MITSUBISHI ELECTRIC

Changes for the Better

MITSUBISHI CNC

Programming Manual (Machining Center System)

C70

IB-1500269(ENG)-E

Introduction

This manual is a guide for using the C70.
Programming 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 as "Restrictions" or "Usable State" in this manual, the instruction manual issued by the machine tool builder takes precedence over this manual.**

⚠ **Items not described in this manual 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.**

Precautions for Safety

Always read the specifications issued by the machine tool builder, 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.

The meanings of the pictorial signs are given below.

The following sings indicate prohibition and compulsory.

	This sign indicates prohibited behavior (must not do). **For example, indicates "Keep fire away".**
	This sign indicates a thing that is compulsory (must do). **For example, indicates "it must be grounded".**

The meaning of each pictorial sing is as follows.

CAUTION	CAUTION rotated object	CAUTION HOT	Danger Electric shock risk	Danger explosive
Prohibited	Disassembly is prohibited	KEEP FIRE AWAY	General instruction	Earth ground

⚠DANGER

Not applicable in this manual.

⚠WARNING

1. Items related to operation

- ⚠ **If the operation start position is set in a block which is in the middle of the program and the program is started, the program before the set block is not executed. Please confirm that G and F modal and coordinate values are appropriate. If there are coordinate system shift commands or M, S, T and B commands before the block set as the start position, carry out the required commands using the MDI, etc. If the program is run from the set block without carrying out these operations, there is a danger of interference with the machine or of machine operation at an unexpected speed, which may result in breakage of tools or machine tool or may cause damage to the operators.**
- ⚠ **Under the constant surface speed control (during G96 modal), if the axis targeted for the constant surface speed control moves toward the spindle center, the spindle rotation speed will increase and may exceed the allowable speed of the workpiece or chuck, etc. In this case, the workpiece, etc. may jump out during machining, which may result in breakage of tools or machine tool or may cause damage to the operators.**

⚠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.**
- ⚠ **Items not described in this manual 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.**

2. Items related to operation

- ⚠ **Before starting actual machining, always carry out graphic check, dry run operation and single block operation to check the machining program, tool offset amount, workpiece offset amount and 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.**
- ⚠ **If the tool offset amount is changed during automatic operation (including during single block stop), it will be validated from the next block or blocks onwards.**
- ⚠ **Do not make the synchronous spindle rotation command OFF with one workpiece chucked by the basic spindle and synchronous spindle during the spindle synchronization.**
Failure to observe this may cause the synchronous spindle stop, and hazardous situation.

3. Items related to programming

- ⚠ **The commands with "no value after G" will be handled as "G00".**
- ⚠ **";" "EOB" and "%" "EOR" are expressions used for explanation. The actual codes are: For ISO: "CR, LF", or "LF" and "%".**
Programs created on the Edit screen are stored in the NC memory in a "CR, LF" format, but 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 the multi-part system, take special care to the movements of the programs for other part systems.**

Disposal

(Note)This symbol mark is for EU countries only.
This symbol mark is according to the directive 2006/66/EC Article 20 Information for end-users and Annex II.

Your MITSUBISHI ELECTRIC product is designed and manufactured with high quality materials and components which can be recycled and/or reused.
This symbol means that batteries and accumulators, at their end-of-life, should be disposed of separately from your household waste.
If a chemical symbol is printed beneath the symbol shown above, this chemical symbol means that the battery or accumulator contains a heavy metal at a certain concentration.
This will be indicated as follows:
Hg: mercury (0,0005%), Cd: cadmium (0,002%), Pb: lead (0,004%)
In the European Union there are separate collection systems for used batteries and accumulators.
Please, dispose of batteries and accumulators correctly at your local community waste collection/recycling centre.

Please, help us to conserve the environment we live in!

CONTENTS

1

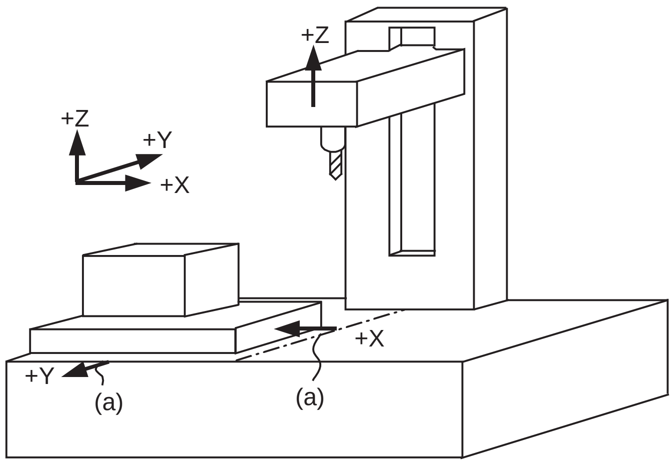
Control Axes

1.1 Coordinate Words and Control Axes

Function and purpose

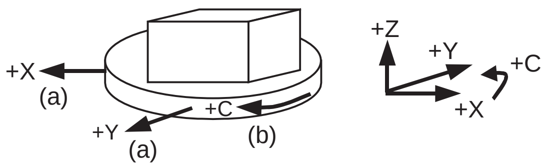
In the standard specifications, there are 3 control axes, however, by adding an additional axis, up to 16 axes can be controlled. The designation of the processing direction responds to those axes and uses a coordinate word made up of alphabet characters that have been decided beforehand.

X-Y table



(a) Direction of table movement

X-Y and rotating table



(a) Direction of table movement (b) Direction of table rotation

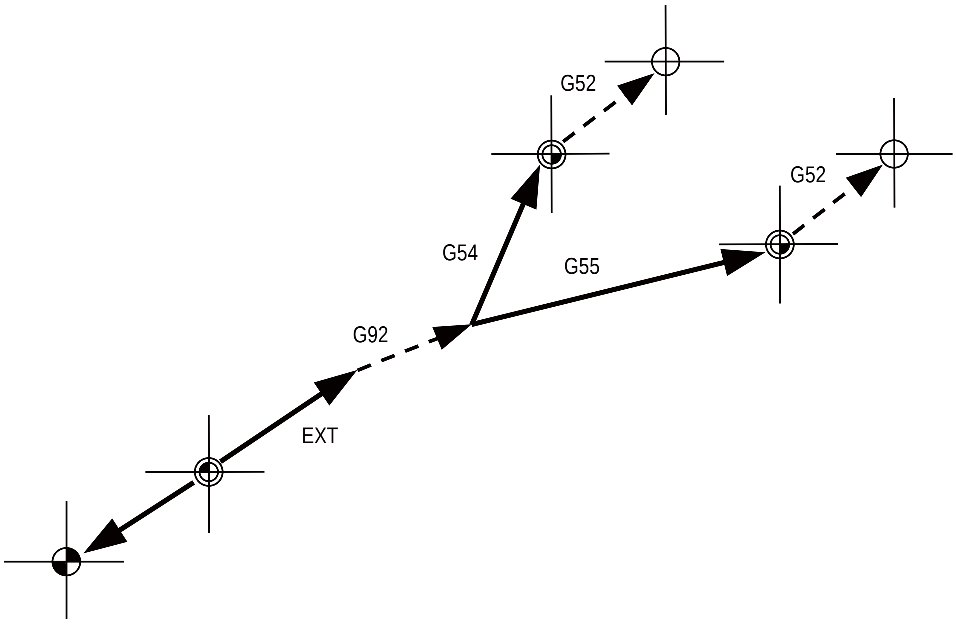
1.2 Coordinate Systems and Coordinate Zero Point Symbols


Reference position:
A specific position to establish coordinate systems and change tools


Basic machine coordinate zero point:
A position specific to machine


Workpiece coordinate zero points (G54 to G59)
A coordinate zero point used for workpiece machining

The basic machine coordinate system is the coordinate system that expresses the position (tool change position, stroke end position, etc.) that is specific to the machine.
Workpiece coordinate systems are used for workpiece machining.
Upon completion of the dog-type reference position return, the parameters are referred and the basic machine coordinate system and workpiece coordinate systems (G54 to G59) are automatically set.
The offset of the basic machine coordinate zero point and reference position is set by a parameter. (Normally, set by machine manufacturers)
Workpiece coordinate systems can be set with coordinate systems setting functions, workpiece coordinate offset measurement (additional specification), and etc.

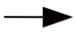


 Reference position

 Basic machine coordinate zero point

 Workpiece coordinate zero points

 Local coordinate zero point

 Offset set by a parameter

 Offset set by a program
("0" is set when turning the power ON)

G52	Local coordinate system offset (*1)
G54	Workpiece coordinate (G54) system offset (*1)
G55	Workpiece coordinate (G55) system offset
G92	G92 Coordinate system shift
EXT	External workpiece coordinate offset

(*1) G52 offset is independently possessed by G 54 to G59 respectively.

The local coordinate systems (G52) are valid on the coordinate systems designated by workpiece coordinate systems 1 to 6.
Using the G92 command, the basic machine coordinate system can be shifted and made into a hypothetical machine coordinate system. At the same time, workpiece coordinate systems 1 to 6 are also shifted.

2

Least Command Increments

2.1 Least Command Increments

Function and purpose

These are the units used for the movement amounts in the program. They are expressed in millimeters, inches or degrees (°).

2.2 Input Setting Unit

Function and purpose

These are the units of setting data which are used, as with the compensation amounts, in common for all axes.
The input command units can be selected from the following types for each axis with the parameters. The input setting units can be selected from the following types common to axes. (For further details on settings, refer to the Instruction Manual.)

Detailed description

	Input unit parameters		Linear axis		Rotary axis (°)
			Millimeter	Inch	
Input command unit	#1015 cunit	= 10	0.001	0.0001	0.001
		= 1	0.0001	0.00001	0.0001
Min. movement unit	#1003 iunit	= B	0.001	0.0001	0.001
		= C	0.0001	0.00001	0.0001
Input setting unit	#1003 iunit	= B	0.001	0.0001	0.001
		= C	0.0001	0.00001	0.0001

Precautions

(1) Inch/metric changeover can be handled by either a parameter screen (#1041 I_inch: valid only when the power is turned ON) or G commands (G20 or G21).
However, the changeover by a G command applies only to the input command units, 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 in order to correspond to input setting units.

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

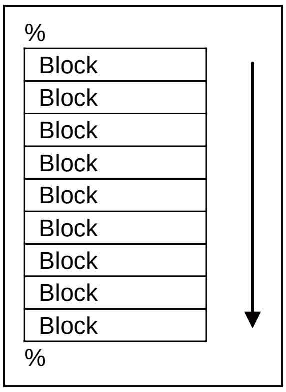
(3) When performing a circular interpolation between the axes whose input command units are different, the center command (I, J, K) and the radius command (R) are designated by the input setting units. (Use a decimal point to avoid confusion.)

3

Program Formats

3.1 Program Format

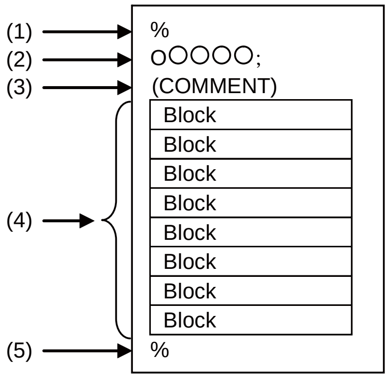
A collection of commands assigned to an NC to move a machine is called "program".
A program is a collection of units called "block" which specifies a sequence of machine tool operations.
Blocks are written in the order of the actual movement of a tool.
A block is a collection of "words" which constitutes a command to an operation.
A word is a collection of characters (alphabets, numerals, signs) arranged in a specific sequence.



Detailed description

Program

A program format looks as follows.



(1) Program start
Input an End Of Record (EOR, %) at the head of a program.
It is automatically added when writing a program on an NC. When using an external device, do not forget to input it at the head of a program.

(2) Program No.
Program Nos. are used to classify programs by main program unit or subprogram unit. They are designated by the address "O" followed by numbers of up to 8 digits. Program Nos. must be written at the head of programs. A setting is available to prohibit O8000s and O9000s from editing (edit lock). Refer to the instruction manual for the edit lock.

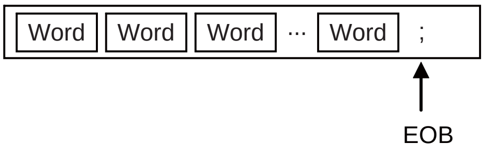
(3) Comment
Data between control out "(" and control in ")" is ignored.
Information including program names and comments can be written in.

(4) Program section
A program is a collection of several blocks.

(5) Program end
Input an end of record (EOR, %) at the end of a program.
It is automatically added when writing a program on an NC.

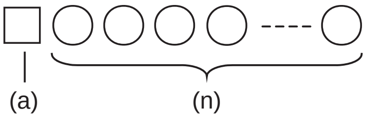
Block and word

[Block]



A block is a least command increment, consisting of words.
It contains the information which is required for a tool machine to execute a specific operation. One block unit constitutes a complete command.
The end of each block is marked with an End of Block (EOB, expressed as ";" for the sake of convenience).

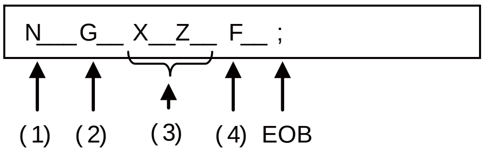
[Word]



(a) Alphabet (address)
(n) Numerals

A word consists of a set of an alphabet, which is called an address, and numerals (numerical information).
Meanings of the numerical information and the number of significant digits of words differ according to an address.

The major contents of a word are described below.



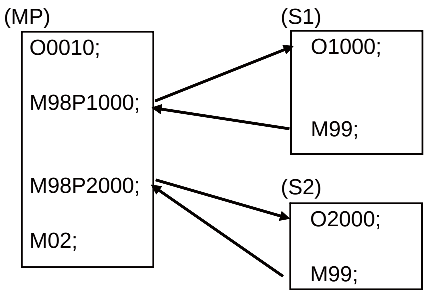
(1) Sequence No.
A "sequence No." consists of the address "N" followed by numbers of up to 5 digits. It is used as an index when searching a necessary block in a program (as branch destination and etc.).
It does not affect the operation of a tool machine.

(2) Preparatory functions (G code, G function)
"Preparatory function (G code, G function)" consists of the address G followed by numbers of 2 or 3 digits (it may include 1 digit after the decimal point). G codes are mainly used to designate functions, such as axis movements and setting of coordinate systems. For example, G00 executes a positioning and G01 executes a linear interpolation.

(3) Coordinate words
"Coordinate words" specify the coordinate position and movement amounts of tool machine axes. They consist of an address which indicates each axis of a tool machine followed by numerical information (+ or - signs and numerals).
X, Y, Z, U, V, W, A, B and C are used as address. Coordinate positions and movement amounts are specified by either "incremental value commands" or "absolute value commands".

(4) Feed Functions (F functions)
"Feed Functions (F functions)" designate the speed of a tool relative to a workpiece. They consist of the address F followed by numbers.

Main program and subprograms



(MP) Main program
(S1) Subprogram 1
(S2) Subprogram 2

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.
If a command is issued to call a subprogram while a main program is being executed, the subprogram will be executed. And when the subprogram is completed, the main program will be resumed.
Refer to the description of subprogram control for the details of the execution of subprograms.

Precautions

(1) Since the semicolon in the parentheses will not result in an EOB, it is 1 block
(Example 1) 2 blocks
G0 X-1000;
G1 X-2000 F500;
(Example 2) 1 blocks
(G0 X-1000;)
G1 X-2000 F500;

(2) When there is no number following the alphabetic character in the actual program, the numeric value following the alphabetic character is handled as a 0.
(Example) G28XYZ; -> G28X0Y0Z0;

3.2 Program/sequence/block numbers; 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 "0" 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 5 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.
O12345678 (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
%			

3.3 Optional Block Skip

3.3.1 Optional Block Skip; /

Function and purpose

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

Detailed description

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 blocks 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

(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 No. 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.

3.3.2 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 n signal is turned ON. For a block with the "/n" code inside the block (not at the head of the block), the program is operated according to the value of the parameter "#1226 aux10/bit1" setting.
When the optional block skip n signal is OFF, the block with "/n" is executed.

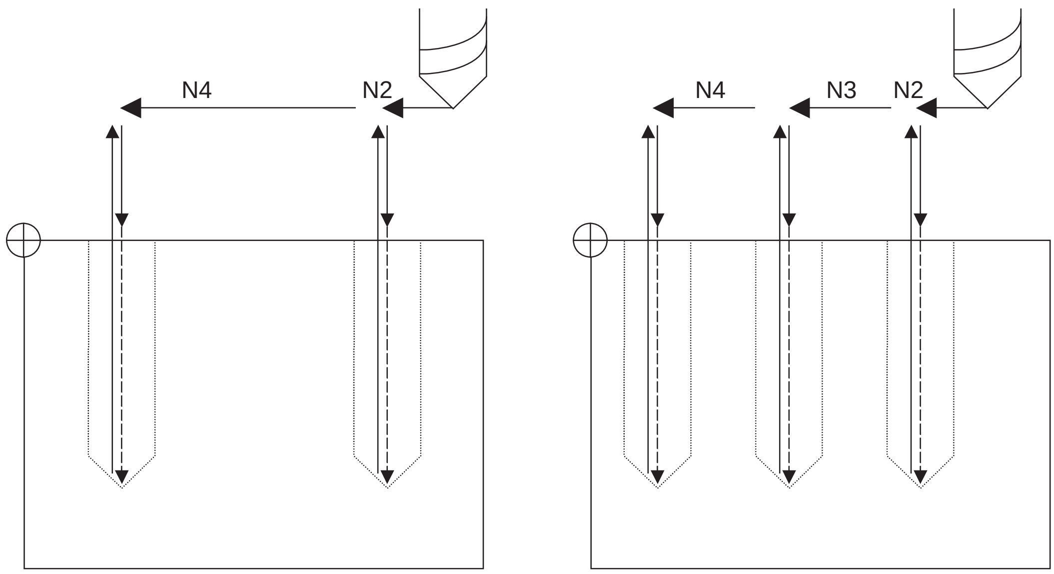
Program example

(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.

```
      N1 G54 ;
      N2 G90 G81 X50. Z-20. R3. F100 ;
/5    N3 X30. ;
      N4 X10. ;
      N5 G80 ;
      M02 ;
```

Part 1
Optional block skip 5 signal ON

Part 2
Optional block skip 5 signal OFF



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

```
         N01 G90 Z3. M03 S1000 ;
/1/2     N02 G00 X50. ;
/1/2     N03 G01 Z-20. F100 ;
/1/2     N04 G00 Z3. ;
/1  /3   N05 G00 X30. ;
/1  /3   N06 G01 Z-20. F100 ;
/1  /3   N07 G00 Z3. ;
  /2/3   N08 G00 X10. ;
  /2/3   N09 G01 Z-20. F100 ;
  /2/3   N10 G00 Z3. ;
         N11 G28 X0 M05 ;
         N12 M02 ;
```

(a) Optional block skip 1 signal ON
(Optional block skip 2.3 signal OFF)
N01 -> N08 -> N09 -> N10 -> N11 -> N12

(b) Optional block skip 2 signal ON
(Optional block skip 1.3 signal OFF)
N01 -> N05 -> N06 -> N07 -> N11 -> N12

(c) Optional block skip 3 signal ON
(Optional block skip 1.2 signal OFF)
N01 -> N02 -> N03 -> N04 -> N11 -> N12

(3) When the parameter "#1226 aux10/bit1" is "1"and 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 n signals corresponding to the command is ON.

```
N01 G91 G28 X0.Y0.Z0.;
N02 G01 F1000;
N03 X1. /1 Y1. /2 Z1.;
N04 M30;
```

N03 block will operate as follows.
(a) Optional block skip 1 signal ON
Optional block skip 2 signal OFF
"Y1. Z1." is ignored.
(b) Optional block skip 1 signal OFF
Optional block skip 2 signal ON
"Z1." is ignored.

3.4 G code list

3.4.1 Modal, unmodal

G codes define the operation modes of each block in programs.
G codes can be modal or unmodal command.
Modal commands always designate one of the G codes in the group as the NC operation mode. The operation mode is maintained until a cancel command is issued or other G code among the same group is commanded.
An unmodal command designates the NC operation mode only when it is issued. It is invalid for the next block.

3.4.2 Table of G Code Lists

G code	Group	Function	Section
Δ 00	01	Positioning	6.1
* 01	01	Linear interpolation	6.2
02	01	Circular interpolation CW (clockwise)	6.3 6.4 6.7 6.13
03	01	Circular interpolation CCW (counterclockwise)	6.3 6.4 6.7 6.13
04	00	Dwell	8.1 15.3
06			
07			
08			
09	00	Exact stop check	7.9
10	00	Programmable compensation input/ Programmable parameter input Programmable current limitation	12.4 13.9 15.5
11	00	Programmable compensation input cancel	12.4 13.11
12	00	Circular cutting CW (clockwise)	13.8
13	00	Circular cutting CCW (counterclockwise)	13.8
14			
15			
16			
* 17	02	Plane selection X-Y	6.5
Δ 18	02	Plane selection Z-X	6.5
Δ 19	02	Plane selection Y-Z	6.5
Δ 20	06	Inch command	5.2
* 21	06	Metric command	5.2
22			
23			
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	13.11
30.2	00	Tool change position return 2	13.11
30.3	00	Tool change position return 3	13.11

G code	Group	Function	Section
30.4	00	Tool change position return 4	13.11
30.5	00	Tool change position return 5	13.11
30.6	00	Tool change position return 6	13.11
31	00	Skip/Multi-step skip function 2	15.2 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	6.6
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	Automatic tool length measurement	15.1
37.1	00	Special fixed cycle (grid)	13.2
38	00	Tool radius compensation vector designation	12.3
39	00	Tool radius compensation corner arc	12.3
* 40	07	Tool radius compensation cancel	12.3
41	07	Tool radius compensation left	12.3
42	07	Tool radius compensation right	12.3
43	08	Tool length offset (+)	12.2
44	08	Tool length offset (-)	12.2
* 49	08	Tool length offset cancel	12.2
* 50.1	19	G command mirror image OFF	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	14.5
* 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 sets extended	14.10
60	00	Unidirectional positioning	6.8
61	13	Exact stop check mode	7.10
61.1	13	High-accuracy control mode	13.12
62	13	Automatic corner override	7.12
63	13	Tapping mode	7.13
* 64	13	Cutting mode	7.14
65	00	User macro call	13.5.2.1
66	14	User macro modal call A	13.5.2.2
66.1	14	User macro modal call B	13.5.2.3
* 67	14	User macro modal call cancel	13.5.1
70		User fixed cycle	
71		User fixed cycle	
72		User fixed cycle	
73	09	Fixed cycle (stepping)	13.1.10
74	09	Fixed cycle (reverse tapping)	13.1.11
75		User fixed cycle	
76	09	Fixed cycle (fine boring)	13.1.12
77		User fixed cycle	
78		User fixed cycle	
79		User fixed cycle	

G code	Group	Function	Section
* 80	09	Fixed cycle cancel	13.1
81	09	Fixed cycle (drilling/spot drilling)	13.1.1
81.1	09	Chopping	13.17
82	00	Fixed cycle (drilling/counter boring)	13.1.2
83	09	Fixed cycle (deep drilling)	13.1.3
84	09	Fixed cycle (tapping)	13.1.4
85	09	Fixed cycle (boring)	13.1.5
86	09	Fixed cycle (boring)	13.1.6
87	09	Fixed cycle (back boring)	13.1.7
88	09	Fixed cycle (boring)	13.1.8
89	09	Fixed cycle (boring)	13.1.9
Δ 90	03	Absolute command	5.1
* 91	03	Incremental command	5.1
92	00	Coordinate system setting/Spindle clamp speed setting;	14.6 10.3
93			
* 94	05	Feed per minute (Asynchronous feed)	7.4
Δ 95	05	Feed per revolution (Synchronous feed)	7.4
Δ 96	17	Constant surface speed control ON	10.2
* 97	17	Constant surface speed control OFF	10.2
* 98	10	Fixed cycle initial level return	13.1.14
99	10	Fixed cycle R point level return	13.1.14
113	00	Spindle synchronization control cancel	10.5.1
114.1	00	Spindle synchronization control ON	10.5.1
115	00	Start point designation synchronization (type1)	13.15
116	00	Start point designation synchronization (type2)	13.16

Precautions

(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.

(2) If two or more G codes from the same code are commanded, the latter G code will be valid.

(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.

(4) Whether the modal is initialized or not depends on each reset input.
(a) "Reset 1"
The modal is initialized when the reset initial parameter "#1151 rstinit" turns ON.
(b) "Reset 2" and "Reset & rewind"
The modal is initialized when the signal is input.
(c) Resetting when emergency stop is canceled
Follows "Reset 1".
(d) When modal is automatically reset at the start of individual functions such as reference position return.
Follows "Reset & rewind".

⚠ CAUTION **1. The commands with "no value after G" will be handled as "G00".**

3.5 Precautions Before Starting Machining

⚠ CAUTION	1. 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. 2. Before starting actual machining, always carry out graphic check, dry run operation and single block operation to check the machining program, tool offset amount, workpiece offset amount and etc.

4

Pre-read Buffers

4.1 Pre-read Buffers

Function and purpose

During automatic processing, the contents of one block ahead 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.

Detailed description

The specifications of pre-read buffers in 1 block are as follows:

(1) The data of 1 block is stored in this buffer.
(2) When comments 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.
(3) The pre-read buffer contents are cleared with resetting.
(4) When the single block function is ON during continuous operation, the pre-read buffer stores the next block's data and then stops operation.
(5) The way to prohibit the M command which operates the external controls from pre-reading, and to make it to recalculate, is as follows:
Identify the M command which operates the external controls by a PLC, and turn on the "recalculation request" on PLC output signal. (When the "recalculation request" is turned ON, the program that has been pre-read is recalculated.)

Precautions

(1) Depending on whether the program is executed continuously or by single blocks, the timing of the validation/invalidation of the external control signals including optional block skip, 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 for the program pre-read with the buffer register.

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/G91 X__ Y__ Z__ α__ ;

G90	Absolute command
G91	Incremental command
X,Y,Z,α	Coordinate values (α is the additional axis.)

Detailed description

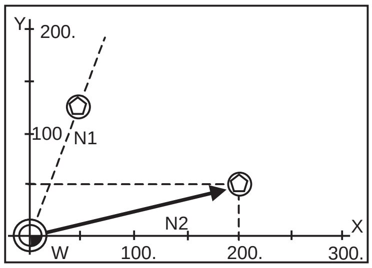
(1) Regardless of the current position, in the absolute value mode, it is possible to move to the position of the workpiece coordinate system that was designated in the program.

N1 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.

N2 G90 G01 X200. Y50. F100 ;
N2 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.



Tool

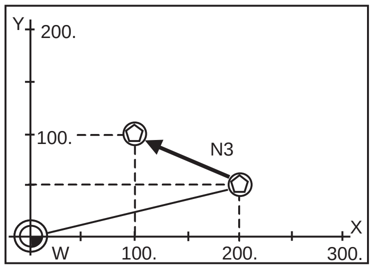
(2) For the next block, the last G90/G91 command that was given becomes the modal.

(G90) N3 X100. Y100. ;

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

(G91) N3 X-100. Y50. ;

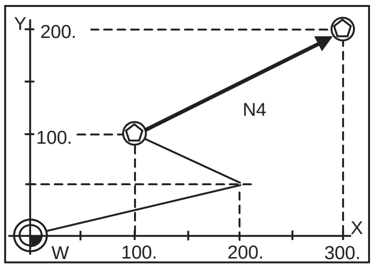
The X axis moves to -100.mm and the Y axis to +50.0mm as an incremental value, and as a result X moves to 100.mm and Y to 100.mm.



(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.

N4 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 Conversion ; G20,G21

Function and purpose

The commands can be changed between inch and metric with the G20/G21 command.

Command format

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. It is invalid for rotation axes.
If the machining program command unit changes to an inch unit at G20 when the initial inch is OFF, the setting unit of the tool offset amount will remain metric. Thus, take note to the setting value.

(Example 1) With decimal point input type 1

Axis	Input command unit type (cunit)	Command example	Metric output (#1016 iout=0)		Inch output (#1016 iout=1)	
			G21	G20	G21	G20
X Y Z	10 10 10	X100; Y100; Z100;	0.100 mm 0.100 mm 0.100 mm	0.254 mm 0.254 mm 0.254 mm	0.0039 inch 0.0039 inch 0.0039 inch	0.0100 inch 0.0100 inch 0.0100 inch
X Y Z	1 1 1	X100; Y100; Z100;	0.0100mm 0.0100mm 0.0100mm	0.0254 mm 0.0254 mm 0.0254 mm	0.00039inch 0.00039inch 0.00039inch	0.00100inch 0.00100inch 0.00100inch

(Note 1) When changing between G20 and G21 with program commands, it is necessary in advance, to convert the parameters, variables, and the offsets for the tool diameter, tool position, tool length, to the units in the input settings of the input setting unit system (for each axis) that have inch or metric commands, and make the settings using the parameter tape.

(Example 2) Input setting unit #1015 cunit=10, #1041 I_inch=0

Position command unit 0.001mm
Compensation amount setting unit ..... When the compensation amount is 0.05mm for 0.001mm

In the above example, when changing from G21 to G20, the compensation amount must be set to 0.002 (0.05/ 25.4 ≒ 0.002).

(Note 2) Since the data before the change will be executed at the command unit after the change, command the F speed command for the change so that it is the correct speed command for the command unit system applied after the change.

5.3 Decimal Point Input

Function and purpose

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

Detailed description

(1) The decimal point command is valid for the distances, angles, times, speeds and scaling rate, in machining programs. (Note, only after G51)

(2) In decimal point input type I and type II, the values of the data commands without the decimal points are shown in the table below.

Command	Command unit	Type I	Type II
X1;	cunit=10	1 (µm, 10^{-4} inch, 10^{-3} °)	1 (mm, inch, °)
	cunit= 1	0.1	1

(3) 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.

(4) See below for the number of significant digits in decimal point commands.
(Input command unit cunit = 10)

	Movement command (linear)		Movement command (rotary)		Feedrate		Dwell	
	Integer	Decimal part	Integer	Decimal part	Integer	Decimal part	Integer	Decimal part
MM (millimeter)	0. to 99999.	.000 to .999	0. to 99999.	.000 to .999	0. to 60000.	.00 to .99	0. to 99999.	.000 to .999
INCH (inch)	0. to 9999.	.0000 to .9999	99999. (359.)	.0 to .999	0. to 2362.	.000 to .999	.0 to .99	.000 to .999

(5) The decimal point command is valid even for commands defining the variable data used in subprograms.

(6) 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 μ m, 10 μ m, etc.) or mm can be selected. This selection can be made with parameter "#1078 Decpt2".

(7) 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.

Decimal point input I, II and decimal point command validity

Decimal point input I and II will result as follows when decimal points are not used in an address which a decimal point command is valid. Whether an address is valid or invalid for the decimal point command is shown in the table below.
Both decimal point input I and II will produce the same result when a command uses a decimal point.

(1) Decimal point input I
The least significant digit of command data matches the command unit.
(Example) When "X1" is commanded in 1 μm system, the same result occurs as for an "X0.001" command.

(2) Decimal point input II
The least significant digit of command data matches the command unit.
(Example) When "X1" is commanded in 1 μm system, the same result occurs as for an "X1." command.

-Addresses used, validity of decimal point commands-

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)	
	Valid	Spindle synchronization acceleration/deceleration time constant	
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	Offset numbers (tool position, tool radius)	
	Valid	Automatic tool length measurement: deceleration distance d	
	Invalid	Data setting: byte type data	
	Valid	Parameter input by program; numerical value parameter	
	Invalid	Synchronous spindle selection	
E	Valid	Inch thread: number of ridges, precision thread: lead	
F	Valid	Feedrate	
	Valid	Thread lead	
	Valid	Number of Z axis pitch in synchronous tap	
G	Valid	Preparatory function code	
H	Invalid	Tool length offset number	
	Invalid	Sequence numbers in subprograms	
	Invalid	Programmable parameter input: bit type data	
	Invalid	Basic spindle selection	
I	Valid	Coordinates for arc center	
	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
J	Valid	Coordinates for arc center	
	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	,J

Address	Decimal Point Command	Application	Remarks
K	Valid	Coordinates for arc center	
	Valid	Tool radius compensation vector components	
	Invalid	Number of holes of the special fixed cycle	
L	Invalid	Number of fixed cycle and subprogram repetitions	
	Invalid	Program tool compensation input type selection	L2, L10, L11 L12, L13
	Invalid	Parameter input by program selection	L50
	Invalid	Programmable parameter input: 2-word type data	4 bytes
	Invalid	Timing-synchronization between part systems	
M	Invalid	Miscellaneous function codes	
N	Invalid	Sequence numbers	
	Invalid	Programmable parameter input: data numbers	
O	Invalid	Program numbers	
P	Valid	Dwell time	Parameter
	Invalid	Subprogram program call: program No.	
	Invalid	Dwell at tap cycle hole base	
	Invalid	Number of holes of the special fixed cycle	
	Invalid	Amount of helical pitch	
	Invalid	Offset number (G10)	
	Invalid	Constant surface speed control axis number	
	Invalid	Programmable parameter input: parameter No.	
	Invalid	Multi-step skip function 2 signal command	
	Invalid	Subprogram return destination sequence No.	
	Invalid	2nd, 3rd, 4th reference position return number	
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	
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	
	Valid	Spindle synchronization phase shift amount	
S	Invalid	Spindle function codes	
	Invalid	Maximum spindle clamp speed	
	Invalid	Constant surface speed control: surface speed	
	Invalid	Programmable parameter input: part system No.	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) Decimal points are all valid in user macro arguments.

Program example

(1) Program example of decimal point valid address

Program example	Decimal point command 1		Decimal point command 2
	When 1 = 1μm	When 1 = 10μm	When 1 = 1mm
G0 X123.45 (decimal points are all mm points)	X123.450 mm	X123.450 mm	X123.450 mm
G0 X12345	X12.345 mm (last digit is 1μm unit)	X123.450 mm	X12345.000 mm
#111=123 #112=5.55 X#111 Y#112	X123.000 mm Y5.550 mm	X123.000 mm Y5.550 mm	X123.000 mm Y5.550 mm
#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

Precautions

(1) If an arithmetic operator is inserted, the data will be handled as data with a decimal point.
(Example1) G00 X123+0 ;
This is the X axis 123mm command. It will not be 123 μ m.

6

Interpolation Functions

6.1 Positioning (Rapid Traverse) ; G00

Function and purpose

This command is accompanied by coordinate words and performs high-speed positioning of a tool, from the present point (start point) to the end point specified by the coordinate words.

Command format

G00 X__ Y__ Z__α__,I__ ; ... Positioning (Rapid Traverse)

X, Y, Z, α	Coordinate values (α is the additional axis.) An absolute position or incremental position is indicated based on the state of G90/G91 at that time.
,I	In-position width. This is valid only in the commanded block. A decimal point command will result in a program error. A block that does not contain this address will follow the parameter "#1193 inpos"settings. 1 to 999999 (μm).

The command addresses are valid for all additional axes.

Detailed description

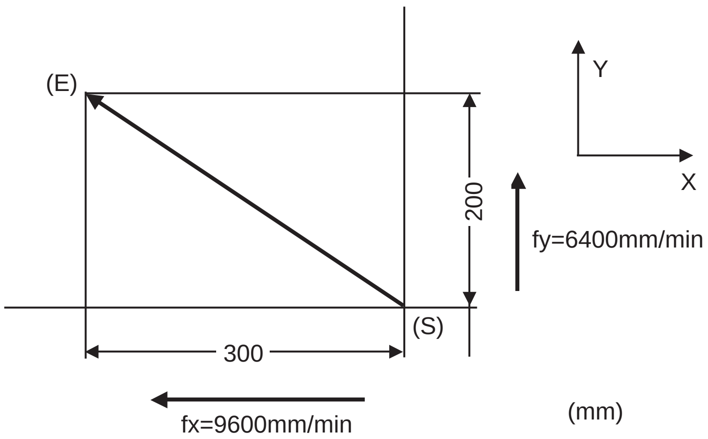
(1) Positioning will be performed at the rapid traverse rate set in the parameter "#2001 rapid".
(2) G00 command belongs to the 01 group and is modal. When G00 command is successively issued, the following blocks can be specified only by the coordinate words.
(3) In the G00 mode, acceleration and deceleration are always carried out at the start point and end point of the block. Before advancing to the next block, a commanded deceleration or an in-position check is conducted at the end point to confirm that the movement is completed for all the moving axes in each part system.
(4) G functions (G72 to G89) in the 09 group are cancelled (G80) by the G00 command.

⚠ CAUTION **1. The commands with "no value after G" will be handled as "G00".**

Tool path

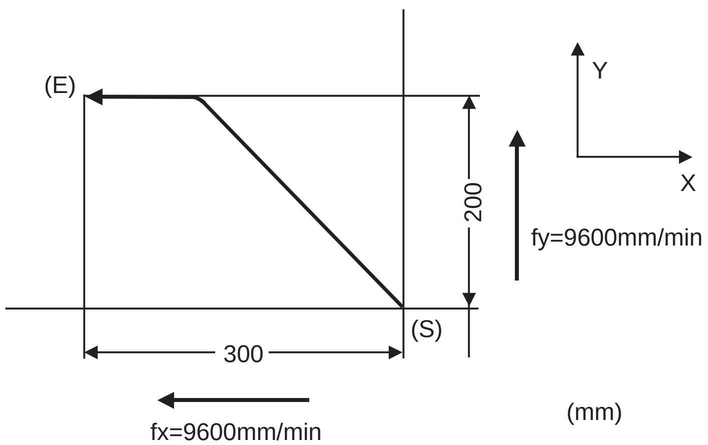
Whether the tool moves along a linear or non-linear path can be selected by the parameter "#1086 G0Intp". The positioning time does not change according to the path.

(1) Linear path: When the parameter "#1086 G0Intp" is set to "0"
In positioning, a tool follows the shortest path which connects the start point and the end point. 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 X-axis and Y-axis rapid traverse rates are both 9600mm/min;
G91 G00 X-300000 Y200000 ; (With an input setting unit of 0.001mm)
The tool will follow the path shown in the figure below.



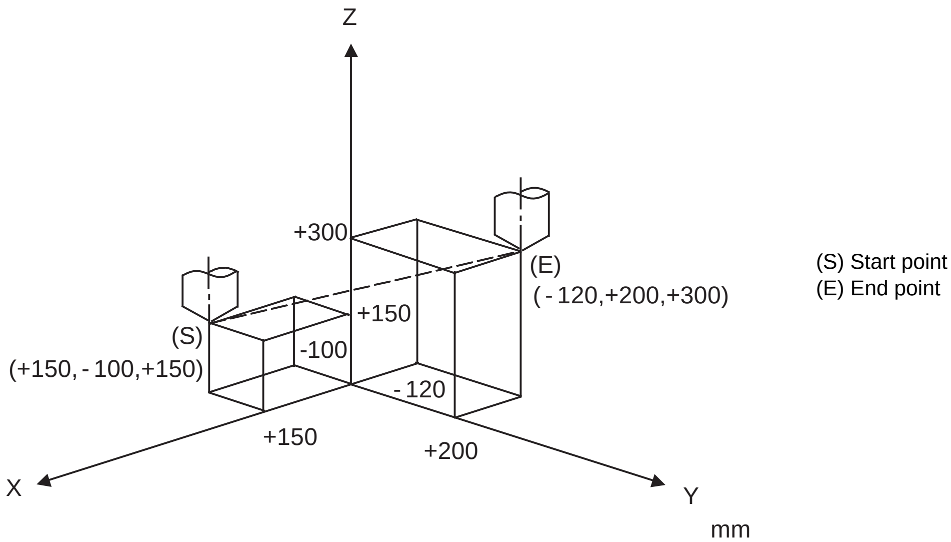
(S) Start point
(E) End point
(fx) Actual X axis rate
(fy) Actual Y axis rate

(2) Non-linear path: When the parameter "#1086 G0Intp" is set to "1"
In positioning, 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 X-axis and Y-axis rapid traverse rates are both 9600mm/min;
G91 G00 X-300000 Y200000 ; (With an input setting unit of 0.001mm)
The tool will follow the path shown in the figure below.



(S) Start point
(E) End point
(fx) Actual X axis rate
(fy) Actual Y axis rate

Program example




```
G91 G00 X-270. Y300. Z150. ;
```

Precautions for deceleration check

There are two methods for the deceleration check; commanded deceleration method and in-position check method. Select a method with the parameter "#1193 inpos".
A block with an in-position width command performs an in-position check with a temporarily changed in-position width. (Programmable in-position width command)
The deceleration check method set in basic specification parameter "#1193 inpos" is used for blocks that do not have the in-position width command.
During cutting feed and when the error detection is ON, the in-position check is forcibly carried out.

Rapid traverse (G00)		**#1193 inpos**	
		0	**1**
,I command	No	Commanded deceleration method (Commanded deceleration check which varies according to the type of acceleration/ deceleration, set in "#2003 smgst" bit3-0)	In-position check method (In-position check by "#2077 G0inps", "#2224 SV024")
	Yes	In-position check method (In-position check by ",I", "#2077 G0inps", "#2224 SV024")	

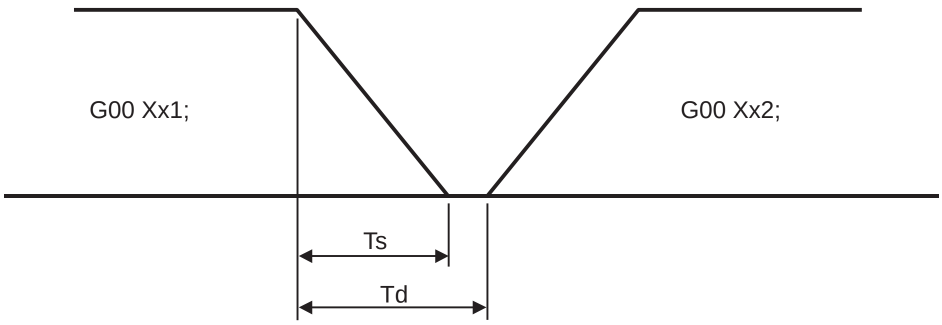
Cutting feedrate (G01)		**#1193 inpos**	
		0	**1**
,I command	No	Commanded deceleration method (Commanded deceleration check which varies according to the type of acceleration/ deceleration, set in "#2003 smgst" bit7-4)	In-position check method (In-position check by "#2078 G1inps", "#2224 SV024")
	Yes	In-position check method (In-position check by ",I", "#2078 G1inps", "#2224 SV024")	

* Following descriptions are for the case of rapid traverse. For G01, interpret the parameters into suitable ones.

Commanded deceleration method when "inpos" = "0"

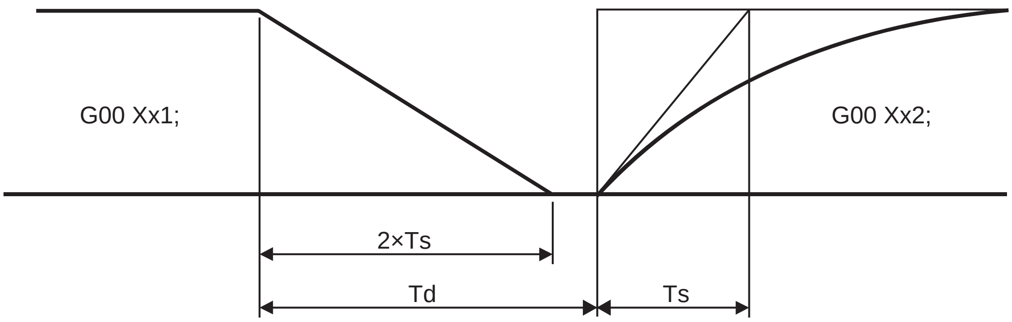
Upon completion of the rapid traverse (G00), the next block will be executed after the deceleration check time (Td) has elapsed.
The deceleration check time (Td) is as follows, depending on the acceleration/deceleration type set in the parameter "#2003 smgst".

(1) Linear acceleration/linear deceleration



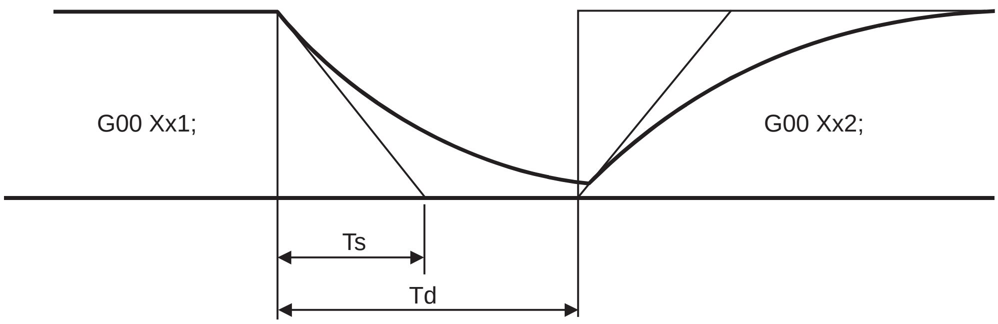
(Ts) Acceleration/deceleration time constant (Td) Deceleration check time: Td = Ts + (0 to 7ms)

(2) Exponential acceleration/linear deceleration



(Ts) Acceleration/deceleration time constant (Td) Deceleration check time: Td = 2 × Ts + (0 to 7ms)

(3) Exponential acceleration/exponential deceleration (Primary delay)

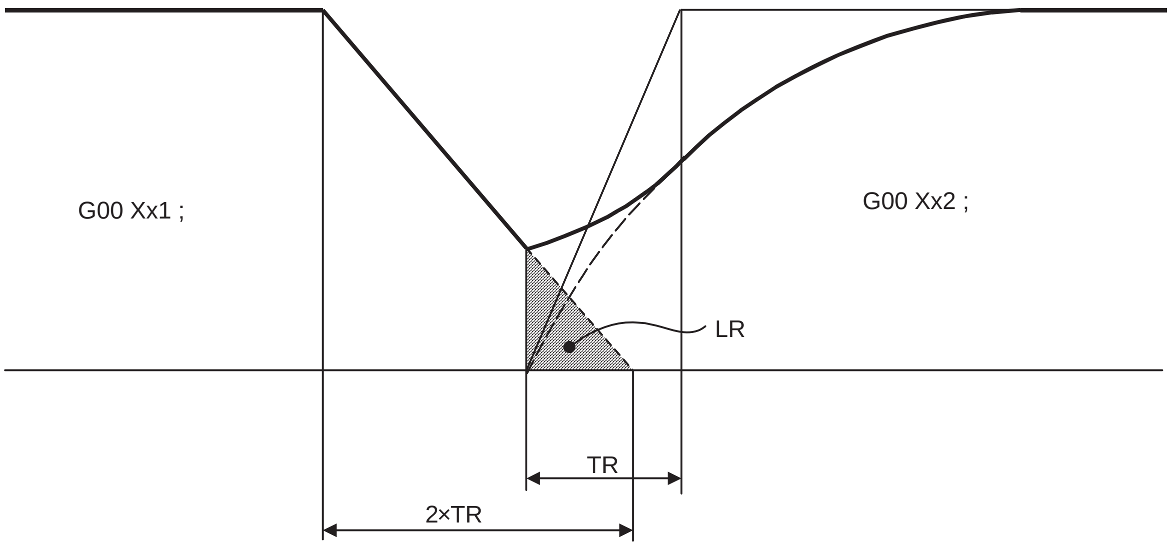


(Ts) Acceleration/deceleration time constant (Td) Deceleration check time: Td = 2 × Ts + (0 to 7ms)

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

In-position check method 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.
The confirmation of the remaining distance should be done with the imposition width.
The bigger one of the servo parameter "#2224 SV024" or G0 in-position width "#2077 G0inps" (For G01, in-position width "#2078 G1inps"), will be adapted as the in-position width.

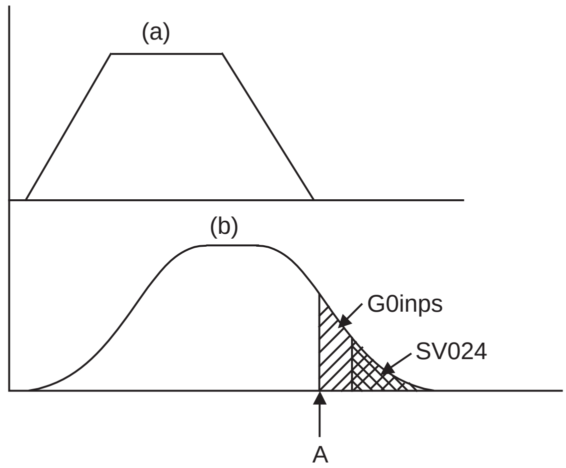


(TR) Rapid traverse acceleration/deceleration time constant
(LR) In-position width

The in-position width LR indicates the remaining distance from the previous block at the start of the next block (shaded area of the figure above).

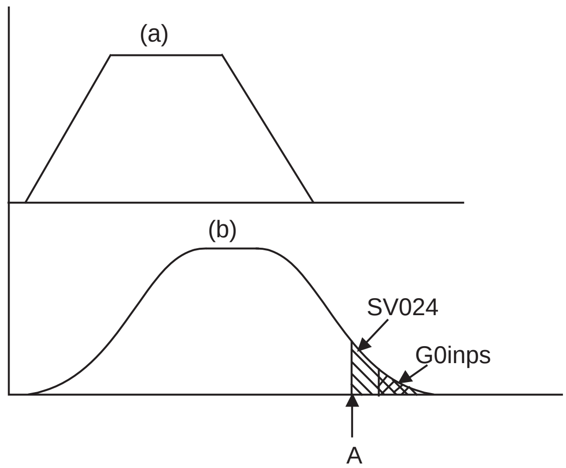
The purpose of the rapid traverse deceleration check is to minimize the positioning time. The bigger the setting value for the in-position width, the shorter the time is, but the remaining distance of the previous block at the start of the next block also becomes larger, and this could become an obstacle 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 effect of time reduction for positioning as in-position width setting value.

(1) In-position check by the G0inps: When SV024 < G0inps (Stop is judged at A in the figure)



(a) Command to motor
(b) Outline of motor movement

(2) In-position check using SV024: When G0inps < SV024 (Stop is judged at A in the figure)



(a) Command to motor
(b) Outline of motor movement

Programmable in-position width command

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

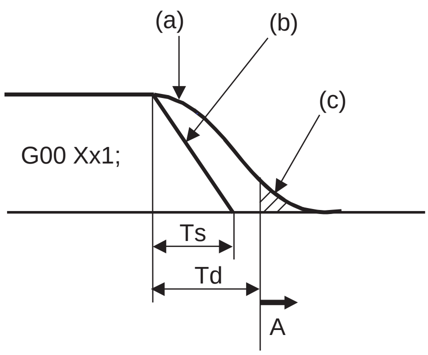
G00 X_ Y_ Z_ ,I_ ;	
X,Y,Z	Positioning coordinate value of each axis
,I	In-position width

Execution of the next block starts after confirming that the position error amount of the positioning (rapid traverse: G00) command block is less than the in-position width issued in this command.
The bigger one of in-position width (SV024, G0inps (For G01, G1inps)) with parameter or in-position width specified by program will be adapted as the in-position width.
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 In-position check

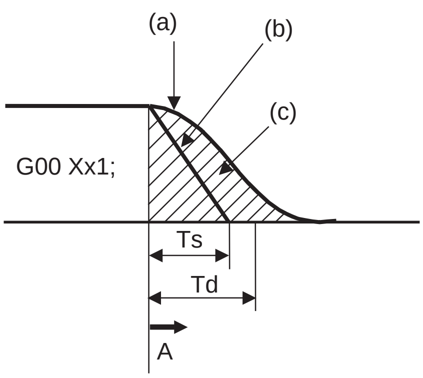
The differences between the in-position check with parameter and with programmable command are as follows:

(1) In-position check with parameter
After completing deceleration of the command system (A), the servo system's position error amount and the parameter setting value (in-position width) are compared.



(a) Servo machine position
(b) Command
(c) In-position width (Servo system position error amount)
(Ts) Acceleration/deceleration time constant
(Td) Deceleration check time: Td = Ts + (0 to 7ms)

(2) In-position check with programmable command (",I" address command)
After starting deceleration of the command system (A), the position error amount and commanded in-position width are compared.



(a) Servo machine position
(b) Command
(c) In-position width (Error amount between command end point and machine position)
(Ts) Acceleration/deceleration time constant
(Td) Deceleration check time: Td = Ts + (0 to 7ms)

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 current 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__ α__ F__ ,I__ ; ... Linear interpolation

X,Y,Z,α	Coordinate values (α is the additional axis.) 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. This is valid only in the commanded block. A decimal command will result in a program error. A block that does not contain this address will follow the parameter "#1193 inpos" settings. 1 to 999999 (μm)

Detailed description

(1) G01 command is a modal command in the 01 group. When G01 command is issued in succession, it can only be issued with coordinate words in subsequent blocks.

(2) The feedrate for a rotary axis is commanded by °/min (decimal point position unit). (F300=300°/min)

(3) The G functions (G72 to G89) in the 09 group are cancelled (G80) by the G01 command.

Programmable in-position width command for linear interpolation

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

G01 X_ Y_ Z_ F_ ,I_ ;	
X,Y,Z	Linear interpolation coordinate value of each axis
F	Feedrate
,I	In-position width

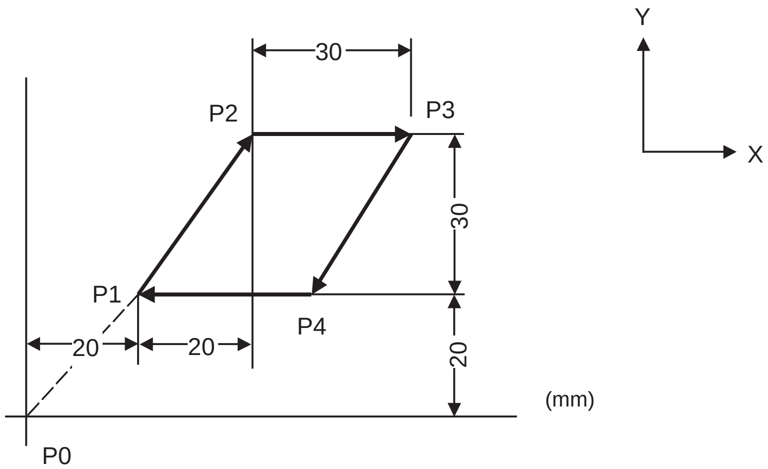
The commanded in-position width is valid in the linear interpolation command only when carrying out deceleration check.

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

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

Program example

(Example) Cutting in the sequence of P1 -> P2 -> P3 -> P4 -> P1 at 300mm/min feedrate.
However, P0 -> P1 is for tool positioning.



G91 G00 X20. Y20. ;	P0 -> P1
G01 X20. Y30. F300 ;	P1 -> P2
X30. ;	P2 -> P3
X-20. Y-30. ;	P3 -> P4
X-30. ;	P4 -> P1

6.3 Circular Interpolation ; G02,G03

Function and purpose

These commands serve to move the tool along a circular.

Command format

G02 X__ Y__ I__ J__ F__ ; ... Circular interpolation : Clockwise (CW)

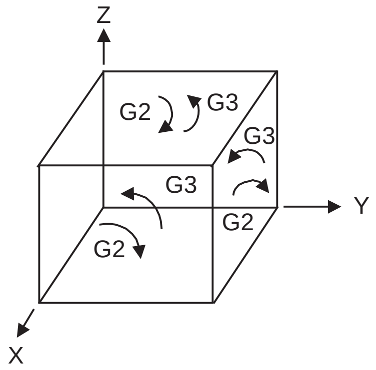
G03 X__ Y__ I__ J__ F__ ; ... Circular interpolation : Counterclockwise (CCW)

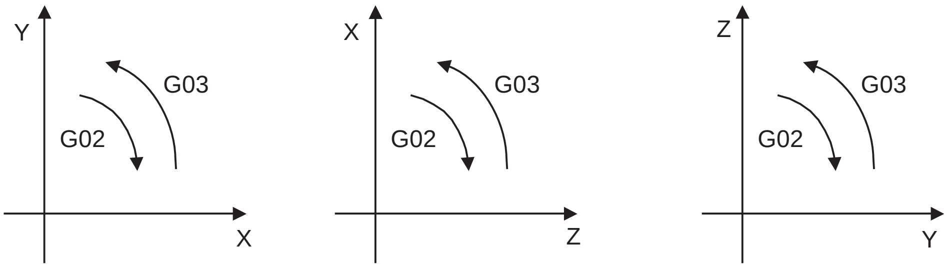
X,Y	End point
I,J	Arc center
F	Feedrate

Detailed description

(1) 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 a program command unit. Caution is required for the circular command of an axis for which the program command unit (#1015 cunit) differs. Command with a decimal point to avoid confusion.

(2) G02 (G03) is a modal command of the 01 group. When G02 (G03) command is issued continuously, the next block and after can be commanded with only coordinate words.
The circular rotation direction is distinguished by G02 and G03.
G02 Clockwise (CW)
G03 Counterclockwise (CCW)





G17(X-Y) plane G18(Z-X) plane G19(Y-Z) plane

(3) An arc which extends for more than one quadrant can be executed with a single block command.

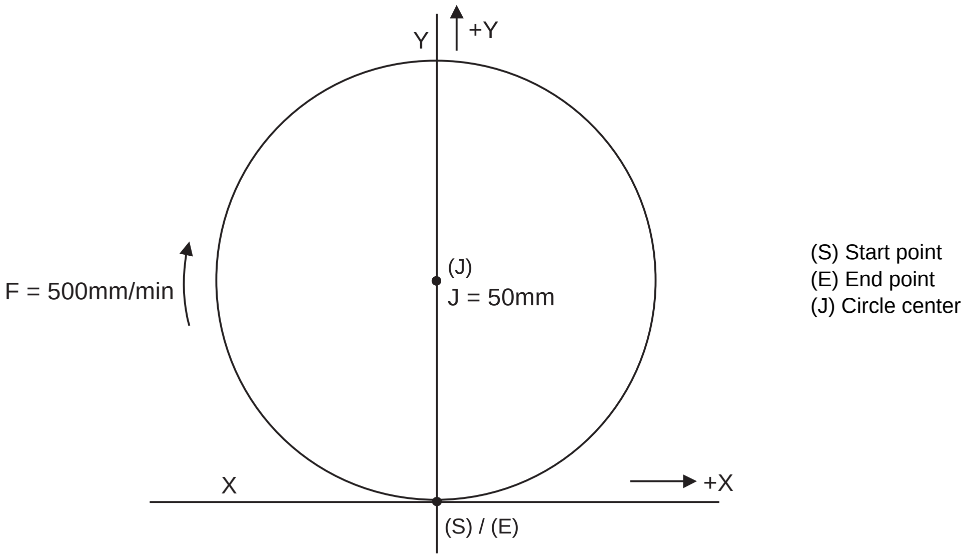
(4) The following information is needed for circular interpolation.
(a) Plane selection : Is there an arc parallel to one of the XY, ZX or YZ planes?
(b) Rotation direction : Clockwise (G02) or counterclockwise (G03)
(c) Circular end point coordinates : Given by addresses X, Y, Z
(d) Circular center coordinates : Given by addresses I, J, K (incremental value commands)
(e) Feedrate : Given by address F

Plane selection

The arc exists in the following three planes (refer to the figure in the “Detailed description”), and are selected by 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)

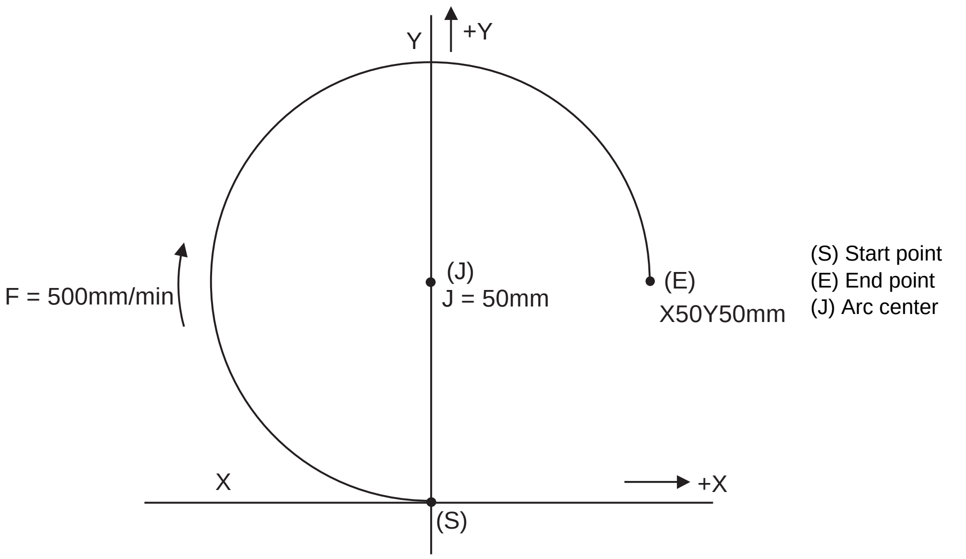
Program example

(Example 1)



G02 J50. F500;	Circle command

(Example 2)



G91 G02 X50.Y50. J50. F500;	3/4 command

Precautions

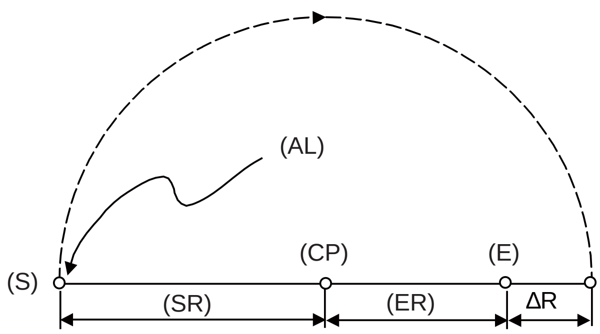
(1) The terms "clockwise" (G02) and "counterclockwise" (G03) used for circular operations are defined as a case where, in a right-hand coordinate system, the negative direction is viewed from the positive direction of the coordinate axis which is at right angles to the plane in question.

(2) If all the end point coordinates are omitted or the end point is at the same position as the start point, commanding the center using I, J and K is the same as commanding a 360°arc (perfect circle).

(3) The following occurs when the start and end point radius do not match in a circular command :

(a) Program error (P70) occurs at the circular start point when error ΔR is greater than parameter "#1084 RadErr".

(G91) G02 X9.899 I5. ;

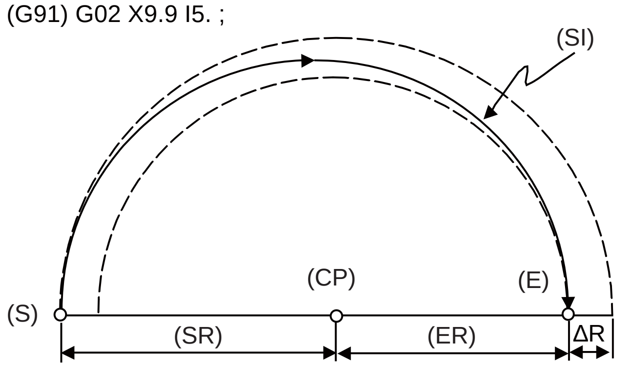


#1084 RadErr Parameter value 0.100
Start point radius=5.000
End point radius=4.899
ErrorΔR=0.101

(S) Start point
(CP) Center
(E) End point
(SR) Start point radius
(ER) End point radius
(AL) Alarm stop

(b) Spiral interpolation in the direction of the commanded end point will be conducted when error ΔR is less than the parameter value.

(G91) G02 X9.9 I5. ;



#1084 RadErr Parameter value 0.100
Start point radius=5.000
End point radius=4.900
ErrorΔR=0.100

(S) Start point
(CP) Center
(E) End point
(SR) Start point radius
(ER) End point radius
(SI) Spiral interpolation

6.4 R Specification Circular Interpolation ; G02,G03

Function and purpose

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

Command format

G02 X__ Y__ R__ F__ ; ... R specification circular interpolation Clockwise (CW)

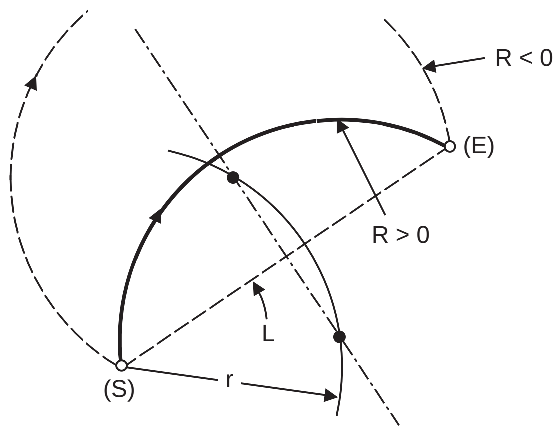
G03 X__ Y__ R__ F__ ; ... R specification circular interpolation Counterclockwise (CCW)

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

The arc radius is commanded with a program command unit. Caution is required for the arc command of an axis for which the program command unit (#1015 cunit) differs. Command with a decimal point to avoid confusion.

Detailed description

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



(S) Start point
(E) End point

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

$$\frac{L}{2 \cdot r} \leqq 1$$

When L/2 - r > parameter (#1084 RadErr), an alarm will occur.

Where L is the line from the start point to the end point. If an R specification and I, J, (K) specification are given at the same time in the same block, the circular command with the R specification takes precedence. In the case of a full-circle command (where the start and end points coincide), an R specification circular command will be completed immediately even if it is issued and no operation will be executed. An I, J, (K) specification circular command should therefore be used in such a case.

Program example

(Example 1)

G02 Xx1 Yy1 Rr1 Ff1 ;	XY plane R-specified arc

(Example 2)

G03 Zz1 Xx1 Rr1 Ff1 ;	ZX plane R-specified arc

(Example 3)

G02 Xx1 Yy1 Ii1 Jj1 Rr1 Ff1 ;	XY plane R-specified arc (When the R specification and I, J, (K) specification are contained in the same block, the circular command with the R specification takes precedence.)

(Example 4)

G17 G02 Ii1 Jj1 Rr1 Ff1 ;	XY plane This is an R-specified arc, but as this is a circle command, it will be completed immediately.

6.5 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.
If the 3 basic axes and the parallel axes corresponding to these basic axes are entered as parameters, the commands can select the plane composed of any 2 axes which are not parallel axes. If a rotary axis is registered as a parallel axis, the commands can select the plane containing the rotary axis.
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

G17 ; ... Plane selection X-Y

G18 ; ... Plane selection Z-X

G19 ; ... Plane selection Y-Z

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

Detailed description

Parameter entry

	#1026-1028base_I,J,K	#1029-1031aux_I,J,K
I	X	U
J	Y	
K	Z	V

Table 1 Examples of plane selection parameter entry

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) Axis addresses assigned in the same block as the plane selection (G17, G18, G19) command determine which of the basic axes or parallel axes are to be in the actual plane selected.
For the parameter entry example in Table 1.
G17 X__Y__ ; XY plane
G18 X__V__ ; VX plane
G18 U__V__ ; VU plane
G19 Y__Z__ ; YZ plane
G19 Y__V__ ; YV plane

(2) The plane will not changeover at a block where a plane selection G code (G17, G18, G19) is not commanded.
G17 X__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 is assumed that the axis addresses of the 3 basic axes have been omitted.
For the parameter entry example in Table 1.
G17 ; XY plane
G17 U__ ; UY plane
G18 U__ ; ZU plane
G18 V__ ; VX plane
G19 Y__ ; YZ plane
G19 V__ ; 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 entry example in Table 1.
G17 U__Z__ ;
If the above is commanded, the UY plane will be selected, and Z will move regardless of the plane.

(5) When the basic axes or their parallel axes are duplicated and assigned in the same block as the plane selection G code (G17, G18, G19), the plane is determined in the order of basic axes, and then parallel axes.
For the parameter entry example in Table 1.
G17 U__Y__W__ ;
If the above is commanded, the UY plane will be selected, and W will move regardless of the plane.

(Note 1) When the power is turned ON or when the system is reset, the plane set by the parameter "#1025 I_plane" is selected.

6.6 Thread Cutting

6.6.1 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__ α__) F__ Q__ ; ... Normal lead thread cutting

Z (X Y α)	Thread end point
F	Lead of long axis (axis which moves the most) direction
Q	Thread cutting start shift angle (0-360°)

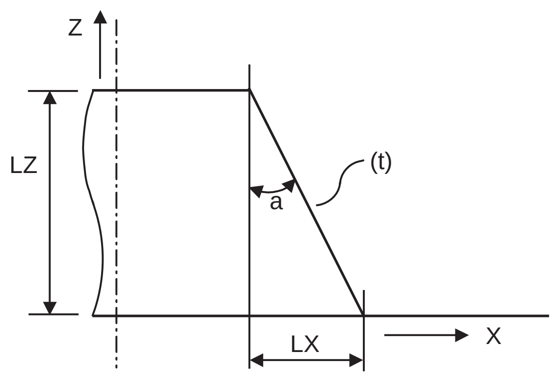
G33 Z__(X__ Y__ α__) E__ Q__ ; ... Precision lead thread cutting

Z (X Y α)	Thread end point
E	Lead of long axis (axis which moves most) direction
Q	Thread cutting start shift angle (0-360°) (Integer)

Detailed description

(1) The E command is also used for the number of ridges in inch thread cutting, and whether the number of ridges or precision lead is to be designated can be selected by parameter setting.(Parameter "#1229 set 01/bit" is set to "1" for precision lead designation.)

(2) The lead in the long axis direction is commanded for the taper thread lead.



(t) Tapered thread section

When a < 45°, lead is in Z-axis direction.
When a < 45°, lead is in X-axis direction.
When a = 45°, lead can be in either Z or X-axis direction.

Thread cutting metric input

Input setting unit	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)
Least Command Increments	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(=1.00000) (1.=1.00000)	1(=1.000) (1.=1.000)
Command range	0.001 - 999.999	0.00001 - 999.99999	0.03 - 999.99	0.00001 - 99.9999	0.000001 - 99.99999	0.1 - 2559999.999

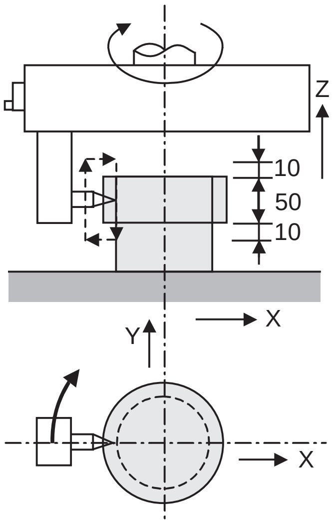
Thread cutting inch input

Input setting unit	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)
Least Command Increments	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(=1.000000) (1.=1.000000)	1(=1.0000) (1.=1.0000)
Command range	0.0001 - 99.9999	0.000001 - 39370078	0.0255 - 9999.9999	0.00001 - 3937009	0.000001 - 3937009	0.25401 - 999.9999

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

(3) The thread cutting will start by the one rotation synchronous signal from the encoder installed on the spindle.
(4) The spindle rotation speed should be kept constant throughout from the rough cutting until the finishing.
(5) 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 occur at the end point of the block following the block in which thread cutting is completed (no longer G33 mode).
(6) 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 occur.
(7) In order to protect the lead during thread cutting, a cutting feedrate which has been converted may sometimes exceed the cutting feed clamp rate.
(8) 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.
(9) The spindle rotation speed is subject to the following restriction :
1 <= R <= Maximum feedrate/Thread lead
Where R <= Tolerable speed of encoder (r/min)
R: Spindle rotation 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.)
(10) The thread cutting start angle is designated with an integer of 0 to 360.

Program example



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 G0 X-210. ;	The tool is evaded in the X axis direction.
N115 Z110. M0 ;	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.
N118 G33 Z40. ;	The second thread cutting is executed.

6.6.2 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__ (X_ Y_ α_)E__ Q__ ; ... Inch thread cutting

Z (X Y α)	Thread end point
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 number of ridges or 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".)

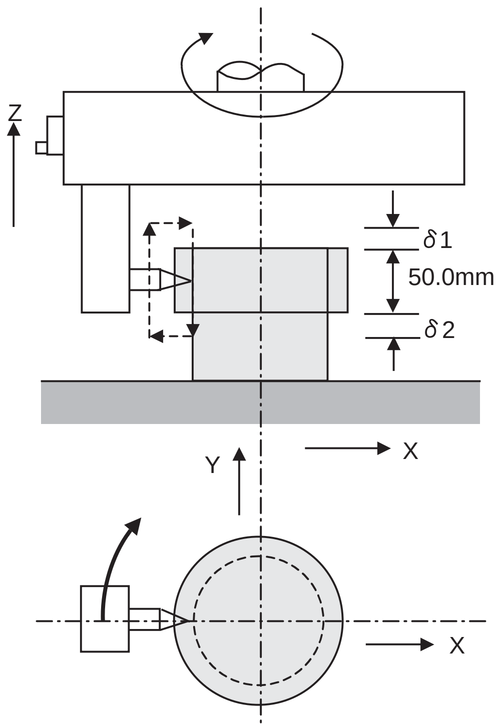
(3) The E command value should be set within the lead value range when converted to lead.

(4) See Section "Constant lead thread cutting" for other details.

Program example

Thread lead ..... 3 threads/inch (= 8.46666 ...)

When programmed with δ 1= 10 mm, δ 2=10 mm using metric input

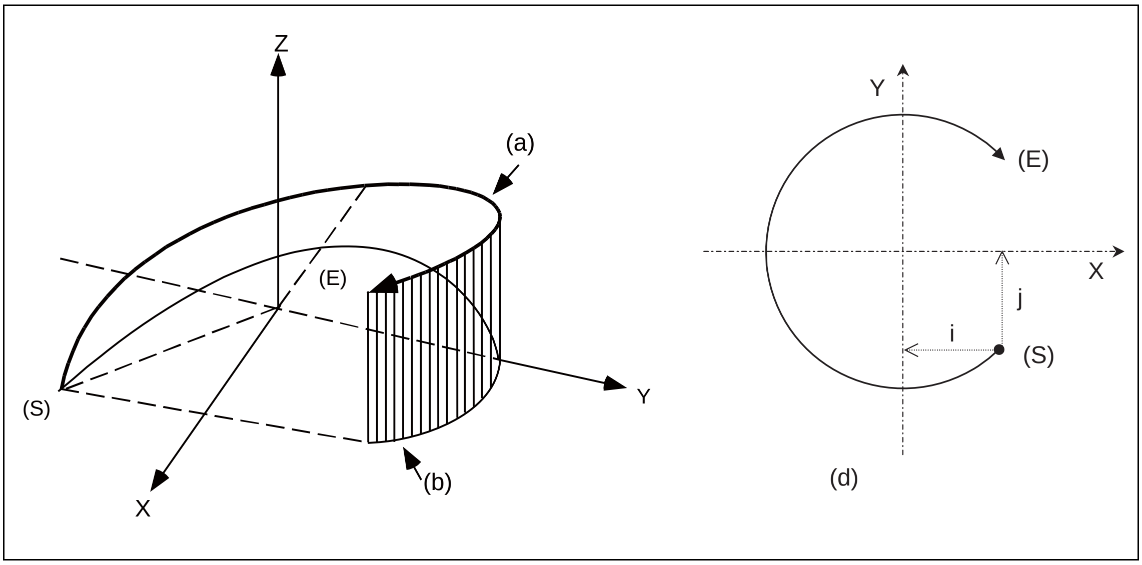


N210 G90 G0 X-200. Y-200. S50 M3 ;	
N211 Z110. ;	
N212 G91 G33 Z-70. E3.0 ;	(First thread cutting)
N213 M19 ;	
N214 G90 G0 X-210. ;	
N215 Z110. M0 ;	
N216 X-200. ; M3 ;	
N217 G04 X2.0 ;	
N218 G91 G33 Z-70. ;	(Second thread cutting)

6.7 Helical Interpolation ; G17 to G19, G02, G03

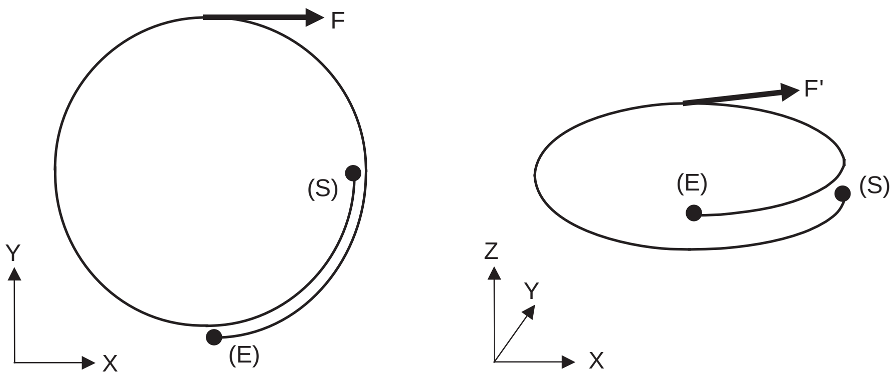
Function and purpose

This function is for circularly interpolating 2 axes on the selected plane and simultaneously interpolating the other axis linearly in synchronization with the circular motion.
When this interpolation is performed with 3 orthogonal axes, the tool will travel helically.



(a) Program command path (b) XY plane projection path in command program
(d) XY plane path (projection path) (S) Start point (E) End point

Normally, the helical interpolation speed is designated with the tangent speed F' including the 3rd axis interpolation element as shown in the figure in the lower left. However, when designating the arc plane element speed, the tangent speed F on the arc plane is commanded as shown in the figure in the lower right. The NC automatically calculates the helical interpolation tangent speed F' so that the tangent speed on the arc plane is F.



(S) Start point (E) End point

Command format

G17/G18/G19 G02/G03 X__ Y__ Z__ I__ J__ P__ F__ ; ... Helical interpolation command (Specify arc center)

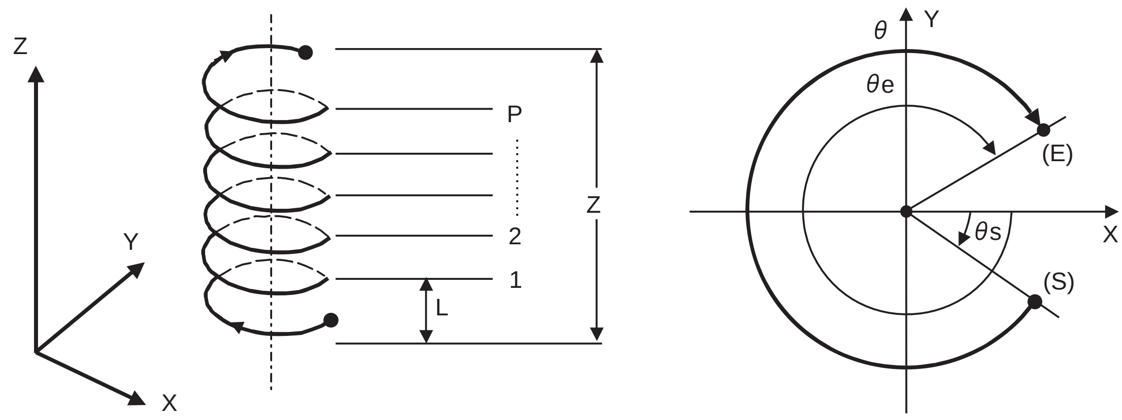
G17/G18/G19 G02/G03 X__ Y__ Z__ R__ F__ ; ... Helical interpolation command (Specify radius (R))

G17/G18/G19	Arc plane (G17: X-Y plane, G18: Z-X plane, G19: Y-Z plane)
G02/G03	Arc rotation direction (G02: clockwise, G03: counterclockwise)
X, Y	Arc end point coordinates
Z	Linear axis end point coordinates
I, J	Arc center coordinates
P	Number of pitches
R	Arc radius
F	Feedrate

Either an absolute value or incremental value can be used for the arc end point coordinate value command and the linear axis 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 and arc radius value are commanded with a program command unit. Caution is required for the helical interpolation command of an axis for which the program command unit (#1015 cunit) differs.
Command with a decimal point to avoid confusion.

Detailed description

Normal speed designation



(S) Start point (E) End point

(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 and Z axis composite element directions speed.

(3) Pitch L is obtained with the following expression.

$$\mathbf{L} = \frac{Z}{(2\pi \cdot P + \theta)/2\pi}$$

$$\theta = \theta e - \theta s = \tan^{-1}\frac{ye}{xe} - \tan^{-1}\frac{ys}{xs} \quad (0 \leq \theta < 2\pi)$$

xs, ys are the start point coordinates from the arc center
xe, ye 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.

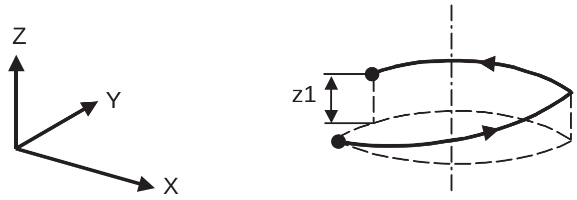
The plane for an additional axis can be selected as with circular interpolation.

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

In addition to the basic command methods above, the command methods shown in the following "Program example" can be used. Refer to the section "Plane Selection; G17,G18,G19" for the arc planes selected with these command methods.

Program example

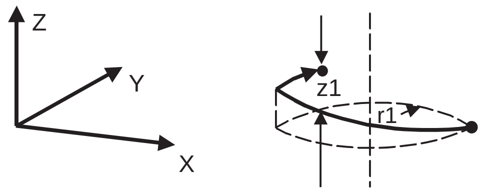
(Example 1)



G17 ;	XY plane
G03 Xx1 Yy1 Zz1 Ii1 Jj1 P0 Ff1;	XY plane arc, Z axis linear

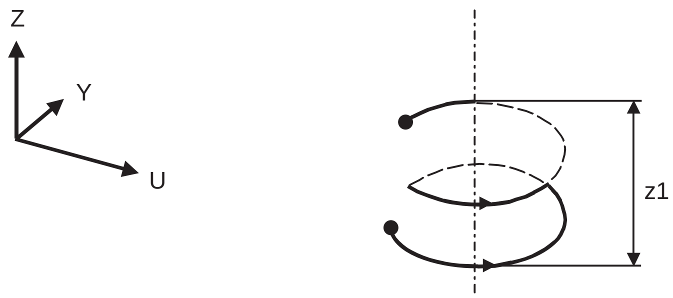
(Note) If pitch No. is 0, address P can be omitted.

(Example 2)



G17 ;	XY plane
G02 Xx1 Yy1 Zz1 Rr1 Ff1;	XY plane arc, Z axis linear

(Example 3)



G17 G03 Uu1 Yy1 Zz1 Ii1 Jj1 P2 Ff1;	UY plane arc, Z axis linear

(Example 4)

G18 G03 Xx1 Uu1 Zz1 Ii1 Kk1 Ff1;	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 Xx1 Uu1 Yy1 Zz1 Ii1 Jj1 Kk1 Ff1;	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.

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 constant direction.

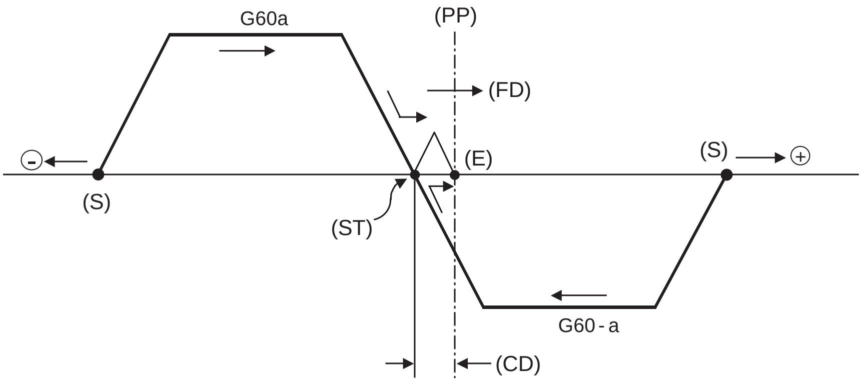
Command format

G60 X__ Y__ Z__ α__; ... Unidirectional positioning

α	Additional 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.



(S) Start point (E) End point (ST) Stop once
(PP) Positioning position (FD) Final advance direction (CD) G60 creep distance

(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) Unidirectional positioning is not performed for the drilling axis during drilling fixed cycles.
(8) Unidirectional 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) Unidirectional 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) will occur when the G60 command is assigned with an NC system which has not been provided with this particular specification.

7

Feed Functions

7.1 Rapid Traverse Rate

Function and purpose

The rapid traverse rate can be set with parameters independently for each axis. The available speed ranges are from 1 mm/min to 1000000 mm/min. The upper limit is subject to the restrictions limited 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.

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 and G03 commands.

Examples (asynchronous feed)

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 μm or 10 μm)

Command mode	F command range	Feedrate command range	Remarks
mm/min	0.001 to 1000000.000	0.001 to 1000000.000 mm/min	
inch/min	0.0001 to 39370.0787	0.0001 to 39370.0787 inch/min	
°/min	0.001 to 1000000.000	0.01 to 1000000 °/min	

(Note 1) A program error (P62) will occur when there is no F command in the first cutting command (G01, G02, G03) after the power has been turned 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.)
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 and G03 modal.
The F1-digit command can also be used for fixed cycle.

Detailed description

Set the corresponding speed of F1 to F5 with the base specification parameters "#1185 spd_F1" to "#1189 spd_F5" respectively.
Operation alarm "104" will occur when the feedrate is 0.

Operation method

(1) Make the F1-digit command valid. (Set the base specification parameter "#1079 F1digt" to 1.)
(2) Set F1 to F5. (Base specification parameter "1185 spd_F1" to "#1189 spd_F5")

Special notes

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

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

(2) F1 to F5 are invalid in the G00 mode and the rapid traverse rate is established instead.
(3) If F0 is used in the G02 or G03 mode, a program error (P121) will occur.
(4) 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.
(5) 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.
(6) 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.
(7) During a F1-digit command, the F1-digit number and F1-digit command signal are output as the PLC interface signals.

F1-digit and G commands

(1) 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

(2) 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.

7.4 Feed Per Minute/Feed Per Revolution (Asynchronous Feed/ Synchronous Feed) ; G94,G95

Function and purpose

Feed per minute (asynchronous feed)

By issuing the G94 command, the commands from that block are issued directly by the numerical value following F as the feedrate per minute (mm/min, inch/min).

Feed per revolution (synchronous feed)

By issuing the G95 command, the commands from that block are issued directly by the numerical value following F as the feedrate per spindle revolution (mm/rev, inch/rev).
When this command is used, the rotary encoder must be attached to the spindle.

Command format

G94; ... Feed per minute (mm/min) (asynchronous feed)

G95; ... Feed per revolution (mm/rev) (synchronous feed)

Detailed description

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

(1) The F code command range is as follows.

Metric input

Input unit system	**B(0.001mm)**		**C(0.0001mm)**	
Command Mode	Feed per minute	Feed per revolution	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)	F(mm/min)	F(mm/rev)
Minimum command unit	1 (= 1.00) (1. = 1.00)	1 (= 0.01) (1. = 1.00)	1 (= 1.000) (1. = 1.000)	1 (= 0.01) (1. = 1.00)
Command range	0.01 - 1000000.00	0.001 - 999.999	0.001 - 100000.000	0.0001 - 99.9999

Inch input

Input unit system	**B(0.0001inch)**		**C(0.00001inch)**	
Command Mode	Feed per minute	Feed per revolution	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)	F(inch/min)	F(inch/rev)
Minimum command unit	1 (= 1.000) (1. = 1.000)	1 (= 0.001) (1. = 1.000)	1 (= 1.0000) (1. = 1.0000)	1 (= 0.001) (1. = 1.000)
Command range	0.001 - 100000.0000	0.0001 - 999.9999	0.0001 - 10000.00000	0.00001 - 99.99999

(2) The effective rate (actual movement speed of machine) under per-revolution feed conditions is given in the following formula (Formula 1).
FC = F × N × OVR ...... (Formula 1)
FC : Effective rate (mm/min, inch/min)
F : Commanded feedrate (mm/rev, inch/rev)
N : Spindle rotation 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.

Precautions

(1) The effective rate (mm/min or inch/min), which is produced by converting the commanded speed, the spindle rotation 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.
(2) When the above effective rate exceeds the cutting feed clamp rate, it is clamped at that clamp rate.
(3) If the spindle rotation speed is zero when feed per revolution is executed, operation alarm "105" occurs.
(4) Feedrate during the machine lock is the command speed.
(5) Under dry run conditions, feed per minute applies and movement results at the externally set rate (mm/min,inch/min, or °/min).
(6) The fixed cycle G84 (tapping cycle) and G74 (reverse tapping cycle) are executed according to the feed mode that is already designated.
(7) Whether feed per minute (G94) or feed per revolution (G95) is to be established when the power is turned ON or when M02 or M30 is executed can be selected by setting parameter "#1074 I_Sync".

7.5 Feedrate Designation and Effects on Control Axes

Function and purpose

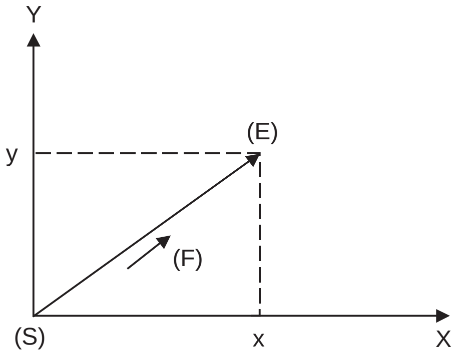
It has already been mentioned that a machine has a number of control axes. These control axes can be divided into linear axes which control linear movement and rotary axes which control rotary movement. The feedrate is designed to assign the displacement speed of these axes, and the effect exerted on 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.

Detailed description

When controlling linear axes

Both when only one machine axis is to be controlled and when there are two or more axes to be controlled simultaneously, the feedrate 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:



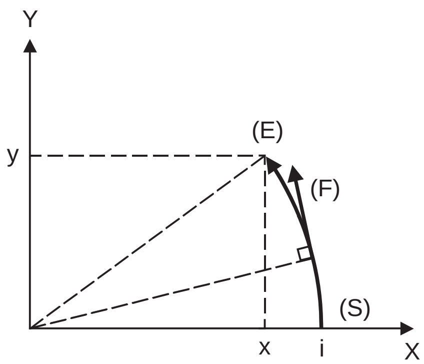
$$fx = f \times \frac{x}{\sqrt{x^2 + y^2}}$$

$$fy = f \times \frac{z}{\sqrt{x^2 + y^2}}$$

(S) Tool start point　　(E) Tool end point　　(F) Speed in this direction is "f".
fx: Feedrate for X axis　　fy: Feedrate for Y axis

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.



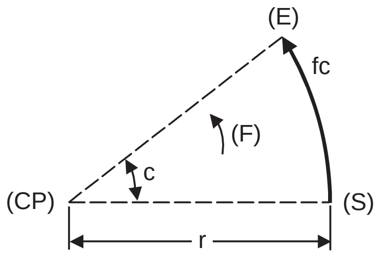
(S) Tool start point
(E) Tool end point
(F) Speed in this direction is "f".

In this case, the feedrate of the X and Y 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 (C) is to be controlled
("f" units = ° /min)



(S) Tool start point
(E) Tool end point
(CP) Center of rotation
(F) Angular speed is "f".

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

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

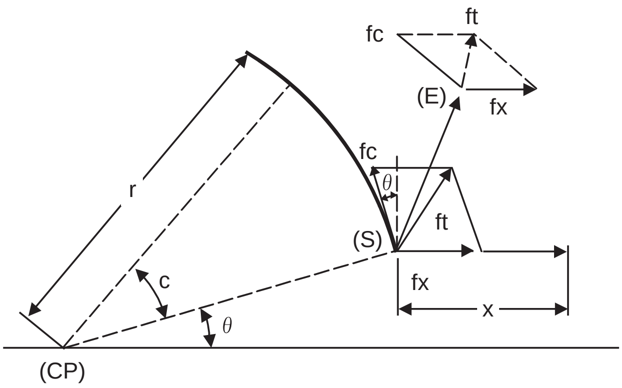
Therefore, the feedrate to be designated in the program must be :

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

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° 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, in the components for each axis of the numerical values assigned by F will be the same as previously described in "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 feedrate 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 values is "c":



(S) Tool start point
(E) Tool end point
(CP) Center of rotation

Size and direction are fixed for fx.
Size is fixed for fc but direction varies.
Both size and direction vary for ft.

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

$$fx = f \times \frac{x}{\sqrt{x^2 + c^2}} \quad \text{...... (1)}$$

$$\omega = f \times \frac{c}{\sqrt{x^2 + c^2}} \quad \text{...... (2)}$$

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

$$fc = \omega \times \frac{\pi \times r}{180} \quad \text{...... (3)}$$

If the speed in the tool advance direction at start point (S) 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:

$$ftx = -r\sin\left(\frac{\pi}{180}\theta\right) \times \frac{\pi}{180}\omega + fx \quad \text{...... (4)}$$

$$fty = -r\cos\left(\frac{\pi}{180}\theta\right) \times \frac{\pi}{180}\omega \quad \text{...... (5)}$$

r is the distance between center of rotation and tool (in mm units)
θ is the angle between the (S) 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:

$$ft = \sqrt{ftx^2 + fty^2}$$

$$= f \times \frac{\sqrt{\left(x^2 - x \times c \times r\sin\left(\frac{\pi}{180}\theta\right)\frac{\pi}{90}\right) + \left(\frac{\pi \times r \times c}{180}\right)^2}}{x^2 + c^2} \quad \text{...... (6)}$$

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

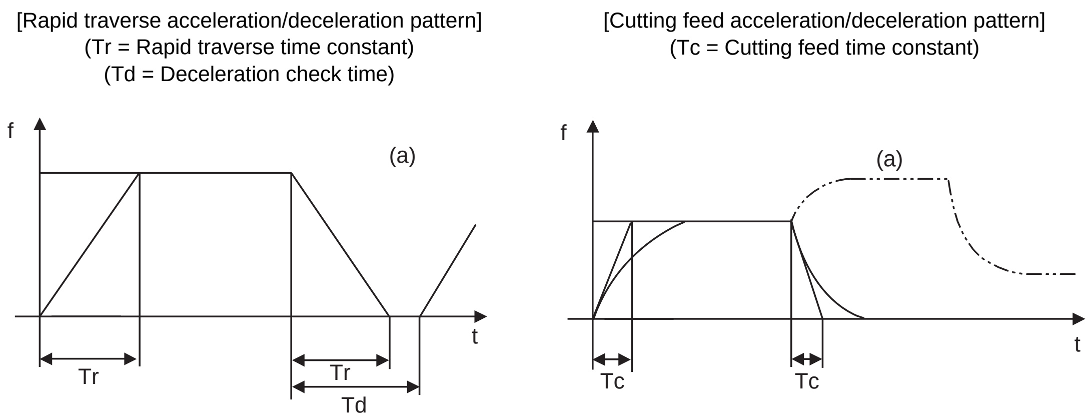
$$f = ft \times \frac{x^2 + c^2}{\sqrt{x^2 - x \times c \times r\sin\left(\frac{\pi}{180}\theta\right)\frac{\pi}{90} + \left(\frac{\pi \times r \times c}{180}\right)^2}} \quad \text{...... (7)}$$

"ft" in formula (6) is the speed at the (S) 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.

7.6 Automatic Acceleration/Deceleration

Function and purpose

The rapid traverse and manual feed acceleration/deceleration pattern is linear acceleration and linear deceleration. Time constant TR can be set independently for each axis using parameters in 1ms steps from 1 to 500ms.
The cutting feed (not manual feed) acceleration/deceleration pattern is exponential acceleration/deceleration. Time constant Tc can be set independently for each axis using parameters in 1ms steps from 1 to 500ms. (Normally, the same time constant is set for all axes.)



(a) With continuous commands

With rapid traverse and manual feed, the following block is executed after the command pulse of the present block has become "0" and the tracking error of the acceleration/deceleration circuit has become "0". However, with cutting feed, the following block is executed as soon as the command pulse of the present block becomes "0" although an external signal (error detection) can detect that the tracking error of the acceleration/deceleration circuit has reached "0" and the following block can be executed. When the in-position check has been made valid (selected by parameter "#1193 inpos" during the deceleration check, it is first confirmed that the tracking error of the acceleration/deceleration circuit has reached "0", then it is checked that the position deviations less than the parameter setting value "#2204 SV024", and finally the following block is executed. It depends on the machine as to whether the error detection function can be activated by a switch or M function and so reference should be made to the instructions issued by the machine tool builder.

7.7 Rapid Traverse Constant Inclination Acceleration/Deceleration

Function and purpose

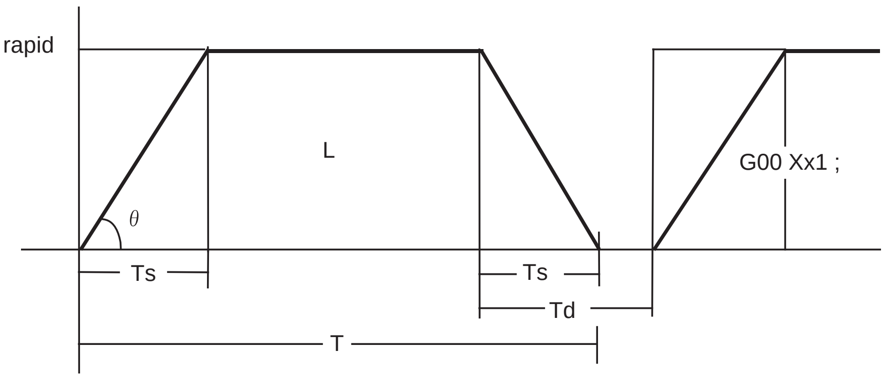
This function performs acceleration and deceleration at a constant inclination during linear acceleration/ deceleration in the rapid traverse mode. Compared to the method of acceleration /deceleration after interpolation, the constant inclination acceleration/deceleration method makes for improved cycle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



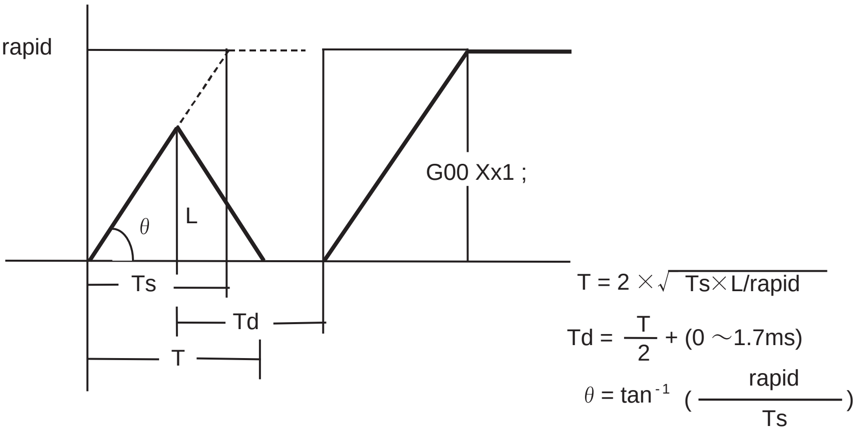
$$T = \frac{L}{rapid} + Ts$$

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

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

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

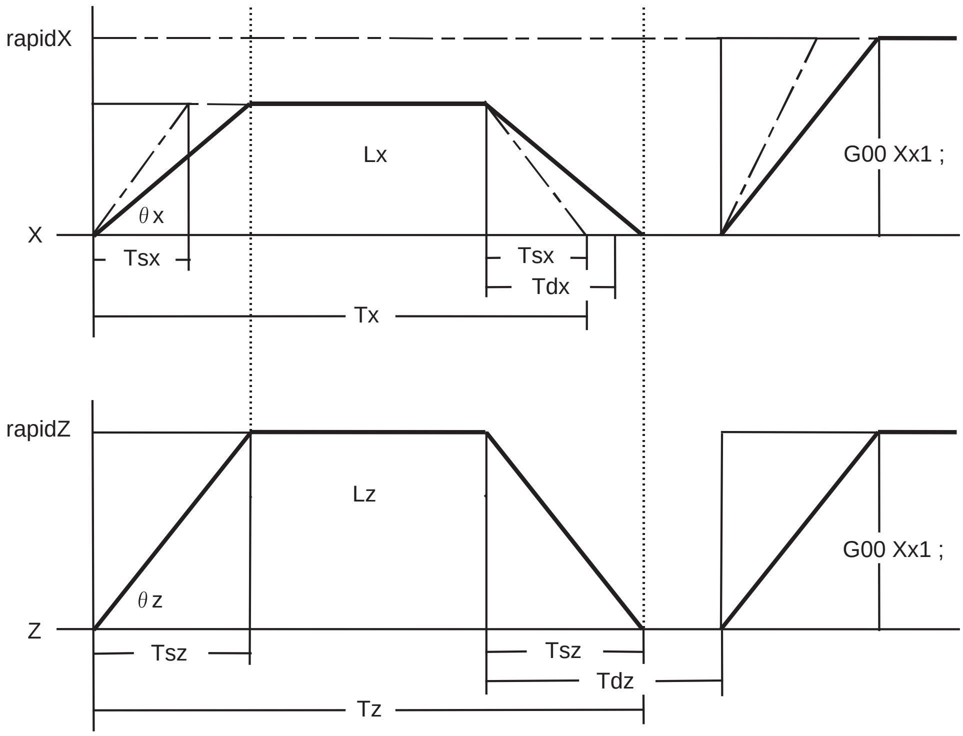
[When the interpolation distance is shorter than the acceleration and deceleration distance]



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

(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, Tsx < Tsz, Lx ≠ Lz)]



When Tsz is greater than Tsx, Tdz is also greater than Tdx, and Td = Tdz in this block.

(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).

7.8 Speed Clamp

Function and purpose

This function exercises control over the actual cutting feedrate in which override has been applied to the cutting feedrate command so that the speed clamp value which has been preset independently for each axis is not exceeded.

(Note) Speed clamping is not applied to feed per rotation and thread cutting.

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 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 the parameter "#1193 inpos".
The in-position width is set into parameter the servo parameter "#2224 sv024".

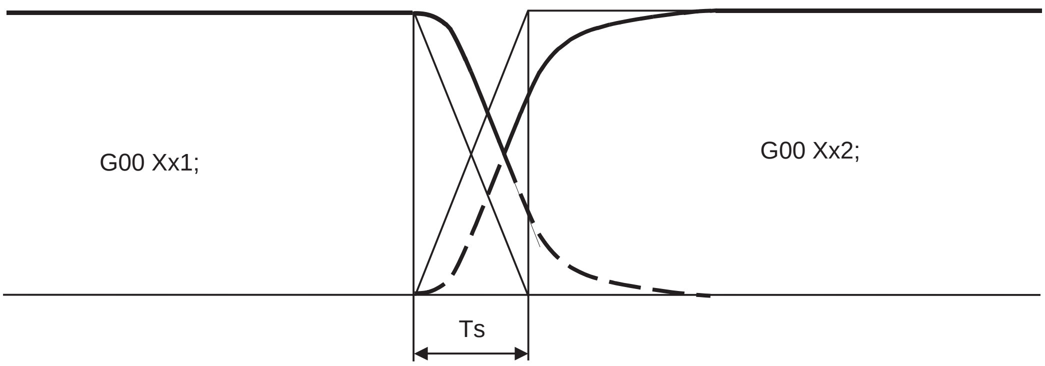
Command format

G09 ; ... Exact stop check

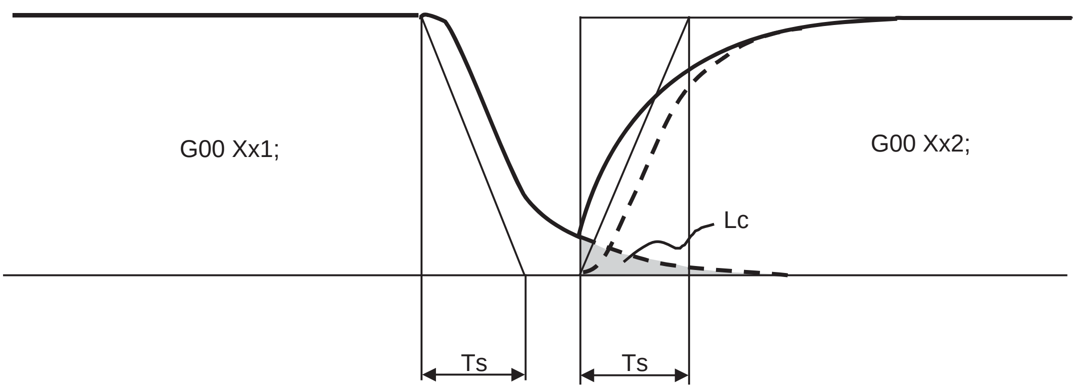
The exact stop check command G09 has an effect only with the cutting command (G01 - G03) in its particular block.

Detailed description

[With continuous cutting feed]

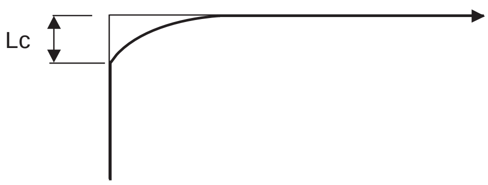


[With cutting feed in-position check]



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

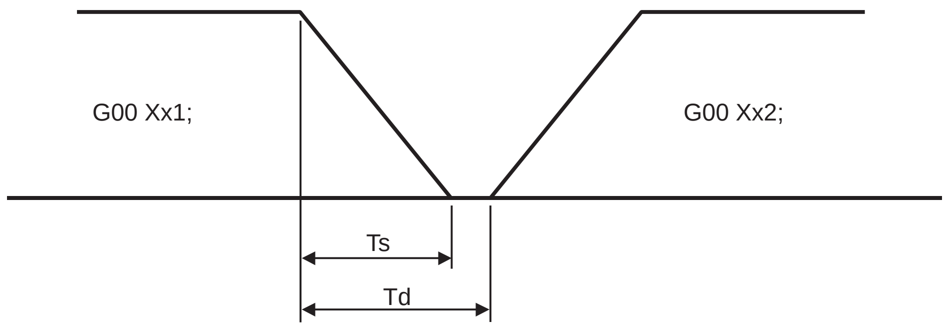
As shown in the figure above, the remaining distance (shaded area in the above figure) of the previous block when the next block is started can be set into the servo parameter "#2224 sv024" as the in-position width "Lc".
The setting unit for the servo parameter "#2224 SV024" is 0.0005mm or 0.00005inch.
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.

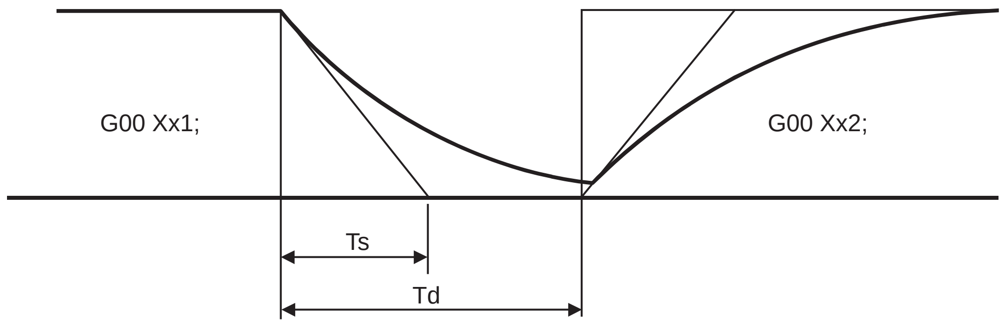
With deceleration check

(1) With linear acceleration/deceleration



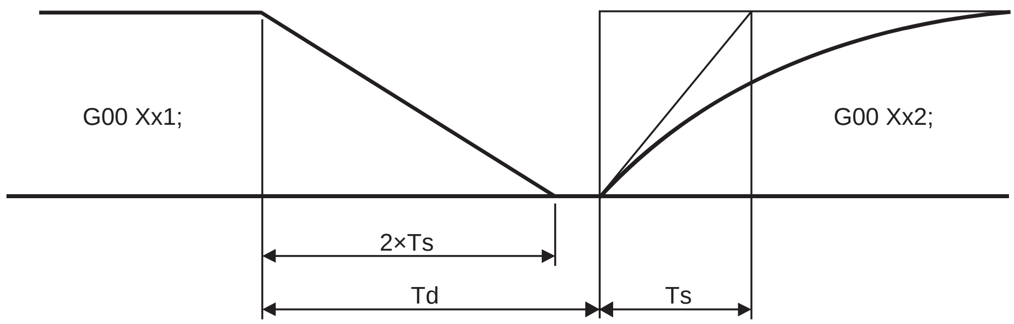
TS: Acceleration/deceleration time constant Td: Deceleration check time Td = Ts + (0 to 14ms)

(2) With exponential acceleration/deceleration



TS: Acceleration/deceleration time constant Td: Deceleration check time Td = 2×Ts + (0 to 14ms)

(3) With exponential acceleration/linear deceleration



TS: Acceleration/deceleration time constant Td: Deceleration check time Td = 2×Ts + (0 to 14ms)

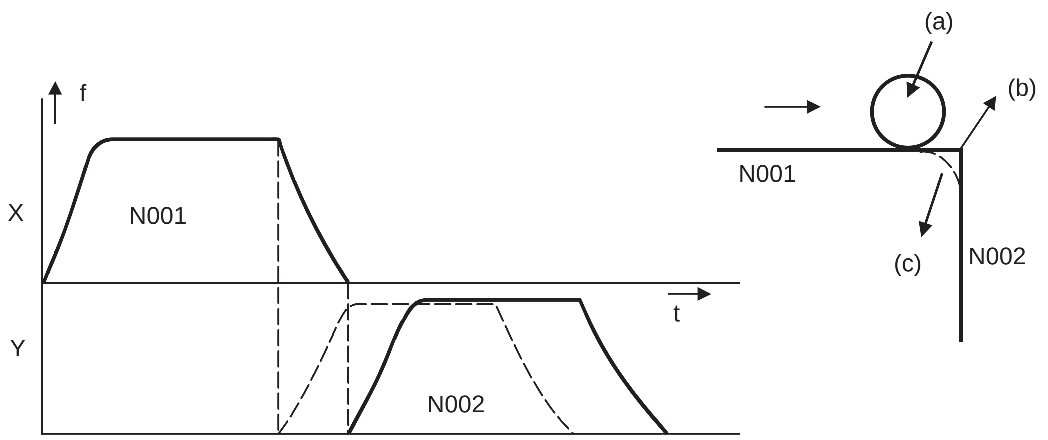
The time required for the deceleration check during cutting feed is the longest among the cutting feed deceleration check times of each axis determined by the cutting feed acceleration/deceleration time constants and by the cutting feed acceleration/ deceleration mode of the axes commanded simultaneously.

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

Program example

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

[Exact stop check result]



(a) Tool (b) With G09 (c) Without G09

f: Commanded speed t: Time

Solid line indicates speed pattern with G09 command

Broken line indicates speed pattern without G09 command

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.

The modal command is released by the following commands.

G61.1..........High-accuracy control
G62 .... Automatic corner override
G63 .... Tapping mode
G64 .... Cutting mode

Command format

G61 ; ... Exact stop check mode

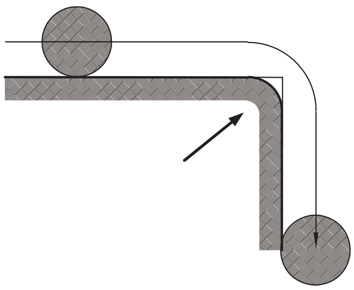
In-position check is executed when the G61 command has been selected, and thereafter, the in-position check is executed at the end of the cutting command block until the check mode is canceled.

7.11 Deceleration Check

Function and purpose

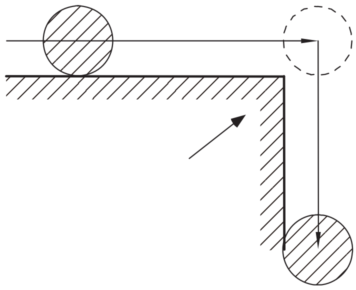
With the deceleration check function, a deceleration stop is executed at the block seam before the next block is executed, preventing corner roundness by reducing the machine shock that occurs when the control axis feedrate is suddenly changed.

N010 G90 G01 X100 ;
N011 G01 Y-50 ;



Corner rounding occurs because the N011 block is started before the N010 command is completely finished.

N010 G09 G90 G01 X100 ;
N011 G01 Y-50 ;



A sharp edge is formed because the N011 block is started after the N010 remaining distance has reached the command deceleration check width or the in-position check width.

Detailed description

Conditions for Executing the Deceleration Check

(1) Deceleration check during rapid traverse
During the rapid traverse mode, deceleration check is carried out at the block seam before executing the next block.

(2) Deceleration check during cutting feed
The deceleration check is carried out at the block joints (before executing the next block) during cutting feed when any one of the following conditions is valid.
(a)When the error detect switch (external signal) is ON.
(b)When G09 (exact stop check) is commanded in the same block.
(c)When G61 (exact stop check mode) has been selected.
(Note)The G61 command is a modal command. The modal is canceled by the following commands.
G61.1: High accuracy control
G62: Automatic corner override
G63: Tapping mode
G64: Cutting mode
(d) When the next block is rapid traverse and the deceleration check during rapid traverse and cutting feed is valid ("#1193 inpos" is set to "2" or "3").

(Note) If any of the following conditions is met, a deceleration check is carried out regardless of commands.
- Cutting feed in the synchronous tapping mode and the next block is rapid traverse.
- Cutting feed in G61.1 (high-accuracy control mode) and the next block is rapid traverse.
- Changing from G64 (cutting mode) to G61.1 (high-accuracy control mode).

Deceleration Check and Parameters

Select the deceleration check method with these parameters.
[Base specification parameter] #1193 inpos Deceleration check method selection

#1193 inpos	Command mode			
	Rapid traverse	G09 + G01	G01 -> G00	G01 -> G01
0	Command deceleration check	Command deceleration check	Deceleration is not applied	Deceleration is not applied
1	In-position	In-position	Deceleration is not applied	Deceleration is not applied
2	Command deceleration check	Command deceleration check	Command deceleration check	Deceleration is not applied
3	In-position	In-position	Command deceleration check	Deceleration is not applied

(Note) When G0 acceleration/deceleration before interpolation is valid ("#1205 G0bdcc" is set to 1) and the high-accuracy control mode is OFF, a deceleration check is always carried out at G01 and G00 block.

Operation when G0 acceleration/deceleration before interpolation is valid

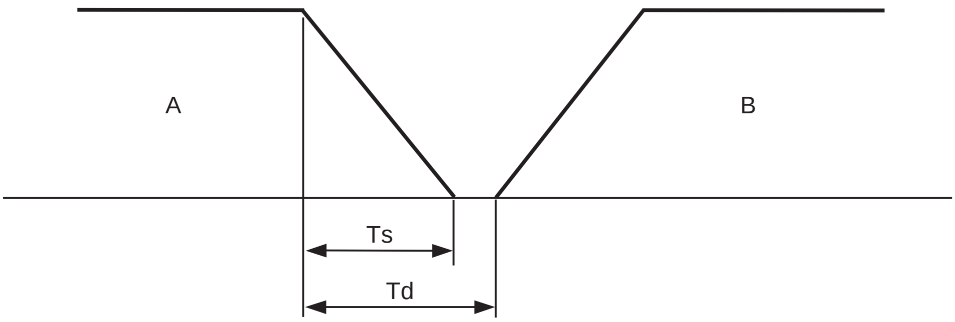
#1193 inpos	Command mode			
	Rapid traverse	G09 + G01	G01 -> G00	G01 -> G01
0	Command deceleration check	Command deceleration check	Command deceleration check	Deceleration is not applied
1	In-position check	In-position check	In-position check	Deceleration is not applied
2	Command deceleration check	Command deceleration check	Command deceleration check	Deceleration is not applied
3	In-position check	In-position check	Command deceleration check	Deceleration is not applied

List of parameters for each axis

#	Item		Detail	Setting range (unit)
2077	G0inps	G0 in-position width	This parameter becomes valid when executing an in-position check at rapid traverse rate. Between SV024 and this parameter, the parameter with a larger value will be applied. (Note) This parameter is valid when executing an in-position check (#1193 inpos:1/3).	0.000 to 99.999(mm)
2078	G1inps	G1 in-position width	This parameter becomes valid when executing an in-position check at cutting feedrate. Between SV024 and this parameter, the parameter with a larger value will be applied. (Note) This parameter is valid when executing an in-position check (#1193 inpos:1/3).	0.000 to 99.999(mm)
2224	SV024	In-position detection width	Set the in-position detection width. (Note) This parameter is valid when executing an in-position check (#1193 inpos:1/3).	0 to 32767(μm)

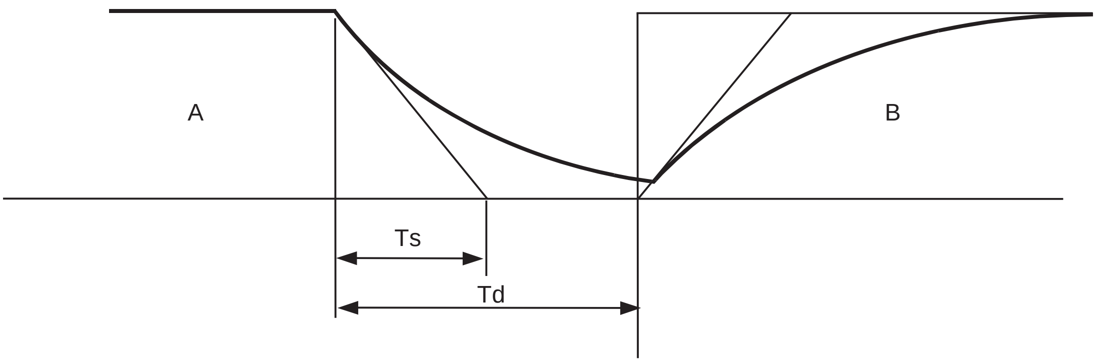
Deceleration Check Method

(1) Command deceleration check
After interpolation for one block has been completed, the completion of the command system deceleration is confirmed before execution of the next block. The time required for the deceleration check is determined according to the acceleration/deceleration mode and acceleration/deceleration time constant.
(a)For linear acceleration/deceleration



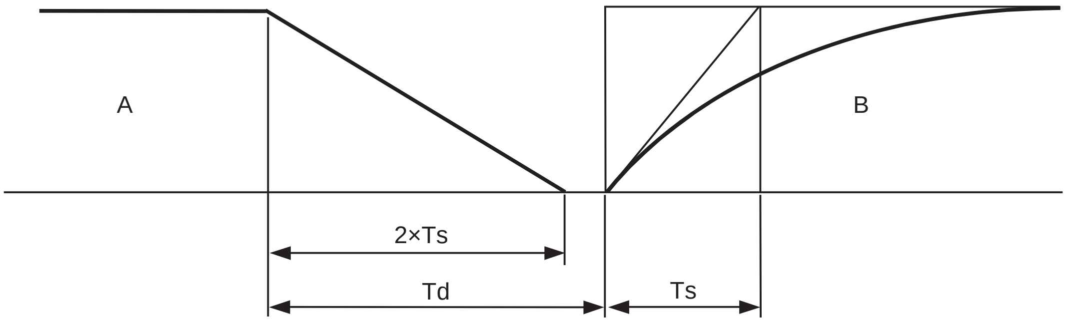
A: Previous block B: Next block
Ts : Acceleration/deceleration time constant
Td : Deceleration check time Td = Ts + (0 to 7ms)

(b)For exponential acceleration/deceleration



A: Previous block B: Next block
Ts : Acceleration/deceleration time constant
Td : Deceleration check time Td = Ts + (0 to 7ms)

(c)For exponential acceleration and linear deceleration

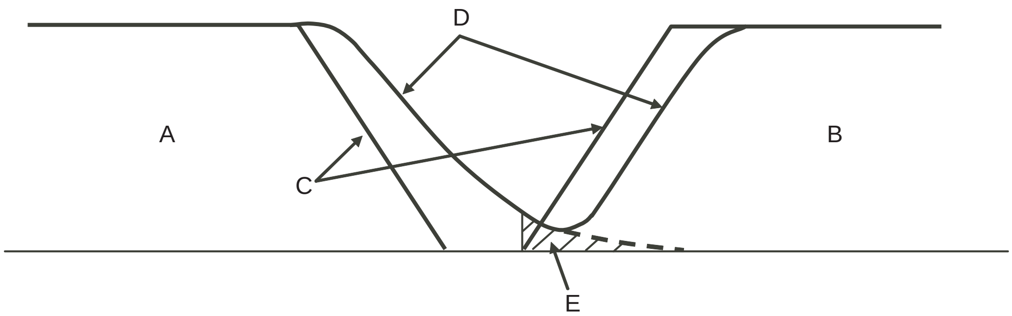


A: Previous block B: Next block
Ts : Acceleration/deceleration time constant
Td : Deceleration check time Td = Ts + (0 to 7ms)

The deceleration check time required during rapid traverse is the longest rapid traverse deceleration check time of all axes. This check time is determined by the rapid traverse acceleration/deceleration mode and rapid traverse acceleration/deceleration time constant of simultaneously commanded axes. The deceleration check time required during cutting feed is determined in the same manner. It is the longest cutting feed deceleration check time of all axes. This check time is determined by the cutting feed acceleration/deceleration mode and cutting feed acceleration/deceleration time constant of simultaneously commanded axes.

(2) In-position check

With the in-position check, after the commanded deceleration check is carried out it is confirmed that the servo system's position error amount is less than the value set in the parameters before executing the next block.



A: Previous block B: Next block C: Command
D: Servo E: In-position width (with servo parameter INP)

7.12 Automatic Corner Override ; G62

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 feedrate.
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.

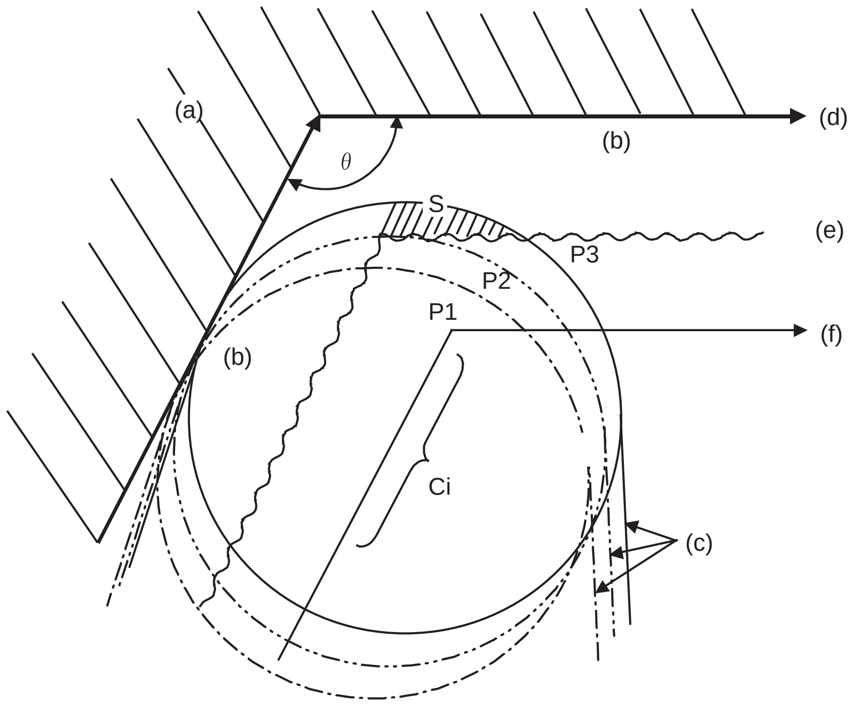
Command format

G62 ; ... Automatic Corner Override

Detailed description

Machining inside corners

When cutting an inside corner, as shown in the figure below, 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.



(a) Workpiece (b) Machining allowance (c) Tool
(d) Programmed path (finished shape) (e) Workpiece surface shape
(f) Tool center path
θ : Max. angle at inside corner
Ci : Deceleration range (IN)

[Operation]

(1) When automatic corner override is not to be applied :
When the tool moves in the order of P1 -> P2 -> P3 in the above figure, the machining allowance at P3 increase by an amount equivalent to the area of shaded section S and so that tool load increases.

(2) When automatic corner override is to be applied :
When the inside corner angle θ in the above figure is less than the angle set in the parameter, the override set into the parameter is automatically applied in the deceleration range Ci.

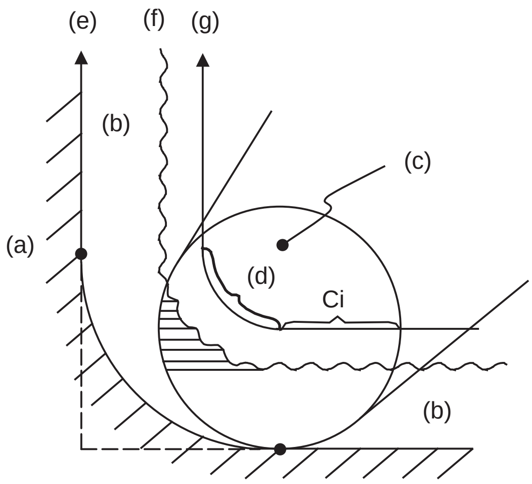
[Parameter setting]

The following parameters are set into the machining parameters :

#	Parameter	Setting range
#8007	OVERRIDE	0 to 100 [%]
#8008	Max. angle at inside corner	0 to 180 [°]
#8009	DSC. ZONE	0 to 99999.999 [mm] or 0 to 3937.000 [inch]

Refer to the Instruction Manual for details on the setting method.

Automatic corner R

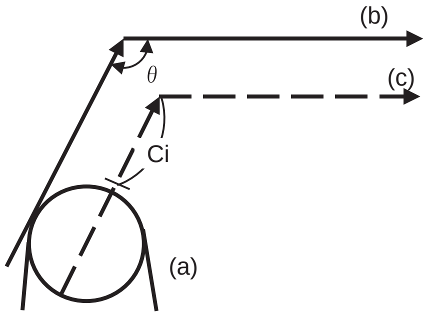


(a) Workpiece
(b) Machining allowance
(c) Corner R center
(d) Corner R section
(e) Programmed path
(f) Workpiece surface shape
(g) Tool center path

(1) The override set in the parameter is automatically applied at the deceleration range Ci and corner R section for inside offset with automatic corner R. (There is no angle check.)

Application example

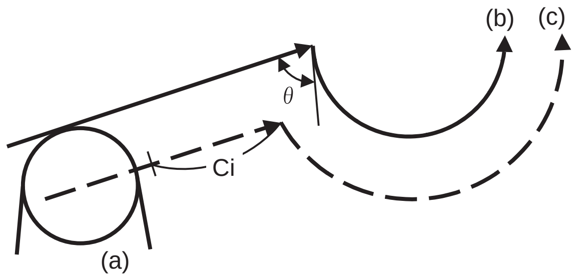
(1) Linear - linear corner



(a) Tool
(b) Program
(c) Tool center

The override set in the parameter is applied at Ci.

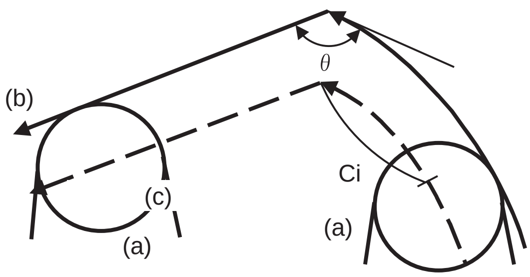
(2) Linear - arc (outside offset) corner



(a) Tool
(b) Program
(c) Tool center

The override set in the parameter is applied at Ci.

(3) Arc (inside offset) - linear corner

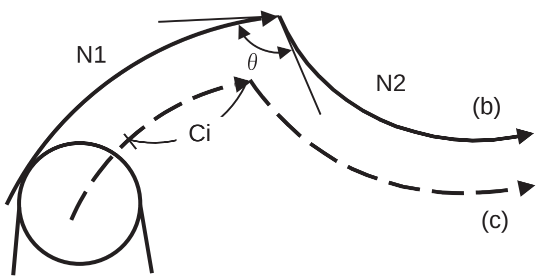


(a) Tool
(b) Program
(c) Tool center

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 offset) - arc (outside offset) corner



(a) Tool
(b) Program
(c) Tool center

The override set in the parameter is applied at Ci.

Relation with other functions

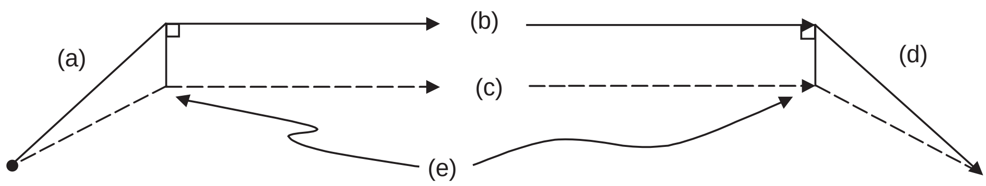
Function	Override at corner
Cutting feed override	Automatic corner override is applied after cutting feed override has been applied.
Override cancel	Automatic corner override is not canceled by override cancel.
Speed clamp	Valid after automatic corner override
Dry run	Automatic corner override is invalid.
Synchronous feed	Automatic corner override is applied to the synchronous feedrate.
Thread cutting	Automatic corner override is invalid.
G31 skip	Program error occurs 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.

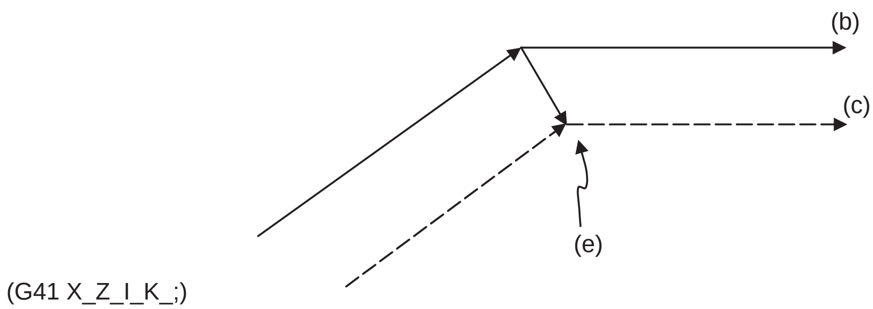
(2) Even if the automatic corner override mode is entered, the automatic corner override will not be applied until the tool radius compensation mode is entered.

(3) Automatic corner override will not be applied on a corner where the tool radius compensation is started or canceled.



(a) Start-up block (b) Program
(c)Tool center (d) Cancel block
(e) Automatic corner override will not be applied.

(4) Automatic corner override will not be applied on a corner where the tool radius compensation I, K vector command is issued.



(b) Program (c) Tool center
(e) Automatic corner override will not be applied.

(5) Automatic corner override will not be applied when intersection calculation cannot be executed. Intersection calculation cannot be executed in the following case.
-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),automatic corner override (G62), or cutting mode (G64) command.
The machine is in the cutting mode status when its power is turned ON.

Command format

G63; ... Tapping mode

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 check mode (G61), high-accuracy control mode (G61.1), automatic corner override (G62), or tapping mode (G63).
The machine is in the cutting mode status when its power is turned ON.

Command format

G64; ... Cutting mode

8

Dwell

8.1 Dwell (Time Designation) ; 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__ /P__ ; ... Dwell (Time designation)

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) The dwell time command range is as follows.
0.001 to 99999.999(s)

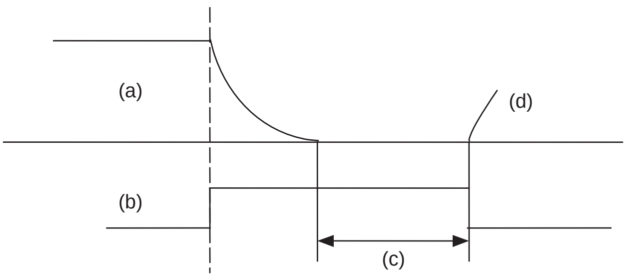
(3) The dwell time setting unit applied when there is no decimal point can be made 1s by setting 1 in the parameter "#1078 Decpt2". This is effective only for X and P for which the decimal command is valid.

(4) 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.

(5) The dwell is valid during the interlock.

(6) The dwell is valid even for the machine lock.

(7) 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.



(a) Previous block cutting command
(b) Dwell command
(c) Dwell time
(d) Next block

Program example

Command	Dwell time [s]	
	#1078 Decpt2 = 0	#1078 Decpt2 = 1
G04 X500 ;	0.5	500
G04 X5000 ;	5	5000
G04 X5. ;	5	5
G04 X#100 ;	1000	1000
G04 P5000 ;	5	5000
G04 P12.345 ;	12.345	12.345
G04 P#100 ;	1000	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) 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 seconds.

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)

Function and purpose

The miscellaneous functions are also known as M functions, and they command auxiliary functions, such as spindle forward and reverse rotation, operation stop and coolant ON/OFF.

Detailed description

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 Mm1 Mm2 Mm3 Mm4 ;

When five or more commands are issued in a block, only the last four will be valid.
The output signal is an 8-digit BCD code and start signal.
The eight commands of M00, M01, M02, M30, M96, M97, M98 and M99 are used as auxiliary commands for specific objectives and so they cannot be used as general auxiliary commands. Therefore, 92 miscellaneous functions are available.
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. The machine specifications determine which sequence applies.

(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 except M96, M97, M98 and M99.

Program stop : M00

When the NC has read this function, it stops reading the next block. As far as the NC system's functions are concerned, it only stops reading the next block. 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 M01 command is read when the optional stop switch on the machine operation board is ON, it will stop reading the next block and perform the same operation as the M00.
If the optional stop switch is OFF, the M01 command is ignored.

(Example)	The status and operation of optional stop switch
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 cueing up the machining program. Whether the program is actually cued up 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 cueing up the program 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 cueing up 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.

Macro interruption; M96, M97

M96 and M97 are M codes for user macro interrupt control.
The M code for user macro interrupt control is processed internally, and is not output externally.
To use M96 and M97 as miscellaneous functions, change to another M code with the parameter (#1109 subs_M, #1110 M96_M and #1111 M97_M).

Subprogram call/completion : M98, M99

These commands are used as the return instructions from branch destination subprograms and branches to subprograms.
M98 and M99 are processed internally and M code signals and strobe signals are not output.

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 machining program’s cueing up operations and the initialization of modals by resetting differ according the machine specifications.

9.2 Secondary Miscellaneous Functions (A8-digits, B8-digits or C8-digits)

Function and purpose

These serve to assign the indexing table positioning and etc. 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.

Detailed description

Select the address A, B or C that is used for the secondary miscellaneous function by a parameter (#1170 M2name). (Except the address that is used for the axis name.)
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 gives 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**
Secondary miscellaneous function	A	-	○	○
	B	○	-	○
	C	○	○	-

Precautions

When A has been assigned as the secondary miscellaneous function address, the following command cannot be used.
- Geometric command

10

Spindle Functions

10.1 Spindle Functions

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.

10.2 Constant Surface Speed Control ; G96,G97

Function and purpose

These commands automatically control the spindle rotation 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 point speed constant during the cutting.

Command format

G96 S__ P__ ; ... Constant surface speed ON

S	Surface speed (1 to 99999999 m/min)
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 rotation speed is controlled so that the surface speed is 200m/min.
:	
G97 G01 X50. Z100. F300 S500 ;	The spindle rotation speed is controlled to 500r/min.
:	
M02 ;	The modal returns to the initial value.

(4) Constant surface speed control can be commanded on the selected spindle (nth spindle) / the 2nd spindle.
Select which spindle (the selected spindle or 2nd one) the commands are made to by the spindle selection G codes (G43.1 and G44.1).
Select which spindle (the selected spindle or 2nd one) is valid as the initial state with the parameter (base specifications parameter "#1199 Sselect").

(5) Select whether calculating the surface speed at rapid traverse command is performed constantly or only at the block end point.

Precautions

Under the constant surface speed control (during G96 modal), if the axis targeted for the constant surface speed control moves toward the spindle center, the spindle rotation speed will increase and may exceed the allowable speed of the workpiece or chuck, etc. In this case, the workpiece, etc. may jump out during machining, which may result in breakage of tools or machine or may cause damage to the operators. Thus make sure to use this control while the "spindle speed clamp" is enabled.
When the constant surface speed control is commanded, keep enough distance from the program zero point.

Program example

(1) When the parameter "1146 Sclamp" is set to "0".
G96 S200 ; ... The spindle rotation speed is controlled so that the surface speed is 200m/min.
G92 S4000 Q200 ; ... The spindle rotation speed is clamped up to 4000r/min and down to 200r/min.
M3 ; ... The rotation command to the spindle

(2) When the parameter "1146 Sclamp" is set to "1".
G92 S4000 Q200 ; ... The spindle rotation speed is clamped up to 4000r/min and down to 200r/min.
G96 S200 ; ... The spindle rotation speed is controlled so that the surface speed is 200m/min.
M3 ; ... The rotation command to the spindle

(Note) For safety, issue the rotation command to the spindle after G92.

⚠ WARNING

1. Under the constant surface speed control (during G96 modal), if the axis targeted for the constant surface speed control moves toward the spindle center, the spindle rotation speed will increase and may exceed the allowable speed of the workpiece or chuck, etc. In this case, the workpiece, etc. may jump out during machining, which may result in breakage of tools or machine or may cause damage to the operators.

10.3 Spindle Clamp Speed Setting ; G92

Function and purpose

The maximum clamp rotation speed of the spindle can be assigned by address S following G92 and the minimum clamp rotation speed by address Q.
Use this command when the spindle speed needs to be limited depending on the workpiece to be machined, the chuck to be mounted on the spindle and the tool specifications, etc.

Command format

G92 S__ Q__ ; ... Spindle Clamp Speed Setting

S	Maximum clamp rotation speed (r/min)
Q	Minimum clamp rotation speed (r/min)

Detailed description

(1) Besides this command, parameters can be used to set the rotation 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 rotation speed ranges based on the parameters and based on "G92 Ss Qq ;".

(2) Set in the parameters "#1146 Sclamp" and "#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.

(Note 1) G92S command and rotation speed clamp operation

		Sclamp=0		**Sclamp=1**	
		aux11/bit5=0	**aux11/bit5=1**	**aux11/bit5=0**	**aux11/bit5=1**
Command	**In G96**	ROTATION SPEED CLAMP COMMAND		ROTATION SPEED CLAMP COMMAND	
	In G97	SPINDLE ROTATION SPEED COMMAND		ROTATION SPEED CLAMP COMMAND	
Operation	**In G96**	ROTATION SPEED CLAMP EXECUTION		ROTATION SPEED CLAMP EXECUTION	
	In G97	NO ROTATION SPEED CLAMP		ROTATION SPEED CLAMP EXECUTION	NO ROTATION SPEED CLAMP

Spindle clamp speed command can be issued to the nth spindle or the 1st spindle.
Use the spindle selection command G code (G43.1/G44.1) to set to which spindle the command is to be issued.
Use the base specification parameter "#1199 Sselect" to set to which spindle the command is to be issued at the initial state.

(Note2) The address Q following the G92 command is handled as the spindle speed clamp command regardless of the constant surface mode.

(3) The command value of the spindle clamp speed will be cleared by modal reset (reset 2 or reset & rewind).
Note that the modal is retained if the parameter "#1210 RstGmd / bit19" is ON.
When the power is turned ON, the setting will be cleared to 0.

Precautions

(1) Once the maximum clamp speed and the minimum clamp speed are set using the spindle clamp speed setting (G92 S__ Q __), the maximum speed clamp will not be cancelled even if the command "G92 S0" is issued. Even when G92 S0 is commanded, the value of Q__ is kept enabled and Q__ is greater than S0.

(2) Note that if the spindle clamp speed setting (G92 S__ Q__) is not commanded, the speed may increase to the machine's maximum specified speed that is set by the parameter.
Especially when the constant surface speed control (G96 S__) is commanded, command the spindle clamp speed setting as well as the spindle maximum rotation speed. As the tool moves closer to the spindle center, the spindle rotation speed will increase and may exceed the allowable speed of the workpiece or chuck, etc.

⚠ WARNING

1. The spindle clamp speed setting command is a modal command, but make sure to confirm that the G and F modal and coordinate values are appropriate if the operation is started from a block in the middle of the program. If there are coordinate system shift commands or M, S, T and B commands before the block set as the start position, carry out the required commands using the MDI, etc. If the program is run from the set block without carrying out these operations, the machine interference may occur or the machine may operate at an unexpected speed.

10.4 Spindle/C Axis Control

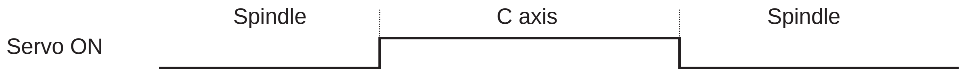
Function and purpose

This function enables one spindle to also be used as a C axis (rotation axis) by an external signal.

Detailed description

Spindle/C axis changeover

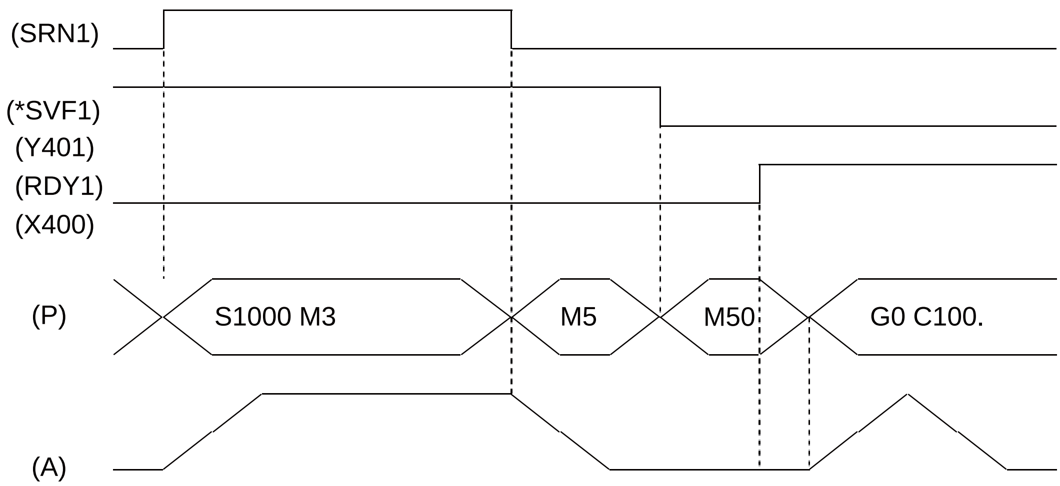
Changeover between the spindle and C axis is done by the C axis Servo ON signal.



At servo OFF ...... Spindle (C axis control not possible)
At servo ON ...... C axis (spindle control not possible)

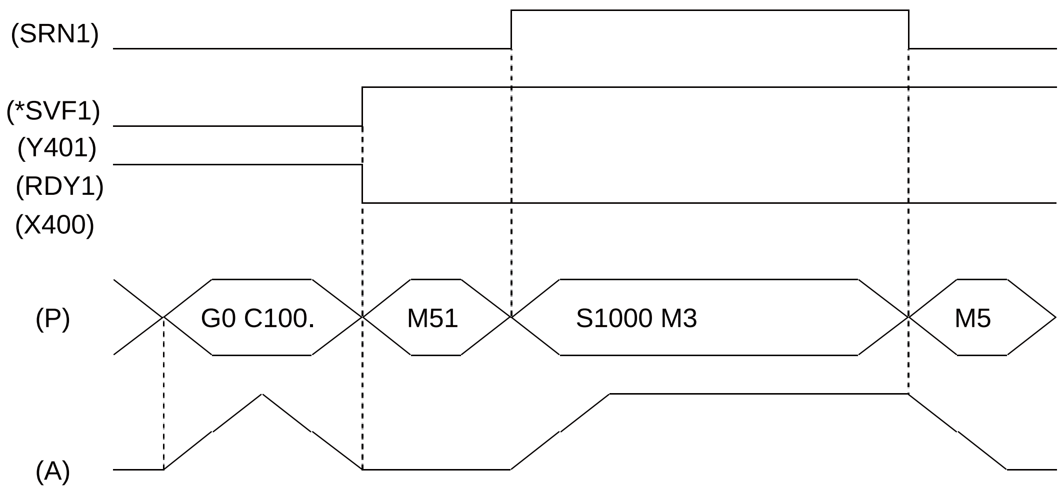
C axis potion data
The NC's internal C axis position data is updated even for the spindle rotation during spindle control.
The C axis coordinate value counter is held during spindle control, and is updated according to the amount moved during spindle control when the C axis servo READY is turned ON. (The C axis position at servo ON may differ from the position just before the previous servo OFF.)

Changeover timing chart example



Switch from spindle to C axis

(SRN1)	Spindle forward run start	(*SVF1)	Servo OFF(B contact)
(RDY1)	Servo ready	(P)	Program command
(A)	Spindle position shift amount		



Switch from C axis to spindle

(SRN1)	Spindle forward run start	(*SVF1)	Servo OFF(B contact)
(RDY1)	Servo ready	(P)	Program command
(A)	Spindle position shift amount		

(Note) M codes in the above figures indicate;
- M3 : Spindle forward run
- M5 : Spindle stop
- M50: C axis servo ON
- M51: C axis servo OFF

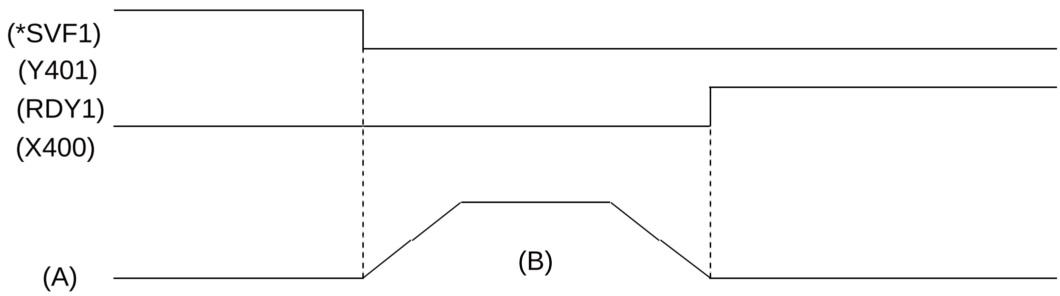
They formulate sequence programs.

The operation of zero point return

The operation of the zero point return when switching from the spindle to the C axis can be selected by the spindle specification parameter "#3106 zrn_typ/bit8" either from zero point return or deceleration stop.
If the first command to the spindle after turning the power ON is to switch to the C axis, Z-phase detection will be carried out before returning to the zero point in order to establish the coordinate.

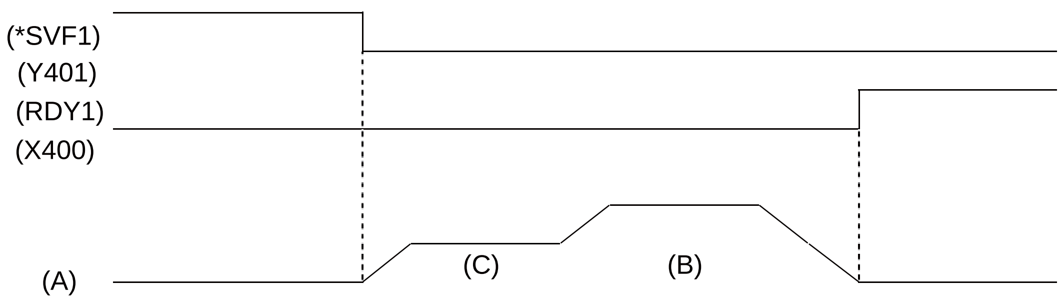
Zero point return type

In case of the zero point return type ("#3106 zrn_typ/bit8" is "0"), the zero point return is executed when switching from the spindle to the C axis by the C axis servo ON to establish the zero point.



The operation of zero point return type (when Z-phase is detected)

(*SVF1)	Servo OFF (B contact)	(RDY1)	Servo ready
(A)	Spindle position shift amount	(B)	Zero point return



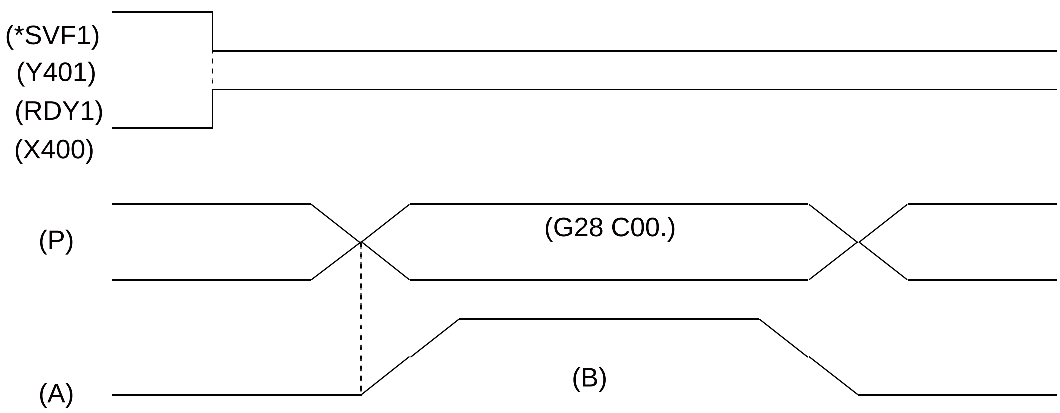
The operation of zero point return type (when Z-phase is not detected)

(*SVF1)	Servo OFF (B contact)	(RDY1)	Servo ready
(A)	Spindle position shift amount	(B)	Zero point return
(C)	Z-phase detection		

Deceleration stop type

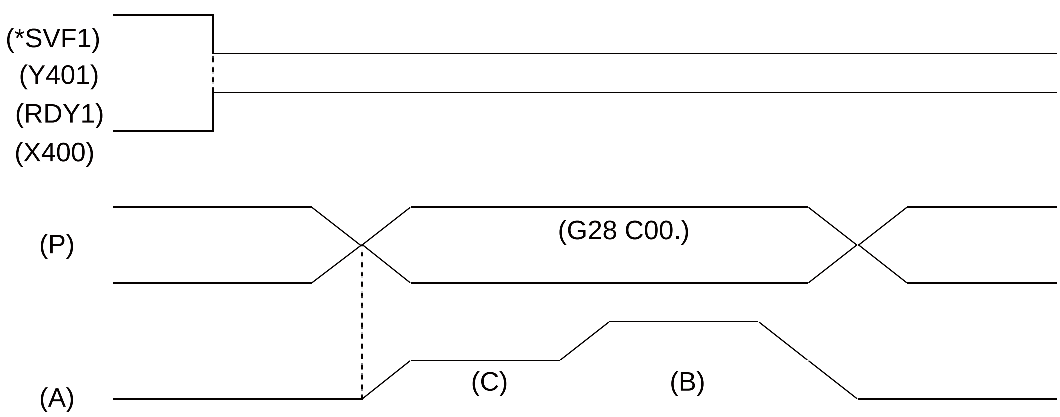
In case of the deceleration stop type ("#3106 zrn_typ/bit8" is "1"), C axis servo ON will only switch the spinlde to the C axis and will not establish the zero point.
A coordinate must be established to carry out an automatic operation. Use the base specification parameter "#1226 aux10/bit3" to select whether to insert an automatic zero point return operation before issuing a movement command.
Manual operation is valid even when the zero point is not established.

(1) When not inserting a zero point return
When "#1226 aux10/bit8" is "0", the zero point return is executed when commanded. The zero point return for the C axis is not established right after switching to the C axis. So the program error (P430) will occur if a command other than zero point return is commanded. (Even when the Z-phase is already detected, the zero point return must be executed as the C axis is not established. Once the zero point for the C axis is established, the zero point will continue to be established after switching to the C axis.)



The operation of zero point return of deceleration stop type without inserting a zero point return (when Z-phase is detected)

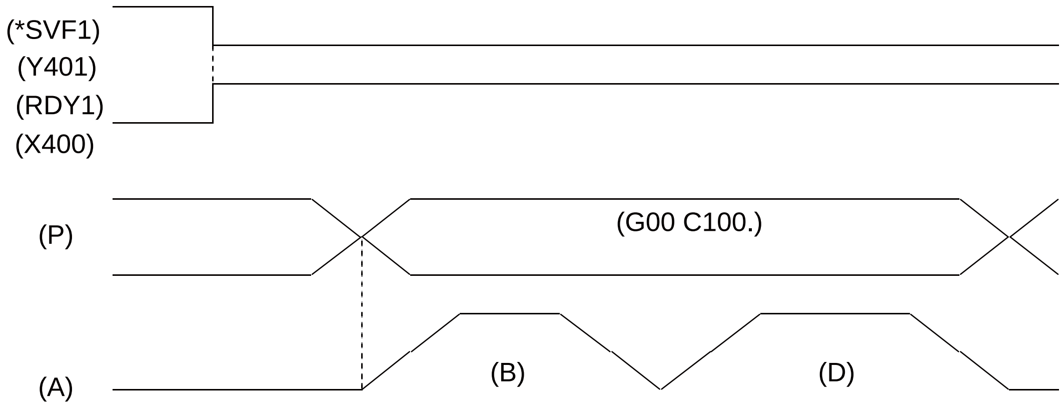
(*SVF1)	Servo OFF (B contact)	(RDY1)	Servo ready
(P)	Program command	(A)	Spindle position shift amount
(B)	Zero point return		



The operation of zero point return of deceleration stop type without inserting a zero point return (when Z-phase is not detected)

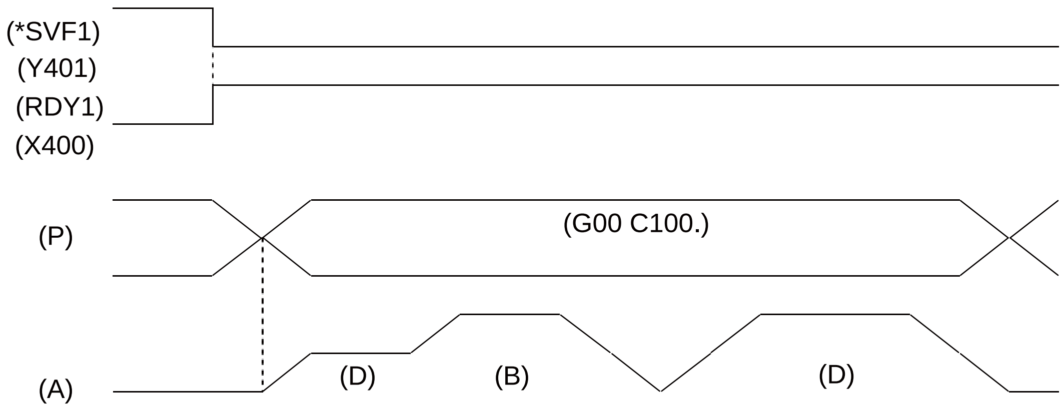
(*SVF1)	Servo OFF (B contact)	(RDY1)	Servo ready
(P)	Program command	(A)	Spindle position shift amount
(B)	Zero point return	(C)	Z-phase detection

(2) When inserting a zero point return automatically
When "#1226 aux10/bit3" is "1", the zero point return is executed automatically before moving if the movement command is commanded without establishing the zero point.
However, the zero point return is not inserted if the movement command is commanded while the zero point is established.



The operation of zero point return of deceleration stop type when inserting a zero point return (when Z-phase is detected)

(*SVF1)	Servo OFF (B contact)	(RDY1)	Servo ready
(P)	Program command	(A)	Spindle position shift amount
(B)	Zero point return	(D)	Positioning



The operation of zero point return of deceleration stop type when inserting a zero point return (when Z-phase is not detected)

(*SVF1)	Servo OFF (B contact)	(RDY1)	Servo ready
(P)	Program command	(A)	Spindle position shift amount
(B)	Zero point return	(C)	Z-phase detection
(D)	Positioning		

The operation when there is a discrepancy between units

When the setting unit for the part system to use the spindle and C-axis "#1003 iunit" differs from the spindle unit "#3035 spunit", the error "Y51 Spindle/C axis unit illegal 0202" will appear and the interlock state will be applied.
However, be aware that the zero point return by the servo ON will be executed even if the error "Y51 Spindle/C axis unit illegal 0202" appears when the spindle/C axis is the zero point return type.
When the unit is not set (blank is displayed), it will be handled as the standard setting value "B".

10.5 Spindle Synchronization

Function and purpose

In a machine having two or more spindles, this function controls the rotation speed and phase of one spindle (basic spindle) in synchronization with the rotation of the other spindle (synchronous spindle).
The function is used when the rotation speed of the two spindles must be matched, for example, if a workpiece grasped by the first spindle is to be grasped by a second spindle, or if the spindle rotation speed has to be changed when one workpiece is grasped by both the first and second spindles.
There are two types of spindle synchronization: Spindle synchronization l and Spindle synchronization ll.

The spindle synchronization control l

The designation of the synchronous spindle and start/stop of the synchronization are executed by commanding G codes in the machining program.

The spindle synchronization function ll

The selections of the synchronized spindle and synchronization start, etc., are all designated from the PLC. Refer to the instruction manual issued by the machine tool builder for details.

Common setting for the spindle synchronization control l and ll

When the spindle synchronization control is carried out, the followings must be set.

- Chuck close
- Error temporary cancel
- Phase monitor
- Multi-speed acceleration/deceleration

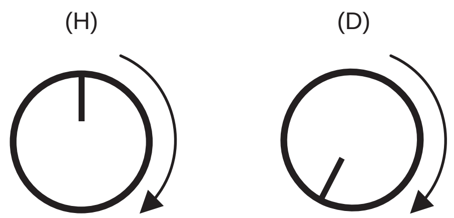
For details, refer to the "Precautions for Using Spindle Synchronization Control".

10.5.1 Spindle Synchronization Control I ; G114.1

Function and purpose

There are two types of spindle synchronization mode: The rotation synchronization mode and the phase synchronization mode

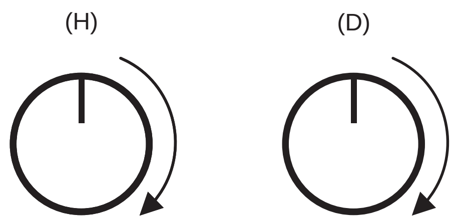
Rotation synchronization mode: Rotation speed of the basic spindle and synchronous spindle is controlled to be the same.



(H) Basic spindle
(D) Synchronous spindle

The rotation speed is synchronized.
The phases (Z-phase) are not aligned.

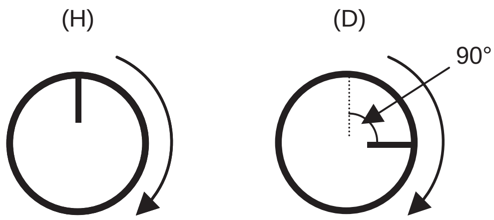
Phase synchronization mode: The rotation speed of the basic spindle and the synchronous spindle is controlled to be the same while their phases (Z phase) are aligned.
(Example 1) Phase synchronization with a phase error of "0"



(H) Basic spindle
(D) Synchronous spindle

The rotation speed is synchronized.
The phases (Z-phase) are aligned.

(Example 2) Phase synchronization with a phase error of "90°"



(H) Basic spindle
(D) Synchronous spindle

The rotation speed is synchronized.
The phases (Z-phase) are aligned.

Spindle Synchronization Control I designates a synchronous spindle and starts/ends synchronization by a G command in a machining program.

Command format

G114.1 H__ D__ R__ A__ ; ... Spindle synchronization control ON

H	Basic spindle selection
D	Synchronous spindle selection
R	Synchronous spindle phase shift amount
A	Spindle synchronization acceleration/deceleration time constant

G113 ; ... Spindle synchronization control cancel

Spindle synchronization control ON (G114.1) command designates the basic spindle and synchronous spindle, and synchronizes the two designated spindles. By commanding the synchronous spindle phase shift amount, the phases of the basic spindle and synchronous spindle can be aligned.
Spindle synchronization cancel (G113) cancels the synchronous state of the two spindles rotating in synchronization with the spindle synchronization command.

Address	Meaning of address	Command range (unit)	Remarks
H	Basic spindle selection Select the No. of the spindle to be used as the basic spindle from the two spindles.	1 to 7 1: 1st spindle 2: 2nd spindle 3: 3rd spindle 4: 4th spindle 5: 5th spindle 6: 6th spindle 7: 7th spindle	- A program error (P35) will occur if a value exceeding the command range or spindle No. without specifications is commanded. - A program error (P33) will occur if there is no command. - A program error (P610) will occur if a spindle not serially connected is commanded.
D	Synchronous spindle selection Select the No. of the spindle to be synchronized with the basic spindle from the two spindles.	1 to 7 or -1 to -7 1: 1st spindle 2: 2nd spindle 3: 3rd spindle 4: 4th spindle 5: 5th spindle 6: 6th spindle 7: 7th spindle	- A program error (P35) will occur if a value exceeding the command range or spindle No. without specifications is commanded. - A program error (P33) will occur if there is no command. - A program error (P33) will occur if the same spindle as that commanded for the basic spindle selection is designated. - The rotation direction of the synchronous spindle in respect to the basic spindle is commanded with the D sign. - A program error (P610) will occur if a spindle not serially connected is commanded.
R	Synchronous spindle phase shift amount Command the shift amount from the Z-phase point (one rotation signal) of the synchronous spindle.	0 to 359.999 (°) or 0 to 35999 (°* 10^{-3})	- A program error (P35) will occur if a value exceeding the command range is commanded. - The commanded shift amount is effective in the clockwise direction of the basic spindle. - The commanded shift amount's minimum resolution is as follows: For semi-closed (Only gear ratio 1:1) 360/4096 (°) For full closed (360/4096) * K (°) K: Spindle and encoder gear ratio - If there is no R command, the phases will not be aligned.
A	Spindle synchronization acceleration/deceleration time constant Command the acceleration/deceleration time constant for when the spindle synchronous command rotation speed changes. (Command this to accelerate or decelerate at a speed slower than the time constant set in the parameters.)	0.001 to 9.999(s) or 1 to 9999 (ms)	- A program error (P35) will occur if a value exceeding the command range is commanded. - If the commanded value is smaller than the acceleration/deceleration time constant set with the parameters, the value set in the parameters will be applied.

Detailed description

Rotation speed and rotation direction

(1) The rotation speed and rotation direction of the basic spindle and synchronous spindle during spindle synchronization control are the rotation speed and rotation direction commanded for the basic spindle. Note that the rotation direction of the synchronous spindle can be reversed from the basic spindle through the program.

(2) The basic spindle's rotation speed and rotation direction can be changed during spindle synchronization control.

(3) The synchronous spindle's rotation command is also valid during spindle synchronization control.
When spindle synchronization control is commanded, if neither a forward run command nor reverse run command is commanded for the synchronous spindle, the synchronization standby state will be entered without starting the synchronous spindle's rotation. If the forward run command or reverse run command is input in this state, the synchronous spindle will start rotation. The synchronous spindle's rotation direction will follow the direction commanded in the program.
If spindle stop is commanded for the synchronous spindle during spindle synchronization control (when both the forward run and reverse run commands are turned OFF), the synchronous spindle rotation will stop.

(4) The rotation speed command (S command) and constant surface speed control are invalid for the synchronous spindle during spindle synchronization control. Note that the modal is updated, so these will be validated when the spindle synchronization is canceled.

(5) The constant surface speed can be controlled by issuing a command to the basic spindle even during spindle synchronization control.

Rotation synchronization

(1) When rotation synchronization control (command with no R address) is commanded with the G114.1 command, the synchronous spindle rotating at an arbitrary rotation speed will accelerate or decelerate to the rotation speed commanded beforehand for the basic spindle, and will enter the rotation synchronization state.

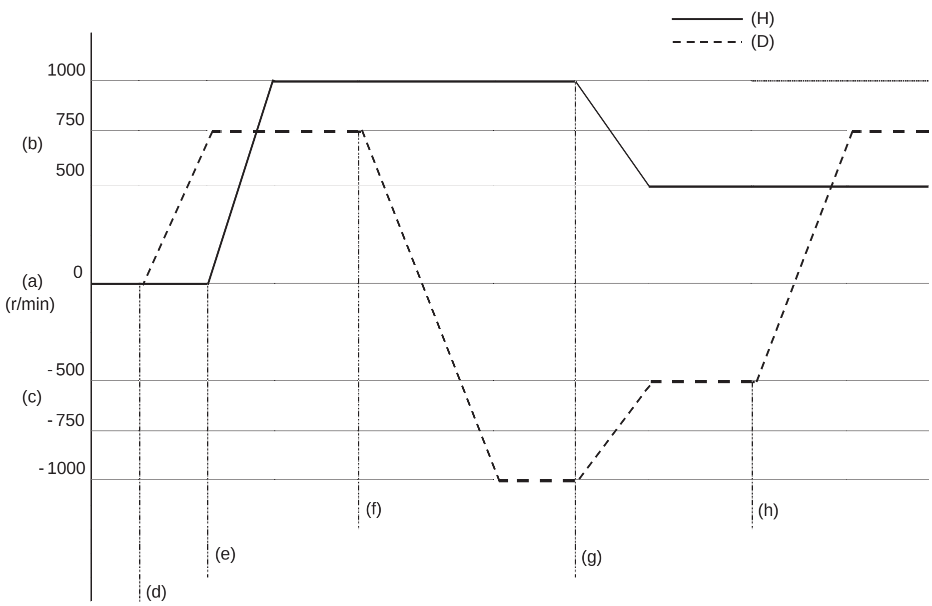
(2) If the basic spindle's commanded rotation speed is changed during the rotation synchronization state, acceleration/deceleration will be carried out while maintaining the synchronization state following the spindle acceleration/deceleration time constants set in the parameters, and the commanded rotation speed will be achieved.

(3) In the rotation synchronization state, the basic spindle can be controlled to the constant surface speed even when two spindles are grasping one workpiece.

(4) Operation will take place in the following manner.

M23 S2=750 ;	...... Forward rotate 2nd spindle (synchronous spindle) at 750 r/min (speed command)
:	
M03 S1=1000 ;	...... Forward rotate 1st spindle (basic spindle) at 1000 r/min (speed command)
:	
G114.1 H1 D-2 ;	...... Synchronize 2nd spindle (synchronous spindle) to 1st spindle (basic spindle) with reverse run.
:	
S1=500 ;	...... Change 1st spindle (basic spindle) rotation speed to 500 r/min.
:	
G113 ;	..... Cancel spindle synchronization

<Operation>



(H) Basic spindle
(D) Synchronous spindle
(a) Rotation speed
(b) Forward run
(c) Reverse run
(d) 2nd spindle (synchronous spindle) forward run
(e) 1st spindle (basic spindle) forward run
(f) 2nd spindle (synchronous spindle) reverse run synchronization
(g) 1st spindle (basic spindle) rotation speed change
(h) Spindle synchronization cancel

Phase synchronization

(1) When phase synchronization (command with R address) is commanded with the G114.1 command, the synchronous spindle rotating at an arbitrary rotation speed will accelerate or decelerate to the rotation speed commanded beforehand for the basic spindle, and will enter the rotation synchronization state. Then, the phase is aligned so that the rotation phase commanded with the R address is reached, and the phase synchronization state is entered.

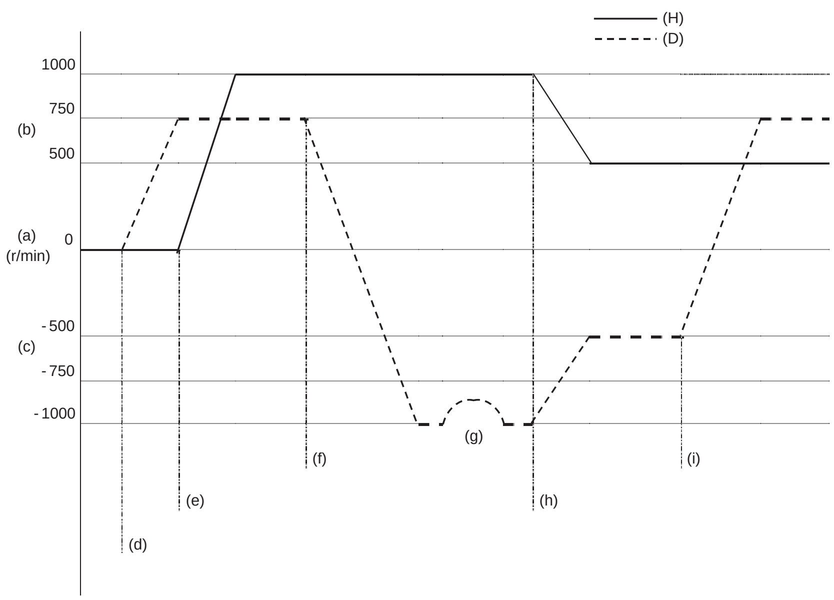
(2) If the basic spindle's commanded rotation speed is changed during the phase synchronization state, acceleration/deceleration will be carried out while maintaining the synchronization state following the spindle acceleration/deceleration time constants set in the parameters, and the commanded rotation speed will be achieved.

(3) In the phase synchronization state, the basic spindle can be controlled to the constant surface speed even when two spindles are grasping one workpiece.

(4) Operation will take place in the following manner.

M23 S2=750 ;	...... Forward rotate 2nd spindle (synchronous spindle) at 750 r/min (speed command)
:	
M03 S1=1000 ;	...... Forward rotate 1st spindle (basic spindle) at 1000 r/min (speed command)
:	
G114.1 H1 D-2 Rxx ; :	...... Synchronize 2nd spindle (synchronous spindle) to 1st spindle (basic spindle) with reverse run. Shift phase of synchronous spindle by R command value.
:	
S1=500 ;	...... Change 1st spindle (basic spindle) rotation speed to 500 r/min.
:	
G113 ;	..... Cancel spindle synchronization

<Operation>



(H) Basic spindle	(D) Synchronous spindle	
(a) Rotation speed	(b) Forward run	(c) Reverse run
(d) 2nd spindle (synchronous spindle) forward run	(e) 1st spindle (basic spindle) forward run	
(f) 2nd spindle (synchronous spindle) reverse run synchronization	(g) Phase alignment	
(h) 1st spindle (basic spindle) rotation speed change	(i) Spindle synchronization cancel	

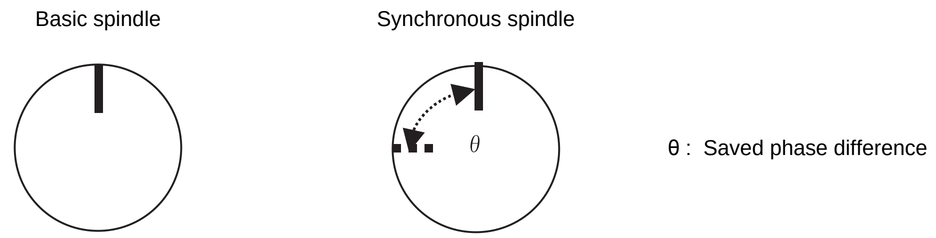
(Note 1) When "#3130 syn_spec/bit1" = "0", the phase synchronization is conducted by the step synchronization method without acceleration/deceleration. And when "#3130 syn_spec/bit1" = "1", it is conducted by the multi-step acceleration/deceleration method (mentioned later).

Spindle synchronization phase shift amount calculation function

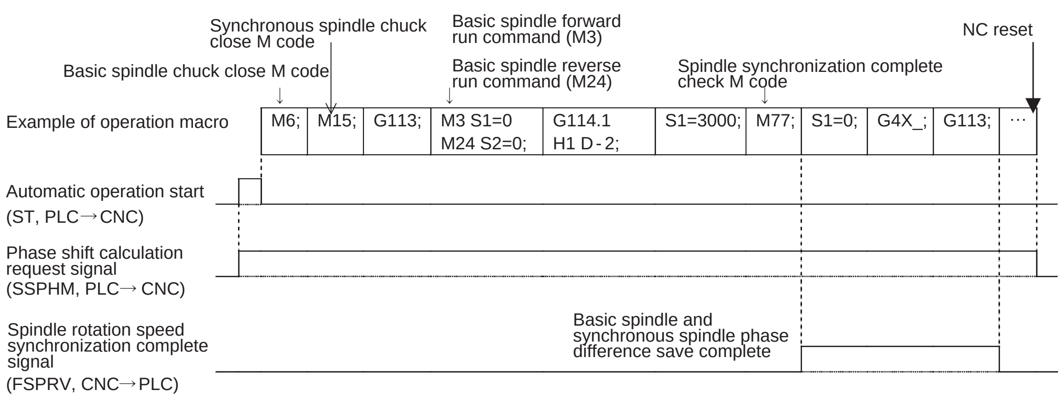
The spindle phase shift amount calculation function obtains and saves the phase difference of the basic spindle and synchronous spindle by turning the PLC signal ON when the phase synchronization command is executed. When the phase is positioned to the automatically saved phase difference before executing the phase synchronization control command, phases can be aligned easier when re-grasping profile materials.

[Saving the basic spindle and synchronous spindle phase difference]

(1) Set a profile material in the main spindle (basic spindle).
(2) Set the profile material in the rear spindle.
(3) Turn the phase shift calculation request signal (SSPHM) ON.
(4) Input a rotation command, with 0 speed, for the main spindle (basic spindle) and rear spindle (synchronous spindle).
<Example> M3 S1=0 M24 S2 = 0;
(5) Execute the rotation synchronization signal (with no R address command).
<Example> G114.1 H1 D-2;
(6) Rotate the main spindle at the speed actually used when re-grasping.
<Example> S1 = 3000;
(7) Check that the phase difference has been saved by looking at the spindle speed synchronization complete signal.
(8) Stop both spindles.
(9) Turn the phase shift calculation request signal OFF.

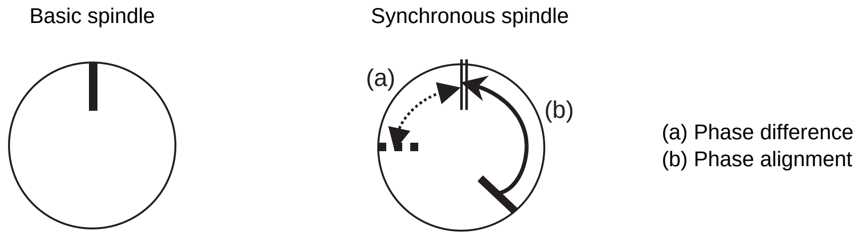


<Example of operation>

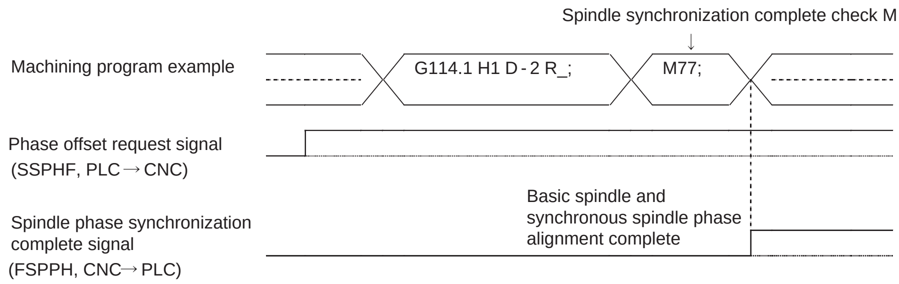


[Automatic phase alignment of basic spindle and synchronous spindle]

(1) Turn the phase offset request signal ON.
(2) Issue the phase synchronization command (with R command).
<Example> G114.1 H1 D-2 R0;
(3) The phase is aligned by offsetting the phase synchronization command by the phase difference obtained with the spindle synchronization phase shift calculation function. The state in which the synchronous spindle phase shift amount designation R value is 0 is the same as the reference state (state obtained with phase shift calculation request signal).



<Example of operation>

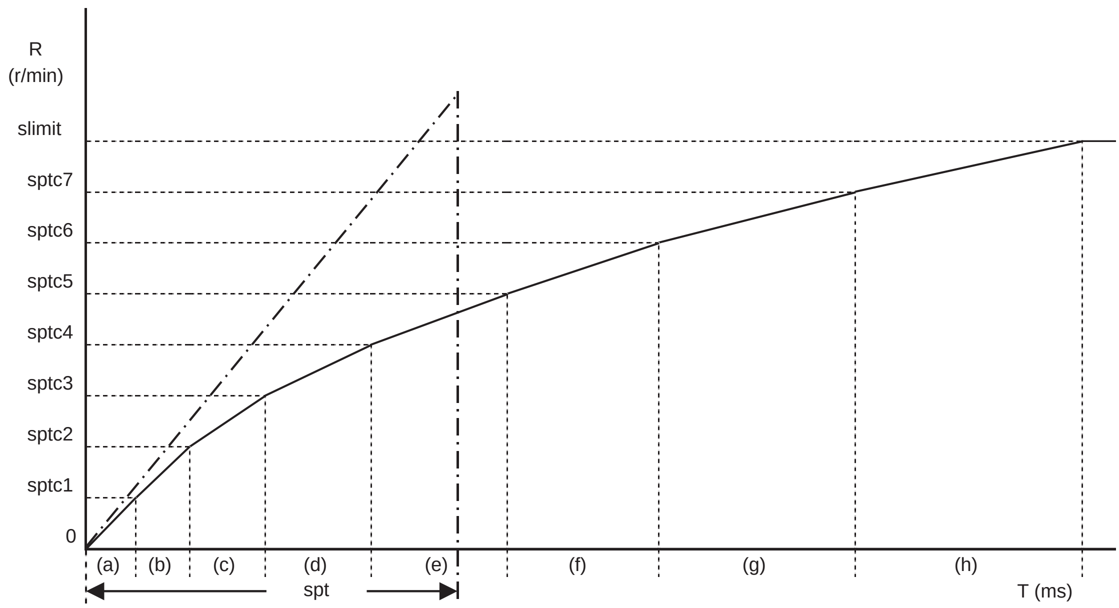


Multi-step acceleration/deceleration

Acceleration/deceleration time constants for up to eight steps can be selected according to the spindle rotation speed for the acceleration/deceleration during spindle synchronization.
The acceleration/deceleration in each step is as follows.

Time required from minimum rotation speed to maximum rotation speed in each step
= [Time constant without multi-step acceleration/deceleration] * [magnification of time constant in each step] * [Rate of rotation speed width in each step respect to rotation speed width up to limit rotation speed]

Time required to rotate to sptc1 set rotation speed from stopped state (a)
= spt (or A command when G114.1 is commanded) * sptc1/slimit
Time required to reach sptc2 set rotation speed from sptc1 (b)
= spt (or A command when G114.1 is commanded) * spdiv1 * (sptc2 - sptc1)/slimit
Time required to reach sptc3 set rotation speed from sptc2 (c)
= spt (or A command when G114.1 is commanded) * spdiv2 * (sptc3 - sptc2)/slimit
Time required to reach sptc4 set rotation speed from sptc3 (d)
= spt (or A command when G114.1 is commanded) * spdiv3 * (sptc4 - sptc3)/slimit
Time required to reach sptc5 set rotation speed from sptc4 (e)
= spt (or A command when G114.1 is commanded) * spdiv4 * (sptc5 - sptc4)/slimit
Time required to reach sptc6 set rotation speed from sptc5 (f)
= spt (or A command when G114.1 is commanded) * spdiv5 * (sptc6 - sptc5)/slimit
Time required to reach sptc7 set rotation speed from sptc6 (g)
= spt (or A command when G114.1 is commanded) * spdiv6 * (sptc7 - sptc6)/slimit
Time required to reach sptc8 set rotation speed from sptc7 (h)
= spt (or A command when G114.1 is commanded) * spdiv7 * (slimit - sptc7)/slimit



R: Rotation speed   T: Time

To decrease the number of acceleration/deceleration steps during spindle synchronization, set one of the following for the unnecessary step.
Magnification for time constant changeover speed (spdiv7 to spdiv1) = 0 (or 1)
Spindle synchronous multi-step acceleration/deceleration changeover speed (sptc7 to sptc1) = Limit rotation speed (slimit) or higher

Precautions

(1) To carry out the spindle synchronization, it is required to command spindle rotation for both basic spindle and synchronous spindle. Note that the rotating direction of the synchronous spindle follows the rotating direction of the basic spindle and rotating direction designation by "D" address.
(2) The spindle rotating with spindle synchronization control will stop when emergency stop is applied.
(3) The rotation speed clamp during spindle synchronization control mode will follow the smaller clamp value set for the basic spindle or synchronous spindle.
(4) Orientation of the basic spindle and synchronous spindle is not possible during the spindle synchronization control mode. To carry out orientation, cancel the spindle synchronization control mode first.
(5) The rotation speed command (S command) is invalid for the synchronous spindle during the spindle synchronization control mode. Note that the modal will be updated, so this command will be validated when spindle synchronization control is canceled.
(6) The constant surface speed control is invalid for the synchronous spindle during the spindle synchronization control mode. Note that the modal will be updated, so the constant surface speed control will be validated when spindle synchronization control is canceled.
(7) The rotation speed command (S command) and constant surface speed control for the synchronous spindle will be validated when spindle synchronization control is canceled. Thus, attention must be paid because the synchronous spindle may start different operations when the control is canceled.
(8) Be aware that the phase shift amount will not be obtained correctly if the phase synchronization command is executed with the phase shift calculation request signal ON although the phase difference is not obtained by the signal.
(9) The spindle Z-phase encoder position parameter (sppst) is invalid (ignored) when using the spindle synchronous phase shift amount calculation function.
This parameter (sppst) is valid when the phase offset request signal is OFF.
(10) If the phase synchronization command (command with R address) is issued while the phase shift calculation request signal is ON, the error "M01 OPERATION ERROR 1106" will occur.
(11) Turn the phase shift calculation request signal ON when the basic spindle and synchronous spindle are both stopped. If the phase shift calculation request signal is turned ON while either of the spindles is rotating, the error "M01 OPERATION ERROR 1106" will occur.
(12) If the phase synchronization command R0 (<Ex.> G114.1 H1 D-2 R0) is commanded while the phase offset request signal is ON, the basic spindle and synchronous spindle phases will be aligned to the phase error of the basic spindle and synchronous spindle saved in the NC memory.
(13) If a value other than the phase synchronization command R0 (<Ex.> G114.1 H1 D-2 R100) is commanded while the phase offset request signal is ON, the phase error obtained by adding the value commanded with the R address command to the phase error of the basic spindle and synchronous spindle saved in the NC memory will be used to align the basic spindle and synchronous spindle.
(14) The phase offset request signal will be ignored when the phase shift calculation request signal (SSPHM) is ON.
(15) The phase error of the basic spindle and synchronous spindle saved in the NC is valid only when the phase shift calculation signal is ON and for the combination of the basic spindle selection (H_) and synchronous spindle (D_) commanded with the rotation synchronization command (no R address). For example, if the basic spindle and synchronous spindle phase error are saved as "G114.1 H1 D-2 ;", the saved phase error will be valid only when the phase offset request signal is ON and "G114.1 H1 D_2 R*** ;" is commanded. If "G114.1 H2 D-1 R*** ;" is commanded in this case, the phase shift amount will not be calculated correctly.

(16) The basic spindle and synchronous spindle phase error saved in the NC is held until the next spindle synchronous phase shift calculation (rotation synchronization command is completed with phase shift calculation request signal ON).
(17) Synchronous tapping can not be used during spindle synchronization control mode.
(18) When the spindle synchronization commands are being issued with the PLC I/F method (#1300 ext36/ bit7 OFF), a program error (P610) will occur if the spindle synchronization control is commanded with G114.1/G113.
(19) Chuck close must always be set. If not, machine may suffer an excessive load or an alarm may occur.

Cautions on programming

(1) To enter the rotation synchronization mode while the basic spindle and synchronous spindle are chucking the same workpiece, turn the basic spindle and synchronous spindle rotation commands ON before turning the spindle synchronization control mode ON.

$1 (1st part system)		$2 (2nd part system)	
:		:	
M6;	1st spindle chuck close	:	
:		M25 S2=0;	2nd spindle stops at S=0
:		:	
!2;		!1;	Waiting-and-simultaneous operation between part systems
M5 S1=0;	1st spindle stops at S=0	M15;	2nd spindle chuck close
:		M24;	2nd spindle rotation command ON
M3;	1st spindle rotation command ON	:	
!2;		!1;	Waiting-and-simultaneous operation between part systems
:		G114.1 H1 D-2 ;	Rotation synchronization mode ON
:		:	
S1=1500;	Synchronous rotation at S=1500	:	
:		:	
S1=0;	Both spindles stop		
G113	Synchronization mode cancel		

(2) To chuck the same workpiece with the basic spindle and synchronous spindle in the phase synchronization mode, align the phases before chucking.

$1 (1st part system)		$2 (2nd part system)	
:		:	
M6;	1st spindle chuck close	:	
:		:	
M3 S1=1500;	1st spindle rotation command ON	:	
:		G114.1 H1 D-2 R0;	Phase synchronization mode ON
:		:	
:		M24;	2nd spindle rotation command ON
:		:	
:		M15;	2nd spindle chuck close (Note 1)
:		:	

(Note 1) Close the chuck after confirming that the spindle phase synchronization complete signal (FSPPH) has turned ON (phase alignment complete).

⚠ CAUTION

1. Do not make the synchronous spindle rotation command OFF with one workpiece chucked by the basic spindle and synchronous spindle during the spindle synchronization control mode. Failure to observe this may cause the synchronous spindle stop, and hazardous situation.

10.5.2 Spindle Synchronization Control II

Function and purpose

With the spindle synchronous control II, selection of the spindles and synchronization start, etc., are all designated from the PLC.

Detailed description

Basic spindle and synchronous spindle selection

Select the basic spindle and synchronous spindle for synchronous control from the PLC.

Device No.	Signal name	Abbrev.	Explanation
R2357	Spindle synchronization Basic spindle selection	-	Select a serially connected spindle to be controlled as the basic spindle. (0: 1st spindle) 1: 1st spindle 2: 2nd spindle 3: 3rd spindle 4: 4th spindle 5: 5th spindle 6: 6th spindle 7: 7th spindle (Note1) Spindle synchronization will not take place if a spindle not connected in serial is selected. (Note2) If "0" is designated, the 1st spindle will be controlled as the basic spindle.
R2358	Spindle synchronization Synchronous spindle selection	-	Select a serially connected spindle to be controlled as the synchronous spindle. (0: 2nd spindle) 1: 1st spindle 2: 2nd spindle 3: 3rd spindle 4: 4th spindle 5: 5th spindle 6: 6th spindle 7: 7th spindle (Note3) Spindle synchronization control will not take place if a spindle not connected in serial is selected or if the same spindle as the basic spindle is selected. (Note4) If "0" is designated, the 2nd spindle will be controlled as the synchronous spindle.

Starting spindle synchronization

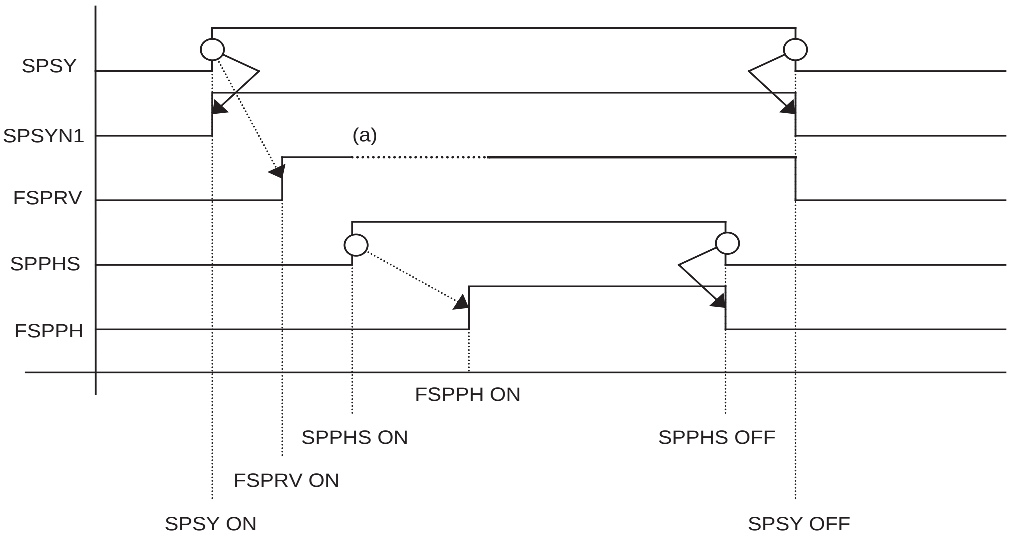
The spindle synchronization control mode is entered by inputting the spindle synchronization signal (SPSY).
The synchronous spindle will be controlled in synchronization with the rotation speed commanded for the basic spindle during the spindle synchronization control mode.
When the difference of the basic spindle and synchronous spindle rotation speeds reaches the spindle synchronization rotation speed reach level setting value (#3050 sprlv), the spindle rotation speed synchronization complete signal (FSPRV) will be output.
The synchronous spindle's rotation direction is designated with the spindle synchronization rotation direction designation as the same as the basic spindle or the reverse direction.

Device No.	Signal name	Abbrev.	Explanation
Y332	Spindle synchronization	SPSY	The spindle synchronization control mode is entered when this signal turns ON.
X32A	In spindle synchronization	SPSYN1	This notifies that the mode is the spindle synchronization.
X32B	Spindle rotation speed synchronization completion	FSPRV	This turns ON when the difference of the basic spindle and synchronous spindle rotation speeds reaches the spindle rotation speed reach level setting value during the spindle synchronization control mode. This turns OFF when the spindle synchronization control mode is canceled, or when an error exceeding the spindle rotation speed reach level setting value occurs during the spindle synchronization control mode.
Y334	Spindle synchronous rotation direction	SPSDR	Designate the basic spindle and synchronous spindle rotation directions for spindle synchronization control. 0: The synchronous spindle rotates in the same direction of the basic spindle. 1: The synchronous spindle rotates in the reverse direction of the basic spindle.

Spindle phase alignment

Spindle phase synchronization starts when the spindle phase synchronization control signal (SPPHS) is input during the spindle synchronization control mode.
The spindle phase synchronization complete signal is output when the spindle synchronization phase reach level setting value (#3051 spplv) is reached.
The synchronous spindle's phase shift amount can also be designated from the PLC.

Device No.	Signal name	Abbrev.	Explanation
Y333	Spindle phase synchronization	SPPHS	Spindle phase synchronization starts when this signal is turned ON during the spindle synchronization control mode. (Note 1) If this signal is turned ON in a mode other than the spindle synchronization control mode, it will be ignored.
X32C	Spindle phase synchronization completion	FSPPH	This signal is output when the spindle synchronization phase reach level is reached after starting spindle phase synchronization.
R2359	Spindle synchronization Phase shift amount	-	Designate the synchronous spindle's phase shift amount. Unit: 360°/4096



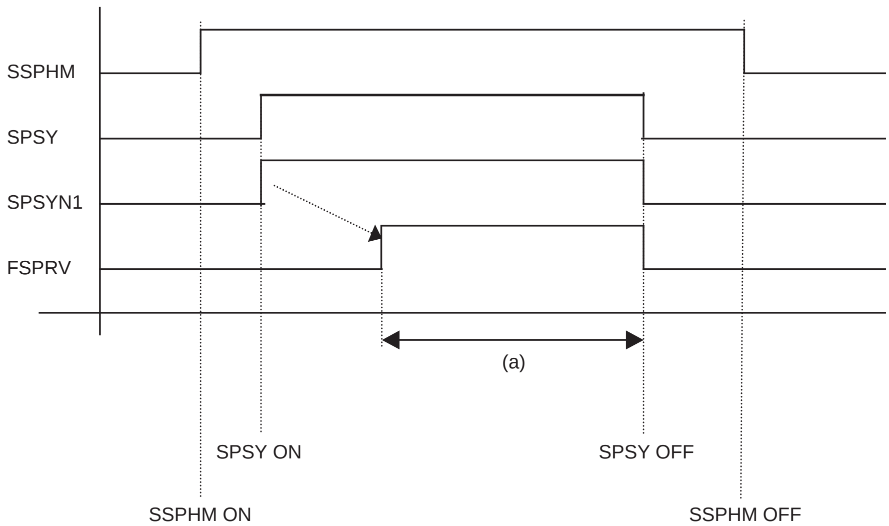
(a) Turns OFF temporarily to change the rotation speed during phase synchronization.

SPSY : Spindle synchronization
SPSYN1 : In spindle synchronization
FSPRV : Spindle rotation speed synchronization completion
SPPHS : Spindle phase synchronization
FSPPH : Spindle phase synchronization completion

Calculating the spindle synchronization phase shift amount and requesting phase offset

The spindle phase shift amount calculation function obtains and saves the phase difference of the basic spindle and synchronous spindle by turning the "phase shift calculation request" signal ON during spindle synchronization. When calculating the spindle phase shift, the synchronous spindle can be rotated with the handle, so the relation of the phases between the spindles can also be adjusted visually.
If the spindle phase synchronization control signal is input while the phase offset request signal (SSPHF) is ON, the phases will be aligned using the position shifted by the saved phase shift amount as a reference. This makes aligning of the phases easier when grasping the material that the shape of one end differs from the other end.

Device No.	Signal name	Abbrev.	Explanation
Y335	Phase shift calculation request	SSPHM	If spindle synchronization is carried out while this signal is ON, the phase difference of the basic spindle and synchronous spindle will be obtained and saved.
Y336	Phase offset request	SSPHF	If spindle phase synchronization is carried out while this signal is ON, the phase will be aligned using the position shifted by the saved phase shift amount as a basic position.
R55	Spindle synchronization phase error output	-	The delay of the synchronous spindle in respect to the basic spindle is output. Unit:360°/4096 (Note 1) If either the basic spindle or synchronous spindle has not passed through the Z phase, etc., and the phase cannot be calculated, -1 will be output. (Note 2) This data is output only while calculating the phase shift or during spindle phase synchronization.
R59	Spindle synchronization Phase offset data	-	The phase difference saved with phase shift calculation is output. Unit:360°/4096 (Note 3) This data is output only during spindle synchronization control.



(a) The phase difference in this interval is saved. (The synchronous spindle can be controlled with the handle.)

SSPHM : Phase shift calculation request
SPSY : Spindle synchronization
SPSYN1 : In spindle synchronization signal
FSPRV : Spindle rotation speed synchronization completion

(Note 1) The phases cannot be aligned while calculating the phase shift.
(Note 2) The synchronous spindle cannot be rotated with the handle when the manual operation mode is set to the handle mode.

Precautions and restrictions

(1) When carrying out spindle synchronization, a rotation command must be issued to both the basic spindle and synchronous spindle. The synchronous spindle's rotation direction will follow the basic spindle rotation direction and spindle synchronization rotation direction designation regardless of whether a forward or reverse run command is issued.
(2) The spindle synchronization control mode will be entered even if the spindle synchronization control signal is turned ON while the spindle rotation speed command is ON. However, synchronous control will not actually take place. Synchronous control will start after the rotation speed command has been issued to the basic spindle, and then the spindle synchronization complete signal will be output.
(3) The spindle rotating with spindle synchronization control will stop when emergency stop is applied.
(4) An operation error will occur if the spindle synchronization control signal is turned ON while the basic spindle and synchronous spindle designations are illegal.
(5) The rotation speed clamp during spindle synchronization control will follow the smaller clamp value set for the basic spindle or synchronous spindle.
(6) Orientation of the basic spindle and synchronous spindle is not possible during the spindle synchronization control mode. To carry out orientation, cancel the spindle synchronization control mode first.
(7) The rotation speed command (S command) is invalid for the synchronous spindle during the spindle synchronization control mode. Note that the modal will be updated, so this command will be validated when spindle synchronization control is canceled.
(8) The constant surface speed control is invalid for the synchronous spindle during the spindle synchronization control mode. Note that the modal will be updated, so the constant surface speed control will be validated when spindle synchronization control is canceled.
(9) The rotation speed command (S command) and constant surface speed control for the synchronous spindle will be validated when spindle synchronization control is canceled. Thus, attention must be paid because the synchronous spindle may start different operations when the control is canceled.
(10) Be aware that the phase shift amount will not be obtained correctly if the phase synchronization command is executed with the phase shift calculation request signal ON although the phase difference is not obtained by the signal.
(11) The spindle Z phase encoder position parameter (sppst) is invalid (ignored) when using the spindle synchronous phase shift amount calculation function.
This parameter (sppst) is valid when the phase offset request signal is OFF.
(12) If spindle phase synchronization is started while the phase shift calculation request signal is ON, the error "M01 OPERATION ERROR 1106" will occur.
(13) Turn the phase shift calculation request signal ON when the basic spindle and synchronous spindle are both stopped. If the phase shift calculation request signal is turned ON while either of the spindles is rotating, the error "M01 OPERATION ERROR 1106" will occur.
(14) The phase offset request signal will be ignored when the phase shift calculation request signal (SSPHM) is ON.
(15) "M01 OPERATION ERROR 1106" will occur when a spindle No. out of specifications is designated in the R registers to set the basic spindle and the synchronous spindle, or when the spindle synchronization control signal (SPSY) is turned ON with R resister value illegal.
(16) The phase shift amount saved in the NC is held until the next phase shift is calculated. (This value is saved even when the power is turned OFF.)
(17) Synchronous tapping can not be used during spindle synchronization control mode.
(18) Chuck close must always be set. If not, machine may suffer an excessive load or an alarm may occur.

10.5.3 Precautions for Using Spindle Synchronization Control

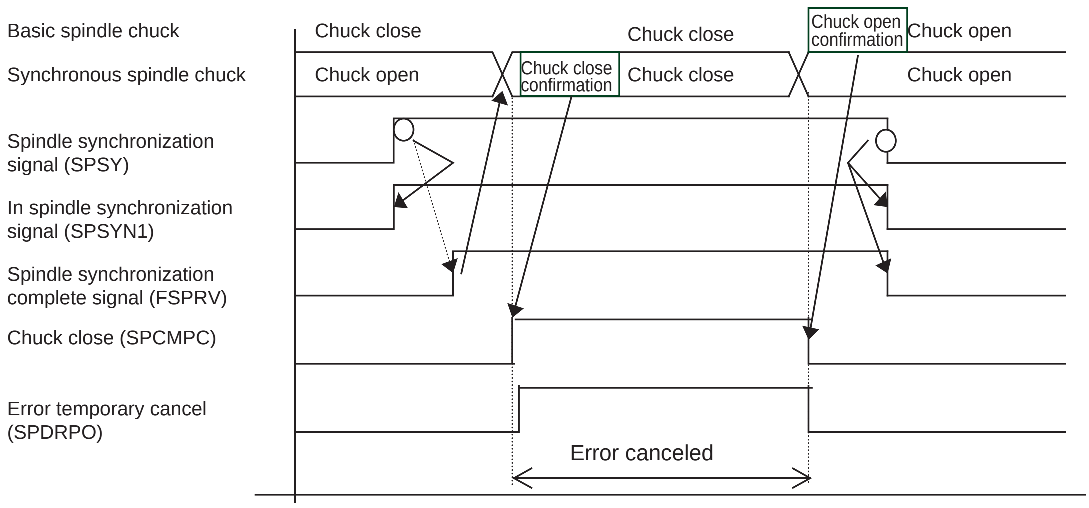
Precautions

Some PLC signals must be set when spindle synchronization control I or II is used. If these signals are not set, an excessive load or an alarm may occur. Refer to the instruction manual issued by the machine tool builder for details. In this section, each function and the signal are explained.

Chuck close signal

The synchronous spindle side carries out droop compensation while the chuck is opened, and aligns itself with the basic spindle. However, when the chuck is closed, the droop compensation is added, and the synchronization error with the base increases. Droop compensation is prevented with the chuck close signal and the position where the chuck is grasped is maintained with position compensation.

Device No.	Signal name	Abbrev.	Explanation
Y331	Chuck close	SPCMPC	This turns ON when the chuck of both spindles are closed. This signal is ON while the basic spindle and the synchronous spindle grasp the same workpiece.
X32D	Chuck close confirmation	SPCMP	This turns ON when the chuck close signal is received during the spindle synchronization control mode.



(Note 1) Use the error temporary cancel only when there is still an error between the spindle and synchronization with the chuck close signal.

Error temporary cancel function

When spindle synchronization is carried out while grasping the workpiece with the basic spindle and rotating, if the chuck is closed to grasp the workpiece with the synchronous spindle, the speed will fluctuate due to external factors and an error will occur. If spindle synchronization is continued without compensating this error, the workpiece will twist.
This torsion can be prevented by temporarily canceling this error.

Device No.	Signal name	Abbrev.	Explanation
Y337	Error temporary cancel	SPDRPO	The error is canceled when this signal is ON. When this signal turns ON, the gap between the basic spindle position and the synchronous spindle position is saved. When this signal is ON, the saved gap is canceled and spindle synchronization is carried out.

(Note 1) Even if the chuck close signal (SPCMPC) is OFF, the error will be canceled while this signal (SPDRPO) is ON.

(Note 2) Turn this signal ON after the both chucks of basic spindle side and synchronous spindle side are closed to grasp the workpiece.
Turn this signal OFF if even one chuck is opened.

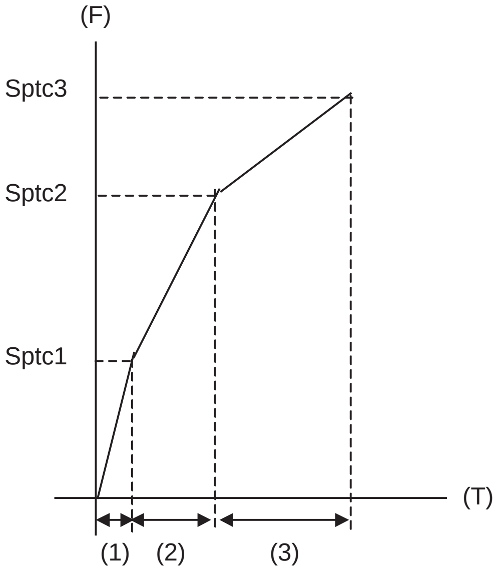
Phase error monitor

The phase error can be monitored during spindle phase synchronization.

Device No.	Signal name	Abbrev.	Explanation
R56	Spindle synchronization Phase error monitor	-	The phase error during spindle phase synchronization control is output as a pulse unit.
R57	Spindle synchronization Phase error monitor (lower limit value)	-	The lower limit value of the phase error during spindle phase synchronization control is output as a pulse unit.
R58	Spindle synchronization Phase error monitor (upper limit value)	-	The upper limit value of the phase error during spindle phase synchronization control is output as a pulse unit.

Multi-step acceleration/deceleration

Up to eight steps of acceleration/deceleration time constants for spindle synchronization can be selected according to the spindle rotation speed.



(F) Rotation speed
(T) Time

(1) Time required from stopped state to sptc1 setting rotation speed
spt * (sptc1/maximum rotation speed)

(2) Time required from sptc1 to sptc2 setting rotation speed
spt * ((sptc2-sptc1)/maximum rotation speed) * spdiv1

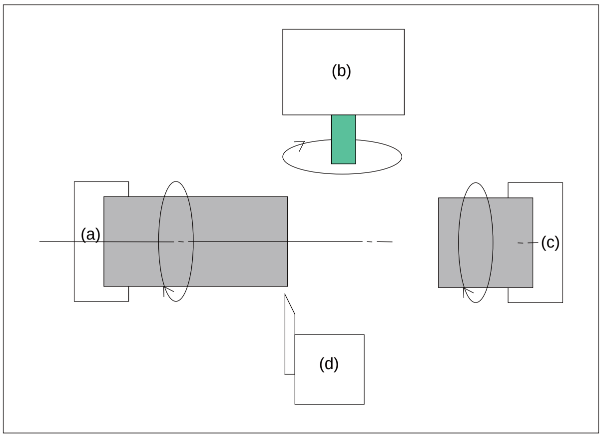
(3) Time required from sptc2 to sptc3 setting rotation speed
spt * ((sptc3-sptc2)/maximum rotation speed) * spdiv2

10.6 Multiple-spindle Control

10.6.1 Multiple spindle command ; S ○ =

Function and purpose

Spindle rotation command for up to 7 spindles is provided.
Although the S***** command is normally used to designate the spindle rotation speed, the Sn=***** command is also used for multiple spindle control.
S commands can be issued from the machining program of any part systems.
Number of usable spindles differ the machine model, confirm the specifications of the model used.



(a) 1st spindle
(b) Tool spindle (3rd spindle)
(c) 2nd spindle
(d) Turret 1

Command format

Sn=*** ... S6-digit binary data**

n	Designate the spindle number with one numeric character. (1 to 7)
*****	Rotation speed or constant surface speed command value.

Detailed description

(1) Each spindle command is delimited by the details of n.
(Example)
S1 = 3500 ; 1st spindle 3500(r/min) command
S2 = 1500 ; 2nd spindle 1500(r/min) command
S3 = 2000 ; 3rd spindle 2000(r/min) command
S4 = 2500 ; 4th spindle 2500(r/min) command
S5 = 2000 ; 5th spindle 2000(r/min) command
S6 = 3000 ; 6th spindle 3000(r/min) command
S7 = 3500 ; 7th spindle 3500(r/min) command

(2) Multiple spindles can be commanded in one block.

(3) If two or more commands are issued to the same spindle in a block, the command issued last will be valid.
(Example) S1 = 3500 S1 = 3600 S1 = 3700 ; S1 = 3700 will be valid.

(4) The S***** command and Sn=***** command can be used together.
The spindle targeted for the S***** command is normally the 1st spindle, however, the S***** command can be used for 2nd or following spindle according to the spindle selection command.

(5) The commands for each spindle can be commanded from the machining program of any part systems. The spindles will rotate with the speed commanded last.
If the S commands are issued from two or more part systems, the command from the part system of largest No. will be valid.

10.6.2 Spindle selection command (Multiple-spindle Control II); G43.1, G44.1

Function and purpose

This function controls which spindle's rotation the cutting follows, in addition, designates the spindle to be selected when "S*****" command is issued.

Command format

G43.1 ... Selected spindle (nth spindle) control mode ON

G44.1 ... Selected spindle (nth spindle) control mode ON

Detailed description

(1) G43.1 and G44.1 are modal G codes.

(2) The spindle control mode entered when the power is turned ON or reset depends on the parameter setting.
Designate the spindle No. to be selected in G43.1 modal with the parameter (basic specifications parameter "#1199 Sselect").
This parameter is provided for every part system to set as follows.

#	Items		Details	Setting range (unit)
1199	Sselect	Select initial spindle control	Select the initial condition of spindle control when power is turned ON or reset.	0: Selected spindle control mode (G43.1) 1: 2nd spindle control mode (G44.1)
21049	SPname		Designate the spindle No. selected for the G43.1 modal in each part system.	0: 1st spindle 4: 4th spindle 1: 1st spindle 5: 5th spindle 2: 2nd spindle 6: 6th spindle 3: 3rd spindle 7: 7th spindle

Reset the NC after changing "#1199 Sselect " and "#21049 SPname" parameters. It is no use to turn the power OFF once and ON again.

(3) If the S command is issued in the same as the spindle selection commands (G43.1, and G44.1), which spindle the S command is valid for depends on the order that G43.1, G44.1, and S command are issued.
When S command precedes the G codes, it follows the G43.1 / G44.1 mode before S command is issued.
When G codes precede, it follows the G43.1 / G44.1 mode issued in the same block.

(4) G43.1 and G44.1 commands can be issued from every part system.

(5) The following functions change after the spindle selection command.
(a) Per rotation command (synchronous feed)
Even if F is commanded in the G95 mode, the per rotation feedrate for the selected spindle (nth spindle) will be applied during G43.1 mode and for the 2nd spindle during G44.1 mode.
(b) S commands (S*****, Sn=*****), constant surface speed control, thread cutting

Function	G43.1 mode	G44.1 mode
S command during G97/G96 Constant surface speed control Upper limit / Lower limit of spindle rotation speed command during constant surface speed control (G92 S_ Q) Thread cutting	Command control for the 1st spindle. (Note 1)	Command control for the 2nd spindle.

(Note 1) The spindle selected during G43.1 mode depends on the parameter "#21049 SPname".

(6) The Sn=***** command can be used to command the other spindle even if it is commanded during G43.1 or G44.1 mode.
Note that the rotation speed designation will be applied for such command even if the G96 mode is ON.
(Example) When "SPname" = 0;

	Rotation speed	
G43.1;	1st spindle	2nd spindle
G97 S1000; :	1000(r/min)	0(r/min)
S2 = 2000; :		2000(r/min)
G96 S100; :	100(m/min) (Note1)	
S2 = 2500; :		2500(r/min)
G44.1 S200; :		200(m/min)
S1 = 3000; :	3000(r/min)	
G97 S4000; :		4000(r/min)

(Note 1) The constant surface speed control will be switched to the 2nd spindle by G44.1 command.
Therefore, the 1st spindle retains its rotation speed as that of "G44.1 S200;" command.
The 1st spindle rotation speed will be 3000 (r/min) when "S1=3000;" command is issued.

11

Tool Functions (T command)

11.1 Tool Functions (T8-digit BCD)

Function and purpose

The tool functions are also known as T functions and they assign the tool numbers. They are designated with an 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 completion of the movement.
(2) The T function is executed simultaneously with the movement command.

Processing and completion sequences are required for all T commands.

12

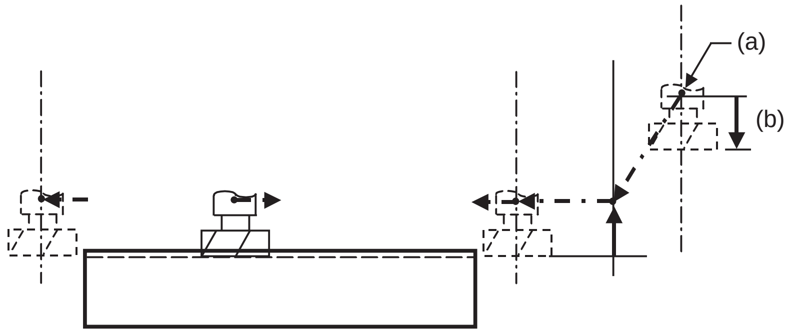
Tool Compensation Functions

12.1 Tool compensation

Function and purpose

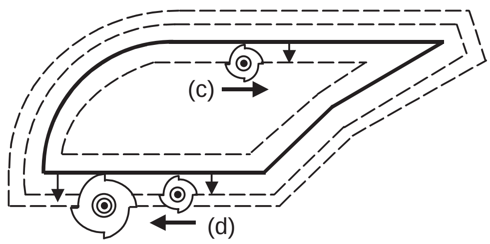
The basic tool offset function includes the tool length offset and tool radius compensation. Each offset amount is designated with the tool offset No. Each offset amount is input from the setting and display unit or the program.

Tool length offset



(Side view)

Tool radius compensation



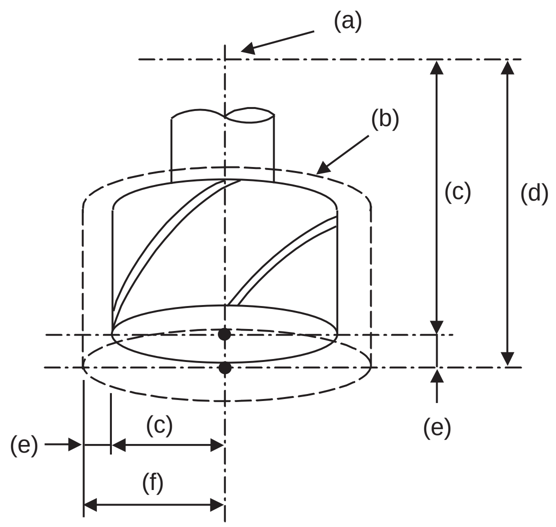
(Plane view)

(a) Reference position (b) Tool length (c) Right compensation (d) Left compensation

Tool offset memory

There are two types of tool offset memories for setting and selecting the tool offset amount. (The type used is determined by the machine maker specifications.)
The offset amount or 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 offset memory	Classification of length offset, radius compensation	Classification of shape offset, wear compensation
Type 1	Not applied	Not applied
Type 2	Applied	Applied



(a) Reference position　(b) Reference tool　(c) Shape
(d) Tool length offset　(e) Wear amount　(f) Tool radius compensation

Type 1

One offset amount corresponds to one offset No. as shown on the right. Thus, these can be used commonly regardless of the tool length offset amount, tool radius compensation amount, shape offset amount and wear compensation amount.

(D1) = a1, (H1) = a1

(D2) = a2 , (H2) = a2

:   :

(Dn) = an, (Hn) = an

Offset No.	Offset amount
1	a1
2	a2
3	a3
:	:
:	:
n	an

Type 2

The shape offset amount related to the tool length, wear compensation amount, shape offset related to the tool radius and the wear compensation amount can be set independently for one offset No. as shown below.

The tool length offset 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

:   :

(Hn) = bn + cn, (Dn) = dn + en

Offset No.	Tool length (H)		Tool radius (D)/(Position offset)	
	Shape offset amount	Wear compensation amount	Shape offset 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

1. If the tool offset amount is changed during automatic operation (including during single block stop), it will be validated from the next block or multiple blocks onwards.

Tool offset No. (H/D)

This address designates the tool offset No.

(1) H is used for the tool length offset, and D is used for the tool position offset and tool radius compensation.

(2) The tool offset No. that is designated once does not change until a new H or D is designated.

(3) The offset 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 offset 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 offset amount for each offset No. is preset with the setting and display unit.

Input setting unit	Shape offset amount		Wear compensation amount	
	Metric system	Inch system	Metric system	Inch system
#1015 cunit=100	±99999.99 (mm)	±9999.999 (inch)	±9999.99 (mm)	±999.999 (inch)
#1015 cunit=10	±9999.999 (mm)	±999.9999 (inch)	±999.999 (mm)	±99.9999 (inch)

12.2 Tool Length Offset/Cancel ; G43,G44/G49

Function and purpose

The end position of the movement command for each axis can be offset 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 offset amount using this function.

Command format

G43 Zz Hh ; ... Tool length offset + start

G44 Zz Hh ; ... Tool length offset - start

G49 Zz ; ... Tool length offset cancel

Detailed description

Tool length offset movement amount

The movement amount is calculated with the following expressions when the G43 or G44 tool length offset command or G49 tool length offset cancel command is issued.

	Z axis movement amount	
G43 Zz Hh1	; z +(lh1)	Offset in + direction by tool offset amount
G44 Zz Hh1	; z -(lh1)	Offset in - direction by tool offset amount
G49 Zz ;	; z -(+)(lh1)	Offset amount cancel

lh1; Offset amount for offset No. h1

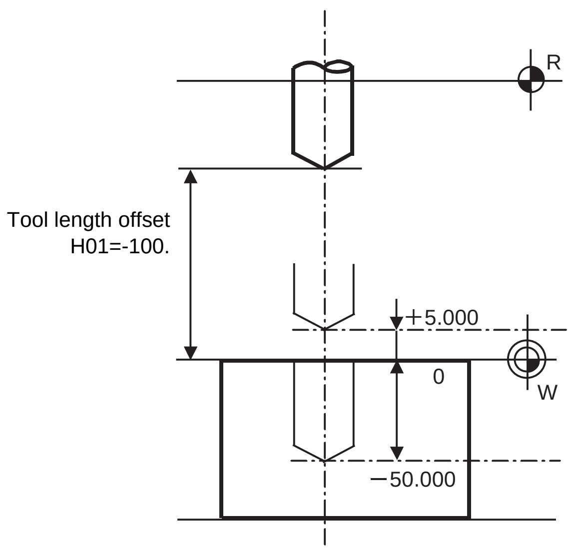
Regardless of the absolute value command or incremental value command, the actual end point will be the point offset by the offset amount designated for the programmed movement command end point coordinate value.
The G49 (tool length offset cancel) mode is entered when the power is turned ON or when 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 ;
```

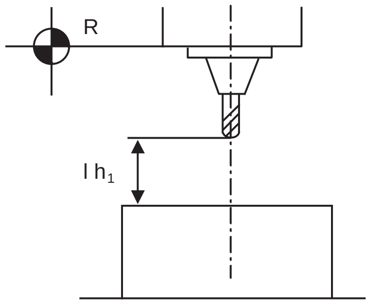


Offset No.

(1) The offset amount differs according to the compensation type.

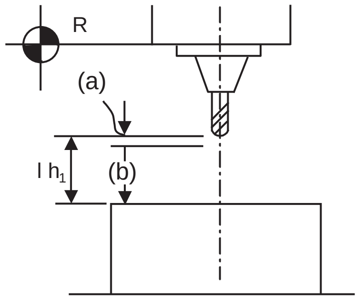
Type 1

G43 Hh1;
When the above is commanded, the compensation amount lh1 commanded with compensation No. h1 will be applied commonly regardless of the tool length offset amount, tool radius compensation amount, shape offset amount or wear compensation amount.



Type 2

G43 Hh1;
When the above is commanded, the compensation amount lh1 commanded with compensation No. h1 will be as follows.
lh1: Shape offset (b) + wear compensation amount (a)



(2) The valid range of the offset No. will differ according to the specifications (No. of offset sets).

(3) If the commanded offset No. exceeds the specification range, the program error (P170) will occur.

(4) Tool length cancel will be applied when H0 is designated.

(5) The offset No. commanded in the same block as G43 or G44 will be valid for the following modals.
(Example 3)

```
 G43  Zz1  Hh1  ;Tool length offset is executed with h1.
   :
 G45 Xx1 Yy1 Hh6 ;
   :
 G49 Zz2 ;   ........................ The tool length offset is canceled.
   :
 G43 Zz2 ;   ........................ Tool length offset is executed again with h1.
   :
```

(6) If G43 is commanded in the G43 modal, an offset of the difference between the offset No. data will be executed.
(Example 4)

```
G43  Zz1  Hh1 ; ............  Becomes the z1 + (lh1) movement.
   :
G43  Zz2  Hh2 ; ............  Becomes the z2+(lh2-lh1) movement.
   :
```

The same applies for the G44 command in the G44 modal.

Axis valid for tool length offset

(1) When parameter "#1080 Dril_Z" is set to "1", the tool length offset is always applied on the Z axis.

(2) 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 ; ............  + offset to X axis
        :
G49  Xx2 ;
        :
G44  Yy1  Hh2 ; ............  -offset to Y axis
        :
G49  Yy2 ;
        :
 G43  α α1 Hh3 ;............  + offset to additional axis
        :
G49  α α1 ;
        :
G43  Xx3  Yy3  Zz3 ;............  Offset is applied on Z axis
    :
G49 ;
```

The handling of the additional axis will follow the parameters "#1029 to 1031 aux_I, J and K" settings.
If the tool length offset is commanded for the rotary axis, set the rotary axis name for one of the parallel axes.

(3) If H (offset No.) is not designated in the same block as G43, the Z axis will be valid.
(Example 6)

```
G43  Hh1 ;............  Offset and cancel to X axis
        :
G49  ;
```

Movement during other commands in tool length offset modal

(1) If reference point return is executed with G28 and manual operation, the tool length offset will be canceled when the reference point return is completed.
(Example 7)

```
 G43 Zz1 Hh1 ;
   :
 G28 Zz2 ;   ..................  Canceled when reference point is reached. (Same as G49)
   :
 G43 Zz2  Hh2 ;
   :
 G49 G28  Zz2 ; ............  The tool length offset will be included when positioning the intermediate point.
                               Canceled when reference point is reached.
```

(2) The movement is commanded to the G53 machine coordinate system, the axis will move to the machine position when the tool offset 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 offset amount.

12.3 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

G40 X__Y__; ... Tool radius compensation cancel

G41 X__Y__; ... Tool radius compensation (left)

G42 X__Y__; ... Tool radius compensation (right)

G38 I__J__; ... Change or hold of compensation vector (Can be commanded only during the radius compensation mode.)

G39 X__Y__; ... Corner changeover (Can be commanded only during the radius compensation mode.)

Detailed description

The number of sets for the compensation differ according to machine specification. (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.

12.3.1 Tool Radius Compensation Operation

Detailed description

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 a compensation cancel command (G40) is issued

The compensation 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.

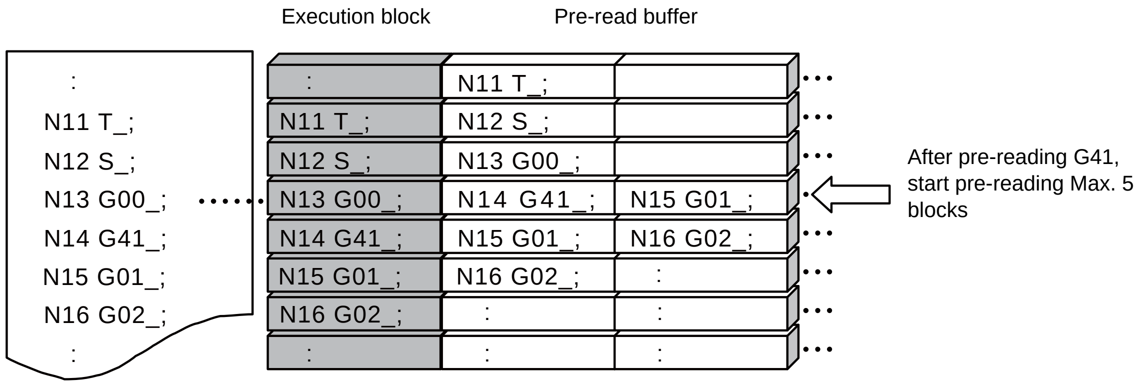
Tool radius compensation start (startup)

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 <= max. offset No.
(3) The movement command of positioning (G00) or linear interpolation (G01) is issued.

Whether in continuous or single block operation, compensation always starts after reading three blocks, or if the three blocks do not contain any movement command, up to five continuous blocks wil be pre-read.
In compensation mode, too, up to 5 blocks are pre-read and the compensation is arithmetically processed.

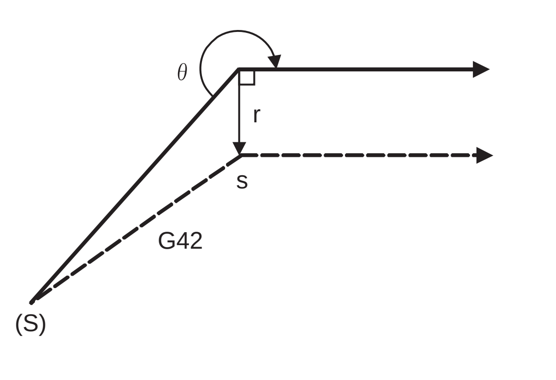
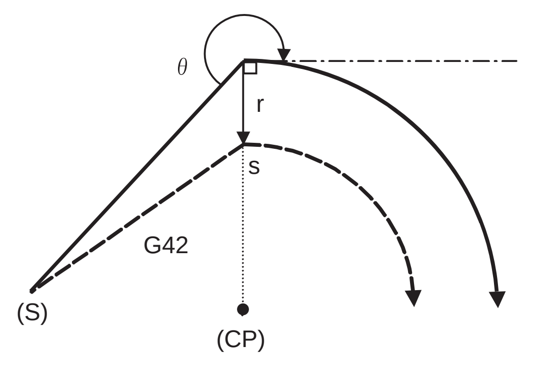
[Control state diagram]



There are two ways of starting the compensation operation: type A and type B.
The type can be selected with parameter "#1229 set01".
This type is used in common with the compensation cancel type.

Start operation for tool radius compensation

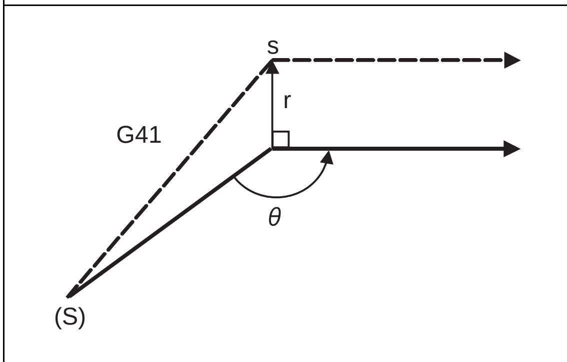
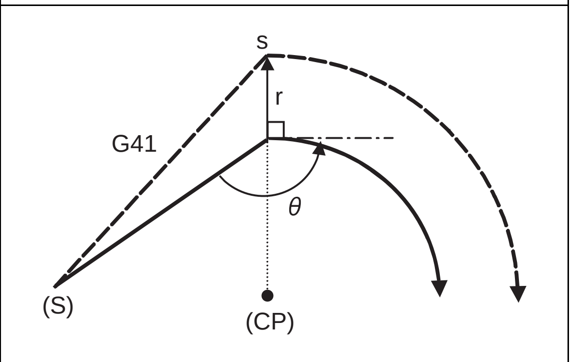
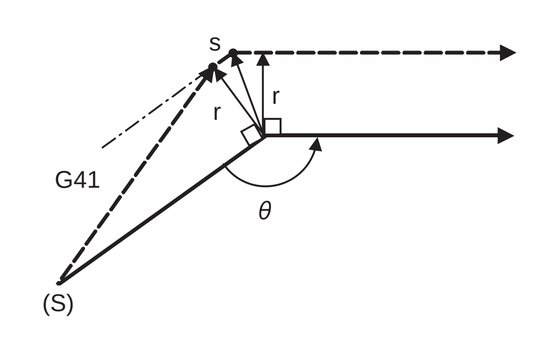
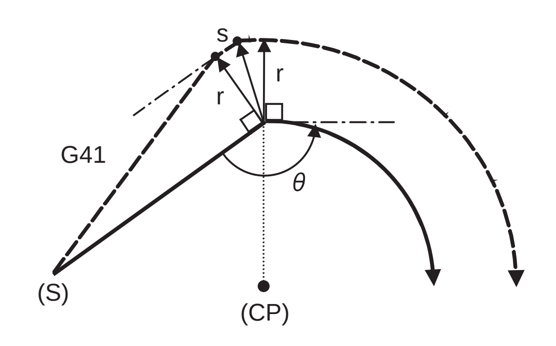
(1) Machining an inside corner

Linear -> Linear	Linear -> Circular
	

(S) Start point (CP) Center of circular r: Compensation amount s: Stop point with single block

Program path Tool center path

(2) Machining an outside corner (obtuse angle) [90° <= θ < 180°]

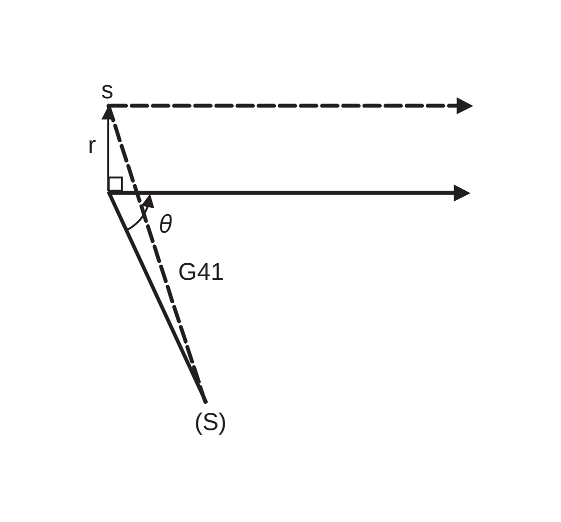
Linear -> Linear (Type A)	Linear -> Circular (Type A)
	
Linear -> Linear (Type B)	**Linear -> Circular (Type B)**
	

(S) Start point (CP) Center of circular r: Compensation amount s: Stop point with single block

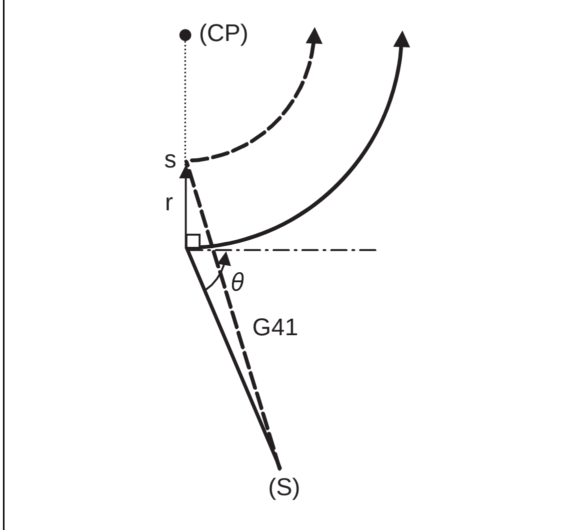
Program path Tool center path

(3) Machining an outside corner (acute angle) [θ < 90°]

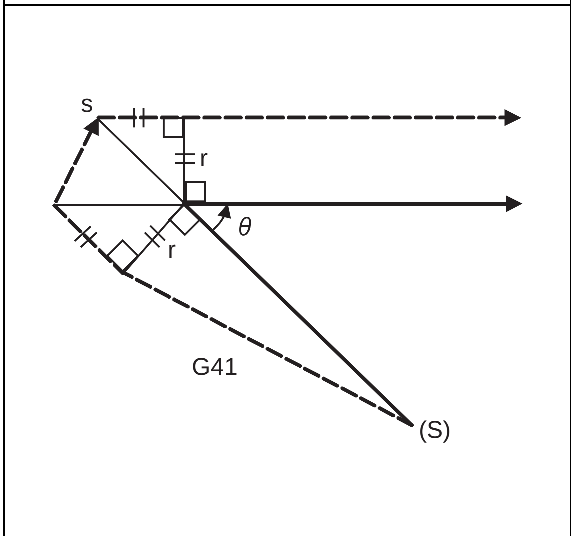
Linear -> Linear (Type A)



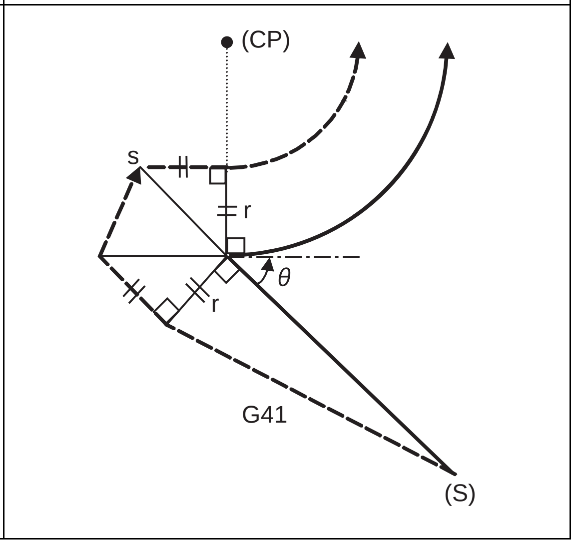
Linear -> Circular (Type A)



Linear -> Linear (Type B)



Linear -> Circular (Type B)



(S) Start point (CP) Center of circular r: Compensation amount s: Stop point with single block

——— Program path - - - - Tool center path

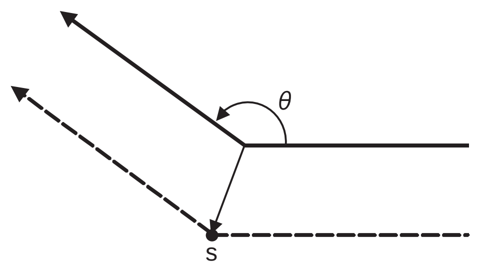
(Note 1) If there is no axis movement command in the same block as G41 or G42, compensation is performed perpendicularly to the next block's direction.

Operation in compensation mode

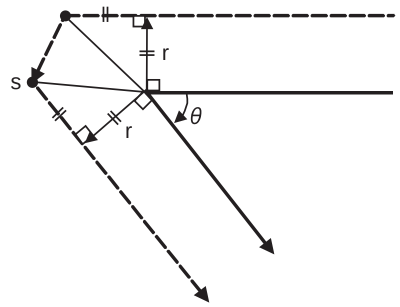
Calculate the tool center path from the linear line/circular arc to perform compensation to the program path (G00, G01, G02, G03).
Even if the same compensation command (G41, G42) is issued in the compensation mode, the command will be ignored.
When 4 or more blocks without movement command are continuously specified in the compensation mode, overcutting or undercutting will occur.
When the M00 command has been issued during tool radius compensation, pre-reading is prohibited.

(1) Machining an outside corner

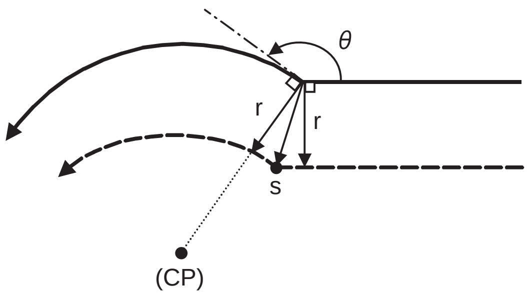
Linear -> Linear (90°<= θ < 180°)



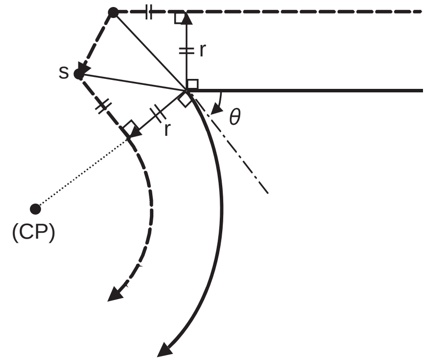
Linear -> Linear (0° < θ < 90°)



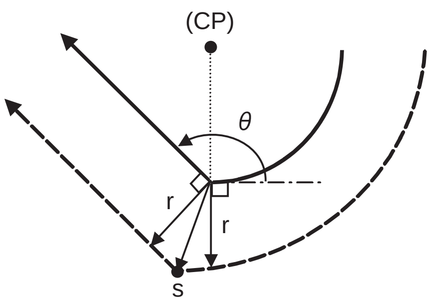
Linear -> Circular (90° <= θ < 180°)



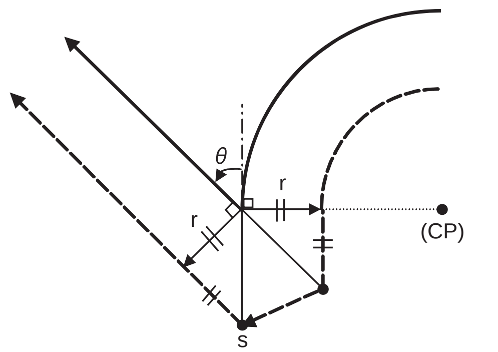
Linear -> Circular (0° < θ < 90°)



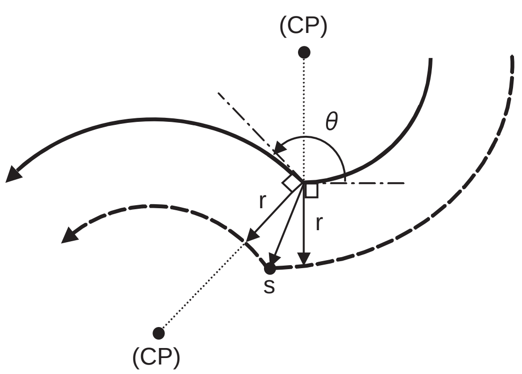
Circular -> Linear (90° <= θ < 180°)



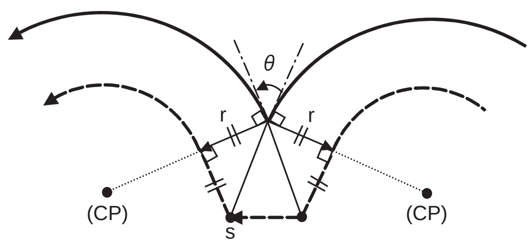
Circular -> Linear (0° < θ < 90°)



Circular -> Circular (90° <= θ < 180°)



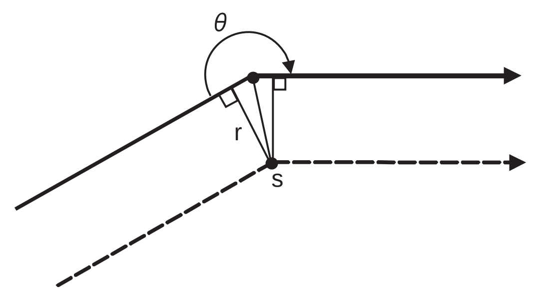
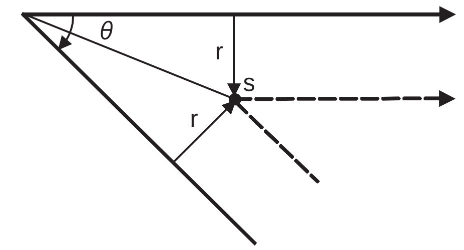
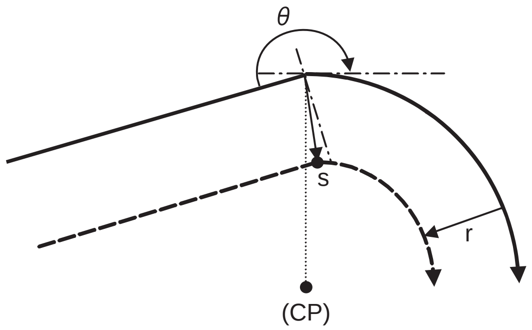
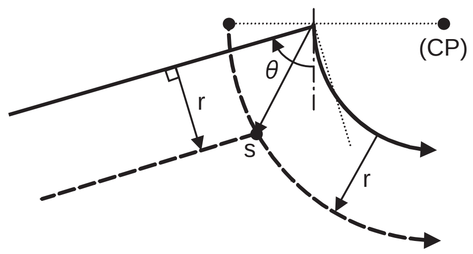
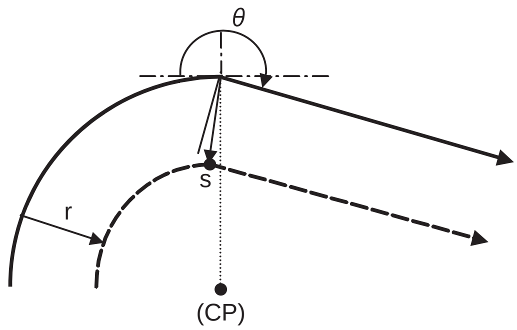
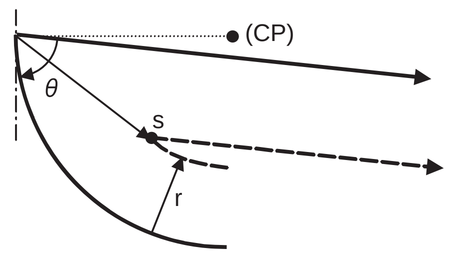
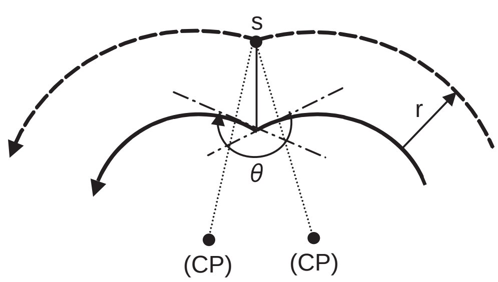
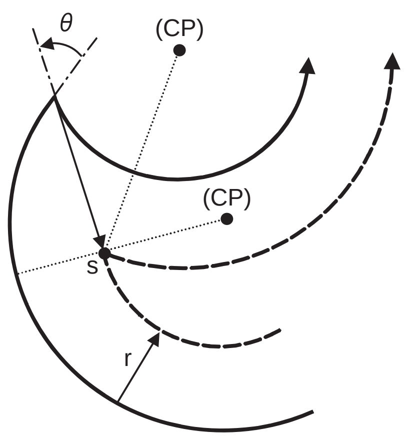
Circular -> Circular (0° < θ < 90°)



(CP) Center of circular r: Compensation amount s: Stop point with single block

Program path Tool center path

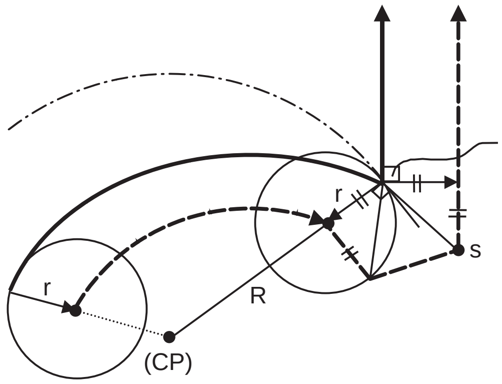
(2) Machining an inside corner

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 -> Circular (Obtuse angle)	**Circular -> Circular (Acute angle)**
	

(CP) Center of circular   r: Compensation amount   s: Stop point with single block

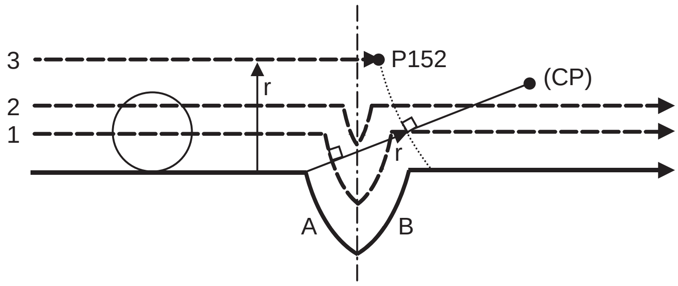
Program path   Tool center path

(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.
Normal circular command
If the error after compensation is within the parameter value ("#1084 RadErr"), it is interpolated as a spiral arc.



(E) End point of circular
(CP) Center of circular
r: Compensation amount

(4) When the inner intersection point does not exist
In cases like the figure below, the intersection point of circulars A and B may not exist depending on the compensation amount.
In such cases, program error (P152) appears and the tool stops at the end point of the previous block.
In the pattern 1 and 2 in this figure, machining is possible because compensation amount r is small.
In pattern 3, compensation r is so large that an intersection does not exist and program error (P152) will occur.



(CP) Center of circular
r: Compensation amount

Program path
Tool center path

Tool radius compensation cancel

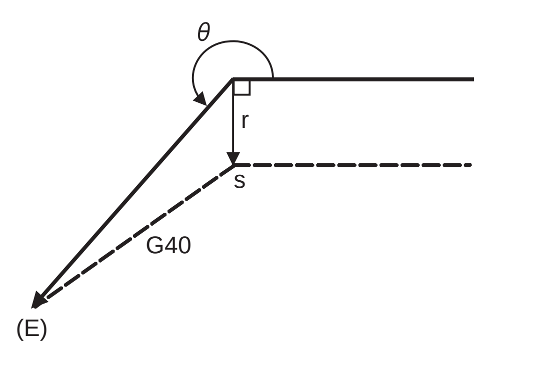
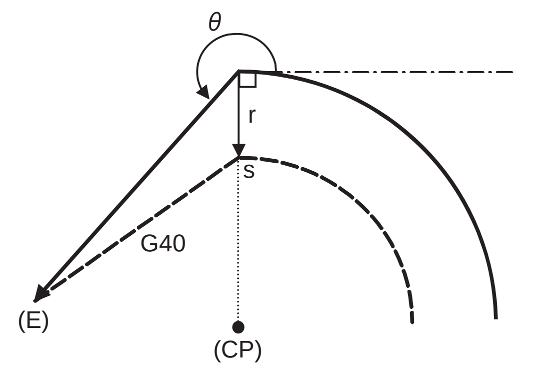
If either of the following conditions is met in the tool radius compensation mode, the compensation will be canceled.
However, the movement command must be a circular command.
If the compensation is canceled by a circular command, program error (P151) will occur.

(1) The G40 command has been executed.
(2) Executed the compensation No. D00.

The cancel mode is established once the compensation cancel command has been read, 5-block pre-reading is suspended and 1-block pre-reading will be operated.

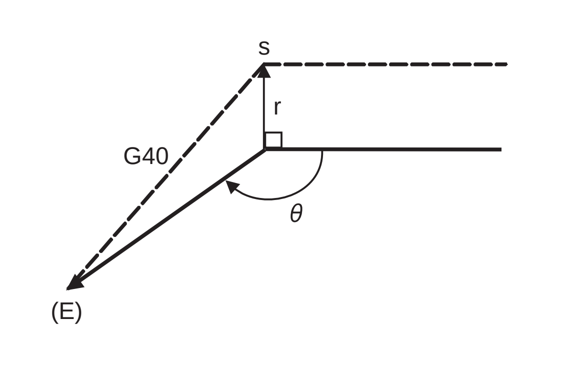
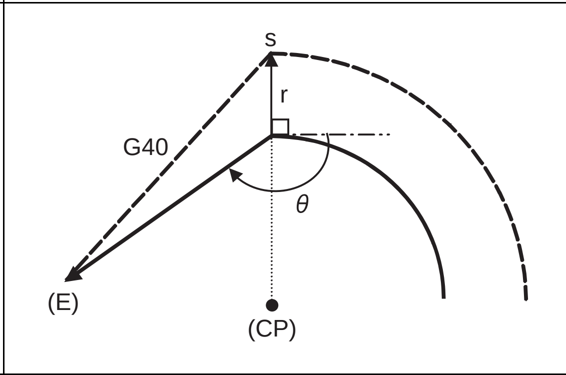
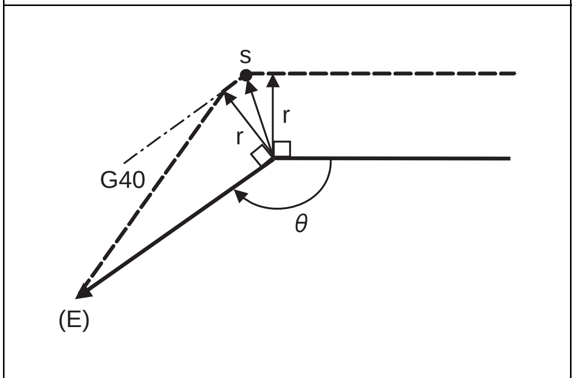
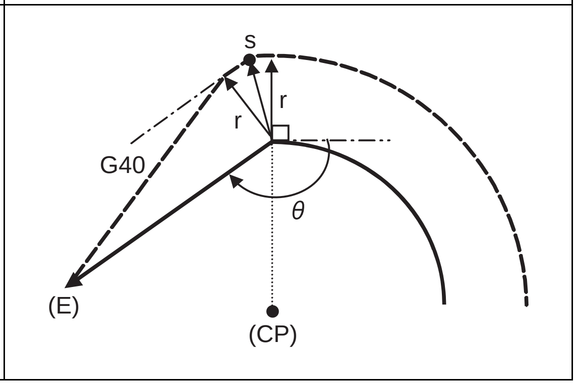
Tool radius compensation cancel operation

(1) Machining an inside corner

Linear -> Linear	Circular -> Linear
	

(E) End point (CP) Center of circular r: Compensation amount s: Stop point with single block

(2) Machining an outside corner (obtuse angle) [90° <= θ < 180°]

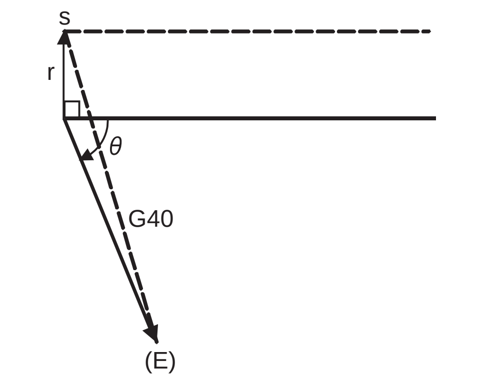
Linear -> Linear (Type A)	**Circular -> Linear (Type A)**
	
Linear -> Linear (Type B)	**Circular -> Linear (Type B)**
	

(E) End point (CP) Center of circular r: Compensation amount s: Stop point with single block

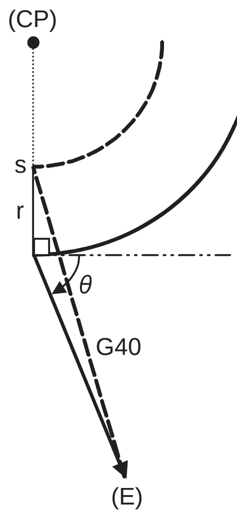
Program path Tool center path

(3) Machining an outside corner (acute angle) [θ < 90°]

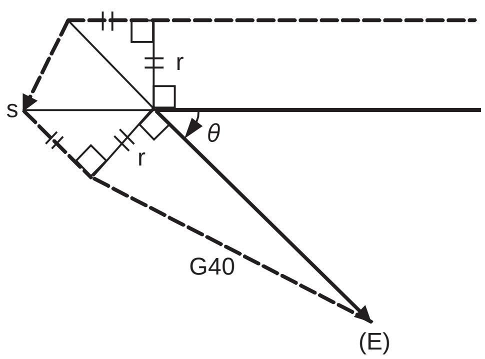
Linear -> Linear (Type A)



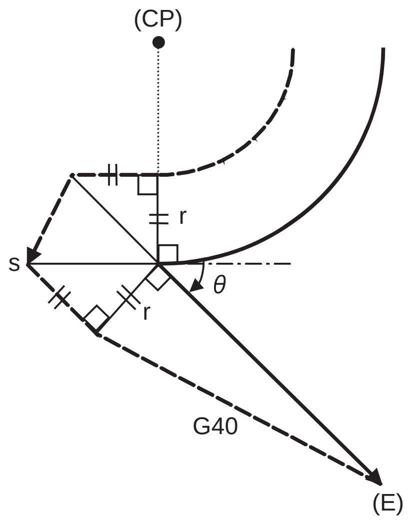
Circular -> Linear (Type A)



Linear -> Linear (Type B)



Circular -> Linear (Type B)



(E) End point (CP) Center of circular r: Compensation amount s: Stop point with single block

Program path

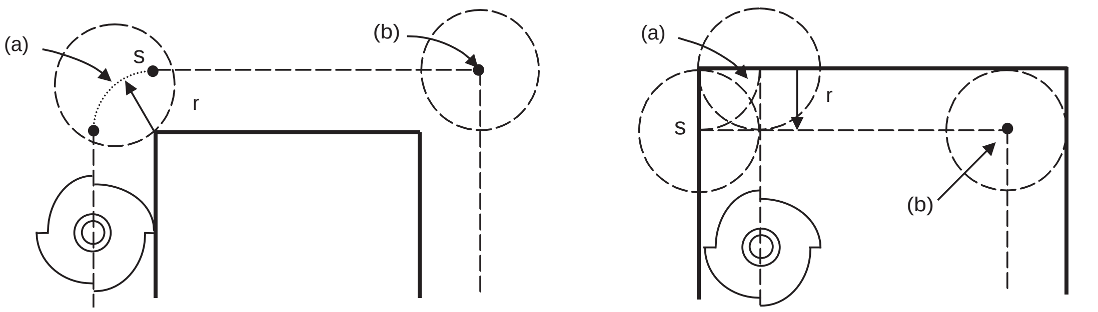
Tool center path

12.3.2 Other Commands and Operations during Tool Radius Compensation

Detailed description

Insertion of corner arc

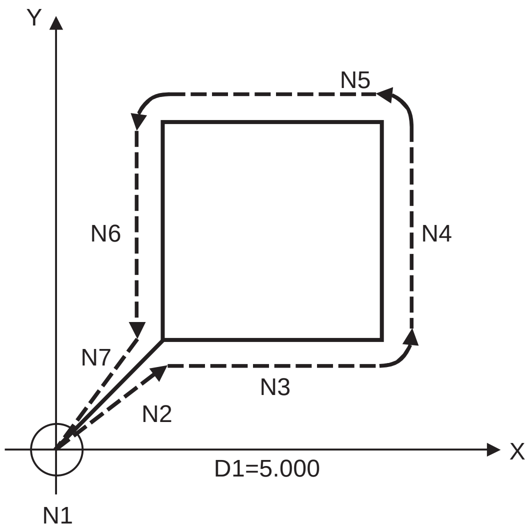
An arc that uses the compensation amount as the radius is inserted without calculating the point of intersection at the workpiece corner when G39 (corner arc) is commanded.



(With G39 command) (No G39 command) (With G39 command) (No G39 command)

[For outer side compensation] **[For inner side compensation]**

(a) Inserted circular (b) Point of intersection r: Compensation amount s: Stop point with single block




```
N1 G28 X0 Y0 ;
N2 G91 G01 G42 X20. Y20. D1 F100 ;
N3 G39 X40. ;
N4 G39 Y40. ;
N5 G39 X-40. ;
N6 Y-40. ;
N7 G40 X-20. Y-20. ;
N8 M02 ;
```

Program path

Tool center path

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;

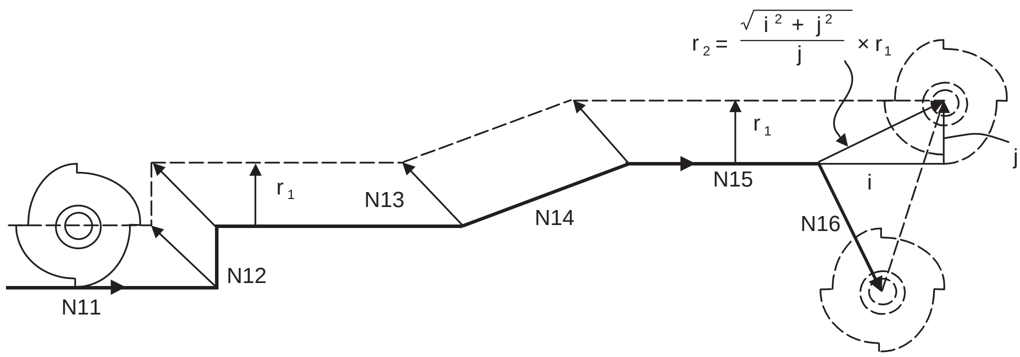
This can be used for pick feed, etc.

(2) Changing of vector

A new compensation vector direction can be commanded with I, J and K, and a new compensation 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.)



N11 G1 Xx11;
N12 G38 Yy12;
N13 G38 Xx13;
N14 G38 Xx14 Yy14;
N15 G38 Xx15 Ii Jj Dd2; ... Vector change
N16 G40 Xx16 Yy16;

Program path

Tool center path

The compensation amount "d" vector is created in the commanded i and j vector direction.

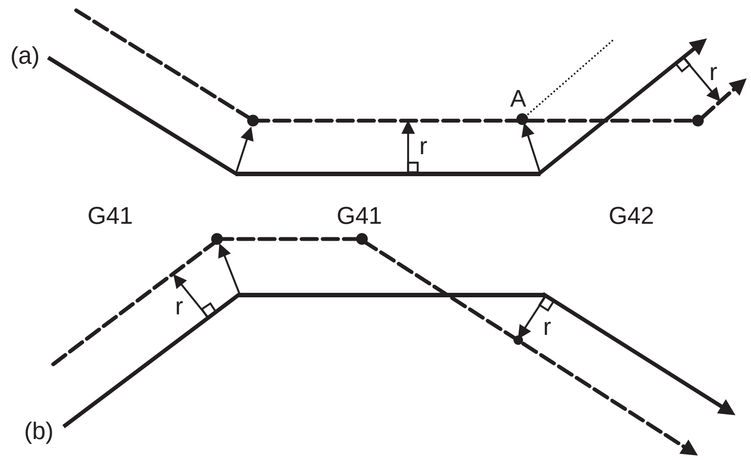
Changing the compensation direction during tool radius compensation

The compensation direction is determined by the tool radius compensation commands (G41, G42) and compensation amount sign.

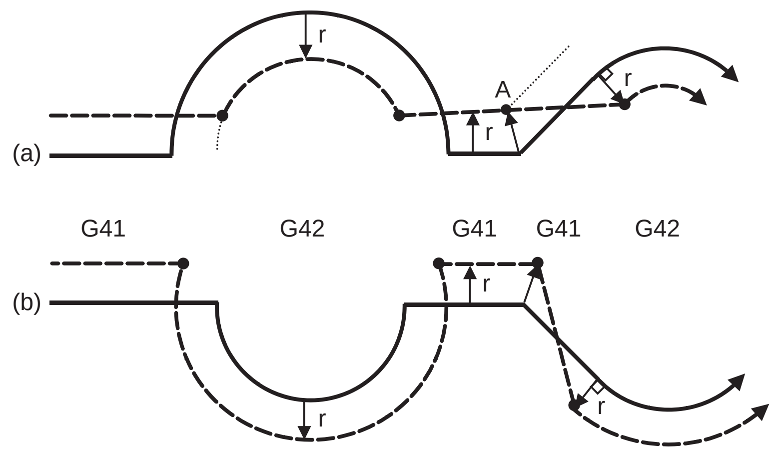
G code	Compensation amount sign +	Compensation amount sign -
G41	Left-side compensation	Right-side compensation
G42	Right-side compensation	Left-side compensation

The compensation direction can be changed by changing the compensation command during the compensation mode without canceling the mode.
However, it is impossible to change the direction in the compensation start block and the next block.

(1) Linear -> Linear
(a) When there is a point of intersection (A) when the compensation direction is changed.
(b) When there is no point of intersection when the compensation direction is changed.



(2) Linear <-> Circular
(a) When there is a point of intersection (A) when the compensation direction is changed.
(b) When there is no point of intersection when the compensation direction is changed.

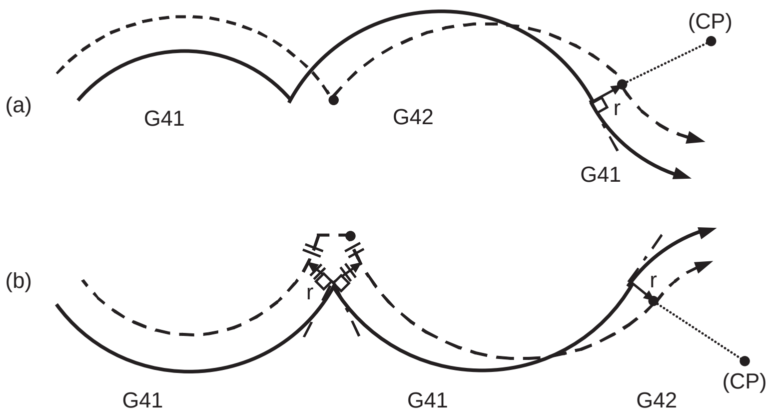


Program path   Tool center path

(3) Circular -> Circular

(a) When there is a point of intersection when the compensation direction is changed.

(b) When there is no point of intersection when the compensation direction is changed.



(CP) Center of circular

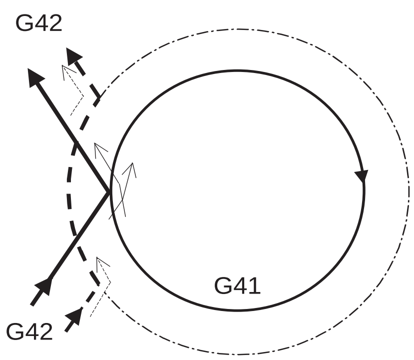
(4) Linear return

(5) Arc exceeding 360° due to compensation

In the case below, it is possible that the arc may exceed 360°

a. Changing the compensation direction by switching between G41/G42.

If the arc exceeds 360°, compensation will be performed as shown in the figure and uncut section will be left.




```
:
G42 G01 X_ Y_;
G41 G02 X_ Y_ I_ K_;
G42 G01 X_ Y_;
:
```

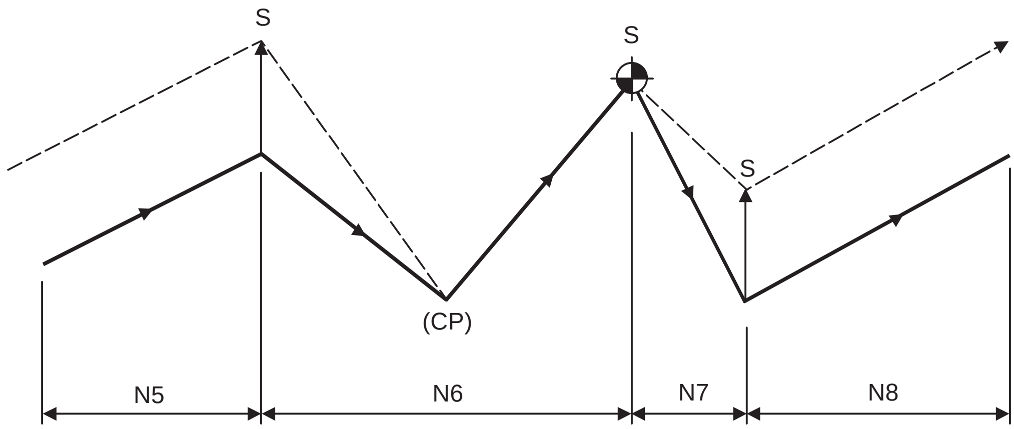
Program path   Tool center path

Section left uncut

Command for eliminating compensation vectors temporarily

When the following command is issued in the compensation mode, the compensation vectors are temporarily eliminated and then, compensation mode will automatically return.
In this case, the compensation is not canceled, and the tool goes directly from the intersection point vector to the point without vectors, in other words, to the programmed command point. When returning to the compensation mode, it goes directly to the intersection point.

(1) Reference position return command
Temporarily no compensation vectors at intermediate point. (Reference position when there is no intermediate point)

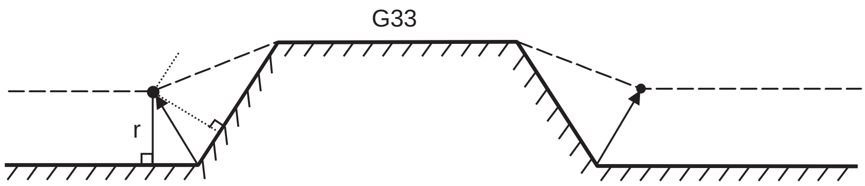



```
 (G41)  :                                                        (CP) Intermediate Point
N5 G91 G01 X60. Y30. ;
N6 G28 X50. Y-40. ;      <-Temporarily no compensation vectors at intermediate point.
N7        X30. Y-60. ;        (Reference position when there is no intermediate point)
N8        X70. Y40. ;
          :
```

(2) 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.

(3) G33 thread cutting command
Tool radius compensation does not apply to the G33 block.



Blocks without movement

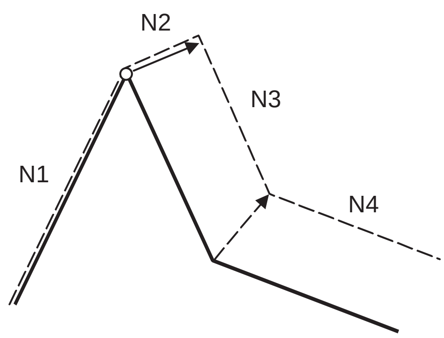
The following blocks are known as blocks without movement.

M03 ;	M command
S12 ;	S command
T45 ;	T command
G04 X500 ;	Dwell
G22 X200. Y150. Z100 ;	Machining prohibited region setting
G10 L10 P01 R50 ;	Compensation amount setting
G92 X600. Y400. Z500. ;	Coordinate system setting
(G17) Z40. ;	Movement outside the compensation plane
G90 ;	G code only
G91 X0 ;	Movement amount 0

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.

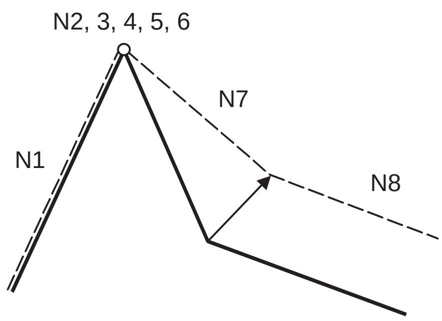
N1 X30.Y60. :
N2 G41 D10 ; <- Block without movement
N3 X20.Y-50. ;
N4 X50.Y-20. ;



Compensation vector cannot be created when there are four or more successive blocks without movement, or when pre-reading prohibiting M command is issued.

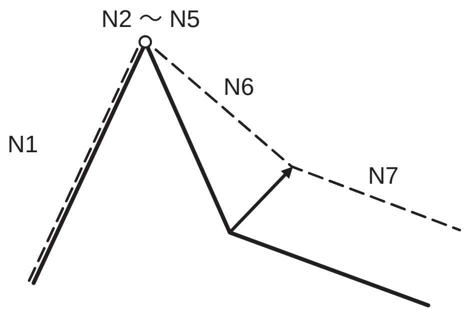
N1 X30.Y60. :
N2 G41 D10 ;
N3 G4 X1000 ;
N4 F100 ;
N5 S500 ;
N6 M3 ;
N7 X20.Y-50. ;
N8 X50.Y-20. ;

$\left.\begin{array}{l} \text{N2} \\ \text{N3} \\ \text{N4} \\ \text{N5} \\ \text{N6} \end{array}\right\}$ Block without movement



N1 G41 X30. Y60. D10 ;
N2 G4 X1000 ;
N3 F100 ;
N4 S500 ;
N5 M3 ;
N6 X20. Y-50. ;
N7 X50. Y-20. ;

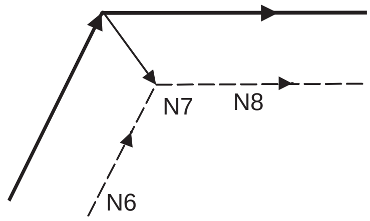
$\left.\begin{array}{l} \text{N2} \\ \text{N3} \\ \text{N4} \\ \text{N5} \end{array}\right\}$ Block without movement



(2) When command is assigned in the compensation mode

Compensation vector will be created as normal when there are not four or more successive blocks without movement, or when pre-read prohibiting M command is not issued.

N6 G91 X100. Y200. ;
N7 G04 X P1000 ; ... Block without movement
N8 X200. ;



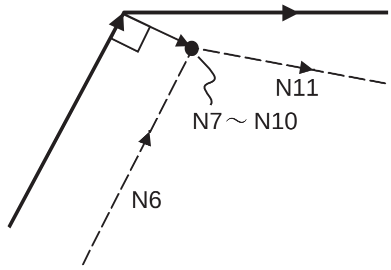
Block N7 is executed at N7 in the figure.

Compensation vector will be created perpendicularly to the end point of the previous block when there are four or more successive blocks without movement, or when pre-read prohibiting M command is issued.

In this case, a cut may occur.

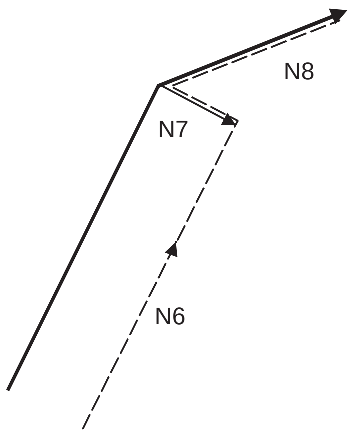
N6 X100. Y200. ;
N7 G4 X1000 ;
N8 F100 ;
N9 S500 ;
N10 M4 ;
N11 X100 ;

Block without movement



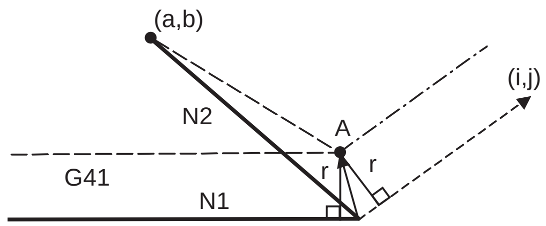
(3) When commanded together with compensation cancel

N6 X100. Y200. ;
N7 G40 M5 ;
N8 X100. Y50. ;



When I, J, K are commanded in G40

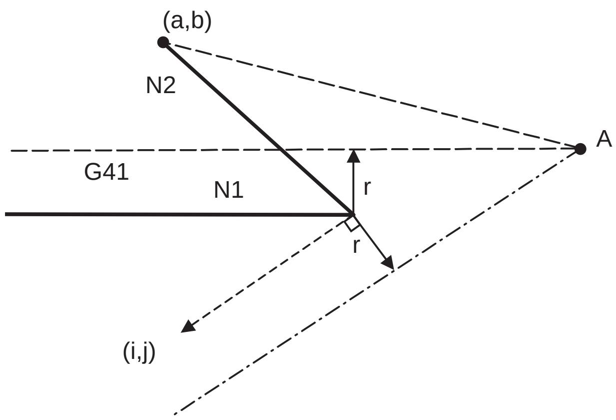
(1) If the final movement command block in the four blocks before the G40 block is the G41 or G42 mode, it will be assumed that the movement is commanded in the vector I, J or K direction from the end point of the final movement command. After interpolating between the hypothetical tool center path and point of intersection, it will be canceled. The compensation direction will not change.



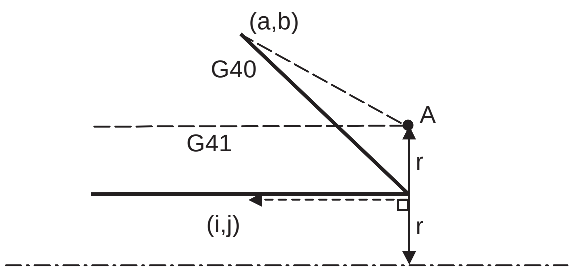
N1 (G41) G1 X_ ;
N2 G40 Xa Yb Ii Jj ;

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.

[When the I and j symbols in the above program example are incorrect]

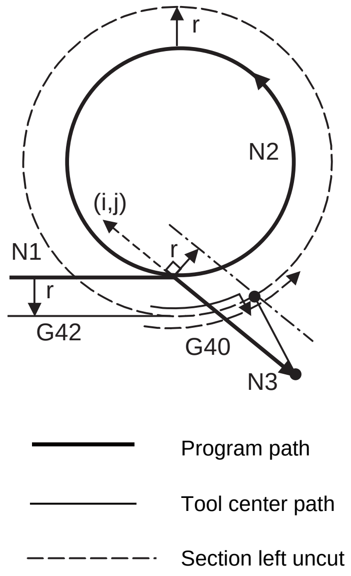


If the compensation vector obtained with point of intersection calculation is extremely large, a perpendicular vector will be created in the block before G40.



Program path
Tool center path
Hypothetical tool center path

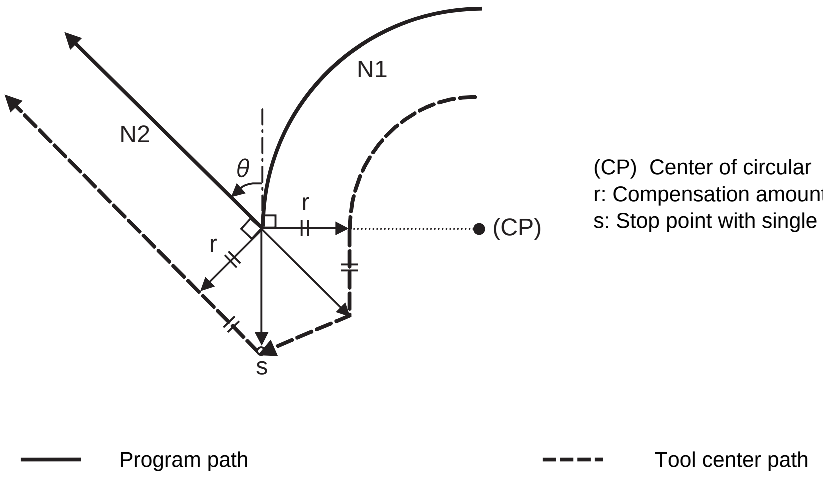
(2) If the arc is 360° or more due to the details of I, J and K at G40 after the arc command, an uncut section will occur.



N1 (G42,G91) G01 X200. ;
N2 G02 J150. ;
N3 G40 G1 X150. Y-150. I-100. J100. ;

Corner movement

When a multiple number of compensation vectors are created at the joints between movement command blocks, the tool will move in a straight line between these 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.



(CP) Center of circular
r: Compensation amount
s: Stop point with single block

12.3.3 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 (X-Y plane) G41/G42 X__ Y__ I__ J__ ;

G18 (Z-X plane) G41/G42 X__ Z__ I__ K__ ;

G19 (Y-Z plane) G41/G42 Y__ Z__ J__ K__ ;

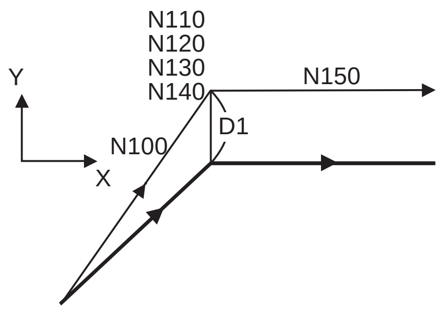
Assign a linear command (G00, G01) in a movement mode.

Detailed description

I, J type vectors (G17 X-Y plane selection)

This section describes the new I,J type vectors (G17 plane) created by this command. (Similar descriptions apply to vector K,I for the G18 plane and to J, K for the G19 plane.)
As shown in the following figures, I, J type vectors create compensation vectors which are perpendicular to the direction designated by I, J and equivalent to the compensation amount, without the intersection point calculation of the programmed path. The I, J vectors 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

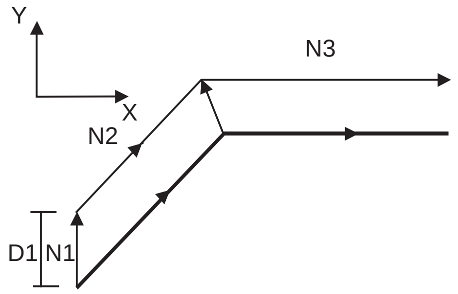



```
      (G40)
        ≀
N100  G91  G41  X100. Y100. I150. D1 ;
N110  G04  X1000 ;
N120  G01  F1000 ;
N130  S500 ;
N140  M03 ;
N150  X150. ;
        ≀
```

Program path

Tool center path

(2) When there are no movement commands at the compensation start.

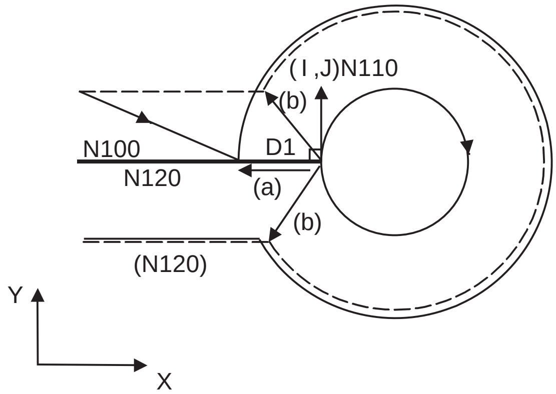



```
(G40)
  ~
N1  G41  I150. D1 ;
N2  G91  X100. Y100. ;
N3  X150. ;
  ~
```

Program path

Tool center path

(3) When I, J has been commanded in the mode (G17 plane)




```
  ~
(G17  G41  G91)
N100  G41  G00 X150. J50. ;
N110  G02  I150. ;
N120  G00  X-150. ;
  ~
```

(a) I, J type vector (b) Intersection point calculation type vector

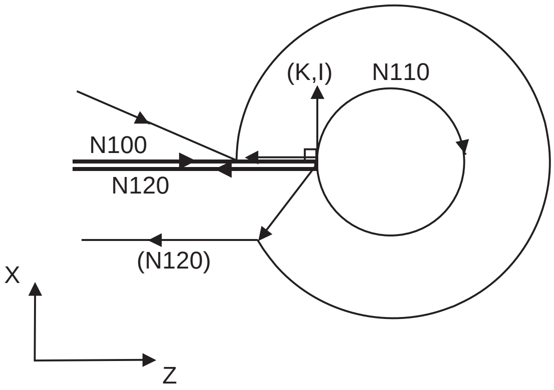
Program path

Tool center path

Path after intersection point calculation

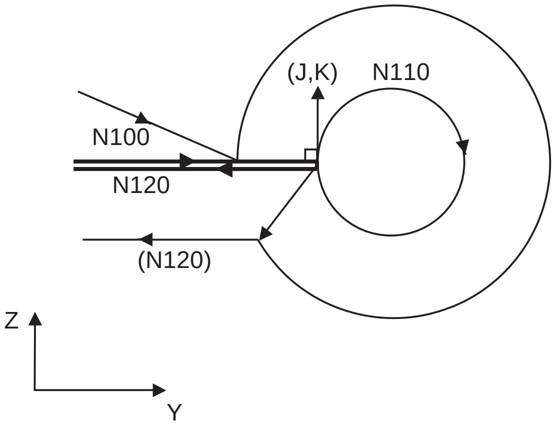
(Reference)

(a) G18 plane



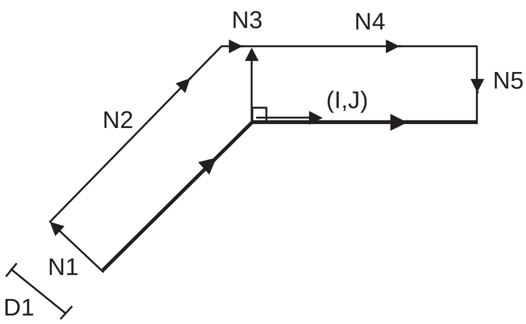

```
~
(G18  G41  G91)
N100  G41  G00  Z150. I50. ;
N110  G02  K50. ;
N120  G00  Z - 150. ;
~
```

(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

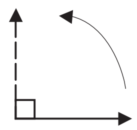
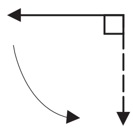



```
N1  G41  D1  G01  F1000 ;
N2  G91  X100. Y100. ;
N3  G41  I50. ;
N4  X150. ;
N5  G40 ;
```

Offset vector direction

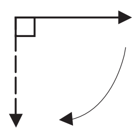
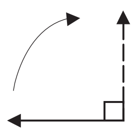
(1) In G41 mode

Direction produced by rotating the direction commanded by I,J by 90° to the left when looking at the zero point from the forward direction of the Z axis (3rd axis).

(Example 1) With I100.		(Example 2) With I-100.	
			
	(100, 0) IJ direction		(-100, 0) IJ direction
	Offset vector direction		Offset vector direction

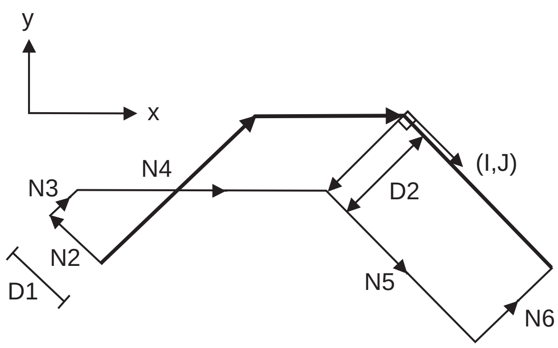
(2) In G42 mode

Direction produced by rotating the direction commanded by IJ by 90° to the right when looking at the zero point from the forward direction of the Z axis (3rd axis).

(Example 1) With I100.		(Example 2) With I-100.	
			
	(100, 0) IJ direction		(-100, 0) IJ direction
	Offset vector direction		Offset vector direction

Selection of offset modal

G41 and G42 modals can be switched over at any time.



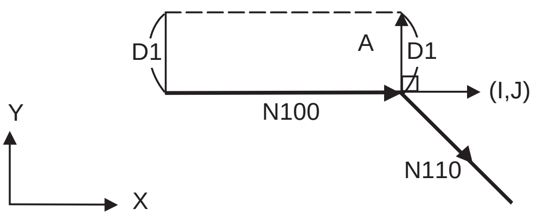

```
N1  G28  X0 Y0  ;
N2  G41  D1  F1000  ;
N3  G01  G91  X100.  Y100.  ;
N4  G42  X100.  I100.  J - 100.
      D2  ;
N5  X100.  Y- 100.  ;
N6  G40  ;
N7  M02  ;
%
```

Compensation amount for offset vectors

The compensation amount is determined by the compensation No. (modal) in a block with the IJ designation.

<Example 1>

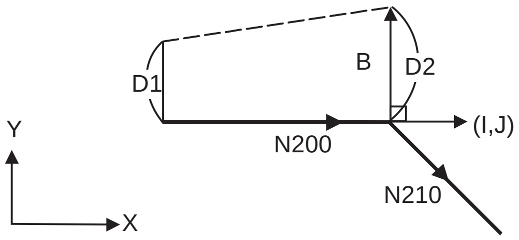
Vector A is the compensation amount registered in compensation No. modal D1 of the N100 block.




```
(G41  D1  G91)
        ≀
N100  G41  X150.  I50.  ;
N110  X100.  Y-100.  ;
        ≀
```

<Example 2>

Vector B is the compensation amount registered in compensation No. modal D2 of the N200 block.

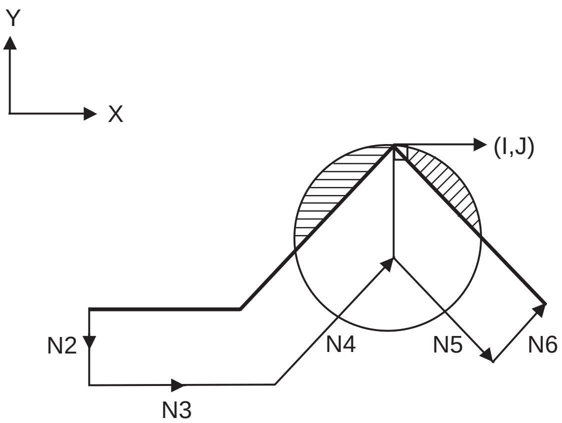



```
(G41  D1  G91)
        ≀
N200  G41  X150.  I50.  D2 ;
N210  X100.  Y-100.  ;
        ≀
```

Precautions

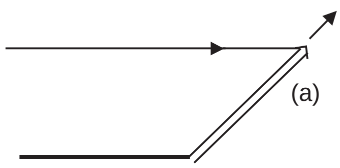
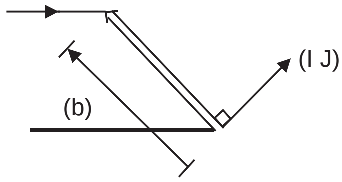
(1) Issue the I,J type vector in a linear mode (G0, G1). If it is in an arc mode at the start of compensation, program error (P151) will occur.
When it is in the offset mode as well as in the arc mode, I,J will be designated at the center of the circular.

(2) When the I,J type vector is designated, it will not be deleted (Interference avoidance) even if there is interference. Consequently, overcutting may occur.
In the figure below, cutting will occur in the shaded section.



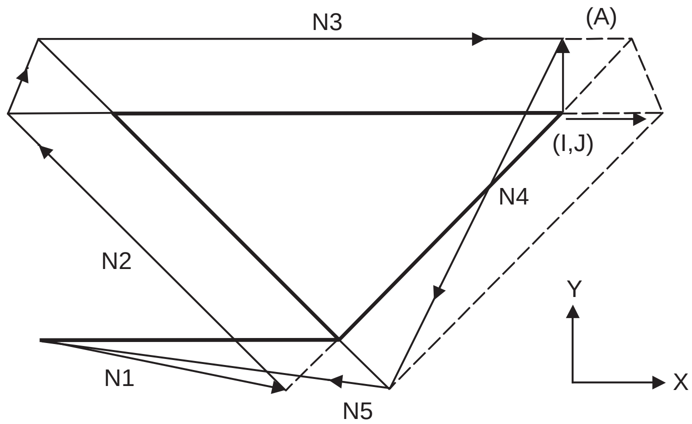

```
N1  G28  X0 Y0  ;
N2  G42  D1  F1000  ;
N3  G91  X100.  ;
N4  G42  X100.  Y100. I10.  ;
N5  X100.  Y-100.  ;
N6  G40  ;
N7  M02  ;
```

(3) The vectors differ for the G38 I _J_ (K_) command and the G41/G42 I_J_(K_) command.

	G38	G41/G42
Example	: (G41) : G38 G91 X100. I50. J50. ; :	: (G41) : G41 G91 X100. I50. J50. ; :
		
	Vector in IJ direction having a compensation amount (a) size	Vector perpendicular in IJ direction and having a compensation amount (b) size

(4) Refer to the following table for the compensation methods depend on the presence or absence of G41/ G42 command and I, K, (J) command.

G41/G42	I, J (K)	Compensation methods
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



N1 G91 G01 G41 X200. D1 F1000 ;
N2 X-150. Y150. ;
N3 G41 X300. I50. ;
N4 X-150. Y-150. ;
N5 G40 X-200. ;

During the I, J type vector compensation, the A insertion block will not exist.

12.3.4 Interrupts during Tool Radius Compensation

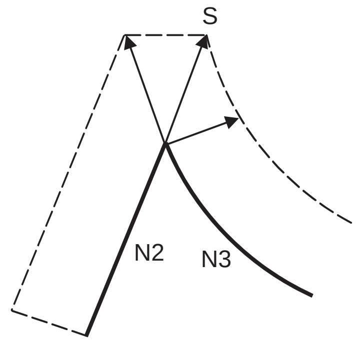
Detailed description

MDI interruption

Tool radius compensation is valid in any automatic operation mode - whether memory or MDI mode.
The figure below shows what happens by MDI interruption after stopping the block during memory mode.
S in the figure indicates the stop position with single block.

(1) Interrupt without movement (tool path does not change)

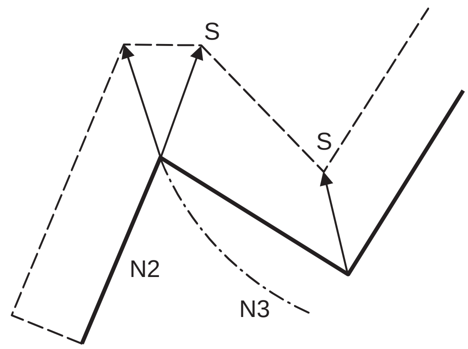
Automatic operation		MDI interruption
N1 G41 D1;		
N2 X20. Y50. ;		
	<---	S1000 M3;
N3 G3 X40. Y-40. R70. ;		



(2) Interrupt with movement
The compensation vectors are automatically re-calculated in the movement block after interrupt.

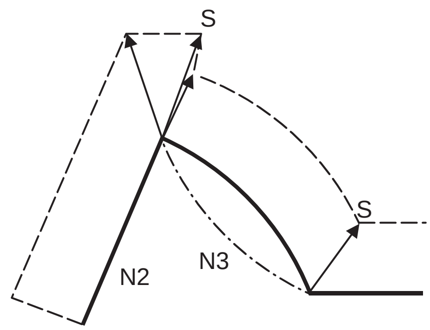
With linear interrupt

Automatic operation		MDI interruption
N1 G41 D1;		
N2 X20. Y50. ;		
	<---	X50. Y-30. ;
		X30. Y50. ;
N3 G3 X40.Y-40. R70. ;		



With circular interruption

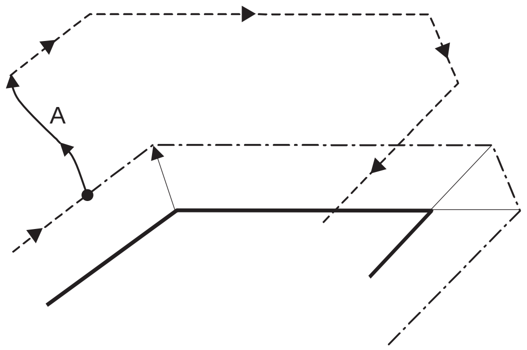
Automatic operation		MDI interruption
N1 G41 D1;		
N2 X20. Y50. ;		
	<---	G2 X40. Y-40. R70. ;
		G1 X40. ;
N3 G3 X40. Y-40. R70. ;		

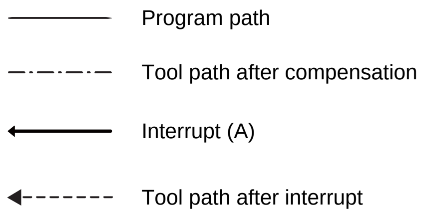


Manual interruption

(1) Interrupt with manual absolute OFF.

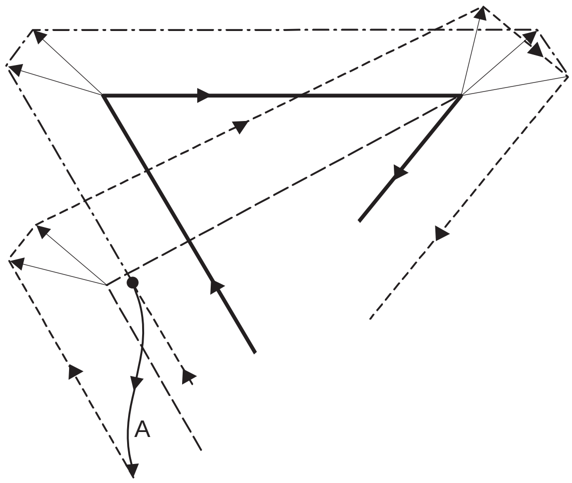
The tool path will deviate from the compensated path by the interrupt amount.

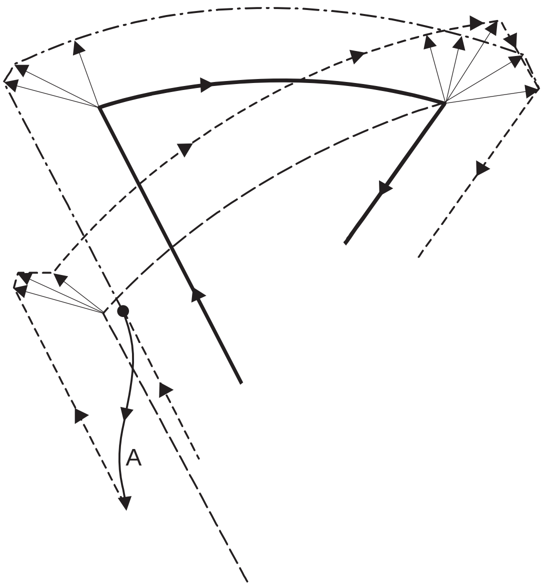




(2) Interrupt with manual absolute ON.

In the incremental value mode, the same operation will be performed as the 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.





Program path

Tool path after compensation

Interrupt (A)

Tool path after interrupt

12.3.5 General precautions for tool radius compensation

Precautions

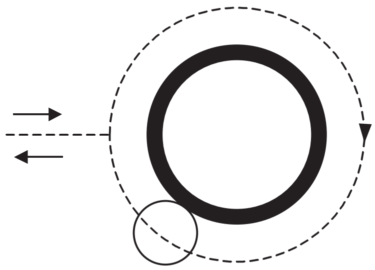
Assigning the compensation amounts

(1) 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 offset amounts for tool position offset.

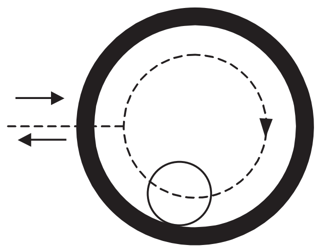
(2) Compensation amounts are normally changed when a different tool has been selected in the compensation cancel mode. However, when an amount is changed during the compensation mode, the vectors at the end point of the block are calculated using the compensation amount designated in that block.

Compensation amount symbols and tool center path

If the compensation 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 compensation amount is programmed as positive (+). However, if the tool path center is programmed as shown in (a) and the compensation 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.)



G41 offset amount (+) or G42 offset amount (-)
(a)



G41 offset amount (-) or G42 offset amount (+)
(b)

---- Tool center path

12.3.6 Changing of Compensation No. during Compensation Mode

Function and purpose

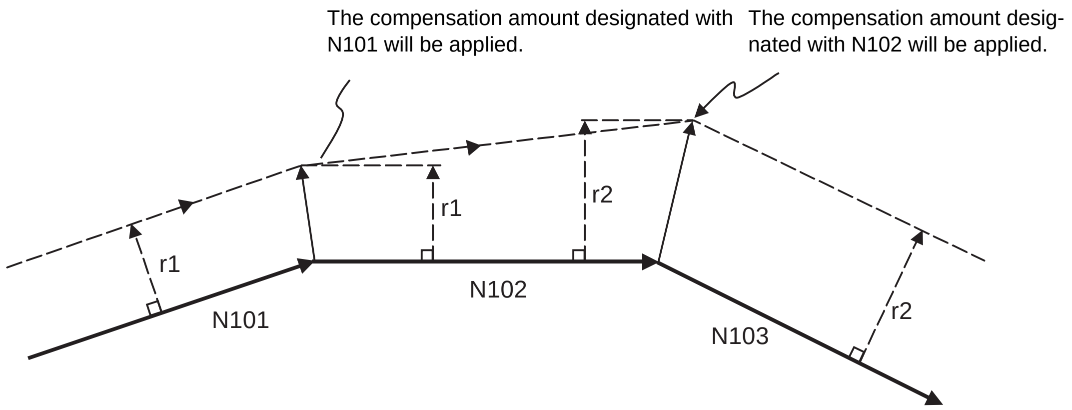
As a principle, the compensation No. must not be changed during the compensation mode. If changed, the movement will be as shown below.

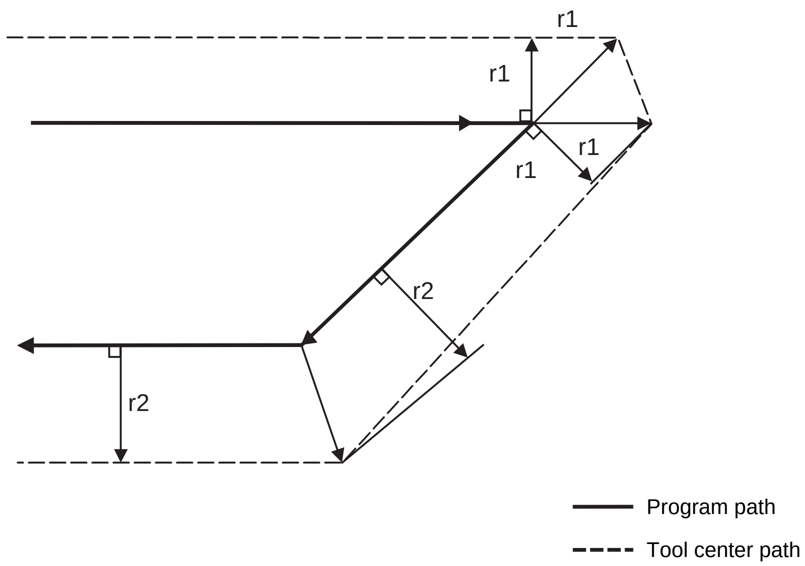
When compensation No. (compensation amount) is changed:

G41 G01 .... Dr1 ;
(α = 0,1,2,3)

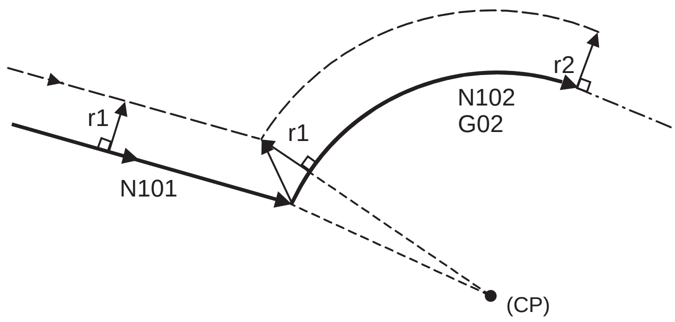
N101 G0α Xx1 Yy1 ;
N102 G0α Xx2 Yy2 Dr2 ; .... Compensation No. changed
N103 Xx3 Yy3 ;

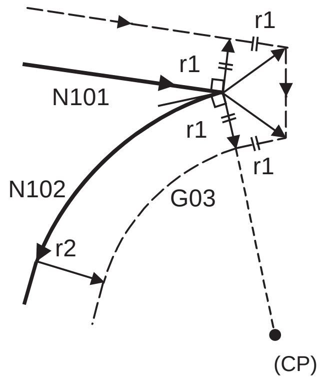
During linear -> linear





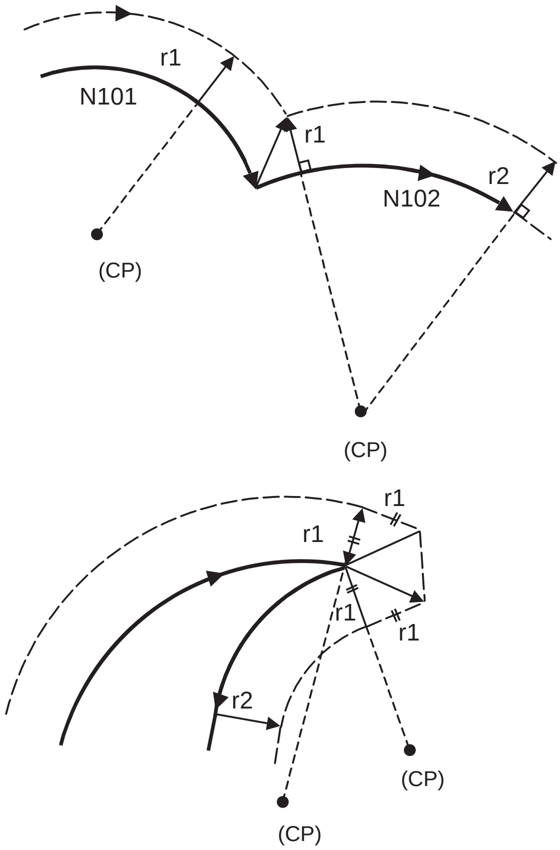
Linear ->circular





(CP) Arc center
Program path
Tool center path

Circular -> circular



(CP) Arc center
Program path
Tool center path

12.3.7 Start of Tool Radius Compensation and Z Axis Cut in Operation

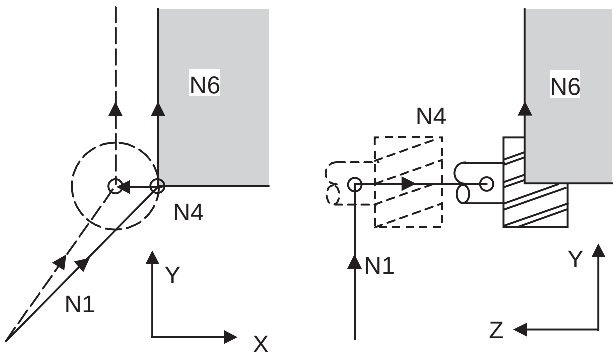
Function and purpose

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.

Program example

When the following type of program is created:

```
N1 G91 G00 G41 X500. Y500. D1 ;
N2 S1000 ;
N3 M3 ;
N4 G01 Z-300. F1 ;
N6        Y100. F2 ;
    :
    :
```

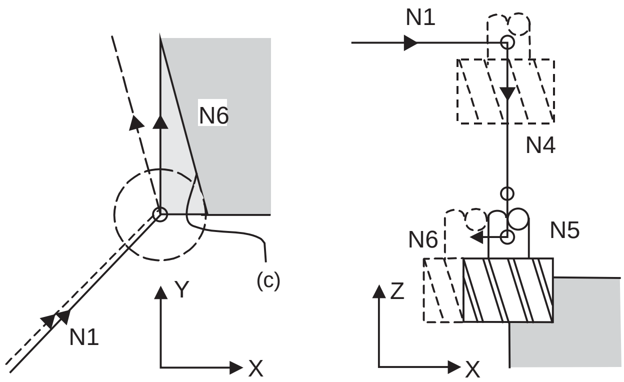


N4 Z axis lowers (1 block)

---- Tool center path

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 X500. Y500. D1 ;
N2 S1000 ;
N3 M3 ;
N4        Z-250. ;
N5 G01 Z-50. F1 ;
N6        Y100. F2 ;
```

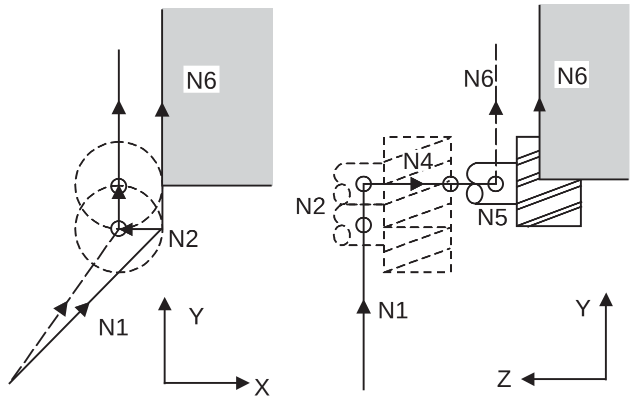


(c) Cut in

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.

```
N1 G91 G00 G41 X500. Y400. D1 ;
N2 Y100. S1000 ;
N3 M3 ;
N4        Z-250. ;
N5 G01 Z-50. F1 ;
N6        Y100. F2 ;
```



The movement is correctly compensated as the same direction as the N6 advance direction is commanded in N2.

12.3.8 Interference Check

Function and purpose

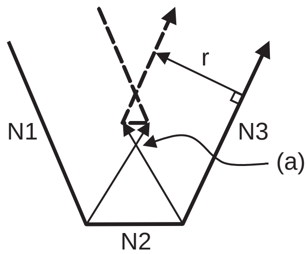
When tool radius is larger than the program path, a tool, compensated by the tool radius compensation function, may sometimes cut into the workpiece. This is known as interference, and interference check is the function which prevents this from occurring.
The table below shows the three functions of interference check and each can be selected for use by parameter.

	Function	Parameter		Operation
		#8102 COLL. ALM OFF	#8103 COLL. CHK OFF	
(1)	Interference check alarm function	0	0	Operation stops with a program error before executing a block which will cause cutting.
(2)	Interference check avoidance function	1	0	The tool path is changed to prevent cutting from occurring.
(3)	Interference check invalid function	0/1	1	Cutting continues as is, even if the workpiece is cut into. Use in the fine segment program.

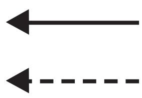
Detailed description

Conditions viewed as interference

When there is a movement command in three of the five pre-read blocks, and if the compensation calculation vectors which are created at the contacts of movement commands intersect each other, it will be viewed as interference.



r : Compensation amount   (a) Vectors intersect

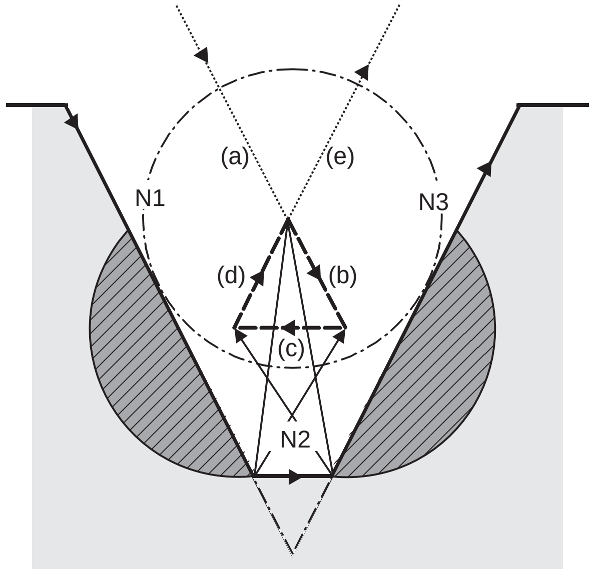


Program path

Tool center path

(Example 1)When operating a program including a short segment with a tool with a large radius
Cutting will occur in the shaded section.

(G41)
N1 G91 G1 X50. Y-100. ;
N2 X70. Y-100. ;
N3 X120. Y0 ;



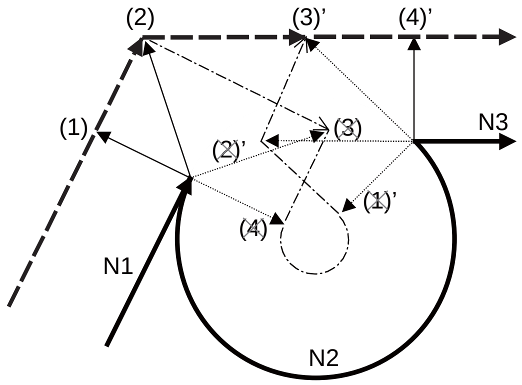
(1) With alarm function
An alarm is given before N1 is executed. The buffer correction function can thus be used to change N1 to the following, enabling machining to continue: N1 G1 X20. Y-40.;

(2) With avoidance function
The intersection of N1 and N3 is calculated to create interference avoidance vectors.
Tool center path is (a) -> (e).

(3) With interference check invalid function
The tool passes while cutting the N1 and N3 line.
Tool center path is (a)->(b)->(c)->(d)->(e) .

(Example 2) When operating a program including a small circular with a tool with a large radius

Cutting occurs near the start point/end point of the circular in the following figure.



Interference check processing

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)'

(1) With alarm function

The alarm occurs before N1 is executed.

(2) With avoidance function

With the above process, the vectors (1), (2), (3)' and (4)' will remain as the valid vectors. The tool center path will follow the path which connects these vectors, as the interference avoidance path.

(Thick broken line path)

(3) With interference check invalid function

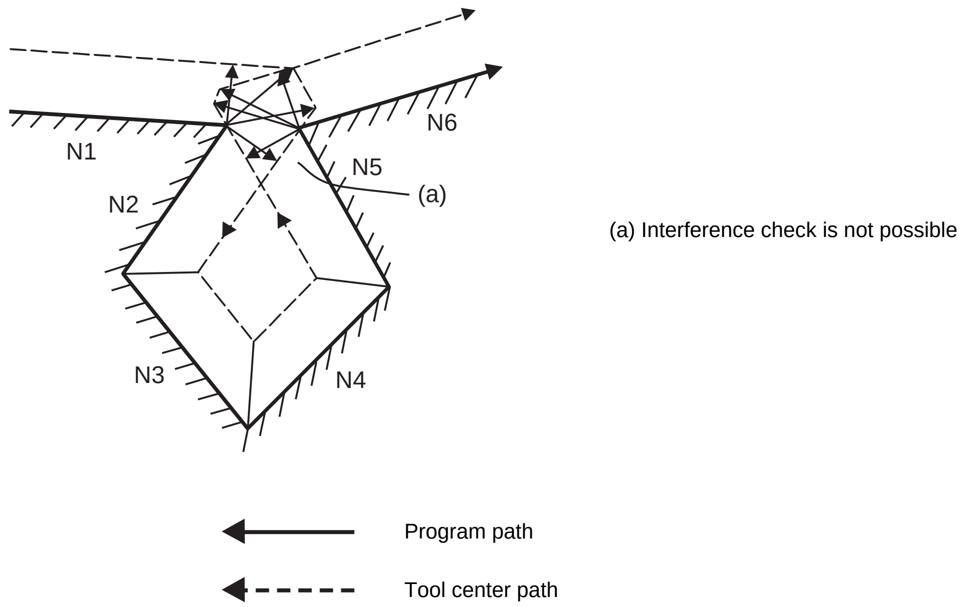
The tool center path will follow the path which connects (1), (2), (3), (4), (1)', (2)', (3)', (4)', as the interference avoidance path while cutting.

(Thin broken line path)

When interference check cannot be executed

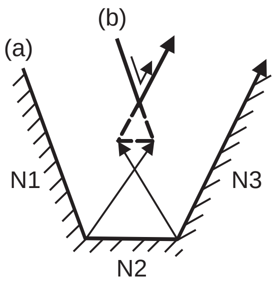
(1) When three of the movement command blocks cannot be pre-read
(When there are three or more blocks in the five pre-read blocks that do not have movement)

(2) When there is an interference following the fourth movement block

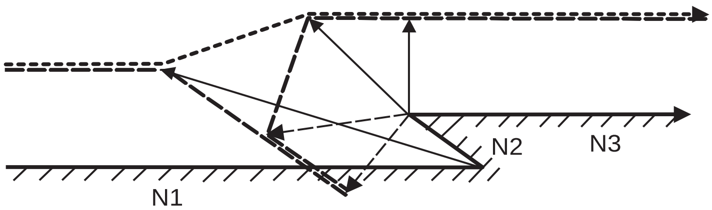


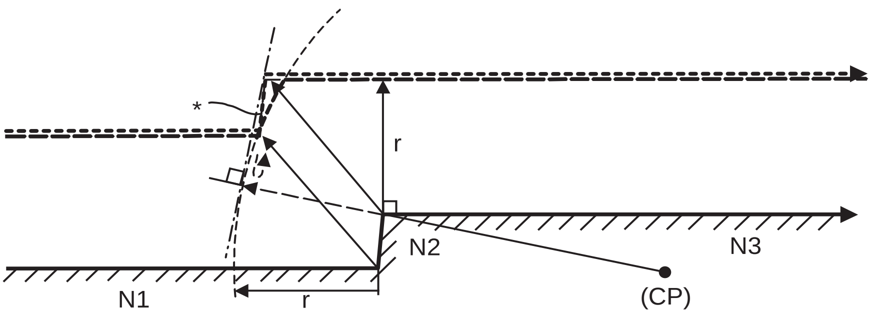
Operation when interference avoidance function is valid

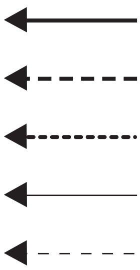
The movement will be as shown below when the interference avoidance check is valid.



(a) Program path (b) Tool center path







Program path

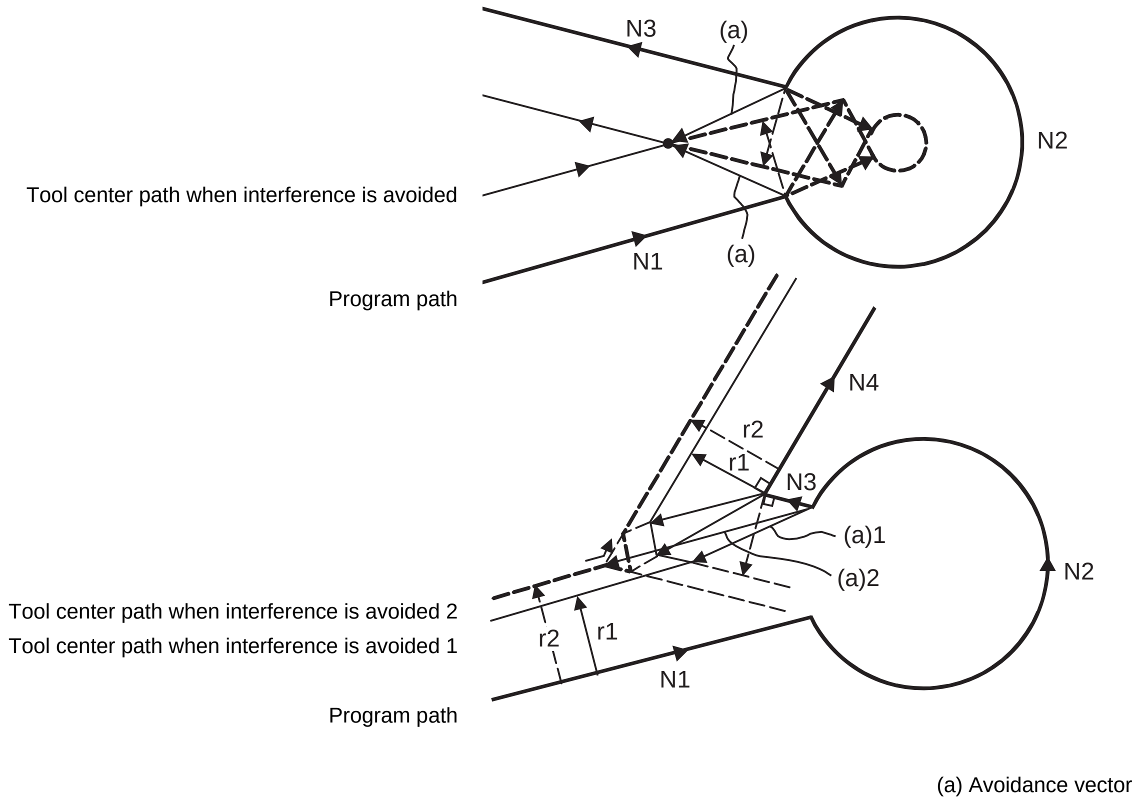
Tool center path without interference check

Tool center path when interference is avoided (*: Linear movement)

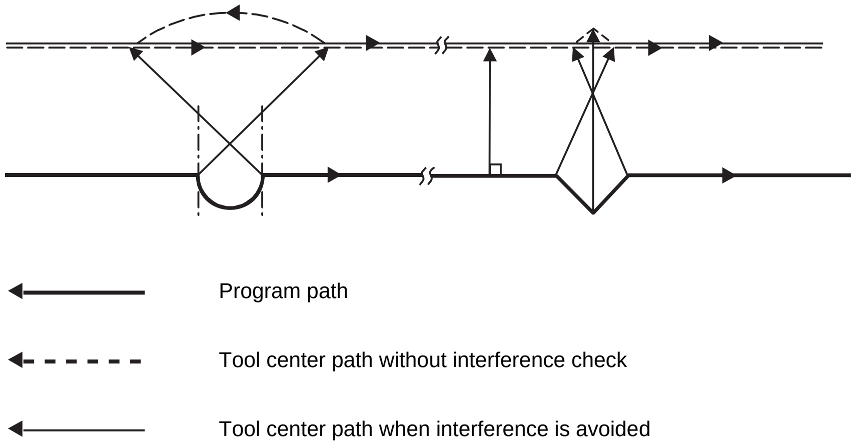
Valid vector

Invalid vector

If all of the line vectors for the interference avoidance are deleted, create a new avoidance vector as shown in below to avoid the interference.



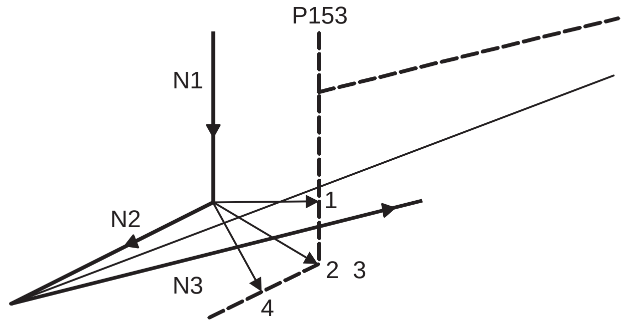
In the case of the figure below, the groove will be left uncut.



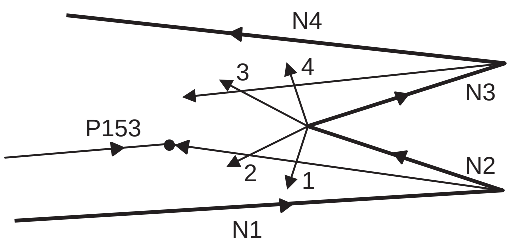
Interference check alarm operation

The interference check alarm occurs under the following conditions.

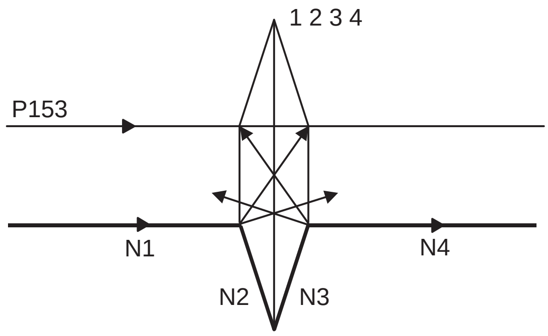
(1) When the interference check alarm function has been selected
When all the vectors at the end of its own block have been deleted.
When, as shown in the figure below, vectors 1 through 4 at the end point of the N1 block have all been deleted, program error (P153) will occur prior to N1 execution.



(2) When the interference check avoidance function has been selected
(Example 1) When there are valid vectors at the end point of the following blocks even when all the vectors at the end point of its own block have been deleted.
When, in the figure below, 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 and the operation stops.



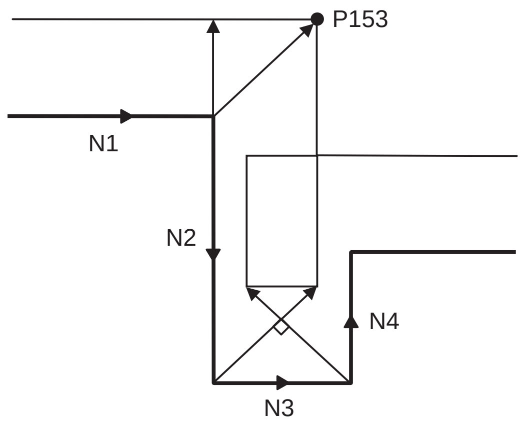
In the case shown in the figure below, the tool will move in the reverse direction at N2.
Program error (P153) now occurs before executing N1 and the operation stops.

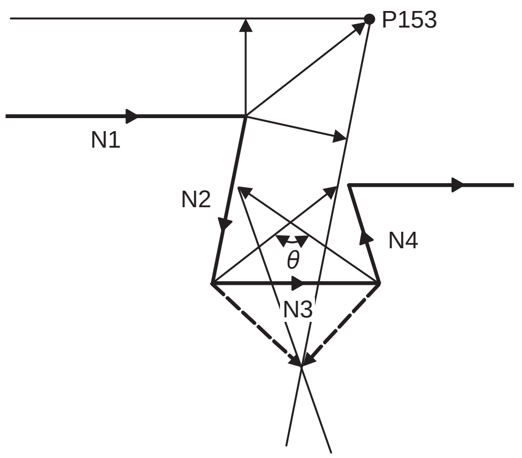


(Example 2) When avoidance vectors cannot be created

Even when, as in the figure below, the conditions for creating the avoidance vectors are satisfied, it may still be impossible to create avoidance vectors, or the interference vectors may interfere with N3.

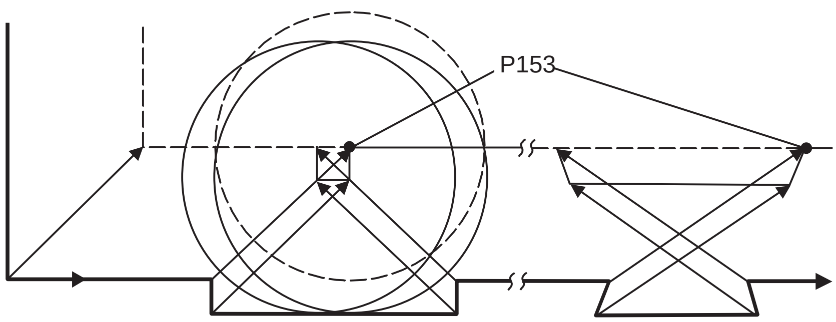
Program error (P153) will occur at the N1 end point when the vector intersecting angle is more than 90° and the operation will stop.





(Example 3) When the program advance direction and the advance direction after compensation are reversed

When grooves, narrower than the tool radius with parallel or widening bottom, are programmed, it will still be regarded as interference even if there is actually no interference.



12.4 Programmable Compensation Input ; G10,G11

Function and purpose

The tool offset and workpiece offset can be set or changed by the program 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

G90 (G91) G10 L2 P__ X__ Y__ Z__ ; ... Workpiece offset input

P	0 : External workpiece 1 : G54 2 : G55 3 : G56 4 : G57 5 : G58 6 : G59 Other than the above or P command is omitted: Currently selected workpiece coordinate system

(Note) The compensation 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.

G10 L10 P__ R__ ; ... Tool compensation input (For tool offset memory I)

P	Offset No.
R	Offset amount

G10 L10 P__ R__ ; ... Tool compensation input (For tool offset memory II) Tool length compensation shape compensation

G10 L11 P__ R__ ; ... Tool compensation input (For tool offset memory II) Tool length compensation wear compensation

G10 L12 P__ R__ ; ... Tool compensation input (For tool offset memory II) Tool radius shape compensation

G10 L13 P__ R__ ; ... Tool compensation input (For tool offset memory II) Tool radius wear compensation

G11 ; ... Compensation input cancel

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 G21,G22, G54 to G59, G90 or G91.

(4) Normally, a program error (P45) occurs when G10/G11 and a fixed cycle are commanded in a same block. When the parameter "#1241 set13/bit0 No G-CODE COMB. Error" is ON, the program error can be avoided but the fixed cycle command will be ignored.

(5) The workpiece offset input command (L2 or L20) should not be issued in the same block as the tool compensation input command (L10).

(6) If an illegal L No. or offset 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 can be omitted when the workpiece offset is input.

(11) When the P command is omitted, it will be handled as the currently selected workpiece compensation input.

Program example

(1) Input the compensation amount

```
······; G10 L10 P10 R-12345 ; G10 L10 P05 R98765 ; G10 L10 P30 R2468 ; ······
```

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 P10 R-500 ;	(The mode is the G91 mode, so -500 is added.)
N4 G01 G90 G43 Z-100000 H10 ;	(Z=-101500)

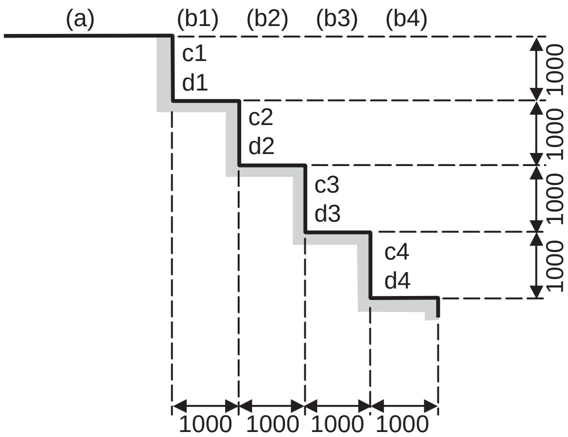
(Example 2) Assume that H10 = -1000 is already set.

Main program

N1 G00 X100000 ;	a
N2 #1=-1000 ;	
N3 M98 P1111 L4 ;	b1, b2, b3, b4

Subprogram O1111

N1 G01 G91 G43 Z0 H10 F100 ;	c1, c2, c3, c4
G01 X1000;	d1, d2, d3, d4
#1=#1-1000 ;	
G90 G10 L10 P10 R#1 ;	
M99 ;	



(Note)Final offset amount will be H10= -5000.

(Example 3) The program for Example 2 can also be written as follows.

Main program

N1 G00 X100000 ;
N2 M98 P1111 L4 ;

Subprogram O1111

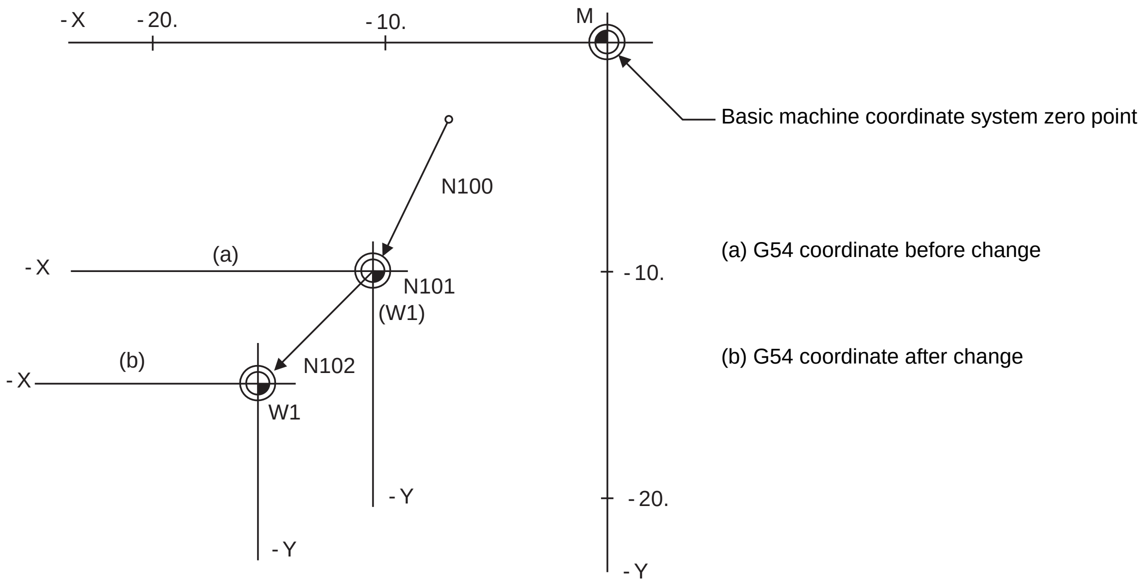
N1 G01 G91 G43 Z0 H10 F100 ;
N2 G01 X1000 ;
N3 G10 L10 P10 R-1000 ;
N4 M99 ;

(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.

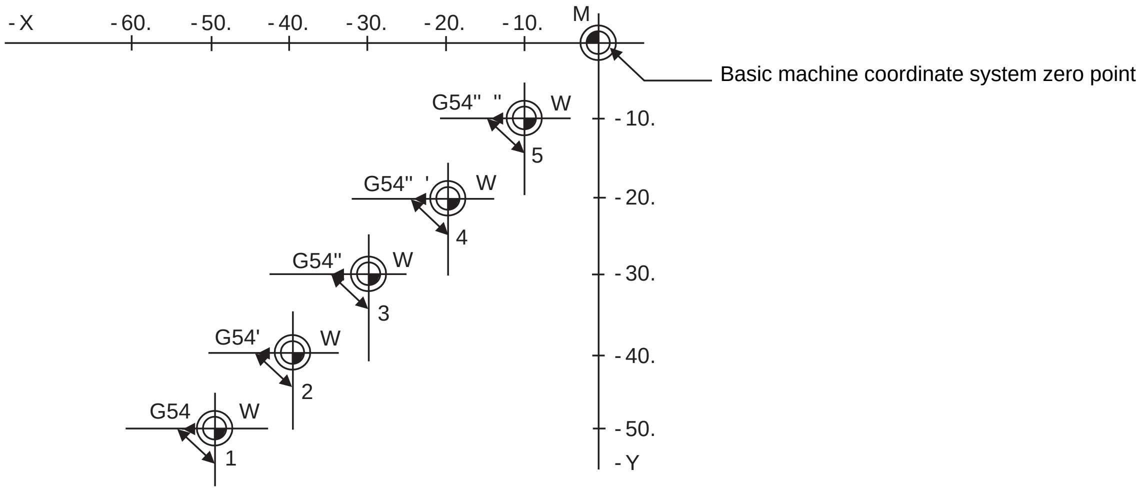
X=0 Y=0	->	X=+5.000 Y=+5.000

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. ;
	M98 P200 L5 ;
	M02 ;
	%
Subprogram O200	N1 G90 G54 G10 L2 P1 X#1 Y#1 ;
	N2 G00 X0 Y0 ;
	N3 X-5. F100 ;
	N4 X0 Y-5. ;
	N5 Y0 ;
	N6 #1=#1+#2 ;
	N7 M99 ;
	%



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 P10 R-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.

13

Program Support Functions

13.1 Fixed cycles

Function and purpose

These fixed cycles are used to perform prepared sequences of machining programs, such as positioning, hole drilling, boring and tapping in a block. The machining sequences available are listed in the table below.
By editing the standard fixed cycle subprograms, the fixed cycle sequences can be changed by the user. The user can also register and edit an original fixed cycle program. For the standard fixed cycle subprograms, refer to the list of the fixed cycle subprograms in the appendix of the operation manual. The list of fixed cycle functions for this control unit is shown below.

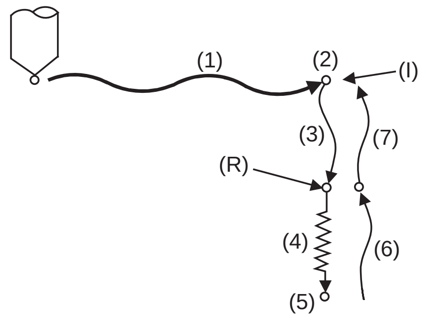
G code	Hole machining start (-Z direction)	Operation at hole bottom		Return operation (+Z direction)	Application
		Dwell	Spindle		
G80	-	-	-	-	Cancel
G81	Cutting feed	-	-	Rapid traverse	Drill, spot drilling cycle
G82	Cutting feed	Yes	-	Rapid traverse	Drill, counter boring cycle
G83	Intermittent feed	-	-	Rapid traverse	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 traverse	Boring cycle
G87	Rapid traverse	-	Forward rotation	Cutting feed	Back boring cycle
G88	Cutting feed	Yes	Stop	Rapid traverse	Boring cycle
G89	Cutting feed	Yes	-	Cutting feed	Boring cycle
G73	Intermittent feed	Yes	-	Rapid traverse	Stepping cycle
G74	Cutting feed	Yes	Forward rotation	Cutting feed	Reverse tapping cycle
G76	Cutting feed	-	Oriented spindle stop	Rapid traverse	Fine boring cycle

A fixed cycle mode can be canceled by G80 command and G command in the 01 group. At the same time, various other data will also be cleared to zero.

Detailed description

Basic operations of fixed cycle for drilling

There are 7 actual operations which are each described below.



(1) This indicates the X and Y axes positioning, and executes positioning with G00.

(2) This is an operation done after positioning is completed (at the initial point), and when G87 is commanded, the M19 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.

(3) The tool is positioned to the R point by rapid traverse.

(4) Hole machining is conducted by cutting feed.

(5) This operation takes place at the hole bottom position, and depending on the fixed cycle mode, the operation can be the spindle stop (M05), the rotary tool reverse rotation (M04), rotary tool forward rotation (M03), dwell or tool shift.

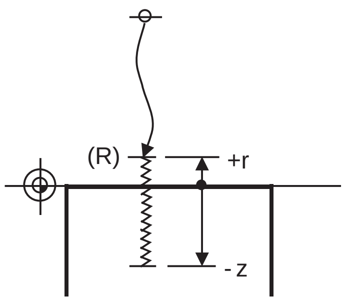
(6) The tool is retracted to the R point at the cutting feed or the rapid traverse rate, depending on the fixed cycle mode.

(7) The tool is returned to the initial point at rapid traverse rate.

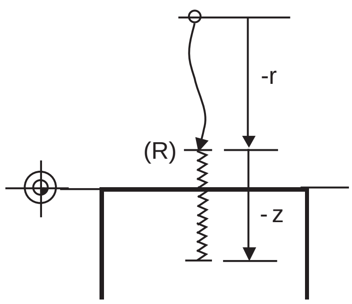
(Note) Whether the fixed cycle is to be completed at operation 6 or 7 can be selected by G98/G99 G commands. (Refer to "Initial point and R point level return; G98, G99")

Difference between absolute value command and incremental value command

For absolute value



For incremental value



(R) R point

Positioning plane and hole drilling axis

The fixed 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 commands, and the hole drilling axis is the axis perpendicular (X, Y, Z or their 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.
An arbitrary 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.

(Example 1) When G17 (X-Y plane) is selected, and the axis parallel to the Z axis is set as the W axis.
G81 ....Z_ ; The Z axis is used as the hole drilling axis.
G81 ......W_ ; The W 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 of the hole drilling axis must be carried out with the fixed cycle canceled.

In the following explanations on the movement in each fixed 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 commands the in-position width for commanding the fixed cycle from the machining program. The commanded in-position width is valid only in the five fixed cycles; G81 (drill, spot drill), G82 (drill, counter boring), G83 (deep drill cycle), G84 (tap cycle) and G74 (reverse tap cycle). 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 for positioning axis (position error amount)	0.001 to 999.999 (mm)	If a value exceeding the command range is commanded, a program error (P35) will occur.
,J	In-position width for drilling axis (position error amount)		

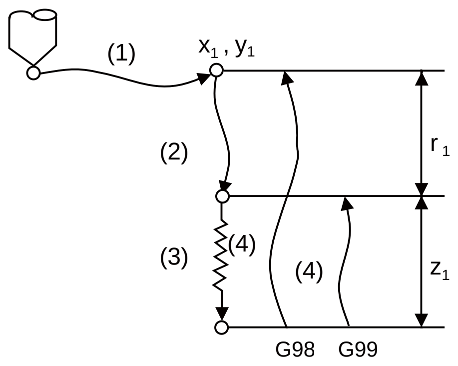
13.1.1 Drilling, spot drilling ; G81

Command format

G81 Xx1 Yy1 Zz1 Rr1 Ff1 Ll1,Ii1,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G0 Xx1 Yy1
(2)	-	Invalid	G0 Zr1
(3)	-	Invalid	G1 Zz1 Ff1
(4)	-	Valid	G98 mode G0Z - (z1+r1) G99 mode G0Z - z1

The operation stops at after the (1), (2) and (4) commands during single block operation.

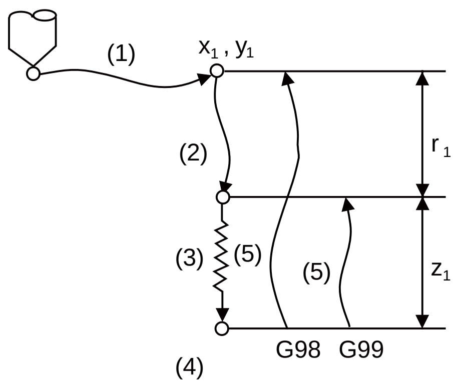
13.1.2 Drilling, counter boring ; G82

Command format

G82 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1 ,Ii1 ,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G0 Xx1 Yy1
(2)	-	Invalid	G0 Zr1
(3)	-	Invalid	G1 Zz1 Ff1
(4)	-	-	G4Pp1(Dwell)
(5)	-	Valid	G98 mode G0Z - (z1+r1) G99 mode G0Z - z1

The operation stops at after the (1), (2) and (5) commands during single block operation.

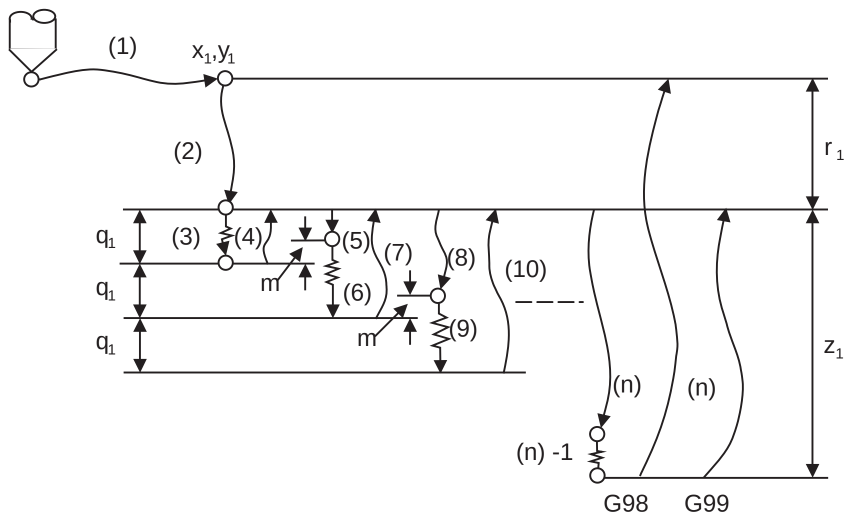
13.1.3 Deep hole drilling cycle ; G83

Command format

G83 Xx1 Yy1 Zz1 Rr1 Qq1 Ff1 Ll1 ,Ii1 ,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Qq1	Cut amount for each cutting pass (incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G0 Xx1 Yy1
(2)	-	Invalid	G0 Zr1
(3)	-	Invalid	G1 Zq1 Ff1
(4)	-	Invalid	G0 Z-q1
(5)	-	Invalid	G0 Z(q1-m)
(6)	-	Invalid	G1 Z(q1+m) Ff1
(7)	-	Invalid	G0 Z -2*q1
(8)	-	Invalid	G0 Z (2*q1-m)
(9)	-	Invalid	G1 Z(q1+m) Ff1
(10)	-	Invalid	G0 Z-3*q1
:			
(n)-1	-	Invalid	
(n)	-	Valid	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1

When executing a second and following cuttings in the G83 as shown above, the movement will change from rapid traverse to cutting feed "m" mm before the position machined last. After reaching the hole bottom, the axis will return according to the G98 or G99 mode.
"m" will differ according to the parameter "#8013 G83 return". Program so that q1 > m.
The operation stops at after the (1), (2) and (n) commands during single block operation.

13.1.4 Tapping cycle ; G84

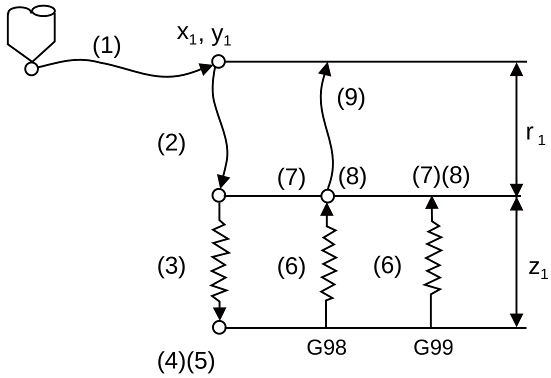
Command format

G84 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1,Rr2 Ss1 ,Ss2 Ll1 ,Ii1 ,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Z-axis feed amount (tapping pitch) per spindle rotation (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
,Rr2	Synchronization method selection (r2=1 synchronous, r2=0 asynchronous) (When omitted, the mode will follow the setting of parameter "#1229/bit4 synchronous tap")
Ss1	Spindle rotation speed command (Note 1) At a synchronous tapping mode, "Sn = *****" type S command will be ignored. (n:spindle number, *****: rotation speed) (Note 2) If an S command is issued during synchronous tapping modal, a program error (P186) will occur.
,Ss2	Spindle rotation speed during return
Ll1	Designation of number of repetitions (0 to 9999). When ″0″ is set, no execution
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

During the asynchronous tapping mode, F address is regarded as cutting federate.

Detailed description



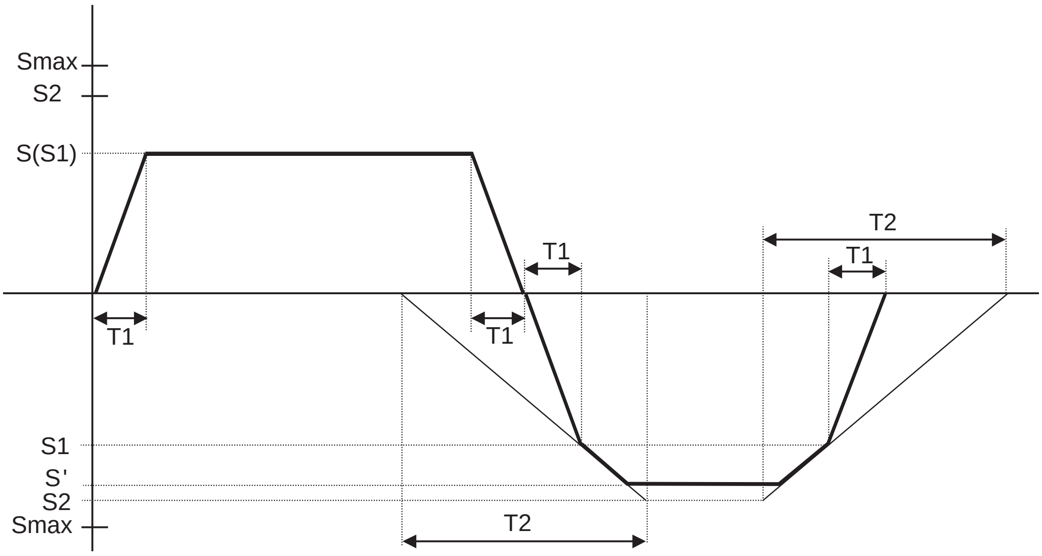
Operation pattern	i1	j1	Program
(1)	Valid	-	G0 Xx1 Yy1
(2)	-	Invalid	G0 Zr1
(3)	-	Invalid	G1 Zz1 Ff1
(4)	-	-	G4 Pp1
(5)	-	-	M4 (Spindle reverse rotation)
(6)	-	Invalid	G1 Z-z1 Ff1
(7)	-	-	G4 Pp1
(8)	-	-	M3 (Spindle forward rotation)
(9)	-	Valid	G98 mode G0 Z-r1 G99 mode No movement

When r2 = 1, the synchronous tapping mode will be applied, and when r2 = 0, the asynchronous tapping mode will be applied. If there is no r2 command, the mode will follow the parameter setting.
When G84 is being executed, the override will be canceled and the override will automatically be set to 100%.
Dry run is valid for the positioning command when the control parameter "G00 DRY RUN" is on. If the feed hold button is pressed during G84 execution, the movement will not stop immediately, and instead, will stop after (6) during sequences (3) to (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.

Spindle acceleration/deceleration pattern during synchronous tapping

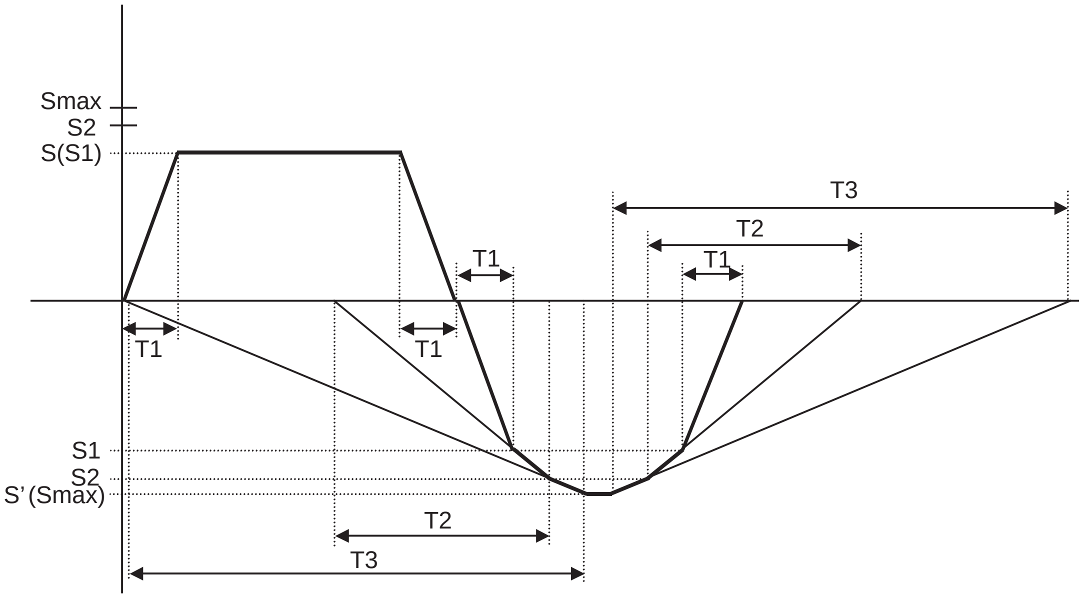
This function enables to make spindle acceleration/deceleration pattern closer to that of the speed loop by dividing the spindle and drilling axis acceleration/deceleration pattern into up to three stages during synchronous tapping.
The acceleration/deceleration pattern can be set up to three stages for each gear.
When returning from the hole bottom, rapid return is possible at the spindle rotation speed during return. The spindle rotation speed during return is held as modal information.

(1) When tap rotation speed < spindle rotation speed during return <= synchronous tapping changeover spindle rotation speed 2



S	Command spindle rotation speed
S'	Spindle rotation speed during return
S1	Tapping rotation speed (spindle basic specification parameters #3013 to #3016)
S2	Synchronous tapping changeover spindle rotation speed 2 (spindle basic specification parameters #3037 to #3040)
Smax	Maximum rotation speed (spindle basic specification parameters #3005 to #3008)
T1	Tapping time constant (spindle basic specification parameters #3017 to #3020)
T2	Synchronous tapping changeover time constant 2 (spindle basic specification parameters #3041 to #3044)

(2) When synchronous tapping changeover spindle rotation speed 2 < spindle rotation speed during return



S	Command spindle rotation speed
S'	Spindle rotation speed during return
S1	Tapping rotation speed (spindle basic specification parameters #3013 to #3016)
S2	Synchronous tapping changeover spindle rotation speed 2 (spindle basic specification parameters #3037 to #3040)
Smax	Maximum rotation speed (spindle basic specification parameters #3005 to #3008)
T1	Tapping time constant (spindle basic specification parameters #3017 to #3020)
T2	Synchronous tapping changeover time constant 2 (spindle basic specification parameters #3041 to #3044)
T3	Synchronous tapping changeover time constant 3 (spindle basic specification parameters #3045 to #3048)

Feedrate for tapping cycle and tapping return

The feedrates for the tapping cycle and tapping return are as shown below.

(1) Selection of synchronous tapping cycle/asynchronous tapping cycle

<table>
<tr><th><Program>
G84…, Rxx</th><th>Control parameter
Synchronous tapping</th><th>Synchronous/asyn-
chronous</th></tr>
<tr><td>,R00</td><td>-</td><td rowspan="2">Asynchronous</td></tr>
<tr><td rowspan="2">,Rxx
No designation</td><td>OFF</td></tr>
<tr><td>ON</td><td rowspan="2">Synchronous</td></tr>
<tr><td>,R01</td><td>-</td></tr>
</table>

- is irrelevant to the setting

(2) Selection of asynchronous tapping cycle feedrate

<table>
<tr><th>G94/G95</th><th>Control parameter
F1-digit valid</th><th>F command value</th><th>Feed designation</th></tr>
<tr><td rowspan="3">G94</td><td>OFF</td><td>F designation</td><td rowspan="2">Feed per minute</td></tr>
<tr><td rowspan="2">ON</td><td>Other than F0 to F8</td></tr>
<tr><td>F0 to F8 (no decimal point)</td><td>F1-digit feed</td></tr>
<tr><td>G95</td><td>-</td><td>F designation</td><td>Feed per revolution</td></tr>
</table>

- is irrelevant to the setting

(3) 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 spindle rotation speed, the spindle rotation speed value will be valid even during return. If the spindle rotation speed is not 0 during return, the tapping retract override value will be invalid.

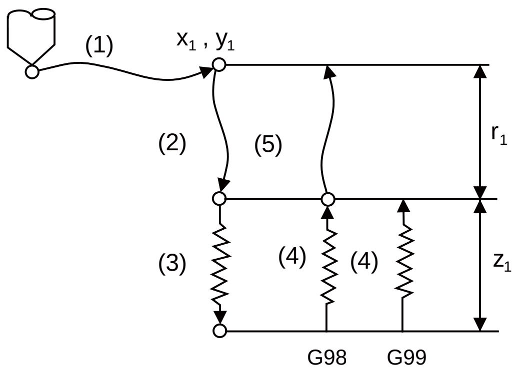
13.1.5 Boring ; G85

Command format

G85 Xx1 Yy1 Zz1 Rr1 Ff1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution

Detailed description



Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zz1 Ff1
(4)	G1 Z-z1 Ff1
(5)	G98 mode G0Z-r1 G99 mode No movement

The operation stops at after the (1), (2) and (4) or (5) commands during single block operation.

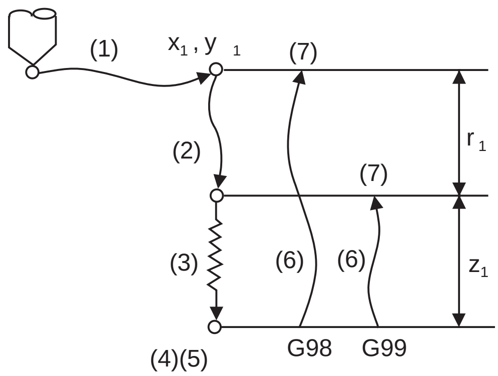
13.1.6 Boring ; G86

Command format

G86 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1 ;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution

Detailed description



Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zz1 Ff1
(4)	G4 Pp1
(5)	M5 (spindle stop)
(6)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1
(7)	M3 (Spindle forward rotation)

The operation stops at after the (1), (2) and (7) commands during single block operation.

13.1.7 Back boring ; G87

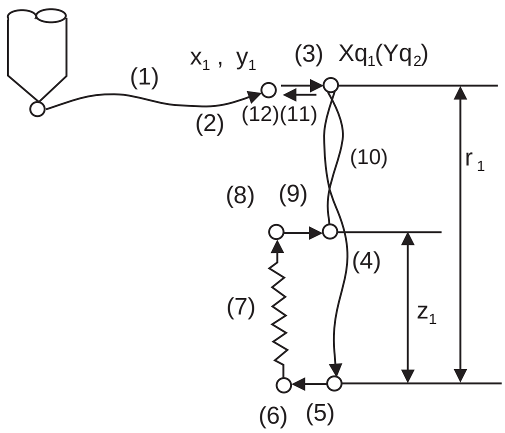
Command format

G87 Xx1 Yy1 Zz1 Rr1 Iq1 Jq2 Kq3 Ff1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Iq1 Jq2 Kq3	Designation of shift amount (incremental value) (modal) The command addresses for each plane selection are as follow; G17 plane: IJ G18 plane: KI G19 plane: JK Depending on the parameter setting, the shift amount can be designated by Q address. Refer to "Designation of shift amount (I,J,K)".
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution

(Note) Take care to the z1 and r1 designations. (The signs of z1 and r1 must be opposite) There is no R point return.

Detailed description

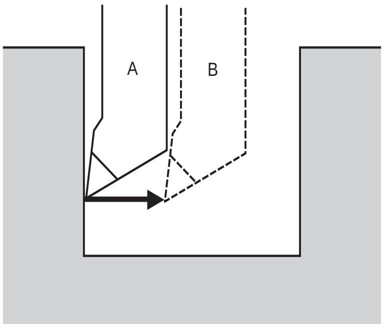


Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	M19 (Spindle orientation)
(3)	G0 Xq1 (Yq2) (shift)
(4)	G0 Zr1
(5)	G1 X-q1(Y-q2)Ff1 (shift)
(6)	M3 (Spindle forward rotation)
(7)	G1 Zz1 Ff1
(8)	M19 (Spindle orientation)
(9)	G0 Xq1 (Yq2) (shift)
(10)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - (r1+z1)
(11)	G0 X-q1(Y-q2) (shift)
(12)	M3 (Spindle forward rotation)

The operation stops at after the (1), (4), (6) and (11) commands during single block operation.

Designation of shift amount (I,J,K)

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.



A: Tool position during cutting
B: Tool position when positioning to the hole bottom and, also, when escaping after cutting

The command addresses to designate the shift amount for each plane selection are as follow;
G17 plane: IJ
G18 plane: KI
G19 plane: JK

The shift amount is executed with linear interpolation, and the feedrate 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 modal during the fixed 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 sign for the Q value is ignored and the value is handled as a positive value. The Q value is a modal during the fixed cycle, and will also be used as the G83, G73 and G76 cutting amount.

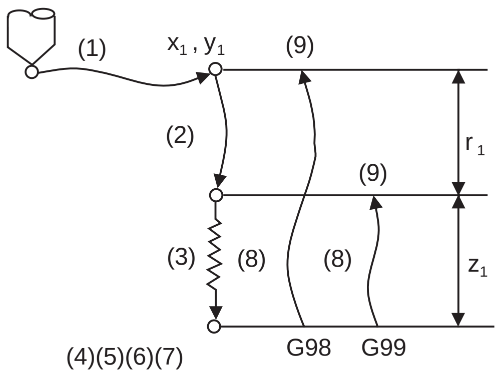
13.1.8 Boring ; G88

Command format

G88 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution

Detailed description



Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zz1 Ff1
(4)	G4 Pp1
(5)	M5 (Spindle stop)
(6)	Stop when single block stop switch is ON
(7)	Automatic start switch ON
(8)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1
(9)	M3 (Spindle forward rotation)

The operation stops at after the (1), (2), (6) and (9) commands during single block operation.

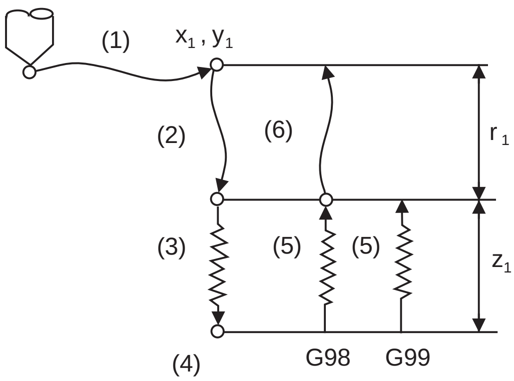
13.1.9 Boring ; G89

Command format

G89 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution

Detailed description



Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zz1 Ff1
(4)	G4 Pp1
(5)	G1 Z-z1 Ff1
(6)	G98 mode G0 Z - r1 G99 mode No movement

The operation stops at after the (1), (2) and (5) or (6) commands during single block operation.

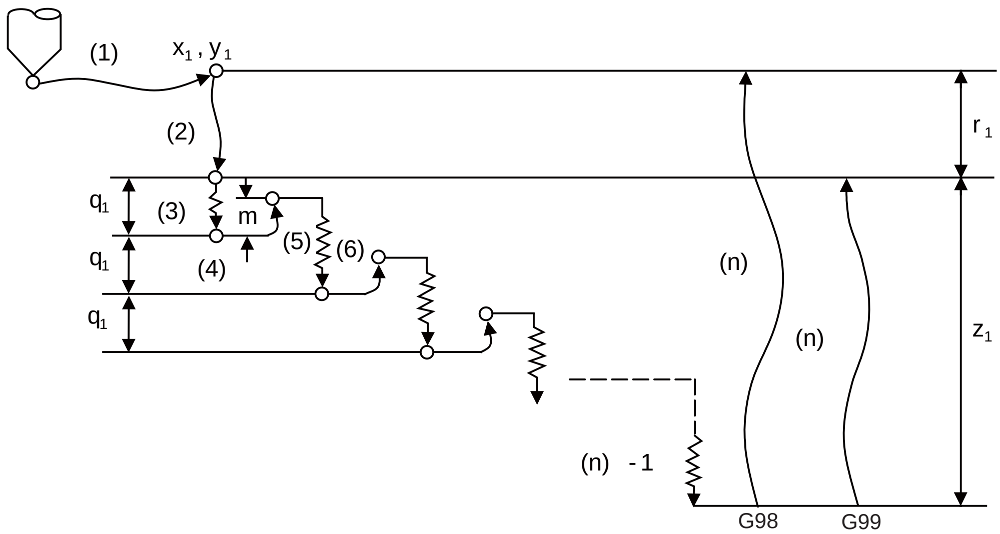
13.1.10 Stepping cycle ; G73

Command format

G73 Xx1 Yy1 Zz1 Qq1 Rr1 Ff1 Pp1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Qq1	Cut amount for each cutting pass (incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution

Detailed description



Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zq1 Ff1
(4)	G4 Pp1
(5)	G0 Z-m
(6)	G1 Z(q1+m) Ff1
:	
(n)-1	
(n)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1

When executing a second and following cutting in the G73 as shown above, the movement will return several "m" mm 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.

13.1.11 Reverse tapping cycle ; G74

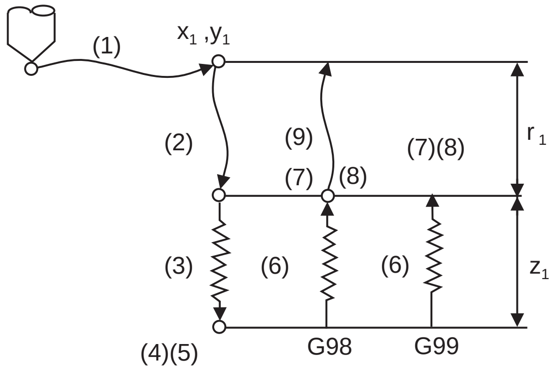
Command format

G74 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 ,Rr2 Ss1 ,Ss2 Ll1 ,Ii1,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Z-axis feed amount (tapping pitch) per spindle rotation (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
,Rr2	Synchronization method selection (r2=1 synchronous, r2=0 asynchronous) (modal) (When omitted, the mode will follow the setting of parameter "#1229/bit4 synchronous tap")
Ss1	Spindle rotation speed (Note) At a synchronous tapping mode, "Sn = *****" type S command will be ignored. (n:spindle number, *****: rotation speed) (Note) If an S command is issued during synchronous tapping modal, a program error (P186) will occur.
,Ss2	Spindle rotation speed during return
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

(Note) During the asynchronous tapping mode, F address is regarded as cutting federate.

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G0 Xx1 Yy1
(2)	-	Invalid	G0 Zr1
(3)	-	Invalid	G1 Zz1 Ff1
(4)	-	-	G4 Pp1
(5)	-	-	M3 (Spindle forward rotation)
(6)	-	Invalid	G1 Z-z1 Ff1
(7)	-	-	G4 Pp1
(8)	-	-	M4 (Spindle reverse rotation)
(9)	-	Valid	G98 mode G0 Z-r1 G99 mode No movement

When r2 = 1, the synchronous tapping mode will be applied, and when r2 = 0, the asynchronous tapping mode will be applied. If there is no r2 command, mode will follow the parameter setting.
When G74 is executed, the override will be canceled and the override will automatically be set to 100%.
Dry run is valid for the positioning command when the parameter "#1085 G00 Drn" is set to "1".
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.

Spindle acceleration/deceleration pattern during synchronous tapping

Refer to "Tapping cycle; G84".

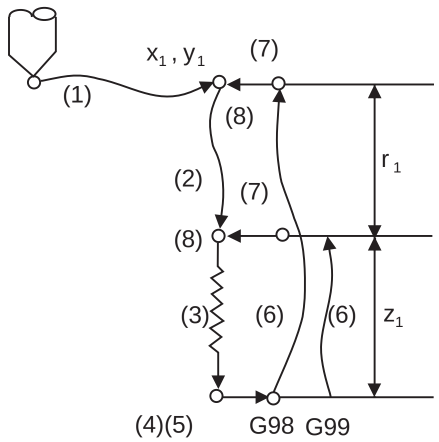
13.1.12 Fine boring ; G76

Command format

G76 Xx1 Yy1 Zz1 Rr1 Iq1 Jq2 Kq3 Ff1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Iq1 Jq2 Kq3	Designation of shift amount (incremental value) (modal) The command addresses for each plane selection are as follow; G17 plane: IJ G18 plane: KI G19 plane: JK Depending on the parameter setting, the shift amount can be designated by Q address. Refer to "Designation of shift amount (I,J,K)".
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution

Detailed description

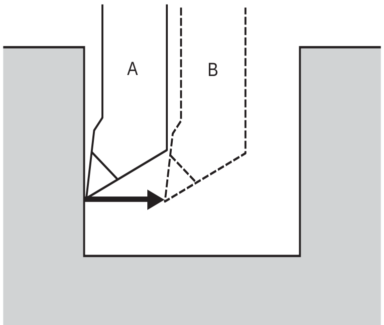


Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zz1 Ff1
(4)	M19 (Spindle orientation)
(5)	G1 Xq1 (Yq2)Ff1 (shift)
(6)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1
(7)	G0 X-q1 (Y-q2) (shift)
(8)	M3 (Spindle forward rotation)

The operation stops at after the (1), (2) and (7) commands during single block operation.

Designation of shift amount (I,J,K)

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.



A: Tool position during cutting
B: Tool position when escaping after cutting

The command addresses to designate the shift amount for each plane selection are as follow;
G17 plane: IJ
G18 plane: KI
G19 plane: JK

The shift amount is executed with linear interpolation, and the feedrate 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 modal during the fixed 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 IGNR" and "#8208 G76/87 (-)". The sign for the Q value is ignored and the value is handled as a positive value.
The Q value is a modal during the fixed cycle, and will also be used as the G83, G87 and G73 cutting amount.

13.1.13 Precautions for using a fixed cycle

Precautions

(1) Before the fixed 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 fixed 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 fixed cycle mode, the hole drilling operation will be executed. If there is no data, the hole will not be drilled.
Note that even when the X axis data exists, the hole will not be drilled if the data is a dwell (G04) time command.

(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 fixed 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 fixed cycle, the following will occur.

m = 00 to 03, 33 n = Fixed cycles

Gm Gn X_Y_Z_R_Q_P_L_F_;

Gm	: Execution	Gn	: Ignore		
X_Y_Z	: Execution	R_Q_P_L	: Ignore	F	: Record

Note that for the G02 and G03 commands, R will be handled as the arc radius.

(5) If M00 or M01 is commanded in a same block with a fixed cycle or during a fixed cycle mode, the fixed cycle will be ignored. Instead, M00 and M01 will be output after positioning. The fixed cycle is executed if X, Y, Z or R is commanded.

(6) If an M function is commanded in the same block as the fixed 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 designation of No. of times, the above control will be executed only for the first drilling.

(7) If another control axis (ex. rotary axis, additional axis) is commanded in the same block as the fixed cycle control axis, the fixed cycle will be executed after the other control axis is moved.

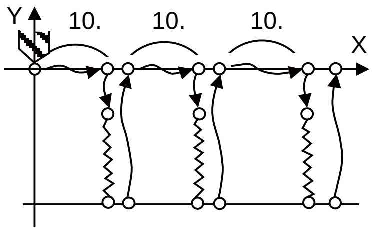
(8) If the No. of repetitions L is not designated, L1 will be set. If L0 is designated in the same block as the fixed cycle G code command, the hole machining data will be recorded, but the hole machining will not be executed.
(Example) G73 X __ Y __ Z __ R __ Q __ P __ F __ L0 __ ;
Memorize only the codes with an execution address

(9) When the fixed cycle is executed, the modal command commanded in the fixed cycle program will be valid only in the fixed cycle subprogram. The modal of the program that called out the fixed cycle will not be affected.

(10) Other subprograms cannot be called from the fixed cycle subprogram.

(11) Decimal points in the movement command will be ignored during the fixed cycle subprogram.

(12) If the No. of repetitions L is 2 or more during the incremental value mode, the positioning will also be incremental each time.
(Example) G91 G81 X10. Z-50. R-20. F100. L3;



(13) 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.

13.1.14 Initial Point and R Point Level Return ; G98,G99

Function and purpose

Whether to use R point or initial level as the return level in the final sequence of the fixed cycle can be selected.

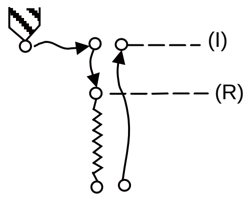
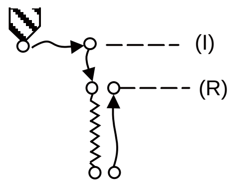
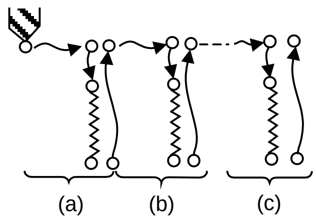
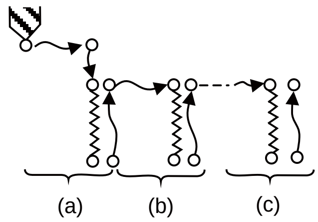
Command format

G98 ; ... Initial level return

G99 ; ... R point level return

Detailed description

The relation of the G98/G99 mode and the number of repetition designation is as shown below.

No. of hole drilling times	Program example	G98 (At power ON, at cancel with M02, M30, and reset button)	G99
Only one execution	G81 X100. Y100. Z-50. R25. F1000 ;	 Initial level return is executed.	 R point level return is executed.
Two or more executions	G81 X100. Y100. Z-50. R25. L5 F1000 ;	 Initial level return is always executed.	

(a) First time (b) Second time (c) Last time

13.1.15 Setting of Workpiece Coordinates in Fixed Cycle Mode

Function and purpose

The designated axis moves in the workpiece coordinate system set for the axis.
The Z axis becomes valid from the R point positioning after positioning is completed or from Z axis movement.

(Note) When the workpiece coordinates change, re-program the addresses Z and R, even if the values are the same.

(Example)
```
G54 Xx1 Yy1 Zz1;
G81 Xx1 Yy2 Zz2 Rr2;
G55 Xx3 Yy3 Zz2 Rr2;   ... Re-command even if Z and R are the same as the previous value.
Xx4 Yy4;
Xx5 Yy5;
```

13.2 Special Fixed Cycle

Function and purpose

The special fixed cycle is used with the standard fixed cycle.
Before using the special fixed cycle, record the hole machining data except for the positioning data (except for X, Y plane) by the standard fixed cycle.
Even after the special fixed cycle is executed, the recorded standard fixed cycle will be kept 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 carried out.

13.2.1 Bolt hole cycle ; G34

Function and purpose

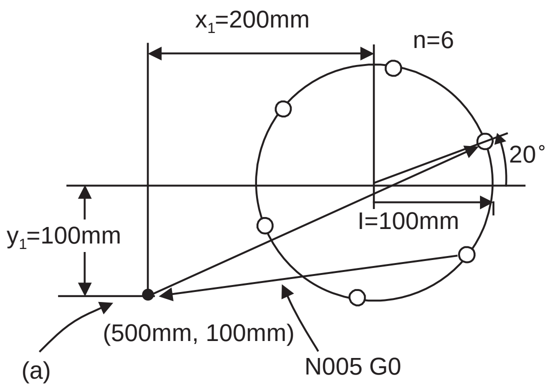
This function is to drill "n" holes, dividing the circumference by "n", on the circumference with a radius R centering the coordinates designated with X and Y. The drilling starts at the point which makes the angle θ with X axis. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G34 will not hold the data after the command is completed.

Command format

G34 Xx1 Yy1 Ir Jθ Kn ;

Xx1,Yy1	Positioning of bolt hole cycle center. This will be affected by G90/G91.
Ir	Radius r of the circle. The unit follows the input setting unit, and is given with a positive No.
Jθ	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°.
Kn	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, a program error (P221) occurs.

Program example




```
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 ;
```

(a) Position before G34 is executed

As shown in the example, the tool position after the G34 command is completed is above 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.

13.2.2 Line at angle ; G35

Function and purpose

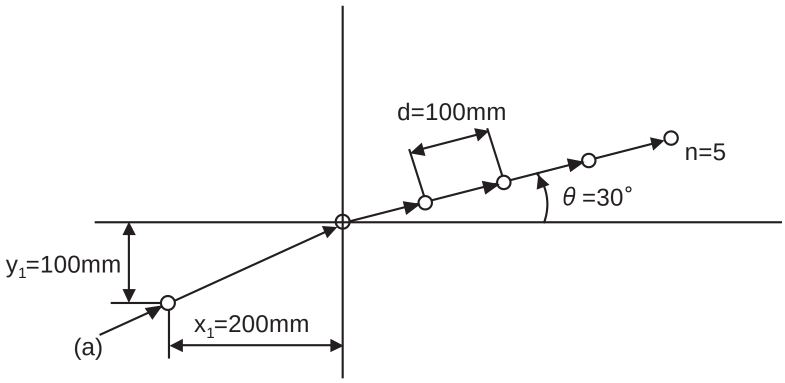
Using the position designated by X and Y as the start point, the n holes will be drilled with interval d in the direction which makes an angle θ with X axis. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G35 will not hold the data after the command is completed.

Command format

G35 Xx1 Yy1 Id Jθ Kn ;

Xx1,Yy1	Designation of start point coordinates. This will be affected by G90/G91.
Id	Interval d. The unit follows the input setting unit. If d is negative, the drilling will take place in the direction symmetrical to the center of the start point.
Jθ	Angle θ. 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°.)
Kn	No. of holes n to be drilled. 1 to 9999 can be designated, and the start point is included.

Program example




```
G91 ;
G81   Z-10.000 R5.000 L0 F100 ;
G35   X200.000 Y100.000
      I100.000 J30.000 K5 ;
```

(a) Position before G35 is executed

(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 G command of group 0 is issued in the same block as the G35 command, the command issued later has the priority.
(Example)G35 G28 Xx1 Yy1 Ii1 Jj1 Kk1 ; G35 is ignored G 28 is executed as Xx1 Yy1
(Note 4) If there is G72 to G89 command in the same block as the G35 command, the fixed cycle will be ignored, and the G35 command will be executed.

13.2.3 Arc ; G36

Function and purpose

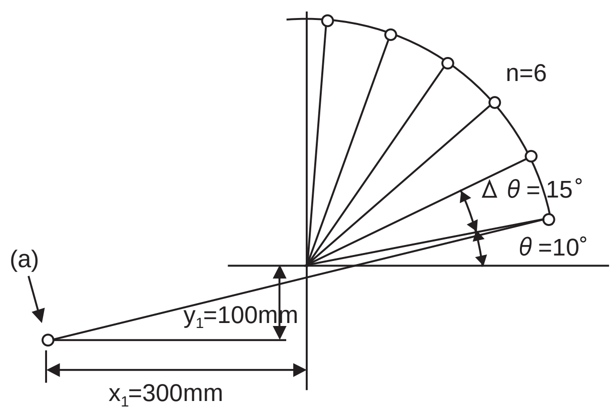
The "n" holes aligned with the angle interval $\Delta\theta$ will be drilled starting at the point which makes the angle θ with the X axis on the circumference with a radius R centering the coordinates designated with X and Y. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G36 will not hold the data after the command is completed.

Command format

G36 Xx1 Yy1 Ir Jθ PΔθ Kn ;

Xx1,Yy1	Center coordinates of arc. This will be affected by G90/G91.
Ir	Radius r of arc. The unit follows the input setting unit, and is given with a positive No.
Jθ	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°.
P Δθ	Angle interval θ. 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°.)
Kn	No. of holes n to be drilled. The setting range is 1 to 9999.

Program example




```
N001    G91 ;
N002    G81  Z-10.000 R5.000 F100 ;
N003    G36  X300.000 Y100.000 I300.000
             J10.000 P15000 K 6 ;
```

(a) Position before G36 is executed

13.2.4 Grid ; G37.1

Function and purpose

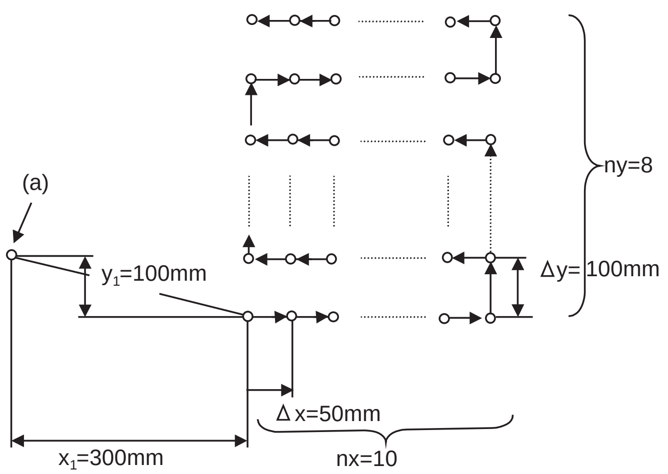
The nx points on a grid are drilled with an interval Δx parallel to the X axis, starting at the position designated with X, Y. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G37.1 will not hold the data after the command is completed.

Command format

G37.1 Xx1 Yy1 IΔx Pnx JΔy Kny ;

Xx1,Yy1	Designation of start point coordinates. This will be affected by G90/G91.
I Δx	Interval Δx of the X axis. The unit will follow the input setting unit. If Δx 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.
Pnx	No. of holes nx in the X axis direction. The setting range is 1 to 9999.
J Δy	Interval Δy of the Y axis. The unit will follow the input setting unit. If Δy 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.
Kny	No. of holes ny in the Y axis direction. The setting range is 1 to 9999.

Program example




```
G91 ;
G81    Z-10.000 R5.000 F20 ;
G37.1  X300.000 Y-100.000 I50.000 P10
       J100.000 K8 ;
```

(a) Position before G37.1 is executed

(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 G, L, N, X, Y, I, P, J, K, F, M, S or B is programmed in the same block as the G37.1 command, that address will be ignored.
(Example) Qq1 will be ignored in the following case.
G37.1 Xx1 Yy1 Ii1 Pp1 Jj1 Kk1 Qq1 ;

(Note 3) If G command of group 0 is issued in the same block as the G37.1 command, the command issued later has the priority.

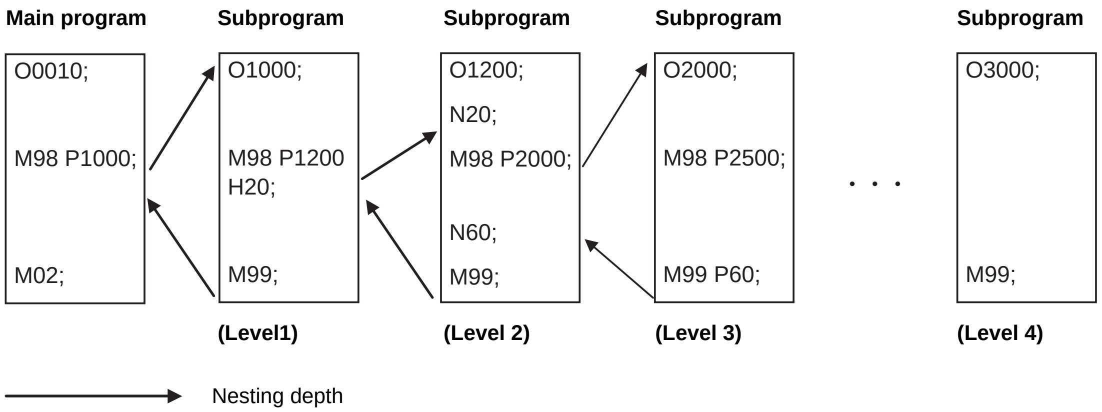
(Note 4) If there is G72 to G89 command in the same block as the G37.1 command, the fixed cycle will be ignored, and the G37.1 command will be executed.

13.3 Subprogram Control; M98, M99

13.3.1 Subprogram Call ; M98,M99

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. M98 serves to call subprograms and M99 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 subprogram control functions and fixed cycle functions.

	Case 1	Case 2	Case 3	Case 4
1. Subprogram control	No	Yes	Yes	No
2. Fixed cycles	No	No	Yes	Yes
Function				
1. Memory mode	○	○	○	○
2. Subprogram call	×	○	○	×
3. Subprogram variable designation (Note 2)	×	○	○	×
4. Subprogram nesting level call (Note 3)	×	○	○	×
5. Fixed cycles	×	×	○	○
6. Editing subprogram for fixed cycle	×	×	○	○

(Note 1) ○ denotes available functions and × denotes unavailable functions.

(Note 2) Variables cannot be transferred with the M98 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

M98 P__ H__ L__ ; ... Subprogram call

P	Program No. of subprogram to be called (own program if omitted) Note that P can be omitted only during memory mode and MDI mode. (Max. 8 digits)
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 as L1, and is not executed when L0.) (1 to 9999 times depending on the 4-digit value) For instance, For instance, M98 P1 L3 ; is equivalent to the following: M98 P1 ; M98 P1 ; M98 P1 ;

M99 P__ H__ Q__ R__ L__ ; ... Return to main program from subprogram

P	Sequence number of return destination (return to the block that follows the calling block if omitted)
H	Program number of return destination (return to the main program at calling if omitted)
Q	Sequence number to start searching of return destination (the block that follows the calling block will be handled as the search start position if omitted)
R	Sequence number to finish searching of return destination (the block that precedes the calling block will be handled as the search finish position if omitted)
L	Number of times after repetition number has been changed ("-1" if omitted)

Detailed description

Creating and registering subprograms

Subprograms have the same format as machining programs for normal memory mode, except that the subprogram completion instruction M99 (P_) ; must be registered as an independent block in the last block.

O******** ;	Program No. as subprogram No.
.... ; : .... ;	Main body of subprogram
M99 ;	Subprogram return command
%(EOR)	Registration completion code

(1) The above program is registered 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 subprogram Nos. ranging from 1 to 99999999 designated by the optional specifications can be used.

(3) Main program and subprograms are registered in the order they were read without distinction. Therefore, main programs and subprograms should not be given the same Nos. (If they are, error "E11" will be displayed at registration.)
Registration example

```
  ;
O○○○○  ;
............................  ;            Subprogram A
  :
M99     ;
%
```

```
O △△△△  ;
............................  ;            Subprogram B
  :
M99     ;
%
```

```
O****  ;
............................  ;            Subprogram C
  :
M99     ;
%
```

(4) Main programs can be entered in the memory or program by MDI operation but subprograms must be entered in the memory.

(5) Besides the M98 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

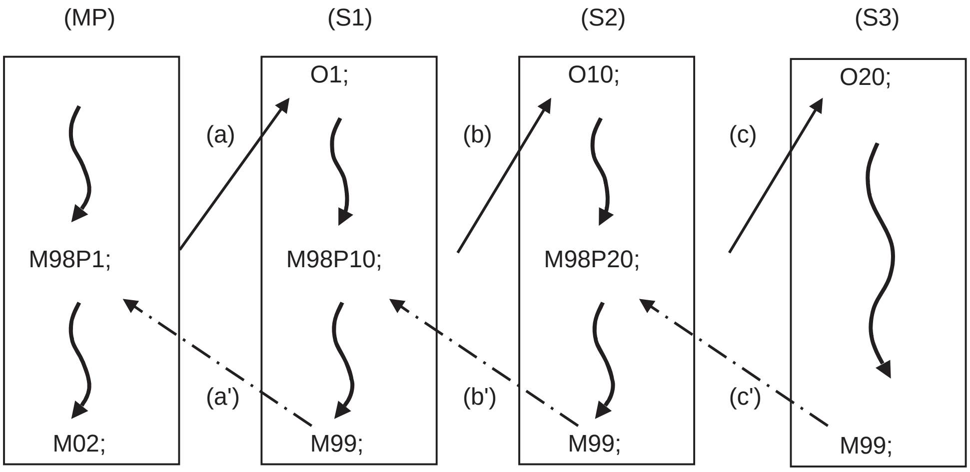
(6) Subprogram nesting is not subject to the following commands which can be called even beyond the 8th nesting level.
- Fixed cycles

(7) To repeatedly use the subprogram, it can be repeated l1 times by programming M98 Pp1 Ll1;.

Program example

Program example 1

When there are 3 subprogram calls (known as 3 nesting levels)



(M) Main program
(S1) Subprogram 1
(S2) Subprogram 2
(S3) Subprogram 3

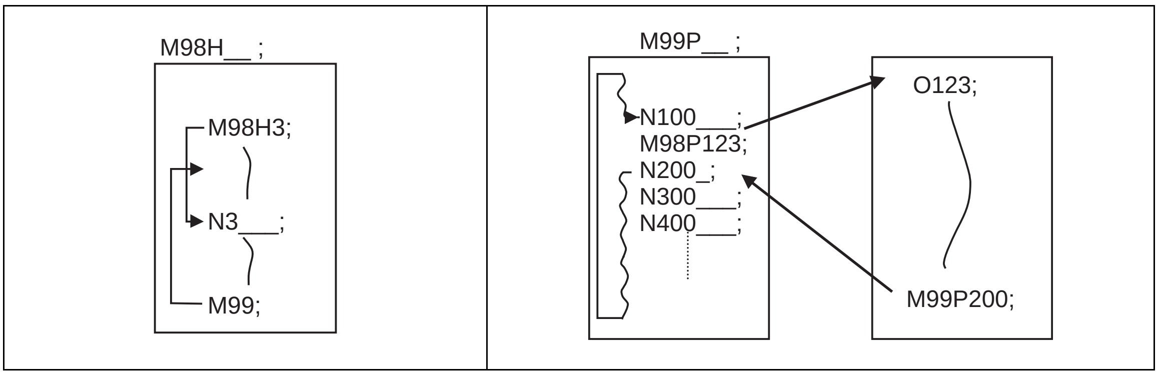
Sequence of execution : (a)-(b)-(c)-(c')-(b')-(a')

(1) For nesting, the M98 and M99 commands should always be paired off on a 1:1 basis; (a)' for (a), (b)' for (b), etc.

(2) Modal information is rewritten in the order of execution sequence without distinction between main programs and subprograms. Therefore, after calling a subprogram, attention must be paid to the modal data status when programming.

Program example 2

The M98 H_ ; M99 P_ ; commands designate the sequence Nos. in a program with a call instruction.



Precautions

(1) Program error (P232) will occur when the designated P (program No.) cannot be found.

(2) The M98 P_ ; M99 ; block does not perform a single block stop. If any address except O, N, P, L or H is used, single block stop can be executed. (With "X100. M98 P100 ;, the operation branches to O100 after X100. is executed.)

(3) When M99 is commanded by the main program, operation returns to the head. (This is same for MDI.)

(4) Note that it takes time to search when the sequence No. is designated by M99 P_ ;.

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 variable values depending on the condition of executing the program.

Command format

#* = ○○○○○○○○;**

#* = [formula] ;**

Detailed description

Variable expressions

		Example
#m	m = value consisting of 0 to 9	#100
# [f]	f = one of the followings 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) will occur when a variable No. is negative.
(Note 4) Examples of incorrect variable expressions are given below.

Incorrect	Correct
#6/2	# [6/2] (#6/2 is regarded as [#6] /2)
#--5	# [-[-5]]
#- [#1]	# [-#1]

Types of variables

The following table gives the types of variables.

<table>
<tr><th colspan="3">Variable set option</th><th colspan="2">No.</th><th>Function</th></tr>
<tr><td colspan="3">Common variables</td><td>Variables common to all part systems</td><td>Variables for each part system</td><td rowspan="9">- Can be used in common throughout main, sub and macro programs.</td></tr>
<tr><td rowspan="8"></td><td rowspan="4">1 part system</td><td>100 sets</td><td>500 to 549</td><td>100 to 149</td></tr>
<tr><td>200 sets</td><td>500 to 599</td><td>100 to 199</td></tr>
<tr><td>300 sets</td><td>500 to 699</td><td>100 to 199</td></tr>
<tr><td>600 sets</td><td>500 to 999</td><td>100 to 199</td></tr>
<tr><td rowspan="4">Multi-part system
(n = number of part systems)</td><td>50 + 50 sets</td><td>500 to 549</td><td>100 to 149 * n</td></tr>
<tr><td>100 + 100 sets</td><td>500 to 599</td><td>100 to 199 * n</td></tr>
<tr><td>200 + 100 sets</td><td>500 to 699</td><td>100 to 199 * n</td></tr>
<tr><td>500 + 100 sets</td><td>500 to 999</td><td>100 to 199 * n</td></tr>
<tr><td colspan="3">Local variables</td><td colspan="2">1 to 33</td><td>Can be used as local variables in macro programs.</td></tr>
<tr><td colspan="3">System variable</td><td colspan="2">From 1000</td><td>Application is fixed by system.</td></tr>
<tr><td colspan="3">Fixed cycle variables</td><td colspan="2">1 to 32</td><td>Local variables in fixed cycle programs.</td></tr>
</table>

(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) Variable names can be set for #500 to #519.

(Note 4) Variable names cannot be used for address "O" and "N".

Variable quotations

Variables can be used for all addresses except O, N and / (slash).

(1) When the variable value is used directly:
X#1 .... Value of #1 is used as the X value.
(2) When the complement of the variable value is used:
X-#2 .... Value with the #2 sign changed is used as the X value.
(3) 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.).
(4) When defining the variable arithmetic formula:
#1 = #3 + #2 - 100 .... Value of the operation result of #3 + #2 - 100. is used as the #1 value.
X［#1 + #3 + 1000］........ 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 ;

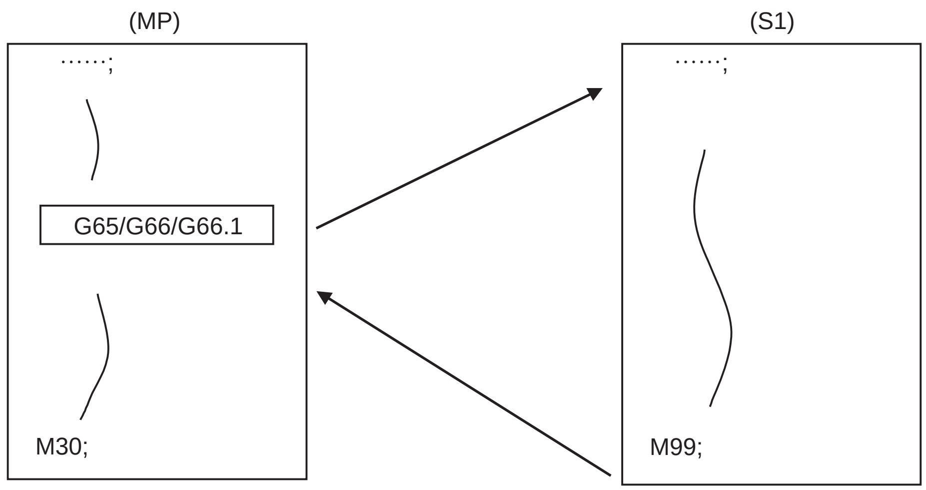
(Note 2) Up to five sets of square parentheses [ ] may be used.
#543 = -[[[[[#120]/2+15.]*3-#100]/#520+#125+#128]*#130+#132]
(Note 3) There are no restrictions on the number of characters and number of variables for variable definition.
(Note 4) The variable values should be within the range of 0 to ±99999999.
If this range is exceeded, the arithmetic operations may not be conducted properly.
(Note 5) The variable definitions become valid from the next command.
#1 = 100 ; .... #1 = 100
#1 = 200 #2 = #1 + 200 ; ..... #1 = 200, #2 = 400
#3 = #1 + 300 ; .... #3 = 500
(Note 6) Variable quotations are always regarded as having a decimal point at the end.
When #100 = 10
X#100 ; is treated as X10.

13.5 User Macro

13.5.1 User Macro

Function and purpose

A group of control and arithmetic instructions can be registered and used as a macro program to make it one integrated function.
Macro programs use variables, control and arithmetic instructions to create subprograms which function to provide special-purpose controls.
By combining the user macros with variable commands, it is possible to use the macro program call, arithmetic operations, data input/output with PLC, control, decision, branch and many other instructions for measurement and other such applications.



(MP) Main program (S1) Macro program (subprogram)

These special-purpose control functions (macro programs) are called by the macro call instructions from the main program when needed.

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 (G66, G66.1) cancel

Detailed description

(1) When the G66 or G66.1 command is entered, the specified user macro program will be called every time a block is executed or after a movement command in blocks with a movement command is executed, until the G67 (cancel) command is entered.
(2) The G66 (G66.1) and G67 commands must be paired in a same program.

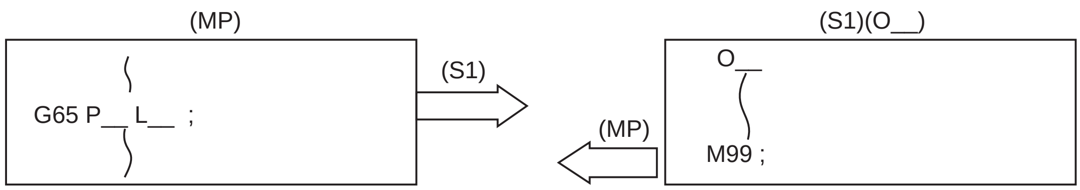
13.5.2 Macro Call Instruction

Function and purpose

Macro call commands include the simple calls which call only the instructed block and the modal calls (types A and B) which call a block in the call modal.

13.5.2.1 Simple Macro Calls ; G65

Function and purpose



M99 is used to terminate the user macro subprogram.

(MP) Main program (S1) Subprogram

Command format

G65 P__ L__ argument ; ... Simple macro calls

P	Program No.
L	Number of repetitions:

Detailed description

When the argument must be transferred as a local variable to a user macro subprogram, the actual value should be designated after the address.
In this case, regardless of the address, a sign and decimal point can be used in the argument. There are 2 ways in which arguments are designated.

Argument designation I

Format : A_ B_ C_ ... X_ Y_ Z_

(1) Arguments can be designated using any address except G, L, N, O and P.
(2) I, J and K must be designated in alphabetical order.
 I_ J_ K_ ...... Correct
 J_ I_ K_ ...... Incorrect
(3) Except for I, J and K, there is no need for designation in alphabetical order.
(4) Addresses which do not need to be designated can be omitted.
(5) The following table shows the correspondence between the addresses which can be designated by argument designation I and the variable numbers in the user macro main body.

Address and variable No. correspondence		Addresses available for call instructions	
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

Argument designation II

Format : A__ B__ C__ I__ J__ K__ I__ J__ K__ ...

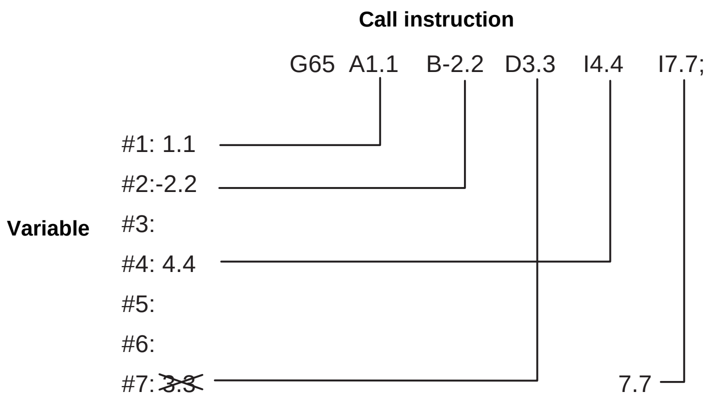
(1) In addition to address A, B and C, up to 10 groups of arguments with I, J, K serving as 1 group can be designated.
(2) When the same address is duplicated, designate the addresses in the specified order.
(3) Addresses which do not need to be designated can be omitted.
(4) The following table shows the correspondence between the addresses which can be designated by argument designation II and the variable numbers in the user macro main body.

Argument specification II address	Variable in macro		Argument specification II address	Variable in macro
A	#1		J5	#17
B	#2		K5	#18
C	#3		I6	#19
I1	#4		J6	#20
J1	#5		K6	#21
K1	#6		I7	#22
I2	#7		J7	#23
J2	#8		K7	#24
K2	#9		I8	#25
I3	#10		J8	#26
J3	#11		K8	#27
K3	#12		I9	#28
I4	#13		J9	#29
J4	#14		K9	#30
K4	#15		I10	#31
I5	#16		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.

Using arguments designations I and II together

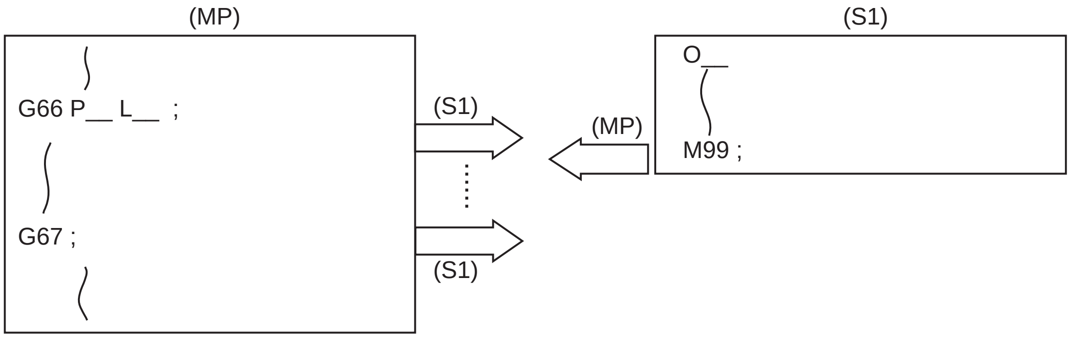
If addresses corresponding to the same variable are commanded when both types I and II are used to designate arguments, the latter address will become valid.
(Example 1)



In the above example, I7.7 argument is valid when both arguments D3.3 and I7.7 are commanded for the #7 variable.

13.5.2.2 Modal Call A (Movement Command Call) ; G66

Function and purpose



(MP) Main program (S1) Subprogram

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. A number of user macro subprograms are designated with "L".
The argument is the same as for a simple call.

Command format

G66 P__ L__ argument ; ... Macro modal call A

P	Program No.
L	Number of repetitions:
Argument	Specify variable data

Detailed description

(1) When the G66 command is entered, the specified user macro program will be called after the movement command in a 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 a same program.
A program error will occur when G67 is issued without G66.

(Example) Drill cycle

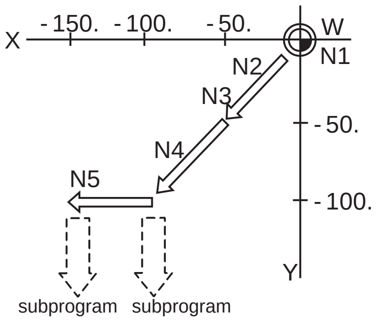
Main program

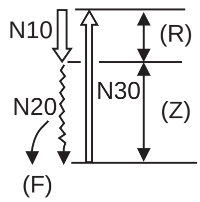
```
N1 G90 G54 G0 X0 Y0 Z0;
N2 G91 G00 X-50.Y-50.Z-200.;
N3 G66 P9010 R-10.Z-30.F100;
N4 X-50.Y-50.;      To subprogram after axis command execution
N5 X-50.;           To subprogram after axis command execution
N6 G67;
  ~
```

Subprogram

```
O 9010
N10 G00 Z #18 M0;
N20 G09 G01 Z #26 F#9;
N30 G00 Z-〔#18+#26〕;
M99
```

To main program





(F) Argument F (R) Argument R (Z) Argument Z

(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.

13.5.2.3 Modal Call B (for each block) ; G66.1

Function and purpose

The specified user macro subprogram is called unconditionally for each command block which is assigned between G66.1 and G67 and the subprogram will be repeated for the number of times specified in L.
The argument is the same as for a simple call.

Command format

G66.1 P__ L__ argument ; ... Modal call B

P	Program No.
L	Number of repetitions

Detailed description

(1) In the G66.1 mode, everything except the 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 O and N will function as the argument.

(2) All significant blocks in the G66.1 mode are handled as when G65P__ is assigned at the head of a block.
(Example 1)
In "G66.1 P1000 ; " mode,
N100 G01 G90 X100. Y200. F400 R1000 ; is same as
N100 G65 P1000 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 as normal NC command values.

(4) Program number O, sequence numbers N and modal G codes are updated as modal information.

13.5.2.4 G Code Macro Call

Function and purpose

User macro subprogram with prescribed program numbers can be called merely by issuing the G code command.

Command format

G argument ; ... G code macro call**

G**	G code for macro call

Detailed description

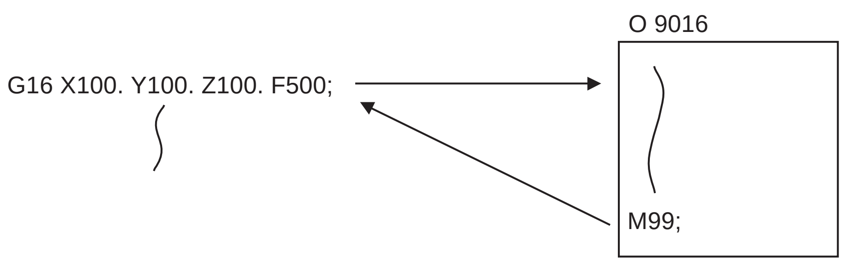
(1) The above instruction functions in the same way as the instructions below, however, the correspondence between M codes and instructions can be set by parameters.
a : M98 P △△△△ ;
b : G65 P △△△△ Argument ;
c : G66 P △△△△ Argument ;
d : G66.1 P △△△△ Argument ;
When the parameters corresponding to c 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 "**" which conducts the macro call and the macro program number P △△△△ to be called is set by parameters.

(3) Up to 10 G codes from G100 to G255 can be used with this instruction. (G01 to G99 can also be used depending on the parameter "#1081 Gmac_P")

(Note 1) G101 to G110 and G200 to G202 are user macro I codes, but if the parameters are set as the G code call codes, the G code call will be the priority, and these codes cannot be used for user macro I.

(4) These commands cannot be issued during a user macro subprogram which has been called by a G code.



13.5.2.5 Miscellaneous Command Macro Call (for M, S, T, B Code Macro Call)

Function and purpose

The user macro subprogram of the specified program number can be called merely by issuing an M (or S, T, B) code. (Registered M code and all S, T and B codes.)

Command format

M** ; (or S** ;, T** ;, B** ;) ... Miscellaneous command macro call

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, however, the correspondence between M codes and instructions can be set by parameters. (Same for S, T and B codes)

a:M98 P**** ; b:G65 P**** M** ;	M98, M** are not output.
c:G66 P**** M** ;	
d:G66.1 P**** M** ;	

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 P**** to be called is set by parameters. Up to 10 M codes from M00 to M95 can be registered.
Note that the codes to be registered should exclude those basically required for the machine and M0, M1, M2, M30 and M96 to M99.

(3) As with M98, 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 registered miscellaneous commands above are issued in a user macro subprogram which are called by an M code, it will not be regarded as a macro call and will be handled as a normal miscellaneous command. (Same for S, T and B codes)

(5) All S, T and B codes call the subprograms in the prescribed program numbers of the corresponding S, T and B functions.

(6) Up to 10 M codes can be set. However, if not using up 10 codes, set the parameters as shown below.

```
[MACRO FILE (1)]

          <CODE> < TYPE> <PROGRAM-No.>
 M[01]      20      0          8000   ---- For M20 commands, set to call O8000 in type 0 (M98 type)
 M[02]      21      0          8001   ---- For M21 commands, set to call O8001 in type 0 (M98 type)
 M[03]    9999      0     199999999
 M[04]    9999      0     199999999
 M[05]    9999      0     199999999   ---- Set parameters that will not be used as shown at the left.
             :      :             :
             :      :             :
 M[10]    9999      0     199999999
```

13.5.2.6 Detailed Description for Macro Call Instruction

Detailed description

Differences between M98 and G65 commands

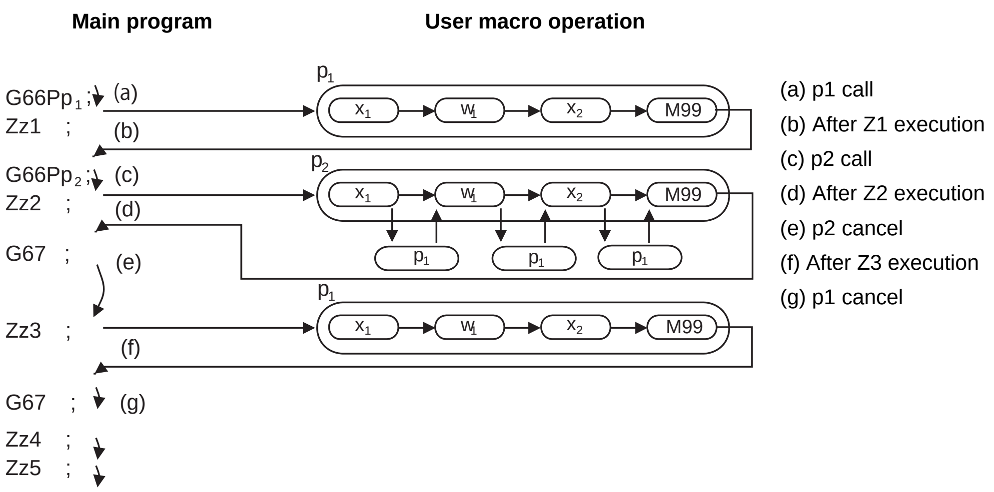
(1) The argument can be designated for G65 but not for M98.
(2) The sequence number can be designated for M98 but not for G65, G66 and G66.1.
(3) M98 executes subprograms after all the commands except M, P, H and L in the M98 block are executed, but G65 branches directly to the subprogram without any further operation.
(4) When any address except O, N, P, H or L is included in the M98 block, the single block stop will be conducted, but not for the G65.
(5) The level of the M98 local variables is fixed but it varies in accordance with the nesting depth for G65. ("#1" before and after M98, for instance, has the same significance, but they have different significance in G65.)
(6) The M98 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 by simple call or modal call.
The argument for a macro call instruction is valid only within 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 programs of each macro call of each level.

(Note 1) When a G65, G66, G66.1 G code macro call or miscellaneous command macro call is conducted, this is regarded as a nesting level and the level of the local variables is also incremented by one.
(Note 2) With modal call A, the designated user macro subprogram is called every time a movement command is executed. However, when the G66 command is duplicated, the next user macro subprogram is called to movement commands in the macro every time an axis is moved.
User macro subprograms are called from the one commanded last.

(Example 1)



13.5.3 Variable

Function and purpose

Both the variable specifications and user macro specifications are required for the variables which are used with the user macros.
The compensation amounts of the local, common and system variables among the variables for this 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".)

Detailed description

Use of multiple variable

When the user macro specifications are applied, variable Nos. can be turned into variables (multiple uses of variables) or replaced by <formula>.
Only one of the four basic arithmetic rule (+, -, * , /) operations can be conducted with <formula>.
(Example 1) Multiple uses 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=10 #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 designations of variables

#10=5;	<Formula>##10 = 100; is handled in the same manner as # [#10] = 100.
##10=100 ;	In which case, #5 = 100.

(Example 3) Replacing variable Nos. with <formula>

#10=5 ;	
#[#10 + 1] = 1000 ;	In which case, #6 = 1000.
#[#10 - 1] = -1000 ;	In which case, #4 = -1000.
#[#10 * 3] = 100 ;	In which case, #15 = 100.
#[#10/2] = -100 ;	In which case, #2 = -100.

Undefined variables

When applying the user macro specifications, variables which have not been used even once after the power was switched on or local variables which were not specified by the G65, G66 or G66.1 commands, can be used as <Blank>. Also, variables can forcibly be set to <Blank>.
Variable #0 is always used as the <Blank> and cannot be defined in the left-side member.

(1) Arithmetic expressions
#1 = #0 ; ....#1 = <Blank>
#2 = #0 + 1; .... #2 = 1
#3 = 1 + #0; .... #3 = 1
#4 = #0 * 10; .... #4 = 0
#5 = #0 + #0; .... #5 = 0

Note that <Blank> in an arithmetic expression is handled in the same way as 0.
<Blank> + <Blank> = 0
<Blank> + <Constant> = Constant
<Constant> + <Blank> = Constant

(2) Variable quotations
When only the undefined variables are quoted, they are ignored including the address itself.
When #1 = <Blank>
G0 X#1 Y1000 ; ....Equivalent to G0 Y1000 ;
G0 X#1 + 10 Y1000 ;.................. Equivalent to G0 X10 Y1000 ;

(3) Conditional expressions
<Blank> differs from "0", only for EQ and NE. (#0 is <Blank>.)

When #101 = <Blank>	When #101 = 0
#101EQ#0 <Blank> = <Blank> established	#101EQ#0 0 = <Blank> not established
#101NE0 <Blank> ≠ 0 established	#101NE0 0 ≠ 0 not established
#101GE#0 <Blank> >= 0 established	#101GE#0 0 >= <Blank> established
#101GT0 <Blank> > 0 not established	#101GT0 0 > 0 not established
#101LE#0 <Blank> <= <Blank> established	#101LE#0 0 <= <Blank> established
#101LT0 <Blank> < 0 not established	#101LT0 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.

13.5.4 Types of Variables

13.5.4.1 Common Variables

Detailed description

Common variables can be used commonly from any position. Number of the common variables sets depends on the specifications.
Refer to the explanation about Variable Commands for details.

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 alphanumerics and it must begin with a letter. Do not use "#" in variable names. It causes an alarm when the program is executed.

SETVNn [NAME1,NAME2, ...] ;

n	Head No. of variable to be named
NAME1	#n name (variable name)
NAME2	#n + 1 name (variable name)

Variable names are separated by a comma (,).

(1) Once variable names have been set, they will not be cleared even when the power is turned off.
(2) Variables in programs can be quoted by their variable names. In this case, the variables should be enclosed in square parentheses [ ].
(Example 1) G01X [#POINT1] ;
[#NUMBER]=25;
(3) The variable Nos., data and variable names are displayed 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) Do not use characters (SIN, COS, etc.) predetermined by the NC and used for operation commands at the head of a variable name.

13.5.4.2 Local Variables (#1 to #33)

Detailed description

Local variables can be defined as an <argument> when a macro subprogram is called, and also used locally within main programs and subprograms. They can be duplicated because there is no relationship between macros. (up to 4 levels)

G65 P__ L__ <argument> ;

P	Program No.
L	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 designation I]

Call command		Argument address	Local variable No.		Call command		Argument address	Local variable No.
G65 G66	G66.1				G65 G66	G66.1		
○	○	A	#1		○	○	Q	#17
○	○	B	#2		○	○	R	#18
○	○	C	#3		○	○	S	#19
○	○	D	#7		○	○	T	#20
○	○	E	#8		○	○	U	#21
○	○	F	#9		○	○	V	#22
×	× *	G	#10		○	○	W	#23
○	○	H	#11		○	○	X	#24
○	○	I	#4		○	○	Y	#25
○	○	J	#5		○	○	Z	#26
○	○	K	#6				-	#27
×	× *	L	#12				-	#28
○	○	M	#13				-	#29
×	× *	N	#14				-	#30
×	×	O	#15				-	#31
×	× *	P	#16				-	#32
							-	#33

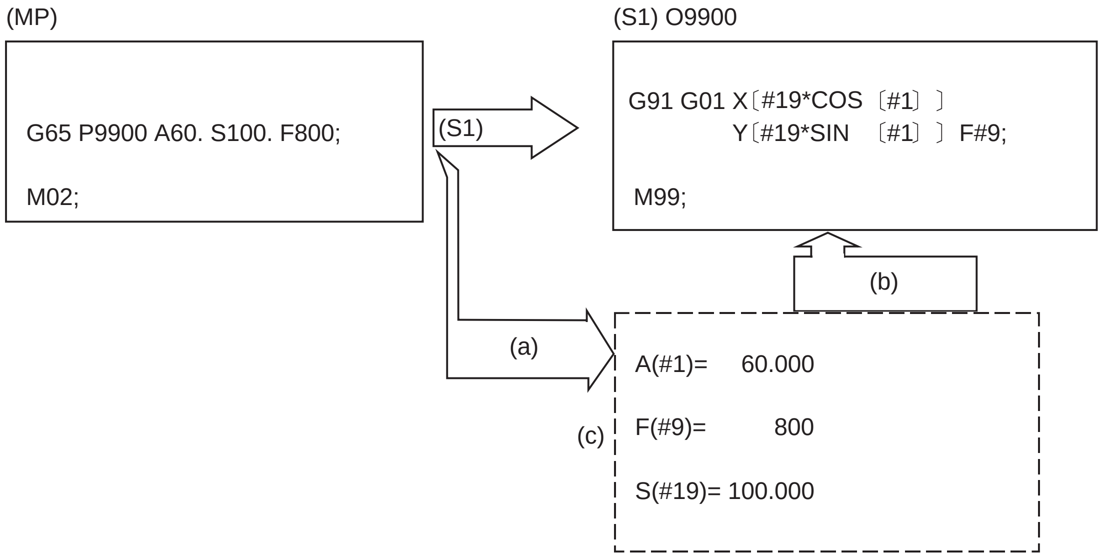
"×" in the above table denotes argument addresses which cannot be used. However, provided that the G66.1 mode has been established, an argument address denoted by the asterisk can be added for use.
The hyphen (-) mark indicates that there is no corresponding address.

[Argument designation II]

Argument designation II address	Variable in macro		Argument designation II address	Variable in macro
A	#1		J5	#17
B	#2		K5	#18
C	#3		I6	#19
I1	#4		J6	#20
J1	#5		K6	#21
K1	#6		I7	#22
I2	#7		J7	#23
J2	#8		K7	#24
K2	#9		I8	#25
I3	#10		J8	#26
J3	#11		K8	#27
K3	#12		I9	#28
I4	#13		J9	#29
J4	#14		K9	#30
K4	#15		I10	#31
I5	#16		J10	#32
			K10	#33

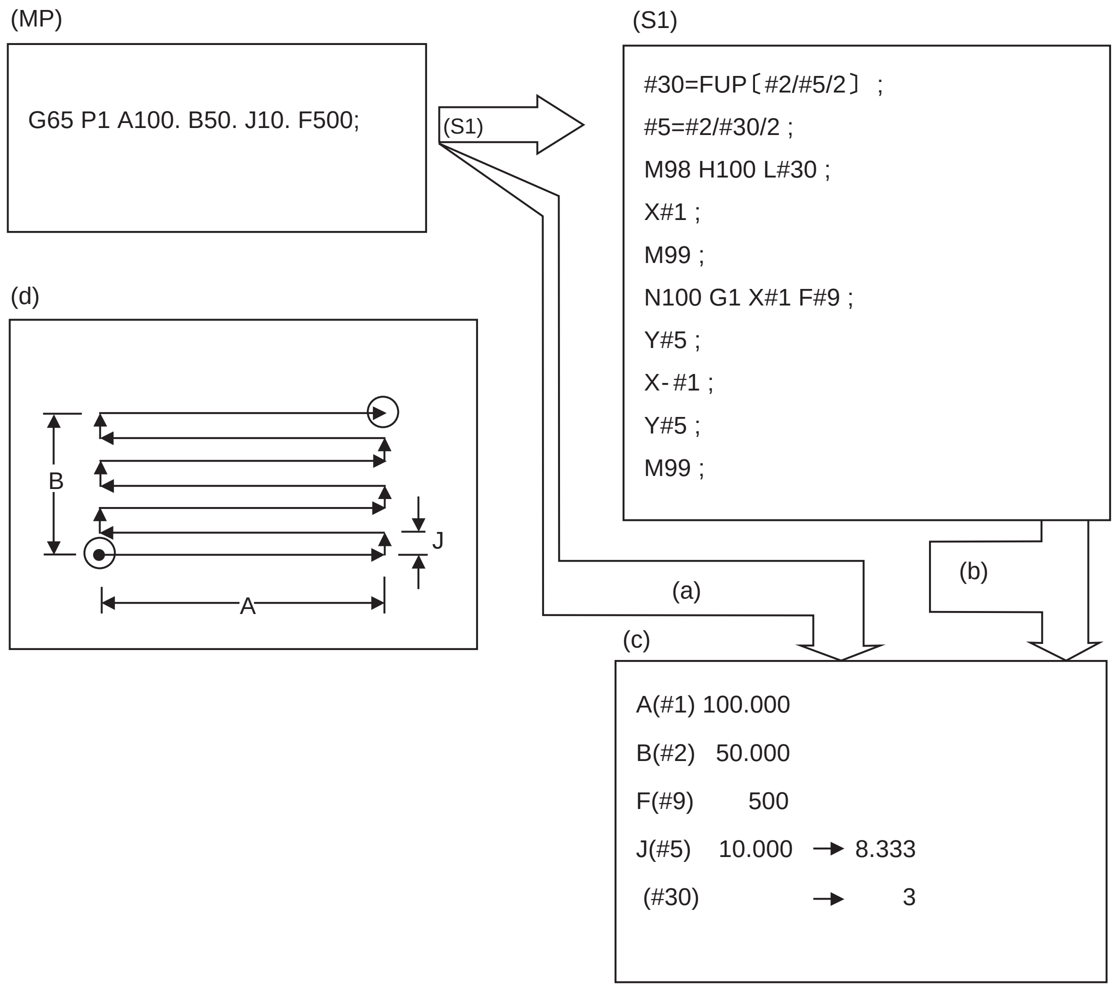
(Note 1) The numbers 1 to 10 accompanying I, J and K indicate the sequence of the commanded sets, and are not required in the actual command.

(1) Local variables in subprograms can be defined by means of the <argument> designation during macro call. (Local variables can be used freely in those subprograms.)



(MP) Main program
(S1) Subprogram
(a) Local variables set by argument
(b) Refer to the local variables and control the movement, etc.
(c) Local variable data table

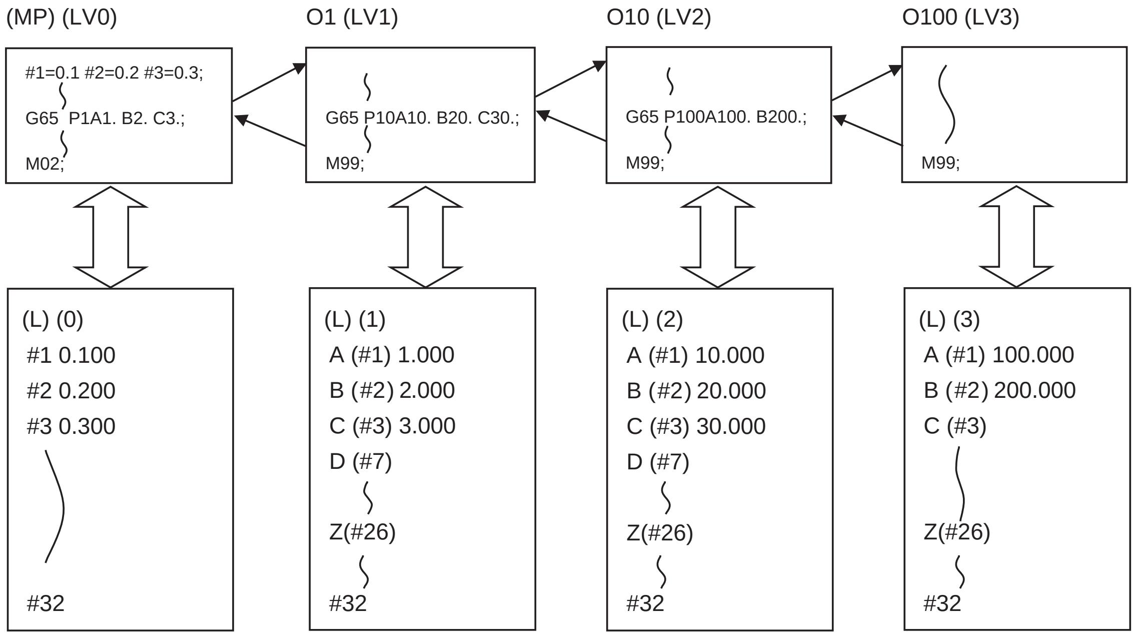
(2) The local variables can be used freely in that subprogram.



(MP) Main program
(a)Local variables set by argument
(c) Local variable data table
(S1) Subprogram
(b) The local variables can be changed in the subprogram.
(d) Example of front surface milling

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.



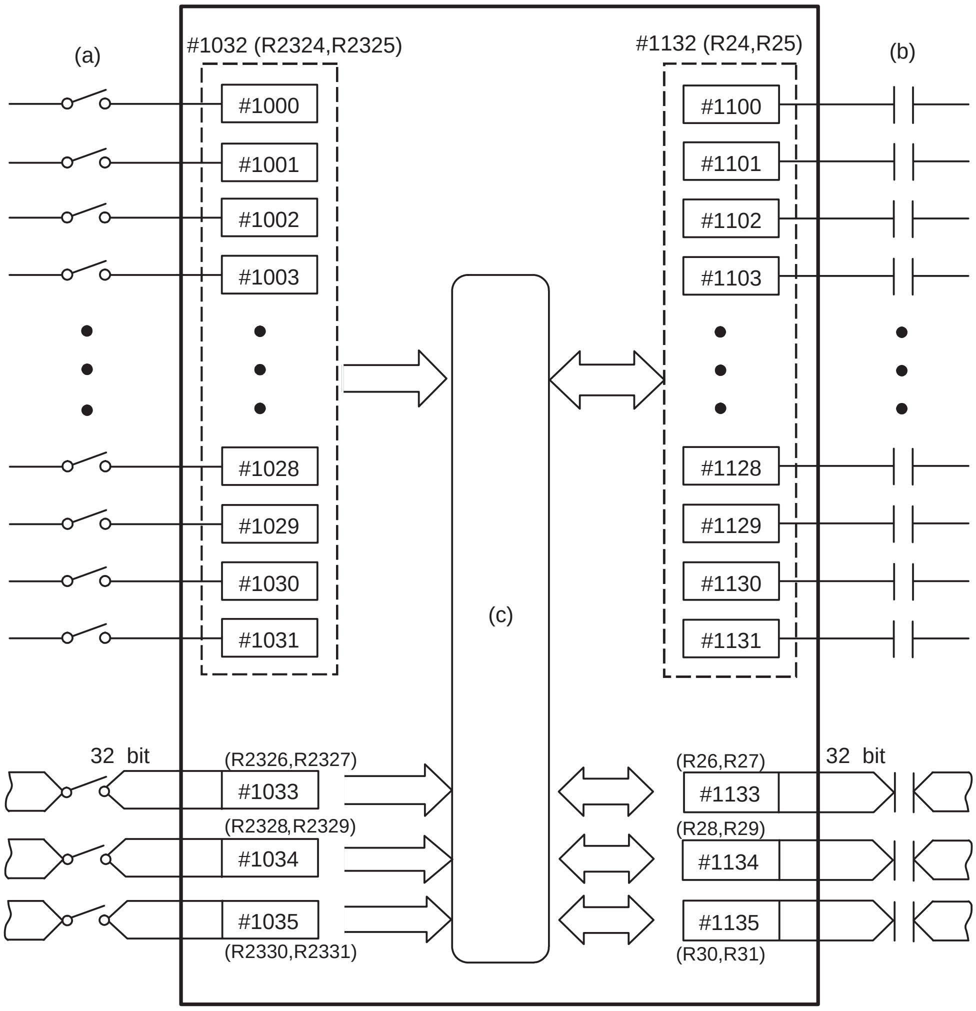
(MP) Main program
(LV0 - 3) Macro level 0 - 3
(L) Local variables

The status of the local variables is displayed on the setting and display unit.
Refer to the Instruction Manual for details.

13.5.4.3 Macro Interface Inputs/Outputs (#1000 to #1035, #1100 to #1135, #1200 to #1295, #1300 to #1395)

Function and purpose

The status of the interface input signals can be ascertained by reading out the values of variable numbers #1000 to #1035, #1200 to #1295.
The interface output signals can be sent by substituting values in variable Nos. #1100 to #1135, #1300 to #1395.



(a) Input signal (b) Output signal (c) Macro instructions

Detailed description

Macro interface inputs (#1000 to #1035, #1200 to #1295) : PLC -> NC

A variable value which has been read out can be only 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 No. #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 Nos. #1033 to #1035.
Variable Nos. #1000 to #1035, #1200 to #1295 are for readout only, and nothing can be placed in the left side member of their operation formula.
Input here refers to input to the control unit.
To use the macro interface function by part system, set the bit selection parameter "#6454/bit0".
Refer to (2) for the signals provided for each part system.

(1) Macro interface common to part systems (input)

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1000	1	Register R2324 bit 0	#1016	1	Register R2325 bit 0
#1001	1	Register R2324 bit 1	#1017	1	Register R2325 bit 1
#1002	1	Register R2324 bit 2	#1018	1	Register R2325 bit 2
#1003	1	Register R2324 bit 3	#1019	1	Register R2325 bit 3
#1004	1	Register R2324 bit 4	#1020	1	Register R2325 bit 4
#1005	1	Register R2324 bit 5	#1021	1	Register R2325 bit 5
#1006	1	Register R2324 bit 6	#1022	1	Register R2325 bit 6
#1007	1	Register R2324 bit 7	#1023	1	Register R2325 bit 7
#1008	1	Register R2324 bit 8	#1024	1	Register R2325 bit 8
#1009	1	Register R2324 bit 9	#1025	1	Register R2325 bit 9
#1010	1	Register R2324 bit 10	#1026	1	Register R2325 bit 10
#1011	1	Register R2324 bit 11	#1027	1	Register R2325 bit 11
#1012	1	Register R2324 bit 12	#1028	1	Register R2325 bit 12
#1013	1	Register R2324 bit 13	#1029	1	Register R2325 bit 13
#1014	1	Register R2324 bit 14	#1030	1	Register R2325 bit 14
#1015	1	Register R2324 bit 15	#1031	1	Register R2325 bit 15

System variable	No. of points	Interface input signal
#1032	32	Register R2324, R2325
#1033	32	Register R2326, R2327
#1034	32	Register R2328, R2329
#1035	32	Register R2330, R2331

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1200	1	Register R2326 bit 0	#1216	1	Register R2327 bit 0
#1201	1	Register R2326 bit 1	#1217	1	Register R2327 bit 1
#1202	1	Register R2326 bit 2	#1218	1	Register R2327 bit 2
#1203	1	Register R2326 bit 3	#1219	1	Register R2327 bit 3
#1204	1	Register R2326 bit 4	#1220	1	Register R2327 bit 4
#1205	1	Register R2326 bit 5	#1221	1	Register R2327 bit 5
#1206	1	Register R2326 bit 6	#1222	1	Register R2327 bit 6
#1207	1	Register R2326 bit 7	#1223	1	Register R2327 bit 7
#1208	1	Register R2326 bit 8	#1224	1	Register R2327 bit 8
#1209	1	Register R2326 bit 9	#1225	1	Register R2327 bit 9
#1210	1	Register R2326 bit 10	#1226	1	Register R2327 bit 10
#1211	1	Register R2326 bit 11	#1227	1	Register R2327 bit 11
#1212	1	Register R2326 bit 12	#1228	1	Register R2327 bit 12
#1213	1	Register R2326 bit 13	#1229	1	Register R2327 bit 13
#1214	1	Register R2326 bit 14	#1230	1	Register R2327 bit 14
#1215	1	Register R2326 bit 15	#1231	1	Register R2327 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1232	1	Register R2328 bit 0	#1248	1	Register R2329 bit 0
#1233	1	Register R2328 bit 1	#1249	1	Register R2329 bit 1
#1234	1	Register R2328 bit 2	#1250	1	Register R2329 bit 2
#1235	1	Register R2328 bit 3	#1251	1	Register R2329 bit 3
#1236	1	Register R2328 bit 4	#1252	1	Register R2329 bit 4
#1237	1	Register R2328 bit 5	#1253	1	Register R2329 bit 5
#1238	1	Register R2328 bit 6	#1254	1	Register R2329 bit 6
#1239	1	Register R2328 bit 7	#1255	1	Register R2329 bit 7
#1240	1	Register R2328 bit 8	#1256	1	Register R2329 bit 8
#1241	1	Register R2328 bit 9	#1257	1	Register R2329 bit 9
#1242	1	Register R2328 bit 10	#1258	1	Register R2329 bit 10
#1243	1	Register R2328 bit 11	#1259	1	Register R2329 bit 11
#1244	1	Register R2328 bit 12	#1260	1	Register R2329 bit 12
#1245	1	Register R2328 bit 13	#1261	1	Register R2329 bit 13
#1246	1	Register R2328 bit 14	#1262	1	Register R2329 bit 14
#1247	1	Register R2328 bit 15	#1263	1	Register R2329 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1264	1	Register R2330 bit 0	#1280	1	Register R2331 bit 0
#1265	1	Register R2330 bit 1	#1281	1	Register R2331 bit 1
#1266	1	Register R2330 bit 2	#1282	1	Register R2331 bit 2
#1267	1	Register R2330 bit 3	#1283	1	Register R2331 bit 3
#1268	1	Register R2330 bit 4	#1284	1	Register R2331 bit 4
#1269	1	Register R2330 bit 5	#1285	1	Register R2331 bit 5
#1270	1	Register R2330 bit 6	#1286	1	Register R2331 bit 6
#1271	1	Register R2330 bit 7	#1287	1	Register R2331 bit 7
#1272	1	Register R2330 bit 8	#1288	1	Register R2331 bit 8
#1273	1	Register R2330 bit 9	#1289	1	Register R2331 bit 9
#1274	1	Register R2330 bit 10	#1290	1	Register R2331 bit 10
#1275	1	Register R2330 bit 11	#1291	1	Register R2331 bit 11
#1276	1	Register R2330 bit 12	#1292	1	Register R2331 bit 12
#1277	1	Register R2330 bit 13	#1293	1	Register R2331 bit 13
#1278	1	Register R2330 bit 14	#1294	1	Register R2331 bit 14
#1279	1	Register R2330 bit 15	#1295	1	Register R2331 bit 15

(2) Macro interface by part system (input)

System variable	No. of points	Interface input signal						
		$1	$2	$3	$4	$5	$6	$7
		R2470	R2570	R2670	R2770	R2870	R2970	R3070
#1000	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1001	1	bit1	bit1	bit1	bit1	bit1	bit1	bit1
#1002	1	bit2	bit2	bit2	bit2	bit2	bit2	bit2
#1003	1	bit3	bit3	bit3	bit3	bit3	bit3	bit3
#1004	1	bit4	bit4	bit4	bit4	bit4	bit4	bit4
#1005	1	bit5	bit5	bit5	bit5	bit5	bit5	bit5
#1006	1	bit6	bit6	bit6	bit6	bit6	bit6	bit6
#1007	1	bit7	bit7	bit7	bit7	bit7	bit7	bit7
#1008	1	bit8	bit8	bit8	bit8	bit8	bit8	bit8
#1009	1	bit9	bit9	bit9	bit9	bit9	bit9	bit9
#1010	1	bit10	bit10	bit10	bit10	bit10	bit10	bit10
#1011	1	bit11	bit11	bit11	bit11	bit11	bit11	bit11
#1012	1	bit12	bit12	bit12	bit12	bit12	bit12	bit12
#1013	1	bit13	bit13	bit13	bit13	bit13	bit13	bit13
#1014	1	bit14	bit14	bit14	bit14	bit14	bit14	bit14
#1015	1	bit15	bit15	bit15	bit15	bit15	bit15	bit15

System variable	No. of points	Interface input signal						
		$1	$2	$3	$4	$5	$6	$7
		R2471	R2571	R2671	R2771	R2871	R2971	R3071
#1016	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1017	1	bit1	bit1	bit1	bit1	bit1	bit1	bit1
#1018	1	bit2	bit2	bit2	bit2	bit2	bit2	bit2
#1019	1	bit3	bit3	bit3	bit3	bit3	bit3	bit3
#1020	1	bit4	bit4	bit4	bit4	bit4	bit4	bit4
#1021	1	bit5	bit5	bit5	bit5	bit5	bit5	bit5
#1022	1	bit6	bit6	bit6	bit6	bit6	bit6	bit6
#1023	1	bit7	bit7	bit7	bit7	bit7	bit7	bit7
#1024	1	bit8	bit8	bit8	bit8	bit8	bit8	bit8
#1025	1	bit9	bit9	bit9	bit9	bit9	bit9	bit9
#1026	1	bit10	bit10	bit10	bit10	bit10	bit10	bit10
#1027	1	bit11	bit11	bit11	bit11	bit11	bit11	bit11
#1028	1	bit12	bit12	bit12	bit12	bit12	bit12	bit12
#1029	1	bit13	bit13	bit13	bit13	bit13	bit13	bit13
#1030	1	bit14	bit14	bit14	bit14	bit14	bit14	bit14
#1031	1	bit15	bit15	bit15	bit15	bit15	bit15	bit15

System variable	No. of points	Interface input signal						
		$1	$2	$3	$4	$5	$6	$7
#1032	32	R2470, R2471	R2570, R2571	R2670, R2671	R2770, R2771	R2870, R2871	R2970, R2971	R3070, R3071
#1033	32	R2472, R2473	R2572, R2573	R2672, R2673	R2772, R2773	R2872, R2873	R2972, R2973	R3072, R3073
#1034	32	R2474, R2475	R2574, R2575	R2674, R2675	R2774, R2775	R2874, R2875	R2974, R2975	R3074, R3075
#1035	32	R2476, R2477	R2576, R2577	R2676, R2677	R2776, R2777	R2876, R2877	R2976, R2977	R3076, R3077

Macro interface outputs (#1100 to #1135, #1300 to #1395) : NC -> PLC

Output signals can only be 0 or 1.
All the output Nos. from #1100 to #1131 can be sent at once by substituting a value in variable No. #1132.
Similarly, the output signals #1300 to #1311, #1332 to #1363, and #1364 to #1395 can be sent by substituting values to the variable Nos. #1133 to #1135. (2^0 to 2^{31})
The status of the writing and output signals can be read in order to compensate the #1100 to #1135, #1300 to #1395 output signals.
Output here refers to the output from the NC side.
To use the macro interface function by part system, set the bit selection parameter "#6454/bit0".Refer to (2) for the signals provided for each part system.

(1) Macro interface common to part systems (output)

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1100	1	Register R24 bit 0	#1116	1	Register R25 bit 0
#1101	1	Register R24 bit 1	#1117	1	Register R25 bit 1
#1102	1	Register R24 bit 2	#1118	1	Register R25 bit 2
#1103	1	Register R24 bit 3	#1119	1	Register R25 bit 3
#1104	1	Register R24 bit 4	#1120	1	Register R25 bit 4
#1105	1	Register R24 bit 5	#1121	1	Register R25 bit 5
#1106	1	Register R24 bit 6	#1122	1	Register R25 bit 6
#1107	1	Register R24 bit 7	#1123	1	Register R25 bit 7
#1108	1	Register R24 bit 8	#1124	1	Register R25 bit 8
#1109	1	Register R24 bit 9	#1125	1	Register R25 bit 9
#1110	1	Register R24 bit 10	#1126	1	Register R25 bit 10
#1111	1	Register R24 bit 11	#1127	1	Register R25 bit 11
#1112	1	Register R24 bit 12	#1128	1	Register R25 bit 12
#1113	1	Register R24 bit 13	#1129	1	Register R25 bit 13
#1114	1	Register R24 bit 14	#1130	1	Register R25 bit 14
#1115	1	Register R24 bit 15	#1131	1	Register R25 bit 15

System variable	No. of points	Interface output signal
#1132	32	Register R24, R25
#1133	32	Register R26, R27
#1134	32	Register R28, R29
#1135	32	Register R30, R31

(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.
<Blank> is treated as 0.
Any number except 0 and <Blank> is treated as 1.
Any value less than 0.00000001 is indefinite.

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1300	1	Register R26 bit 0	#1316	1	Register R27 bit 0
#1301	1	Register R26 bit 1	#1317	1	Register R27 bit1
#1302	1	Register R26 bit 2	#1318	1	Register R27 bit 2
#1303	1	Register R26 bit 3	#1319	1	Register R27 bit 3
#1304	1	Register R26 bit 4	#1320	1	Register R27 bit 4
#1305	1	Register R26 bit 5	#1321	1	Register R27 bit 5
#1306	1	Register R26 bit 6	#1322	1	Register R27 bit 6
#1307	1	Register R26 bit 7	#1323	1	Register R27 bit 7
#1308	1	Register R26 bit 8	#1324	1	Register R27 bit 8
#1309	1	Register R26 bit 9	#1325	1	Register R27 bit 9
#1310	1	Register R26 bit 10	#1326	1	Register R27 bit 10
#1311	1	Register R26 bit 11	#1327	1	Register R27 bit 11
#1312	1	Register R26 bit 12	#1328	1	Register R27 bit 12
#1313	1	Register R26 bit 13	#1329	1	Register R27 bit 13
#1314	1	Register R26 bit 14	#1330	1	Register R27 bit 14
#1315	1	Register R26 bit 15	#1331	1	Register R27 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1332	1	Register R28 bit 0	#1348	1	Register R29 bit 0
#1333	1	Register R28 bit 1	#1349	1	Register R29 bit 1
#1334	1	Register R28 bit 2	#1350	1	Register R29 bit 2
#1335	1	Register R28 bit 3	#1351	1	Register R29 bit 3
#1336	1	Register R28 bit 4	#1352	1	Register R29 bit 4
#1337	1	Register R28 bit 5	#1353	1	Register R29 bit 5
#1338	1	Register R28 bit 6	#1354	1	Register R29 bit 6
#1339	1	Register R28 bit 7	#1355	1	Register R29 bit 7
#1340	1	Register R28 bit 8	#1356	1	Register R29 bit 8
#1341	1	Register R28 bit 9	#1357	1	Register R29 bit 9
#1342	1	Register R28 bit 10	#1358	1	Register R29 bit 10
#1343	1	Register R28 bit 11	#1359	1	Register R29 bit 11
#1344	1	Register R28 bit 12	#1360	1	Register R29 bit 12
#1345	1	Register R28 bit 13	#1361	1	Register R29 bit 13
#1346	1	Register R28 bit 14	#1362	1	Register R29 bit 14
#1347	1	Register R28 bit 15	#1363	1	Register R29 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1364	1	Register R30 bit 0	#1380	1	Register R31 bit 0
#1365	1	Register R30 bit 1	#1381	1	Register R31 bit 1
#1366	1	Register R30 bit 2	#1382	1	Register R31 bit 2
#1367	1	Register R30 bit 3	#1383	1	Register R31 bit 3
#1368	1	Register R30 bit 4	#1384	1	Register R31 bit 4
#1369	1	Register R30 bit 5	#1385	1	Register R31 bit 5
#1370	1	Register R30 bit 6	#1386	1	Register R31 bit 6
#1371	1	Register R30 bit 7	#1387	1	Register R31 bit 7
#1372	1	Register R30 bit 8	#1388	1	Register R31 bit 8
#1373	1	Register R30 bit 9	#1389	1	Register R31 bit 9
#1374	1	Register R30 bit 10	#1390	1	Register R31 bit 10
#1375	1	Register R30 bit 11	#1391	1	Register R31 bit 11
#1376	1	Register R30 bit 12	#1392	1	Register R31 bit 12
#1377	1	Register R30 bit 13	#1393	1	Register R31 bit 13
#1378	1	Register R30 bit 14	#1394	1	Register R31 bit 14
#1379	1	Register R30 bit 15	#1395	1	Register R31 bit 15

(2) Macro interface by part system (output)

System variable	No. of points	Interface output signal						
		$1	$2	$3	$4	$5	$6	$7
		R170	R270	R370	R470	R570	R670	R770
#1100	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1101	1	bit1	bit1	bit1	bit1	bit1	bit1	bit1
#1102	1	bit2	bit2	bit2	bit2	bit2	bit2	bit2
#1103	1	bit3	bit3	bit3	bit3	bit3	bit3	bit3
#1104	1	bit4	bit4	bit4	bit4	bit4	bit4	bit4
#1105	1	bit5	bit5	bit5	bit5	bit5	bit5	bit5
#1106	1	bit6	bit6	bit6	bit6	bit6	bit6	bit6
#1107	1	bit7	bit7	bit7	bit7	bit7	bit7	bit7
#1108	1	bit8	bit8	bit8	bit8	bit8	bit8	bit8
#1109	1	bit9	bit9	bit9	bit9	bit9	bit9	bit9
#1110	1	bit10	bit10	bit10	bit10	bit10	bit10	bit10
#1111	1	bit11	bit11	bit11	bit11	bit11	bit11	bit11
#1112	1	bit12	bit12	bit12	bit12	bit12	bit12	bit12
#1113	1	bit13	bit13	bit13	bit13	bit13	bit13	bit13
#1114	1	bit14	bit14	bit14	bit14	bit14	bit14	bit14
#1115	1	bit15	bit15	bit15	bit15	bit15	bit15	bit15

System variable	No. of points	Interface output signal						
		$1	$2	$3	$4	$5	$6	$7
		R171	R271	R371	R471	R571	R671	R771
#1116	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1117	1	bit1	bit1	bit1	bit1	bit1	bit1	bit1
#1118	1	bit2	bit2	bit2	bit2	bit2	bit2	bit2
#1119	1	bit3	bit3	bit3	bit3	bit3	bit3	bit3
#1120	1	bit4	bit4	bit4	bit4	bit4	bit4	bit4
#1121	1	bit5	bit5	bit5	bit5	bit5	bit5	bit5
#1122	1	bit6	bit6	bit6	bit6	bit6	bit6	bit6
#1123	1	bit7	bit7	bit7	bit7	bit7	bit7	bit7
#1124	1	bit8	bit8	bit8	bit8	bit8	bit8	bit8
#1125	1	bit9	bit9	bit9	bit9	bit9	bit9	bit9
#1126	1	bit10	bit10	bit10	bit10	bit10	bit10	bit10
#1127	1	bit11	bit11	bit11	bit11	bit11	bit11	bit11
#1128	1	bit12	bit12	bit12	bit12	bit12	bit12	bit12
#1129	1	bit13	bit13	bit13	bit13	bit13	bit13	bit13
#1130	1	bit14	bit14	bit14	bit14	bit14	bit14	bit14
#1131	1	bit15	bit15	bit15	bit15	bit15	bit15	bit15

System variable	No. of points	Interface output signal						
		$1	$2	$3	$4	$5	$6	$7
#1132	32	R170, R171	R270, R271	R370, R371	R470, R471	R570, R571	R670, R671	R770, R771
#1133	32	R172, R173	R272, R273	R372, R373	R472, R473	R572, R573	R672, R673	R772, R773
#1134	32	R174, R175	R274, R275	R374, R375	R474, R475	R574, R575	R674, R675	R774, R775
#1135	32	R176, R177	R276, R277	R376, R377	R476, R477	R576, R577	R676, R677	R776, R777

13.5.4.4 Tool Offset

Detailed description

Tool compensation data can be read and set using the variable numbers.

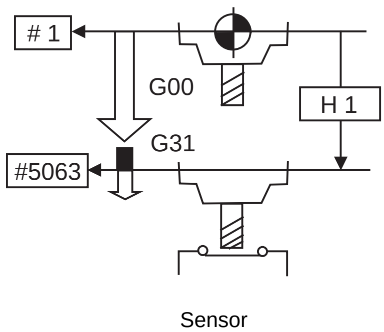
Variable number range		Type 1	Type 2
#10001 to #10000+n	#2001 to #2000+n	○	○(Length dimension)
#11001 to #11000+n	#2201 to #2200+n	×	○(Length wear)
#16001 to #16000+n	#2401 to #2400+n	×	○(Radius dimension)
#17001 to #17000+n	#2601 to #2600+n	×	○(Radius wear)

Tool data can be read and values substituted using the variable numbers.
Either the numbers in the #10000 order or #2000 order can be used.
The last 3 digits of the variable numbers correspond to the tool offset number.
n corresponds to the No. of tool offset sets.
If there are 400 tool offset sets and type 2 is being used, avoid variable Nos. in the #2000 order, and instead use the #10000 order.
The tool offset data are configured as data with a decimal point in the same way as for other variables.
Consequently, this decimal point must be commanded when data below the decimal point is to be entered.

Programming example	Common variables	Tool offset data
#101=1000; #10001=#101; #102=#10001;	#101=1000.0 #102=1000.0	H1=1000.000

(Example 1) Calculation and tool offset data setting

G28 Z0 T01 ;	Zero point return
M06;	Tool change (T0101)
#1=#5003 ;	Start point memory
G00 Z-500. ;	Rapid traverse to safe position
G31 Z-100. F100;	Skip measurement
#10001=#5063-#1 ;	Measurement distance calculation and tool offset data setting



(Note 1) In (Example 1), no consideration is given to the delay in the skip sensor signal.
#5003 is the Z axis start point position and #5063 indicates the position at which the skip signal is input while G31 is being executed in the Z axis skip coordinates.

13.5.4.5 Workpiece Coordinate System Offset (#5201 - #532n)

Detailed description

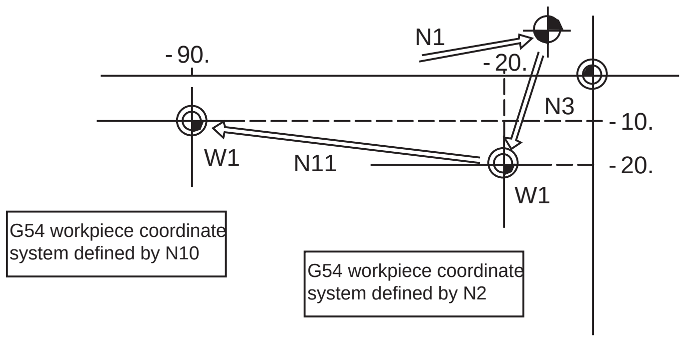
By using variable Nos #5201 to #532n, it is possible to read out the workpiece coordinate system compensation data or to substitute values.

(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.

Coordinate name	1st axis	2nd axis	3rd axis	4th axis	.....	nth axis	Remarks
External workpiece offset	#5201	#5202	#5203	#5204	.....	#520n	External workpiece offset specifications are required.
G54	#5221	#5222	#5223	#5224	.....	#522n	Workpiece coordinate system offset specifications are required.
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	

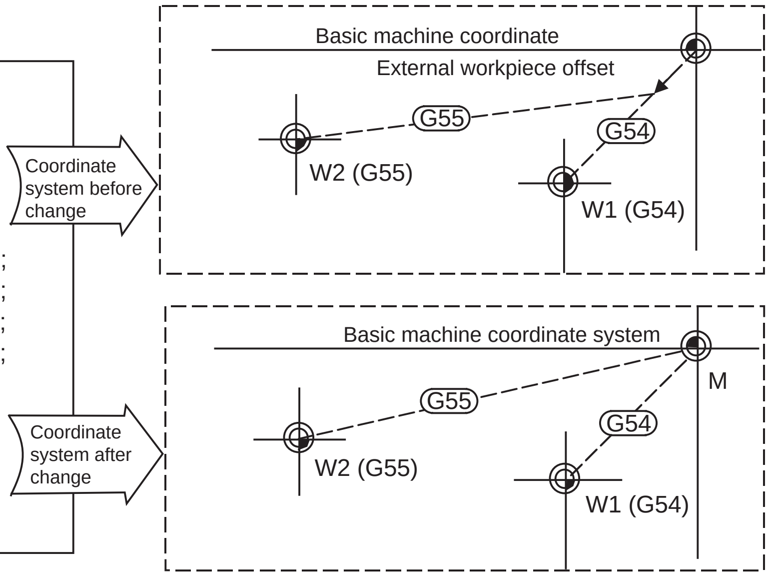
(Example 1)

```
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 X0Y0 ;

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 compensation values are added to the workpiece coordinate (G54, G55) system compensation values without changing the position of the workpiece coordinate systems.

13.5.4.6 NC Alarm (#3000)

Detailed description

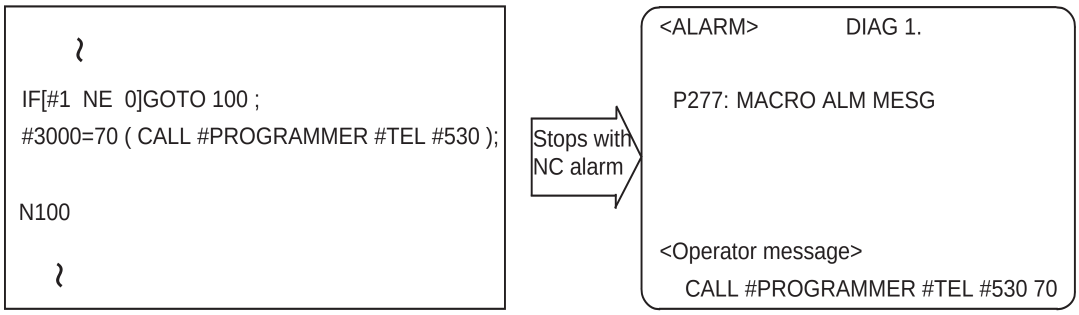
The NC unit can be forcibly set to the alarm state by using variable No. #3000.

#3000= 70 (CALL #PROGRAMMER #TEL #530) ;

70	Alarm No.
CALL #PROGRAMMER #TEL #530	Alarm message

Any alarm number from 1 to 9999 can be specified.
The alarm message must be written in 31 or less characters.
NC alarm 3 signal (program error) is output.
The "P277: MACRO ALM MESG" appears in the <ALARM> column on "DIAG 1." screen and the alarm No. and alarm message "70: (CALL #PROGRAMMER #TEL #530)"will appear in the <Operator massage>.

Example of program (alarm when #1 = 0)



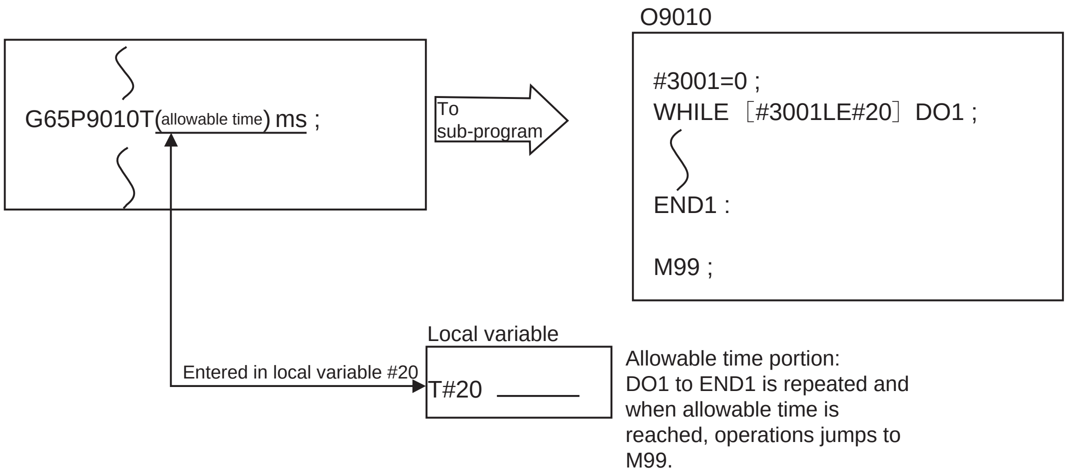
(Note 1) Alarm No. does not display 0. Any number exceeding 9999 cannot be displayed.
(Note 2) The characters following the first alphabet letter in the right member is treated as an alarm message. Therefore, a number cannot be designated as the first character of an alarm message. It is recommended that the alarm messages be enclosed in round parentheses.
(Note 3) Only the system that the alarm numbers and alarm messages are issued in #3000 can be forcibly set to the alarm state. However, operator messages will be displayed on each screen as they are common system.

13.5.4.7 Integrating Time (#3001, #3002)

Detailed description

The integrating time during the power is turned ON or the automatic start is running, can be read or values can be substituted by using variable Nos. #3001 and #3002.

Type	Variable No.	Unit	Contents when power is switched on	Initialization of contents	Count condition
Power ON	3001	1ms	Same as when power is switched off	Substitute values to variables	At all times while power is ON
Automatic start	3002				In-automatic start



13.5.4.8 Suppression of Single Block Stop and Miscellaneous Function Finish Signal Waiting (#3003)

Detailed description

By substituting the values below in variable No. #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	Wait
1	Suppressed	Wait
2	Not suppressed	Not wait
3	Suppressed	Not wait

(Note 1) Variable No. #3003 is set to zero by NC reset.

13.5.4.9 Feed Hold, Feedrate Override, G09 Valid/Invalid (#3004)

Detailed description

By substituting the values below in variable No. #3004, it is possible to make the feed hold, feedrate override and G09 functions either valid or invalid in the subsequent blocks.

Contents (value)	#3004		
	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 No. #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.

13.5.4.10 Message Display and Stop (#3006)

Detailed description

By using variable No. #3006, the operation stops after the previous block is executed and, if message display data is commanded, the corresponding message will be indicated on the operator message area.

```
#3006 =  1( TAKE FIVE );
```

TAKE FIVE	Message

The message should be written in 31 or less characters and should be enclosed by round parentheses.

(Note 1) Only "1" is valid to set the number on #3006. Block Stop is not possible and the messages will not be displayed if the number other than "1" is set.

(Note 2) Block stop is possible only for the system which is issued to set "1" on #3006, however, the operator messages are displayed on each part system's alarm screen as they are common to part systems.

13.5.4.11 Mirror Image (#3007)

Detailed description

By reading variable No. #3007, it is possible to ascertain the status of mirror image of the each axis at the point.
Each axis corresponds to a bit of #3007.
When the bits are 0, the mirror image function is invalid.
When the bits are 1, the mirror image function 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

13.5.4.12 G Command Modals (#4001-#4021, #4201-#4221)

Detailed description

Using variable Nos. #4001 to #4021, it is possible to read the modal commands which have been issued in previous blocks.
Similarly, it is possible to read the modals in the block being executed with variable Nos. #4201 to #4221.

Variable No.		Function	
Pre-read block	Execution block		
#4001	#4201	Interpolation mode	G00 : 0, G01 : 1,G 02 : 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 radius compensation	G40 : 40, G41 : 41, G42 : 42
#4008	#4208	Tool length offset	G43:43, G44:44, G49:49
#4009	#4209	Fixed cycle	G80 : 80, G73-74 : 73-74, G76 : 76, G81-89 : 81-89
#4010	#4210	Return level	G98 : 98, G99 : 99
#4011	#4211		
#4012	#4212	Workpiece coordinate system	G54-G59 : 54-59
#4013	#4213	Acceleration/deceleration	G61-G64 : 61-64, 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 P300 X100. Y100. ;
M02 ;
O300
#1 = #4003 ;   = -> Group 03 G modal (pre-read) #1 = 91.0
#2  = #4203 ;    = -> Group 03 G modal (now being executed) #2 = 90.0
G#1 X#24 Y#25 ;
M99 ;
%
```

13.5.4.13 Other Modals (#4101 - #4120, #4301 - #4320)

Detailed description

Using variable Nos. #4101 to #4120, it is possible to read the modal commands which have been issued in previous blocks.
Similarly, it is possible to read the modals in the block being executed with variable Nos. #4301 to #4320.

Variable No.		Modal information		Variable No.		Modal information
Pre-read	Execution			Pre-read	Execution	
#4101	#4301			#4111	#4311	Tool length offset 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 O
#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

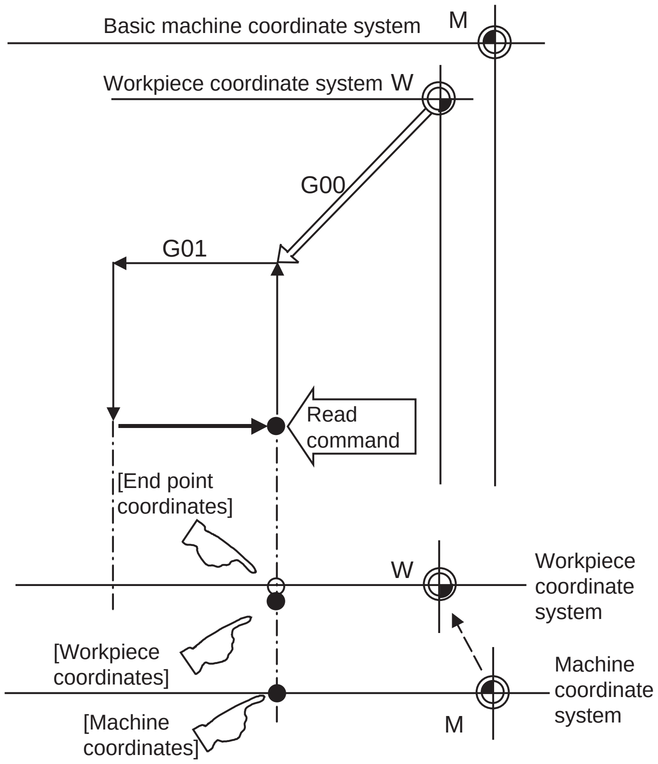
13.5.4.14 Position Information (#5001 - #5140 + n)

Detailed description

Using variable Nos. #5001 to #5104, it is possible to read the end point coordinates, machine coordinates, workpiece coordinates, skip coordinates and servo deviation amounts in the last block.

Position information	Axis No.						Remarks (reading during movement)
	1	2	3	4	...	n	
End point coordinate of the last block	#5001	#5002	#5003	#5004	...	#5000+n	Yes
Machine coordinate	#5021	#5022	#5023	#5024	...	#5020+n	No
Workpiece coordinate	#5041	#5042	#5043	#5044	...	#5040+n	No
Skip coordinate	#5061	#5062	#5063	#5064	...	#5060+n	Yes
Servo deviation amount	#5101	#5102	#5103	#5104	...	#5100+n	Yes

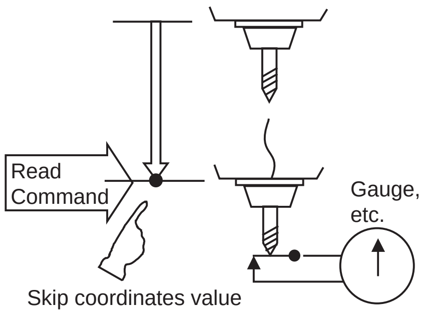
(Note) The number of axes which can be controlled differs according to the specifications.
The last digit of the variable No. corresponds to the control axis No.



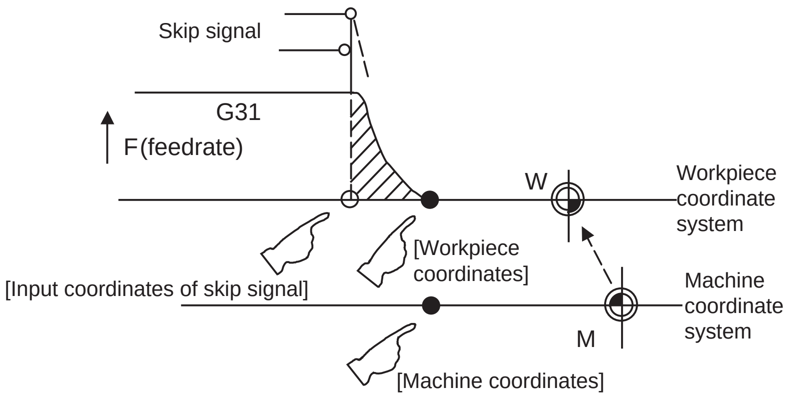
(1) The positions of the end point coordinates and skip coordinates are positions in the workpiece coordinate system.

(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 skip coordinates indicates the position where the skip signal is turned ON in the G31 block. If the skip signal does not turn ON. they will be the end point position.
(For further details, refer to the section on Automatic Tool Length Measurement.)



(4) The end point coordinates indicate the tool nose position regardless of the tool compensation and other such factors. On the other hand, the machine coordinates, workpiece coordinates and skip coordinates indicate the tool reference point position with consideration given to tool compensation.

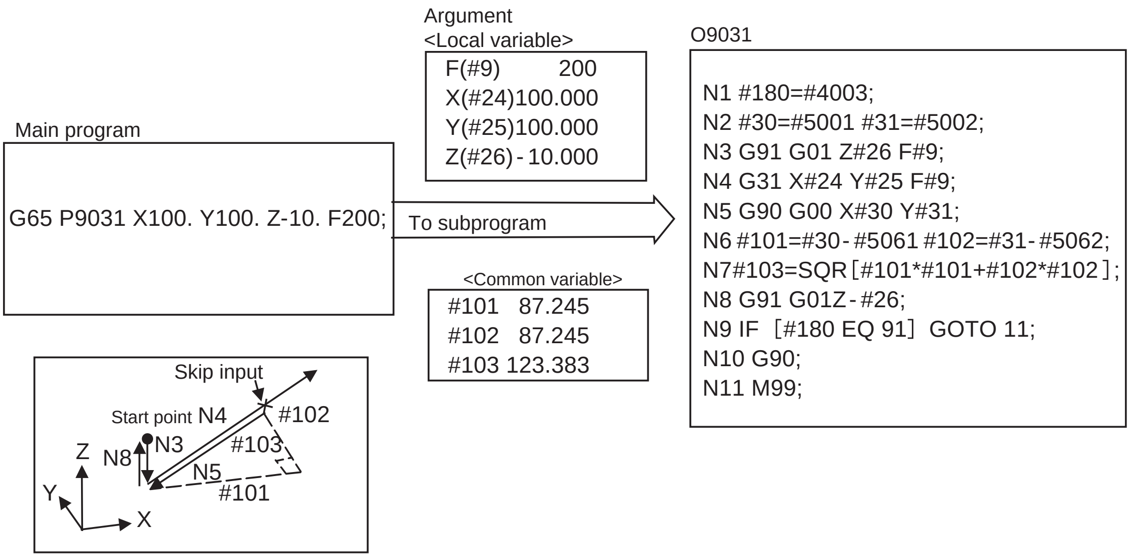


For " ● ", check stop and then proceed to read.
For " ○ ", reading is possible during movement.

The skip signal input coordinates value is the position in the workpiece coordinate system. The coordinate value in variable Nos. #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, refer to the section on "Skip Function".

(Example 1) Example of workpiece position measurement

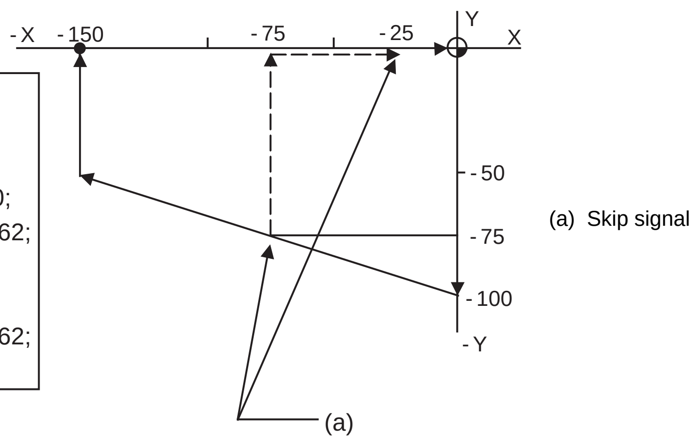
An example to measure the distance from the measured reference position to the workpiece edge is shown below.



#101	X axis measurement amount	N1	G90/G91 modal recording
#102	Y axis measurement amount	N2	X, Y start point recording
#103	Measurement linear segment amount	N3	Z axis entry amount
		N4	X, Y measurement (Stop at skip input)
#5001	X axis measurement start point	N5	Return to X, Y start point
#5002	Y axis measurement 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	Main program return

(Example 2) Reading of skip input coordinates

```
N1 G91 G28 X0 Y0;
N2 G90 G00 X0 Y0;
N3 X0 Y-100.;
N4 G31 X-150. Y-50. F80;
N5 #111=#5061#112=#5062;
N6 G00 Y0;
N7 G31 X0;
N8 #121=#5061#122=#5062;
N9 M02;
```



#111=-75.+ε	#112=-75.+ε
#121=-25.+ε	#122=-75.+ε

ε 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.

13.5.4.15 External workpiece coordinate offset (#2501, #2601)

Detailed description

The external workpiece coordinate offset can be read using variables #2501 and #2601. By substituting a value in these variables, the external workpiece coordinate offset can be changed.

Axis No.	External workpiece coordinate offset
1	#2501
2	#2601

13.5.4.16 Number of Workpiece Machining Times (#3901, #3902)

Detailed description

The number of workpiece machining times can be read using variables #3901 and #3902.
By substituting a value in these variable Nos., the number of workpiece machining times can be changed.

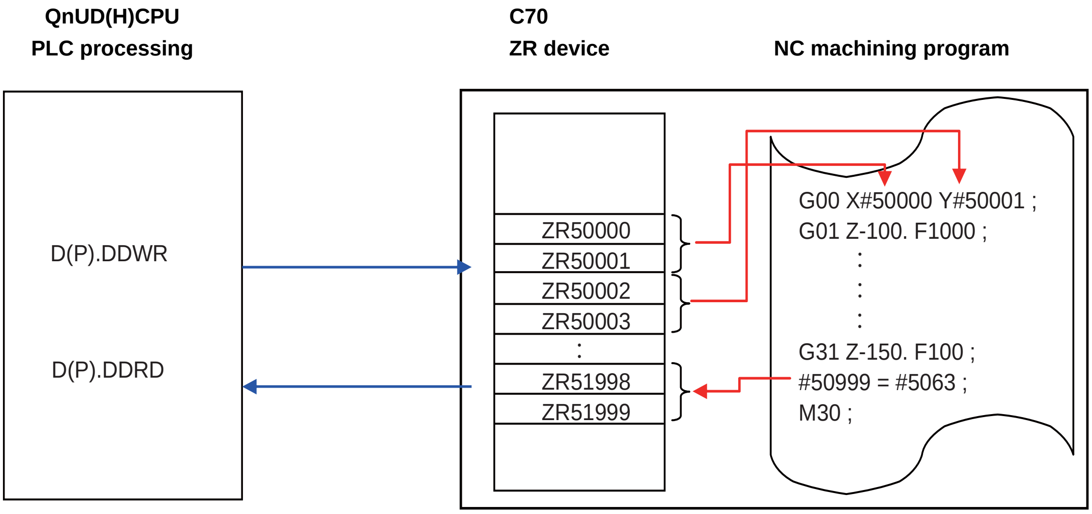
Type	Variable No.	Data setting range
Number of workpiece machining times	#3901	0 to 999999
Maximum workpiece value	#3902	

(Note) The number of workpiece machining times must be a positive value.

13.5.4.17 ZR device access variable

Detailed description

By using variable Nos.#50000 to #51199, machining programs can read the ZR device data in NC side that is accessible with PLC I/F command, in the NC side or values can be substituted from QnUD(H)CPU.



ZR device access variable and ZR device No. correspondence table

Variable Nos.	ZR device Nos.
#50000	ZR50000,ZR50001
#50001	ZR50002,ZR50003
#50002	ZR50004,ZR50005
:	:
#5000 + n	ZR50000+2n, ZR50000+2n+1
:	:
#51199	ZR52398,ZR52399

These variables can be decimal points valid or invalid depending on the variable numbers as shown in the table below.

Decimal points Valid/Invalid	Variable Nos.	The number of sets
Valid	#50000 to #50999	1000
Invalid	#51000 to #51199	200

Decimal point position can be varied depending on the parameter "#1003 iunit (input setting units)" and "#1041 I_inch (initial inch)" settings. When setting the values in the ZR device, consider these parameter settings and then decide the decimal point position.

Readout variables

(1) For decimal points valid variable numbers (#50000 to #50999)
It is used for addresses (X,Y,Z,U,V,W,A,B,C,I,J,K,F,E,Q,R) where a decimal point is valid such as the distances, angles, times and speeds in machining programs.

When a variable is used as below in machining programs, the data set in the device ZR50000 and ZR50001 will be referred.

G0 X#50000 ;

ZR50000, ZR50001

Example of relationship between the setting value for ZR device and the variable commands

Device	Value	#50000	Parameter	Command
ZR50001	0x0001	0x1e240(hexadecimal) = 123456(decimal)	iunit = B I_inch = 0	X123.456
ZR50000	0xe240			

As the variable #50000 is decimal points valid variable number, it means the same as the decimal points valid variable number command as shown in the "Command" column in the table above.

Input unit "iunit" for the valid decimal point address and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command position when "123456" is set to #50000 by X#50000 (where the X axis is the linear axis) command	X123.456	X12.3456	X12.3456	X1.23456
Command position when "123456" is set to #50000 (where the C axis is the rotary axis) command	C123.456	C12.3456	C12.346 (Note)	C1.2346 (Note)

(Note) For rotary axis, the displayed number of digits below the decimal point is the same as the metric system when the inch system is selected although the decimal point position of the data is the same as the linear axis.
Therefore, if "123456" is set to ZR50000,1 and inunit = B the decimal point position is the same as "C12.3456" is commanded when the rotation axis is commanded. Similarly, if inunit = C the decimal point position is the same as "C12.3456" is commanded.

Among the addresses where the decimal point command is valid, the number of digit of below the decimal point for address F is different from other addresses.
The number of digit for below the significant decimal point will be rounded off to the nearest increment.

Input unit "iunit" for address F and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
Decimal point position	2 digits below the decimal point	3 digits below the decimal point	3 digits below the decimal point	4 digits below the decimal point
Command speed when "123456" is set to #50010 by F#50010 command	F123.46	F12.346	F12.346	F1.2346

When using these variable numbers for the decimal point invalid addresses (D,H,L,M,N,O,S,T) as shown below, the value which is rounded off below the decimal point in the variable becomes the command value.

N#50010 ;

Input unit "iunit" for the decimal point invalid addresses and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command when "234567" is set to #50010 by F#50010 command	N235	N23	N23	N2

The following table shows the valid range of these variables for the addresses where the decimal point command is valid.

The valid range of these variables for the addresses where the decimal point command is valid

	Movement command (linear)	Movement command (rotation)	Feedrate (Note 1)	Dwell (Note 1)
mm	-99999999 to 99999999	-99999999 to 99999999	-1000000000 to 1000000000	-99999999 to 99999999
inch	-99999999 to 99999999	-99999999 to 99999999	-39370.0787 to 39370.0787	-99999999 to 99999999

(Note 1) "-" will be ignored even "-" is set to Feedrate or Dwell.

(Note 2) The program error (P35) will occur if the command is executed by setting the value, which exceeds the address command range, to these variable numbers.

(2) For variable number commanded is decimal point invalid (#51000 to #51199)
As shown in the example of program below, the data which is set to these variables becomes the command value when these variables are used for the decimal point invalid addresses (D,H,L,M,N,O,S,T) regardless of parameter "#1003 iunit (input unit)" or "#1041 I_inch (initial inch)".

S#51000 ;

Input unit "iunit" for the decimal point invalid addresses and the decimal point position with mm/inch

	Metric system (I_inch = 0)		**Inch system (I_inch = 1)**	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command when "500" is set to #51000 by S#51000 command	S500	S500	S500	S500

Valid range of these variables for decimal point invalid address is within the address command range for variables. The program error (P35) will occur if a value exceeding the command range set to the variable and executed.
If these variables are executed to the decimal point valid address as shown in the example of program below, the command value is expressed as below since these variables will be treated as the data with a decimal point.

X#51000 ;

Input unit "iunit" for the decimal point valid addresses and the decimal point position with mm/inch

	Metric system (I_inch = 0)		**Inch system (I_inch = 1)**	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command when "500" is set to #51000 by X#51000 command	X500.000	X500.0000	X500.0000	X500.00000

Substituting in variables

(1) For variable numbers which are decimal point valid (#50000 to #50999)

When substituting value without decimal point into decimal point valid variable as below, the value will be shifted by the number of the digits in fractional part of the value whose unit is set by the parameter #1003 and be set to ZR devices, regardless of the "#1078 Decpt2(Decimal point type 2)" setting.

#50100 = 123 ;

→ ZR50200, ZR50201

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		**Inch system (I_inch = 1)**	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #50100 by #50100 = 123 command	123000	1230000	1230000	12300000

When the value with decimal point is substituted into the variable as shown below, the value with decimal point will be substituted.

#50101 = 987.654 ;

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		**Inch system (I_inch = 1)**	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #50101 by #50101 = 987.654 command	987654	9876540	9876540	98765400

When a variable with decimal point such as a coordinate variable or a common variable is substituted into the decimal point valid variable as shown below, the coordinate variable or the common variable will be substituted.

#50200 = #5063 (#5063: Skip coordinate)

The table below lists the values which are set to the ZR devices when variable #5063 is substituted into variable #50200.

Substituted value into ZR device when variable is substituted

iunit	Metric system (I_inch = 0) Readout value of #5063	Inch System (I_inch = 1) Readout value of #5063	Substituted value for #50200	Value	Device
B	-123.456	-12.3456	-123456 = 0xfffe1dc0	0xfffe	ZR50401
C	-12.3456	-1.23456		0x1dc0	ZR50400

(2) For variable numbers with decimal point invalid (#51000 to #51199)
When the value without decimal point is substituted into the decimal point invalid variable as shown below, the substituted value is set to ZR device regardless of parameter "#1003 iunit (input unit)", "1041 I_inch (initial inch)", "#1078 Decpt2 (decimal point type 2)".

#51100 = 123 ;

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #51100 by #51100 = 123 command	123	123	123	123

When the value with decimal point is substituted into the variable, the value which is rounded off decimal point will be substituted.

#51101 = 7.543 ; or #51101 = 7.456 ;

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #51101 by #51101 = 73543 command	8	8	8	8
Substituted value into #51101 by #51101 = 73546 command	7	7	7	7

When a variable with decimal point such as a coordinate variable or a common variable is substituted into the decimal point valid variable as shown below, the value which is rounded off after decimal point will be substituted.

#51102 = #5021 ;

Using ZR device access variables in user control statement

ZR device access variables can be used in user macro control statement.
Note that the variable data and true/false of the conditions are different between when using decimal point valid variables (#50000 to #50999) and when using decimal point invalid variables (#51000 to #51199).

(1) For variable number with decimal point valid (#50000 to #50999)
(Example)
IF [#50100 EQ 1] GOTO 30 ;
G00 X100. ;
N30 ;
The table below lists #50100’s values that allows the condition of the user macro control statement as shown above to be true.

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
The #50100's value to jump to N30 with control statement #50100 EQ 1.	1000	10000	10000	100000

(2) For variable number with decimal point invalid (#51000 to #51199)
(Example)
IF [#51100 EQ 1] GOTO 30 ;
G00 X100. ;
N30 ;
The table below lists #51100's values that allow the condition of the user macro control statement as shown above to be true.

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
The #50100's value to jump to N30 with control statement #50100 EQ 1.	1	1	1	1

When using a decimal point invalid variable in user macro control statement, the condition becomes true regardless of the parameter "#1003 iunit()" and "#1041 I_inch(Inicial state(inch))" settings.

Common variable and substitution between ZR device access values

(1) For decimal point valid variable (#50000 to #50999)
When a common variable is substituted into a decimal point valid variable, the value is substitute by the setting of parameter "#1003 iunit (input unit)", "#1041 I_inch (initial inch)" as show below.
#101 = 123.45678 ;
#50200 = #101 ;

The value of ZR device access variable when a common variable substituted

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
#101 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #101 by #101 = 123.45678 command	123.4568	123.4568	123.4568	123.4568
#502000 decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #50200 by #50200 = #101 command	123457	1234568	1234568	12345678

The number of decimal of the data is treated differently in between ZR devices which is long type, and common variables which is double type.

When the decimal point valid ZR device access variable is substituted into common variable as shown in the table below, the value is substituted by the settings of parameter "#1003 iunit (input setting unit)", "#1041 I_inch (initial inch)".

#50201 = 123.45678 ;
#102 = #50201 ;

Common variable value when ZR device access variable is substituted into common variable.

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	**iunit = C**	**iunit = B**	**iunit = C**
#50201 decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #502001 by #50201 = 123.45678 command	123457	1234568	1234568	12345678
#102 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #102 by #102 = #50201 command	123.4570	123.4568	123.4568	123.4568

(2) For decimal point invalid variable (#51000 to #51199)
When a common variable is substituted into the decimal point invalid variable, the value which is rounded off below the decimal point is substituted regardless of parameter "#1003 iunit (input unit)", "#1041 I_inch (initial inch)".

#101 = 123.4567 ;
#51100 = #101 ;

ZR device access variable data when a common variable is substituted

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
#101 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #101 by #101 = 123.4567 command	123.4567	123.4567	123.4567	123.4567
#51100 decimal point position	No value below the decimal point	No value below the decimal point	No value below the decimal point	No value below the decimal point
Substituted value into #51100 by #51100 = #101 command	123	123	123	123

#101 = 123.5432 ;
#51100 = #101 ;

ZR device access variable data when a common variable is substituted

	Metric system (I_inch = 0)		Inch system (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
#101 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #101 by #101 = 123.5432 command	123.5432	123.5432	123.5432	123.5432
#51100 decimal point position	No value below the decimal point	No value below the decimal point	No value below the decimal point	No value below the decimal point
Substituted value into #51100 by #51100 = #101 command	124	124	124	124

13.5.4.18 Tool Life Management (#60000 - #64700)

Detailed description

Definition of variable Nos.

(1) 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 No. If a group No. is not designated, the data of the group registered first is read.This is valid until reset.

(2) Tool life management system variable No. (Read)
#60001 to #64700
| a | b | c | d | e |

| a | : "6" Fix (Tool life management)

| b | c | : Details of data classification

Data class	Details	Remarks
00	For control	Refer by data types
05	Group No.	Refer by registration No.
10	Tool No.	Refer by registration No.
15	Tool data flag	Refer by registration No.
20	Tool status	Refer by registration No.
25	Life data	Refer by registration No.
30	Usage data	Refer by registration No.
35	Tool length offset data	Refer by registration No.
40	Tool radius compensation data	Refer by registration No.
45	Auxiliary data	Refer by registration No.

The group No. and life data are common for the group.

| d | e | : Registration No. or data type
Registration No.

1 to 200

Data type

Type	Details
1	Number of registered tools
2	Life current value
3	Tool selection No.
4	Number of remaining registered tools
5	Execution signal
6	Cutting time cumulative value (min)
7	Life end signal
8	Life prediction signal

List of variables

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 min 0 to 9999 times
60003	Tool selection No.	For each group (Designate Group No. #60000)	Registration No. of tool being used. Designated group's selected tool registration No. (If a tool is not selected, the first tool of ST:1, 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	For each group (Designate Group No. #60000)	No. of first registered tool that has not reached its life.	0 to 200
60005	Execution signal	For each group (Designate Group No. #60000)	"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 (min)	For each group (Designate Group No. #60000)	Indicates the time that this group is used in the program being executed.	(Not used)
60007	Life end signal	For each group (Designate Group No. #60000)	"1" when lives of all tools in this group have expired. "1" when all registered tools in the designated group reach lives.	0/1
60008	Life prediction signal	For each group (Designate Group No. #60000)	"1" when selecting a new tool with the 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. (Group No. #60000/ registration No. *** is designated.) Note that the group No./ method and life are common for the groups.	This group's No.	1 to 99999999
61000 +***	Tool No.		Tool No.	1 to 99999999
61500 +***	Tool data flag		Parameters such as usage data count method, length compensation method, and radius compensation method. 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 : Tool not used 1 : Tool in use 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 min 0 to 9999 times
63000 +***	Usage data		Usage time or No. of uses for each tool	0 to 4000 min 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

Program example

(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.
                          Designated program No. is valid until reset.
#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 tool No.
%
```

(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 ; ...... #111 =-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.
                         The group 10 life data is registered through the commands from G10 to G11.
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 life data with the G10 command.
#111 = #61001 ;   .....  Reads the group 10, #1 tool No.
G10 L3 ;          .....  Starts the life management data registration.
                         The group 10 life data is registered through the commands from G10 to G11.
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.  #111 = -1.
```

Precautions

(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.

13.5.5 Operation Commands

Function and purpose

A variety of 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 operation	#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 operation	#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 tan θ uses sinθ/cosθ.
	#i = ASIN [#k]	Arcsine (Note 4)
	#i = ATAN [#k]	Arctangent (ATAN or ATN may be used)
	#i = ACOS [#k]	Arccosine
	#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 compensation 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)

Operation Commands	Common variables after execution
#101 =1000 ; #10001 =#101 ; #102 =#10001 ;	#101 1000.000 #102 1000.000

(Note 3) The <formula> after a function must be enclosed in the square parentheses [ ].

(Note 4) Operation results differ depending on the setting of the parameter "#1273 ext09/bit0".

Detailed description

Sequence of operations

(1) The sequence of the operations (a) to (c) is performed in the following order; the function, the multiplication operation and the addition operation.

```
#101=#111+#112*SIN〔#113〕
```

(a) Function
(b) Multiplication operation
(c) Addition operation

(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=SQRT〔〔〔#111-#112〕*SIN〔#113〕+#114〕*#115〕;
```

First pair of brackets
Second pair of brackets
Third pair of brackets

Examples of operation 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 sub-stitution =	#1 = 1000 #2 = 1000.	#1 1000.000 #2 1000.000	
	#3 = #101 #4 = #102	#3 100.000 #4 200.000	From common variables
	#5 = #10001 (#10001 = -10.)	#5 -10.000	From offset amount
(3) Addition and sub-traction + -	#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.] (Note) SIN [60] is equivalent to SIN [60.]	#501 #502 #503 #504 #505 #506	0.866 0.866 866.025 866.025 866.025 866.025
(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.] (Note) COS [45] is equivalent to COS [45.]	#541 #542 #543 #544 #545 #546	0.707 0.707 707.107 707.107 707.107 707.107
(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.] (Note) TAN [60] is equivalent to TAN [60.]	#551 #552 #553 #554 #555 #556	1.732 1.732 1732.051 1732.051 1732.051 1732.051

(12) Arcsine ASIN	#531 = ASIN [100.500 / 201.] #532 = ASIN [100.500 / 201] #533 = ASIN [0.500] #534 = ASIN [-0.500]	#531 #532 #533 #534	30.000 30.000 30.000 -30.000	
		(Note) When #1273/bit 0 is set to 1, #534 will be 330°.		
(13) Arctangent ATN or ATAN	#561 = ATAN [173205 / 100000] #562 = ATAN [173205 / 100000.] #563 = ATAN [173.205 / 100] #564 = ATAN [173.205 / 100.] #565 = ATAN [1.73205]	#561 #562 #563 #564 #565	60.000 60.000 60.000 60.000 60.000	
(14) Arccosine ACOS	#521 = ACOS [100 / 141.421] #522 = ACOS [100. / 141.421]	#521 #522	45.000 45.000	
(15) Square root SQR or SQRT	#571 = SQRT [1000] #572 = SQRT [1000.] #573 = SQRT [10. * 10. + 20. * 20] (Note) In order to increase the accuracy, proceed with the operation inside parentheses as much as possible.	#571 #572 #573	31.623 31.623 22.360	
(16) Absolute value ABS	#576 = -1000 #577 = ABS [#576] #3 = 70. #4 = -50. #580 = ABS [#4 - #3]	#576 #577 #580	-1000.000 1000.000 120.000	
(17) BIN, BCD	#1 = 100 #11 = BIN [#1] #12 = BCD [#1]	#11 #12	64 256	
(18) Rounding off RND or ROUND	#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 #22 #23 #24 #25 #26 #27 #28	5 5 5 5 -5 -5 -5 -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 #22 #23 #24 #25 #26 #27 #28	4.000 4.000 4.000 4.000 -4.000 -4.000 -4.000 -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 #22 #23 #24 #25 #26 #27 #28	5.000 5.000 5.000 5.000 -5.000 -5.000 -5.000 -5.000	
(21) Natural logarithms LN	#10 = LN [5] #102 = LN [0.5] #103 = LN [-5]	#101 #102 Error	1.609 -0.693 "P282"	
(22) Exponents EXP	#104 = EXP [2] #105 = EXP [1] #106 = EXP [-2]	#104 #105 #106	7.389 2.718 0.135	

Sequence of operations

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	2.33×10^{-10}	5.32×10^{-10}	Min. |ε/b|, |ε/c|
a = b*c	1.55×10^{-10}	4.66×10^{-10}	Relative error |ε/a|
a = b/c	4.66×10^{-10}	1.86×10^{-9}	
a = √b	1.24×10^{-9}	3.73×10^{-9}	
a = SIN [b] a = COS [b]	5.0×10^{-9}	1.0×10^{-8}	Absolute error |ε|°
a = ATAN [b/c]	1.8×10^{-6}	3.6×10^{-6}	

(Note)　SIN/COS is calculated for the function TAN.

Precautions

(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.

13.5.6 Control Commands

Function and purpose

The flow of programs can be controlled by IF-GOTO- and WHILE-DO-END.

Detailed description

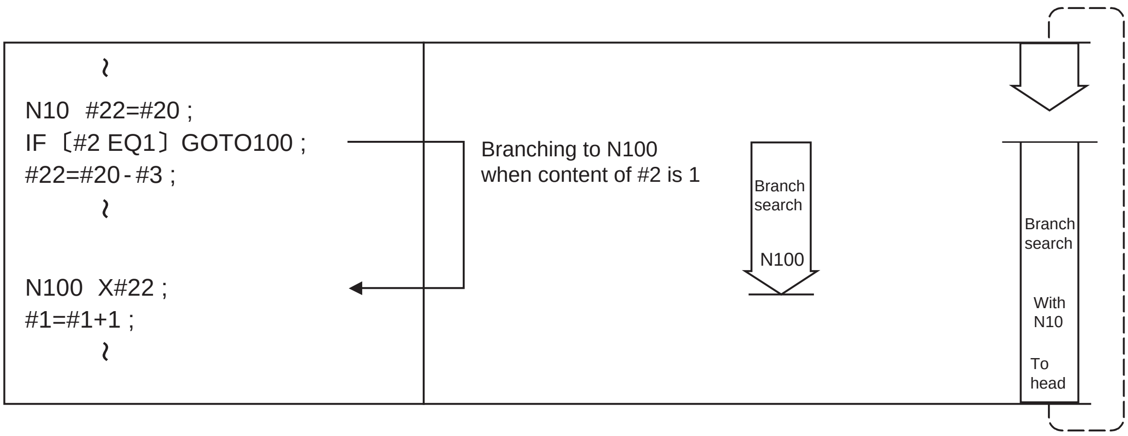
Branching

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 branches 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. If not, program error (P231) will occur. 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 "Nn" must always be at the head of the block. Otherwise, program error (P231) will occur.
If "/" is at the head of the block and "Nn" follows, control can be branched to the sequence number.



(Note 1) When searching the sequence number of the branch destination, 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 time compared with branch searches in the forward direction.

(Note 2) EQ and NE should be compared only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.

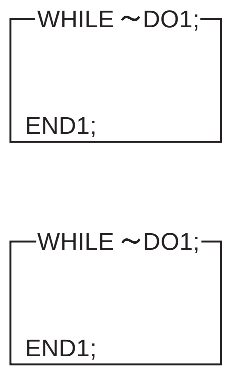
Repetitions

WHILE [conditional expression] DOm ; (m =1, 2, 3 ..... 127)
:
END m ;

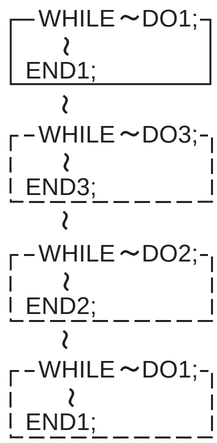
While the conditional expression is established, the blocks from the following block to ENDm are repeatedly executed; when it is not established, execution moves to the block following ENDm. DOm may come before WHILE.

"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 Nos. range from 1 to 127. (DO1, DO2, DO3, ... DO127) Up to 27 nesting levels can be used.

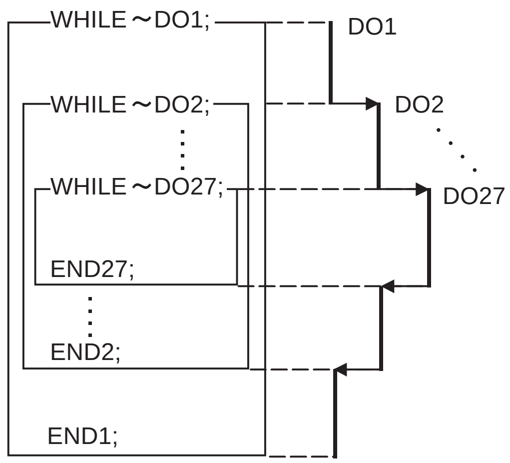
(1) Same identification No. can be used any number of times.



(2) Any number may be used as the WHILE-DOm identification No.

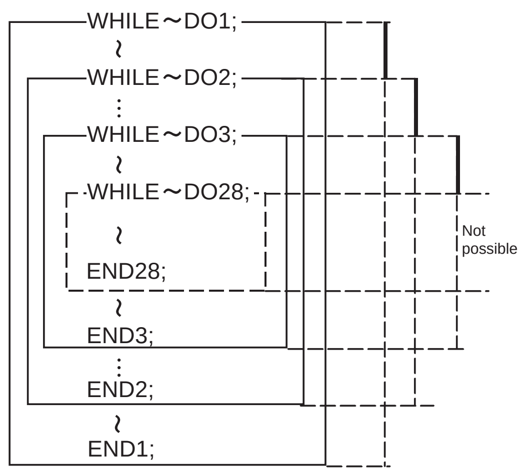


(3) Up to 27 nesting levels can be used for WHILE-DOm.
"m" is any number from 1 to 127 for the nesting depth.

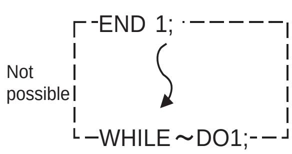


(Note) For nesting, "m" which has been used once cannot be used.

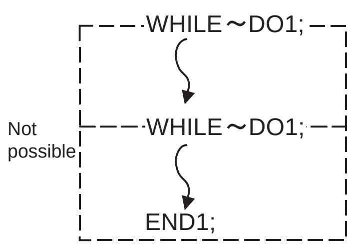
(4) The number of WHILE-DOm nesting levels cannot exceed 27.



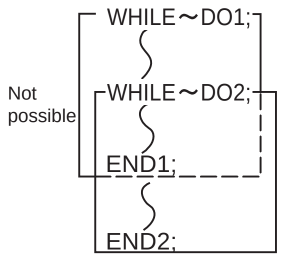
(5) WHILE - DOm must be designated first and ENDm last.



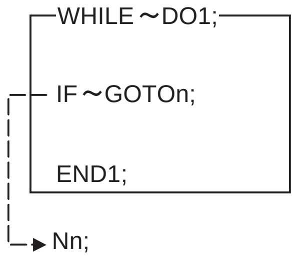
(6) WHILE - DOm and ENDm must correspond on a 1:1 (pairing) basis in a same program.



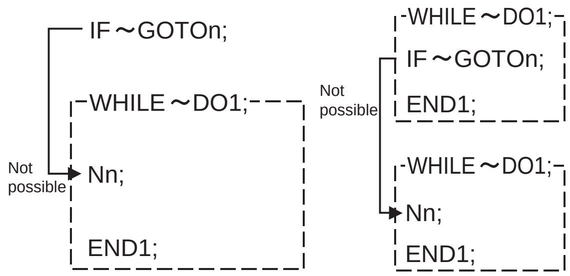
(7) Two WHILE - DOm's must not overlap.



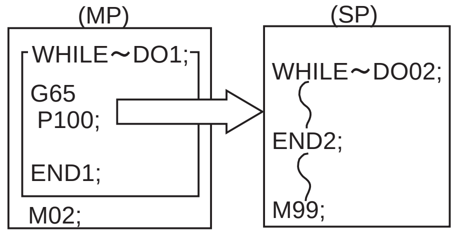
(8) Branching externally out of the WHILE - DOm range, is possible.



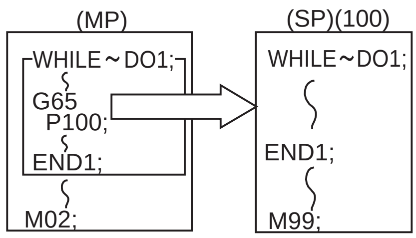
(9) No branching into WHILE - DOm, is possible.



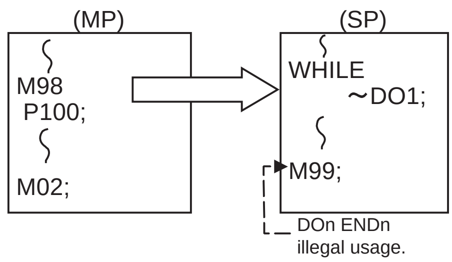
(10) Subprograms can be called by M98, G65 or G66 between WHILE - DOm's.



(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.



(12) A program error will occur in M99 if WHILE and END are not paired in the subprogram (including macro subprogram).



(MP) Main program (SP) Subprogram

(Note) As the fixed cycles G73 and G83 and the special fixed cycle G34 use WHILE, these will be added multiple times.

13.5.7 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.
By setting the parameter "#8101 macro single", the macro statements can be processed concurrently with the execution of the executable statement.
(During normal machining, set the parameter OFF to process all the macro statements together, and during a program check, set it ON to execute the macro statements block by block. Setting can be chosen depending on the purpose.)

Program example

```
N1 G91 G28 X0 Y0 ;                 ......(1)
N2 G92 X0 Y0 ;                     ......(2)
N3 G00 X-100. Y-100. ;             ......(3)
N4 #101 = 100. * COS[210.] ;       ......(4)
                                                  (4),(5) Macro statements
N5 #103 = 100. * SIN[210.] ;       ......(5)
N6 G01 X#101 Y#103 F800 ;          ......(6)
```

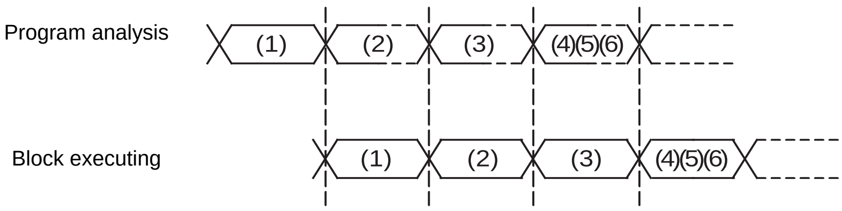
Macro statements are:

(a) Arithmetic commands (block including =)
(b) Control commands (block including GOTO, DO-END, etc.)
(c) Macro call commands (including macro calls based on G codes and cancel commands (G65, G66, G66.1, G67))

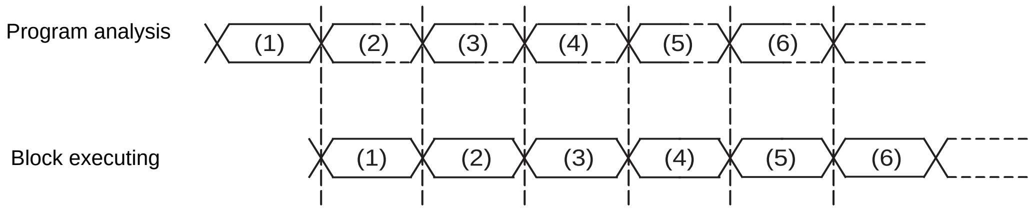
Execution statements refer to statements other than macro statements.

Flow of processing by the Program Example in the previous page

<Macro single OFF>



<Macro single ON>



Machining program display

<Macro single OFF>

[Executing] N3 G00 X-100. Y-100. ; [Next command] N6 G01 X#101 Y#103 F800 ;	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.

<Macro single ON>

[Executing] N3 G00 X-100. Y-100. ; [Next command] N4 #101=100. *COS[210.] ;	N4 is processed in parallel with the control of the executable statement of N3, and it is displayed as the next command. After N3 is finished, N5 and N6 are analyzed, and then N6 is executed. So the machine control is held on standby during the N5 and N6 analysis time.

13.5.8 Actual examples of using user macros

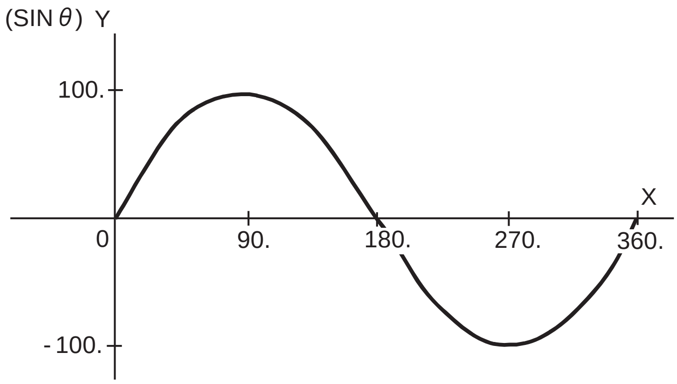
Program example

The following three examples will be described.
(Example 1) SIN curve
(Example 2) Bolt hole circle
(Example 3) Grid

(Example 1) SIN curve

G65 Pp1 Aa1 Bb1 Cc1 Ff1;

a1; Initial value 0°
b1; Final value 360°
c1; R of R*SINθ
f1; Feedrate



Main Program

```
G65 P9910 A0 B360.C100.F100;
```

To subprogram

O9910(Subprogram)

```
WHILE〔#1LE#2〕DO1;
#101=#3*SIN〔#1〕;
G90 G01 X#1 Y#10 F#9;
#1=#1+10.;
END1;
M99;
```

(Note 1)

Local variables set by argument

```
#1=0
#2=360.000
#3=100.000
#9=100.000
```

(Note 1) Commanding with one block is possible when G90G01X#1Y[#3*SIN[#1]] F#9; is issued.

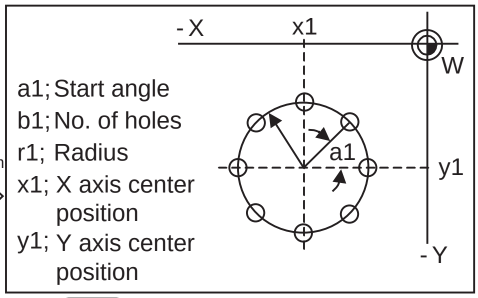
(Example 2) Bolt hole circle

After defining the hole data with fixed cycle (G72 to G89), the macro command is issued as the hole position command.

Main program

```
        ~
G81 Z-100.R50.F300L0
G65 P9920 Aa1 Bb1 Rr1 Xx1 Yy1;
        ~
```

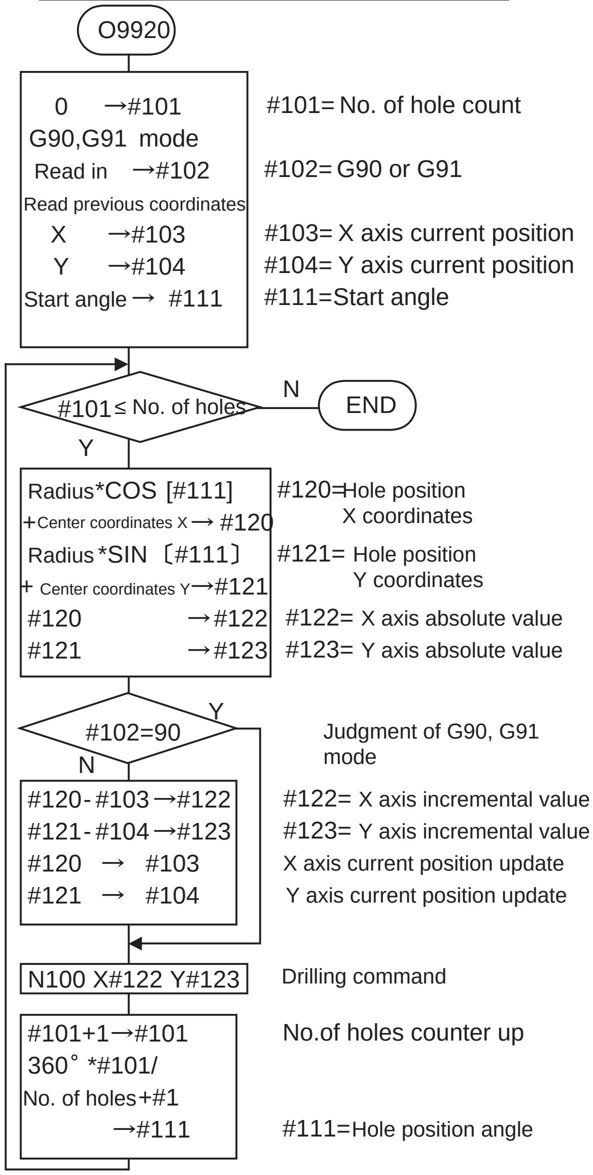
To subprogram



O9920(Subprogram)

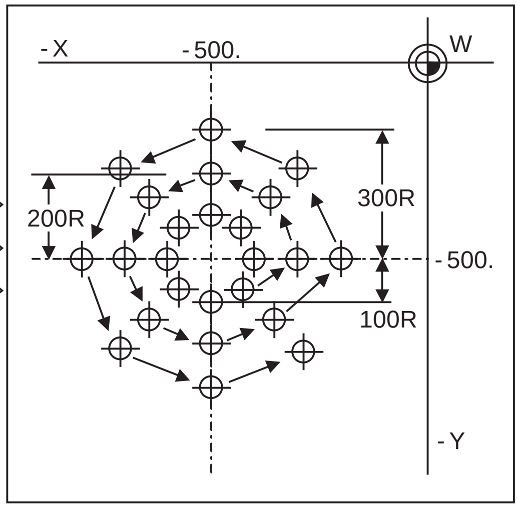
```
#101=0;
#102=#4003;
#103=#5001;        (Note 1)
#104=#5002;
#111=#1;

WHILE〔#101LT#2〕DO1;

#120=#24+#18*COS〔#111〕;
#121=#25+#18*SIN 〔#111〕;  (Note 1)

#122=#120 #123=#121;
IF〔#102EQ90〕GOTO100;

#122=#120-#103;
#123=#121-#104;  (Note 1)

#103=#120;
#104=#121;  (Note 1)

N100 X#122 Y#123;
#101=#101+1;              (Note 1)
#111=#1+360.*#101/#2;

END1;
M99;
```

(Note 1) The processing time can be shortened by programming in one block.



```
G28 X0 Y0 Z0;
T1 M06;
G90 G43 Z100.H01;
G54 G00 X0 Y0;
G81 Z-100.R3.F100 L0 M03;
G65 P9920 X-500.Y-500.A0 B8 R100.;
G65 P9920 X-500.Y-500.A0 B8 R200.;
G65 P9920 X-500.Y-500.A0 B8 R300.;
        ≀
```

To subprogram
To subprogram
To subprogram

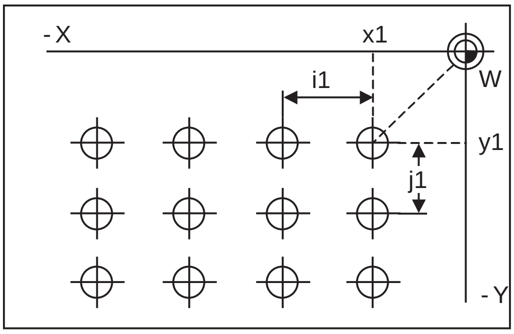


(Example 3) Grid

After defining the hole data with the fixed cycle (G72 to G89), macro call is commanded as a hole position command.

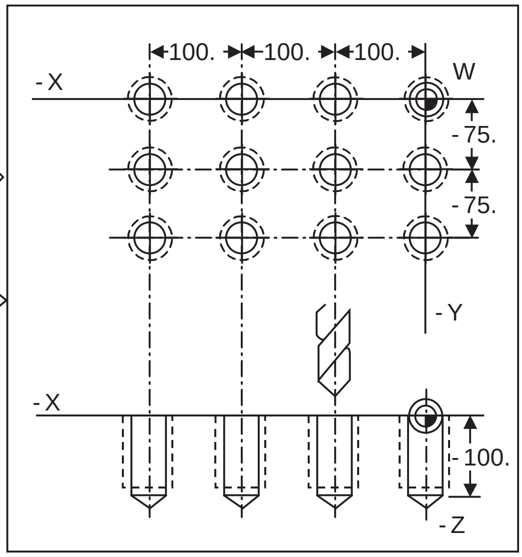
```
G81 Zz1 Rr1 Ff1;
G65 Pp1 Xx1 Yy1 Ii1 Jj1 Aa1 Bb1;
            x1;  X axis hole position
            y1;  Y axis hole position
            i1;  X axis interval
            j1;  Y axis interval
            a1;  No. of holes in X direction
            b1;  No. of holes in Y direction
```

Subprogram is on next page




```
G28 X0 Y0 Z0;
T1 M06;
G90 G43 Z100.H01;
G54 G00 X0 Y0;
G81 Z-100.R3.F100 L0 M03;
G65 P9930 X0 Y0 I-100. J-75. A5B3;
        ≀
G84 Z-90. R3. F250 M03;
G65 P9930 X0 I-100. J-75. A5B3;
        ≀
```

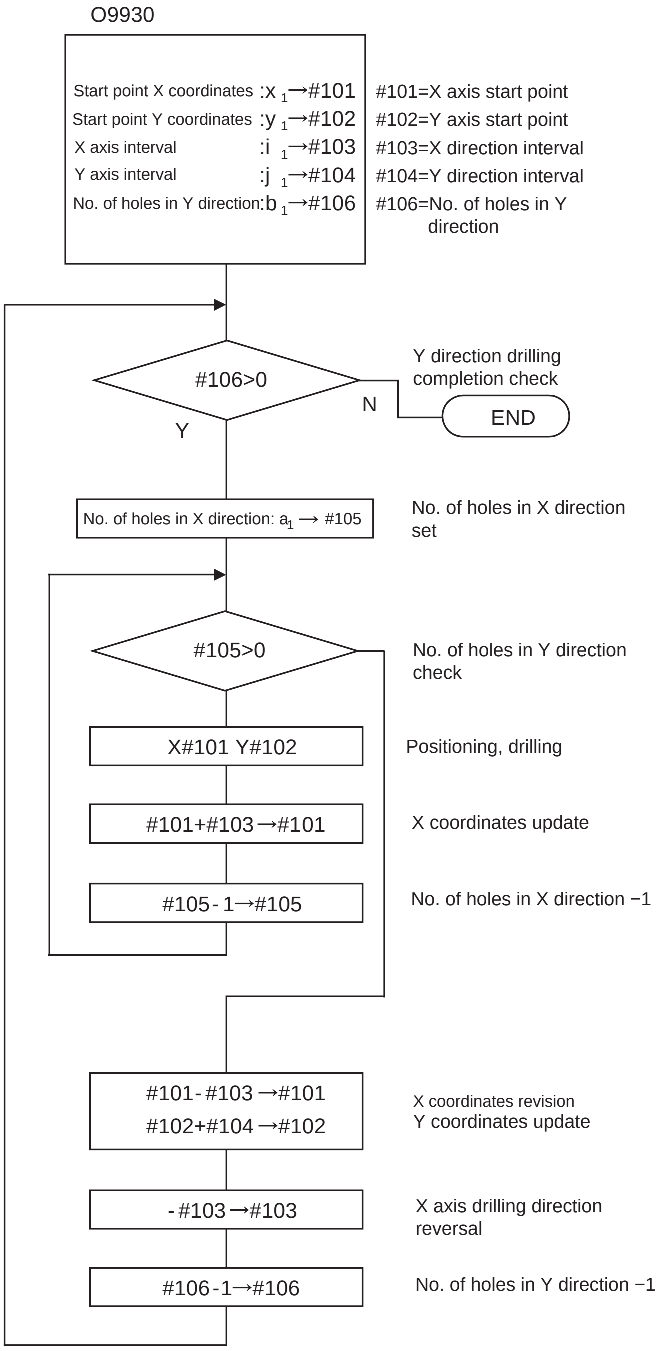
To subprogram
To subprogram



O9930 (Subprogram)

```
      #101=#24;
      #102=#25;                     (Note 1)
      #103=#4;
      #104=#5;
      #106=#2;

      WHILE〔#106GT0〕DO1;

      #105=#1;

      WHILE〔#105GT0〕DO2;

G90 X#101 Y#102;

      #101=#101+#103;               (Note 1)
      #105=#105-1;

      END2;

      #101=#101-#103;
      #102=#102+#104;               (Note 1)
      #103=-#103;
      #106=#106-1;

      END1;

      M99;
```

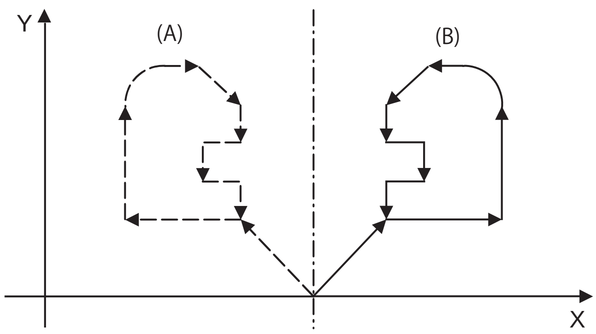
(Note 1) The processing time can be shortened by programming in one block.



13.6 G command mirror image ; G50.1,G51.1

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), a symmetrical shape (B) can be machined on the right side by applying mirror image and executing the program.



Mirror axis

Command format

G51.1 Xx1 Yy1 Zz1 ... Mirror image ON

x1, y1, z1	Mirror image center coordinates (Mirror image will be applied regarding this position as a center)

G50.1 Xx2 Yy2 Zz2 ... Mirror image OFF

x2, y2, z2	Mirror image cancel axis (The values of x2, y2, z2 will be ignored.)

Detailed description

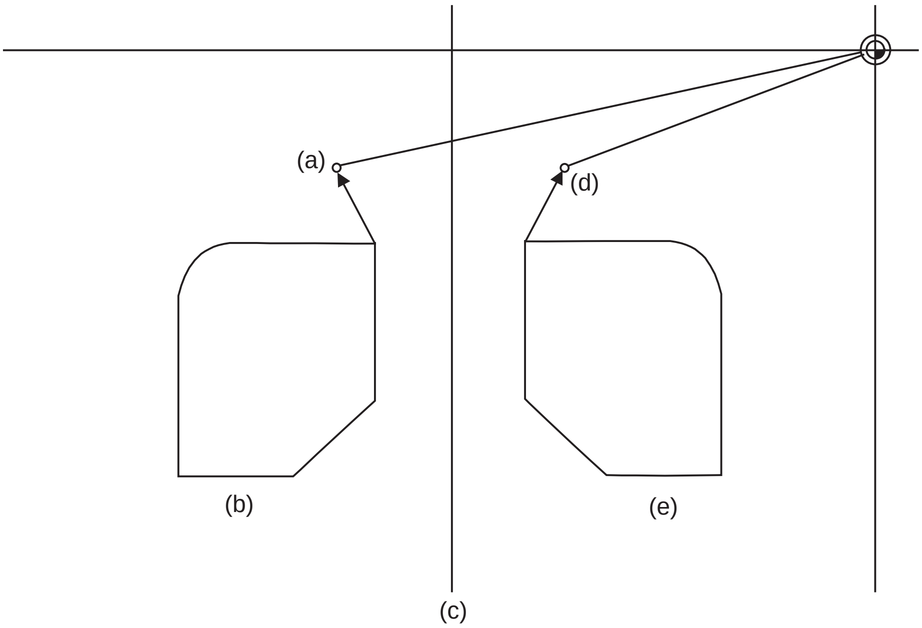
(1) At G51.1, command the mirror image axis and the coordinate to be a center of mirror image with the absolute command or incremental command.

(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 of 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 point return during mirror image
If the reference point 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 point after the intermediate point.



(a) Intermediate point when mirror is applied (b) Path on which mirror is applied (c) Mirror center
(d) Intermediate point (e) Programmed path

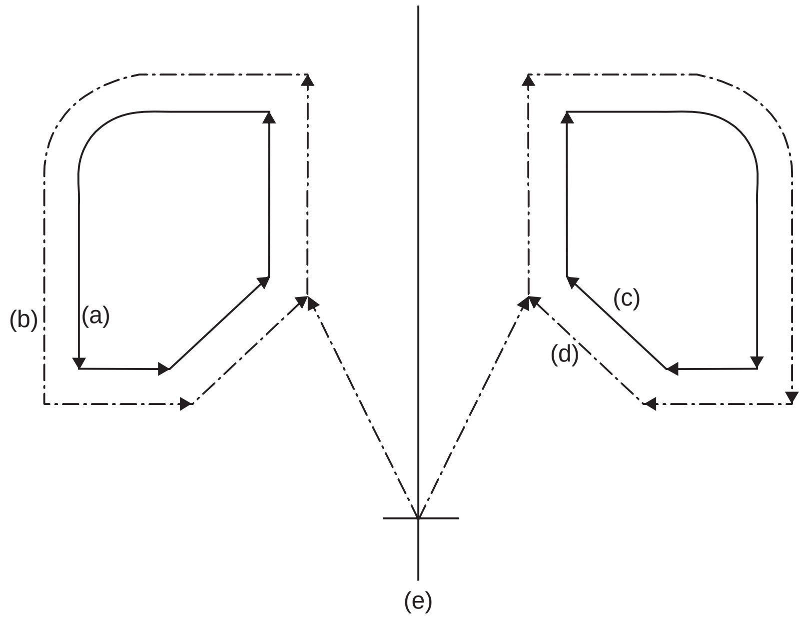
(6) Return from zero point during mirror image
If the return command (G29) from the zero point is commanded during the mirror image, the mirror will be applied on the intermediate point.

(7) The mirror image will not be applied on the G53 command.

Relation 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.

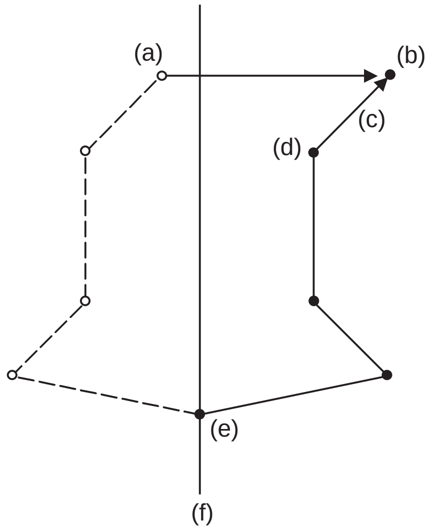


(a) Programmed path
(b) When only radius compensation is applied
(c) When only mirror image is applied
(d) When both mirror image and radius compensation are applied
(e) Mirror center

Precautions

CAUTION **1. Turn the mirror image ON and OFF at the mirror image center.**

If mirror image is not canceled at the mirror center, the absolute value and machine position will deviate as shown below. (This state will last until an absolute value command (positioning with G90 mode) is issued, or a reference point return with G28 or G30 is executed.) 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.



(a) Absolute value (position commanded in program) (b) Machine position
(c) When moved with the incremental command after mirror cancel (d) Mirror cancel command
(e) Mirror axis command (f) Mirror center

13.7 Corner Chamfering/Corner Rounding I

Function and purpose

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 ; G01 X_ Y_ ,C

Function and purpose

This chamfers a corner by connecting the both side of the hypothetical corner which would appear as if chamfering is not performed, by the amount commanded by ",C_".

Command format

N100 G01 X__ Y__ ,C__ ;
N200 G01 X__ Y__ ;

,C	Length up to chamfering starting point or end point from imaginary corner

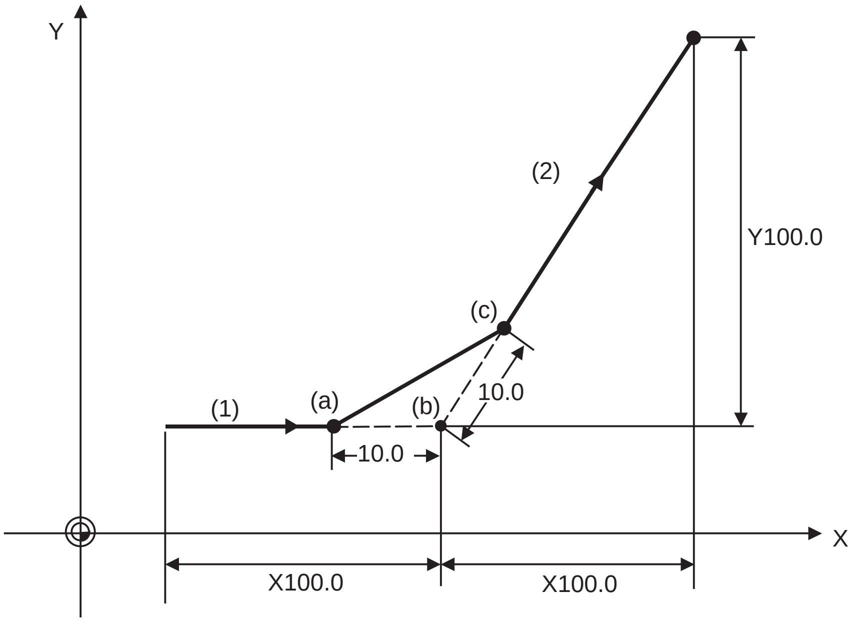
Chamfering is performed at the point where N100 and N200 intersect.

Detailed description

(1) The start point of the block following the corner chamfering is the hypothetical corner intersection point.
(2) The ",C" command will be interpreted as a C command if there is no "," (comma).
(3) When both the corner chamfer and corner rounding commands exist in the same block, the latter command is valid.
(4) Tool offset is calculated for the shape which has already been subjected to corner chamfering.
(5) Program error (P381) will occur when there is an arc command in the block following the corner chamfering block.
(6) Program error (P382) will occur when the block following the corner chamfering block does not have a linear command.
(7) Program error (P383) will occur when the movement amount in the corner chamfering block is less than the chamfering amount.
(8) Program error (P384) will occur when the movement amount in the block following the corner chamfering block is less than the chamfering amount.

Program example

(1) G91 G01 X100. ,C10.;
(2) X100. Y100.;



(a) Chamfering start point (b) Hypothetical corner intersection point (c) Chamfering end point

13.7.2 Corner Rounding ; G01 X_ Y_ ,R_

Function and purpose

The hypothetical 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 G01 X__ Y__ ;
```

,R	Circular radius of corner rounding

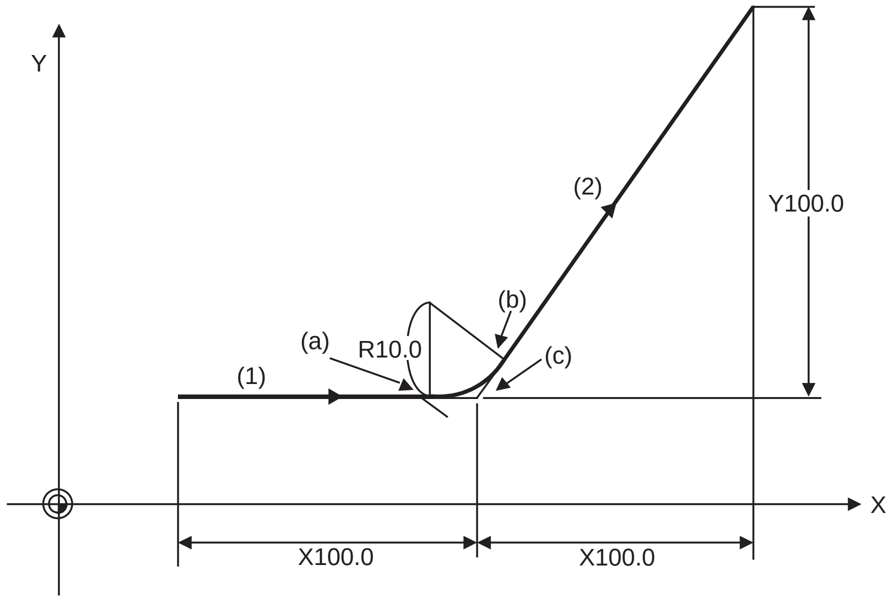
Corner rounding is performed at the point where N100 and N200 intersect.

Detailed description

(1) The start point of the block following the corner rounding is the hypothetical corner intersection point.
(2) The ",R" command will be interpreted as a R command if there is no "," (comma).
(3) When both the corner chamfer and corner rounding commands exist in the same block, the latter command is valid.
(4) Tool offset is calculated for the shape which has already been subjected to corner rounding.
(5) Program error (P381) will occur when there is an arc command in the block following the corner rounding block.
(6) Program error (P382) will occur when the block following the corner rounding block does not have a linear command.
(7) Program error (P383) will occur when the movement amount in the corner rounding block is less than the R value.
(8) Program error (P384) will occur when the movement amount in the block following the corner rounding block is less than the R value.

Program example

(1) G91 G01 X100. ,R10.;
(2) X100. Y100.;



(a) Corner rounding start point (b) Corner rounding end point (c) Hypothetical corner intersection point

13.8 Circular 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

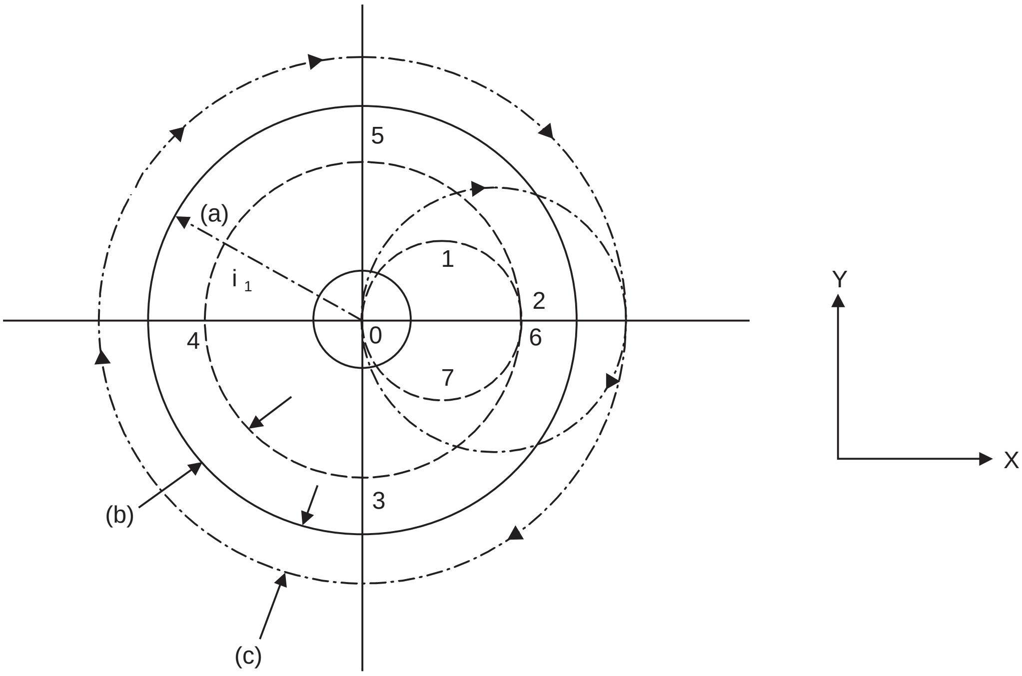
G12 I__ D__ F__ ; ... Circular cutting Clockwise (CW)

G13 I__ D__ F__ ; ... Circular cutting Counterclockwise (CCW)

I	Radius of circle (incremental value), the sign 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 sign + for the offset amount indicates reduction, and - indicates enlargement.
(2) The circle cutting is executed on the plane G17, G18 or G19 currently selected.



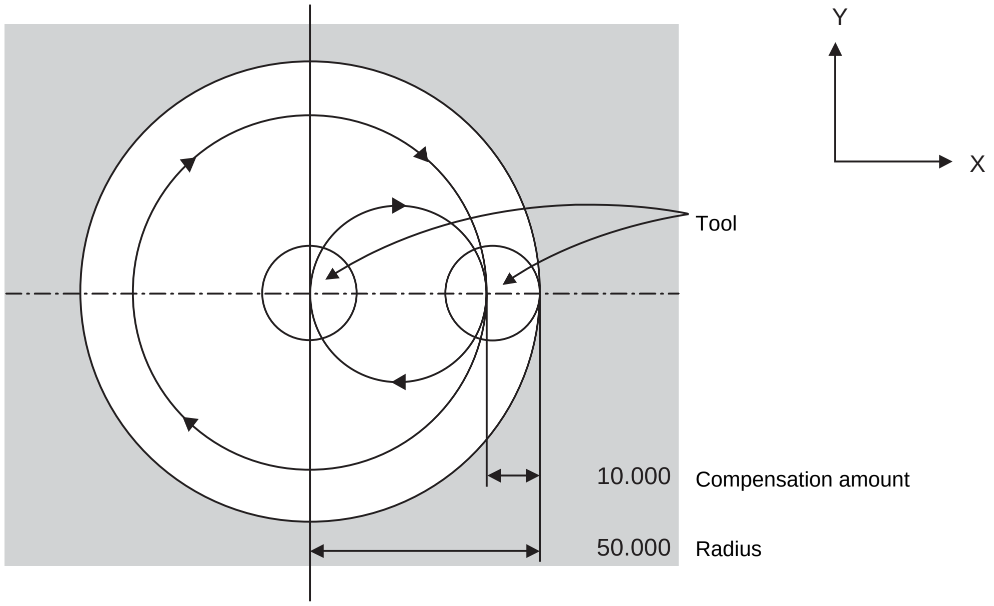
———— Compensation amount sign +   —·—·— Compensation amount sign -

(a) Circle radius   (b) d1 offset amount +   (c) d1 offset amount -

For G12 (tool center path) 0->1->2->3->4->5->6->7->0
For G13 (tool center path) 0->7->6->5->4->3->2->1->0

Program example

(Example 1) G12 I50.000 D01 F100 ;
When compensation amount is +10.000mm



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 compensated 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, the program error (P32) will occur.

13.9 Programmable Parameter Input ; G10 L70, G11

Function and purpose

The parameters set from the setting and display unit can be changed in the machining programs.

Command format

G10 L70 ;...Data setting start command **P__ S__ A__ H□__ ; .... Bit parameter** **P__ S__ A__ D__ ; .... Numerical value parameter** **P__ S__ A__ <character string> ; ...... Character string parameter** **P__ S__ A__ ,character string; ...... Character string parameter ...**

P	Parameter No.
S	Part system No.
A	Axis No.
H	Data
D	Data
<character string>	
,character string	

G11 ; ... Data setting end command

(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 the 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) Character strings must be with "," or put in angled brackets "<" and ">" when commanded.
If these brackets are not provided, the program error (P33) will occur.
Up to 31 characters can be set.

(Note 7) Command G10 L70, G11 in independent blocks. A program error (P33, P421) will occur if not commanded in independent blocks.

(Note 8) When the data with a decimal point is commanded without a decimal point, it will be handled as decimal point valid.

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".
P7501 <X> ;	Sets "X" to "#7501".
G11 ;	

13.10 Macro Interruption ; M96,M97

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.

Command format

M96 P__ H__ ; ... User macro interruption enable

M96	User macro interruption command
P	Interrupt program No.
H	Interrupt sequence No.

M97 ; ... User macro interruption disable

M97	User macro interruption end command

Detailed description

The user macro interrupt function is enabled and disabled by the M96 and M97 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 interruption enable period from when M96 is issued to when M97 is issued or the NC is reset, a user macro interruption is caused to execute the program specified by P__ 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 M97 command is issued or the system is reset.
M96 and M97 are processed internally as user macro interrupt control M codes.

Interrupt enable conditions

A user macro interruption is enabled only during execution of a program.
The requirements for the user macro interrupt are as follows:

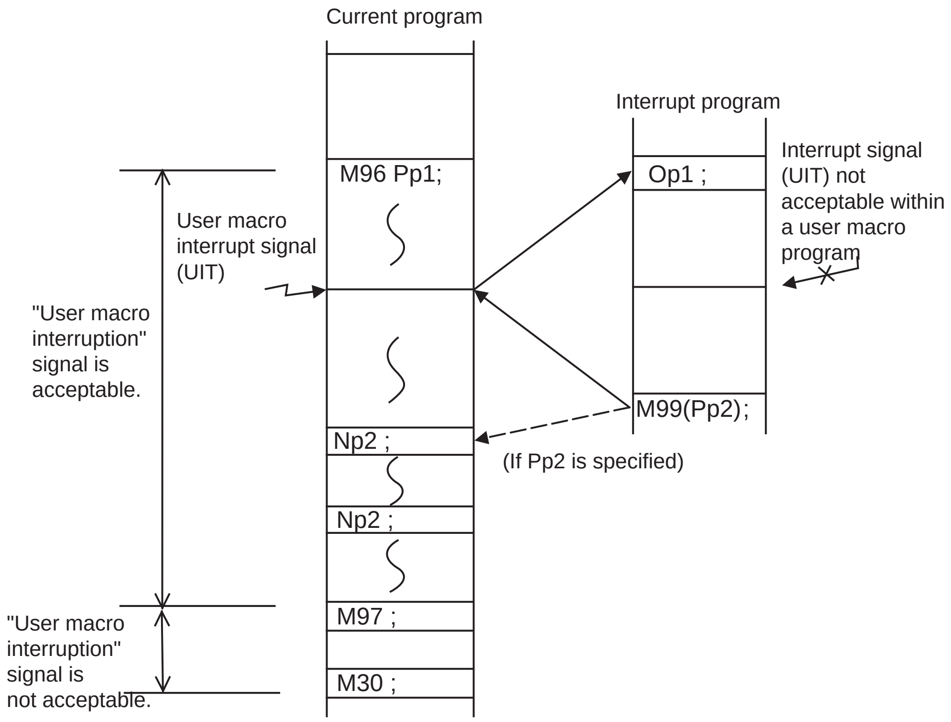
(1) A memory operation mode or MDI has been selected.
(2) The system is running in automatic mode.
(3) No user macro interruption is being processed.

(Note 1) A macro interruption 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 M96Pp1 ; command is issued by the current program, interrupt program Op1 is executed. When an M99; command is issued by the interrupt program, control returns to the main program.

(2) If M99Pp2 ; is specified, the blocks from the one next to the interrupted block to the last one are searched for the block with sequence number Np2 ;. Control thus returns to the block with sequence number Np2 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]

(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.

(2) 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.

(3) If the interrupted 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.

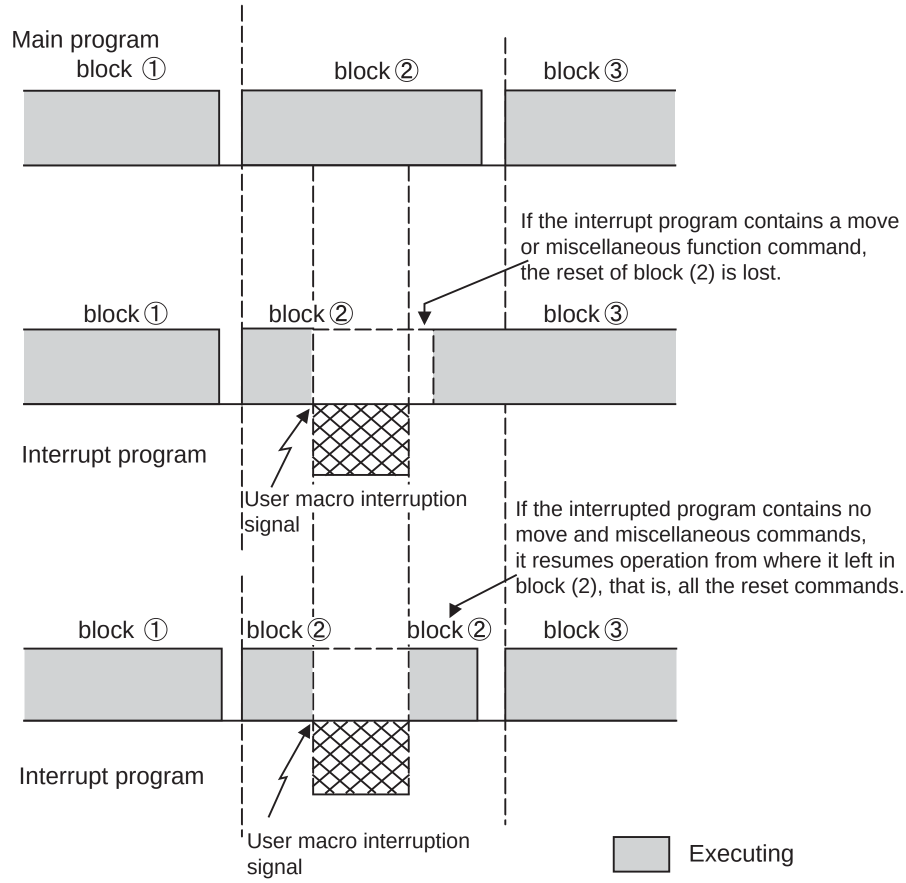
[Type 2]

(1) When an interrupt signal (UIT) is input, the program completes the commands in the current block, then transfers control to the interrupt program.

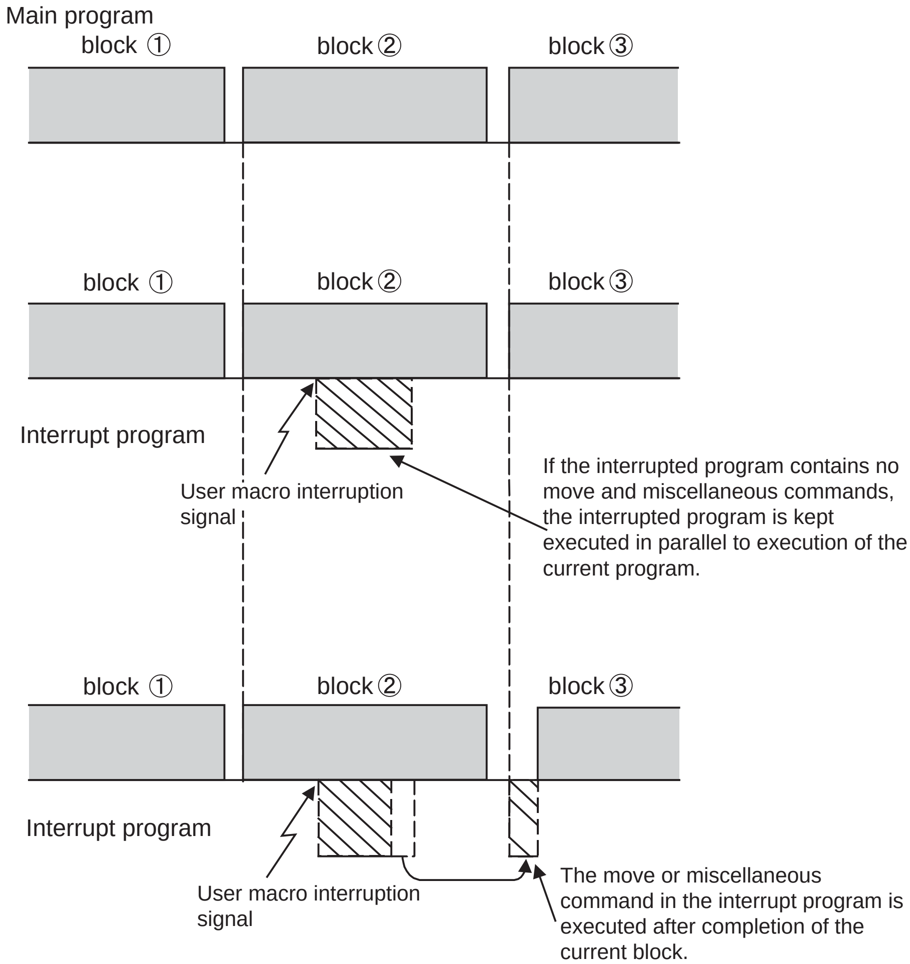
(2) If the interrupt program contains no move and miscellaneous function (MSTB) commands, the interrupt program is executed without interrupting execution of the current block.

However, if the interrupt program has not ended even after the execution of the original block is completed, the system may stop machining temporarily.

[Type 1]


Main program
block ①
block ②
block ③
If the interrupt program contains a move or miscellaneous function command, the reset of block (2) is lost.
block ①
block ②
block ③
Interrupt program
User macro interruption signal
If the interrupted program contains no move and miscellaneous commands, it resumes operation from where it left in block (2), that is, all the reset commands.
block ①
block ②
block ②
block ③
Interrupt program
User macro interruption signal
Executing

[Type 2]


Main program
block ①
block ②
block ③
block ①
block ②
block ③
Interrupt program
User macro interruption signal
If the interrupted program contains no move and miscellaneous commands, the interrupted program is kept executed in parallel to execution of the current program.
block ①
block ②
block ③
Interrupt program
User macro interruption signal
The move or miscellaneous command in the interrupt program is executed after completion of the current block.

Calling method

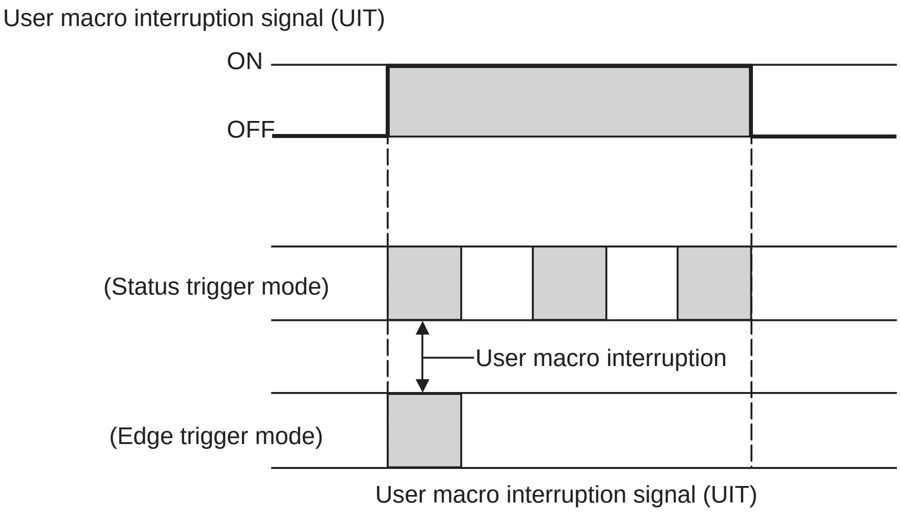
User macro interruption is classified into the following two types depending on the way an interrupt program is called. These two types of interrupt are selected by parameter "#1229 set01/bit0".
Both types of interrupt are added to the calculation of the nest level. The subprograms and user macros called in the interrupt program are also added to the calculation of the nest level.

Subprogram type interrupt	The user macro interruption program is called as a subprogram. As with calling by M98, the local variable level remains unchanged before and after an interrupt.
Macro type interruption	The user macro interpretation 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 interruption signal (UIT)

A user macro interruption signal (UIT) is accepted in the following two modes: These two modes are selected by a parameter "#1112 S_TRG".

Status trigger mode	The user macro interruption 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 M96, 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.



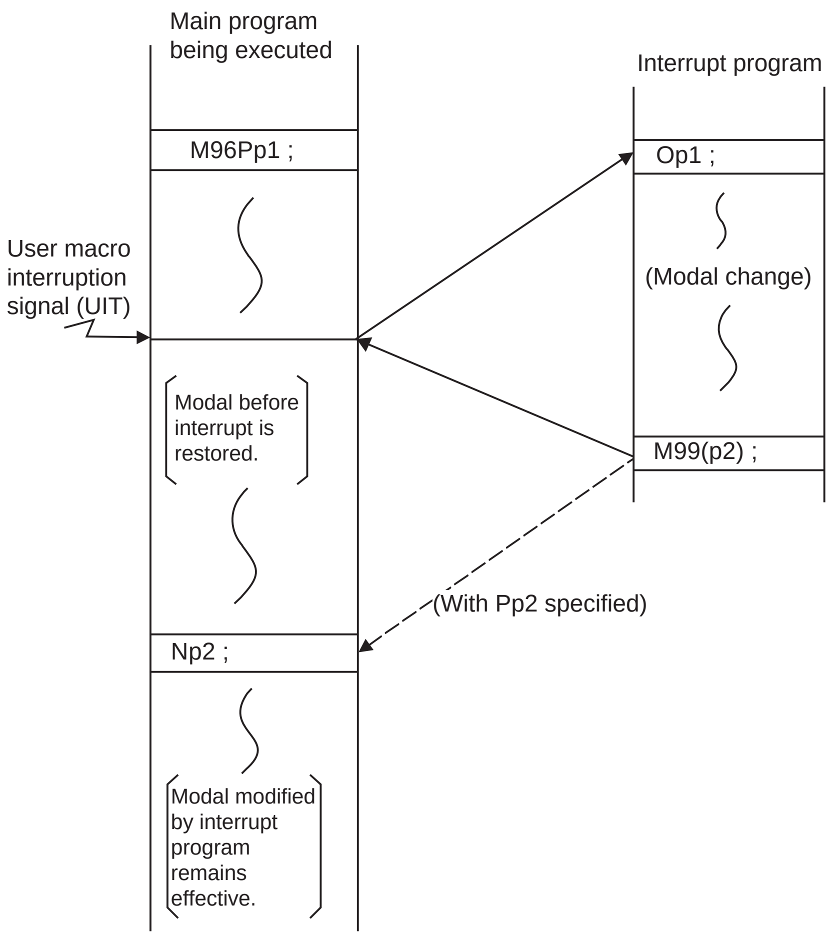
Returning from user macro interruption

M99 (P__) ;
An M99 command is issued in the interrupt program to return to the main program.
Address P 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 No. If it is not found, all the blocks before the interrupted one are then searched. Control thus returns to the block with sequence No. that is found first in the above search.
(This is equivalent to M99P__ used after M98 calling.)

Modal information affected by user macro interruption

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 M99;	The change of modal information by the interrupt program is invalidated and the original modal information is 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 M99P__ ;	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 M99P__; from a program called by M98, etc.



Modal information affected by user macro interruption

Modal information variables (#4401 to #4520)

Modal information when control passes to the user macro interruption program can be known by reading system variables #4401 to #4520.
The unit specified with a command applies.

System variable	Modal information	
#4401 : #4421	G code (group01) : G code (group21)	Some groups are not used.
#4507	D code	
#4509	F code	
#4511	H code	
#4513	M code	
#4514	Sequence No.	
#4515	Program No.	
#4519	S code	
#4520	T code	

The above system variables are available only in the user macro interrupt program.
If they are used in other programs, program error (P241) will occur.

M code for control of user macro interruption

The user macro interruption is controlled by M96 and M97. However, these commands may have been used for other operation. To be prepared for such case, these command functions can be assigned to other M codes.
(This invalidates program compatibility.)

User macro interrupt control with alternate M codes is possible by setting the alternate M code in parameters "#1110 M96_M" and "#1111 M97_M" and by validating the setting by selecting parameter "#1109 subs_M".
(M codes 03 to 97 except 30 are available for this purpose.)

If the parameter "#1109 subs_M" used to enable the alternate M codes is not selected, the M96 and M97 codes remain effective for user macro interrupt control.
In either case, the M codes for user macro interrupt control are processed internally and not output to the outside.

Parameters

Refer to the Instruction Manual for details on the setting methods.

(1) Subprogram call validity "#1229 set 01/bit 0"
1: Subprogram type user macro interruption
0: Macro type user macro interruption

(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 interruption control "#1109 subs_M"
1: Valid
0: Invalid

(5) Alternate M codes for user macro interruption
Interrupt enable M code (equivalent to M96) "#1110 M96_M"
Interrupt disable M code (equivalent to M97) "#1111 M97_M"
M codes 03 to 97 except 30 are available.

Precautions

(1) If the user macro interruption 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 No. (M99P__;) must be specified with a command to return from the user macro interrupt program. If no sequence No. is specified, control cannot return to the main program normally.

13.11 Tool Change Position Return ; G30.1 - G30.6

Function and purpose

By specifying the tool change position in a parameter "#8206 tool change" and also specifying a tool change 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 change position and the order in which the axes begin to return can be changed by commands.

Command format

G30.n ; ... Tool change position return

n = 1 to 6: Specify the axes that return to the tool change position and the order in which they return.

Detailed description

Commands and return order are given below.

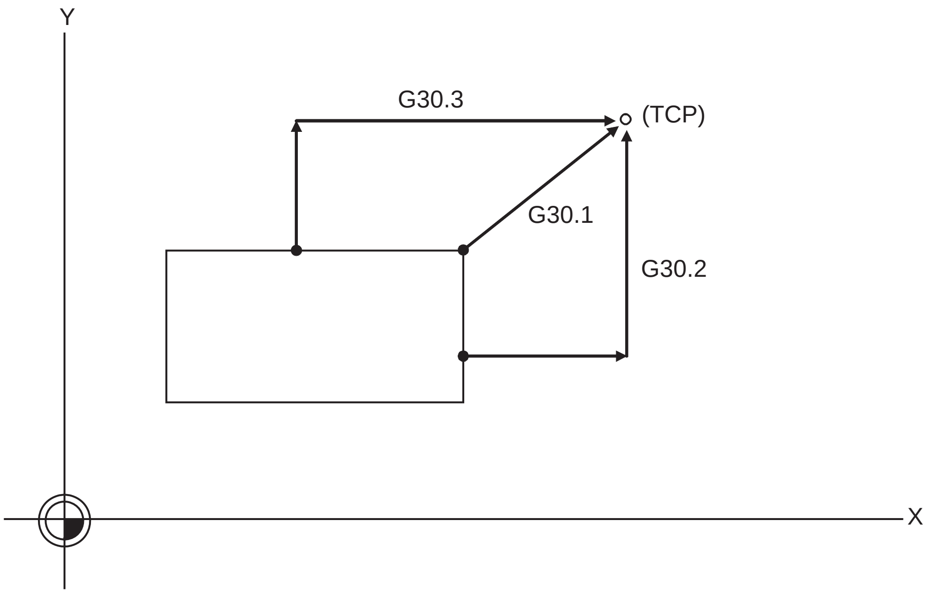
Command	Return order
G30.1	Z axis -> X axis - Y axis (-> additional axis)
G30.2	Z axis -> X axis -> Y axis (-> additional axis)
G30.3	Z axis -> Y axis -> X axis (-> additional axis)
G30.4	X axis -> Y axis - Z axis (-> additional axis)
G30.5	Y axis -> X axis - Z axis (-> additional axis)
G30.6	X axis - Y axis - Z axis (-> additional axis)

(Note 1) An arrow (->) indicates the order of axes that begin to return. A hyphen (-) indicates that the axes begin to return simultaneously. (Example: "Z axis -> X axis - Y axis" indicates that the Z axis returns to the tool change position, then the X axis and Y axis do at the same time.)

(1) The tool change position return on/off for the additional axis can be set with parameter "#1092 Tchg_A" for the additional axis.
For the order for returning to the tool change position, the axes return after the standard axis completes the return to the tool change position (refer to above table).
The additional axis alone cannot return to the tool change position.

Operation example

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.)



TCP : Tool change position

(1) 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.)

(2) 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.)

(3) 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.)

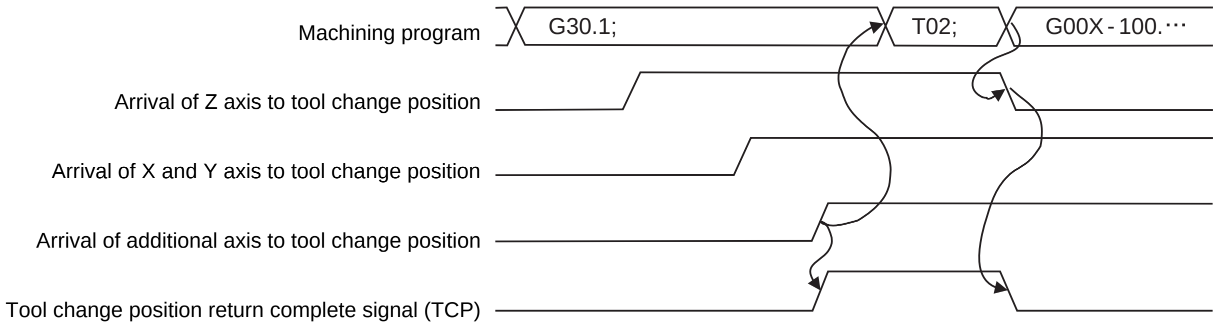
(4) G30.4 command : The X axis returns to the tool changing position, then the Y axis and Z axis simultaneously do the same thing. (If tool changing position return is on for an added axis, the added axis also return to the tool changing position after the X, Y and X axes reach the tool changing position.)

(5) 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.)

(6) G30.6command :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.)

(7) 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.

[TCP signal output timing chart] (G30.1 command with tool change position return for additional axes set ON)



(8) 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.

(9) 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 change position. To make the next axis tool change position return, therefore, a cycle start needs to be specified.

13.12 High-accuracy control ; G61.1

Function and purpose

The following troubles occurred when using normal control:

(1) Corner rounding occurred at linear and linear-connected corners because the next command movement started before the previous command finished. (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.
(Fig. 2)

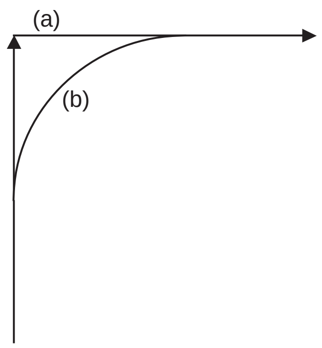


Fig. 1 Rounding at linear corners

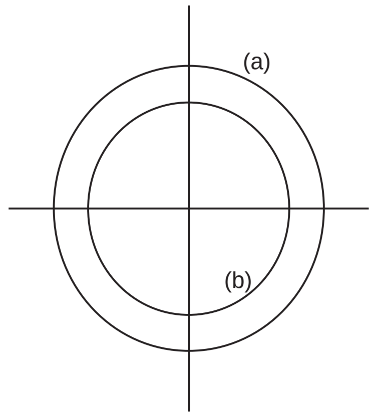


Fig. 2 Radius reduction error in circle commands

(a) Commanded path (b) Actual path

This function controls the operation so the lag is eliminated in control systems and servo systems. With this function, machining accuracy can be improved, especially during high-speed machining, and machining time can be reduced. The high-accuracy control function is configured of the following functions.

(1) Pre-interpolation acceleration/deceleration (linear acceleration/deceleration)
(2) Optimum speed control
(3) Vector accuracy interpolation
(4) Feed forward

Command format

G61.1 F__ ; ... High-accuracy control mode ON

F	Feedrate command

The high-accuracy control mode is validated from the block containing the G61.1 command. This function is valid only for the first part system.
The "G61.1" high-accuracy control mode is canceled with one of the functions of G code group 13.
- G61 (Exact stop check mode)
- G62 (Automatic corner override)
- G63 (Tapping mode)
- G64 (Cutting mode)

Detailed description

(1) The feedrate command F is clamped by the rapid traverse rate or maximum cutting feedrate set with the parameters.
(2) 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).

<table>
<tr><th colspan="2">Parameter</th><th>Default state</th><th colspan="3">Resetting</th></tr>
<tr><th>Reset initial (#1151)</th><th>Initial high accuracy (#1148)</th><th>Power ON</th><th>Reset 1</th><th>Reset 2</th><th>Reset & rewind</th></tr>
<tr><td>OFF</td><td rowspan="2">OFF</td><td rowspan="2">OFF</td><td>Hold</td><td colspan="2" rowspan="2">OFF</td></tr>
<tr><td>ON</td><td>OFF</td></tr>
<tr><td>OFF</td><td rowspan="2">ON</td><td rowspan="2">Hold</td><td rowspan="2">ON</td><td rowspan="2">ON</td><td rowspan="2">Hold</td></tr>
<tr><td>ON</td></tr>
</table>

<table>
<tr><th colspan="2">Parameter</th><th>Emergency stop</th><th>Emergency stop cancel</th></tr>
<tr><th>Reset initial (#1151)</th><th>Initial high accuracy (#1148)</th><th>Emergency stop switch,
External emergency stop</th><th>Emergency stop switch,
External emergency stop</th></tr>
<tr><td>OFF</td><td rowspan="2">OFF</td><td rowspan="2">Hold</td><td>Hold</td></tr>
<tr><td>ON</td><td>OFF</td></tr>
<tr><td>OFF</td><td rowspan="2">ON</td><td rowspan="2">Hold</td><td>Hold</td></tr>
<tr><td>ON</td><td>ON</td></tr>
</table>

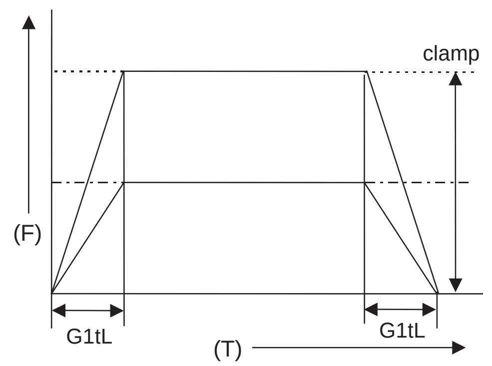
<table>
<tr><th colspan="2">Parameter</th><th>Block interruption</th><th>Block stop</th><th>NC alarm</th><th>OT</th></tr>
<tr><th>Reset initial (#1151)</th><th>Initial high accuracy (#1148)</th><th>Mode changeover (automatic/manual), or Feed hold</th><th>Single block</th><th>Servo alarm</th><th>H/W OT</th></tr>
<tr><td>OFF</td><td rowspan="2">OFF</td><td colspan="4" rowspan="4">Hold</td></tr>
<tr><td>ON</td></tr>
<tr><td>OFF</td><td rowspan="2">ON</td></tr>
<tr><td>ON</td></tr>
</table>

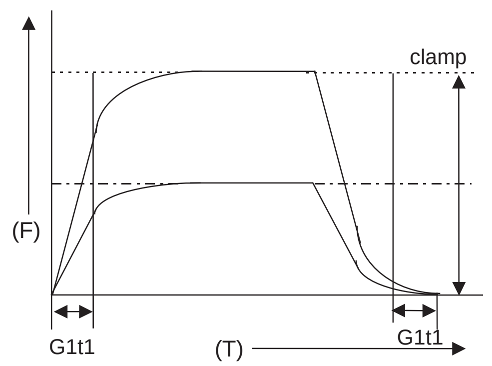
H (hold): Modal hold
ON: Switches to high-accuracy mode
As for G61.1, the mode is switched to the high-accuracy mode, even if the other modes (G61 to G64) are valid.
OFF: The status of the high-accuracy mode is OFF.

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.

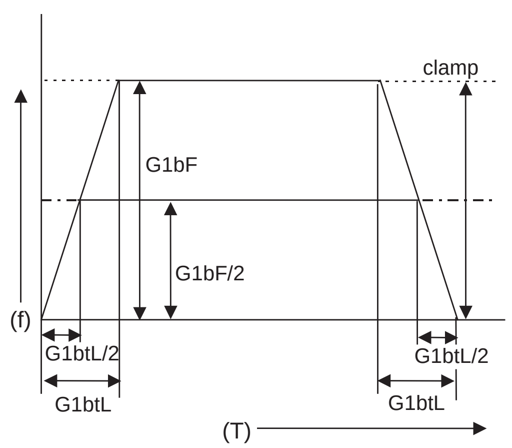
(1) Basic patterns of acceleration/deceleration control in linear interpolation commands

<table>
<tr><th></th><th>Acceleration/deceleration waveform pattern</th><th></th></tr>
<tr><td>Normal mode</td><td>




(F) Speed of each axis (T) Time</td><td>(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.

(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.

#2002 clamp : G01 clamp speed
#2007 G1tL : Linear type acceleration/deceleration time constant
#2008 G1t1 : Exponential type acceleration/deceleration time constant</td></tr>
<tr><td>High-accuracy control mode</td><td>


(f) Combined speed (T) Time</td><td>(a) Because of the constant inclination type linear acceleration/deceleration, the acceleration/deceleration time is reduced as the command speed becomes slower.
(b) Only one acceleration/deceleration time constant (common for each axis) exists in a system.

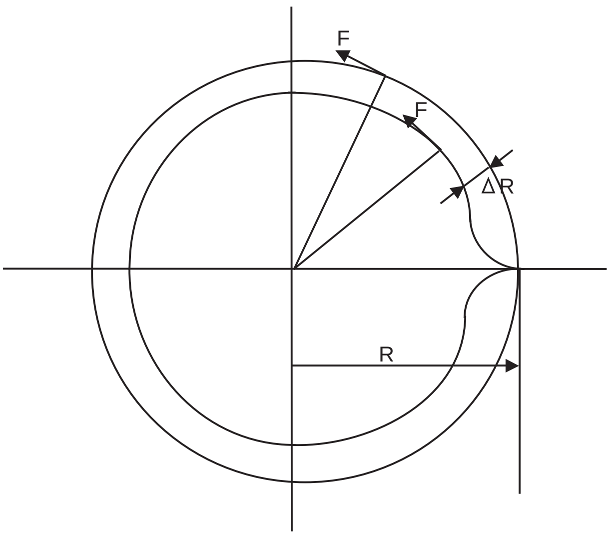
#2002 clamp : G01 clamp speed
#1206 G1bF : Target speed
#1207 G1btL : Acceleration/deceleration time to target speed
(Note) G1bF and G1btL are values for specifying the inclination of the acceleration/deceleration time. The actual cutting feed maximum speed is clamped by the "#2002 clamp" value.</td></tr>
</table>

(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.



R : Commanded radius (mm) ΔR : Radius error (mm) F : Cutting feedrate (mm/min)

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 $\Delta R = \frac{1}{2R}\left(\frac{1}{12}Ts^2 + Tp^2\right)\left(\frac{F}{60}\right)^2$ Exponential function acceleration/deceleration $\Delta R = \frac{1}{2R}\left(Ts^2 + Tp^2\right)\left(\frac{F}{60}\right)^2$	Linear acceleration/deceleration $\Delta R = \frac{1}{2R}\left\{Tp^2\left(1 - Kf^2\right)\right\}\left(\frac{F}{60}\right)^2$ (a) Because the item Ts can be ignored by using the pre-interpolation acceleration/deceleration control method, the radius reduction error amount can be reduced. (b) Item Tp can be negated by making Kf = 1.

Ts : Acceleration/deceleration time constant in the CNC (s)

Tp : Servo system position loop time constant (s)

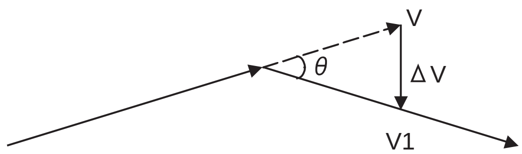
Kf : Feed forward coefficient

Optimum speed control

(1) Optimum corner deceleration

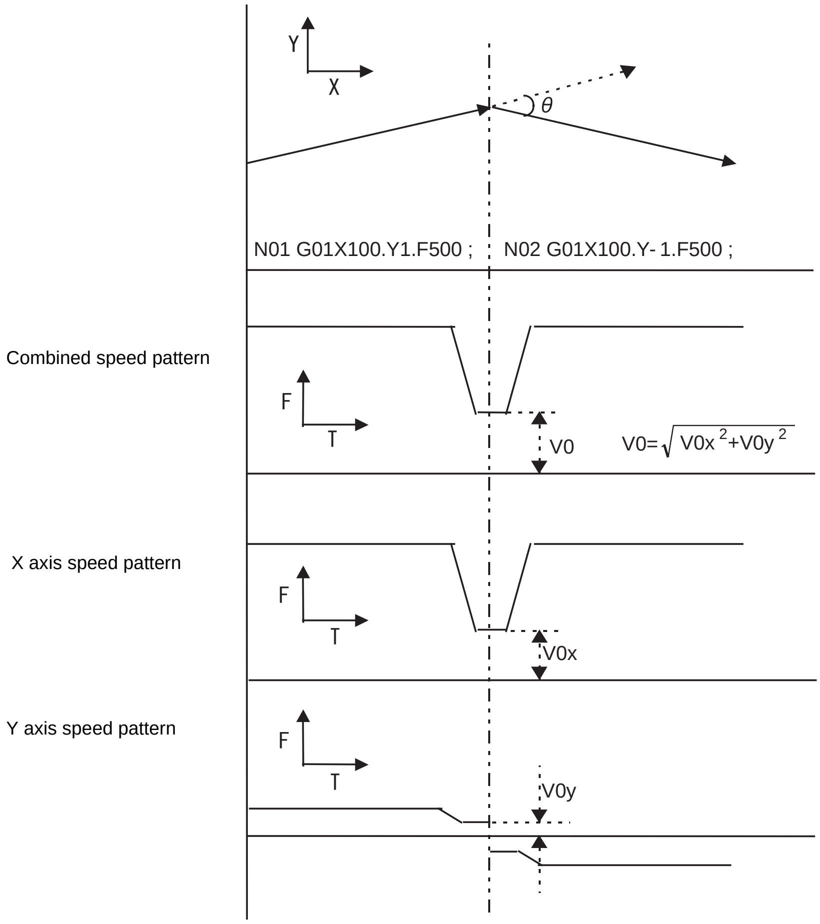
By calculating the angle of the seam between blocks, and carrying out acceleration/ deceleration control in which the corner is passed at the optimum speed, highly accurate edge machining can be realized.
When entering in a corner, optimum speed for the corner (optimum corner speed) is calculated from the angle with the next block. The machine decelerates to the speed in advance, and then accelerates back to the command speed after passing the corner.
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 ΔV occurs due to the change in the direction of progress after passing the corner at the speed V.



V : Speed before entering the corner   ΔV : Speed change at the corner   V1 : Speed after passing the corner

The corner speed V is controlled so that ΔV 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 optimum corner speed is represented by V0. V0 is obtained from the pre-interpolation acceleration/deceleration tolerable value (ΔV') and the corner angle (outside angle) θ.

ΔV' = G1bF/G1btL

To further reduce the corner speed V0 (to further improve the edge accuracy), the V0 value can be reduced in the machining parameter "#8019 R COMPEN".

V0' = V0*(100 - Ks) / 100 (Ks : R COMPEN)

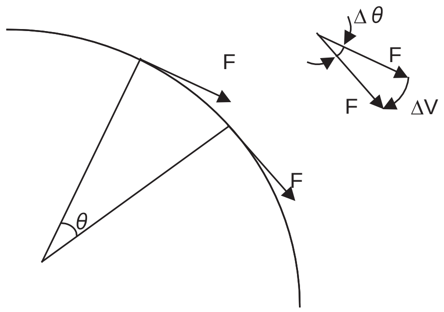
(Note 1) In this case, the cycle time may increase due to the increase in the time required for acceleration/deceleration.

(F) Speed
(T) Time

(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 enough in relation 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.



F : Commanded speed (mm/min)
R : Commanded arc radius (mm)
Δθ : Angle change per interpolation unit
ΔV : Speed change per interpolation unit

The tool is fed with the arc clamp speed F so that ΔV does not exceed the tolerable acceleration/deceleration speed before interpolation ΔV.

$$F \leqq \sqrt{R*\Delta V*60*1000}\ \text{(mm/min)}$$

$$\Delta V = \frac{\text{G1bF(mm/min)}}{\text{G1btL(ms)}}$$

When the above F' expression is substituted in the expression for the maximum logical arc radius reduction error amount ΔR, explained in the section "Pre-interpolation acceleration/deceleration", the commanded radius R is eliminated, and ΔR does not rely on R.

$$\Delta R \leqq \frac{1}{2R}\left\{Tp^2\left(1 - Kf^2\right)\right\}\left(\frac{F}{60}\right)^2$$

$$\leqq \frac{1}{2}\left\{Tp^2\left(1 - Kf^2\right)\right\}\left(\frac{\Delta V' * 1000}{60}\right)$$

ΔR : 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 ΔR by the set percentage.

$$\Delta R' = \frac{\Delta R * (100 - Ks)}{100}\ \text{(mm)}$$

ΔR' : Maximum arc radius reduction error amount
Ks : R COMPEN (%)

After setting the "R COMPEN", the above ΔR' will appear on the parameter screen.

#8019 R COMPEN (0.078) 50

Accuracy coefficient setting value

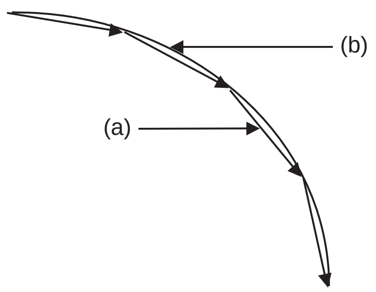
Δ R'

(Note 1) When the "R COMPEN" is set, the arc clamp speed will drop, so in a machining program with many arc commands, the machining time will take longer.

(Note 2) 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.

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.

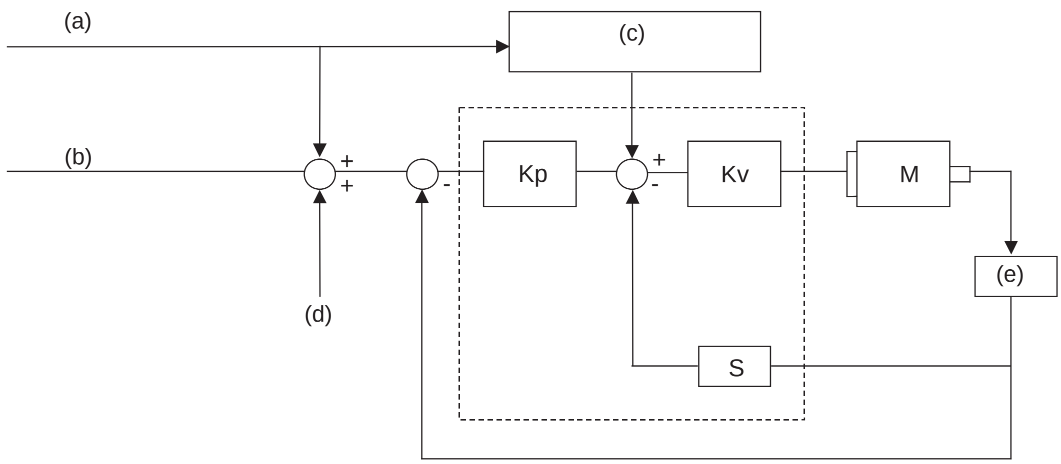


(a) Commanded path (b) Vector accuracy interpolation

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



Kp : Position loop gain Kv : Speed loop gain M : Motor S : Segment

(a) Command during pre-interpolation acceleration/deceleration
(b) Command during post-interpolation acceleration/deceleration
(c) Feed forward control
(d) Machine error compensation amount
(e) Detector

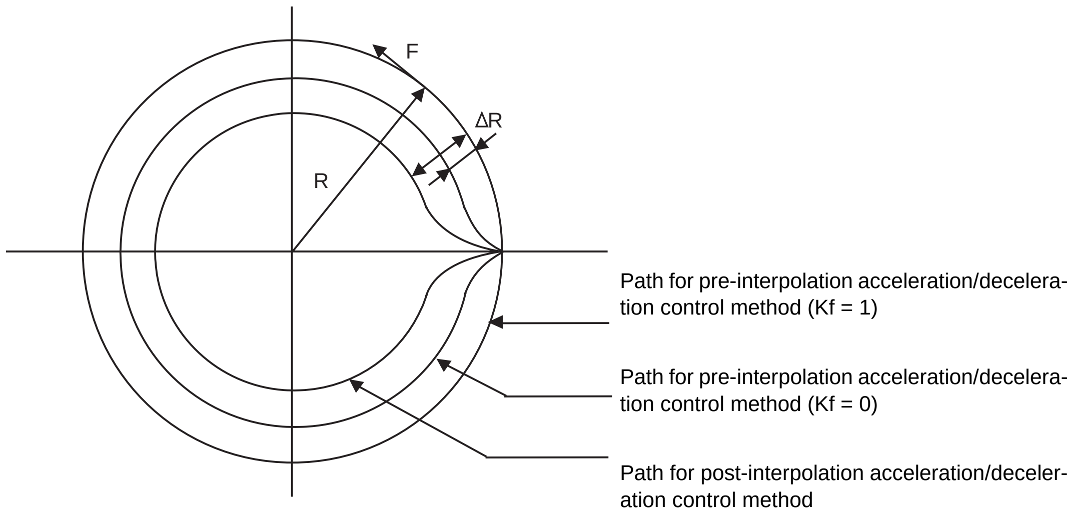
(2) Reduction of arc radius reduction error amount using feed forward control

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 ΔR in the high-accuracy control mode is obtained with the following expression.

<table>
<tr><th>Feed forward control</th><th>SHG control + Feed forward control</th></tr>
<tr><td colspan="2">$$\Delta R = \frac{1}{2R}\left\{Tp^2\left(1 - Kf^2\right)\right\}\left(\frac{F}{60}\right)^2$$
R : Arc radius (mm)
F : Cutting feedrate (mm/min)
Tp : Position loop time constant (s)
Kf : Feed forward coefficient (fwd_g/100)</td></tr>
<tr><td colspan="2">By setting Kf to the following value, the delay elements caused by the position loop in the servo system can be eliminated, and, logically, ΔR can be set to 0.</td></tr>
<tr><td>Kf = 1 (Feed forward gain 100%)</td><td>The equivalent feed forward gain to set Kf to 1 can be obtained with the following expression.
$$100\sqrt{1 - \left\{1 - \left(\frac{fwd\_g}{50}\right)^2\right\}\left(\frac{\text{PGN1 for conventional control}}{2 * \text{PGN1 for SHG control}}\right)^2}$$</td></tr>
</table>

The feed forward gain can be set independently for G00 and G01.

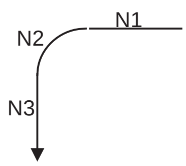


(Note) If the machine vibrates when Kf is set to 1, Kf must be lowered or the servo system must be adjusted.

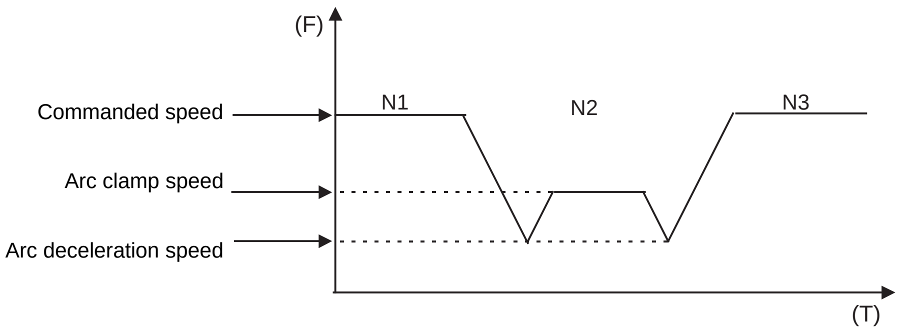
Arc entrance/exit speed control

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 cireft". The deceleration speed is designated with the base specification parameter "#1209 cirdcc".

(Example 1)When not using corner deceleration

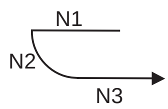
<Program>	<Operation>
G61.1 ; . . N1 G01 X-10. F3000 ; N2 G02 X-5. Y-5. J-2.5 ; N3 G01 Y-10. ; . .	

<Speed pattern>

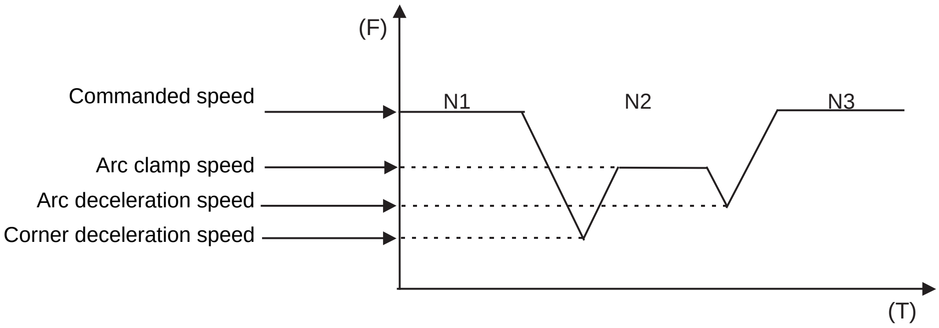


(F) Speed
(T) Time

(Example 2)When using corner deceleration

<Program>	<Operation>
G61.1 ; . . N1 G01 X-10. F3000 ; N2 G02 X5. Y-5. I2.5 ; N3 G01 X10. ; . .	

<Speed pattern>

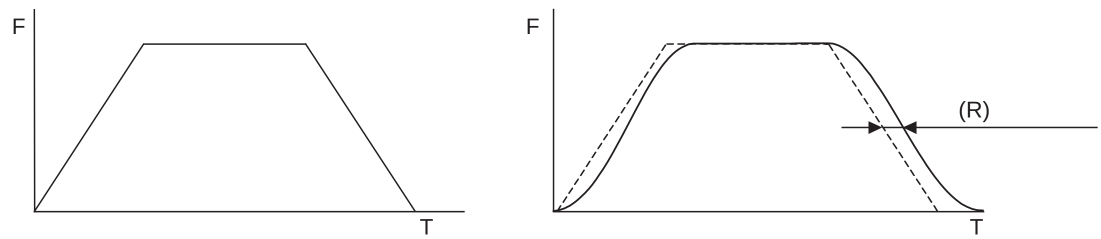


(F) Speed
(T) Time

S-pattern filter control

This control interpolates the fluctuations in the segments further smoothly, when they are 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.
Set the S-pattern filter with "#1131 Fldoc".
The S-pattern filter can be set as "3.5/7.1/14.28.4/56.8[ms]".

<When filter is invalid> <When filter is valid>



(R) Parameter setting value

Command to drive unit

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) This function may not be usable, depending on the model.

13.13 Coordinate rotation by program ; G68/G69

Function and purpose

When machining a complicated shape located in a rotated position in respect to the coordinate system, this function enables to machine the rotated shape with the program for the shape before rotation on the local coordinate system and with the rotation angle designated by the program coordinate rotation command.

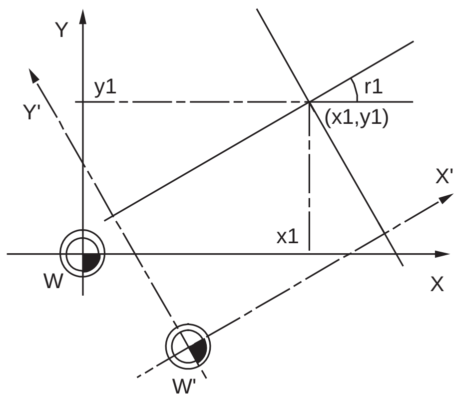
Command format

G68 X__ Y__ R__; ... Coordinate rotation ON

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 +.

G69 ; ... Coordinate rotation cancel

Select the command plane with G17 to G19.



W : Original local coordinate   W' : Rotated local coordinate system   r1 : Rotation angle   (x1, y1) Rotation center

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 "r1" depends on the G90/G91 modal.

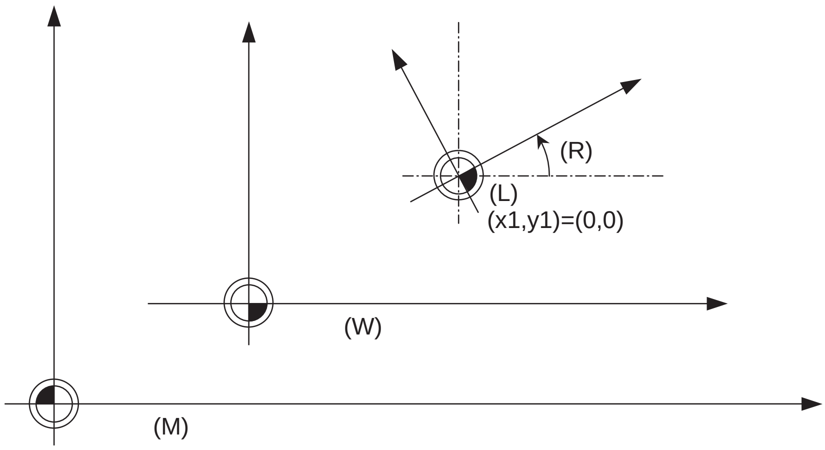
(2) If the rotation center coordinates (x1, y1) are omitted, the position where the G68 command was executed will be the rotation center.

(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 is commanded. Thus, the command of rotation angle "r1"can be omitted.
If the rotation angle is omitted in spite that G68 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.



(R) Rotation angle
(L) Local coordinate system
(W) Workpiece coordinate system
(M) Basic machine coordinate system

(7) The coordinate rotation command during coordinate rotation is processed as the changes of center coordinates and rotation angle.

(8) If M02 or M30 is commanded or the reset signal is input during the coordinate rotation mode, the coordinate rotation mode will be canceled.

(9) G68 is displayed on the modal information screen during the coordinate rotation mode. When the mode is canceled, the display changes to G69. (The modal value is not displayed for the rotation angle command R.)

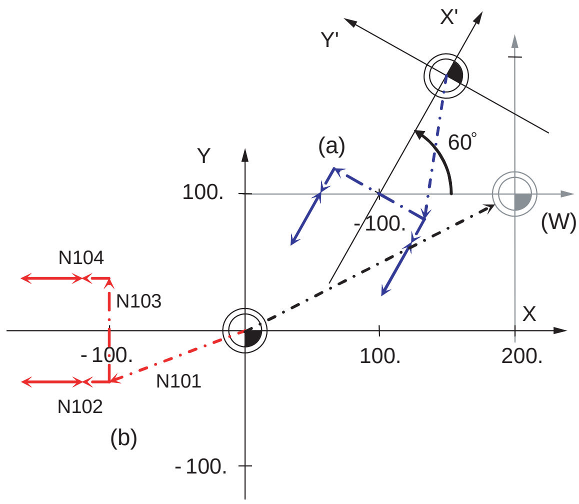
(10) The program coordinate rotation function is valid only in the automatic operation mode.

Program example

Program coordinate rotation by absolute command

N01 G28 X0. Y0.;	
N02 G54 G52 X200. Y100. ;	Local coordinate designation
N03 T10 ;	
N04 G68 X-100. Y0. R60. ;	Coordinate rotation ON
N05 M98 H101 ;	Subprogram execution
N06 G69 ;	Coordinate rotation cancel
N07 G54 G52 X0 Y0 ;	Local coordinate system cancel
N08 M02 ;	End

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 M99



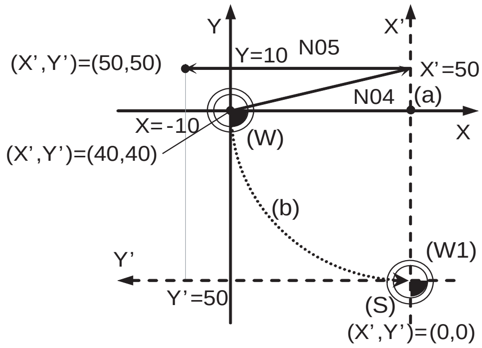
(a) Actual machining shape (b) Program coordinate (W) Local coordinates (before rotation)

Operation when only one axis was commanded by the first movement command

Command basically two axes in the rotation plane by an absolute value immediately after the coordinate rotation command.
When commanding one axis only, the following two kinds of operations 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 when "N04" is "X50.Y0.". The end point is calculated on the assumption that the start point rotates along with the coordinates' rotation.

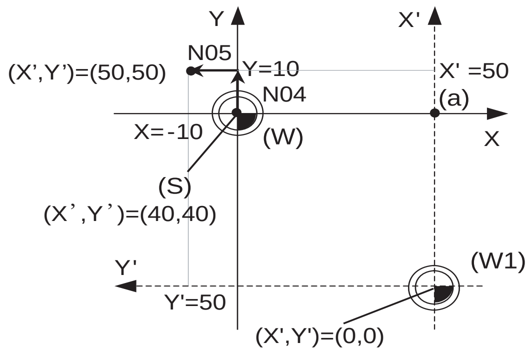
N01 G17 G28 X0. Y0.;	
N02 G90 G92 G53 X0. Y0.;	
N03 G68 X40. Y0. R90.;	Coordinate rotation ON
N04 X50.;G04 X5. ;	
N05 Y50.;	
N06 G69 ;	Coordinate rotation cancel
N07 M02 ;	End



Machine movement path (S) Start point
(a) Rotation center (b) The start point is rotated virtually
(W) Local coordinate system before rotation (W') Local coordinate system after rotation

(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 X40. Y0. R90.;	Coordinate rotation ON
N04 X50.;G04 X5. ;	
N05 Y50.;	
N06 G69 ;	Coordinate rotation cancel
N07 M02 ;	End



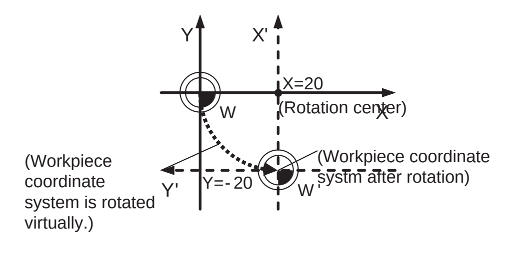
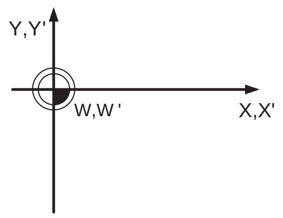
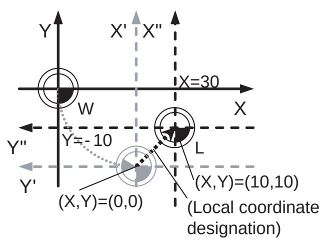
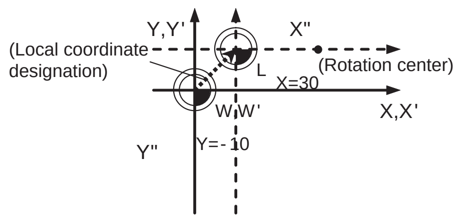
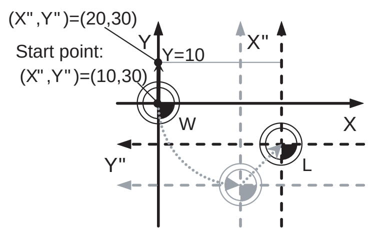
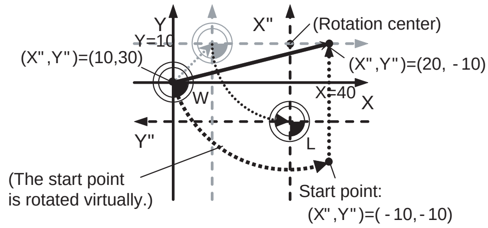
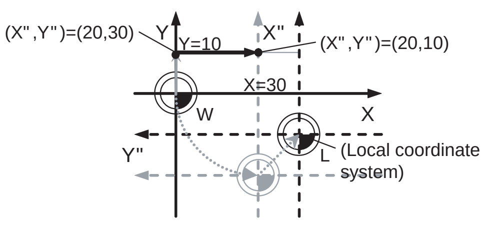
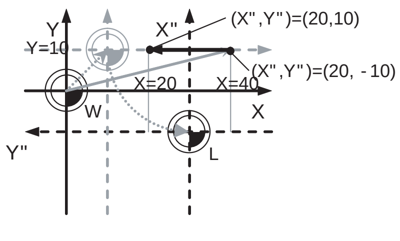
Machine movement path (S) Start point
(a) Rotation center
(W) Local coordinate system before rotation (W') Local coordinate system after rotation

Local coordinate designation during program coordinate rotation

(1) When "#19003 PRG coord rot type" is "0", the position commanded on the rotated coordinate system is set as the local coordinate zero point.

(2) When "#19003 PRG coord rot type" is "1", the position commanded on the coordinate system before it is rotated, is set as the local coordinate zero point and the local coordinate will be rotated.

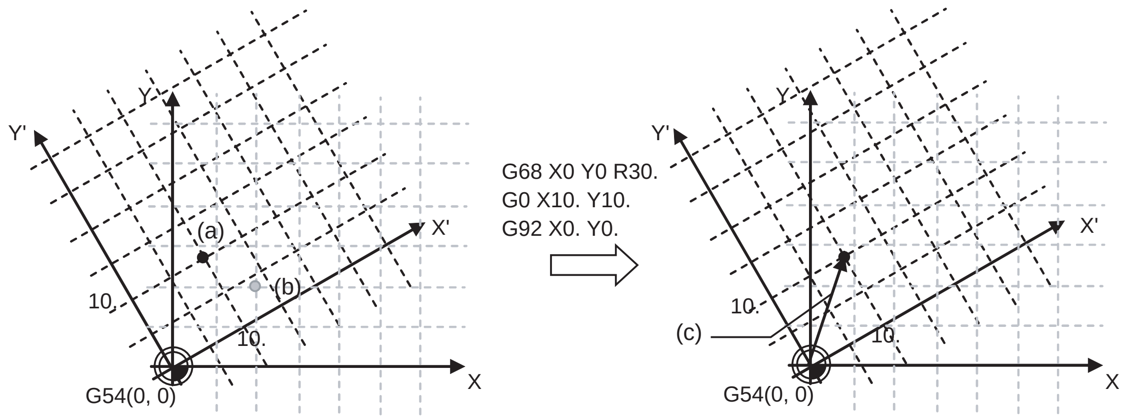
```
N01 G17 G28 X0. Y0.;
N02 G90 G92 G53 X0. Y0.;
N03 G68 X20. Y0. R90.;      Coordinate rotation ON
N04 G52 X10. Y10.;          Local coordinate designation
N05 X20.;
N06 Y10.;
N07 G69 ;                   Coordinate rotation cancel        W : Workpiece coordinate system
N08 M02 ;                   End                               L : Local coordinate system
```

	(1) Operation of #19003 = 0	(2) Operation of #19003 = 1
N03		
	Workpiece coordinate system is rotated virtually.	Workpiece coordinate system is not rotated.
N04		
	The workpiece coordinate zero point after rotation is considered as (X,Y)=(0,0). The position after shifted by 10 each in the X and Y direction is set as the local coordinate zero point. * The direction of the shift is not the direction of X' and Y'.	Designate the local coordinate system on the workpiece coordinate system.
N05		
	The commanded axis moves on the rotation coordinate system. * Axis without movement command does not move.	The commanded axis moves on the rotation coordinate system. * Axis without movement command moves to the position on rotation coordinate system.
N06		

Coordinate system designation during program coordinate rotation

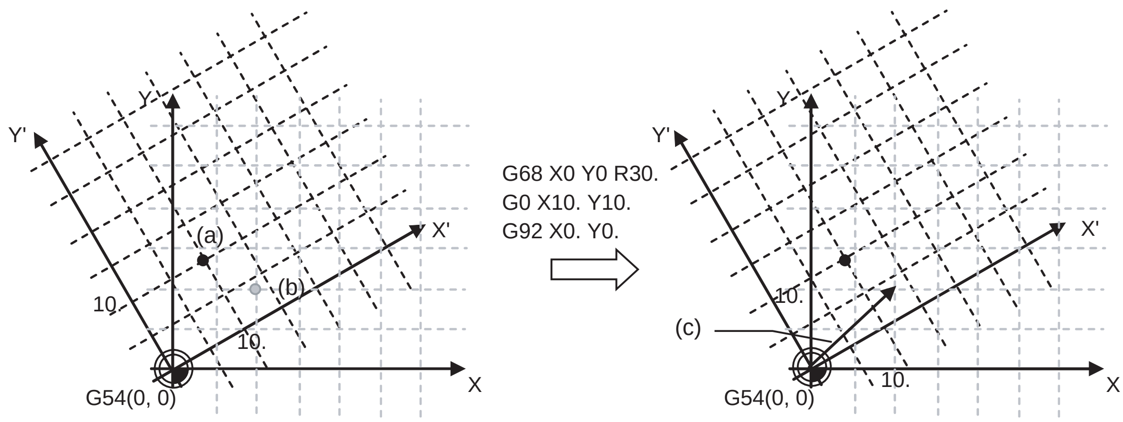
When the coordinate system setting (G92) is executed during program coordinate rotation (G68), this program operates same as "Local coordinate designation during program coordinate rotation".

(1) When "#19003 PRG coord rot type" is "0", the position is preset to the current position commanded on the rotated coordinate system.
(Ex.) Designation on the coordinate system (X'-Y') after rotation



(a) Position after rotation (b) Commanded position (c) G92 shift amount

(2) When "#19003 PRG coord rot type" is "1", the position is preset to the current position commanded on the coordinate system before rotation. The coordinate system is rotated after the position is commanded.
(Ex.) Setting on the coordinate system (X-Y) after rotation



(a) Position after rotation (b) Commanded position (c) G92 shift amount

(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 in respect to the basic machine coordinate system)

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, high-speed high-accuracy I or II is commanded during the coordinate rotation mode.

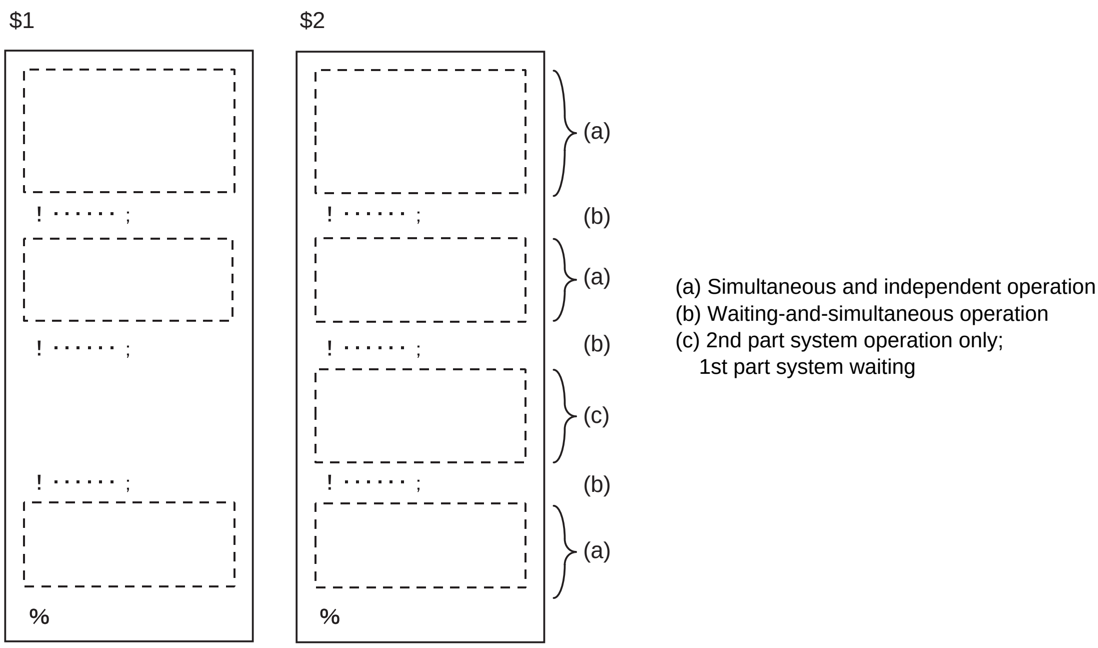
Precautions

(1) Always command an absolute value for the movement command immediately after G68 and G69.
(2) If the manual absolute is ON and interrupted the coordinate rotation axis, then, do not use automatic operation for the following absolute value command.
(3) The intermediate point during reference point 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 on the coordinate system which the command was issued. (The same position in respect to 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 G0Intp" = 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 compensation during the coordinate rotation mode is carried out to 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) On the display, the positions after rotation is always displayed on 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 command. The plane cannot be selected in the same block as the G68 command.
(13) The coordinates can be rotated for the rotation axis. The angle will be interpreted as the length when rotating.

13.14 Waiting-and-simultaneous Operation (! code) ; !L

Function and purpose

The multi-axis, multi-part system complex control NC system can simultaneously run multiple machining programs independently. The 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.

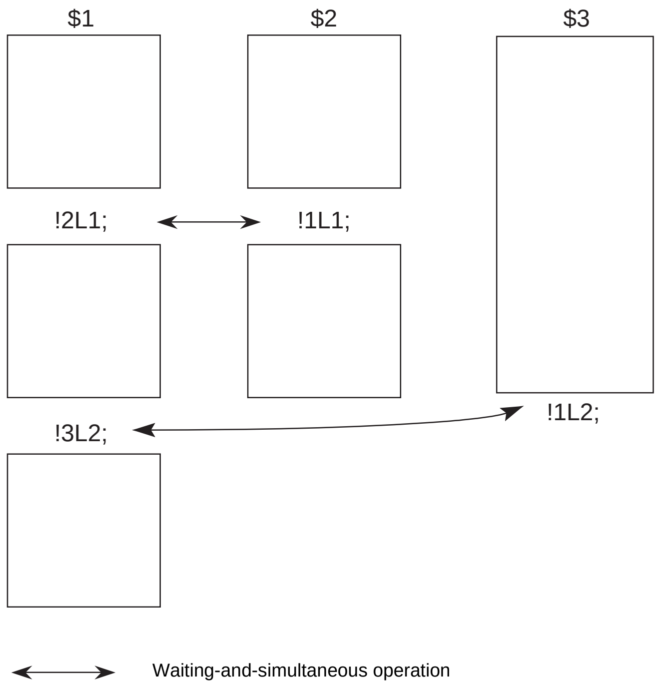


⚠ CAUTION	**1. When programming a multi-part system, carefully observe the movements caused by other part systems' programs.**

Command format

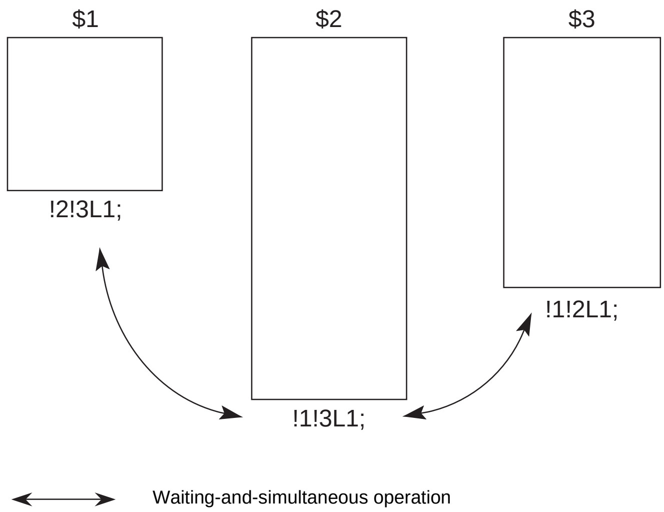
!nLl; ... Command for synchronizing with nth part system

n	Part system number
L	Waiting-and-simultaneous Operation No. 01 to 9999



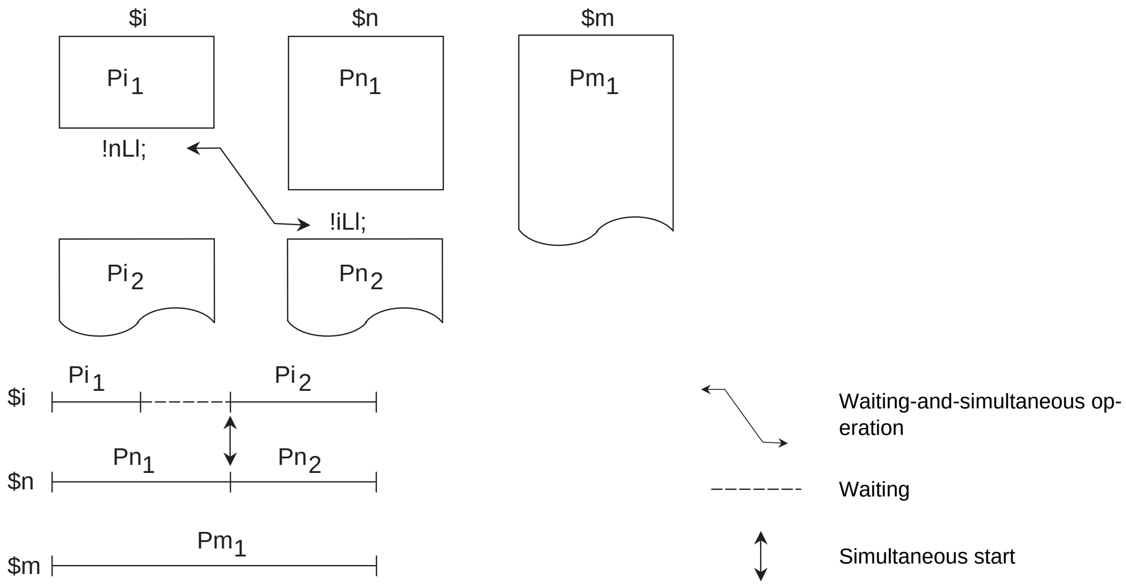
!n!m...Ll; ... Command for synchronizing among three part systems

n,m	Part system number n ≠ m
L	Waiting-and-simultaneous Operation No. 01 to 9999

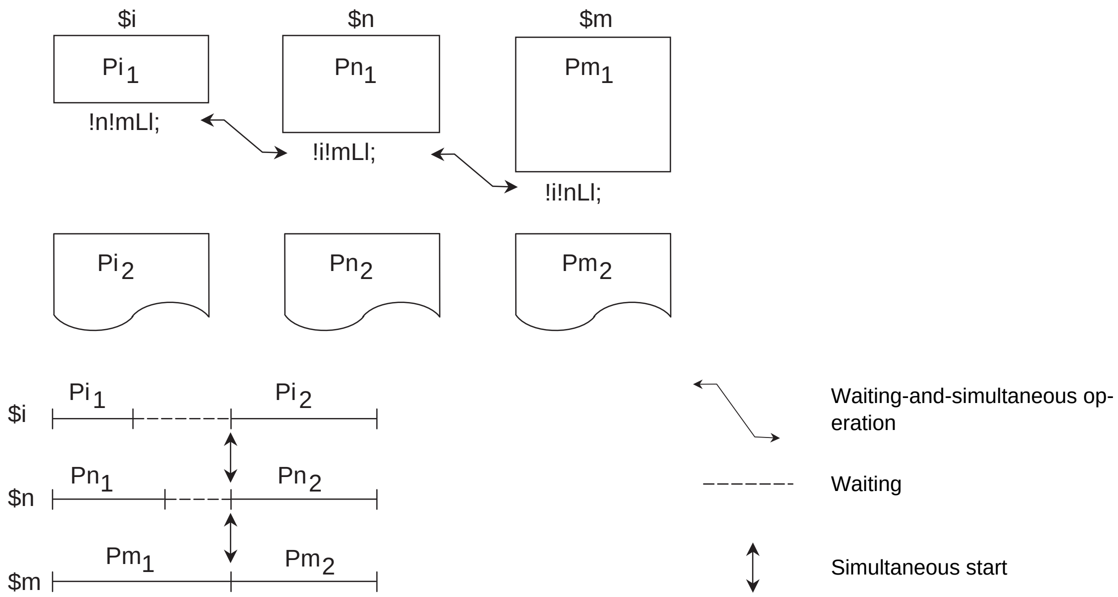


Detailed description

(1) When the "!nLl" code is issued from the part system "i", the operation of that program will wait until the "!iLl" code is issued from the part system "n".
When the "!iLl" code is issued, the programs of both part systems "i" and "n" will start running simultaneously.



(2) Synchronizing among three part systems is as follows. When the "!n!mLl" command is issued from the part system "i", the program of part system "i" operation will wait until the "!i!mLl" command is issued from the part system "n" and the "!i!nLl" command is issued from the part system "m".
When the synchronizing commands are issued, programs of part systems "i", "n" and "m" will start operating simultaneously.



(3) Program error (P35) occurs when an illegal system number has been issued.

(4) The synchronizing 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 : Wait before executing movement command.
1 : Wait after executing movement command.

(5) If there is no movement command in the same block as the waiting-and-simultaneous operation, when the next block movement starts, synchronization may not be secured between the part systems. To synchronize the part systems when movement starts after waiting, issue the movement command in the same block as the waiting-and-simultaneous operation.

(6) Waiting-and-simultaneous operation is done only while the part system to be waited is operating automatically. If this is not possible, the waiting-and-simultaneous operation will be ignored and operation will advance to the next block.

(7) The L command is the waiting-and-simultaneous identification No. The same Nos. are waited but when they are omitted, the Nos. are handled as L0.

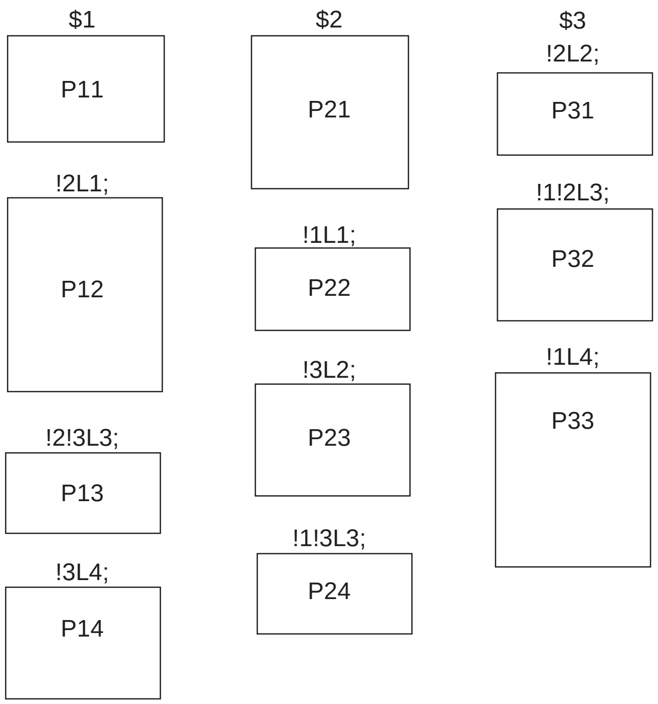
(8) The synchronizing command designates the number of the other part system number to be synchronized, and can also be issued along with its own part system number.
(Example) Part system "i" command: !i!n!mLl;

(9) When the part system No. is omitted (when only "!" is issued), part system 1 will be handled as "!2" and part system 2 as "!1". The command using only "!" cannot be used for synchronizing with part system 3 and following.
If the command using only "!" is used for part system 3 or following, the program error (P33) will occur.

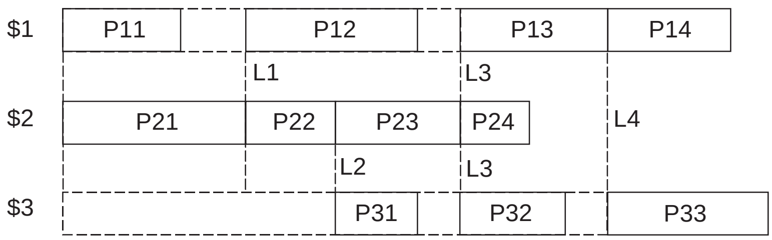
(10) "SYN" will appear in the operation status section during synchronization. The synchronizing signal will be output to the PLC I/F. ($1: X63C, $2: X6BC, $3: X73C, $4: X7BC, $5: X83C, $6: X8BC, $7: X93C)

Operation example

Example of waiting-and-simultaneous operation between part systems



The above programs are executed as follows:



13.15 Start Point Designation Timing Synchronization (Type 1) ; G115

Function and purpose

The part system can wait for the other part system to reach the start point before starting itself.
The waiting-and-simultaneous point can be set in the middle of a block.

Command format

!L__　G115　X__　Y__　Z__ ;

!L	Waiting-and-simultaneous operation
G115	G command
X　Y　Z	Start point (Command axis and workpiece coordinate values for checking waiting of other part system.)

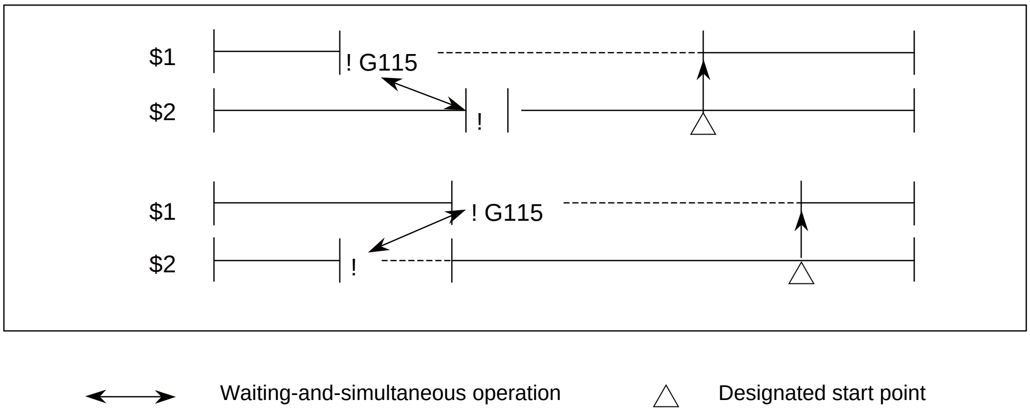
Detailed description

(1) Designate the start point using the workpiece coordinates of the other part system (ex. $2).

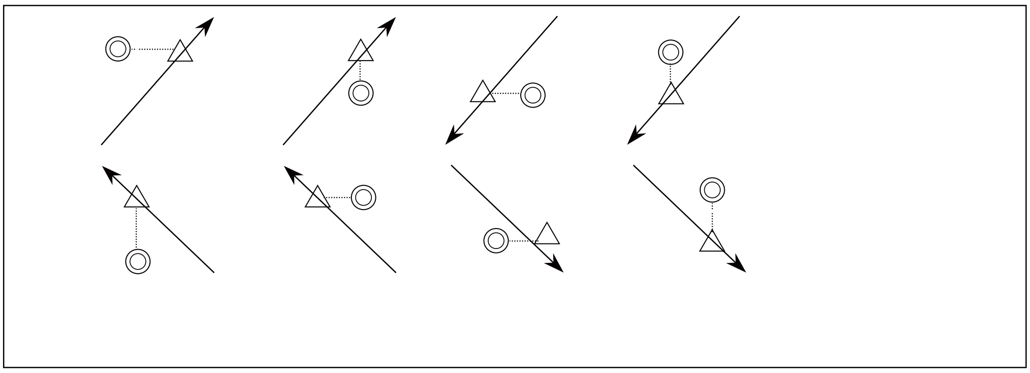
(2) The start point check is executed only for the axis designated by G115.
(Example) !L2 G115 X100. ;
Once the other part system reaches X100., the own part system will start. The other axes are not checked.

(3) The other part system starts first when waiting is executed.

(4) The own part system waits for the other part system to move and reach the designated start point, and then starts.

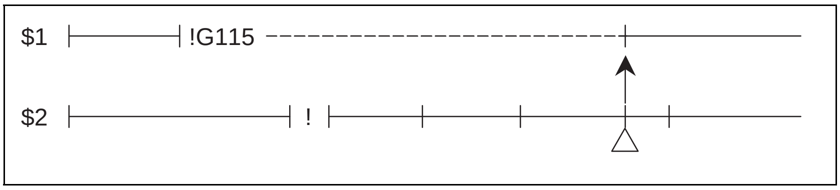


(5) When the start point designated by G115 is not on the next block movement path of the other part system, the own part system starts once all the designated axes of the other part system reach the designated start point.



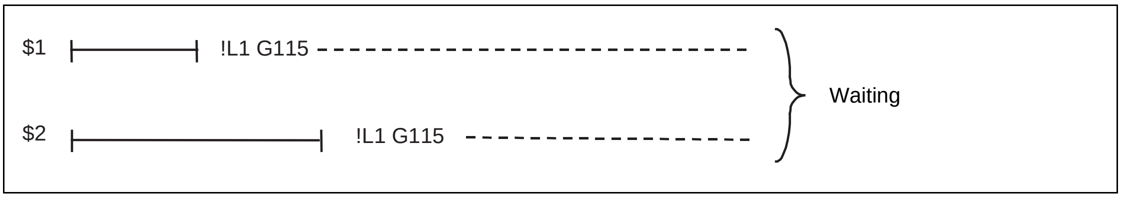
Movement　Command point　Actual start point

(6) The following operation is executed by parameters (base specification parameter #1229 set01/bit5) when the start point cannot be determined by the next block movement of the other system.
(a) When the parameter is ON
Operation waits until the start point is reached by the movement in the next and subsequent blocks.



(b) When the parameter is OFF
The own part system starts upon completion of the next block movement.

(7) The waiting status continues when the G115 command has been duplicated between part systems.



(8) A program error (P33) occurs when the G115 command is issued for 3 part systems.

(9) The single block stop function does not apply for the G115 block.

(10) A program error (P32) will occur if an address other than an axis is designated in G115 command block.

13.16 Start Point Designation Timing Synchronization (Type 2) ; G116

Function and purpose

The own part system can make the other part system to wait until it reaches the start point.
The waiting-and-simultaneous point can be set in the middle of a block.

Command format

!L__ G116 X__ Y__ Z__ ;

!L	Waiting-and-simultaneous operation
G116	G command
X Y Z	Start point (Command axis and workpiece coordinate values for checking waiting of the own part system.)

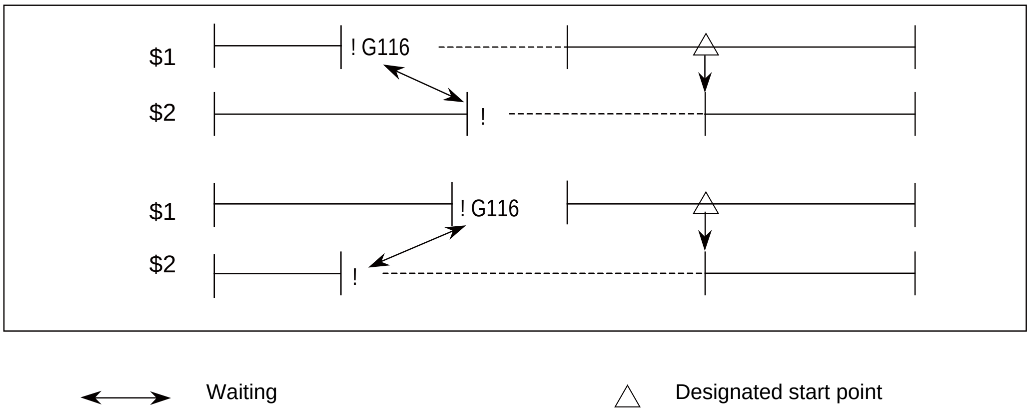
Detailed description

(1) Designate the start point using the workpiece coordinates of the own part system (ex. $1).

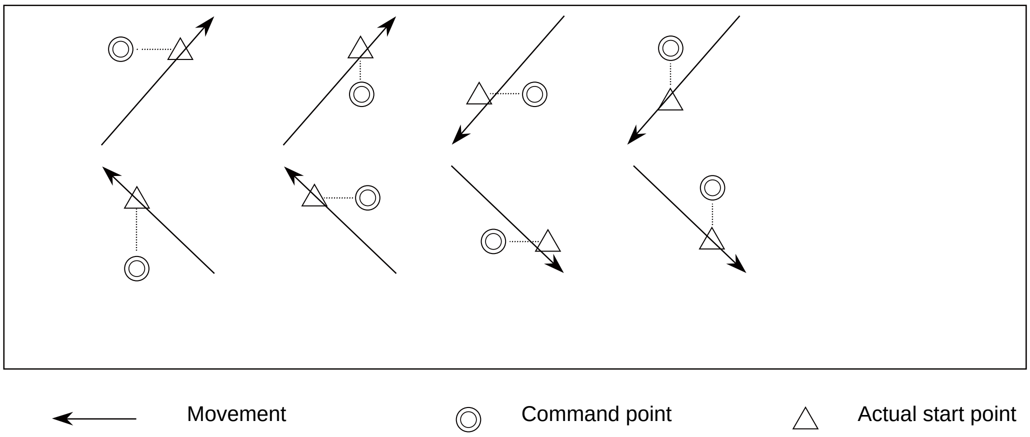
(2) The start point check is executed only for the axis designated by G116.
(Example) !L1 G116 X100. ;
Once the own part system reaches X100., the other part system (ex. $2) will start. The other axes are not checked.

(3) The own part system starts first when waiting is executed.

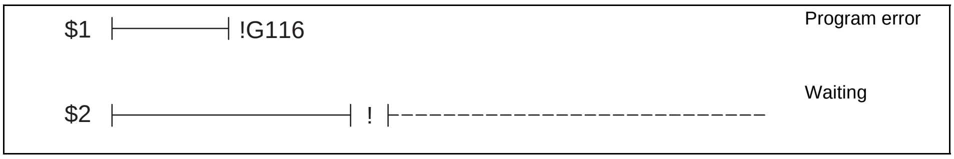
(4) The other part system waits for the own part system to move and reach the designated start point, and then starts.



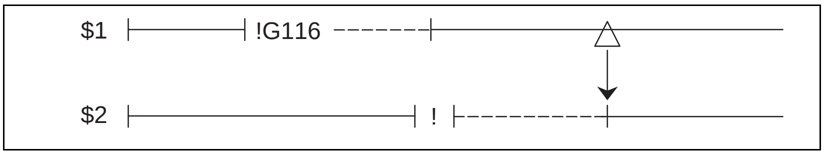
(5) When the start point designated by G116 is not on the next block movement path of own part system, the other part system starts once all the designated axes of the own part system reach the designated start point.



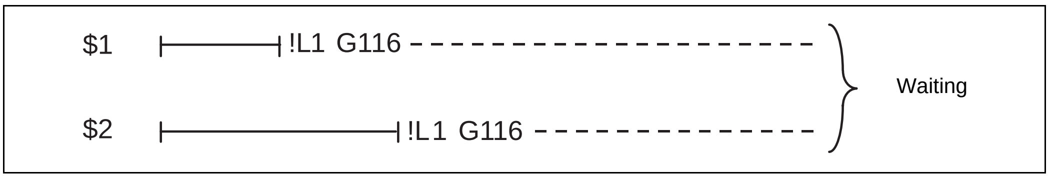
(6) The next operation is executed by parameters (base specification parameter #1229 set01/bit5) when the start point cannot be determined by the next block movement of the own part system.
(a) When the parameter is ON
Program error "P33" occurs before the own part system moves.



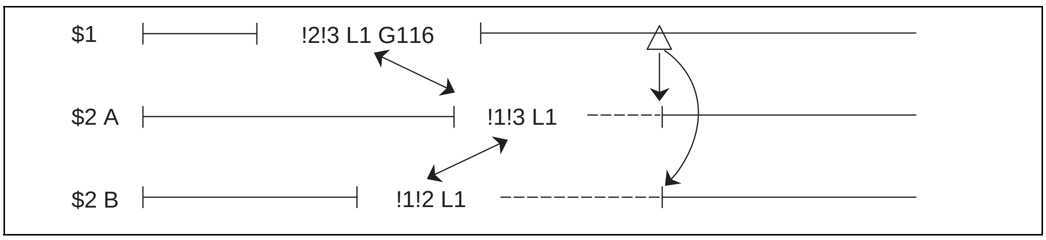
(b) When the parameter is OFF
The other part system starts upon completion of the next block movement.



(7) The waiting status continues when the G116 command has been duplicated between part systems.



(8) The two other part systems start when the G116 command is issued for 3 part systems.



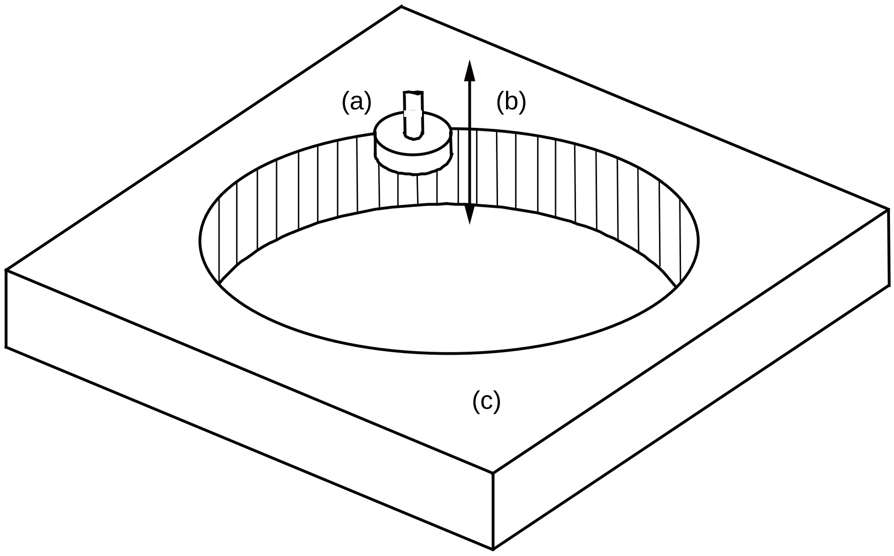
(9) The single block stop function does not apply for the G116 block.

(10) A program error (P32) will occur if an address other than an axis is designated in G116 command block.

13.17 Chopping ; G81.1

Function and purpose

This function continuously raises and lowers the chopping axis independently of the program operation when workpiece contours are to be cut.
There are two types of command for the chopping function: a command by the machining program and a command by a signal from the PLC.



(a) Grindstone
(b) Chopping operation
(c) Workpiece

Command format

G81.1 Z__ Q__ F__; ... Starting the chopping operation

Z	The upper dead point (Select the chopping axis with commanded axis address)
Q	The distance between the upper dead point and the lower dead point. Command with incremental.
F	The feedrate during chopping (mm/min).

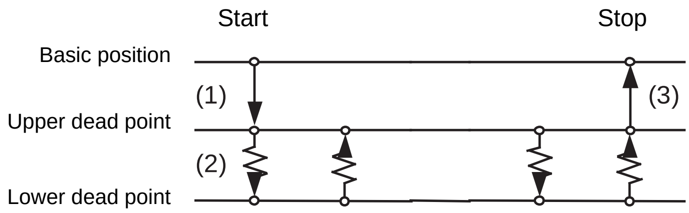
G80 ... Cancelling the chopping operation

Detailed description

(1) When "#1323 chopsel (chopping command method)" is set to "0", the program error (P610) will occur.
(2) Command starting chopping operation (G81.1) in independent blocks. Program error (P33) will occur when commanding G81.1 in the same block with other G command.
(3) When starting chopping action (G81.1), a command for other address (apart from N command) cannot be issued. If other address is commanded, program error (P33) will occur.
(4) Only one axis can be designated for the chopping axis. Program error (P33) will occur when designating two or more axes.
(5) During the chopping operation, movement command does not work. If a movement command is issued, "M01 Operation Error 0151" will occur and all axes will be in the interlock state.
(6) Cancelling the chopping operation (G80) can cancel the fixed cycle modal.
(7) The position where starting the chopping operation (G81.1) is assigned will be the base position for chopping.
(8) Rapid traverse override can be valid for the following movement; travelling from the base position to the upper dead point in starting chopping operation, and traveling from the upper dead point to the base point after the operation. Only when G81.1 commanded, rapid traverse override can be set valid/invalid by external signal. During the chopping operation, the rapid traverse override cannot be switched.
(9) Chopping override can be set during the chopping operation.
Chopping override is only valid for chopping axis and it does not affect other axes.
Also, the axis in chopping operation does not get affected by other override.
If "0 %" is assigned to the chopping override, "M01 operation error 0150" will occur.
(10) During the chopping mode, the upper dead point/the lower dead point and feedrate can be changed by commanding G81.1. However, it is not possible to change the chopping axis. Changing chopping axis can cause the program error (P33).
(11) The chopping operation will be started with G81.1 command, but the machine will not travel to the upper dead point/the lower dead point since the tool offset is done at the initial level of the operation. Until the error amount of command position and feedback position reach the allowable error value, checking with M commands etc. is necessary.
(12) When speed (F) command is large and stroke of the chopping is short, the chopping operation may be done slower than the command speed.

Chopping operation

Operation of the chopping axis



(1) Starting the chopping operation
The chopping mode is entered by issuing the G81.1 command and the chopping operation will be initiated using the current position as a basic position. Chopping is operated after traveling from the base position to the upper dead point with rapid traverse.

(2) During the chopping operation
The axis travels repeatedly between the upper dead point and the lower dead point by designated number of cycles or the feedrate. During the chopping operation, compensation amount is calculated from the machine operation (feedback position at the motor end) and the compensation is conducted so that the machine movement reaches the upper dead point and the lower dead point. (Refer to Chopping compensation.) Traveling during the chopping operation will be performed by soft acceleration/ deceleration.

(3) Stopping the chopping operation
The chopping operation is stopped by issuing the G80 command.
After the chopping operation is performed to the upper dead point, the chopping axis will travel to the base point with rapid traverse. The chopping axis will travel to the lower dead point once even during the travel from the upper dead point to the lower dead point. Acceleration/deceleration will be performed by linear acceleration/deceleration for travelling from the basic point to the upper dead point.

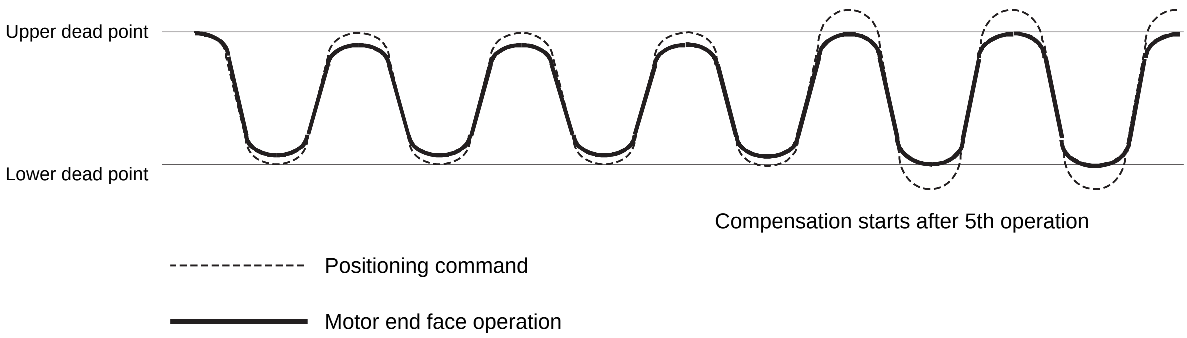
Interrupt operation during chopping

When the interruption, which affects the chopping axis, occurs during the chopping operation, the chopping axis performs as follow.

Interrupt operation	Chopping performance	Chopping mode
Reset	Travel to the basic position with rapid traverse after travelling to the upper dead point immediately.	Cancel
Feed hold	Performance can be continued without stopping.	Hold
Block stop		
Axis interlock	Decelerates and stops. The chopping operation starts again after cancelling.	
Door interlock II		
Door interlock I	Decelerates and stops.	Cancel
Servo OFF		
Axis detachment		
Stroke end		
Emergency stop	Stops immediately.	
Program error	Travel to the basic position with rapid traverse after travelling to the upper dead point immediately.	Hold

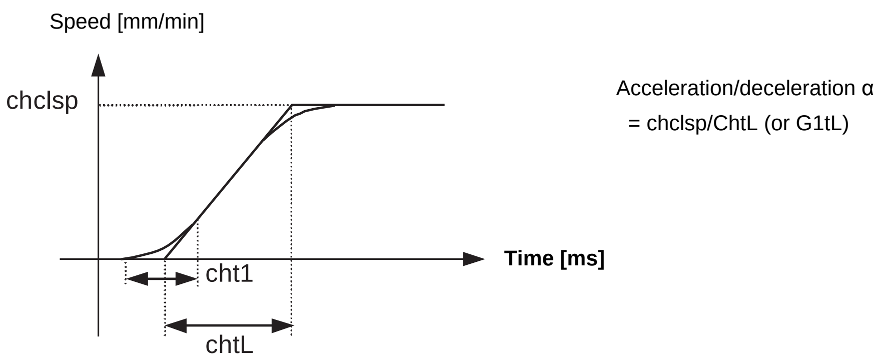
Chopping compensation operation (Compensation value sequential update type)

This function uses the method that the compensation amount is calculated with the machine operation (feedback position of a motor end face) rather than using in-position check for assigning the position since the positioning command is compensated for high-speed repeated operation. The compensation amount for assigning the position can be calculated from the difference between command position and feedback position every four cycle. Then the compensation amount is added on the positioning command for the next cycle to eliminate the difference between command position and feedback position.



Chopping feedrate

The feedrate of the chopping axis will be clamped at the clamp speed (#2081 chclsp) of the chopping axis. When 0 is set to the chopping clamp speed, it will be clamped at the G1 clamp speed (#2002 clamp). The acceleration/deceleration time constant can be set by chopping acceleration/deceleration time constant (#2141 chtL). When 0 is set to the chopping axis acceleration/deceleration time constant, the linear acceleration/deceleration time constant (#2007 G1tL) will be executed.



Program example

(1) Machining condition
Chopping axis : Z axis
Basic point coordinate : (Machine) -20.0, the upper dead point : -25.0, the lower dead point : -45.0
Chopping speed : 1000[mm/min]

(2) Program

O1000();	
N0010 G91 G54;	
N0020 G28 X0 Y0 Z0;	
N0030 G90 G00 X10. Y10.;	
N0040 G53 Z-20.;	Z-20.0 of the machine coordinate is the basic point
N0050 G91;	
N0060 G81.1 Z-5. Q-20. F1000;	Command Z axis to the chopping axis
N0070 M70;	Waiting to be finished chopping compensation
N0080 G00 X-10. Y-10.;	
N0090 G01 X50. F300;	
N0100 G01 Y50.;	
N0110 G01 X-50.;	
N0120 G01 Y-50.;	
N0130 G00 X10. Y10.	
N0140 G80;	Chopping axis returns to the basic point
N0150 G28 Z0.;	
N0160 G28 X0 Y0;	
N0170 M30;	

14

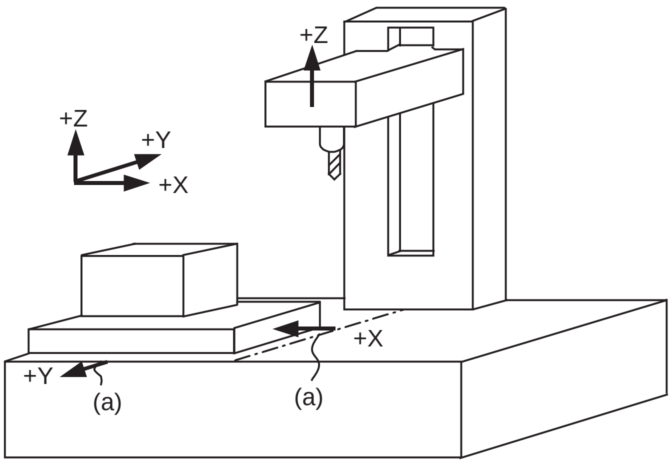
Coordinate System Setting Functions

14.1 Coordinate Words and Control Axes

Function and purpose

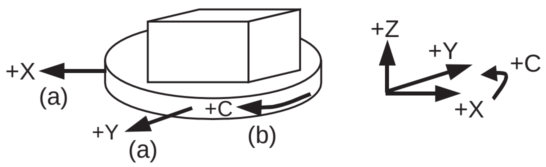
In the standard specifications, there are 3 control axes, however, by adding an additional axis, up to 16 axes can be controlled. The designation of the processing direction responds to those axes and uses a coordinate word made up of alphabet characters that have been decided beforehand.

X-Y table



(a) Direction of table movement

X-Y and rotating table

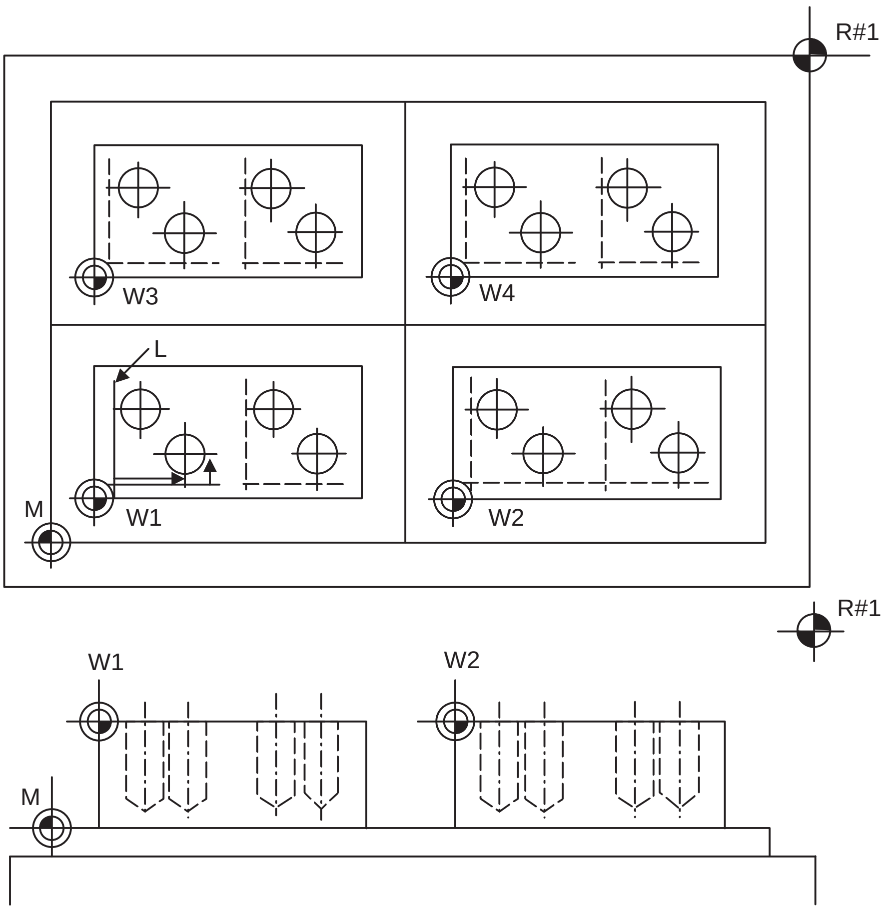


(a) Direction of table movement (b) Direction of table rotation

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 basic point 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.
Upon completion of the reference position return, the basic machine coordinate system and workpiece coordinate systems (G54 to G59) are automatically set with reference to the parameters.
The basic machine coordinate system is set so that the first reference position is brought to the position specified by the parameter from the basic machine coordinate zero point (machine zero point).

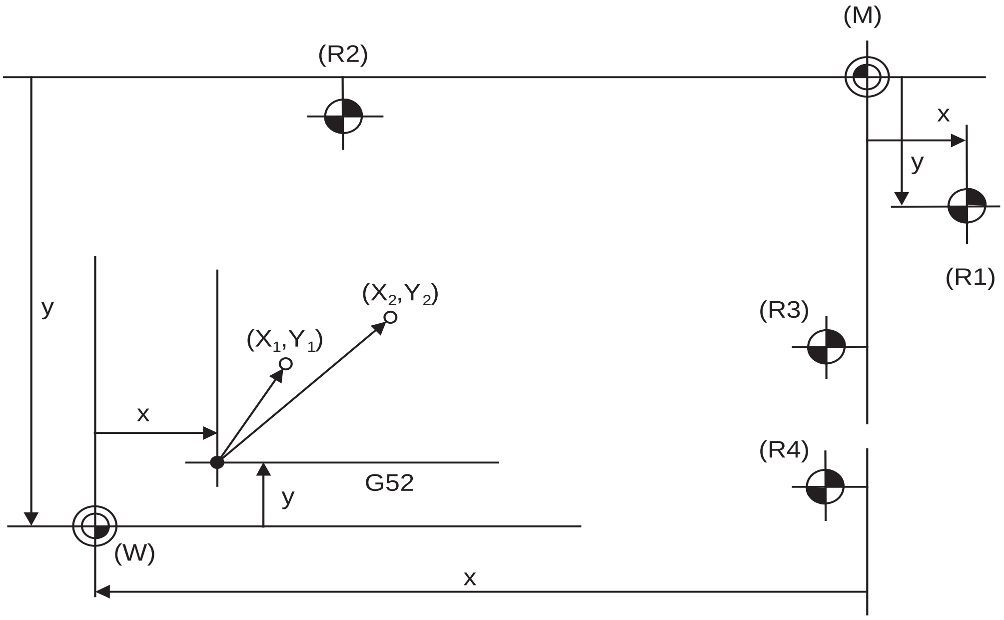


M : Basic machine coordinate system
W : Workpiece coordinate system
L : Local coordinate system

14.3 Machine Zero Point and 2nd, 3rd, 4th Reference Position (Zero point)

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) point 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.

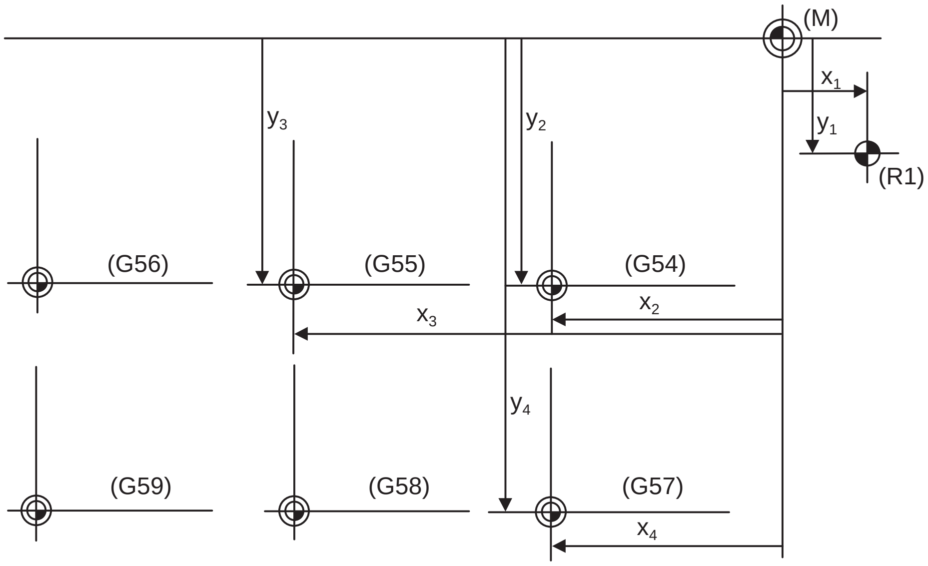


(M) Basic machine coordinate system (G52) Local coordinate system
(R1) 1st reference position (R2) 2nd reference position (W) Workpiece coordinate systems (G54 to G59)
(R3) 3rd reference position (R4) 4th reference position

14.4 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.
The actual machining program is programmed over the coordinate systems which have been set above.



(M) Basic machine coordinate system (R1) 1st reference position
(G54) Workpiece coordinate system 1 (G55) Workpiece coordinate system 2 (G56) Workpiece coordinate system 3
(G57) Workpiece coordinate system 4 (G58) Workpiece coordinate system 5 (G59) Workpiece coordinate system 6

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, after deciding at which position the first reference position should be set in the basic machine coordinate system and then set the zero point positions of the workpiece coordinate systems.

(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 interrupt will be canceled.

(4) The dog-type reference position return will be executed when the first time manual reference position return or the first time automatic reference position return is executed after the power has been turned ON. It will be also executed when the dog-type is selected by the parameter for the manual reference position return or the automatic reference position return for the second time onwards.

⚠CAUTION

1. If the workpiece coordinate offset amount is changed during automatic operation (including during single block operation), it will be validated from the next block or after multiple blocks of the command.

14.5 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.

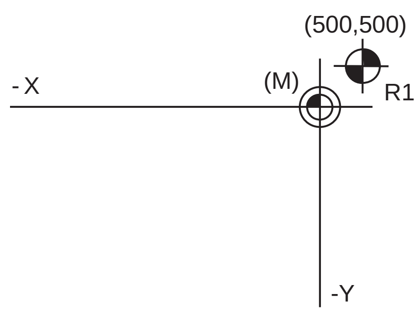
Command format

(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) All the G53 command move at rapid traverse rate.
(8) If the G53 command and G28 command (reference position return) are issued in the same block, the command issued last will be valid.



(M) Basic machine coordinate system
(R1) 1st reference position

1st reference position coordinate value: X=+500 and Y=+500

14.6 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.

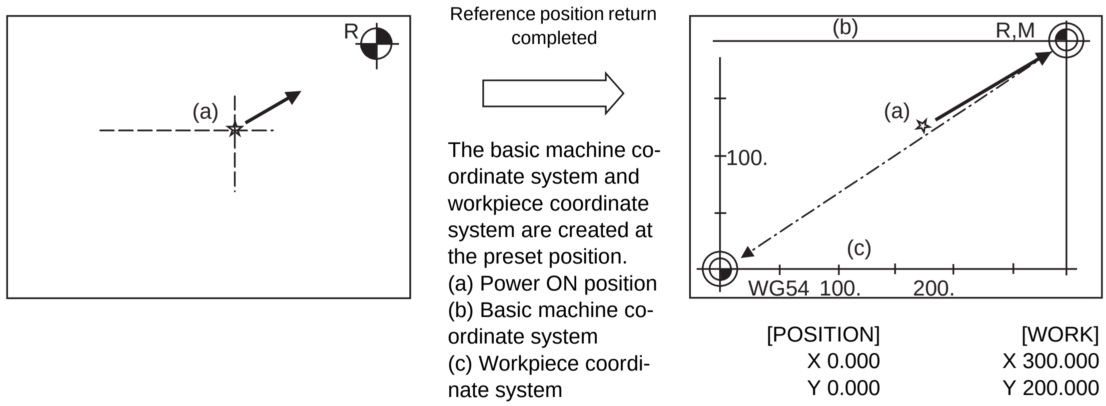
Command format

G92 X__ Y__ Z__ α__ ;

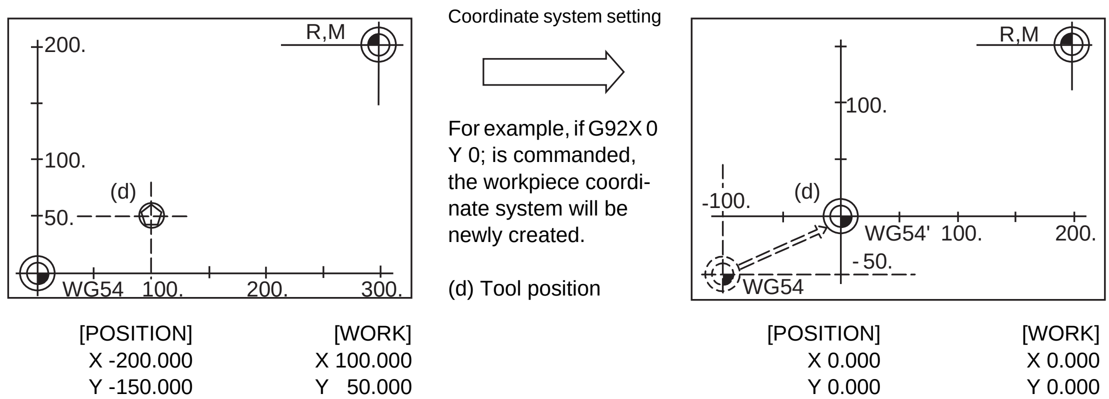
α	Additional axis

Detailed description

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.

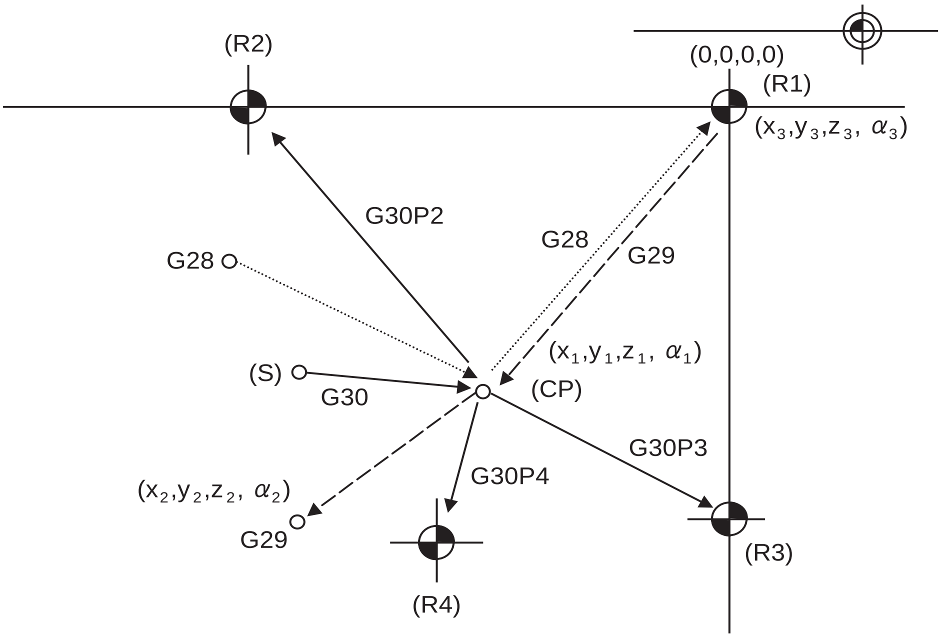


(Note) If the workpiece coordinate system deviated because the axis is moved manually when the manual absolute position switch is OFF, etc., the workpiece coordinate system can be corrected with the following steps.
(1) Execute reference position return while the coordinate system is deviated.
(2) After 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.

14.7 Reference Position (Zero point) Return ; G28,G29

Function and purpose

After the commanded axes have been positioned by G0, they are returned respectively at rapid traverse to the first reference position when G28 is commanded.
By commanding G29, the axes are first positioned independently at high speed to the G28 or G30 intermediate point and then positioned by G0 to the commanded position.



(CP) Intermediate point	(R1) 1st reference position	(R2) 2nd reference position
(S) Start point	(R3) 3rd reference position	(R4) 4th reference position

Command format

G28 Xx1 Yy1 Zz1 αα1; ... Automatic reference position return

G29 Xx2 Yy2 Zz2 αα2; ... Start point return

αα1/αα2	Additional axis

Detailed description

(1) The G28 command is equivalent to the following:
G00 Xx1 Yy1 Zz1 α α1;
G00 Xx3 Yy3 Zz3 α α3;
In this case, Xx3, Yy3, Zz3 and α3 are the reference position coordinates and they are set by parameter "#2037 G53ofs" as the distance from the basic machine coordinate system zero point.

(2) After the power has been switched on, the axes which have not been subject to manual reference position return 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 subsequence 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 Xx1 Yy1 Zz1 α α1;
G00 Xx2 Yy2 Zz2 α α2
Rapid traverse (non-interpolation type) applies independently for each axis for the positioning from the reference position to the intermediate point.
In this case, x1 y1 z1 and α1 are the coordinate value of the G28 or G30 intermediate point.

(5) Program error (P430) occurs when G29 is executed without executing automatic reference position (zero point) return (G28) after the power has been turned 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 (x1, y1, z1, α1) 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.

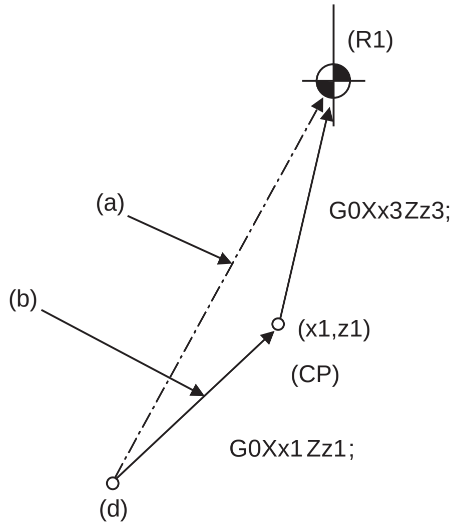
(9) The tool offset will be canceled during reference position return unless it is already canceled, and the offset amount will be cleared.

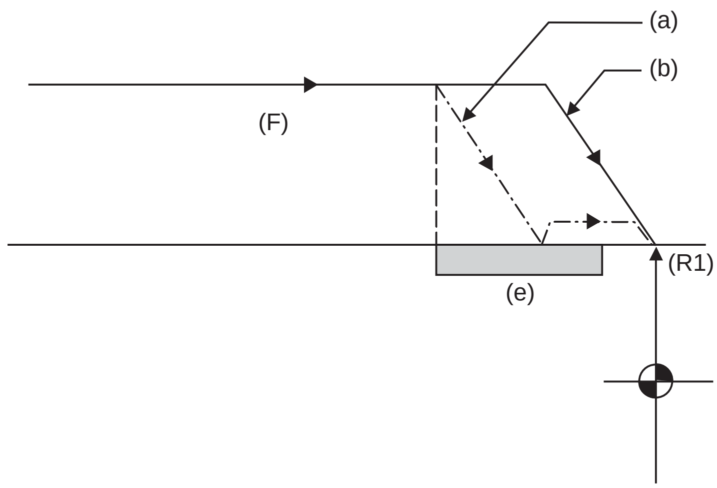
(10) Control from the intermediate point to the reference (zero) point is ignored for reference (zero) point return in the machine lock status. The next block is executed when the commanded axis survives as far as the intermediate point.

(11) Mirror image is valid from the start point to the intermediate point during reference (zero) point return in the mirror image mode 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) point and the tool will move to the reference (zero) point.

Program example

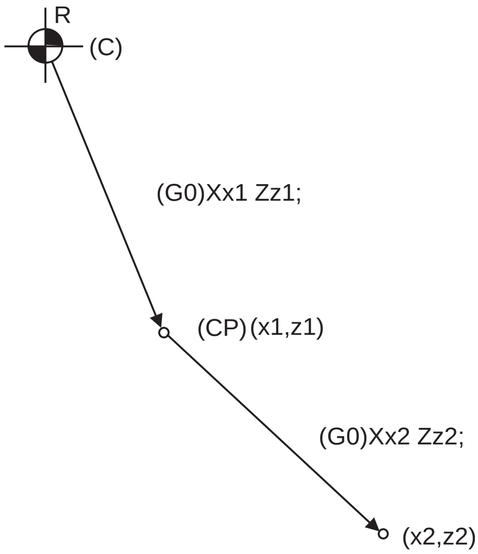
(Example 1) G28 Xx1 Zz1 ;





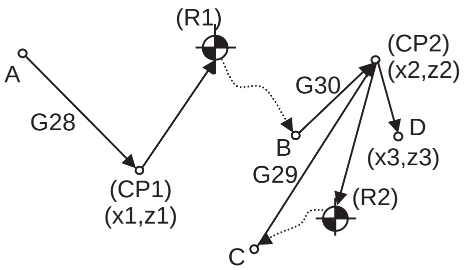
(a) 1st operation after power has been turned ON	(b) 2nd and subsequent operations	(d) Return start position
(e) Near-point dog	(F) Rapid traverse rate	(CP) Intermediate point
(R1) Reference position (#1)		

(Example 2) G29 Xx2, Zz2 ;



(C) Current position (CP) G28, G30 Intermediate point

(Example 3) G28 Xx1 Zz1 ;
: (From point A to 1st reference position)
:
G30 Xx2 Zz2 ;
: (From point B to 2nd reference position)
:
G29 Xx3 Zz3 ;
(From point C to point D)

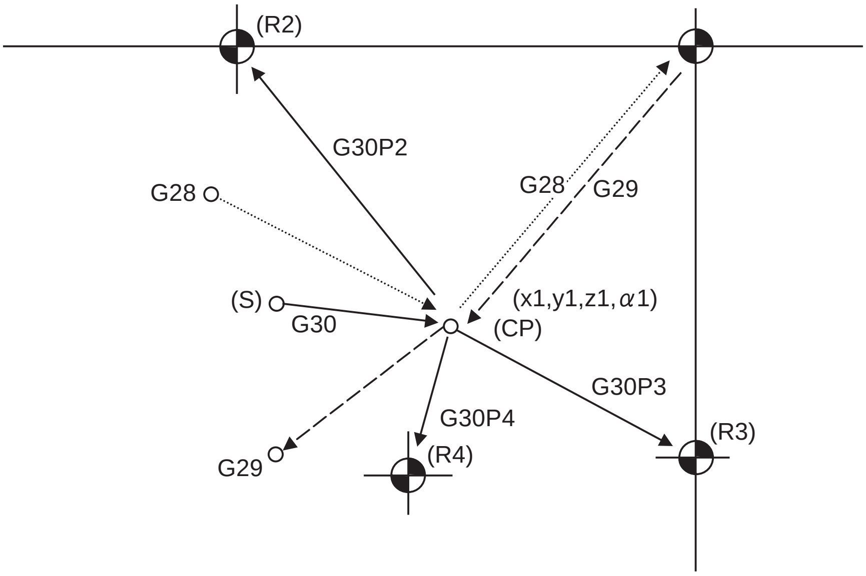


(CP1) Old intermediate point (CP2) New intermediate point
(R1) Reference position (#1) (R2) 2nd reference position (#2)

14.8 2nd, 3rd, and 4th Reference Position (Zero point) Return ; G30

Function and purpose

The tool can return to the second, third, or fourth reference position by specifying G30 P2 (P3 or P4).



(S) Start point
(CP) Intermediate point
(R2) 2nd reference position
(R3) 3rd reference position
(R4) 4th reference position

Command format

G30 P2(P3,P4)Xx1 Yy1 Zz1 αα1;

αα1	Additional axis

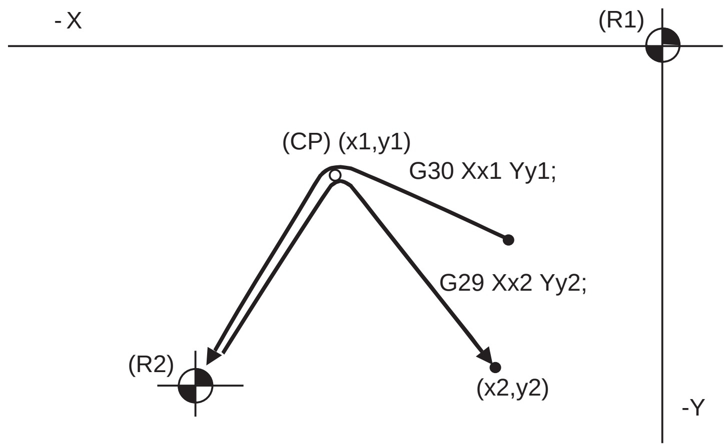
Detailed description

(1) The 2nd, 3rd, or 4th reference position return is specified by P2, P3, or P4.
A command without P or with other designation method will return the tool to the 2nd reference position.

(2) In the 2nd, 3rd, or 4th reference position return mode, as in the 1st reference position return mode, the tool returns to the 2nd, 3rd, or 4th reference position via the intermediate point specified by G30.

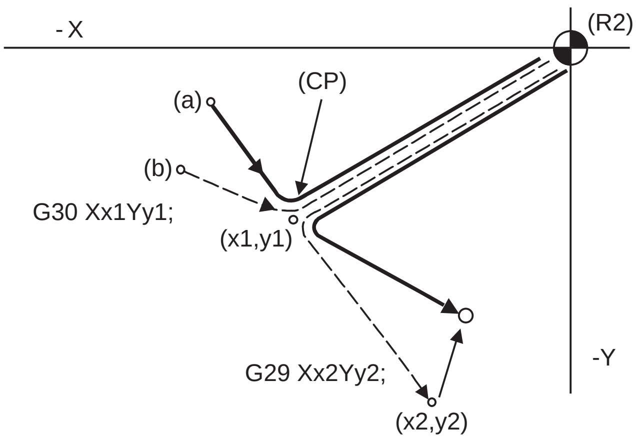
(3) The 2nd, 3rd, and 4th reference position coordinates refer to the positions specific to the machine, and these can be checked with the setting and display unit.

(4) If G29 is commanded after completion of returning to the 2nd, 3rd, and 4th reference position, the intermediate position used last is used as the intermediate position for returning by G29.



(CP) Intermediate point (R1) 1st reference position (R2) 2nd reference position

(5) With reference position return on a plane during compensation, the tool moves without tool radius compensation (zero compensation) from the intermediate point as far as the reference position. With a subsequent G29 command, the tool move without tool radius compensation from the reference position to the intermediate point and it moves with such compensation until the G29 command from the intermediate point.

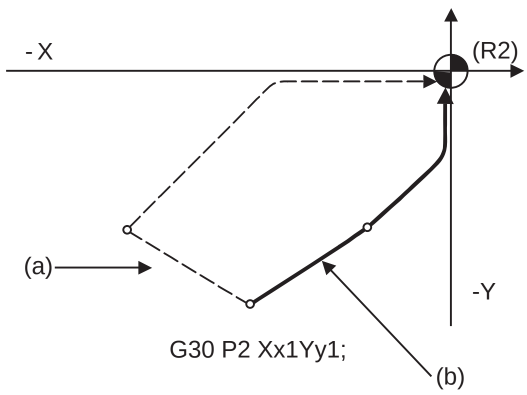


(a) Tool nose center path (b) Program path
(CP) Intermediate point (R1) 1st reference position (R2) 2nd reference position

(6) The tool length offset amount for the axis involved is canceled after the 2nd, 3rd and 4th reference position return.

(7) With second, third and fourth reference (zero) point returns in the machine lock status, control from the intermediate point to the reference (zero) point 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 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 position and the tool moves to the reference position.



(a) X-axis mirror image (b) No mirror image (R2) 2nd reference 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 1st 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 1st reference position and return to the 1st reference position, it is possible to check whether the tool has returned to the reference position after the program has been run.

Command format

X__ Y__ Z__ P__ ; ... Check command

X Y Z	Return control axis
P	Check No. 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 1st 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 48 sets of added workpiece coordinate systems, as well as 6 workpiece coordinate systems, which are used by the programmer for programming (G54 to G59). (The 48 sets are options.)

(3) Among the workpiece coordinate systems currently selected by these commands, any workpiece coordinate system with coordinates which have been commanded by the current position of the tool is reset. (The "present position of the tool" includes the offset amounts for tool radius, tool length and tool position.)

(4) A hypothetical machine coordinate system with coordinates which have been commanded by the current position of the tool is set by this command.
(The "present position of the tool" includes the offset amounts for tool radius, tool length and tool position offset.) (G54,G92)

Command format

(G90) G54 to G59 ... Workpiece coordinate system selection

(G54 to G59) G92 X__ Y__ Z__ α__ ; ... Workpiece coordinate system setting

α	Additional axis

G54.1 Pn ; ... Workpiece coordinate system selection (extended: P1 to P48)

G54.1 Pn ;
G92 X__ Y__ Z__ ; ... Workpiece coordinate system setting (extended: P1 to P48)

G10 L20 Pn X__ Y__ Z__ ; ... Workpiece coordinate system offset amount setting (extended: P1 to P48)

Detailed description

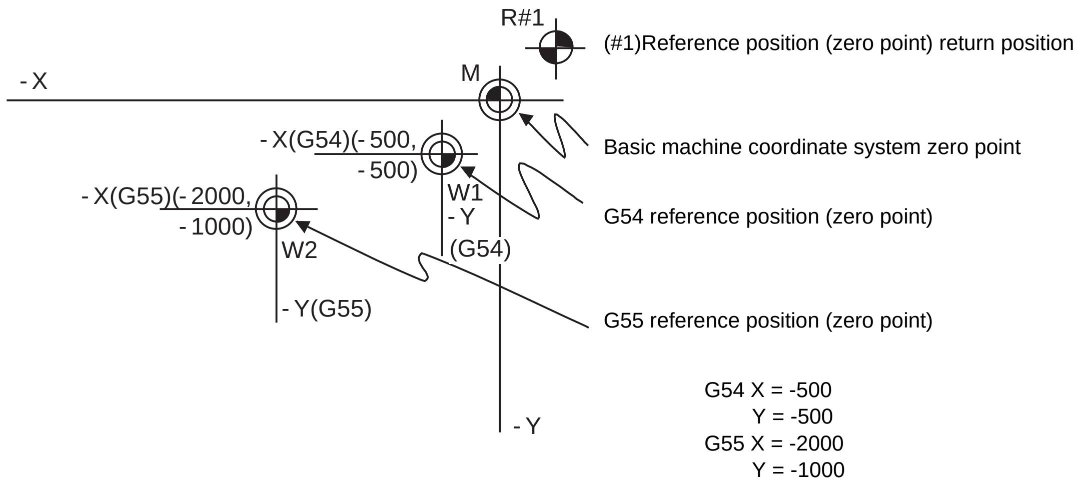
(1) With any of the G54 through G59 commands, the tool radius compensation amounts for the commanded axes will not be canceled even if workpiece coordinate system selection is commanded.

(2) The G54 workpiece coordinate system is selected when the power is turned ON.

(3) Commands G54 through G59 are modal commands (group 12).

(4) The coordinate system will move with G92 in a workpiece coordinate system.

(5) The offset setting amount in a workpiece coordinate system denotes the distance from the basic machine coordinate system zero point.



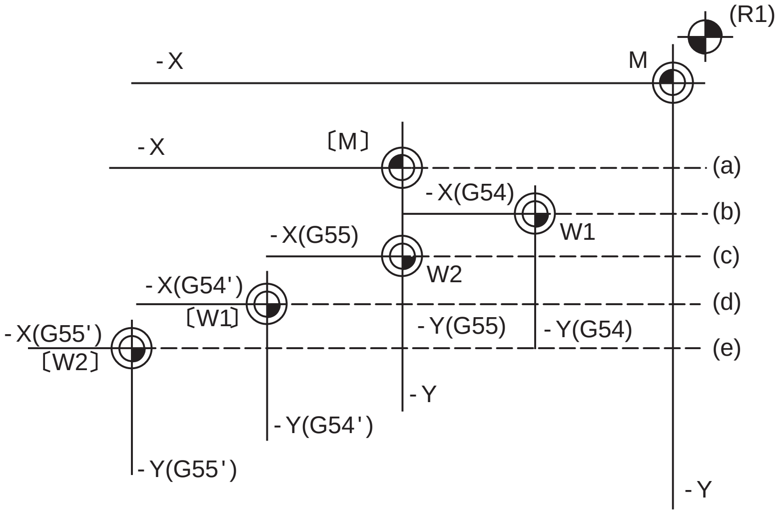
(6) The offset settings of workpiece coordinate systems can be changed any number of times. (They can also be changed by G10 L2 Pp1 Xx1 Yy1 Zz1.)

[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 48 : 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 ; G10 Xx Yy Zz ;	When there is no L value, it is regarded as L2 (workpiece 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 to 6 (G55 to G59) will move in parallel and new workpiece coordinate systems 2 to 6 will be set.

(8) A hypothetical machine coordinate system is formed at the position which deviates from the new workpiece reference position (zero point) by an amount equivalent to the workpiece coordinate system offset amount.



After the power has been switched on, the hypothetical machine coordinate system is matched with the basic machine coordinate system by the first automatic (G28) or manual reference position (zero point) return.

(R1) Reference position 1
(a) Hypothetical machine coordinate system based on G92
(b) Old workpiece 1 (G54) coordinate system
(c) Old workpiece 2 (G55) coordinate system
(d) New workpiece 1 (G54) coordinate system
(e) New workpiece 2 (G55) coordinate system

(9) By setting the hypothetical machine coordinate system, the new workpiece coordinate system will be set at a position which deviates from that hypothetical machine coordinate system by an amount equivalent to the workpiece coordinate system offset amount.

(10) When the first automatic (G28) or manual reference position (zero point) return is completed after the power has been turned ON, the basic machine coordinate system and workpiece coordinate systems are set automatically in accordance with the parameter settings.

(11) If G54 X- 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.

(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.

(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.
[Variable Nos. of the extended workpiece coordinate offset system]

	1st axis to 16th axis			**1st axis to 16th axis**
P1	#7001 to #7016		P25	#7481 to #7496
P2	#7021 to #7036		P26	#7501 to #7516
P3	#7041 to #7056		P27	#7521 to #7536
P4	#7061 to #7076		P28	#7541 to #7556
P5	#7081 to #7096		P29	#7561 to #7576
P6	#7101 to #7116		P30	#7581 to #7596
P7	#7121 to #7136		P31	#7601 to #7616
P8	#7141 to #7156		P32	#7621 to #7636
P9	#7161 to #7176		P33	#7641 to #7656
P10	#7181 to #7196		P34	#7661 to #7676
P11	#7201 to #7216		P35	#7681 to #7696
P12	#7221 to #7236		P36	#7701 to #7716
P13	#7241 to #7256		P37	#7721 to #7736
P14	#7261 to #7276		P38	#7741 to #7756
P15	#7281 to #7296		P39	#7761 to #7776
P16	#7301 to #7316		P40	#7781 to #7796
P17	#7321 to #7336		P41	#7801 to #7816
P18	#7341 to #7356		P42	#7821 to #7836
P19	#7361 to #7376		P43	#7841 to #7856
P20	#7381 to #7396		P44	#7861 to #7876
P21	#7401 to #7416		P45	#7881 to #7896
P22	#7421 to #7436		P46	#7901 to #7916
P23	#7441 to #7456		P47	#7921 to #7936
P24	#7461 to #7476		P48	#7941 to #7956

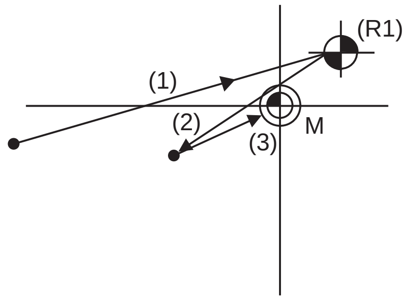
⚠ **CAUTION**

1. If the workpiece coordinate system offset amount is changed during single block stop, the new setting will be valid from the next block.

Program example

(Example 1)

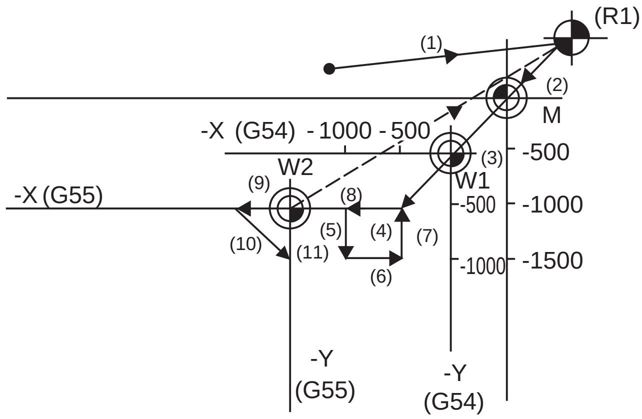
(1) G28 X0 Y0 ;
(2) G53 X-1000 Y-500 ;
(3) G53 X0 Y0 ;



When the 1st reference position coordinate position is zero, the basic machine coordinate system zero point and reference position (zero point) return position (#1) will coincide.

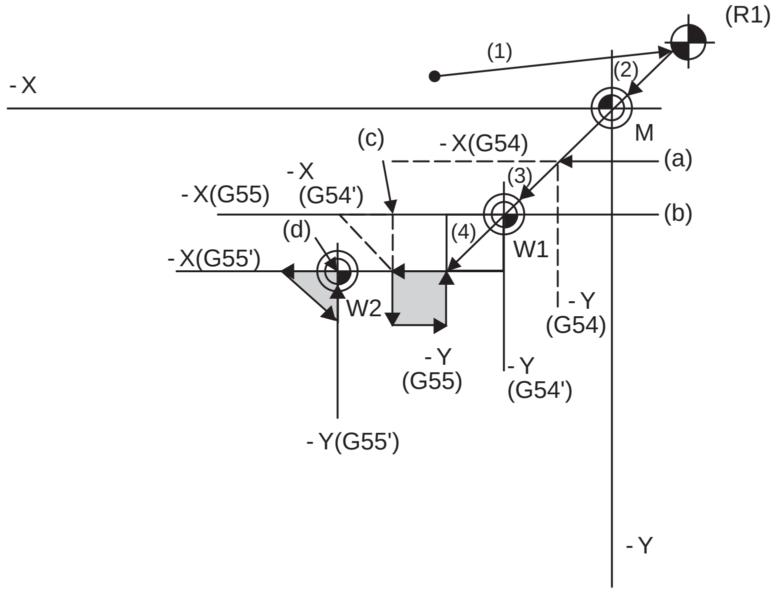
(Example 2)

(1) G28 X0 Y0 ;
(2) G90 G00 G53 X0 Y0 ;
(3) G54 X-500 Y-500 ;
(4) G01 G91 X-500 F100 ;
(5) Y-500 ;
(6) X+500 ;
(7) Y+500 ;
(8) G90 G00 G55 X0 Y0 ;
(9) G01 X-500 F200 ;
(10) X0 Y-500 ;
(11) G90 G28 X0 Y0 ;



(Example 3) When workpiece coordinate system G54 (-500, -500) has deviated in Example 2. (It is assumed that (3) to (10) in Example 2 have been entered in subprogram 1111.)

(1)	G28 X0 Y0 ;	
(2)	G90 G00 G53 X0 Y0 ;	(This is not required when there is no G53 offset.)
(3)	G54 X-500 Y-500 ;	Amount by which workpiece coordinate system deviates
(4)	G92 X0 Y0 ;	New workpiece coordinate system is set.
(5)	M98 P1111 ;	



(a) Old G54 coordinate system
(b) New G54 coordinate system
(c) Old G55 coordinate system
(d) New G55 coordinate system
(R1) Reference position return position

(Note) The workpiece coordinate system will deviate each time when steps (3) to (5) are repeated. The reference position return (G28) command should therefore be issued upon completion of the program.

(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

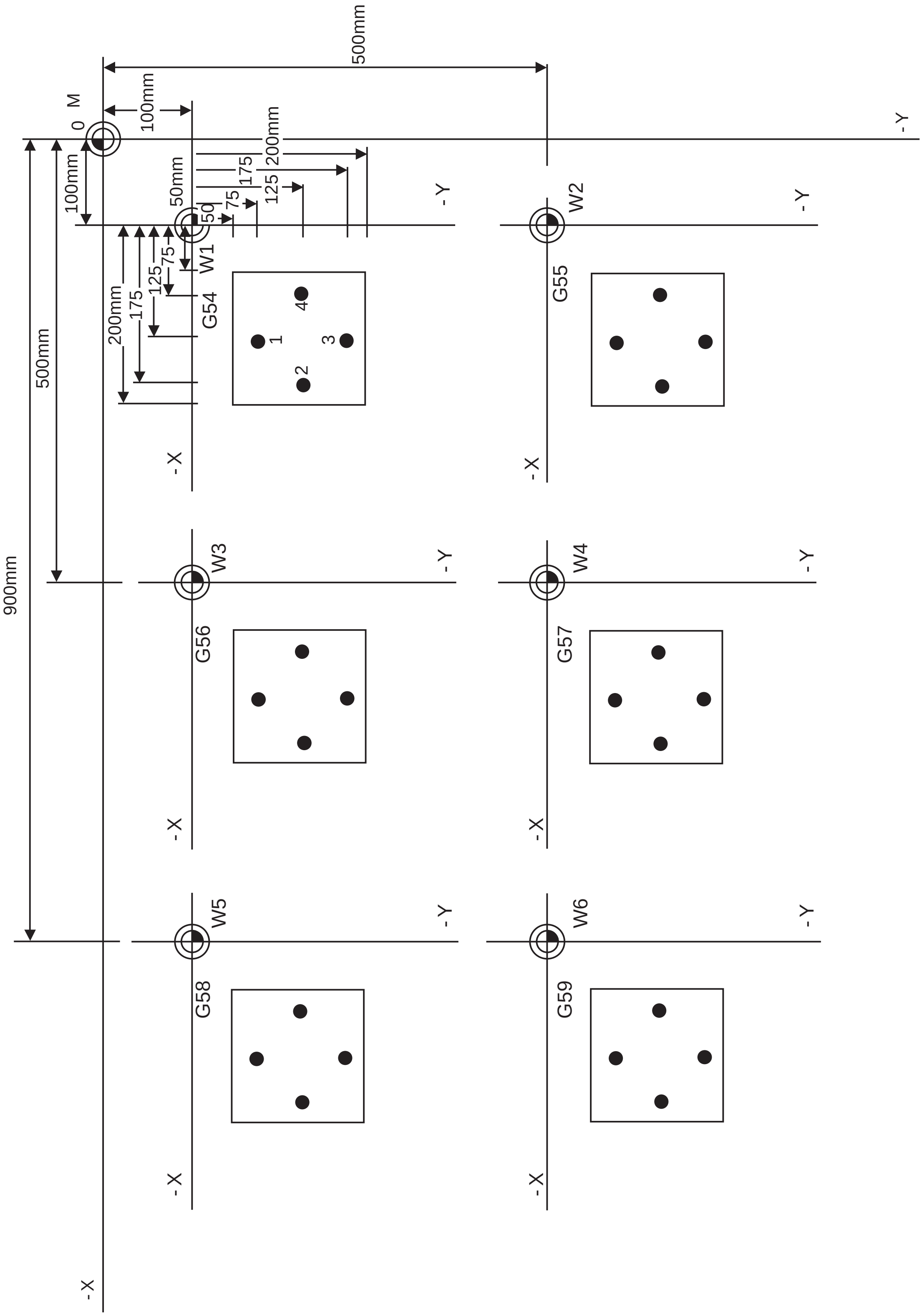
```
Workpiece 1 X=-100.000 Y=-100.000.....G54
Workpiece 2 X=-100.000 Y=-500.000.....G55
Workpiece 3 X=-500.000 Y=-100.000.....G56
Workpiece 4 X=-500.000 Y=-500.000.....G57
Workpiece 5 X=-900.000 Y=-100.000.....G58
Workpiece 6 X=-900.000 Y=-500.000.....G59
```

(2) Machining program (subprogram)

```
O100;
N1  G90 G0 G43 X-50. Y-50. Z-100. H10 ;            .....Positioning
N2  G01 X-200. F50 ;                               .....Surface cutting
        Y-200. ;                                   .....Surface cutting
        X-50. ;                                    .....Surface cutting
        Y-50. ;                                    .....Surface cutting
N3  G28 X0 Y0 Z0 ;
     :
     :
     :
N4  G98 G81 X-125. Y-75. Z-150. R-100. F40 ;       ..... Drilling 1
        X-175. Y-125. ;                            ..... Drilling 2
        X-125. Y-175. ;                            ..... Drilling 3
        X- 75. Y-125. ;                            ..... Drilling 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. ;                            ..... Tapping 2
        X-125. Y-175. ;                            ..... Tapping 3
        X- 75. Y-125. ;                            ..... Tapping 4
    G80 ;
    M99 ;
```

(3) Positioning program (main)

```
G28 X0 Y0 Z0 ;                         ..... When power is turned ON
N1  G90 G54 M98 P100 ;
N2      G55 M98 P100 ;
N3      G57 M98 P100 ;
N4      G56 M98 P100 ;
N5      G58 M98 P100 ;
N6      G59 M98 P100 ;
N7  G28 X0 Y0 Z0 ;
N8  M02 ;
%
```


500mm
100mm
M
0
200mm
175
125
75
50
50mm
100mm
500mm
900mm
W1
G54
W2
G55
W3
G56
W4
G57
W5
G58
W6
G59
1
2
3
4
-X
-Y

14.11 Local Coordinate System Setting ; G52

Function and purpose

The local coordinate systems can be set 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.

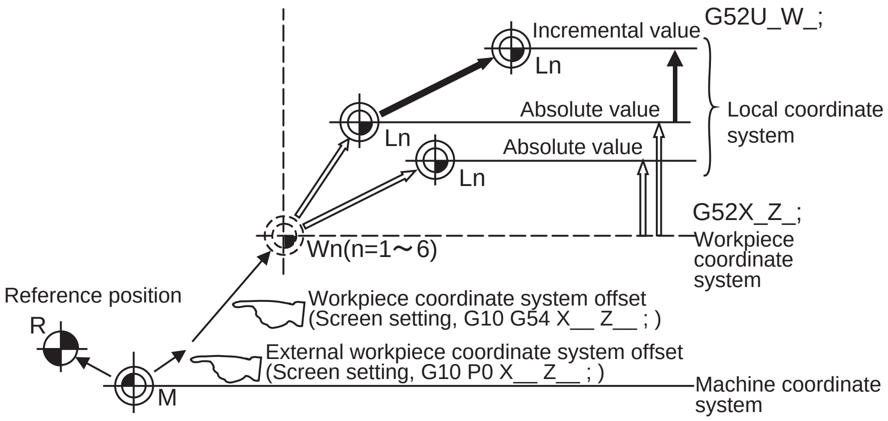
Command format

G54(G54 to G59) G52 X__ Y__ Z__ α__ ;

α	Additional axis

Detailed description

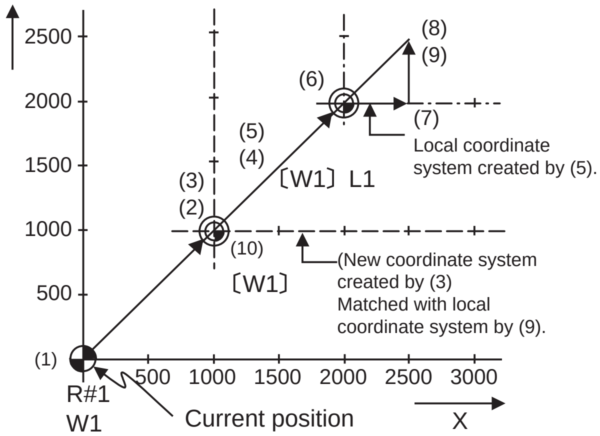
(1) The G52 command is valid until a new G52 command is issued, and the tool does not move. This command comes in handy for employing another coordinate system without changing the zero point positions of the workpiece coordinate systems (G54 to G59).
(2) The local coordinate system offset will be cleared by the dog-type manual reference (zero) point return or reference (zero) point return performed after the power has been switched ON.
(3) The local coordinate system is canceled by (G54 to G59) G52 X0 Y0 Z0 α 0;.
(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)

(1)	G28 X0 Y0 ;
(2)	G00 G90 X1. Y1. ;
(3)	G92 X0 Y0 ;
(4)	G00 X500 Y500 ;
(5)	G52 X1. Y1. ;
(6)	G00 X0 Y0 ;
(7)	G01 X500 F100 ;
(8)	Y500 ;
(9)	G52 X0 Y0 ;
(10)	G00 X0 Y0 ;

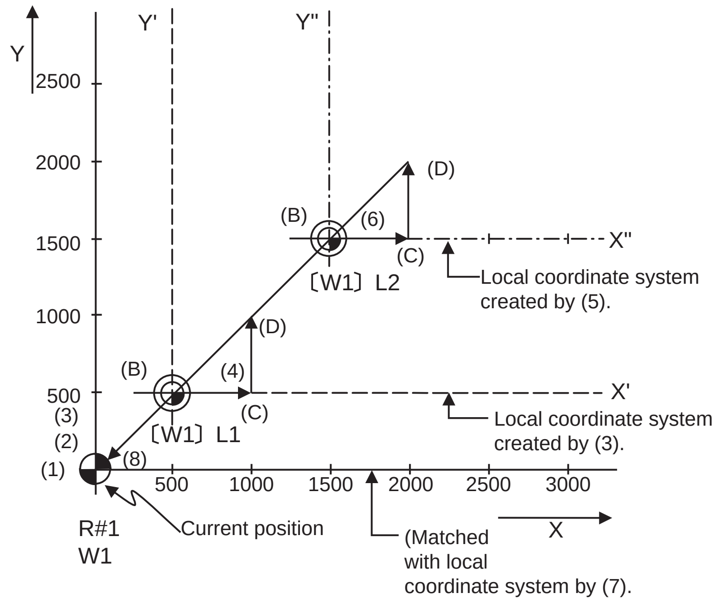


The local coordinate system is created by (5), canceled (9) and matched with the coordinate system for (3).

(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.)

(1)	G28 X0 Y0 ;
(2)	G92 X0 Y0 ;
(3)	G91 G52 X500 Y500 ;
(4)	M98 P100 ;
(5)	G52 X1. Y1. ;
(6)	M98 P100 ;
(7)	G52 X-1.5 Y-1.5 ;
(8)	G00 G90 X0 Y0 ;
	M02 ;
(A)	O100 ;
(B)	G90 G00 X0 Y0 ;
(C)	G01 X500 ;
(D)	Y500 ;
(E)	G91 ;
(F)	M99 ;



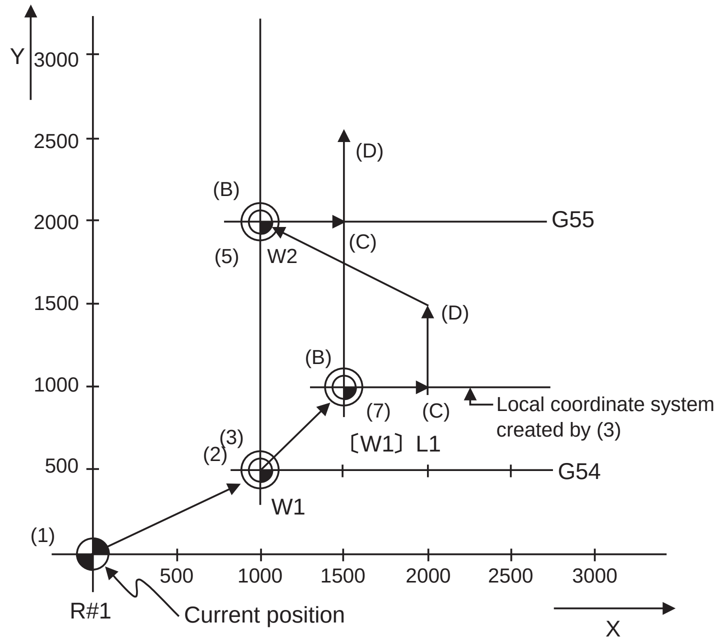
The local coordinate system X'Y' is created at the XY coordinate system (500,500) position by (3).
The local coordinate system X"Y" is created at the X'Y' coordinate system (1000,1000) position by (5).
The local coordinate system is created at the X"Y" coordinate system (-1500, -1500) position by (7). 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

```
(1)   G28 X0 Y0 ;
(2)   G00 G90 G54 X0 Y0 ;
(3)   G52 X500 Y500 ;
(4)   M98 P200 ;
(5)   G00 G90 G55 X0 Y0 ;
(6)   M98 P200 ;
(7)   G00 G90 G54 X0 Y0 ;
         :
      M02 ;
(A)   O200 ;
(B)   G00 X0 Y0 ;
(C)   G01 X500 F100 ;
(D)   Y500 ;
(E)   M99 ;
       %
```

	G54	G55
X	1000	1000
Y	500	2000

Workpiece coordinate system (parameter setting value)

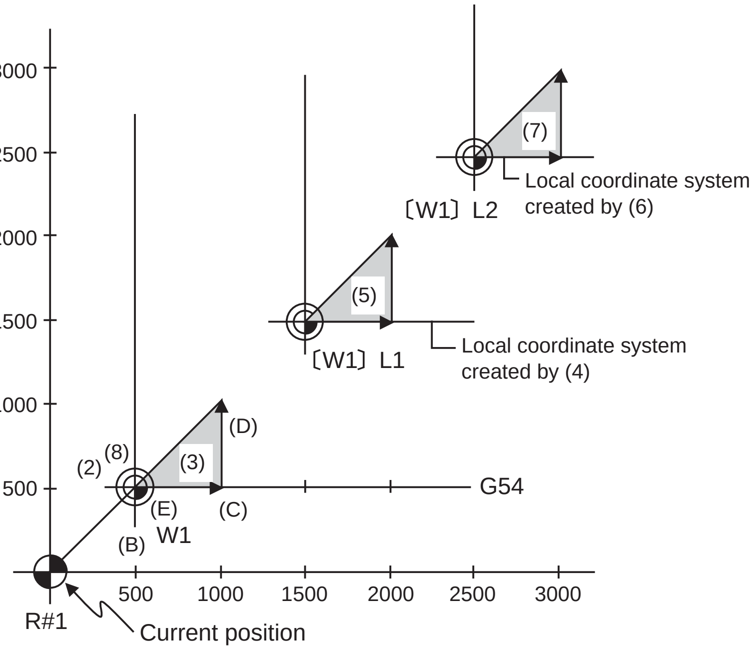


The local coordinate system is created at the G54 coordinate system (500,500) position by (3), but the local coordinate system is not created for the G55 coordinate system.
During the movement for (7), the axis moves to the G54 local coordinate system's reference position (zero point).
The local coordinate system is canceled by G90G54G52X0Y0;.

(Example 4) Combination of workpiece coordinate system G54 and multiple local coordinate systems

(1)	G28 X 0 Y0 ;	
(2)	G00 G90 G54 X0 Y0 ;	
(3)	M98 P300;	
(4)	G52 X1. Y1. ;	
(5)	M98 P300 ;	
(6)	G52 X2. Y2. ;	
(7)	M98 P300 ;	
(8)	G52 X0 Y0 ;	
	:	
	M02 ;	
(A)	O300 ;	
(B)	G00 X0 Y0 ;	
(C)	G01 X500 F100 ;	
(D)	Y500 ;	
(E)	X0 Y0 ;	
(F)	M99 ;	
	%	

	G54	
X	500	Workpiece coordinate offset (parameter setting value)
Y	500	



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 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 and 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 value 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.

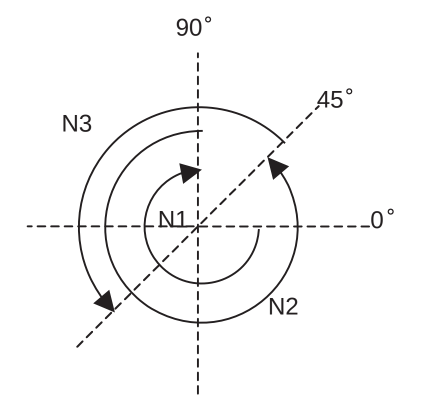
<table>
<tr><th rowspan="3"></th><th colspan="3">Rotary axis</th><th colspan="2" rowspan="3">Linear axis</th></tr>
<tr><th colspan="2">Rotating type rotary axis</th><th>Linear type rotary axis</th></tr>
<tr><th>Short-cut invalid</th><th>Short-cut valid</th><th>Workpiece coordinate position linear type</th></tr>
<tr><td>#8213 setting value</td><td>0</td><td>1</td><td colspan="2">2</td><td>-</td></tr>
<tr><td>Workpiece coordinate position</td><td colspan="2">Displayed in the range of 0° to 359.999°.</td><td colspan="3">Displayed in the range of 0° to ± 99999.999°.</td></tr>
<tr><td>Machine coordinate position/relative position</td><td colspan="3">Displayed in the range of 0° to 359.999°.</td><td colspan="2">Displayed in the range of 0° to ± 99999.999°.</td></tr>
<tr><td>ABS command</td><td>The incremental amount from the end point to the current position is divided by 360 degrees, and the axis moves by the remainder amount according to the sign.</td><td>Moves with a short-cut to the end point.</td><td colspan="3">In the same manner as the normal linear axis, it moves according to the sign by the amount obtained by subtracting the current position from the end point (without rounding up to 360 degrees).</td></tr>
<tr><td>INC command</td><td colspan="5">Moves in the direction of the commanded sign by the commanded incremental amount starting at the current position.</td></tr>
<tr><td rowspan="2">Reference position return</td><td colspan="5">Depends on the absolute command or the incremental command during the movement to the intermediate point.</td></tr>
<tr><td colspan="3">Returns with movement within 360 degrees.</td><td colspan="2">Moves and returns in the R point direction for the difference from the current position to the R point.</td></tr>
</table>

Operation example

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°.)

Rotary type (short-cut invalid)

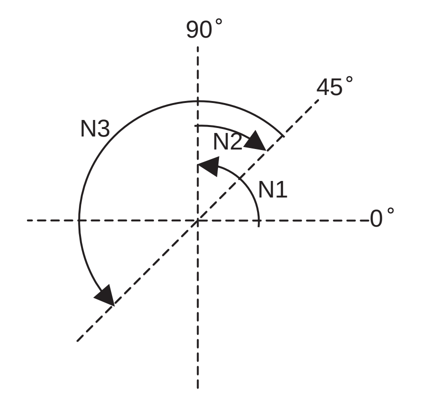
(1) The machine coordinate position, workpiece coordinate position and relative position are displayed in the range of 0° to 359.999°.
(2) For the absolute position command, the axis moves according to the sign by the remainder amount obtained by dividing by 360°.



Program	Workpiece	Machine
G28 C0.		
N1 G90 C-270.	90.000	90.000
N2 C405.	45.000	45.000
N3 G91 C180.	225.000	225.000

Rotary type (short-cut valid)

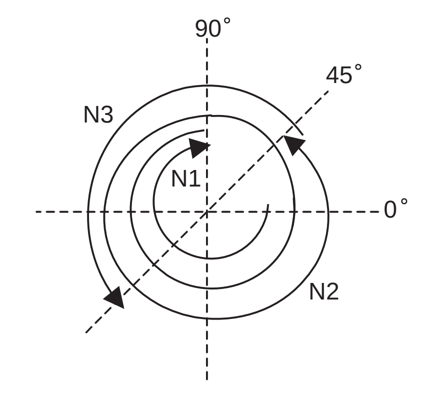
(1) The machine coordinate position, workpiece coordinate position and relative position are displayed in the range of 0° to 359.999°.
(2) For the absolute position command, the axis rotates to the direction having less amount of movement to the end point.



Program	Workpiece	Machine
G28 C0.		
N1 G90 C-270.	90.000	90.000
N2 C405.	45.000	45.000
N3 G91 C180.	225.000	225.000

Linear type (workpiece coordinate position linear type)

(1) 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°.
(2) The movement is the same as the linear axis.
(3) During reference position return, the axis moves in the same manner as the linear axis until the intermediate point is reached. The axis returns with a rotation within 360° from the intermediate point to the reference position.
(4) 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	Machine	Relative position
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
	After the power is turned ON again		
	Workpiece	Machine	
	225.000	225.000	

15

Measurement Support Functions

15.1 Automatic Tool Length Measurement ; G37

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 being 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 compensation 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.

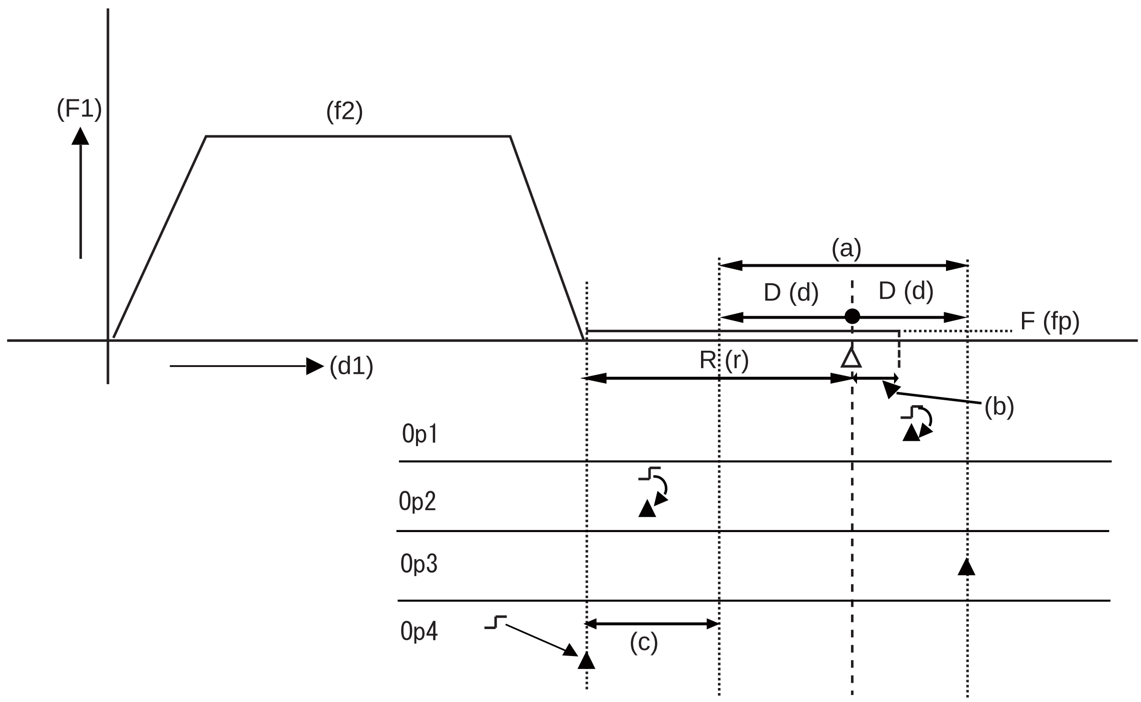
Command format

G37 Z__ R__ D__ F__ ; ... Automatic tool length measurement command

Z	Measuring axis address and coordinates of measurement position ------X,Y,Z,α (α 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_ or F_ is omitted, the value set in the parameter is used instead. <Parameter> ("AUTO TLM" on machining parameter screen) - #8004 SPEED (measuring feedrate): 0 to 60000 [mm/min] - #8005 ZONE r: 0 to 99999.999 [mm] - #8006 ZONE d: 0 to 99999.999 [mm]

Detailed description

(1) Operation with G37 command



Op1 : Normal completion as it is measurement within the allowable range.

Op2 : Alarm stop (P607) as it is outside of the measurement allowable range.

Op3 : Alarm stop (P607) as the sensor is not detected.

Op4 : Alarm stop (P607) as it is outside of the measurement allowable range. However if there is no (c) area, normal completion will occur.

(a) Measurement allowable range	(b) Compensation amount	(d1) Distance
(F1) Speed	(f2) Rapid traverse rate	(d) Measurement range
(r) Deceleration range		

△ Measuring position	▲ Stop point	Sensor output

(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.

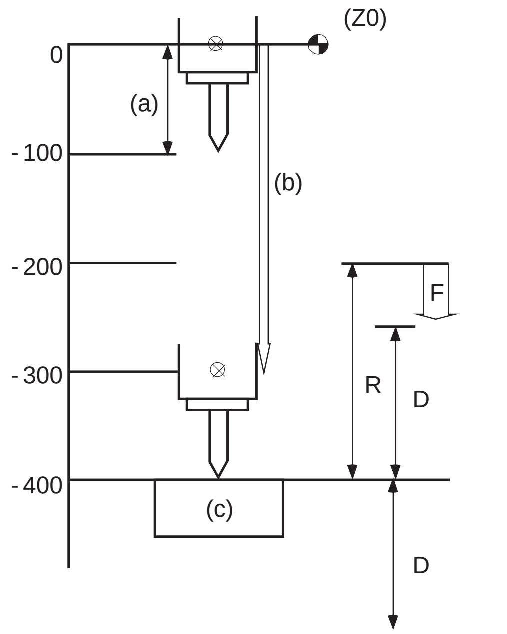
(4) An updated offset amount is valid unless it is assigned from the following Z axis (measurement axis) command of the G37 command.

(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.
Maximum measuring error [mm] = Measuring speed [mm/min] * 1/60 * 0.2 [ms]/1000

(6) The machine position coordinates at that point in time are read by sensor signal detection, and the machine will overtravel and stop at a position equivalent to the servo droop.
Maximum overtravel [mm] = Measuring speed [mm/min] * 1/60 * 1/Position loop gain [1/s]
The standard position loop gain is 33 (1/s).

Operation example

For new measurement



G28 Z0;
T01;
M06 T02;
G90 G00 G43 Z0 H01;
G37 Z-400 R200 D150 F1;

Coordinate value when reached at the measurement position = -300
-300-(-400)=100
0+100=100 H01=100

(a) Tool length
(b) Movement amount by tool length measurement
(c) Measuring device

(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.

Precautions

(1) Program error (P600) occurs if G37 is commanded when the automatic tool length measurement function is not provided.
(2) Program error (P604) will occur when no axis has been commanded in the G37 block or when two or more axes have been commanded.
(3) Program error (P605) will occur when the H code is commanded in the G37 block.
(4) Program error (P606) will occur when G43_H code is not commanded prior to the G37 block.
(5) Program error (P607) will occur when the sensor signal is input outside the allowable measuring range or when the sensor signal is 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 or the parameter setting data must meet the following conditions:
| Measurement point start point | > R command or parameter r > D command or parameter d
(8) When the D address and parameter d in (7) above are zero, the operation will be completed normally only when the commanded measurement point and sensor signal detection point coincide. Otherwise, program error (P607) will occur.
(9) When the R and D addresses as well as parameters r and d in (7) above are all zero, program error (P607) will occur 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) must be commanded together with the G43H_ command that designates the offset No.
G43 H_;
G37 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__ ;

X,Y,Z,α	Axis coordinate value; they are commanded as absolute or incremental values according to the G90/G91 modal when commanded. α is the additional axis.
F	Feedrate (mm/min)

Detailed description

(1) If Ff is commanded as the feedrate in the same block as the G31 command block, command feed f will apply; if not commanded, 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) Normally, the machine will not automatically accelerate or decelerate with the G31 block.
However, setting the base specification parameter "#21101 add01/bit3" to "1" allows the automatic acceleration/deceleration valid.
In such case, the acceleration/deceleration will apply following to the cutting feed acceleration/deceleration pattern set with the axis specification parameter "#2003 smgst".
Since the deceleration at skip signal input follows the cutting feed acceleration/deceleration pattern mentioned above, the coasting amount from the skip signal input to stop may be larger than the normal specifications (when automatic acceleration/deceleration is invalid)

(3) The stop condition (such as feed hold, stroke end) is also valid for the G31 block.

(4) With the normal specifications, override and dry run are invalid during execution of G31 block.
However, setting the base specification parameter "#21101 add01/bit3" to "1" allows the override and dry run.

(5) The G31 command is unmodal and it needs to be commanded each time.

(6) If the skip command is input at the start of the G31 command, the G31 command will be completed immediately.
When a skip signal has not been input until the completion of the G31 block, the G31 command will also be completed upon completion of the movement commands.

(7) When the G31 command is issued during tool radius compensation, the program error (P608) will occur.

(8) When there is no F command in the G31 command and the parameter speed is also zero, the program error (P603) will occur.

(9) With machine lock or with the Z axis cancel switch ON when only the Z axis is commanded, the skip signal will be ignored and execution will continue as far as the end of the block.

(10) Signal input contact can be selected, depending on the parameter "#1258 set30/bit0 skip I/F changeover" setting.
0: A contact (Skip operation will be performed as the signal turns ON.)
1: B contact (Skip operation will be performed as the signal turns OFF.)

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.

```
:
G90 G00  X-100.  ;
G31 X-200.  F60  ;        (Skip command)
#101=#5061 ;              Skip signal input coordinate position (workpiece coordinate system) is readout to #101.
:
```

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 between deceleration start and stop after responding to the skip signal is short, so the machine can be stopped precisely with a small coasting amount. The coasting amount can be calculated from the following formula.

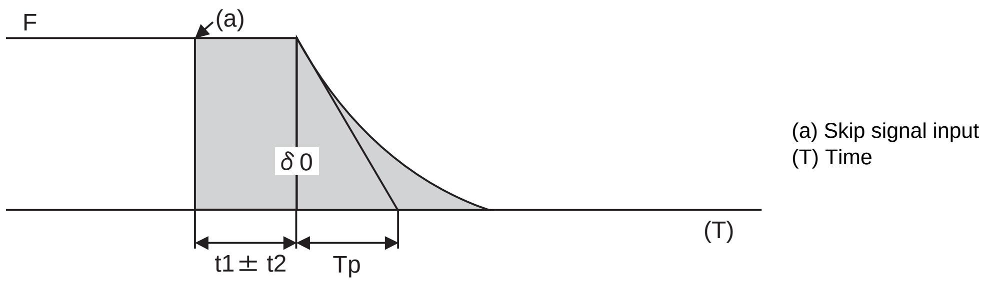
$$\delta 0 = \frac{F}{60} \times Tp + \frac{F}{60} \times (t1 \pm t2)$$

$$= \underbrace{\frac{F}{60} \times (Tp+t1)}_{\delta 1} \pm \underbrace{\frac{F}{60} \times t2}_{\delta 2}$$

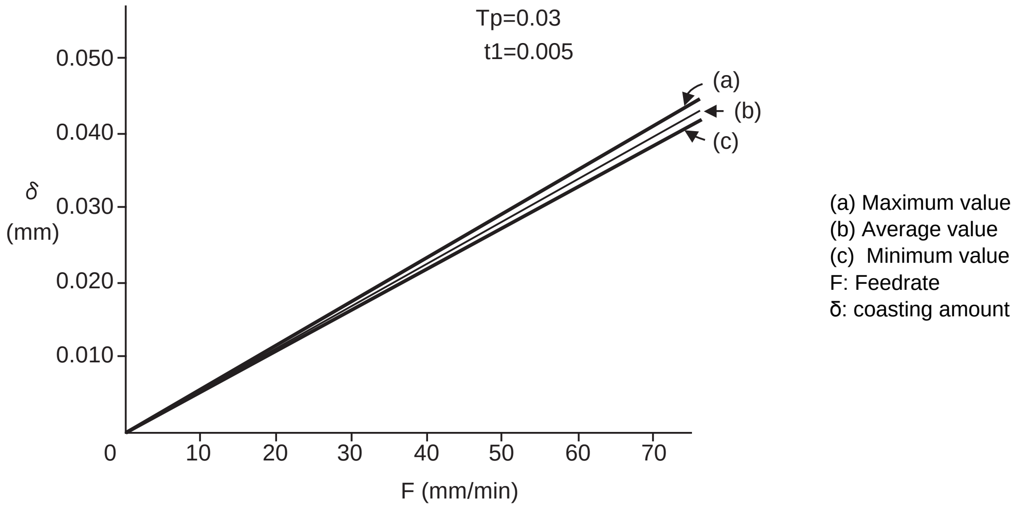
δ0	:	Coasting amount (mm)
F	:	G31 skip speed (mm/min)
Tp	:	Position loop time constant (s) = (position loop gain)$^{-1}$
t1	:	Response delay time (s) = (time taken from the detection to the arrival of the skip signal at the controller)
t2	:	Response error time 0.001 (s)

When G31 is used for calculation, the value calculated from the section indicated by δ1 in the above equation can be compensated, however, δ2 results in calculation error.

Stop pattern with skip signal input is shown below.



The relationship between the coasting amount and speed when Tp is 30ms and t1 is 5ms is shown in the following figure.



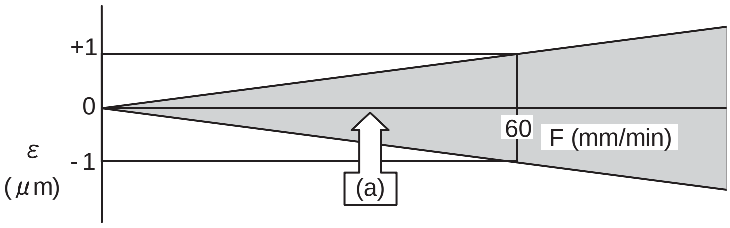
(Note) When the base specification parameter "#21101 add01/bit3" is set to "1", the automatic acceleration/deceleration becomes valid for the deceleration at skip signal input.
Thus, the coasting amount from the skip signal input to stop may be larger than when the automatic acceleration/deceleration is invalid.

Readout error of skip coordinates mm

(1) Skip signal input coordinate readout
The coasting amount based on the position loop time constant Tp and cutting feed time constant Ts is not included in the skip signal input coordinate values.
Therefore, the workpiece coordinate values applying when the skip signal is input can be readout within the error range in the following formula as the skip signal input coordinate values. However, coasting based on response delay time t1 results in a measurement error and so compensation must be provided.

$\varepsilon = \pm (F/60) * t\,2$

ε : Readout error
F : Feedrate
t2 : Response error time 0.001 (s)
(a) Measurement value



Readout error of skip signal input coordinates
Readout error with a 60mm/min feedrate is as shown below and the measurement value is within readout error range of ±1 μ m:

$\varepsilon = \pm (60/60) * 0.001 = \pm 0.001$ (mm)

(2) Readout of other coordinates
The readout coordinate values include the coasting amount. Therefore, when coordinate values at the time of skip signal input is required, reference should be made to the section on the G31 coasting amount to compensate the coordinate value. As in the case of (1), the coasting amount based on the delay error time t2 cannot be calculated, and this generates a measuring error.

Examples of compensating for coasting

(1) Compensating for skip signal input coordinates

```
:
G31 X100.F100 ;         Skip command
G04;                    Machine stop check
#101=#5061 ;            Skip signal input coordinate readout
#102=#110*#111/60 ;     Coasting based on response delay time
#105=#101-#102 ;        Skip signal input coordinates
:
```

#110 = Skip feedrate: #111 = Response delay time t1;

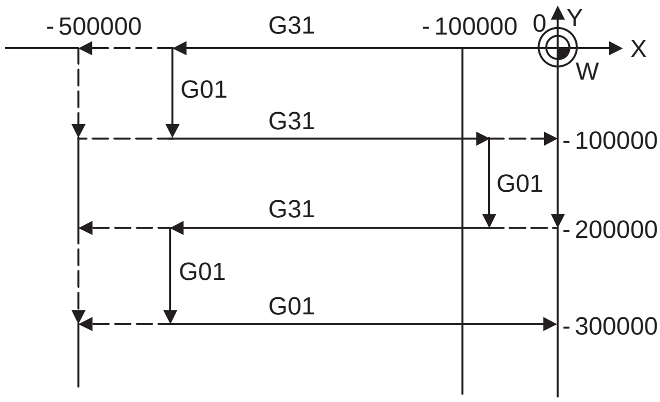
(2) Compensating for workpiece coordinates

```
:
G31 X100.F100 ;           Skip command
G04;                      Machine stop check
#101=#5061 ;              Skip signal input coordinate readout
#102=#110*#111/60 ;       Coasting based on response delay time
#103=#110*#112/60 ;       Coasting based on position loop time constant
#105=#101-#102-#103 ;     Skip signal input coordinates
:
```

#110 = Skiop feedrate; #111 = Response delay time t1; #112 = Position loop time constant Tp;

Operation example

```
G90  G00  X-100000 Y0;
     G31  X-500000 F100;
     G01  Y-100000;
     G31  X-0 F100;
          Y-200000;
     G31  X-50000 F100;
          Y-300000;
          X0
```



15.3 Multi-step Skip Function 1 ; 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 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 feedrates corresponding to the G31.1, G31.2, and G31.3 commands can be set by parameters.
(4) 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			
	4	3	2	1
1	-	-	-	○
2	-	-	○	-
3	-	-	○	○
4	-	○	-	-
5	-	○	-	○
6	-	○	○	-
7	-	○	○	○
8	○	-	-	-
9	○	-	-	○
10	○	-	○	-
11	○	-	○	○
12	○	○	-	-
13	○	○	-	○
14	○	○	○	-
15	○	○	○	○

(Skip when " ○ " signal is input.)

(5) Other commands work the same as the G31 (skip function) command.

Operation example

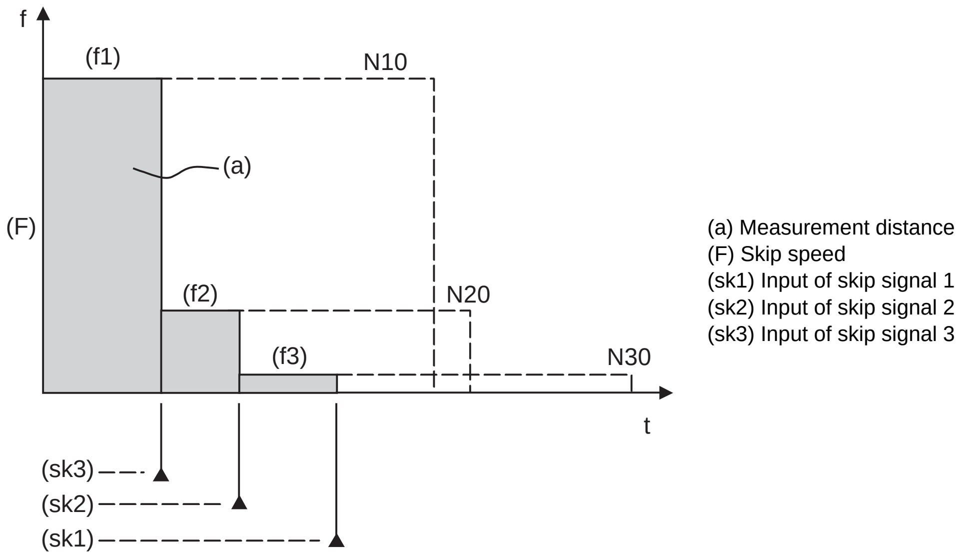
(1) The multi-step skip function enables the following control, thereby improving measurement accuracy and shortening the time required for measurement.

[Parameter settings]

Skip condition	Skip speed
G31.1 : 7	20.0mm/min (f1)
G31.2 : 3	5.0mm/min (f2)
G31.3 : 1	1.0mm/min (f3)

[Program example]

```
N10 G31.1 X200.0 ;
N20 G31.2 X40.0 ;
N30 G31.3 X1.0 ;
```



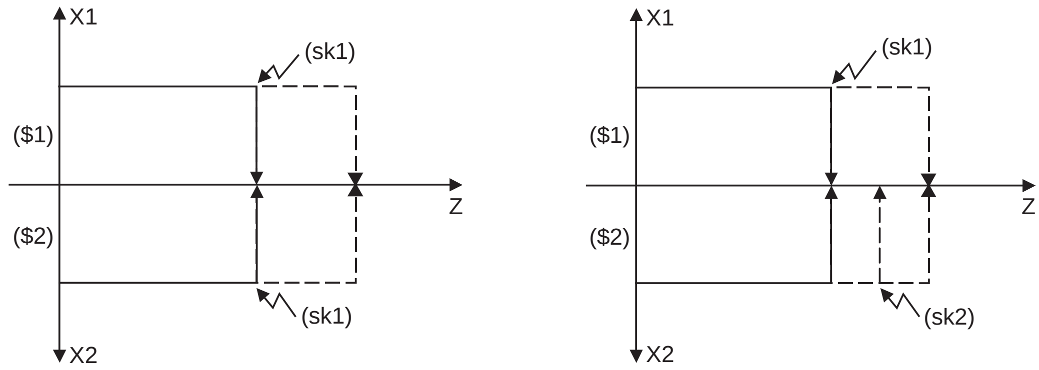
(Note 1) If skip signal 1 is input before skip signal 2 in the above operation, N20 is skipped at that point and N30 is also ignored.

(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 P

Function and purpose

During linear interpolation by the skip command (G31), operation can be skipped according to the conditions of the skip signal parameter Pp which indicates external skip signals 1 to 4.
If multi-step skip commands are issued simultaneously in different part systems as shown in the left figure, 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 as shown in the right figure. The skip operation is the same as a normal skip command (G31 without P command).



[Same skip signals input in both 1st and 2nd part systems]

[Different skip signals input in 1st and 2nd part systems]

($1) 1st part system ($2) 2nd part system
(sk1) Skip signal 1 (sk2) Skip signal 2

If the skip condition specified by the parameter "#1173 dwlskp" (indicating external skip signals 1 to 4) 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__ Y__ Z__ α__ P__ F__ ;

X Y Z α	Command format axis coordinate word and target coordinates
P	Skip signal command
F	Feedrate (mm/min)

Detailed description

(1) The skip speed is specified by command speed f. Note that the F modal is not updated.

(2) The skip signal is specified by skip signal command p. The command range of "p" is from 1 to 15. If outside the range is commanded, program error (P35) will occur.

Skip signal command P	Valid skip signal			
	4	**3**	**2**	**1**
1	-	-	-	○
2	-	-	○	-
3	-	-	○	○
4	-	○	-	-
5	-	○	-	○
6	-	○	○	-
7	-	○	○	○
8	○	-	-	-
9	○	-	-	○
10	○	-	○	-
11	○	-	○	○
12	○	○	-	-
13	○	○	-	○
14	○	○	○	-
15	○	○	○	○

(Skip when " ○ " signal is input.)

(3) The specified skip signal command is a logical sum of the skip signals.
(Example) G31 P5 F100 ;
Operation is skipped if skip signal 1 or 3 is input.

(4) If skip signal parameter Pp is not specified, it works as a skip function (G31), not as a multi-step skip function. If speed parameter Ff is not specified, the skip speed set by the parameter "#1174 skip_F" will apply.

[Relations between skip and multi-step skip]

Skip specifications	×		○	
	Condition	**Speed**	**Condition**	**Speed**
G31 X100 ; (Without P and F)	Program error (P601)		Skip 1	#1174 skip_F
G31 X100 P5 ; (Without F)	Program error (P602)		Command value	#1174 skip_F
G31 X100 F100 ; (Without P)	Program error (P601)		Skip 1	Command value
G31 X100 P5 F100 ;	Program error (P602)		Command value	Command value

(5) If skip specification is effective and P is specified as an axis address, skip signal parameter P will be given a priority. The axis address "P" will be ignored.
(Example) G31 P500 F100 ;
This is regarded as a skip signal. (The program error (P35) will occur.)

(6) Other than above, the same detailed description as "Skip function; G31" applies.

15.5 Programmable Current Limitation ; G10 L14 ;

Function and purpose

This function allows the current limit value of the NC 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 range: 1 to 300

Precautions

(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 with external signals 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 until the end, and then move to the next block with the droops still accumulated.

[Interlock mode]
During the occurrence of the droops, it enters to the internal interlock state and the next movement will not be carried out.
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) 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 300%. 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".

Appendix1

Order of G Function Command Priority

(Command in a separate block when possible)
Upper level: When commanded in the same block
○ : Indicates that both commands are executed simultaneously
Lower level: When commanded during each modal

Commanded G code	G Group						
	01 G00-G03, G33	02 G17-G19	03 G90, G91	05 G94, G95	06 G20, G21	07 G40-G42	08 G43, G44 G49
G00-G03 Positioning/ interpolation	G command commanded last is valid. Group 1 modal is updated	○ ○ Also possible during arc modal	○ ○	○ ○	○ ○	Arc and G41, G42 cause error P151 ○ Tool radius is compensated, and then axes move.	Arc and G43 to G49 cause error P70 ○ The G49 movement in the arc modal moves with G01
G04 Dwell	○ Group 1 modal is updated G04 is executed ○	○ ○	○ ○	○ ○	○ ○	G04 is executed G40 to G42 are ignored (Note) ○	G04 is executed G43 to G49 are ignored (Note) ○
G09 Exact stop check	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G10, G11 Parameter input by program	○ G10 is priority for axis No movement I, J, K rotation input ○	○ G10 is used for axis, so the selected plan axis will be the basic axis. ○	○ ○	○ ○	○ ○	G10 to G11 are executed G40 to G42 are ignored (Note) ○	G10 to G11 are executed G43 to G49 are ignored (Note) ○
G17 to G19 Plane selection	○ ○	○ G command commanded last is valid. ○	○ ○	○ ○	○ ○	○ Plane axis changeover during radius compensation causes error P112	○ ○
G20, G21 Inch/metric changeover	○ ○	○ ○	○ ○	○ ○	Possible in same block ○	○ ○	○ ○
G27 to G30 Reference position compare/ return	○ G00 to G03, and G33 modals are updated G27 to G30 are executed ○	○ ○	○ ○	○ ○	○ ○	G27 to G30 are executed G40 to G42 are ignored (Note) ○	G27 to G30 are executed G43 to G49 are ignored (Note) ○
G31 to G31.3 Skip	○ ○	○ ○	○ ○	○ ○	○ ○	Error:P608 Error:P608	○ ○
G33 Thread cutting	G command commanded last is valid. ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○

(Note) Normally, a program error (P45) occurs when these G codes are commanded in a same block. The program error can be avoided depending on a parameter setting but one of the G commands will be ignored. So attention must be paid when setting.

Commanded G code	G Group						
	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 Positioning/ interpolation	Group 1 command is executed Group 9 is canceled	○ ○	○ ○	○ ○	G66 to G67 are executed G00 to G03 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 (Note) ○	○ ○	G04 is executed Group 12 is changed ○	○ ○	G04 is executed G66-G67 are ignored (Note) ○	○ ○	G04 is executed G50.1 and G51.1 are ignored (Note) ○
G09 Exact stop check	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G10, G11 Parameter input by program	G10 to G11 are executed G73 to G89 are ignored (Note) ○	○ ○	○ 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	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G20, G21 Inch/metric changeover	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G27 to G30 Reference position compare/ return	○ ○	○ ○	○ ○	○ ○	G66 to G67 are executed G27 to G30 are ignored (Note) ○	○ ○	G27 to G30 are executed G50.1 and G51.1 are ignored (Note) ○
G31 to G31.3 Skip	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G33 Thread cutting	Group 1 command is executed Group 9 is canceled ○	○ ○	○ ○	○ ○	G66 to G67 are executed G33 modal is updated ○	○ ○	○ ○

Commanded G code	G Group						
	01 G00 to G03, G33	02 G17 to G19	03 G90, G91	05 G94, G95	06 G20, G21	07 G40-G42	08 G43, G44 G49
G37 Automatic tool length measurement	G37 is executed G00 to G03 and G33 are ignored ○	○ ○	○ ○	○ ○	○ ○	G37 is executed G40 to G42 are ignored (Note) ○	G37 is executed G43 to G49 are ignored (Note) ○
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. ○	○ ○
G43, G44, G49 Length compensation	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 (Note) ○	G52 is executed G43 to G49 are ignored (Note) ○
G53 Machine coordinate system	○ ○	○	○ ○	○	○ ○	G53 is executed G40 to G42 are ignored (Note) ○	G53 is executed G43 to G49 are ignored (Note) ○
G54 to G59 Workpiece coordinate system	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G60 Unidirectional positioning	○ ○	○ ○	○ ○	○ ○	○ ○	G60 is executed G40 to G42 are ignored (Note)	G60 is executed G43 to G49 are ignored (Note)
G61 to G64 Mode selection	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G65 Macro call	G65 is executed G00 to G03 and G33 modals are updated ○	○ ○	○ ○	○ ○	○ ○	○ ○	G65 is executed G43 to G49 modals are updated (Note) ○
G66 to G67 Macro call	G66 to G67 are executed G00 to G03 and G33 modals are updated ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ G66 to G67 are executed G43 to G49 modals are updated ○

Commanded G code	G Group						
	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
G37 Automatic tool length measurement		○ ○	○	○	G66 to G67 are executed G37 is ignored (Note) ○	○ ○	G37 is executed G50.1 and G51.1 are ignored (Note) ○
G40 to G42 Tool radius compensation	Error:P155 Error:P155	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G43, G44, G49 Length compensation	○ ○	○ ○	○ ○	○ ○	G66 to G67 are executed G43 to G49 modals are updated ○	○ ○	○ ○
G50.1 G51.1 Program mirror image	○ ○	○ ○	○ ○	○ ○	G66 to G67 are executed G50.1 and G51.1 are ignored ○	○ ○	G command commanded last is valid. ○
G52 Local coordinate system	G52 is executed G73 to G89 are ignored (Note)	○ ○	○ ○	○ ○	G52 is executed G66-G67 are ignored (Note) ○	○ ○	G52 is executed G50.1 and G51.1 are ignored (Note) ○
G53 Machine coordinate system		○ ○	○ ○	○ ○	G53 is executed G66-G67 are ignored (Note) ○	○ ○	G53 is executed G50.1 and G51.1 are invalid (Note) ○
G54 to G59 Workpiece coordinate system	○ ○	○ ○	G command commanded last is valid. ○	○ ○	G66 to G67 are executed G54 and G59 modals are updated ○	○ ○	○ ○
G60 Unidirectional positioning		○ ○	○ ○	○ ○	G60 is executed G66-G67 are ignored (Note)	○ ○	G60 is executed G50.1 and G51.1 are ignored
G61 to G64 Mode selection	○ ○	○ ○	○ ○	G command commanded last is valid. ○	○ ○	○ ○	○ ○
G65 Macro call	G65 is executed G73 to G89 are ignored (Note) ○	○ ○	○ ○	○ ○	Error ○	○ ○	G65 is executed G50.1 and G51.1 are ignored (Note) ○
G66 to G67 Macro call	G66 to G67 are executed G73 to G89 are ignored ○	○ ○	G66 to G67 are executed G54 and G59 modals are updated ○	○ ○	G command commanded last is valid.	○ ○	G66 to G67 are executed G50.1 and G51.1 are ignored ○

Appendix1 Order of G Function Command Priority

Commanded G code	G Group						
	01 G00 to G03 G33	02 G17 to G19	03 G90, G91	05 G94, G95	06 G20, G21	07 G40 to G42	08 G43, G44 G49
G73 to G89 Fixed cycles	G73 to G89 are canceled G01 to G33 modals are updated ○	○ ○	○ ○	○ ○	○ ○	Error:P155 Fixed 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	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○

Commanded G code	G Group						
	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
G73 to G89 Fixed cycles	G command commanded last is valid. ○	○ ○	○ ○	○ ○	G66 to G67 are executed G73 to G89 are ignored ○	○ ○	○ All axes become mirror center ○
G90, G91 Absolute value/ incremental value	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G92 Coordinate system setting	G92 is executed G73 to G89 are ignored ○	○ ○	○	○ ○	○	○ ○	○ Note that G92 is priority for axis ○
G94, G95 Synchronous/ asynchronous	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G96, G97 Constant surface speed control	○ ○	○ ○	○ ○	○ ○	○ ○	G command commanded last is valid. ○	○ ○
G98, G99 Initial point/ R point return	○ ○	G command commanded last is valid. ○	○ ○	○ ○	○ ○	○ ○	○ ○

Appendix 2

Program Errors

(Note) Program error messages are displayed in abbreviation on the screen.

P10 EXCS. AXIS. No.

Details

The number of axis addresses commanded in a block is exceeds the specifications.

Remedy

- Divide the alarm block command into two.
- Check the specifications.

P11 AXIS ADR. ERROR

Details

The axis address commanded by the program does not match any of the ones set by the parameter.

Remedy

- Correct the axis names in the program.

P20 DIVISION ERROR

Details

The issued axis command cannot be divided by the command unit.

Remedy

- Correct the program.

P30 PARITY H

Details

The number of holes per character on the paper tape is even for EIA code and odd for ISO code.

Remedy

- Check the paper tape.
- Check the tape puncher and tape reader.

P31 PARITY V

Details

The number of characters per block on the paper tape is odd.

Remedy

- Make the number of characters per block on the paper tape even.
- Set the parameter parity V selection OFF.

P32 ADDRESS. ERROR

Details

An address not listed in the specifications has been used.

Remedy

- Correct the program address.
- Correct the parameter settings.
- Check the specifications.

P33 FORMAT ERROR

Details

The command format in the program is not correct.

Remedy

- Correct the program.

P34 G-CODE ERROR

Details

The commanded G code is not in the specifications.
An illegal G code was commanded during the coordinate rotation command (G68).

Remedy

- Correct the G code address in the program.

Details

G51.2 or G50.2 was commanded when "#1501 polyax (Rotational tool axis number)" was set to "0".
G51.2 or G50.2 was commanded when the tool axis was set to the linear axis ("#1017 rot (Rotational axis)" is set to"0").

Remedy

- Correct the parameter settings.

P35 CMD-VALUE OVER

Details

The setting range for the addresses has been exceeded.
The program coordinates overflowed because commands to the linear type rotary axis accumulated in one direction.

Remedy

- Correct the program.

P36 PROGRAM END ERR

Details

"EOR" has been read during memory mode.

Remedy

- Enter the M02 and M30 command at the end of the program.
- Enter the M99 command at the end of the subprogram.

P37 PROG. No. ZERO

Details

"0" has been specified for program or sequence No.

Remedy

- Designate program Nos. within a range from 1 to 99999999.
- Designate sequence Nos. within a range from 1 to 99999.
- Add M02 or M03 to the end of the program running in FTP operation.

P39 NO SPEC ERR

Details

- A non-specified G code was commanded.
- The selected operation mode is out of specifications.

Remedy

- Check the specifications.

P45 G-CODE COMB.

Details

The combination of G codes in a block is inappropriate.
A part of unmodal G codes and modal G codes cannot be commanded in a same block.

Remedy

Correct the combination of G codes.
Separate the incompatible G codes into different blocks.

P60 OVER CMP. LENG.

Details

The commanded movement distance is excessive (over 2^{31}).

Remedy

- Correct the command range for the axis address.

P62 F-CMD. NOTHING

Details

- No feed rate command has been issued.
- There is no F command in the cylindrical interpolation or polar coordinate interpolation immediately after the G95 mode is commanded.

Remedy

- 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.
- Specify F with a thread lead command.

P65 No G05P3 SPEC

Details

Remedy

- Check whether the specifications are provided for the high-speed mode III.

P70 ARC ERROR

Details

- There is an error in the arc start and end points as well as in the arc center.
- The difference of the involute curve through the start point and the end point is large.
- When arc was commanded, one of the two axes configuring the arc plane was a scaling valid axis.

Remedy

- Correct the numerical values of the addresses that specify the start and end points, arc center as well as the radius in the program.
- Correct the "+" and "-" directions of the address numerical values.
- Check for the scaling valid axis.

P71 ARC CENTER

Details

- An arc center cannot be obtained in R-specified circular interpolation.
- A curvature center of the involute curve cannot be obtained.

Remedy

- Correct the numerical values of the addresses in the program.
- Correct the start and end points if they are inside of the base circle for involute interpolation. When carrying out tool radius compensation, make sure that the start and end points after compensation will not be inside of the base circle for involute interpolation.
- Correct the start and end points if they are at an even distance from the center of the base circle for involute interpolation.

P72 NO HELICAL SPEC

Details

A helical command has been issued though it is out of specifications.

Remedy

- Check whether the specifications are provided for the helical cutting.
- An Axis 3 command has been issued by the circular interpolation command. If there is no helical specification, move the linear axis to the next block.

P90 NO THREAD SPEC

Details

A thread cutting command was issued though it is out of specifications.

Remedy

- Check the specifications.

P93 SCREW PITCH ERR

Details

An illegal thread lead (thread pitch) was specified at the thread cutting command.

Remedy

- Correct the thread lead for the thread cutting command.

P111 PLANE CHG (CR)

Details

Plane selection commands (G17, G18, G19) were issued during a coordinate rotation (G68) was being commanded.

Remedy

- Always command G69 (coordinate rotation cancel) after the G68 command, and then issue a plane selection command.

P112 PLANE CHG (CC)

Details

- 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.
- Plane selection commands were issued after completing nose R compensation commands when there were no further axis movement commands after G40, and compensation has not been cancelled.

Remedy

- Issue plane selection commands after completing (axis movement commands issued after G40 cancel command) tool radius compensation and nose R compensation commands.

P113 ILLEGAL PLANE

Details

The circular command axis does not correspond to the selected plane.

Remedy

- Select a correct plane before issuing a circular command.

P122 NO AUTO C-OVR

Details

An auto corner override command (G62) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Delete the G62 command from the program.

P130 2nd AUX. ADDR

Details

The 2nd miscellaneous function address, commanded in the program, differs from the address set in the parameters.

Remedy

- Correct the 2nd miscellaneous function address in the program.

P131 NO G96 SPEC

Details

A constant surface speed control command (G96) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Issue a rotation speed command (G97) instead of the constant surface speed control command (G96).

P132 SPINDLE S = 0

Details

No spindle rotation speed command has been issued.

Remedy

- Correct the program.

P133 G96 P-No. ERR

Details

The illegal No. was specified for the constant surface speed control axis.

Remedy

- Correct the parameter settings and program that specify the constant surface speed control axis.

P134 G96 Clamp Err.

Details

The constant surface speed control command (G96) was issued without commanding the spindle speed clamp (G92/G50).

Remedy

Press the reset key and carry out the remedy below.
- Check the program.
- Issue the G92/G50 command before the G96 command.
- Command the constant surface speed cancel (G97) to switch to the rotation speed command.

P150 NO C-CMP SPEC

Details

- Tool radius compensation commands (G41 and G42) were issued though they are out of specifications.
- Nose R compensation commands (G41, G42, and G46) were issued though they are out of specifications.

Remedy

- Check the specifications.

P151 G2, 3 CMP. ERR

Details

A compensation command (G40, G41, G42, G43, G44, or G46) has been issued in the arc modal (G02 or G03).

Remedy

- Issue the linear command (G01) or rapid traverse command (G00) in the compensation command block or cancel block.
  (Set the modal to linear interpolation.)

P152 I.S.P NOTHING

Details

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.

Remedy

- Correct the program.

P153 I.F ERROR

Details

An interference error has occurred while the tool radius compensation command (G41 or G42) or nose R compensation command (G41, G42 or G46) was being executed.

Remedy

- Correct the program.

P155 F-CYC ERR (CC)

Details

A fixed cycle command has been issued in the radius compensation mode.

Remedy

- Issue a radius compensation cancel command (G40) to cancel the radius compensation mode that has been applied since the fixed cycle command was issued.

P156 BOUND DIRECT

Details

A shift vector with undefined compensation direction was found at the start of G46 nose R compensation.

Remedy

- Change the vector to that which has the defined compensation direction.
- Change the tool to that which has a different tip point No.

P157 SIDE REVERSED

Details

During G46 nose R compensation, the compensation direction is reversed.

Remedy

- Change the G command to that which allows the reversed compensation direction (G00, G28, G30, G33, or G53).
- Change the tool to that which has a different tip point No.
- Enable "#8106 G46 NO REV-ERR".

P158 ILLEGAL TIP P.

Details

An illegal tip point No. (other than 1 to 8) was found during G46 nose R compensation.

Remedy

- Correct the tip point No.

P170 NO CORR. NO.

Details

No compensation No. (DOO, TOO or HOO) command was given when the radius compensation (G41, G42, G43 or G46) command was issued. Otherwise, the compensation No. is larger than the number of sets in the specifications.

Remedy

- Add the compensation No. command to the compensation command block.
- Check the number of sets for the tool compensation Nos. and correct the compensation No. command to be within the number of sets.

P171 NO G10 SPEC

Details

Compensation data input by program (G10) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P172 G10 L-No. ERR

Details

An address of G10 command is not correct.

Remedy

- Correct the address L No. of the G10 command.

P173 G10 P-No. ERR

Details

The compensation No. at the G10 command is not within the permitted number of sets in the specifications.

Remedy

- Check the number of sets for the tool compensation Nos. and correct the address P designation to be within the number of sets.

P174 NO G11 SPEC

Details

Compensation data input by program cancel (G11) was commanded though there is no specification of compensation data input by program.

Remedy

- Check the specifications.

P177 LIFE COUNT ACT

Details

Registration of tool life management data with G10 was attempted when the "usage data count valid" signal was ON.

Remedy

- The tool life management data cannot be registered during the usage data count. Turn the "usage data count valid" signal OFF.

P178 LIFE DATA OVER

Details

The number of registration groups, total number of registered tools or the number of registrations per group exceeded the range in the specifications.

Remedy

- Correct the number of registrations.

P179 GROUP NO. ILL.

Details

- A duplicate group No. was found at the registration of the tool life management data with G10.
- A group No. that was not registered was designated during the T****99 command.
- An M code command, which must be issued as a single command, coexists in the same block as that of another M code command.
- The M code commands set in the same group exist in the same block.

Remedy

- Register the tool life data once for one group: commanding with a duplicate group No. is not allowed.
- Correct to the group No.

P180 NO BORING CYC.

Details

A fixed cycle command (G72 - G89) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Correct the program.

P181 NO S-CMD (TAP)

Details

Spindle rotation speed (S) has not been commanded in synchronous tapping.

Remedy

- Command the spindle rotation speed (S) in synchronous tapping.
- 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.

P182 SYN TAP ERROR

Details

- Connection to the main spindle unit was not established.
- The synchronous tapping was attempted with the spindle not serially connected under the multiple-spindle control I.

Remedy

- Check connection to the main spindle.
- Check that the main spindle encoder exists.
- Set 1 to the parameter #3024 (sout).

P183 PTC/THD No.

Details

The pitch or number of threads has not been commanded in the tap cycle of a fixed cycle for drilling command.

Remedy

- Specify the pitch data and the number of threads by F or E command.

P184 NO PTC/THD CMD

Details

- The pitch or the number of threads per inch is illegal in the tap cycle of the fixed cycle for drilling command.
- The pitch is too small for the spindle rotation speed.
- The thread number is too large for the spindle rotation speed.

Remedy

- Correct the pitch or the number of threads per inch.

P185 NO SYN TAP SPC

Details

Synchronous tapping cycle (G84/G74) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P187 Tap SP clamp 0

Details

The external spindle speed clamp signal was turned ON without setting the tapping spindle's external spindle speed when commanding the synchronous tapping.

Remedy

- Set the external spindle speed clamp speed parameter.
- Turn the external spindle speed clamp signal OFF.

P190 NO CUTTING CYC

Details

A lathe cutting cycle command was issued though it is out of specifications.

Remedy

- Check the specification.
- Delete the lathe cutting cycle command.

P191 TAPER LENG ERR

Details

In the lathe cutting cycle, the specified length of taper section is illegal.

Remedy

- Set the smaller radius value than the axis travel amount in the lathe cycle command.

P192 CHAMFERING ERR

Details

Chamfering in the thread cutting cycle is illegal.

Remedy

- Set a chamfering amount not exceeding the cycle.

P200 NO MRC CYC SPC

Details

The compound type fixed cycle for turning machining I (G70 to G73) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P201 PROG. ERR (MRC)

Details

- The subprogram, called with a compound type fixed cycle for turning machining I command, has at least one of the following commands: reference position return command (G27, G28, G29, G30); thread cutting (G33, G34); fixed cycle skip-function (G31, G31.n).
- An arc command was found in the first movement block of the finished shape program in compound type fixed cycle for turning machining I.

Remedy

- Delete G27, G28, G29, G30, G31, G33, G34, and fixed cycle G codes from the subprogram called with the compound type fixed cycle for turning machining I commands (G70 to G73).
- Delete G02 and G03 from the first movement block of the finished shape program in compound type fixed cycle for turning machining I.

P202 BLOCK OVR (MRC)

Details

The number of blocks in the shape program of the compound type fixed cycle for turning machining I is over 50 or 200 (the maximum number differs according to the model).

Remedy

- Set a 50/200 or less value for the number of blocks in the shape program called by the compound type fixed cycle for turning machining I commands (G70 to G73). (The maximum number differs according to the model).

P203 CONF. ERR (MRC)

Details

A proper shape will not obtained by executing the shape program for the compound type fixed cycle for turning machining I (G70 to G73).

Remedy

- Correct the shape program for the compound type fixed cycle for turning machining I (G70 to G73).

P204 VALUE ERR (MRC)

Details

A command value of the compound type fixed cycle for turning machining (G70 to G76) is illegal.

Remedy

- Correct the command value of the compound type fixed cycle for turning machining (G70 to G76).

P210 NO PAT CYC SPC

Details

A compound type fixed cycle for turning machining II (G74 to G76) command was commanded though it is out of specifications.

Remedy

- Check the specifications.

P220 NO SPECIAL CYC

Details

There are no special fixed cycle specifications.

Remedy

- Check the specifications.

P221 NO HOLE (S-CYC)

Details

"0" has been specified for the number of holes in special fixed cycle mode.

Remedy

- Correct the program.

P222 G36 ANGLE ERR

Details

A G36 command specifies "0" for angle intervals.

Remedy

- Correct the program.

P223 G12 G13 R ERR

Details

The radius value specified with a G12 or G13 command is below the compensation amount.

Remedy

- Correct the program.

P224 NO G12, G13 SPC

Details

There are no circular cutting specifications.

Remedy

- Check the specifications.

P230 NESTING OVER

Details

Over 8 times of subprogram calls have been done in succession from a subprogram.

- A M198 command was found in the program in the data server.
- The program in the IC card has been called more than once (the program in the IC card can be called only once during nested).

Remedy

- Correct the program so that the number of subprogram calls does not exceed 8 times.

P231 NO N-NUMBER

Details

The sequence No., commanded at the return from the subprogram or by GOTO in the subprogram call, was not set.

Remedy

- Specify the sequence Nos. in the call block of the subprogram.

P232 NO PROGRAM No.

Details

- The machining program has not been found when the machining program is called.
- The file name of the program registered in IC card is not corresponding to O No.

Remedy

- Enter the machining program.
- Check the subprogram storage destination parameters.
- Ensure that the external device (including IC card) that contains the file is mounted.

P241 NO VARl NUMBER

Details

The variable No. commanded is out of the range specified in the specifications.

Remedy

- Check the specifications.
- Correct the program variable No.

P242 EQL. SYM. MSSG.

Details

The "=" sign has not been commanded when a variable is defined.

Remedy

- Designate the "=" sign in the variable definition of the program.

P243 VARIABLE ERR.

Details

An invalid variable has been specified in the left or right side of an operation expression.

Remedy

- Correct the program.

P260 NO COOD-RT SPC

Details

A coordinate rotation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P270 NO MACRO SPEC

Details

A macro specification was commanded though it is out of specifications.

Remedy

- Check the specifications.

P271 NO MACRO INT.

Details

A macro interruption command has been issued though it is out of specifications.

Remedy

- Check the specifications.

P272 MACRO ILL.

Details

An executable statement and a macro statement exist together in the same block.

Remedy

- Place the executable statement and macro statement in separate blocks in the program.

P273 MACRO OVERCALL

Details

The number of macro call nests exceeded the limit imposed by the specifications.

Remedy

- Correct the program so that the macro calls do not exceed the limit imposed by the specifications.

P275 MACRO ARG. EX.

Details

The number of argument sets in the macro call argument type II has exceeded the limit.

Remedy

- Correct the program.

P276 CALL CANCEL

Details

A G67 command was issued though it was not during the G66 command modal.

Remedy

- Correct the program.
- Issue G66 command before G67 command, which is a call cancel command.

P277 MACRO ALM MESG

Details

An alarm command has been issued in #3000.

Remedy

- Refer to the operator messages on the diagnosis screen.
- Refer to the instruction manual issued by the machine tool builder.

P280 EXC. [,]

Details

Over five times have the parentheses "[" or "]" been used in a single block.

Remedy

- Correct the program so that the number of "[" or "]" is five or less.

P281 [,] ILLEGAL

Details

A single block does not have the same number of commanded parentheses "[" as that of "]".

Remedy

- Correct the program so that "[" and "]" parentheses are paired up properly.

P282 CALC. IMPOSS.

Details

The arithmetic formula is incorrect.

Remedy

- Correct the formula in the program.

P283 DIVIDE BY ZERO

Details

The denominator of the division is zero.

Remedy

- Correct the program so that the denominator for division in the formula is not zero.

P290 IF SNT. ERROR

Details

There is an error in the "IF[<conditional>]GOTO(" statement.

Remedy

- Correct the program.

P291 WHILE SNT. ERR

Details

There is an error in the "WHILE[<conditional>]DO(-END(" statement.

Remedy

- Correct the program.

P292 SETVN SNT. ERR

Details

There is an error in the "SETVN(" statement when the variable name setting was made.

Remedy

- Correct the program.
- The number of characters in the variable name of the SETVN statement must be 7 or less.

P293 DO-END EXCESS

Details

The number of DO-END nesting levels in the "WHILE[<conditional>]DO(-END(" statement has exceeded 27.

Remedy

- Correct the program so that the nesting levels of the DO-END statement does not exceed 27.

P294 DO-END MMC.

Details

The DOs and ENDs are not paired off properly.

Remedy

- Correct the program so that the DOs and ENDs are paired off properly.

P295 WHILE/GOTO TPE

Details

There is a WHILE or GOTO statement on the tape during FTP operation.

Remedy

- Apply memory mode operation instead of FTP operation that does not allow the execution of the program with a WHILE or GOTO statement.

P296 NO ADR (MACRO)

Details

A required address has not been specified in the user macro.

Remedy

- Correct the program.

P297 ADR-A ERR.

Details

The user macro does not use address A as a variable.

Remedy

- Correct the program.

P298 PTR OP (MACRO)

Details

User macro G200, G201, or G202 was specified during tape or MDI mode.

Remedy

- Correct the program.

P300 VAR. NAME ERROR

Details

The variable names have not been commanded properly.

Remedy

- Correct the variable names in the program.

P301 VAR. NAME DUPLI

Details

A duplicate variable name was found.

Remedy

- Correct the program so that no duplicate name exists.

P360 NO PROG.MIRR.

Details

A mirror image (G50.1 or G51.1) command has been issued though the programmable mirror image specifications are not provided.

Remedy

- Check the specifications.

P380 NO CORNER R/C

Details

The corner R/C was issued though it is out of specifications.

Remedy

- Check the specifications.
- Delete the corner chamfering/corner rounding command in the program.

P381 NO ARC R/C SPC

Details

Corner chamfering II or corner rounding II was commanded in the arc interpolation block though it is out of specifications.

Remedy

- Check the specifications.

P382 CORNER NO MOVE

Details

The block next to corner chamfering/ corner rounding is not a travel command.

Remedy

- Replace the block succeeding the corner chamfering/ corner rounding command by G01 command.

P383 CORNER SHORT

Details

The travel distance in the corner chamfering/corner rounding command was shorter than the value in the corner chamfering/corner rounding command.

Remedy

- Set the smaller value for the corner chamfering/corner rounding than the travel distance.

P384 CORNER SHORT

Details

The travel distance in the following block in the corner chamfering/corner rounding command was shorter than the value in the corner chamfering/corner rounding command.

Remedy

- Set the smaller value for the corner chamfering/corner rounding than the travel distance in the following block.

P385 G0 G33 IN CONR

Details

A block with corner chamfering/corner rounding was given during G00 or G33 modal.

Remedy

- Correct the program.

P390 NO GEOMETRIC

Details

A geometric command was issued though it is out of specifications.

Remedy

- Check the specifications.

P391 NO GEOMETRIC 2

Details

There are no geometric IB specifications.

Remedy

- Check the specifications.

P392 LES AGL (GEOMT)

Details

The angular difference between the geometric line and line is 1° or less.

Remedy

- Correct the geometric angle.

P393 INC ERR (GEOMT)

Details

The second geometric block has a command with an incremental value.

Remedy

- Issue a command with an absolute value in the second geometric block.

P394 NO G01 (GEOMT)

Details

The second geometric block contains no linear command.

Remedy

- Issue the G01 command.

P395 NO ADRS (GEOMT)

Details

The geometric format is invalid.

Remedy

- Correct the program.

P396 PL CHG. (GEOMT)

Details

A plane switching command was issued during geometric command processing.

Remedy

- Complete the plane switching command before geometric command processing.

P397 ARC ERR (GEOMT)

Details

In geometric IB, the circular arc end point does not contact or cross the next block start point.

Remedy

- Correct the geometric circular arc command and the preceding and following commands.

P398 NO GEOMETRIC1B

Details

A geometric command was issued though the geometric IB specifications are not provided.

Remedy

- Check the specifications.

P420 NO PARAM IN

Details

Parameter input by program (G10) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P421 PRAM. IN ERROR

Details

- The specified parameter No. or set data is illegal.
- An illegal G command address was input in parameter input mode.
- A parameter input command was issued during fixed cycle modal or nose R compensation.
- G10L50, G10L70, G11 were not commanded in independent blocks.

Remedy

- Correct the program.

P430 AXIS NOT RET.

Details

- A command was issued to move an axis, which has not returned to the reference position, away from that reference position.
- A command was issued to an axis removal axis.

Remedy

- Execute reference position return manually.
- Disable the axis removal on the axis for which the command was issued.

P431 NO 2ndREF. SPC

Details

A command for second, third or fourth reference position return was issued though there are no such command specifications.

Remedy

- Check the specifications.

P434 COLLATION ERR

Details

One of the axes did not return to the reference position when the reference position check command (G27) was executed.

Remedy

- Correct the program.

P435 G27/M ERROR

Details

An M command was issued simultaneously in the G27 command block.

Remedy

- Place the M code command, which cannot be issued in a G27 command block, in separate block from G27 command block.

P436 G29/M ERROR

Details

An M command was issued simultaneously in the G29 command block.

Remedy

- Place the M code command, which cannot be issued in a G29 command block, in separate block from G29 command block.

P438 NOT USE (G52)

Details

A local coordinate system command was issued during execution of the G54.1 command.

Remedy

- Correct the program.

P450 NO CHUCK BARR.

Details

The chuck barrier on command (G22) was specified although the chuck barrier is out of specifications.

Remedy

- Check the specifications.

P460 TAPE I/O ERROR

Details

An error has occurred in the tape reader. Otherwise an error has occurred in the printer during macro printing.

Remedy

- Check the power and cable of the connected devices.
- Correct the I/O device parameters.

P461 FILE I/O ERROR

Details

- A file of the machining program cannot be read.

Remedy

- In memory mode, the programs stored in memory may have been destroyed. Output all of the programs and tool data and then format the system.

P501 Cross (G110) impossible

Details

Mixed control (cross axis control) command (G110) was issued in the following modes.

- During nose R compensation mode
- During pole coordinate interpolation mode
- During cylindrical interpolation mode
- During balance cut mode
- During fixed cycle machining mode
- During facing turret mirror image
- During constant surface speed control mode
- During hobbing mode
- During axis name switch

Remedy

- Correct the program.

P503 Illegal G110 axis

Details

- The commanded axis does not exist.
- The mixed control (cross axis control) (G110) was commanded to the axis for which the mixed control (cross axis control) is disabled.
- The number of axes included in the mixed control (cross axis control) (G110) command is exceeding the maximum number of axes per part system.

Remedy

- Correct the program.

P600 NO AUTO TLM.

Details

An automatic tool length measurement command (G37) was issued though it is out of specifications.

Remedy

- Check the specifications.

P601 NO SKIP SPEC.

Details

A skip command (G31) was issued though it is out of specifications.

Remedy

- Check the specifications.

P602 NO MULTI SKIP

Details

A multiple skip command (G31.1, G31.2 or G31.3) was issued though it is out of specifications.

Remedy

- Check the specifications.

P603 SKIP SPEED 0

Details

The skip speed is "0".

Remedy

- Specify the skip speed.

P604 TLM ILL. AXIS command

Details

No axis was specified in the automatic tool length measurement block. Otherwise, two or more axes were specified.

Remedy

- Specify only one axis.

P605 T-CMD IN BLOCK

Details

The T code is in the same block as the automatic tool length measurement block.

Remedy

- Specify the T code before the automatic tool length measurement block.

P606 NO T-CMD BEFOR

Details

The T code was not yet specified in automatic tool length measurement.

Remedy

- Specify the T code before the automatic tool length measurement block.

P607 TLM ILL. SIGNL

Details

The measurement position arrival signal turned ON before the area specified by the D command or "#8006 ZONE d". Otherwise, the signal remained OFF to the end.

Remedy

- Correct the program.

P608 SKIP ERROR (CC)

Details

A skip command was issued during radius compensation processing.

Remedy

- Issue a radius compensation cancel (G40) command or remove the skip command.

P609 NO PLC SKIP

Details

PLC skip has been commanded (L to G31) while PLC skip is out of specifications.

Remedy

- Check the specifications.

P610 ILLEGAL PARA.

Details

- G114.1 was commanded when the spindle synchronization with PLC I/F command was selected.
- Spindle synchronization was commanded to a spindle that is not connected serially.

Remedy

- Check the program.
- Check the argument of G114.1 command.
- Check the state of spindle connection.

P990 PREPRO S/W ERR

Details

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.

Remedy

- Delete some or all of the combinations of commands that require pre-reading.

Index

M

N

O

P

R

S

T

U

V

W

Z

Revision History

Date of revision	Manual No.	Revision details
Nov. 2006	IB(NA)1500269-A	First edition created.
Jan. 2007	IB(NA)1500269-B	Second edition created.
Jun. 2009	IB(NA)1500269-C	Third edition created. Corrections are made corresponding to C70 S/W version B2. Following chapter is added. - 13.18 Chopping Following chapter is revised. - 13.9 Parameter Input by Program; G10, G11 Following chapter is deleted - 13.6.7 External output commands - Appendix 1. Parameter Input by Program N No. Correspondence Table Mistakes were corrected.
Jul. 2010	IB(NA)1500269-D	Fourth edition created. - Reviewed "Precautions for Safety". - Corrected the items below. - 10.5 Constant Surface Speed Control; G96, G97 - 10.6 Spindle Clamp Speed Setting; G92 - Corrected the mistakes.
Mar. 2011	IB(NA)1500269-E	Fifth edition created. Corrections are made corresponding to C70 S/W version C5. Following chapter is added. -15.5 Programmable Current Limitation Following chapters are revised. - 13.5.4 Types of Variables - 13.5.5 Operation Commands - Apendix1 Order of G Function Command Priority Structure of the following chapters are changed. - 3 Program Formats ("Data Formats" in previous version) - 6.1 Positioning (Rapid Traverse) ;G00 - 13.1 Fixed cycles Corrected the mistakes.

Global Service Network

AMERICA

MITSUBISHI ELECTRIC AUTOMATION INC. (AMERICA FA CENTER)

Central Region Service Center
500 CORPORATE WOODS PARKWAY, VERNON HILLS, ILLINOIS 60061, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

Michigan Service Satellite
ALLEGAN, MICHIGAN 49010, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-876-6535

Ohio Service Satellite
LIMA, OHIO 45804, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

Minnesota Service Satellite
RICHFIELD, MINNESOTA 55423, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

West Region Service Center
5665 PLAZA DRIVE, CYPRESS, CALIFORNIA 90630, U.S.A.
TEL: +1-714-220-4796 / FAX: +1-714-229-3818

East Region Service Center
200 COTTONTAIL LANE SOMERSET, NEW JERSEY 08873, U.S.A.
TEL: +1-732-560-4500 / FAX: +1-732-560-4531

Pennsylvania Service Satellite
ERIE, PENNSYLVANIA 16510, U.S.A.
TEL: +1-814-897-7820 / FAX: +1-814-987-7820

South Region Service Center
2810 PREMIERE PARKWAY SUITE 400, DULUTH, GEORGIA 30097, U.S.A.
TEL: +1-678-258-4500 / FAX: +1-678-258-4519

Texas Service Satellites
GRAPEVINE, TEXAS 76051, U.S.A.
TEL: +1-817-251-7468 / FAX: +1-817-416-5000
FRIENDSWOOD, TEXAS 77546, U.S.A.
TEL: +1-832-573-0787 / FAX: +1-678-573-8290

Florida Service Satellite
SATELLITE BEACH, FLORIDA 32937, U.S.A.
TEL: +1-321-610-4436 / FAX: +1-321-610-4437

Canada Region Service Center
4299 14TH AVENUE MARKHAM, ONTARIO L3R OJ2, CANADA
TEL: +1-905-475-7728 / FAX: +1-905-475-7935

Mexico Region Service Center
MARIANO ESCOBEDO 69 TLALNEPANTLA, 54030 EDO. DE MEXICO
TEL: +52-55-9171-7662 / FAX: +52-55-9171-7649

Monterrey Service Satellite
MONTERREY, N.L., 64720, MEXICO
TEL: +52-81-8365-4171 / FAX: +52-81-8365-4171

Brazil Region Service Center
ACESSO JOSE SARTORELLI, KM 2.1 CEP 18550-000, BOITUVA-SP, BRAZIL
TEL: +55-15-3363-9900 / FAX: +55-15-3363-9911

Brazil Service Satellites
PORTO ALEGRE AND CAXIAS DO SUL BRAZIL
TEL: +55-15-3363-9927
SANTA CATARINA AND PARANA STATES
TEL: +55-15-3363-9927

EUROPE

MITSUBISHI ELECTRIC EUROPE B.V. (EUROPE FA CENTER)
GOTHAER STRASSE 10, 40880 RATINGEN, GERMANY
TEL: +49-2102-486-0 / FAX: +49-2102-486-5910

Germany Service Center
KURZE STRASSE. 40, 70794 FILDERSTADT-BONLANDEN, GERMANY
TEL: + 49-711-3270-010 / FAX: +49-711-3270-0141

France Service Center
25, BOULEVARD DES BOUVETS, 92741 NANTERRE CEDEX FRANCE
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

France (Lyon) Service Satellite
120, ALLEE JACQUES MONOD 69800 SAINT PRIEST FRANCE
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

Italy Service Center
VIALE COLLEONI 7-PALAZZO SIRIO CENTRO DIREZIONALE COLLEONI,
20041 AGRATE BRIANZA MILANO ITALY
TEL: +39-039-60531-342 / FAX: +39-039-6053-206

Italy (Padova) Service Satellite
VIA SAVELLI 24 - 35129 PADOVA ITALY
TEL: +39-039-60531-342 / FAX: +39-039-6053-206

U.K. Service Center
TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.
TEL: +44-1707-27-6100 / FAX: +44-1707-27-8992

Spain Service Center
CTRA. DE RUBI, 76-80-APDO. 420
08190 SAINT CUGAT DEL VALLES, BARCELONA SPAIN
TEL: +34-935-65-2236 / FAX: +34-935-89-1579

Poland Service Center
UL.KRAKOWSKA 50, 32-083 BALICE, POLAND
TEL: +48-12-630-4700 / FAX: +48-12-630-4727

Poland (Wroclaw) Service Center
UL KOBIERZYCKA 23, 52-315 WROCLAW, POLAND
TEL: +48-71-333-77-53 / FAX: +48-71-333-77-53

Turkey Service Center
BAYRAKTAR BULVARI, NUTUK SOKAK NO.5, YUKARI DUDULLU
ISTANBUL, TURKEY
TEL: +90-216-526-3990 / FAX: +90-216-526-3995

Czech Republic Service Center
TECHNOLOGICKA 374/6,708 00 OSTRAVA-PUSTKOVEC, CZECH REPUBLIC
TEL: +420-59-5691-185 / FAX: +420-59-5691-199

Russia Service Center
213, B.NOVODMITROVSKAYA STR., 14/2, 127015 MOSCOW, RUSSIA
TEL: +7-495-748-0191 / FAX: +7-495-748-0192

Sweden Service Center
STRANDKULLEN, 718 91 FROVI, SWEDEN
TEL: +46-581-700-20 / FAX: +46-581-700-75

Bulgaria Service Center
4 A. LYAPCHEV BOUL., 1756 - SOFIA, BULGARIA
TEL: +359-2-8176000 / FAX: +359-2-9744061

Ukraine (Kharkov) Service Center
APTEKARSKIY PEREULOK 9-A, OFFICE 3, 61001 KHARKOV, UKRAINE
TEL: +38-57-732-7744 / FAX: +38-57-731-8721

Ukraine (Kiev) Service Center
4-B, M. RASKOVOYI STR., 02660 KIEV, UKRAINE
TEL: +38-044-494-3355 / FAX: +38-044-494-3366

Belarus Service Center
703, OKTYABRSKAYA STR., 16/5, 220030 MINSK, BELARUS
TEL: +375-17-210-4626 / FAX: +375-17-227-5830

South Africa Service Center
P.O. BOX 9234, EDLEEN, KEMPTON PARK GAUTENG, 1625 SOUTH AFRICA
TEL: +27-11-394-8512 / FAX: +27-11-394-8513

ASEAN

MITSUBISHI ELECTRIC ASIA PTE. LTD. (ASEAN FA CENTER)

Singapore Service Center
307 ALEXANDRA ROAD #05-01/02 MITSUBISHI ELECTRIC BUILDING SINGAPORE 159943
TEL: +65-6473-2308 / FAX: +65-6476-7439

Indonesia Service Center
THE PLAZZA OFFICE TOWER, 28TH FLOOR JL.M.H. THAMRIN KAV.28-30, JAKARTA, INDONESIA
TEL: +62-21-2992-2333 / FAX: +62-21-2992-2555

Malaysia (KL) Service Center
60, JALAN USJ 10 /1B 47620 UEP SUBANG JAYA SELANGOR DARUL EHSAN, MALAYSIA
TEL: +60-3-5631-7605 / FAX: +60-3-5631-7636

Malaysia (Johor Baru) Service Center
NO. 16, JALAN SHAH BANDAR 1, TAMAN UNGKU TUN AMINAH, 81300 SKUDAI, JOHOR MALAYSIA
TEL: +60-7-557-8218 / FAX: +60-7-557-3404

Vietnam Service Center-1
ROOM 1004, 1005, FLOOR 10, 255 TRAN HUNG DAO CO GIANG WARD, DIST. 1, HCMC, VIETNAM
TEL: +84-8-3838-6931 / FAX: +84-8-3838-6932

Vietnam Service Center-2
LOT G10 - AREA 4 - HIEP BINH CHANH WARD - THU DUC DISTRICT - HCMC, VIETNAM
TEL: +84-8-2240-3587 / FAX: +84-8-3726-7968

Vietnam (Hanoi) Service Center
5FL, 59 - XA DAN STR., DONG DA DIST., HN, VIETNAM
TEL: +84-4-3573-7646 / FAX: +84-4-3573-7650

Philippines Service Center
UNIT NO.411, ALABAMG CORPORATE CENTER KM 25. WEST SERVICE ROAD
SOUTH SUPERHIGHWAY, ALABAMG MUNTINLUPA METRO MANILA, PHILIPPINES 1771
TEL: +63-2-807-2416 / FAX: +63-2-807-2417

MITSUBISHI ELECTRIC AUTOMATION (THAILAND) CO., LTD. (THAILAND FA CENTER)

BANG-CHAN INDUSTRIAL ESTATE NO.111 SOI SERITHAI 54
T.KANNAYAO, A.KANNAYAO, BANGKOK 10230, THAILAND
TEL: +66-2906-8255 / FAX: +66-2906-3239

Thailand Service Center
898/19,20,21,22 S.V. CITY BUILDING OFFICE TOWER 1, FLOOR 7
RAMA III RD., BANGPONGPANG, YANNAWA, BANGKOK 10120, THAILAND
TEL: +66-2-682-6522 / FAX: +66-2-682-9750

INDIA

MITSUBISHI ELECTRIC ASIA PVT. LTD.

FIRST & SECOND FLOOR, AVR BASE, MUNICIPAL NO.BC-308,
HENNURE BANASWADI ROAD, HRBR RING ROAD, BANGALORE-560 043, INDIA
TEL: +91-80-4020-1600 / FAX: +91-80-4020-1699

India (Pune) Service Center
EL-3, J BLOCK, M.I.D.C., BHOSARI PUNE 411026, INDIA
TEL: +91-20-2710-2000 / FAX: +91-20-2710-2185

India (Bangalore) Service Center
S 615, 6TH FLOOR, MANIPAL CENTER, BANGALORE 560001, INDIA
TEL: +91-80-509-2119 / FAX: +91-80-532-0480

India (Gurgaon) Service Center
UNIT NUMBER 107, 1ST FLOOR, SEVA CORPORATE PARK, MG ROAD, NEAR IFFCO CHOWK,
GURGAON, HARYANA INDIA
TEL: +91-124-4982901 / FAX: +91-124-4982900

OCEANIA

MITSUBISHI ELECTRIC AUSTRALIA LTD.

Oceania Service Center
348 VICTORIA ROAD, RYDALMERE, N.S.W. 2116 AUSTRALIA
TEL: +61-2-9684-7269 / FAX: +61-2-9684-7245

CHINA

MITSUBISHI ELECTRIC AUTOMATION (CHINA) LTD. (CHINA FA CENTER)

China (Shanghai) Service Center
4/F., ZHI FU PLAZA, NO. 80 XIN CHANG ROAD, HUANG PU DISTRICT,
SHANGHAI 200003, CHINA
TEL: +86-21-2322-3030 / FAX: +86-21-2308-2830
China (Ningbo) Service Dealer
China (Wuxi) Service Dealer
China (Jinan) Service Dealer
China (Wuhan) Service Satellite

China (Beijing) Service Center
9/F, OFFICE TOWER 1, HENDERSON CENTER, 18 JIANGUOMENNEI DAJIE,
DONGCHENG DISTRICT, BEIJING 100005, CHINA
TEL: +86-10-6518-8830 / FAX: +86-10-6518-3907
China (Beijing) Service Dealer

China (Tianjin) Service Center
B-2 801/802, YOUYI BUILDING, NO.50 YOUYI ROAD, HEXI DISTRICT,
TIANJIN 300061, CHINA
TEL: +86-22-2813-1015 / FAX: +86-22-2813-1017
China (Shenyang) Service Satellite
China (Changchun) Service Satellite

China (Chengdu) Service Center
ROOM 407-408, OFFICE TOWER AT SHANGRI-LA CENTER, NO. 9 BINJIANG DONG ROAD,
JINJIANG DISTRICT, CHENGDU, SICHUAN 610021, CHINA
TEL: +86-28-8446-8030 / FAX: +86-28-8446-8630

China (Shenzhen) Service Center
ROOM 2512-2516, 25/F., GREAT CHINA INTERNATIONAL EXCHANGE SQUARE, JINTIAN RD.S.,
FUTIAN DISTRICT, SHENZHEN 518034, CHINA
TEL: +86-755-2399-8272 / FAX: +86-755-8218-4776
China (Xiamen) Service Dealer
China (Dongguan) Service Dealer

KOREA

MITSUBISHI ELECTRIC AUTOMATION KOREA CO., LTD. (KOREA FA CENTER)

Korea Service Center
1480-6, GAYANG-DONG, GANGSEO-GU SEOUL 157-200, KOREA
TEL: +82-2-3660-9602 / FAX: +82-2-3664-8668

Korea Taegu Service Satellite
603 CRYSTAL BUILDING 1666, SANBYEOK-DONG, BUK-KU, DAEGU, 702-010, KOREA
TEL: +82-53-604-6047 / FAX: +82-53-604-6049

TAIWAN

MITSUBISHI ELECTRIC TAIWAN CO., LTD. (TAIWAN FA CENTER)

Taiwan (Taichung) 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

Taiwan (Taipei) Service Center
3RD. FLOOR, NO.122 WUKUNG 2ND RD., WU-KU HSIANG, TAIPEI HSIEN, TAIWAN R.O.C.
TEL: +886-2-2299-2205 / FAX: +886-2-2298-1909

Taiwan (Tainan) Service Center
2F(C),1-1, CHUNGHWA-RD., YONGKANG CITY, TAINAN HSIEN, TAIWAN R.O.C.
TEL: +886-6-313-9600 / FAX: +886-6-313-7713

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.

MITSUBISHI CNC

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING,2-7-3 MARUNOUCHI,CHIYODA-KU,TOKYO 100-8310,JAPAN

MODEL	C70
MODEL CODE	100—019
Manual No.	IB-1500269

IB(NA)1500269(ENG)-E(1103)MEE
Made in JAPAN