

Changes for the Better

MITSUBISHI CNC

**Programming Manual (Machining Center System)
E70 Series**








Introduction

This manual is a guide for using the MITSUBISHI CNC E70 Series.

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.**

General precautions




- (1) Refer to the following documents for details on handling
MITSUBISHI CNC E70 Series Instruction Manual IB-1501186


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".









<p> DANGER</p> <p>When the user may be subject to imminent fatalities or major injuries if handling is mistaken.</p>
<p> WARNING</p> <p>When the user may be subject to fatalities or major injuries if handling is mistaken.</p>
<p> CAUTION</p> <p>When the user may be subject to injuries or when physical damage may occur if handling is mistaken.</p>

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 following signs indicate prohibition and compulsory.

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

The meaning of each pictorial sign 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

For Safe Use

Mitsubishi CNC is designed and manufactured solely for applications to machine tools to be used for industrial purposes.



Do not use this product in any applications other than those specified above, especially those which are substantially influential on the public interest or which are expected to have significant influence on human lives or properties.

 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 compensation amount, workpiece compensation 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 compensation amount is changed during automatic operation (including during single block stop), it will be validated from the next block or blocks onwards.

3. Items related to programming

-  The commands with "no value after G" will be handled as "G00".
-  ";" "EOB" and "%" "EOR" are 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!

Trademarks

MELDAS, MELSEC, EZSocket, EZMotion, iQ Platform, MELSOFT, GOT, CC-Link, CC-Link/LT and CC-Link IE are either trademarks or registered trademarks of Mitsubishi Electric Corporation in Japan and/or other countries.

Ethernet is a registered trademark of Xerox Corporation in the United States and/or other countries.

Microsoft® and Windows® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.

CompactFlash and CF are either trademarks or registered trademarks of SanDisk Corporation in the United States and/or other countries.

UNIX is a registered trademark of The Open Group in the United States and/or other countries.

Intel® and Pentium® are either trademarks or registered trademarks of Intel Corporation in the United States and/or other countries.

Other company and product names that appear in this manual are trademarks or registered trademarks of the respective companies.

本製品の取扱いについて

(日本語/Japanese)

本製品は工業用(クラス A)電磁環境適合機器です。販売者あるいは使用者はこの点に注意し、住商業環境以外での使用をお願いいたします。

Handling of our product

(English)

This is a class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

본 제품의 취급에 대해서

(한국어/Korean)

이 기기는 업무용(A급) 전자파적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며 가정외의 지역에서 사용하는 것을 목적으로 합니다.

CONTENTS

1 Control Axes	1
1.1 Coordinate Words and Control Axes	2
1.2 Coordinate Systems and Coordinate Zero Point Symbols	3
2 Least Command Increments	5
2.1 Input Setting Unit	6
2.2 Input Command Increment Tenfold	7
3 Program Formats	9
3.1 Program Format	10
3.2 File Format	14
3.3 Optional Block Skip	16
3.3.1 Optional Block Skip; /	16
3.4 G code	18
3.4.1 Modal, unmodal	18
3.4.2 G Code Lists	18
3.5 Precautions Before Starting Machining	21
4 Pre-read Buffers	23
4.1 Pre-read Buffers	24
5 Position Commands	25
5.1 Position Command Methods ; G90,G91	26
5.2 Inch/Metric Conversion ; G20,G21	28
5.3 Decimal Point Input	29
6 Interpolation Functions	33
6.1 Positioning (Rapid Traverse) ; G00	34
6.2 Linear Interpolation ; G01	41
6.3 Circular Interpolation ; G02,G03	43
6.4 R Specification Circular Interpolation ; G02,G03	48
6.5 Plane Selection ; G17,G18,G19	50
6.6 Thread Cutting	52
6.6.1 Constant Lead Thread Cutting ; G33	52
6.6.2 Inch Thread Cutting ; G33	56
6.7 Helical Interpolation ; G17 to G19, G02, G03	58
6.8 Unidirectional Positioning ; G60	64
7 Feed Functions	65
7.1 Rapid Traverse Rate	66
7.2 Cutting Feedrate	67
7.3 F1-digit Feed	68
7.4 Feed Per Minute/Feed Per Revolution (Asynchronous Feed/Synchronous Feed) ; G94,G95	71
7.5 Feedrate Designation and Effects on Control Axes	74
7.6 Rapid Traverse Constant Inclination Acceleration/Deceleration	79
7.7 Cutting Feed Constant Inclination Acceleration/Deceleration	84
7.8 Exact Stop Check ; G09	91
7.9 Exact Stop Check Mode ; G61	95
7.10 Deceleration Check	96
7.10.1 G1 -> G0 Deceleration Check	98
7.10.2 G1 -> G1 Deceleration Check	99
7.11 Automatic Corner Override ; G62	100
7.12 Tapping Mode ; G63	106
7.13 Cutting Mode ; G64	107
8 Dwell	109

8.1 Dwell (Time Designation) ; G04	110
9 Miscellaneous Functions	113
9.1 Miscellaneous Functions (M8-digits)	114
9.2 Secondary Miscellaneous Functions (A8-digits, B8-digits or C8-digits)	116
10 Spindle Functions	117
10.1 Spindle Functions.....	118
10.2 Constant Surface Speed Control ; G96,G97.....	119
10.3 Spindle Clamp Speed Setting ; G92	121
10.4 Spindle/C Axis Control	123
11 Tool Functions (T command)	127
11.1 Tool Functions (T8-digit BCD)	128
12 Tool Compensation Functions	129
12.1 Tool Compensation.....	130
12.2 Tool Length Compensation/Cancel ; G43,G44/G49	134
12.3 Tool Radius Compensation ; G38,G39/G40/G41,G42.....	138
12.3.1 Tool Radius Compensation Operation.....	139
12.3.2 Other Commands and Operations during Tool Radius Compensation.....	148
12.3.3 G41/G42 Commands and I, J, K Designation	158
12.3.4 Interrupts during Tool Radius Compensation	164
12.3.5 General Precautions for Tool Radius Compensation.....	167
12.3.6 Changing of Compensation No. during Compensation Mode.....	168
12.3.7 Start of Tool Radius Compensation and Z Axis Cut in Operation.....	171
12.3.8 Interference Check	173
12.3.9 Workpiece Coordinate Changing during Radius Compensation.....	183
12.4 Compensation Data Input by Program ; G10.....	185
12.5 Inputting the Tool Life Management Data ; G10,G11	190
12.5.1 Inputting the Tool Life Management Data by G10 L3 Command ; G10 L3,G11	190
12.5.2 Inputting the Tool Life Management Data by G10 L30 Command ; G10 L30,G11	192
12.5.3 Precautions for Inputting the Tool Life Management Data.....	195
13 Program Support Functions	197
13.1 Fixed Cycles.....	198
13.1.1 Drilling, Spot Drilling ; G81.....	202
13.1.2 Drilling, Counter Boring ; G82.....	203
13.1.3 Deep Hole Drilling Cycle ; G83	204
13.1.3.1 Deep Hole Drilling Cycle.....	204
13.1.4 Tapping Cycle ; G84	205
13.1.5 Boring ; G85.....	213
13.1.6 Boring ; G86.....	214
13.1.7 Back Boring ; G87.....	215
13.1.8 Boring ; G88.....	217
13.1.9 Boring ; G89.....	218
13.1.10 Stepping Cycle ; G73.....	219
13.1.11 Reverse Tapping Cycle ; G74.....	220
13.1.12 Circular Cutting ; G75	222
13.1.13 Fine Boring ; G76.....	224
13.1.14 Precautions for Using a Fixed Cycle.....	226
13.1.15 Initial Point and R Point Level Return ; G98,G99.....	228
13.1.16 Setting of Workpiece Coordinates in Fixed Cycle Mode	229
13.1.17 Drilling Cycle with High-Speed Retract.....	230
13.1.18 Acceleration/Deceleration Mode Change in Hole Drilling Cycle	234
13.2 Special Fixed Cycle.....	236
13.2.1 Bolt Hole Cycle ; G34.....	237
13.2.2 Line at Angle ; G35	238
13.2.3 Arc ; G36.....	239
13.2.4 Grid ; G37.1	240

13.3 Subprogram Control; M98, M99, M198	241
13.3.1 Subprogram Call ; M98,M99.....	241
13.3.2 Subprogram Call ; M198.....	247
13.4 Variable Commands	248
13.5 User Macro	250
13.5.1 User Macro	250
13.5.2 Macro Call Instruction	251
13.5.2.1 Simple Macro Calls ; G65.....	251
13.5.2.2 Modal Call A (Movement Command Call) ; G66.....	255
13.5.2.3 Modal Call B (for each block) ; G66.1.....	257
13.5.2.4 G Code Macro Call	259
13.5.2.5 Miscellaneous Command Macro Call (for M, S, T, B Code Macro Call)	260
13.5.2.6 Detailed Description for Macro Call Instruction	261
13.5.3 ASCII Code Macro	263
13.5.4 Variable.....	267
13.5.5 Types of Variables	269
13.5.5.1 Common Variables	269
13.5.5.2 Local Variables (#1 to #33)	270
13.5.5.3 Macro Interface Inputs/Outputs (#1000 to #1035, #1100 to #1135, #1200 to #1295, #1300 to #1395)	275
13.5.5.4 Tool Compensation	282
13.5.5.5 Workpiece Coordinate System Compensation (#5201 - #532n)	283
13.5.5.6 NC Alarm (#3000)	284
13.5.5.7 Integrating Time (#3001, #3002)	285
13.5.5.8 Suppression of Single Block Stop and Miscellaneous Function Finish Signal Waiting (#3003)	285
13.5.5.9 Feed Hold, Feedrate Override, G09 Valid/Invalid (#3004)	286
13.5.5.10 Message Display and Stop (#3006)	286
13.5.5.11 Mirror Image (#3007)	287
13.5.5.12 G Command Modals (#4001-#4021, #4201-#4221)	288
13.5.5.13 Other Modals (#4101 - #4120, #4301 - #4320)	289
13.5.5.14 Position Information (#5001 - #5160 + n)	290
13.5.5.15 Number of Workpiece Machining Times (#3901, #3902)	293
13.5.5.16 Coordinate Rotation Parameter	293
13.5.5.17 Tool Life Management (#60000 - #64700)	294
13.5.5.18 Reading The Parameters (#100000-#100002, #100010)	299
13.5.5.19 Reading PLC data (#100100-#100103,#100110)	302
13.5.5.20 Time Reading Variables (#3001, #3002, #3011, #3012)	306
13.5.5.21 R Device Access Variables (#50000 - #50749, #51000 - #51749, #52000 - #52749) ..	308
13.5.6 Operation Commands	314
13.5.7 Control Commands	319
13.5.8 External Output Commands ; POPEN, PCLOS, DPRNT	322
13.5.9 Precautions	326
13.5.10 Actual Examples of Using User Macros.....	328
13.6 G Command Mirror Image ; G50.1,G51.1.....	332
13.7 Corner Chamfering I / Corner Rounding I	336
13.7.1 Corner Chamfering I ; G01 X_ Y_ ,C_.....	336
13.7.2 Corner Rounding I ; G01 X_ Y_ ,R_.....	338
13.7.3 Corner Chamfering Expansion/Corner Rounding Expansion	340
13.7.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding	342
13.8 Corner Chamfering II / Corner Rounding II	343
13.8.1 Corner Chamfering II ; G01/G02/G03 X_ Y_ ,C_.....	343
13.8.2 Corner Rounding II ; G01/G02/G03 X_ Y_ ,R_.....	345
13.8.3 Corner Chamfering Expansion/Corner Rounding Expansion	346
13.8.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding	346
13.9 Linear Angle Command ; G01 X_/Y_ A_/A_.....	347
13.10 Geometric	348
13.10.1 Geometric I ; G01 A_.....	348
13.10.2 Geometric IB	350
13.10.2.1 Geometric IB (Automatic calculation of two-arc contact) ; G02/G03 P_Q_/R_.....	351

13.10.2.2 Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 P_Q_H_	353
13.10.2.3 Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 R_H_	356
13.11 Circular Cutting ; G12,G13.....	359
13.12 Parameter Input by Program ; G10 L70/L100, G11	361
13.13 Macro Interruption ; M96,M97	363
13.14 Tool Change Position Return ; G30.1 - G30.6	371
13.15 Scaling ; G50/G51	374
13.16 Coordinate Rotation by Program ; G68/G69.....	378
14 Coordinate System Setting Functions	385
14.1 Coordinate Words and Control Axes.....	386
14.2 Basic Machine, Workpiece and Local Coordinate Systems.....	387
14.3 Machine Zero Point and 2nd, 3rd, 4th Reference Position (Zero point)	388
14.4 Automatic Coordinate System Setting	389
14.5 Basic Machine Coordinate System Selection ; G53.....	391
14.6 Coordinate System Setting ; G92.....	392
14.7 Reference Position (Zero point) Return ; G28,G29.....	394
14.8 2nd, 3rd, and 4th Reference Position (Zero point) Return ; G30	398
14.9 Reference Position Check ; G27	401
14.10 Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1)	402
14.11 Local Coordinate System Setting ; G52	411
14.12 Coordinate System for Rotary Axis.....	415
15 Measurement Support Functions	419
15.1 Automatic Tool Length Measurement ; G37	420
15.2 Skip Function ; G31.....	424
15.3 Multi-step Skip Function 1 ; G31.n ,G04	429
15.4 Multi-step Skip Function 2 ; G31 P.....	431
15.5 Programmable Current Limitation ; G10 L14 ;	433
Appendix 1 Program Errors	435
Appendix 2 Order of G Function Command Priority	469



Control Axes

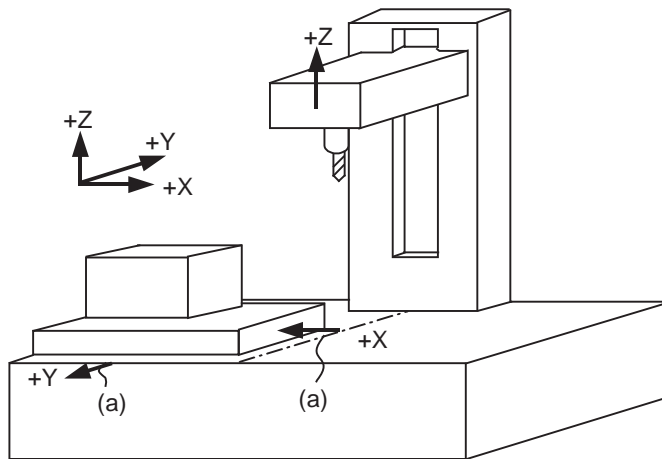
1.1 Coordinate Words and Control Axes



Function and purpose

In the standard specifications, there are 3 control axes. 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

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

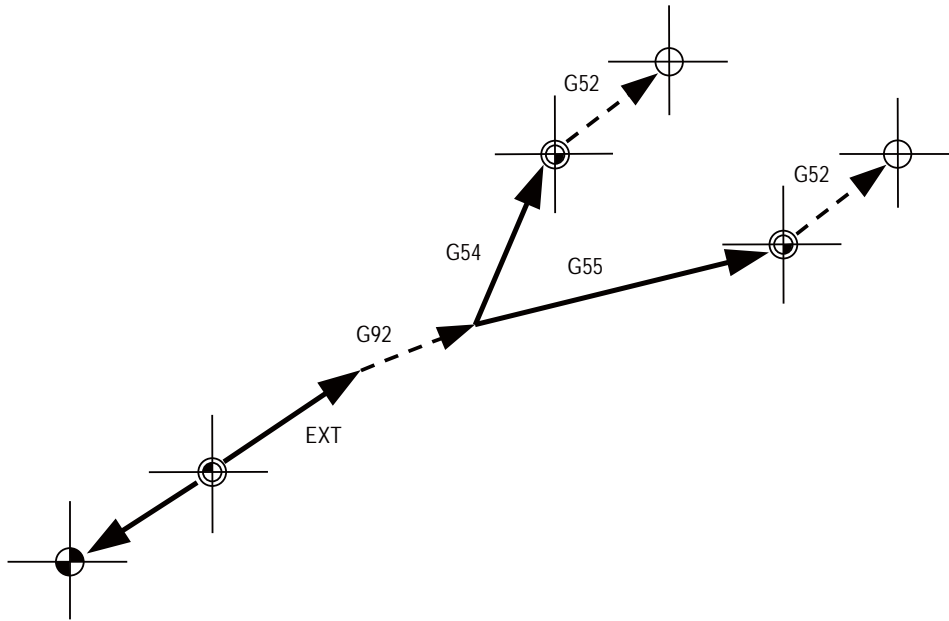
The basic machine coordinate system is the coordinate system that expresses the position (tool change position, stroke end position, etc.) that is specific to the machine.







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, 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.

Least Command Increments

2.1 Input Setting Unit



Function and purpose

The input setting units are the units of setting data including tool compensation amounts and workpiece coordinates compensation.

The program command units are the units of movement amounts in programs.

These are expressed with mm, inch or degree (°).



Detailed description

Program command units for each axis and input setting units, common for all axes, are determined by the setting of parameters as follows.

	Parameter		Linear axis		Rotary axis (°)
			Millimeter	Inch	
Input setting unit	#1003 iunit	= B	0.001	0.0001	0.001
		= C	0.0001	0.00001	0.0001
Program command unit	#1015 cunit	= 0	Follow #1003 iunit		
		= 1	0.0001	0.00001	0.0001
		= 10	0.001	0.0001	0.001
		= 100	0.01	0.001	0.01
		= 1000	0.1	0.01	0.1
		= 10000	1.0	0.1	1.0



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 program command units, and not to the input setting units. Consequently, the tool compensation 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 program 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.)

2.2 Input Command Increment Tenfold



Function and purpose

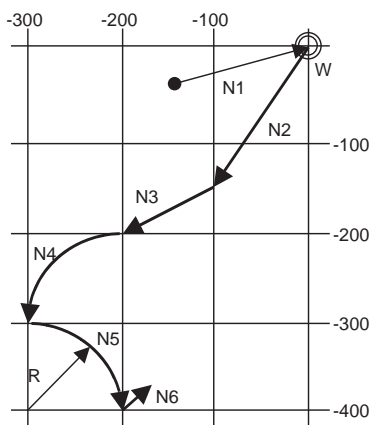
The program's command increment can be multiplied by an arbitrary scale with the parameter designation. This function is valid when a decimal point is not used for the command increment. The scale is set with the parameter "#8044 UNIT*10".



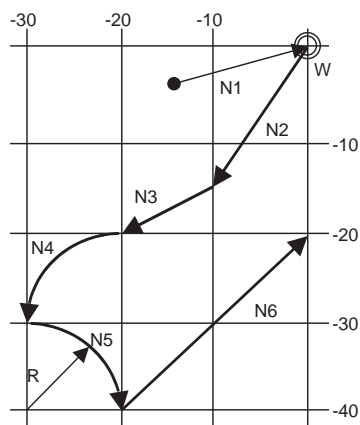
Detailed description

- (1) When running a machining program already created with a 10 μ m input command increment with a CNC unit for which the command increment is set to 1 μ m and this function's parameter value is set to "10", this function enables the same machining as the original program.
- (2) When running a machining program already created with a 1 μ m input command increment with a CNC unit for which the command increment is set to 0.1 μ m and this function's parameter value is set to "10", this function enables the same machining as the original program.
- (3) This function cannot be used for the dwell function G04_X_(P_);.
- (4) This function cannot be used for the compensation amount of the tool compensation input.
- (5) This function can be used when decimal point type I is valid, but cannot be used when decimal point type II is valid.
- (6) This function cannot be used for a tool shape setting command (in G10L100 format).

Program example (Machining program : programmed with 1=10 μ m) (CNC unit is 1=1 μ m system)	"UNIT*10" parameter			
	10		1	
	X	Y	X	Y
N1 G90 G00 X0 Y0;	0	0	0	0
N2 G91 X-10000 Y-15000;	-100.000	-150.000	-10.000	-15.000
N3 G01 X-10000 Y-5000 F500;	-200.000	-200.000	-20.000	-20.000
N4 G03 X-10000 Y-10000 J-10000;	-300.000	-300.000	-30.000	-30.000
N5 X10000 Y-10000 R10000;	-200.000	-400.000	-20.000	-40.000
N6 G01 X20.000 Y20.000	-180.000	-380.000	0.000	-20.000



UNIT*10 ON



UNIT*10 OFF



Program Formats

3.1 Program Format

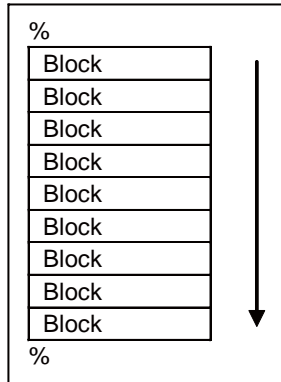
A collection of commands assigned to an NC to move a machine is called "program".

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 units called "word" which constitutes a command to an operation.

A word is a collection of characters (alphabets, numerals, signs) arranged in a specific sequence.

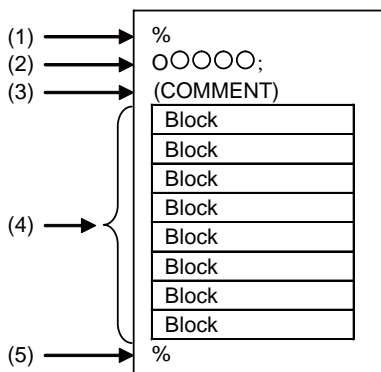




Detailed description

Program

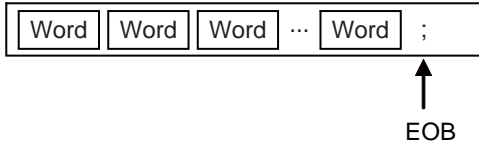
A program format looks as follows.



- (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. For details, refer to the description of the file format.
- (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]

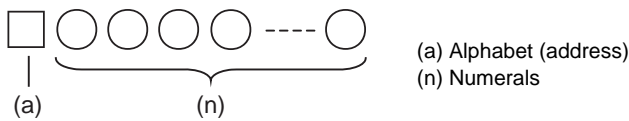


A block is a least command increment, consisting of words.

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.

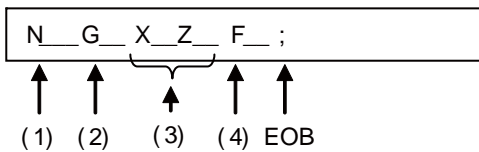
[Word]



A 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.

(Note) Leading zeros can be omitted from numerals.

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 6 digits (Normally 3 or 4 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

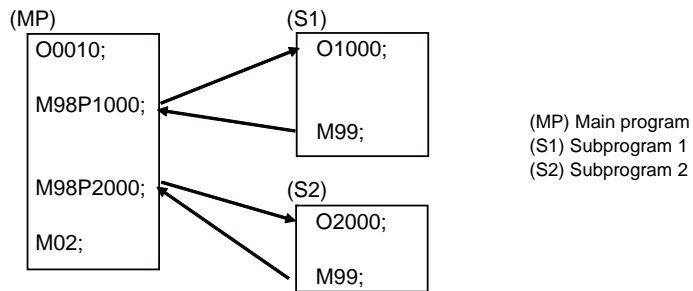
"Coordinate words" specify the coordinate position and movement amounts of tool machine axes. They consist of an address which indicates each axis of a tool machine followed by numerical information (+ or - signs and numerals).

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



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.

3.2 File Format



Function and purpose

Program file can be created using NC edit screen and PC.

It can be input/output between NC memory and an external I/O device. For the details of input/output method, refer to the instruction manual.

Program file format differs depending on the device which creates the program.



Detailed description

Devices available for input/output

Devices which can input/output program files are as follows.

- NC memory
- Serial
- Memory card (front IC card)
- USB memory
- Ethernet

Program file format

The file format for each external I/O device is as follows.

- (1) NC memory (Creates program on NC)

```
(COMMENT) ;
G28XYZ ;
:
:
M02 ;
%
```

End of record (EOR, %)	The end of record (EOR, %) is automatically added. It does not need to be input purposely.
Program No. (O No.)	Not necessary.
File transfer	When multiple programs within the NC memory are transferred to an external device as serial, they will be integrated into one file in the external device. When a file containing multiple programs in an external device is transferred to NC memory as serial, it will be divided into one file per one program.

(2) External device (except for serials, such as memory card, USB memory)

[Single program]	[Multiple programs]
<pre> CRLF (COMMENT) CRLF G28 XYZ CRLF : : : M02 CRLF % ^Z </pre>	<pre> CRLF O100(COMMENT) CRLF G28 XYZ CRLF : : : M02 CRLF O101(COMMENT1) CRLF : : M02 CRLF % ^Z </pre>

End of record (EOR, %)	The first line (from % to LF, or CR LF) will be skipped. Also, the content after the second % will not be transferred. "%" must be included in the first line because if not, the necessary information when transferring a file to an NC memory cannot be transferred.
Program No. (O No.)	O No. before (COMMENT) will be ignored and the file name will be given the priority.
File transfer	Transfer and check of multiple programs between external devices, except for serial <_> serial, are not available. When a file containing multiple programs in an external device is transferred to NC memory as serial, it will be divided into one file per one program. When transferring divided programs one by one from an external device, which is not serial, (multiple programs) to an NC memory, the head program name can be omitted like "(COMMENT)" only when the transferring destination file name is designated to the file name field of device B.
Program name	Program name should be designated with up to 32 alphanumeric characters.
End of block (EOB, ;)	When the I/O parameter "CR output" is set to "1", EOB becomes CRLF.

(3) External device (serial)

```

% LF
O100(COMMENT) LF
G28 XYZ LF
:
:
:
M02 LF
%
                    
```

End of record (EOR, %)	The first line (from % to LF, or CR LF) will be skipped. Also, the content after the second % will not be transferred. "%" must be included in the first line because if not, the necessary information when transferring a file to an NC memory cannot be transferred.
File transfer	Transfer and check of multiple programs between external devices, except for serial <_> serial, are not available. When transferring a file as serial, the head program name can be omitted like "(COMMENT)" only when the transferring destination file name is designated to the file name field of device B.
Program name	Program name should be designated with up to 32 alphanumeric characters.
End of block (EOB, ;)	When the I/O parameter "CR output" is set to "1", EOB becomes CRLF.

3.3 Optional Block Skip

3.3.1 Optional Block Skip; /



Function and purpose

This function selectively ignores a section of a machining program from a "/" (slash code) to the end of the block.



Detailed description

Provided that the optional block skip switch is ON, sections of a machining program from a "/" to the end of the block 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, one machining program can be used to machine different parts by inserting the "/" into those specific blocks.



Program example

- When the parameter "#1274 ext10/bit4" is set to "0" and the parameter "#1226 aux10/bit1" is set to "0":
A "/" placed in the middle of a block is always interpreted as a division instruction regardless of the optional block skip signal state being ON or OFF.

G00 X0. Z0.;	
#101 = [100. / 4] ;	Sets "25." to #101. (As the result of execution of a division instruction)
G00 Z[100. / 4] ;	Moves Z axis to "25.". (As the result of execution of a division instruction)
#102 = 100. / #101 ;	Sets "4." to #102. (As the result of execution of a division instruction)
M30 ;	

- When the parameter "#1274 ext10/bit4" is set to "0" and the parameter "#1226 aux10/bit1" is set to "1":
A "/" placed in a bracketed ("["]) expression is interpreted as a division instruction.
As for a "/" that appears in any other contexts, the section of the block following the "/" will be skipped if the optional skip signal is ON, and the "/" itself will be ignored if the optional skip signal is OFF.
- Operation example of a case when optional block skip signal is ON:

G00 X0. Z0.;	
#101 = [100. / 4] ;	Sets "25." to #101. (As the result of execution of a division instruction)
G00 X100. / Z200. ;	Moves X axis to "100.". No Z axis movements made. (As the result of skipping the section of the block after "/")
G00 Z[100. / 4] ;	Moves Z axis to "25.". (As the result of execution of a division instruction)
#102 = 100. / #101 ;	Sets "100." to #102. (As the result of skipping the section of the block after "/")
M30 ;	

- Operation example of a case when optional block skip signal is OFF:

G00 X0. Z0.;	
#101 = [100. / 4] ;	Sets "25." to #101. (As the result of execution of a division instruction)
G00 X100. / Z200. ;	Moves X axis to "100." and Z axis to "200.". (As the result of ignoring "/")
G00 Z[100. / 4] ;	Moves Z axis to "25.". (As the result of execution of a division instruction)
#102 = 100. / #101 ;	Program error "P242 = not defined at vrble set" occurs. (As the result of ignoring "/")
M30 ;	

- (3) When the parameter "#1274 ext10/bit4" is set to "1":

When a "/" is placed in a bracketed expression or when an expression that includes a "/" is on the right side of an equation, the "/" is interpreted as a division instruction.

As for a "/" that appears in any other contexts, the section of the block following the "/" will be skipped if the optional skip signal is ON, and the "/" itself will be ignored if the optional skip signal is OFF.

- Operation example of a case when optional block skip signal is ON:

G00 X0. Z0.;	
#101 = [100. / 4] ;	Sets "25." to #101. (As the result of execution of a division instruction)
G00 X100. / Z200. ;	Moves X axis to "100.". No Z axis movements made. (As the result of skipping the section of the block after "/")
G00 Z[100. / 4] ;	Moves Z axis to "25.". (As the result of execution of a division instruction)
#102 = 100. / #101 ;	Sets "4." to #102. (As the result of execution of a division instruction)
M30 ;	

- Operation example of a case when optional block skip signal is OFF:

G00 X0. Z0.;	
#101 = [100. / 4] ;	Sets "25." to #101. (As the result of execution of a division instruction)
G00 X100. / Z200. ;	Moves X axis to "100." and Z axis to "200.". (As the result of ignoring "/")
G00 Z[100. / 4] ;	Moves Z axis to "25.". (As the result of execution of a division instruction)
#102 = 100. / #101 ;	Sets "4." to #102. (As the result of execution of a division instruction)
M30 ;	



Precautions

- (1) When parameter "#1274 ext10/bit4" = "0" and parameter "#1226 aux10/bit1" = "0", 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. /Z25. ;NG (User macro, a division instruction; a program error results.)
/N20 G1 X25. Z25. ;OK
When parameter "#1274 ext10/bit4" = "0" and parameter "#1226 aux10/bit1" = "1", a "/" placed in the middle of a block functions as a starting point of the optional skip. To use a "/" as a division instruction, bracket (enclose in square brackets) the necessary section.
- (2) A space immediately followed by a "/" at the very beginning of a block is always regarded as equal to a "/" at the head of a block regardless of the value set in parameter "#1226 aux10/bit1".
- (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.

3.4 G code

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 G Code Lists

G code	Group	Function	Section
Δ 00	01	Positioning	6.1
Δ 01	01	Linear interpolation	6.2
02	01	Circular interpolation CW (clockwise) R-specified circular interpolation CW Helical interpolation CW	6.3 6.4 6.7
03	01	Circular interpolation CCW (counterclockwise) R-specified circular interpolation CCW Helical interpolation CCW	6.3 6.4 6.7
04	00	Dwell	8.1
09	00	Exact stop check	7.8
10	00	Compensation Data Input by Program Parameter Input by Program	12.4 13.12
11	00	Cancel (Compensation Data Input by Program, Parameter Input by Program)	12.4 13.12
12	00	Circular cut CW (clockwise)	13.11
13	00	Circular cut CCW (counterclockwise)	13.11
14			
Δ 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
Δ 21	06	Metric command	5.2
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	13.14
30.2	00	Tool change position return 2	13.14
30.3	00	Tool change position return 3	13.14
30.4	00	Tool change position return 4	13.14
30.5	00	Tool change position return 5	13.14
30.6	00	Tool change position return 6	13.14
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

G code	Group	Function	Section
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 compensation (+)	12.2
44	08	Tool length compensation (-)	12.2
* 49	08	Tool length compensation cancel	12.2
* 50	11	Scaling cancel	13.15
51	11	Scaling ON	13.15
* 50.1	19	G command mirror image cancel	13.6
51.1	19	G command mirror image ON	13.6
52	00	Local coordinate system setting	14.11
53	00	Basic machine coordinate system selection	14.5
* 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	7.9
62	13	Automatic corner override	7.11
63	13	Tapping mode	7.12
* 64	13	Cutting mode	7.13
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	13.5.2
68	16	Coordinate rotation by program ON	13.16
* 69	16	Coordinate rotation by program cancel	13.16
70	09	User fixed cycle	
71	09	User fixed cycle	
72	09	User fixed cycle	
73	09	Fixed cycle (step)	13.1.10
74	09	Fixed cycle (reverse tap)	13.1.11
75	09	Fixed cycle (circle cutting cycle)	13.1.12
76	09	Fixed cycle (fine boring)	13.1.13
77	09	User fixed cycle	
78	09	User fixed cycle	
79	09	User fixed cycle	
* 80	09	Fixed cycle cancel	13.1
81	09	Fixed cycle (drill/spot drill)	13.1.1
82	09	Fixed cycle (drill/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

G code	Group	Function	Section
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 value command	5.1
Δ 91	03	Incremental command value	5.1
92	00	Coordinate system setting / Spindle clamp speed setting	14.6
Δ 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
Δ 97	17	Constant surface speed control OFF	10.2
* 98	10	Fixed cycle Initial level return	13.1.15
99	10	Fixed cycle R point level return	13.1.15
100 - 225	00	User macro (G code call) Max. 10	13.5.2



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 compensation 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.

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__ ;

G90	Absolute command
G91	Incremental command
X,Y,Z	Coordinate values



Detailed description

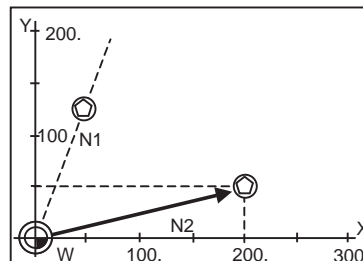
- (1) Regardless of the current position, in the absolute value mode, it is possible to move to the position of the workpiece coordinate system that was designated in the program.

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

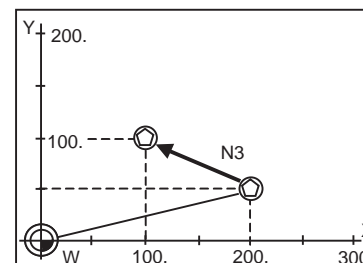
- (2) For the next block, the last G90/G91 command that was given becomes the modal.

(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. ;

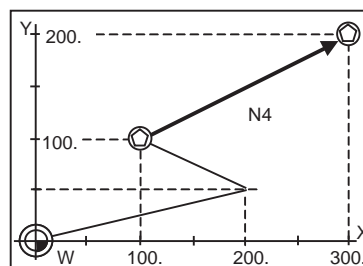
The X axis moves to -100.mm and the Y axis to +50.0mm as an incremental value, and as a result X moves to 100.mm and Y to 100.mm.



- (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.

Output unit, command unit and setting unit

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

NC axis

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

PLC axis

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



Precautions

- (1) The parameter and tool data will be input/output with the unit set by "#1041 I_inch".
If "#1041 I_inch" is not found in the parameter input data, the unit will follow the unit currently set to NC.
- (2) The unit of read/write used in PLC window is fixed to metric unit regardless of a parameter and G20/G21 command modal.

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=10000	1000 (μ m, 10^{-4} inch, 10^{-3}°)	1 (mm, inch, $^\circ$)
	cunit= 1000	100	1
	cunit= 100	10	1
	cunit= 10	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) In decimal point command, the valid range of command value is as shown below. (Input command unit cunit = 10)

	Movement command (linear)	Movement command (rotary)	Feedrate	Dwell
Input unit [mm]	-99999.999 to 99999.999	-99999.999 to 99999.999	0.001 to 1000000.000	0 to 99999.999
Input unit [inch]	-9999.9999 to 9999.9999		0.0001 to 1000000.0000	

- (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 (1μ m, 10μ 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.
- (8) "Input command increment tenfold" is applied in the decimal point type I mode, but not in the decimal point type II mode.

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)	
B	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function code	
C	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function code	
	Valid	Corner chamfering amount	,C
D	Invalid	Compensation numbers (tool position, tool radius)	
	Valid	Automatic tool length measurement: deceleration distance d	
	Invalid	Data setting: byte type data	
	Invalid	Subprogram storing device number	,D
E	Valid	Inch thread: number of ridges, precision thread: lead	
F	Valid	Feedrate, automatic tool length measurement speed	
	Valid	Thread lead	
	Valid	Number of Z axis pitch in synchronous tap	
G	Valid	Preparatory function code	
H	Invalid	Tool length compensation number	
	Invalid	Sequence numbers in subprograms	
	Invalid	Parameter input by program: bit type data	
	Invalid	Basic spindle selection	
I	Valid	Arc center coordinates	
	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	

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	
	Invalid	Number of drilling cycle repetitions	
L	Invalid	Number of fixed cycle and subprogram repetitions	
	Invalid	Tool compensation data input by program/workpiece offset input: type selection	L2, L20, L10, L11 L12, L13,
	Invalid	Parameter input by program: data setting selection	L70
	Invalid	Parameter input by program: 2-word type data	4 bytes
	Invalid	Tool life data	
M	Invalid	Miscellaneous function codes	
N	Invalid	Sequence numbers	
	Invalid	Parameter input by program: data numbers	
O	Invalid	Program numbers	
P	Invalid/Valid	Dwell time	Parameter
	Invalid	Subprogram program call: program No.	
	Invalid/Valid	Dwell at tap cycle hole base	Parameter
	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	Parameter input by program: broad classification number	
	Invalid	Multi-step skip function 2 signal command	
	Invalid	Subprogram return destination sequence No.	
	Invalid	2nd, 3rd, 4th reference position return number	
	Valid	Scaling magnification	
	Invalid	Extended workpiece coordinate system No.	
	Invalid	Tool life data group No.	
	Invalid	Machining purpose	
Q	Valid	Cut amount of deep hole drill cycle	
	Valid	Shift amount of back boring	
	Valid	Shift amount of fine boring	
	Invalid	Minimum spindle clamp speed	
	Valid	Starting shift angle for screw cutting	
	Invalid	Tool life data management method	
	Invalid	Machining condition	
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	
	Invalid	Skip acceleration/deceleration time constant	
S	Invalid	Spindle function codes	
	Invalid	Maximum spindle clamp speed	
	Invalid	Constant surface speed control: surface speed	
	Invalid	Parameter input by program: word type data	2 bytes
T	Invalid	Tool function codes	
U	Valid	Coordinate position data	
V	Valid	Coordinate position data	
W	Valid	Coordinate position data	
X	Valid	Coordinate position data	
	Valid	Dwell time	

Address	Decimal Point Command	Application	Remarks
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 = 1mm
	When 1 = 1 μm	When 1 = 10 μm	
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 1 μm 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 μm.

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__,I__ ; ... Positioning (Rapid Traverse)

X, Y, Z	Coordinate values. 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 block that does not contain this address will follow the parameter "#1193 inpos" settings. 1 to 999999

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.
- (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.

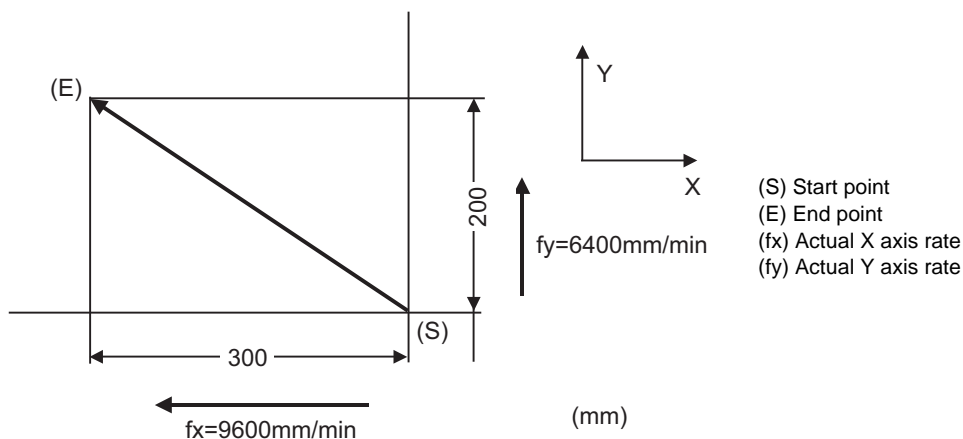
- (1) Linear path: When the parameter "#1086 G0Intp" is set to "0"

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.



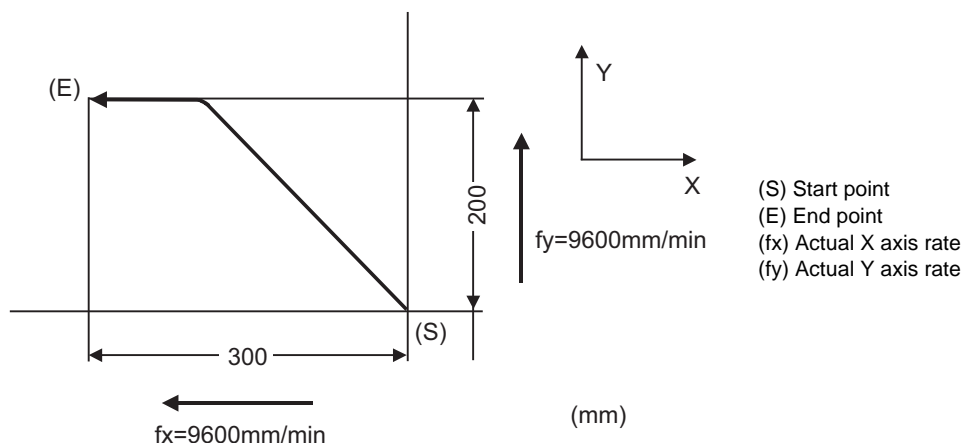
- (2) Non-linear path: When the parameter "#1086 G0Intp" is set to "1"

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;

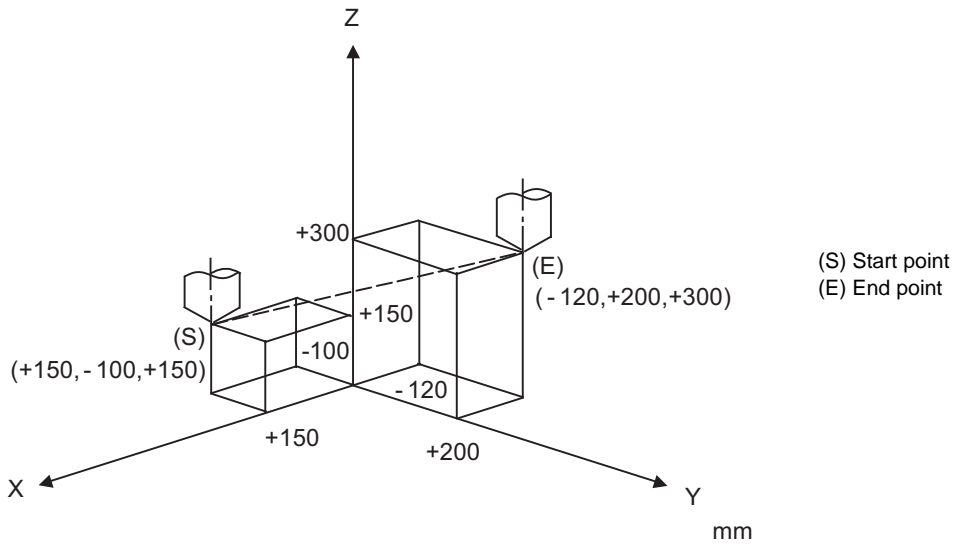
G91 G00 X-300000 Y200000 ; (With an input setting unit of 0.001mm)

The tool will follow the path shown in the figure below.





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")	

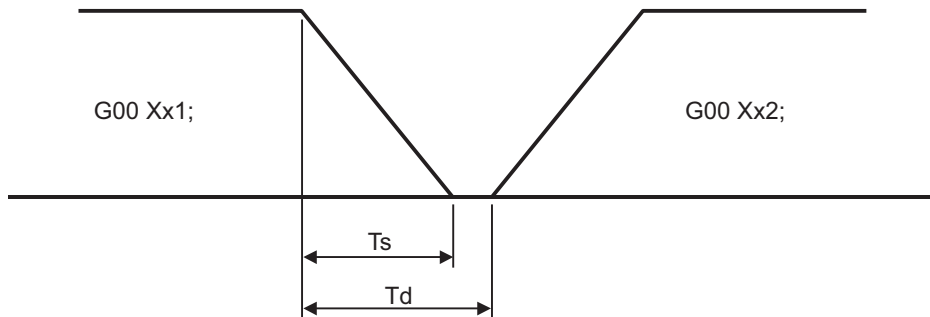
* Following descriptions are for the case of rapid traverse. For G01, interpret the parameters into suitable ones.

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".

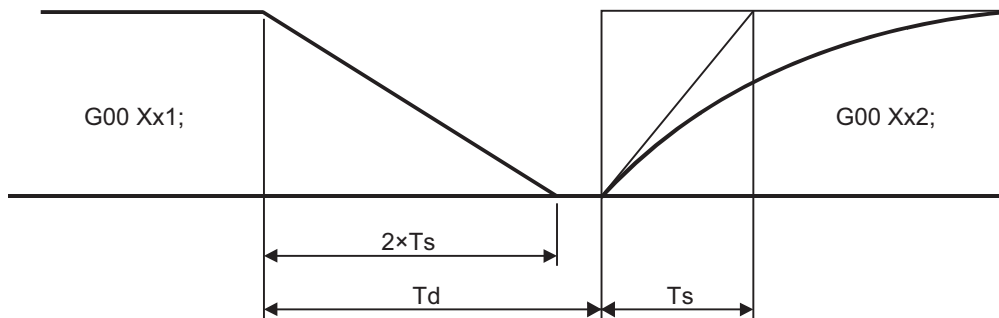
(1) Linear acceleration/linear deceleration



(Ts) Acceleration/deceleration time constant

(Td) Deceleration check time: $T_d = T_s + (0 \text{ to } 7\text{ms})$

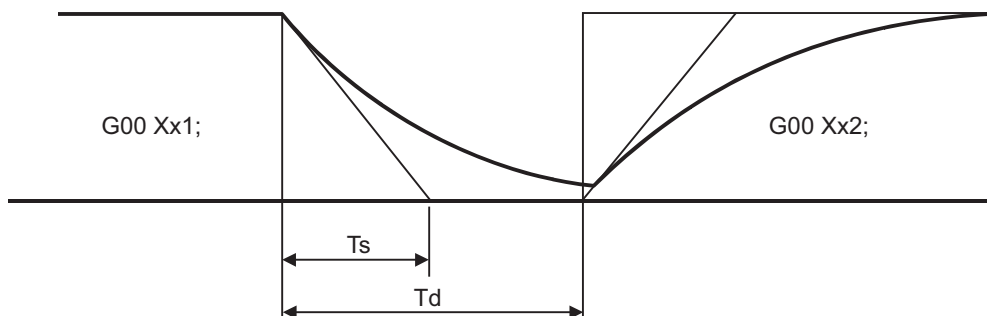
(2) Exponential acceleration/linear deceleration



(Ts) Acceleration/deceleration time constant

(Td) Deceleration check time: $T_d = 2 \times T_s + (0 \text{ to } 7\text{ms})$

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



(Ts) Acceleration/deceleration time constant

(Td) Deceleration check time: $T_d = 2 \times T_s + (0 \text{ to } 7\text{ms})$

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.

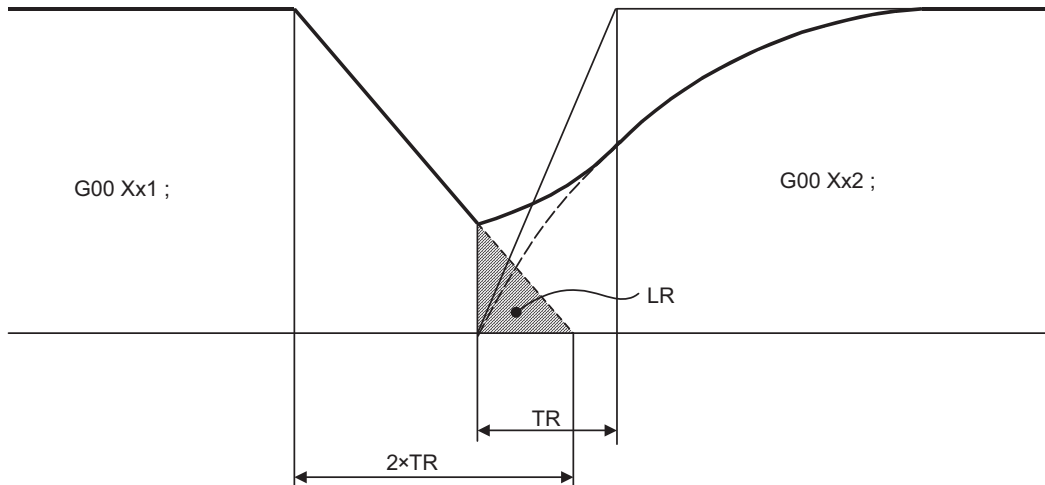
In-position check method when "inpos" = 1

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.

G0 deceleration pattern



(TR) Rapid traverse acceleration and deceleration time constant

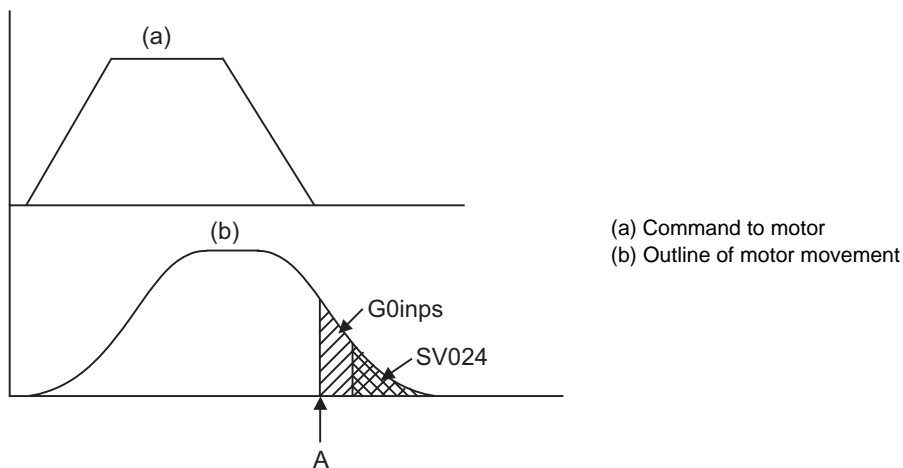
(LR) In-position width

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).

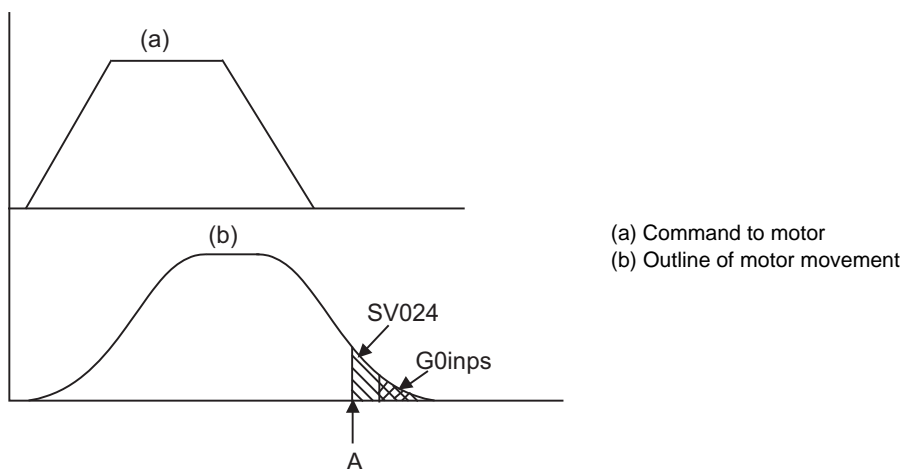
The purpose of the rapid traverse deceleration check is to minimize the positioning time. The bigger the setting value for the in-position width, the shorter the time is, but the remaining distance of the previous block at the start of the next block also becomes larger, and this could become an obstacle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



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



Programmable in-position width command

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

G00 X_ Z_ ,I_ ;	
X,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.

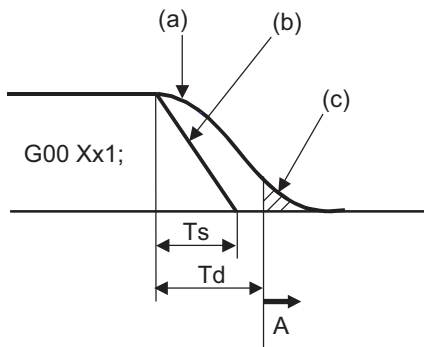
When there are several movement axes, the system confirms that the position error amount of each movement axis is less than the in-position width issued in this command before executing the next block.

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

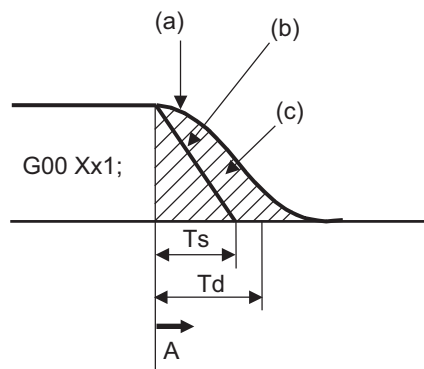
After completing deceleration of the command system (A), the servo system's position error amount and the parameter setting value (in-position width) are compared.



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

(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: $T_d = T_s + (0 \text{ to } 7\text{ms})$

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_ F_ ,I_ ; ... Linear interpolation

X,Y,Z	Coordinate values. 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 block that does not contain this address will follow the parameter "#1193 inpos" settings. 1 to 999999



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.

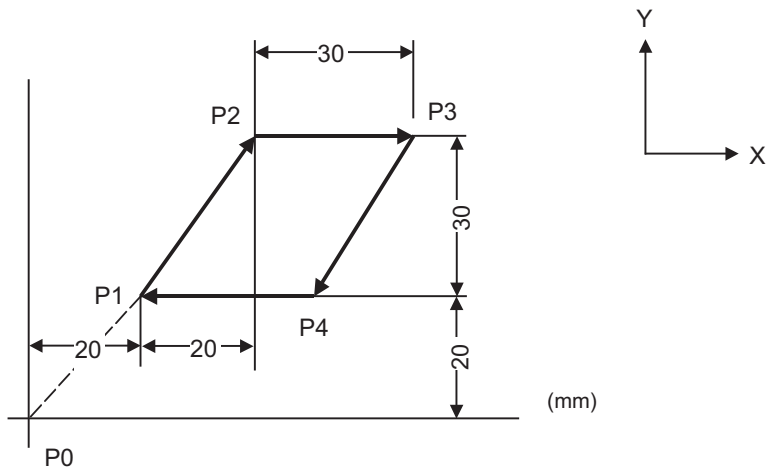
- When the error detection switch is ON.
- When G09 (exact stop check) is commanded in the same block.
- When G61 (exact stop check mode) is selected.

(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)
--

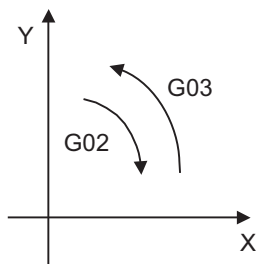
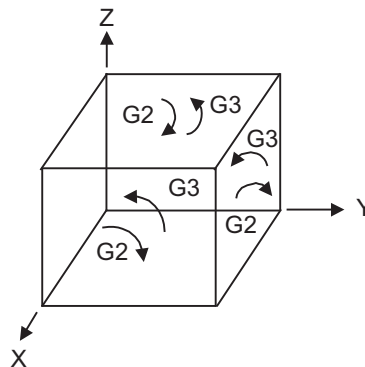
X,Y	End point
I,J	Arc center
F	Feedrate



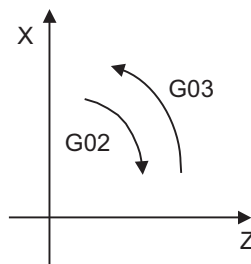
Detailed description

- (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.

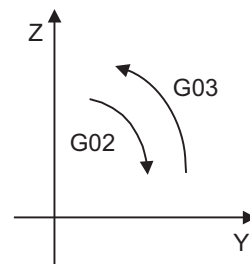
- (2) G02 (G03) is a modal command of the 01 group. When G02 (G03) command is issued continuously, the next block and after can be commanded with only coordinate words.
 The circular rotation direction is distinguished by G02 and G03.
 G02 Clockwise (CW)
 G03 Counterclockwise (CCW)



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)

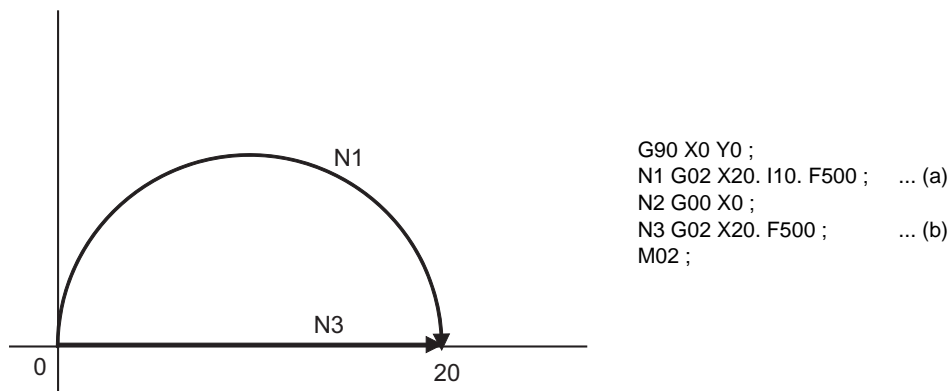
Change into linear interpolation command

Program error (P33) will occur when the center and radius are not designated at circular command.

When the parameter "#11029 Arc to G1 no Cent (Change command from arc to linear when no arc center designation)" is set, the linear interpolation can be operated up to the end point coordinate value only for that block. However, a modal is the circular modal.

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

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



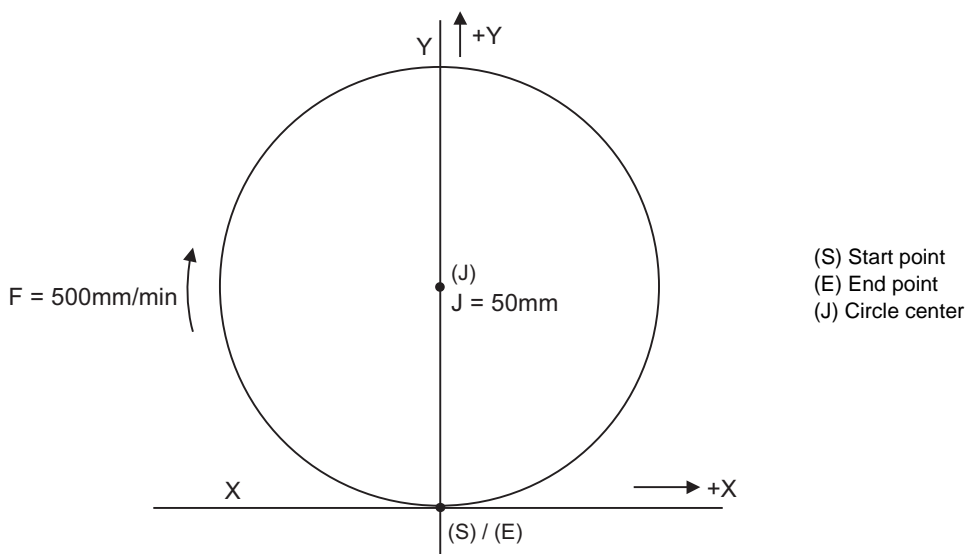
(a) The circular interpolation (G02) is executed because there is a center command.

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



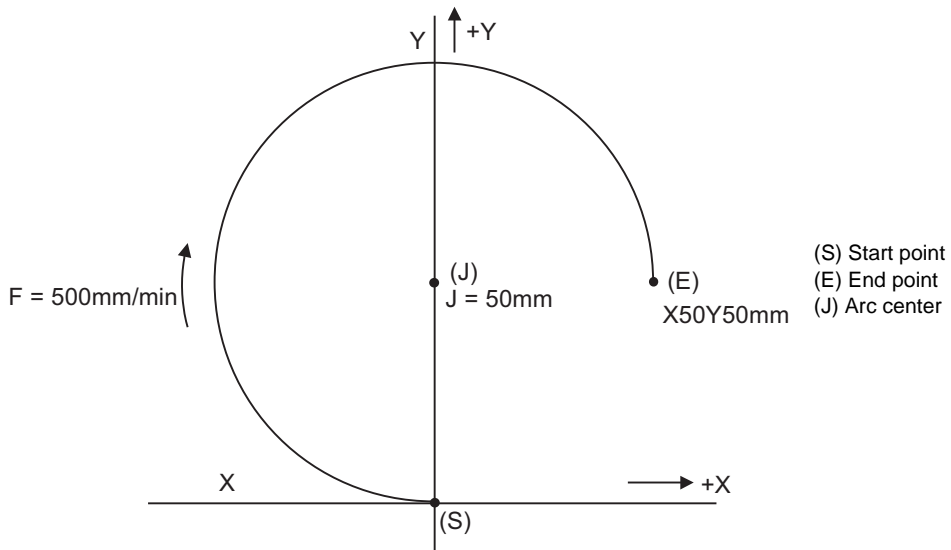
Program example

(Example 1)



G02 J50. F500;	Circle command
----------------	----------------

(Example 2)



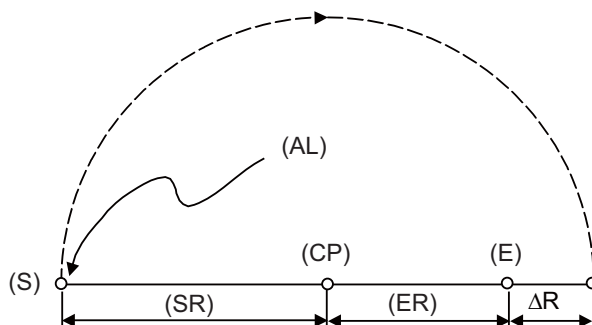
G91 G02 X50.Y50. J50. F500;	3/4 command
-----------------------------	-------------



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. ;

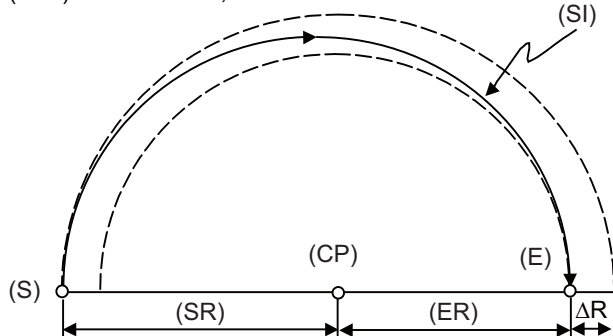


#1084 RadErr Parameter value 0.100
 Start point radius=5.000
 End point radius=4.899
 Error ΔR =0.101

(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
 Error ΔR =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

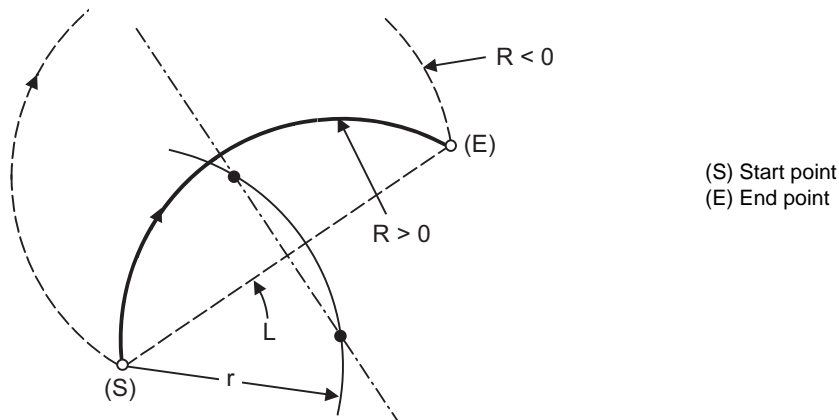
The arc radius is commanded with a program command unit. Caution is required for the arc command of an axis for which the program command unit (#1015 cunit) differs. Command with a decimal point to avoid confusion.



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.



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

$$\frac{L}{2 \cdot r} \leq 1 \quad \text{When } L/2 - r > \text{parameter (\#1084 RadErr), an alarm will occur.}$$

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.

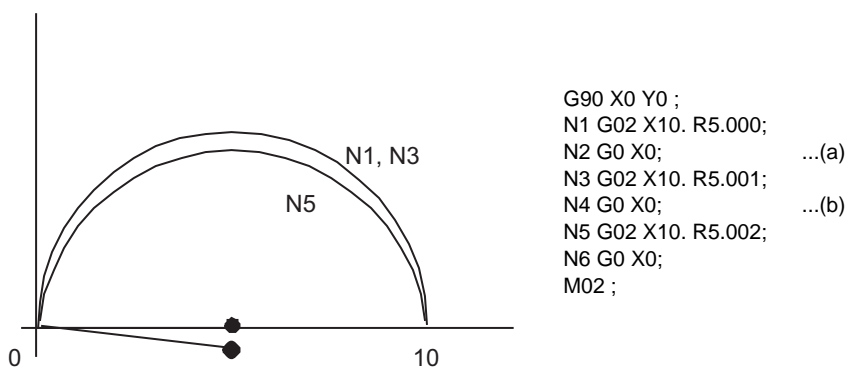
Circular center coordinate compensation

When the error margin between "the segment connecting the start and end points" and "the commanded radius $\times 2$ " is less than the setting value, "the midpoint of segment connecting the start and end points" is compensated as the circular center, because the required semicircle is not obtained by calculation error in R specification circular interpolation.

Set the setting value to the parameter "#11028 Tolerance Arc Cent (Tolerable correction value of arc center error)".

(Example) "#11028 Tolerance Arc Cent" = "0.000 (mm)"

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



(a) Compensate the center coordinate: Same as N1 path

(b) Do not compensate the center coordinate: Slightly inside N1 path

Calculation error margin compensation allowance value: 0.002 mm

Segment connecting the start and end points: 10.000

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

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

Therefore, this example is shown in the above figure.



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) When the axis address is commanded to the same block as the plane selection G code (G17, G18, G19), the commanded axes will travel.

- (5) 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.

- (6) 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

Normal lead thread cutting

G33 Z__(X__Y__) F__ Q__ ;

Z (X Y)	Thread end point
F	Lead of long axis (axis which moves the most) direction
Q	Thread cutting start shift angle (0.001 - 360.000°)

Precision lead thread cutting

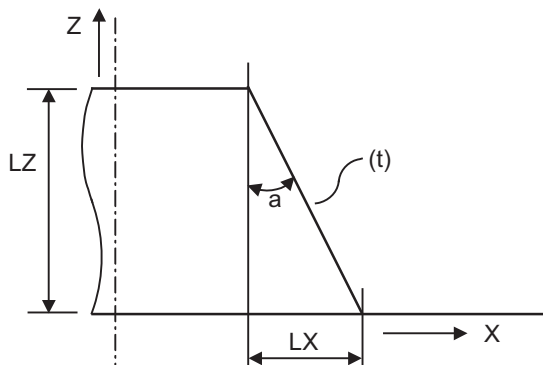
G33 Z__(X__Y__) E__ Q__ ;

Z (X Y)	Thread end point
E	Lead of long axis (axis which moves most) direction
Q	Thread cutting start shift angle (0.001 - 360.000°)



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 "#8156 Fine thread cut E" 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 $a < 45^\circ$, lead is in Z-axis direction.When $a < 45^\circ$, lead is in X-axis direction.When $a = 45^\circ$, lead can be in either Z or X-axis direction.

Thread cutting metric input

Input setting unit	B (0.001mm)		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Least Command Increments	1(=0.001) (1.=1.000)	1(=0.0001) (1.=1.0000)	1(=1.00) (1.=1.00)
Command range	0.001 - 999.999	0.0001 - 999.9999	0.03 - 999.99

Input setting unit	C (0.0001mm)		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Least Command Increments	1(=0.0001) (1.=1.0000)	1(=0.00001) (1.=1.00000)	1(=1.000) (1.=1.000)
Command range	0.0001 - 999.9999	0.00001 - 999.99999	0.026 - 222807.017

Thread cutting inch input

Input setting unit	B (0.0001inch)		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Least Command Increments	1(=0.0001) (1.=1.0000)	1(=0.00001) (1.=1.00000)	1(=1.000) (1.=1.000)
Command range	0.0001 - 39.3700	0.00001 - 39.37007	0.025 - 9999.999

Input setting unit	C (0.00001inch)		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Least Command Increments	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)	1(=1.0000) (1.=1.0000)
Command range	0.00001 - 39.37007	0.000001 - 39.370078	0.0255 - 9999.9999

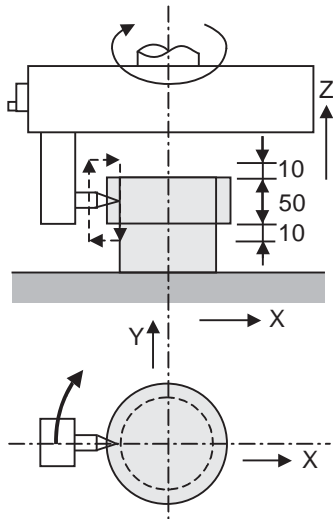
(Note 1) It is not possible to assign a lead where the feedrate as converted into feed per minute exceeds the maximum cutting feedrate.

- (3) The constant surface speed control function should not be used for taper thread cutting commands or scrolled thread cutting commands.
- (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 :

$$1 \leq R \leq \text{Maximum feedrate/Thread lead}$$
 Where $R \leq \text{Tolerable speed of encoder (r/min)}$
 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) A program error (P97) may occur when the result of the expression (9) is $R < 1$ because the thread lead is very large to the highest cutting feedrate.
- (11) Dry run is valid for thread cutting but the feedrate based on dry run is not synchronized with the spindle rotation.
The dry run signal is checked at the start of thread cutting and any switching during thread cutting is ignored.
- (12) Synchronous feed applies for the thread cutting commands even with an asynchronous feed command (G94).
- (13) Spindle override and cutting feed override are invalid and the speeds are fixed to 100% during thread cutting.
- (14) When a thread cutting is commanded during tool radius compensation, the compensation is temporarily canceled and the thread cutting is executed.
- (15) When the mode is switched to another automatic mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops.
- (16) When the mode is switched to the manual mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops. In the case of a single block, the following block which does not contain a thread cutting command (G33 mode is cancelled) is first executed and then the automatic operation stops. Note that automatic operation is stopped until the G33 command axis starts moving.
- (17) The thread cutting command waits for the single rotation synchronization signal of the rotary encoder and starts movement.
- (18) The thread cutting start shift angle is not modal. If there is no Q command with G33, this will be handled as "Q0".
- (19) The automatic handle interrupt/interruption is valid during thread cutting.
- (20) If a value exceeding 360.000 is command in G33 Q, a program error (P35) will occur.
- (21) G33 cuts one row with one cycle. To cut two rows, change the Q value, and issue the same command.



Program example



N110 G90 G0 X-200. Y-200. S50 M3 ;	The spindle center is positioned to the workpiece center, and the spindle rotates in the forward direction.
N111 Z110. ;	
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_) E_ Q_ ; ... Inch thread cutting

Z (X Y)	End point of thread cutting
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

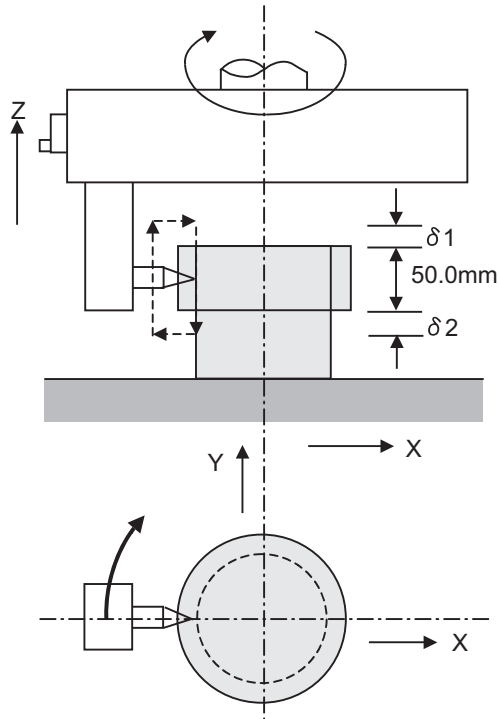
- (1) The number of ridges in the long axis direction is assigned as the number of ridges per inch.
- (2) The E code is also used to assign the precision lead length, and whether the number of ridges or precision lead length is to be designated can be selected by parameter setting. (The number of ridges is designated by setting the parameter "#8156 Fine thread cut E" 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 $\delta 1 = 10 \text{ mm}$, $\delta 2 = 10 \text{ mm}$ 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)

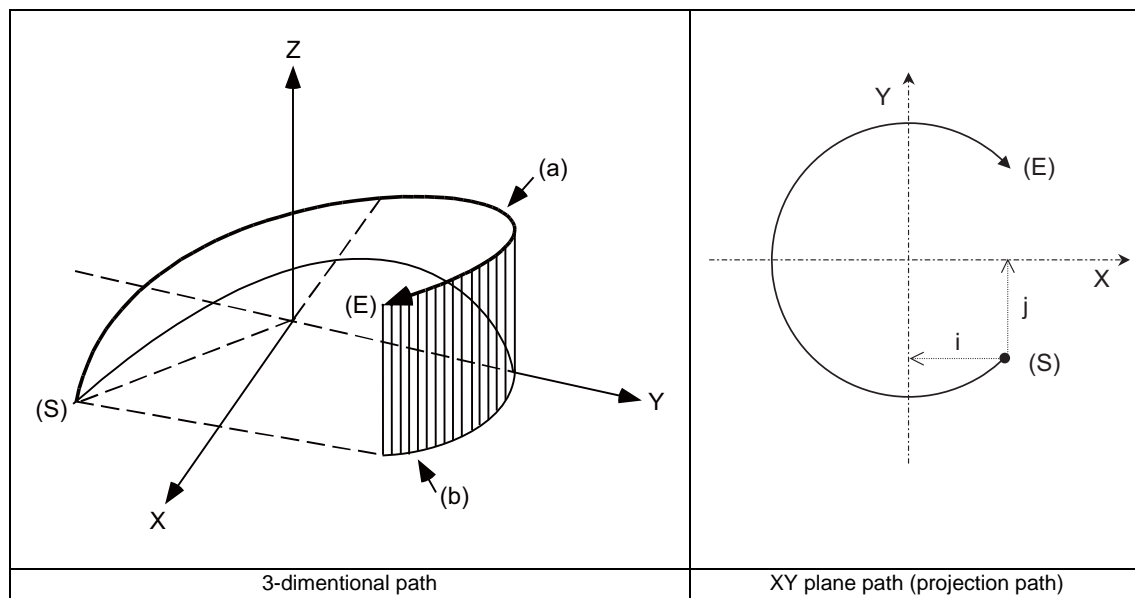
6.7 Helical Interpolation ; G17 to G19, G02, G03



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
- (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.

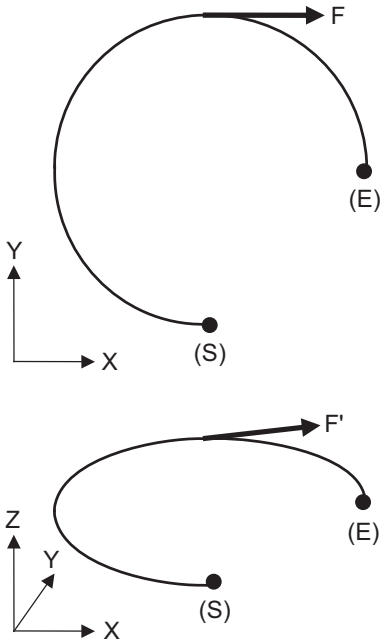


Detailed description

Speed designation during the helical interpolation

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

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

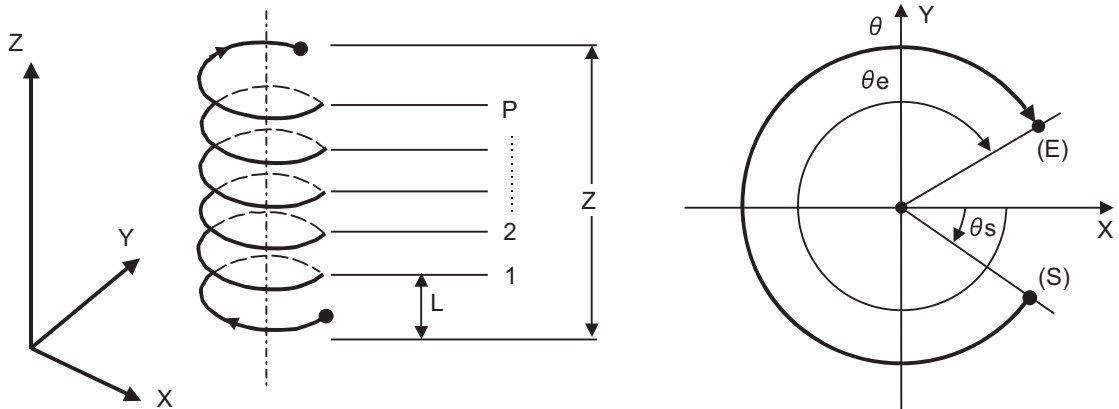


(S) Start point
(E) End point

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

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

Normal speed designation



(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.

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

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

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

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

(Note) The pitch No. P command range is 0 to 9999.
 The pitch No. designation (P command) cannot be made with the R-specified arc.

(5) Plane selection

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

XY plane circular, Z axis linear

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

ZX plane circular, Y axis linear

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

YZ plane circular, X axis linear

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

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 described 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.

Arc plane element speed designation

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

(Example)

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

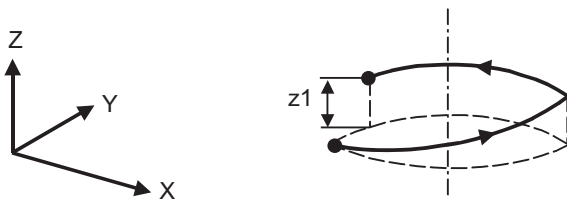
When the arc plane element speed designation is selected, only the helical interpolation speed command is converted to the speed commanded with the arc plane element. The other linear and arc commands operate as normal speed commands.

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



Program example

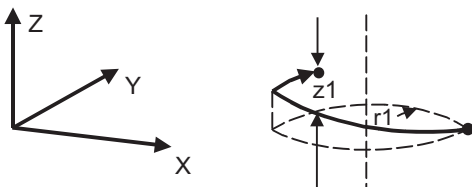
(Example 1)



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

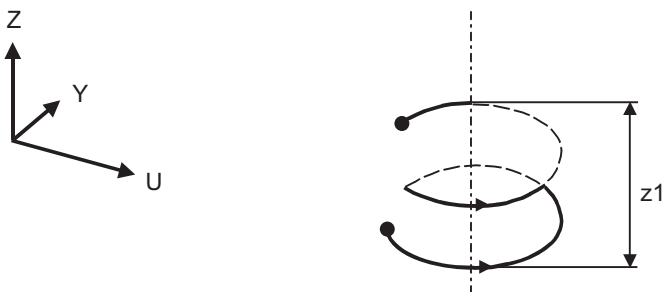
(Note) If pitch No. is 0, address P can be omitted.

(Example 2)



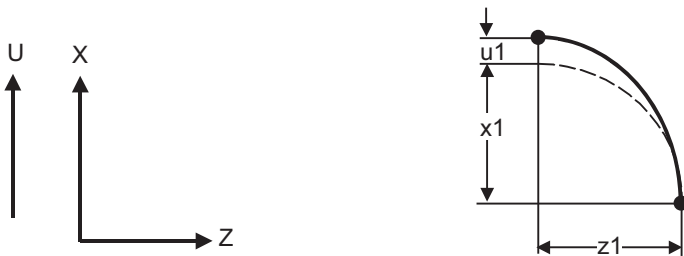
G17 ;	XY plane
G02 Xx1 Yy1 Zz1 Rr1 Ff1;	XY plane arc, Z axis linear

(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
----------------------------------	-----------------------------

(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.



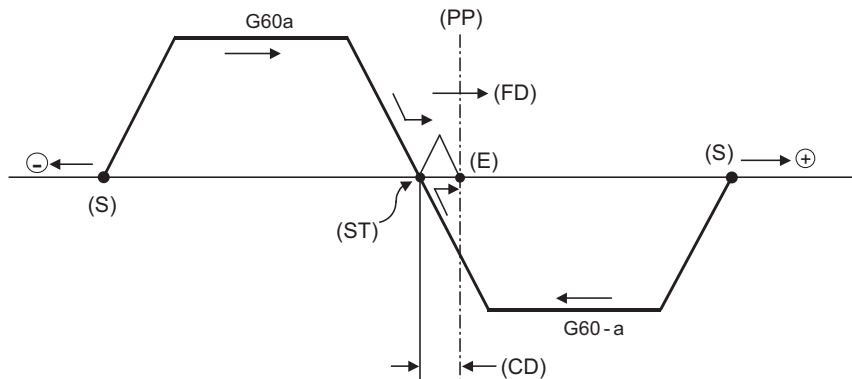
Command format

G60 X__ Y__ Z__ α __; ... Unidirectional positioning



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.

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 10000000 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 numerals.
The cutting feedrate is valid for the G01, G02 and G03 commands.

Examples Feed per minute (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 μ m)

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

(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, but the acceleration/deceleration method follows the rapid traverse setting.)

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



Detailed description

- (1) To validate the F1-digit feed, set the parameter "#1079 F1digit" to "1".
- (2) The override function of the feedrate which is set in accordance to the F1-digit is performed by using the 1st manual handle. (Feedrate cannot be changed with the 2nd or 3rd handle.)

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

$$\pm \Delta F = \frac{FM}{K} \times (\pm P)$$

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

- K : Operation constant
(This is the number of FM divisions, and is the calculated constant of the increment/decrement speed per scale of the manual handle pulse generator.)
This is set with the base specification parameter "#1507 F1_K".
- FM : This is the clamp speed of F1 to F5
This is set with the base specification parameter "#1506 F1_FM".
- P : The number of manual handle pulses

Set the corresponding speed of F1 to F5 with the base specification parameters "#1185 spd_F1" to "#1189 spd_F5" respectively.

The increase/reduction range is from "0" to the set value of the parameter "#1506 F1_FM".

The error "M01 operation error 0104" will occur when the feedrate is 0.

- (3) Use of both the F1-digit command and normal cutting feedrate command is possible when the F1-digit is valid.
(Example 1)
F0 Rapid traverse rate
F1 to F5 F1-digit
F6 or more Normal cutting feedrate command
- (4) The F1-digit command is valid in a G01, G02, or G03 modal.
- (5) The F1-digit command can also be used for fixed cycle.
- (6) The number of manual handle pulses is 1 pulse per scale unit regardless of the scaling factor.
- (7) During a F1-digit command, the F1-digit number and F1-digit command signal are output as the PLC 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.

Example of arithmetic constant K setting

When the handle scale unit is to be made ± 10 mm/min.

FM is made 15000 mm/min:

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

Therefore, K is 1500.

The feedrate is made F (1 to 5) ± 10 (mm/min) by rotating the handle by one scale unit.

The conditions for validation of the speed change with the handle

When all the following conditions are established, the speed change is valid with the handle.

- (1) The operation must be automatic (program run, MDI).
- (2) The operation must be in automatic start.
- (3) The operation must be in cutting feed, and F1-digit feedrate command must be issued.
- (4) The F1-digit feed parameter must be ON.
- (5) The F1-digit feed switch must be ON.
- (6) The operation must not be in machine lock.
- (7) The operation must not be in dry run. .
- (8) Upper limit of F1-digit feedrate (#1506) and F1-digit feedrate change constant (#1507) are not zero.

Conversion of the F1-digit feedrate changing with handle operation

Once the F1-digit feedrate is changed with manual handle, the changed feedrate is valid until parameter is changed or the power is turned OFF. Changed feedrate can be checked only during operation. Note that if the parameter "#1246 set18/bit16"(switchover of the F1-digit feedrate changing method) is set to "1", even after the power is turned OFF, operation is kept at the changed feedrate by changing to the feedrate after changing the parameter of F1-digit feedrate. In this case, if the screen is not switched, the changed parameter values are not reflected to display.



Precautions

- (1) If F1-digit feedrate changing valid signal is turned ON when F1-digit feed is commanded and changing feedrate with the manual handle is valid and applied to the following conditions, it is clamped at the upper limit of F1-digit feedrate.
F1-digit feedrate > Upper limit of F1-digit feedrate
- (2) When the F1-digit feedrate is 0 with the handle operation during the F1-digit feedrate command, the alarm "0104 F1-digit feed rate zero" will occur. This alarm can be eliminated when the F1-digit feedrate is set in larger value than 0 with the handle operation.
- (3) When K is 0, the speed is not changed.
- (4) F1 to F5 are invalid in the G00 mode and the rapid traverse rate is established instead.
- (5) If F0 is used in the G02 or G03 mode, the program error (P121) will occur. The error will be eliminated if the F0 command is rewritten.
- (6) When F1. to F5. (with decimal point) are assigned, the 1mm/min to 5mm/min (direct numerical value command) are established instead of the F1-digit feed command.
- (7) 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.
- (8) 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 (°)/mm.
- (9) Even if the F1-digit feed is commanded during feed per revolution (G95), it is executed as a normal F command (direct numerical value command).
- (10) When both the F1-digit feedrate changing and the manual speed command are present, the manual speed command will have the priority.
- (11) In the synchronous tapping command, the speed cannot be changed with the handle.

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

G94/G95 commands are modal commands.

(Ex.) After the G95 command is assigned, the G95 command is valid until the G94 command is next assigned.

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

Metric input

Input Setting unit	B(0.001mm)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)
Minimum command unit	1 (=1.000) (1.=1.000)	1 (=0.001) (1.=1.000)
Command range	0.001 - 1000000.000	0.001 - 999.999

Input Setting unit	C(0.0001mm)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)
Minimum command unit	1 (=1.0000) (1.=1.0000)	1 (=0.0001) (1.=1.0000)
Command range	0.0001 - 1000000.0000	0.0001 - 999.9999

Inch input

Input Setting unit	B(0.0001inch)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)
Minimum command unit	1 (=1.0000) (1.=1.0000)	1 (=0.0001) (1.=1.0000)
Command range	0.0001 - 100000.0000	0.0001 - 999.9999

Input Setting unit	C(0.00001inch)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)
Minimum command unit	1 (=1.00000) (1.=1.00000)	1 (=0.00001) (1.=1.00000)
Command range	0.00001 - 100000.00000	0.00001 - 999.99999

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

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

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, the error "M01 operation error 0105" occurs.
- (4) Feedrate in the machine lock status is the command speed.
- (5) Under dry run conditions, feed per minute applies and movement results at the manual feedrate (mm/min or inch/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.

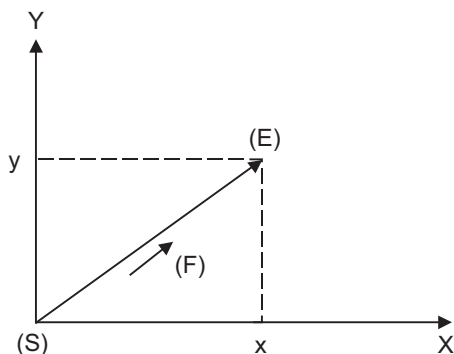


Detailed description

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:



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

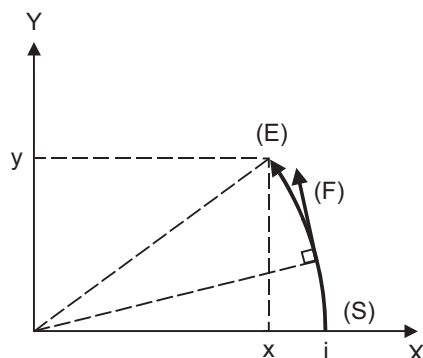
$$f_y = f \times \frac{y}{\sqrt{x^2 + y^2}}$$

- (S) Tool start point
- (E) Tool end point
- (F) Speed in this direction is "f".
- f_x : Feedrate for X axis
- f_y : 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".

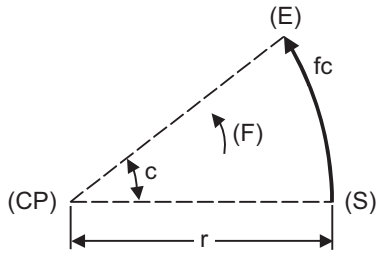
When controlling rotary axes

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" :

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

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

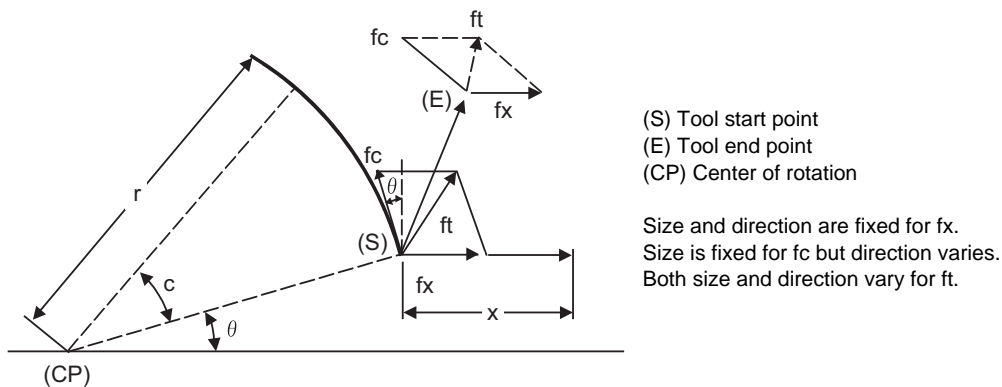
$$f = f_c \times \frac{180}{\pi \cdot r}$$

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":



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

$$f_x = f \times \frac{x}{\sqrt{x^2 + c^2}} \quad \dots\dots (1)$$

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

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

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

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:

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

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

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:

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

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

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

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

"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 Rapid Traverse Constant Inclination Acceleration/Deceleration



Function and purpose

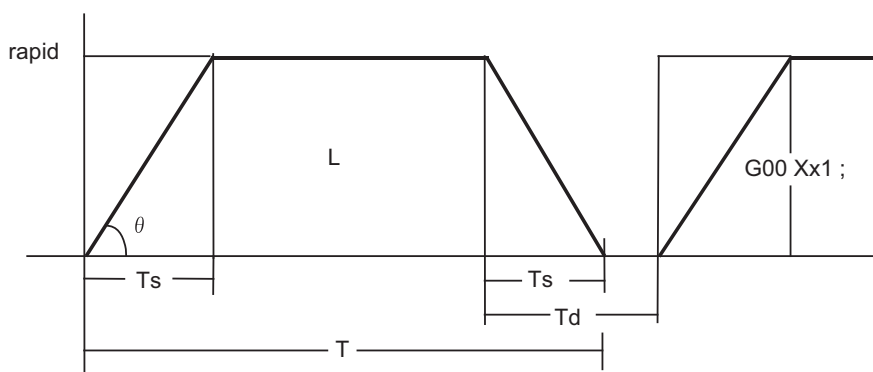
This function performs acceleration and deceleration at a constant inclination during linear acceleration/ deceleration in the rapid traverse mode. Compared to the method of acceleration /deceleration after interpolation, the constant inclination acceleration/deceleration method makes for improved cycle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



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

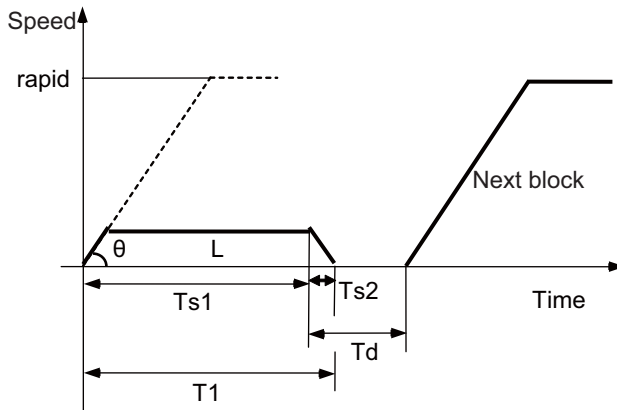
$$T_d = T_s + (0 \sim 14\text{ms})$$

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

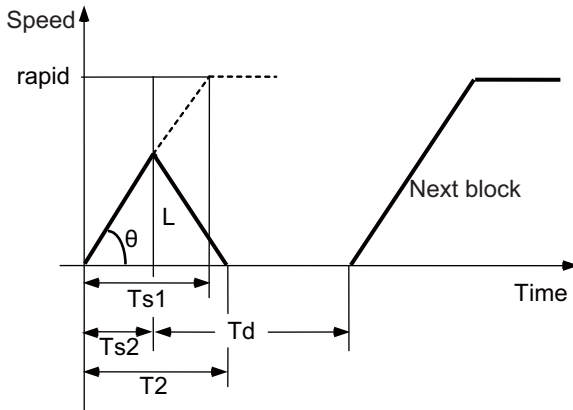
- rapid : Rapid traverse rate
- Ts : Acceleration/deceleration time constant
- Td : Command deceleration check time
- θ : Acceleration/deceleration inclination
- T : Interpolation time
- L : Interpolation distance

[When the interpolation distance is so short that the rapid traverse rate is not achieved]

In case of time-constant acceleration/deceleration:



In case of inclination-constant acceleration/deceleration:



$$T1 = Ts1 + Ts2$$

$$T2 = 2 \times \sqrt{Ts1 \times L / \text{rapid}}$$

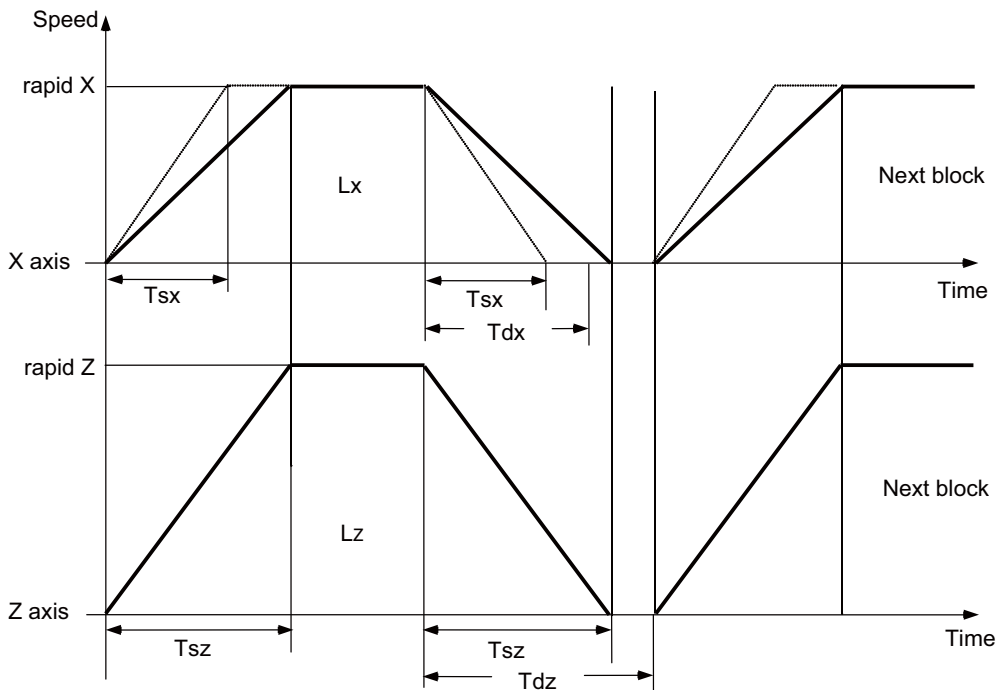
$$Td = T2 / 2 + (0 \text{ to } 14 \text{ ms})$$

$$\theta = \frac{\tan^{-1}(\text{rapid})}{Ts1}$$

- rapid : Rapid traverse rate (Axis specification parameter #2001 rapid)
- Ts1 : Acceleration/deceleration time (Axis specification parameter #2004 G0tL)
- Ts2 : Acceleration/deceleration time to reach the maximum speed
- Td : Command deceleration check time
- θ : Acceleration/deceleration inclination
- T1 : Interpolation time (time-constant acceleration/deceleration)
- T2 : Interpolation time (inclination-constant acceleration/deceleration)
- 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, $T_{sx} < T_{sz}$, $L_x \neq L_z$)]

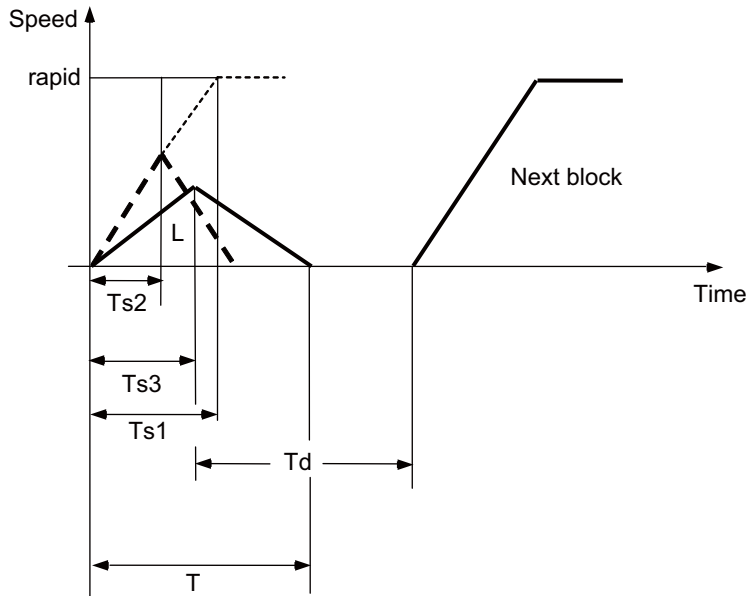


When T_{sz} is greater than T_{sx} ,
 T_{dz} is also greater than T_{dx} , and
 $T_d = T_{dz}$ in this block.

- T_{sx} : X axis acceleration/deceleration time
- T_{sz} : Z axis acceleration/deceleration time
- T_{dx} : X axis commanded deceleration check time
- T_{dz} : Z axis commanded deceleration check time
- L_x : X axis interpolation distance
- L_z : Z axis interpolation distance

- (4) If a minimum time constant for inclination-constant acceleration/deceleration has been set by the parameter, acceleration/deceleration speed is adjusted to prevent the acceleration/deceleration time calculated by interpolation distance from going below the minimum time constant.

[When the interpolation distance is so short that the acceleration/deceleration time is shorter than the minimum time constant for constant inclination acceleration/deceleration]



$$T = 2 \times Ts2$$

$$Td = \frac{T}{2} + (0 \text{ to } 14 \text{ ms})$$

- rapid : Rapid traverse speed (Axis specification parameter #2001 rapid)
- Ts1 : Acceleration/deceleration time (Axis specification parameter #2004 G0tL)
- Ts2 : Acceleration/deceleration time to reach the maximum speed
- Ts3 : Minimum time for inclination-constant acceleration/deceleration (Axis specification parameter #2198 G0tMin)
- Td : Command deceleration check time
- T1 : Interpolation time
- L : Interpolation distance

- (5) Use the rapid traverse time constant changeover request signal to switch the rapid traverse constant. The time constant is switched in the block next to where the rapid traverse time constant changeover request signal is turned ON/OFF.

	Basic rapid traverse time constant (signal OFF)	Rapid traverse time constant for switching (signal ON)
Rapid traverse time constant	#2004 G0tL	#2598 G0tL_2
Rapid traverse constant (primary delay) / 2nd step time constant of soft acceleration/deceleration	#2005 G0t1	#2599 G0t1_2

(Note) When #2598 is "0", use #2004. When #2599 is "0", use #2005.

- (6) 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).
- (7) This function is valid only for G0 (rapid traverse).



Precautions

- (1) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, and "#1219 aux03 bit7: Time constant setting changeover for soft acceleration/deceleration" is set to "1: Accelerating time is obtained by $G0tL+G0t1 (G1tL+G1t1)$ ", acceleration/deceleration speed is adjusted to prevent the sum of the 1st step and 2nd step acceleration/deceleration times from going below the minimum time constant for constant-inclination acceleration/deceleration.
- (2) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, if the acceleration/deceleration time is shorter than $G0tL$ (or $G1tL$), the 2nd step time constant will be reduced by the same rate as the 1st step time constant.
- (3) If a commanded travel distance in a block is small, acceleration/deceleration time becomes quite short when the constant-inclination acceleration/deceleration method is enabled. Although this does contribute to reducing the cycle time, this can also be a cause of machine vibrations. In such case set the minimum time constant for constant-inclination acceleration/deceleration in parameter "#2198 $G0tMin$ ". By this setting, the acceleration/deceleration speed is controlled so that the acceleration/deceleration time should not fall below the preset value.

7.7 Cutting Feed Constant Inclination Acceleration/Deceleration



Function and purpose

This function performs linear acceleration/deceleration at a constant inclination in the cutting feed mode. The inclination-constant acceleration/deceleration method can be more beneficial in reducing cycle time in comparison to the time-constant acceleration/deceleration method.



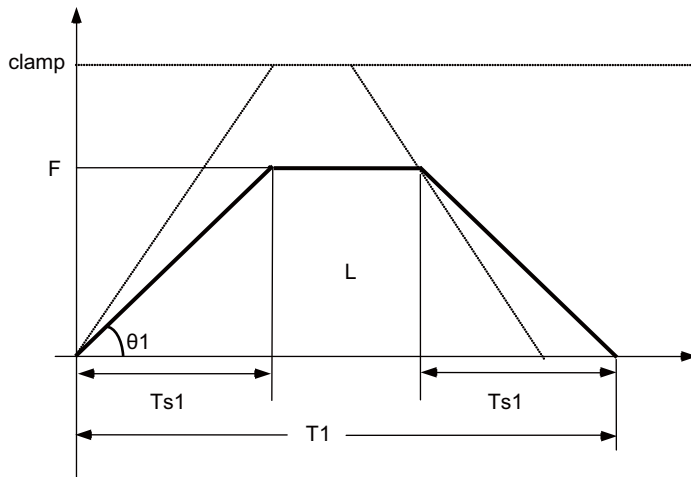
Detailed description

- (1) Cutting feed constant inclination acceleration/deceleration function is effective only when the commanded cutting feed acceleration/deceleration mode is linear method or soft method in a linear interpolation command.
- (2) The program format of linear interpolation when cutting feed constant inclination acceleration/deceleration is executed is the same as when this function is invalid (time constant acceleration/deceleration).

- (3) The acceleration/deceleration patterns in the case where cutting feed constant inclination acceleration/deceleration is performed are as follows.

[When the interpolation distance is long enough for the rapid traverse rate to be achieved]

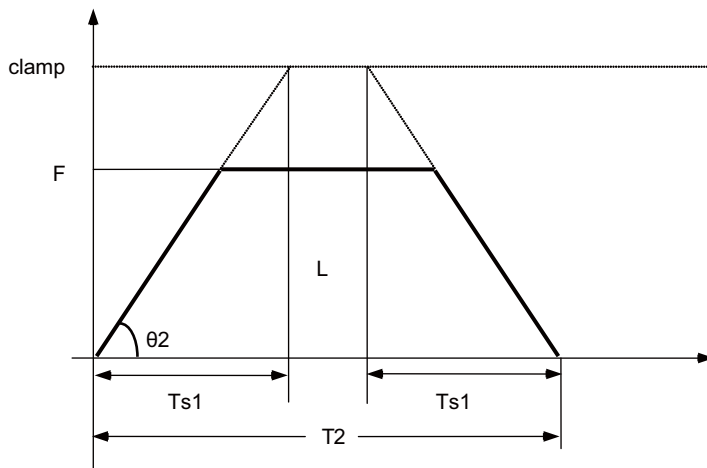
In case of time-constant acceleration/deceleration:



$$T1 = \frac{L}{F} + Ts1 \times 2$$

$$\theta1 = \tan^{-1} \left(\frac{F}{Ts1} \right)$$

In case of inclination-constant acceleration/deceleration:



$$T2 = \frac{L}{F} + \frac{Ts1 \times F}{clamp}$$

$$\theta2 = \tan^{-1} \left(\frac{clamp}{Ts1} \right)$$

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")

F : Cutting feed rate

Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")

$\theta 1$: Acceleration/deceleration inclination (time-constant acceleration/deceleration)

$\theta 2$: Acceleration/deceleration inclination (inclination-constant acceleration/deceleration)

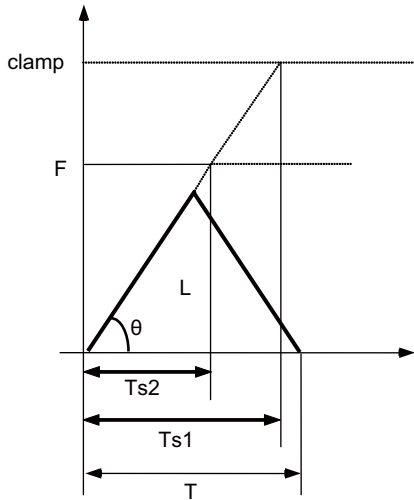
T1 : Interpolation time (time-constant acceleration/deceleration)

T2 : Interpolation time (inclination-constant acceleration/deceleration)

L : Interpolation distance

In the case of time-constant acceleration/deceleration, the acceleration/deceleration inclination is determined by the cutting feed rate. In the case of inclination-constant acceleration/deceleration, it's determined by the maximum cutting feed rate; therefore, the cycle time will be shorter than in the former case.

[When the interpolation distance is so short that the cutting feed rate is not achieved]



$$T = 2 \times \sqrt{Ts2 \times L / \text{clamp}}$$

$$\theta = \tan^{-1} \left(\frac{\text{clamp}}{Ts1} \right)$$

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")

F : Cutting feed rate

Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")

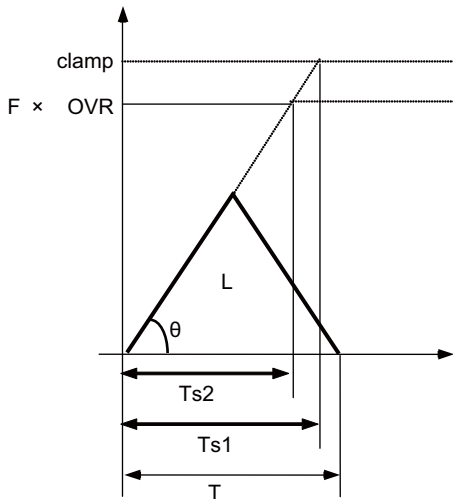
Ts2 : Acceleration/deceleration time to reach the cutting feed rate

θ : Acceleration/deceleration inclination

T : Interpolation time

L : Interpolation distance

[When the interpolation distance is so short that the maximum cutting feed rate is not achieved and the override for cutting feed inclination-constant acceleration/deceleration is activated]



$$T = 2 \times \sqrt{Ts2 \times L \times \text{OVR} / \text{clamp}}$$

$$\theta = \tan^{-1} \left(\frac{\text{clamp}}{Ts1} \right)$$

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")

F : Cutting feed rate

OVR : Maximum override value for cutting feed inclination-constant acceleration/deceleration (Base Specifications Parameter "#1367 G1AccOVRMax")

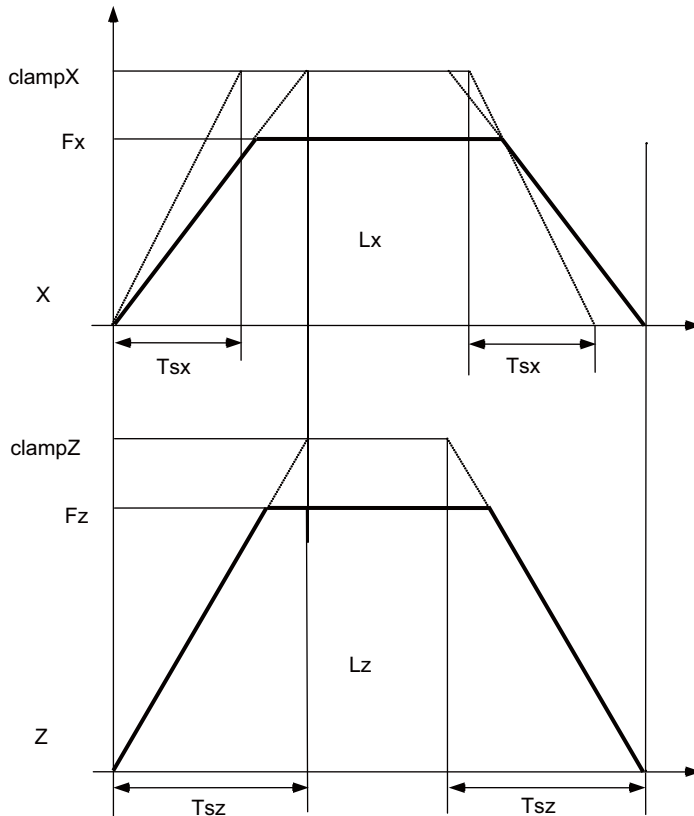
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")

Ts2 : Acceleration/deceleration time to reach the cutting feed rate

θ : Acceleration/deceleration inclination

[2-axis simultaneous interpolation (When $T_{sx} < T_{sz}$, $L_x \neq L_z$)]

When 2-axis simultaneous interpolation is performed during linear interpolation inclination-constant acceleration/deceleration, the acceleration/deceleration time of each axis will be set to the longest one among the acceleration/deceleration times determined by maximum cutting feed rates (axis specification parameter "#2002 clamp"), cutting feed acceleration/deceleration time constant (axis specification parameter "#2007 G1tL"), cutting feed rates (F) and interpolation distances (L) of the simultaneously commanded axes. Should the acceleration/deceleration time constant of an axis be greater than the cutting feed acceleration/deceleration time constant (axis specification parameter "#2007 G1tL"), that axis accelerates/decelerates in accordance with the cutting feed acceleration/deceleration time constant (axis specification parameter "#2007 G1tL").

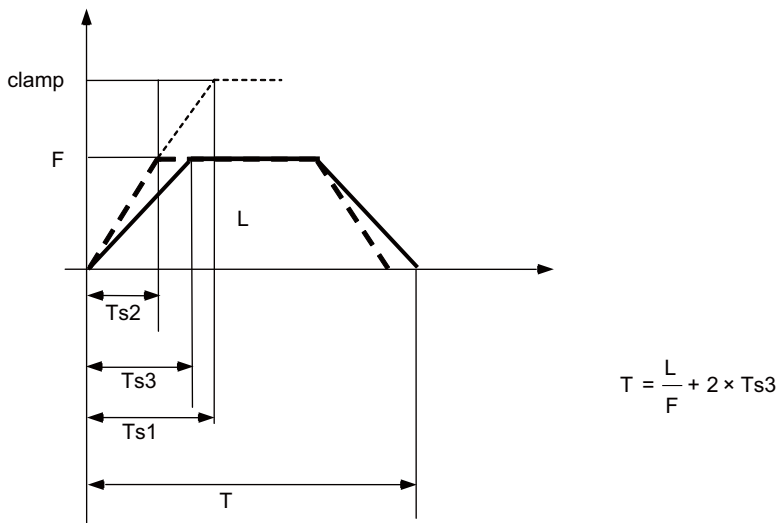


- Tsx : X axis acceleration/deceleration time
- Tsz : Z axis acceleration/deceleration time
- Lx : X axis interpolation distance
- Lz : Z axis interpolation distance
- Fx : X axis feed rate
- Fz : Z axis feed rate

When $T_{sx} < T_{sz}$, the acceleration/deceleration time of the block (T_s) will be equal to the acceleration/deceleration time T_{sz} (Z axis acceleration/deceleration time).

[When the feed rate is so low that the acceleration/deceleration time is shorter than the minimum time constant for inclination-constant acceleration/deceleration]

Acceleration/deceleration speed is adjusted to prevent the acceleration/deceleration time calculated by the cutting feed rate from going below the minimum time constant.



clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")

F : Cutting feed rate

Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")

Ts2 : Acceleration/deceleration time to reach the cutting feed rate

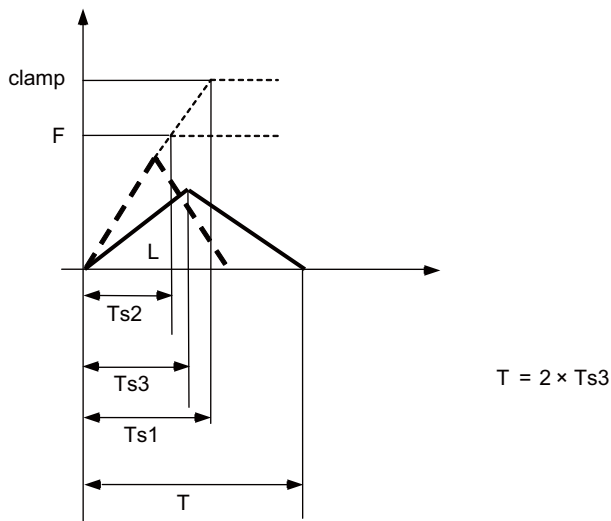
Ts3 : Minimum time for inclination-constant acceleration/deceleration (Axis specification parameter "#2199 G1tMin")

T : Interpolation time

L : Interpolation distance

[When the interpolation distance is so short that the acceleration/deceleration time is shorter than the minimum time constant for inclination-constant acceleration/deceleration]

Acceleration/deceleration speed is adjusted to prevent the acceleration/deceleration time calculated by interpolation distance from going below the minimum time constant.



clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")

F : Cutting feed rate

Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")

Ts2 : Acceleration/deceleration time to reach the cutting feed rate

Ts3 : Minimum time for inclination-constant acceleration/deceleration (Axis specification parameter "#2199 G1tMin")

T : Interpolation time

L : Interpolation distance



Precautions

- (1) If a value greater than 100 (%) is designated for cutting feed override under cutting feed inclination-constant acceleration/deceleration control, the acceleration/deceleration inclination becomes steeper as the feedrate increases. To use the cutting feed override function at a rate higher than 100%, set the parameter "#1367 G1AccOVRMax" accordingly. When the setting of this parameter is between 0 and 99, the override value is handled as 100% even if the specified cutting feed override is over 100%.
- (2) If there are one or more NC control axes that are set to soft acceleration/deceleration for G1, the parameter "#1367 G1AccOVRMax" setting will be ignored and the cutting feed override value is handled as 100%.
- (3) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, and "#1219 aux03 bit7: Time constant setting changeover for soft acceleration/deceleration" is set to "1: Accelerating time is obtained by $G0tL+G0t1 (G1tL+G1t1)$ ", acceleration/deceleration speed is adjusted to prevent the sum of the 1st step and 2nd step acceleration/deceleration times from going below the minimum time constant for inclination-constant acceleration/deceleration.
- (4) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, if the acceleration/deceleration is shorter than $G0tL$ (or $G1tL$), the 2nd step time constant will be reduced by the same rate as the 1st step time constant.
- (5) If the commanded travel distance in the block is small or the commanded linear interpolation (G1) feed rate is low, acceleration/deceleration time becomes quite short when the inclination-constant acceleration/deceleration method is enabled. Although this does contribute to reduce the cycle time, this can also be a cause of machine vibrations. Such vibrations can be prevented by setting the minimum time constant for inclination-constant acceleration/deceleration in parameter "#2198 G0tMin". By this setting, the acceleration/deceleration speed is controlled so that the acceleration/deceleration time should not fall below the preset value.

7.8 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.

A deceleration check is performed when the G09 (exact stop check) command has been designated in the same block. The G09 command is unmodal.

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

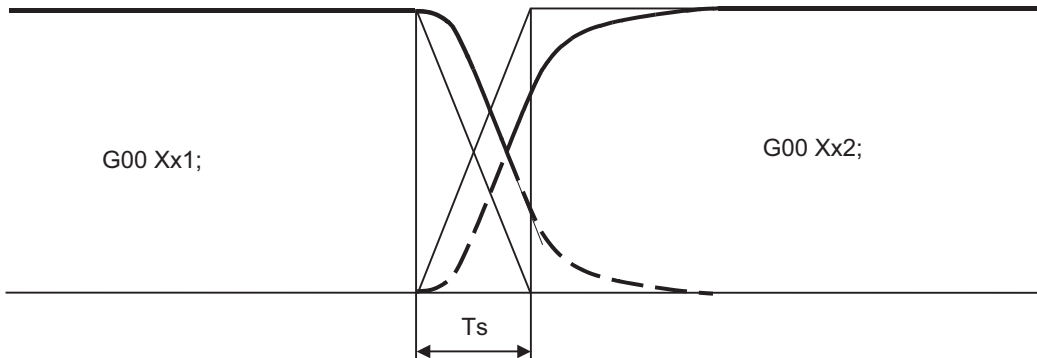
G09 ; ... Exact stop check

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

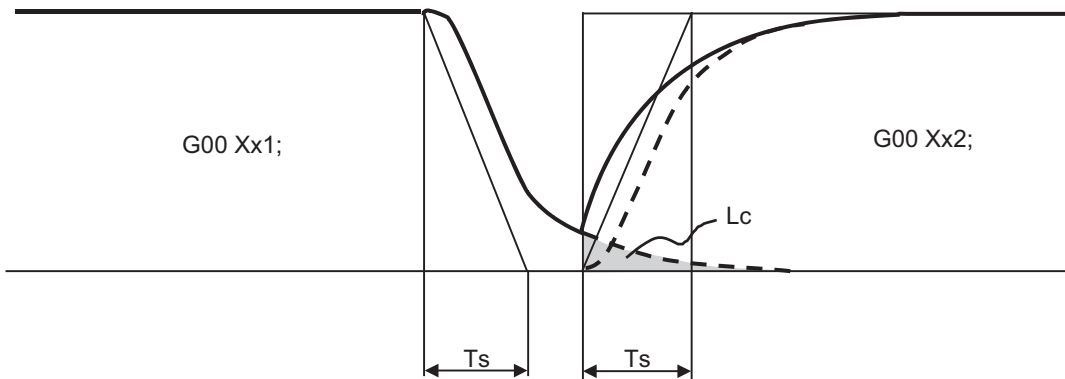


Detailed description

[With continuous cutting feed]



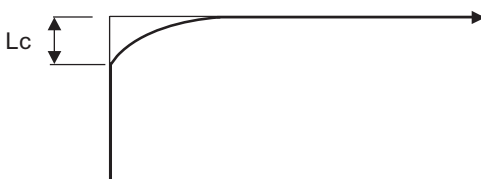
[With cutting feed in-position check]



Ts : Cutting feed acceleration/deceleration time constant

Lc : In-position width

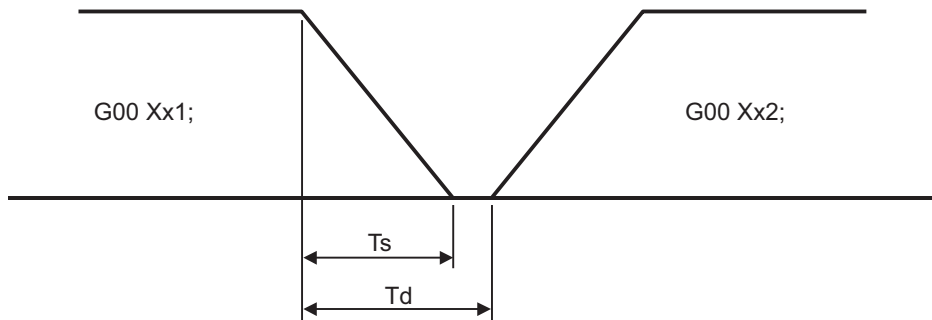
As shown in the figure above, the remaining distance (shaded area in the above figure) of the previous block when the next block is started can be set into the servo parameter "#2224 sv024" as the in-position width "Lc". The in-position width is designed to reduce the roundness at the workpiece corners 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

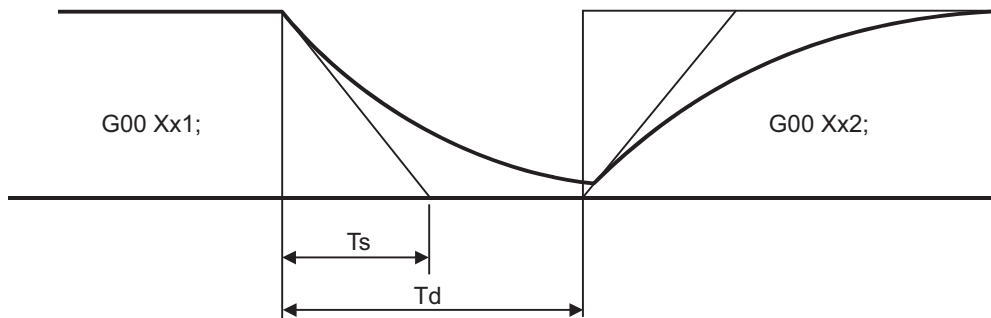
- (1) With linear acceleration/deceleration



TS: Acceleration/deceleration time constant

Td: Deceleration check time $T_d = T_s + (0 \text{ to } 14\text{ms})$

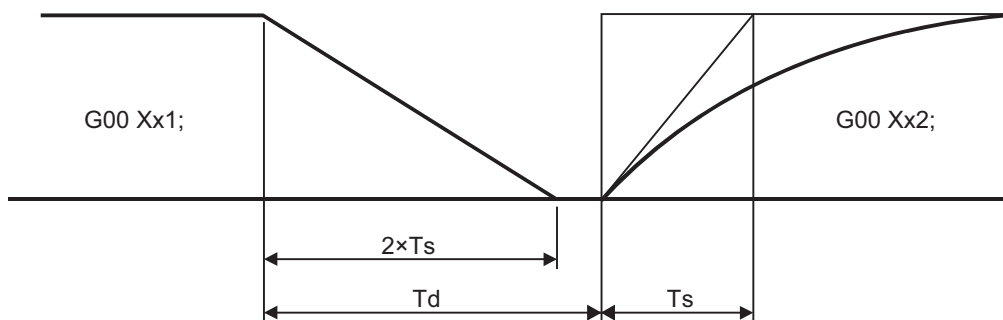
- (2) With exponential acceleration/deceleration



TS: Acceleration/deceleration time constant

Td: Deceleration check time $T_d = 2 \times T_s + (0 \text{ to } 14\text{ms})$

- (3) With exponential acceleration/linear deceleration



TS: Acceleration/deceleration time constant

Td: Deceleration check time $T_d = 2 \times T_s + (0 \text{ to } 14\text{ms})$

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.

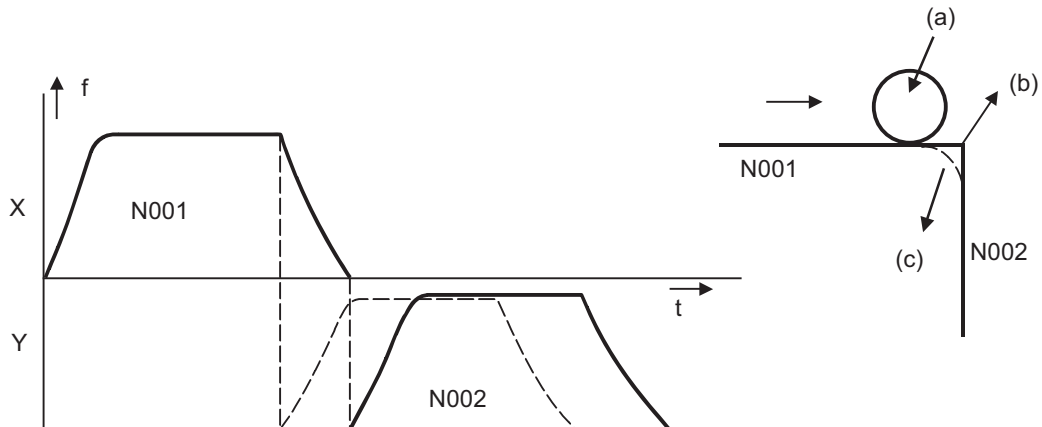
- (Note 1) To execute exact stop check in a fixed cycle cutting block, insert command G09 into the fixed cycle subprogram.



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

7.9 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.

- 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.

7.10 Deceleration Check



Function and purpose

The deceleration check is a function that determines the method of the check at the completion of the axis movement block's movement.

The deceleration check includes the in-position check and commanded speed check method.

The G0 and G1 deceleration check method combination can be selected. (Refer to section "Deceleration check combination".)

With this function, the deceleration check in the reverse direction of G1 -> G0 or G1 -> G1 can be changed by changing the parameter setting.

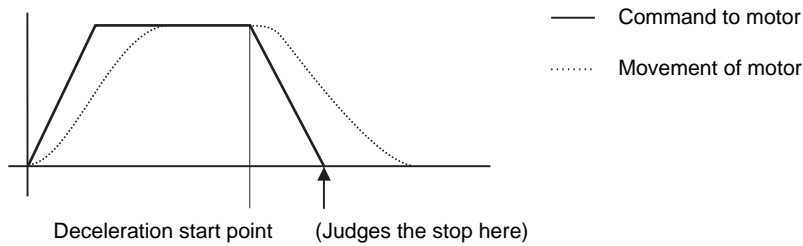


Detailed description

Types of deceleration check

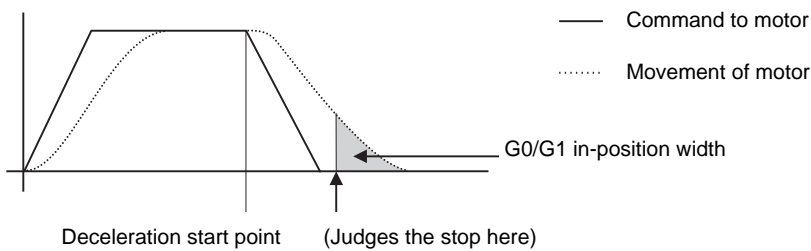
(1) Command speed check

With the commanded speed check, the completion of deceleration is judged when the command to the motor is completed.



(2) In-position check

With the in-position check, the completion of deceleration is judged when the motor moves to the in-position width designated with the parameter.



Designating deceleration check

The deceleration check by designating a parameter includes "deceleration check specification type 1" and "deceleration check specification type 2". The setting is selected with the parameter "#1306 InpsTyp".

- (1) Deceleration check specification type 1 ("#1306 InpsTyp" = 0)

The G0 and G1 deceleration check method can be selected with the base specification parameter deceleration check method 1 (#1193 inpos) and deceleration check method 2 (#1223 AUX07/BIT-1).

Parameter	Rapid traverse command	Parameter	Other than rapid traverse command (G1: other than G0 command)	
			G1+G9 -> XX	G1 -> XX
inpos (#1193)	G0 -> XX (G0+G9 -> XX)	AUX07/BIT-1 (#1223/BIT-1)	G1+G9 -> XX	G1 -> XX
0	Command deceleration check	0	Command deceleration check	No deceleration check
1	In-position check	1	In-position check	

(Note 1) XX expresses all commands.

- (2) Deceleration check specification type 2 ("#1306 InpsTyp" = 1)

Rapid traverse and cutting feed in-position are designated with the "#1193 inpos" parameter.

Parameter	Command block		
	G0	G1+G9	G1
#1193 inpos			
0	Command deceleration check	Command deceleration check	No deceleration check
1	In-position check	In-position check	No deceleration check

(Note 1) "G0" means the rapid traverse, and "G1" means the cutting feed.

7.10.1 G1 -> G0 Deceleration Check



Detailed description

In G1 -> G0 continuous blocks, the parameter "#1502 G0lpfg" can be changed to change the deceleration check in the reverse direction.

	Same direction	Reverse direction
G0lpfg:0		 (a) The acceleration rate is excessive due to the composite speed of G1 and G0.
G0lpfg:1		 Command deceleration



Program example

When there is a deceleration check in the movement of several axes:

(1)

G91 G1 X100. Y100. F4000 ; G0 X-100. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.
---	--

(2)

G91 G1 X100. Y-100. F4000 ; G0 X80. Y100. ;	A deceleration check is carried out, because the Y axis moves in the reverse direction in the program.
--	--

(3)

G90 G1 X100. Y100. F4000 ; G0 X80. Y100. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program. (When the program start position is X0 Y0)
---	--

(4)

G91 G1 X100. Y100. F4000 ; G0 X100. Y100. ;	A deceleration check is not carried out, because both the X axis and the Y axis move in the same direction in the program.
--	--

(5)

G91 G1 X100. Y80. F4000 ; G0 X80. ;	A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Y axis movement command in the program.
--	--

7.10.2 G1 -> G1 Deceleration Check



Detailed description

In G1 -> G1 continuous blocks, the parameter "#1503 G1lpfg" can be changed to change the deceleration check of the reverse direction.

	Same direction	Reverse direction
G1lpfg:0		
G1lpfg:1		



Program example

When there is a deceleration check in the movement of several axes:

(1)

G91 G1 X100. Y100. F4000 ; G1 X-100. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.
---	--

(2)

G91 G1 X100. Y-100. F4000 ; G1 X80. Y100. ;	A deceleration check is carried out, because the Y axis moves in the reverse direction in the program.
--	--

(3)

G90 G1 X100. Y100. F4000 ; G1 X80. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program. (When the program start position is X0 Y0)
---	--

(4)

G91 G1 X100. Y100. F4000 ; G1 X100. Y100. ;	A deceleration check is not carried out, because both the X axis and the Y axis move in the same direction in the program.
--	--

(5)

G91 G1 X100. Y80. F4000 ; G1 X80. ;	A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Y axis movement command in the program.
--	--

7.11 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), tapping mode (G63), or cutting mode (G64) command is issued.



Command format

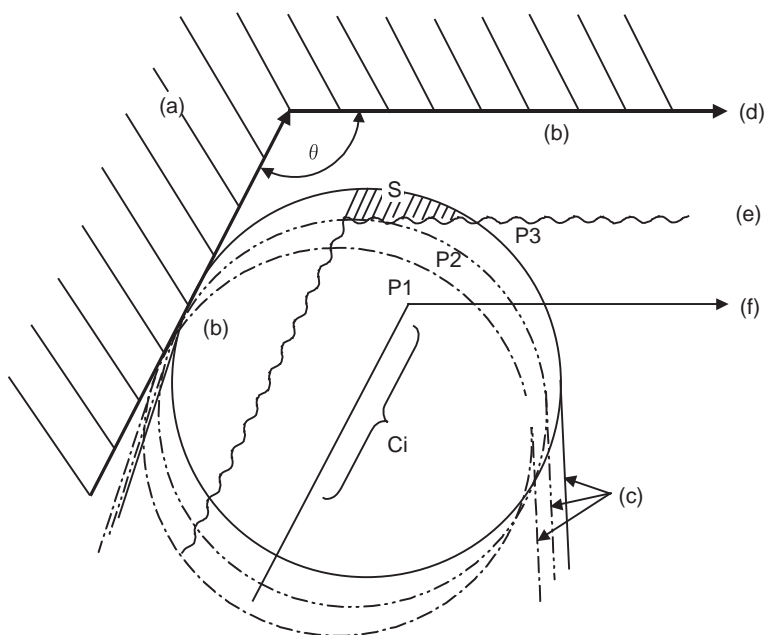
G62 ; ... Automatic Corner Override
--



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]

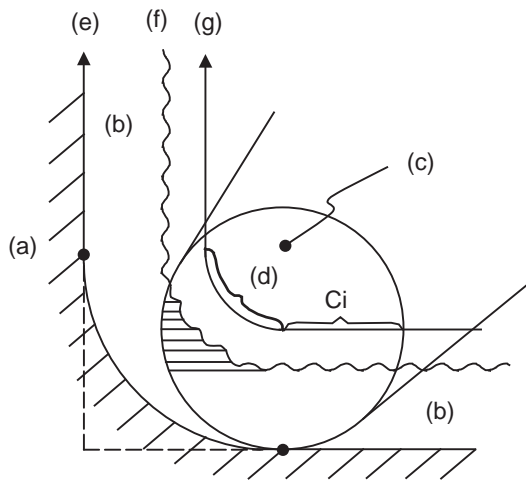
- When automatic corner override is not to be applied :
 When the tool moves in the order of P1 -> P2 -> P3 in the above figure, the machining allowance at P3 increase by an amount equivalent to the area of shaded section S and so that tool load increases.
- When automatic corner override is to be applied :
 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. Refer to the Instruction Manual for details on the setting method.

#	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]

Automatic corner R



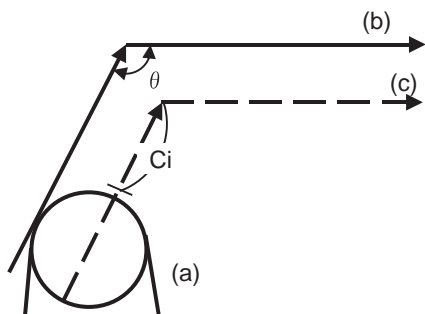
- | | |
|----------------------|-----------------------------|
| (a) Workpiece | (b) Machining allowance |
| (c) Corner R center | (d) Corner R section |
| (e) Programmed path | (f) Workpiece surface shape |
| (g) Tool center path | |

- (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

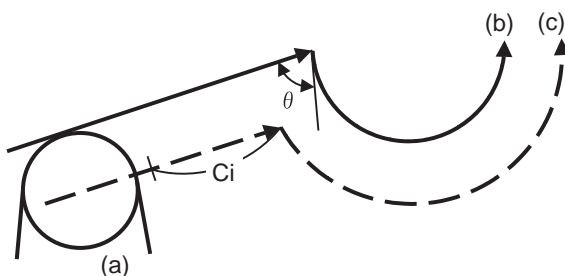
(1) Linear - linear corner



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

The override set in the parameter is applied at Ci.

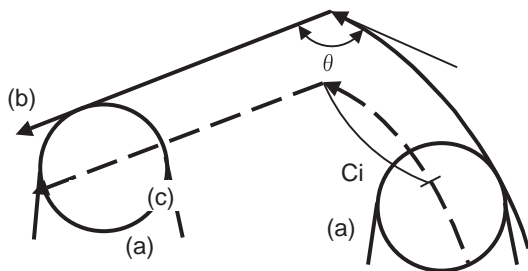
(2) Linear - arc (outside offset) corner



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

The override set in the parameter is applied at Ci.

(3) Arc (inside offset) - linear corner

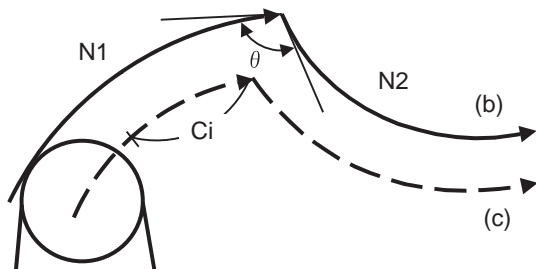


(a) Tool
(b) Program
(c) Tool center

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.



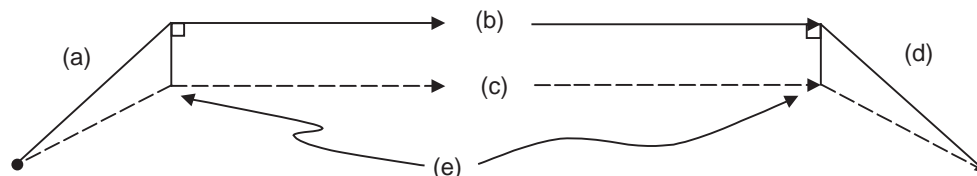
Relation with other functions

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

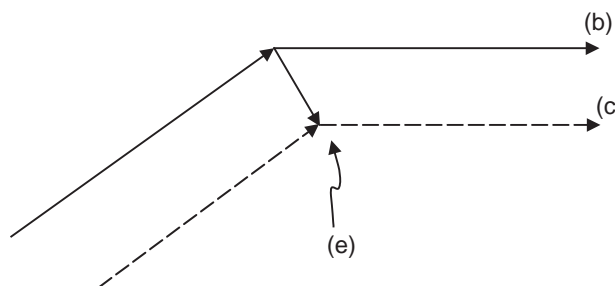
(b) Program

(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.



(G41 X_Z_I_K;)

(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.

7.12 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), 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

7.13 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.

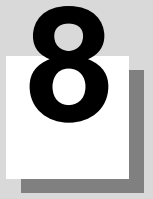
G64 is released by the exact stop check mode (G61), automatic corner override (G62), or tapping mode (G63).

The machine is in the cutting mode status when its power is turned ON.



Command format

G64; ... Cutting mode



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.

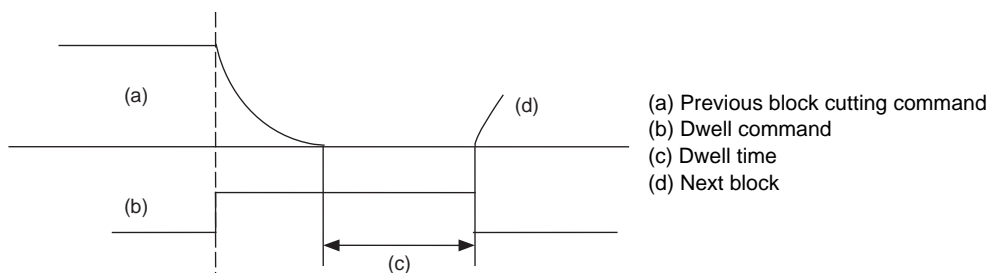


Detailed description

- (1) When designating the dwell time with X, the decimal point command is valid.
- (2) When designating the dwell time with P, the availability of the decimal point command can be selected with the parameter (#8112). When the decimal point command is invalid in the parameter setting, the command below the decimal point issued with P is ignored.
- (3) When the decimal point command is valid or invalid, the dwell time command range is as follows.

Command range when the decimal point command is valid	Command range when the decimal point command is invalid
0 to 99999.999(s)	0 to 99999999 (ms)

- (4) The dwell time setting unit applied when there is no decimal point can be made 1s by setting 1 in the parameter "#1078 Decpt2". This is effective only for X and P for which the decimal command is valid.
- (5) When a cutting command is in the previous block, the dwell command starts calculating the dwell time after the machine has decelerated and stopped. When it is commanded in the same block as an M, S, T or B command, the calculation starts simultaneously.
- (6) The dwell is valid during the interlock.
- (7) The dwell is valid even for the machine lock.
- (8) The dwell can be canceled by setting the parameter "#1173 dwlskp" beforehand. If the set skip signal is input during the dwell time, the remaining time is discarded, and the following block will be executed.





Program example

Command	Dwell time [s]			
	#1078 Decpt2 = 0		#1078 Decpt2 = 1	
	G04P DECIMAL PNT-N	G04P DECIMAL PNT-P	G04P DECIMAL PNT-N	G04P DECIMAL PNT-P
G04 X500 ;	0.5		500	
G04 X5000 ;	5		5000	
G04 X5. ;	5		5	
G04 X#100 ;	1000		1000	
G04 P5000 ;	5		5	5000
G04 P12.345 ;	0.012	12.345	0.012	12.345
G04 P#100 ;	1	1000	1	1000

(Note 1) The above examples are the results under the following conditions.

- Input setting unit 0.001mm or 0.0001inch
- #100 = 1000 ;

(Note 2) "G04P DECIMAL PNT-P" is a control parameter (#8112).

(Note 3) If the input setting unit is 0.0001inch, the X before G04 will be multiplied by 10. For example for "X5. G04 ;", the dwell time will be 50 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.



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.

The number of M commands that can be issued within the same block is determined by the parameter setting (#12005 Mfig).

(Example) G00 Xx Mm1 Mm2 Mm3 Mm4 ;

When five or more commands are issued in a block, only the last four will be valid.

Whether to BCD output or binary output the second miscellaneous function can be selected by a parameter. 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 to 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.)

The secondary miscellaneous function can be issued up to 4 sets in a block. The number of M commands to be issued within a same block is determined by the parameter setting (#12011 Bfig).

Whether to BCD output or binary output the secondary miscellaneous function can be selected by a parameter.

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.



Precautions

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

- Linear angle commands (,A can 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
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

```

F300;
:
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.

```



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
Q	Minimum clamp rotation speed



Detailed description

- 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 S_ Q_ ;".
 - 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

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

- 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

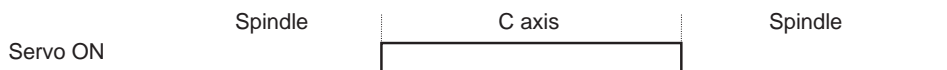
This function enables a spindle (MDS-D or later) to also be used as a C axis (rotary axis) by an external signal.



Detailed description

Spindle/C axis changeover

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



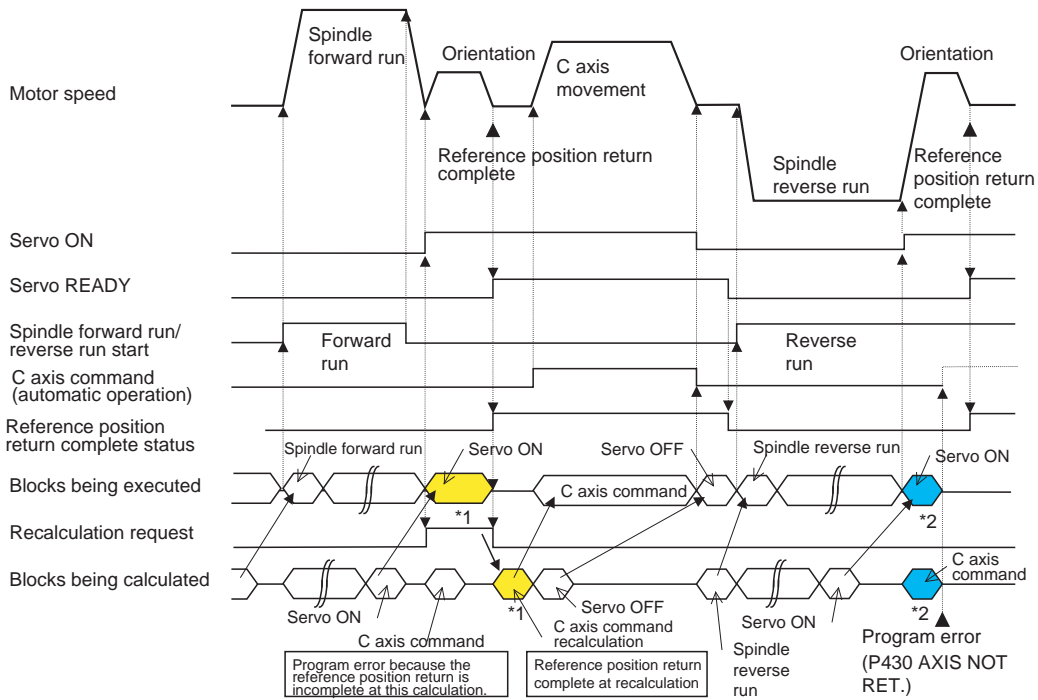
At servo OFF Spindle (C axis control not possible)

At servo ON C axis (spindle control not possible)

The C axis is in a reference position return incomplete state.

- (1) Reference position return state
Reference position return is incomplete when the Z phase has not been passed.
Reference position return is complete when the Z phase has been passed.
- (2) C axis position 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



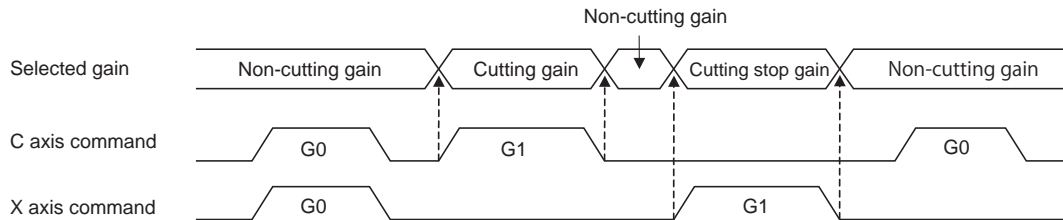
(Note 1) For axis commands, the reference position return complete is checked at calculation. Thus, when the C axis servo ON command and C axis command are continuous, the program error (P430) will occur as shown above in *2.

In response to this kind of situation, the following two processes must be carried out on user PLC, as shown above in *1.

- Input the recalculation request signal with a servo ON command.
- Wait for the completion of the servo ON command until the C axis enters a servo READY state.

C axis gain

The C axis gain is changed over (the optimum gain is selected) by the C axis cutting condition. During C axis cutting feed, cutting gain is applied. During other axis' cutting feed (C axis face turning), non-cutting stop gain is applied. Non-cutting gain is applied in all other cases.



(Note 1) There are 1st to 3rd cutting gains, which are selected with the ladder.

Deceleration check in movement including spindle/C-axis

The deceleration check in a movement command including the spindle/C-axis is as the table described below when the following condition is fulfilled. When the different values are set for the position loop gain in non-cutting mode (spindle parameter #3203 PGCO) and the position loop gain in cutting mode (spindle parameter #3330 PGC1 to #3333 PGC4). That is because a vibration and so on occurs in the machine when the gain is changed during the axis movement.

Parameter	Rapid traverse command	Parameter	Other than rapid traverse command (G1: other than G0 command)	
			G1 -> G0 (G0+G9 -> XX)	G1 -> G1
Inpos (#1193)	G0 -> XX (G0+G9 -> XX)	AUX07/bit1 (#1223/bit1)	G1 -> G0 (G0+G9 -> XX)	G1 -> G1
0	Command deceleration check	0	In-position check (Applicable only to SV024)	No deceleration check
1	In-position check	1		

(Note 1) When G1 command is issued, the in-position check is performed regardless of the deceleration check parameter.

(Note 2) XX expresses all commands.



Precautions and restrictions

- (1) A reference position return cannot be executed by the orientation when there is no Z phase in the detector (PLG, ENC, other).
Replace the detector with one having a Z phase, or if using the detector as it is, set the position control changeover to "After deceleration stop" in the parameters (Spindle parameters, SP129 bitE:1), and set the axis to "Axis without zero point" (Zero point return parameters, noref:1).
- (2) The program error (P430) will occur if a C axis command is issued during servo OFF or during orientation.
- (3) Do not execute a servo OFF during a C axis command.
The remaining C axis commands will be cleared at servo ON.
(If servo OFF is executed during C axis control, the feed will stop and spindle control will occur.)
- (4) If servo ON is executed during spindle rotation, the rotation will stop and C axis control will occur.
- (5) Dog-type reference position return is not possible for the C axis.
Set the reference position return to the orientation method in the parameters (Spindle parameters, SP129 bitE: 0), or set the axis to "Axis without reference position (zero point)" (Zero point return parameters "#2031 noref: 1").

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.

Tool Compensation Functions

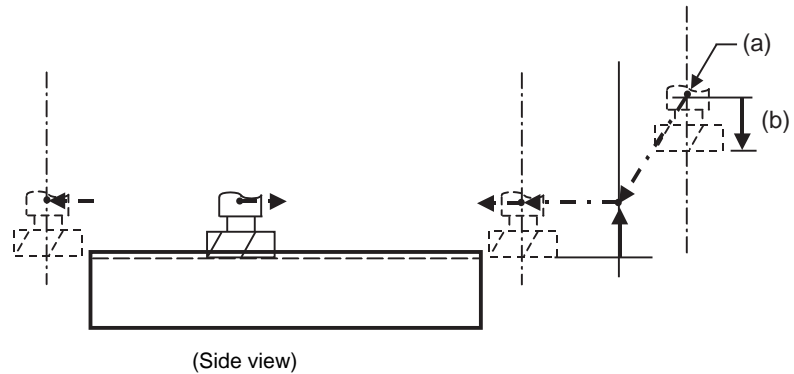
12.1 Tool Compensation



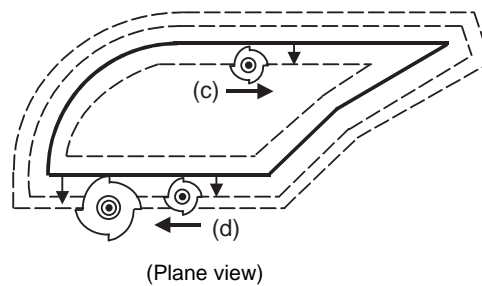
Function and purpose

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

Tool length compensation



Tool radius compensation



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

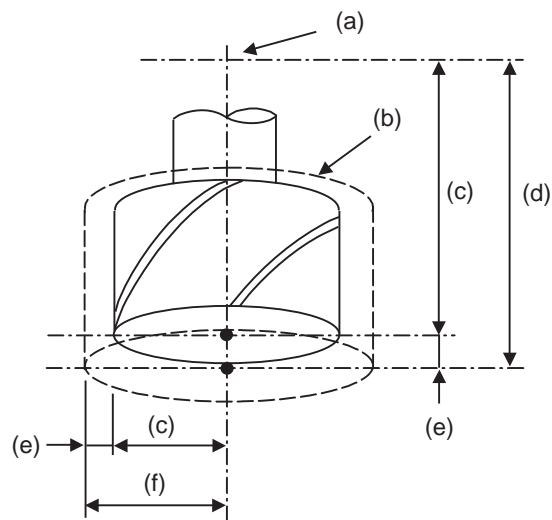
Tool compensation memory

There are two types of tool compensation memories for I and II, setting and selecting the tool compensation amount. (The type used is determined by the machine maker specifications.)

The compensation amount settings are preset with the setting and display unit.

Type I is selected when parameter "#1037 cmdtyp" is set to "1", and type II is selected when set to "2".

Type of tool compensation memory	Classification of length compensation, radius compensation	Classification of shape compensation, wear compensation
Type I	Not applied	Not applied
Type II	Applied	Applied



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

Type I

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

$(D1) = a1, (H1) = a1$

$(D2) = a2, (H2) = a2$

: :

$(Dn) = an, (Hn) = an$

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

Type II

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

The tool length compensation amount is set with H, and the tool radius compensation amount with D.

$(H1) = b1 + c1, (D1) = d1 + e1$

$(H2) = b2 + c2, (D2) = d2 + e2$

: :

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

Compensation No.	Tool length (H)		Tool radius (D)/(Position compensation)	
	Shape compensation amount	Wear compensation amount	Shape compensation amount	Wear compensation amount
1	b1	c1	d1	e1
2	b2	c2	d2	e2
3	b3	c3	d3	e3
:	:	:	:	:
:	:	:	:	:
n	bn	cn	dn	en

 CAUTION

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

Tool compensation No. (H/D)

This address designates the tool compensation No.

- (1) H is used for the tool length compensation, and D is used for the tool position compensation and tool radius compensation.
- (2) The tool compensation No. that is designated once does not change until a new H or D is designated.
- (3) The compensation No. can be commanded once in each block. (If two or more Nos. are commanded, the latter one will be valid.)
- (4) The No. of compensation sets that can be used will differ according to the machine.
For 40 sets: Designate with the H01 to H40 (D01 to D40) numbers.
- (5) If a value larger than this is set, the program error (P170) will occur.
- (6) The setting value ranges are as follows for each No.

The compensation amount for each compensation No. is preset with the setting and display unit.

Setting	Shape compensation amount		Wear compensation amount	
	Metric system	Inch system	Metric system	Inch system
#1003 = B	± 999.999 (mm)	± 99.9999 (inch)	± 999.999 (mm)	± 99.9999 (inch)
#1003 = C	± 999.9999 (mm)	± 99.99999 (inch)	± 999.9999 (mm)	± 99.99999 (inch)

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



Function and purpose

The end position of the movement command for each axis can be compensated by the preset amount when this command is used. A continuity can be applied to the program by setting the actual deviation from the tool length value decided during programming as the compensation amount using this function.



Command format

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

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

G49 Zz ; ... Tool length compensation cancel



Detailed description

Tool length compensation movement amount

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

Z axis movement amount

G43 Zz Hh1 ; z +(lh1)	Compensation in + direction by tool compensation amount
G44 Zz Hh1 ; z -(lh1)	Compensation in - direction by tool compensation amount
G49 Zz ; z -(+)(lh1)	Compensation amount cancel

lh1; Compensation amount for compensation No. h1

Regardless of the absolute value command or incremental value command, the actual end point will be the point compensated by the compensation amount designated for the programmed movement command end point coordinate value.

The G49 (tool length compensation cancel) mode is entered when the power is turned ON or when M02 has been executed.

(Example 1)

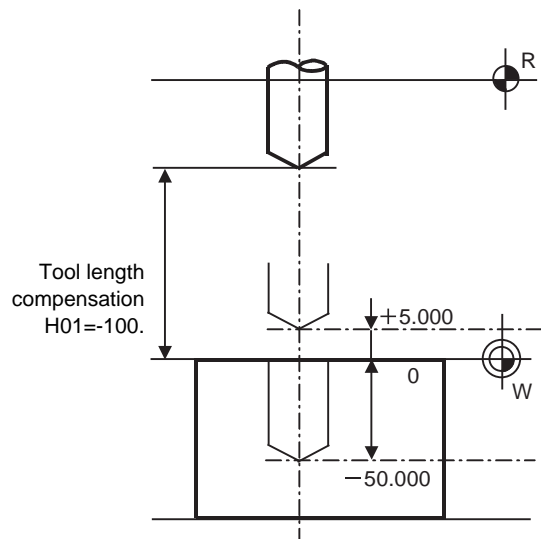
For absolute value command H01=-100000

```
N1 G28 Z0 T01 M06 ;
N2 G90 G92 Z0 ;
N3 G43 Z5000 H01 ;
N4 G01 Z-50000 F500 ;
```

(Example 2)

For incremental value command H01=-100000

```
N1 G28 Z0 T01 M06 ;
N2 G91 G92 Z0 ;
N3 G43 Z5000 H01 ;
N4 G01 Z-55000 F500 ;
```



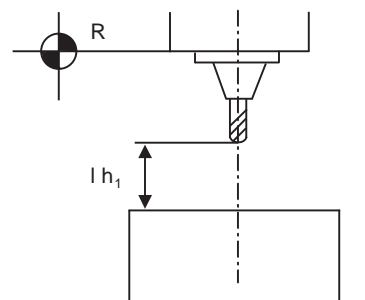
Compensation No.

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

Type I

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 compensation amount, tool radius compensation amount, shape compensation amount or wear compensation amount.

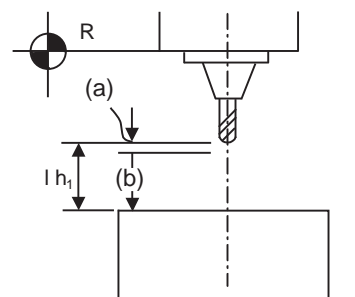


Type II

G43 Hh1;

When the above is commanded, the compensation amount lh1 commanded with compensation No. h1 will be as follows.

lh1: Shape compensation (b) + wear compensation amount (a)



- (2) The valid range of the compensation No. will differ according to the specifications (No. of compensation sets).
- (3) If the commanded compensation No. exceeds the specification range, the program error (P170) will occur.
- (4) Tool length cancel will be applied when H0 is designated.
- (5) The compensation No. commanded in the same block as G43 or G44 will be valid for the following modals.

(Example 3)

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

- (6) If G43 is commanded in the G43 modal, a compensation of the difference between the compensation No. data will be executed.

(Example 4)

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

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

Axis valid for tool length compensation

- (1) When parameter "#1080 Dril_Z" is set to "1", the tool length compensation 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 ;      ..... + compensation to X axis
:
G49 Xx2 ;
:
G44 Yy1 Hh2 ;      ..... - compensation to Y axis
:
G49 Yy2 ;          .....
:
G43 Xx3 Yy3 Zz3 ; ..... Compensation is applied on Z axis
:
G49 ;
```

If the tool length compensation is commanded for the rotary axis, set the rotary axis name for one of the parallel axes.

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

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

Movement during other commands in tool length compensation modal

- (1) If reference point return is executed with G28 and manual operation, the tool length compensation 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 compensation 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 compensation amount is canceled.
When the G54 to G59 workpiece coordinate system is returned to, the position returned to will be the coordinates shifted by the tool compensation amount.

Movement by tool length compensation command

By setting the parameter "#1247/bit0(Movement by tool length compensation command)", it is possible to select whether or not to move the axis by the compensation amount when Tool length compensation (G43/ G44) or Tool length compensation cancel (G49) is independently commanded.

	Without movement (#1247/bit0 = 1)	With movement (#1247/bit0 = 0)
Independent tool length compensation command		
	* The axis does not move when Tool length compensation is commanded independently.	* The axis moves by the compensation amount when Tool length compensation is commanded independently.
Tool length compensation command with movement		
	* The axis moves by the compensation amount when the tool length compensation with travel is commanded.	* The axis moves by the compensation amount when the tool length compensation with travel is commanded.

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__D__; ... Tool radius compensation (left)

G42 X__Y__D__; ... 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 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)

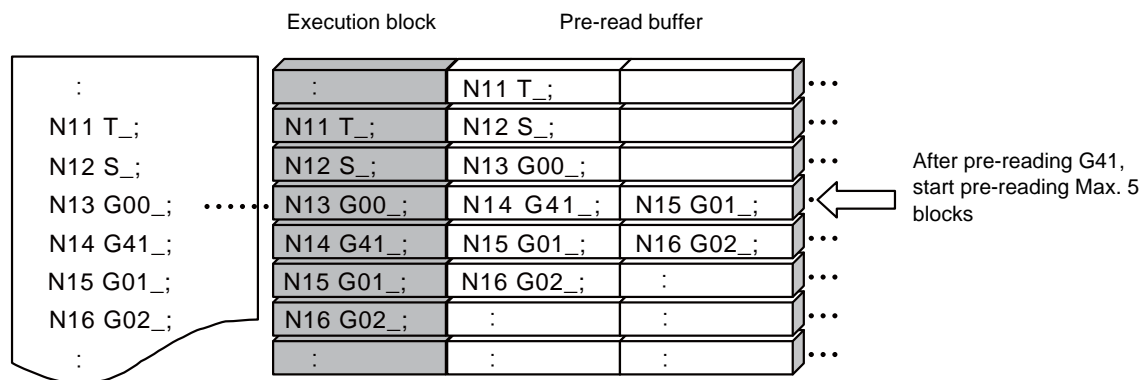
Tool radius compensation starts when all the following conditions are met in the compensation cancel mode.

- (1) The movement command is issued after G41 or G42.
- (2) The tool radius compensation offset No. is $0 < D \leq \text{max. offset No.}$
- (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 will be pre-read.

In compensation mode, too, up to 5 blocks are pre-read and the compensation is arithmetically processed.

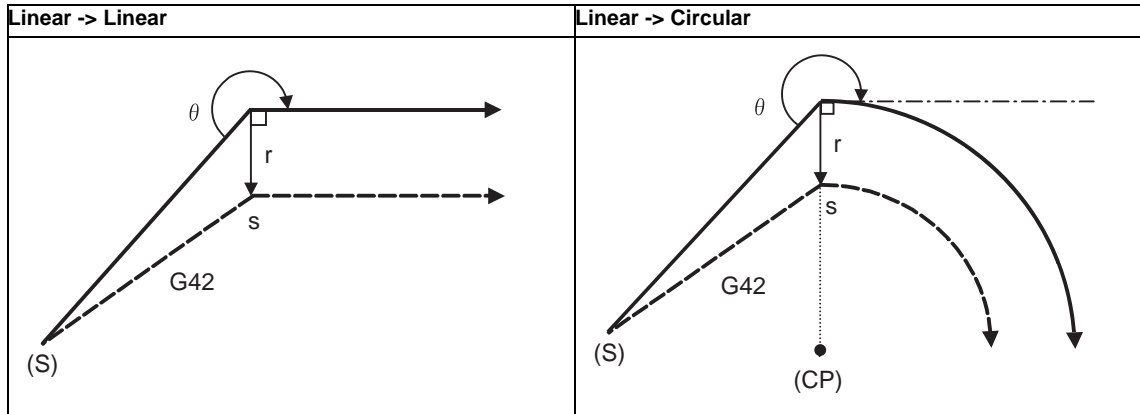
[Control state diagram]



There are two ways of starting the compensation operation: type A and type B. The type can be selected with bit 2 of parameter "#8157 Radius comp type B". This type is used in common with the compensation cancel type.

Start operation for tool radius compensation

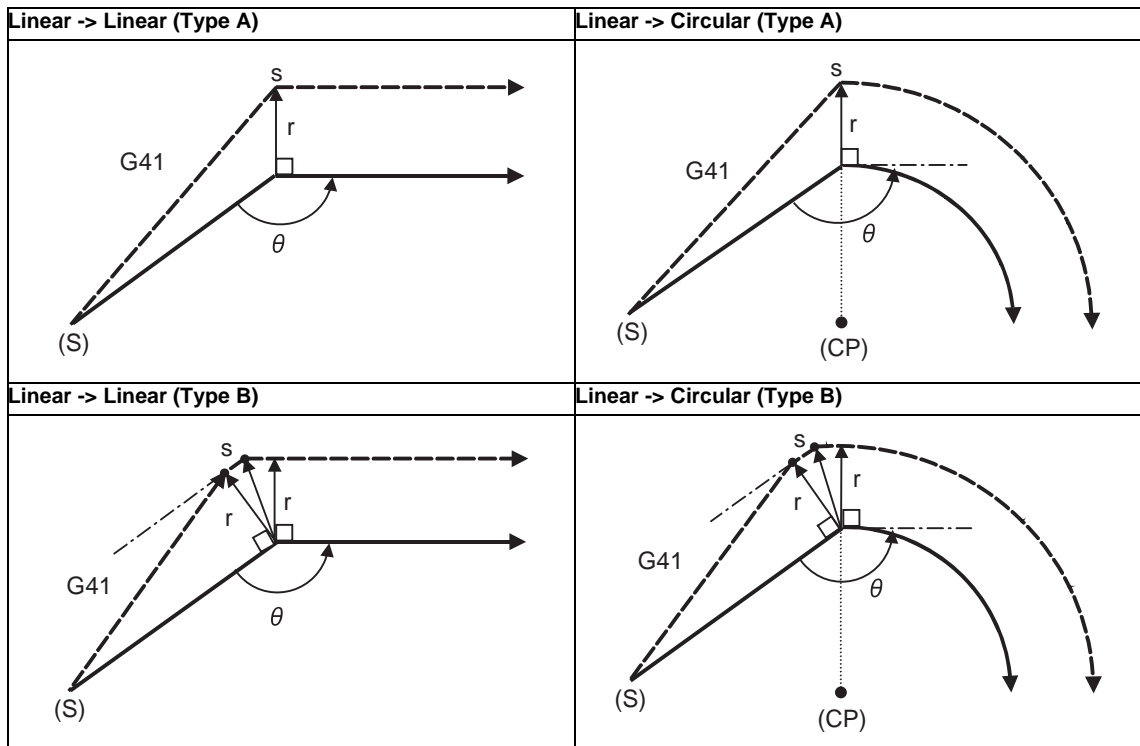
(1) Machining an inside corner



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

—— Program path - - - - Tool center path

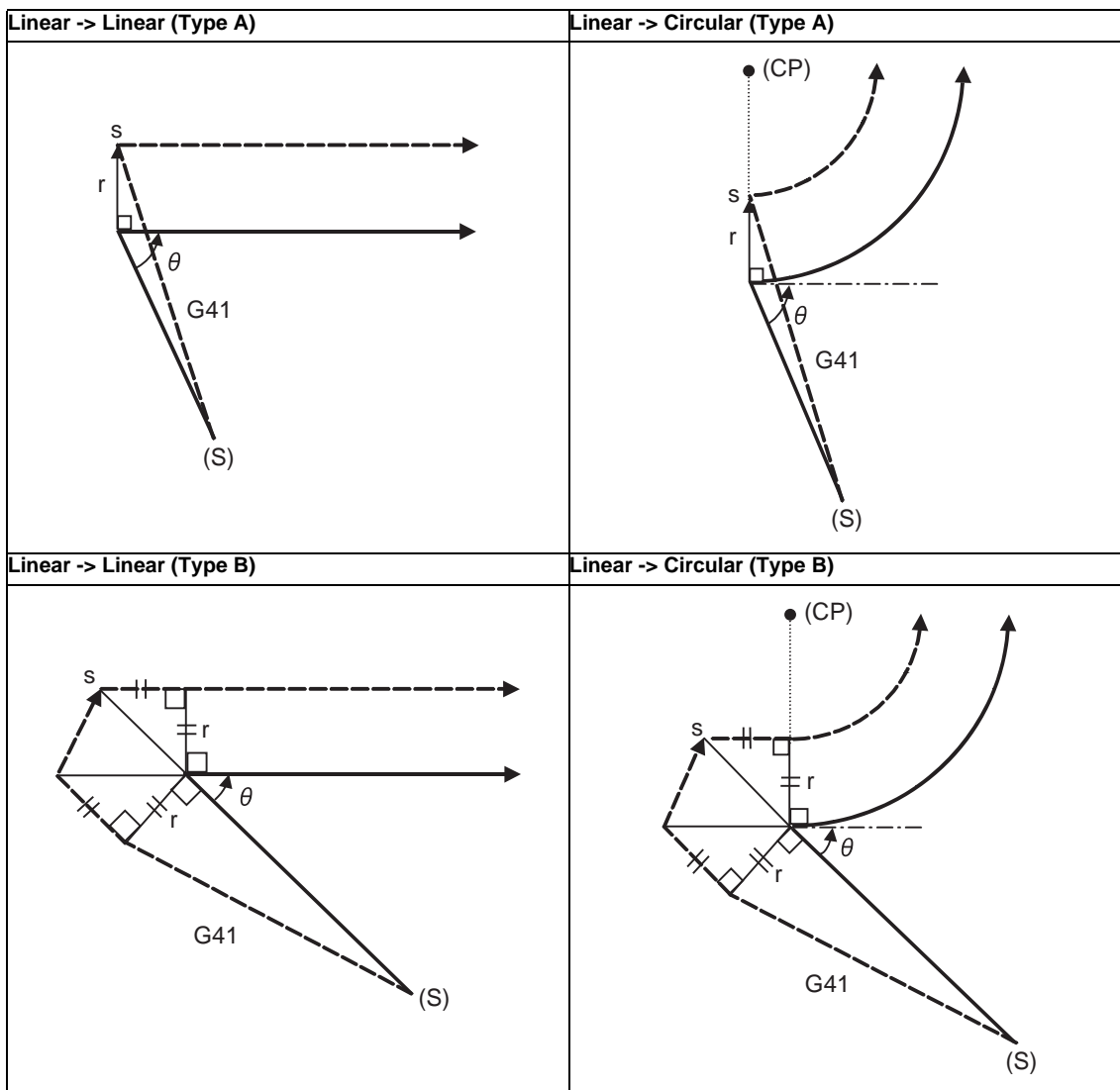
(2) Machining an outside corner (obtuse angle) [$90^\circ \leq \theta < 180^\circ$]



(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) [$\theta < 90^\circ$]



(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

<p>Linear -> Linear ($90^\circ \leq \theta < 180^\circ$)</p>	<p>Linear -> Linear ($0^\circ < \theta < 90^\circ$)</p>
<p>Linear -> Circular ($90^\circ \leq \theta < 180^\circ$)</p>	<p>Linear -> Circular ($0^\circ < \theta < 90^\circ$)</p>
<p>Circular -> Linear ($90^\circ \leq \theta < 180^\circ$)</p>	<p>Circular -> Linear ($0^\circ < \theta < 90^\circ$)</p>
<p>Circular -> Circular ($90^\circ \leq \theta < 180^\circ$)</p>	<p>Circular -> Circular ($0^\circ < \theta < 90^\circ$)</p>

(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

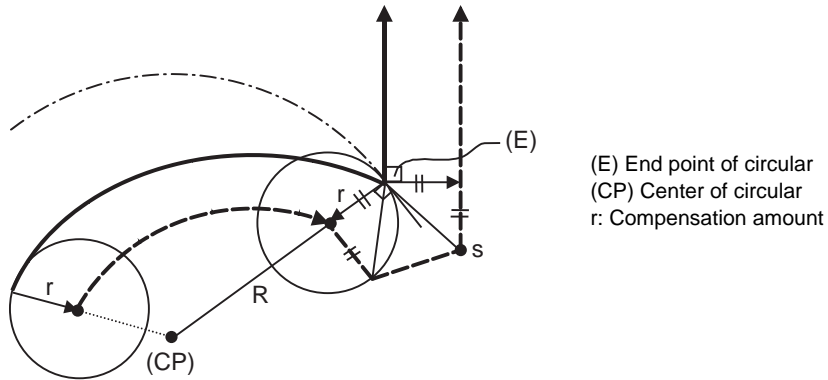
- (3) When the arc end point is not on the arc

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.



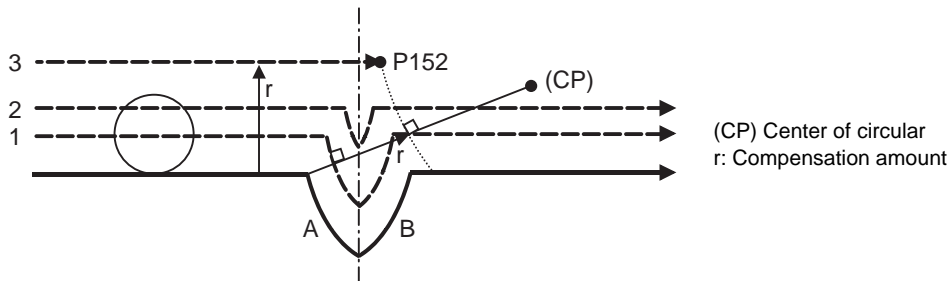
- (4) When the inner intersection point does not exist

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.



— 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, this must be any movement command except a circular command.

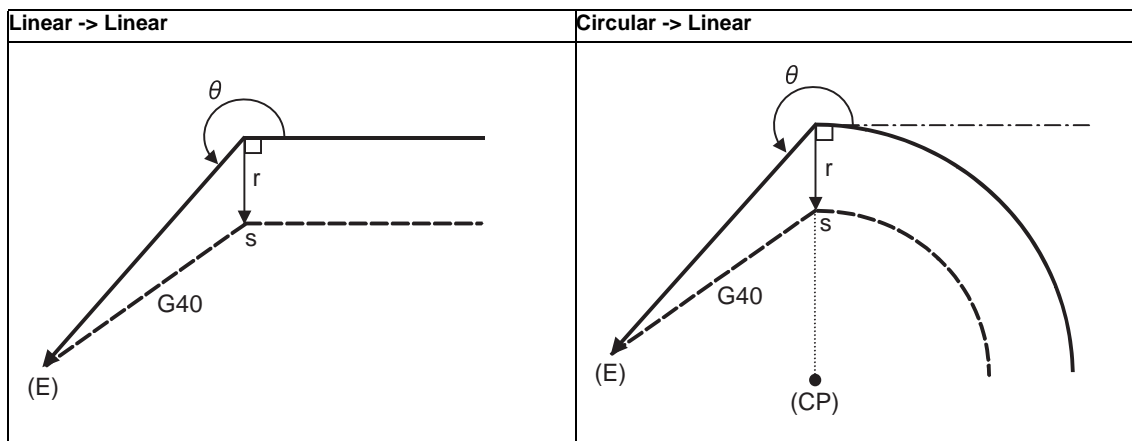
If the compensation is canceled by a circular command, program error (P151) will occur.

- (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



(E) End point

(CP) Center of circular

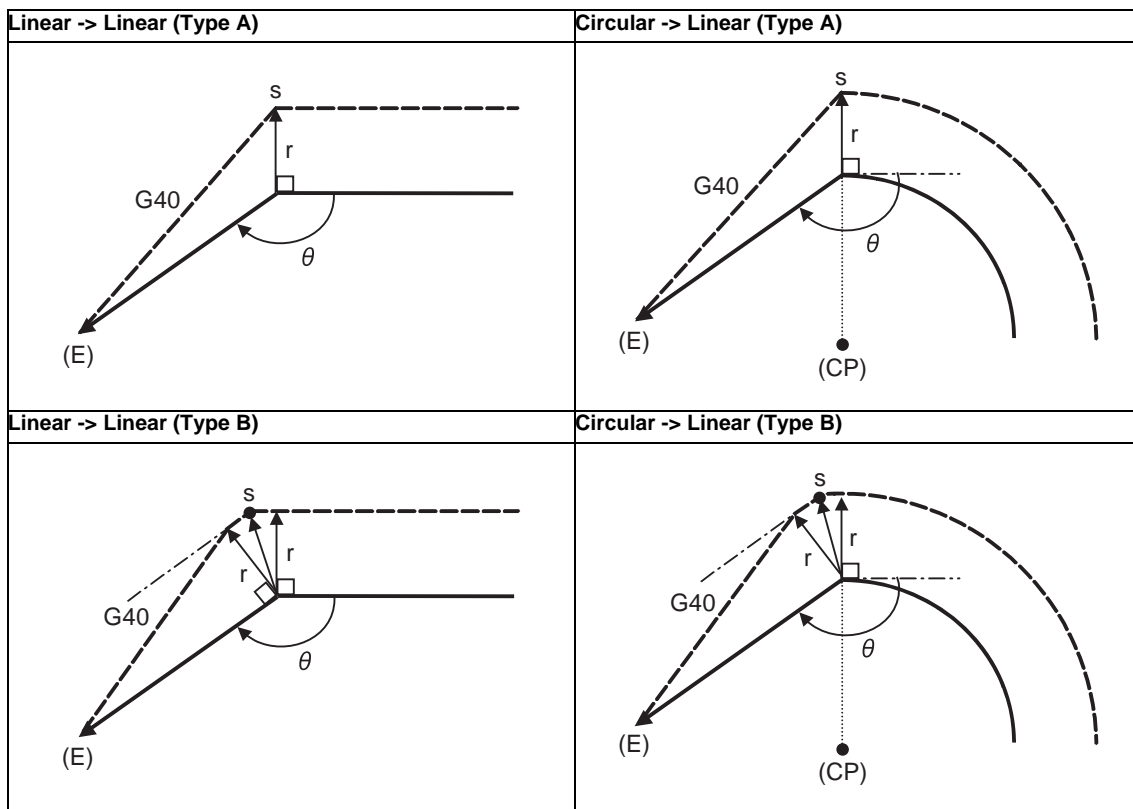
r: Compensation amount

s: Stop point with single block

—— Program path

----- Tool center path

(2) Machining an outside corner (obtuse angle) [$90^\circ \leq \theta < 180^\circ$]



(E) End point

(CP) Center of circular

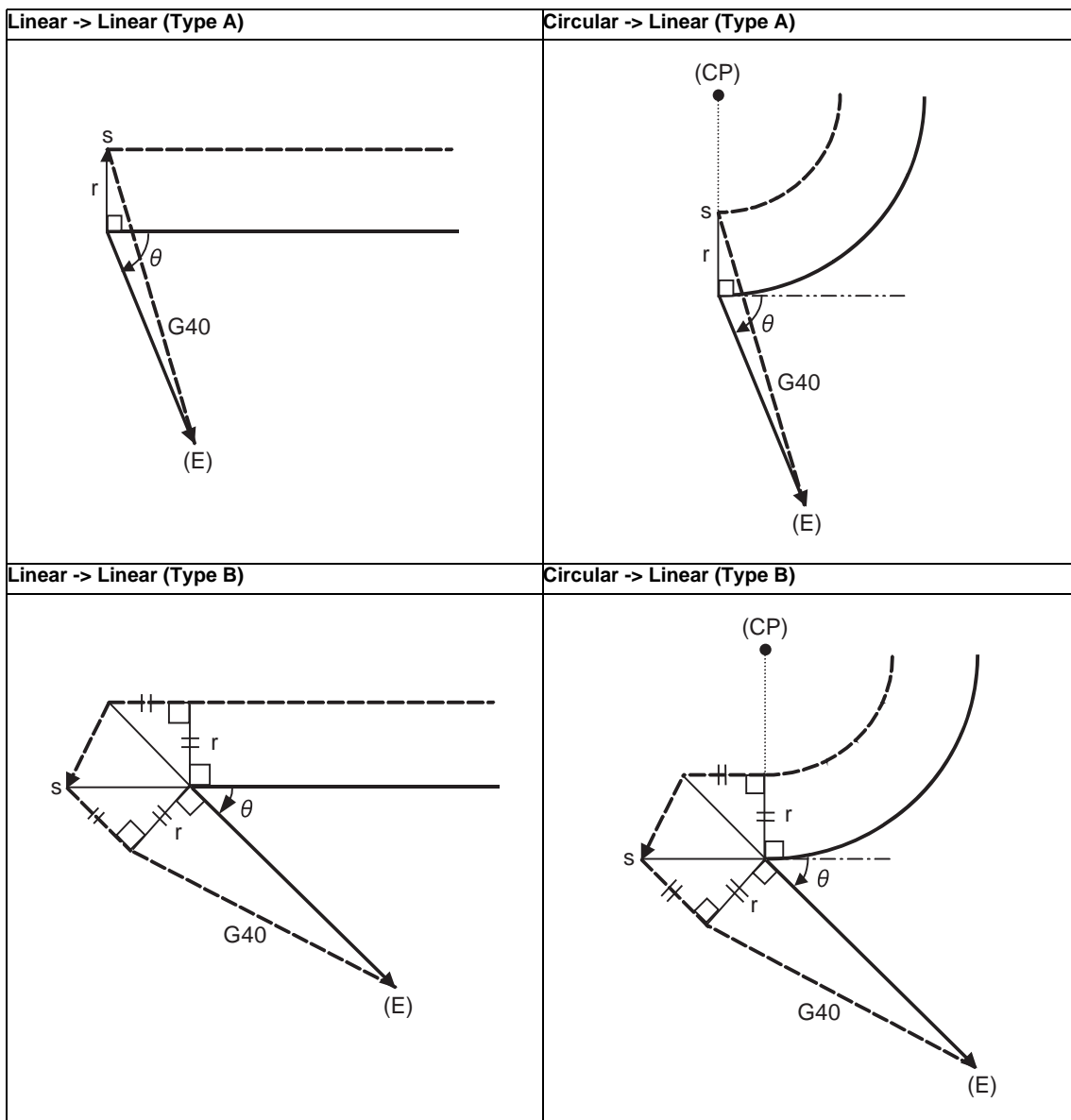
r: Compensation amount

s: Stop point with single block

— Program path

- - - Tool center path

(3) Machining an outside corner (acute angle) [$\theta < 90^\circ$]



(E) End point

(CP) Center of circular

r: Compensation amount

s: Stop point with single block

— Program path

--- Tool center path

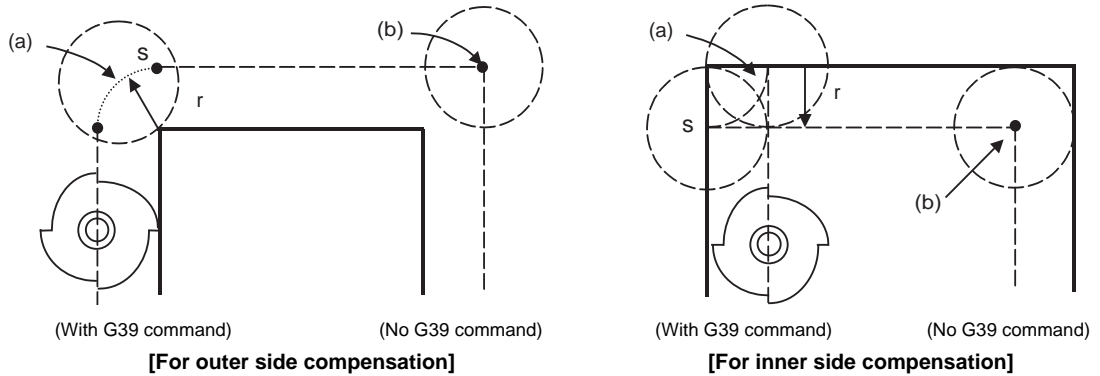
12.3.2 Other Commands and Operations during Tool Radius Compensation



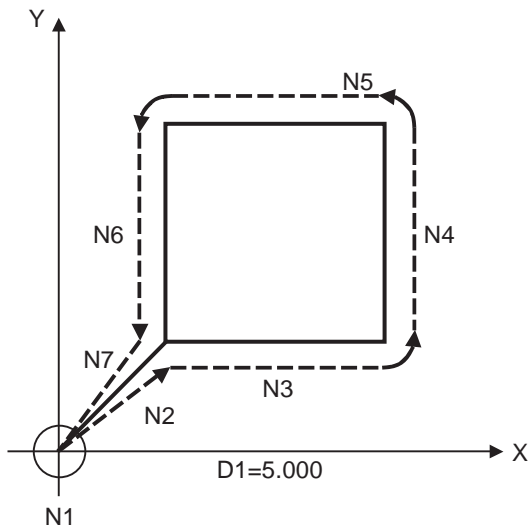
Detailed description

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.



- (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.

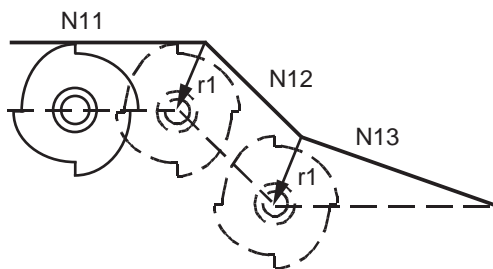
(1) Holding of vector

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.

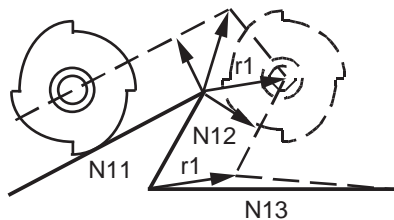
[Holding the inside compensation vector]



N11 G1 Xx11 ;
N12 G38 Xx12 Yy12 ;
N13 G40 Xx13 ;

r1:N11-N12 Vector at N11-N12 block intersection calculation

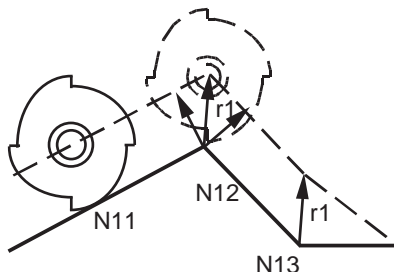
[Holding the outside compensation acute angle]



N11 G1 Xx11 Yy11 ;
N12 G38 Xx12 Yy12 ;
N13 G40 Xx13 ;

r1: Vector at N11-N12 block intersection calculation

[Holding the outside compensation obtuse angle]



N11 G1 Xx11 Yy11 ;
N12 G38 Xx12 Yy12 ;
N13 G40 Xx13 ;

r1: Vector at N11-N12 block intersection calculation

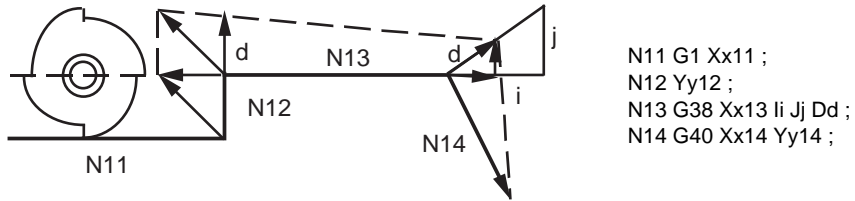
— Program path
- - - Tool center path

(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 Yy12 ;
N13 G38 Xx13 Ii Jj Dd ;
N14 G40 Xx14 Yy14 ;
```

— Program path
 ---- Tool center path

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

(Note) If G38 is commanded in the same block as the circular block (G02/G03) I and J commands, I and J will be handled as the G38 vector, and an error will occur.

Changing the compensation direction during tool radius compensation

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

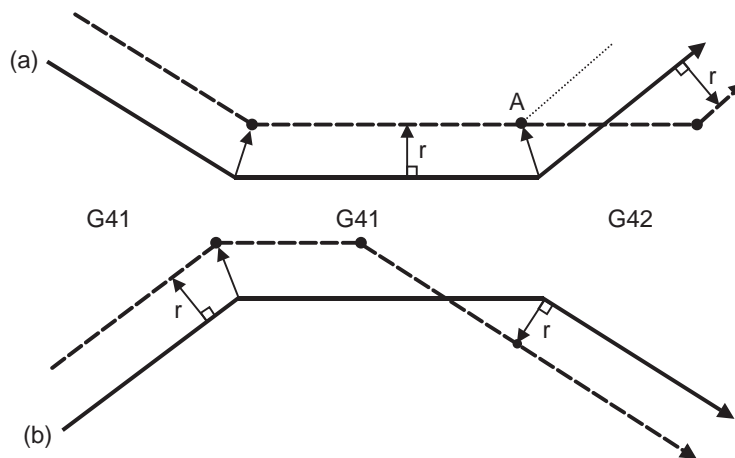
G code	Compensation amount sign +	Compensation amount sign -
G41	Left-side compensation	Right-side compensation
G42	Right-side compensation	Left-side compensation

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.

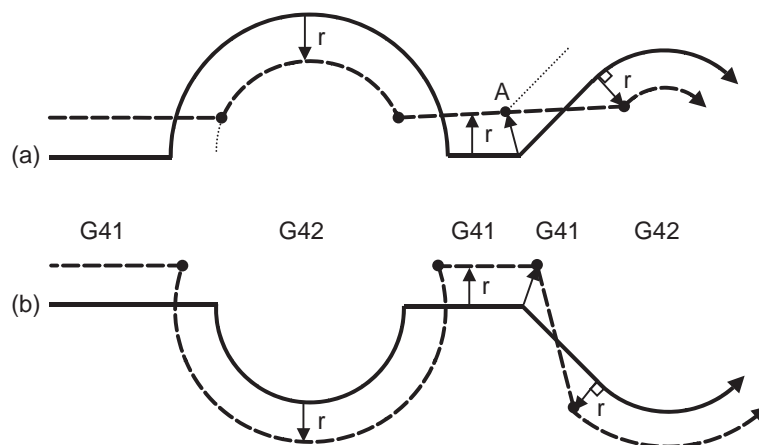
(1) Linear -> Linear

- (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.

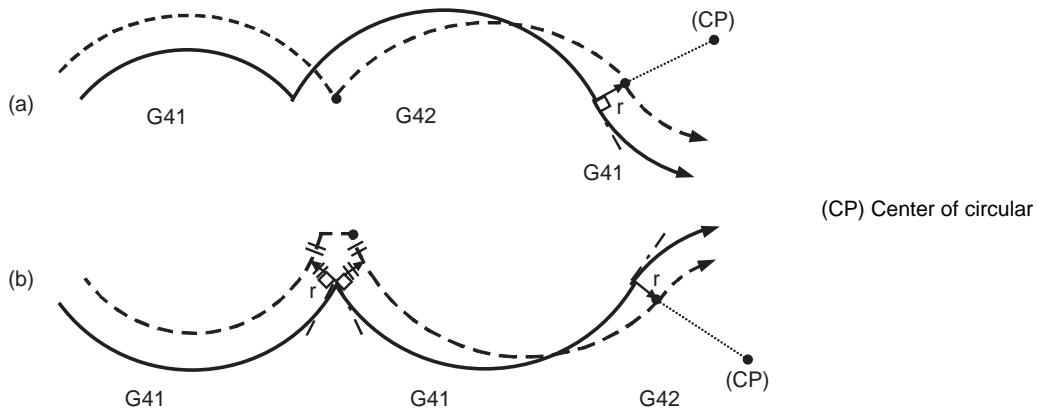


—— Program path

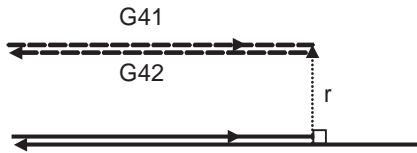
----- Tool center path

(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.



(4) Linear return

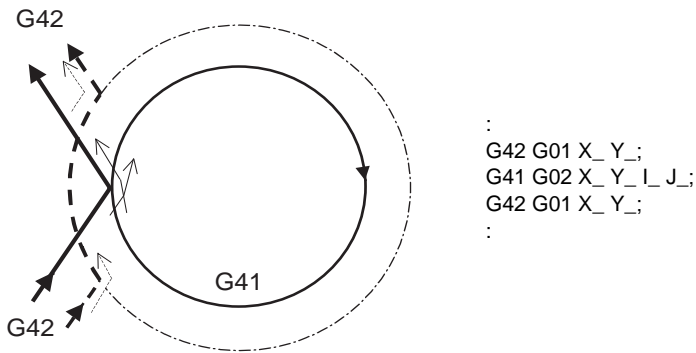


(5) Arc exceeding 360° due to compensation

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.



Program path
 Tool center path
 Section left uncut

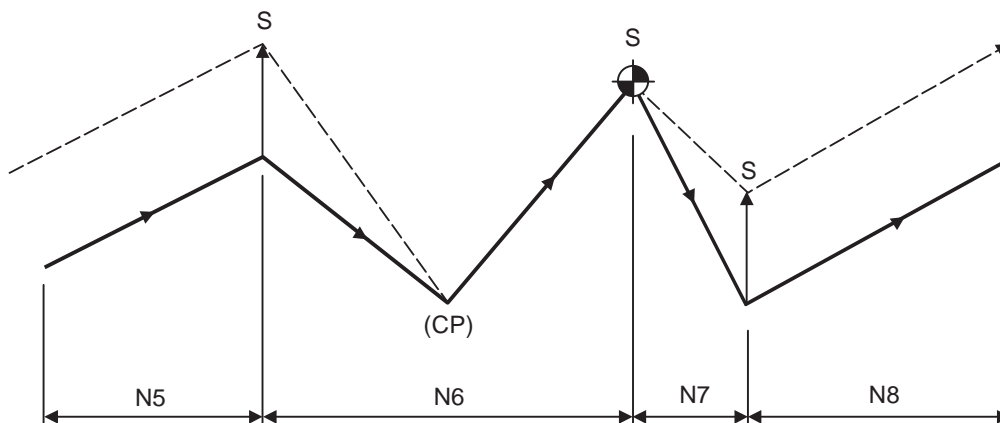
Command for eliminating compensation vectors temporarily

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)



(CP) Intermediate Point

(G41) :

N5 G91 G01 X60. Y30. ;

N6 G28 X50. Y-40. ;

<- Temporarily no compensation vectors at intermediate point.
(Reference position when there is no intermediate point)

N7 X30. Y-60. ;

N8 X70. Y40. ;

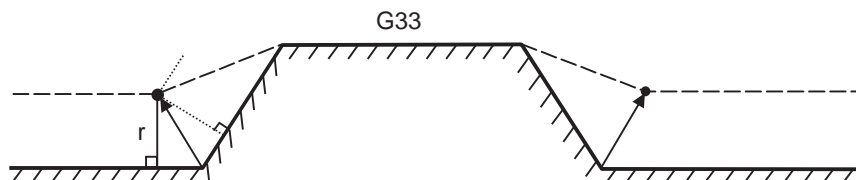
:

(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.

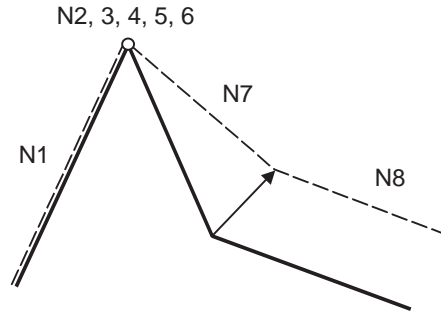
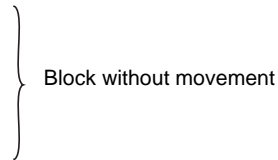
M03 ;	M command
S12 ;	S command
T45 ;	T command
G04 X500 ;	Dwell
G22 X200. Y150. Z100 ;	Machining prohibited region setting
G10 L10 P01 R50 ;	Compensation amount setting
G92 X600. Y400. Z500. ;	Coordinate system setting
(G17) Z40. ;	Movement outside the compensation plane
G90 ;	G code only
G91 X0 ;	Movement 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
 Compensation vector cannot be created when there are four or more successive blocks without movement, or when pre-reading prohibiting M command is issued.

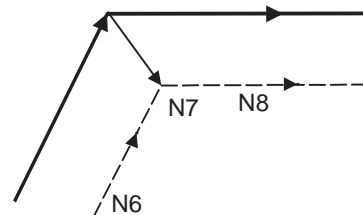
```

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



- (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. ;
```



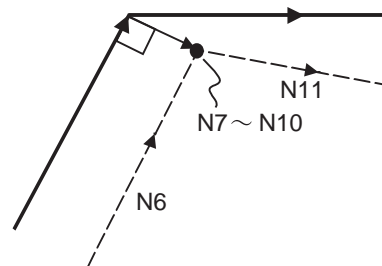
Block N7 is executed at N7 in the figure.

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.

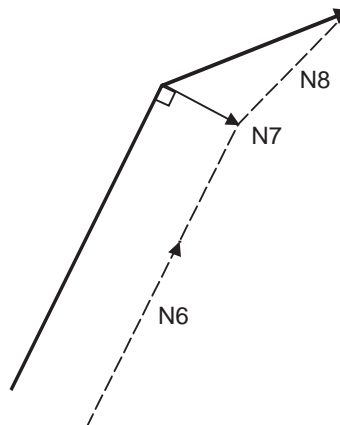
```
N6 X100. Y200. ;
N7 G4 X1000 ;
N8 F100 ;
N9 S500 ;
N10 M4 ;
N11 X100 ;
```

} Block without movement



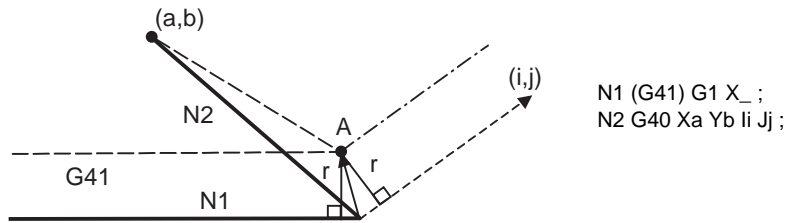
- (3) When commanded together with compensation cancel
 Only the compensation vectors are canceled when a block without movement is commanded together with the G40 command.

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



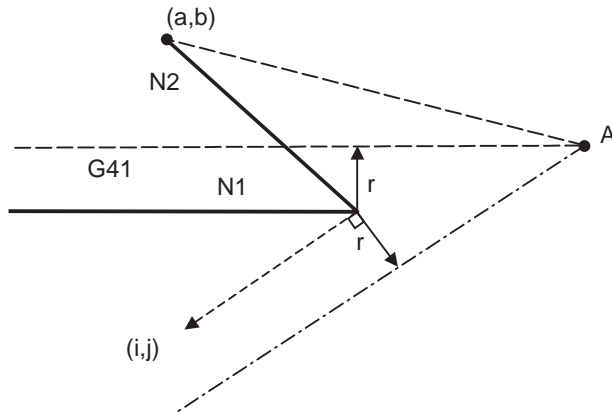
When I, J, K are commanded in G40

- (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.

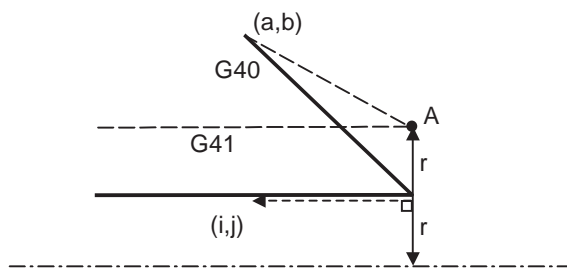


In this case, the point of intersection will always be obtained, regardless of the compensation direction, even when the commanded vector is incorrect as shown below.

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

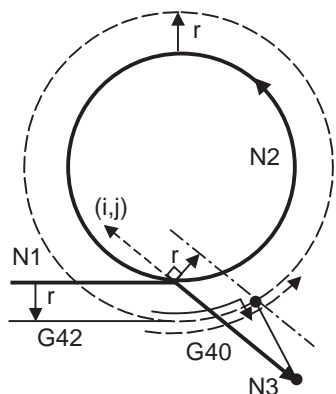


If the compensation vector obtained with point of intersection calculation is extremely large, a perpendicular vector will be created in the block before G40.



- 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.



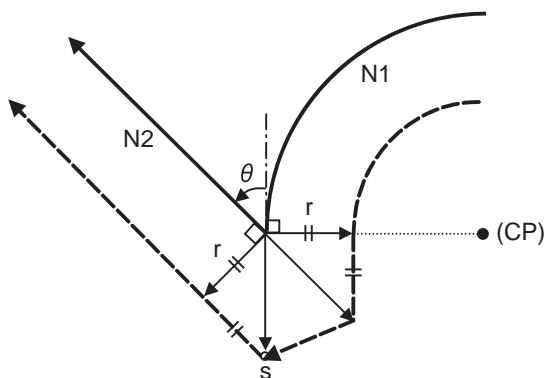
```
N1 (G42,G91) G01 X200. ;
N2 G02 J150. ;
N3 G40 G1 X150. Y-150. I-100. J100. ;
```

- Program path
- - - Tool center path
- · · Section left uncut

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

- Program path
- - - Tool center path

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.



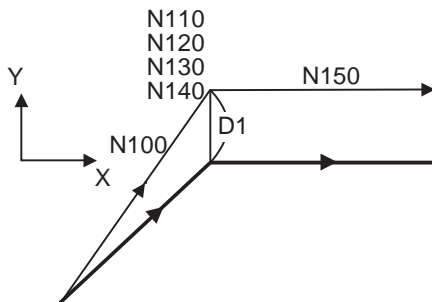
Detailed description

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

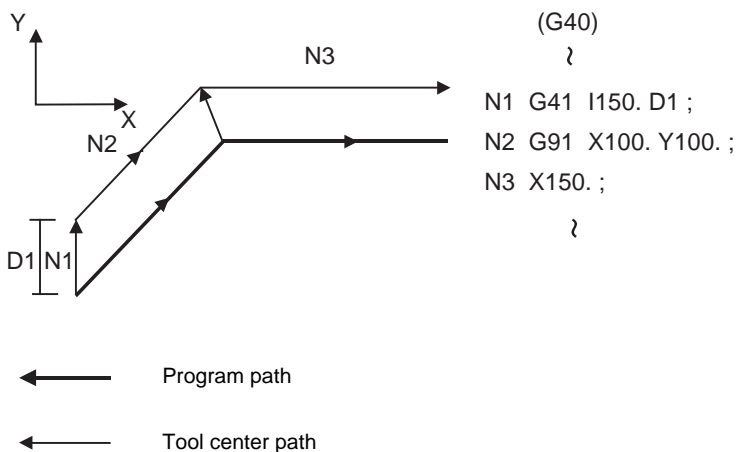


```

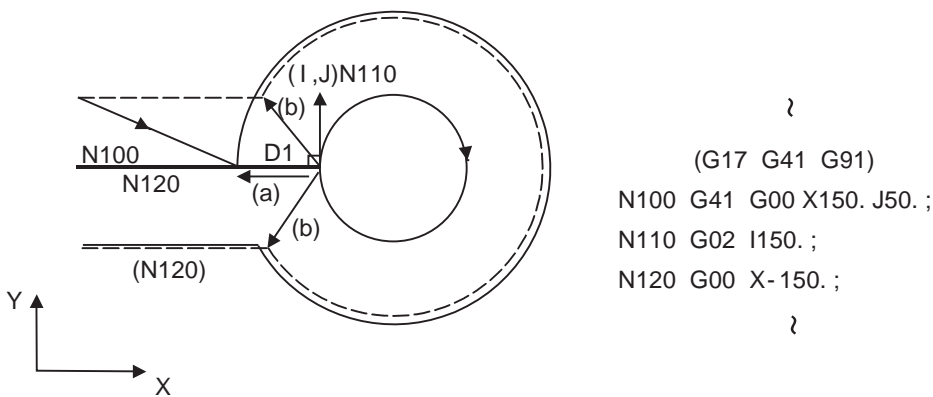
(G40)
?
N100 G91 G41 X100. Y100. I150. D1 ;
N110 G04 X1000 ;
N120 G01 F1000 ;
N130 S500 ;
N140 M03 ;
N150 X150. ;
?
    
```

- ← Program path
- ← Tool center path

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



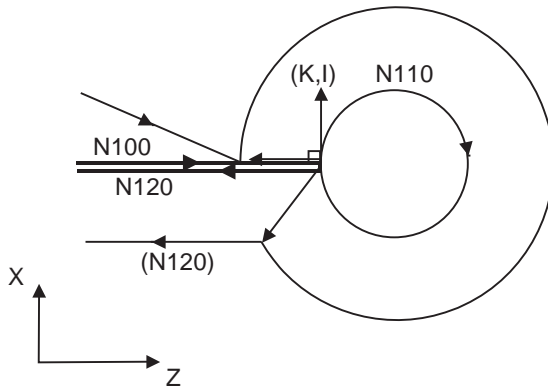
(3) When I, J has been commanded in the mode (G17 plane)



(a) I, J type vector

(b) Intersection point calculation type vector

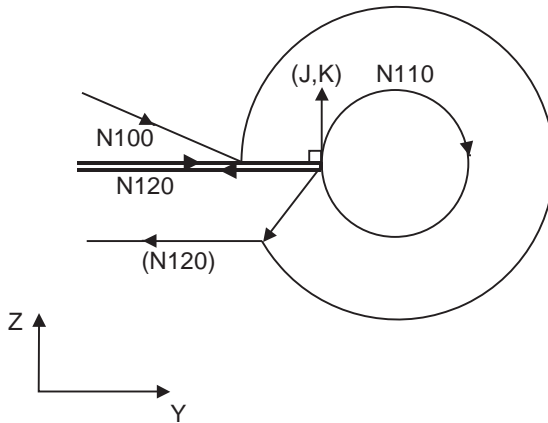
(Reference)
 (a) G18 plane



```

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

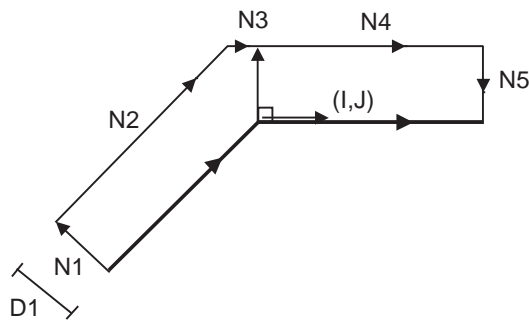
(b) G19 plane



```

    }
    (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 ;
    
```

Compensation 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
	Compensation vector direction		Compensation vector direction

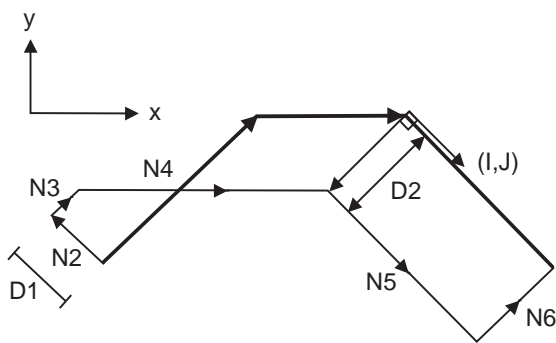
(2) In G42 mode

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
	Compensation vector direction		Compensation vector direction

Selection of compensation modal

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



```

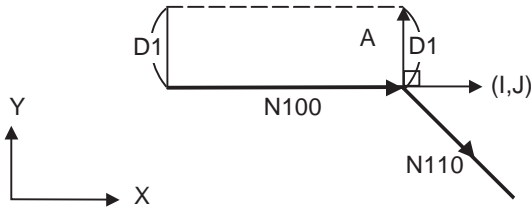
N1 G28 X0 Y0 ;
N2 G41 D1 F1000 ;
N3 G01 G91 X100. Y100. ;
N4 G42 X100. I100. J -100.
    D2 ;
N5 X100. Y-100. ;
N6 G40 ;
N7 M02 ;
%
```

Compensation amount for compensation vectors

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

<Example 1>

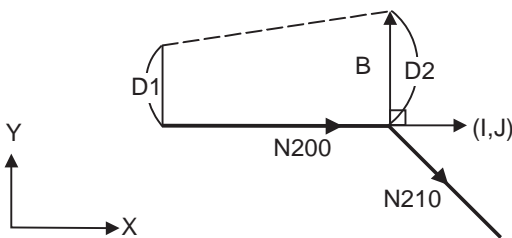
Vector A is the compensation amount registered in offset No. modal D1 of the N100 block.



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

<Example 2>

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

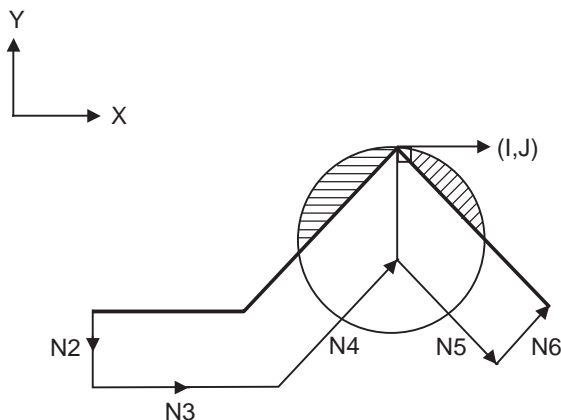


```
(G41 D1 G91)
    }
N200 G41 X150. I50. D2 ;
N210 X100. Y-100. ;
    }
```



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 compensation 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.



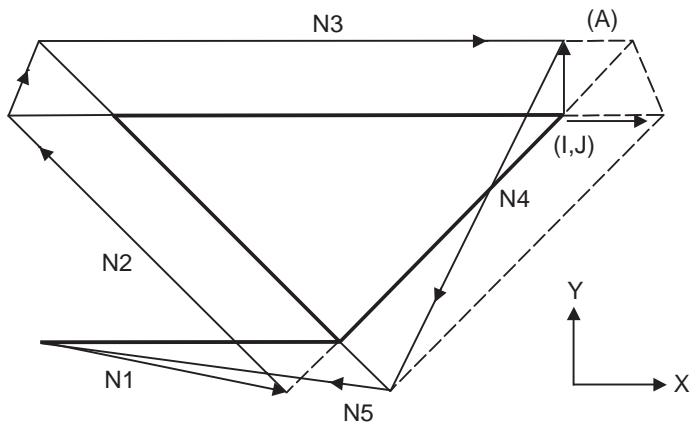
```
N1 G28 X0 Y0 ;
N2 G42 D1 F1000 ;
N3 G91 X100. ;
N4 G42 X100. Y100. I10. ;
N5 X100. Y-100. ;
N6 G40 ;
N7 M02 ;
```

(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

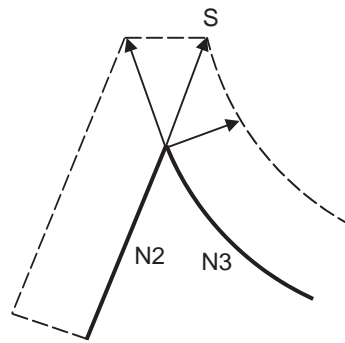
MDI interruption

Tool radius compensation is valid in any automatic operation mode - whether tape, memory or MDI mode. The figure below shows what happens by MDI interruption after stopping the block during tape or 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. ;	

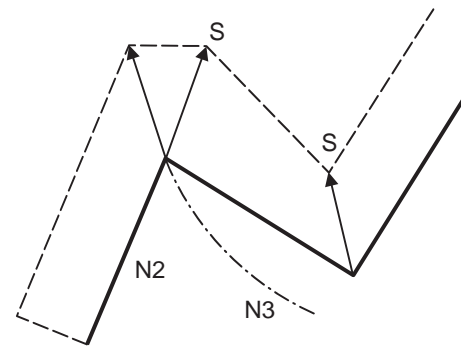


(2) Interrupt with movement

The compensation vectors are automatically re-calculated in the movement block after interrupt.

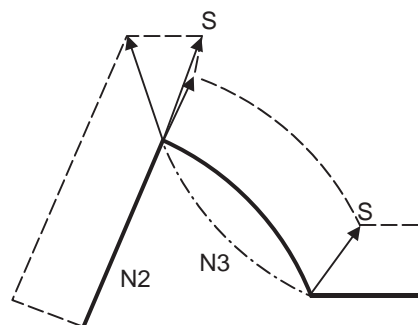
With linear interrupt

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



With circular interruption

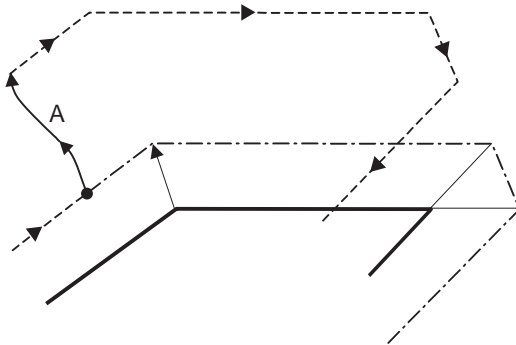
Automatic operation	MDI interruption
N1 G41 D1;	
N2 X20. Y50. ;	
	<--- G2 X40. Y-40. R70. ;
	G1 X40. ;
N3 G3 X40. Y-40. R70. ;	



Manual interruption

(1) Interrupt with manual absolute OFF.

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

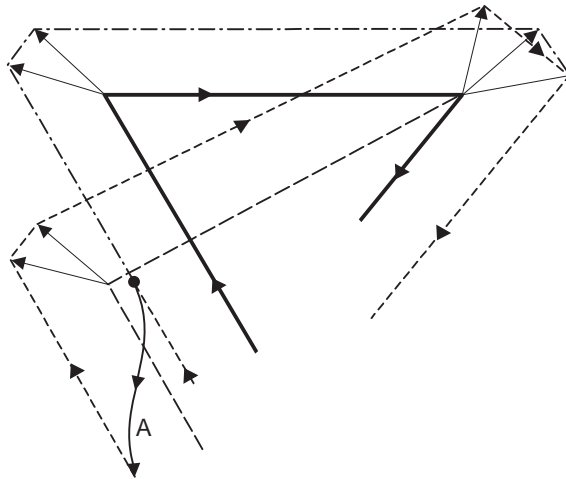


- Program path
- - - - - Tool path after compensation
- ← Interrupt (A)
- ← - - - - Tool path after interrupt

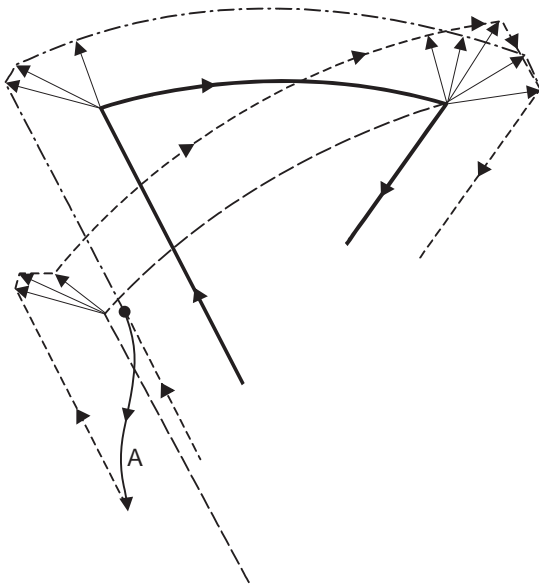
(2) Interrupt with manual absolute ON.

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.

[Line-Line-Line]



[Line-arc-Line]



- 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

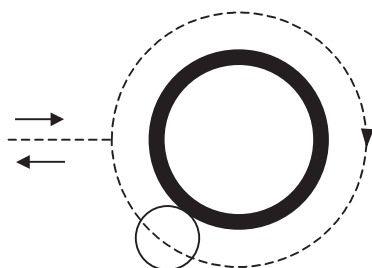
- (1) The offset amounts can be designated with the D code by designating an offset amount No. Once designated, the D code is valid until another D code is commanded. If an H code is designated, the program error (P170) No COMP No will occur.
Besides being used to designate the compensation amounts for tool radius compensation, the D codes are also used to designate the compensation amounts for tool position compensation.
- (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

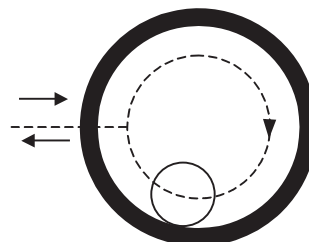
If the compensation amount is negative (-), the figure will be the same as if G41 and G42 are interchanged. Thus, the axis that was rotating around the outer side of the workpiece will rotate around the inner side, and vice versa.

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 ;

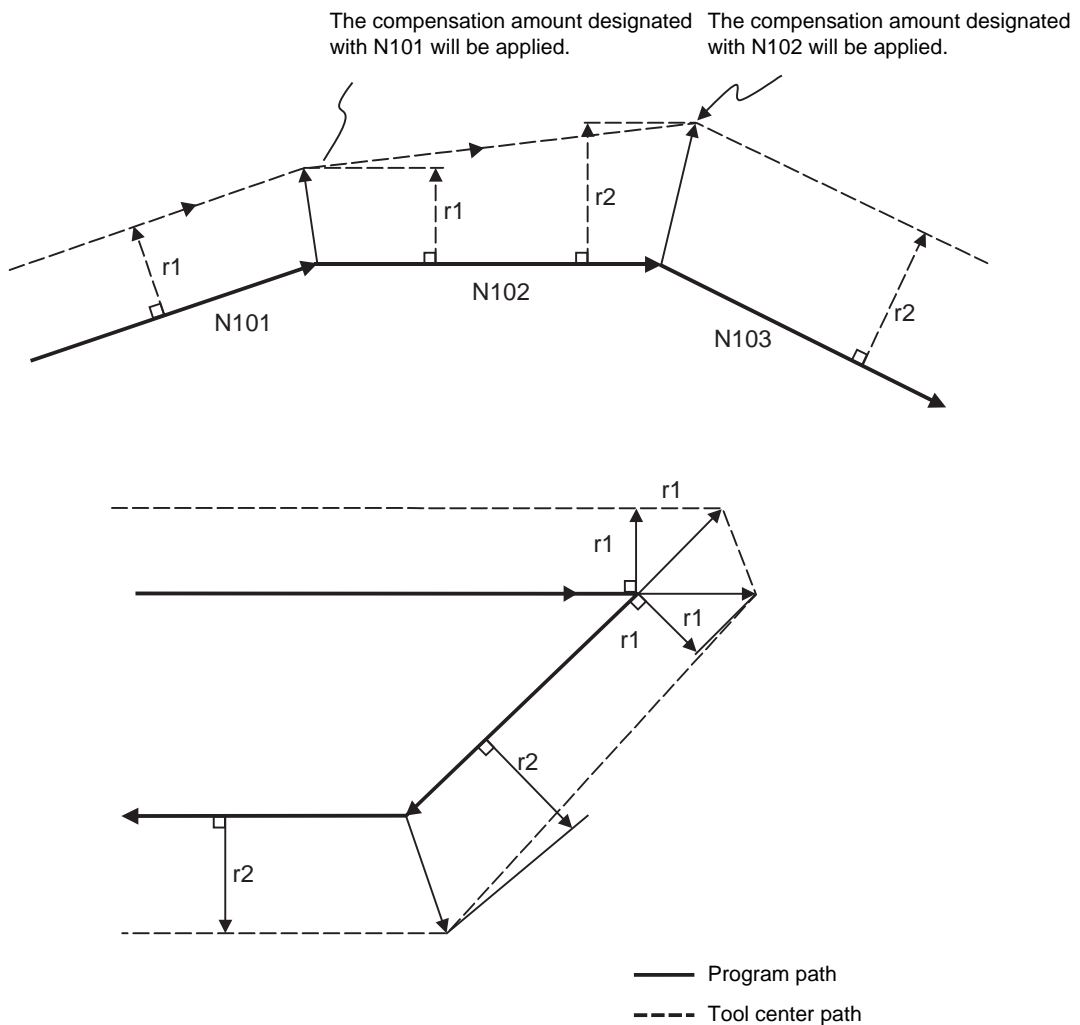
($\alpha = 0,1,2,3$)

N101 G0 α Xx1 Yy1 ;

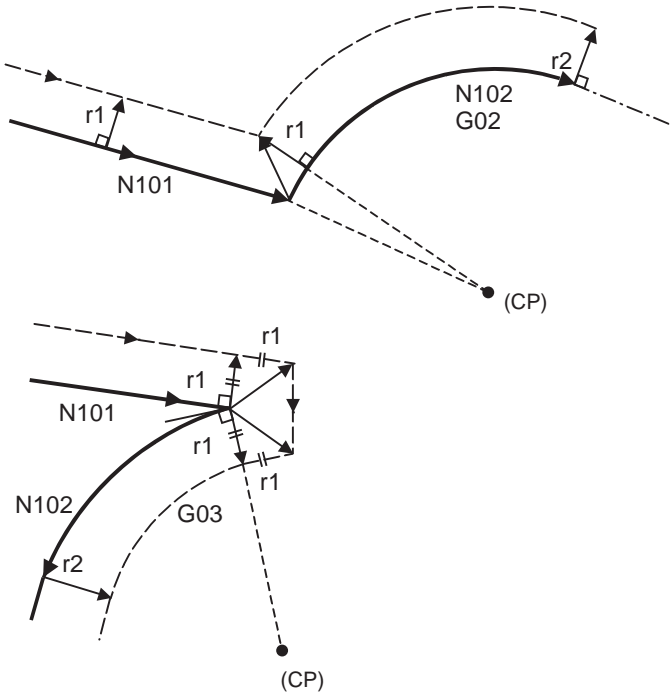
N102 G0 α Xx2 Yy2 Dr2 ; Compensation No. changed

N103 Xx3 Yy3 ;

During linear -> linear

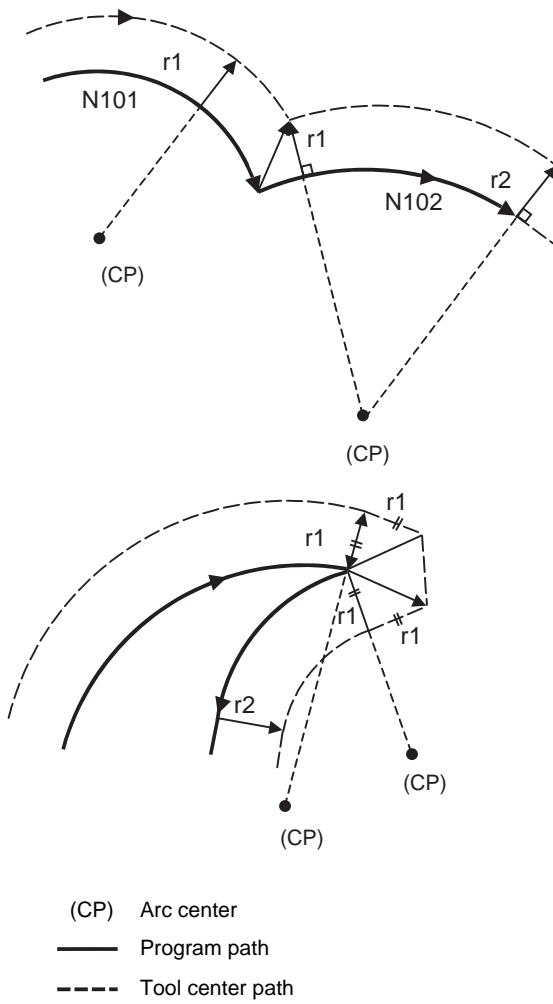


Linear ->circular



- (CP) Arc center
- Program path
- - - Tool center path

Circular -> circular



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



Function and purpose

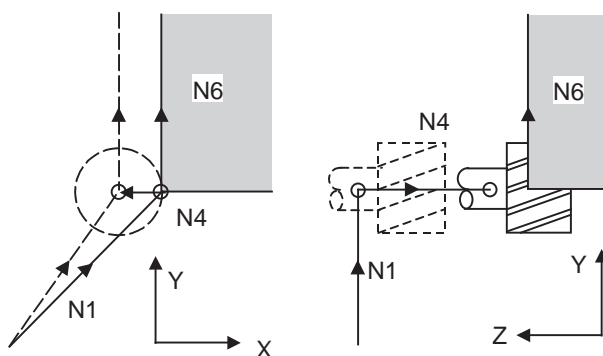
Often when starting cutting, a method of applying a radius compensation (normally the XY plane) beforehand at a position separated for the workpiece, and then cutting in with the Z axis is often used. When using this method, create the program so that the Z axis movement is divided into the two steps 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 ;
:
:
```



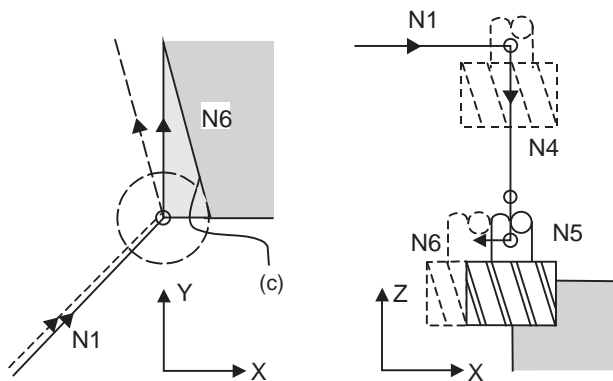
N4 Z axis lowers (1 block)

----- 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

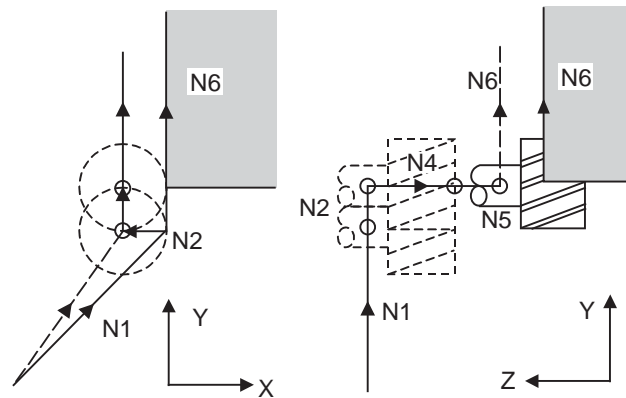
In this case, the four blocks N2 to N5 do not have a command in the XY plane, so when the N1 compensation is started, the program cannot be read to the N6 block.

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.

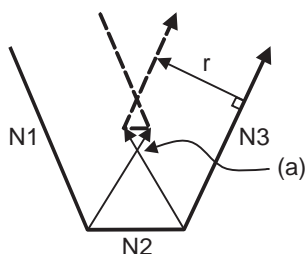
	Function	Parameter		Operation
		#8102 COLL. ALM OFF	#8103 COLL. CHK OFF	
(1)	Interference check alarm function	0	0	Operation stops with a program error (P153) before executing a block which will cause cutting.
(2)	Interference check avoidance function	1	0	The tool path is changed to prevent cutting from occurring. If the path cannot be changed, a program error (P153) occurs and the program will be stopped.
(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

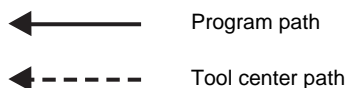
Conditions viewed as interference

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.



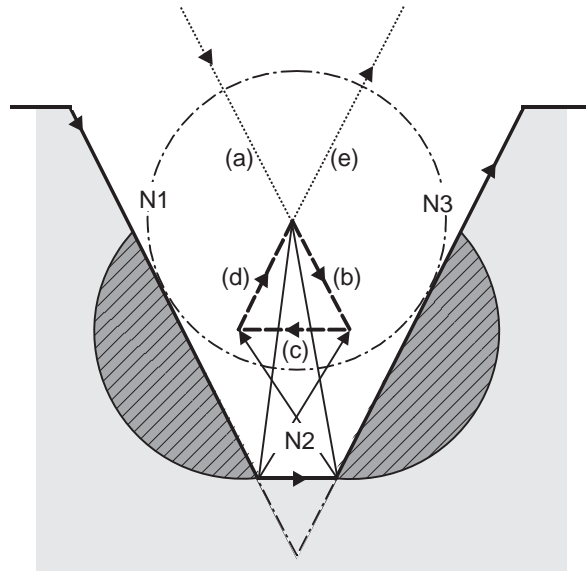
r : Compensation amount

(a) Vectors intersect



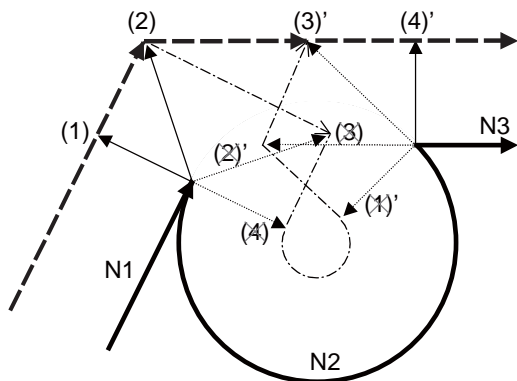
(Example 1) When operating a program including a short segment with a tool with a large radius
Cutting will occur in the shaded section.

```
(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.

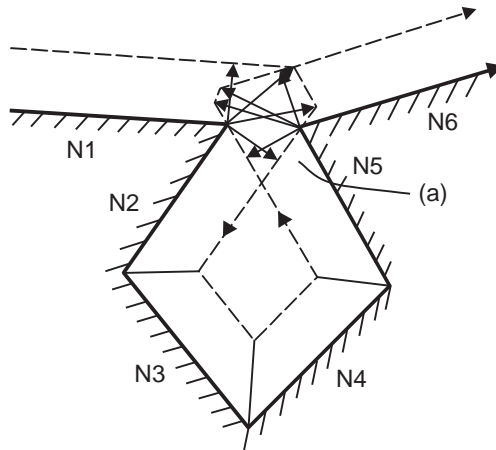
← - - - - (Thick broken line path)

- (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.

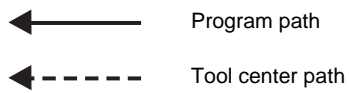
← - . - . - . (Thin broken line path)

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

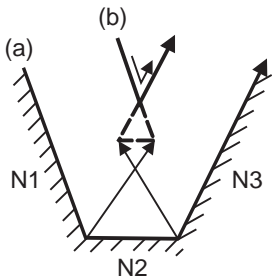


(a) Interference check is not possible

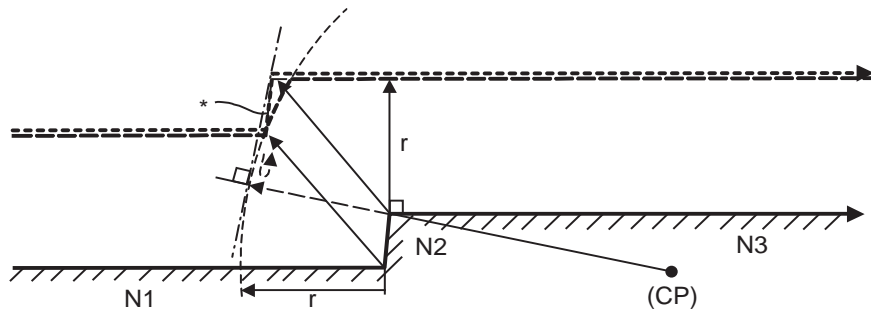
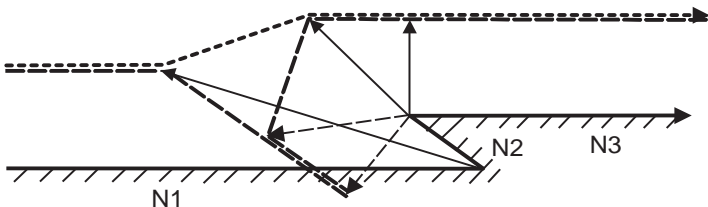


Operation when interference avoidance function is valid

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

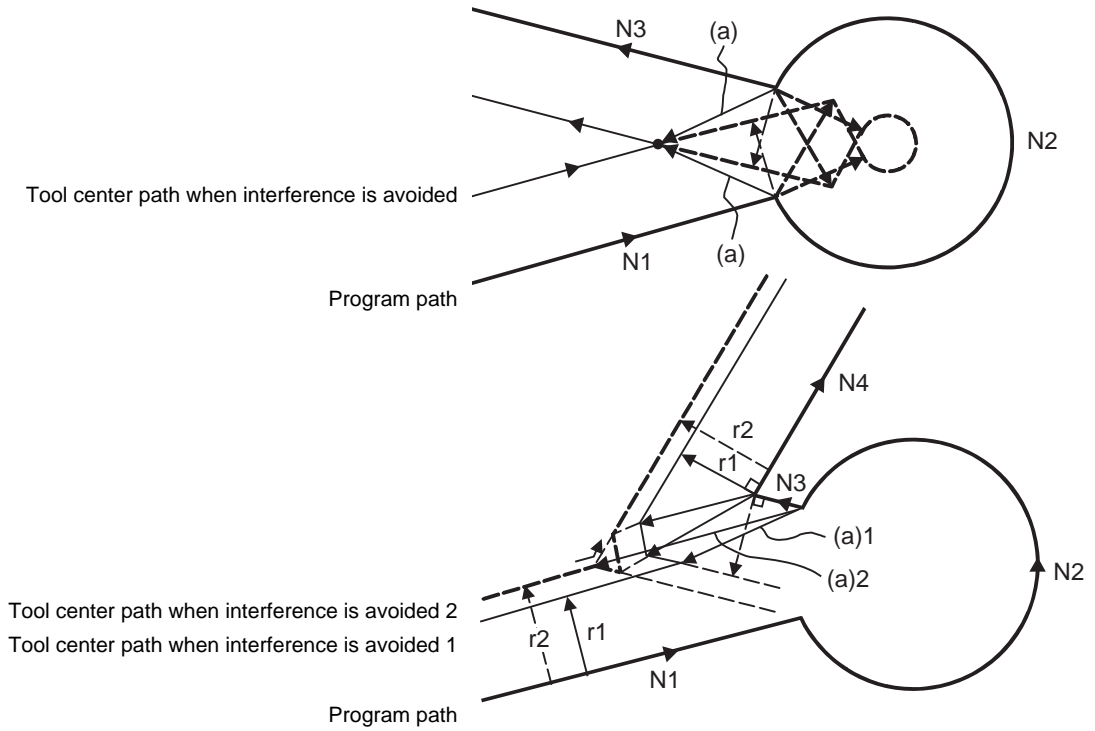


(a) Program path
(b) Tool center path



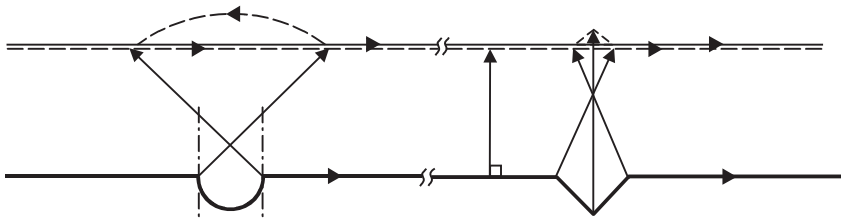
- ← Program path
- ← - - - - - Tool center path without interference check
- ← ······· Tool center path when interference is avoided (*: Linear movement)
- ← Valid vector
- ← - - - - - Invalid vector

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.



(a) Avoidance vector

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

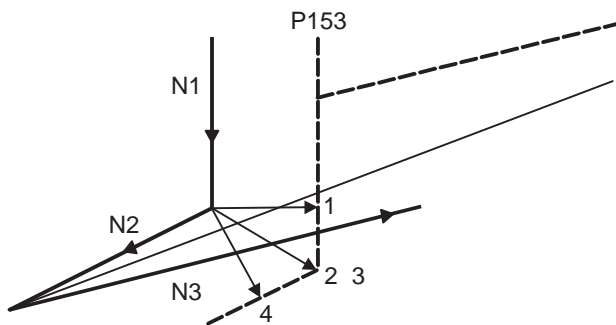


- ← Program path
- ← - - - - - Tool center path without interference check
- ← Tool center path when interference is avoided

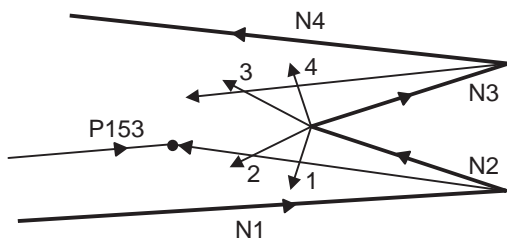
Interference check alarm operation

The interference check alarm occurs under the following conditions.

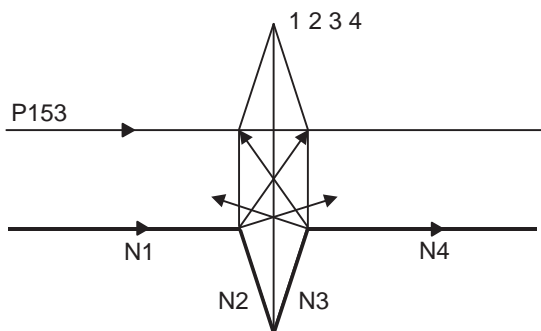
- (1) When the interference check alarm function has been selected
 When all the vectors at the end of its own block have been deleted.
 When, as shown in the figure below, vectors 1 through 4 at the end point of the N1 block have all been deleted, program error (P153) will occur prior to N1 execution.



- (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.



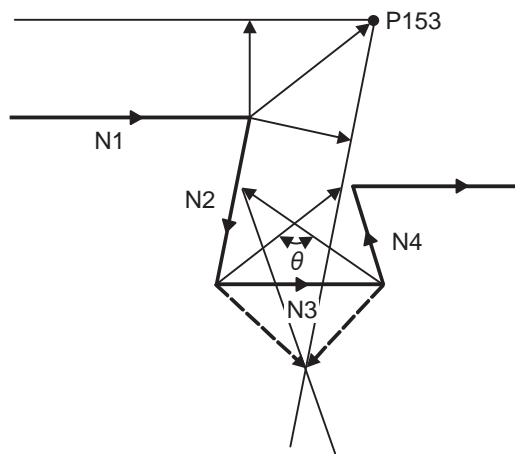
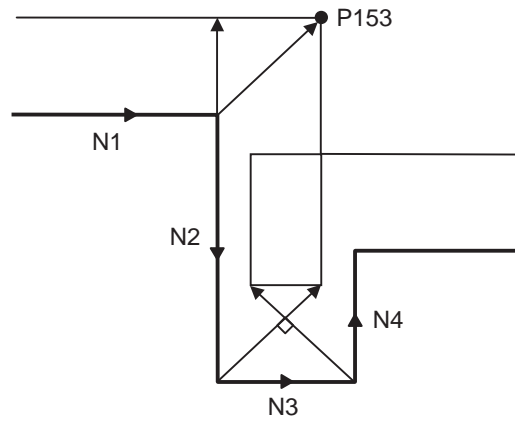
In the case shown in the figure below, the tool will move in the reverse direction at N2. Program error (P153) now occurs before executing N1 and the operation stops.



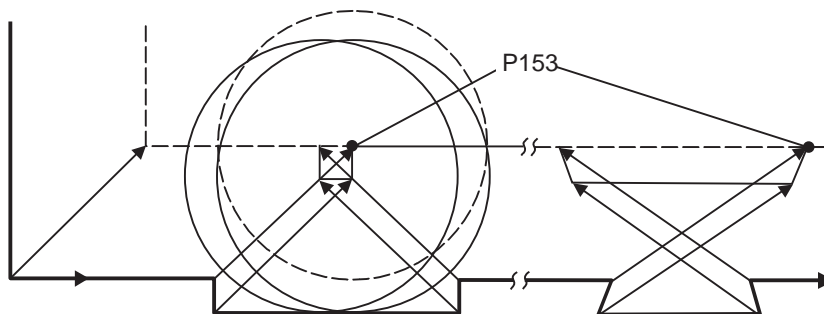
(Example 2) When avoidance vectors cannot be created

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.

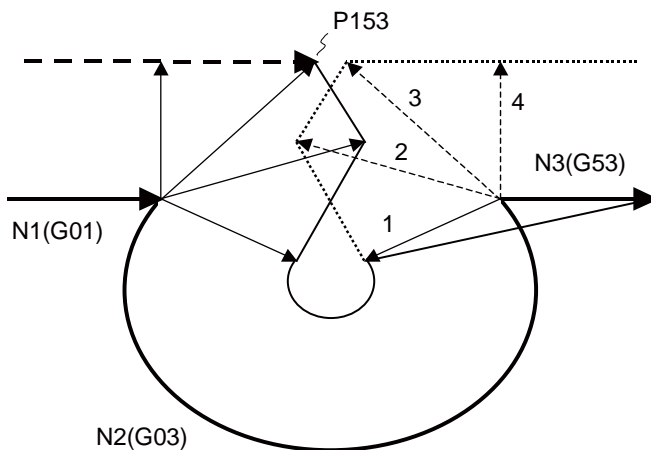


(Example 3) When the program advance direction and the advance direction after compensation are reversed
 When grooves, narrower than the tool radius with parallel or widening bottom, are programmed, it will still be regarded as interference even if there is actually no interference.



(Example 4) Command for eliminating compensation vectors temporarily
 Interference check will be executed even at the end point of the block right before the command to eliminate compensation vectors temporarily, as well as when the command is not issued. It may be regarded as an interference even if there is actually no interference. If regarded as an interference, program error (P153) will occur.

In the figure below, only vector 1 is left as an end point vector in N2 because of the N3 G53 command to temporarily eliminate compensation vectors. However, the interference check will still be conducted to vector 1 to 4 and an interference will be detected. Program error (P153) now occurs at the end point of the previous block and the operation stops.

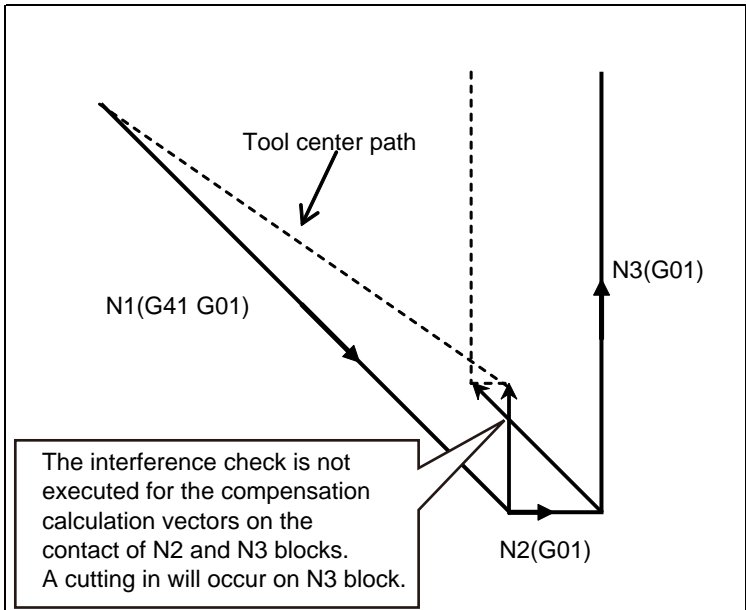


- ← Program path
- ← - - - - Tool center path
- Tool center path when the interference check is invalid
- Tool center path when the interference check is invalid in N3 due to a command (G01 etc.) not to eliminate compensation vectors.
- ← Valid vector
- ← - - - - Invalid vector (Invalid, however, subject to interference check)

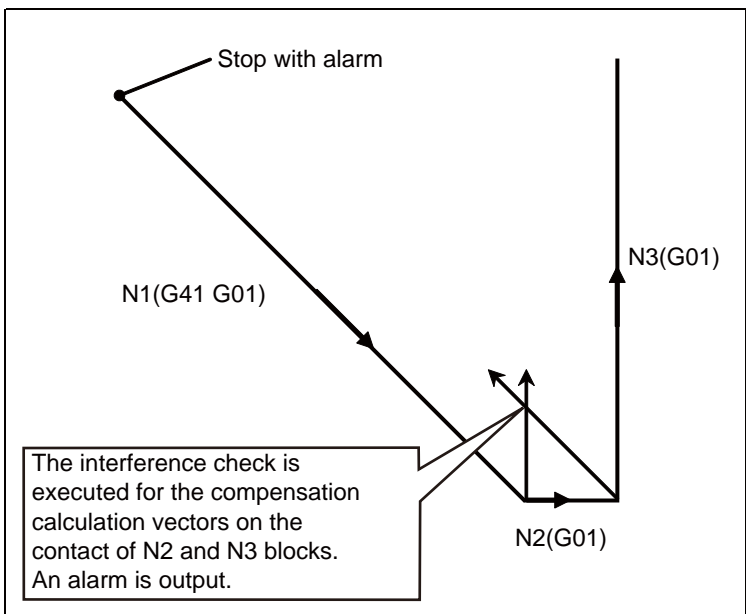
Interference check for start-up block

When starting compensation operation, the tool center path is determined with the movement command of the same block as G41/G42 and the next movement command. The interference check is not executed at that time. Set the parameter "#1241 set13/bit1" to check interference. Note that an alarm is output and the operation is stopped even when the collision avoidance setting "#8102 COLL. ALM OFF" is set to "1" and that the interference avoidance is not applied.

- (a) Interference check for start-up block invalid
"#1241 set13/bit1" = "0" or "#8103 COLL. CHK OFF" = "1"



- (b) Interference check for start-up block valid
"#1241 set13/bit1" = "1" and "#8103 COLL. CHK OFF" = "0"



12.3.9 Workpiece Coordinate Changing during Radius Compensation



Function and purpose

When the tool radius compensation is executed, the tool center path is calculated based on the position on the coordinate system. The based coordinate system is different depending on setting of the parameter "#1246/bit2 Switch coordinate systems for radius compensation".

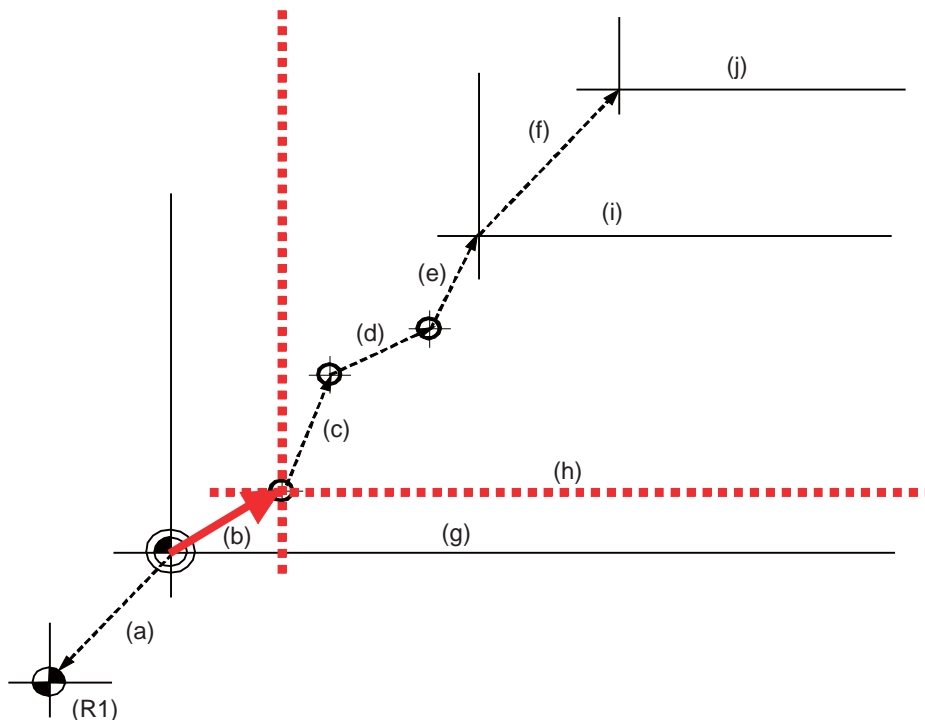


Detailed description

When the parameter is "0", the tool radius compensation is calculated based on the position on the workpiece coordinate system.

When the parameter is "1", the tool radius compensation is calculated based on the position on the program coordinate system.

The program coordinate systems are defined as shown in the figure below.



- (R1) 1st reference position
- (a) 1st reference position offset
- (b) Interrupt amount offset
- (c) Extended workpiece coordinate system offset
- (d) G92 offset
- (e) Workpiece coordinate system offset
- (f) Local coordinate system offset
- (g) G53 Basic machine coordinate system
- (h) Program coordinate system
- (i) G54 to G59/G54.1Pn Workpiece coordinate system/Extended workpiece coordinate system
- (j) G52 Local coordinate system

The coordinate system changed by parameter is as follows.

G90 G54 G00 X15. Y20.;
N1 G41 D3 X5. Y10. ;
N2 G01 Y-20. F1000 ;
N3 G40 X30. ;
M30 ;

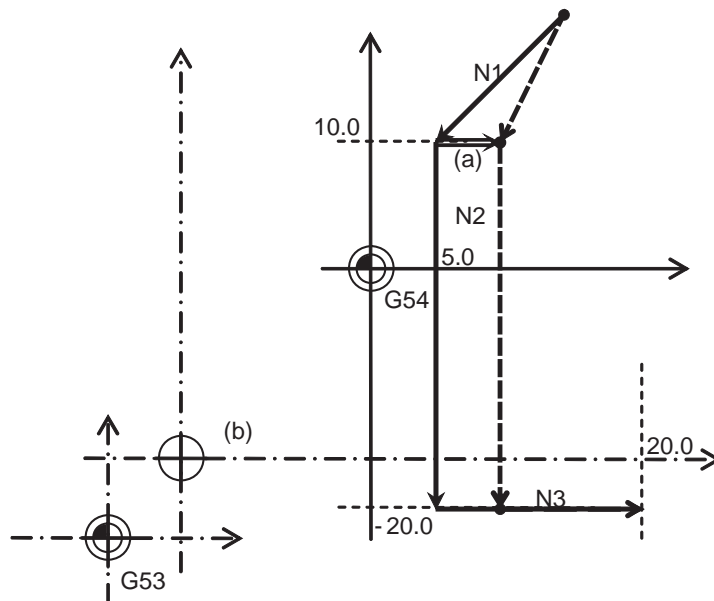
D3 = 5.000

G54 offset

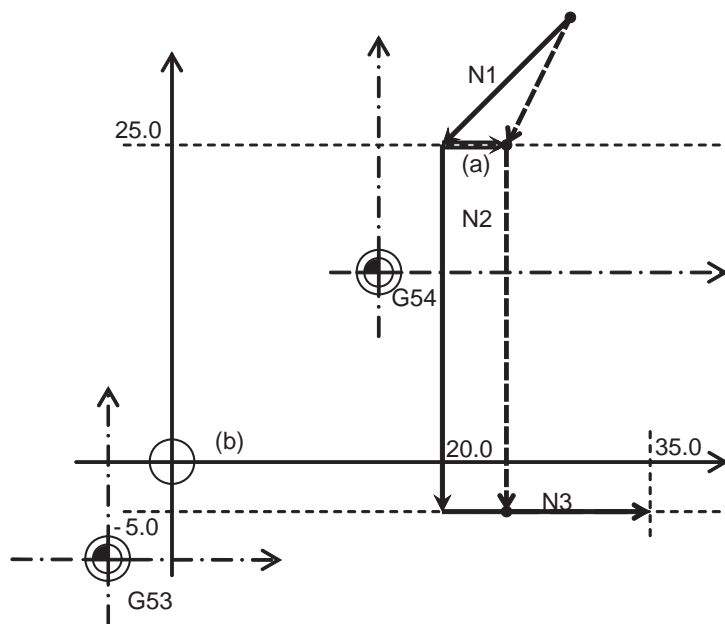
X15.000

Y15.000

(1) Parameter = 0



(2) Parameter = 1



— Program path
 - - - Tool center path

(a) Compensation vector
 (b) Program coordinate system

12.4 Compensation Data Input by Program ; G10



Function and purpose

The tool compensation 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

Workpiece offset input

G90 (G91) G10 L2 P__ X__ Y__ Z__ ;

P	0 : External workpiece
	1 : G54
	2 : G55
	3 : G56
	4 : G57
	5 : G58
	6 : G59

(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.

Tool compensation input

Type I

G10 L10 P__ R__ ; ... For tool compensation memory

P	Compensation No.
R	Compensation amount

Type II

G10 L10 P__ R__ ; ... For tool compensation memory Tool length compensation shape compensation

G10 L11 P__ R__ ; ... For tool compensation memory Tool length compensation wear compensation

G10 L12 P__ R__ ; ... For tool compensation memory Tool radius shape compensation

G10 L13 P__ R__ ; ... For tool compensation memory Tool radius wear compensation

(Note) Type I is selected when parameter "#1037 cmdtyp" is set to "1", and type II is selected when set to "2".



Detailed description

- (1) G10 is an unmodal command and is valid only in the commanded block.
- (2) The G10 command does not contain movement, but must not be used with G commands other than G54 to G59, G90 or G91.
- (3) Do not command G10 in the same block as the fixed cycle and sub-program call command. This will cause malfunctioning and program errors.
- (4) The workpiece offset input command (L2 or L20) should not be issued in the same block as the tool compensation input command (L10).
- (5) If an illegal L No. or compensation No. is commanded, the program errors (P172 and P170) will occur respectively.
If the offset amount exceeds the maximum command value, the program error (P35) will occur.
- (6) Decimal point inputs can be used for the offset amount.
- (7) 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.
- (8) 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.
- (9) L2 can be omitted when the workpiece offset is input.
- (10) When the P command is omitted, it will be handled as the currently selected workpiece compensation input.
- (11) The setting range for the compensation amount is given below.
Program error (P35) occurs for any value not listed in the table after command unit conversion.
With an incremental value command, the setting range for the compensation amount is the sum of the present setting value and command value.

Setting	Compensation amount			
	#11050(T-ofs digit type)=0		#11050(T-ofs digit type)=1	
	Metric system	Inch system	Metric system	Inch system
#1004=B	± 999.999 (mm)	± 99.9999 (inch)	± 9999.999 (mm)	± 999.9999 (inch)
#1004=C	± 999.9999 (mm)	± 99.99999 (inch)	± 9999.9999 (mm)	± 999.99999 (inch)



Program example

- (1) **Input the compensation amount**

```
.....; G10 L10 P10 R-12.345 ; G10 L10 P05 R9.8765 ; G10 L10 P30 R2.468 ; .....
```

H10=-12.345 H05=9.8765 H30=2.468

(2) Updating of compensation amount

(Example 1) Assume that H10 = -1000 is already set.

N1 G01 G90 G43 Z-100000 H10 F100 ;	(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)

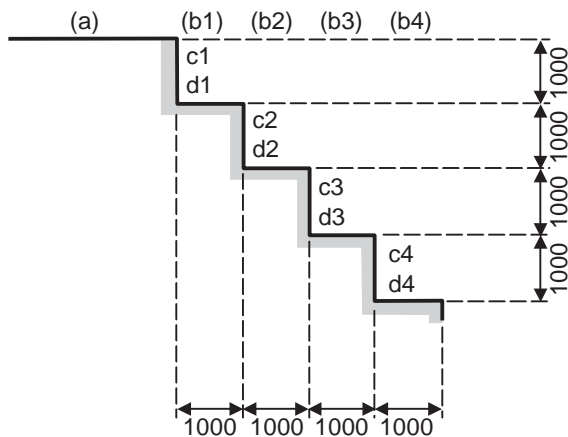
(Example 2) Assume that H10 = -1000 is already set.

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 ;

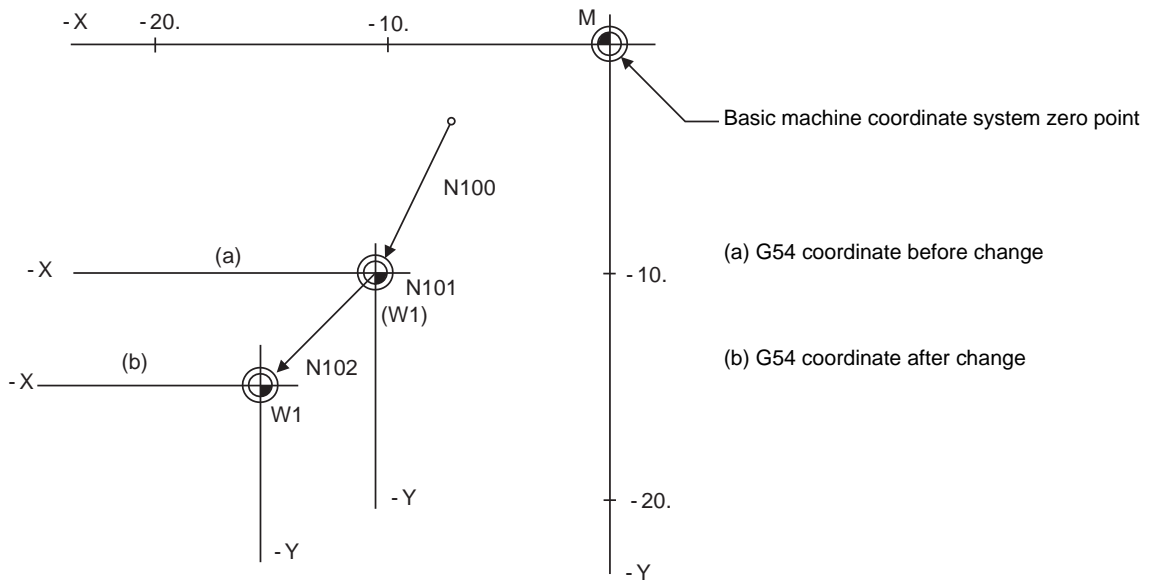
Subprogram O1111

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.

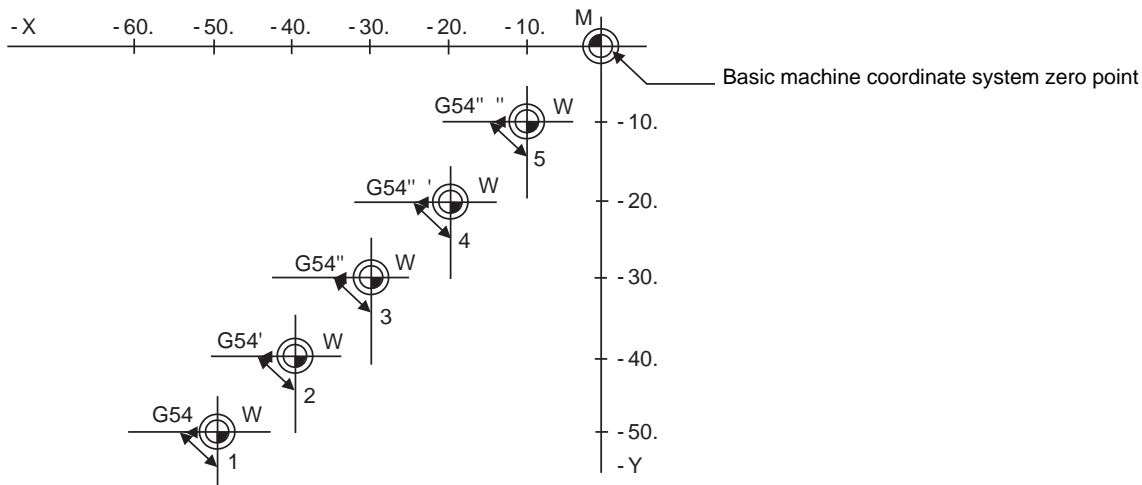
$$\begin{matrix} X=0 & \rightarrow & X=+5.000 \\ Y=0 & & Y=+5.000 \end{matrix}$$

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.

12.5 Inputting the Tool Life Management Data ; G10,G11

12.5.1 Inputting the Tool Life Management Data by G10 L3 Command ; G10 L3,G11



Function and purpose

Using the G10 command (unmodal command), the tool life management data can be registered, changed and added to, and preregistered groups can be deleted.



Command format

Start of life management data registration

```
G10 L3 ;
P_L_Q_ ; (First group)
T_H_D_ ;
T_H_D_ ;
P_L_Q_ ; (Next group )
T_H_D_ ;
```

P	Group No.
L	Life
Q	Control method
T	Tool No. The spare tools are selected in the order of the tool Nos. registered here.
H	Length compensation No.
D	Radius compensation No.

Start of life management data change or addition

```
G10 L3 P1;
P_L_Q_ ; (First group)
T_H_D_ ;
T_H_D_ ;
P_L_Q_ ; (Next group)
T_H_D_ ;
```

P	Group No.
L	Life
Q	Control method
T	Tool No.
H	Length compensation No.
D	Radius compensation No.

Start of life management data deletion

```
G10 L3 P2;
P_ ; (First group)
P_ ; (Second group)
```

P	Group No.
---	-----------

End of life management data registration, change, addition or deletion

```
G11 ;
```



Detailed description

Command range

Item		Command range
Group No.	(Pn)	1 to 99999999 (Only group No. 1 can be used for the tool life management III)
Life	(Ln)	0 to 65000 times (No. of times control method) 0 to 4000 minutes (time control method)
Control method	(Qn)	1 to 3 1: Number of mounts control 2: Time control 3: Number of cutting times control
Tool No.	(Tn)	1 to 99999999
Length compensation No.	(Hn)	0 to 200
Radius compensation No.	(Dn)	0 to 200



Operation example

	Program example	Operation
Data registration	G10 L3 ; P10 L10 Q1 ; T10 H10 D10 ; G11 ; M02 ;	1. After deleting all group data, the registration starts. 2. Group No. 10 is registered. 3. Tool No. 10 is registered in group No. 10. 4. The registration ends. 5. The program ends.
Group change, addition	G10 L3 P1 ; P10 L10 Q1 ; T10 H10 D10 ; G11 ; M02 ;	1. Changing and addition of the group and tool starts. 2. The change and addition operation takes place in the following manner. (1) When group No. 10 has not been registered. - Group No. 10 is additionally registered. - Tool No. 10 is registered in group No. 10. (2) When group No. 10 has been registered, but tool No. 10 has not been registered. - Tool No. 10 is additionally registered in group No. 10 (3) When group No. 10 and tool No. 10 have been both registered. - The tool No. 10 data is changed. 3. The group and tool change and addition ends. 4. The program ends.
Group deletion	G10 L3 P2 ; P10 ; G11 ; M02 ;	1. The group deletion starts. 2. The group No. 10 data is deleted. 3. The group deletion ends. 4. The program ends.

12.5.2 Inputting the Tool Life Management Data by G10 L30 Command ; G10 L30,G11



Function and purpose

Using the G10 command (unmodal command), the tool life management data can be registered, changed and added to, and preregistered groups can be deleted.

To specify additional compensation amount or direct compensation amount by control method, the length compensation and diameter compensation can be registered/changed with the tool compensation amount format.



Command format

Start of life management data registration

```
G10 L30 ;
P_L_Q_ ; (First group)
T_H_R_ ;
T_H_R_ ;
P_L_Q_ ; (Next group )
T_H_R_ ;
```

P	Group No.
L	Life
Q	Control method
T	Tool No. The spare tools are selected in the order of the tool Nos. registered here.
H	Length compensation No. or length compensation amount
R	Radius compensation No. or Radius compensation amount

Start of life management data change or addition

```
G10 L30 P1;
P_L_Q_ ; (First group)
T_H_R_ ;
T_H_R_ ;
P_L_Q_ ; (Next group)
T_H_R_ ;
```

P	Group No.
L	Life
Q	Length compensation data format, radius compensation data format, control method
T	Tool No.
H	Length compensation No. or length compensation amount
D	Radius compensation No. or radius compensation amount

Start of life management data deletion

```
G10 L30 P2;
P_ ; (First group)
P_ ; (Second group)
```

P	Group No.
---	-----------

End of life management data registration, change, addition or deletion

G11 ;



Detailed description

Command range

Item		Command range
Group No.	(Pn)	1 to 99999999 (Only group No. 1 can be used for the tool life management III)
Tool No.	(Tn)	1 to 99999999
Control method	(Qabc)	abc:Three integer digits a.Tool length compensation data format 0: Compensation No. 1: Incremental value compensation amount 2: Absolute value compensation amount b.Tool radius compensation data format 0: Compensation No. 1: Incremental value compensation amount 2: Absolute value compensation amount c.Tool management method 0: Usage time 1: Number of mounts 2: Number of usages
Life	(Ln)	0 to 4000 minutes (usage time) 0 to 65000 times (number of mounts) 0 to 65000 times (number of usages)
Length compensation (No./amount)	(Hn)	0 to 200 (compensation No.) ± 999.999 (incremental value compensation amount) (*) ± 999.999 (absolute value compensation amount) (*)
Radius compensation (No./amount)	(Rn)	0 to 200 (compensation No.) ± 999.999 (incremental value compensation amount) (*) ± 999.999 (absolute value compensation amount) (*)

(*) Refer to (16) in "12.5.3 Precautions for Inputting the Tool Life Management Data" for the data range of compensation amount.



Operation example

	Program example	Operation
Data registration	G10 L30 ; P10 L10 Q001 ; T10 H10 R10 ; G11 ; M02 ;	<ol style="list-style-type: none"> 1. After deleting all group data, the registration starts. 2. Group No. 10 is registered. Tool management method is number of mounts Compensation No. method is applied to tool length compensation and tool radius compensation. 3. Tool No. 10 is registered in group No. 10. 4. The registration ends. 5. The program ends.
Group change, addition	G10 L30 P1 ; P10 L10 Q122 ; T10 H10 R0.25 ; G11 ; M02 ;	<ol style="list-style-type: none"> 1. Changing and addition of the group and tool starts. 2. The change and addition operation takes place in the following manner. (1) When group No. 10 has not been registered. [Group No. 10 is additionally registered.] About the change and addition tool Tool management method is number of usages Tool length compensation is incremental value compensation amount method Tool radius compensation is absolute value compensation amount method [Tool No. 10 is registered in group No. 10.] The absolute value compensation amount "0.25" is set to tool radius compensation. (2) When group No. 10 has been registered, but tool No. 10 has not been registered. - Tool No. 10 is additionally registered in group No. 10 (3) When group No. 10 and tool No. 10 have been both registered. - The tool No. 10 data is changed. 3. The group and tool change and addition ends. 4. The program ends.
Group deletion	G10 L30 P2 ; P10 ; G11 ; M02 ;	<ol style="list-style-type: none"> 1. The group deletion starts. 2. The group No. 10 data is deleted. 3. The group deletion ends. 4. The program ends.

12.5.3 Precautions for Inputting the Tool Life Management Data



Relation with other functions

- (1) During the following operations, the tool usage data will not be counted.
 - Machine lock
 - Miscellaneous function lock
 - Dry run
 - Single block
 - Skip



Precautions

- (1) The tool life data is registered, changed, added to or deleted by executing the program in the memory or MDI mode.
- (2) The group No. and tool No. cannot be commanded in duplicate. The program error (P179) will occur.
- (3) When two or more addresses are commanded in one block, the latter address will be valid.
- (4) If the life data (L_) is omitted, the life data for that group will be "0".
- (5) If the control method (Q_) is omitted, the control method for that group will follow the base specification parameter "#1106 Tcount".
Note that when carrying out the No. of cutting times control method, command the method from the program.
- (6) If the control method (Q_) is not designated with 3-digit by G10 L30 command, the omitted high-order are equivalent to "0".
Therefore, "Q1" is equivalent to "Q001", and "Q12" is equivalent to "Q012".
- (7) If the length compensation No. (H_) is omitted, the length compensation No. for that group will be "0".
- (8) If the radius compensation No. (D_) is omitted, the radius compensation No. for that group will be "0".
- (9) Programming with a sequence No. is not possible between G10 L3 or G10 L30 and G11. The program error (P33) will occur.
- (10) If the usage data count valid signal (YC8A) is ON, G10 L3 or G10 L30 cannot be commanded. The program error (P177) will occur.
- (11) The registered data is held even if the power is turned OFF.
- (12) When G10 L3 or G10 L30 is commanded, the commanded group and tool will be registered after all of the registered data is erased.
- (13) The change and addition conditions in the G10L3P1 or G10 L30 P1 command are as follows.
 - (a) Change conditions
Both the commanded group No. and tool No. are registered.
-> Change the commanded tool No. data.
 - (b) Additional conditions
Neither the commanded group No. nor tool No. is registered.
-> Additionally register the commanded group No. and tool No. data.
The commanded group No. is registered, but the commanded tool No. is not registered.
-> Additionally register the commanded tool No. data to the commanded group No.
- (14) The setting range of the tool compensation No. is described in the table below.

No. of tool offset sets	Setting range
200 sets	1-200

- (15) Only group No. 1 can be used to register, change and add for the tool life management III.
- (16) The data range of compensation amount changes as follows according to the parameter "#11050 T-ofs digit type(Tool compensation digit type)".

#11050	Data range
0: Set with a 3-digit integer	Absolute value compensation amount ± 9999.999
	Incremental value compensation amount ± 9999.999
1: Set with a 4-digit integer	Absolute value compensation amount ± 999.999
	Incremental value compensation amount ± 999.999

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)	Retract at high speed	Application
		Dwell	Spindle			
G80	-	-	-	-	-	Cancel
G81	Cutting feed	-	-	Rapid traverse	Yes	Drill, spot drilling cycle
G82	Cutting feed	Yes	-	Rapid traverse	-	Drill, counter boring cycle
G83	Intermittent feed	-	-	Rapid traverse	Yes	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	Yes	Stepping cycle
G74	Cutting feed	Yes	Forward rotation	Cutting feed	-	Reverse tapping cycle
G75	Cutting feed	-	-	Rapid traverse	-	Circular cutting cycle
G76	Cutting feed	-	Oriented spindle stop	Rapid traverse	-	Fine boring cycle

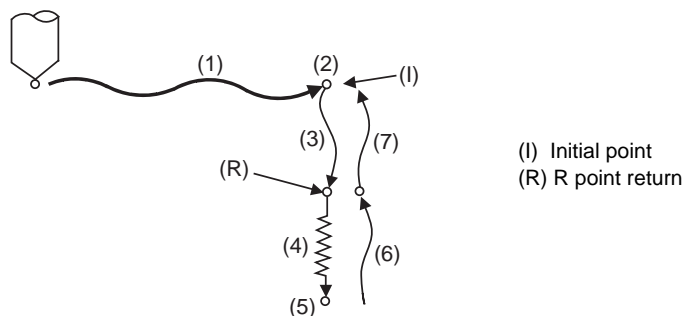
A fixed cycle mode can be canceled by G80 command and other hole machining modes or G command in the 01 group. At the same time, various other data will also be cleared to zero.



Detailed description

Basic operations of fixed cycle for drilling

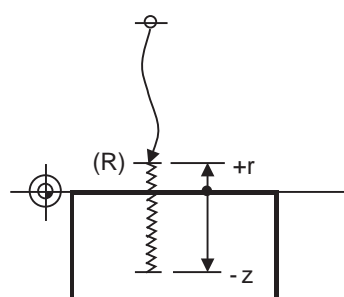
There are 7 actual operations which are each described below.



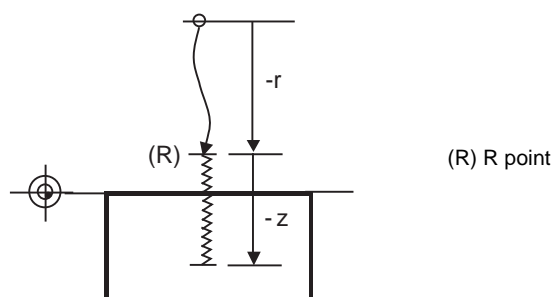
- (1) This indicates the X and Y axes positioning, and executes positioning with G00.
 - (2) This is an operation done after positioning is completed (at the initial point), and when G87 is commanded, the M19 command is output from the control unit to the machine. When this M command is executed and 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



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.

G81Z_ ; The Z axis is used as the hole drilling axis.

G81W_ ; 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 eight fixed cycles; G81 (drill, spot drill), G82 (drill, counter boring), G83 (deep drill cycle), G84 (tap cycle), G85 (boring), G89 (boring), G73 (step cycle) and G74 (reverse tap cycle). 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)		

In-position check in fixed cycle

When L (number of repetitions) is designated twice or more times in the fixed cycle, the commanded in-position width will be valid in the repetition block (5) to (8) below.

```

:
G91 G81 X-50. Z-50. R-50. L2 F2000 ,I0.2 ,J0.3;
:
    
```

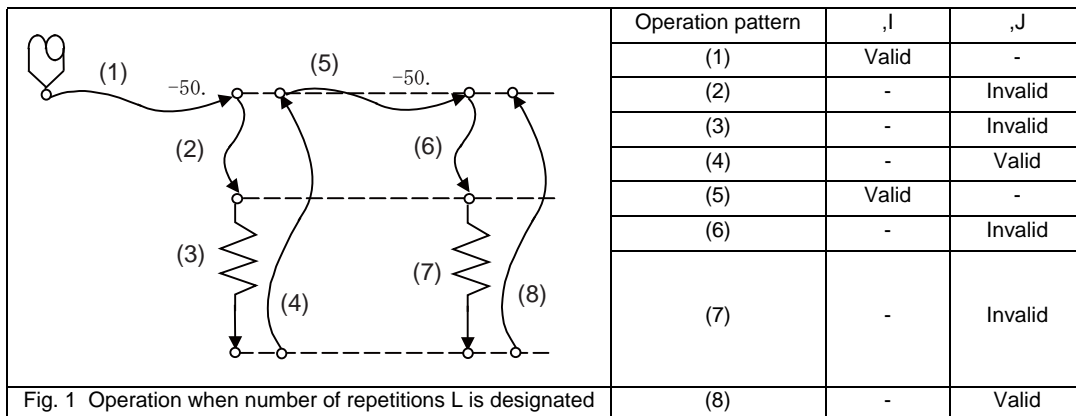


Fig. 1 Operation when number of repetitions L is designated

In the following machining program, the commanded in-position width is valid for the Fig. 2 block. In the (B) block, the in-position width (, I) commanded regarding to positioning in the previous block (A) is invalid (5). However, when returning from the hole bottom, the in-position width (, J) commanded in the previous block (A) is valid (8).

To validate the in-position width for positioning, command again as shown in block (C) (9).

```

:
G91 G81 X-50. Z-50. R-50. F2000 ,I0.2 ,J0.3 ; ..... (A)
X-10. ; ..... (B)
X-10.,I0.2 ; ..... (C)
    
```

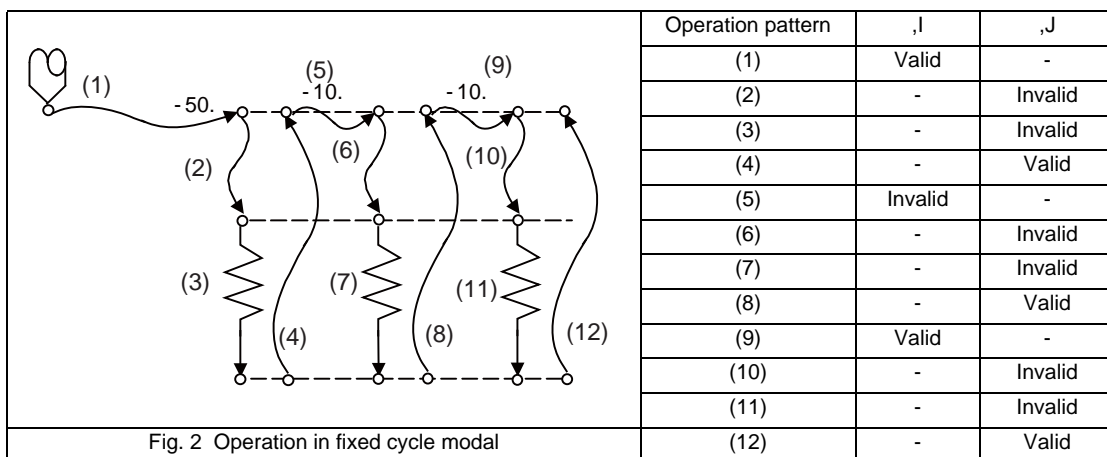


Fig. 2 Operation in fixed cycle modal

13.1.1 Drilling, Spot Drilling ; G81



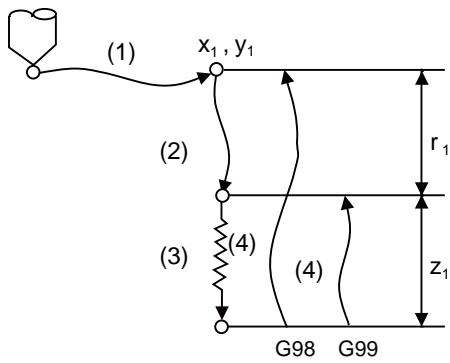
Command format

G81 Xx1 Yy1 Zz1 Rr1 Ff1 LI1,li1,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)
LI1	Designation of number of repetitions. (0 to 9999) When " 0" is set, no execution
,li1	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



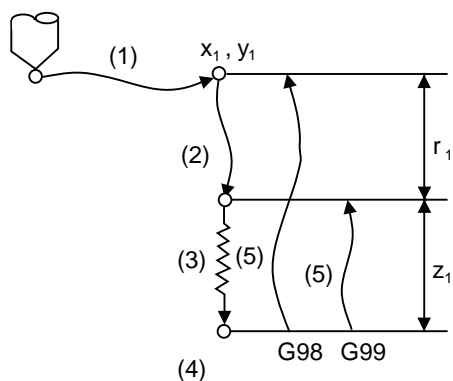
Command format

G82 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1 ,li1 ,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
,li1	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

13.1.3.1 Deep Hole Drilling Cycle

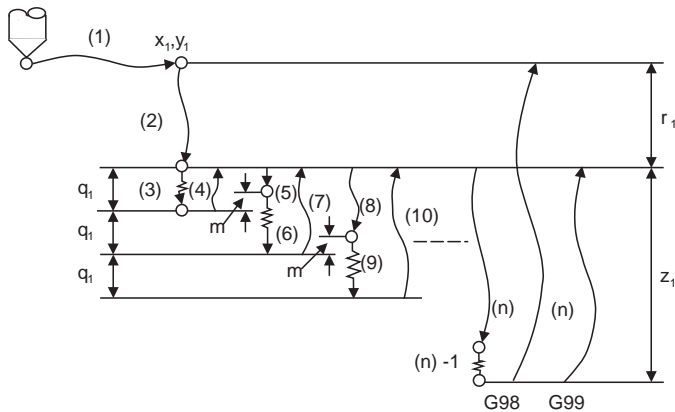


Command format

G83 Xx1 Yy1 Zz1 Rr1 Qq1 Ff1 L1 ,li1 ,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)
L1	Designation of number of repetitions. (0 to 9999) When " 0" is set, no execution
,li1	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 LI1 ,li1 ,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	During synchronous tapping: Designation of drilling axis feed amount (tapping pitch) per spindle revolution During asynchronous tapping: Designation of the feedrate for cutting feed
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 "#8159 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
LI1	Designation of number of repetitions (0 to 9999). When " 0" is set, no execution
,li1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

(Note) ,S command is held as a modal information.

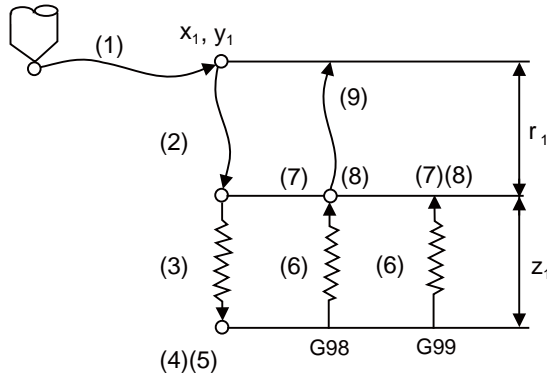
When the value is smaller than the spindle rotation speed (S command), the value of spindle rotation speed is valid even at the return.

If the spindle rotation speed is not "0" at the return, the tapping return override value ("#1172 tapovr") is invalid.



Detailed description

Normal tapping cycle



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.

When it is interrupted by such as the emergency stop during the tapping cycle, a tool can be taken out from the workpiece by enabling the "Tap retract" signal (TRV) .

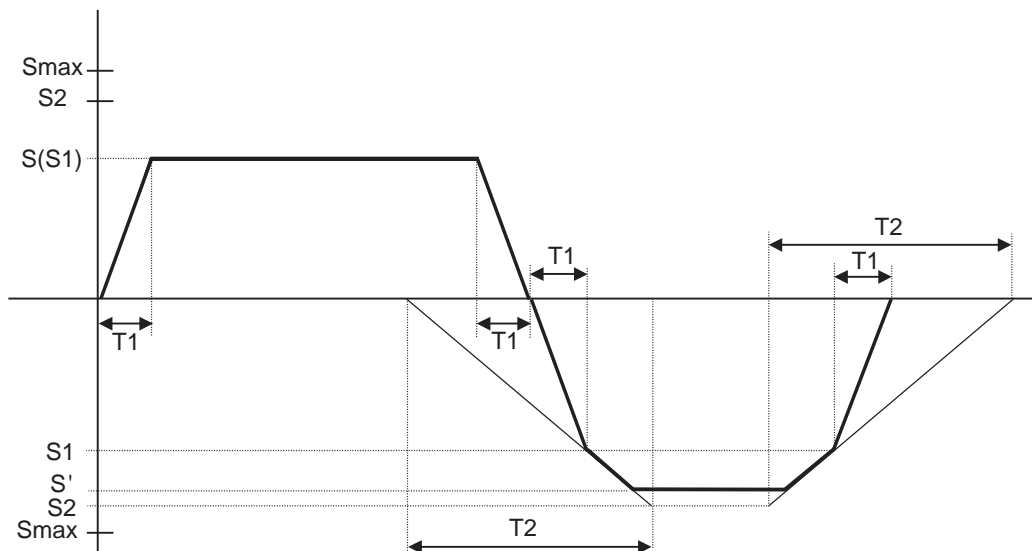
Spindle acceleration/deceleration pattern during synchronous tapping

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.

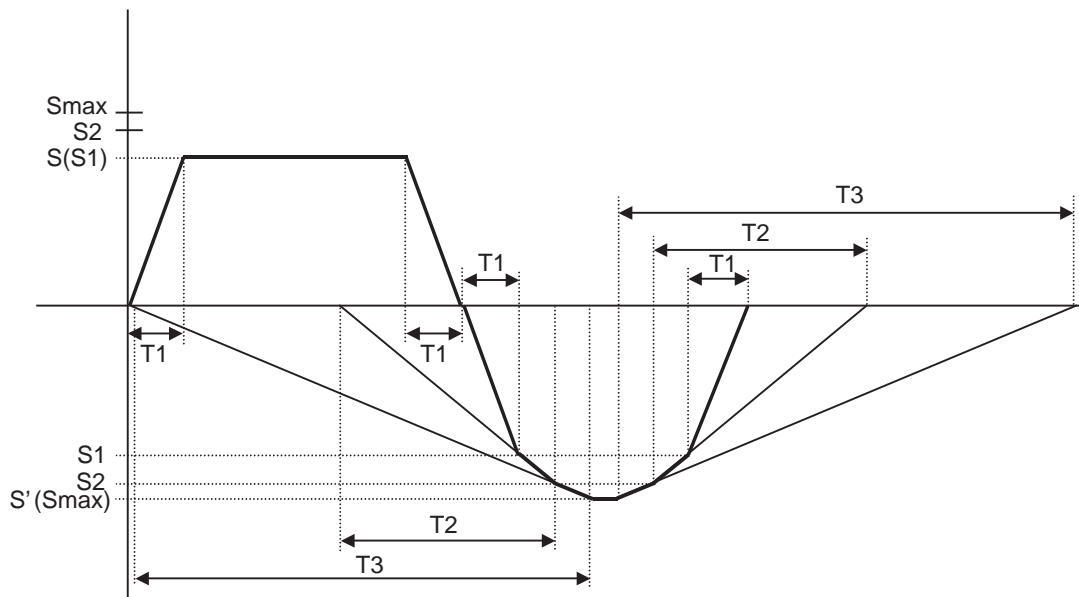
When returning from the hole bottom, rapid return is possible at the spindle rotation speed during return. The spindle rotation speed during return is held as modal information.

- (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

<Program> G84···, Rxx	Control parameter Synchronous tapping	Synchronous/asynchronous
,R00	-	Asynchronous
,Rxx	OFF	
No designation	ON	Synchronous
,R01	-	

- is irrelevant to the setting

(2) Selection of asynchronous tapping cycle feedrate

G94/G95	Control parameter F1-digit valid	F command value	Feed designation
G94	OFF	F designation	Feed per minute
	ON	Other than F0 to F8	
G95	-	F0 to F8 (no decimal point)	F1-digit feed
		F designation	Feed per revolution

- 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 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.

M code for forward/reverse rotation command

The M code set with the parameter "#3028 sprcmn" is output as the M code for spindle forward/reverse rotation that is output at "hole bottom" or at "R point" during asynchronous tapping cycle.

Note that the M code for forward rotation is output as "M3" and that for reverse rotation is as "M4" if the parameter "#3028 sprcmm" is set to "0".

Feed per minute command of the synchronous tapping

Enable the feed per minute command of the synchronous tapping by the setting of parameter "#1268 ext04/bit2". When this parameter is valid, G94 and G95 modal will be applied.

	During G94 modal (feed per minute)	During G95 modal (feed per revolution)
#1268/bit2 = ON	Feed per minute	Feed per revolution
#1268/bit2 = OFF	Feed per revolution	Feed per revolution

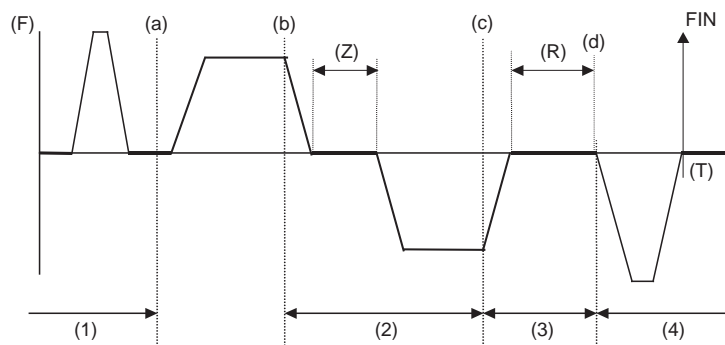
Parameter setting values and tapping axis

(1) Synchronous tapping in-position check settings

#1223 aux07				"P" designation of G84/G74 command	In-position check during synchronous tapping
bit3	bit4	bit5	bit2		
In-position check during synchronous tapping	Hole bottom	R point	I point -> R point		
0	-	-	-	-	Carry out in-position check at I point -> R point/R point/hole bottom.
1	-	-	-	No "P" designation Example : G84 F1. Z-5. S1000 R-5.	Hole bottom: Do not carry out in-position check. R point: Do not carry out in-position check. I point -> R point: Do not carry out in-position check.
1	1	1	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Carry out in-position check by tap in-position width. R point: Carry out in-position check. I point -> R point: Carry out in-position check.
1	1	0	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Carry out in-position check by tapping in-position width. R point: Do not carry out in-position check. I point -> R point: Carry out in-position check.
1	0	1	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Do not carry out in-position check. R point: Carry out in-position check. I point -> R point: Carry out in-position check.
1	0	0	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Do not carry out in-position check. R point: Do not carry out in-position check. I point -> R point: Carry out in-position check.
1	1	1	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Carry out in-position check by tapping in-position width. R point: Carry out in-position check. I point -> R point: Do not carry out in-position check.
1	1	0	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Carry out in-position check by tapping in-position width. R point: Do not carry out in-position check. I point -> R point: Do not carry out in-position check.
1	0	1	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom: Do not carry out in-position check. R point: Carry out in-position check. I point -> R point: Do not carry out in-position check.
1	0	0	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	Hole bottom : Do not carry out in-position check. R point: Do not carry out in-position check. I point → R point: Do not carry out in-position check.

(Note1) The I point refers to the initial point.

(2) In-position width and tapping axis movement for a synchronous tapping in-position check



(Z) Hole bottom

(R) R point

- (a) In-position completion of the G0 feed from the R point
- (b) G1 deceleration start at tapping cut-in
- (c) G1 deceleration start at tapping return
- (d) Start of G0 feed to the R point

- (1) Section in which the in-position check is carried out by G0inps.
- (2) Section in which the in-position check is carried out by TapInp.
- (3) Section in which the in-position check is carried out by G1inps.
- (4) Section in which the in-position check is carried out by sv024.

R point: In-position check by the G1inps

I point: In-position check by the G0inps

Hole bottom: In-position check by the Tapinps

13 Program Support Functions

(3) Relation between the parameter setting values and tapping axis movement for a synchronous tapping in-position check

#1223 aux07				Hole bottom wait time	Operation at hole bottom	Operation at R point	Operation at I point -> R point
bit3	bit4	bit5	bit2				
Synchronous tapping in-position check	Hole bottom	R point	I point -> R point				
0	-	-	-	Time designated by "P". Several 10ms as processing time when no "P".	Operation determined by setting of inpos (#1193) and aux 07 (#1223): bit 1 parameters.	Operation determined by setting of inpos (#1193) and aux 07 (#1223): bit 1 parameters.	Operation determined by setting of inpos (#1193) and aux 07 (#1223): bit 1 parameters.
1	0	0	1	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in the left column elapses.		Wait until completion of in-position check by G0inps.
1	0	1	1	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in the left column elapses.	Wait until completion of in-position check by G1inps.	Wait until completion of in-position check by G0inps.
1	1	0	1	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until dwell time in the left column elapses after completion of in-position check.		Wait until completion of in-position check by G0inps.
1	1	1	1	Larger value of "P" and TapDwl (#1313) is valid. Several 10ms as processing time when both of them are "0".	Wait until dwell time in the left column elapses after completion of in-position check.	Wait until completion of in-position check by G1inps.	Wait until completion of in-position check by G0inps.
1	0	0	0	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in the left column elapses.		
1	0	1	0	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until time in the left column elapses.	Wait until completion of in-position check by G1inps.	
1	1	0	0	Larger value of "P" and TapDwl (#1313) is valid. No dwell executed if both values are 0.	Wait until dwell time in the left column elapses after completion of in-position check.		
1	1	1	0	Larger value of "P" and TapDwl (#1313) is valid. Several 10ms as processing time when both of them are "0".	Wait until dwell time in the left column elapses after completion of in-position check.	Wait until completion of in-position check by G1inps.	

(Note1) The I point refers to the initial point.

Remarks: The processing time of several 10ms is needed when an in-position check is not carried out at the hole bottom.

Also, the processing time of between 0 to 14.2ms is required at the R point.

And the spindle gain changeover time at I point -> R point, too.

13.1.5 Boring ; G85



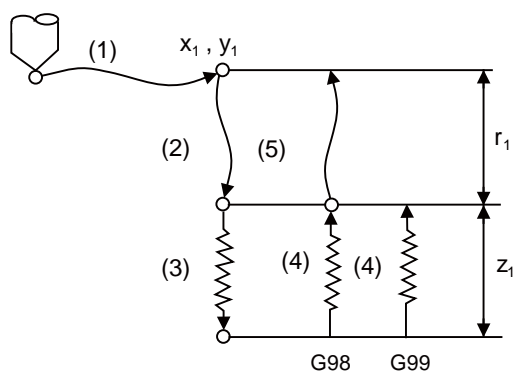
Command format

G85 Xx1 Yy1 Zz1 Rr1 Ff1 L1 ,li1 ,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)
L1	Designation of number of repetitions. (0 to 9999) When "0" is set, no execution
,li1	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)	-	Invalid	G1 Z-z1 Ff1
(5)	-	Invalid	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



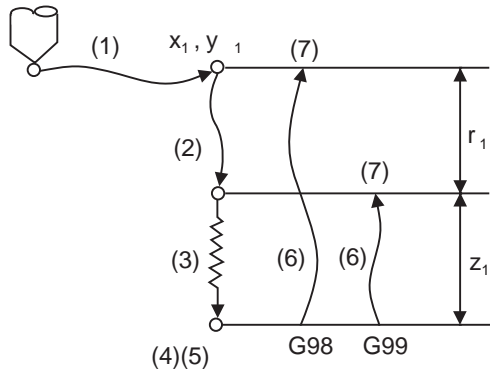
Command format

G86 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 LI1 ;

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)
LI1	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



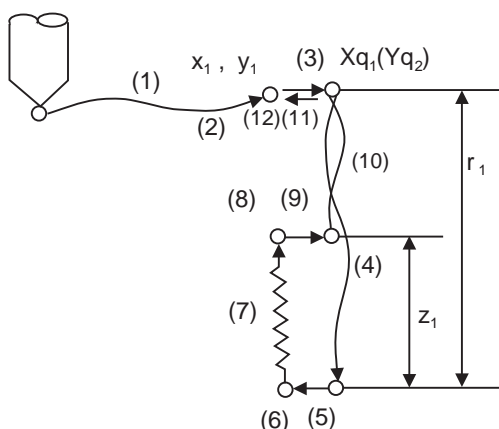
Command format

G87 Xx1 Yy1 Zz1 Rr1 Iq1 Jq2 Kq3 Ff1 L1;

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)
L1	Designation of number of repetitions. (0 to 9999) When " 0" is set, no execution



Detailed description

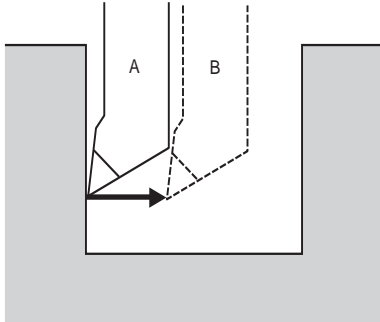


Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	M19 (Spindle orientation)
(3)	G0 Xq1 (Yq2) (shift)
(4)	G0 Zr1
(5)	G1 X-q1 (Y-q2) Ff1 (shift)
(6)	M3 (Spindle forward rotation)
(7)	G1 Zz1 Ff1
(8)	M19 (Spindle orientation)
(9)	G0 Xq1 (Yq2) (shift)
(10)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - (r1+z1)
(11)	G0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



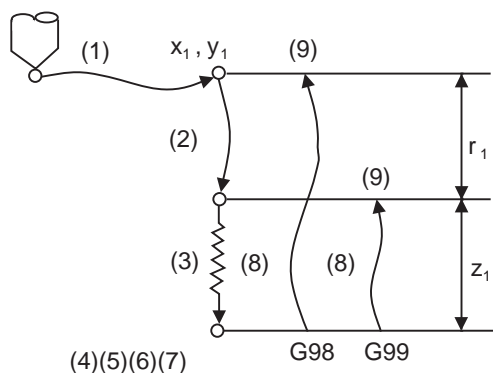
Command format

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



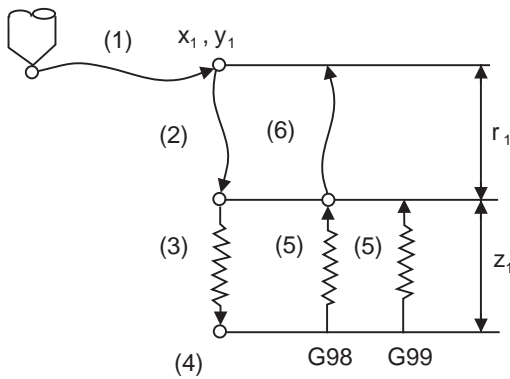
Command format

G89 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 LI1 ,li1,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) (modal)
LI1	Designation of number of repetitions. (0 to 9999) When " 0" is set, no execution
,li1	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)	-	-	G4 Pp1
(5)	-	Invalid	G1 Z-z1 Ff1
(6)	-	Valid	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



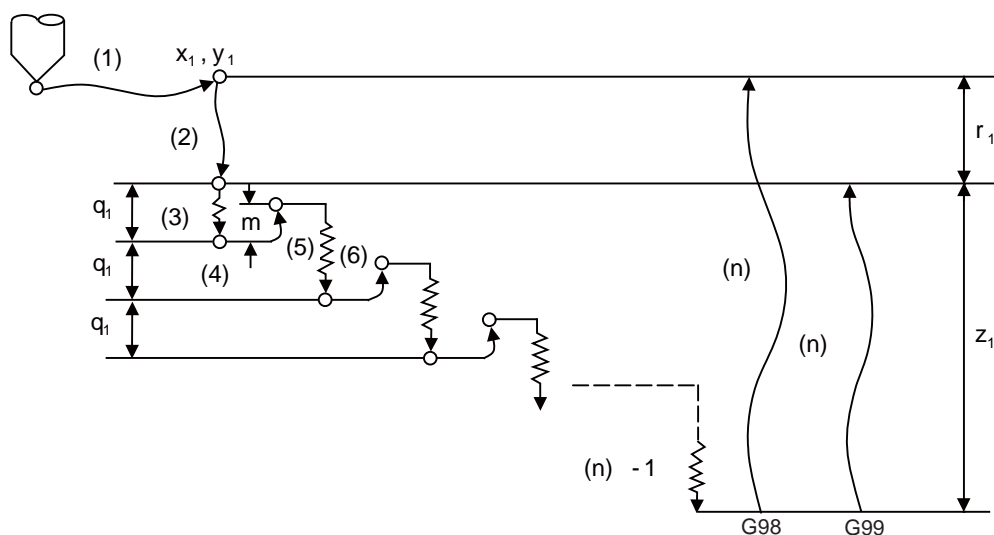
Command format

G73 Xx1 Yy1 Zz1 Qq1 Rr1 Ff1 Pp1 Ll1 ,li1 ,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)
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
,li1	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)	-	-	G4 Pp1
(5)	-	Invalid	G0 Z-m
(6)	-	Invalid	G1 Z(q1+m) Ff1
:			
(n)-1	-	Invalid	
(n)	-	Valid	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

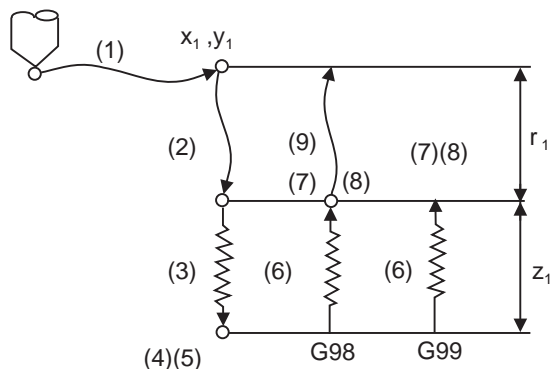
G74 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 ,Rr2 Ss1 ,Ss2 LI1 ,li1,Jj1;

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 "#8159 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
LI1	Designation of number of repetitions. (0 to 9999) When " 0" is set, no execution
,li1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

(Note) When asynchronous tapping mode is applied, F address becomes the cutting feed speed.



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".

Feedrate for tapping cycle and tapping return

Refer to "Tapping cycle; G84".

M code for forward/reverse rotation command

Refer to "Tapping cycle; G84".

Parameter setting values and tapping axis

Refer to "Tapping cycle; G84".

13.1.12 Circular Cutting ; G75



Function and purpose

Circle cutting starts with the X and Y axes positioned at the center of the circle, and the Z axis cuts into the commanded position. Then, the tool cuts the inner circumference of the circle drawing a true circle and returns to the center position.



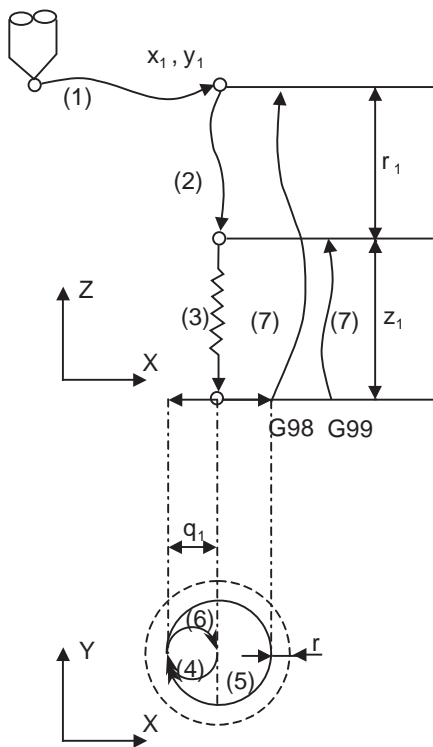
Command format

G75 Xx1 Yy1 Zz1 Rr1 Qq1 Pp1 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)
Qq1	Radius of outer circumference (modal)
Pp1	Tool radius compensation No. (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)	Gn X-(q1-r) I-(q1/2) Inner circumference half circle	n:q1 >= 0 -> G02 q1 < 0 -> G03 r: Tool radius compensation amount of the No. commanded with p1.
(5)	Iq1	
(6)	X(q1-r) I(q1/2) Inner circumference half circle	
(7)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1	

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

13.1.13 Fine Boring ; G76



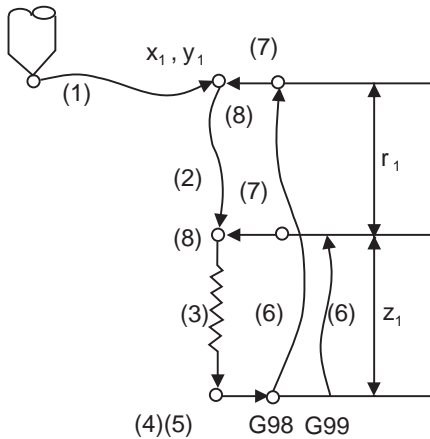
Command format

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



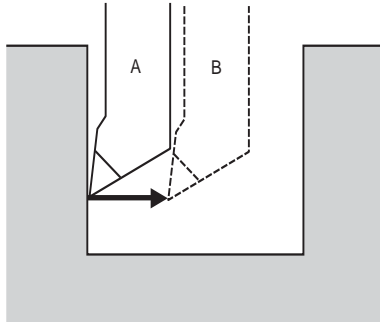
Operation pattern	Program
(1)	G0 Xx1 Yy1
(2)	G0 Zr1
(3)	G1 Zz1 Ff1
(4)	M19 (Spindle orientation)
(5)	G1 Xq1 (Yq2)Ff1 (shift)
(6)	G98 mode G0 Z - (z1+r1) G99 mode G0 Z - z1
(7)	G0 X-q1 (Y-q2) (shift)
(8)	M3 (Spindle forward rotation)

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.

13.1.14 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

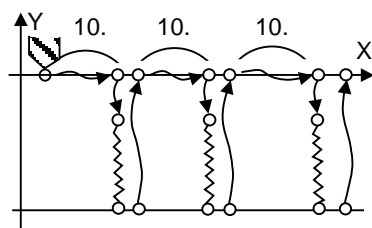
Gm Gn X_Y_Z_R_Q_P_L_F_;

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 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
- (8) 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.
- (9) Other subprograms cannot be called from the fixed cycle subprogram.
- (10) Decimal points in the movement command will be ignored during the fixed cycle subprogram.
- (11) 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;



- (12) If the spindle rotation speed value during return is smaller than the spindle rotation speed, the spindle rotation speed value is valid even during return.
- (13) If the 2nd and 3rd acceleration/deceleration stage inclinations following the spindle rotation speed and time constants set in the parameters are each steeper than the previous stage's inclination, the previous stage's inclination will be valid.
- (14) If the values set in the spindle base specification parameter "tap rotation speed" and "the synchronous tap changeover spindle rotation speed 2" exceed the maximum rotation speed, the spindle rotation speed will be clamped at the maximum rotation speed.
- (15) If the spindle rotation speed during return is not 0, the tap return override value will be invalid.
- (16) In a block where the movement direction of any axis reverses as shown below, the servo system load will greatly increase, so do not command the in-position width in the machining program.
 G0 X100. ,I10.0;
 X-200.;
- (17) If the in-position width commanded with the programmable in-position width command is increased, the positioning time and linear interpolation time can be reduced. However, the positioning error amount of the previous block will also increase when the next block starts, so the actual machining could be obstructed.
- (18) The in-position width and the positioning error amount are compared at a set time, so the positioning error amount at the in-position point will be smaller than the commanded in-position width.
- (19) If the in-position width commanded with the programmable in-position command is small, the commanded deceleration check or in-position check following the parameters may be carried out first.
- (20) Synchronous or asynchronous tapping can be selected with the M function.

Basic specification parameters

#	Item		Details	Setting range
1272 (PR)	ext08	bit 1	M-function synchronous tapping cycle valid.	0: Invalid 1: Valid

Synchronous tapping cannot be selected with the M function when this parameter is OFF.

Basic specification parameters

#	Item	Details	Setting range
1513	stapM	M code for synchronous tapping selection	0 to 99999999

The synchronous tapping mode is selected with the miscellaneous function code set with this parameter. The M function can be commanded just before or in the same block as the tapping command. When using this parameter, validate "#1272 ext08/bit1 M function synchronous tap cycle".

The selection of synchronous or asynchronous tapping will follow the combination shown below.

Program command (,R0/1)	Combination											
	0	0	0	0	1	1	1	1	No command			
#8159 Synchronous tap	0	0	1	1	0	0	1	1	0	0	1	1
M function code (M**)	x	○	x	○	x	○	x	○	x	○	x	○
Synchronous/asynchronous selection	A	A	A	A	S	S	S	S	A	S	S	S

- x : Does not command
- : Commands
- A : Asynchronous tap
- S : Synchronous tap

(Note1) Do not use M00, 01, 02, 30, 98 or 99.

- (21) Even when the parameter "#1151 rstinit" is OFF, the fixed cycle will be canceled if NC reset 1 is carried out while executing the fixed cycle.

13.1.15 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 ;	<p>Initial level return is executed.</p>	<p>R point level return is executed.</p>
Two or more executions	G81 X100. Y100. Z-50. R25. L5 F1000 ;	<p>Initial level return is always executed.</p>	

- (a) First time
- (b) Second time
- (c) Last time

13.1.16 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;

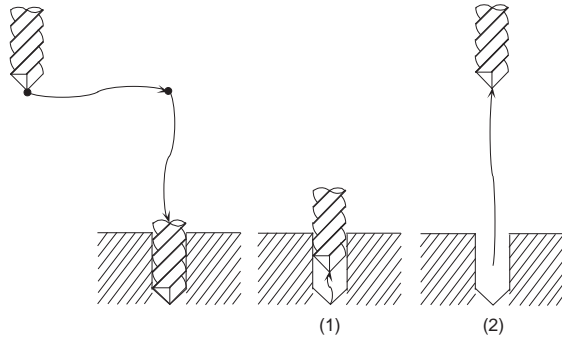
Xx5 Yy5;

13.1.17 Drilling Cycle with High-Speed Retract



Function and purpose

This function retracts the drill from the hole bottom at high speed in drilling machining. This helps extending the drill life by reducing the time of drilling in vain at hole bottom.



The drill moves up at high-speed (1) and returns to the initial point or R point in rapid traverse (2).



Command format

The command format is the same as fixed cycle.

**Detailed description**

- (1) This function is available only when "#8123 H-spd retract ON" is enabled in the following fixed cycles.
 - G81 (Drill spot drilling cycle)
 - G83 (Deep whole drilling cycle)
 - G73 (Step cycle)

- (2) When "#8123 H-spd retract ON" is ON, the axis is retracted from the hole bottom at high speed with lost motion compensation.
 - (a) Set the lost motion compensation type 2 or 3 to the servo parameter. Then set the following parameters to adjust the retract amount.
 - #2170 Lmc1QR (Lost motion compensation gain 1 for high-speed retract)
(correspond to "#2216 SV016 LMC1 Lost motion compensation 1")
 - #2171 Lmc2QR (lost motion compensation gain 2 for high-speed retract)
(correspond to "#2241 SV041 LMC2 Lost motion compensation 2")

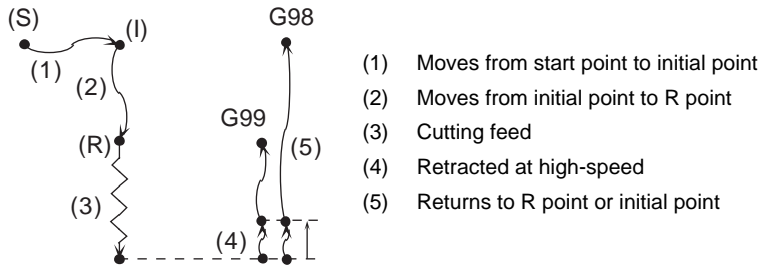
 - (b) Set the following parameters for lost motion compensation timing or lost motion compensation 3 spring constant/ viscous coefficient in addition to the ordinary lost motion compensations.
 - #2172 LmcdQR (Lost motion compensation timing for high-speed retract)
(correspond to "#2239 SV039 LMCD Lost motion compensation timing)
 - #2173 LmckQR (Lost motion compensation 3 spring constant for high-speed retract)
(correspond to "#2285 SV085 LMCK Lost motion compensation 3 spring constant")
 - #2174 LmccQR (Lost motion compensation 3 viscous coefficient for high-speed retract)
(correspond to "#2286 SV086 LMCc Lost motion compensation 3 viscous coefficient")

- (3) While G80 (Fixed cycle cancel) command is issued, this function is canceled by issuing any other fixed cycle of the same group (Group 9) or any Group 1 command.



Details of Operation

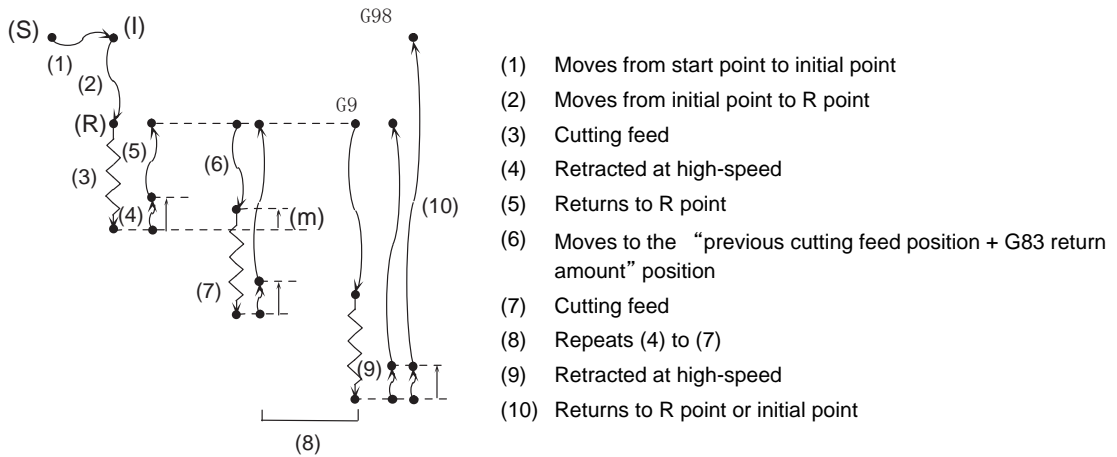
Operation at G81 command



- (I) Initial point
- (S) Start point
- (R) R point

During single block operation, the axis stops after (1), (2) and (5) only.

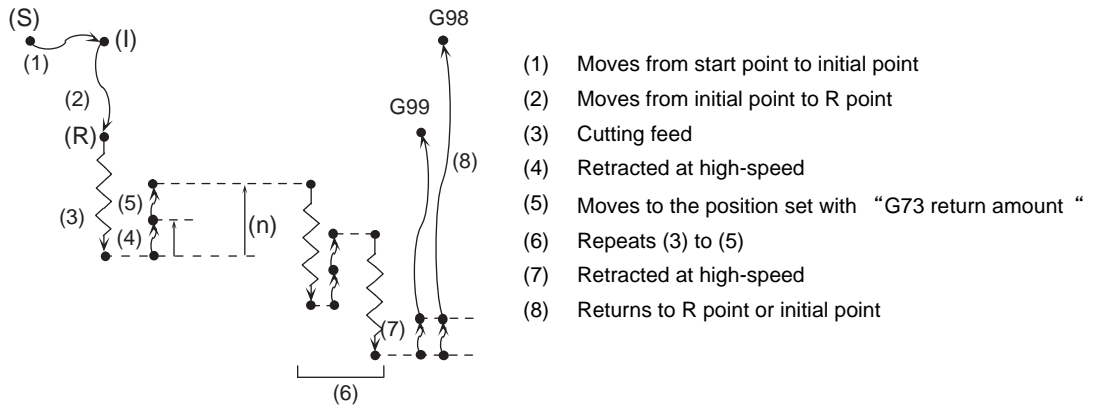
Operation at G83 command



- (I) Initial point
- (S) Start point
- (R) R point
- (m) G83 Return amount

During single block operation, the axis stops after (1), (2) and (10) only.

Operation at G73 command



- (I) Initial point
- (S) Start point
- (R) R point
- (n) G73 Return amount

During single block operation, the axis stops after (1), (2) and (8) only.

If a dwell command is issued, the high-speed retract will be executed after the command.

13.1.18 Acceleration/Deceleration Mode Change in Hole Drilling Cycle



Function and purpose

This function switches the acceleration/deceleration mode for hole drilling cycle between the inclination-constant method and the acceleration/deceleration after interpolation.



Command format

The command formats are the same as those of the fixed cycles G81,G82,G83 and G73.



Detailed description

With parameter "#1253 set25/bit2 Acceleration/deceleration mode change in hole drilling cycle" enabled, operation will be as follows.

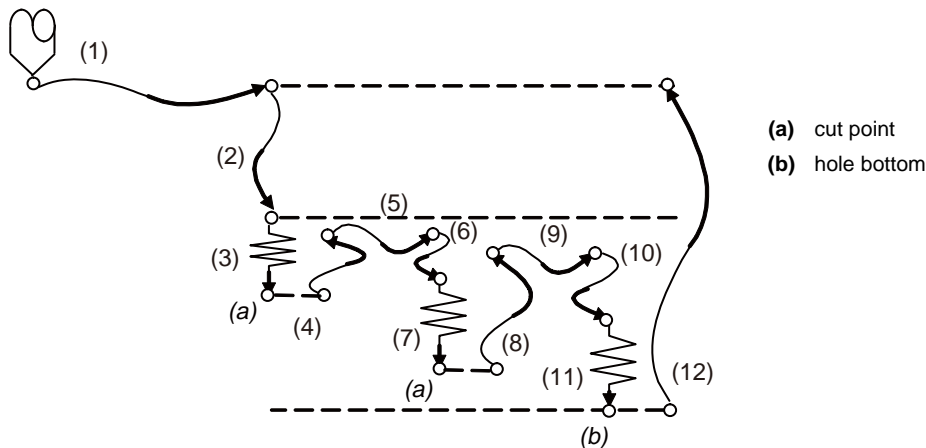
- (1) Acceleration/deceleration mode will be either linear or soft method. (Unless soft acceleration/deceleration is applied, the linear method will always be applied.)
- (2) The operation follows the parameter settings that determine whether the inclination-constant or post-interpolation acceleration/deceleration is applied. "#2001 rapid (rapid traverse rate)" and "#2004 G0tL (G0 time constant (linear))" provide G0 (rapid traverse) acceleration/deceleration inclination, and "#2002 clamp (cutting feedrate for clamp)" and "#2007 G1tL (G1 time constant (linear))" provide G1 (cutting feed) acceleration/deceleration inclination. Refer to "Rapid Traverse Constant Inclination Acceleration/Deceleration" for details of constant inclination acceleration/deceleration.



Operation example

Operation example of "acceleration/deceleration mode change in hole drilling cycle" being enabled

The below illustrates the processes of hole-bottom deceleration check of a drilling axis following the parameter "#19417 Hole dec check 2" settings.



#19417		G81	G82	G83	G73
0	(a)Cut point	Perform deceleration check.		Perform no deceleration check.	
	(b)Hole bottom	Perform no deceleration check.			
1	(a)Cut point	Perform no command deceleration check.		Perform command deceleration check.	
	(b)Hole bottom	Perform command deceleration check.			
2	(a)Cut point	Perform command deceleration check.			Perform in-position check (sv024).
	(b)Hole bottom	Perform in-position check (sv024).			

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.

The tool is positioned to the hole drilling position when the special fixed cycle is executed. The drilling operation is executed with the fixed cycle for drilling.

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.

If the special fixed cycle is commanded without commanding the fixed cycle for drilling, positioning will be executed following the current 01 group modal G code.

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.



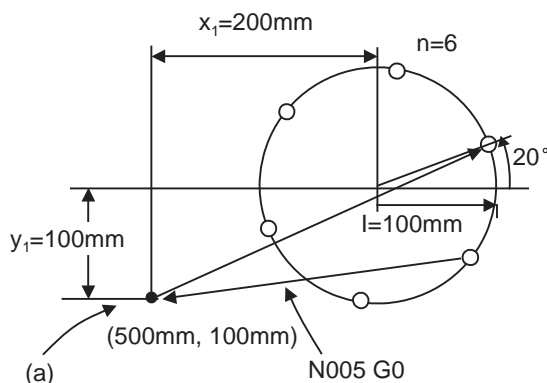
Command format

G34 Xx1 Yy1 Ir J θ 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.
J θ	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



(a) Position before G34 is executed

```

N001 G91;
N002 G81 Z-10.000 R5.000 L0 F200 ;
N003 G90 G34 X200.000 Y100.000 I100.000 J20.000 K6;
N004 G80 ; ----- (G81 cancel)
N005 G90 G0 X500.000 Y100.000 ;

```

As shown in the example, the tool position after the G34 command is completed is 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.

(Note 1) If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G34 command, a program error (P32) will occur.

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.



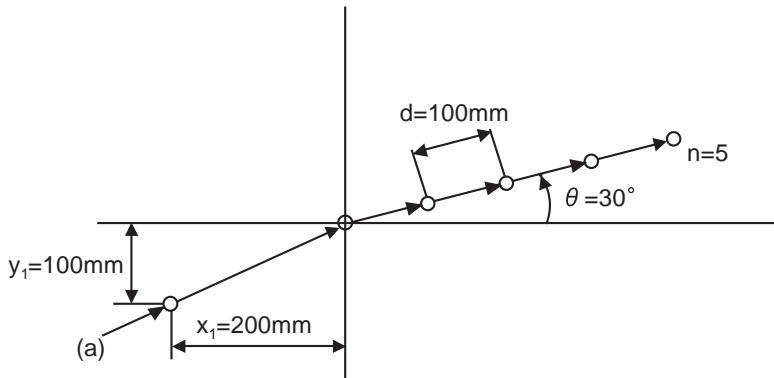
Command format

G35 Xx1 Yy1 Id J θ 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.
J θ	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 an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G35 command, a program error (P32) will occur.
- (Note 4) If 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 5) 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.



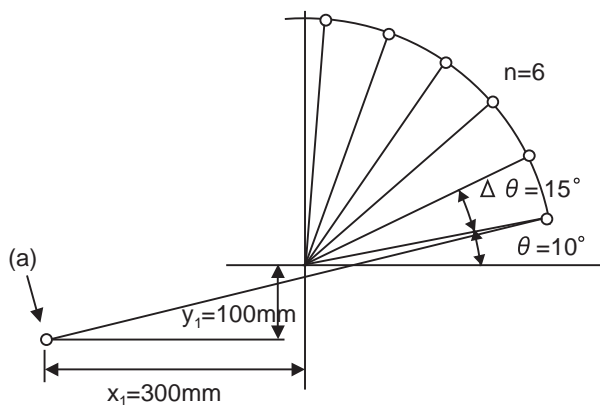
Command format

G36 Xx1 Yy1 Ir J θ P $\Delta \theta$ 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.
J θ	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 $\Delta \theta$	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



(a) Position before G36 is executed

```

N001 G91 ;
N002 G81 Z-10.000 R5.000 F100 ;
N003 G36 X300.000 Y100.000 I300.000
      J10.000 P15000 K 6 ;

```

(Note 1) If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G36 command, a program error (P32) will occur.

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.

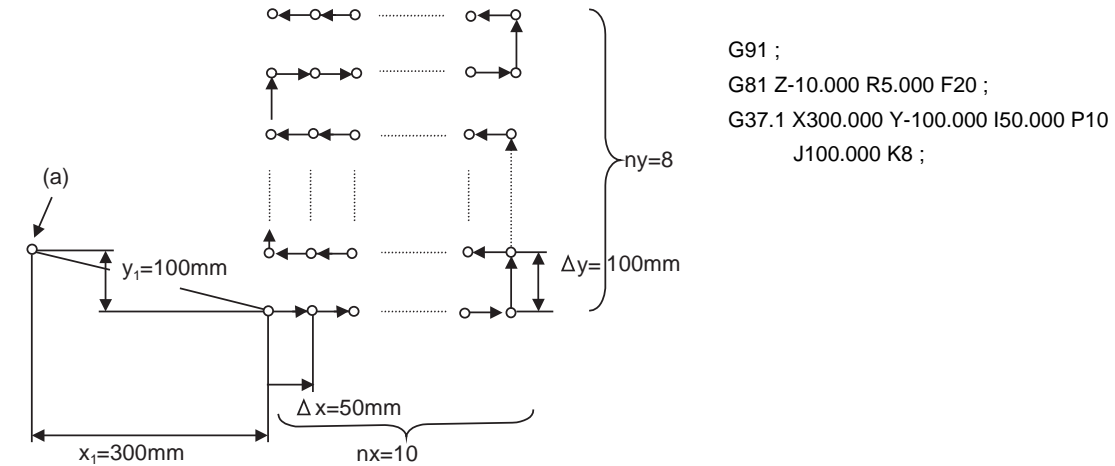


Command format

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



(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 the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G37.1 command, a program error (P32) will occur.
- (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.
- (Note 5) If the G22/G23 command is programmed in the same block as the G37.1 command, the G22/G23 command will be ignored, and the G37.1 command will be executed.

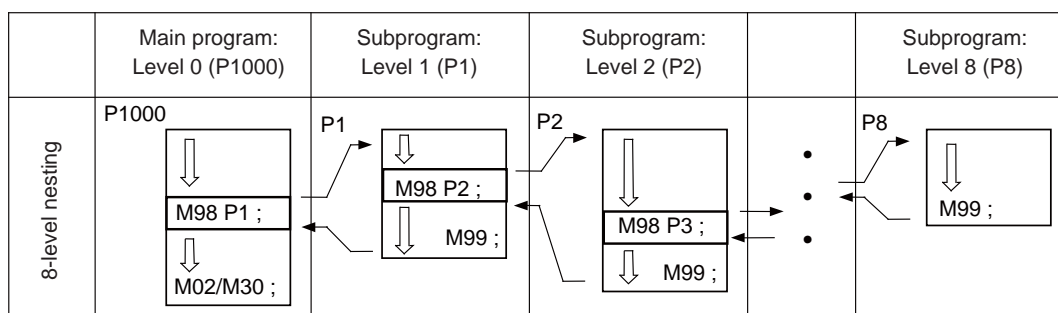
13.3 Subprogram Control; M98, M99, M198

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 tape memory/editing functions, subprogram control functions and fixed cycle functions.

	Case 1	Case 2	Case 3	Case 4
1. Tape memory and editing	Yes	Yes	Yes	Yes
2. Subprogram control	No	Yes	Yes	No
3. Fixed cycles	No	No	Yes	Yes
Function				
1. Memory mode	○	○	○	○
2. Tape editing (main memory)	○	○	○	○
3. Subprogram call	×	○	○	×
4. Subprogram variable designation (Note 2)	×	○	○	×
5. Subprogram nesting level call (Note 3)	×	○	○	×
6. Fixed cycles	×	×	○	○
7. 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.

(Note 3) A maximum of 8 nesting levels form the nesting depth.



Command format

Subprogram call

M98 P__ H__ L__ ,D__ ;

M98 <file name> H__ L__ ,D__ ;

P	<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) Use a parameter to specify a 4- or 8-digit subprogram No. starting with O. However, if the commanded value is bigger than the digit number set with parameter, a subprogram call is carried out as commanded.</p>
<File name>	<p>File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets <>. (The file name can have up to 32 characters including the extension.) (Example) M98 <BUHIN-12. RAF> ;</p>
H	<p>Sequence No. in subprogram to be called (head block if omitted) (Max. 5 digits)</p>
L	<p>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 ;</p>
,D	<p>Subprogram device No. (0 to 4). The subprogram is searched according to the setting of parameter "#8890 Subpro srch odr D0" to "#8894 Subpro srch odr D4" when ,D is omitted. The device No. is set to the parameter, such as "#8880 Subpro stor D0: dev".</p>

Return to main program from subprogram

M99 P__ ;

P	Sequence No. of return destination (returned to block that follows the calling block)
---	---



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

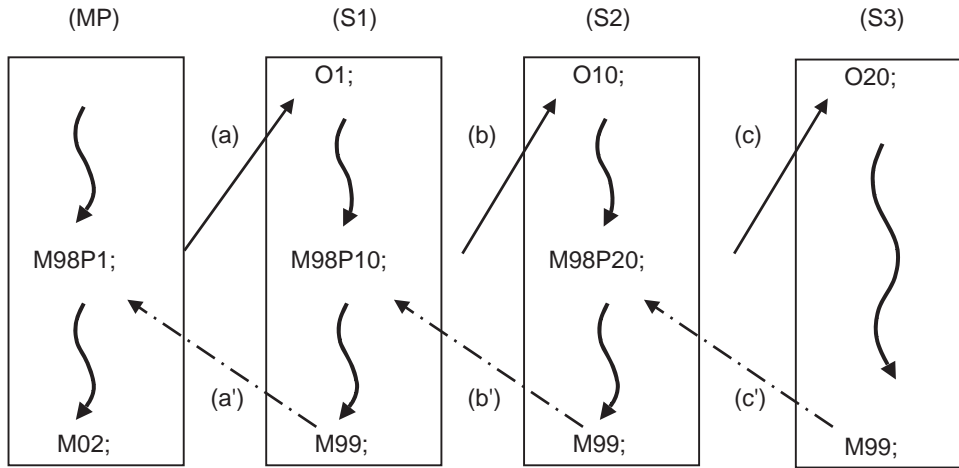
- (1) The above program is registered by editing operations at the setting and display unit. For further details, refer to the section on "program editing" in the Instruction Manual.
- (2) Only those subprogram Nos. ranging from 1 to 99999999 designated by the optional specifications can be used. When there are no program Nos. on the tape, they are registered as the setting No. for "program input."
- (3) Up to 8 nesting levels can be used for calling programs from subprograms and program error (P230) will occur if this number is exceeded.
- (4) Main programs 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.)
- (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
 - Pattern 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

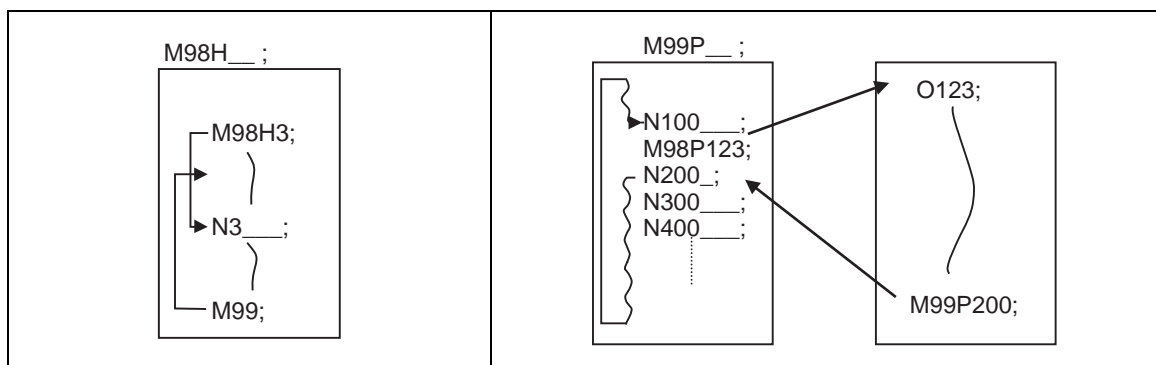
(S3) Subprogram 3

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.





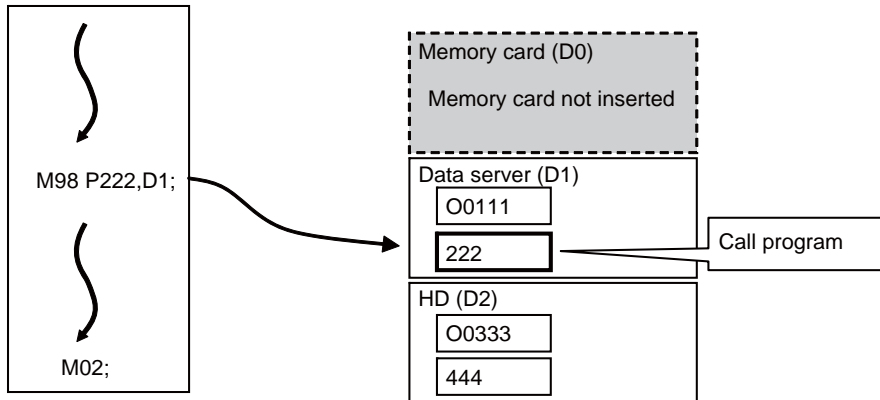
Precautions

- (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) Branching from tape to the subprogram with M98 P_ ; is possible, but the return destination sequence No. cannot be designated with M99 P_ ;. (P_ is ignored.)
- (5) Note that it takes time to search when the sequence No. is designated by M99 P_ ;.
- (6) When using a file name for the subprogram, specify the file name with 32 characters or less, including the extension. If a file name exceeding 32 characters is specified, a program error (P232) will occur.
- (7) All the programs are registered as files. For example, when calling a file "0100" as a subprogram, "0100" cannot be searched with M98P100 or M98P0100. When numerical values are specified after P, 0 is ignored. In this case, it is regarded that the program No. (file) "100" is specified. To call a program like "0100" , specify the file name using the M98<0100> format.
- (8) A subprogram added O No. is searched with the parameter setting (#8129="1" or "2") which calls a subprogram with O No. as priority.
If a subprogram with O No. is not found, a subprogram with a name specified with the P command is searched.
(Note) To prevent any unintended program call, avoid using such program names that may be confused.
(For example, 123, O0123 and O00000123 can be considered identical.)

Refer to the next page for operation examples of subprogram search with the setting which calls subprograms with O No. as priority.

(a) With designation of device No.
Only the designated devices are subject to search.

Main program



[Parameter setting]

- #8129 Subpro No. select = 1 (Four-digit program No. beginning with O No.)
- #8880 Subpro stor D0 dev = R(Memory card)
- #8882 Subpro stor D1 dev = D(Data server)
- #8884 Subpro stor D2 dev = G(Hard disc)

(b) Without designation of device No.

A subprogram with O No. is searched according to the settings of #8890(D0 in order of subprogram search) to #8894(D4 in order of subprogram search). (Refer to the solid line arrows a and b.)

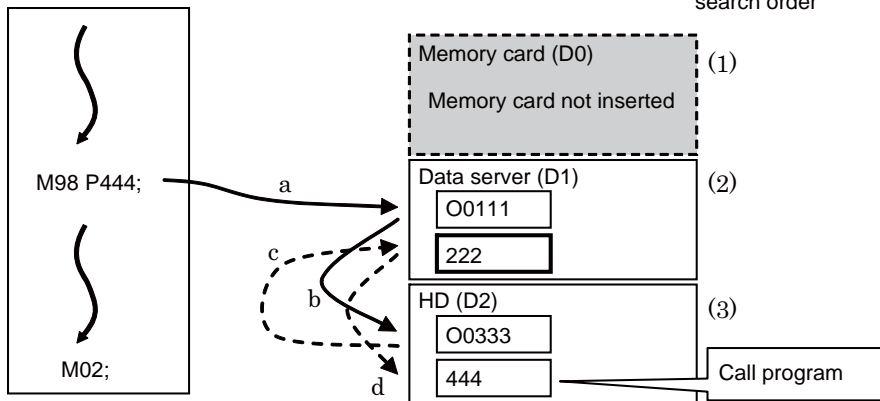
If a subprogram with O No. is not found, subprograms with a name designated with the P command are searched in order of the subprogram search setting. (Refer to the broken lines c,d.)

If none of the designated subprogram storage locations are subject to search, memories are searched.

* If any device or directory designated as the subprogram storage location is not found due to a reason such as absence, poor contact and contact failure of a memory card, the said device or directory will be excluded from the search target.

Main program

Subprogram search order



[Parameter setting]

- #8129 Subpro No. select = 1 (Four-digit program No. beginning with O No.)
- #8880 Subpro stor D0 dev = R(Memory card)
- #8882 Subpro stor D1 dev = D(Data server)
- #8884 Subpro stor D2 dev = G(Hard disc)
- #8890 Subpro srch odr D0 = 1
- #8891 Subpro srch odr D1 = 2
- #8892 Subpro srch odr D2 = 3

13.3.2 Subprogram Call ; M198



Function and purpose

Programs registered in the front IC card (CF card) can be called as subprograms. To call a program in the CF card as a subprogram, command in the main program as shown below.



Command format

Subprogram call

M198 P__ L__ ;

M198 <File name> L__ ;

P	Program No. in CF card to be called as subprogram. (Max. 8 digits) Use a parameter to specify a 4- or 8-digit subprogram No. starting with O. However, if the commanded value is bigger than the digit number set with parameter, a subprogram call is carried out as commanded.
<File name>	File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets <>. (The file name can have up to 32 characters including the extension.)
L	Number of subprogram repetitions. (Max. 4 digits) This can be omitted. (In this case, the subprogram will be called once.) When "L0" is designated, the subprogram call will not be executed.

Return to main program from subprogram

M99 ;



Detailed description

- (1) The subprogram can be called with the M198 command once in the subprogram nest. The subprogram can be called only from the memory or MDI program.
- (2) The section from the head of the program to the first LF (line feed code, 0x0A hexadecimal) is invalid, and is not run or displayed. Note that if the head starts with a 0 No., the program will be valid from the head.
- (3) Refer to "Subprogram call ; M98,M99" for <File name> and calling the subprogram with O No.

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

```
##*** = 00000000 ;
```

```
##*** = [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.

Type	No.		Function
Common variables	Common variables 1	Common variables 2	- Can be used in common throughout main, sub and macro programs.
200 sets	500 to 599	100 to 199	
Local variables	1 to 33		Can be used as local variables in macro programs.
System variable	From 1000		Application is fixed by system.
Fixed cycle variables	1 to 32		Local variables in fixed cycle programs.

(Note 1) All common variables are retained even when the power is turned OFF.

(Note 2) When the power is turned OFF or reset, the common variables can be set to <null> by setting the parameter (#1128 RstVC1, #1129 PwrVC1).

(Note 3) When inputting the common variable data, if the following illegal variable No. data exist in the input file, the illegal variable No. data is ignored and only the correct common variable data will be input.

- Variable data which is not common variables such as local variables (#1 to #33) and system variables (from #1000), etc.
- Variable data with which conditions of number of common variable sets do not match.

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

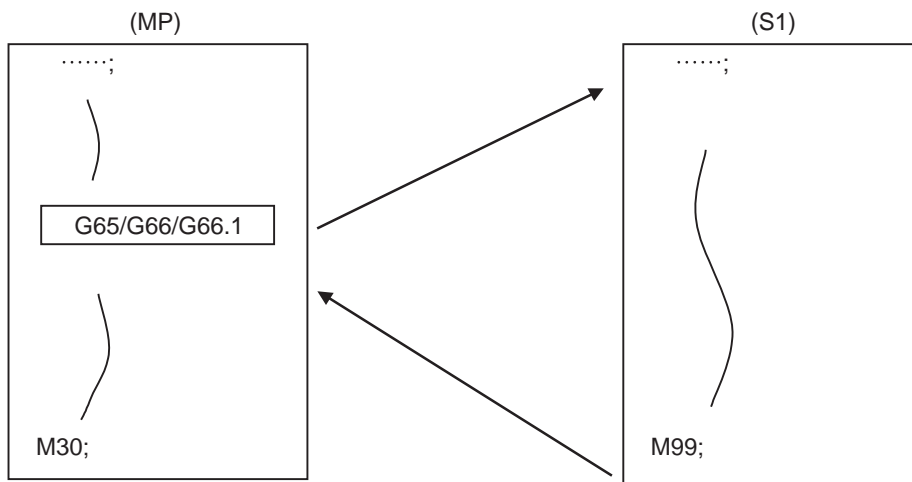


Function and purpose

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



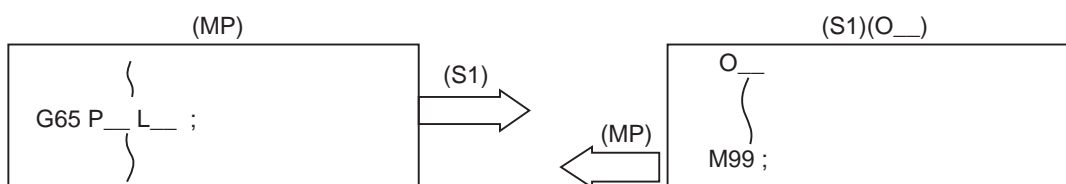
Function and purpose

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

G65 <File name> L__ argument ; ... Simple macro calls

P	Program No. Use a parameter to specify a 4- or 8-digit subprogram No. starting with O.
<File name>	File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets <>. (The file name can have up to 32 characters including the extension.)
L	Number of repetitions:
Argument	Specify variable data



Detailed description

- (1) 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_

- (a) Arguments can be designated using any address except G, L, N, O and P.
- (b) I, J and K must be designated in alphabetical order.
I_ J_ K_ Correct
J_ I_ K_ Incorrect
- (c) Except for I, J and K, there is no need for designation in alphabetical order.
- (d) Addresses which do not need to be designated can be omitted.
- (e) The following table shows the correspondence between the addresses which can be designated by argument designation I and the variable 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	x	x *
H	#11	○	○
I	#4	○	○
J	#5	○	○
K	#6	○	○
L	#12	x	x *
M	#13	○	○
N	#14	x	x *
O	#15	x	x
P	#16	x	x *
Q	#17	○	○
R	#18	○	○
S	#19	○	○
T	#20	○	○
U	#21	○	○
V	#22	○	○
W	#23	○	○
X	#24	○	○
Y	#25	○	○
Z	#26	○	○

○ : Can be used
 x : 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__ ...

- (a) In addition to address A, B and C, up to 10 groups of arguments with I, J, K serving as 1 group can be designated.
- (b) When the same address is duplicated, designate the addresses in the specified order.
- (c) Addresses which do not need to be designated can be omitted.
- (d) The following table shows the correspondence between the addresses which can be designated by argument designation II and the variable numbers in the user macro main body.

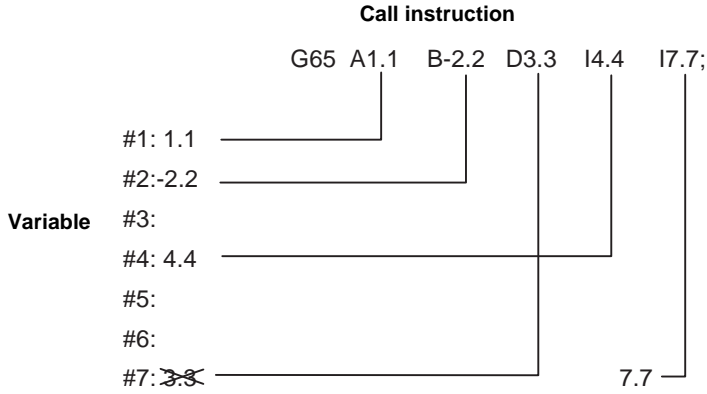
Argument specification II address	Variable in macro		Argument specification II address	Variable in macro
A	#1		J5	#17
B	#2		K5	#18
C	#3		I6	#19
I1	#4		J6	#20
J1	#5		K6	#21
K1	#6		I7	#22
I2	#7		J7	#23
J2	#8		K7	#24
K2	#9		I8	#25
I3	#10		J8	#26
J3	#11		K8	#27
K3	#12		I9	#28
I4	#13		J9	#29
J4	#14		K9	#30
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.

- (2) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.

However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.

(Example)

When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "G65 P12" command.

- (3) In the following cases, a subprogram of P command value without O No. is called even with a setting to call a subprogram with O No.

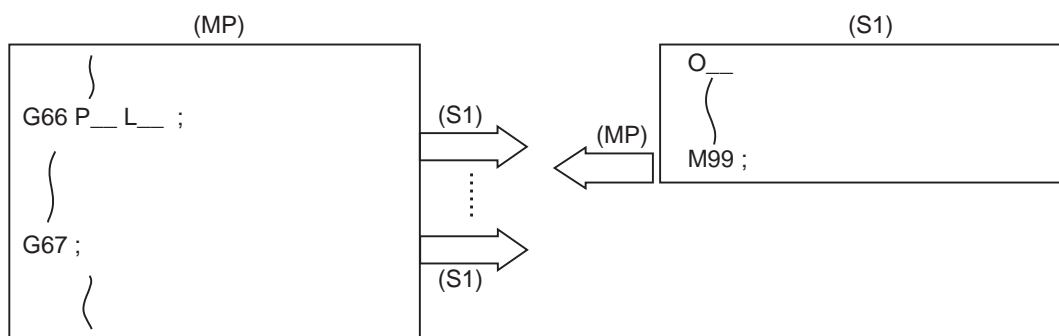
- The digit number of P command value is over the digit number of the program number set with parameter "#8129 subprogram number selection".

- A subprogram starting with commanded O No. does not exist.

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

Modal call A

G66 P__ L__ argument ;

G66 <File name> L__ argument ;

P	Program No. Use a parameter to specify a 4- or 8-digit subprogram No. starting with O.
<File name>	File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets < >. (The file name can have up to 32 characters including the extension.)
L	Number of repetitions:
Argument	Specify variable data

Modal call end

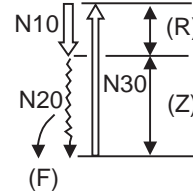
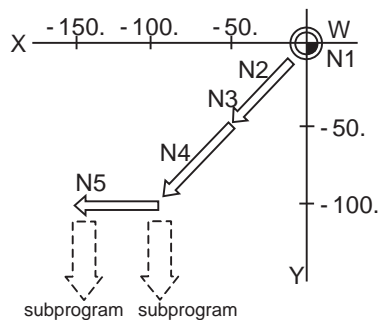
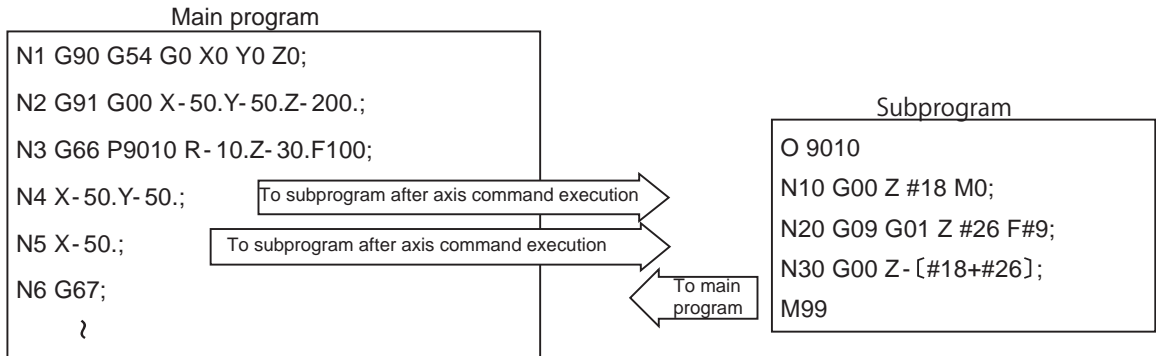
G67 ;



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



(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.

- (3) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.

(Example)

When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "G66 P12" command.

- (4) In the following cases, a subprogram of P command value without O No. is called even with a setting to call a subprogram with O No.
 - The digit number of P command value is over the digit number of the program number set with parameter "#8129 subprogram number selection".
 - A subprogram starting with commanded O No. does not exist.

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

Modal call B

G66.1 P__ L__ argument ;

G66.1 <File name> L__ argument ;

P	Program No. Use a parameter to specify a 4- or 8-digit subprogram No. starting with O.
<File name>	File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets < >. (The file name can have up to 32 characters including the extension.)
L	Number of repetitions
Argument	Specify variable data

Modal call end

G67 ;



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.
- (5) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.
(Example)
When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "G66.1 P12" command.
- (6) In the following cases, a subprogram of P command value without O No. is called even with a setting to call a subprogram with O No.
 - The digit number of P command value is over the digit number of the program number set with parameter "#8129 subprogram number selection".
 - A subprogram starting with commanded O No. does not exist.

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 $\Delta\Delta\Delta\Delta$;
 - b : G65 P $\Delta\Delta\Delta\Delta$ Argument ;
 - c : G66 P $\Delta\Delta\Delta\Delta$ Argument ;
 - d : G66.1 P $\Delta\Delta\Delta\Delta$ 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 $\Delta\Delta\Delta\Delta$ to be called is set by parameters.
- (3) Up to 10 G codes from G100 to G999 can be used with this instruction. (G codes from G01 to G99 used in the system can also be used with 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 in a program which has been called by a G code macro. If issued in such a program, they will be handled as ordinary G commands.

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**** ;	M98, M** are not output.
b:G65 P**** M** ;	
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 M9999 can be registered. Note that the codes to be registered should exclude those basically required for the machine and M0, M1, M2, M30, M96 to M99 and M198.
- (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.

(Note) When "1 to 3" is set to "#7002 M[01] type", the macro call will be equivalent call to G65/G66/G66.1. In this case, the alphabet before the M,S,T,B code macro is not handled as an argument. For example, commanding the M code and T code in the same block changes the operation depending on the order of the address.

(Preparation)
Register M06 to M code macro

(Command)
M06 T02 ... The value of T is handled as the variable #20 in the macro. The value is entered in the T code at the same time.
T02 M06 ... The value is not entered in the variable#20 in macro. The value is entered in the T code.

13.5.2.6 Detailed Description for Macro Call Instruction



Detailed description

Differences between M98 and G65 commands

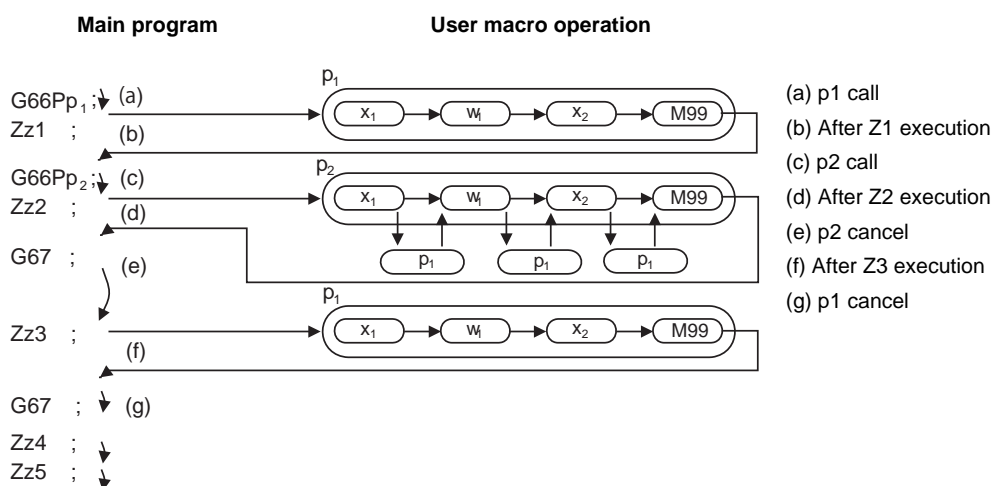
- (1) The argument can be designated for G65 but not for M98.
- (2) The sequence number can be designated for M98 but not for G65, G66 and G66.1.
- (3) M98 executes subprograms after all the commands except M, P, H and L in the M98 block are executed, but G65 branches directly to the subprogram without 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.

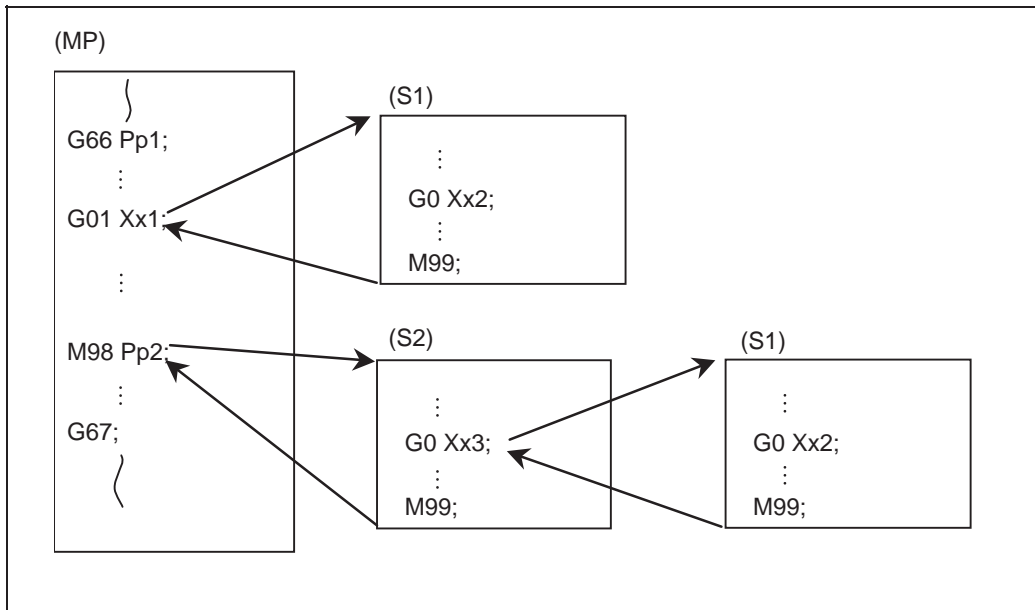
- (Note 1) When a G65, G66, G66.1 G code macro call or miscellaneous command macro call is conducted, this is regarded as a nesting level and the level of the local variables is also incremented by one.
- (Note 2) With modal call A, the designated user macro subprogram is called every time a movement 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)



(Note 3) When M98 command is executed in G66(G66.1) modal, the program designated by G66(G66.1) will be executed after completing the movement command in the subprogram called by M98 (in case of G66.1, after completing each block).

(Example 2)



(MP) Main program

(S1),(S2) Subprogram

When the program numbers of p1 and p2 are same, the program numbers of (S1) and (S2) will be same.

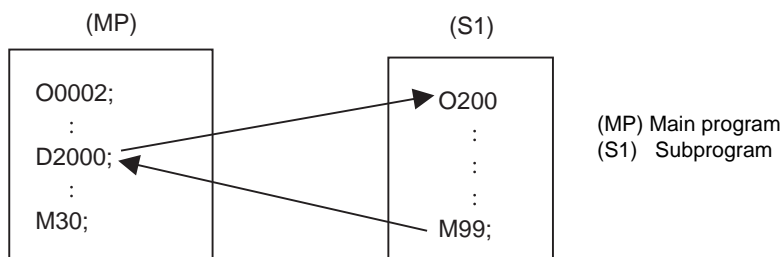
13.5.3 ASCII Code Macro



Function and purpose

A macro program can be called out by setting the correspondence of a subprogram (macro program) preregistered with the parameters to codes, and then commanding the ASCII code in the machining program. This function can be used in addition to the G, M, S, T and B miscellaneous command macro call function.

(Execution example 1) M98 type

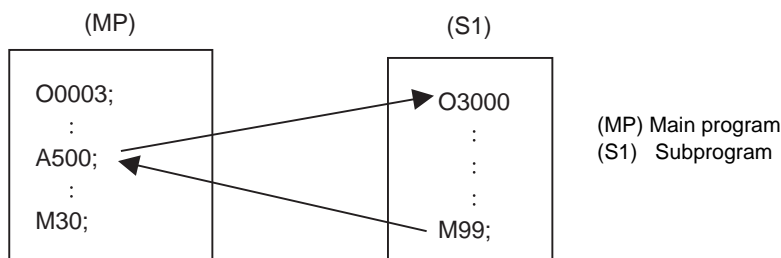


After outputting 2000 to common variable #146, the program No. 200 subprogram is called with the M98 subprogram call type.

Parameter

#7401 (ASCII [01] Valid/Invalid)	1 (Valid)
#7402 (ASCII [01] Code)	D
#7403 (ASCII [01] Type)	0 (M98 type)
#7404 (ASCII [01] Program No.)	200
#7405 (ASCII [01] Variable)	146

(Execution example 2) G65 type



After outputting 500 to local variable #1, the program No. 3000 subprogram is called out with the G65 macro call type.

Parameter

#7411 (ASCII [02] Valid/Invalid)	1 (Valid)
#7412 (ASCII [02] Code)	A
#7413 (ASCII [02] Type)	1 (G65 type)
#7414 (ASCII [02] Program No.)	3000
#7415 (ASCII [02] Variable)	100 (Not used)



Command format

<input type="checkbox"/> **** ; ... Designates the address and code	
<input type="checkbox"/>	ASCII code for calling out a macro (one character)
****	Value or expression output to variable (Setting range: ± 999999.9999)



Detailed description

- (1) The command above performs same operations as commands listed below. The correspondence of commands is set for each ASCII code with the parameters.
 - 0 : M98 P****;
 - 1 : G65 P**** <Argument>;
 - 2 : G66 P**** <Argument>;
 - 3 : G66.1 P**** <Argument>;

When parameters corresponding to 2 and 3 above are set, the modal call will be canceled. Thus, command the cancel command (G67) after commanding the call code or during the user macro.

- (2) The ASCII code for calling the macro and the program No. P**** to be called are set with the parameters. Up to two ASCII codes can be registered.
- (3) The code section is output to the variables, but the output destination differs according to the call type and address.

(a) For M98 type

The code section is output to a common variable and the variable No. is set with a parameter. When corresponding to the first address (parameter #7401), the section is output to the common variable which is indicated by the first variable No. (parameter #7404).

(b) For G65/G66/G66.1 type

The code section is output to a local variable. The variable No. differs according to the address, and corresponds to the following table.

Address	#	Address	#	Address	#
A	1	K	6	U	21
B	2	L	12	V	22
C	3	M	13	W	23
D	7	N	14	X	24
E	8	O	15	Y	25
F	9	P	16	Z	26
G	10	Q	17		
H	11	R	18		
I	4	S	19		
J	5	T	20		

(Note) The following addresses can be used.
A, B, D, F, H, I, J, K, M, Q, R, S, T



Precautions

Calling a macro with an ASCII code from a macro-called program

A macro cannot be called with an ASCII code from a macro-called program with an ASCII code.

The other patterns are shown below.

If it is judged that a macro cannot be called, the command will be handled as a normal command.

		Called side			
		ASCII	GMSTB macro	G65/66/66.1	M98
Calling side	ASCII	×	×	○	○
	GMSTB macro	×	×	○	○
	G65/66/66.1	○	○	○	○
	M98	○	○	○	○

Nest level of macro call command

Up to 4 nesting levels are available for macro subprogram calls using simple call (G65) and modal call (G66/G66.1).

The macro call command's argument is valid only in the called macro level.

Since the macro call nest level is four, the argument can be used in the program as a local variable for each macro call.

Nest level of macro call command

Counting the main program as 0, up to eight levels of subprograms can be called (M98) from a subprogram. The following commands are used for subprogram nesting.

- (1) M98
- (2) G65 G66 G66.1
- (3) G code call Miscellaneous function call (M/S/T/B)
- (4) MDI interruption
- (5) Automatic Tool Length Measurement
- (6) Multiple-step skip function

The following commands can be issued regardless of nesting.

- (7) Fixed cycle
- (8) Macro interruption

Order of command priority

If "M" is designated for the ASCII code address, it may overlap with the codes basically necessary for that machine. In this case, commands will be identified with the following priority using code values.

- (1) M98,M99 (subprogram call command)
M00 (program stop command), M01 (optional stop command)
M02,M30,M198,M199 (end command)
M96,M97 (macro interruption command)
- (2) When corresponding to miscellaneous code (M) call macro command
- (3) When corresponding to ASCII code macro command
- (4) Used as normal command

If "S", "T" and "B" are designated for the ASCII code address, commands will be identified with the following priority using code values.

- (a) When corresponding to miscellaneous code (S, T, B) call macro command
- (b) When corresponding to ASCII code macro command
- (c) Used as normal command

If the other addresses do not correspond to the ASCII code macro command, they will be identified as normal commands. If the command to be used, overlaps with an ASCII code macro command, it must be commanded in the macro-called program with the ASCII code.

Note that there are cases where the command will be unconditionally handled as a normal command, as explained in below.

Conditions where the address set is handled as a normal command

- (1) When there is a data setting command (G10) in the same block.
- (2) When ASCII code macro call is executed after the G code macro call command in the same block (also applies for M, S, T, B and ASCII)
(Example) When address "D" (G65 type) is set in the ASCII code macro, and M50 is set in the macro call (G65 type).

M50 D200 ; Execute M code macro with argument (200 set in #7)

- (3) When inputting parameters
- (4) When there is a comma (,) before the address. Example) ,D ,R, etc.
- (5) When commanded in fixed cycle
- (6) When commanded in macro subprogram called with G code macro call
(Also applies when macro is called with M, S, T, B or ASCII)

13.5.4 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; ##10=100 ;	<Formula>##10 = 100; is handled in the same manner as # [#10] = 100. In which case, #5 = 100.
----------------------	--

(Example 3) Replacing variable Nos. with <formula>

#10=5 ; #[#10 + 1] = 1000 ; #[#10 - 1] = -1000 ; #[#10 * 3] = 100 ; #[#10/2] = -100 ;	In which case, #6 = 1000. In which case, #4 = -1000. In which case, #15 = 100. In which case, #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.

13.5.5 Types of Variables

13.5.5.1 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 alphanumeric characters and it must begin with a letter. Do not use "#" in variable names. It causes an alarm when the program is executed.

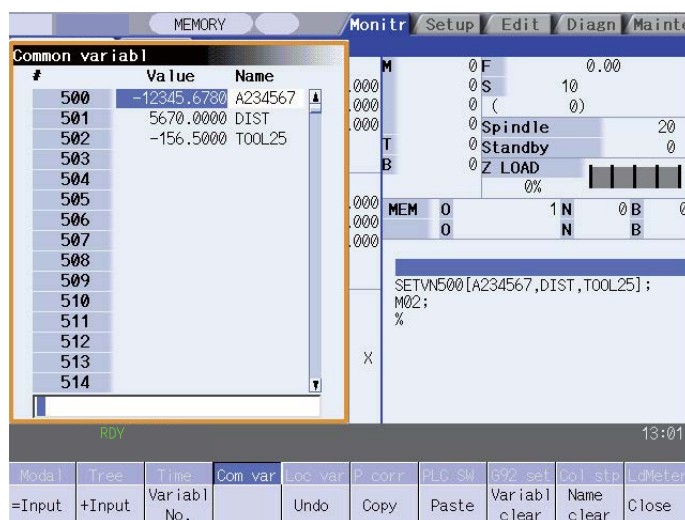
```
SETVn [ NAME1,NAME2, ..... ] ;
```

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] ;
- (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] ;



- (Note) Do not use characters (SIN, COS, etc.) predetermined by the NC and used for operation commands at the head of a variable name.

13.5.5.2 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
x	x *	G	#10	○	○	W	#23
○	○	H	#11	○	○	X	#24
○	○	I	#4	○	○	Y	#25
○	○	J	#5	○	○	Z	#26
○	○	K	#6			-	#27
x	x *	L	#12			-	#28
○	○	M	#13			-	#29
x	x *	N	#14			-	#30
x	x	O	#15			-	#31
x	x *	P	#16			-	#32
						-	#33

"x" 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.

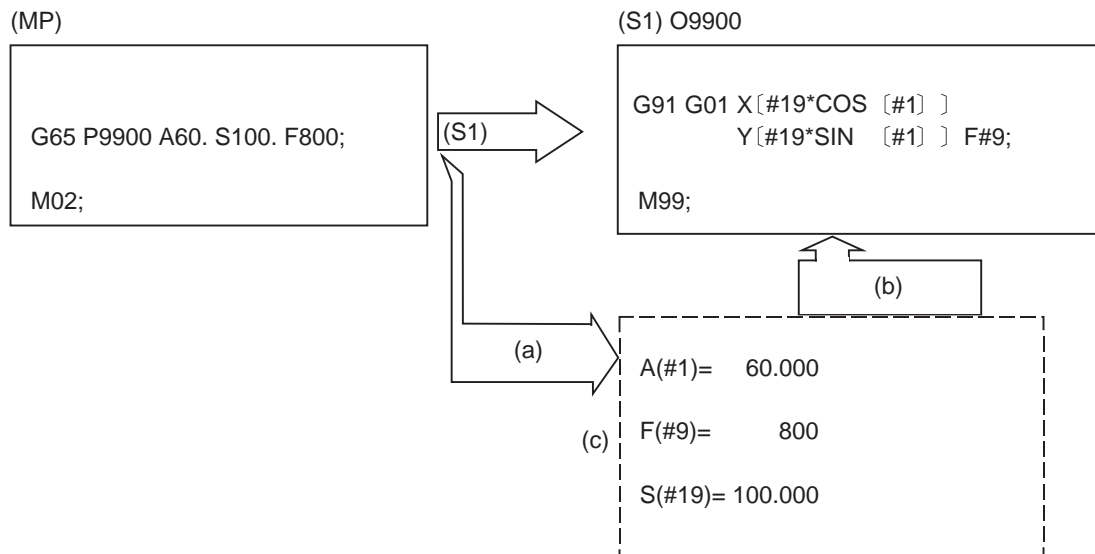
The hyphen (-) mark indicates that there is no corresponding address.

[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

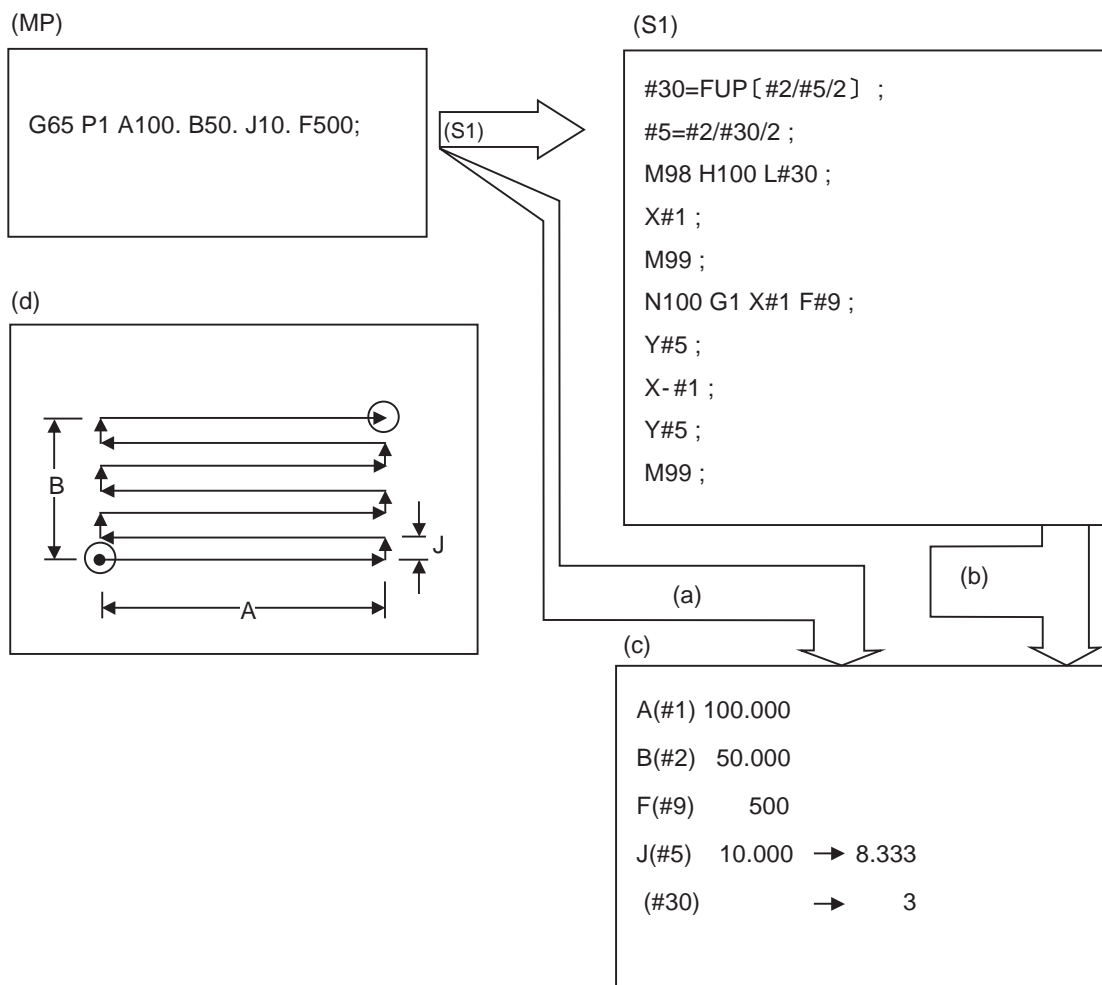
(S1) Subprogram

(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

(S1) Subprogram

(a) Local variables set by argument

(b) The local variables can be changed in the subprogram.

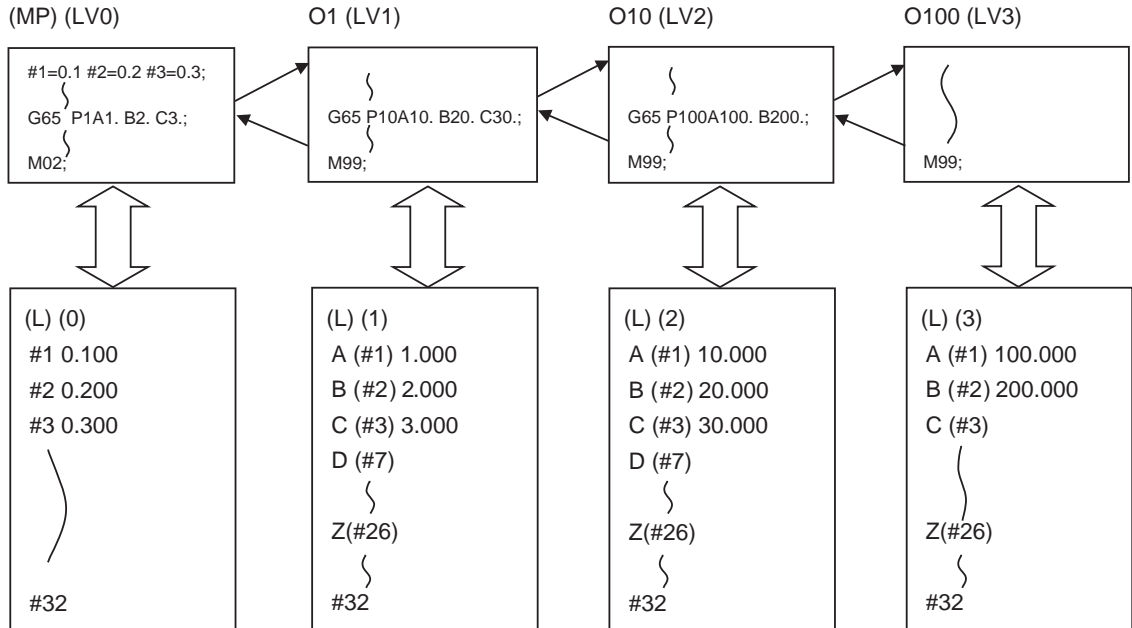
(c) Local variable data table

(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
 (L) Local variables
 (LV0 - 3) Macro level 0 - 3

The status of the local variables is displayed on the setting and display unit.
 Refer to the Instruction Manual for details.

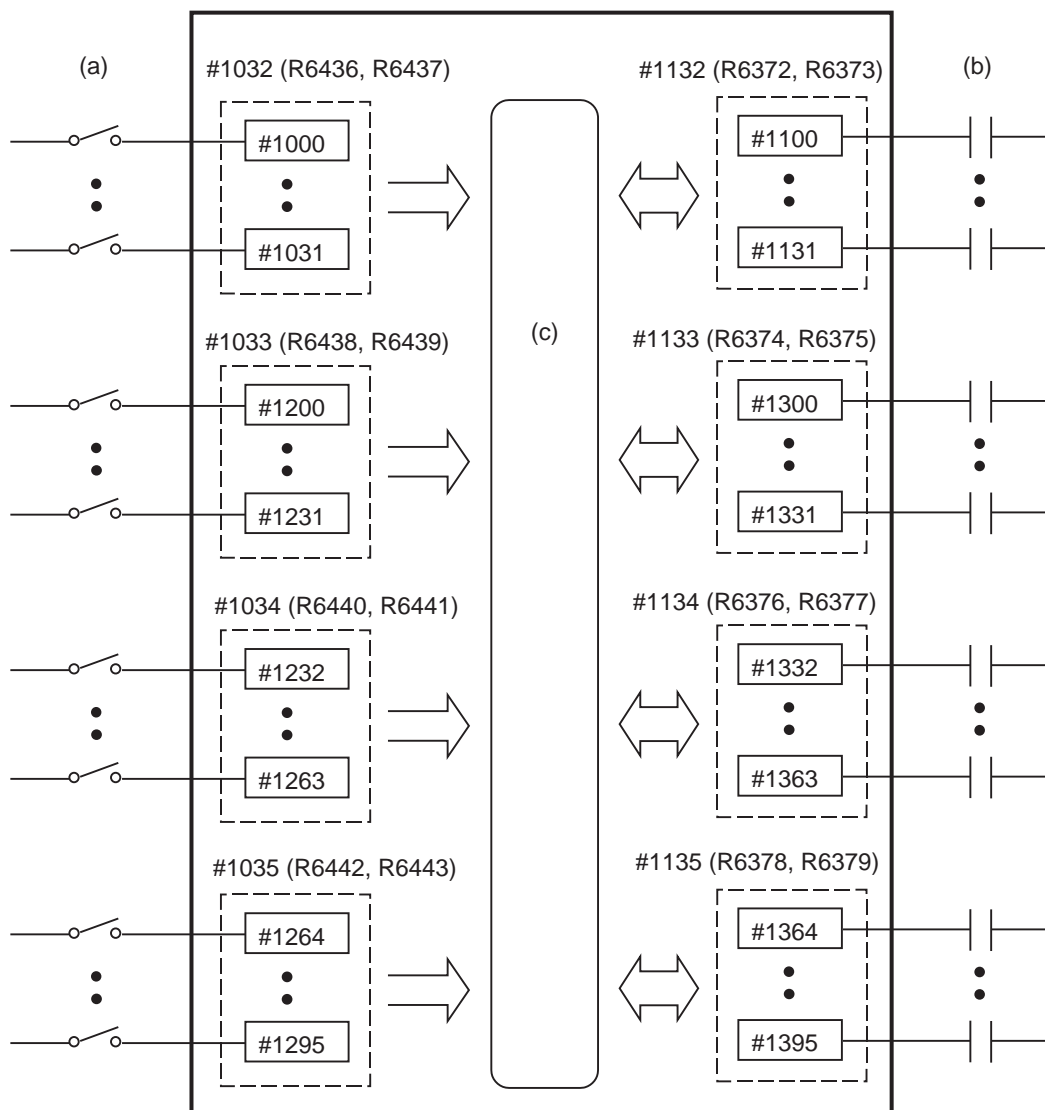
13.5.5.3 Macro Interface Inputs/Outputs (#1000 to #1035, #1100 to #1135, #1200 to #1295, #1300 to #1395)



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.

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1000	1	Register R6436 bit 0	#1016	1	Register R6437 bit 0
#1001	1	Register R6436 bit 1	#1017	1	Register R6437 bit 1
#1002	1	Register R6436 bit 2	#1018	1	Register R6437 bit 2
#1003	1	Register R6436 bit 3	#1019	1	Register R6437 bit 3
#1004	1	Register R6436 bit 4	#1020	1	Register R6437 bit 4
#1005	1	Register R6436 bit 5	#1021	1	Register R6437 bit 5
#1006	1	Register R6436 bit 6	#1022	1	Register R6437 bit 6
#1007	1	Register R6436 bit 7	#1023	1	Register R6437 bit 7
#1008	1	Register R6436 bit 8	#1024	1	Register R6437 bit 8
#1009	1	Register R6436 bit 9	#1025	1	Register R6437 bit 9
#1010	1	Register R6436 bit 10	#1026	1	Register R6437 bit 10
#1011	1	Register R6436 bit 11	#1027	1	Register R6437 bit 11
#1012	1	Register R6436 bit 12	#1028	1	Register R6437 bit 12
#1013	1	Register R6436 bit 13	#1029	1	Register R6437 bit 13
#1014	1	Register R6436 bit 14	#1030	1	Register R6437 bit 14
#1015	1	Register R6436 bit 15	#1031	1	Register R6437 bit 15

System variable	No. of points	Interface input signal
#1032	32	Register R6436, R6437
#1033	32	Register R6438, R6439
#1034	32	Register R6440, R6441
#1035	32	Register R6442, R6443

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1200	1	Register R6438 bit 0	#1216	1	Register R6439 bit 0
#1201	1	Register R6438 bit 1	#1217	1	Register R6439 bit 1
#1202	1	Register R6438 bit 2	#1218	1	Register R6439 bit 2
#1203	1	Register R6438 bit 3	#1219	1	Register R6439 bit 3
#1204	1	Register R6438 bit 4	#1220	1	Register R6439 bit 4
#1205	1	Register R6438 bit 5	#1221	1	Register R6439 bit 5
#1206	1	Register R6438 bit 6	#1222	1	Register R6439 bit 6
#1207	1	Register R6438 bit 7	#1223	1	Register R6439 bit 7
#1208	1	Register R6438 bit 8	#1224	1	Register R6439 bit 8
#1209	1	Register R6438 bit 9	#1225	1	Register R6439 bit 9
#1210	1	Register R6438 bit 10	#1226	1	Register R6439 bit 10
#1211	1	Register R6438 bit 11	#1227	1	Register R6439 bit 11
#1212	1	Register R6438 bit 12	#1228	1	Register R6439 bit 12
#1213	1	Register R6438 bit 13	#1229	1	Register R6439 bit 13
#1214	1	Register R6438 bit 14	#1230	1	Register R6439 bit 14
#1215	1	Register R6438 bit 15	#1231	1	Register R6439 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1232	1	Register R6440 bit 0	#1248	1	Register R6441 bit 0
#1233	1	Register R6440 bit 1	#1249	1	Register R6441 bit 1
#1234	1	Register R6440 bit 2	#1250	1	Register R6441 bit 2
#1235	1	Register R6440 bit 3	#1251	1	Register R6441 bit 3
#1236	1	Register R6440 bit 4	#1252	1	Register R6441 bit 4
#1237	1	Register R6440 bit 5	#1253	1	Register R6441 bit 5
#1238	1	Register R6440 bit 6	#1254	1	Register R6441 bit 6
#1239	1	Register R6440 bit 7	#1255	1	Register R6441 bit 7
#1240	1	Register R6440 bit 8	#1256	1	Register R6441 bit 8
#1241	1	Register R6440 bit 9	#1257	1	Register R6441 bit 9
#1242	1	Register R6440 bit 10	#1258	1	Register R6441 bit 10
#1243	1	Register R6440 bit 11	#1259	1	Register R6441 bit 11
#1244	1	Register R6440 bit 12	#1260	1	Register R6441 bit 12
#1245	1	Register R6440 bit 13	#1261	1	Register R6441 bit 13
#1246	1	Register R6440 bit 14	#1262	1	Register R6441 bit 14
#1247	1	Register R6440 bit 15	#1263	1	Register R6441 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1264	1	Register R6442 bit 0	#1280	1	Register R6443 bit 0
#1265		Register R6442 bit 1	#1281		Register R6443 bit 1
#1266		Register R6442 bit 2	#1282		Register R6443 bit 2
#1267		Register R6442 bit 3	#1283		Register R6443 bit 3
#1268		Register R6442 bit 4	#1284		Register R6443 bit 4
#1269		Register R6442 bit 5	#1285		Register R6443 bit 5
#1270		Register R6442 bit 6	#1286		Register R6443 bit 6
#1271		Register R6442 bit 7	#1287		Register R6443 bit 7
#1272		Register R6442 bit 8	#1288		Register R6442 bit 8
#1273		Register R6442 bit 9	#1289		Register R6443 bit 9
#1274		Register R6442 bit 10	#1290		Register R6443 bit 10
#1275		Register R6442 bit 11	#1291		Register R6443 bit 11
#1276		Register R6442 bit 12	#1292		Register R6443 bit 12
#1277		Register R6442 bit 13	#1293		Register R6443 bit 13
#1278		Register R6442 bit 14	#1294		Register R6443 bit 14
#1279		Register R6442 bit 15	#1295		Register R6443 bit 15

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.

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1100	1	Register R6372 bit 0	#1116	1	Register R6373 bit 0
#1101	1	Register R6372 bit 1	#1117	1	Register R6373 bit 1
#1102	1	Register R6372 bit 2	#1118	1	Register R6373 bit 2
#1103	1	Register R6372 bit 3	#1119	1	Register R6373 bit 3
#1104	1	Register R6372 bit 4	#1120	1	Register R6373 bit 4
#1105	1	Register R6372 bit 5	#1121	1	Register R6373 bit 5
#1106	1	Register R6372 bit 6	#1122	1	Register R6373 bit 6
#1107	1	Register R6372 bit 7	#1123	1	Register R6373 bit 7
#1108	1	Register R6372 bit 8	#1124	1	Register R6373 bit 8
#1109	1	Register R6372 bit 9	#1125	1	Register R6373 bit 9
#1110	1	Register R6372 bit 10	#1126	1	Register R6373 bit 10
#1111	1	Register R6372 bit 11	#1127	1	Register R6373 bit 11
#1112	1	Register R6372 bit 12	#1128	1	Register R6373 bit 12
#1113	1	Register R6372 bit 13	#1129	1	Register R6373 bit 13
#1114	1	Register R6372 bit 14	#1130	1	Register R6373 bit 14
#1115	1	Register R6372 bit 15	#1131	1	Register R6373 bit 15

System variable	No. of points	Interface output signal
#1132	32	Register R6372, R6373
#1133	32	Register R6374, R6375
#1134	32	Register R6376, R6377
#1135	32	Register R6378, R6379

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1300	1	Register R6374 bit 0	#1316	1	Register R6375 bit 0
#1301	1	Register R6374 bit 1	#1317	1	Register R6375 bit 1
#1302	1	Register R6374 bit 2	#1318	1	Register R6375 bit 2
#1303	1	Register R6374 bit 3	#1319	1	Register R6375 bit 3
#1304	1	Register R6374 bit 4	#1320	1	Register R6375 bit 4
#1305	1	Register R6374 bit 5	#1321	1	Register R6375 bit 5
#1306	1	Register R6374 bit 6	#1322	1	Register R6375 bit 6
#1307	1	Register R6374 bit 7	#1323	1	Register R6375 bit 7
#1308	1	Register R6374 bit 8	#1324	1	Register R6375 bit 8
#1309	1	Register R6374 bit 9	#1325	1	Register R6375 bit 9
#1310	1	Register R6374 bit 10	#1326	1	Register R6375 bit 10
#1311	1	Register R6374 bit 11	#1327	1	Register R6375 bit 11
#1312	1	Register R6374 bit 12	#1328	1	Register R6375 bit 12
#1313	1	Register R6374 bit 13	#1329	1	Register R6375 bit 13
#1314	1	Register R6374 bit 14	#1330	1	Register R6375 bit 14
#1315	1	Register R6374 bit 15	#1331	1	Register R6375 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1332	1	Register R6376 bit 0	#1348	1	Register R6377 bit 0
#1333	1	Register R6376 bit 1	#1349	1	Register R6377 bit 1
#1334	1	Register R6376 bit 2	#1350	1	Register R6377 bit 2
#1335	1	Register R6376 bit 3	#1351	1	Register R6377 bit 3
#1336	1	Register R6376 bit 4	#1352	1	Register R6377 bit 4
#1337	1	Register R6376 bit 5	#1353	1	Register R6377 bit 5
#1338	1	Register R6376 bit 6	#1354	1	Register R6377 bit 6
#1339	1	Register R6376 bit 7	#1355	1	Register R6377 bit 7
#1340	1	Register R6376 bit 8	#1356	1	Register R6377 bit 8
#1341	1	Register R6376 bit 9	#1357	1	Register R6377 bit 9
#1342	1	Register R6376 bit 10	#1358	1	Register R6377 bit 10
#1343	1	Register R6376 bit 11	#1359	1	Register R6377 bit 11
#1344	1	Register R6376 bit 12	#1360	1	Register R6377 bit 12
#1345	1	Register R6376 bit 13	#1361	1	Register R6377 bit 13
#1346	1	Register R6376 bit 14	#1362	1	Register R6377 bit 14
#1347	1	Register R6376 bit 15	#1363	1	Register R6377 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1364	1	Register R6378 bit 0	#1380	1	Register R6379 bit 0
#1365	1	Register R6378 bit 1	#1381	1	Register R6379 bit 1
#1366	1	Register R6378 bit 2	#1382	1	Register R6379 bit 2
#1367	1	Register R6378 bit 3	#1383	1	Register R6379 bit 3
#1368	1	Register R6378 bit 4	#1384	1	Register R6379 bit 4
#1369	1	Register R6378 bit 5	#1385	1	Register R6379 bit 5
#1370	1	Register R6378 bit 6	#1386	1	Register R6379 bit 6
#1371	1	Register R6378 bit 7	#1387	1	Register R6379 bit 7
#1372	1	Register R6378 bit 8	#1388	1	Register R6379 bit 8
#1373	1	Register R6378 bit 9	#1389	1	Register R6379 bit 9
#1374	1	Register R6378 bit 10	#1390	1	Register R6379 bit 10
#1375	1	Register R6378 bit 11	#1391	1	Register R6379 bit 11
#1376	1	Register R6378 bit 12	#1392	1	Register R6379 bit 12
#1377	1	Register R6378 bit 13	#1393	1	Register R6379 bit 13
#1378	1	Register R6378 bit 14	#1394	1	Register R6379 bit 14
#1379	1	Register R6378 bit 15	#1395	1	Register R6379 bit 15

(Note 1) The last values of the system variables #1100 to #1135, #1300 to #1395 sent are retained as 1 or 0. (They are not cleared even with resetting.)

(Note 2) The following applies when any number except 1 or 0 is substituted into #1100 to #1131, #1300 to #1395.

<Blank> is treated as 0.

Any number except 0 and <Blank> is treated as 1.

Any value less than 0.00000001 is indefinite.

13.5.5.4 Tool Compensation



Detailed description

Tool compensation data can be read and set using the variable numbers.

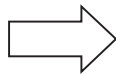
Variable number range		Type 1	Type 2
#10001 to #10000+n	#2001 to #2000+n	○	○ (Length dimension)
#11001 to #11000+n	#2201 to #2200+n	×	○ (Length wear)
#16001 to #16000+n	#2401 to #2400+n	×	○ (Radius dimension)
#17001 to #17000+n	#2601 to #2600+n	×	○ (Radius wear)

"n" in the table corresponds to the tool No. Maximum "n" value is the number of tool compensation sets. The #10000s and #2000s are equivalent functions, however, the maximum value of "n" for #2000 order is "200".

The tool compensation data is configured as data with a decimal point in the same way as other variables. When "# 10001=100;" is programmed, 100.000 is set in tool compensation data.

Programming example

```
#101=100;
#10001=#101;
#102=#10001;
```



Common variables

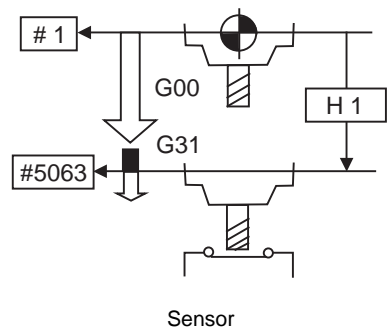
```
#101=100.0
#102=100.0
```

Tool compensation data

```
H1=100.000
```

(Example 1) Calculation and tool offset data setting

- G28 Z0 T01 ; Reference position 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 compensation 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.

13.5.5.5 Workpiece Coordinate System Compensation (#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	Remarks
External workpiece offset	#5201	#5202	#5203	
G54	#5221	#5222	#5223	
G55	#5241	#5242	#5243	
G56	#5261	#5262	#5263	
G57	#5281	#5282	#5283	
G58	#5301	#5302	#5303	
G59	#5321	#5322	#5323	

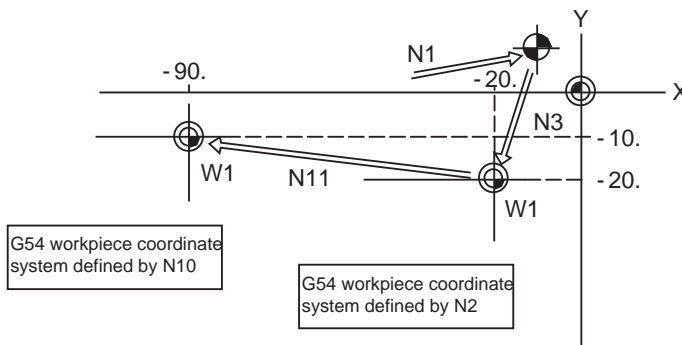
(Example 1)

```

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 ;
    
```



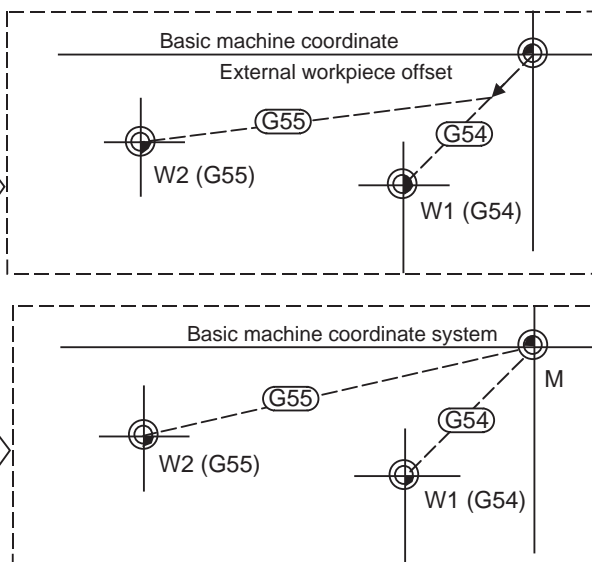
(Example 2)

```

N100 #5221=#5221+#5201 ;
      #5222=#5222+#5202 ;
      #5241=#5241+#5201 ;
      #5242=#5242+#5202 ;
      #5201=0 #5202=0;
    
```

Coordinate system before change

Coordinate system after change



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.

13.5.5.6 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 message " (CALL #PROGRAMMER #TEL #530)" and alarm No. (70) will appear in the <Operator message>.

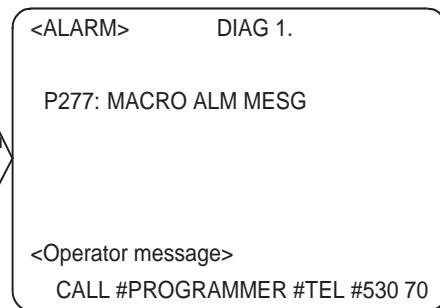
Example of program (alarm when #1 = 0)

```

}
IF[#1 NE 0]GOTO 100 ;
#3000=70 ( CALL #PROGRAMMER #TEL #530 );

N100
}
    
```

Stops with
NC alarm



- (Note 1) If the numerical value "0" or larger than "9999" is commanded as an alarm No., the alarm No. will be invalid and cannot be displayed. However, it will be set as the alarm state and the commanded alarm message will be displayed.
- (Note 2) An alarm message can be commanded by enclosing the message that exists right after the alarm No. with round parentheses. If other character strings are commanded between the alarm No. and the alarm message enclosed by the parentheses, the alarm message will be invalid and not displayed. However, it turns to the alarm state and the commanded alarm No. is displayed.
- (Note 3) If a character string with 32 or more is commanded, the character string of 32th or after is not displayed.
- (Note 4) The space in character strings of the alarm message is ignored and not displayed on the screen. To separate character strings, insert a character such as "." (period) instead.

13.5.5.7 Integrating Time (#3001, #3002)

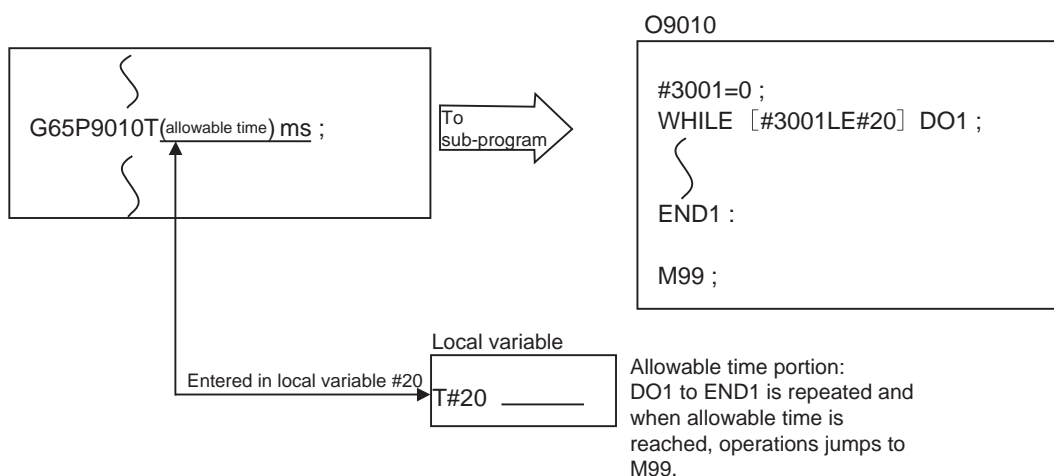


Detailed description

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

The integrating time returns to zero in about 2.44×10^{11} ms (approximately 7.7 years).



13.5.5.8 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.

13.5.5.9 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.

(Note 3) When the feed hold is set to invalid with #3004, the following will occur when the feed hold switch is pressed.

- During thread cutting, block stop will be carried out at the end of the next block of the block where thread cutting is completed.
- During tapping with tap cycle, block stop will be carried out after the operation of R point return.
- In the case other than above, block stop will be carried out after the termination of the currently executing block.

13.5.5.10 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 and the stop No. will be indicated on the operator message area.

#3006 = 1(TAKE FIVE);	
1 to 9999	Stop No. (When a value other than 1 - 9999 is set, the command will be invalidated.)
TAKE FIVE	Message (Nothing will be displayed if no message is designated.)

The message should be written in 31 or less characters and should be enclosed by round parentheses.

13.5.5.11 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

13.5.5.12 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, G02 : 2, G03 : 3, G33 : 33
#4002	#4202	Plane selection	G17 : 17, G18 : 18, G19 : 19
#4003	#4203	Absolute/incremental	G90 : 90, G91 : 91
#4004	#4204	No variable No.	
#4005	#4205	Feed designation	G94 : 94, G95 : 95
#4006	#4206	Inch/metric	G20 : 20, G21 : 21
#4007	#4207	Tool radius compensation	G40 : 40, G41 : 41, G42 : 42
#4008	#4208	Tool length compensation	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, G54.1:54.1
#4013	#4213	Acceleration/deceleration	G61-G64 : 61-64
#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 ;
%
```

13.5.5.13 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 compensation No. H
#4102	#4302		#4112	#4312	
#4103	#4303		#4113	#4313	Miscellaneous function M
#4104	#4304		#4114	#4314	Sequence number N
#4105	#4305		#4115	#4315	Program number O (Note 1)
#4106	#4306		#4116	#4316	
#4107	#4307	Tool radius compensation No. D	#4117	#4317	
#4108	#4308		#4118	#4318	
#4109	#4309	Feedrate F	#4119	#4319	Spindle function S
#4110	#4310		#4120	#4320	Tool function T
			#4130	#4330	Extended workpiece coordinate system No. P

(Note 1) The programs are registered as files. When the program No. (file name) is read with #4115, #4315, the character string will be converted to a value.

(Example 1) The file name "123" is the character string 0x31, 0x32, 0x33, so the value will be $(0x31-0x30)*100 + (0x32-0x30)*10 + (0x33-0x30) = 123.0$. Note that if the file name contains characters other than numbers, it will be "blank".

(Example 2) If the file name is "123ABC", it contains characters other than numbers; so, the result will be "blank".

13.5.5.14 Position Information (#5001 - #5160 + n)

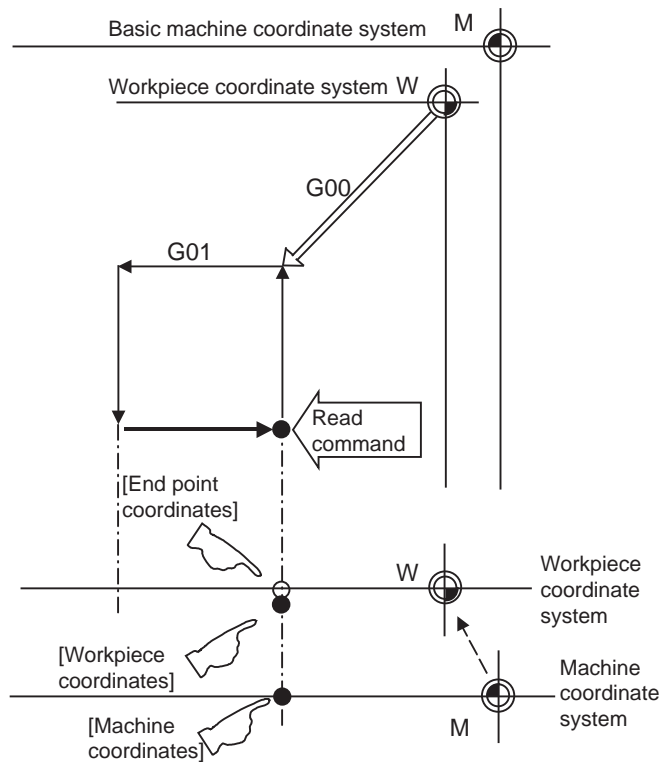


Detailed description

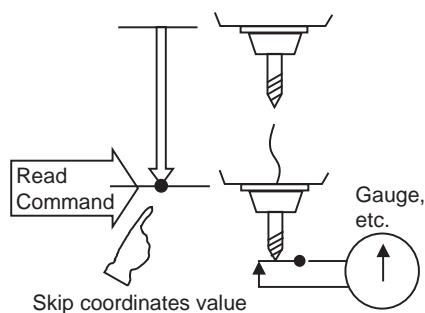
Using variable Nos. #5001 to #5160, it is possible to read the end point coordinates, machine coordinates, workpiece coordinates, skip coordinates, tool position compensation amount and servo deviation amounts in the last block.

Position information	Axis No.			Reading during movement
	1	2	3	
End point coordinate of the last block	#5001	#5002	#5003	Yes
Machine coordinate	#5021	#5022	#5023	No
Workpiece coordinate	#5041	#5042	#5043	No
Skip coordinate	#5061	#5062	#5063	Yes
Tool position compensation amount	#5081	#5082	#5083	No
Servo deviation amount	#5101	#5102	#5103	Yes
Macro interruption stop Start point coordinates	#5121	#5122	#5123	Yes
Macro interruption stop End point coordinates	#5141	#5142	#5143	Yes

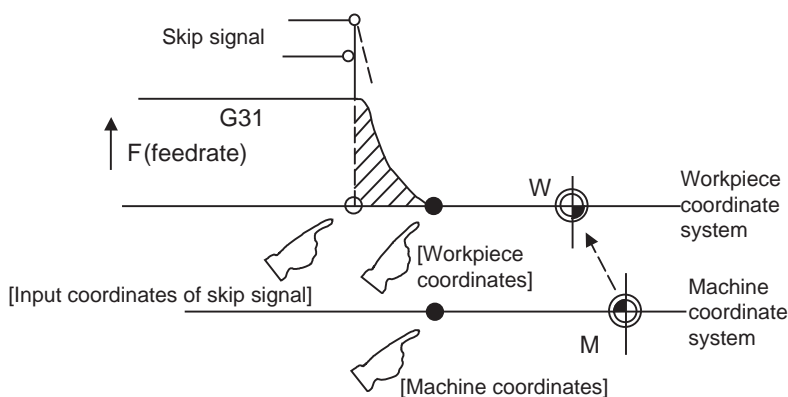
(Note) The number of axes which can be controlled differs according to the specifications. The last digit of the variable No. corresponds to the control axis No.



- (1) The position of the end point coordinates is position 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.

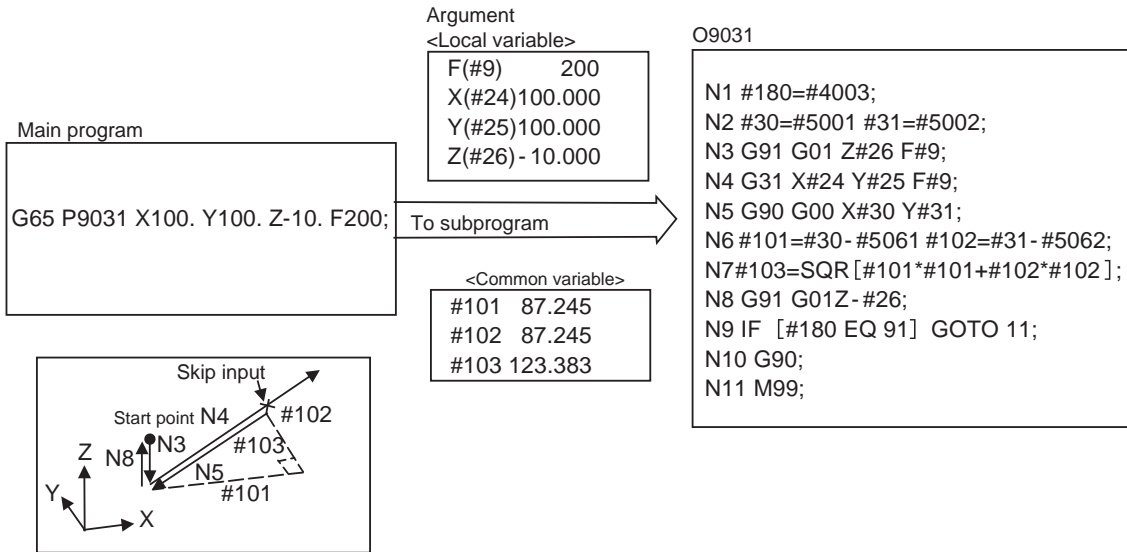
Skip coordinate value is the position on the workpiece coordinate system.

The coordinate value in variable Nos. #5061 to #5060+n 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".

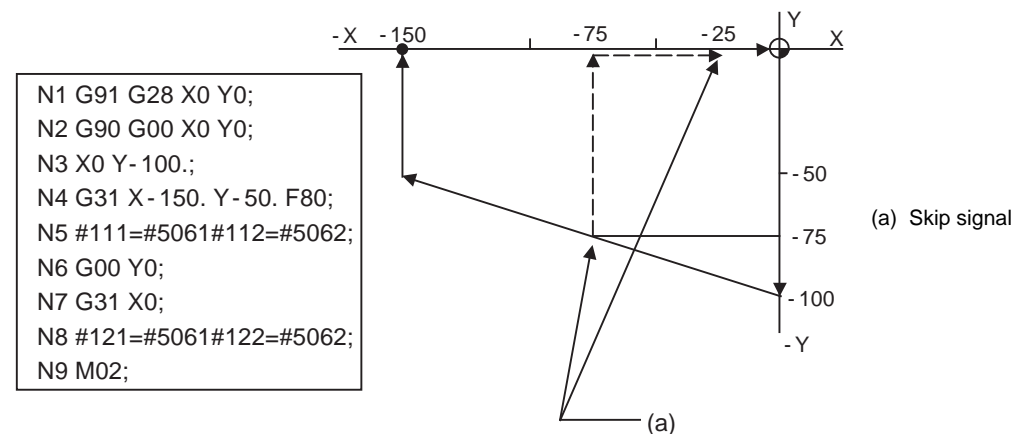
(Example 1) Example of workpiece position measurement

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) |
| | | N5 | Return to X, Y start point |
| #5001 | X axis measurement start point | N6 | X, Y measurement incremental value calculation |
| #5002 | Y axis measurement start point | N7 | Measurement linear segment calculation |
| | | N8 | Z axis escape |
| #5061 | X axis skip input point | N9,N10 | G90/G91 modal return |
| #5062 | Y axis skip input point | N11 | Main program return |

(Example 2) Reading of skip input coordinates



#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.

13.5.5.15 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.

13.5.5.16 Coordinate Rotation Parameter



Detailed description

The following variables can be read by the system variables of the variable command.

Note that writing is not possible onto these variables.

Variable No.	Description
#30060	Control axis No. on the coordinate rotation plane (horizontal axis)
#30061	Control axis No. on the coordinate rotation plane (vertical axis)
#30062	Coordinate rotation center (horizontal axis)
#30063	Coordinate rotation center (vertical axis)
#30064	Coordinate rotation angle
#30065	SIN data for the coordinate rotation angle [SIN(Coordinate rotation angle)]
#30066	COS data for the coordinate rotation angle [COS(Coordinate rotation angle)]
#30067	Coordinate rotation vector (horizontal axis)
#30068	Coordinate rotation vector (vertical axis)

13.5.5.17 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. When the tool life management III are provided, the group No. other than 1 cannot be used.

- (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 compensation 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 65000 times
60003	Tool selection No.		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		No. of first registered tool that has not reached its life.	0 to 200
60005	Execution signal		"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)		Indicates the time that this group is used in the program being executed.	(Not used)
60007	Life end signal		"1" when lives of all tools in this group have expired. "1" when all registered tools in the designated group reach lives.	0/1
60008	Life prediction signal		"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 minutes 0 to 9999 times
63000 +***	Usage data		Usage time or No. of uses for each tool	0 to 4000 minutes 0 to 65000 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 ± 999.999 (*1) Increment value compensation amount ± 999.999 (*1)
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 ± 999.999 (*1) Increment value compensation amount ± 999.999 (*1)
64500 +***	Auxiliary data		Spare data	0 to 65535

(*1) Refer to (16) in "12.5.3 Precautions for Inputting the Tool Life Management Data" for the data range of compensation amount.



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 life data other than group 10 is registered 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.
- (5) When the tool life management III are provided, the group No. other than 1 cannot be used.

13.5.5.18 Reading The Parameters (#100000-#100002, #100010)



Function and purpose

Parameter data can be read in with the system variables.

Variable No.	Application
#100000	Parameter No. designation
#100001	System reserve
#100002	Axis No./spindle No. designation
#100010	Parameter value read



Detailed description

The parameter values are read in with the following four blocks using these four system variables.

#100000 = 1001; Designates the parameter No.
 #100001 = 0; Always set "0"
 #100002 = 1; Designates the axis No./spindle No.
 #100 = #100010; Reads the parameter value.

Parameter No. designation (#100000)

The parameter to be read in is designated by substituting the parameter No. in this system variable.

If the parameters are read without designating this No., the parameters will be read in the same manner as if the minimum parameter No. (#1) is designated. Once designated, the setting is held until the parameter No. is designated again or until it is reset.

A program error (P39) will occur if a nonexistent parameter No. is set.

System reserve (#100001)

Make sure that it's always set to "0". For a PLC axis, set it to "10".

Axis No. /spindle No. designation (#100002)

(1) System variable for axis No. /spindle No. designation

The axis No./spindle No. of the parameter to be read in is designated by substituting an index value in this system variable. This designation will be ignored when reading in parameters that are neither for a specific axis nor spindle.

If the parameters are read without designating this number, the parameters will be read in the same manner as when the index value 1 (1st axis/1st spindle) is designated. Once designated, the setting is held until the index value is designated again or until it is reset.

A program error (P39) will occur if a nonexistent axis/spindle No. is set.

(2) Index values

Index values	Axis parameter	Spindle parameter
1	1st axis	1st spindle
2	2nd axis	-
3	3rd axis	-

Reading the parameters (#100010)

The designated parameter data is read with this system variable.

Data to be read as follows, depending on the parameter type.

Type	Read in data
Value	The values displayed on the Parameter screen are output.
Text	ASCII codes are converted into decimal values.



Program example

- (1) To read the parameter [#1002 axisno Number of axes]

```
#100000 = 1002; ..... Designates [#1002].
#100001 = 0;
#101 = #100010; ..... Reads the number of axes.
#100001 = 10; ..... Designates [PLC axis].
#110 = #100010; ..... Reads the number of PLC axes.
```

- (2) To read the axis parameter [#2037 G53ofs #1 reference point]

[Conditions]

#2037 G53ofs	<1st axis>	<2nd axis>
	1000.000	200.000

```
#100002 = 1; ..... Designates [1st axis].
#100000 = 2037; ..... Designates [#2037].
#101 = #100010; ..... Reads the [#1 reference point] for the 1st axis.
                      (#101=100.000)
#100002 = 2; ..... Designates [2nd axis].
#102 = #100010; ..... Reads the [#1 reference point] for the 2nd axis.
                      (#102=200.000)
```

- (3) To read each parameter for axis and spindle

```
#100002 = 1; ..... Designates [1st spindle].
#100000 = 3001; ..... Designates [#3001].
#101 = #100010; ..... Reads the [#3001 slimt1 Limit rotation speed gears 00] for 1st spindle.
#100000 = 3002; ..... Designates [#3002].
#102 = #100010; ..... Reads the [#3002 slimt2 Limit rotation speed gears 01] for 1st spindle.
```



Precautions

- (1) The number of axes and spindles is set at the maximum number specified by the model.
- (2) The inch/metric changeover function for the setting and display is valid for the readout data.

13.5.5.19 Reading PLC data (#100100-#100103,#100110)



Function and purpose

PLC data can be read in with the system variables.

Variable No.	Application
#100100	Device type designation
#100101	Device No. designation
#100102	Number of read bytes designation
#100103	Read bit designation
#100110	Reading PLC data

(Note 1) The readable devices are limited.



Detailed description

The PLC data is read in with the following five blocks using these five system variables.

- #100100 = 1; Designates the device type.
- #100101 = 0; Designates the device No.
- #100102 = 1; Designates the number of bytes.
- #100103 = 2; Designates the bit. (Valid only when reading word device bits.)
- #100=#100110; Reads in the PLC data.

Device designation (#100100)

(1) System variable for device designation

The type of device to be read in can be designated by substituting the device designation value in this system variable.

If the data is read without designating this number, the data will be read in the same manner as when the minimum device designation value (0: M device) is designated. Once designated, the setting is held until the device is designated again or until it is reset.

A program error (P39) will occur if a nonexistent device is set.

(2) Device designation value

Device designation value	Device	Unit	Device No.	Device designation value	Device	Unit	Device No.
0	M	Bit	M0 to M10239	10	F	Bit	F0 to F1023
1	D	Word	D0 to D2047	13	L	Bit	L0 to L511
2	C	Bit	C0 to C255	18	V	Bit	V0 to V255
4	X *	Bit	X0 to X1FFF	19	ST	Bit	ST0 to ST63
5	Y *	Bit	Y0 to Y1FFF	20	SD	Word	SD0 to SD127
6	R	Word	R0 to R32767	21	SB *	Bit	SB0 to SB1FF
7	T	Bit	T0 to T703	22	SW *	Word	SW0 to SW1FF
9	SM	Bit	SM0 to SM127	23	B *	Bit	B0 to B1FFF
				24	W *	Word	W0 to W1FFF

The unit indicates the amount of data per device No. "Word" is 16 bits, and "Bit" is one bit.

* is a device that expresses the device No. as a hexadecimal.

Device No. designation (#100101)

The device to be read in is designated by substituting the device No. in this system variable.

Convert a device expressed as a hexadecimal into a decimal when designating.

If the data is read without designating this number, the data will be read in the same manner as when the minimum device No. (0) is designated. Once designated, the setting is held until the device No. is designated again or until it is reset.

A program error (P39) will occur if a nonexistent device No. is set.

Number of bytes designation (#100102)

(1) System variable for number of bytes designation

The reading size is designated by substituting the number of bytes designation value in this system variable.

If the data is read without designating this number, the data will be read in the same manner as when the minimum device designation value (0: M device) is designated. Once designated, the setting is held until the number of bytes is designated again or until it is reset.

A program error (P39) will occur if a number of bytes that does not exist in the specifications is set.

(2) Number of bytes designation value

Number of bytes designation value	Read in data			Operation	
	Size	Sign	Range	Word device	Bit device
0	1 bit	-	0 to 1	The number of bits designated is read in.	The bits for the designated device No. are read in.
1	1 byte	No	0 to 255	The low-order byte is read in.	8 bits are read in from the designated device No.
101		Yes	-128 to 127		
2	2 byte	No	0 to 65535	Two bytes are read in.	16 bits are read in from the designated device No.
102		Yes	-32768 to 32767		
4	4 byte	No	0 to 4294967295	The designated device (L) and next device (H) are read in.	32 bits are read in from the designated device No.
104		Yes	-2147483648 to 2147483647		

0 to 4 are designated without a sign, and 101 to 104 are designated with a sign.

Bit designation (#100103)

(1) System variable for bit designation

The bit to be read in is designated by substituting the bit designation value in this system variable.

This designation is valid only when reading the bits for a 16-bit device, and is invalid for the others.

If the data is read without designating this number, the data will be read in the same manner as if the minimum bit designation value (0: bit 0) is designated. Once designated, the setting is held until the bit is designated again or until it is reset.

A program error (P39) will occur if a nonexistent bit is set.

(2) Bit designation value

Bit designation value	Read in bit
0	Bit 0
1	Bit 1
:	:
15	Bit 15

Reading PLC data (#100110)

The data for the designated device is read in with this system variable.

Refer to the table for number of bytes designation for details on the range of data read in.



Program example

To read a bit device

#100100 = 0;	Designates [M device].
#100101 = 0;	Designates [Device No. 0].
#100102 = 0;	Designates [Bit].
#100 = #100110;	Reads M0 (one bit).
#100102 = 1;	Designates [1 bytes].
#101 = #100110;	Reads M0 to M7 (8 bits). (If M7 to M0 is 0001 0010, this will be #102 = 18 (0x12).)
#100102 = 102;	Designates [Signed two bytes].
#102 = #100110;	Reads M0 to M15 (16 bits). (If M15 to M0 is 1111 1110 1101 1100, this will be #102 = -292 (0xFEDC).)
#100102 = 4;	Designates [4 bytes].
#104 = #100110;	Reads M0 to M31 (32 bits). (If M31 to M0 is 0001 0010 0011 0100 0101 0110 0111 1000, #104 = 305419896 (0x12345678).)

(2) To read a word device

#100100 = 1;	Designates [D device].
#100101 = 0;	Designates [Device No. 0].
#100102 = 0;	Designates [Bit].
#100103 = 1;	Designates [Bit 1].
#100 = #100110;	Reads the D0 bit 1. (If D0 = 0x0102, #101 =1.)
#100102 = 1;	Designates [1 bytes].
#101 = #100110;	Reads the low-order byte of D0. (If D0 = 0x0102, #101 =2.)
#100102 = 2;	Designates [2 bytes].
#102 = #100110;	Reads D0. (If D0 = 0x0102, #102 =258.)
#100102 = 104;	Designates [Signed four bytes].
#104 = #100110;	Reads D0 and D1. (If D0 = 0xFFFFE and D1 = 0xFFFF, #104 =-2.)



Precautions

- (1) As the PLC data is read asynchronously from the ladder execution, the data is not necessarily the one which was gained when the program was executed. Be careful when reading devices which are changing.
- (2) If reading of a nonexistent device is attempted by designating the device No. and number of bytes, the 0 value will be read in only for the nonexistent section.

13.5.5.20 Time Reading Variables (#3001, #3002, #3011, #3012)



Detailed description

The following operations can be carried out using the system variable extension for the user macro time.

- (1) By adding time information system variable #3011 and #3012, the current date (#3011) and current time (#3012) can be read and written.
- (2) By adding parameter #1273/bit1, the unit (millisecond unit/hour unit) of the cumulative time during system variable #3002 automatic start, can be changed.

Variable No.	Details
#3001	The cumulative time during power ON can be read and the value can be substituted. The unit is millisecond.
#3002	The cumulative time during automatic start can be read and the value can be substituted. The unit can be changed between millisecond and hour with parameter #1273/bit1.
#3011	The current date can be read and written. YYYY/MM/DD is read as a YYYYMMDD value. Command range for year/month/date setting Year (YYYY) : 2000 to 2099 Month (MM) : 1 to 12 Date (DD) : 1 to maximum number of days in month
#3012	The current time can be read and written. HH/MM/SS is read as a value "HHMMSS". When a value "HHMMSS" is written in, it will be set as HH/MM/DD. Command range for hour/minute/second setting Hour (HH) : 0 to 23 (24-hour system) Minute (MM) : 0 to 59 Second (SS) : 0 to 59

- (3) The cumulative time returns to 0 at approx. 2.44×10^{11} milliseconds (approx. 7.7 years).
- (4) If a negative value or a value exceeding 244335917226 milliseconds (67871.08811851 hours for #3002 time designation) is set for the cumulative time, a program error (P35) will occur.
- (5) If a value exceeding the command range is set for the date or time, a program error (P35) will occur.
- (6) Always set the month/date/hour/minute/second as a two-digit value when setting the date and time.
If the value only has one digit, always add 0. (February 14, 2001 => #3001= 20010214 ;, etc.)



Program example

Example of use (#3011, #3012)

(Example 1) To read the current date (February 14, 2001) in common variable #100

#100 = #3011 ; (20010214 is inserted in #100)

(Example 2) To write current time (18 hours, 13 minutes, 6 seconds) into system variable #3012

#3012 = 181306 ; (The command value cumulative time #2: time is set to 18:13:06.)

(Example 3) By setting the following program example, the machining start/end time (year/month/date/hour/minute/second) can be viewed.

```
#100=#3011 ; => Machining start year/month/date
#101=#3012 ; => Machining start hour/minute/second
G28 X0 Y0 Z0 ;

G92 ;
G0 X50. ;
:
:
:
#102=#3011 ; .... Machining end year/month/date
#103=#3012 ; .... Machining end hour/minute/second
M30 ;
```



Precautions

Limits and precautions for using time reading variable

- (1) #3011 reads the date as an eight-digit value, so the difference between the two dates read in will not be the difference of days.
- (2) #3012 reads the time as a six-digit value, so the difference between the two times read in will not be the difference of hours.

13.5.5.21 R Device Access Variables (#50000 - #50749, #51000 - #51749, #52000 - #52749)



Function and purpose

By using variable Nos. #50000 to #50749, #51000 to #51749, #52000 to #52749, it is possible to read data (R8300 to R9799, R18300 to R19799, R28300 to R29799) and substitute value in the R device user backup area.

Variable No.	R device	
#50000	R8300,R8301	User backup area (1500 points)
#50001	R8302,R8303	
:		
#50749	R9798,R9799	

Variable No.	R device	
#51000	R18300,R18301	User backup area (1500 points)
#51001	R18302,R18303	
:		
#51749	R19798,R19799	

Variable No.	R device	
#52000	R28300,R28301	User backup area (1500 points)
#52001	R28302,R28303	
:		
#52749	R29798,R29799	



Detailed description

These variables read and write the two words of R device.

Data range of these variables is -2147483648 to 2147483647.

Depending on the setting of the PLC bit selection parameter "#6455/ bit0 to 2", these variables can be changed between decimal point valid or invalid for each user backup area.

The position of the decimal point when decimal point valid is selected, varies according to the parameters "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).


#1041 I_inch	#1003 iunit	
	B	C
0 : Metric	Three digits after the decimal point	Four digits after the decimal point
1 : Inch	Four digits after the decimal point	Five digits after the decimal point

These variables are retained even when the power is off.

Access from a machining program to R device

[Reading variables]

When the variable #50000 is used in a machining program as shown below, the data set in device R8300 and R8301 will be referred.

GO X#50000 ; 	Device	Value	#50000 0x1e240 (hexadecimal) = 123456 (decimal)
	R8301	0x0001	
	R8300	0xe240	

- (1) When decimal point invalid is selected

Regardless of the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), the data set in the R device will be the command value.

In case of the above example, the command value will be "X123456."

- (2) When decimal point valid is selected

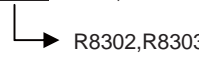
The data set in the R device will be read as a data with a decimal point.

The position of the decimal point will be as follows, according to the settings of the parameters "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch	#1003 iunit	
	B	C
0: Metric	X123.456	X12.3456
1: Inch	X12.3456	X1.23456

[Substitution into variables]

When substituting a value to the variable #50000 in a machining program as shown below, data will be set in the device R8302 and R8303.

#50001 = 123 ;


- (1) When decimal point invalid is selected

Regardless of the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), substituted value will be set in the R device.

#50001	Device	Value
123 (decimal) = 0x7b (hexadecimal)	R8303	0x0000
	R8302	0x007b

When a value with a decimal point is substituted to a variable like "#50001 = 123.456 ;", the numbers after the decimal point will be truncated and "123" will be set.

(2) When decimal point valid is selected

According to the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), values which are shifted for the number of decimals will be set in the R device, as shown below.

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50001		123000 (decimal) = 0x1e078 (hexadecimal)	1230000 (decimal) = 0x12c4b0 (hexadecimal)
Device	R8303	0x0001	0x0012
	R8302	0xe078	0xc4b0

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50001		1230000 (decimal) = 0x12c4b0 (hexadecimal)	12300000 (decimal) = 0xbbaee0 (hexadecimal)
Device	R8303	0x0012	0x00bb
	R8302	0xc4b0	0xae0

When a value with a decimal point is substituted to a variable like "#50001 = 123.456 ;", the value will directly be set.

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50001		123456 (decimal) = 0x1e240 (hexadecimal)	1234560 (decimal) = 0x12d680 (hexadecimal)
Device	R8303	0x0001	0x0012
	R8302	0xe240	0xd680

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50001		1234560 (decimal) = 0x12d680 (hexadecimal)	12345600 (decimal) = 0xbc6100 (hexadecimal)
Device	R8303	0x0012	0x00bc
	R8302	0xd680	0x6100

If the number of decimals of the substituted data exceeds the number of significant figures, the value will be rounded off to the number of significant figures and will be set.

When "#50001 = 123.4567899 ;".

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50001		123457 (decimal) = 0x1e241 (hexadecimal)	1234568 (decimal) = 0x12d688 (hexadecimal)
Device	R8303	0x0001	0x0012
	R8302	0xe241	0xd688

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50001		1234568 (decimal) = 0x12d688 (hexadecimal)	12345679 (decimal) = 0xbc614f (hexadecimal)
Device	R8303	0x0012	0x00bc
	R8302	0xd688	0x614f

Use of R device access variables in control command

These variables can be used in control command.

However, note that the variable value and the condition of true/false differ between decimal point valid variables and invalid variables.

```
IF [#50003 EQ 1] GOTO 30 ;
G00 X100 ;
N30
```

- (1) When decimal point invalid is selected

Regardless of the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), R device value of #50003 whose condition is true, will be "1".

#50003	Device	Value
1 (decimal) = 0x01 (hexadecimal)	R8307	0x0000
	R8306	0x0001

- (2) When decimal point valid is selected

The condition is true when #50003 is "1". So the R device value of #50003 will be as follows depending on the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch	0 : Metric		
#1003 iunit	B	C	
#50003	1000 (decimal) = 0x3e8 (hexadecimal)	10000 (decimal) = 0x2710 (hexadecimal)	
Device	R8307	0x0000	0x0000
	R8306	0x03e8	0x2710

#1041 I_inch	1 : Inch		
#1003 iunit	B	C	
#50003	10000 (decimal) = 0x2710 (hexadecimal)	100000 (decimal) = 0x186a0 (hexadecimal)	
Device	R8307	0x0000	0x0001
	R8306	0x2710	0x86a0

Substitution between R device access variable and other variables.

[Substitution into R device access variables]

Common variables and coordinates variables can be substituted to the R device access variables.

(Example 1) Common variables

```
#101 = -123.456 ;
#50004 = #101 ;
```

(Example 2) #5063 : Skip coordinates #5063

```
#50004 = #5063 ;
```

(1) When decimal point invalid is selected

Regardless of the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), the value which is rounded off will be set.

- When the common variable and coordinate variable in the above example are "-123.456"

#50004	Device	Value
-123 (decimal) = 0xfffff85 (hexadecimal)	R8309	0xffff
	R8308	0x0085

(2) When decimal point valid is selected

Substitution will be as follows according to the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50004		-123456 (decimal) = 0xfffe1dc0 (hexadecimal)	-1234560 (decimal) = 0xffed2980 (hexadecimal)
Device	R8309	0xfffe	0xffed
	R8308	0x1dc0	0x2980

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50004		-1234560 (decimal) = 0xffed2980 (hexadecimal)	-12345600 (decimal) = 0xff439f00 (hexadecimal)
Device	R8309	0xffed	0xff43
	R8308	0x2980	0x9f00

[Substitution of R device access variables]

```
#50005 = 123.456789 ;
#102 = #50005
```

- (1) When decimal point invalid is selected
Regardless of the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), #102 will be "123".
- (2) When decimal point valid is selected
Substitution will be as follows according to the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch	0: Metric	
#1003 iunit	B	C
#102	123.4570	123.4568

#1041 I_inch	1 : Inch	
#1003 iunit	B	C
#102	123.4568	123.4568

**Precautions**

- (1) The position of a decimal point changes depending on the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch). Fix the decimal point position while considering these parameter settings when setting a number to an R device.
- (2) These variables do not handle <Blank>. If #0<Blank> is substituted, it will be converted into "0". Therefore, when comparing this variable after substituting #0 #0<Blank> and #0 #0<Blank> with a conditional expression (EQ), it will not be formed.
- (3) A program error (P35) will occur when a value exceeding an effective range is substituted into these variables.
- (4) When these values are used as decimal point invalid, the settings of "#1078 Decpt2" (Decimal point type 2) and "#8044 UNIT*10" will not be applied.
- (5) When a graphic is being checked, writing into R device will not be executed even if a value is substituted into these variables.
These variables (reference to the R device value) during a graphic check read the currently set R device value.

13.5.6 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 $\tan \theta$ uses $\sin \theta / \cos \theta$.
	#i = ASIN [#k]	Arcsine
	#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 [].

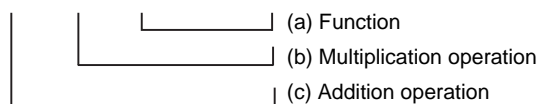


Detailed description

Sequence of operations

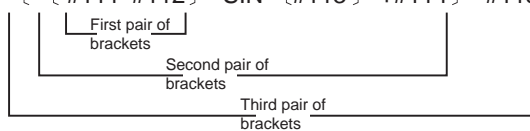
- (1) The sequence of the operations (a) to (c) is performed in the following order; the function, the multiplication operation and the addition operation.

```
#101=#111+#112*SIN [#113]
```



- (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] ] ;
```



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 substitution =	#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 compensation amount				
(3) Addition and subtraction + -	#11 = #1 + 1000 #12 = #2 - 50. #13 = #101 + #1 #14 = #10001 - 3. (#10001 = -10.) #15 = #10001 + #102	#11 2000.000 #12 950.000 #13 1100.000 #14 -13.000 #15 190.000					
(4) Multiplication and division * /	#21 = 100 * 100 #22 = 100. * 100 #23 = 100 * 100. #24 = 100. * 100. #25 = 100 / 100 #26 = 100. / 100 #27 = 100 / 100. #28 = 100. / 100. #29 = #10001 * #101 (#10001 = -10.) #30 = #10001 / #102	#21 10000.000 #22 10000.000 #23 10000.000 #24 10000.000 #25 1.000 #26 1.000 #27 1.000 #28 1.000 #29 -1000.000 #30 -0.050					
	(5) Remainder MOD	#19 = 48 #20 = 9 #31 = #19 MOD #20	#19/#20 = 48/9 = 5 with 3 over #31 = 3				
	(6) Logical sum OR	#3 = 100 #4 = #3 OR 14	#3 = 01100100 (binary) 14 = 00001110 (binary)				
			#4 = 01101110 = 110				
	(7) Exclusive OR XOR	#3 = 100 #4 = #3 XOR 14	#3 = 01100100 (binary) 14 = 00001110 (binary)				
			#4 = 01101010 = 106				
	(8) Logical product AND	#9 = 100 #10 = #9 AND 15	#9 = 01100100 (binary) 15 = 00001111 (binary)				
			#10 = 00000100 = 4				
	(9) Sin SIN	#501 = SIN [60] #502 = SIN [60.] #503 = 1000 * SIN [60] #504 = 1000 * SIN [60.] #505 = 1000. * SIN [60] #506 = 1000. * SIN [60.] (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	



Precautions

(1) Notes on logical relation

EQ, NE, GT, LT, GE and LE conduct the same calculation as addition and subtraction. Pay careful attention to errors. For example, to determine whether or not #10 and #20 are equal in the following example, it is not always possible to judge correctly because of the error.

IF [#10 EQ #20]

Therefore when the difference between #10 and #20 falls within the designated error range, both values should be considered equal.

IF [ABS [#10 - #20] LT 200000]

13.5.7 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.

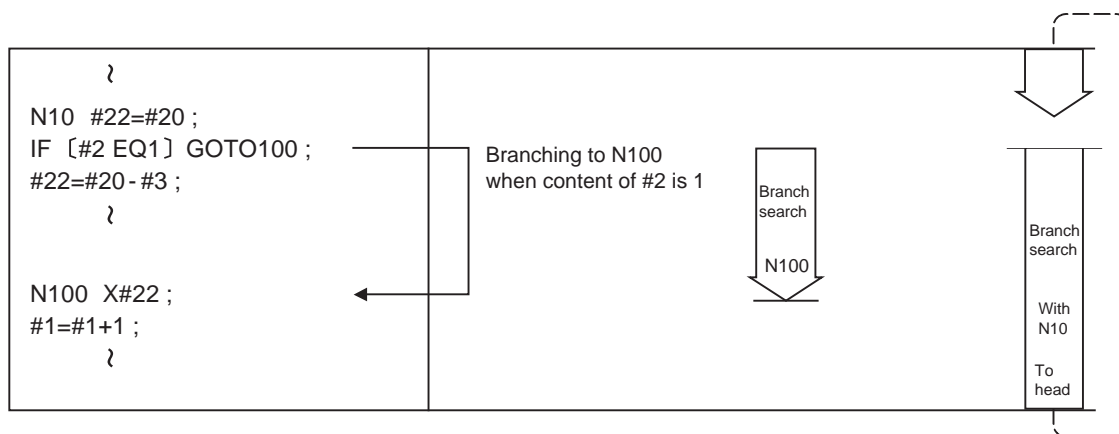
The following types of [conditional expressions] are available.

#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] D0m ; (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. D0m may come before WHILE.

"WHILE [conditional expression] D0m" 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. (D01, D02, D03, D0127) 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-D0m identification No.

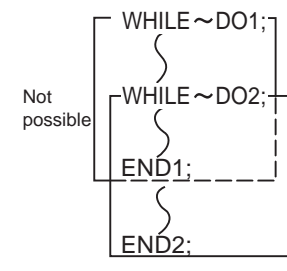
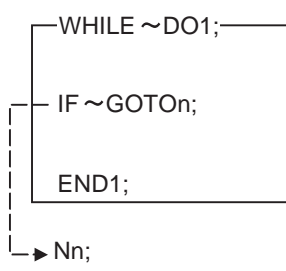
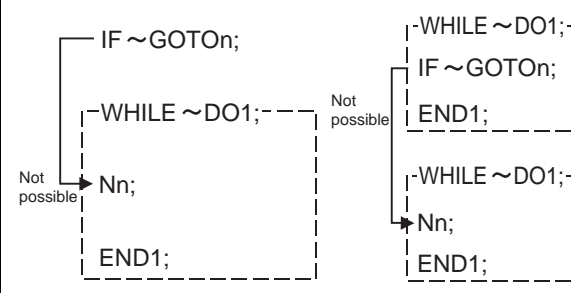
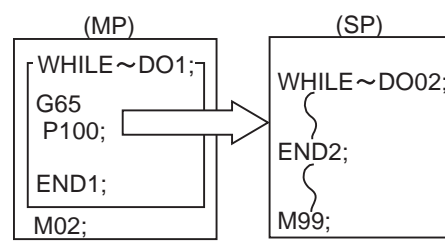
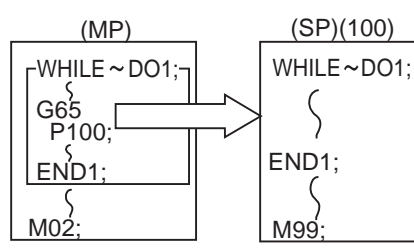
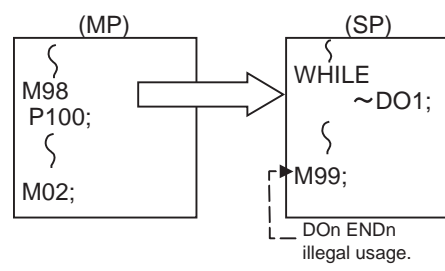
(3) Up to 27 nesting levels can be used for WHILE-D0m. "m" is any number from 1 to 127 for the nesting depth.

(4) The number of WHILE-D0m nesting levels cannot exceed 27.

(Note) For nesting, "m" which has been used once cannot be used.

(5) WHILE - D0m must be designated first and ENDm last.

(6) WHILE - D0m and ENDm must correspond on a 1:1 (pairing) basis in a same program.

<p>(7) Two WHILE - DOm's must not overlap.</p> 	<p>(8) Branching externally out of the WHILE - DOm range, is possible.</p> 
<p>(9) No branching into WHILE - DOm, is possible.</p> 	<p>(10) Subprograms can be called by M98, G65 or G66 between WHILE - DOm's.</p> 
<p>(11) Calls can be initiated by G65 or G66 between WHILE - DOm's and commands can be issued again from 1. Up to 27 nesting levels are possible for the main program and subprograms.</p> 	<p>(12) A program error will occur in M99 if WHILE and END are not paired in the subprogram (including macro subprogram).</p> 

(MP) Main program

(SP) Subprogram

(Note) Even if a fixed cycle containing WHILE is called, the nesting level will be counted up.

13.5.8 External Output Commands ; POPEN, PCLOS, DPRNT



Function and purpose

Besides the standard user macro commands, the following macro instructions are also available as external output commands. They are designed to output the variable values or characters to external devices. The data output port can be chosen from RS-232C interface or memory card.



Command format

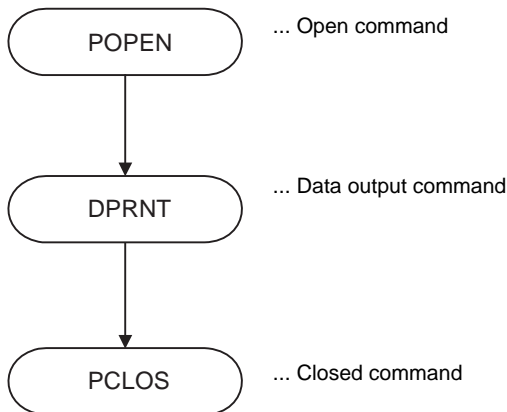
POPEN ... Open command

PCLOS ... Closed command

DPRNT ... Data output command

POPEN	For preparing the data outputs
PCLOS	For terminating the data outputs
DPRNT	For character output and digit-by-digit variable numerical output

Command sequence



Detailed description

Open command : POPEN

- (1) The command is issued before the series of data output commands.
- (2) The DC2 control code and % code are output from the NC system to the external output device.
- (3) Once POPEN; is issued, it will remain valid until PCLOS; is issued.

Close command : PCLOS

- (1) This command is issued when all the data outputs are completed.
- (2) The DC4 control code and % code are output from the NC system to the external output device.
- (3) This command is used together with the open command and it should not be issued unless the open mode has been established.
- (4) Issue the close command at the end of the program even when the operation is suspended by resetting or some other operation during data output.

Data output command : DPRNT

DPRNT [l1#v1 [d1 c1] l2#v2 [d2 c2]] ;

l1	Character string	
v1	Variable No.	
d1	Significant digits above decimal point	c + d <= 8
c1	Significant digits after decimal point	

- (1) The character output and decimal output of the variable values are done with ISO codes.
- (2) The commanded character string is output as it is by the ISO code.
Alphanumerics (A to Z, 0 to 9) and special characters (+, -, *, /) can be used.
Note that asterisk (*) is output as a space code.
- (3) The required significant digits above and below the decimal point of the variable values are each commanded within square parentheses. As a result, the commanded number of digits of variable values are output in ISO code in decimal notation from the high-order digits including the decimal point. In this case, trailing zeroes are not omitted.
- (4) Leading zeroes are omitted.
The omitted leading zero can be replaced by a space by the setting of a parameter.
This can justify the last digit of the data output to the printer.
- (5) Linefeed (LF) code will be output to the end of the output data.
And by setting the parameter "#9112 to #9512 DEV0 - 4 CR OUTPUT" to "1", (CR) code will be written in just before EOB (LF) code.

Data output port

- (1) The output port can be selected by the parameter "#9007 MACRO PRINT PORT".
- (2) When the port is a memory card, the file name of the port can be designated by the parameter "#9054 MACRO PRINT FILE".
- (3) When the port is a memory card, the port directory is fixed to root directory.



Example of use

<Parameter setting>

#1127 DPRINT (DPRINT alignment)	= 1 (Align the minimum digit and output)
#9007 MACRO PRINT PORT	= 9 (Output to a memory card by an external output command)
#9008 MACRO PRINT DEV.	= 0 (Device 0 is selected for an external output command)
#9054 MACRO PRINT FILE	= DPRNT_OUT (File name to store output data of an external output command)
#9112 DEV0 CR OUTPUT	= 1 (Insert the CR code just before the LF code)

<Machining program>

```
#1=12.34;
#2=#0
#100=-123456789.;
#500=-0.123456789;
POPEN;
DPRNT[]; (Note)
DPRNT[VAL-CHECK];
DPRNT[1234567890];
DPRNT[#1[44]];
DPRNT[#2[44]];
DPRNT[#100[80]];
DPRNT[#500[80]];
DPRNT[#100[08]];
DPRNT[#500[08]];
PCLOS;
M30;
%
```

(Note) Designate a blank line when opening an output file on an edit screen. If not, it will be regarded that no information is provided in the head line of the file on the NC edit screen.

<Output file (file name = DPRNT_OUT)>

Blank										Linefeed code	
(CR)	(LF)									(CR)	(LF)
V	A	L	-	C	H	E	C	K		(CR)	(LF)
1	2	3	4	5	6	7	8	9	0	(CR)	(LF)
			1	2	.	3	4	0	0	(CR)	(LF)
				0	.	0	0	0	0	(CR)	(LF)
-	2	3	4	5	6	7	8	9		(CR)	(LF)
							-	0		(CR)	(LF)
-	.	0	0	0	0	0	0	0	0	(CR)	(LF)
-	.	1	2	3	4	5	6	7	9	(CR)	(LF)

Values above the number of significant figures are rounded down

Values below the number of significant figures are rounded off



Precautions

- (1) An external output command during restart-searching is ignored.
After using the restart-search type 2 to restart-search between a POPEN command and a PCLOS command, execute a POPEN command by such as an MDI interruption before restarting the program.
- (2) An external output command during graphic check is ignored.
- (3) A program error (P460) will occur if an external output command is issued when the output device is unable to output due to a lack of connection, a low free space etc.
- (4) The NC automatically conducts a closing processing when it is reset between a POPEN command and a PCLOS command. So, execute a POPEN command by such as an MDI interruption before executing the rest of the machining program.
- (5) If a program error occurs between a POPEN command and a PCLOS command, NC will not automatically conduct a closing processing. So, there is no need to execute a POPEN command by such as an MDI interruption before executing the rest of the machining program.
- (6) If a program error occurs to the output port due to the setting of the memory card, execute an NC reset and close the output file before demounting the card.
- (7) When the output port is a memory card, the output file may be destroyed if the card is dismounted or the power is turned off without issuing a PCLOS command or NC reset after a POPEN command is issued.
- (8) When the data is output to a memory card, the maximum number of files that can be created is determined by the FAT16 format.

13.5.9 Precautions



Precautions

When the user macro commands are employed, it is possible to use the M, S, T and other NC control commands together with the arithmetic, decision, branching and other macro commands for preparing the machining programs. When the former commands are made into executable statements and the latter commands into macro statements, the macro statement processing should be accomplished as quickly as possible in order to minimize the machining time, because such processing is not directly related to machine control.

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:

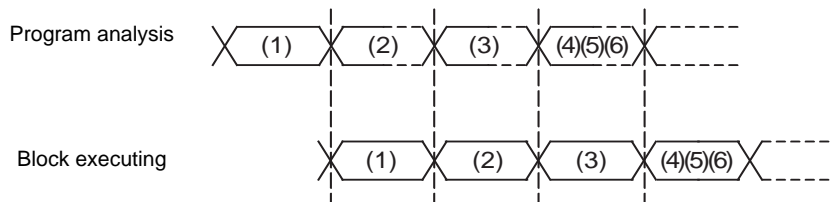
- (a) Arithmetic commands (block including =)
- (b) Control commands (block including GOTO, DO-END, etc.)
- (c) Macro call commands (including macro calls based on G codes and cancel commands (G65, G66, G66.1, G67))

Execution statements refer to statements other than macro statements.

Flow of processing by the Program Example in the previous page

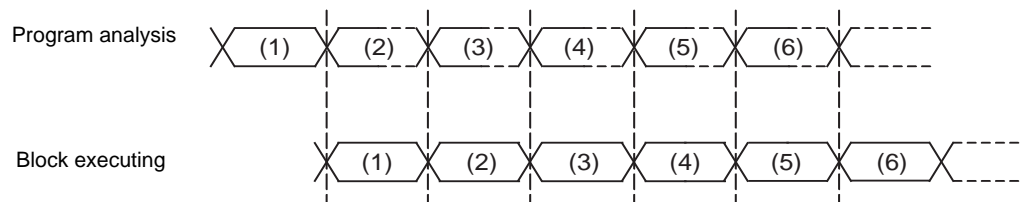
<Macro single OFF>

N4, N5 and N6 are processed in parallel with the control of the executable statement of N3. If the N4, N5 and N6 analysis is in time during N3 control, the machine movement will be continuously controlled.



<Macro single ON>

N4 is processed in parallel with the control of the executable statement of N3. 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.



13.5.10 Actual Examples of Using User Macros



Program example

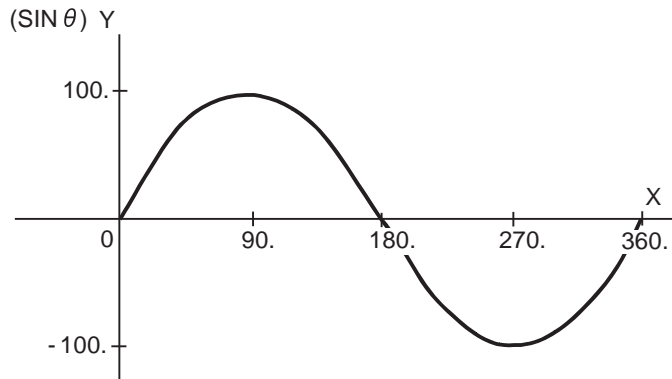
The following three examples will be described.

- (Example 1) SIN curve
- (Example 2) Bolt hole circle
- (Example 3) Grid

(Example 1) SIN curve

G65 Pp1 Aa1 Bb1 Cc1 Ff1;

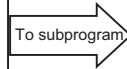
a1; Initial value 0°
 b1; Final value 360°
 c1; R of R*SIN θ
 f1; Feedrate



Main Program

```

}
G65 P9910 A0 B360.C100.F100;
}
    
```

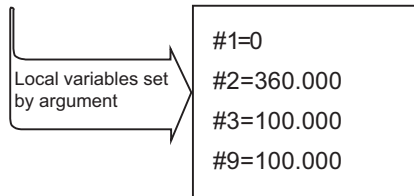


O9910(Subprogram)

```

WHILE [#1LE#2] DO1;
#101=#3*SIN [#1] ;
G90 G01 X#1 Y#10 F#9;
#1=#1+10.;
END1;
M99;
    
```

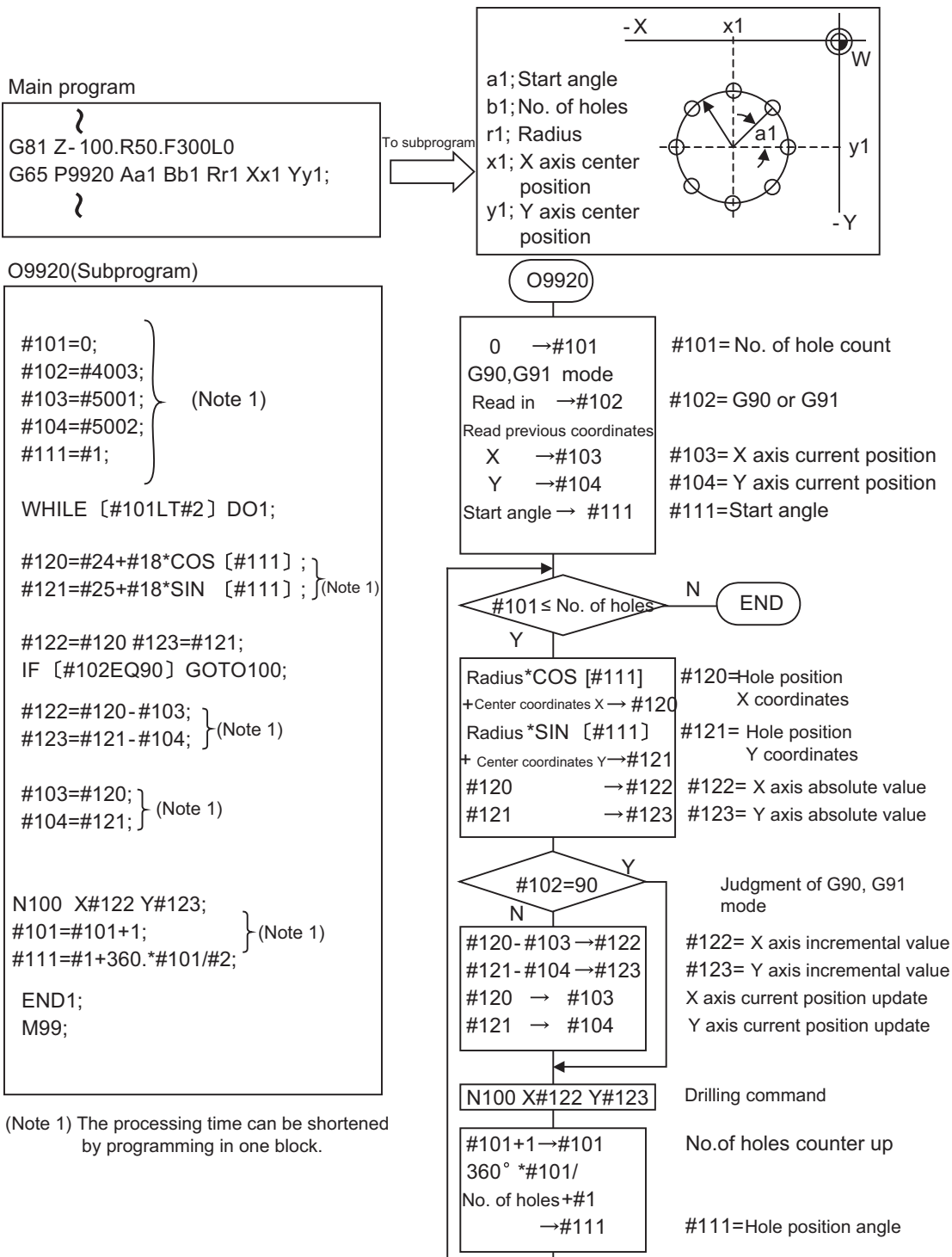
} (Note 1)



(Note 1) Commanding with one block is possible when G90G01X#1Y[#3*SIN[#1]] F#9; is issued.

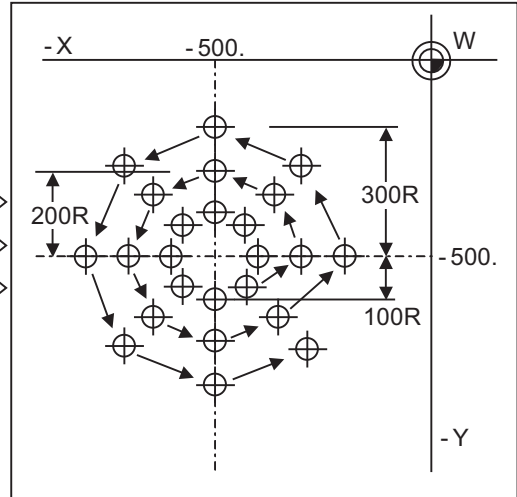
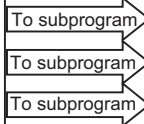
(Example 2) Bolt hole circle

After defining the hole data with fixed cycle (G72 to G89), the macro command is issued as the hole position command.



(Note 1) The processing time can be shortened by programming in one block.

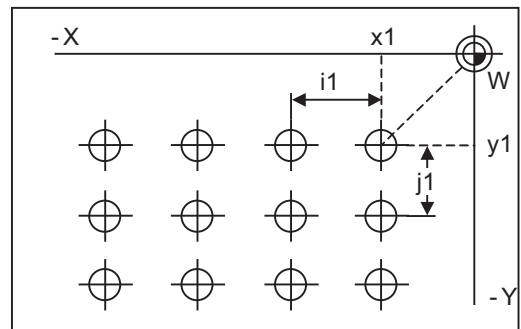
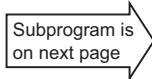
```
G28 X0 Y0 Z0;
T1 M06;
G90 G43 Z100.H01;
G54 G00 X0 Y0;
G81 Z-100.R3.F100 L0 M03;
G65 P9920 X-500.Y-500.A0 B8 R100.;
G65 P9920 X-500.Y-500.A0 B8 R200.;
G65 P9920 X-500.Y-500.A0 B8 R300.;
}
```



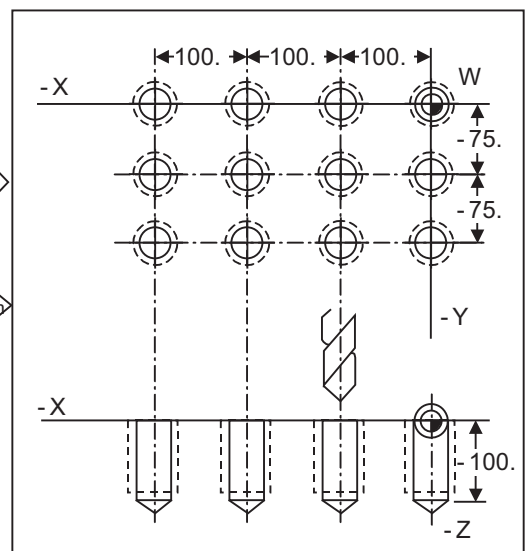
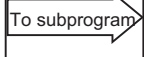
(Example 3) Grid

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
```



```
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;
}
```



O9930 (Subprogram)

```

#101=#24;
#102=#25;

#103=#4;
#104=#5;

#106=#2;

WHILE [#106GT0] DO1;

#105=#1;

WHILE [#105GT0] DO2;

G90 X#101 Y#102;

#101=#101+#103;
#105=#105-1;

END2;

#101=#101-#103;
#102=#102+#104;

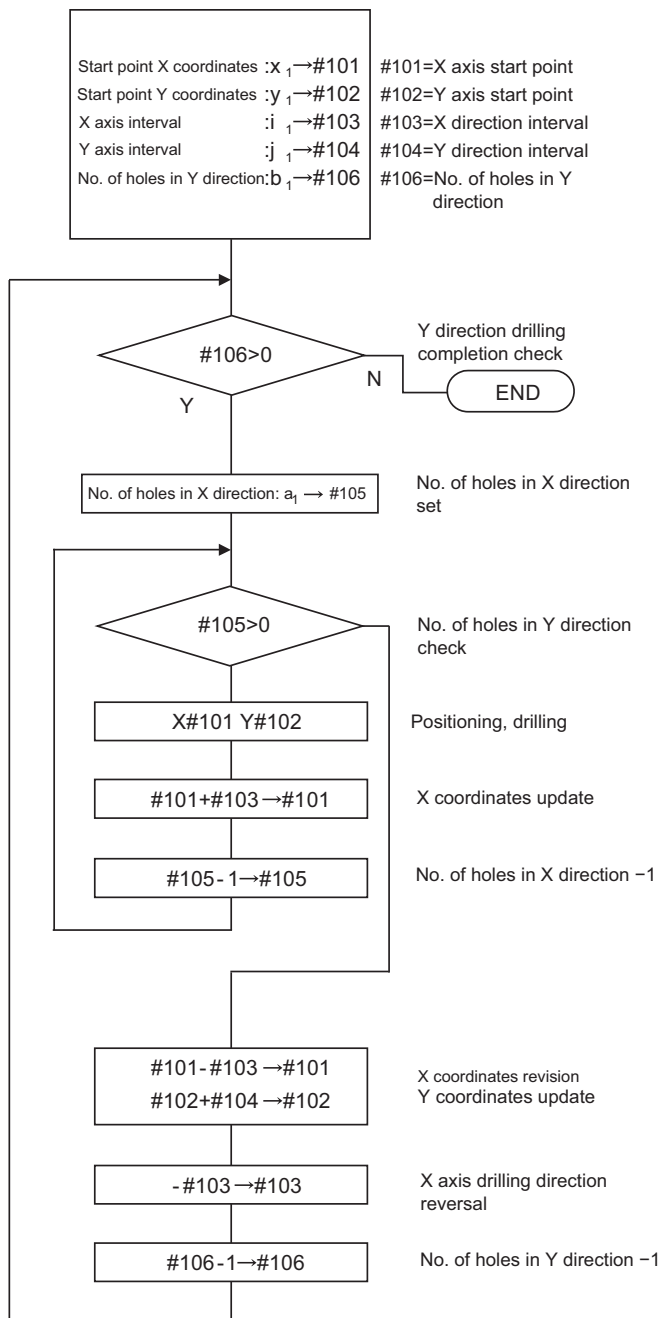
#103=-#103;
#106=#106-1;

END1;

M99;
    
```

(Note 1) The processing time can be shortened by programming in one block.

O9930



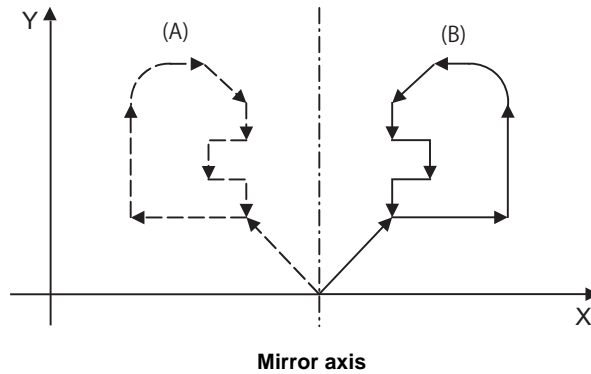
13.6 G Command Mirror Image ; G50.1,G51.1



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.



Command format

Mirror image ON

G51.1 Xx1 Yy1 Zz1

x1, y1, z1	Mirror image center coordinates (Mirror image will be applied regarding this position as a center)
------------	---

Mirror image OFF

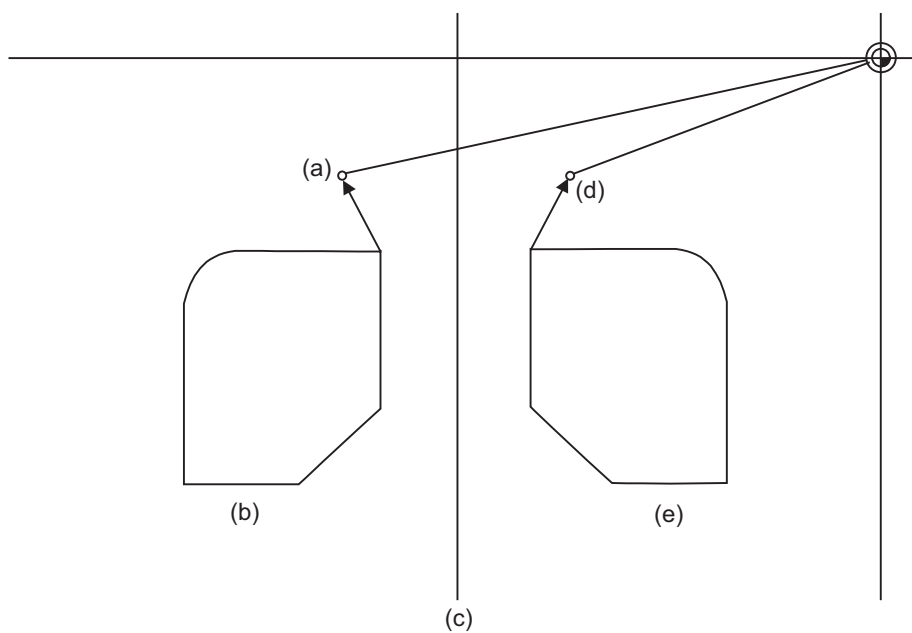
G50.1 Xx2 Yy2 Zz2

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

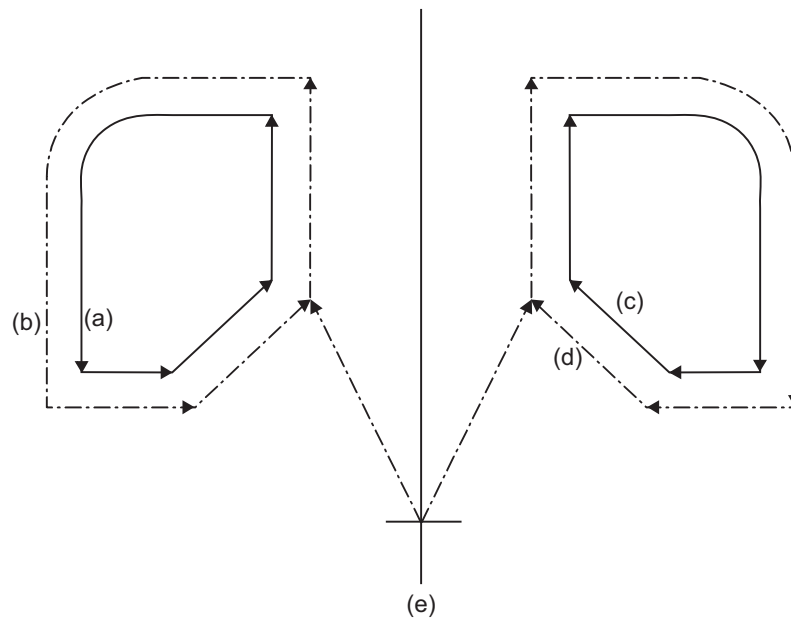
- (6) Return from zero point during mirror image
If the return command (G29) from the zero point is commanded during the mirror image, the mirror will be applied on the intermediate point.
- (7) The mirror image will not be applied on the G53 command.



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

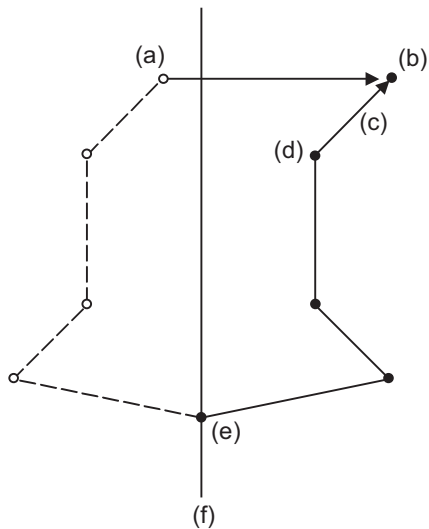


Precautions

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 I / 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 I ; 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

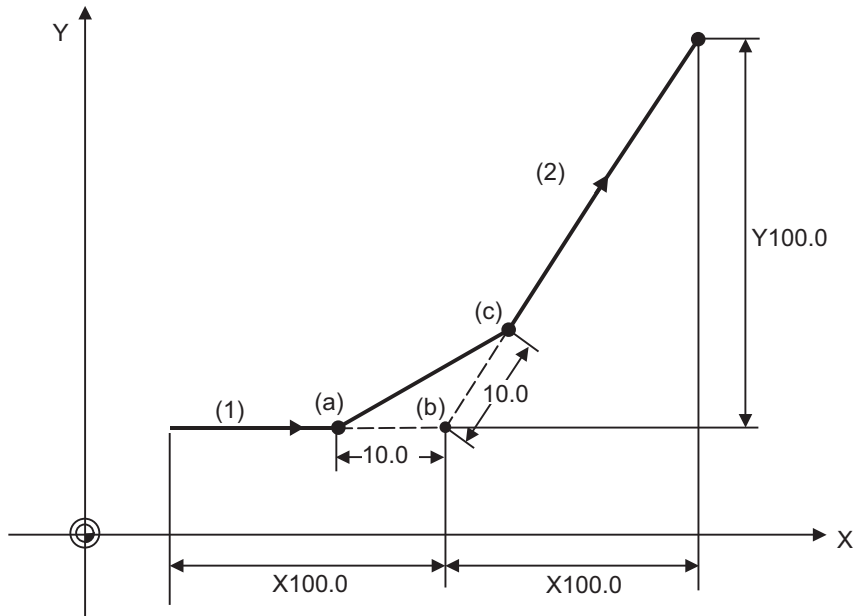
- (1) The start point of the block following the corner chamfering is the hypothetical corner intersection point.
- (2) If there are multiple or duplicate corner chamfering commands in a same block, the last command will be valid.
- (3) When both the corner chamfer and corner rounding commands exist in the same block, the latter command is valid.
- (4) Tool compensation is calculated for the shape which has already been subjected to corner chamfering.
- (5) When the block following a command with corner chamfering does not contain a linear command, a corner chamfering/corner rounding II command will be executed.
- (6) Program error (P383) will occur when the movement amount in the corner chamfering block is less than the chamfering amount.
- (7) Program error (P384) will occur when the movement amount in the block following the corner chamfering block is less than the chamfering amount.
- (8) Program error (P382) will occur when a movement command is not issued in the block following the corner chamfering I command.



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 I ; 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

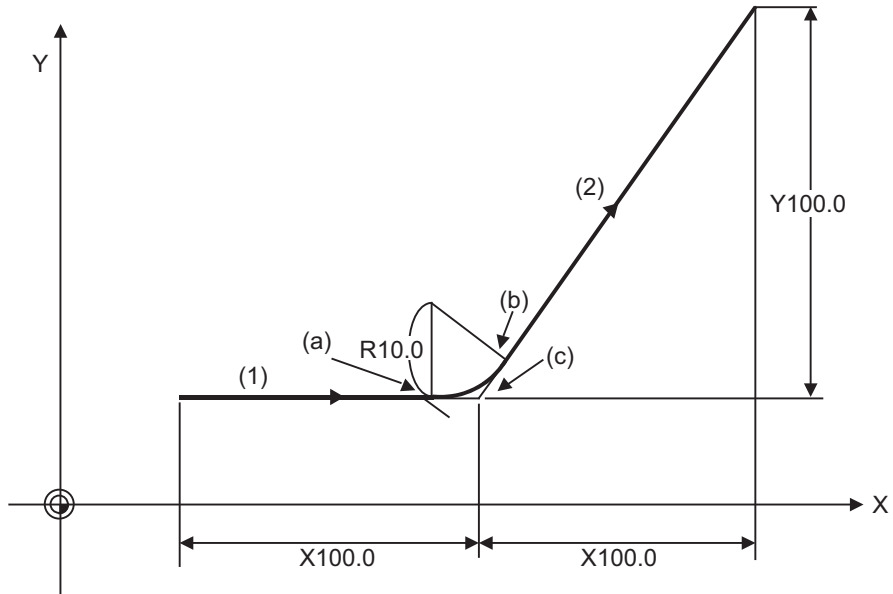
- (1) The start point of the block following the corner rounding is the hypothetical corner intersection point.
- (2) When both the corner chamfer and corner rounding commands exist in the same block, the latter command is valid.
- (3) Tool compensation is calculated for the shape which has already been subjected to corner rounding.
- (4) When the block following a command with corner rounding does not contain a linear command, a corner chamfering/corner rounding II command will be executed.
- (5) Program error (P383) will occur when the movement amount in the corner rounding block is less than the R value.
- (6) Program error (P384) will occur when the movement amount in the block following the corner rounding block is less than the R value.
- (7) Program error (P382) will occur if a movement command is not issued in the block following the corner rounding.



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

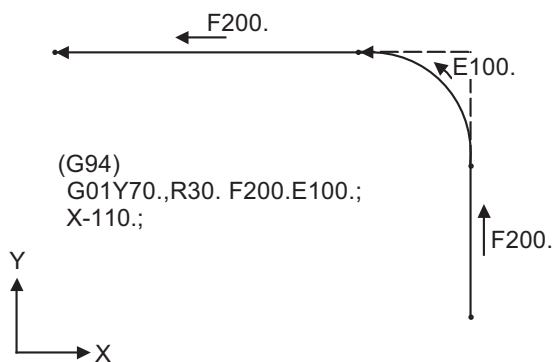
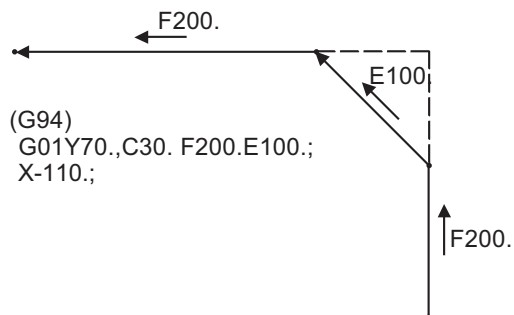
13.7.3 Corner Chamfering Expansion/Corner Rounding Expansion



Function and purpose

Using an E command, the feedrate can be designated for the corner chamfering and corner rounding section. In this way, the corner section can be cut into a correct shape.

Example

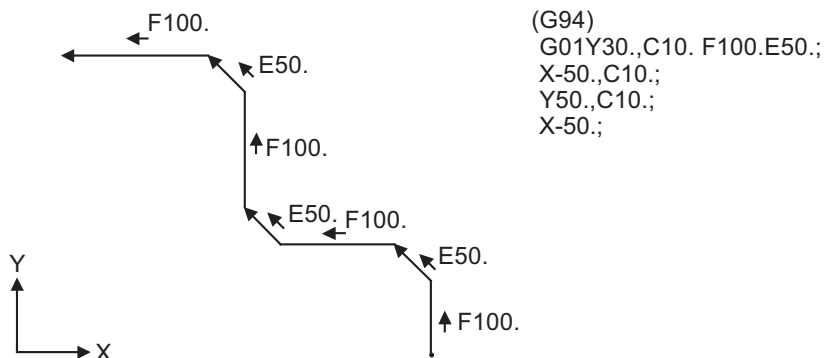




Detailed description

- (1) The E command is modal. It is also valid for the feed in the next corner chamfering/corner rounding section.

Example

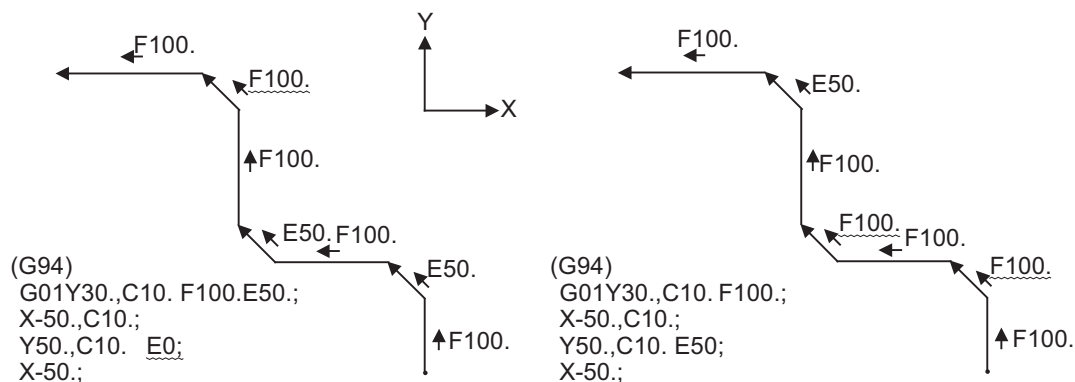


- (2) E command modal has separate asynchronous feedrate modal and synchronous feedrate modal functions.

Which one is validated depends on the asynchronous/synchronous mode (G94/G95).

- (3) When the E command is 0, or when there has not been an E command up to now, the corner chamfering/corner rounding section feedrate will be the same as the F command feedrate.

Example



- (4) E command modal is not cleared even if the reset button is pressed.
It is cleared when the power is turned OFF. (In the same manner as F commands.)
- (5) All E commands except those shown below are at the corner chamfering/corner rounding section feedrate.
- E commands during thread cutting modal
 - E commands during thread cutting cycle modal

13.7.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding

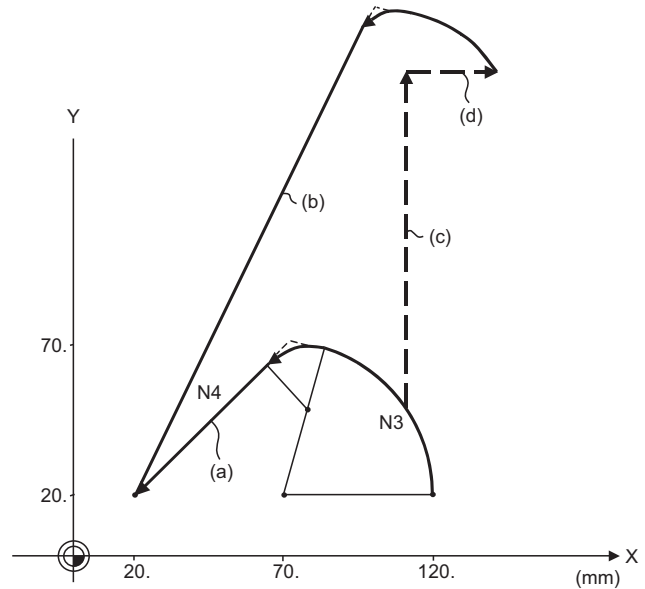


Detailed description

(1) Shown below are the operations of manual interruption during corner chamfering or corner rounding.

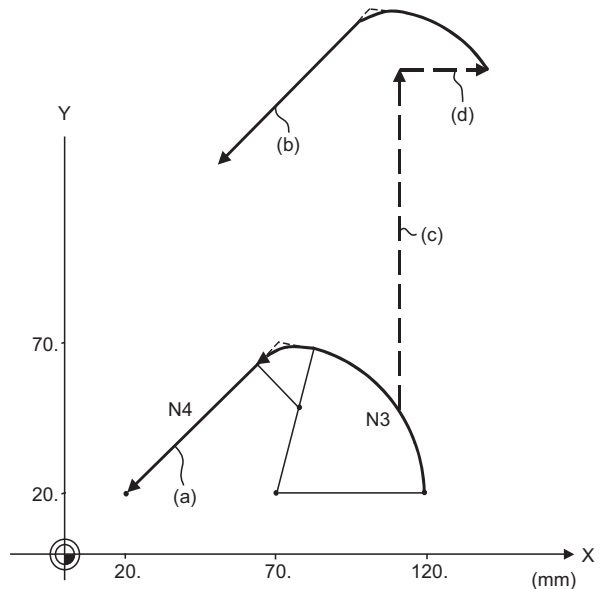
With an absolute value command and manual absolute switch ON.

```
N1 G28 XY;
N2 G00 X120.Y20. ;
N3 G03 X70. Y70.I-50. ,R20. F100 ;
N4 G01 X20. Y20. ;
:
```



With an incremental value command and manual absolute switch OFF

```
N1 G28 XY;
N2 G00 X120. Y20. ;
N3 G03 X-50. Y50. I-50. ,R20. F100 ;
N4 G01 X-50. Y-50.;
:
```



- (a) When interrupt is not applied
- (b) When interrupt is applied
- (c) X-axis interrupt
- (d) Z-axis interrupt

(2) With a single block during corner chamfering or corner rounding, the tool stops after these operations are executed.

13.8 Corner Chamfering II / Corner Rounding II



Function and purpose

Corner chamfering and corner rounding can be performed by adding ",C" or ",R" to the end of the block which is commanded first among the block that forms a corner with continuous arbitrary angle lines or arcs.

13.8.1 Corner Chamfering II ; G01/G02/G03 X_ Y_ ,C_



Function and purpose

The corner is chamfered by commanding ",C" in the 1st block of the two continuous blocks containing an arc. For an arc, this will be the chord length.



Command format

```
N100 G03 X__ Y__ I__ J__ ,C__ ;
N200 G01 X__ Y__ ;
```

.C	Length up to chamfering starting point or end point from hypothetical corner
----	--

Corner chamfering is performed at the point where N100 and N200 intersect.



Detailed description

- (1) The corner chamfering and corner rounding options are required to use this function. A program error (P381) will occur if the function is commanded without the option.
- (2) The start point of the block following the corner chamfering is the hypothetical corner intersection point.
- (3) If there are multiple or duplicate corner chamfering commands in a same block, the last command will be valid.
- (4) When both corner chamfering and corner rounding are commanded in the same block, the latter command will be valid.
- (5) Tool compensation is calculated for the shape which has already been subjected to corner chamfering.
- (6) Program error (P385) will occur when positioning or thread cutting is commanded in the corner chamfering command block or in the next block.
- (7) Program error (P382) will occur when the block following corner chamfering contains a G command other than group 01 or another command.
- (8) Program error (P383) will occur when the movement amount in the block, commanding corner chamfering, is less than the chamfering amount.
- (9) Program error (P384) will occur when the movement amount is less than the chamfering amount in the block following the block commanding corner chamfering.
- (10) Even if a diameter is commanded, it will be handled as a radial command value during corner chamfering.
- (11) Program error (P382) will occur when a movement command is not issued in the block following the corner chamfering II command.

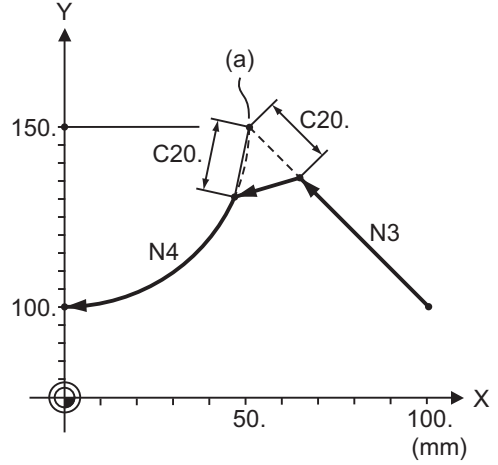


Program example

(1) Linear - arc

```

Absolute value command
N1 G28 XY;
N2 G90 G00 X100. Y100.;
N3 G01 X50.Y150.,C20. F100;
N4 G02 X0 Y100. I-50. J0;
:
Relative value command
N1 G28 XY;
N2 G91 G00 X100. Y100.;
N3 G01 X-50.Y50.,C20. F100;
N4 G02 X-50. Y-50. I-50. J0;
:
    
```

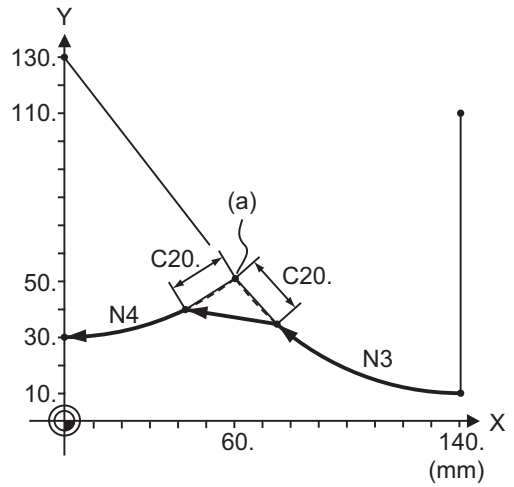


(a) Hypothetical corner intersection point

(2) Arc - arc

```

Absolute value command
N1 G28 XY;
N2 G91 G00 X140. Y10.;
N3 G02 X60.Y50.I0 J100. ,C20. F100;
N4 X0 Y30.I-60.J80.;
:
Relative value command
N1 G28 XY;
N2 G91 G00 X140. Y10.;
N3 G02 X-80.Y40. R100. ,C20. F100;
N4 X-60. Y-20. I-60. J80.;
:
    
```



(a) Hypothetical corner intersection point

13.8.2 Corner Rounding II ; G01/G02/G03 X_ Y_ ,R_



Function and purpose

The corner is rounded by commanding ",R_" in the 1st block of the two continuous blocks containing an arc.



Command format

```
N100 G03 X__ Y__ I__ J__ ,R__ ;
N200 G01 X__ Y__ ;
```

,R	Arc radius of corner rounding
----	-------------------------------

Corner rounding is performed at the point where N100 and N200 intersect.



Detailed description

- (1) The corner chamfering and corner rounding options are required to use this function. Program error (P381) will occur if the function is commanded without the option.
- (2) The start point of the block following the corner rounding is the hypothetical corner intersection point.
- (3) When both corner chamfering and corner rounding are commanded in a same block, the latter command will be valid.
- (4) Tool compensation is calculated for the shape which has already been subjected to corner rounding.
- (5) Program error (P385) will occur when positioning or thread cutting is commanded in the corner rounding command block or in the next block.
- (6) Program error (P382) will occur when the block following corner rounding contains a G command other than group 01 or another 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 is less than the R value in the block following the corner rounding.
- (9) Even if a diameter is commanded, it will be handled as a radial command value during corner rounding.
- (10) A program error (P382) will occur if a movement command is not issued in the block following corner rounding.

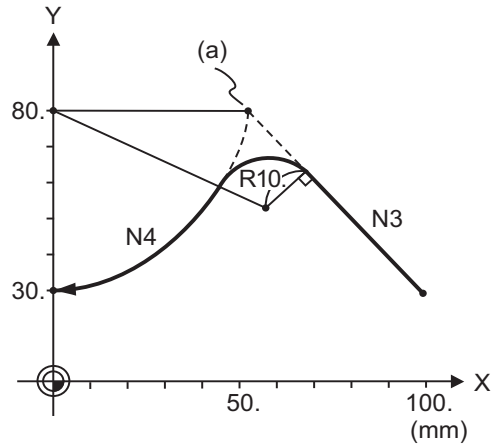


Program example

(1) Linear - arc

```

Absolute value command
N1 G28 XY;
N2 G90 G00 X100. Y30.;
N3 G01 X50.Y80.,R10. F100;
N4 G02 X0 Y30. I-50.J0;
:
Relative value command
N1 G28 XY;
N2 G91 G00 X100. Y30.;
N3 G01 X-50.Y50.,R10. F100;
N4 G02 X-50. Y-50. I-50.J0;
:
    
```

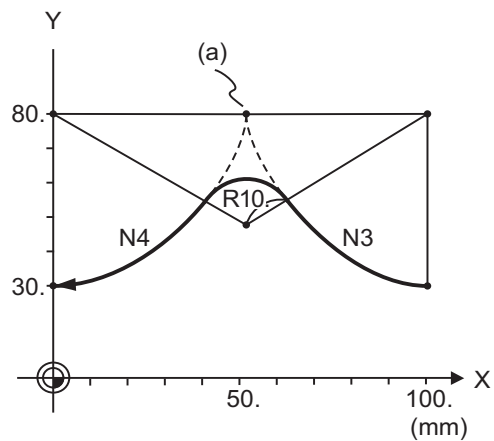


(a) Hypothetical corner intersection point

(2) Arc - arc

```

Absolute value command
N1 G28 XY;
N2 G90 G00 X100. Y30.;
N3 G02 X50.Y80. R50.,R10.F100;
N4 X0 Y30. R50.;
:
Relative value command
N1 G28 XY;
N2 G91 G00 X100. Y30.;
N3 G02 X-50.Y50. I0 J50.,R10.F100;
N4 X-50. Y-50. I-50. J0;
:
    
```



(a) Hypothetical corner intersection point

13.8.3 Corner Chamfering Expansion/Corner Rounding Expansion

For details, refer to "Corner Chamfering I / Corner Rounding I: Corner Chamfering Expansion / Corner Rounding Expansion".

13.8.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding

For details, refer to "Corner Chamfering I / Corner Rounding I: Interrupt during Corner Chamfering Interrupt during / Corner Rounding".

13.9 Linear Angle Command ; G01 X_/Y_ A_/A_



Function and purpose

The end point coordinates are automatically calculated by commanding the linear angle and one of the end point coordinate axes.



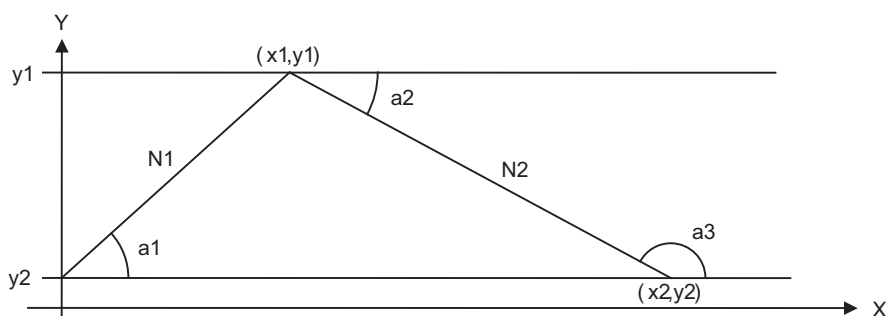
Command format

```
N1 G01 Xx1(Yy1) Aa1;
N2 G01 Xx2(Yy2) A-a2; (A-a2 can also be set as Aa 3.)
```

```
N1 G01 Xx1(Yy1) ,Aa1;
N2 G01 Xx2(Yy2) ,A-a2;
```

This designates the angle and the X or Y axis coordinates.

Select the command plane with G17 to G19.



Detailed description

- (1) As seen from the + direction of the horizontal axis of the selected plane, the counterclockwise (CCW) direction is considered to be + and the clockwise direction (CW) -.
- (2) Either of the axes on the selected plane is commanded for the end point.
- (3) The angle is ignored when the angle and the coordinates of both axes are commanded.
- (4) When only the angle has been commanded, this is treated as a geometric command.
- (5) The angle of either the start point (a1) or end point (a2) may be used.
- (6) This function is valid only for the G01 command; it is not valid for other interpolation or positioning commands.
- (7) The range of slope "a" is between -360.000 and 360.000.
When a value outside this range is commanded, it will be divided by 360 (degrees) and the remainder will be commanded.
(Example) If 400 is commanded, 40° (remainder of 400/360) will become the command angle.
- (8) If an address A is used for the axis name or the 2nd miscellaneous function, use ",A" as the angle.
- (9) If "A" and ",A" are commanded in a same block, ",A" will be interpreted as the angle.

13.10 Geometric

13.10.1 Geometric I ; G01 A_



Function and purpose

When it is difficult to calculate the intersection point of two straight lines in a continuous linear interpolation command, the end point of the first straight line will be automatically calculated inside the CNC and the movement command will be controlled, provided that the slope of the first straight line as well as the end point coordinates and slope of the second straight line are commanded.

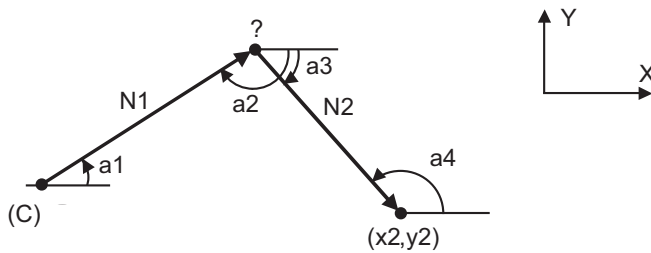
(Note) If the parameter (#1082 Geomet) is set to 0, geometric I will not function.



Command format

N1 G01 Aa1 (A-a2) Ff1;
N2 Xx2 Yy2 A-a4 (A-a3) Ff2;

Aa1, A-a2, A-a3, Aa4	Angle
Ff1, Ff2	Feedrate
Xx2, Yy2	Next block end point coordinates



(C) Current position



Detailed description

- (1) Program error (P396) will occur when the geometric command is not on the selected plane.
- (2) As seen from the + direction of the horizontal axis of the selected plane, the counterclockwise (CCW) direction is considered to be + and the clockwise direction (CW) -.
- (3) The range of slope "a" is $-360.000 \leq a \leq 360.000$.
When a value outside this range is commanded, it will be divided by 360 (degrees) and the remainder will be commanded.
(Example) If 400. is commanded, 40° (remainder of $400/360$) will become the command angle.
- (4) The slope of the line can be commanded on either the start or end point side. Whether the commanded slope is on the start or end point side is identified automatically inside the NC unit.
- (5) The end point coordinates of the second block should be commanded with absolute values. If incremental values are used, program error (P393) will occur.
- (6) The feedrate can be commanded for each block.
- (7) When the angle where the two straight lines intersect is less than 1° , program error (P392) will occur.
- (8) Program error (P396) will occur when the plane is changed in the 1st block and 2nd block.
- (9) This function is ignored when address A is used for the axis name or as the 2nd miscellaneous function.
- (10) Single block stop is possible at the end point of the 1st block.
- (11) Program error (P394) will occur when the 1st and 2nd blocks do not contain the G01 or G33 command.



Relation with other functions

(1) Corner chamfering and corner rounding can be commanded after the angle command in the 1st block.

<p>(Example 1) N1 Aa1 ,Cc1 ; N2 Xx2 Yy2 Aa2 ;</p>	
<p>(Example 2) N1 Aa1 ,Rr1 ; N2 Xx2 Yy2 Aa2 ;</p>	

(2) The geometric command I can be issued after the corner chamfering or corner rounding command.

<p>(Example 3) N1 Xx2 Yy2 ,Cc1 ; N2 Aa1 ; N3 Xx3 Yy3 Aa2 ;</p>	
---	--

(3) The geometric command I can be issued after the linear angle command.

<p>(Example 4) N1 Xx2 Aa1 ; N2 Aa2 ; N3 Xx3 Yy3 Aa3 ;</p>	
--	--

13.10.2 Geometric IB

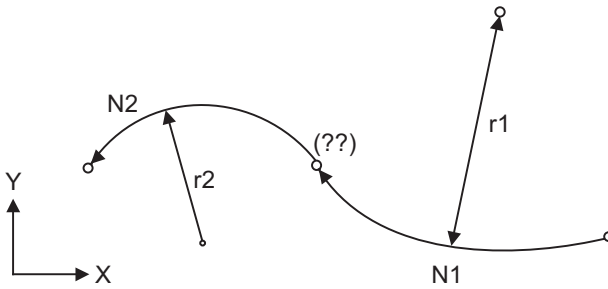


Function and purpose

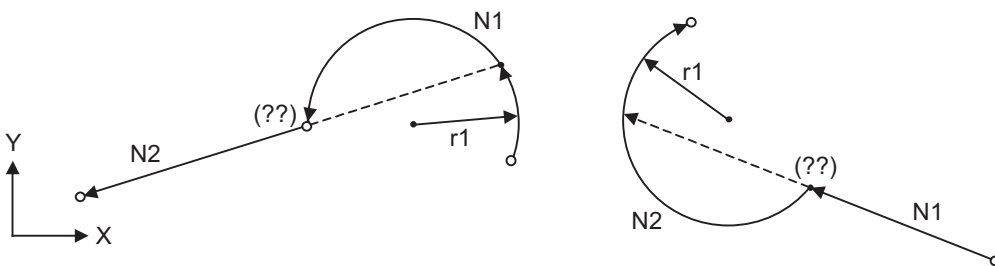
With the geometric IB function, the contact and intersection are calculated by commanding a arc center point or linear angle in the movement commands of two continuous blocks (only blocks with arc commands), instead of commanding the first block end point.

(Note) If the parameter (#1082 Geomet) is not set to 2, geometric IB will not function.

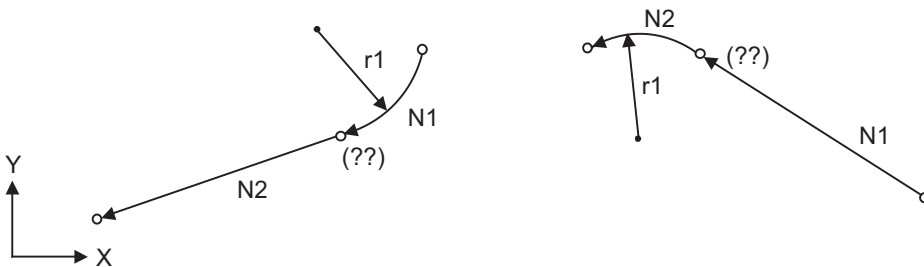
Two-arc contact



Linear - arc (arc - linear) intersection



Linear - arc (arc - linear) contact



13.10.2.1 Geometric IB (Automatic calculation of two-arc contact) ; G02/G03 P_Q_/R_



Function and purpose

When the contact of two continuous contacting arcs is not indicated in the drawing, it can be automatically calculated by commanding the 1st circular center coordinate value or radius, and the 2nd arc end point absolute value and center coordinate value or radius.



Command format

```
N1 G02(G03) Pp1 Qq1 Ff1;
N2 G03(G02) Xx2 Yy2 Pp2 Qq2 Ff2;
```

```
N1 G02(G03) Pp1 Qq1 Ff1;
N2 G03(G02) Xx2 Yy2 Rr2 Ff2;
```

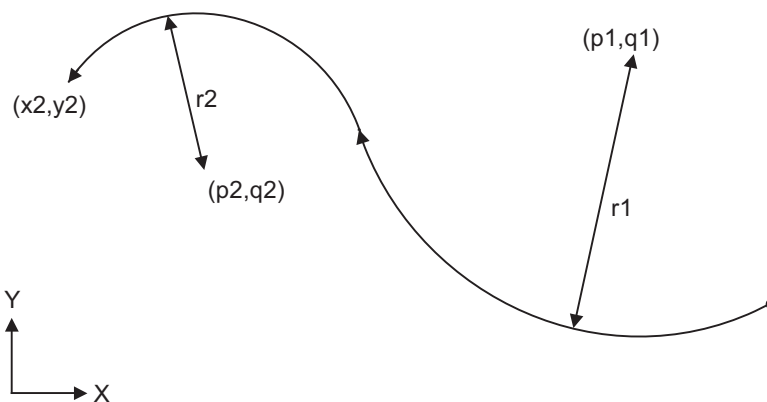
```
N1 G02(G03) Rr1 Ff1;
N2 G03(G02) Xx2 Yy2 Pp2 Qq2 Ff2;
```

P,Q	X and Y axes circular center coordinate absolute value (diameter/radius value command) The center address for the 3rd axis is commanded with A.
R	Arc radius (when a (-) sign is attached, the arc is judged to be 180° or more)

* I and J (X and Y axes arc center coordinate incremental value) commands can be issued instead of P and Q.

1st block arc : Incremental amount from the start point to the center

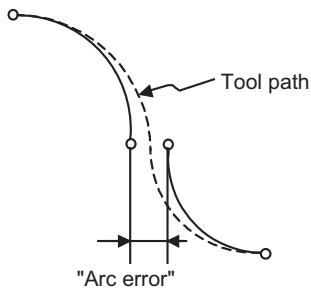
2nd block arc : Incremental amount from the end point to the center



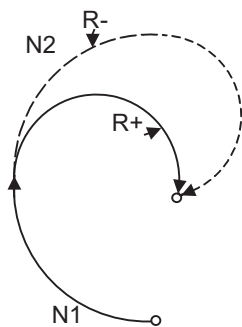


Detailed description

- (1) Program error (P393) will occur before the 1st block if the 2nd block is not a coordinate absolute value command.
- (2) Program error (P398) will occur before the 1st block if there is no geometric IB specification.
- (3) Program error (P395) will occur before the 1st block if there is no R (here, the 1st block is designated with P, Q (I, J)) or P, Q (I, J) designation in the 2nd block.
- (4) Program error (P396) will occur before the 1st block if another plane selection command (G17 to G19) is issued in the 2nd block.
- (5) Program error (P397) will occur before the 1st block if two arcs that do not contact are commanded.
- (6) The contact calculation accuracy is $\pm 1 \mu\text{m}$ (fractions rounded up).
- (7) Single block operation stops at the 1st block.
- (8) When I or J is omitted, the values are regarded as I0 and J0. P and Q cannot be omitted.
- (9) The error range in which the contact is obtained is set in parameter "#1084 RadErr".



- (10) For an arc block perfect circle command (arc block start point = arc block end point), the R designation arc command finishes immediately, and there is no operation. Thus, use a PQ (IJ) designation arc command.
- (11) G codes of the G modal group 1 in the 1st/2nd block can be omitted.
- (12) Addresses being used as axis names cannot be used as command addresses for arc center coordinates or arc radius.
- (13) When the 2nd block arc inscribes the 1st block arc and the 2nd block is an R designation arc, the R+ sign becomes the inward turning arc command, and the R- sign becomes the outward turning arc command.



13.10.2.2 Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03

P_Q_H_



Function and purpose

When the contact point of a shape in which a line and arc contact is not indicated in the drawing, it can be automatically calculated by commanding the following program.



Command format

```
N1 G01 Aa1(A-a2) Ff1 ;
N2 G02(G03) Xx2 Yy2 Pp2 Qq2 Hh2 Ff2 ;
```

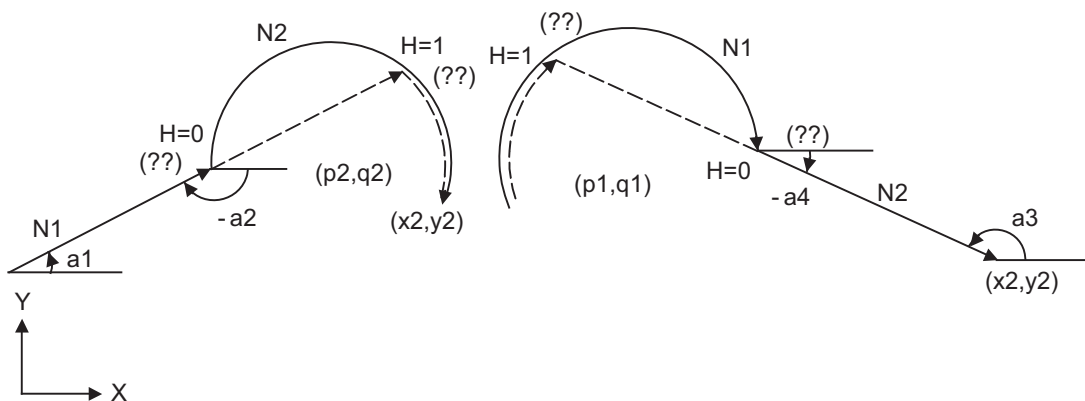
```
N1 G02(G03) Pp1 Qq1 Hh1 (,Hh1) Ff1 ;
N2 G1 Xx2 Yy2 Aa3 (A-a4) Ff2 ;
```

A	Linear angle (-360.000° to 360.000°)
P,Q	X and Y axes circular center coordinate absolute value (diameter/radius value command) The center address for the 3rd axis is commanded with A.
H (,H)	Selection of linear - arc intersection 0: Intersection of the shorter line 1: Intersection of the longer line

* I and J (X and Y axes arc center coordinate incremental value) commands can be issued instead of P and Q.

1st block arc : Incremental amount from the start point to the center

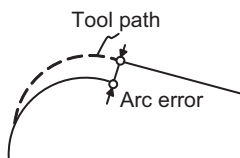
2nd block arc : Incremental amount from the end point to the center



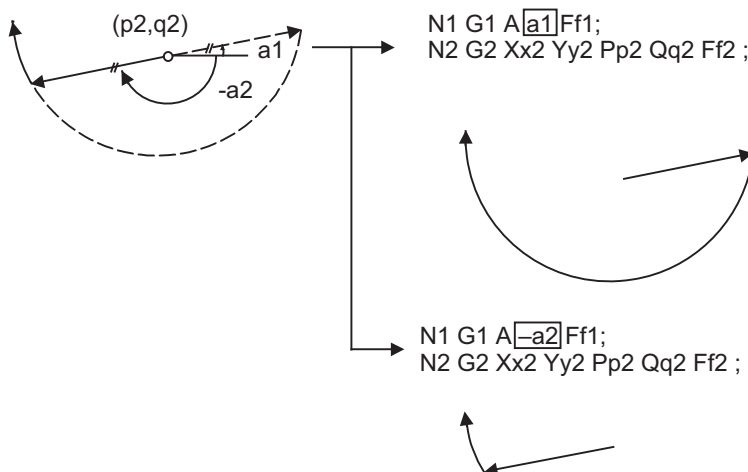


Detailed description

- (1) When the 2nd miscellaneous function address is A, the 2nd miscellaneous function is validated and this function is invalidated.
- (2) Program error (P393) will occur before the 1st block if the 2nd block is not a coordinate absolute value command.
- (3) Program error (P398) will occur before the 1st block if there is no geometric IB specification.
- (4) In case of the 2nd block arc, a program error (P395) will occur before the 1st block if there is no P, Q (I, J) designation. A program error (P395) will also occur if there is no A designation for the line.
- (5) Program error (P396) will occur before the 1st block if another plane selection command (G17 to G19) is issued in the 2nd block.
- (6) Program error (P397) will occur before the 1st block if a straight line and arc that do not contact or intersect are commanded.



- (7) Single block operation stops at the 1st block.
- (8) When I or J is omitted, the values are regarded as I0 and J0. P and Q cannot be omitted.
- (9) When H is omitted, the value is regarded as H0.
- (10) The linear - arc contact is automatically calculated by designating R instead of P, Q (I, J).
- (11) The error range in which the intersect is obtained is set in parameter "#1084 RadErr".
- (12) As seen from the + direction of the horizontal axis of the selected plane, the counterclockwise (CCW) direction is considered to be + and the clockwise direction (CW) -.
- (13) The slope of the line can be commanded on either the start or end point side. Whether designated slope is the starting point or the end point will be automatically identified.
- (14) When the distance to the intersection from the line and arc is same (as in the figure below), the control by address H (short/long distance selection) is invalidated. In this case, the judgment is carried out based on the angle of the line.



- (15) The intersect calculation accuracy is $\pm 1 \mu m$ (fractions rounded up).
- (16) In linear - arc intersections, the arc command can only be PQ (IJ) command. When the arc block start point and arc block end point are the same point, the arc is a perfect circle.
- (17) G codes of the G modal group in the 1st/2nd block can be omitted.
- (18) Addresses being used as axis names cannot be used as command addresses for angles, arc center coordinates or intersection selections.
- (19) When geometric IB is commanded, two blocks are pre-read.



Relation with other functions

Command	Tool path
<p>Geometric IB + corner chamfering</p> <pre>N1 G02 P_Q_H_ ; N2 G01 X_Y_A_.C_ ; G01 X_Y_ ;</pre>	

13.10.2.3 Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 R_H_



Function and purpose

When the intersection of a shape in which a line and arc intersect is not indicated in the drawing, it can be automatically calculated by commanding the following program.

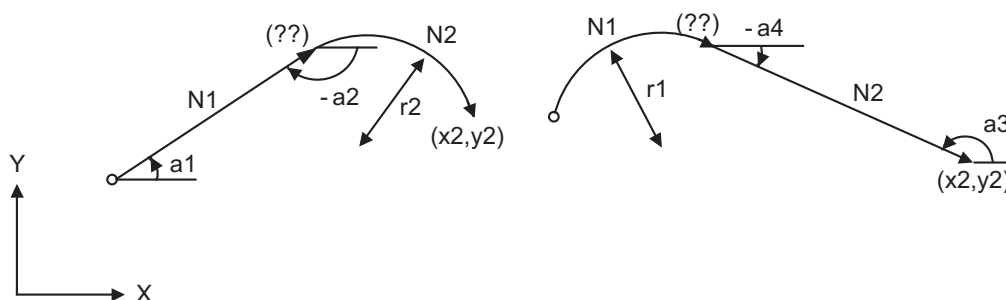


Command format (For G18 plane)

N1 G01 Aa1(A-a2) Ff1;
N2 G03(G02) Xx2 Yy2 Rr2 Ff2;

N1 G03(G02) Rr1 Ff1;
N2 G01 Xx2 Yy2 Aa3(A-a4) Ff2;

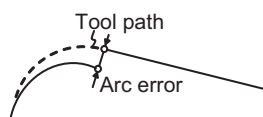
A	Linear angle (-360.000° to 360.000°)
R	Arc radius





Detailed description

- (1) When the 2nd miscellaneous function address is A, the 2nd miscellaneous function is validated and this function is invalidated.
- (2) Program error (P393) will occur before the 1st block if the 2nd block is not a coordinate absolute value command.
- (3) Program error (P398) will occur before the 1st block if there is no geometric IB specification.
- (4) Program error (P396) will occur before the 1st block if another plane selection command (G17 to G19) is issued in the 2nd block.
- (5) A program error (P397) will occur before the 1st block if a straight line and arc that do not contact are commanded.
- (6) In case of the 2nd block arc, a program error (P395) will occur before the 1st block if there is no R designation. A program error (P395) will also occur if there is no A designation for the line.
- (7) Single block operation stops at the 1st block.
- (8) The linear - arc contact is automatically calculated by designating R instead of P, Q (I, J).



- (9) The error range in which the contact is obtained is set in parameter "#1084 RadErr".
- (10) The line slope is the angle to the positive (+) direction of its horizontal axis. Counterclockwise (CCW) is positive (+). Clockwise (CW) is negative (-).
- (11) The slope of the line can be commanded on either the start or end point side. Whether the commanded slope is on the start or end point side is identified automatically inside the NC unit.
- (12) The intersect calculation accuracy is $\pm 1 \mu\text{m}$ (fractions rounded up).
- (13) In linear - arc contact, the arc command can only be an R command. Thus, when the arc block start point = arc block end point, the arc command finishes immediately, and there will be no operation. (Perfect circle command is impossible.)
- (14) G codes of the G modal group 1 in the 1st block can be omitted.
- (15) Addresses being used as axis names cannot be used as command addresses for angles or arc radius.
- (16) When geometric IB is commanded, two blocks are pre-read.



Relation with other functions

Command	Tool path
Geometric IB + corner chamfering N1 G03 R_ ; N2 G01 X_ Y_ A_ ,C_ ; G01 X_ Y_ ;	
Geometric IB + corner rounding II N1 G03 R_ ; N2 G01 X_ Y_ A_ ,R_ ; G01 X_ Y_ ;	

13.11 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)

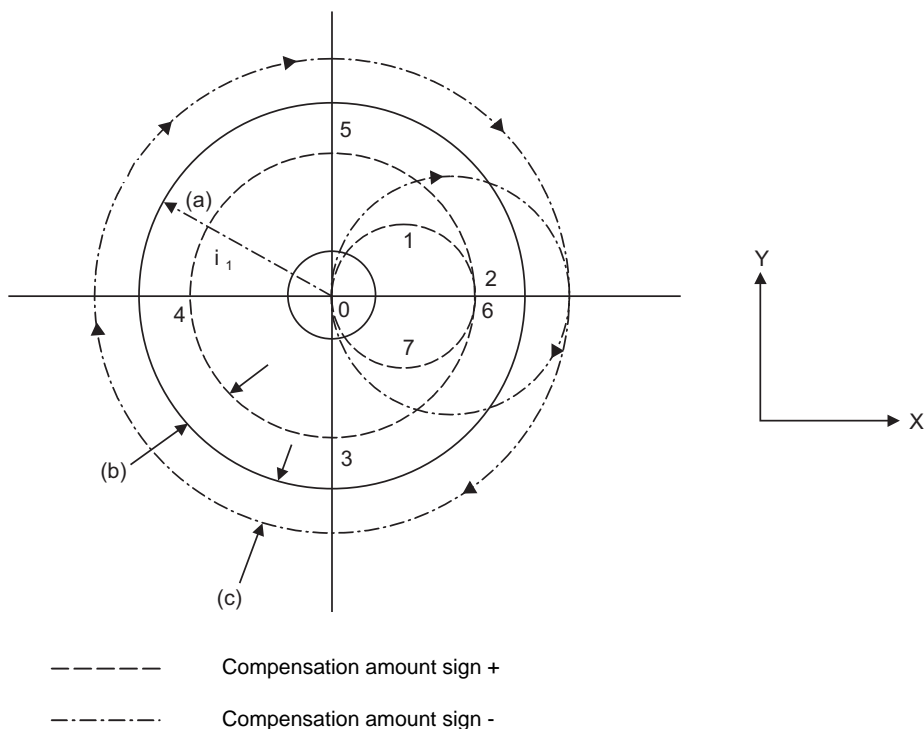
G13 I__ D__ F__ ; ... Circular cutting Counterclockwise (CCW)

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.



(a) Circle radius

(b) d1 offset amount +

(c) d1 offset amount -

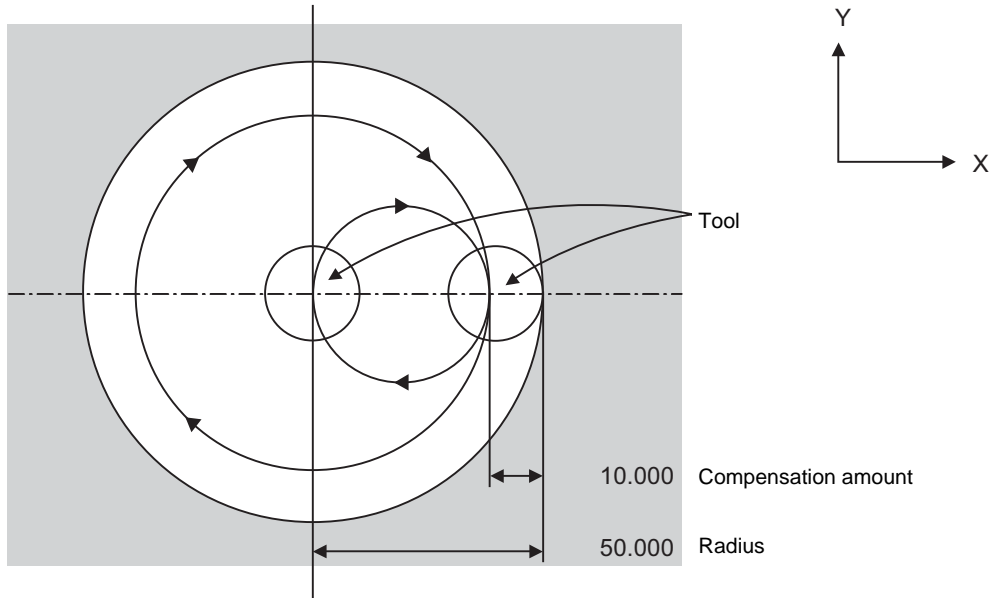
For G12 (tool center path) 0->1->2->3->4->5->6->7->0

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.
But when the parameter "#11034" is set to "1", it operates as follows;
 - (a) Program error will not occur except for an "H" command.
 - (b) Only "D", "F", "I" and "M", "S", "T", "B" will be valid.

13.12 Parameter Input by Program ; G10 L70/L100, G11



Function and purpose

The parameters set from the setting and display unit can be changed in the machining programs.

G10 L70 ... For commanding data with decimal point, and character string data.

The data's command range conforms to the parameter setting range described in Setup Manual.



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	Parameter No.
S	Part system No.
A	Axis No.
H	Data
D	Data
character string	Data

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) Set "1" to the part system No.. It is allowed to omit the address S.
- (Note 3) The axis No. is set in the following manner. "1" for 1st axis, "2" for 2nd axis, and so forth.
If the address A is omitted, the 1st axis will be applied.
As for the parameters common to axes, the command of axis No. will be ignored.
- (Note 4) Address H is commanded with the combination of setting data (0 or 1) and the bit designation □ (0 to 7).
- (Note 5) Only the decimal number can be commanded with the address D.
The value that is smaller than the input setting increment (#1003 iunit) will be round off to the nearest increment.
- (Note 6) The character string must be put in angled brackets "<" and ">".
If these brackets are not provided, the program error (P33) will occur.
Up to 63 characters can be set.
- (Note 7) Command G10 L70, G11 in independent blocks. A program error (P33, P421) will occur if not commanded in independent blocks.
- (Note 8) The following data cannot be changed with the G10 L70 command.
Tool compensation data, workpiece coordinate data, PLC switch, and PLC axis parameter.
- (Note 9) The settings of the parameters with (PR) in the parameter list will be enabled after the power is turned OFF and ON. Refer to the parameter list in your manual.



Program example

(1)When G10 L70 command

G10 L70;	
P6401 H71 ;	Sets "1" to "#6401 bit7".
P8204 S1 A2 D1.234 ;	Sets "1.234" to "#8204 of the 2nd axis".
P8621 <X> ;	Sets "X" to "#8621".
G11 ;	

13.13 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

User macro interruption enable

M96 P__ H__ ;

M96 <File name> H__ ;

P	Interrupt program No. Use a parameter to read out a 4- or 8-digit interrupt program No. starting with O.
<File name>	File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets <>. (The file name can have up to 32 characters including the extension.)
H	Interrupt sequence No.

User macro interruption disable

M97 ;



Detailed description

- (1) 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.
- (2) 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.
- (3) M96 and M97 are processed internally as user macro interrupt control M codes.
- (4) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.
(Example)
When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "M96 P12" command.
- (5) In the following cases, a subprogram of P command value without O No. is called even with a setting to call a subprogram with O No.
 - The digit number of P command value is over the digit number of the program number set with parameter "#8129 subprogram number selection".
 - A subprogram starting with commanded O No. does not exist.

Interrupt enable conditions

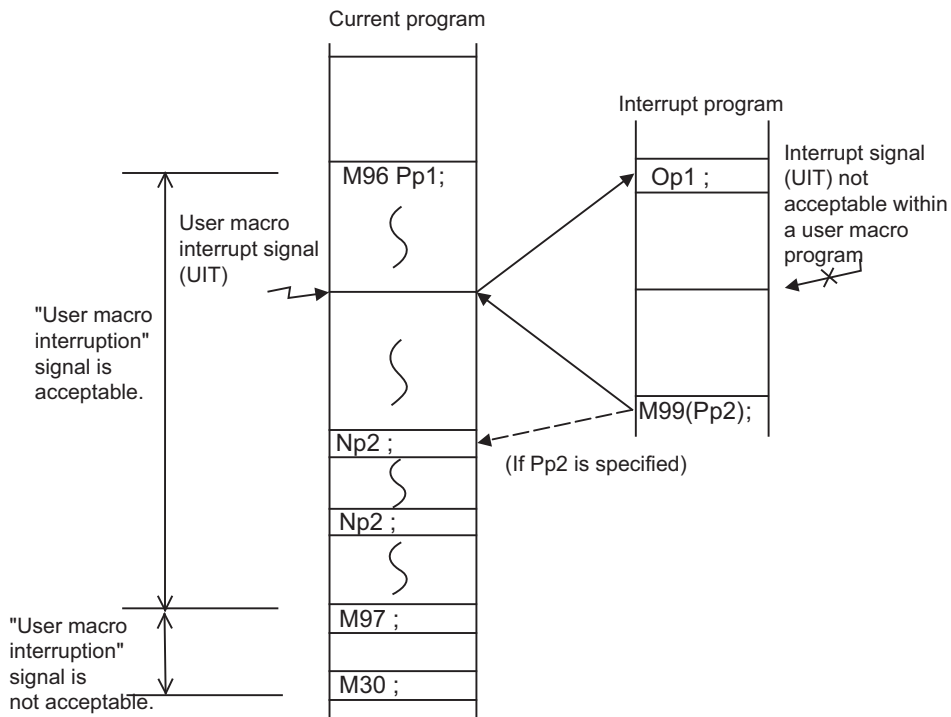
A user macro interruption is enabled only during execution of a program.
The requirements for the user macro interrupt are as follows:

- (1) An automatic 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.

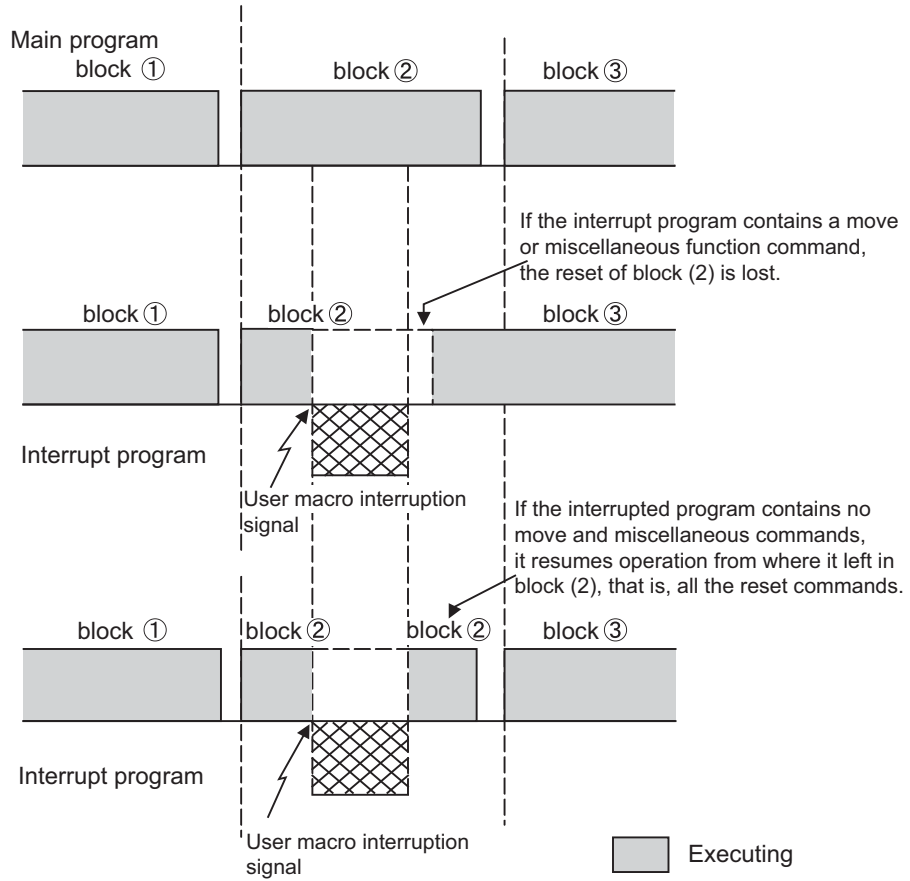
If an interrupt signal (UIT) is input during execution of a miscellaneous function (MSTB) command, the NC system waits for a completion signal (FIN). The system thus executes a move or miscellaneous function command (MSTB) in the interrupt program only after input of FIN.

[Type 2]

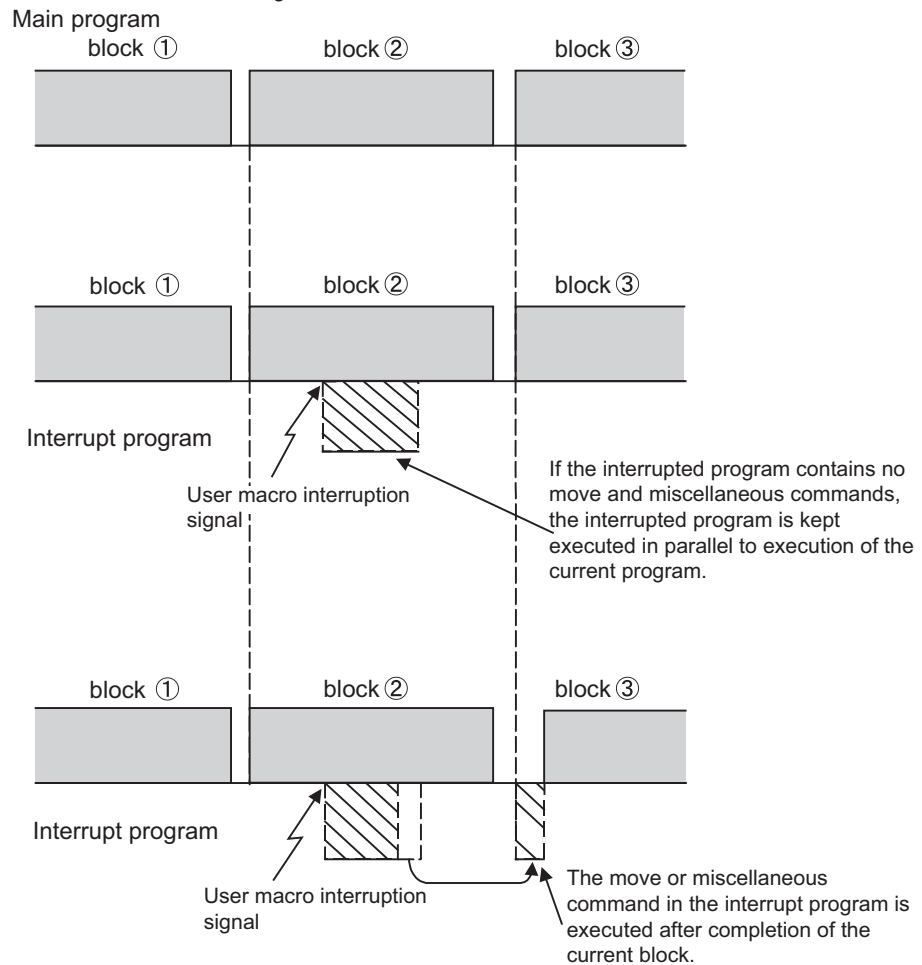
- (1) When an interrupt signal (UIT) is input, the interrupt program will be executed without interrupting execution of the current block in parallel with the executing block.
- (2) If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are completed, then, these commands will be executed.
- (3) 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]



[Type 2]



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 "#8155 Sub-pro interrupt".

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.

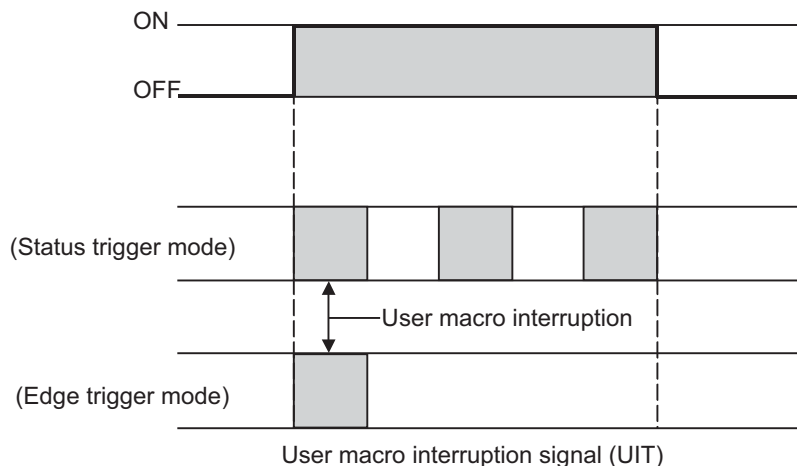
Subprogram type interrupt	The user macro interruption program is called as a subprogram. As with calling by M98, the local variable level remains unchanged before and after an interrupt.
Macro type interruption	The user macro interpretation program is called as a user macro. As with calling by G65, the 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.

User macro interruption signal (UIT)



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.

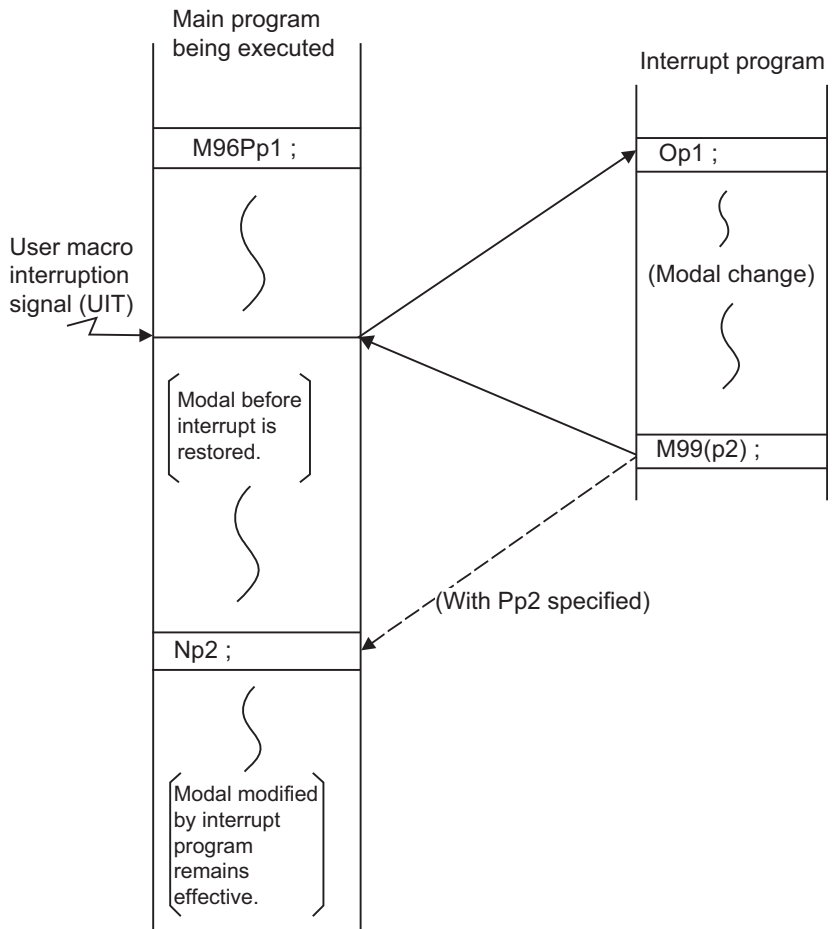
The blocks from the one next to the interrupted block to the last one in the main program are first searched for the block with designated sequence No. If it is not found, all the blocks before the interrupted one are then searched. Control thus returns to the block with sequence No. that is found first 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. (Note 1)	
#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.

(Note 1) The programs are registered as files. When the program No. (file name) is read with #4515, the character string will be converted to a value.

(Example 1)

The file name "123" is the character string 0x31, 0x32, 0x33, so the value will be $(0x31-0x30)*100 + (0x32-0x30)*10 + (0x33-0x30) = 123.0$.

Note that if the file name contains characters other than numbers, it will be "blank".

(Example 2)

If the file name is "123ABC", it contains characters other than numbers, so the result will be "blank".

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

- (1) Subprogram call validity "#8155 Sub-pro interrupt"
 - 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"
- (6) Subprogram number selection "#8129 subprogram number selection"

Select a subprogram number to be called preferentially under subprogram control.

 - 0: Commanded program number
 - 1: 4-digits program number beginning with O No.
 - 2: 8-digits program number beginning with O No.



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.

13.14 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
G30.2	Z axis -> X axis -> Y axis
G30.3	Z axis -> Y axis -> X axis
G30.4	X axis -> Y axis - Z axis
G30.5	Y axis -> X axis - Z axis
G30.6	X axis - Y axis - Z axis

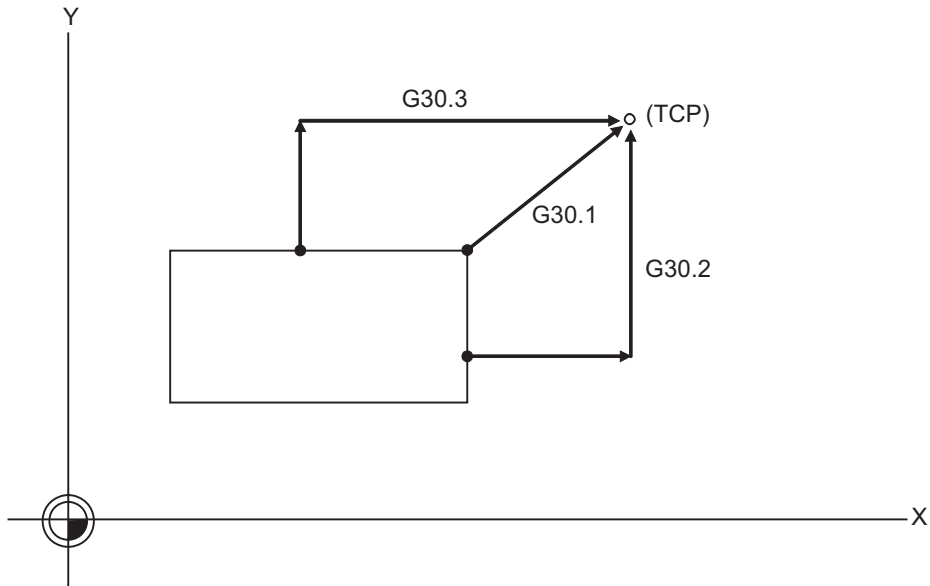
(Note 1) An arrow (->) indicates the order of axes that begin to return. A hyphen (-) indicates that the axes begin to return simultaneously. (Example: "Z axis -> X axis - Y axis" indicates that the Z axis returns to the tool change position, then the X axis and Y axis do at the same time.)

- (1) 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).
- (2) If the axis address is commanded in the same block as the tool change position return command, a program error (P33) will occur.



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.)



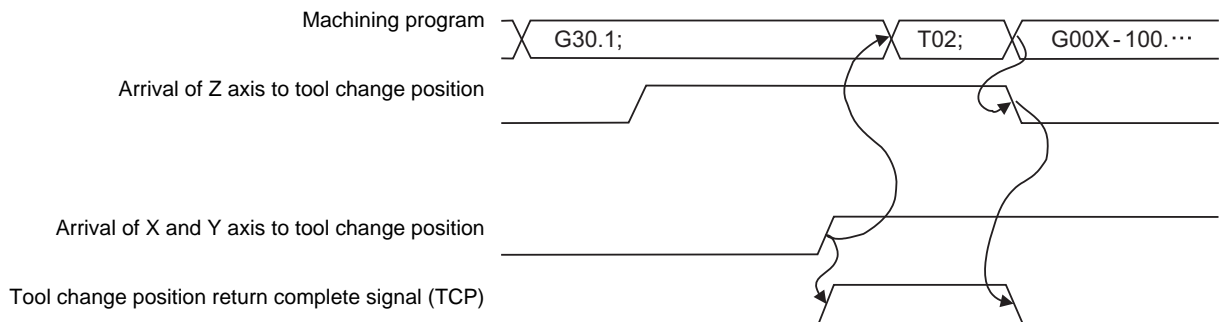
TCP : Tool change position

- (1) G30.1 command: The Z axis returns to the tool changing position, then the X and Y axes simultaneously do the same thing.
- (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.
- (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.
- (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.
- (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.
- (6) G30.6command :The X, Y and Z axes return to the tool changing position simultaneously.

- (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. The TCP signal is then turned OFF when the X or Y axis 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.

13.15 Scaling ; G50/G51



Function and purpose

By multiplying the moving axis command values within the range specified under this command by the factor, the shape commanded by the program can be enlarged or reduced to the desired size.

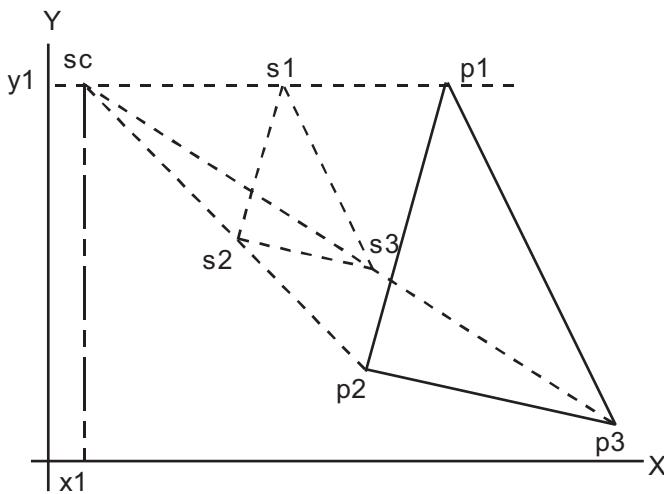


Command format

Scaling ON (set the common scaling factor to the three basic axes)

G51 X__ Y__ Z__ P__ ;

X,Y,Z	Scaling center coordinates
P	Scaling factor



sc : Scaling center
 p1,p2,p3: Program shape
 s1,s2,s3: Shape after scaling

Scaling ON (When setting the scaling factor to each of the three basic axes)

G51 X__ Y__ Z__ I__ J__ K__ ;

X,Y,Z	Scaling center coordinates
I	Scaling factor of basic 1st axis
J	Scaling factor of basic 2nd axis
K	Scaling factor of basic 3rd axis

Scaling cancel

G50 ;



Detailed description

Specifying the scaling axis, scaling center and its factor

Commanding G51 selects the scaling mode. The G51 command only specifies the scaling axis, its center and factor, and does not move the axis.

Though the scaling mode is selected by the G51 command, the axis actually valid for scaling is the axis where the scaling center has been specified.

- (1) Scaling center
 - Specify the scaling center in accordance with the then absolute/incremental mode (G90/G91).
 - The scaling center must be specified also when the current position is defined as a center.
 - As described above, the axis valid for scaling is only the axis whose center has been specified.
- (2) Scaling factor
 - Use the address P or I, J, K to specify the scaling factor.
 - Minimum command unit : 0.000001
 - Command range: Both -99999999 to 99999999 (-99.999999 to 99.999999 times) and -99.999999 to 99.999999 is valid, but the decimal point command is valid only after the G51 command.
 - When the factor is not specified in the same block as G51, the factor set with the parameter "#8072 SCALING P" is used.
 - When the address P and the address I, J, K are commanded in a same block, a factor specified by the address I, J, K is applied for the basic three axes. And a factor specified by the address P is applied for other axes.
 - If changed during the scaling mode, the value of this parameter will not become valid. Scaling is performed with the setting value that was used when G51 was commanded.
 - When the factor is not specified in either the program nor parameter, it is calculated as 1.
- (3) A program error will occur in the following cases.
 - Scaling was commanded though there was no scaling specification. (P350)
 - The upper limit of the factor command range was exceeded in the same block as G51. (P35)
 (When using the machining parameter scaling factor, the factor is calculated as 1, when $-0.000001 < \text{factor} < 0.000001$, or the factor is more than 99.999999 or less than -99.999999.)

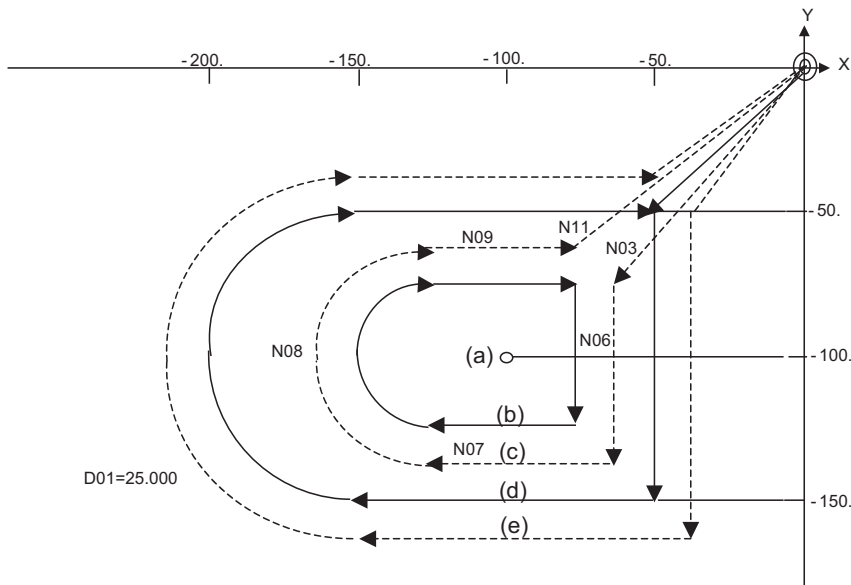
Scaling cancel

When G50 is commanded, scaling is canceled.



Program example

(Example 1)



- (a) Scaling center
- (b) Program path after 1/2 scaling
- (c) Tool path after 1/2 scaling
- (d) Program path when scaling is not applied
- (e) Tool path when scaling is not applied

<Program>

```

N01 G92 X0 Y0 Z0;
N02 G90 G51 X-100. Y-100. P0.5;
N03 G00 G43 Z-200. H02;
N04 G41 X-50. Y-50. D01;
N05 G01 Z-250. F1000;
N06 Y-150. F200;
N07 X-150.;
N08 G02 Y-50. J50.;
N09 G01 X-50.;
N10 G00 G49 Z0;
N11 G40 G50 X0 Y0;
N12 M02;
    
```



Relation with other functions

- (1) G27 reference position check command
When G27 is commanded during scaling, scaling is canceled at completion of the command.
- (2) Reference position return command (G28, G29, G30)
When the G28, G30 or reference position return command is issued during scaling, scaling is canceled at the midpoint and the axis returns to the reference position. When the midpoint is to be ignored, the axis returns to the reference point directly.
When G29 is commanded during scaling, scaling is applied to the movement after the midpoint.
- (3) G60 (unidirectional positioning) command
If the G60 (unidirectional positioning) command is given during scaling, scaling is applied to the final positioning point and is not applied to the creep amount. Namely, the creep amount is uniform regardless of scaling.
- (4) Workpiece coordinate system switching
When the workpiece coordinate system is switched during scaling, the scaling center is shifted by the difference between the offset amounts of the new and old workpiece coordinate systems.
- (5) During coordinate rotation
When scaling is commanded during coordinate rotation, the scaling center rotates. Scaling is executed at that rotated scaling center.
- (6) G51 command
When the G51 command is issued during the scaling mode, the axis whose center was newly specified is also made valid for scaling. Also, the factor under the latest G51 command is made valid.



Precautions

- (1) Scaling is not applied to the compensation amounts of tool radius compensation, tool length compensation and the like. (Compensation is calculated for the shape after scaling.)
- (2) Scaling is valid for only the movement command in automatic operation. It is invalid for manual movement.
- (3) For X, Y and Z, scaling is valid for only the specified axes and is not applied to unspecified axes.
- (4) When an arc is commanded and scaling is valid for one of the two axes configuring the arc plane, a program error (P70) will occur.
- (5) When M02 or M30 is commanded, or when NC reset is carried out during the scaling mode, the mode switches to a cancel mode.
- (6) When the coordinate system is shifted (G92, G52 command) during scaling, the scaling center is also shifted by the difference amount.
- (7) If manual interruption is made during scaling, manual ABS selection is ignored for the movement followed by an incremental value command and operation performed is the same as in manual ABS OFF.

13.16 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.



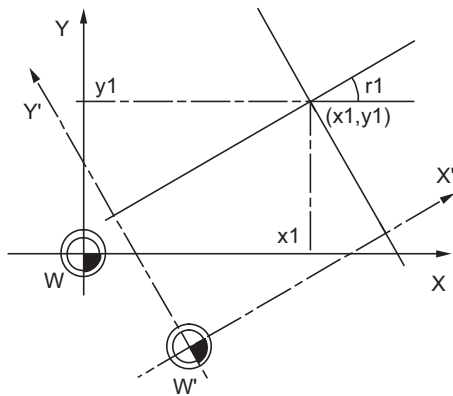
Command format

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

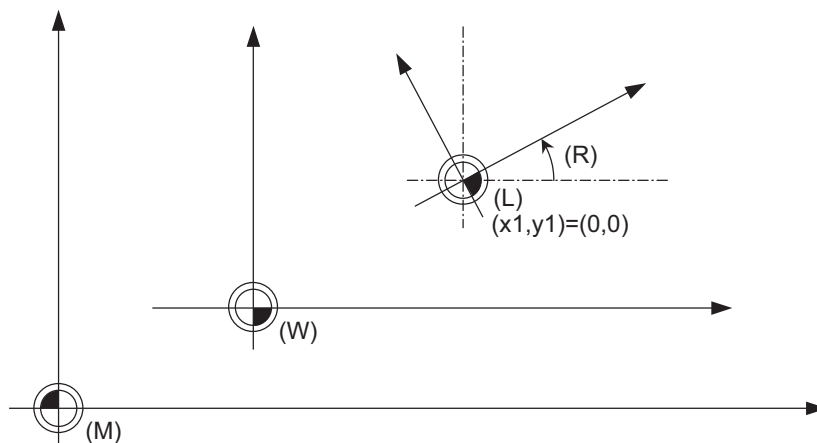
r1 : Rotation angle

(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.

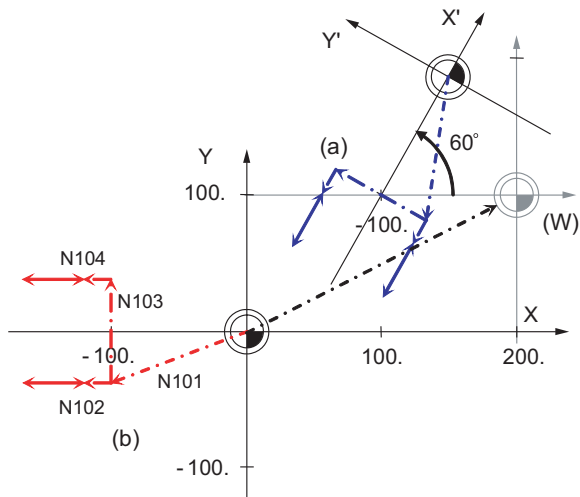


Program example

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. Q-10. F100 ;
N103 G00 Y40. ;
N104 G83 X-150. R-20. Q-10. 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

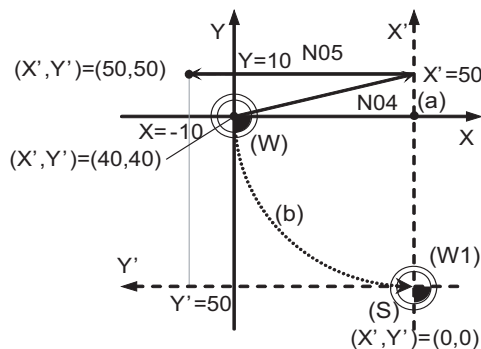
Command basically two axes in the rotation plane by an absolute value immediately after the coordinate rotation command.

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
    
```

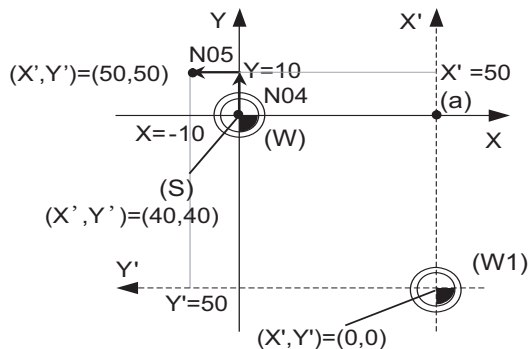


- Machine movement path
- (a) Rotation center
- (W) Local coordinate system before rotation
- (S) Start point
- (b) The start point is rotated virtually
- (W1) Local coordinate system after rotation

- (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
- (a) Rotation center
- (W) Local coordinate system before rotation
- (S) Start point
- (W1) 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
N08 M02 ;                 End
    
```

W : Workpiece coordinate system
L : Local coordinate system

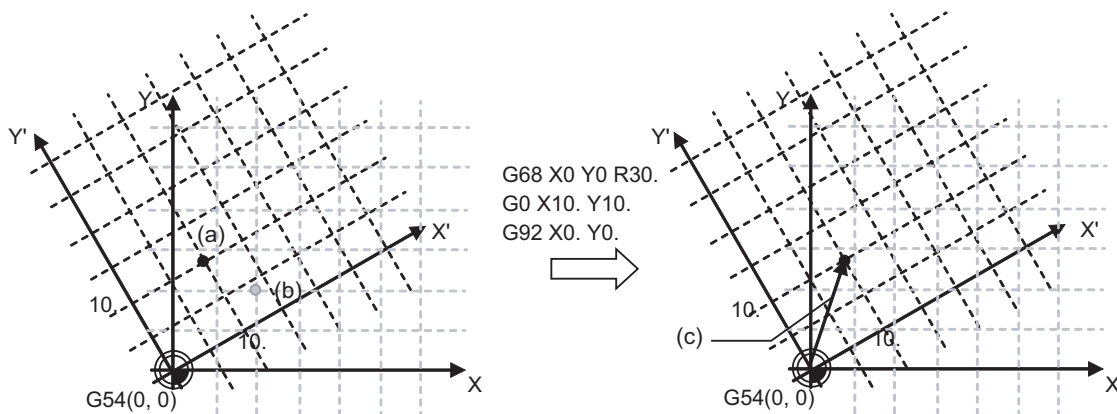
	(1) Operation of #19003 = 0	(2) Operation of #19003 = 1
N03	<p>(Workpiece coordinate system is rotated virtually.)</p>	
	Workpiece coordinate system is rotated virtually.	Workpiece coordinate system is not rotated.
N04	<p>(X, Y) = (10, 10) (Local coordinate designation) (X'', Y'') = (0, 0)</p>	<p>(Local coordinate designation) (Rotation center) (X, Y) = (10, 10) Y = -10</p>
	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	<p>(X'', Y'') = (20, 30) Start point: (X'', Y'') = (10, 30) Y = 10</p>	<p>(Rotation center) (X'', Y'') = (20, -10) Y = 10 Start point: (X'', Y'') = (-10, -10) X = 40</p> <p>(The start point is rotated virtually.)</p>
	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	<p>(X'', Y'') = (20, 10) Y = 10 X = 30 (Local coordinate system)</p>	<p>(X'', Y'') = (20, 10) Y = 10 X = 20 X = 40 (X'', Y'') = (20, -10)</p>

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



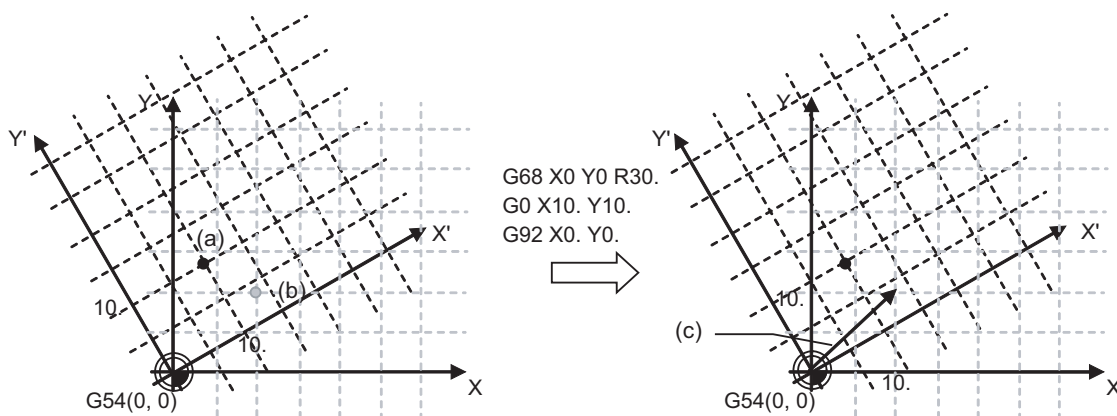
(a) Position after rotation

(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.



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 GOIntp" = 1) is set, each axis will move independently at the respective rapid traverse rates. If the axis must be moved linearly (interpolated) from the start point to the end point (such as during the hole machining cycle), always turn G00 non-interpolation OFF (parameter "#1086 GOIntp" = 0). The feedrate in this case is the composite speed of each axis' rapid traverse rate, so the movement speed will be faster than when moving only one axis (before coordinate rotation).
- (7) If the coordinate rotation specifications are not provided, a program error (P260) will occur when coordinate rotation is commanded.
- (8) The 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.

Coordinate System Setting Functions

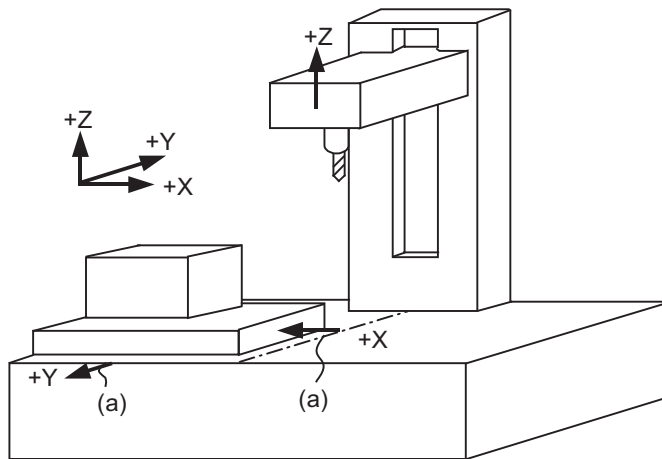
14.1 Coordinate Words and Control Axes



Function and purpose

In the standard specifications, there are 3 control axes. 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

14.2 Basic Machine, Workpiece and Local Coordinate Systems



Function and purpose

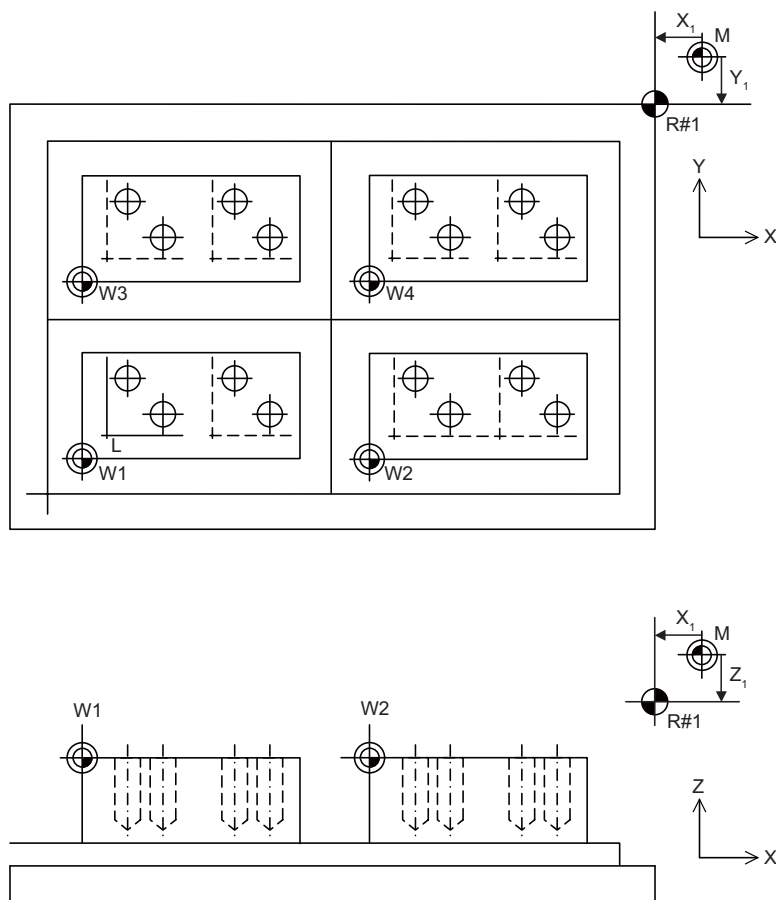
The basic machine coordinate system is fixed in the machine and it denotes that position which is determined inherently by the machine.

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

The local coordinate systems (G52) are valid on the coordinate systems designated by workpiece coordinate systems 1 to 6.

The hypothetical machine coordinate system can be set on the basic machine coordinate system using a G92 command. At this time, the workpiece coordinate system 1 to 6 is also simultaneously shifted.

Also refer to "Coordinate Systems and Coordinate Zero Point symbols".

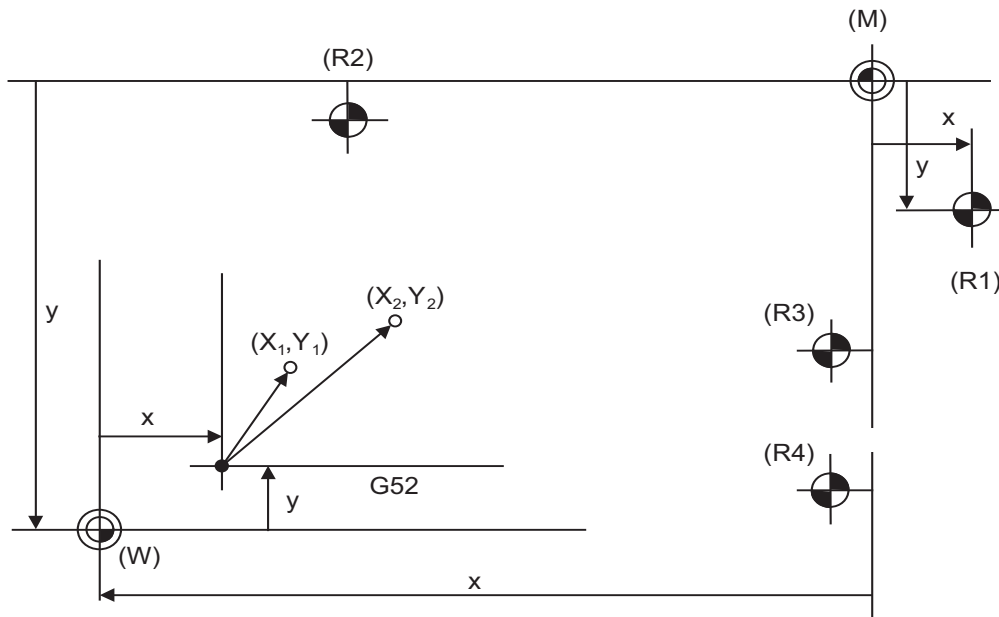
14.3 Machine Zero Point and 2nd, 3rd, 4th Reference Position (Zero point)



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

(R3) 3rd reference position

(R4) 4th reference position

(W) Workpiece coordinate systems (G54 to G59)

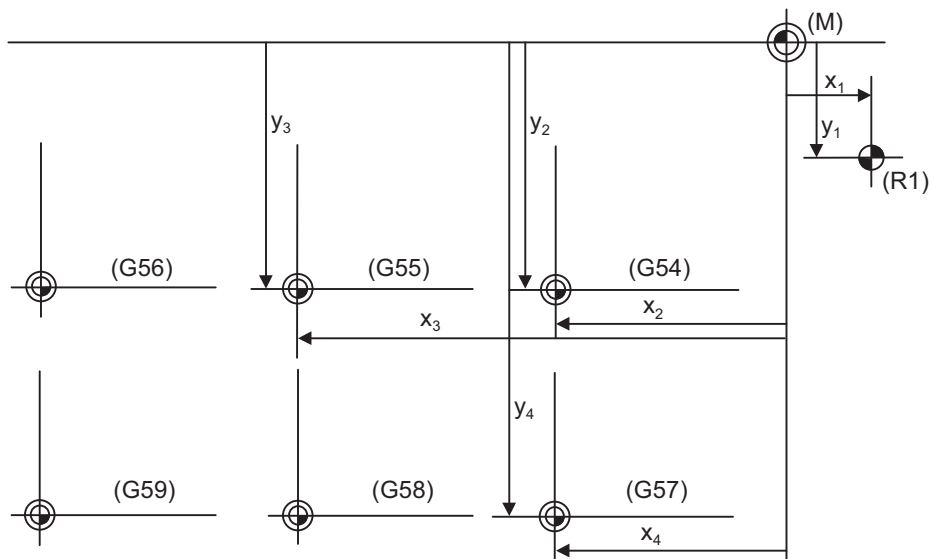
14.4 Automatic Coordinate System Setting



Function and purpose

This function creates each coordinate system according to the parameter values input beforehand from the setting and display unit when the first manual reference position return or the reference position is reached with the 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.



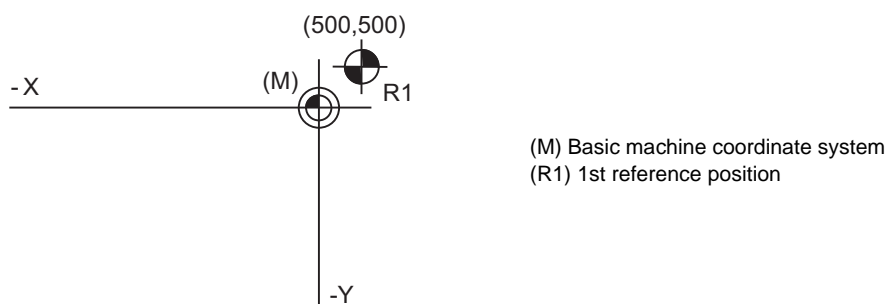
Command format

```
(G90)G53 X__ Y__ Z__ ;
```



Detailed description

- (1) When the power is switched on, the basic machine coordinate system is automatically set as referenced to the reference (zero) position return position, which is determined by the automatic or manual reference (zero) position return.
- (2) The basic machine coordinate system is not changed by the G92 command.
- (3) The G53 command is valid only in the block in which it has been designated.
- (4) In the incremental value command mode (G91), the G53 command provides movement with the incremental value in the coordinate system being selected.
- (5) Even if G53 is commanded, the tool radius compensation amount for the commanded axis will not be canceled.
- (6) The 1st reference position coordinate value indicates the distance from the basic machine coordinate system 0 point to the reference position (zero point) return position.
- (7) The G53 command will move with cutting feedrate or rapid traverse following command modal.
- (8) If the G53 command and G28 command (reference position return) are issued in the same block, the command issued last will be valid.



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.



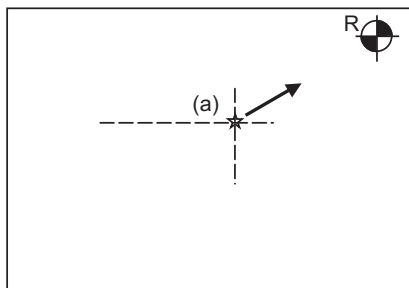
Command format

```
G92 X__ Y__ Z__ α__ ;
```

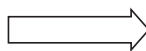


Detailed description

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)

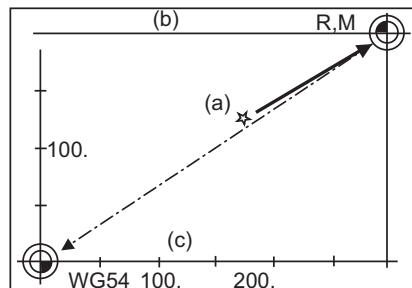


Reference position return completed



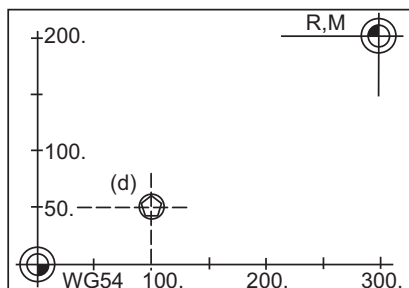
The basic machine coordinate system and workpiece coordinate system are created at the preset position.

- (a) Power ON position
- (b) Basic machine coordinate system
- (c) Workpiece coordinate system

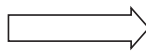


[Position]	[Workpiece]
X 0.000	X 300.000
Y 0.000	Y 200.000

(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.

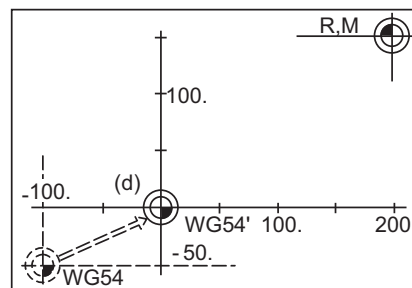


Coordinate system setting



For example, if G92X 0 Y 0; is commanded, the workpiece coordinate system will be newly created.

(d) Tool position



[Position]	[Workpiece]
X -200.000	X 100.000
Y -150.000	Y 50.000

[Position]	[Workpiece]
X 0.000	X 0.000
Y 0.000	Y 0.000

(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.



Precautions

- (1) When the parameter "#1279 ext15/bit5" is set to "1", the shift amount set with the coordinate system setting (G92) will be cleared when the axis reaches to the manual reference position.

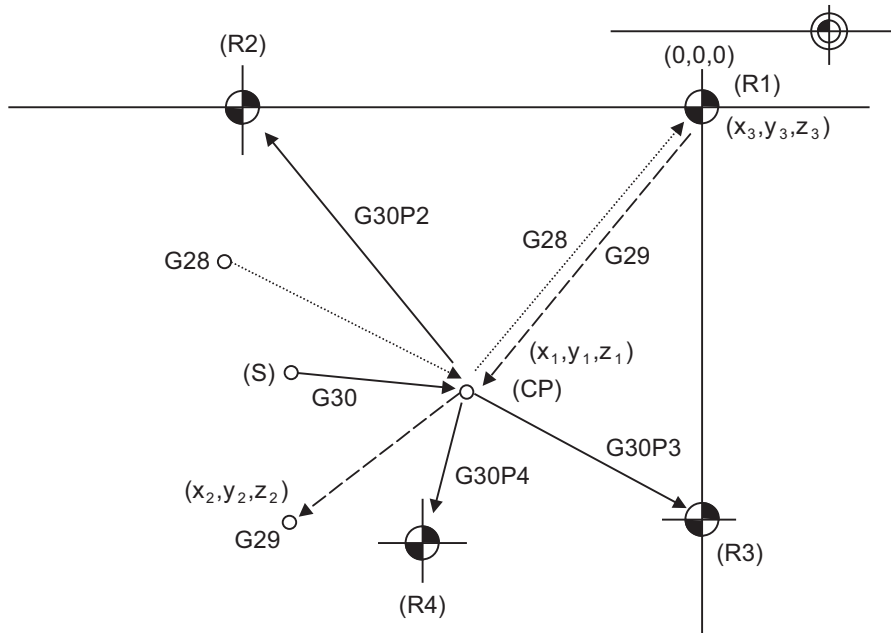
14.7 Reference Position (Zero point) Return ; G28,G29



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
- (S) Start point
- (R1) 1st reference position
- (R2) 2nd reference position
- (R3) 3rd reference position
- (R4) 4th reference position



Command format

G28 Xx1 Yy1 Zz1; ... Automatic reference position return	
X, Y, Z	Coordinate value of the intermediate point
G29 Xx2 Yy2 Zz2; ... Start point return	
X, Y, Z	Coordinate value of the end point



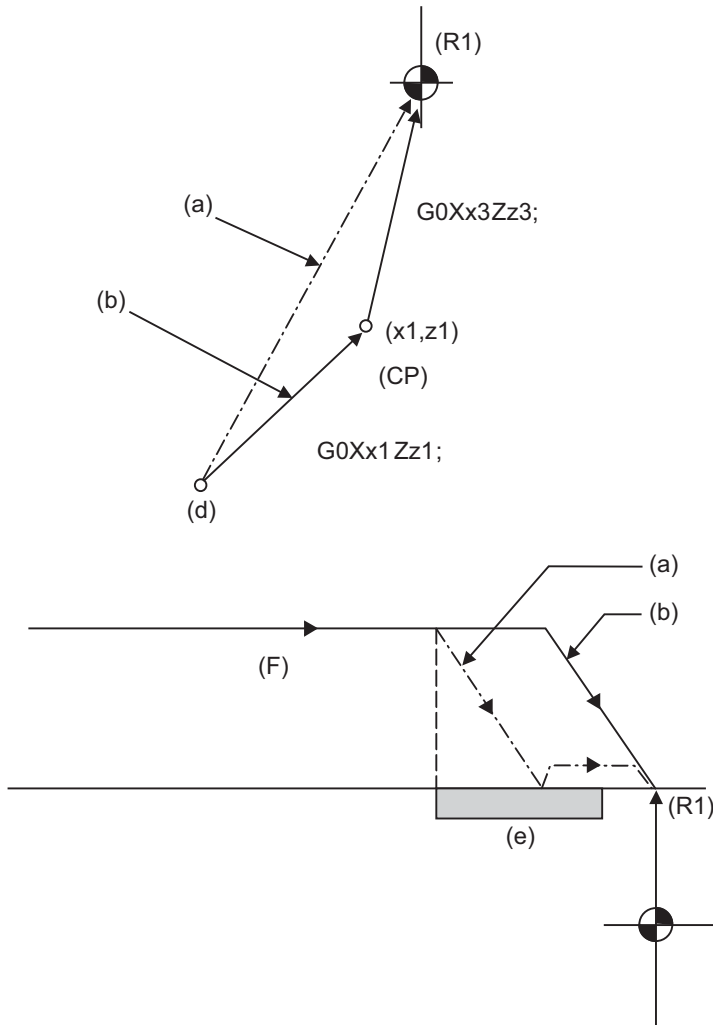
Detailed description

- (1) The G28 command is equivalent to the following:
 G00 Xx1 Yy1 Zz1;
 G00 Xx3 Yy3 Zz3;
 In this case, Xx3, Yy3 and Zz3 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 subsequent returns, the return is made at high speed to the reference (zero) position which was stored at the first time and the direction is not checked.
- (3) When reference 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;
 G00 Xx2 Yy2 Zz2;
 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 and z1 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) of the positioning point are assigned by the position command modal. (G90, G91).
- (8) G29 is valid for either G28 or G30 but the commanded axes are positioned after a return has been made to the latest intermediate point.
- (9) The tool compensation will be canceled during reference position return unless it is already canceled, and the compensation amount will be cleared.
- (10) The intermediate point can be ignored by parameter "#1091 Ignore middle point" setting.
- (11) Control from the intermediate point to the reference position is ignored for reference position return in the machine lock status. The next block is executed when the commanded axis reaches as far as the intermediate point.
- (12) Mirror image is valid from the start point to the intermediate point during reference position 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 position and the tool will move to the reference position.
- (13) When G28/G29/G30 is commanded in single block mode, if "#1279 ext15/bit6 Enable single block stop at middle point" is set to "1", single block stop at middle point will be performed; single block stop at middle point will not be performed if set to "0".
- (14) If the mode is switched to MDI mode or reference point return mode while in a single block stop at middle point, "M01 Operation Error 0013" occurs.
- (15) If the NC is reset while in a single block stop at middle point, the interim position for G29 start position return will not be updated.
- (16) If a miscellaneous function is commanded in the same block as G28/G29, the miscellaneous function completion waiting point will be the end of commanded movement, instead of the interim position.



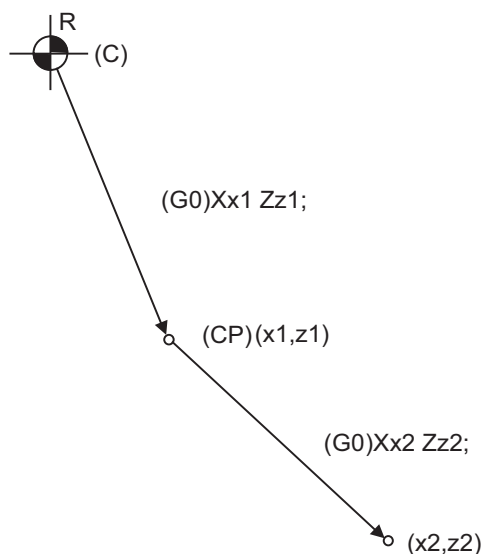
Program example

(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)

:

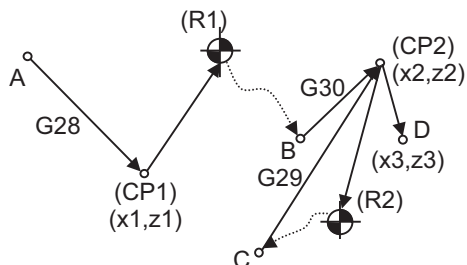
G30 Xx2 Zz2 ;

: (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)

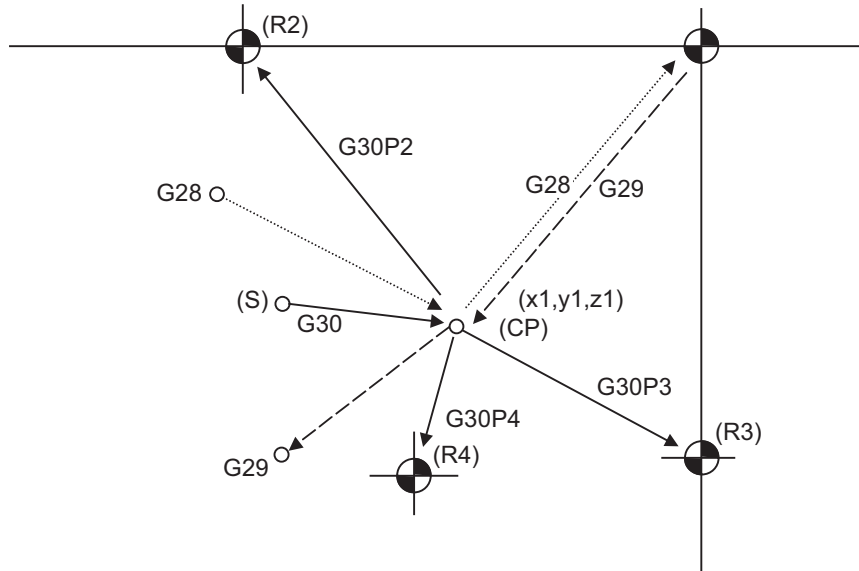
(R2) 2nd reference position (#2)

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

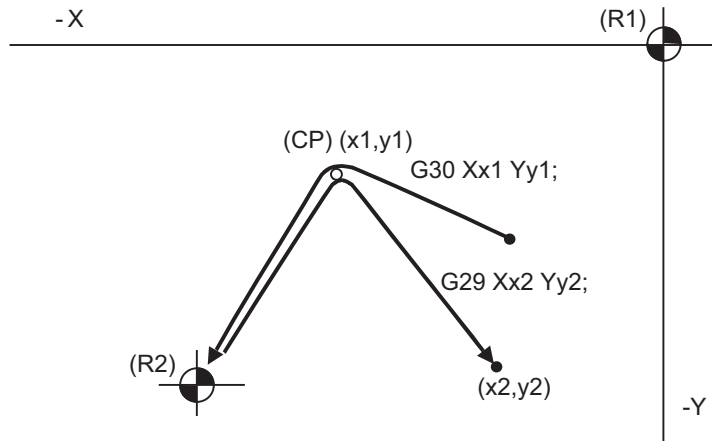
G30 P2(P3,P4)Xx1 Yy1 Zz1;

X, Y, Z	Coordinate value of the intermediate point
P	Reference position No. P2: 2nd reference position return P3: 3rd reference position return P4: 4th reference position return



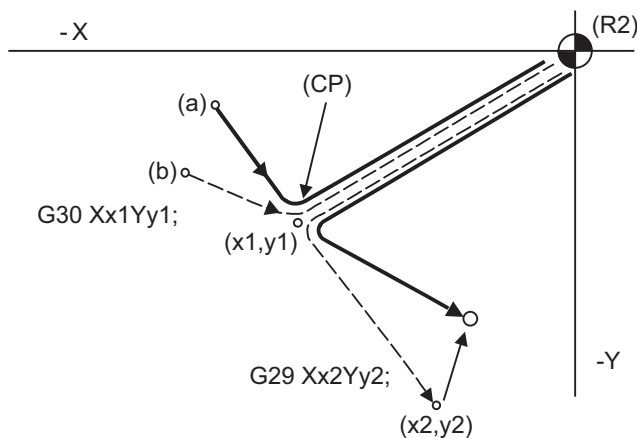
Detailed description

- (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.



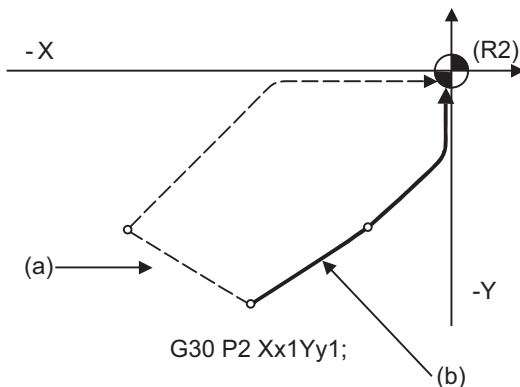
(CP) Intermediate point
 (R1) 1st reference position
 (R2) 2nd reference position

- (5) With reference position return on a plane during compensation, the tool moves without tool radius compensation 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.



(CP) Intermediate point
 (a) Tool nose center path
 (R1) 1st reference position
 (b) Program path
 (R2) 2nd reference position

- (6) The tool length compensation amount for the axis involved is canceled after the 2nd, 3rd and 4th reference position return.
- (7) With second, third and fourth reference point returns in the machine lock status, control from the intermediate point to the reference 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

- (9) If the 2nd, 3rd or 4th reference position is changed while G30 zero point return operation is in pause due to an interlock, "M01 Operation Error" occurs.
- (10) When G28/G29/G30 is commanded in single block mode, if "#1279 ext15/bit6 Enable single block stop at middle point" is set to "1", single block stop at middle point will be performed; single block stop at middle point will not be performed if set to "0".
- (11) If the mode is switched to MDI mode or reference point return mode while in a single block stop at middle point, "M01 Operation Error 0013" occurs.
- (12) If the NC is reset while in a single block stop at middle point, the interim position for G29 start position return will not be updated.
- (13) If a miscellaneous function is commanded in the same block as G30, the miscellaneous function completion waiting point will be the end of commanded movement, instead of the interim 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).
- (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 compensation amounts for tool radius and tool length.)
- (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 compensation amounts for tool radius and tool length.)
(G54,G92)



Command format

(G90) G54 to G59 ... Workpiece coordinate system selection

(G54 to G59) G92 X__ Y__ Z__ ; ... Workpiece coordinate system setting

G54.1 Pn ; ... Workpiece coordinate system selection (P1 to P48)

**G54.1 Pn ;
G92 X__ Y__ Z__ ; ... Workpiece coordinate system setting (P1 to P48)**

G10 L20 Pn X__ Y__ Z__ ; ... Workpiece coordinate system offset amount setting (P1 to P48)

When the designated extended workpiece coordinate system offset amount is rewritten

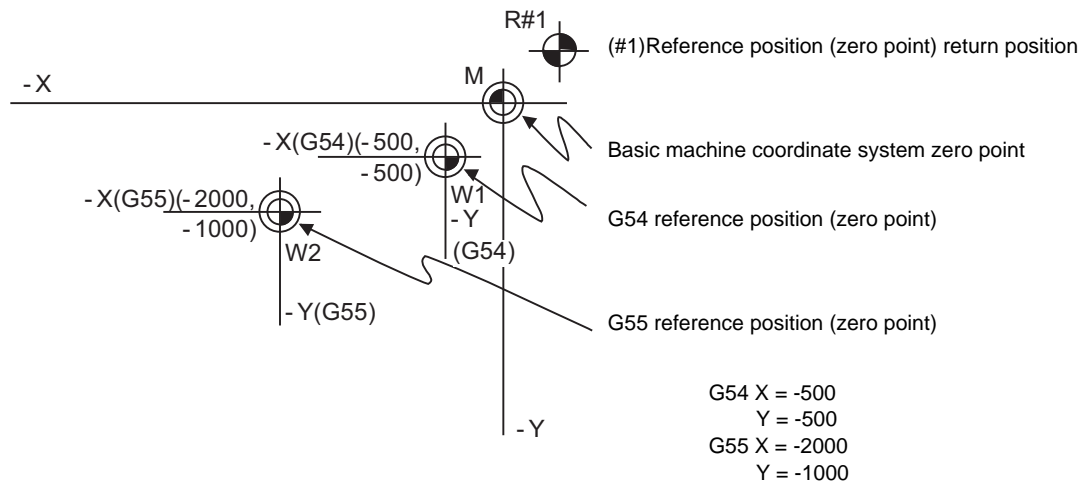
G10 G54.1 Pn X__ Y__ Z__ ; ... Workpiece coordinate system offset amount setting (P1 to P48)

When the extended workpiece coordinate system is selected, and the offset amount is rewritten



Detailed description

- (1) With any of the G54 through G59 commands or G54.1P1 through G54.1P48 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 and G54.1P1 through G54.1P48 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.



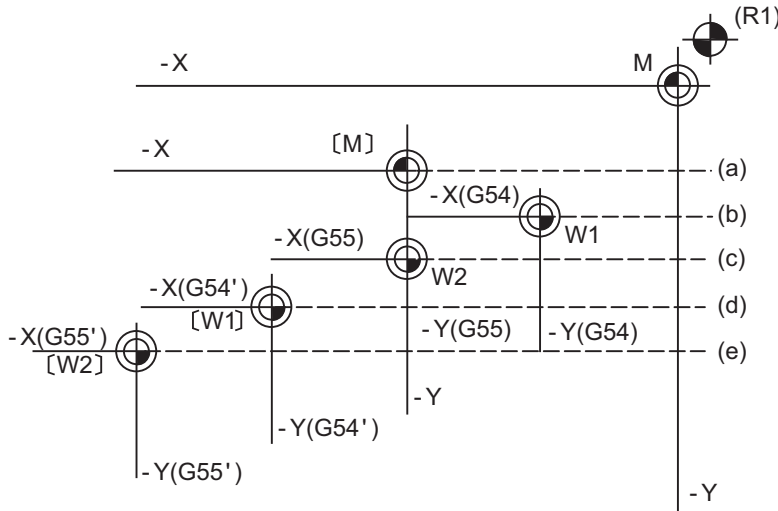
- (6) The offset settings of workpiece coordinate systems can be changed any number of times. (They can also be changed by G10 L2 Pp1 Xx1 Yy1 Zz1.)

[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 ;	Set the offset amount in the designated coordinate system No. by P code. When the currently selected coordinate system is G54 to G59, P1 to P6 corresponds to G54 to G59 respectively. When the external coordinate system is selected, P No. corresponds to G54.1 P1 to P48. If other values are set, the program error (P35) will occur.
G10 Xx Yy Zz ;	Set the offset amount in the currently selected coordinate system.
G10 G54.1 Xx Yy Zz ;	When there is no P code in the same block as G54.1, the program error (P33) will occur.

- (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 or G10L20. If commanded, the P code will be used by the G code with the higher priority or the program error (P33) will occur.
- (13) The local coordinate system cannot be used during G54.1 modal. The program error (P438) will occur when the G52 command is executed during G54.1 modal.
- (14) 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.

- (15) 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 nth axis		1st axis to nth axis
P1	#7001 to #700n	P25	#7481 to #748n
P2	#7021 to #702n	P26	#7501 to #750n
P3	#7041 to #704n	P27	#7521 to #752n
P4	#7061 to #706n	P28	#7541 to #754n
P5	#7081 to #708n	P29	#7561 to #756n
P6	#7101 to #710n	P30	#7581 to #758n
P7	#7121 to #712n	P31	#7601 to #760n
P8	#7141 to #714n	P32	#7621 to #762n
P9	#7161 to #716n	P33	#7641 to #764n
P10	#7181 to #718n	P34	#7661 to #766n
P11	#7201 to #720n	P35	#7681 to #768n
P12	#7221 to #722n	P36	#7701 to #770n
P13	#7241 to #724n	P37	#7721 to #772n
P14	#7261 to #726n	P38	#7741 to #774n
P15	#7281 to #728n	P39	#7761 to #776n
P16	#7301 to #730n	P40	#7781 to #778n
P17	#7321 to #732n	P41	#7801 to #780n
P18	#7341 to #734n	P42	#7821 to #782n
P19	#7361 to #736n	P43	#7841 to #784n
P20	#7381 to #738n	P44	#7861 to #786n
P21	#7401 to #740n	P45	#7881 to #788n
P22	#7421 to #742n	P46	#7901 to #790n
P23	#7441 to #744n	P47	#7921 to #792n
P24	#7461 to #746n	P48	#7941 to #794n

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.

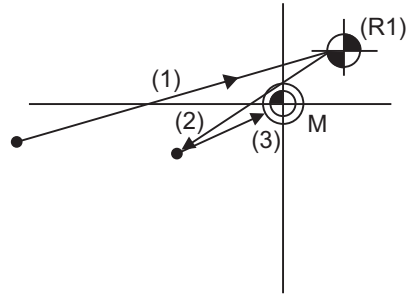
- (16) When the parameter "#1151 Reset ini" is OFF, the modal of G54.1 command will be retained even if the reset 1 is carried out.
- (17) P address (coordinate system selection) of G54.1 cannot be commanded alone even in G54.1 modal. Even if commanded, the designated extended workpiece coordinate system cannot be selected.
(Ex)
P54.1 P5 ; Changed to P5 workpiece coordinate system.
P3 ; Ignored.
G92 X0 Y0 Z0 ; The current position becomes the zero point of P5 workpiece coordinate system.
- (18) When G92 is commanded in the extended workpiece coordinate system, the coordinate system will be sifted.



Program example

(Example 1)

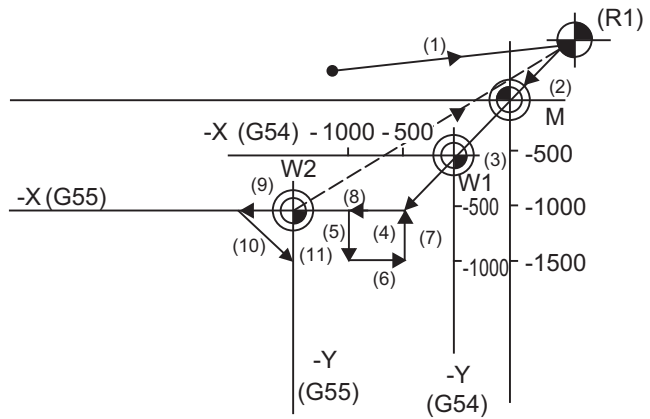
- (1) G28 X0 Y0 ;
- (2) G53 X-1000 Y-500 ;
- (3) G53 X0 Y0 ;



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.

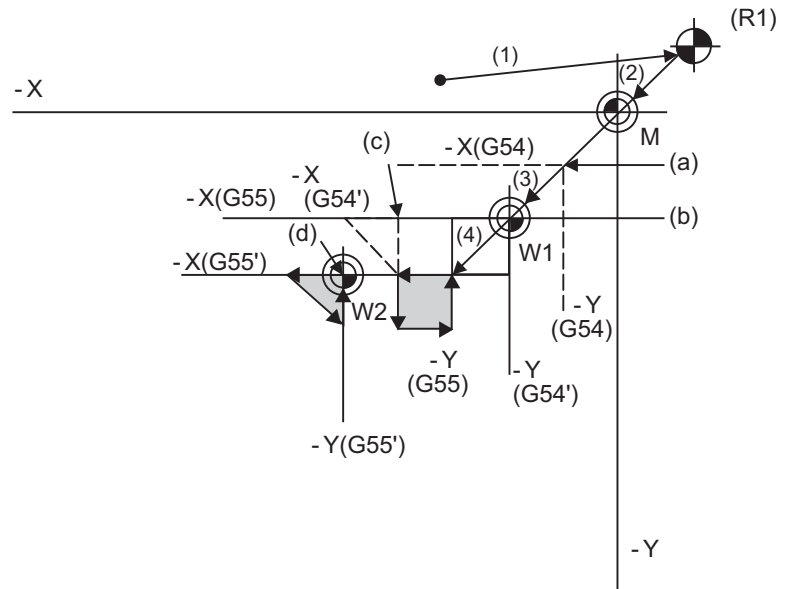
(Example 2)

- (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

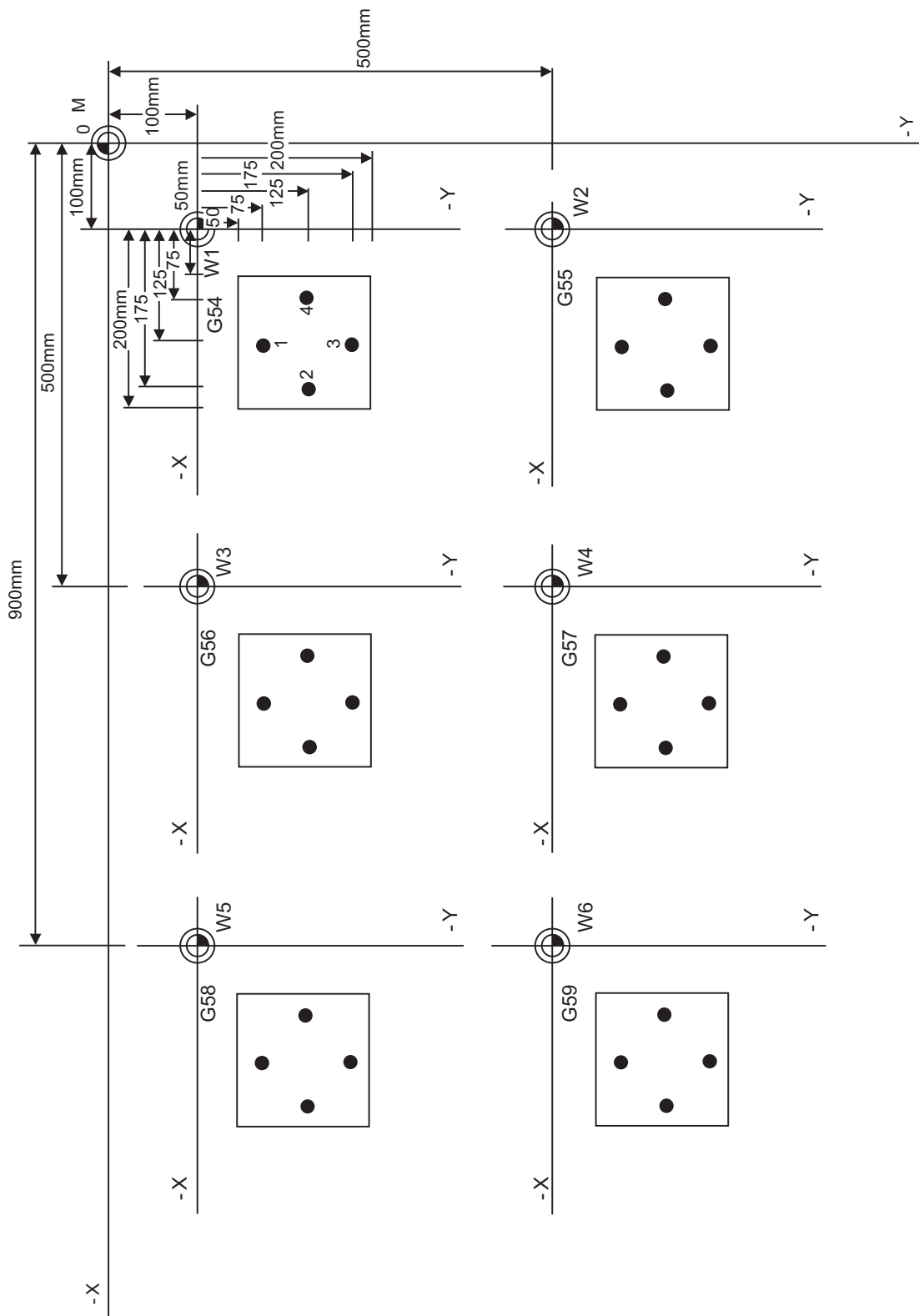
```
Workpiece 1 X=-100.000 Y=-100.000.....G54
Workpiece 2 X=-100.000 Y=-500.000.....G55
Workpiece 3 X=-500.000 Y=-100.000.....G56
Workpiece 4 X=-500.000 Y=-500.000.....G57
Workpiece 5 X=-900.000 Y=-100.000.....G58
Workpiece 6 X=-900.000 Y=-500.000.....G59
```

(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 ;
```

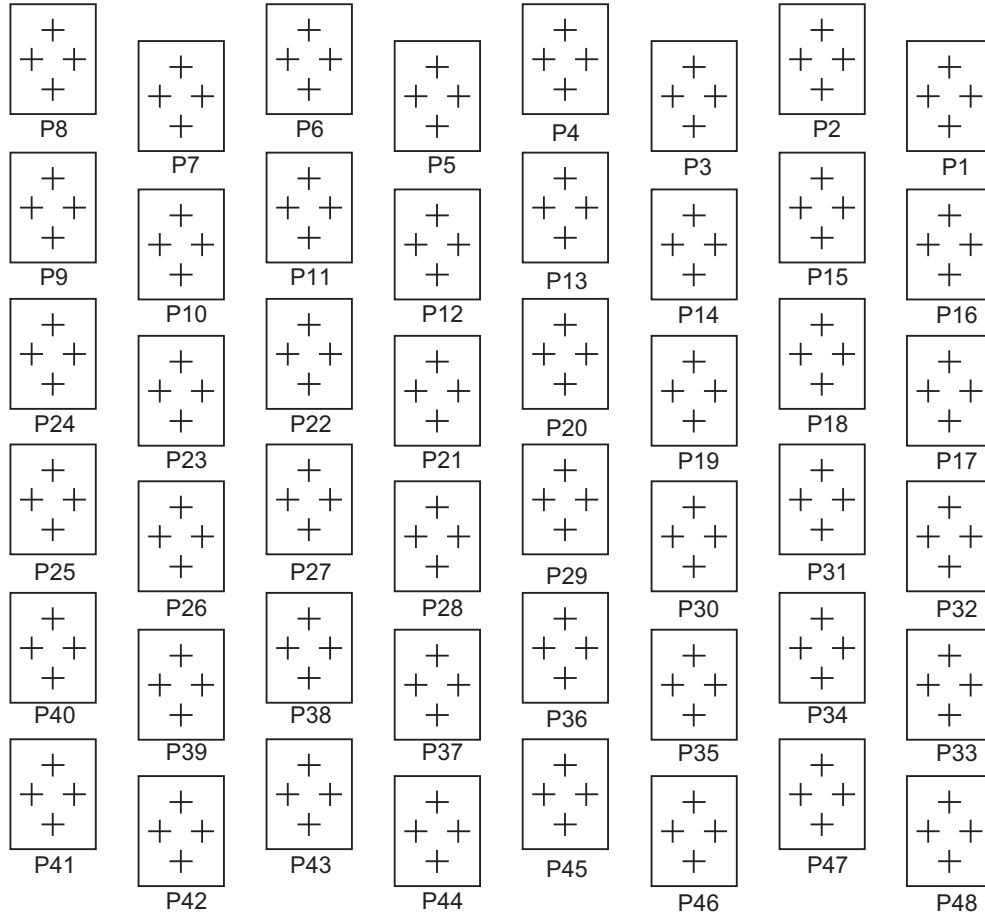
(3) Positioning program (main)

```
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 ;
%
```



(Example 5) Program example when continuously using 48 sets of added workpiece coordinate system offsets.

In this example, the offsets for each workpiece are set beforehand in P1 to P48 when 48 workpieces are fixed on a table, as shown in the drawing below.



```

01000
G28 XYZ ;           Reference position return
#100=1 ;           Initialize added workpiece coordinate
                    system P No.
G90 ;              Absolute value mode
WHILE [#100LE48]D01 ; Repeat P No. to 48
G54.1 P#100 ;      Set workpiece coordinate system
M98 P1001 ;        Call sub-program
#100=#100+1 ;      P No. +1
END1 ;
G28 Z ;            Return to reference position
G28 XY ;
M02 ;
    
```

```

01001
G43 X-10.Y-10.Z-100.H10.;   Contour
G01 X-30.;
Y-30.;
X-10.;
Y-10.;
G00 G40 Z10.;
G98 G81 X-20.Y-15.Z-150.R5.F40;   Drilling
X-25.Y-20.;
X-20.Y-25.;
X-15.Y-20.;
G80 ;
M99 ;
    
```

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.



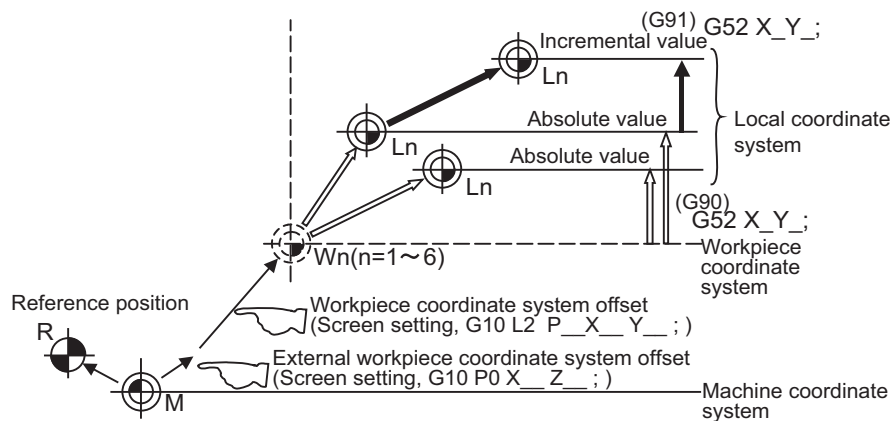
Command format

G54(G54 to G59) G52 X__ Y__ Z__ ;



Detailed description

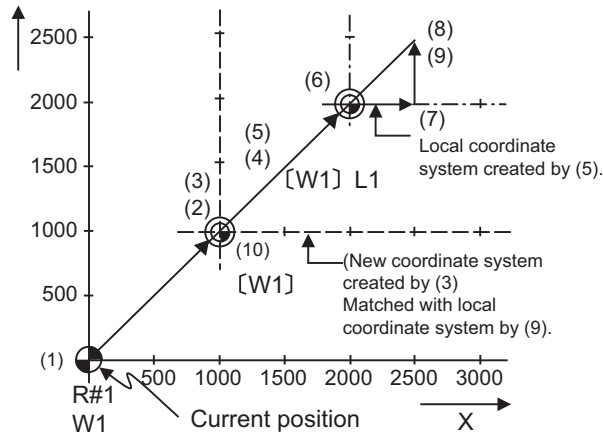
- (1) The G52 command is valid until a new G52 command is issued, and the tool does not move. This command comes in handy for employing another coordinate system without changing the zero point positions of the workpiece coordinate systems (G54 to G59).
- (2) The local coordinate system offset will be 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 ;.
- (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.

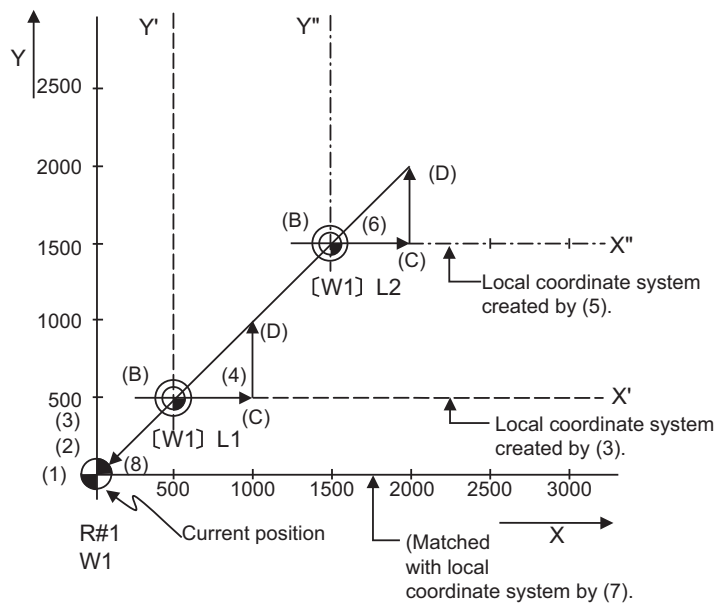
(Example 2) Local coordinates for incremental value mode (The local coordinate system offset is cumulated.)

<Main program>

- (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 ;

<Subprogram>

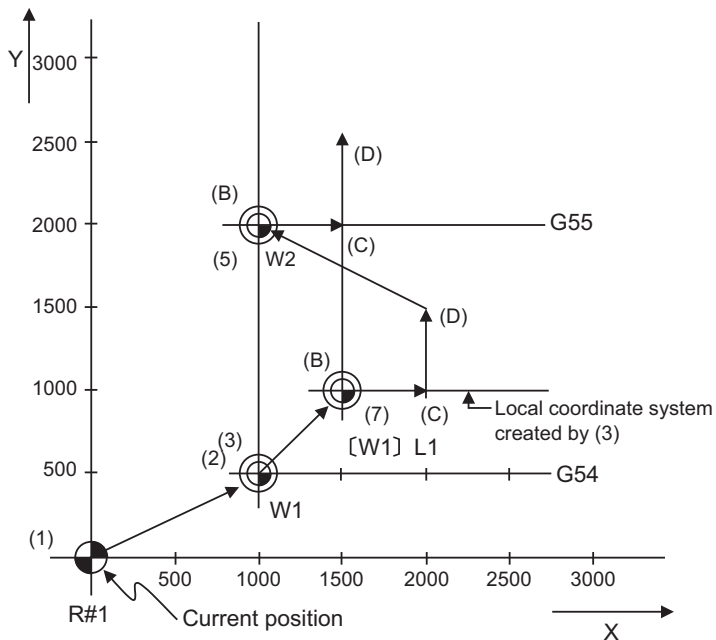
- (A) O100 ;
- (B) G90 G00 X0 Y0 ;
- (C) G01 X500 ;
- (D) Y500 ;
- (E) G91 ;
- (F) M99 ;



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 ; | G54 G55 | |
| (2) | G00 G90 G54 X0 Y0 ; | X 1000 1000 | } Workpiece coordinate system
(parameter setting value) |
| (3) | G52 X500 Y500 ; | Y 500 2000 | |
| (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 ; | | |
| | % | | |



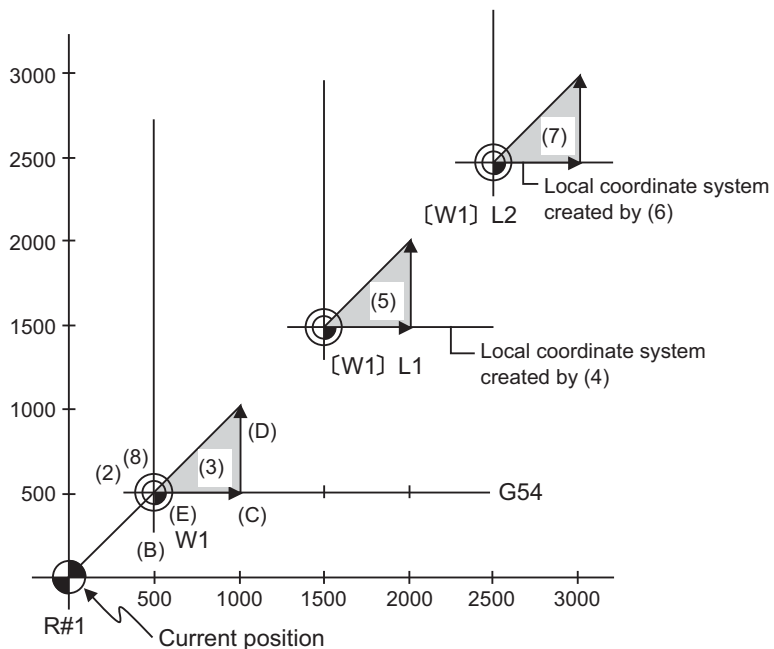
The local coordinate system is created at the G54 coordinate system (500,500) position by (3), but the local coordinate system is not created for the G55 coordinate system.

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 X0 Y0 ; G54
- (2) G00 G90 G54 X0 Y0 ; X 500
- (3) M98 P300 ; Y 500 } Workpiece coordinate offset
 (parameter setting value)
- (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 ;
- %



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.

	Rotary axis				Linear axis
	Rotating type rotary axis		Linear type rotary axis		
	Short-cut invalid	Short-cut valid	Workpiece coordinate position linear type	All-coordinate position linear type	
#8213 setting value	0	1	2	3	-
Workpiece coordinate position	Displayed in the range of 0° to 359.999° .		Displayed in the range of 0° to ± 99999.999° .		
Machine coordinate position/relative position	Displayed in the range of 0° to 359.999° .			Displayed in the range of 0° to ± 99999.999° .	
ABS command	The incremental amount from the end point to the current position is divided by 360 degrees, and the axis moves by the remainder amount according to the sign.	Moves with a short-cut to the end point.	In the same manner as the normal linear axis, 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).		
INC command	Moves in the direction of the commanded sign by the commanded incremental amount starting at the current position.				
Reference position return	Depends on the absolute command or the incremental command during the movement to the intermediate point.				
	Returns with movement within 360 degrees.			Moves and returns in the R point direction for the difference from the current position to the R point.	



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

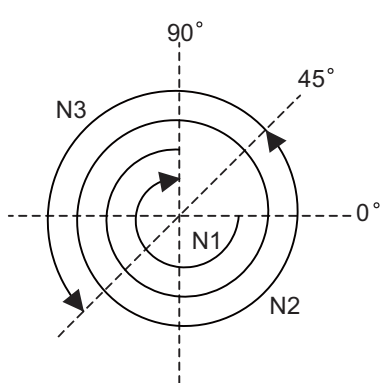
Rotary type (short-cut valid)

- (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)

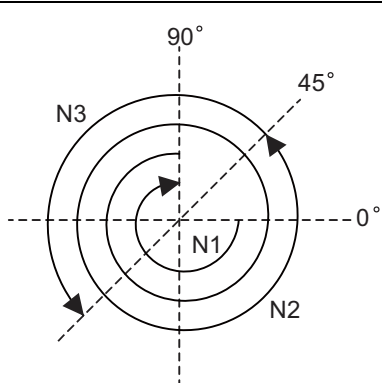
- (1) The coordinate position counter other than the workpiece coordinate position is displayed in the range of 0° to 359.999°.
The workpiece coordinate position is displayed in the range of 0 to ±99999.999°.
- (2) The movement is the same as the linear axis.
- (3) During reference position return, the 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	

Linear type (all coordinate position linear type)

- (1) The workpiece coordinate position counter 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 rotates by the difference from the intermediate point to the reference position and returns to the reference position.
- (4) During absolute position detection, the system starts up at the position where the power was turned OFF when the power is turned ON again.



Program	Workpiece	Machine	Relative position
G28 C0.			
N1 G90 C-270.	-270.000	-270.000	-270.000
N2 C405.	405.000	405.000	405.000
N3 G91 C180.	585.000	585.000	585.000
After the power is turned ON again			
	Workpiece	Machine	
	585.000	585.000	

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

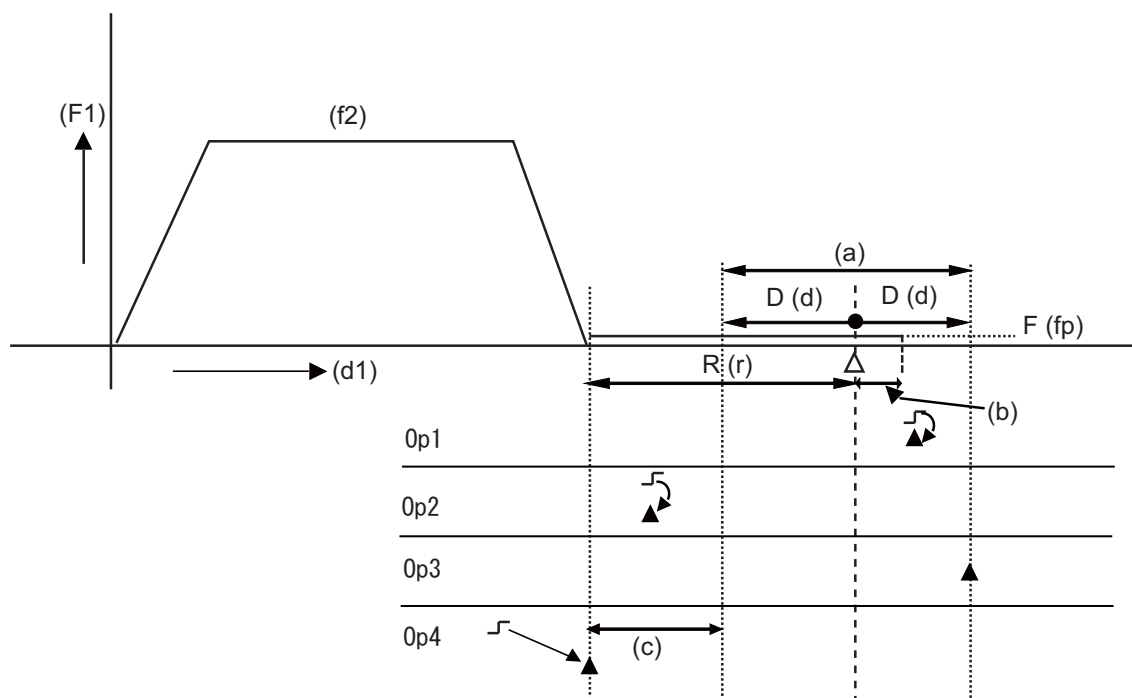
G37 Z__ R__ D__ F__ ; ... Automatic tool length measurement command

Z	Measuring axis address and coordinates of measurement position -----X,Y,Z
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 1000000 [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) Feedrate

(d) Measurement range

(r) Deceleration range

Δ 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.

$$\text{Maximum measuring error [mm]} = \text{Measuring speed [mm/min]} * 1/60 * 0.2 \text{ [ms]}/1000$$

(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.

$$\text{Maximum overtravel [mm]} = \text{Measuring speed [mm/min]} * 1/60 * 1/\text{Position loop gain [1/s]}$$

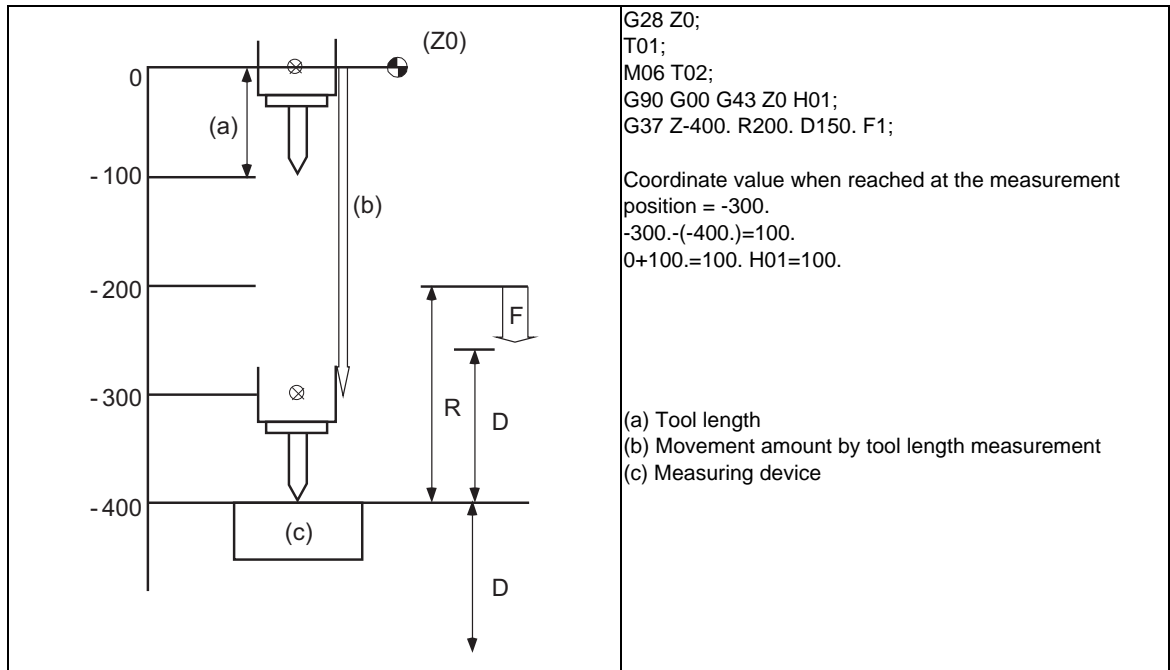
The standard position loop gain is 33 (1/s).



Operation example

For new measurement

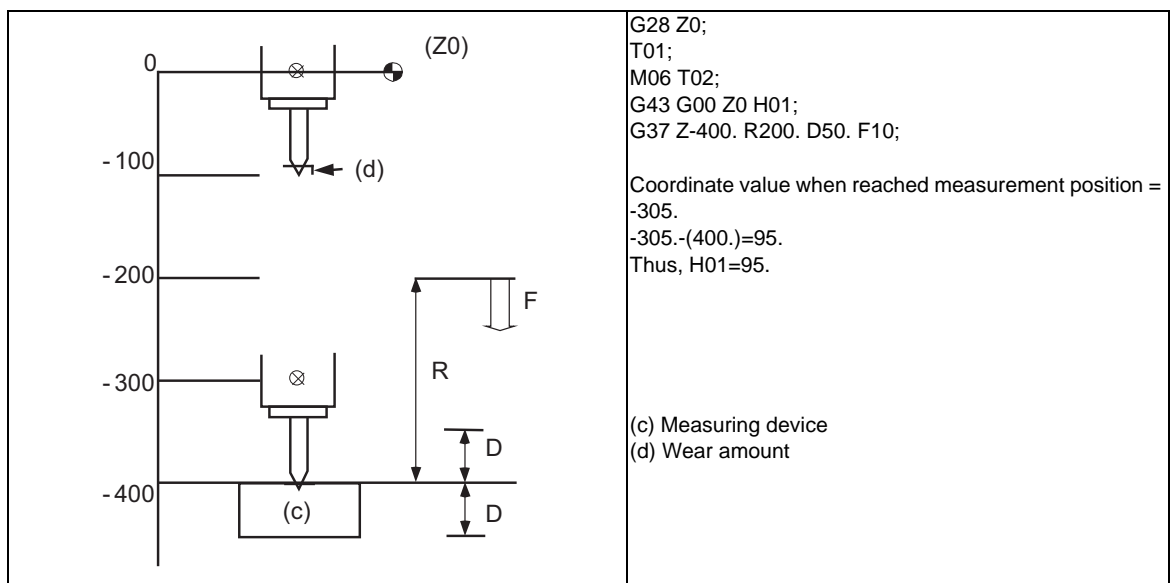
[mm]



(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.

When tool compensation is applied

[mm]



(Note) A measurement for the wear amount is applied when the current tool length compensation amount is other than zero. Thus, length wear will be compensated if length dimension by tool compensation memory type and length wear are differentiated. If not differentiated, length dimension will be compensated.



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) When the measurement allowable range is larger than the measurement command distance, it becomes the measurement allowable range for all axes.
- (11) When the measurement speed movement distance is larger than the measurement command distance, all axes move at the measurement speed.
- (12) When the measurement allowable range is larger than the measurement speed movement distance, the axis moves in the measurement allowable range at the measurement speed.
- (13) 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 coordinate value is read, the remaining distance is discarded and the command in the following block is executed.



Command format

G31 X__ Y__ Z__ R__ F__ ;

X,Y,Z	Axis coordinate value; they are commanded as absolute or incremental values according to the G90/G91 modal when commanded.
R	Acceleration/deceleration command R0: Acceleration/deceleration time constant=0 (No automatic acceleration/deceleration after interpolation.) R1: Acceleration/deceleration time constant valid. Accelerate/decelerate with the time constant set with the parameters "#2102 skip_tL" and "#2103 skip_t1". R0 is applied when it is omitted.
F	Feedrate (mm/min)

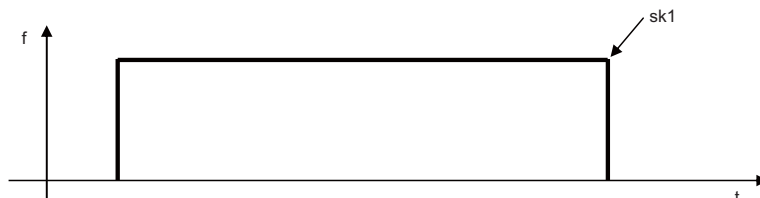


Detailed description

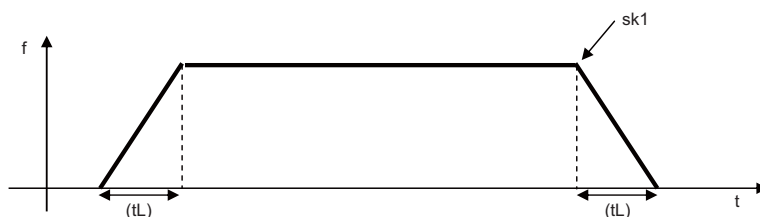
- (1) If Ff is commanded as the feedrate in the same block as the G31 command block, command feed f will apply; if not commanded, the value set in the parameter "#1174 Skip_F" will serve as the feedrate. In either case, the F modal will not be updated.
- (2) The G31 maximum speed will depend on the machine specifications.
- (3) When R0 is commanded or the R command is omitted, the step acceleration/deceleration will be applied to G31 block after the interpolation without performing the automatic acceleration/deceleration. When R1 is commanded, the automatic acceleration/deceleration will be performed according to the cutting feed acceleration/deceleration mode set by the parameter "#2003 smgst" with the time constant set by the parameter "#2102 skip_tL" and "#2103 skip_t1".
Even if G1 constant inclination acceleration/deceleration (the parameter "#1201 G1_acc" is set to "1") is valid, the time constant acceleration and deceleration will be performed.

- (4) When the R1 is commanded with the acceleration and deceleration command, the automatic acceleration and deceleration will be performed after the interpolation even if the skip single is input. Note that if the value of the parameter "#2102 skip_tL" and "#2103 skip_t1" are big, it will not stop immediately.

Acceleration/deceleration when R0 is commanded or R is omitted



Acceleration/deceleration when R1 is commanded



(sk1) Skip signal

(tL) Skip time constant

- (5) Command the acceleration/deceleration command (R0/R01) per G31 command. If not, "the acceleration/deceleration time constant=0 (R0)" will be applied and no automatic acceleration/deceleration will be performed after the interpolation.
- (6) Override is invalid with the G31 command and it is fixed at 100%. Dry run is also invalid. The stop conditions (feed hold, interlock, override zero and stroke end) are valid. External deceleration is also valid.
- (7) The G31 command is unmodal and it needs to be commanded each time.
- (8) 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.
- (9) When the G31 command is issued during tool radius compensation, the program error (P608) will occur.
- (10) When there is no F command in the G31 command and the parameter speed is also zero, the program error (P603) will occur.
- (11) 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.

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.

$$\begin{aligned} \delta 0 &= \frac{F}{60} \times T_p + \frac{F}{60} \times (t_1 \pm t_2) \\ &= \underbrace{\frac{F}{60} \times (T_p + t_1)}_{\delta 1} \pm \underbrace{\frac{F}{60} \times t_2}_{\delta 2} \end{aligned}$$

$\delta 0$: Coasting amount (mm)

F : G31 skip speed (mm/min)

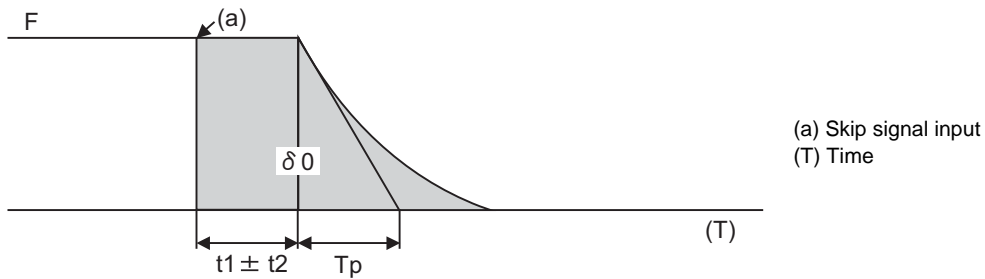
T_p : Position loop time constant (s) = (position loop gain)⁻¹

t_1 : Response delay time (s) = (time taken from the detection to the arrival of the skip signal at the controller via PC)

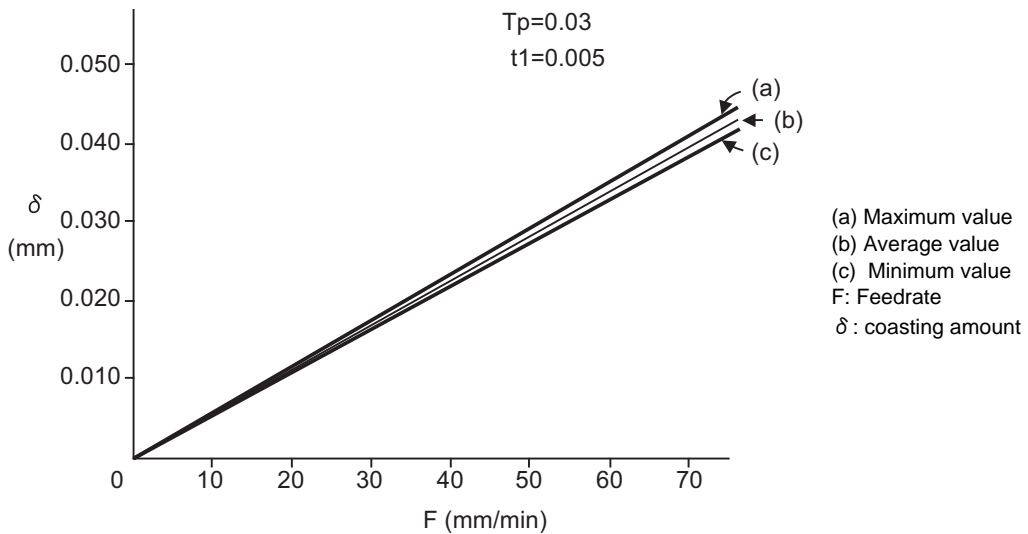
t_2 : Response error time 0.001 (s)

When G31 is used for calculation, the value calculated from the section indicated by $\delta 1$ in the above equation can be compensated, however, $\delta 2$ results in calculation error.

Stop pattern with skip signal input is shown below.



The relationship between the coasting amount and speed when T_p is 30ms and t_1 is 5ms is shown in the following figure.



Readout error of skip coordinates mm

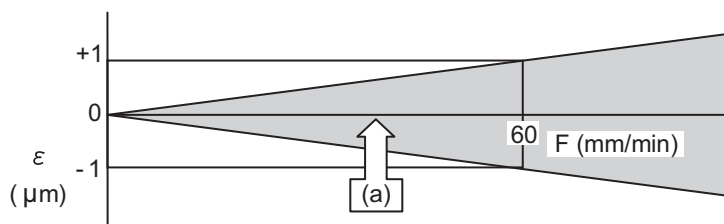
(1) Skip signal input coordinate readout

The coasting amount based on the position loop time constant T_p and cutting feed time constant T_s is not included in the skip signal input coordinate values.

Therefore, the workpiece coordinate values applying when the skip signal is input can be readout within the error range in the following formula as the skip signal input coordinate values. However, coasting based on response delay time t_1 results in a measurement error and so compensation must be provided.

$$\varepsilon = \pm (F/60) * t_2$$

ε : Readout error
 F : Feedrate
 t_2 : 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 $\pm 1 \mu\text{m}$:

$$\varepsilon = \pm (60/60) * 0.001 = \pm 0.001 \text{ (mm)}$$

(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 t_2 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

```

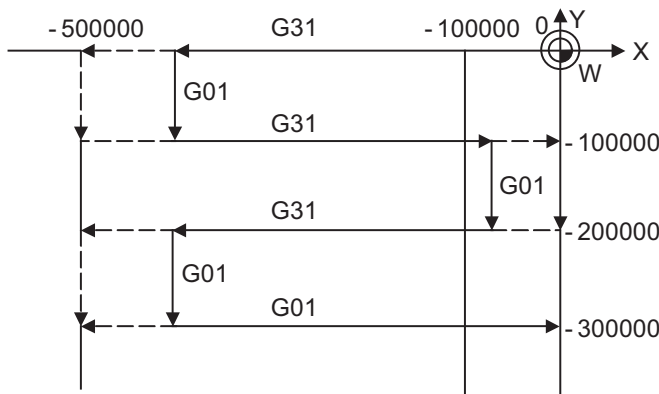
:
G31 X100.F100 ;      Skip command
G04;                Machine stop check
#101=#5061 ;        Skip signal input coordinate readout
#102=#110*#111/60 ; Coasting based on response delay time
#103=#110*#112/60 ; Coasting based on position loop time constant
#105=#101-#102-#103 ; Skip signal input coordinates
:
#110 = Skip 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__ R__ F__ ;

X,Y,Z	Target coordinates
R	Acceleration/deceleration command R0: Acceleration/deceleration time constant=0 (No automatic acceleration/deceleration after interpolation.) R1: Acceleration/deceleration time constant valid. Accelerate/decelerate with the time constant set with the parameters "#2102 skip_tL" and "#2103 skip_t1". R0 is applied when it is omitted.
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

- Command the skip speed by the program command or the parameter. Feedrate G31.1 set with the parameter corresponds to "#1176 skip1f", G31.2 corresponds to "#1178 skip2f", G31.3 corresponds to "#1180 skip3f", and G04 corresponds to "#1173 dwlskp". However, F modal will not be updated in each case.
- A command is skipped if it meets the specified skip signal condition.
- The feedrates corresponding to the G31.1, G31.2, and G31.3 commands can be set by parameters.
- The skip conditions (logical sum of skip signals which have been set) corresponding to the G31.1, G31.2, G31.3 and G04 commands can be set by parameters.

Parameter setting	Valid skip signal		
	1	2	3
1	○		
2		○	
3	○	○	
4			○
5	○		○
6		○	○
7	○	○	○

- Other commands work the same as the G31 (skip function) command.



Operation example

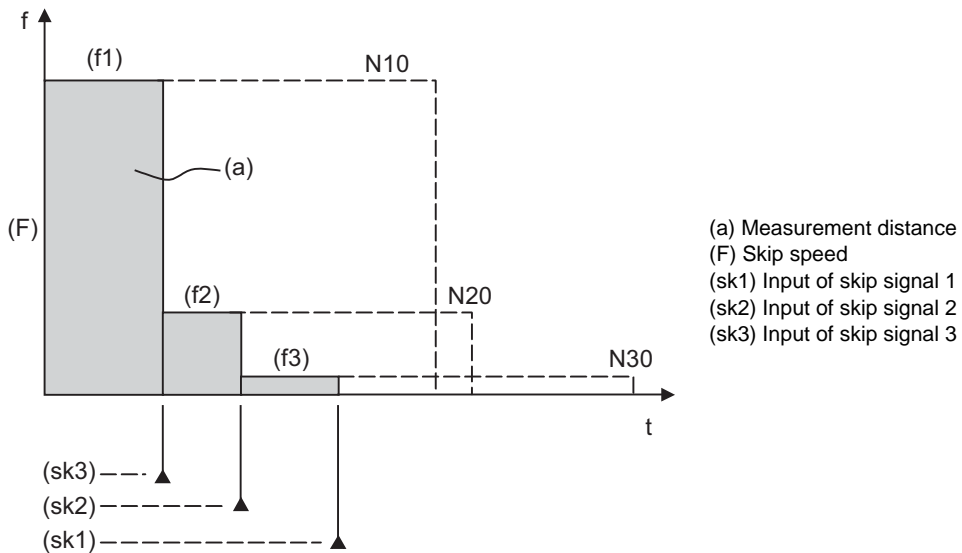
- (1) The multi-step skip function enables the following control, thereby improving measurement accuracy and shortening the time required for measurement.

[Parameter settings]

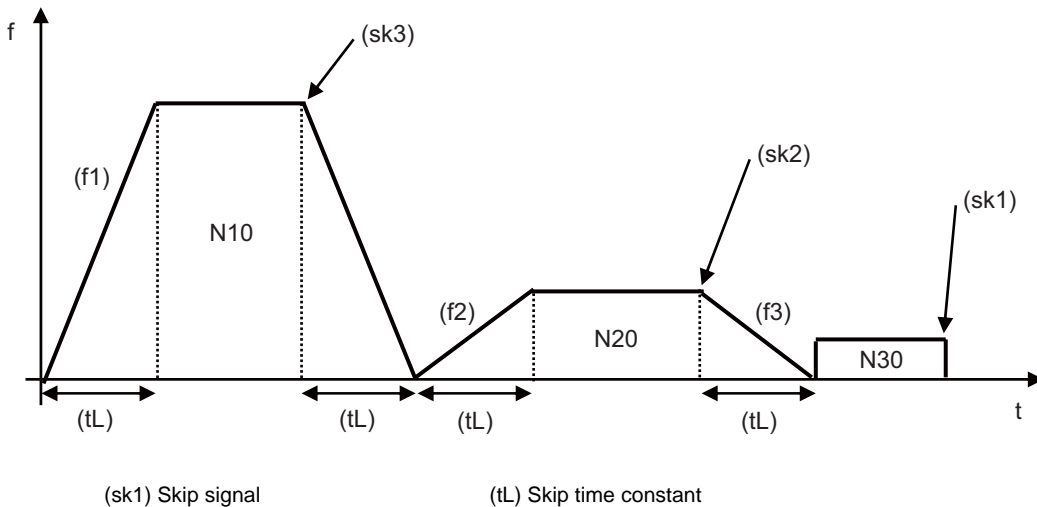
Skip condition	Skip speed
G31.1 : 7	20.0mm/min (f1)
G31.2 : 3	5.0mm/min (f2)
G31.3 : 1	1.0mm/min (f3)

[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.

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.



Command format

G31 X__ Y__ Z__ P__ R__ F__ ;

X Y Z	Target coordinates
P	Skip signal command
R	Acceleration/deceleration command R0: Acceleration/deceleration time constant=0 (No automatic acceleration/deceleration after interpolation.) R1: Acceleration/deceleration time constant valid. Accelerate/decelerate with the time constant set with the parameters "#2102 skip_tL" and "#2103 skip_t1". R0 is applied when it is omitted.
F	Feedrate (mm/min)



Detailed description

- (1) The skip speed is specified by program command or parameter. The feedrate by the parameter is set by "#1174 skip_F". Note that the F modal is not updated in each case.
- (2) The skip signal is specified by skip signal command p. The command range of "p" is from 1 to 255. If outside the range is commanded, program error (P35) will occur.

Skip signal command P	Valid skip signal							
	8	7	6	5	4	3	2	1
1								○
2							○	
3							○	○
4						○		
5						○		○
6						○	○	
7						○	○	○
8					○			
⋮								
⋮								
⋮								
253	○	○	○	○	○	○		○
254	○	○	○	○	○	○	○	
255	○	○	○	○	○	○	○	○

- (3) The specified skip signal command is a logical sum of the skip signals.
(Example) G31 X100. P5 F100 ;
Operation is skipped if skip signal 1 or 3 is input.
- (4) If skip signal parameter Pp is not specified, 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	x		○	
	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 X100. 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. "#2214 SVO14(current limit value in special control)" can be changed. 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 999



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 999%. Commands that exceed this range will cause a program error (P35).
- (5) If a decimal point is designated with the G10 command, only the integer will be valid.
Example) G10 L14 X10.123 ; The current limit value will be set to 10%.
- (6) For the axis name "C", the current limit value cannot be set from the program (G10 command).
To set from the program, set the axis address with an incremental axis name, or set the axis name to one other than "C".

Appendix 1

Program Errors

These alarms occur during automatic operation, and the causes of these alarms are mainly program errors which occur, for instance, when mistakes have been made in the preparation of the machining programs or when programs which conform to the specification have not been prepared.

(Note) This alarm description is common for M700V/M70V/E70 Series.

It is written on the assumption that all option functions are added.

Confirm with the specifications issued by the machine tool builder before starting use.

P10 No. of simultaneous axes over

Details

The number of axis addresses commanded in a block exceeds the specifications.

Remedy

- Divide the alarm block command into two.
- Check the specifications.

P11 Illegal axis address

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.

P29 Not accept command

Details

The command has been issued when it is impossible.

- The normal line control command (G40.1, G41.1, G42.1) has been issued during the modal in which the normal line control is not acceptable.
- The command has been issued during the modal in which the 2-part system synchronous thread cutting is not acceptable.

Remedy

- Correct the program.

P30 Parity H error

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 error

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 Illegal address

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 Illegal G code**Details**

The commanded G code is not in the specifications.
An illegal G code was commanded during the coordinate rotation command.

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 Setting value range over**Details**

The setting range for the addresses has been exceeded.

Remedy

- Correct the program.

P36 Program end error**Details**

"EOR" has been read during tape and 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 O, N number 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.

P38 No spec: Add. Op block skip**Details**

"/n" has been issued while the optional block skip addition is not in the specifications.

Remedy

- Check the specifications.

P39 No specifications**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.

P48 Restart pos return incomplete**Details**

A travel command was issued before the execution of the block that had been restart-searched.

Remedy

- Carry out program restart again.
Travel command cannot be executed before the execution of the block that has been restart-searched.

P49 Invalid restart search**Details**

- Restart search was attempted for the 3-dimensional circular interpolation.
- Restart search was attempted for the mixed control (cross axis control) command (G110).
- Restart search was attempted during the cylindrical interpolation, polar coordinate interpolation, or tool tip center control.
- Restart search was attempted from a block (G68.2) during the inclined surface machining mode or from the inclined surface machining mode cancel command block (G69).
- Restart search was attempted to the program after direct command mode.

Remedy

- Correct the program.
- Correct the restart search position.

P50 No spec: Inch/Metric change**Details**

Inch/Metric changeover (G20/G21) command was issued while the function is out of specifications.

Remedy

- Check the specifications.

P60 Compensation length over**Details**

The commanded movement distance is excessive (over 2^{31}).

Remedy

- Correct the command range for the axis address.

P61 No spec: Unidirectional posit.**Details**

Unidirectional positioning (G60) was commanded while the function is out of specifications.

Remedy

- Check the specifications.

P62 No F command**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.

P63 No spec: High-speed machining**Details**

High-speed machining cancel (G5P0) was commanded while the high-speed machining is out of specifications.

Remedy

- Check the specifications.

P65 No spec: High speed mode 3**Details****Remedy**

- Check whether the specifications are provided for the high-speed mode III.

P70 Arc end point deviation large**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 error**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 spec: Helical cutting**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.

P73 No spec: Spiral cutting**Details**

A spiral command was issued though it is out of specifications.

Remedy

- Issue the G02.1 and G03.1 commands for circular interpolation.
- Check whether the specifications are provided for the spiral cutting.

P74 Can't calculate 3DIM arc**Details**

The 3-dimension circular cannot be obtained because the end block was not specified during 3-dimension circular interpolation supplementary modal.

The 3-dimension circular cannot be obtained due to an interruption during 3-dimension circular interpolation supplementary modal.

Remedy

- Correct the program.

P75 3DIM arc illegal**Details**

An illegal G code was issued during 3-dimension circular interpolation modal. Otherwise, 3-dimension circular interpolation command was issued during a modal for which a 3-dimension circular interpolation command cannot be issued.

Remedy

- Correct the program.

P76 No spec: 3DIM arc interpolat**Details**

G02.4/G03.4 was commanded though there is no 3-dimension circular interpolation specification.

Remedy

- Check the specifications.

P80 No spec: Hypoth ax interpolat**Details**

Hypothetical axis interpolation (G07) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P90 No spec: Thread cutting**Details**

A thread cutting command was issued though it is out of specifications.

Remedy

- Check the specifications.

P91 No spec: Var lead threading**Details**

Variable lead thread cutting (G34) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P93 Illegal pitch vaule**Details**

An illegal thread lead (thread pitch) was specified at the thread cutting command.

Remedy

- Correct the thread lead for the thread cutting command.

P100 No spec: Cylindric interpolat**Details**

A cylindrical interpolation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P110 Plane select during figure rot**Details**

Plane selection (G17/G18/G19) was commanded during figure rotation.

Remedy

- Correct the machining program.

P111 Plane selected while coord rot**Details**

Plane selection commands (G17, G18, G19) were issued during a coordinate rotation was being commanded.

Remedy

- Always command coordinate rotation cancel after the coordinate rotation command, and then issue a plane selection command.

P112 Plane selected while R compen**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 select**Details**

The circular command axis does not correspond to the selected plane.

Remedy

- Select a correct plane before issuing a circular command.

P120 No spec: Feed per rotation**Details**

Feed per rotation (G95) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P121 F0 command during arc modal**Details**

F0 (F 1-digit feed) was commanded during the arc modal (G02/G03).

Remedy

- Correct the machining program.

P122 No spec: Auto corner override**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.

P123 No spec: High-accuracy control**Details**

High-accuracy control command was issued though it is out of specifications.

Remedy

- Check the specifications.

P124 No spec: Inverse time feed**Details**

- The inverse time option is not provided.

Remedy

- Check the specifications.

P125 G93 mode error**Details**

- The issued G code command is illegal during G93 mode.
- G93 command was issued during a modal for which inverse time feed cannot be performed.

Remedy

- Correct the program.

P126 Invalid cmnd in high-accuracy**Details**

An illegal command was issued during the high-accuracy control mode.

- A G code group 13 command was issued during the high-accuracy control mode.
- Milling, cylindrical interpolation or pole coordinate interpolation was commanded during the high-accuracy control mode.

Remedy

- Correct the program.

P127 No spec: SSS Control**Details**

The SSS control valid parameter has been set although there is no SSS control specification.

Remedy

- Check the specifications. Set "#8090 SSS ON" to "0" when there is no SSS control specification.

P128 Machin condtn select I disable**Details**

Machining condition selection I was commanded during the mode where the selection command is unavailable.

Remedy

- Check the program and see whether any unavailable mode is included during machining condition selection I. If any, cancel that mode.

P130 2nd M function code illegal**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 spec: Cnst surface ctrl G96**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 rotation speed S=0**Details**

No spindle rotation speed command has been issued.

Remedy

- Correct the program.

P133 Illegal P-No. G96**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.

P140 No spec: Pos compen cmd**Details**

The position compensation command (G45 to G48) is out of specifications.

Remedy

- Check the specifications.

P141 Pos compen during rotation**Details**

Position compensation was commanded during the figure rotation or coordinate rotation command.

Remedy

- Correct the program.

P142 Pos compen invalid arc**Details**

Position compensation cannot be executed with the issued arc command.

Remedy

- Correct the program.

P150 No spec: Nose R compensation**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 Radius compen during arc mode**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 No intersection**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.
- The compensation amount cannot be calculated during the tool radius compensation for 5-axis machining (G41.2,G42.2).

Remedy

- Correct the program.

P153 Compensation interference**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.

P154 No spec: 3D compensation**Details**

A three-dimensional compensation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P155 Fixed cyc exec during compen

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 R compen direction not defined

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 R compen direction changed

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 point

Details

An illegal tip point No. (other than 1 to 8) was found during G46 nose R compensation.

Remedy

- Correct the tip point No.

P159 C-CMP REMAINS

Details

One of the followings was commanded while the compensation has not been canceled (the compensation amount remained).

- (1) Automatic tool nose R compensation (G143)
- (2) Radius compensation (G145)
- (3) Plane selection (G17 to G19)
- (4) Skip (G31 or G31.1/G31.2/G31.3)
- (5) Fixed cycle for drilling (G81 to G89)
- (6) Compound type fixed cycle II (G74 to G76)

Remedy

- Cancel the compensation (with compensation amount "0") before commanding.
- Command the G00 move block before the block with a command among (1) to (6).

P160 G53 CMP. ERR

Details

- G53 was commanded during nose R compensation (by G41/G42/G46).
- G53 was commanded to the block where the nose R compensation mode is changed (with G40/G41/G42/G46).
- G53 was commanded while nose R compensation amount has not been cleared.

Remedy

- Correct the program.
- When issuing G53 after G40 command, move the compensation plane axis by G00/G01/G02/G03 command before issuing G53.

P161 No spec: 5ax tool R compensate**Details**

Tool radius compensation for 5-axis machining is not included in the specifications.

Remedy

- Check the specifications.

P162 Disable Cmd in 5ax tool R comp**Details**

A command (G or T command, etc) was issued during tool radius compensation for 5-axis machining, although it is disabled during the compensation.

Remedy

- Cancel the tool radius compensation for 5-axis machining.

P163 5 ax tool R comp is disable**Details**

Tool radius compensation for 5-axis machining was commanded in a mode where the command is disabled.

Remedy

- Cancel the mode that disables the command.

P170 No offset number**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.
- When setting the L system tool life management II, the tool group management program was executed with the tool life management disabled.

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.
- When setting the L system tool life management II ("#1096 T_Ltyp="2"), enable the tool life management ("#1103 T_Life="1") to execute a tool group management program.

P171 No spec:Comp input by prog G10**Details**

Compensation data input by program (G10) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P172 G10 L number error**Details**

An address of G10 command is not correct.

Remedy

- Correct the address L No. of the G10 command.

P173 G10 P number error**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 spec:Comp input by prog G11**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 Tool life count active**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 Tool life data entry 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 Illegal group No.**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 spec: Drilling cycle**Details**

A fixed cycle command (G72 - G89) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Correct the program.

P181 No spindle command (Tap cycle)**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 Synchronous 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 No pitch/thread number**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 Pitch/thread number error**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 spec: Sync tapping cycle**Details**

Synchronous tapping cycle (G84/G74) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P186 Illegal S cmd in synchro tap**Details**

S command was issued during synchronous tapping modal.

Remedy

- Cancel the synchronous tapping before issuing the S command.

P190 No spec: Turning cycle**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 length error**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 error**Details**

Chamfering in the thread cutting cycle is illegal.

Remedy

- Set a chamfering amount not exceeding the cycle.

P200 No spec: MRC cycle**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 Program error (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 over (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 D cmnd figure error (MRC)**Details**

A proper shape will not be 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 E cmnd fixed cycle error**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 spec: Pattern cycle**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 spec: Special fixed cycle**Details**

There are no special fixed cycle specifications.

Remedy

- Check the specifications.

P221 No. of special fixed holes = 0**Details**

"0" has been specified for the number of holes in special fixed cycle mode.

Remedy

- Correct the program.

P222 G36 angle error**Details**

A G36 command specifies "0" for angle intervals.

Remedy

- Correct the program.

P223 G12/G13 radius error**Details**

The radius value specified with a G12 or G13 command is below the compensation amount.

Remedy

- Correct the program.

P224 No spec: Circular (G12/G13)**Details**

There are no circular cutting specifications.

Remedy

- Check the specifications.

P230 Subprogram 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 sequence No.**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.

P235 Program editing**Details**

Operation was attempted for the file under program editing.

Remedy

- Execute the program again after completion of program editing.

P240 No spec: Variable commands**Details**

A variable command (with #) was issued though it is out of specifications.

Remedy

- Check the specifications.

P241 No variable No.**Details**

The variable No. commanded is out of the range specified in the specifications.

Remedy

- Check the specifications.
- Correct the program variable No.

P242 = not defined at vrble set

Details

The "=" sign has not been commanded when a variable is defined.

Remedy

- Designate the "=" sign in the variable definition of the program.

P243 Can't use variables

Details

An invalid variable has been specified in the left or right side of an operation expression.

Remedy

- Correct the program.

P244 Invalid set date or time

Details

Date or time was set earlier than current date or time in the system variables (#3011, #3012) when the system lock was valid.

Remedy

- Date or time cannot be changed.
- Correct the program.

P250 No spec: Figure rotation

Details

Figure rotation (M98 I_J_P_H_L_) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P251 Figure rotation overlapped

Details

Figure rotation command was issued during figure rotation.

Remedy

- Correct the program.

P252 Coord rotate in fig. rotation

Details

A coordinate rotation related command (G68, G69) was issued during figure rotation.

Remedy

- Correct the program.

P260 No spec: Coordinates rotation

Details

A coordinate rotation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P261 G code illegal (Coord rot)

Details

Another G code or a T command has been issued in the block of coordinate rotation command.

Remedy

- Correct the program.

P262 Illegal modal (Coord rot)

Details

A coordinate rotation command has been issued during modal in which coordinate rotation is not allowed.

Remedy

- Correct the program.

P270 No spec: User macro**Details**

A macro specification was commanded though it is out of specifications.

Remedy

- Check the specifications.

P271 No spec: Macro interrupt**Details**

A macro interruption command has been issued though it is out of specifications.

Remedy

- Check the specifications.

P272 NC and macro texts in a block**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 call nesting over**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 argument over**Details**

The number of argument sets in the macro call argument type II has exceeded the limit.

Remedy

- Correct the program.

P276 Illegal G67 command**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 alarm message**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 Brackets [] nesting over**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 Brackets [] not paired**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 Calculation impossible**Details**

The arithmetic formula is incorrect.

Remedy

- Correct the formula in the program.

P283 Divided 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.

P288 IF EXCESS**Details**

Over ten times of IF statement nesting have been done.

Remedy

Correct the program so that the number of IF statement nesting does not exceed ten.

P289 IF-ENDIF MMC.**Details**

An IF statement is not ended with ENDIF.
THEN/ELSE has been commanded while there is no IF command.

Remedy

- Correct the program so that IF statements are ended with ENDIFs.
- Put IF[condition] before THEN/ELSE command.

P290 IF sentence error**Details**

There is an error in the "IF[<conditional>]GOTO(" statement.

Remedy

- Correct the program.

P291 WHILE sentence error**Details**

There is an error in the "WHILE[<conditional>]DO(-END(" statement.

Remedy

- Correct the program.

P292 SETVN sentence error**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 nesting over**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 and END not paired**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 in tape**Details**

There is a WHILE or GOTO statement on the tape during tape operation.

Remedy

- Apply memory mode operation instead of tape mode that does not allow the execution of the program with a WHILE or GOTO statement.

P296 No address (macro)**Details**

A required address has not been specified in the user macro.

Remedy

- Correct the program.

P297 Address-A error**Details**

The user macro does not use address A as a variable.

Remedy

- Correct the program.

P298 G200-G202 cmnd in tape**Details**

User macro G200, G201, or G202 was specified during tape or MDI mode.

Remedy

- Correct the program.

P300 Variable name illegal**Details**

The variable names have not been commanded properly.

Remedy

- Correct the variable names in the program.

P301 Variable name duplicated**Details**

A duplicate variable name was found.

Remedy

- Correct the program so that no duplicate name exists.

P310 Not use GMSTB macro code**Details**

G, M, S, T, or B macro code was called during fixed cycle.

Remedy

- Correct the program.
- Correct the parameter settings.

P350 No spec: Scaling command**Details**

The scaling command (G50, G51) was issued though it is out of specifications.

Remedy

- Check the specifications.

P360 No spec: Program mirror**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.

P370 No spec: Facing t-post MR**Details**

The facing turret mirror image specifications are not provided.

Remedy

- Check the specifications.

P371 Facing t-post MR illegal**Details**

- Mirror image for facing tool posts was commanded to an axis in external mirror image or parameter mirror image.
- The commanded mirror image for facing tool posts enables the mirror image for a rotary axis.

Remedy

- Correct the program.
- Correct the parameter settings.

P380 No spec: 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 spec: Arc R/C**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 No corner movement**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 movement 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 next movement 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 Corner during G00/G33**Details**

A block with corner chamfering/corner rounding was given during G00 or G33 modal.

Remedy

- Correct the program.

P390 No spec: Geometric**Details**

A geometric command was issued though it is out of specifications.

Remedy

- Check the specifications.

P391 No spec: Geometric arc**Details**

There are no geometric IB specifications.

Remedy

- Check the specifications.

P392 Angle < 1 degree (GEOMT)**Details**

The angular difference between the geometric line and line is 1° or less.

Remedy

- Correct the geometric angle.

P393 Inc value in 2nd block (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 linear move command (GEOMT)**Details**

The second geometric block contains no linear command.

Remedy

- Issue the G01 command.

P395 Illegal address (GEOMT)**Details**

The geometric format is invalid.

Remedy

- Correct the program.

P396 Plane selected in GEOMT ctrl**Details**

A plane switching command was issued during geometric command processing.

Remedy

- Complete the plane switching command before geometric command processing.

P397 Arc error (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 spec: Geometric1B**Details**

A geometric command was issued though the geometric IB specifications are not provided.

Remedy

- Check the specifications.

P411 Illegal modal G111

Details

- G111 was issued during milling mode.
- G111 was issued during nose R compensation mode.
- G111 was issued during constant surface speed.
- G111 was issued during mixed control (cross axis control).
- G111 was issued during fixed cycle.
- G111 was issued during polar coordinate interpolation.
- G111 was issued during cylindrical interpolation mode.

Remedy

- Before commanding G111, cancel the following commands.
- Milling mode
- Nose R compensation
- Constant surface speed
- Mixed control (cross axis control)
- Fixed cycle
- Polar coordinate interpolation
- Cylindrical interpolation

P412 No spec: Axis name switch

Details

Axis name switch (G111) was issued though it is out of specifications.

Remedy

- Check the specifications.

P420 No spec: Para input by program

Details

Parameter input by program (G10) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P421 Parameter input 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.
- G10L70, G11 were not commanded in independent blocks.

Remedy

- Correct the program.

P430 R-pnt return incomplete

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 spec: 2,3,4th R-point ret

Details

A command for second, third or fourth reference position return was issued though there are no such command specifications.

Remedy

- Check the specifications.

P432 No spec: Start position return**Details**

Start position return (G29) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P433 No spec: R-position check**Details**

Reference position check (G27) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P434 Compare error**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 and M commands in a block**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 and M commands in a block**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 G52 invalid during G54.1**Details**

A local coordinate system command was issued during execution of the G54.1 command.

Remedy

- Correct the program.

P450 No spec: Chuck barrier**Details**

The chuck barrier on command (G22) was specified although the chuck barrier is out of specifications.

Remedy

- Check the specifications.

P451 No spec: Stroke chk bef travel**Details**

Stroke check before travel (G22/G23) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P452 Limit before travel exists**Details**

An illegal command, which places the axis travel start/end point in the prohibited area or moves the axis through the prohibited area, was detected when Stroke check before travel (G22) was commanded.

Remedy

- Correct the coordinate values of the axis address commanded in the program.

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.
- IC card has not been inserted.

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.
- Ensure that the external device (including an IC card, etc) that contains the file is mounted.
- Correct the parameter settings for HD operation or IC card operation.

P462 Computer link commu error**Details**

A communication error occurred during the BTR operation.

Remedy

- "L01 Computer link error" is displayed simultaneously. Take the remedy corresponding to the error No.

P480 No spec: Milling**Details**

- Milling was commanded though it is out of specifications.
- Polar coordinate interpolation was commanded though it is out of specifications.

Remedy

- Check the specifications.

P481 Illegal G code (mill)**Details**

- An illegal G code was used during the milling mode.
- An illegal G code was used during cylindrical interpolation or polar coordinate interpolation.
- The G07.1 command was issued during the tool radius compensation.

Remedy

- Correct the program.

P482 Illegal axis (mill)**Details**

- A rotary axis was commanded during the milling mode.
- Milling was executed though an illegal value was set for the milling axis No.
- Cylindrical interpolation or polar coordinate interpolation was commanded during mirror image.
- Cylindrical interpolation or polar coordinate interpolation was commanded before the tool compensation was completed after the T command.
- G07.1 was commanded when cylindrical interpolation was not possible (there is no rotary axis, or external mirror image is ON).
- An axis other than a cylindrical coordinate system axis was commanded during cylindrical interpolation.

Remedy

- Correct the machining program, parameters and PLC interface signals.

P484 R-pnt ret incomplete (mill)**Details**

- Movement was commanded to an axis that had not completed reference position return during the milling mode.
- Movement was commanded to an axis that had not completed reference position return during cylindrical interpolation or polar coordinate interpolation.

Remedy

- Carry out manual reference position return.

P485 Illegal modal (mill)**Details**

- The milling mode was turned ON during nose R compensation or constant surface speed control.
- A T command was issued during the milling mode.
- The mode was switched from milling to cutting during tool compensation.
- Cylindrical interpolation or polar coordinate interpolation was commanded during the constant surface speed control mode (G96).
- The command unacceptable in the cylindrical interpolation was issued.
- A T command was issued during the cylindrical interpolation or polar coordinate interpolation mode.
- A movement command was issued when the plane was not selected just before or after the G07.1 command.
- A plane selection command was issued during the polar coordinate interpolation mode.
- Cylindrical interpolation or polar coordinate interpolation was commanded during tool radius compensation.
- The G16 plane in which the radius value of a cylinder is "0" was specified.
- A cylindrical interpolation or polar coordinate interpolation command was issued during coordinate rotation by program.

Remedy

- Correct the program.
- Issue G40 or G97 before issuing G12.1.
- Issue a T command before issuing G12.1.
- Issue G40 before issuing G13.1.
- Specify the radius value of a cylinder other than "0", or specify the X axis's current value other than "0" before issuing G12.1/G16.

P486 Milling error**Details**

- The milling command was issued during the mirror image (when parameter or external input is turned ON).
- Polar coordinate interpolation, cylindrical interpolation or milling interpolation was commanded during mirror image for facing tool posts.
- The start command of the cylindrical interpolation or polar coordinate interpolation was issued during the normal line control.

Remedy

- Correct the program.

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.

P511 Synchronization M code error

Details

- Two or more synchronization M codes were commanded in the same block.
- The synchronization M code and "!" code were commanded in the same block.
- Synchronization with the M code was commanded in 3rd part system or more. (Synchronization with the M code is valid only in 1st part system or 2nd part system.)

Remedy

- Correct the program.

P520 Control axis superimposition/Designated axis illegal

Details

An axis which was impossible to superimpose was designated as a master axis or superimposing axis.

Remedy

Correct the program.

P521 Illegal synchronization axis

Details

The axis specified as a basic or synchronous axis of synchronization across part systems cannot be synchronized.

Remedy

- Correct the program.

P530 DEC. POINT ERR

Details

A decimal point was added to the address where the decimal point command is not allowed. ("#1274 ext10/bit0 (Type of address enabling/disabling decimal point command)")

Remedy

Do not add any decimal point to the addresses where the decimal point is not allowed.

P544 No spec: Wk instl err cmp

Details

The workpiece installation error compensation function is out of the specifications.

Remedy

- Check the specifications.

P545 Invld cmd in wk instl err cmp

Details

During workpiece installation error compensation, a command impossible to issue (such as G command) was issued.

Remedy

- Check the program. If you wish to issue a command impossible to issue (such as G command) during workpiece installation error compensation, cancel workpiece installation error compensation once.

P546 Wk instl err cmp cmd invalid**Details**

- Workpiece installation error compensation was commanded in a G modal in which commanding it is not allowed.
- An illegal G command was issued in the block that has a workpiece installation error compensation command.

Remedy

- Check the program. Also check the G modals which were issued at commanding the workpiece installation error compensation, and cancel illegal ones.
- Issue the G command in a separate block.

P547 Illegal wk instl err cmp cmd**Details**

A command in which the rotary axis's travel distance exceeds 180 degrees was issued.

Remedy

- Divide the travel command so that the rotary axis's travel distance per block is less than 180 degrees.

P550 No spec: G06.2(NURBS)**Details**

There is no NURBS interpolation option.

Remedy

- Check the specifications.

P551 G06.2 knot error**Details**

The knot (k) command value is smaller than the value for the previous block.

Remedy

- Correct the program.
- Specify the knot by monotone increment.

P552 Start point of 1st G06.2 err**Details**

The block end point immediately before the G06.2 command and the G06.2 first block command value do not match.

Remedy

- Match the G06.2 first block coordinate command value with the previous block end point.

P554 Invl manual interrupt in G6.2**Details**

Manual interruption was executed in a block that applies the G06.2 mode.

Remedy

- Execute the manual interruption in the block that does not apply the G06.2 mode.

P555 Invalid restart during G06.2**Details**

Restart was attempted from the block that applies G06.2 mode.

Remedy

- Restart from the block that does not apply the G06.2 mode.

P600 No spec: Auto TLM**Details**

An automatic tool length measurement command (G37) was issued though it is out of specifications.

Remedy

- Check the specifications.

P601 No spec: Skip**Details**

A skip command (G31) was issued though it is out of specifications.

Remedy

- Check the specifications.

P602 No spec: Multi skip**Details**

A multiple skip command (G31.1, G31.2, G31.3 or G31 Pn) 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 illegal axis**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 & TLM command in a 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 T cmdnd not found before TLM**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 illegal signal**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 during radius compen**Details**

A skip command was issued during radius compensation processing.

Remedy

- Issue a radius compensation cancel (G40) command or remove the skip command.

P610 Illegal parameter**Details**

- The parameter setting is not correct.
- G114.1 was commanded while the spindle synchronization was selected with the PLC interface signal.
- G110 was commanded while the mixed control (cross axis control) was selected with the PLC interface signal.
- G125 was commanded while the control axis synchronization across part systems was selected with the PLC interface signal.
- G126 was commanded while the control axis superimposition was selected with the PLC interface signal.

Remedy

- Correct the settings of "#1514 expLinax (Exponential function interpolation linear axis)" and "#1515 expRotax (Exponential function interpolation rotary axis)".
- Correct the program.
- Correct the parameter settings.

P611 No spec: Exponential function**Details**

There is no specification for the exponential interpolation.

Remedy

- Check the specifications.

P612 Exponential function error**Details**

A travel command for exponential interpolation was issued during mirror image for facing tool posts.

Remedy

- Correct the program.

P700 Illegal command value**Details**

Spindle synchronization was commanded to a spindle that is not connected serially.

Remedy

- Correct the program.
- Correct the parameter settings.

P900 No spec: Normal line control**Details**

A normal line control command (G40.1, G41.1, or G42.1) was issued though it is out of specifications.

Remedy

- Check the specifications.

P901 Normal line control axis G92**Details**

A coordinate system preset command (G92) was issued to a normal line control axis during normal line control.

Remedy

- Correct the program.

P902 Normal line control axis error**Details**

- The normal line control axis was set to a linear axis.
- The normal line control axis was set to the linear type rotary axis II axis.
- The normal line control axis has not been set.
- The normal line control axis is the same as the plane selection axis.

Remedy

- Correct the normal line control axis setting.

P903 Plane chg in Normal line ctrl

Details

The plane selection command (G17, G18, or G19) was issued during normal line control.

Remedy

- Delete the plane selection command (G17, G18, or G19) from the program of the normal line control.

P920 No spec: 3D coord conv

Details

There is no specification for 3-dimensional coordinate conversion.

Remedy

- Check the specifications.

P921 Illegal G code at 3D coord

Details

The commanded G code cannot be performed during 3-dimensional coordinate conversion modal.

Remedy

- Refer to "MITSUBISHI CNC 700/70 Series Programming Manual (Machining Center System)" for usable G commands.
- When the parameter "#8158 Init const sur spd" is enabled, disable the parameter or issue the constant surface speed control cancel (G97) command.

P922 Illegal mode at 3D coord

Details

A 3-dimensional coordinate conversion command was issued during a modal for which 3-dimensional coordinate conversion cannot be performed.

Remedy

- Refer to "MITSUBISHI CNC 700/70 Series Programming Manual (Machining Center System)" for usable G commands.

P923 Illegal addr in 3D coord blk

Details

A G code and G68 was commanded in a block though the G code cannot be commanded with G68.

Remedy

- Refer to "MITSUBISHI CNC 700/70 Series Programming Manual (Machining Center System)" for usable G commands.

P930 No spec: Tool axis compen

Details

A tool length compensation along the tool axis command was issued though it is out of specifications.

Remedy

- Check the specifications.

P931 Executing tool axis compen

Details

There is a G code that cannot be commanded during tool length compensation along the tool axis.

Remedy

- Correct the program.

P932 Rot axis parameter error

Details

There is an illegal linear axis name or rotary axis name set in the rotary axis configuration parameters. There is an illegal setting in the parameter concerning the configuration of the inclined surface machining axis.

Remedy

- Set the correct value and turn the power ON again.

P940 No spec: Tool tip control**Details**

There is no specification for tool tip center control.

Remedy

- Check the specifications.

P941 Invalid T tip control command**Details**

A tool tip center control command was issued during a modal for which a tool tip center control command cannot be issued.

Remedy

- Correct the program.

P942 Invalid cmd during T tip ctrl**Details**

A G code that cannot be commanded was issued during tool tip center control.

Remedy

- Correct the program.

P943 Tool posture command illegal**Details**

In tool tip center control type 1, if the signs at the tool-side rotary axis or table base-side rotary axis start and finish points differ, a tool base-side rotary axis or table workpiece-side rotary axis rotation exists for the same block, and does not pass a singular point.

In tool tip center control type 2, the posture vector command is incorrect.

Remedy

- Correct the program.

P950 No spec: Tilt face machining**Details**

Inclined surface machining option is not supported.

Remedy

- Check the specifications.

P951 Ill cmd in tilt face machining**Details**

A forbidden command (G command, etc) was issued during inclined surface machining.

Remedy

- Check the program. If you want to execute a command (G command, etc) that is unavailable during inclined surface machining, cancel the inclined surface machining.

P952 Inclined face cut prohibited**Details**

Inclined surface machining was commanded during the mode where the machining is unavailable. Inclined surface machining was commanded during interruption.

Remedy

- Check the program and see whether any unavailable mode is included during inclined surface machining command. If any, cancel that mode.

P953 Tool axis dir cntrl prohibited**Details**

Tool axis direction control was commanded during the mode where the control is unavailable.

Remedy

- Check the program and see whether any unavailable mode is included during tool axis direction control. If any, cancel that mode.

P954 Inclined face command error

Details

The address to issue the inclined surface machining command is incorrect.

Remedy

- Check the program.

P955 Inclined face coord illegal

Details

Impossible to define an inclined surface with the values you specified.

Remedy

- Check the program.

P956 G68.2P10 surface not defined

Details

The coordinate system for the machining surface selected with G68.2P10 has not been defined.

Remedy

- Set the machining surface so that the coordinate system can be defined.

P957 Tool axis dir ctrl cmp amt 0

Details

When the tool axis direction control type 2 (G53.6) was commanded, a tool length compensation No. whose compensation amount is 0 was commanded.

Remedy

- Correct the program. Set the tool length compensation amount, or command a tool length compensation No. whose compensation amount is not 0.

P960 No spec: Direct command mode

Details

G05 P4 was commanded while direct command mode option is OFF.

Remedy

- Check the specifications.

P961 Invalid during dir cmdnd mode

Details

- A G code other than G05 P0 was commanded in direct command mode.
- A sequence No. command, F code command, MSTB command or variable command was issued.
- A corner chamfering command or corner R command was issued.
- A travel command was issued to an axis that had not been command in the G05 P4 block.

Remedy

- Check the program.

P962 Dir cmdnd mode cmdnd invalid

Details

G05 P4 was commanded in a modal where direct command mode is not available.

Remedy

- Check the program.

P963 Illegal direct cmdnd mode cmdnd

Details

The commanded coordinate value was beyond the maximum travel distance in direct command mode.

Remedy

- Correct the coordinate value in direct command mode.

P990 PREPRO error**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.

Appendix 2

Order of G Function Command Priority

Appendix 2 Order of G Function Command Priority

Upper level: When commanded in the same block indicates that both commands are executed simultaneously
 Lower level: When commanded during modal
 Command in a separate block when possible

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	<input type="radio"/> <input type="radio"/> Also possible during arc modal	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	Arc and G41 to G42 cause error P151 <input type="radio"/> Tool radius is compensated, and then axes move.	Arc and G43 to G49 cause error P70 <input type="radio"/> The G49 movement in the arc modal moves with G01
G04 Dwell	<input type="radio"/> Group 1 modal is updated G04 is executed <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	G04 is executed G40 to G42 are ignored (Note) <input type="radio"/>	G04 is executed G43 to G49 are ignored (Note) <input type="radio"/>
G09 Exact stop check	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>
G10, G11 Programmable data setting	<input type="radio"/> G10 is priority for axis No movement I, J, K rotation input <input type="radio"/>	<input type="radio"/> G10 is used for axis, so the selected plan axis will be the basic axis. <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	G10 to G11 are executed G40 to G42 are ignored <input type="radio"/>	G10 to G11 are executed G43 to G49 are ignored <input type="radio"/>
G17 to G19 Plane Selection	<input type="radio"/> <input type="radio"/>	<input type="radio"/> G command commanded last is valid. <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> Plane axis changeover during radius compensation causes error P112	<input type="radio"/> <input type="radio"/>
G20, G21 Inch/metric changeover	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	Possible in same block <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>
G27 to G30 Reference position compare/ return	<input type="radio"/> G00 to G03 modals are updated G27 to G30 are executed <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	G27 to G30 are executed G40 to G42 are ignored (Note) <input type="radio"/>	G27 to G30 are executed G43 to G49 are ignored (Note) <input type="radio"/>
G31 to G31.3 Skip	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	<input type="radio"/> <input type="radio"/>	Error:P608 Error:P608	<input type="radio"/> <input type="radio"/>

(Note) A program error (P45) will occur if they are commanded in the same block. This error can be avoided by setting the parameter "#1241 bit0 (No G-CODE COMB. Error)" to "1", but be aware that one of the G commands is ignored.

Commanded G code	G Group						
	09 G70 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 G70 to G89 except G80 are ignored(Note) ○	○ ○	○ G04 is executed Group 12 is changed ○	○ ○	○	○ ○	G04 is executed G50.1 and G51.1 are ignored (Note) ○
G09 Exact stop check	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G10, G11 Programmable data setting	G10 to G11 are executed G70 to G89 except G80 are ignored ○	○ ○	○ G10 is executed G54 to G59 modals are updated ○	○ ○	G66 to G67 are executed G10 is ignored ○	○ ○	G10 to G11 are executed G50.1 and G51.1 are ignored ○
G17 to G19 Plane Selection	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
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	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○

Appendix 2 Order of G Function Command Priority

Upper level: When commanded in the same block indicates that both commands are executed simultaneously
 Lower level: When commanded during modal
 Command in a separate block when possible

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
G33 Thread cutting	G command commanded last is valid. ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G37 Automatic tool length measurement	G37 is executed G00 to G33 are ignored(Note) ○	○ ○	○ ○	○ ○	○ ○	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 to G42 cause error P151 G41 to 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 Tool 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	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○

(Note) A program error (P45) will occur if they are commanded in the same block. This error can be avoided by setting the parameter "#1241 bit0 (No G-CODE COMB. Error)" to "1", but be aware that one of the G commands is ignored.

Commanded G code	G Group						
	09 G70 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
G33 Thread cutting	Group 1 command is executed Group 9 is canceled ○	○ ○	○ ○	○ ○	G66 to G67 are executed G33 modal is updated ○	○ ○	○ ○
G37 Automatic tool length measurement	G37 is executed G70 to G89 except G80 are ignored (Note) ○	○ ○	○ ○	○ ○	G66 to G67 are executed G37 is ignored(Note) ○	○ ○	G37 is executed G50.1 and G51.1 are ignored(Note) ○
G40 to G42.2 Tool radius compensation	Error:P155 Error:P155	○ ○	○ ○	○ ○	○ ○	○ ○	○ ○
G43, G44, G49 Tool 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 G70 to G89 except G80 are ignored(Note) ○	○ ○	○ ○	○ ○	○	○ ○	G52 is executed G50.1 and G51.1 are ignored(Note) ○
G53 Machine coordinate system	G53 is executed G70 to G89 except G80 are ignored (Note) ○	○ ○	○ ○	○ ○	○	○ ○	G53 is executed G50.1 and G51.1 are ignored(Note) ○
G54 to G59 Workpiece coordinate system	○ ○	○ ○	G command commanded last is valid. ○	○ ○	G66 to G67 are executed G54 and G59 modals are updated ○	○ ○	○ ○

Appendix 2 Order of G Function Command Priority

Upper level: When commanded in the same block indicates that both commands are executed simultaneously
 Lower level: When commanded during modal
 Command in a separate block when possible

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
G61 to G64 Mode selection	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G65 Macro call	G65 is executed G00 to G03 modals are updated <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G65 is executed G43 to G49 modals are updated <input type="radio"/>
G66 to G67 Macro call	G66 to G67 are executed G00 to G03 modals are updated <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/> G66 to G67 are executed G43 to G49 modals are updated <input type="radio"/>
G73 to G89 Fixed cycles	G73 to G89 are canceled G01 to G33 modals are updated <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Error:P155 Fixed cycle during compensation Error:P155	<input type="radio"/>
G90, G91 Absolute value/ incremental value	<input type="radio"/>	<input type="radio"/>	Use in same block <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G92 Coordinate system setting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G94, G95 Synchronous/ asynchronous	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G command commanded last is valid. <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G96, G97 Constant surface speed control	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G98, G99 Fixed cycle Initial point/ R point return	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

(Note) A program error (P45) will occur if they are commanded in the same block. This error can be avoided by setting the parameter "#1241 bit0 (No G-CODE COMB. Error)" to "1", but be aware that one of the G commands is ignored.

Commanded G code	G Group						
	09 G70 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
G61 to G64 Mode selection	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G command commanded last is valid. <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G65 Macro call	G65 is executed G70 to G89 except G80 are ignored <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Error <input type="radio"/>	<input type="radio"/>	G65 is executed G50.1 and G51.1 are ignored <input type="radio"/>
G66 to G67 Macro call	G66 to G67 are executed G70 to G89 except G80 are ignored <input type="radio"/>	<input type="radio"/>	G66 to G67 are executed G54 and G59 modals are updated <input type="radio"/>	<input type="radio"/>	G command commanded last is valid. ↑ <input type="radio"/>	<input type="radio"/>	G66 to G67 are executed G50.1 and G51.1 are ignored <input type="radio"/>
G73 to G89 Fixed cycles	G command commanded last is valid. <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G66 to G67 are executed G73 to G89 are ignored <input type="radio"/>	<input type="radio"/>	<input type="radio"/> All axes become mirror center <input type="radio"/>
G90, G91 Absolute value/ incremental value	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G92 Coordinate system setting	G92 is executed G70 to G89 except G80 are ignored(Note) <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/> Note that G92 is priority for axis <input type="radio"/>
G94, G95 Synchronous/ asynchronous	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
G96, G97 Constant surface speed control	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	G command commanded last is valid. <input type="radio"/>	<input type="radio"/>
G98, G99 Fixed cycle Initial point/ R point return	<input type="radio"/>	G command commanded last is valid. <input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Index

Numbers

2nd, 3rd, and 4th Reference Position (Zero point)
Return ; G30..... 398

A

Acceleration/Deceleration Mode Change in
Hole Drilling Cycle 234
Actual Examples of Using User Macros..... 328
Arc ; G36..... 239
ASCII Code Macro 263
Automatic Coordinate System Setting 389
Automatic Corner Override ; G62..... 100
Automatic Tool Length Measurement ; G37 420

B

Back Boring ; G87 215
Basic Machine Coordinate System Selection ;
G53..... 391
Basic Machine, Workpiece and
Local Coordinate Systems 387
Bolt Hole Cycle ; G34..... 237
Boring ; G85 213
Boring ; G86 214
Boring ; G88 217
Boring ; G89 218

C

Changing of Compensation No. during
Compensation Mode 168
Circular Cutting ; G12,G13..... 359
Circular Cutting ; G75..... 222
Circular Interpolation ; G02,G03 43
Common Variables 269
Compensation Data Input by Program ; G10 185
Constant Lead Thread Cutting ; G33 52
Constant Surface Speed Control ; G96,G97..... 119
Control Commands 319
Coordinate Rotation by Program ; G68/G69 378
Coordinate Rotation Parameter 293
Coordinate System for Rotary Axis 415
Coordinate System Setting ; G92..... 392
Coordinate Systems and Coordinate
Zero Point Symbols 3
Coordinate Words and Control Axes..... 2
Coordinate Words and Control Axes..... 386
Corner Chamfering Expansion/
Corner Rounding Expansion 346
Corner Chamfering Expansion/
Corner Rounding Expansion 340
Corner Chamfering I ; G01 X_ Y_ ,C_ 336
Corner Chamfering I / Corner Rounding I 336
Corner Chamfering II ; G01/G02/G03 X_ Y_ ,C_... 343
Corner Chamfering II / Corner Rounding II 343
Corner Rounding I ; G01 X_ Y_ ,R_..... 338
Corner Rounding II ; G01/G02/G03 X_ Y_ ,R_..... 345
Cutting Feed Constant Inclination
Acceleration/Deceleration 84

Cutting Feedrate 67
Cutting Mode ; G64..... 107

D

Deceleration Check 96
Decimal Point Input 29
Deep Hole Drilling Cycle ; G83..... 204
Deep Hole Drilling Cycle..... 204
Detailed Description for Macro Call Instruction ... 261
Drilling Cycle with High-Speed Retract..... 230
Drilling, Counter Boring ; G82..... 203
Drilling, Spot Drilling ; G81..... 202
Dwell (Time Designation) ; G04..... 110

E

Exact Stop Check ; G09 91
Exact Stop Check Mode ; G61 95
External Output Commands ;
POPEN, PCLOS, DPRNT 322

F

F1-digit Feed..... 68
Feed Hold, Feedrate Override,
G09 Valid/Invalid (#3004) 286
Feed Per Minute/Feed Per Revolution
(Asynchronous Feed/Synchronous Feed) ;
G94,G95 71
Feedrate Designation and Effects on
Control Axes 74
File Format 14
Fine Boring ; G76..... 224
Fixed Cycles 198

G

G code 18
G Code Lists..... 18
G Code Macro Call 259
G Command Mirror Image ; G50.1,G51.1 332
G Command Modals
(#4001-#4021, #4201-#4221) 288
G1 -> G0 Deceleration Check 98
G1 -> G1 Deceleration Check 99
G41/G42 Commands and I, J, K Designation 158
General Precautions for
Tool Radius Compensation..... 167
Geometric 348
Geometric I ; G01 A_..... 348
Geometric IB 350
Geometric IB (Automatic calculation of
linear - arc intersection) ;
G01 A_ , G02/G03 R_ H_ 356
Geometric IB (Automatic calculation of
linear - arc intersection) ;
G01 A_ , G02/G03 P_ Q_ H_ 353
Geometric IB (Automatic calculation of
two-arc contact) ; G02/G03 P_ Q_ /R_ 351
Grid ; G37.1 240

H

Helical Interpolation ; G17 to G19, G02, G03 58

I

Inch Thread Cutting ; G33 56
Inch/Metric Conversion ; G20,G21 28
Initial Point and R Point Level Return ;
G98,G99..... 228
Input Command Increment Tenfold 7
Input Setting Unit 6
Inputting the Tool Life Management Data ;
G10,G11..... 190
Inputting the Tool Life Management Data by
G10 L3 Command ; G10 L3,G11 190
Inputting the Tool Life Management Data by
G10 L30 Command ; G10 L30,G11 192
Integrating Time (#3001, #3002) 285
Interference Check 173
Interrupt during Corner Chamfering/
Interrupt during Corner Rounding 346
Interrupt during Corner Chamfering/
Interrupt during Corner Rounding 342
Interrupts during Tool Radius Compensation 164

L

Line at Angle ; G35 238
Linear Angle Command ; G01 X_/Y_ A_/A_ 347
Linear Interpolation ; G01 41
Local Coordinate System Setting ; G52..... 411
Local Variables (#1 to #33) 270

M

Machine Zero Point and 2nd, 3rd, 4th
Reference Position (Zero point) 388
Macro Call Instruction 251
Macro Interface Inputs/Outputs
(#1000 to #1035, #1100 to #1135,
#1200 to #1295, #1300 to #1395) 275
Macro Interruption ; M96,M97 363
Message Display and Stop (#3006) 286
Mirror Image (#3007) 287
Miscellaneous Command Macro Call
(for M, S, T, B Code Macro Call) 260
Miscellaneous Functions (M8-digits) 114
Modal Call A (Movement Command Call) ;
G66 255
Modal Call B (for each block) ; G66.1 257
Modal, unmodal 18
Multi-step Skip Function 1 ; G31.n ,G04 429
Multi-step Skip Function 2 ; G31 P 431

N

NC Alarm (#3000) 284
Number of Workpiece Machining Times
(#3901, #3902) 293

O

Operation Commands314
Optional Block Skip 16
Optional Block Skip; / 16
Other Commands and Operations during
Tool Radius Compensation 148
Other Modals (#4101 - #4120, #4301 - #4320)289

P

Parameter Input by Program ;
G10 L70/L100, G11361
Plane Selection ; G17,G18,G1950
Position Command Methods ; G90,G9126
Position Information (#5001 - #5160 + n)290
Positioning (Rapid Traverse) ; G0034
Precautions326
Precautions Before Starting Machining21
Precautions for Inputting the
Tool Life Management Data 195
Precautions for Using a Fixed Cycle226
Pre-read Buffers24
Program Format10
Programmable Current Limitation ; G10 L14 ;433

R

R Device Access Variables
(#50000 - #50749, #51000 - #51749,
#52000 - #52749)308
R Specification Circular Interpolation ; G02,G03.....48
Rapid Traverse Constant Inclination
Acceleration/Deceleration79
Rapid Traverse Rate66
Reading PLC data
(#100100-#100103,#100110)302
Reading The Parameters
(#100000-#100002, #100010)299
Reference Position (Zero point) Return ;
G28,G29394
Reference Position Check ; G27401
Reverse Tapping Cycle ; G74220

S

Scaling ; G50/G51374
Secondary Miscellaneous Functions
(A8-digits, B8-digits or C8-digits) 116
Setting of Workpiece Coordinates in
Fixed Cycle Mode229
Simple Macro Calls ; G65.....251
Skip Function ; G31424
Special Fixed Cycle236
Spindle Clamp Speed Setting ; G92..... 121
Spindle Functions118
Spindle/C Axis Control 123
Start of Tool Radius Compensation and
Z Axis Cut in Operation..... 171
Stepping Cycle ; G73219
Subprogram Call ; M198247
Subprogram Call ; M98,M99241
Subprogram Control; M98, M99, M198241

Suppression of Single Block Stop and
Miscellaneous Function Finish Signal Waiting
(#3003) 285

T

Tapping Cycle ; G84 205
Tapping Mode ; G63 106
Thread Cutting 52
Time Reading Variables
(#3001, #3002, #3011, #3012) 306
Tool Change Position Return ; G30.1 - G30.6 371
Tool Compensation 282
Tool Compensation 130
Tool Functions (T8-digit BCD) 128
Tool Length Compensation/Cancel ;
G43,G44/G49 134
Tool Life Management (#60000 - #64700) 294
Tool Radius Compensation ;
G38,G39/G40/G41,G42 138
Tool Radius Compensation Operation 139
Types of Variables 269

U

Unidirectional Positioning ; G60 64
User Macro..... 250

V

Variable 267
Variable Commands 248

W

Workpiece Coordinate Changing during
Radius Compensation 183
Workpiece Coordinate System Compensation
(#5201 - #532n)..... 283
Workpiece Coordinate System Setting and Offset ;
G54 to G59 (G54.1)..... 402

Revision History

Date of revision	Manual No.	Revision details
Mar. 2013	IB(NA)1501200-A	First edition created.
Mar. 2014	IB(NA)1501200-B	<p>- Corrections and additions are made corresponding to S/W version K1.</p> <p>[Added] 13.3.2 Subprogram Call ; M198 13.7.3 Corner Chamfering Expansion/Corner Rounding Expansion 13.7.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding 13.8 Corner Chamfering II/Corner Rounding II [Changed the chapter title] 13.3 Subprogram Control ; M98,M99,M198 [Changed] 3.1 Program Format 3.4.2 G Code Lists 6.6.1 Constant Lead Thread Cutting ; G33 6.6.2 Inch Thread Cutting ; G33 7.6 Rapid Traverse Constant Inclination Acceleration/Deceleration 12.3.4 Interrupts during Tool Radius Compensation 12.3.8 Interference Check 12.4 Compensation Data Input by Program ; G10 12.5.1 Inputting the Tool Life Management Data by G10 L3 Command ; G10 L3,G11 12.5.2 Inputting the Tool Life Management Data by G10 L30 Command ; G10 L30,G11 12.5.3 Precautions for Inputting the Tool Life Management Data 13.1.4 Tapping Cycle ; G84 13.1.11 Reverse Tapping Cycle ; G74 13.1.14 Precautions for Using a Fixed Cycle 13.3.1 Subprogram Call ; M98,M99 13.5.2.1 Simple Macro Calls ; G65 13.5.2.2 Modal Call A (Movement Command Call) ; G66 13.5.2.3 Modal Call B (for each block) ; G66.1 13.5.5.4 Tool Compensation 13.5.5.12 G Command Modals (#4001-#4021, #4201-#4221) 13.5.5.13 Other Modals (#4101 - #4120, #4301 - #4320) 13.5.5.17 Tool Life Management (#60000 - #64700) 13.7.1 Corner Chamfering I ; G01 X_ Y_ ,C_ 13.7.2 Corner Rounding I ; G01 X_ Y_ ,R_ 13.12 Parameter Input by Program ; G10 L70/L100, G11 13.13 Macro Interruption ; M96,M97 14.10 Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1) 15.5 Programmable Current Limitation ; G10 L14 ;</p> <p>- According to the addition of chapters, some chapter numbers are changed. - Corrected the mistakes.</p>

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: +1-847-478-2650

Ohio Service Satellite
LIMA, OHIO 45901, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650
CINCINNATI, OHIO 45201, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

Minnesota Service Satellite
ROGERS, MINNESOTA 55374, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

West Region Service Center
16900 VALLEY VIEW AVE., LAMIRADA, CALIFORNIA 90638, U.S.A.
TEL: +1-714-699-2625 / FAX: +1-847-478-2650

Northern CA Satellite
SARATOGA, CALIFORNIA 95070, U.S.A.
TEL: +1-714-699-2625 / FAX: +1-847-478-2650

Pennsylvania Service Satellite
PITTSBURG, PENNSYLVANIA 15644, U.S.A.
TEL: +1-732-560-4500 / FAX: +1-732-560-4531

Connecticut Service Satellite
TORRINGTON, CONNECTICUT 06790, U.S.A.
TEL: +1-732-560-4500 / FAX: +1-732-560-4531

South Region Service Center
1845 SATTELITE BOULEVARD STE. 450, DULUTH, GEORGIA 30097, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Texas Service Satellites
GRAPEVINE, TEXAS 76051, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519
HOUSTON, TEXAS 77001, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Tennessee Service Satellite
Nashville, Tennessee, 37201, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Florida Service Satellite
WEST MELBOURNE, FLORIDA 32904, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Canada Region Service Center
4299 14TH AVENUE MARKHAM, ONTARIO L3R 0J2, CANADA
TEL: +1-905-475-7728 / FAX: +1-905-475-7935

Canada Service Satellite
EDMONTON, ALBERTA T5A 0A1, CANADA
TEL: +1-905-475-7728 / FAX: +1-905-475-7935

Mexico Region Service Center
MARIANO ESCOBEDO 69 TLALNEPANTLA, 54030 EDO. DE MEXICO
TEL: +52-55-3067-7500 / FAX: +52-55-9171-7649

Monterrey Service Satellite
MONTERREY, N.L., 64720, MEXICO
TEL: +52-81-8365-4171

BRAZIL

MELCO CNC do Brasil Comércio e Serviços S.A

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

EUROPE

MITSUBISHI ELECTRIC EUROPE B.V.
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: +49-711-770598-123 / FAX: +49-711-770598-141

France Service Center DEPARTEMENT CONTROLE NUMERIQUE
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 DEPARTEMENT CONTROLE NUMERIQUE
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 - CENTRO DIREZIONALE COLLEONI PALAZZO SIRIO INGRESSO 1
20864 AGRATE BRIANZA (MB), ITALY
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

Italy (Padova) Service Satellite
VIA G. SAVELLI, 24 - 35129 PADOVA, ITALY
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

U.K. Branch
TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.
TEL: +49-2102-486-0 / FAX: +49-2102-486-5910

Spain Service Center
CTRA. DE RUBI, 76-80-APDO. 420
08173 SAINT CUGAT DEL VALLES, BARCELONA SPAIN
TEL: +34-935-65-2236 / FAX: +34-935-65-1579

Poland Service Center
UL. KRAKOWSKA 50, 32-083 BALICE, POLAND
TEL: +48-12-630-4700 / FAX: +48-12-630-4701

Mitsubishi Electric Turkey A.Ş Ümraniye Şubesi
Turkey Service Center
ŞERİFALİ MAH. NUTUK SOK. NO.5 34775
ÜMRANIYE, İSTANBUL, TURKEY
TEL: +90-216-526-3990 / FAX: +90-216-526-3995

Czech Republic Service Center
KAFKOVA 1853/3, 702 00 OSTRAVA 2, CZECH REPUBLIC
TEL: +420-59-5691-185 / FAX: +420-59-5691-199

Russia Service Center
213, B.NOVOODMITROVSKAYA STR., 14/2, 127015 MOSCOW, RUSSIA
TEL: +7-495-748-0191 / FAX: +7-495-748-0192

MITSUBISHI ELECTRIC EUROPE B.V. (SCANDINAVIA)
Sweden Service Center
HAMMARBACKEN 14 191 49 SOLLENTUNA, SWEDEN
TEL: +46-8-6251000 / FAX: +46-8-966877

Bulgaria Service Center
4 A.LYAPCHEV BOUL., POB 21, BG-1756 SOFIA, BULGARIA
TEL: +359-2-8176009 / FAX: +359-2-9744061

Ukraine (Kharkov) Service Center
APTEKARSKIY LANE 9-A, OFFICE 3, 61001 KHARKOV, UKRAINE
TEL: +380-57-732-7774 / FAX: +380-57-731-8721

Ukraine (Kiev) Service Center
4-B, M. RASKOVOYI STR., 02660 KIEV, UKRAINE
TEL: +380-44-494-3355 / FAX: +380-44-494-3366

Belarus Service Center
OFFICE 9, NEZAVISIMOSTI PR.177, 220125 MINSK, BELARUS
TEL: +375-17-393-1177 / FAX: +375-17-393-0081

South Africa Service Center
5 ALBATROSS STREET, RHODESFIELD, KEMPTON PARK 1619, GAUTENG, 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

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
17 & 17A, JALAN IMPIAN EMAS 5/5, TAMAN IMPIAN EMAS, 81300 SKUDAI, JOHOR MALAYSIA.
TEL: +60-7-557-8218 / FAX: +60-7-557-3404

Philippines Service Center
UNIT NO.411, ALABANG CORPORATE CENTER KM 25, WEST SERVICE ROAD
SOUTH SUPERHIGHWAY, ALABANG MUNTINLUPA METRO MANILA, PHILIPPINES 1771
TEL: +63-2-807-2416 / FAX: +63-2-807-2417

VIETNAM**MITSUBISHI ELECTRIC VIETNAM CO.,LTD**

Vietnam (Ho Chi Minh) Service Center
UNIT 01-04, 10TH FLOOR, VINCOM CENTER 72 LE THANH TON STREET, DISTRICT 1,
HO CHI MINH CITY, VIETNAM
TEL: +84-8-3910 5945 / FAX: +84-8-3910 5946

Vietnam (Hanoi) Service Satellite
SUITE 9-05, 9TH FLOOR, HANOI CENTRAL OFFICE BUILDING, 44B LY THUONG KIET STREET,
HOAN KIEM DISTRICT, HANOI CITY, VIETNAM
TEL: +84-4-3937-8075 / FAX: +84-4-3937-8076

INDONESIA**PT. MITSUBISHI ELECTRIC INDONESIA**

Indonesia Service Center
GEDUNG JAYA 11TH FLOOR, JL. MH. THAMRIN NO.12, JAKARTA PUSAT 10340, INDONESIA
TEL: +62-21-3192-6461 / FAX: +62-21-3192-3942

THAILAND**MITSUBISHI ELECTRIC FACTORY AUTOMATION (THAILAND) CO.,LTD**

Thailand Service Center
12TH FLOOR, SV.CITY BUILDING, OFFICE TOWER 1, NO. 896/19 AND 20 RAMA 3 ROAD,
KWAENG BANGPONGPANG, KHET YANNAWA, BANGKOK 10120, THAILAND
TEL: +66-2-682-6522-31 / FAX: +66-2-682-6020

INDIA**MITSUBISHI ELECTRIC INDIA PVT. LTD.**

India Service Center
2nd FLOOR, TOWER A & B, DLF CYBER GREENS, DLF CYBER CITY,
DLF PHASE-III, GURGAON 122 002, HARYANA, INDIA
TEL: +91-124-4630 300 / FAX: +91-124-4630 399

Ludhiana satellite office
Jamshedpur satellite office

India (Pune) Service Center
EMERALD HOUSE, EL-3, J-BLOCK, MIDC BHOSARI, PUNE – 411 026, MAHARASHTRA, INDIA
TEL: +91-20-2710 2000 / FAX: +91-20-2710 2100

Baroda satellite office
Mumbai satellite office

India (Bangalore) Service Center
PRESTIGE EMERALD, 6TH FLOOR, MUNICIPAL NO. 2,
LAVELLE ROAD, BANGALORE - 560 043, KAMATAKA, INDIA
TEL: +91-80-4020-1600 / FAX: +91-80-4020-1699

Chennai satellite office
Coimbatore satellite office

OCEANIA**MITSUBISHI ELECTRIC AUSTRALIA LTD.**

Australia 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
1-3.5-10,18-23/F, NO.1386 HONG QIAO ROAD, CHANG NING QU,
SHANGHAI 200336, CHINA
TEL: +86-21-2322-3030 / FAX: +86-21-2308-3000

China (Ningbo) Service Dealer
China (Wuxi) Service Dealer
China (Jinan) Service Dealer
China (Hangzhou) 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: +86-10-6518-8030

China (Beijing) Service Dealer

China (Tianjin) Service Center
UNIT 2003, TIANJIN CITY TOWER, NO 35 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
4F KT BUILDING, 1630 SANGYEOK-DONG, BUK-KU, DAEGU 702-835, KOREA
TEL: +82-53-382-7400 / FAX: +82-53-382-7411

TAIWAN**MITSUBISHI ELECTRIC TAIWAN CO., LTD. (TAIWAN FA CENTER)**

Taiwan (Taichung) Service Center (Central Area)
NO.8-1, INDUSTRIAL 16TH RD., TAICHUNG INDUSTRIAL PARK, SITUN DIST.,
TAICHUNG CITY 40768, TAIWAN R.O.C.
TEL: +886-4-2359-0688 / FAX: +886-4-2359-0689

Taiwan (Taipei) Service Center (North Area)
10F, NO.88, SEC.6, CHUNG-SHAN N. RD., SHI LIN DIST., TAIPEI CITY 11155, TAIWAN R.O.C.
TEL: +886-2-2833-5430 / FAX: +886-2-2833-5433

Taiwan (Tainan) Service Center (South Area)
11F-1., NO.30, ZHONGZHENG S. ROAD, YONGKANG DISTRICT, TAINAN CITY 71067, TAIWAN, R.O.C
TEL: +886-6-252-5030 / FAX: +886-6-252-5031

Notice

Every effort has been made to keep up with software and hardware revisions in the contents described in this manual. However, please understand that in some unavoidable cases simultaneous revision is not possible. Please contact your Mitsubishi Electric dealer with any questions or comments regarding the use of this product.

Duplication Prohibited

This manual may not be reproduced in any form, in part or in whole, without written permission from Mitsubishi Electric Corporation.

COPYRIGHT 2013-2014 MITSUBISHI ELECTRIC CORPORATION
ALL RIGHTS RESERVED

MITSUBISHI CNC



MODEL	E70 Series
MODEL CODE	100-342
Manual No.	IB-1501200