



Changes for the Better

MITSUBISHI CNC

Programming Manual (Lathe System)

C70






Introduction

This manual is a guide for using the C70.

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




Details described in this manual


CAUTION

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

Precautions for Safety





Always read the specifications issued by the machine tool builder, this manual, related manuals and attached documents before installation, operation, programming, maintenance or inspection to ensure correct use. Understand this numerical controller, safety items and cautions before using the unit. This manual ranks the safety precautions into "DANGER", "WARNING" and "CAUTION".

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











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

The meanings of the pictorial signs are given below.

The following signs indicate prohibition and compulsory.

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

The meaning of each pictorial 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

 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 (normally X axis for a lathe) moves toward the spindle center, the spindle rotation speed will increase and may exceed the allowable speed of the workpiece or chuck, etc. In this case, the workpiece, etc. may jump out during machining, which may result in breakage of tools or machine tool or may cause damage to the operators.**

CAUTION






1. Items related to product and manual

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

2. Items related to operation

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

3. Items related to programming

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

Disposal



(Note) This symbol mark is for EU countries only.

This symbol mark is according to the directive 2006/66/EC Article 20 Information for end-users and Annex II.

Your MITSUBISHI ELECTRIC product is designed and manufactured with high quality materials and components which can be recycled and/or reused.

This symbol means that batteries and accumulators, at their end-of-life, should be disposed of separately from your household waste.

If a chemical symbol is printed beneath the symbol shown above, this chemical symbol means that the battery or accumulator contains a heavy metal at a certain concentration.

This will be indicated as follows:

Hg: mercury (0,0005%), Cd: cadmium (0,002%), Pb: lead (0,004%)

In the European Union there are separate collection systems for used batteries and accumulators.

Please, dispose of batteries and accumulators correctly at your local community waste collection/recycling centre.

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

Trademarks

MELDAS, MELSEC, EZSocket, EZMotion, iQ Platform, MELSOFT, GOT, CC-Link, CC-Link/LT, 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®, 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.

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	4
2 Least Command Increments	5
2.1 Input Setting Unit	6
3 Program Formats	7
3.1 Program Format	8
3.2 Program/sequence/block numbers; O, N.....	12
3.3 Optional Block Skip	13
3.3.1 Optional Block Skip; /	13
3.3.2 Optional Block Skip Addition ; /n	14
3.4 G code	16
3.4.1 Modal, unmodal	16
3.4.2 G code Lists	16
3.4.3 Table of G Code Lists	17
3.5 Precautions Before Starting Machining	21
4 Pre-read Buffers	23
4.1 Pre-read Buffers	24
5 Position Commands	25
5.1 Absolute/Incremental Value Commands ; G90,G91.....	26
5.2 Radius/Diameter Designation	28
5.3 Inch/Metric Conversion ; G20,G21.....	29
5.4 Decimal Point Input	30
6 Interpolation Functions	35
6.1 Positioning (Rapid Traverse) ; G00	36
6.2 Linear Interpolation ; G01.....	43
6.3 Circular Interpolation ; G02,G03.....	46
6.4 R Specification Circular Interpolation ; G02,G03.....	50
6.5 Plane Selection ; G17,G18,G19.....	52
6.6 Thread Cutting.....	54
6.6.1 Constant Lead Thread Cutting ; G33.....	54
6.6.2 Inch Thread Cutting ; G33	58
6.6.3 Continuous Thread Cutting ; G33.....	60
6.6.4 Variable Lead Thread Cutting ; G34	61
6.7 Helical Interpolation ; G17,G18,G19 and G02,G03	63
7 Feed Functions	67
7.1 Rapid Traverse Rate	68
7.2 Cutting Feedrate	69
7.3 F1-digit Feed.....	70
7.4 Feed Per Minute/Feed Per Revolution (Asynchronous Feed/Synchronous Feed) ; G94,G95	72
7.5 Feedrate Designation and Effects on Control Axes	74
7.6 Thread Cutting Mode	79
7.7 Automatic Acceleration/Deceleration	80
7.8 Rapid Traverse Constant Inclination Acceleration/Deceleration	81
7.9 Speed Clamp	83
7.10 Exact Stop Check ; G09.....	84
7.11 Exact Stop Check Mode ; G61	88
7.12 Deceleration Check.....	89
7.13 Automatic Corner Override ; G62.....	93
7.14 Tapping Mode ; G63.....	99
7.15 Cutting Mode ; G64	100

8 Dwell	101
8.1 Dwell (Time Designation) ; G04	102
9 Miscellaneous Functions	105
9.1 Miscellaneous Functions (M8-digits)	106
9.2 Secondary Miscellaneous Functions (A8-digits, B8-digits or C8-digits)	108
10 Spindle Functions	109
10.1 Spindle Functions.....	110
10.2 Constant Surface Speed Control ; G96,G97.....	111
10.3 Spindle Clamp Speed Setting ; G92.....	113
10.4 Spindle/C Axis Control.....	115
10.5 Spindle Synchronization.....	120
10.5.1 Spindle Synchronization Control I ; G114.1	121
10.5.2 Spindle Synchronization Control II	132
10.5.3 Precautions for Using Spindle Synchronization Control	137
10.6 Multiple-spindle Control.....	140
10.6.1 Multiple-spindle Control I (spindle control command) ; S ○ =	141
10.6.2 Multiple-spindle Control I (spindle selection command) ; G43.1,G44.1.....	142
11 Tool Functions	145
11.1 Tool Functions (T8-digit BCD)	146
12 Tool Compensation Functions	147
12.1 Tool Compensation	148
12.1.1 Tool Compensation Start	149
12.2 Tool Length Compensation	150
12.3 Tool Nose Wear Compensation	152
12.4 Tool Nose R Compensation ; G40,G41,G42,G46	153
12.4.1 Tool Nose Point and Compensation Directions	155
12.4.2 Tool Nose Radius Compensation Operations	158
12.4.3 Other Operations during Tool Nose Radius Compensation	176
12.4.4 G41/G42 Commands and I, J, K Designation	184
12.4.5 Interrupts during Tool Nose Radius Compensation	188
12.4.6 General Precautions for Tool Nose Radius Compensation	191
12.4.7 Interference Check	192
12.5 Compensation Data Input by Program ; G10 L2/L10/L11, G11	198
12.6 Tool Life Management II ; G10 L3, G11	201
12.6.1 Counting the Tool Life	204
13 Program Support Functions	207
13.1 Fixed Cycles for Turning Machining	208
13.1.1 Longitudinal Cutting Cycle ; G77	209
13.1.2 Thread Cutting Cycle ; G78.....	212
13.1.3 Face Cutting Cycle ; G79.....	215
13.2 Compound Type Fixed Cycle for Turning Machining	218
13.2.1 Longitudinal Rough Cutting Cycle ; G71	219
13.2.2 Face Rough Cutting Cycle ; G72	224
13.2.3 Formed Material Rough Cutting Cycle ; G73.....	228
13.2.4 Finishing Cycle ; G70.....	232
13.2.5 Face Cut-Off Cycle ; G74	233
13.2.6 Longitudinal Cut-off Cycle ; G75	235
13.2.7 Compound Thread Cutting Cycle ; G76.....	237
13.2.8 Precautions for Compound Type Fixed Cycle for Turning Machining; G70 to G76	241
13.3 Fixed Cycle for Drilling	243
13.3.1 Face Deep Hole Drilling Cycle 1 (Longitudinal deep hole drilling cycle 1) ; G83 (G87)	246
13.3.2 Face Tapping Cycle (Longitudinal tapping cycle) ; G84 (G88)	248
13.3.3 Face Boring Cycle (Longitudinal boring cycle) ; G85 (G89)	253
13.3.4 Deep Hole Drilling Cycle 2 ; G83.2.....	254
13.3.5 Fixed Cycle for Drilling Cancel; G80	256
13.3.6 Precautions When Using a Fixed Cycle for Drilling	257
13.3.7 Initial Point and R Point Level Return ; G98,G99	258
13.4 Subprogram Control; M98, M99	259
13.4.1 Subprogram Call ; M98,M99.....	259
13.5 Variable Commands	265

13.6 User Macro.....	268
13.6.1 User Macro	268
13.6.2 Macro Call Instruction	269
13.6.2.1 Simple Macro Calls ; G65.....	269
13.6.2.2 Modal Call A (Movement Command Call) ; G66.....	272
13.6.2.3 Modal Call B (for each block) ; G66.1	273
13.6.2.4 G Code Macro Call	274
13.6.2.5 Miscellaneous Command Macro Call (for M, S, T, B Code Macro Call)	275
13.6.2.6 Detailed Description for Macro Call Instruction	276
13.6.3 Variable	277
13.6.4 Types of Variables	279
13.6.4.1 Common Variables	279
13.6.4.2 Local Variables (#1 to #33)	280
13.6.4.3 Macro Interface Inputs/Outputs	
(#1000 to #1035, #1100 to #1135, #1200 to #1295, #1300 to #1395)	283
13.6.4.4 Tool Compensation	292
13.6.4.5 Workpiece Coordinate System Compensation	
(External Workpiece Coordinate Offset) (#5201 - #532n).....	293
13.6.4.6 NC Alarm (#3000)	294
13.6.4.7 Integrating Time (#3001, #3002)	295
13.6.4.8 Suppression of Single Block Stop and Miscellaneous Function Finish Signal Waiting (#3003)	
.....	295
13.6.4.9 Feed Hold, Feedrate Override, G09 Valid/Invalid (#3004)	296
13.6.4.10 Message Display and Stop (#3006)	296
13.6.4.11 Mirror Image (#3007)	297
13.6.4.12 G Command Modals (#4001-#4021, #4201-#4221)	298
13.6.4.13 Other Modals (#4101 - #4140, #4301 - #4340)	299
13.6.4.14 Position Information (#5001 - #5100 + n)	300
13.6.4.15 Number of Workpiece Machining Times (#3901, #3902)	302
13.6.4.16 ZR device access variable	303
13.6.4.17 Tool Life Management (#60000 - #63016)	312
13.6.5 Operation Commands	316
13.6.6 Control Commands	321
13.6.7 Precautions	324
13.7 Mirror Image for Facing Tool Posts ; G68,G69.....	326
13.8 Corner Chamfering/Corner Rounding I	331
13.8.1 Corner Chamfering I ; G01 X_ Z_ ,C	331
13.8.2 Corner Rounding I ; G01 X_ Z_ ,R.....	333
13.8.3 Interrupt during Corner Chamfering/Corner Rounding	335
13.9 Corner Chamfering/Corner Rounding II	336
13.9.1 Corner Chamfering II ; G01/G02/G03 X_ Z_ ,C.....	337
13.9.2 Corner Rounding II ; G01/G02/G03 X_ Z_ ,R.....	339
13.9.3 Interrupt during Corner Chamfering/Corner Rounding	340
13.10 Geometric	341
13.10.1 Geometric I ; G01 A_	341
13.10.2 Geometric IB	343
13.10.2.1 Geometric IB (Automatic calculation of two-arc contact) ; G02/G03 P_ Q_ /R_.....	344
13.10.2.2 Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 P_ Q_ H_	
.....	348
13.10.2.3 Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 R_ H_	
.....	352
13.11 Programmable Parameter Input ; G10 L70, G11.....	356
13.12 Macro Interruption ; M96,M97	358
13.13 Tool Change Position Return ; G30.1 - G30.5.....	366
13.14 High-accuracy control ; G61.1	369
13.15 Coordinate Rotation by Program ; G68.1/G69.1.....	379
13.16 Balance Cut ; G15,G14.....	388
13.17 Timing Synchronization between Part Systems	392
13.17.1 Timing Synchronization between Part Systems ; InL.....	392
13.17.2 Start Point Designation Timing Synchronization (Type 1) ; G115	396
13.17.3 Start Point Designation Timing Synchronization (Type 2) ; G116	398
13.18 2-part System Simultaneous Thread Cutting Cycle.....	400
13.18.1 2-part System Simultaneous Thread Cutting Cycle Parameter Setting Command ; G76.....	400
13.18.2 2-part System Simultaneous Thread Cutting Cycle I ; G76.1	401
13.18.3 2-part System Simultaneous Thread Cutting Cycle II ; G76.2.....	403
13.19 Chopping ; G81.1	407

14 Coordinate System Setting Functions	413
14.1 Coordinate Words and Control Axes	414
14.2 Basic Machine, Workpiece and Local Coordinate Systems	416
14.3 Machine Zero Point and 2nd Reference Position (Zero point)	417
14.4 Automatic Coordinate System Setting	418
14.5 Basic Machine Coordinate System Selection ; G53	419
14.6 Coordinate System Setting ; G92	420
14.7 Reference Position (Zero point) Return ; G28,G29	421
14.8 2nd, 3rd, and 4th Reference Position (Zero point) Return ; G30	425
14.9 Reference Position Check ; G27	428
14.10 Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1)	429
14.11 Local Coordinate System Setting ; G52	434
14.12 Coordinate System for Rotary Axis	435
15 Protection Function	439
15.1 Chuck Barrier/Tailstock Barrier ; G22,G23	440
16 Measurement Support Functions	443
16.1 Automatic Tool Length Measurement ; G37	444
16.2 Skip Function ; G31	448
16.3 Multi-step Skip Function 1 ; G31.n , G04	453
16.4 Multi-step Skip Function 2 ; G31 P/L	456
16.5 Programmable Current Limitation ; G10 L14 ;	459
Appendix 1 Order of G Function Command Priority	461
Appendix 2 Program Errors	463



Control Axes

1.1 Coordinate Words and Control Axes



Function and purpose

In the case of a lathe, axis names (coordinate words) and directions are defined as follows.

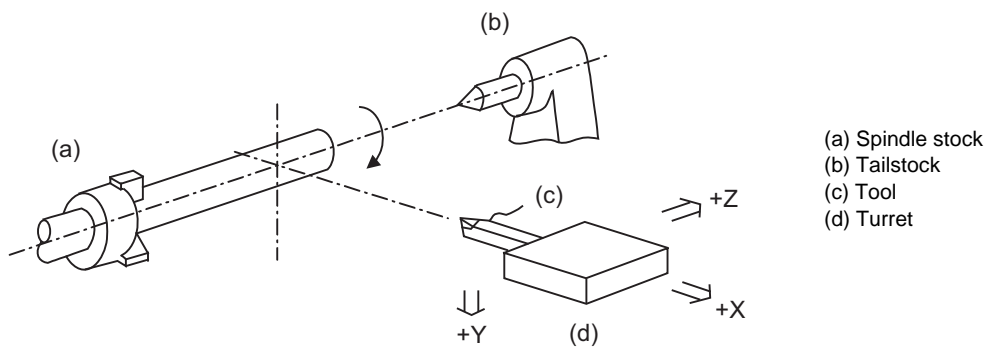
The axis at right angles to the spindle

Axis name: X axis

The axis parallel to the spindle

Axis name: Z axis

Coordinate axes and polarities

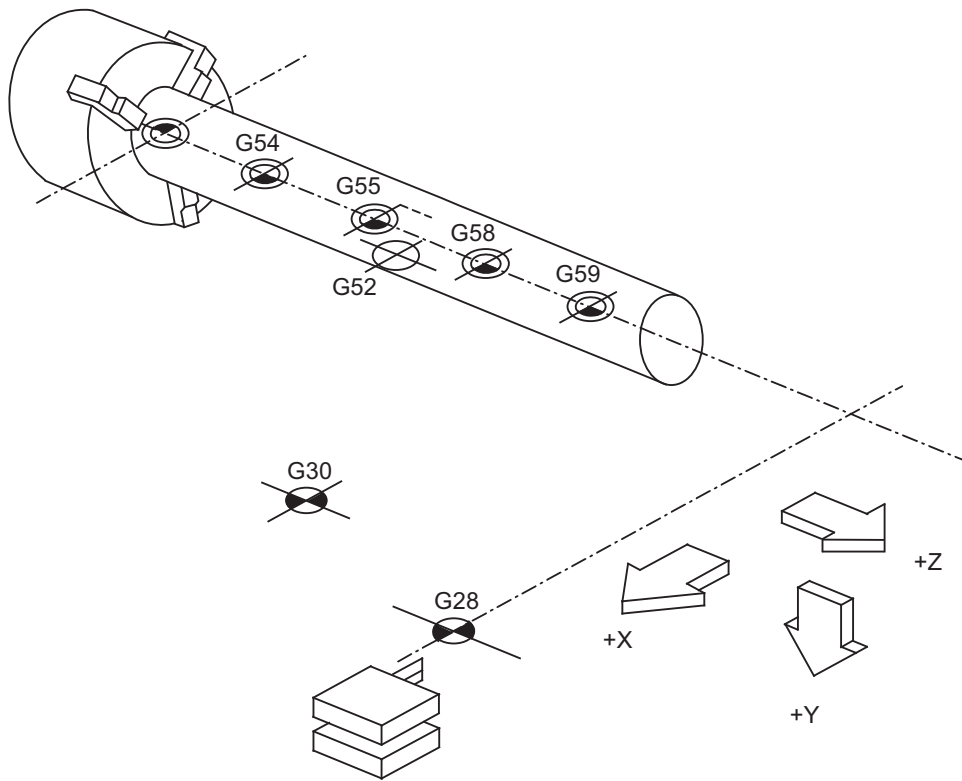






Since coordinates based on the right hand rule are used with a lathe, in the above figure, the positive direction of the Y axis which is at right angles to the X-Z plane is downward.

Note that a circular on the X-Z plane is expressed as clockwise or counterclockwise as seen from the forward direction of the Y axis.




(Refer to the section on circular interpolation.)

Relationship between coordinates



-  Reference position
-  Basic machine coordinate
-  Workpiece coordinate zero points
-  Local coordinate zero point

1.2 Coordinate Systems and Coordinate Zero Point Symbols

- 
 Reference position:
 A specific position to establish coordinate systems and change tools
- 
 Basic machine coordinate zero point:
 A position specific to machine
- 
 Workpiece coordinate zero points (G54 to G59)
 A coordinate zero point used for workpiece machining

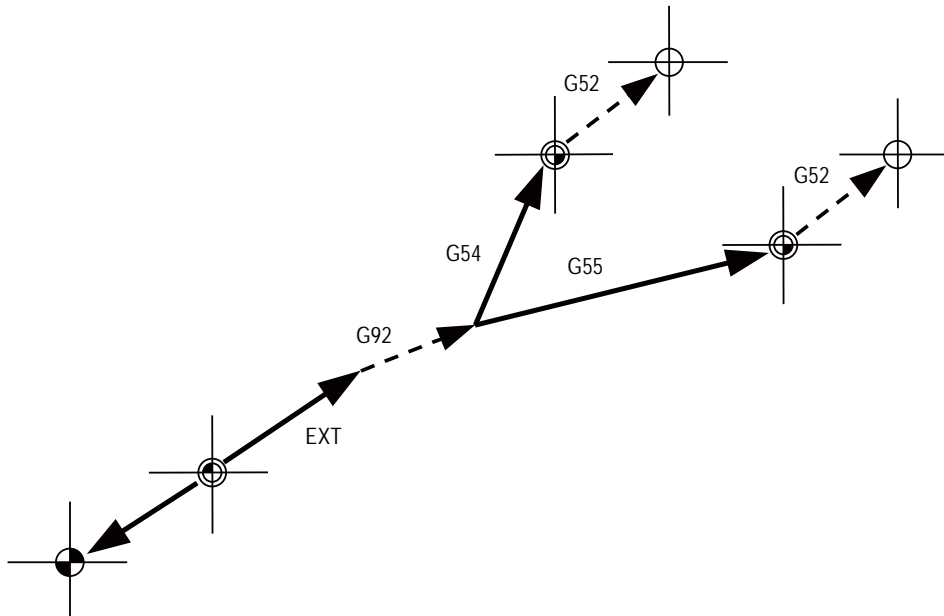
The basic machine coordinate system is the coordinate system that expresses the position (tool change position, stroke end position, etc.) that is specific to the machine.







Workpiece coordinate systems are used for workpiece machining.

Upon completion of the dog-type reference position return, the parameters are referred and the basic machine coordinate system and workpiece coordinate systems (G54 to G59) are automatically set.

The offset of the basic machine coordinate zero point and reference position is set by a parameter. (Normally, set by machine manufacturers)

Workpiece coordinate systems can be set with coordinate systems setting functions, workpiece coordinate offset measurement (additional specification), and etc.



- | | | |
|---|-----|---|
|  Reference position | G52 | Local coordinate system offset (*1) |
|  Basic machine coordinate zero point | G54 | Workpiece coordinate (G54) system offset (*1) |
|  Workpiece coordinate zero points | G55 | Workpiece coordinate (G55) system offset |
|  Local coordinate zero point | G92 | G92 Coordinate system shift |
|  Offset set by a parameter | EXT | External workpiece coordinate offset |
|  Offset set by a program
("0" is set when turning the power ON) | | |

(*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.

	Input unit parameters		Linear axis				Rotary axis (°)
			Millimeter		Inch		
			Diameter command	Radius command	Diameter command	Radius command	
Program command unit	#1015 cunit	= 10	0.001	0.001	0.0001	0.0001	0.001
		= 1	0.0001	0.0001	0.00001	0.00001	0.0001
Min. movement unit	#1003 iunit	= B	0.0005	0.001	0.00005	0.0001	0.001
		= C	0.00005	0.0001	0.000005	0.00001	0.0001
Input setting unit	#1003 iunit	= B	0.001	0.001	0.0001	0.0001	0.001
		= C	0.0001	0.0001	0.00001	0.00001	0.0001



Precautions

- (1) Inch/metric changeover can be handled by either a parameter screen (#1041 I_inch: valid only when the power is turned ON) or G commands (G20 or G21).
However, the changeover by a G command applies only to the program command units, and not to the input setting units. Consequently, the tool offset amounts and other compensation amounts as well as the variable data should be preset in order to correspond to input setting units.
- (2) The millimeter and inch systems cannot be used together.
- (3) When performing a circular interpolation between the axes whose 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.)

Program Formats

3.1 Program Format

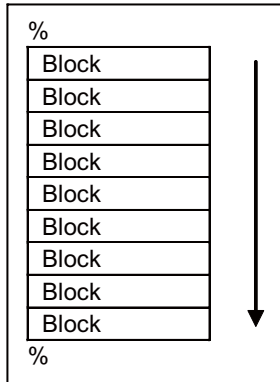
A collection of commands assigned to an NC to move a machine is called "program".

A program is a collection of units called "block" which specifies a sequence of machine tool operations.

Blocks are written in the order of the actual movement of a tool.

A block is a collection of "words" which constitutes a command to an operation.

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

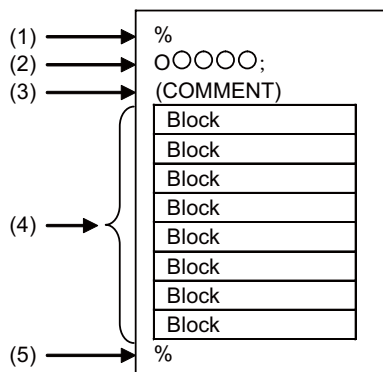




Detailed description

Program

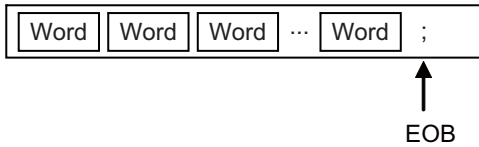
A program format looks as follows.



- (1) Program start
Input an End Of Record (EOR, %) at the head of a program.
It is automatically added when writing a program on an NC. When using an external device, do not forget to input it at the head of a program.
- (2) Program No.
Program Nos. are used to classify programs by main program unit or subprogram unit. They are designated by the address "O" followed by numbers of up to 8 digits. Program Nos. must be written at the head of programs. A setting is available to prohibit O8000s and O9000s from editing (edit lock). Refer to the instruction manual for the edit lock.
- (3) Comment
Data between control out "(" and control in ")" is ignored.
Information including program names and comments can be written in.
- (4) Program section
A program is a collection of several blocks.
- (5) Program end
Input an end of record (EOR, %) at the end of a program.
It is automatically added when writing a program on an NC.

Block and word

[Block]

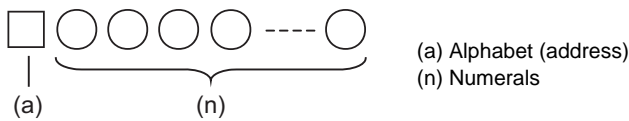


A block is a least command increment, consisting of words.

It contains the information which is required for a tool machine to execute a specific operation. One block unit constitutes a complete command.

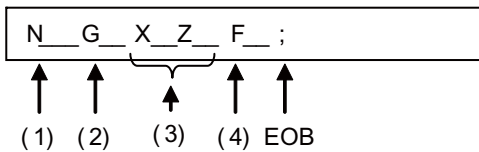
The end of each block is marked with an End of Block (EOB, expressed as ";" for the sake of convenience).

[Word]



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

The major contents of a word are described below.



(1) Sequence No.

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

(2) Preparatory functions (G code, G function)

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

There are 2 types of G code systems, 2 and 3. Refer to the description of G code system for available G codes.

(3) Coordinate words

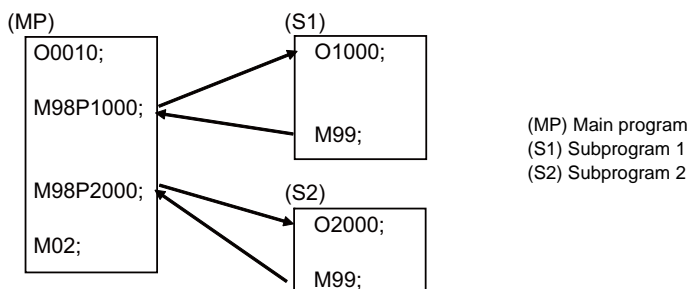
"Coordinate words" specify the coordinate position and movement amounts of tool machine axes. They consist of an address which indicates each axis of a tool machine followed by numerical information (+ or - signs and numerals).

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.



Precautions

- (1) Since the semicolon in the parentheses will not result in an EOB, it is 1 block
 - (Example 1) 2 blocks
 - G0 X-1000;
 - G1 X-2000 F500;
 - (Example 2) 1 blocks
 - (G0 X-1000;)
 - G1 X-2000 F500;
- (2) When there is no number following the alphabetic character in the actual program, the numeric value following the alphabetic character is handled as a 0.
 - (Example) G28XYZ; -> G28X0Y0Z0;

3.2 Program/sequence/block numbers; O, N



Function and purpose

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

- (1) Program numbers are classified by workpiece correspondence or by subprogram units, and they are designated by the address "O" followed by a number with up to 8 digits.
- (2) Sequence numbers are attached where appropriate to command blocks which configure machining programs, and they are designated by the address "N" followed by a number with up to 5 digits.
- (3) Block numbers are automatically provided internally. They are preset to zero every time a program number or sequence number is read, and they are counted up one at a time unless program numbers or sequence numbers are commanded in blocks which are subsequently read.

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

Machining program	Monitor display		
	Program No.	Sequence No.	Block No.
O12345678 (DEMO,PROG) ;	12345678	0	0
N100 G00 G90 X120. Z100. ;	12345678	100	0
G94 S1000;	12345678	100	1
N102 G71 P210 Q220 I0.2 K0.2 D0.5 F600 ;	12345678	102	0
N200 G94 S1200 F300 ;	12345678	200	0
N210 G01 X0 Z95. ;	12345678	210	0
G01 X20. ;	12345678	210	1
G03 X50. Z80. K-15. ;	12345678	210	2
G01 Z55. ;	12345678	210	3
G02 X80. Z40. I15. ;	12345678	210	4
G01 X100. ;	12345678	210	5
G01 Z30. ;	12345678	210	6
G02 Z10. K-15. ;	12345678	210	7
N220 G01 Z0 ;	12345678	220	0
N230 G00 X120. Z150. ;	12345678	230	0
N240 M02 ;	12345678	240	0
%		240	0

3.3 Optional Block Skip

3.3.1 Optional Block Skip; /



Function and purpose

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



Detailed description

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

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

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



Precautions

- (1) Put the "/" code for optional block skip at the beginning of a block. If it is placed inside the block, it is assumed as a user macro, a division instruction.
(Example)
N20 G1 X25. /Z25. ;NG (User macro, a division instruction; a program error results.)
/N20 G1 X25. Z25. ;OK
- (2) Parity checks (H and V) are conducted regardless of the optional block skip switch position.
- (3) The optional block skip is processed immediately before the pre-read buffer.
Consequently, it is not possible to skip up to the block which has been read into the pre-read buffer.
- (4) This function is valid even during a sequence No. search.
- (5) All blocks with the "/" code are also input and output during tape storing and tape output, regardless of the position of the optional block skip switch.

3.3.2 Optional Block Skip Addition ; /n



Function and purpose

Whether the block with "/n (n:1 to 9)" (slash) is executed during automatic operation and searching is selected.
By using the machining program with "/n" code, different parts can be machined by the same program.



Detailed description

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



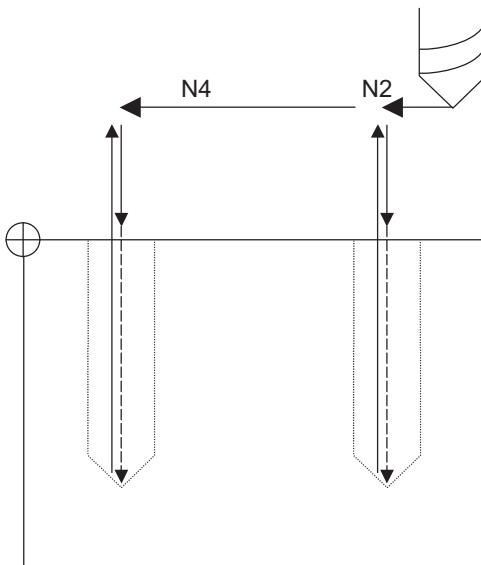
Program example

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

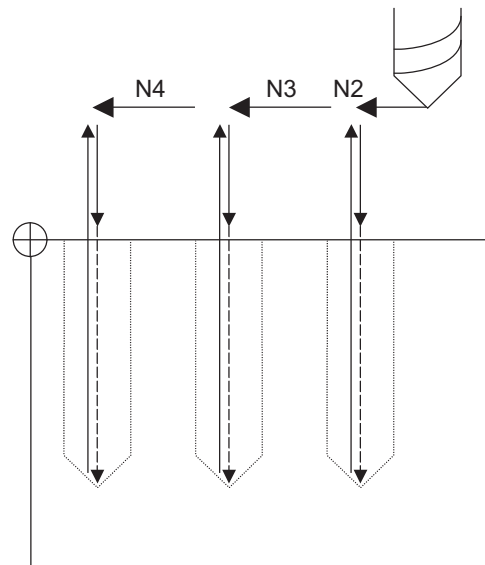
```

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

Part 1
Optional block skip 5 signal ON



Part 2
Optional block skip 5 signal OFF



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

N01 G90 Z3. M03 S1000 ;	(a) Optional block skip 1 signal ON (Optional block skip 2.3 signal OFF)
/1/2 N02 G00 X50. ;	N01 -> N08 -> N09 -> N10 -> N11 -> N12
/1/2 N03 G01 Z-20. F100 ;	
/1/2 N04 G00 Z3. ;	
/1 /3 N05 G00 X30. ;	(b) Optional block skip 2 signal ON (Optional block skip 1.3 signal OFF)
/1 /3 N06 G01 Z-20. F100 ;	N01 -> N05 -> N06 -> N07 -> N11 -> N12
/1 /3 N07 G00 Z3. ;	
/2/3 N08 G00 X10. ;	(c) Optional block skip 3 signal ON (Optional block skip 1.2 signal OFF)
/2/3 N09 G01 Z-20. F100 ;	N01 -> N02 -> N03 -> N04 -> N11 -> N12
/2/3 N10 G00 Z3. ;	
N11 G28 X0 M05 ;	
N12 M02 ;	

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

N01 G91 G28 X0.Y0.Z0.;	N03 block will operate as follows.
N02 G01 F1000;	(a) Optional block skip 1 signal ON
N03 X1. /1 Y1. /2 Z1.;	Optional block skip 2 signal OFF
N04 M30;	"Y1. Z1." is ignored.
	(b) Optional block skip 1 signal OFF
	Optional block skip 2 signal ON
	"Z1." is ignored.

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 codes include the two G code lists 2 and 3.

cmdtyp	G code lists
3	2
4	3

Here, G functions are explained using the G code list 3.

3.4.3 Table of G Code Lists

G code lists		Group	Function	Section
2	3			
Δ G00	Δ G00	01	Positioning	6.1
Δ G01	Δ G01	01	Linear interpolation	6.2
G02	G02	01	Circular interpolation CW, R specification circular interpolation CW, Helical interpolation CW	6.3 6.4 6.7
G03	G03	01	Circular interpolation CCW, R specification circular interpolation CCW, Helical interpolation CCW	6.3 6.4 6.7
G04	G04	00	Dwell	8.1
G09	G09	00	Exact stop check	7.10
G10	G10	00	Programmable parameter input, Compensation data input by program, Tool life management data registration, Programmable current limitation	12.5 12.6 13.11 16.5
G11	G11	00	Programmable parameter input cancel, Compensation data input by program cancel, Tool life management data registration cancel	12.5 12.6 13.11
*G14	*G14	18	● Balance cut OFF	13.16
G15	G15	18	● Balance cut ON	13.16
Δ G17	Δ G17	02	Plane selection X-Y	6.5
Δ G18	Δ G18	02	Plane selection Z-X	6.5
Δ G19	Δ G19	02	Plane selection Y-Z	6.5
Δ G20	Δ G20	06	Inch command	5.3
Δ G21	Δ G21	06	Metric command	5.3
G22	G22	04	Barrier check ON	15.1
*G23	*G23	04	Barrier check OFF	15.1
G27	G27	00	Reference position check	14.9
G28	G28	00	Automatic reference position return	14.7
G29	G29	00	Start position return	14.7
G30	G30	00	2nd, 3rd and 4th reference position return	14.8
G30.1	G30.1	00	Tool change position return 1	13.13
G30.2	G30.2	00	Tool change position return 2	13.13
G30.3	G30.3	00	Tool change position return 3	13.13
G30.4	G30.4	00	Tool change position return 4	13.13
G30.5	G30.5	00	Tool change position return 5	13.13
G31	G31	00	Skip, Multi-step skip function 2	16.2 16.4
G31.1	G31.1	00	Multi-step skip function 1-1	16.3
G31.2	G31.2	00	Multi-step skip function 1-2	16.3
G31.3	G31.3	00	Multi-step skip function 1-3	16.3
G32	G33	01	Thread cutting	6.6.1 6.6.2 6.6.3
G34	G34	01	Variable lead thread cutting	6.6.4
G37	G37	00	Automatic tool length measurement	16.1
*G40	*G40	07	Tool nose R compensation cancel	12.4
G41	G41	07	Tool nose R compensation left	12.4
G42	G42	07	Tool nose R compensation right	12.4
G46	G46	07	Tool nose R compensation (automatic direction identification) ON	12.4

G code lists		Group	Function	Section
2	3			
G43.1	G43.1	20	Selected spindle (nth spindle) control mode ON	10.6.2
G44.1	G44.1	20	2nd spindle control mode ON	10.6.2
G50	G92	00	Spindle clamp speed setting, Coordinate system setting	10.3 14.6
G52	G52	00	Local coordinate system setting	14.11
G53	G53	00	Basic machine coordinate system selection	14.5
*G54	*G54	12	Workpiece coordinate system selection 1	14.10
G55	G55	12	Workpiece coordinate system selection 2	14.10
G56	G56	12	Workpiece coordinate system selection 3	14.10
G57	G57	12	Workpiece coordinate system selection 4	14.10
G58	G58	12	Workpiece coordinate system selection 5	14.10
G59	G59	12	Workpiece coordinate system selection 6	14.10
G61	G61	13	Exact stop check mode	7.11
G61.1	G61.1	13	High-accuracy control	13.14
G62	G62	13	Automatic corner override	7.13
G63	G63	13	Tapping mode	7.14
*G64	*G64	13	Cutting mode	7.15
G65	G65	00	User macro call	13.6.1
G66	G66	14	User macro modal call A	13.6.1
G66.1	G66.1	14	User macro modal call B	13.6.1
*G67	*G67	14	User macro modal call cancel	13.6.1
G68	G68	15	Mirror image for facing tool posts ON	13.7
G69	G69	15	Mirror image for facing tool posts OFF	13.7
G68.1	G68.1	16	Coordinate rotation ON	13.15
G69.1	G69.1	16	Coordinate rotation cancel	13.15
G70	G70	09	Finishing cycle	13.2.4
G71	G71	09	Longitudinal rough cutting cycle	13.2.1
G72	G72	09	Face rough cutting cycle	13.2.2
G73	G73	09	Formed material rough cutting cycle	13.2.3
G74	G74	09	Face cut-off cycle	13.2.5
G75	G75	09	Longitudinal cut-off cycle	13.2.6
G76	G76	09	Compound thread cutting cycle, 2-part system simultaneous thread cutting cycle parameter setting command	13.2.7 13.18.1
G76.1	G76.1	09	● 2-part Simultaneous synchronous thread-cutting cycle (1)	13.18.2
G76.2	G76.2	09	● 2-part Simultaneous synchronous thread-cutting cycle (2)	13.18.3
*G80	*G80	09	Fixed cycle for drilling cancel	13.3.5
G81.1	G81.1	00	Chopping	13.19
G90	G77	09	Longitudinal cutting fixed cycle	13.1.1
G92	G78	09	Thread cutting fixed cycle	13.1.2
G94	G79	09	Face cutting fixed cycle	13.1.3
G79	G83.2	09	Deep hole drilling cycle 2	13.3.4
G83	G83	09	Deep hole drilling cycle 1 (Z axis)	13.3.1
G84	G84	09	Tapping cycle (Z axis)	13.3.2
G85	G85	09	Boring cycle (Z axis)	13.3.3
G87	G87	09	Deep hole drilling cycle 1 (X axis)	13.3.1
G88	G88	09	Tapping cycle (X axis)	13.3.2

G code lists		Group	Function	Section
2	3			
G89	G89	09	Boring cycle (X axis)	13.3.3
Δ G96	Δ G96	17	Constant surface speed control ON	10.2
Δ G97	Δ G97	17	Constant surface speed control OFF	10.2
Δ G98	Δ G94	05	Feed per minute (asynchronous feed)	7.4
Δ G99	Δ G95	05	Feed per revolution (synchronous feed)	7.4
-	Δ G90	03	Absolute value command	5.1
-	Δ G91	03	Incremental value command	5.1
-	*G98	10	Fixed cycle initial level return	13.3.7
-	G99	10	Fixed cycle R point level return	13.3.7
G113	G113	00	Spindle synchronization cancel	10.5.1
G114.1	G114.1	00	Spindle synchronization	10.5.1
G115	G115	00	● Start point designation timing synchronization Type 1	13.17.2
G116	G116	00	● Start point designation timing synchronization Type 2	13.17.3



Precautions

- (1) A program error (P34) will occur if a G code unlisted on the Table of G code lists is commanded.
- (2) An alarm will occur if a G code without additional specifications is commanded.
- (3) A (*) symbol indicates the G code to be selected in each group when the power is turned ON or when a reset is executed to initialize the modal.
- (4) A (Δ) symbol indicates the G code for which parameters selection is possible as an initialization status when the power is turned ON or when a reset is executed to initialize the modal. Note that inch/metric changeover can only be selected when the power is turned ON.
- (5) A (●) symbol indicates a function dedicated for multi-part system.
- (6) If two or more G codes from the same group are commanded in a block, the last G code will be valid.
- (7) 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.
- (8) Whether the modal is initialized or not depends on each reset input.
 - "Reset 1"
 - The modal is initialized when the reset initialization parameter (#1151 rstinit) is ON.
 - "Reset 2 "and "Reset and Rewind"
 - The modal is initialized when the signal is input.
 - Reset at emergency stop release
 - Conforms to "Reset 1".
 - When modal is automatically reset at the start of individual functions such as reference point return.
 - Conforms to "Reset & rewind".



CAUTION

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

3.5 Precautions Before Starting Machining

 **CAUTION**

1. When creating the machining program, select the appropriate machining conditions, and make sure that the performance, capacity and limits of the machine and NC are not exceeded. The examples do not consider the machining conditions.
2. Before starting actual machining, always carry out graphic check, dry run operation and single block operation to check the machining program, tool offset amount, workpiece offset amount and etc.



4

Pre-read Buffers

4.1 Pre-read Buffers



Function and purpose

During automatic processing, the contents of one block ahead are normally pre-read so that program analysis processing is conducted smoothly. However, during tool nose 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 Absolute/Incremental Value Commands ; G90,G91

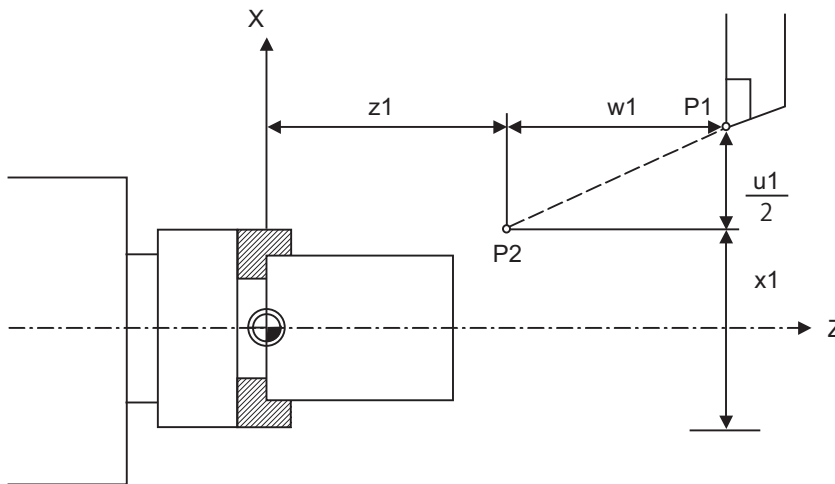


Function and purpose

There are two methods of issuing tool movement amount commands: the absolute value method and the incremental value method.

To designate the coordinates of a point to which to move, the absolute value method issues a command using the distance from the coordinate zero point; on the other hand, the incremental value method issues a command using the distance from the present point.

Use an axis address or a G command to choose between the absolute or incremental value command. Whether an axis address or a G command is valid can be selected according to the parameter setting. The following figure shows what happens when the tool is moved from point P1 to point P2.



- (1) Movement command by an axis address (when "#1076 AbsInc" is "1")
 Absolute value command: G00 Xx1 Zz1 ;
 Incremental value command: G00 Uu1 Ww1 ;
- (2) Movement command by a G command (when "#1076 AbsInc" is "0")
 Absolute value command: G90 G00 Xx1 Zx1 ;
 Incremental value command: G91 G00 Xu1 Zw1 ;



Command format

G90; ... Absolute value command

G91; ... Incremental value command

When the parameter "#1076 AbsInc" is set to "0", a G command selects either the incremental or absolute value commands.

After commanding G90/ G91, coordinates will be commanded with incremental or absolute values.



Detailed description

Selection between incremental or absolute value commands by an axis address

When the parameter "#1076 AbsInc" is set to "1", an axis address selects either the incremental or absolute value commands.

- (1) Set correspondence between addresses and axes with the following parameters.

#1013 axname

#1014 incax

Following table shows the example of when "X,Z,C,Y" are set to "#1013 axname" and "U,W,H,V" are set to "#1014 incax".

		Command method
Absolute value	X axis	Address X
	Z axis	Address Z
	C/Y axis	Address C/Y
Incremental value	X axis	Address U
	Z axis	Address W
	C/Y axis	Address H/V

(Note 1) The C/Y axis is an example of additional axes.

- (2) Absolute and incremental values can be used together in the same block.

(Example) X__ W__ ; ... An absolute value command for X axis and an incremental value command for Z axis



Precautions

- (1) When parameter "#1076 AbsInc" is 1, and H is used for the incremental command address, address H of blocks in M98 and G10 L50 modal will be handled as the parameter of each command, and the axis will not be moved.

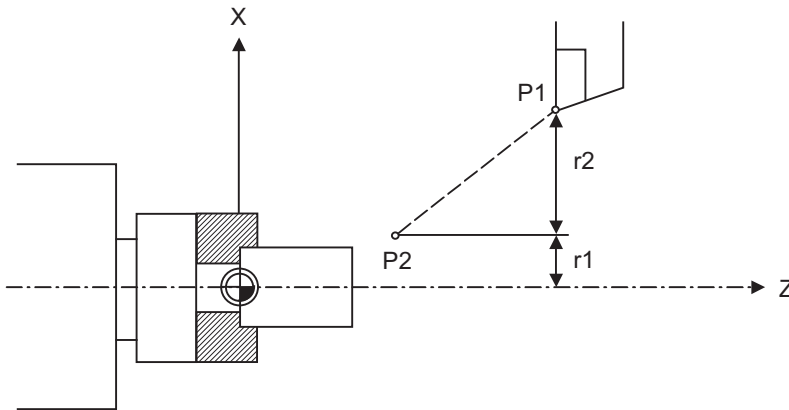
5.2 Radius/Diameter Designation



Function and purpose

On a lathe, a workpiece rotates, so its coordinate positions, dimensions, and commands can be designated by radius/ diameter values. Commands using diameter values are called diameter designation, and commands with radius values are called radius designation.

Either radius or diameter designation can be used depending on the setting of the parameter (#1019 dia). The figure below shows the command procedure when the tool is to be moved from point P1 to point P2.



X command		U command		Remarks
Radius	Diameter	Radius	Diameter	
X = r1	X = 2 * r1	U = r2	U = 2 * r2	Even when a diameter designation has been selected, the U command can exclusively be changed into a radius designation by the parameter "#1077 radius". (Note) "U" is an incremental command address.



Precautions and restrictions

- (1) In the above example, the tool moves from P1 to P2 in the minus direction of the X axis. So when this is using incremental value command, the minus sign is given to the numerical value being commanded.
- (2) In this manual, diameter commands are used in descriptions of both the X and U axes for the sake of convenience.

5.3 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.

(Example) Relationship between input command units and G20/G21 commands
(with decimal point input type I)

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

5.4 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 and speeds in machining programs.
- (2) Refer to the table "Addresses used, validity of decimal point commands" for details on the valid addresses for the decimal point commands.
- (3) In decimal point command, the valid range of command value is as shown below. (for input command increment cunit=10)

	Movement command (linear)		Movement command (rotation)		Feedrate		Dwell (X)	
	Integer	Decimal part	Integer	Decimal part	Integer	Decimal part	Integer	Decimal part
mm (millimeter)	0 to 99999.	.000 to .999	0 to 99999.	.000 to .999	0 to 60000.	.000 to .999	0. to 99999.	.000 to .999
					0 to 999.	.0000 to .9999		
inch (inch)	0 to 9999.	.0000 to .9999	99999. (359.)	.0 to .999	0 to 2362.	.0000 to .9999	0 to 99.	.000 to .999
					0 to 99.	.000000 to .999999		

(Note) The top row gives the feedrate as a per-minute rate and the bottom row as a per revolution rate.

- (4) The decimal point command is valid even for commands defining the variable data used in subprograms.
- (5) Decimal point commands for decimal point invalid addresses are processed as an integer only data which everything below the decimal point is ignored. Decimal point invalid addresses include the followings; D,H,L,M,N,O,P,S,T.
All variable commands, however, are treated as data with decimal points.
- (6) As for the minimum unit when a value is commanded without a decimal point though the decimal point designation is valid, select the minimum input command unit determined by specifications (1 μ m, 10 μ m, etc.) or "mm".
Select which to use by setting the parameter "#1078 Decpt2".

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 μ m 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 μ m 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	2nd miscellaneous function code	
	Valid	Angle data	
	Invalid	MRC program number	
	Invalid	Parameter input by program, axis No.	
	Valid	Deep hole drilling cycle (2) Safety distance	
		Compound type thread cutting cycle Thread cutting start shift angle	
	Valid	Spindle synchronous acceleration/deceleration time constant	
B	Valid	Coordinate position data	
	Invalid	2nd miscellaneous function code	
C	Valid	Coordinate position data	
	Invalid	2nd miscellaneous function code	
	Valid	Corner chamfering amount	,C
	Valid	Program tool offset input Nose R compensation amount (incremental)	
	Valid	Chamfering width (slitting cycle)	
D	Valid	Automatic tool length measurement, deceleration range d	
	Invalid	Parameter input by program, byte type data	
	Invalid	Synchronous spindle No. at spindle synchronization	
	Valid	Parameter input by program, numerical value parameter	
E	Valid	Inch threads Precision thread lead	
	Valid	Corner cutting feedrate	
F	Valid	Feedrate	
	Valid	Thread lead	
G	Valid	Preparatory function code	
H	Valid	Coordinate position data	
	Invalid	Sequence numbers in subprograms	
	Invalid	Subprogram return destination program No.	
	Invalid	Parameter input by program, bit type data	
	Invalid	Selection of linear - arc intersection (geometric)	
	Invalid	Basic spindle No. at spindle synchronization	
I	Valid	Circular center coordinates	
	Valid	Nose R compensation/ tool radius compensation vector components	
	Valid	Deep hole drilling (2) First cut amount	
	Valid	G0/G1 in-position width Hole drilling cycle G0 in-position width	,I
J	Valid	Circular center coordinates	
	Valid	Nose R compensation/ tool radius compensation vector components	
	Invalid	Deep hole drilling (2) Dwell time at return point	
	Valid	Hole drilling cycle G1 in-position width	,J
K	Valid	Circular center coordinates	
	Valid	Nose R compensation/tool radius compensation vector components	
	Invalid	Hole machining cycle Number of repetitions	
	Valid	Deep hole drilling cycle (2) Second and subsequent cut amounts	
	Valid	Thread lead increase/decrease amount (variable lead thread cutting)	

Address	Decimal point command	Application	Remarks
L	Invalid	Subprogram Number of repetitions	
	Invalid	Program tool compensation input type selection	L2 L10 L11
	Invalid	Parameter input by program, selection	L50
	Invalid	Parameter input by program, two-word type data	4 bytes
	Invalid	Part system No. designation in timing synchronization between part systems	
	Invalid	PLC skip signal command of multi-step skip function 2	
M	Invalid	Miscellaneous function codes	
N	Invalid	Sequence numbers	
	Invalid	Parameter input by program, data No.	
O	Invalid	Program numbers	
P	Invalid	Dwell time	
	Invalid	Subprogram call program numbers	
	Invalid	2nd, 3rd and 4th reference point number	
	Invalid	Constant surface speed control, axis number	
	Invalid	MRC finishing shape start sequence number	
	Valid	Cut-off cycle shift amount/cut amount	
	Invalid	Compound type thread cutting cycle Number of cutting passes, chamfering, tool nose angle	
	Valid	Compound type thread cutting cycle Thread height	
	Invalid	Program tool compensation input compensation No.	
	Invalid	Parameter input by program, parameter No.	
	Valid	Coordinate position data	
	Invalid	High-speed skip signal command of multi-step skip function 2	
	Valid	Arc center coordinates (absolute value) (geometric)	
	Invalid	Subprogram return destination sequence No.	
Q	Invalid	Minimum spindle clamp rotation speed	
	Invalid	MRC finishing shape end sequence number	
	Valid	Cut-off cycle Cut amount/shift amount	
	Valid	Compound type thread cutting cycle Minimum cut amount	
	Valid	Compound type thread cutting cycle First cut amount	
	Valid	Deep hole drilling cycle 1 Cut amount of each pass	
	Invalid	Program tool compensation input Hypothetical tool nose point number	
	Invalid	Deep hole drilling cycle (2) Dwell time at cut point	
	Valid	Arc center coordinates (absolute value) (geometric)	
	Valid	Thread cutting start shift angle	

Address	Decimal point command	Application	Remarks
R	Valid	R-designated arc radius	
	Valid	Corner rounding circular radius	,R
	Valid	Automatic tool length measurement, deceleration range r	
	Valid	MRC longitudinal/face escape amount	
	Invalid	MRC shaping division number	
	Valid	Cut-off cycle, return amount	
	Valid	Cut-off cycle, escape amount	
	Valid	Compound type thread cutting cycle, finishing allowance	
	Valid	Compound type thread cutting cycle/turning cycle, taper difference	
	Valid	Hole drilling cycle/deep hole drilling cycle (2), distance to reference point	
	Valid	Program tool compensation input/nose R compensation amount	
	Valid	Coordinate position data	
	Valid	Rough cutting cycle (longitudinal) (face) pull amount	
	Valid	Synchronous tap/ asynchronous tap changeover	,R
Valid	Synchronous spindle phase shift amount		
S	Invalid	Spindle function codes	
	Invalid	Maximum spindle clamp rotation speed	
	Invalid	Constant surface speed control, surface speed	
	Invalid	Parameter input by program, part system No.	2 bytes
T	Invalid	Tool function codes	
U	Valid	Coordinate position data	
	Valid	Program tool compensation input	
	Valid	Rough cutting cycle (longitudinal) cutting amount	
	Valid	Dwell time	
V	Valid	Coordinate position data	
	Valid	Program tool compensation input	
W	Valid	Coordinate position data	
	Valid	Program tool compensation input	
	Valid	Rough cutting cycle (face) cutting amount	
X	Valid	Coordinate position data	
	Valid	Dwell	
	Valid	Program tool compensation input	
Y	Valid	Coordinate position data	
	Valid	Program tool compensation input	
Z	Valid	Coordinate position data	
	Valid	Program tool compensation input	

(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 μ m	When 1 = 10 μ m	
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 μ m unit)	X123.450 mm	X12345.000 mm
#111 = 123 #112 = 5.55 X#111 Z#112	X123.000 mm Z5.550 mm	X123.000 mm Z5.550 mm	X123.000 mm Z5.550 mm
#113 = #111 + #112 (addition)	#113 = 128.550	#113 = 128.550	#113 = 128.550
#114 = #111 - #112 (subtraction)	#114 = 117.450	#114 = 117.450	#114 = 117.450
#115 = #111 * #112 (multiplication)	#115 = 682.650	#115 = 682.650	#115 = 682.650
#116 = #111 / #112 #117 = #112 / #111 (division)	#116 = 22.162 #117 = 0.045	#116 = 22.162 #117 = 0.045	#116 = 22.162 #117 = 0.045



Precautions

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

6

Interpolation Functions

6.1 Positioning (Rapid Traverse) ; G00



Function and purpose

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



Command format

G00 X_/U_ Z_/W_ ,I_ ; ... Positioning (Rapid Traverse)

X_/U_	X axis end point coordinate (X is the absolute value of workpiece coordinate system, U is the incremental value from present position)
Z_/W_	Z axis end point coordinate (Z is the absolute value of workpiece coordinate system, W is the incremental value from present position)
,I	In-position width

The command addresses are valid for all additional axes.



Detailed description

- (1) Positioning will be performed at the rapid traverse rate set in the parameter "#2001 rapid".
- (2) G00 command belongs to the 01 group and is modal. When G00 command is successively issued, the following blocks can be specified only by the coordinate words.
- (3) In the G00 mode, acceleration and deceleration are always carried out at the start point and end point of the block. Before advancing to the next block, a commanded deceleration or an in-position check is conducted at the end point to confirm that the movement is completed for all the moving axes in each part system..
- (4) G functions (G83 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.

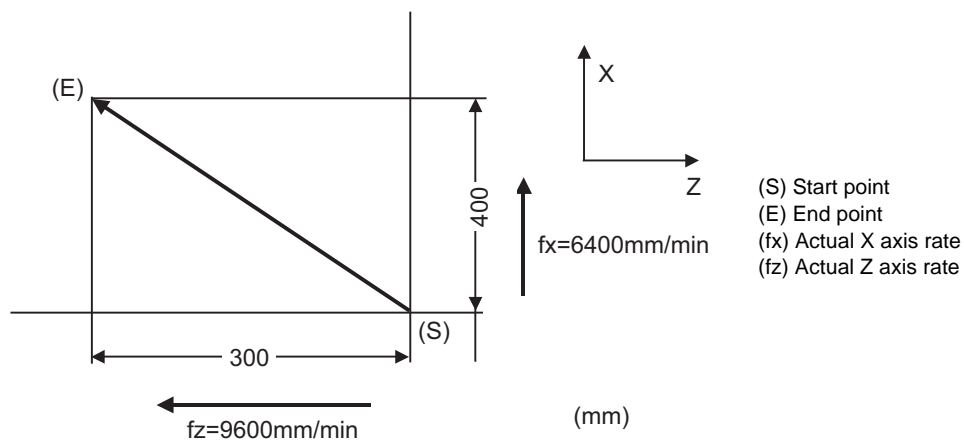
- (1) Linear path: When the parameter "#1086 G0Intp" is set to "0"

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 Z-axis rapid traverse rates are both 9600mm/min;

G00 Z-300000 X400000 ; (With an input setting unit of 0.001mm)

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



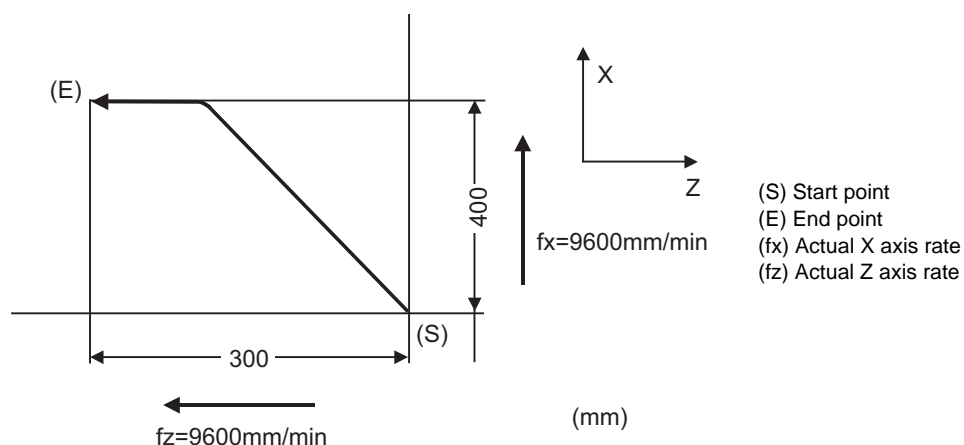
- (2) Non-linear path: When the parameter "#1086 G0Intp" is set to "1"

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 Z-axis rapid traverse rates are both 9600mm/min;

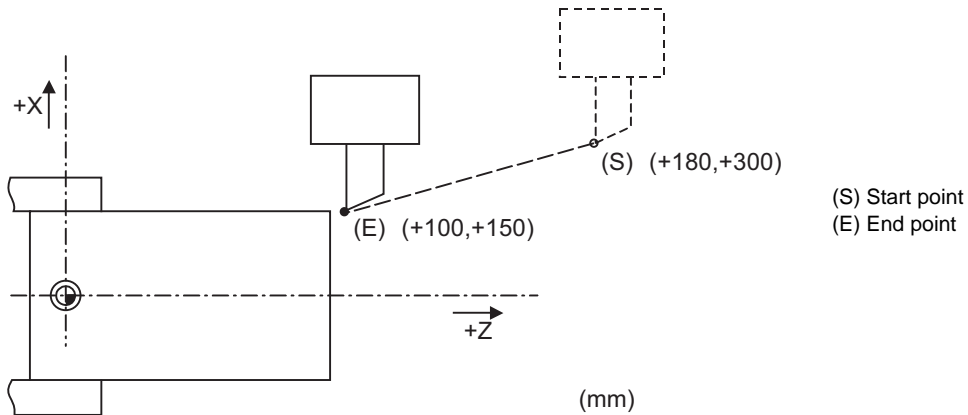
G00 Z-300000 X400000 ; (With an input setting unit of 0.001mm)

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





Program example



G00 X100. Z150. ;	Absolute value command
G00 U-80. W-150. ;	Incremental value command



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")	

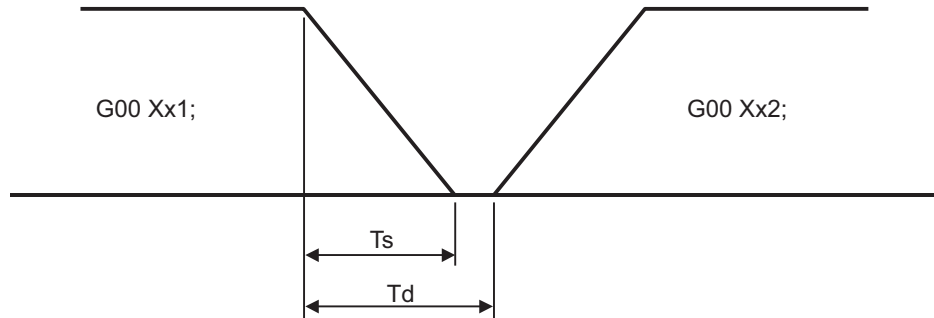
* Following descriptions are for the case of rapid traverse. For G01, interpret the parameters into suitable ones.

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.

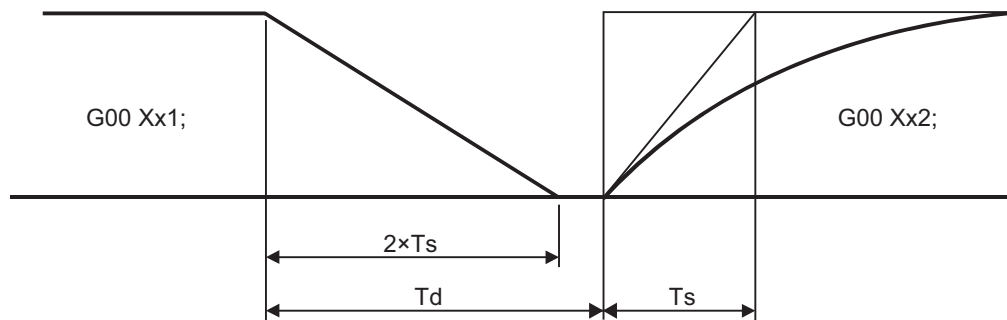
The deceleration check time (Td) is as follows, depending on the acceleration/deceleration type set in the parameter "#2003 smgst".

(1) Linear acceleration/linear deceleration



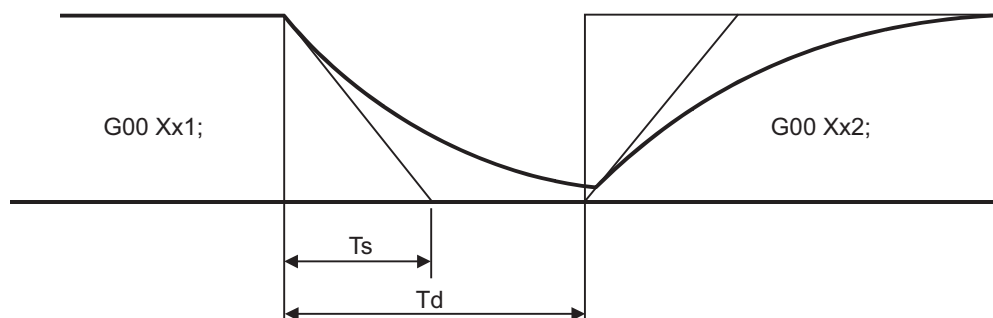
(Ts) Acceleration/deceleration time constant (Td) Deceleration check time: $T_d = T_s + (0 \text{ to } 7\text{ms})$

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

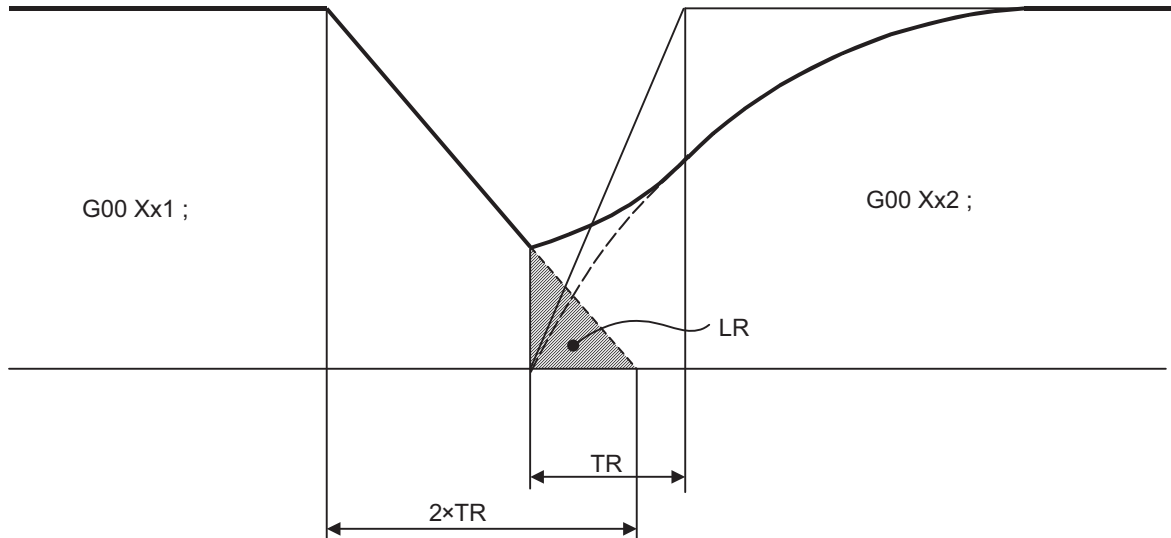
The time required for the deceleration check is the longest among the deceleration check times of each axis determined by the acceleration/deceleration mode and time constants of the axes commanded simultaneously.

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.



(TR) Rapid traverse acceleration/deceleration time constant

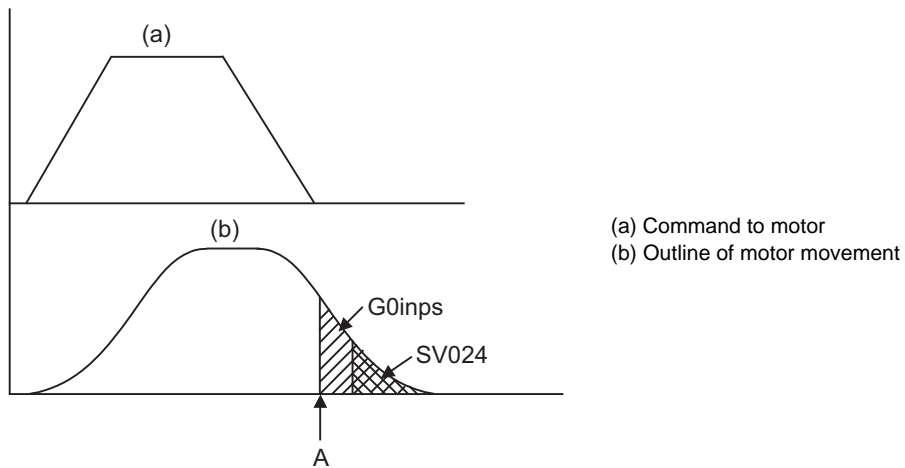
(LR) In-position width

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

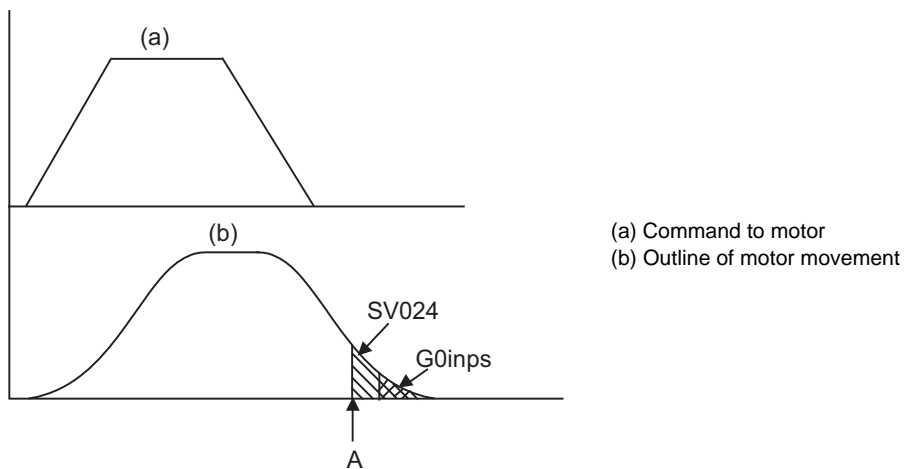
The purpose of the rapid traverse deceleration check is to minimize the positioning time. The bigger the setting value for the in-position width, the shorter the time is, but the remaining distance of the previous block at the start of the next block also becomes larger, and this could become an obstacle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

The check for the remaining distance is done at set intervals. Accordingly, it may not be possible to get the effect of time reduction for positioning as in-position width setting value.

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



- (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.

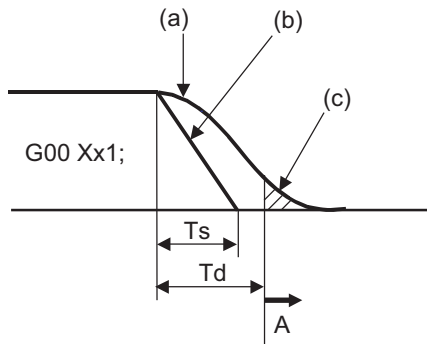
When there are several movement axes, the system confirms that the position error amount of each movement axis in each part system is less than the in-position width issued in this command before executing the next block.

The differences of In-position check

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

(1) In-position check with parameter

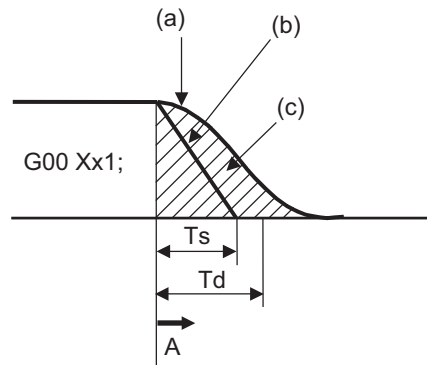
After completing deceleration of the command system (A), the servo system's position error amount and the parameter setting value (in-position width) are compared.



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

(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

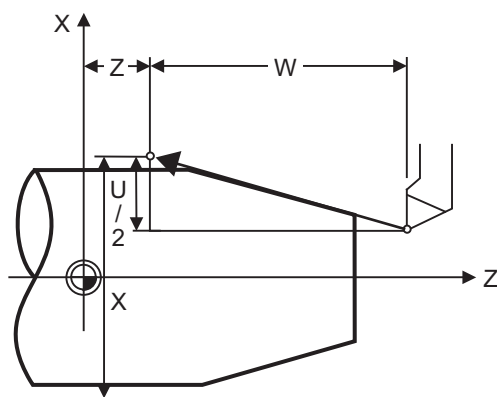
This command is accompanied by coordinate words and a feedrate command. It makes the tool move (interpolate) linearly from its current position to the end point specified by the coordinate words at the speed specified by address F. In this case, the feedrate specified by address F always acts as a linear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



Command format

G01 X__/U__ Z__/W__ α __ F__ ,I__ ; ... Linear interpolation

X,U,Z,W, α	Coordinate values (α is the additional axis.)
F	Feedrate (mm/min or ° /min)
,I	In-position width. This is valid only in the commanded block. A decimal command will result in a program error. A block that does not contain this address will follow the parameter "#1193 inpos" settings. 1 to 999999 (μ m)





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 (G70 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_ Z_ F_ ,I_ ;	
X,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.

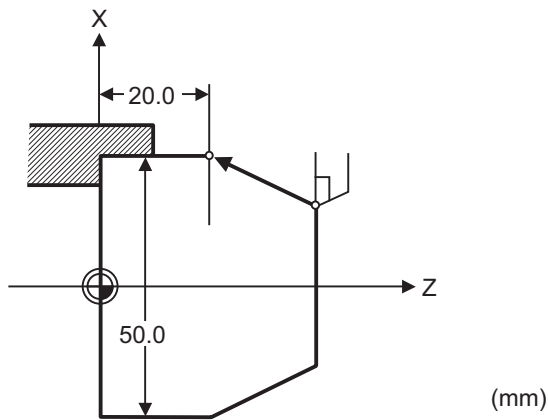
- When the error detection switch is ON.
- When G09 (exact stop check) is commanded in the same block.
- When G61 (exact stop check mode) is selected.

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



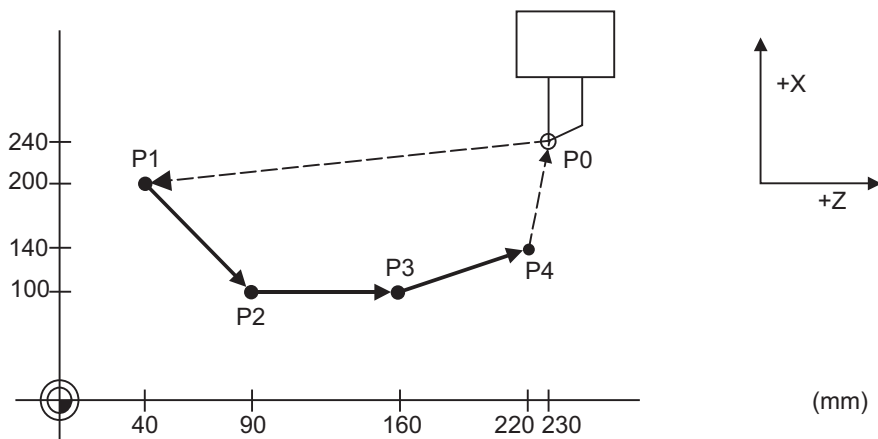
Program example

(Example 1)



G01 X50.0 Z20.0 F300 ;	
------------------------	--

(Example 2) Cutting in the sequence of P1 -> P2 -> P3 -> P4 at 300mm/min feedrate.
However, P0 -> P1 and P4 -> P0 are for tool positioning.



G00 X200. Z40. ;	P0 -> P1
G01 X100. Z90. F300 ;	P1 -> P2
Z160. ;	P2 -> P3
X140. Z220. ;	P3 -> P4
G00 X240. Z230. ;	P4 -> P0

6.3 Circular Interpolation ; G02,G03



Function and purpose

These commands serve to move the tool along a circular.



Command format

G02 X__/U__ Z__/W__ I__ K__ F__ ; ... Circular interpolation : Clockwise (CW)

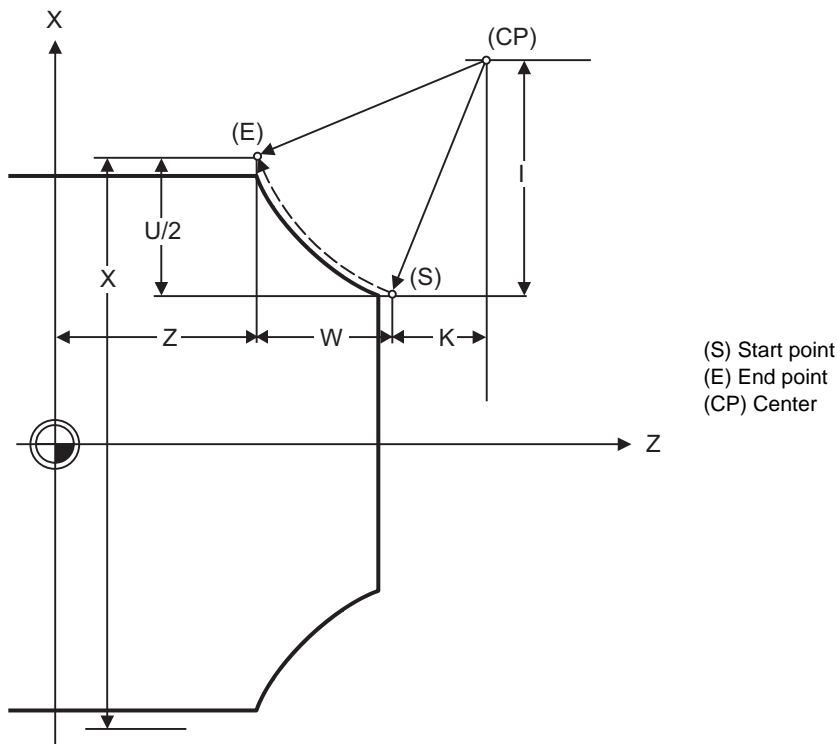
G03 X__/U__ Z__/W__ I__ K__ F__ ; ... Circular interpolation : Counterclockwise (CCW)

X/U	Circular end point coordinates, X axis (X is the absolute value of workpiece coordinate system, U is the incremental value from present position)
Z/W	Circular end point coordinates, Z axis (Z is the absolute value of workpiece coordinate system, W is the incremental value from present position)
I	Circular center, X axis (I is the radius command incremental value of X coordinate at the center as seen from the start point.)
K	Circular center, Z axis (K is the incremental value of Z coordinate at the center as seen from the start point.)
F	Feedrate

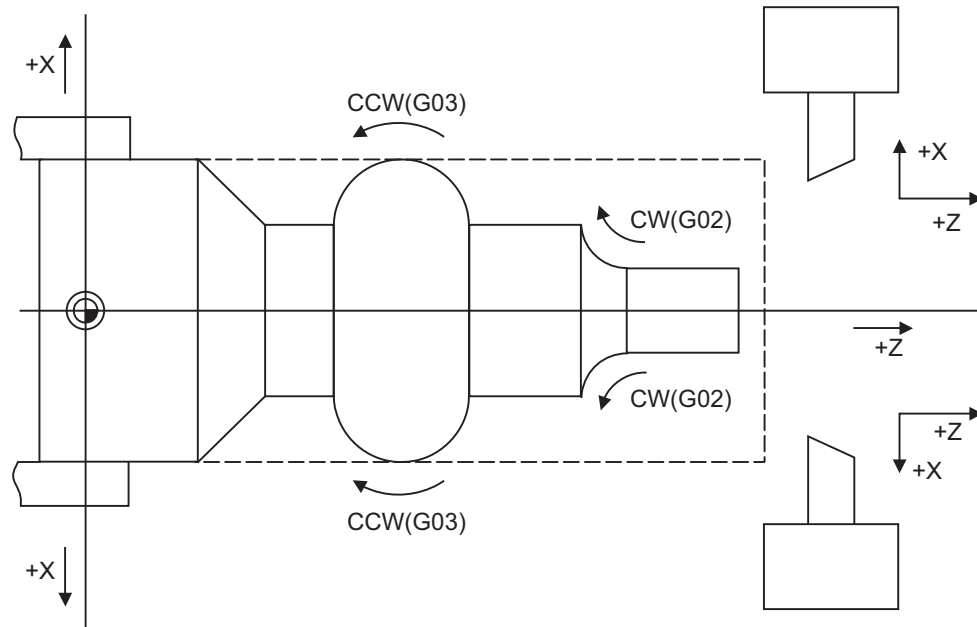


Detailed description

- (1) The arc center coordinate value is commanded with an input setting 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.



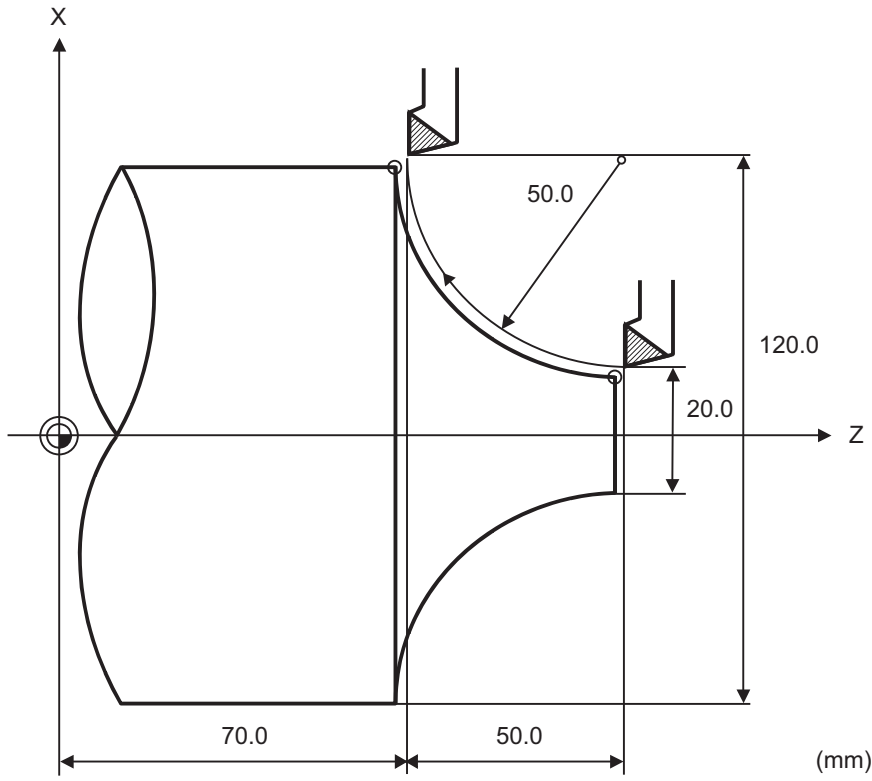
- (2) G02 (G03) is a modal command of the 01 group. When G02 (G03) command is issued continuously, the next block and after can be commanded with only coordinate words.
The circular rotation direction is distinguished by G02 and G03.
G02 Clockwise (CW)
G03 Counterclockwise (CCW)



- (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.
- Rotation direction : Clockwise (G02) or counterclockwise (G03)
 - Circular end point coordinates : Given by addresses X, Z, U, W
 - Circular center coordinates : Given by addresses I, K (incremental value commands)
 - Feedrate : Given by address F
- (5) A program error results when I, K or R is not commanded.
Consideration must be given to the sign for I and K since I is the distance in the X-axis direction to the arc center from the start point and K in the Z-axis direction.
- (6) No T commands can be issued in the G2/G3 modal status.
A program error (P151) will occur if a T command is issued in the G2/G3 modal status.



Program example

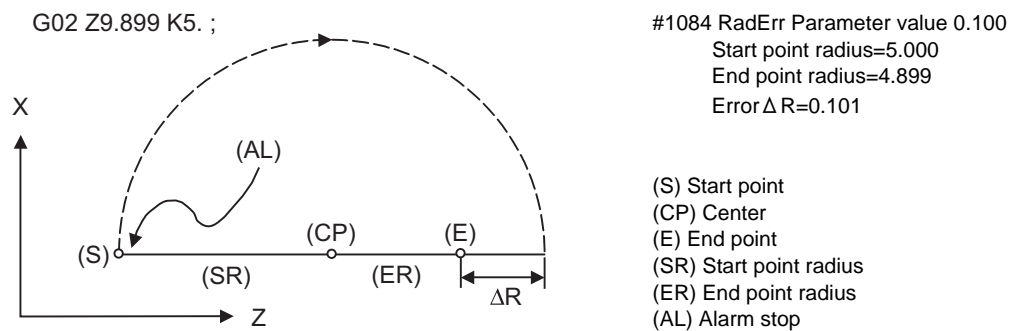


G2 X120.0 Z70.0 I50.0 F200 ;	Absolute value command
G2 U100.0 W-50.0 I50.0 F200 ;	Incremental value command

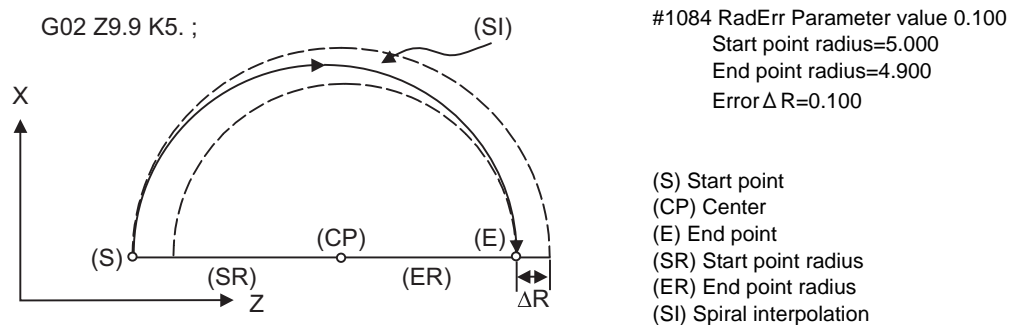


Precautions

- (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 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) results at the circular start point when error ΔR is greater than parameter "#1084 RadErr".



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



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, K) designation, these commands can also be issued by directly designating the circular radius R.



Command format

G02 X/U__ Z/W__ R__ F__ ; ... R specification circular interpolation Clockwise (CW)

G03 X/U__ Z/W__ R__ F__ ; ... R specification circular interpolation Counterclockwise (CCW)

X/U	X axis end point coordinate
Z/W	Z axis end point coordinate
R	Circular radius
F	Feedrate

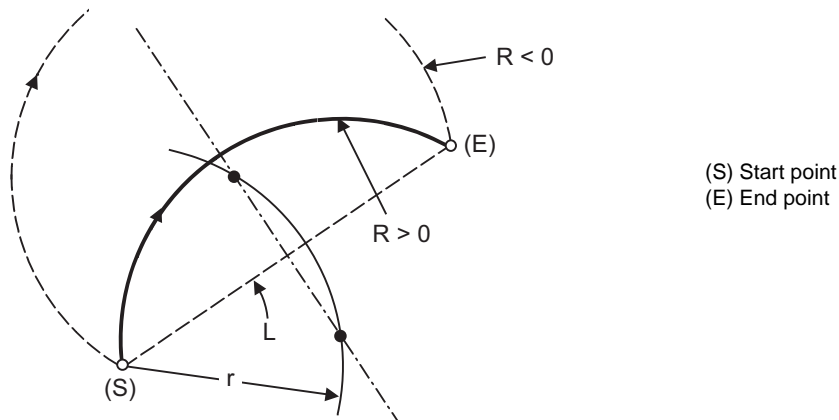
The arc radius is commanded with a program command unit. Caution is required for the arc command of an axis for which the program command unit (#1015 cunit) differs. Command with a decimal point to avoid confusion.



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:

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

Where L is the line from the start point to the end point. If an R specification and I, 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, K specification circular command should therefore be used in such a case.



Program example

(Example 1)

G03 Zz1 Xx1 Rr1 Ff1 ;	R specification circular on Z-X plane
-----------------------	---------------------------------------

(Example 2)

G02 Xx1 Zz1 Ii1 Kk1 Rr1 Ff1 ;	R specification circular on X-Z plane (When the R specification and I, K specification are contained in the same block, the R specification has priority in processing.)
-------------------------------	---

6.5 Plane Selection ; G17,G18,G19



Function and purpose

These commands are used to select the control plane and the plane on which the circular exists. 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 rotation axis is entered as a parallel axis, the commands can select the plane containing the rotation axis.

These commands are used to select:

- The plane for circular interpolation
- The plane for nose R compensation



Command format

G17; ... I-J plane selection

G18 ; ... K-I plane selection

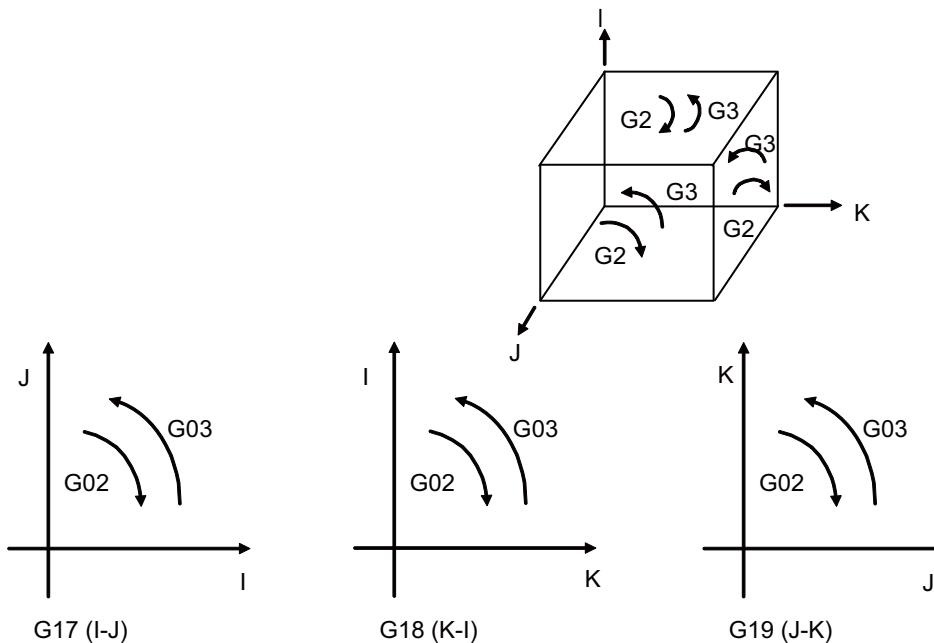
G19; ... J-K plane selection



Detailed description

I, J and K indicate each basic axis or parallel axis.

When the power is turned ON or when the system is reset, the plane set by the parameters "#1025 I_plane" is selected.



Parameter entry

	#1026 to 1028 base_I,J,K	#1029 to 1031 aux_I,J,K	Basic axes and parallel axes can be entered in the parameters. The same axis name can be entered in duplication, but when it is assigned in duplication, the plane is determined by plane selection system (4). The axis which is not registered as the control axis cannot be set.
I	X	Y	
J	Y		
K	Z		

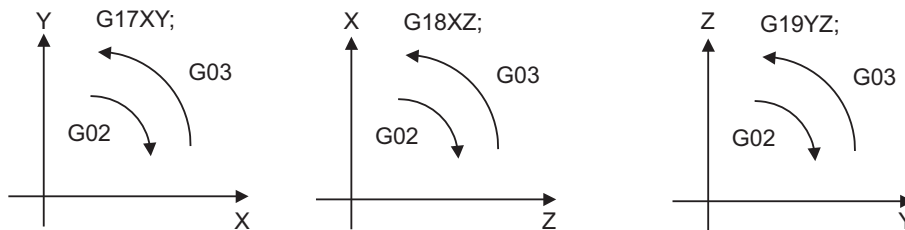
Table 1 Examples of plane selection parameter entry

Plane selection system

This section describes the plane selection shown in the "Table 1 Examples of plane selection parameter entry".

- (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.

(Example)



- (2) Plane selection is not performed with blocks in which the plane selection G code (G17, G18, G19) is not assigned.

G18 X_ Z_ ; Z-X plane

Y_ Z_ ; Z-X plane (no plane change)

- (3) When the axis addresses are omitted in the block containing the plane selection G codes (G17, G18, G19), it is assumed that the axis addresses of the 3 basic axes have been assigned.

G18 ; (Z-X plane = G18 XZ ;)

- (4) 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.

G18 XYZ ; The Z-X plane is selected. Therefore, the Y movement is unrelated to the selected plane.

- (Note 1) When the "2" in the parameter "#1025 I_plane" is kept ON, the G18 plane is selected when the power is turned ON or when the system is reset.

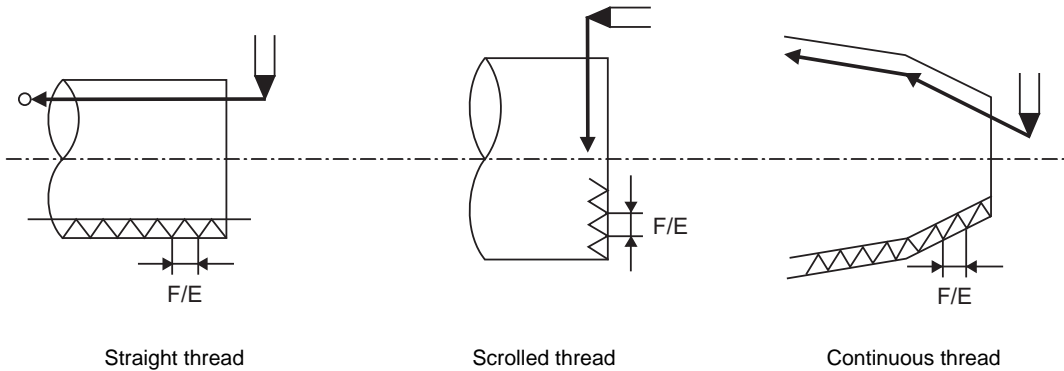
6.6 Thread Cutting

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, tapered thread-cutting, and continuous thread-cutting.



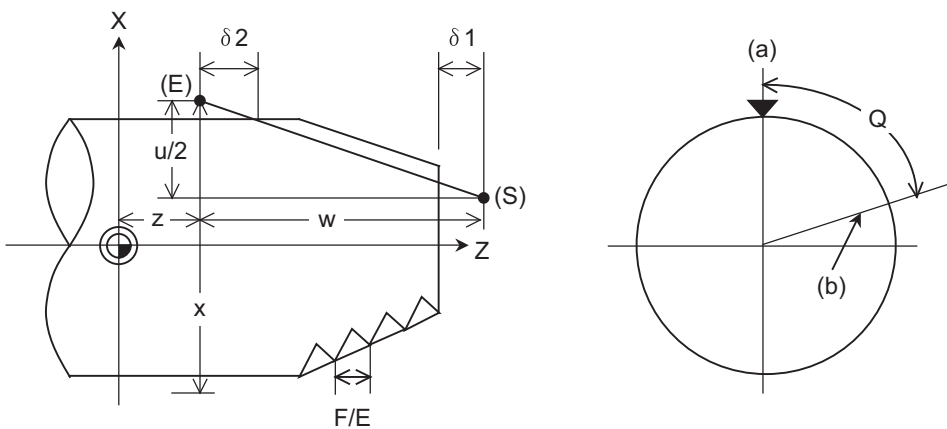
Command format

G33 Z/W__ X/U__ F__ Q__ ; ... Normal lead thread cutting

Z,W,X,U	Thread end point
F	Lead of long axis (axis which moves most) direction
Q	Thread cutting start shift angle, 0.001 to 360.000°

G33 Z/W__ X/U__ E__ Q__ ; ... Precision lead thread cutting

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

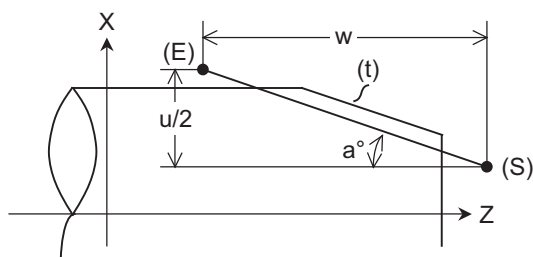


- $\delta 1 >$ Illegal lead at start of thread cutting
- $\delta 2 >$ Illegal lead at end of thread cutting
- (S) Start point (E) End point
- (a) One-rotation synchronization signal
- (b) Thread cutting start position



Detailed description

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



(t) Tapered thread section (E) End point (S) Start point

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 unit system	B (0.001mm)			C (0.0001mm)		
	Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)	F (mm/rev)	E (mm/rev)
Least command increments	1 (= 0.0001), (1.=1.0000)	1 (= 0.00001), (1.=1.00000)	1 (= 1.00), (1.=1.00)	1 (= 0.00001), (1.=1.00000)	1(=0.000001), (1.=1.000000)	1 (= 1.000), (1.=1.000)
Command range	0.0001 to 999.9999	0.00001 to 999.99999	0.03 to 999.99	0.00001 to 99.99999	0.000001 to 99.999999	0.255 to 9999.999

Thread cutting inch input

Input unit system	B (0.0001inch)			C (0.00001inch)		
	Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)	F (inch/rev)	E (inch/rev)
Least command increments	1(=0.000001), (1.=1.000000)	1(=0.0000001), (1.=1.0000000)	1 (= 1), (1.=1.0000)	1(=0.0000001), (1.=1.0000000)	1(=0.00000001), (1.=1.00000000)	1(=1), (1.=1.00000)
Command range	0.000001 to 99.999999	0.000010 to 9.9999999	0.0101 to 9999.9999	0.0000001 to 9.9999999	0.00000001 to 0.99999999	0.10001 to 999.99999

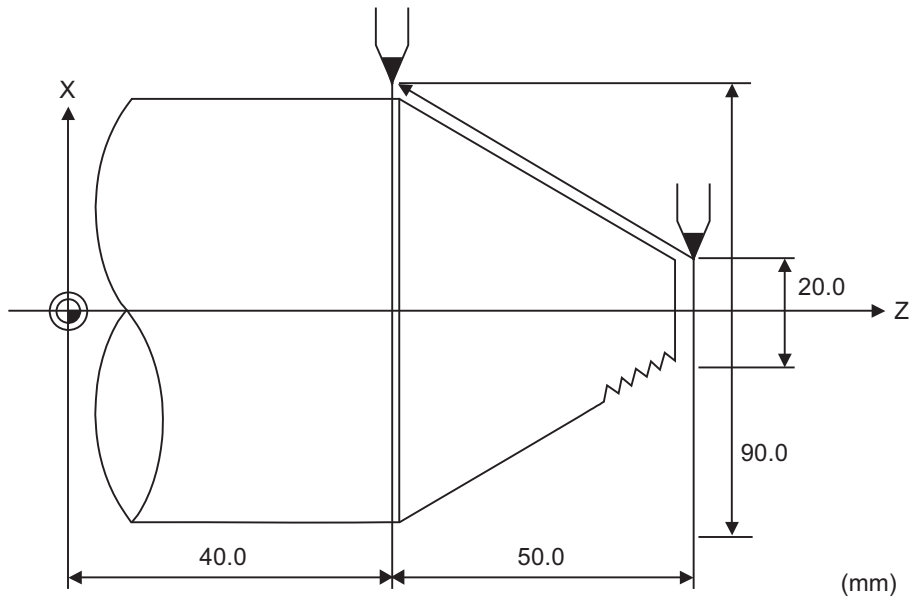
(Note) It is not possible to assign a lead where the feedrate as converted into feed per minute exceeds the maximum cutting feedrate.

- (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 $\delta 1$ and $\delta 2$ to the required thread length.
- (9) The spindle rotation speed is subject to the following restriction :

$$1 \leq R \leq \text{Maximum feed rate/Thread lead}$$
 Where $R \leq$ 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) 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.
- (11) Synchronous feed applies for the thread cutting commands even with an asynchronous feed command (G94).
- (12) Spindle override is invalid and the speeds are fixed to 100% during thread cutting.
- (13) When a thread cutting command is programmed during nose R compensation, the compensation is temporarily canceled and the thread cutting is executed.
- (14) 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.
- (15) 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.
- (16) The thread cutting command waits for the single rotation synchronization signal of the rotary encoder and starts movement.
Note that when using multiple part systems, if one part system issues a thread cutting command during ongoing thread cutting by another part system, the movement will start without waiting for the rotary encoder single rotation synchronization signal. Therefore, carry out timing synchronization between part systems before issuing a thread cutting command with multiple part systems.
- (17) The thread cutting start shift angle is not modal. If there is no Q command with G33, this will be handled as "Q0".
- (18) If a value exceeding 360.000 is command in G33 Q, this will be handled as "Q360.000".
- (19) G33 cuts one row with one cycle. To cut two rows, change the Q value, and issue the same command.



Program example



G33 X90.0 Z40.0 E12.34567 ;	Absolute value command
G33 U70.0 W-50.0 E12.34567 ;	Incremental value command

6.6.2 Inch Thread Cutting ; G33



Function and purpose

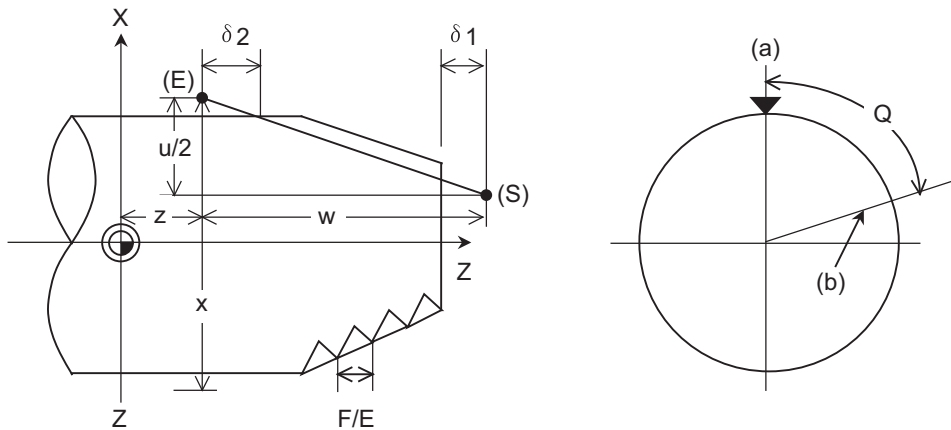
If the number of ridges per inch in the long axis direction is assigned in the G33 command, the feed of the tool synchronized with the spindle rotation will be controlled, which means that constant-lead straight thread-cutting, tapered thread-cutting, and continuous thread-cutting can be performed.



Command format

G33 Z/W__ X/U__ E__ Q__ ; ... Inch thread cutting

Z,W,X,U	Thread end point
E	Number of ridges per inch in the long axis direction (axis which moves the most) (decimal point command can also be assigned)
Q	Thread cutting start shift angle, 0.001 to 360.000°



- $\delta 1 >$ Illegal lead at start of thread cutting
- $\delta 2 >$ Illegal lead at end of thread cutting
- (S) Start point (E) End point
- (a) One-rotation (b) Thread cutting start position
- synchronization signal

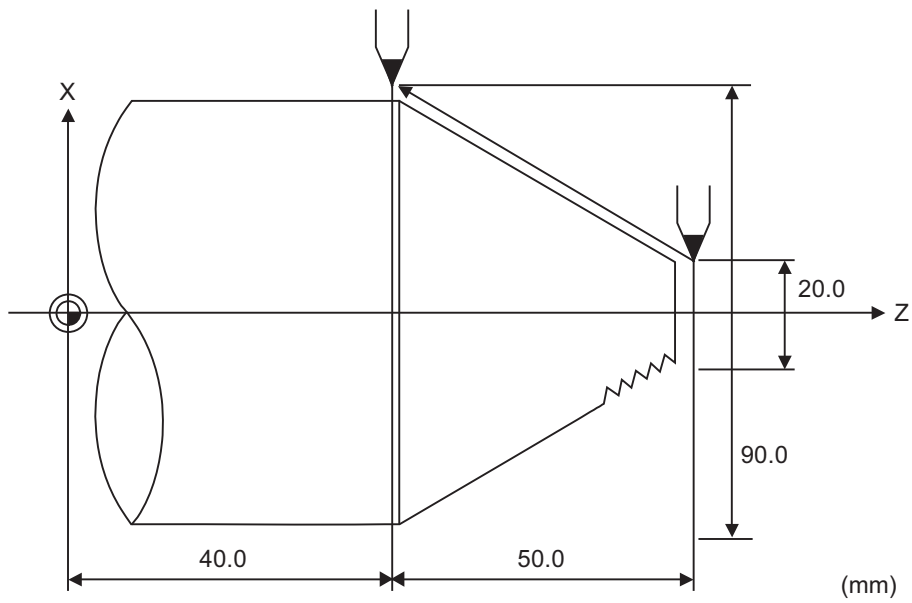


Detailed description

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



Program example



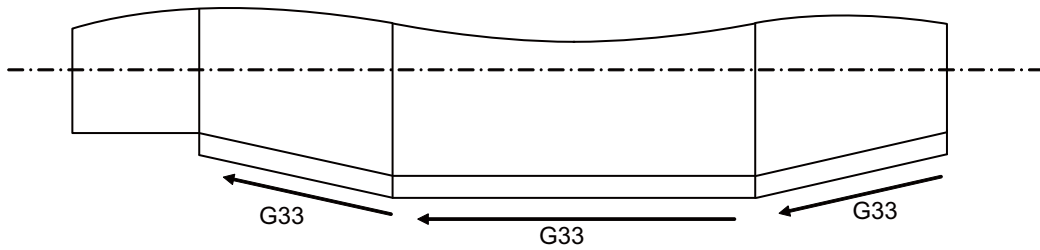
G33 X90.0 Z40.0 E12.0 ;	Absolute value command
G33 U70.0 W-50.0 E12.0 ;	Incremental value command

6.6.3 Continuous Thread Cutting ; G33



Function and purpose

Continuous thread cutting is possible by assigning thread cutting commands continuously. In this way, it is possible to cut special threads whose lead or shape changes.



6.6.4 Variable Lead Thread Cutting ; G34



Function and purpose

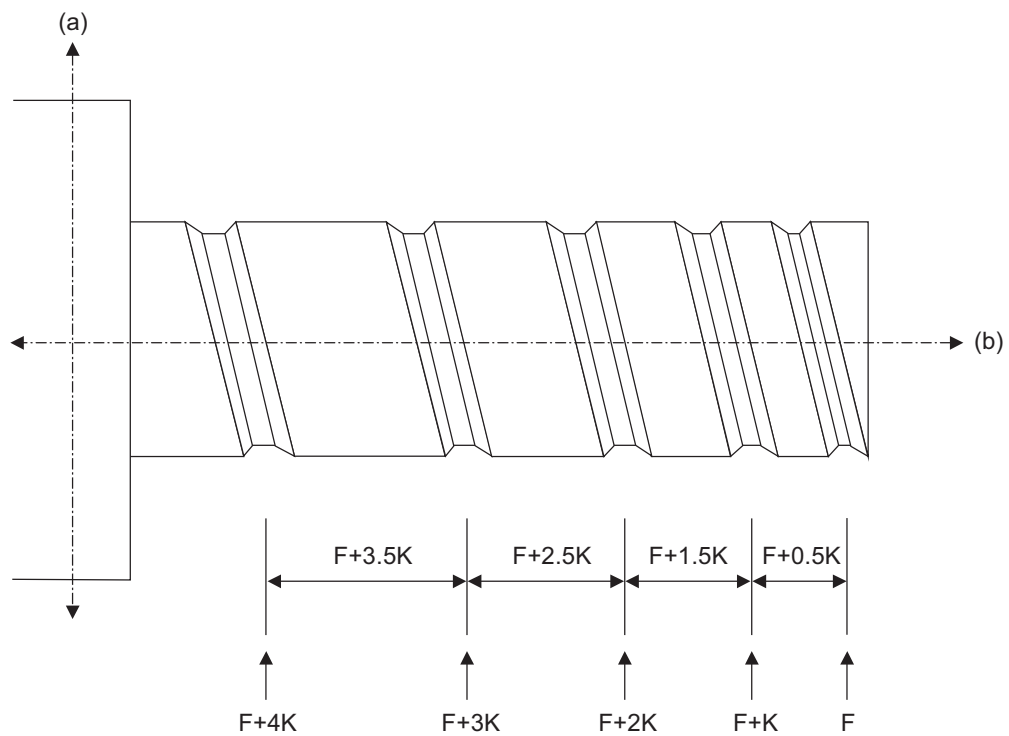
Variable lead thread cutting is enabled by a command specifying a lead increment or decrement amount per turn of the screw.



Command format

G34 X/U__ Z/W__ F/E__ K__ ; ... Variable lead thread cutting

X/U Z/W	Thread end point
F/E	Standard screw lead
K	Lead increment or decrement amount per turn of the screw



(a) Non-lead axis

(b) Lead axis

(F) Lead speed



Detailed description

- (1) The command range is as shown below.
Thread cutting metric input

Input unit system	B (0.001mm)		C (0.0001mm)		B/C
	Command address	F (mm/rev)	E (mm/rev)	F (mm/rev)	
Least command increments	1 (=0.0001) (1.=1.0000)	1 (=0.00001) (1.=1.00000)	1 (=0.00001) (1.=1.00000)	1 (=0.000001) (1.=1.000000)	K (n*mm/rev) n: Number of pitches Same as F or E (signed)
Command range	0.0001 to 999.9999	0.00001 to 999.99999	0.00001 to 99.99999	0.000001 to 99.999999	

Thread cutting inch input

Input unit system	B (0.0001inch)		C (0.00001inch)		C (0.00001inch)
	Command address	F (inch/rev)	E (inch/rev)	F (inch/rev)	
Least command increments	1 (=0.000001) (1.=1.000000)	1 (=0.0000001) (1.=1.0000000)	1 (=0.0000001) (1.=1.0000000)	1 (=0.00000001) (1.=1.00000000)	K (n*inch/rev) n: Number of pitches Same as F or E (signed)
Command range	0.000001 to 99.999999	0.0000001 to 9.9999999	0.0000001 to 9.9999999	0.00000001 to 0.99999999	

- (2) A positive value of K indicates incremental pitches.
Movement amount of one block (n pitches) = (F + K) + (F + 2K) + (F + 3K) + ... + (F + nK)
- (3) A negative value of K indicates decremental pitches.
Movement amount of one block (n pitches) = (F - K) + (F - 2K) + (F - 3K) + ... + (F - nK)
- (4) A program error will occur if the thread lead is not set correctly.

Error No.	Details	Remedy
P 93	Illegal pitch value (1) An invalid value is specified for F/E or K in a thread cutting command. (2) The last lead goes outside of the F/E command range.	The last lead goes outside of the F/E command range. LL : Last lead NP : Number of pitches $LL = \sqrt{F^2 + 2KZ}$ $NP = (-F + LL) / K$

- (5) The other details are the same as G33.
Refer to "Constant lead thread cutting ; G33".

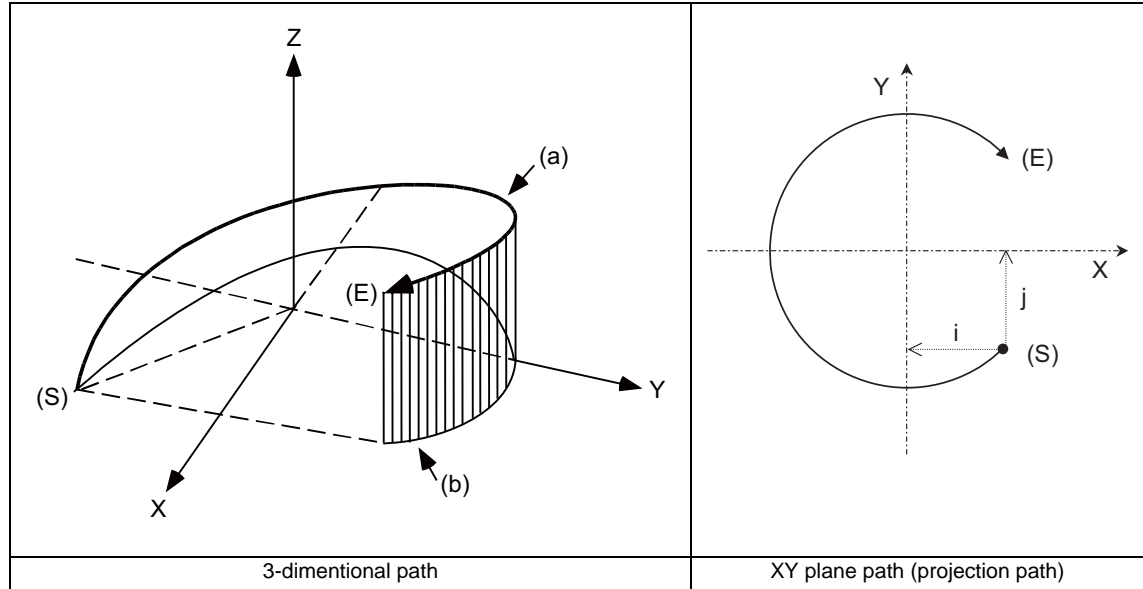
6.7 Helical Interpolation ; G17,G18,G19 and G02,G03



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

(Note) Geometric IB function uses P,Q and A commands on an arc center coordinate (absolute command). So when "#1082 Geomet" is set to "2" (either using geometric I or IB), a program error (P33) will occur and a helical compensation cannot be executed.



Command format

G17/G18/G19 G02(G03) Xx/Uu Yy/Vv Zz/Ww Ii Jj Pp Ff ; ... Helical interpolation

G17/G18/G19 G02(G03) Xx/Uu Yy/Vv Zz/Ww Rr Ff ; ... Helical interpolation

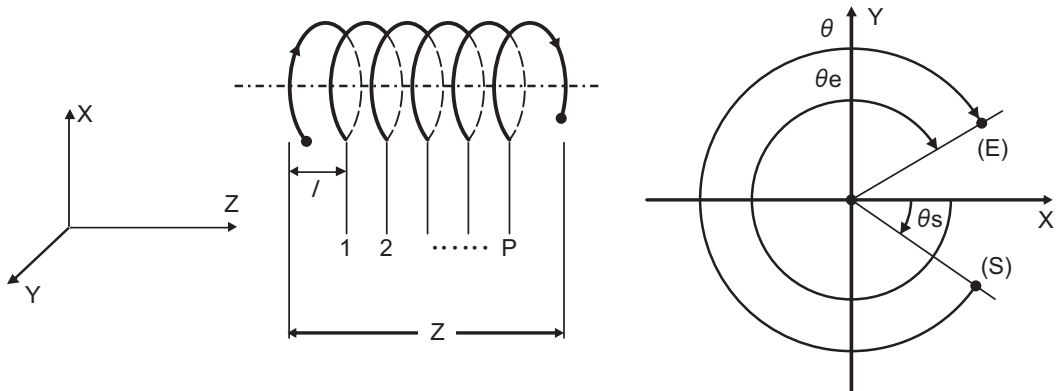
G17/G18/G19	Arc plane (G17: X-Y plane, G18: Z-X plane, G19: Y-Z plane)
G02(G03)	Arc rotation direction (G02: clockwise, G03: counterclockwise)
Xx/Uu, Yy/Vv	Arc end point coordinates
Zz/Ww	Linear axis end point coordinates
Ii, Jj	Arc center coordinates
Pp	Pitch No.
Rr	Arc radius
Ff	Feedrate

(Note 1) In this manual, the following setting descriptions are used: I axis: X, J axis: Y, K axis: Z

(Note 2) Pitch No. can be commanded only when Geometric IB is not performed (#1082 Geomet=0 or 1).



Detailed description



(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.

$$l = \frac{Z}{P + \theta/2\pi}$$

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

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. However, when the starting point is same as the end point, the pitch No. is 1.

(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. P command (pitch command) will be ignored.

- (5) Plane selection

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

XY plane circular, Z axis linear

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

ZX plane circular, Y axis linear

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

YZ plane circular, X axis linear

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

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

UY plane circular, Z axis linear

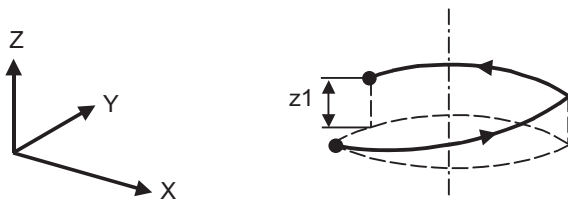
Command the U, Y and Z axis addresses in the G02 (G03) and G17 (plane selection G code) mode.

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



Program example

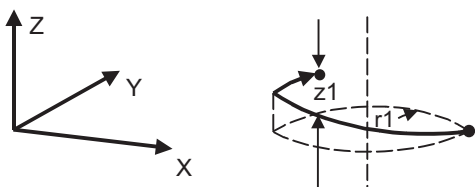
(Example 1)



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

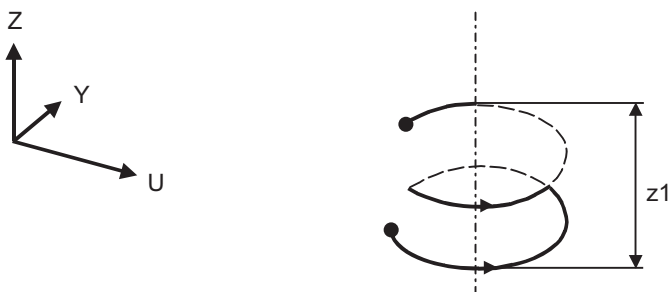
(Note) If pitch No. is 0, address P can be omitted.

(Example 2)



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

(Example 3)



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

(Example 4)



G18 G03 Xx1 Uu1 Zz1 Ii1 Kk1 Ff1;	ZX plane arc, U axis linear
----------------------------------	-----------------------------

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

(Example 5)

G18 G02 Xx1 Uu1 Yy1 Zz1 Ii1 Jj1 Kk1 Ff1;	ZX plane arc, U axis, Y axis linear (The J command is ignored)
--	---

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



Precautions and restrictions

- (1) When Geometric IB is not performed (#1082 Geomet-0 or 1), the pitch No. command of helical interpolation can be issued.
- (2) Up to the number of simultaneous contouring control axes can be commanded simultaneously.
- (3) With helical interpolation, the two axes that configure the plane is the circular interpolation axis, and the other axes are the linear interpolation axes.
- (4) If a helical interpolation is commanded after the corner chamfering or corner R command is commanded or the corner chamfering or corner R command is commanded after the helical interpolation is commanded, the movement of the linear interpolation axis is stopped and only the circular interpolation axis operates during the corner chamfering or corner R commands.
- (5) This function is an option. If it is executed without it, a program error (P72) will occur.
- (6) A program error (P60) will occur if the commanded movement distance (circumference * pitch No. + the last circumference) is excessive.

The P command's program error (P35) is prioritized over a program error (P60).

7

Feed Functions

7.1 Rapid Traverse Rate



Function and purpose

The rapid traverse rate can be set independently for each axis. The available speed ranges are from 1 mm/min to 1000000mm/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. 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.

(Note) Rapid traverse override

Override can be applied by an external input signal for both manual and automatic rapid traverse.

There are 2 types which are determined by the PLC specifications.

Type1 : Override in 4 steps: 1%, 25%, 50%, 100%

Type2 : Override in 1% steps from 0% to 100%

7.2 Cutting Feedrate



Function and purpose

This function specifies the feedrate of the cutting commands, and a feed amount per spindle rotation or feed amount per minute is commanded. Once commanded, it is stored in the memory as a modal value. The feedrate modal is cleared to zero only when the power is turned ON. The maximum cutting feedrate is clamped by the cutting feedrate clamp parameter (whose setting range is the same as that for the cutting feedrate).

The cutting feedrate is assigned with address F and numerals.

The cutting feedrate is valid for the G01, G02, G03, G33 and G34 commands.

Examples (asynchronous feed)

Feedrate					
G1	X100.	Z100.	F200 ;	200.0mm/min	F200 or F200.000 gives the same rate.
G1	X100.	Z100.	F123.4 ;	123.4mm/min	
G1	X100.	Z100.	F56.789 ;	56.789mm/min	

Speed range that can be commanded (when input setting unit is 1 μ m or 10 μ m)

Command mode	Feedrate command range	Feed rate command range	Remarks
mm/min	0.001 to 1000000.000	0.001 to 1000000.000	
inch/min	0.0001 to 39370.0787	0.0001 to 39370.0787	
° /min	0.001 to 1000000.000	0.001 to 1000000.000	

(Note 1) A program error (P62) will occur when there is no F command in the first cutting command (G01, G02, G03, G33, G34) after the power has been turned ON.

7.3 F1-digit Feed



Function and purpose

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

When F0 is assigned, the rapid traverse rate is established and the speed is the same as for G00. (G modal does not change.)

When F1 to F5 is assigned, the feedrate set to correspond to the command serves as the command value. The command greater than F6 is considered to be the normal cutting feedrate.

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

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



Detailed description

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

Operation alarm "104" will occur when the feedrate is 0.

Operation method

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

Special notes

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

F0	Rapid traverse rate
F1 to F5	F1-digit
F6 or more	Normal cutting feedrate command
- (2) F1 to F5 are invalid in the G00 mode and the rapid traverse rate is established instead.
- (3) If F0 is used in the G02 or G03 mode, a program error (P121) will occur.
- (4) When F1. to F5. (with decimal point) are assigned, the 1mm/min to 5mm/min direct commands are established instead of the F1-digit command.
- (5) When the commands are used with the millimeter or degree units, the feedrate set to correspond to F1 to F5 serves as the assigned speed mm (°)/min.
- (6) When the commands are used with inch units, one-tenth of the feedrate set correspond to F1 to F5 serves at the assigned speed inch/min.
- (7) During a F1-digit command, the F1-digit number and F1-digit command signal are output as the PLC interface signals.

F1-digit and G commands

- (1) 01 group G command in same block as F1-digit commands

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

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

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



Function and purpose

Feed per minute (asynchronous feed)

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

Feed per revolution (synchronous feed)

By issuing the G95 command, the commands from that block are issued directly by the numerical value following F as the feedrate per spindle revolution (mm/rev, inch/rev).

When this command is used, the rotary encoder must be attached to the spindle.



Command format

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

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



Detailed description

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 command unit system	B(0.001mm)		C(0.0001mm)	
	Per-minute feed	Per-revolution feed	Per-minute feed	Per-revolution feed
Command mode				
Command address	F(mm/min)	F(mm/rev)	F(mm/min)	F(mm/rev)
Least command increments	1 (= 1.000), (1. = 1.000)	1 (= 0.0001), (1. = 1.0000)	1 (= 1.0000), (1. = 1.0000)	1 (= 0.00001), (1. = 1.00000)
Command range	0.001 to 1000000.000	0.0001 to 999.9999	0.0001 to 100000.0000	0.00001 to 99.99999

Inch input

Input command unit system	B (0.0001inch)		C (0.00001inch)	
	Per-minute feed	Per-revolution feed	Per-minute feed	Per-revolution feed
Command mode	F(inch/min)	E (inch/rev)	F(inch/min)	E (inch/rev)
Least command increments	1 (= 0.01), (1. = 1.0000)	1 (= 0.000001), (1. = 1.000000)	1 (= 0.000001), (1. = 1.000000)	1 (= 0.0000001), (1. = 1.0000000)
Command range	0.0001 to 39370.0787	0.000001 to 99.999999	0.00001 to 3937.00787	0.0000001 to 9.9999999

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

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

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, operation alarm "105" occurs.
- (4) Feedrate during the machine lock is the command speed.
- (5) Under dry run conditions, feed per minute applies and movement results at the externally set rate (mm/min or inch/min).
- (6) 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.

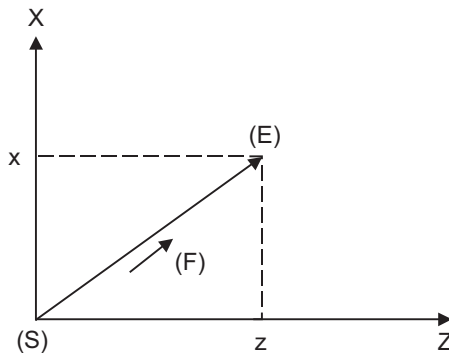


Detailed description

When controlling linear axes

Even when only one machine axis is to be controlled or 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 Z) are to be controlled:



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

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

(S) Tool start point

(E) Tool end point

(F) Speed in this direction is "f".

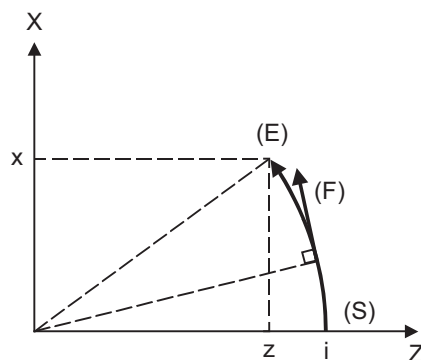
f_x : Feedrate for X axis

f_z : Feedrate for Z 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.

(Note) When the circular interpolation function is used and the tool is moved along the circumference of an arc by the linear control axis, 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".

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

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

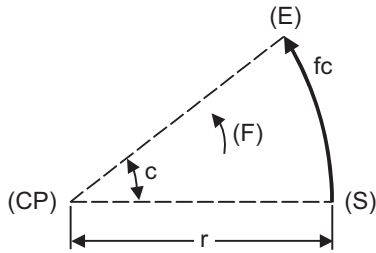
When controlling rotary axes

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

Consequently, the cutting feed in the tool advance direction, or in other words the linear speed, varies according to the distance between the center of rotation and the tool.

This distance must be borne in mind when designating the feedrate in the program.

(Example) When the feedrate is designated as "f" and rotary axis (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

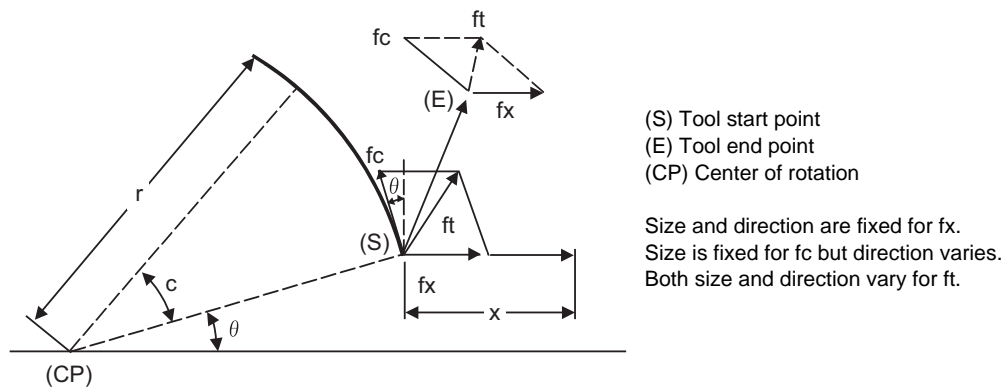
The controller proceeds in exactly the same way whether linear or rotary axes are to be controlled.

When a rotary axis is to be controlled, the numerical value assigned by the coordinate word (C, H) 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 "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 value is "c":



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

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

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

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

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

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:

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

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

Where 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:

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

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

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

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

"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 Thread Cutting Mode



Function and purpose

F command or E commands for thread leads can be issued for the thread cutting mode (G33, G34, G76 G78 commands).

The thread leads command range is shown below.

Thread cutting metric input

Input unit system	B(0.001mm)			C(0.0001mm)		
Command address	F(mm/rev)	E (mm/rev)	E(ridges/inch)	F(mm/rev)	E(mm/rev)	E(ridges/inch)
Least command increments	1 (= 0.0001), (1.=1.0000)	1 (= 0.00001), (1.=1.00000)	1 (= 1.00), (1.=1.00)	1 (= 0.00001), (1.=1.00000)	1(=0.000001), (1.=1.000000)	1 (= 1.000), (1.=1.000)
Command range	0.0001 to 999.9999	0.00001 to 999.99999	0.03 to 999.99	0.00001 to 99.99999	0.000001 to 99.999999	0.255 to 9999.999

Thread cutting inch input

Input unit system	B(0.0001inch)			C(0.00001inch)		
Command address	F(inch/rev)	E(inch/rev)	E(ridges/inch)	F(inch/rev)	E(inch/rev)	E(ridges/inch)
Least command increments	1(=0.000001), (1.=1.000000)	1(=0.0000001), (1.=1.0000000)	1 (= 1.0000), (1.=1.0000)	1(=0.0000001), (1.=1.0000000)	1(=0.00000001), (1.=1.00000000)	1(=1.00000), (1.=1.00000)
Command range	0.000001 to 99.999999	0.000010 to 9.9999999	0.0101 to 9999.9999	0.0000001 to 9.9999999	0.0000010 to 0.99999999	0.10001 to 999.99999

7.7 Automatic Acceleration/Deceleration

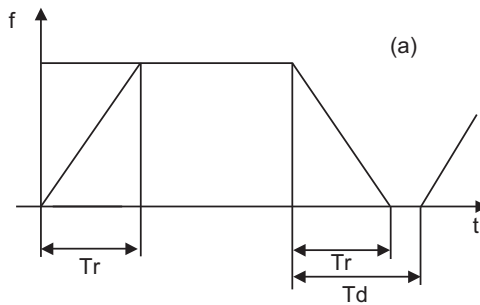


Function and purpose

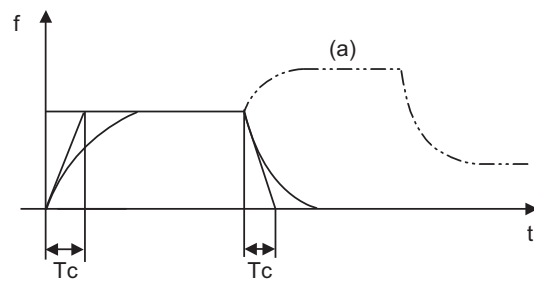
The rapid traverse and manual feed acceleration/deceleration pattern is linear acceleration and linear deceleration. Time constant TR can be set independently for each axis using parameters in 1ms steps from 1 to 500ms.

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

[Rapid traverse acceleration/deceleration pattern]
 (Tr = Rapid traverse time constant)
 (Td = Deceleration check time)



[Cutting feed acceleration/deceleration pattern]
 (Tc = Cutting feed time constant)



(a) With continuous commands

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

7.8 Rapid Traverse Constant Inclination Acceleration/Deceleration



Function and purpose

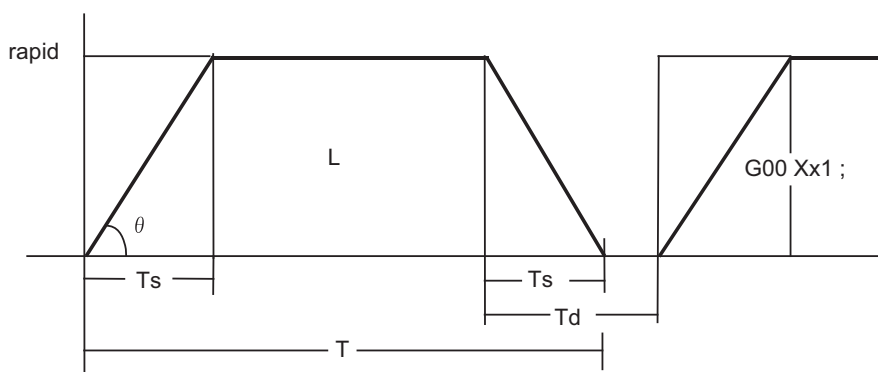
This function performs acceleration and deceleration at a constant inclination during linear acceleration/ deceleration in the rapid traverse mode. Compared to the method of acceleration /deceleration after interpolation, the constant inclination acceleration/deceleration method makes for improved cycle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



Detailed description

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

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



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

$$T_d = T_s + (0 \sim 1.7\text{ms})$$

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

rapid : Rapid traverse rate

Ts : Acceleration/deceleration time constant

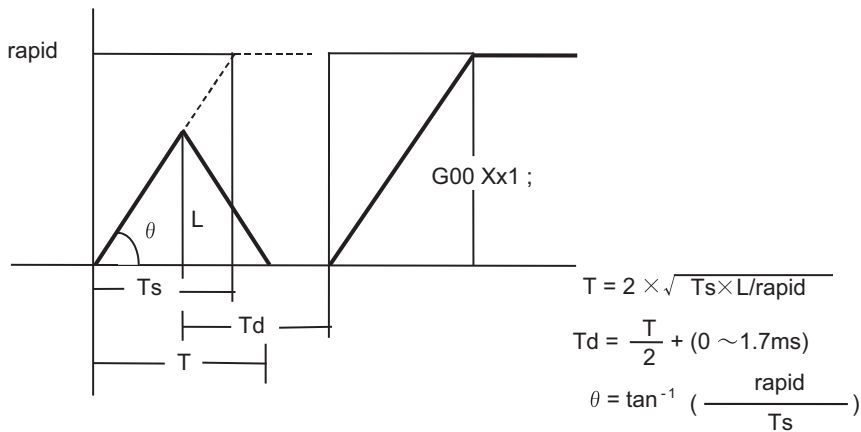
Td : Command deceleration check time

θ : Acceleration/deceleration inclination

T : Interpolation time

L : Interpolation distance

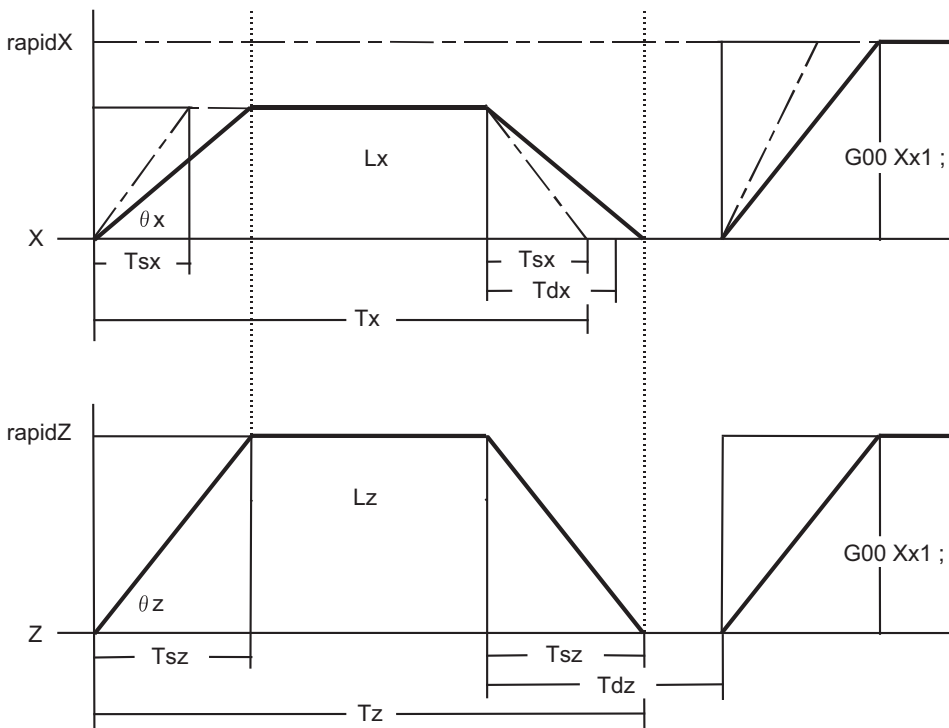
[When the interpolation distance is shorter than the acceleration and deceleration distance]



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

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

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



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

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

7.9 Speed Clamp



Function and purpose

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

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

7.10 Exact Stop Check ; G09



Function and purpose

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

Either the deceleration check time or in-position state is selected with the parameter. (Refer to the section "Deceleration check")

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



Command format

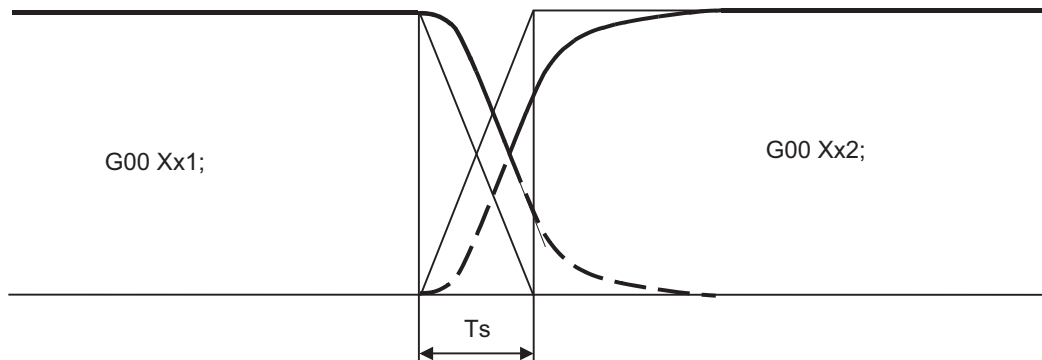
G09 G01 (G02, G03) ; ... Exact stop check

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

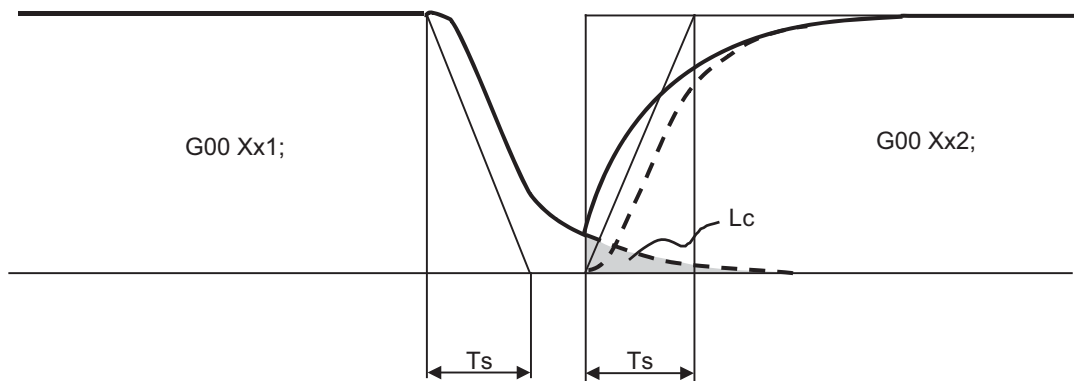


Detailed description

[With continuous cutting feed]



[With cutting feed in-position check]



Ts : Cutting feed acceleration/deceleration time constant

Lc : In-position width

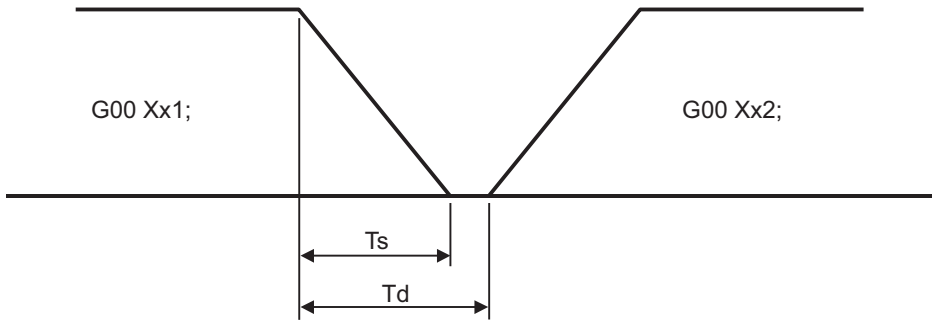
As shown in the figure below, the remaining distance (shaded area in the above figure) of the previous block when the next block is started can be set into the servo parameter "#2224 sv024" as the in-position width "Lc". The setting unit for the servo parameter "#2224 SV024" is 0.0005mm or 0.00005inch.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



To eliminate corner roundness, set the value as small as possible to servo parameter "#2224 sv024" and perform an in-position check or assign the dwell command (G04) between blocks.

With deceleration check

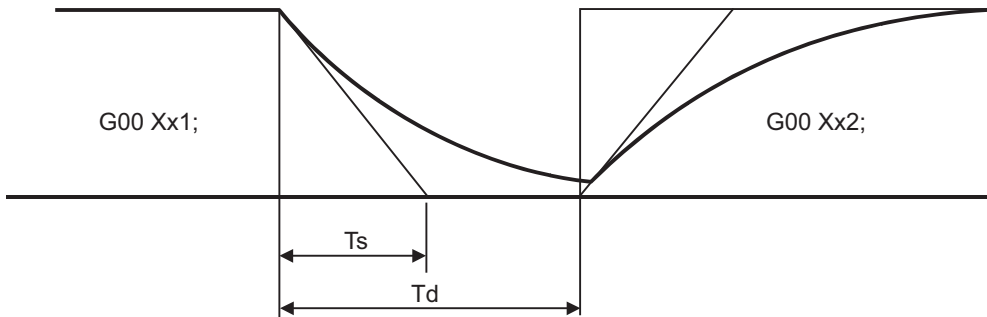
(1) With linear acceleration/deceleration



TS: Acceleration/deceleration time constant

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

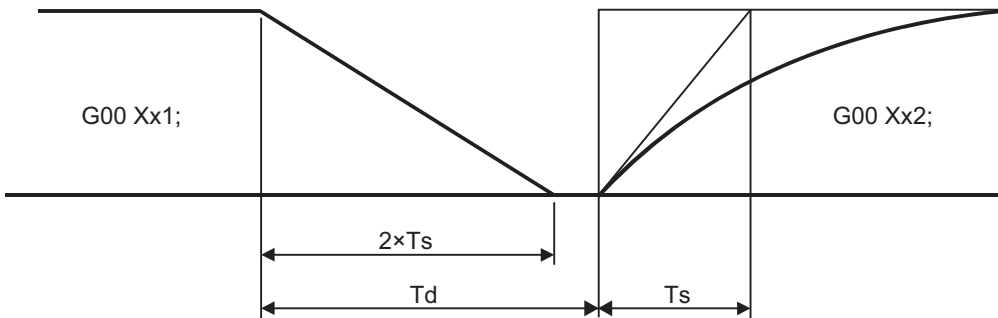
(2) With exponential acceleration/deceleration



TS: Acceleration/deceleration time constant

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

(3) With exponential acceleration/linear deceleration



TS: Acceleration/deceleration time constant

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

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.

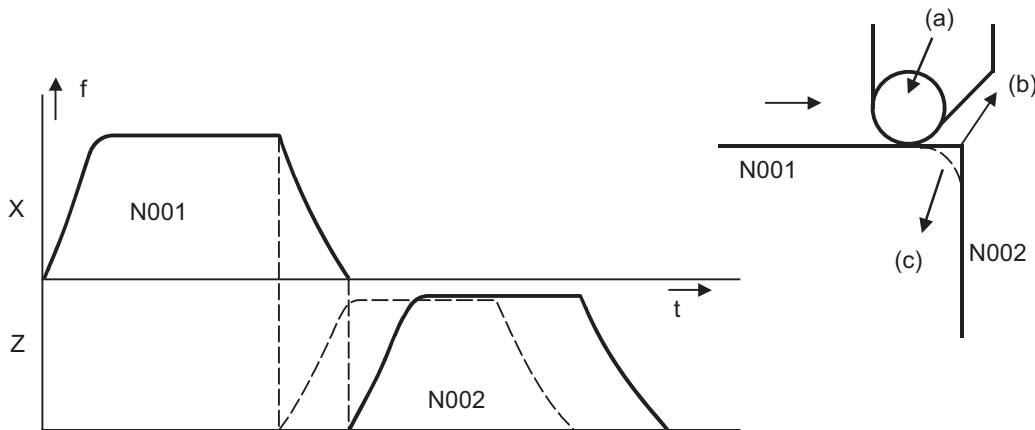
(Note 1) To execute exact stop check in a fixed cycle cutting block, insert command G09 into the fixed cycle subprogram.



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 Z100.000 ;	

[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

7.11 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.

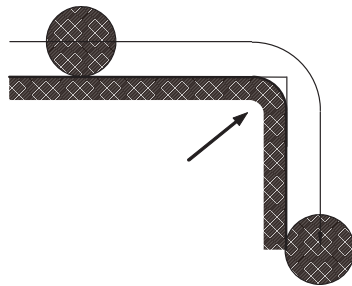
7.12 Deceleration Check



Function and purpose

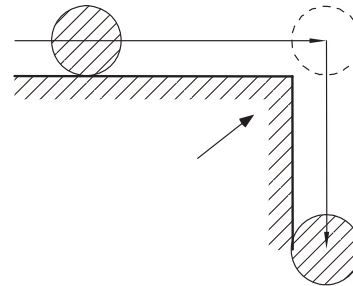
With the deceleration check function, a deceleration stop is executed at the block seam before the next block is executed, preventing corner rounding by reducing the machine shock that occurs when the control axis feedrate is suddenly changed.

```
N010 G90 G01 Z100 ;
N011 G01 X-50 ;
```



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

```
N010 G09 G90 G01 Z100 ;
N011 G01 X-50 ;
```



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



Detailed description

Conditions for Executing the Deceleration Check

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

Deceleration Check and Parameters

Select the deceleration check method with these parameters.

[Base specification parameter] #1193 inpos Deceleration check method selection

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

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

Operation when G0 acceleration/deceleration before interpolation is valid

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

List of parameters for each axis

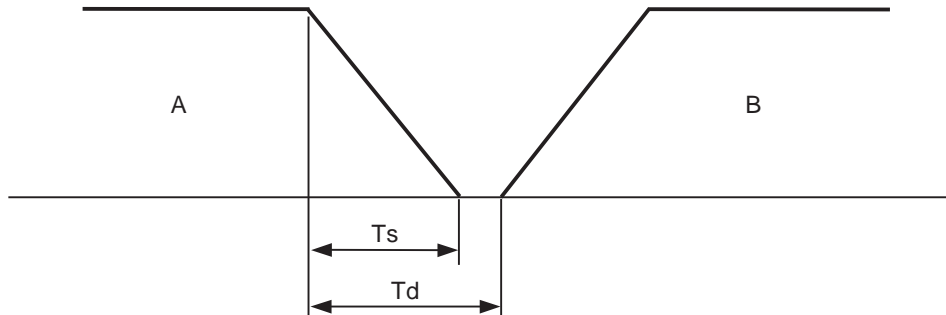
#	Item		Detail	Setting range (unit)
2077	G0inps	G0 in-position width	This parameter becomes valid when executing an in-position check at rapid traverse rate. Between SV024 and this parameter, the parameter with a larger value will be applied. (Note) This parameter is valid when executing an in-position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

Deceleration Check Method

(1) Command deceleration check

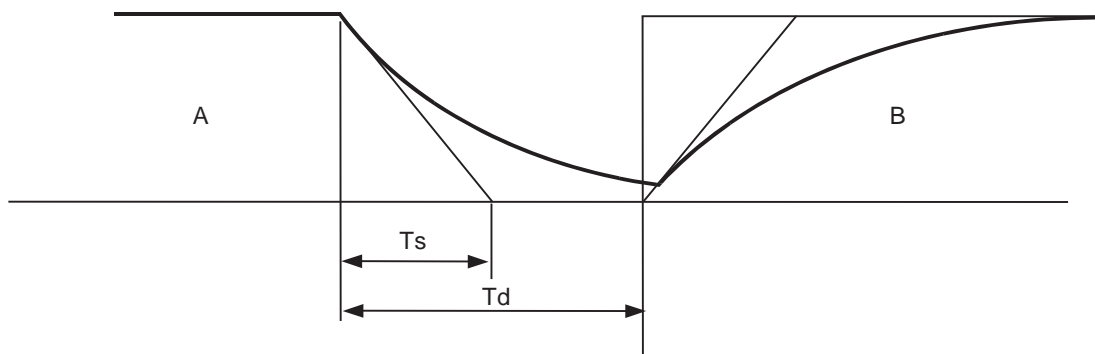
After interpolation for one block has been completed, the completion of the command system deceleration is confirmed before execution of the next block. The time required for the deceleration check is determined according to the acceleration/deceleration mode and acceleration/deceleration time constant.

(a) For linear acceleration/deceleration



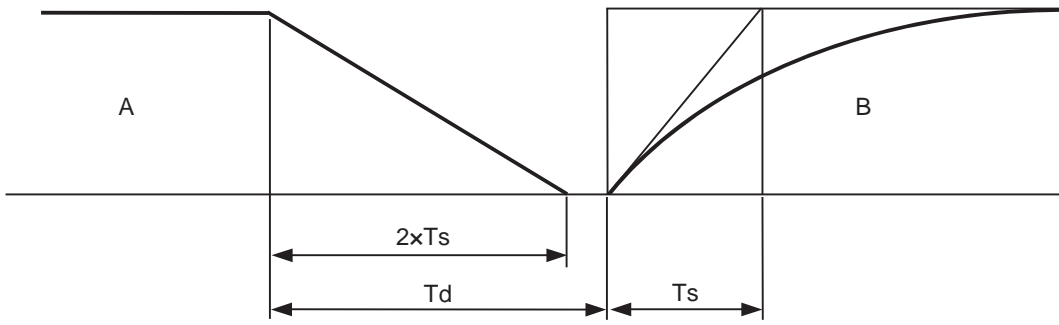
A: Previous block B: Next block
 T_s : Acceleration/deceleration time constant
 T_d : Deceleration check time $T_d = T_s + (0 \text{ to } 7\text{ms})$

(b) For exponential acceleration/deceleration



A: Previous block B: Next block
 T_s : Acceleration/deceleration time constant
 T_d : Deceleration check time $T_d = T_s + (0 \text{ to } 7\text{ms})$

(c) For exponential acceleration and linear deceleration

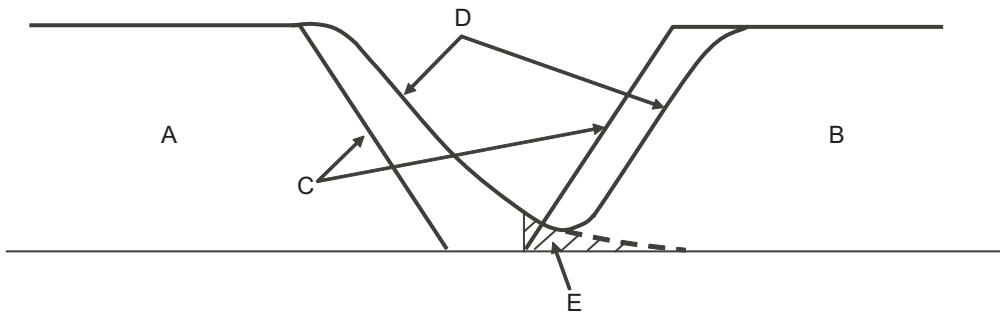


A: Previous block B: Next block
 Ts : Acceleration/deceleration time constant
 Td : Deceleration check time $Td = Ts + (0 \text{ to } 7\text{ms})$

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

(2) In-position check

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



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

7.13 Automatic Corner Override ; G62



Function and purpose

When cutting with nose R compensation, to prevent machining surface distortion due to the increase in the cutting load during cutting of corners, this command automatically applies an override on the cutting feedrate so that the cutting amount is not increased for a set time at the inside corner or automatic corner R.

Automatic corner override is valid until the nose R compensation cancel (G40), exact stop check mode (G61), tapping mode (G63), or cutting mode (G64) command is issued.



Command format

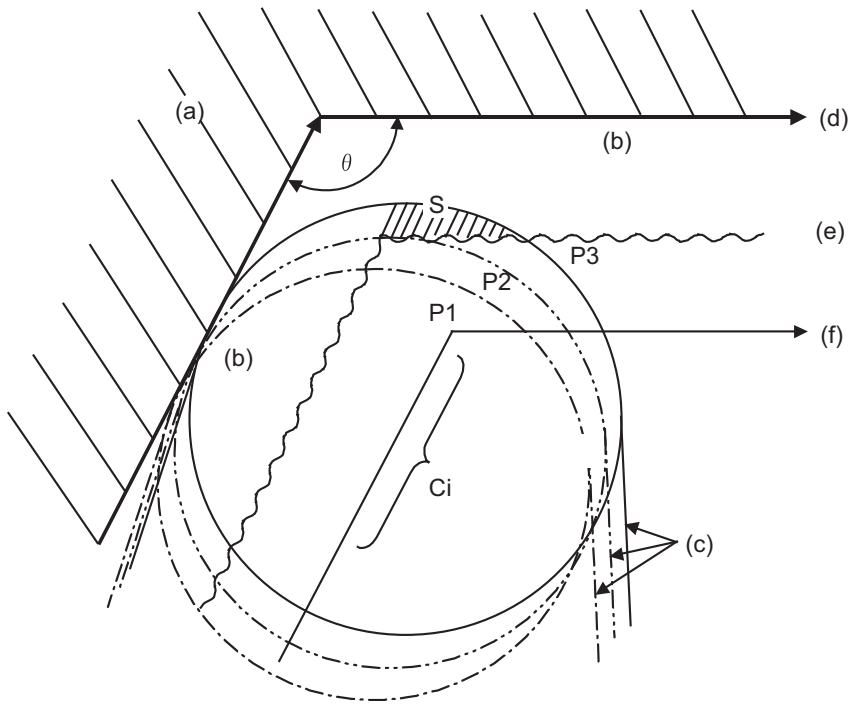
G62 ; ... Automatic Corner Override
--



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) Nose R center path
- θ : Max. angle at inside corner
- Ci : Deceleration range (IN)

[Operation]

- (1) When automatic corner override is not to be applied :
When the tool moves in the order of P1 -> P2 -> P3 in the above figure, the machining allowance at P3 increase by an amount equivalent to the area of shaded section S and so that tool load increases.
- (2) When automatic corner override is to be applied :
When the inside corner angle θ in the above figure is less than the angle set in the parameter, the override set into the parameter is automatically applied in the deceleration range Ci.

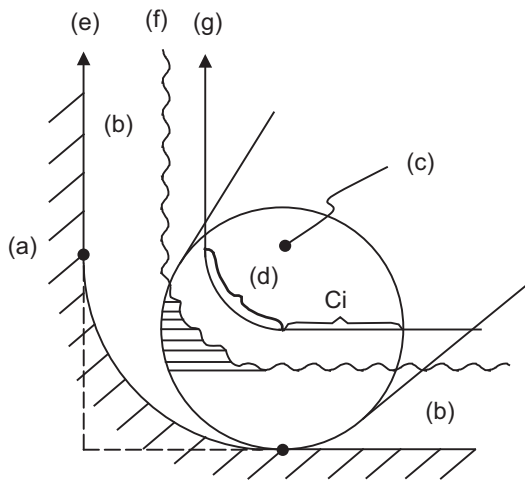
[Parameter setting]

The following parameters are set into the machining parameters :

#	Parameter	Setting range
#8007	OVERRIDE	0 to 100 [%]
#8008	Max. angle at inside corner	0 to 180 [°]
#8009	DSC. ZONE	0 to 99999.99 [mm] or 0 to 3937.000 [inch]

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

Automatic corner R



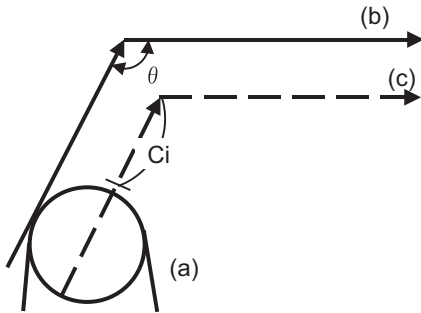
- | | | |
|------------------------|-------------------------|-----------------------------|
| (a) Workpiece | (b) Machining allowance | (c) Corner R center |
| (d) Corner R section | (e) Programmed path | (f) Workpiece surface shape |
| (g) Nose R center path | | |

- (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

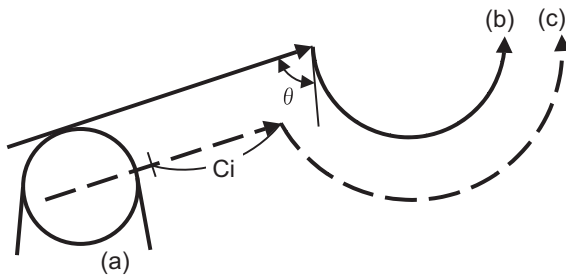
(1) Linear - linear corner



- (a) Tool
- (b) Program
- (c) Nose R center

The override set in the parameter is applied at Ci.

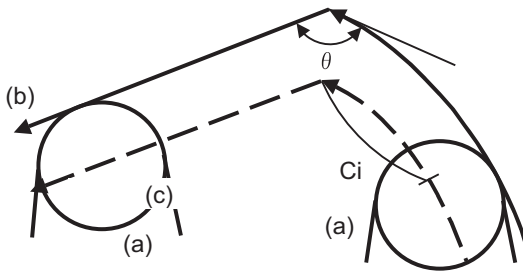
(2) Linear - arc (outside offset) corner



- (a) Tool
- (b) Program
- (c) Nose R center

The override set in the parameter is applied at Ci.

(3) Arc (inside offset) - linear corner

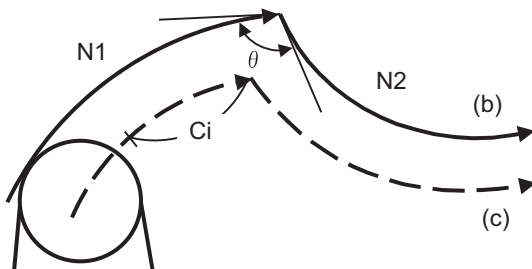


- (a) Tool
- (b) Program
- (c) Nose R center

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) Nose R 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 feed per revolution.
Thread cutting	Automatic corner override is invalid.
G31 skip	Program error occurs with G31 command during nose R compensation.
Machine lock	Valid
Machine lock high speed	Automatic corner override is invalid.
G00	Invalid
G01	Valid
G02,G03	Valid

7.14 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

7.15 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.

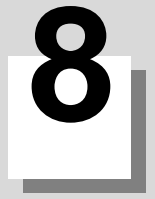
G64 is released by the exact stop check mode (G61), automatic corner override (G62), or tapping mode (G63).

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/U_/P_ ; ... Dwell (Time designation)

X/U/P	Dwell time
-------	------------

The input command unit for the dwell time depends on the parameter.

In addition to the address P and X, the address U (actually, the address corresponding to the X-axis designated with the #1014 incax) can be used. Note that this is invalid when the #1076 AbsInc is set to 0.

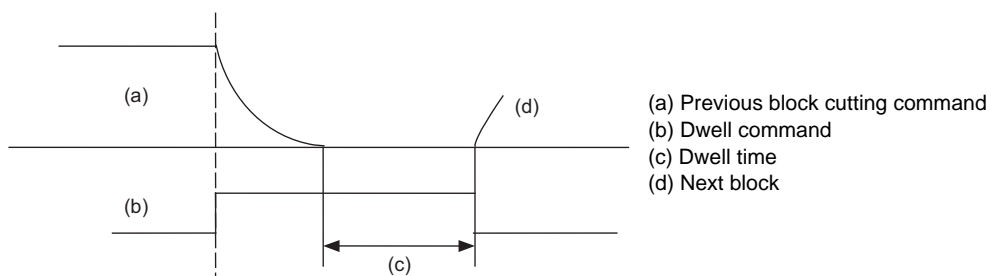


Detailed description

- (1) When designating the dwell time with X or U, the decimal point command is valid.
- (2) 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)

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





Program example

Command	Dwell time [s]	
	#1078 Decpt2 = 0	#1078 Decpt2 = 1
G04 X500 ;	0.5	500
G04 X5000 ;	5	5000
G04 X5. ;	5	5
G04 X#100 ;	1000	1000
G04 U500 ;	0.5	500
G04 U5000 ;	5	5000
G04 U5. ;	5	5
G04 U#100 ;	1000	1000
G04 P5000 ;	5	5000
G04 P12.345 ;	12.345	12.345
G04 P#100 ;	1000	1000

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

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

(Note 2) If the input setting unit is 0.0001inch, the X before G04 will be multiplied by 10. For example for "X5. G04 ;", the dwell time will be 50 seconds.



Precautions and restrictions

- (1) When using this function, command X or U after G04 in order to make sure that the dwell is based on X or U.



Miscellaneous Functions

9.1 Miscellaneous Functions (M8-digits)



Function and purpose

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



Detailed description

These functions are designated by an 8-digit number (0 to 99999999) following the address M with this controller, and up to 4 groups can be commanded in a single block.

(Example) G00 Xx Mm1 Mm2 Mm3 Mm4 ;

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

The output signal is an 8-digit BCD code and start signal.

The eight commands of M00, M01, M02, M30, M96, M97, M98 and M99 are used as auxiliary commands for specific objectives and so they cannot be used as general auxiliary commands. Therefore, 92 miscellaneous functions are available.

Reference should be made to the instructions issued by the machine manufacturer for the actual correspondence between the functions and numerical values.

When the M00, M01, M02, and M30 functions are used, the next block is not read into the pre-read buffer due to pre-read inhibiting.

If the M function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which sequence applies.

- (1) The M function is executed after the movement command.
- (2) The M function is executed at the same time as the movement command.

Processing and completion sequences are required in each case for all M commands except M96, M97, M98 and M99.

Program stop : M00

When the NC has read this function, it stops reading the next block. As far as the NC system's functions are concerned, only the reading of the next block is stopped. Whether such machine functions as the spindle rotation and coolant supply are stopped or not differs according to the machine in question.

Re-start is enabled by pressing the automatic start button on the machine operation board.

Whether resetting can be initiated by M00 depends on the machine specifications.

Optional stop : M01

If the tape reader reads the M01 command when the optional stop switch on the machine operation board is ON, reading of the next block will stop and performs the same operation as the M00.

If the optional stop switch is OFF, the M01 command is ignored.

(Example)

N10 G00 X1000 ;

N11 M01 ;

N12 G01 X2000 Z3000 F600 ;

:

Optional stop switch command is ignored.

Stops at N11 when switch is ON

Next command (N12) is executed without stopping at N11 when switch is OFF

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 indexing the machining program. Whether indexing the machining program 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 indexing 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 indexing the machining program 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.

(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.

Indexing operation other than M02/M03 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 second miscellaneous function by a parameter(#1170 M2name). (Except the address that is used for the axis name and the increment command axis name.)

The second miscellaneous function can be issued up to 4 sets in a block.

Whether to BCD output or binary output the second 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.

The table below gives address combinations. It is not possible to use an address which is the same for the axis name of an additional axis and secondary miscellaneous function.

		Additional axis name		
		A	B	C
2nd miscellaneous function	A	-	○	○
	B	○	-	○
	C	○	○	-



Precautions

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

- Geometric command
- Deep hole drilling cycle 2 commands

Spindle Functions

10.1 Spindle Functions



Function and purpose

These functions are assigned with an 8-digit (0 to 99999999) number following the address S, and one group can be assigned in one block.

The output signal is a 32-bit binary data with sign and start signal.

Processing and completion sequences are required for all S commands.

10.2 Constant Surface Speed Control ; G96,G97



Function and purpose

These commands automatically control the spindle rotation speed in line with the changes in the radius coordinate values as cutting proceeds in the diametrical direction, and they serve to keep the cutting point speed constant during the cutting.



Command format

G96 S__ P__ ; ... Constant surface speed ON

S	Surface speed(1 to 999999 m/min)
P	Constant surface speed control axis(Available No. of axes to be controlled within G92 part system)

G97 ; ... Constant surface speed cancel



Detailed description

- (1) The constant surface speed control axis is set by parameter (#1181 G96_ax).
 0: Fixed at 1st axis (P command invalid)
 1: 1st axis
 2: 2nd axis
 3: 3rd axis
- (2) When the above-mentioned parameter is not zero, the constant surface speed control axis can be assigned by address P.
 (Example) G96_ax = 1

Program	Constant surface speed control axis
G96 S100 ;	1st axis
G96 S100 P3 ;	3rd axis

- (3) Example of selection program and operation

G90 G96 G01 X50. Z100. S200 ;	The spindle rotation speed is controlled so that the surface speed is 200m/min.
:	
G97 G01 X50. Z100. F300 S500 ;	The spindle rotation speed is controlled to 500r/min.
:	
M02 ;	The modal returns to the initial value.

- (4) Constant surface speed control can be commanded on the selected spindle (nth spindle) / the 2nd spindle.
 Select which spindle (the selected spindle or 2nd one) the commands are made to by the spindle selection G codes (G43.1 and G44.1).
 Select which spindle (the selected spindle or 2nd one) is valid as the initial state with the parameter (base specifications parameter "#1199 Sselect").
- (5) Select whether calculating the surface speed at rapid traverse command is performed constantly or only at the block end point.



Precautions

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

10.3 Spindle Clamp Speed Setting ; G92



Function and purpose

The maximum clamp rotation speed of the spindle can be assigned by address S following G92 and the minimum clamp rotation speed by address Q.

Use this command when the spindle speed needs to be limited depending on the workpiece to be machined, the chuck to be mounted on the spindle and the tool specifications, etc.



Command format

G92 S__ Q__ ; ... Spindle Clamp Speed Setting

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



Detailed description

- Besides this command, parameters can be used to set the rotation speed range up to 4 stages in 1 r/min units to accommodate gear selection between the spindle and spindle motor. The lowest upper limit and highest lower limit are valid among the rotation speed ranges based on the parameters and based on "G92 Ss Qq ;".
 - Set in the parameters "#1146 Sclamp" and "#1227 aux11/bit5" whether to carry out rotation speed clamp only in the constant surface speed mode or even when the constant surface speed is canceled.
- (Note 1) G92S command and rotation speed clamp operation

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

Spindle clamp speed command can be issued to the nth spindle or the 1st spindle.

Use the spindle selection command G code (G43.1/G44.1) to set to which spindle the command is to be issued.

Use the base specification parameter "#1199 Sselect" to set to which spindle the command is to be issued at the initial state.

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

- The command value of the spindle clamp speed will be cleared by modal reset (reset 2 or reset & rewind).

Note that the modal is retained if the parameter "#1210 RstGmd / bit19" is ON.

When the power is turned ON, the setting will be cleared to 0.



Precautions

- (1) Once the maximum clamp speed and the minimum clamp speed are set using the spindle clamp speed setting (G92 S__ Q __), the maximum speed clamp will not be cancelled even if the command "G92 S0" is issued. Even when G92 S0 is commanded, the value of Q__ is kept enabled and Q__ is greater than S0.
- (2) Note that if the spindle clamp speed setting (G92 S__ Q__) is not commanded, the speed may increase to the machine's maximum specified speed that is set by the parameter. Especially when the constant surface speed control (G96 S__) is commanded, command the spindle clamp speed setting as well as the spindle maximum rotation speed. As the tool moves closer to the spindle center, the spindle rotation speed will increase and may exceed the allowable speed of the workpiece or chuck, etc.

⚠ WARNING

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

10.4 Spindle/C Axis Control



Function and purpose

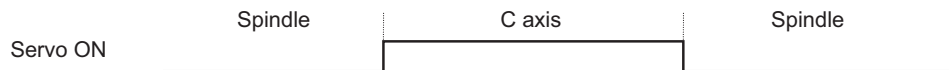
This function enables one spindle to also be used as a C axis (rotation axis) by an external signal.



Detailed description

Spindle/C axis changeover

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



At servo OFF Spindle (C axis control not possible)

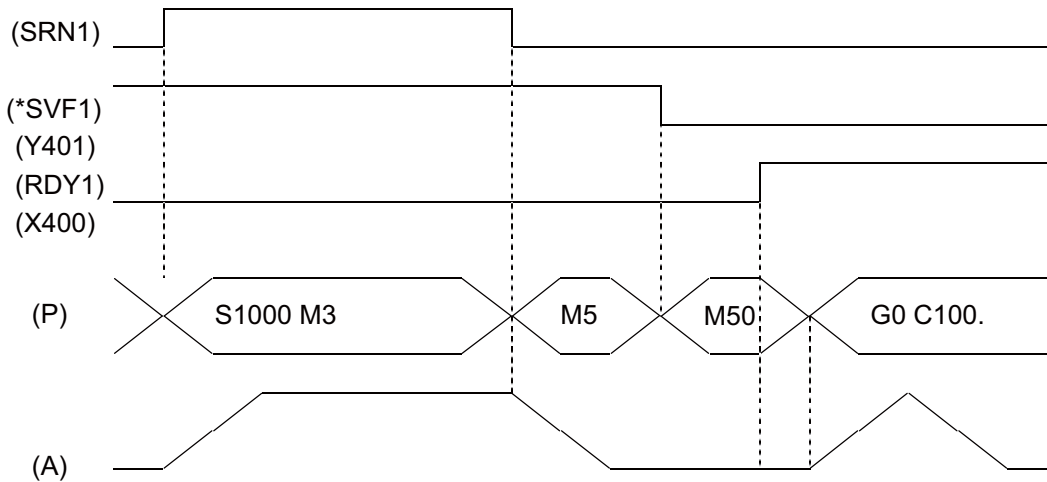
At servo ON C axis (spindle control not possible)

C axis position data

The NC's internal C axis position data is updated even for the spindle rotation during spindle control.

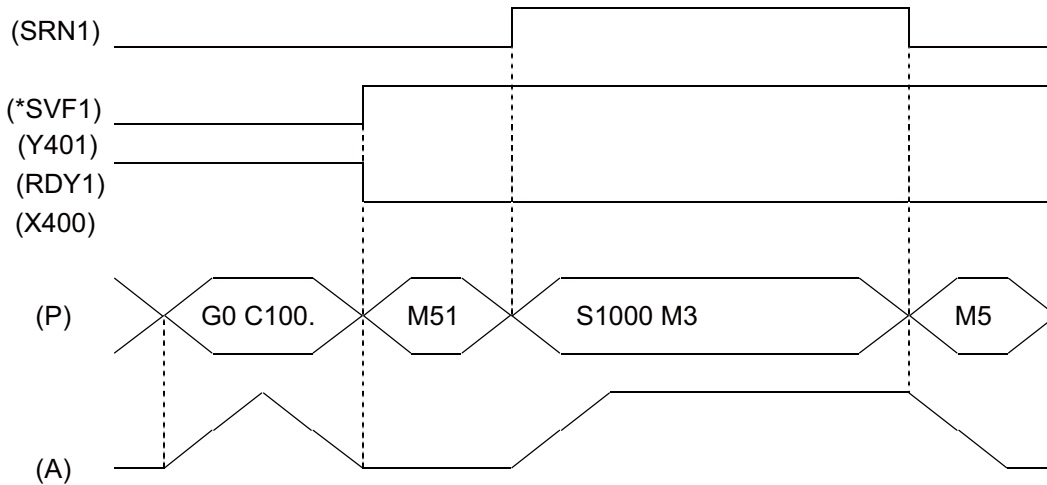
The C axis coordinate value counter is held during spindle control, and is updated according to the amount moved during spindle control when the C axis servo READY is turned ON. (The C axis position at servo ON may differ from the position just before the previous servo OFF.)

Changeover timing chart example



Switch from spindle to C axis

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



Switch from C axis to spindle

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

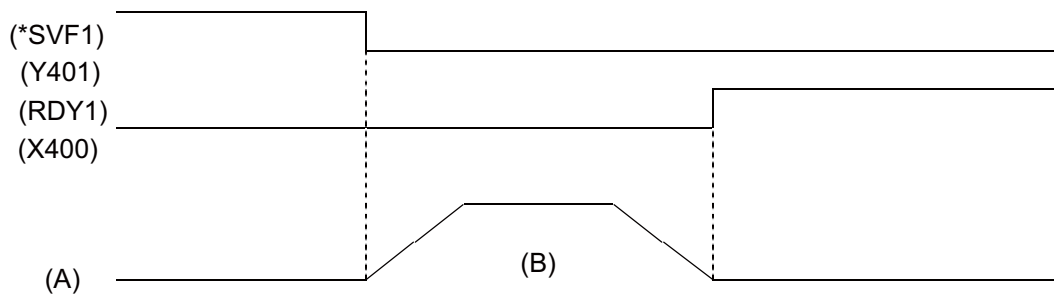
(Note) M codes in the above figures indicate;
 - M3 : Spindle forward run
 - M5 : Spindle stop
 - M50: C axis servo ON
 - M51: C axis servo OFF
 They formulate sequence programs.

The operation of zero point return

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

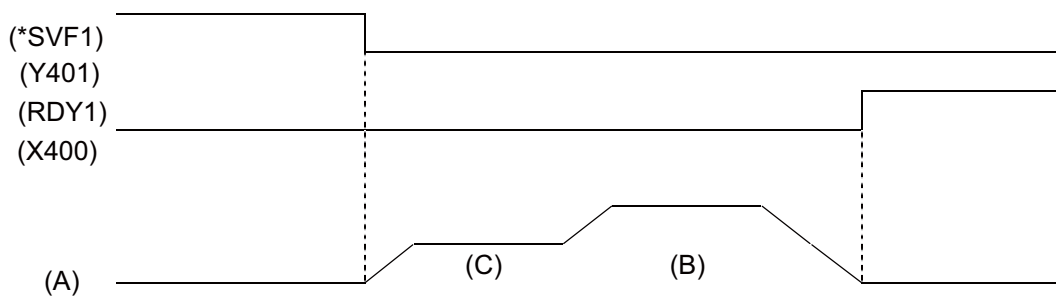
Zero point return type

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



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

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



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

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

Deceleration stop type

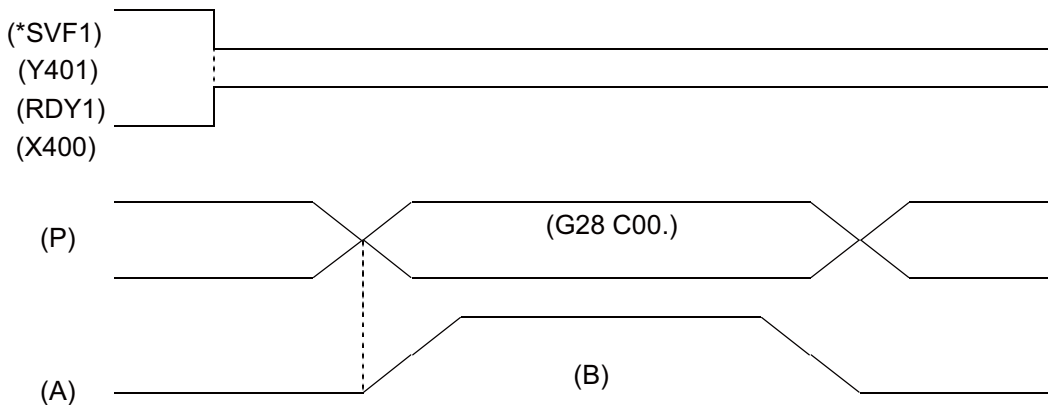
In case of the deceleration stop type ("#3106 zrn_typ/bit8" is "1"), C axis servo ON will only switch the spindle to the C axis and will not establish the zero point.

A coordinate must be established to carry out an automatic operation. Use the base specification parameter "#1226 aux10/bit3" to select whether to insert an automatic zero point return operation before issuing a movement command.

Manual operation is valid even when the zero point is not established.

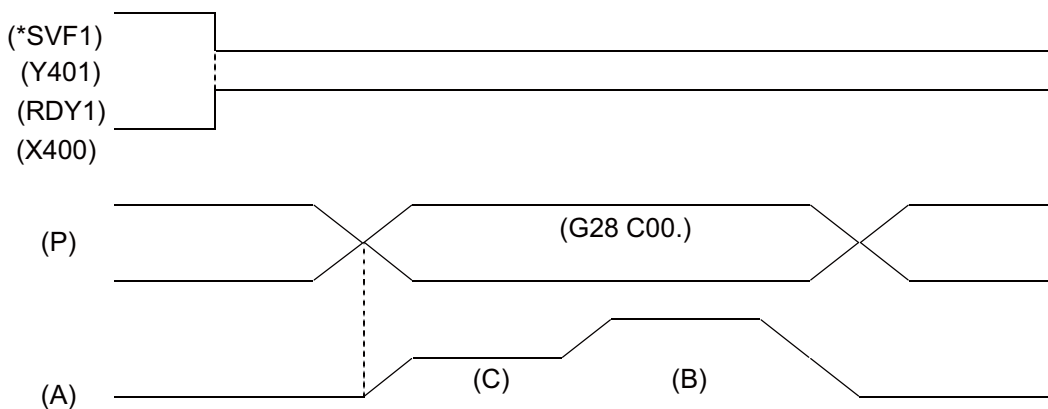
(1) When not inserting a zero point return

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



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

- (*SVF1) Servo OFF (B contact)
- (Y401) Servo ready
- (RDY1) Servo ready
- (X400) Servo ready
- (P) Program command
- (A) Spindle position shift amount
- (B) Zero point return



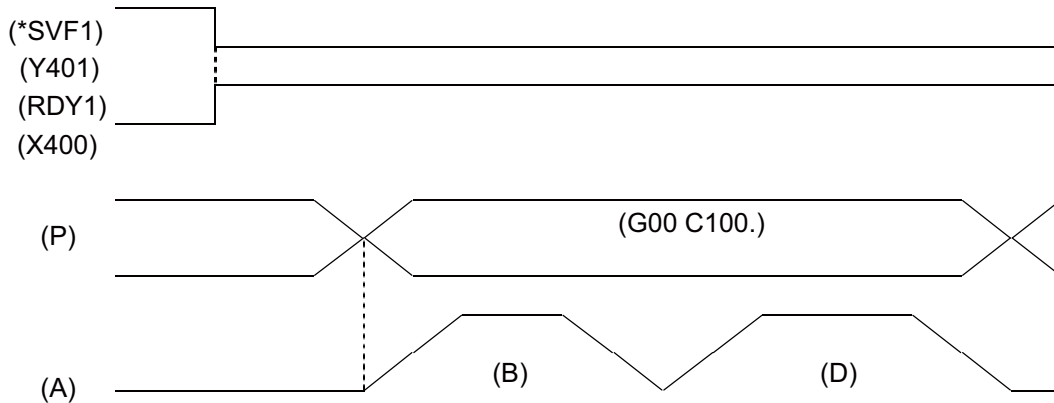
The operation of zero point return of deceleration stop type without inserting a zero point return (when Z-phase is not detected)

- (*SVF1) Servo OFF (B contact)
- (Y401) Servo ready
- (RDY1) Servo ready
- (X400) Servo ready
- (P) Program command
- (A) Spindle position shift amount
- (B) Zero point return
- (C) Z-phase detection

(2) When inserting a zero point return automatically

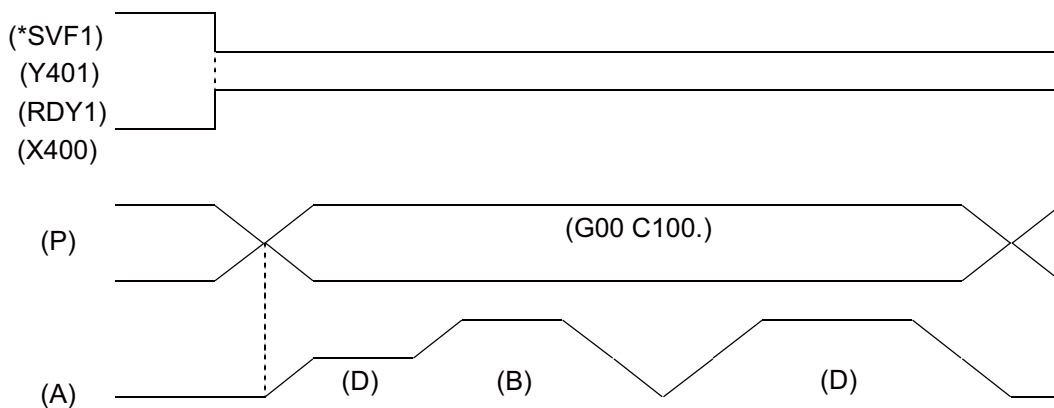
When "#1226 aux10/bit3" is "1", the zero point return is executed automatically before moving if the movement command is commanded without establishing the zero point.

However, the zero point return is not inserted if the movement command is commanded while the zero point is established.



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

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



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

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

The operation when there is a discrepancy between units

When the setting unit for the part system to use the spindle and C-axis "#1003 iunit" differs from the spindle unit "#3035 spunit", the error "Y51 Spindle/C axis unit illegal 0202" will appear and the interlock state will be applied.

However, be aware that the zero point return by the servo ON will be executed even if the error "Y51 Spindle/C axis unit illegal 0202" appears when the spindle/C axis is the zero point return type.

When the unit is not set (blank is displayed), it will be handled as the standard setting value "B".

10.5 Spindle Synchronization



Function and purpose

In a machine having two or more spindles, this function controls the rotation speed and phase of one spindle (basic spindle) in synchronization with the rotation of the other spindle (synchronous spindle).

The function is used when the rotation speed of the two spindles must be matched, for example, if a workpiece grasped by the first spindle is to be grasped by a second spindle, or if the spindle rotation speed has to be changed when one workpiece is grasped by both the first and second spindles.

There are two types of spindle synchronization: Spindle synchronization I and Spindle synchronization II.

The spindle synchronization control I

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

The spindle synchronization function II

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

Common setting for the spindle synchronization control I and II

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

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

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

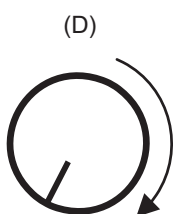
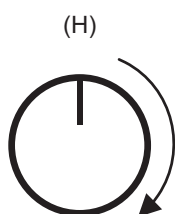
10.5.1 Spindle Synchronization Control I ; G114.1



Function and purpose

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

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

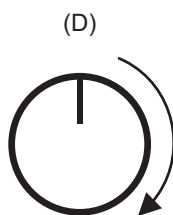
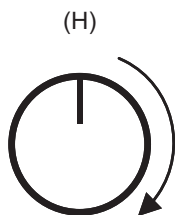


(H) Basic spindle
(D) Synchronous spindle

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

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

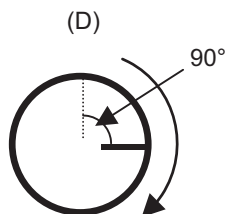
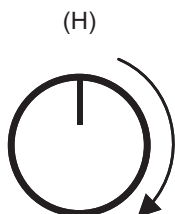
(Example 1) Phase synchronization with a phase error of "0"



(H) Basic spindle
(D) Synchronous spindle

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

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



(H) Basic spindle
(D) Synchronous spindle

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

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



Command format

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

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

G113 ; ... Spindle synchronization control cancel

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

Spindle synchronization cancel (G113) cancels the synchronous state of the two spindles rotating in synchronization with the spindle synchronization command.

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



Detailed description

Rotation speed and rotation direction

- (1) The rotation speed and rotation direction of the basic spindle and synchronous spindle during spindle synchronization control are the rotation speed and rotation direction commanded for the basic spindle. Note that the rotation direction of the synchronous spindle can be reversed from the basic spindle through the program.
- (2) The basic spindle's rotation speed and rotation direction can be changed during spindle synchronization control.
- (3) The synchronous spindle's rotation command is also valid during spindle synchronization control. When spindle synchronization control is commanded, if neither a forward run command nor reverse run command is commanded for the synchronous spindle, the synchronization standby state will be entered without starting the synchronous spindle's rotation. If the forward run command or reverse run command is input in this state, the synchronous spindle will start rotation. The synchronous spindle's rotation direction will follow the direction commanded in the program.
If spindle stop is commanded for the synchronous spindle during spindle synchronization control (when both the forward run and reverse run commands are turned OFF), the synchronous spindle rotation will stop.
- (4) The rotation speed command (S command) and constant surface speed control are invalid for the synchronous spindle during spindle synchronization control. Note that the modal is updated, so these will be validated when the spindle synchronization is canceled.
- (5) The constant surface speed can be controlled by issuing a command to the basic spindle even during spindle synchronization control.

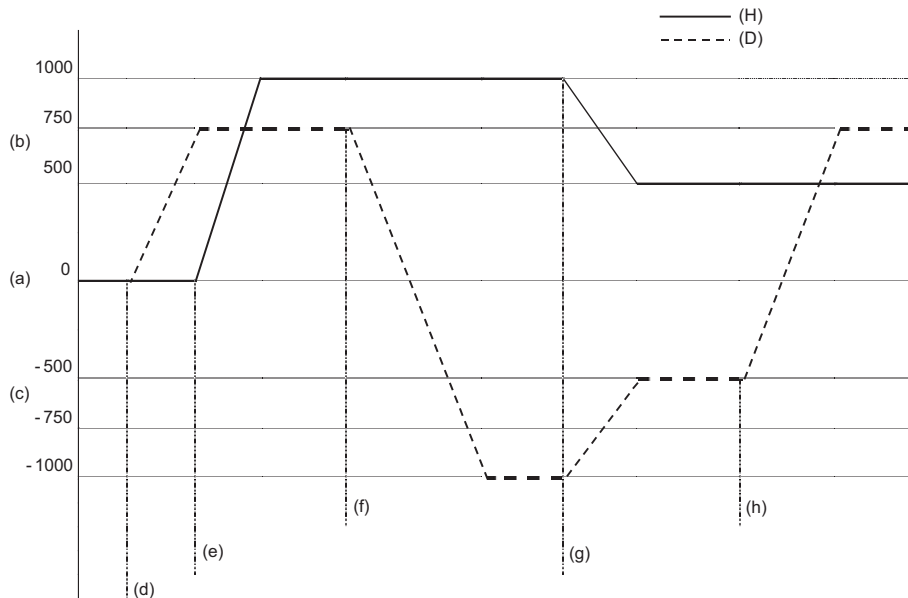
Rotation synchronization

- (1) When rotation synchronization control (command with no R address) is commanded with the G114.1 command, the synchronous spindle rotating at an arbitrary rotation speed will accelerate or decelerate to the rotation speed commanded beforehand for the basic spindle, and will enter the rotation synchronization state.
- (2) If the basic spindle's commanded rotation speed is changed during the rotation synchronization state, acceleration/deceleration will be carried out while maintaining the synchronization state following the spindle acceleration/deceleration time constants set in the parameters, and the commanded rotation speed will be achieved.
- (3) In the rotation synchronization state, the basic spindle can be controlled to the constant surface speed even when two spindles are grasping one workpiece.
- (4) Operation will take place in the following manner.

```

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

<Operation>



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

Phase synchronization

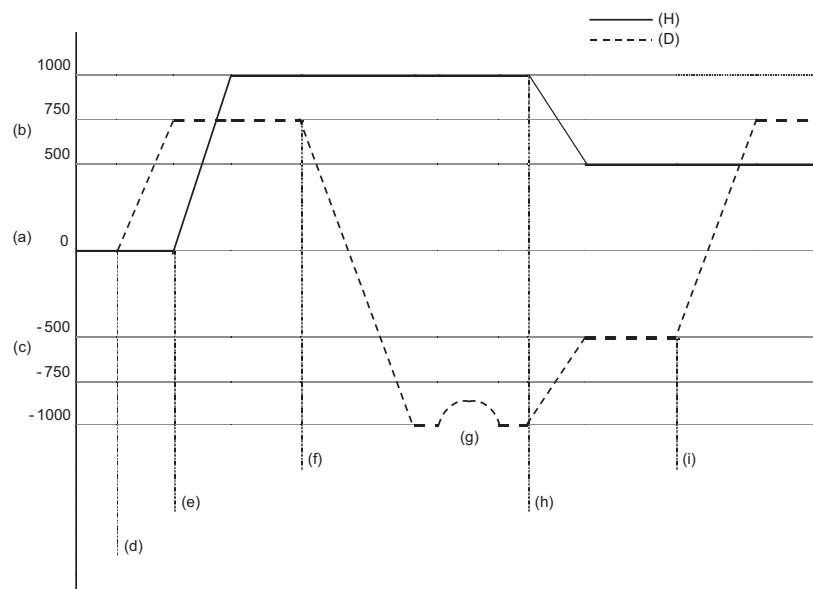
- (1) When phase synchronization (command with R address) is commanded with the G114.1 command, the synchronous spindle rotating at an arbitrary rotation speed will accelerate or decelerate to the rotation speed commanded beforehand for the basic spindle, and will enter the rotation synchronization state. Then, the phase is aligned so that the rotation phase commanded with the R address is reached, and the phase synchronization state is entered.
- (2) If the basic spindle's commanded rotation speed is changed during the phase synchronization state, acceleration/deceleration will be carried out while maintaining the synchronization state following the spindle acceleration/deceleration time constants set in the parameters, and the commanded rotation speed will be achieved.
- (3) In the phase synchronization state, the basic spindle can be controlled to the constant surface speed even when two spindles are grasping one workpiece.
- (4) Operation will take place in the following manner.

```

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

```

<Operation>



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

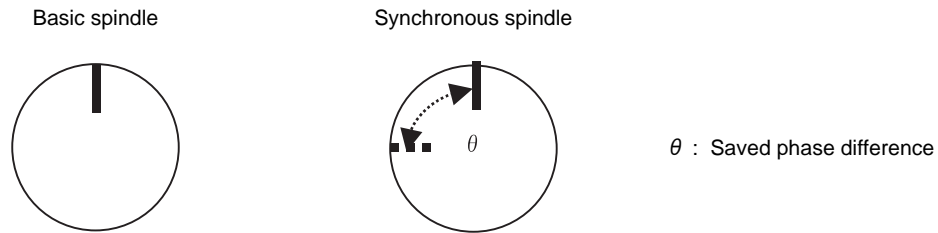
(Note 1) When "#3130 syn_spec/bit1" = "0", the phase synchronization is conducted by the step synchronization method without acceleration/deceleration. And when "#3130 syn_spec/bit1" = "1", it is conducted by the multi-step acceleration/deceleration method (mentioned later).

Spindle synchronization phase shift amount calculation function

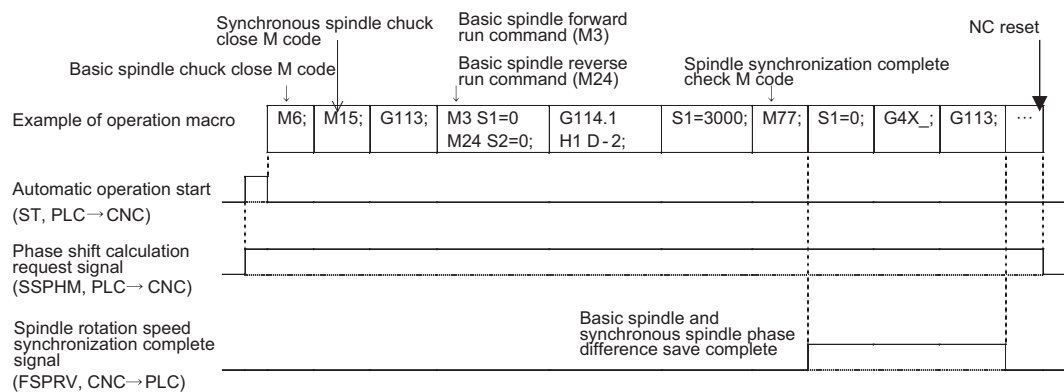
The spindle phase shift amount calculation function obtains and saves the phase difference of the basic spindle and synchronous spindle by turning the PLC signal ON when the phase synchronization command is executed. When the phase is positioned to the automatically saved phase difference before executing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

[Saving the basic spindle and synchronous spindle phase difference]

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

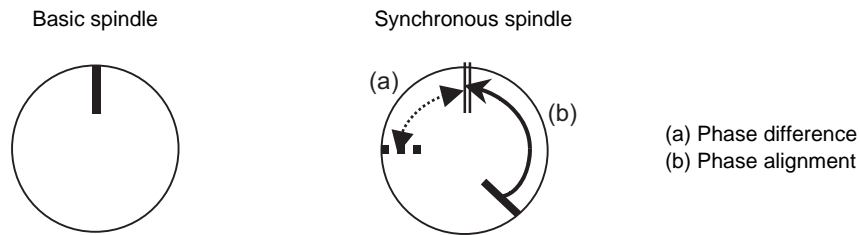


<Example of operation>

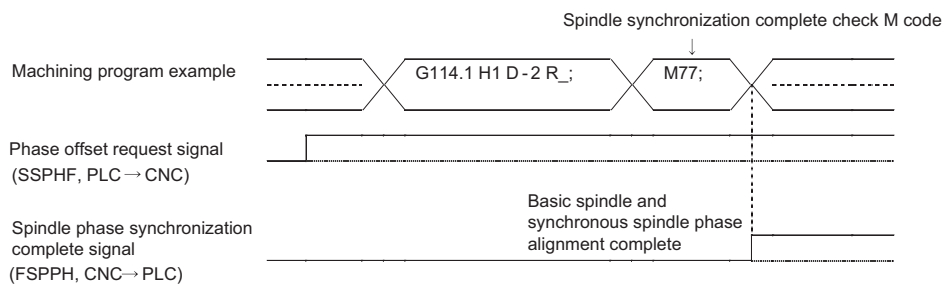


[Automatic phase alignment of basic spindle and synchronous spindle]

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



<Example of operation>



Multi-step acceleration/deceleration

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

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

Time required to rotate to sptc1 set rotation speed from stopped state (a)
= spt (or A command when G114.1 is commanded) * sptc1/slimit

Time required to reach sptc2 set rotation speed from sptc1 (b)
= spt (or A command when G114.1 is commanded) * spdiv1 * (sptc2 - sptc1)/slimit

Time required to reach sptc3 set rotation speed from sptc2 (c)
= spt (or A command when G114.1 is commanded) * spdiv2 * (sptc3 - sptc2)/slimit

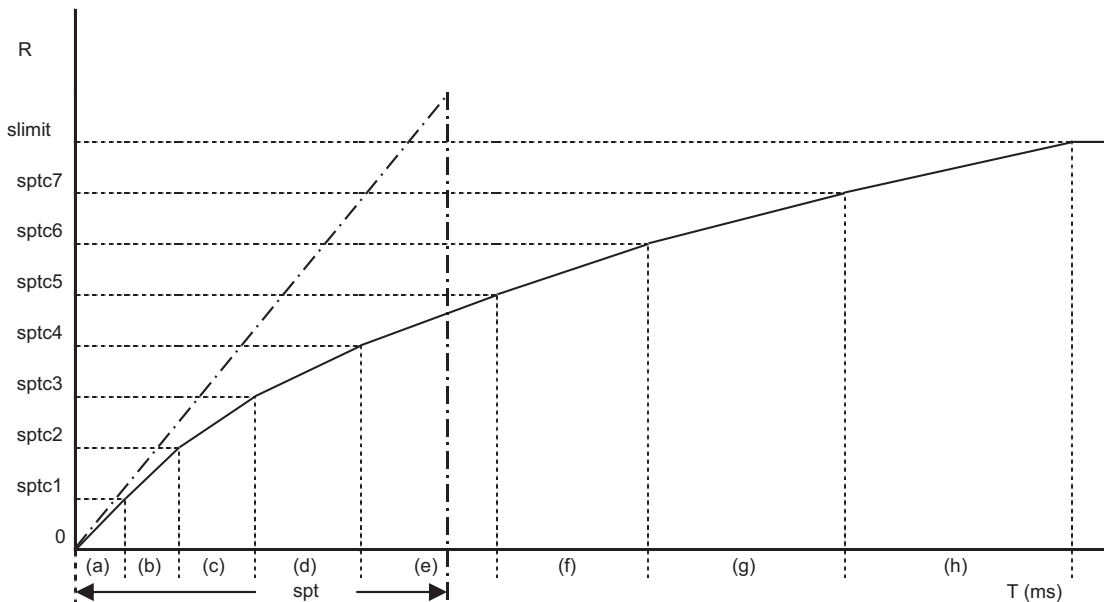
Time required to reach sptc4 set rotation speed from sptc3 (d)
= spt (or A command when G114.1 is commanded) * spdiv3 * (sptc4 - sptc3)/slimit

Time required to reach sptc5 set rotation speed from sptc4 (e)
= spt (or A command when G114.1 is commanded) *
spdiv4 * (sptc5 - sptc4)/slimit

Time required to reach sptc6 set rotation speed from sptc5 (f)
= spt (or A command when G114.1 is commanded) *
spdiv5 * (sptc6 - sptc5)/slimit

Time required to reach sptc7 set rotation speed from sptc6 (g)
= spt (or A command when G114.1 is commanded) *
spdiv6 * (sptc7 - sptc6)/slimit

Time required to reach sptc8 set rotation speed from sptc7 (h)
= spt (or A command when G114.1 is commanded) *
spdiv7 * (slimit - sptc7)/slimit



R: Rotation speed T: Time

To decrease the number of acceleration/deceleration steps during spindle synchronization, set one of the following for the unnecessary step.

Magnification for time constant changeover speed (spdiv7 to spdiv1) = 0 (or 1)

Spindle synchronous multi-step acceleration/deceleration changeover speed (sptc7 to sptc1) = Limit rotation speed (slimit) or higher



Precautions

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



- (18) When the spindle synchronization commands are being issued with the PLC I/F method (#1300 ext36/bit7 OFF), a program error (P610) will occur if the spindle synchronization control is commanded with G114.1/G113.
- (19) Chuck close must always be set. If not, machine may suffer an excessive load or an alarm may occur.

Cautions on programming

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

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

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

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

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



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

10.5.2 Spindle Synchronization Control II



Function and purpose

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



Detailed description

Basic spindle and synchronous spindle selection

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

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

Starting spindle synchronization

The spindle synchronization control mode is entered by inputting the spindle synchronization signal (SPSY). The synchronous spindle will be controlled in synchronization with the rotation speed commanded for the basic spindle during the spindle synchronization control mode.

When the difference of the basic spindle and synchronous spindle rotation speeds reaches the spindle synchronization rotation speed reach level setting value (#3050 sprlv), the spindle rotation speed synchronization complete signal (FSPRV) will be output.

The synchronous spindle's rotation direction is designated with the spindle synchronization rotation direction designation as the same as the basic spindle or the reverse direction.

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

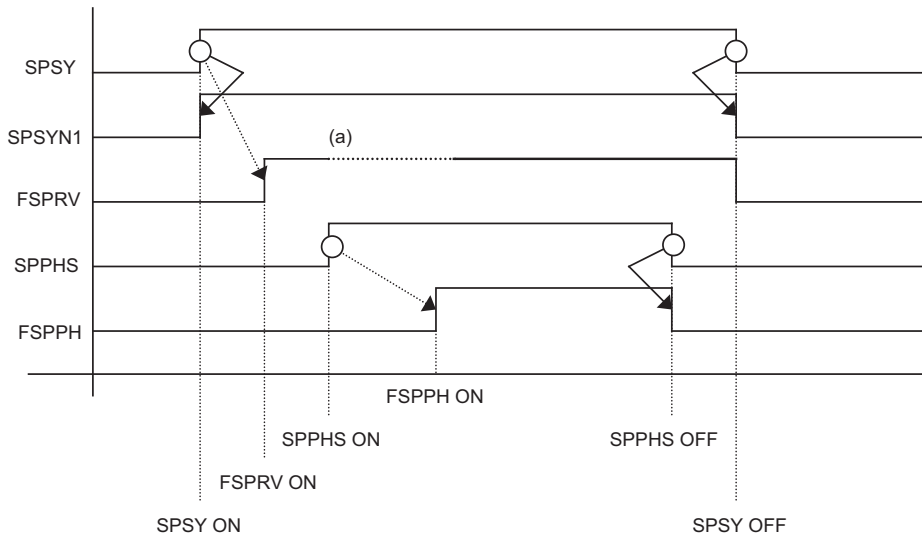
Spindle phase alignment

Spindle phase synchronization starts when the spindle phase synchronization control signal (SPPHS) is input during the spindle synchronization control mode.

The spindle phase synchronization complete signal is output when the spindle synchronization phase reach level setting value (#3051 spplv) is reached.

The synchronous spindle's phase shift amount can also be designated from the PLC.

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



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

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

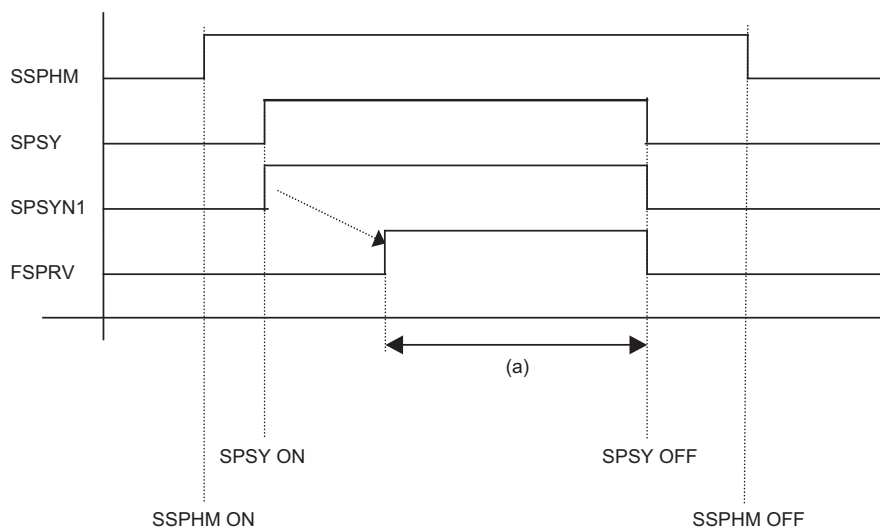
Calculating the spindle synchronization phase shift amount and requesting phase offset

The spindle phase shift amount calculation function obtains and saves the phase difference of the basic spindle and synchronous spindle by turning the "phase shift calculation request" signal ON during spindle synchronization. When calculating the spindle phase shift, the synchronous spindle can be rotated with the handle, so the relation of the phases between the spindles can also be adjusted visually.

If the spindle phase synchronization control signal is input while the phase offset request signal (SSPHF) is ON, the phases will be aligned using the position shifted by the saved phase shift amount as a reference.

This makes aligning of the phases easier when grasping the material that the shape of one end differs from the other end.

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



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

SSPHM : Phase shift calculation request

SPSY : Spindle synchronization

SPSYN1 : In spindle synchronization signal

FSPRV : Spindle rotation speed synchronization completion

(Note 1) The phases cannot be aligned while calculating the phase shift.

(Note 2) The synchronous spindle cannot be rotated with the handle when the manual operation mode is set to the handle mode.



Precautions and restrictions

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

10.5.3 Precautions for Using Spindle Synchronization Control



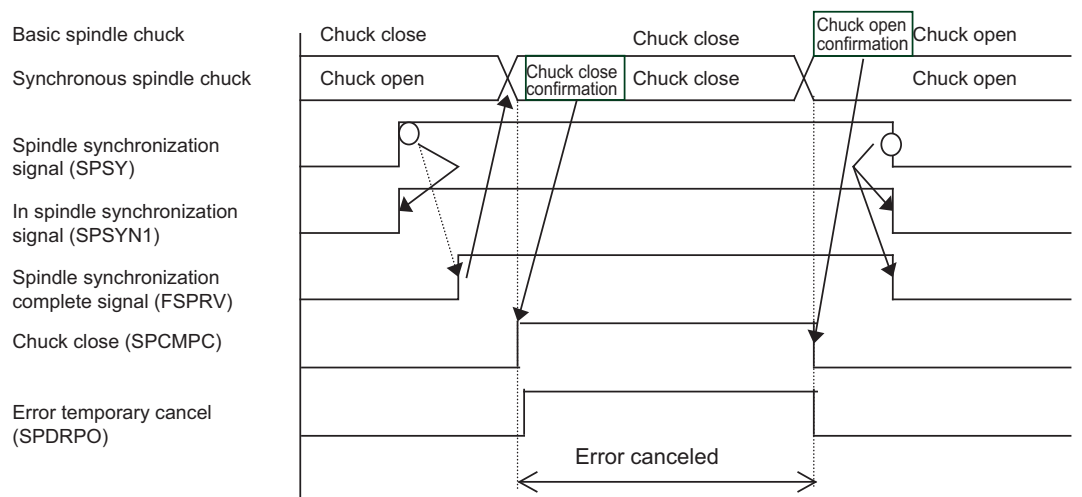
Precautions

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

Chuck close signal

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

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



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

Error temporary cancel function

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

This torsion can be prevented by temporarily canceling this error.

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

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

(Note 2) Turn this signal ON after the both chucks of basic spindle side and synchronous spindle side are closed to grasp the workpiece.

Turn this signal OFF if even one chuck is opened.

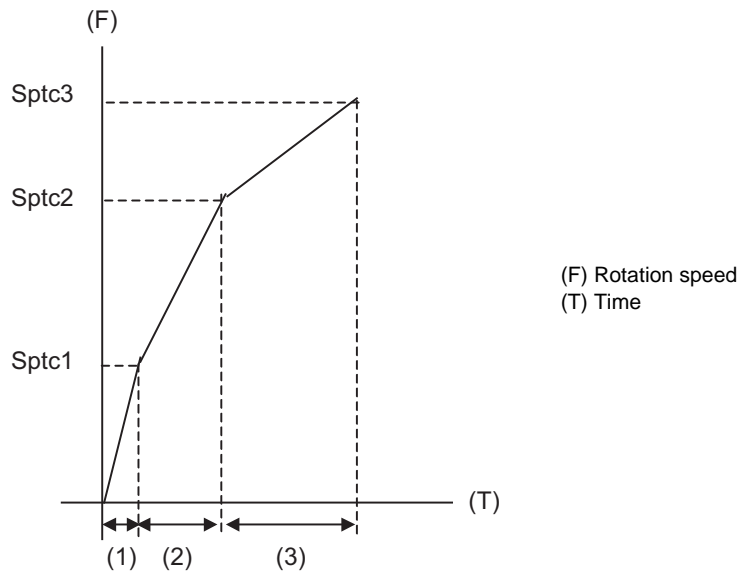
Phase error monitor

The phase error can be monitored during spindle phase synchronization.

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

Multi-step acceleration/deceleration

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



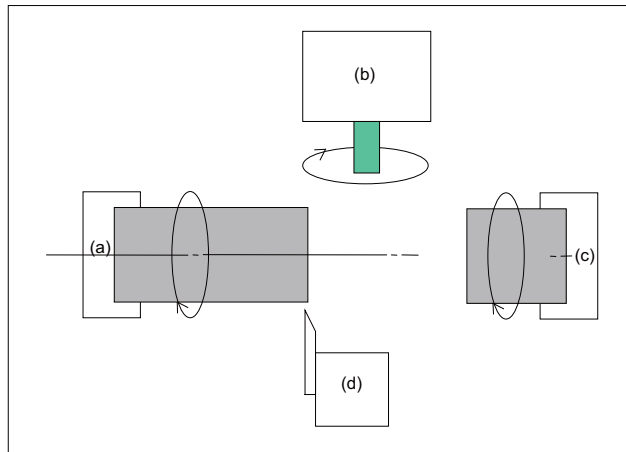
- (1) Time required from stopped state to sptc1 setting rotation speed
 $spt * (sptc1/\text{maximum rotation speed})$
- (2) Time required from sptc1 to sptc2 setting rotation speed
 $spt * ((sptc2-sptc1)/\text{maximum rotation speed}) * spdiv1$
- (3) Time required from sptc2 to sptc3 setting rotation speed
 $spt * ((sptc3-sptc2)/\text{maximum rotation speed}) * spdiv2$

10.6 Multiple-spindle Control



Function and purpose

Multiple spindle control is a function used to control the sub-spindle in a machine tool that has a main spindle (1st spindle) and a sub-spindle (2nd spindle to 4th spindle).



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

10.6.1 Multiple-spindle Control I (spindle control command) ; S ○ =



Function and purpose

Spindle rotation command for up to 7 spindles is provided.

Although the S***** command is normally used to designate the spindle rotation speed, the Sn=***** command is also used for multiple spindle control.

S commands can be issued from the machining program of any part systems.

Number of usable spindles differ the machine model, confirm the specifications of the model used.



Command format

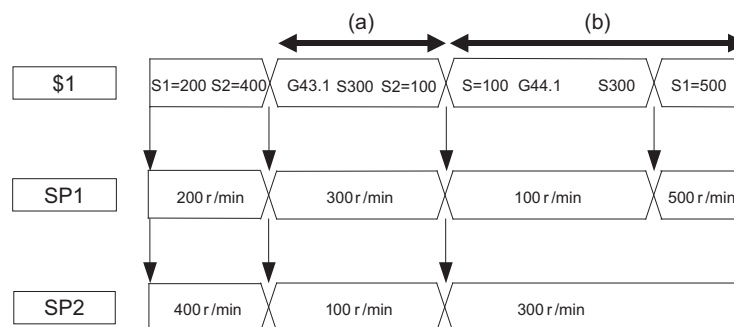
Sn=***** ; ... S6-digit binary data.

n	Designate the spindle number with one numeric character.
*****	Rotation speed or constant surface speed command value.



Detailed description

- (1) Each spindle command is delimited by the details of ○ .
(Example)
S1 = 3500 ; 1st spindle 3500(r/min) command
S2 = 1500 : 2nd spindle 1500(r/min) command
- (2) Multiple spindles can be commanded in one block.
- (3) If two or more commands are issued to the same spindle in a block, the command issued last will be valid.
(Example) S1 = 3500 S1 = 3600 S1 = 3700 ; S1 = 3700 will be valid.
- (4) The S***** command and S ○ =***** command can be used together.
The spindle targeted for the S***** command is delimited by the spindle selection command.
(Example) When G44.1 spindle No. is 2



\$1 : 1st part system execution program

SP1 : 1st spindle rotation speed

SP2 : 2nd spindle rotation speed

(a) 1st spindle control mode (b) 2nd spindle control mode

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

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



Function and purpose

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



Command format

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

G44.1 ; ... 2nd spindle control mode ON



Detailed description

- (1) G43.1 and G44.1 are modal G codes.
- (2) The spindle control mode entered when the power is turned ON or reset depends on the parameter setting.
 Designate the spindle No. to be selected in G43.1 modal with the parameter (basic specifications parameter "#1199 Sselect").
 This parameter is provided for every part system to set as follows.

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

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

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

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

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

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

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

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

(Note1) The constant surface speed control will be switched to the 2nd spindle by G44.1 command. Therefore, the 1st spindle retains its rotation speed as that of "G44.1 S200;" command. The 1st spindle rotation speed will be 3000 (r/min) when "S1=3000;" command is issued.

Tool Functions

11.1 Tool Functions (T8-digit BCD)



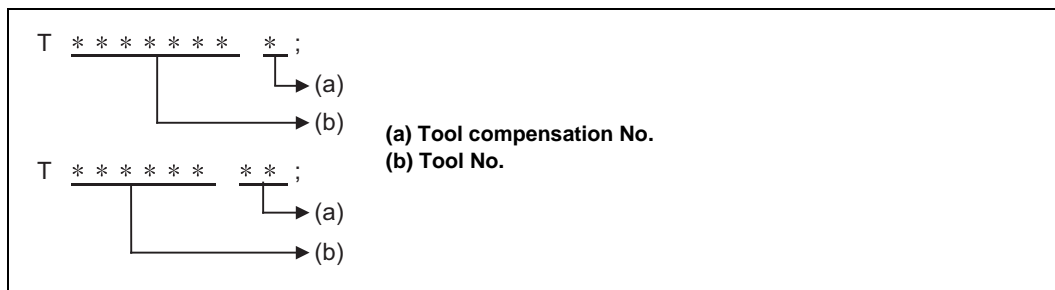
Function and purpose

The tool functions are also known as T functions and they assign the tool numbers and tool compensation number. The designations are made by the 8 digits (0 to 99999999) following the address T. These commands are used with the higher-order six or seven digits indicating the tool No., and the lower-order one or two digits indicating the compensation No.

Which is to be used is determined by the setting in the parameter (#1097 TLno.). The available T commands differ according to each machine, so refer to the instruction manual issued by the machine tool builder. One set of T commands can be issued in one block.



Command format



Refer to the instructions issued by the machine tool builder for the correspondence between the actual tools and the tool Nos. commanded in the program.

BCD codes and start signals are output.

If the T function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which sequence applies.

- (1) The T function is executed after the movement command.
- (2) The T function is executed simultaneously with the movement command.

Processing and completion sequences are required for all T commands.

Tool Compensation Functions

12.1 Tool Compensation



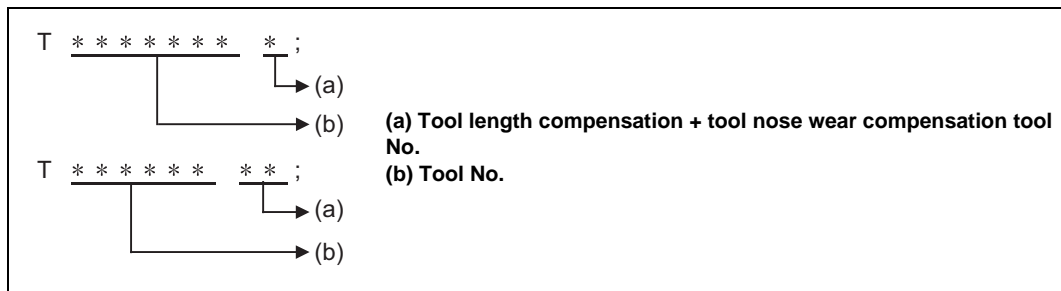
Function and purpose

Tool compensation is performed by the T functions which are commanded with the 3-, 4- or 8-digit number following address T. There are two types of tool compensation: tool length compensation and tool nose wear compensation. The command method can be selected with parameter "#1097 T1digit" and "#1098 TLno.". One set of T commands can be issued in one block.

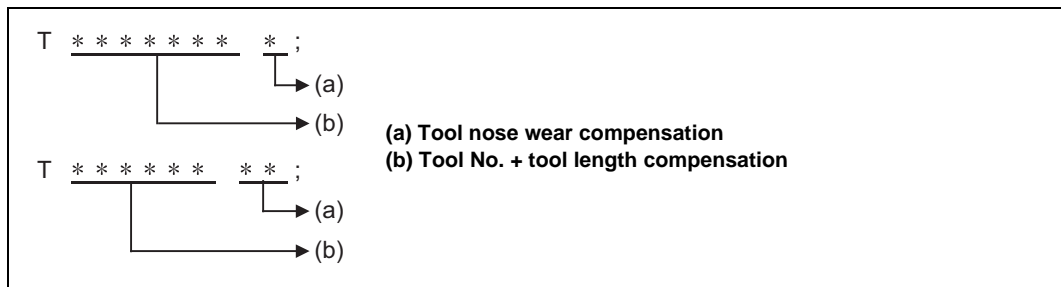


Command format

- (1) When designating the tool length and tool nose wear compensation No. using the last 1 or 2 digits of the T command.



- (2) When differentiating between the tool length compensation No. and tool nose wear compensation No.



The lower two digits of the tool No. are the tool length compensation No.

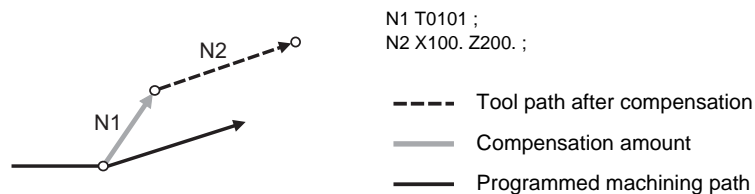
12.1.1 Tool Compensation Start



Detailed description

There are two ways to execute tool compensation and these can be selected by parameters: executing compensation when the T command is executed or executing compensation in the block with a movement command instead of performing compensation when the T command is executed.

(1) Compensation with T command execution



Tool length compensation and tool nose wear compensation are conducted simultaneously.

(Note 1) The movement for a compensation with the T command execution is rapid traverse in a G00 modal. The movement for other modals is cutting feed.

(Note 2) When performing compensation with T command execution, the path is compensated as a linear movement in a circular modal.

(Note 3) When performing compensation with T command execution, if the following G commands are issued in the same block as the T command, compensation will not be performed until other G commands are issued.

However, if an axis is specified by the command, compensation will be performed only to the specified axis.

G04 : Dwell

G10 : Programmable parameter/Compensation data input

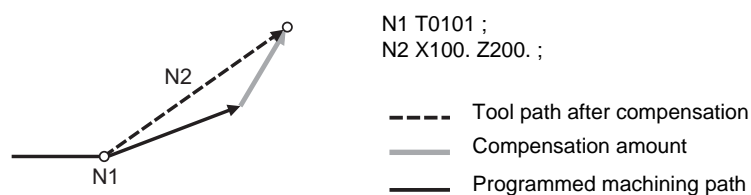
G11 : Parameter input by program cancel

G65 : User macro Simple call

G92 : Coordinate system setting

(Note 4) The following commands will temporarily cancel the compensation amount for the axis with any movement commanded: automatic reference position return (G28), 2nd, 3rd, and 4th reference position return (G30) or basic machine coordinate system selection (G53).

(2) Compensation with movement command



Tool length compensation and tool nose wear compensation are conducted simultaneously.

(Note 1) When performing compensation with a movement command, compensation is performed if the compensation amount is lower than the parameter "#1084 RadErr" when the first compensation is performed with a circular command. If the amount is higher, the program error (P70) will occur. (This also applies when the circular command and T command are in the same block for compensation with T command execution.)

12.2 Tool Length Compensation

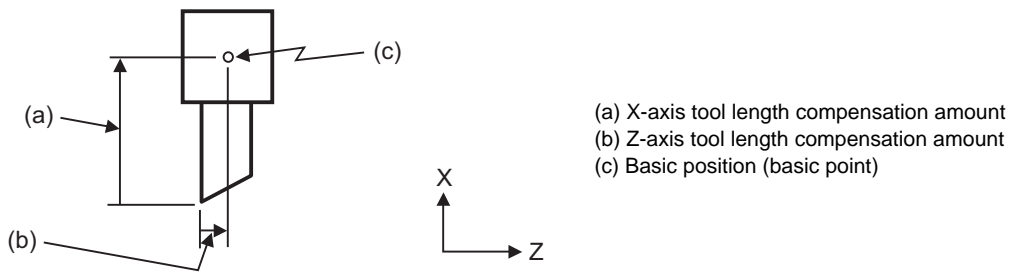


Detailed description

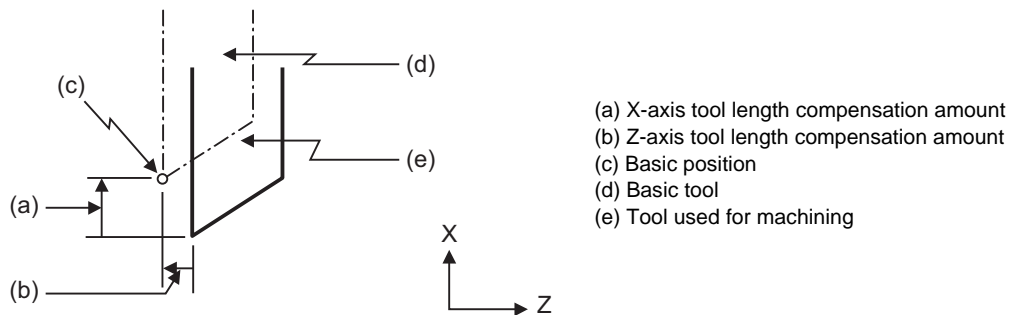
Tool length compensation amount setting

This function compensates the tool length with respect to the programmed basic position. This position may generally be set to either the center position of the turret or the tool nose position of the basic tool.

- (1) Center position of turret

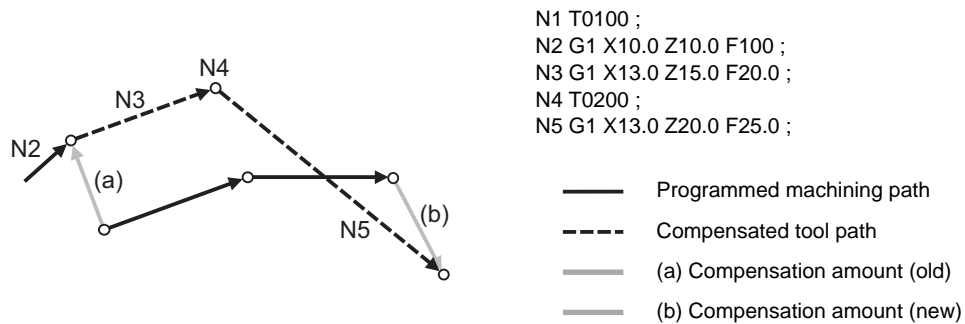


- (2) Tool nose position of basic tool



Tool length compensation No. change

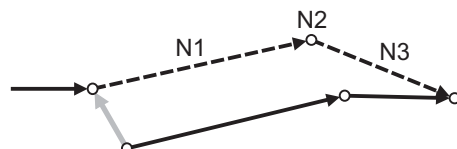
When tool Nos. are changed, the tool length compensation corresponding to the new tool Nos. are added to the movement amounts in the machining program.



In this example, the tool length is compensated with the tool No. and compensation is performed in the block with the movement command.

Tool length compensation cancel

- (1) When the compensation No. of 0 has been assigned
Compensation is canceled when 0 has been assigned as the tool length compensation No. by the T command.

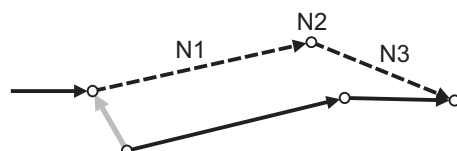


```
N1 X10.0 Z10.0 F10 ;
N2 T0000 ;
N3 G1 X10.0 Z20.0 ;
```

In this case, compensation is performed by the block with the movement command.

— Programmed machining path
- - - Compensated tool path
— Compensation amount

- (2) When the assigned compensation amount is 0
Compensation is canceled when the compensation amount in the tool length compensation No. assigned by the T command is 0.



```
N1 G1 X10.0 Z10.0 F10 ;
N2 T0100 ;
N3 G1 X10.0 Z20.0 ;
```

In this case, compensation is performed by the block with the movement command.

— Programmed machining path
- - - Compensated tool path
— Compensation amount



Precautions

- (1) When G28, G29 or G30 is commanded, the compensation is temporarily canceled. Therefore, the machine moves to the position where the compensation was canceled and the compensation amount is stored in the memory. This means that with the next movement command the machine will move to the compensation position.
- (2) When G28, G29 or G30 and the compensation cancel are commanded in the same block, the machine is moved to the position where the compensation was canceled, however the compensation amount remains stored in the memory. This means that the display coordinate may be displayed including the compensation amount. Issue these command in the separate block so that the compensation amount should not be stored in the memory.
- (3) Even if the compensation amount of the compensation No. currently selected by MDI is changed during automatic operation, the changed compensation amount will not be valid unless a T command with the same No. is executed again.
- (4) The tool length compensation and tool nose wear compensation amounts are cleared by resetting and by emergency stop. They can be retained by parameter "#1099 Treset".

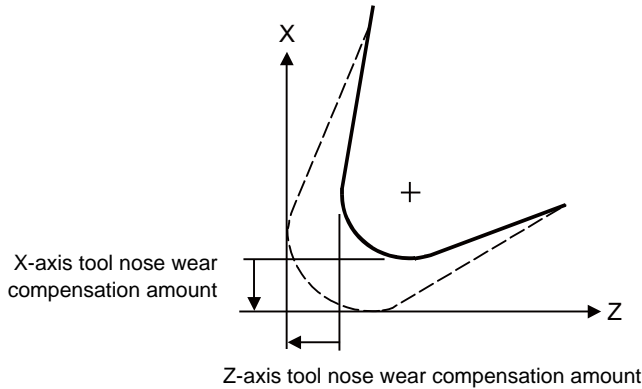
12.3 Tool Nose Wear Compensation



Detailed description

Tool nose wear compensation amount setting

The wear sustained by the tool being used can be compensated.

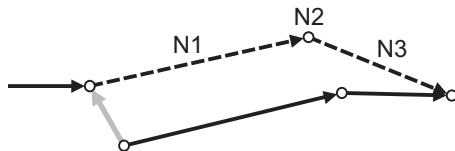


Tool nose wear compensation cancel

Tool nose wear compensation is canceled when 0 has been assigned as the compensation No.

```
N1 G1 X10.0 Z10.0 F10 ;
N2 T0100 ;
N3 G1 X10.0 Z20.0 ;
```

In this case, compensation is performed by the block with the movement command.



- Programmed machining path
- - - Compensated tool path
- Compensation amount



Precautions

- (1) When G28, G29 or G30 is commanded, the compensation is temporarily canceled. Therefore, the machine moves to the position where the compensation was canceled and the compensation amount is stored in the memory. This means that with the next movement command the machine will move to the compensation position.
- (2) When G28, G29 or G30 and the compensation cancel are commanded in the same block, the machine is moved to the position where the compensation was canceled, however the compensation amount remains stored in the memory. Issue these command in the separate block so that the compensation amount should not be stored in the memory.
- (3) Even if the compensation amount of the compensation No. currently selected by MDI is changed during automatic operation, the changed compensation amount will not be valid unless a T command with the same No. is executed again.
- (4) The tool length compensation and tool nose wear compensation amounts are cleared by resetting and by emergency stop. They can be retained by parameter "#1099 Treset".

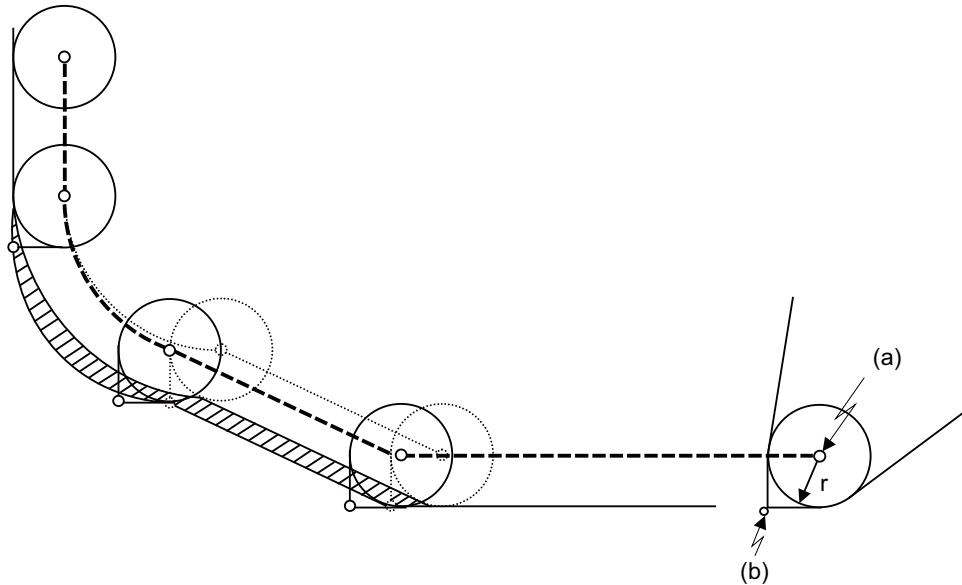
12.4 Tool Nose R Compensation ; G40,G41,G42,G46



Function and purpose

Because a tool nose is generally rounded, a hypothetical tool nose point is used for programming. Due to this roundness of the tool nose, there will be a gap between the programmed shape and the actual cutting shape during taper cutting or circular cutting. Tool nose radius compensation is a function for automatically calculating and offsetting this error by setting the tool nose radius value.

These command codes enable to choose the offset direction to be fixed or automatically identified.



(a) Tool nose center

(b) Hypothetical tool nose point (r) Tool nose R

.....

Tool nose center path with no tool nose radius compensation (Shaded part indicates the cutting shape gap)

Tool nose center path with tool nose radius compensation



Command format

G40(Xx/Uu Zz/Ww) ; ... Tool nose radius compensation cancel

G41(Xx/Uu Zz/Ww) ; ... Tool nose radius compensation left

G42(Xx/Uu Zz/Ww) ; ... Tool nose radius compensation right

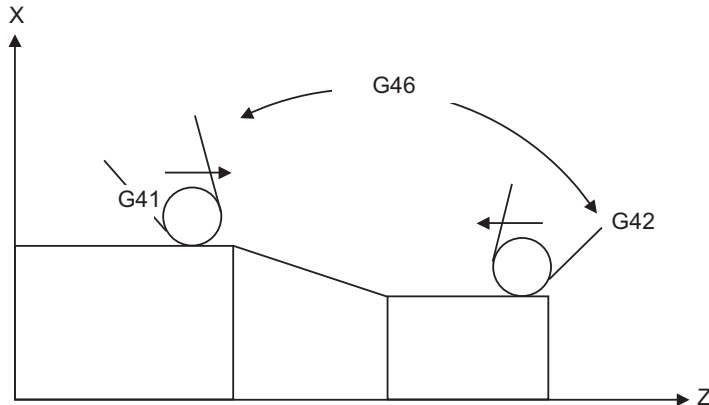
G46(Xx/Uu Zz/Ww) ; ... Tool nose radius compensation (automatic direction identification) ON

X_/U_	X axis end point coordinate (X is the absolute value of workpiece coordinate system, U is the incremental value from current position)
Z_/W_	Z axis end point coordinate (Z is the absolute value of workpiece coordinate system, W is the incremental value from current position)

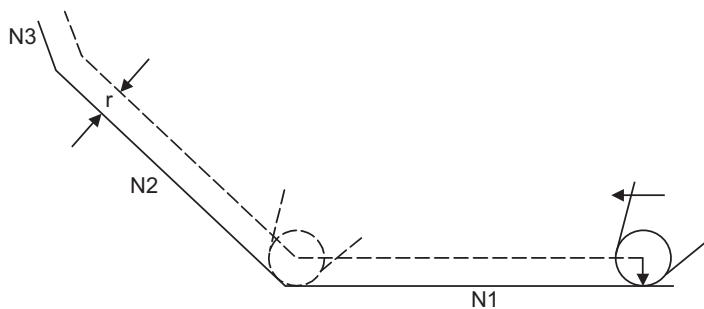


Detailed description

- (1) G41 works on condition that the tool is located on the left of the workpiece to the direction of motion. G42 works on condition that the tool is located on the right of the workpiece to the direction of motion. G46 automatically identifies the compensation direction by the preset hypothetical tool nose point and movement commands in the machining program. G40 cancels the tool nose radius compensation mode.



- (2) Tool nose radius compensation pre-reads the data in the following two movement command blocks (up to 5 blocks when there is no movement command) and controls the tool nose radius center path by the intersection point calculation method so that it is offset from the programmed path by an amount equivalent to the nose radius. In the figure below, "r" is the tool nose radius compensation amount (nose radius). The tool nose radius compensation amount corresponds to the tool length No. and should be preset along with the tool nose point.



- (3) If there are 4 or more blocks without movement amounts among 5 continuous blocks, overcutting or undercutting will occur. Blocks in which optional block skip is valid are ignored.
- (4) Tool nose radius compensation is valid also for fixed cycles (G77 to G79) and rough cutting cycles (G70, G71, G72, G73). However, in the rough cutting cycles, cutting will be done with the compensation mode canceled since the finished shape already includes the compensation amount and, upon completion of the cutting, operation will automatically return to the compensation mode.
- (5) Compensation mode will be temporarily canceled in 1 block before the thread cutting command block.
- (6) A tool nose radius compensation (G41 or G42) command can be issued during tool nose radius compensation (G46). For this, compensation mode does not need to be canceled with G40.
- (7) The compensation plane, movement axes and next advance direction vector follow the plane selection command designated by G17, G18 or G19.
 - G17 ... X-Y plane X, Y, I, J
 - G18 ... Z-X plane Z, X, K, I
 - G19 ... Y-Z plane Y, Z, J, K

12.4.1 Tool Nose Point and Compensation Directions



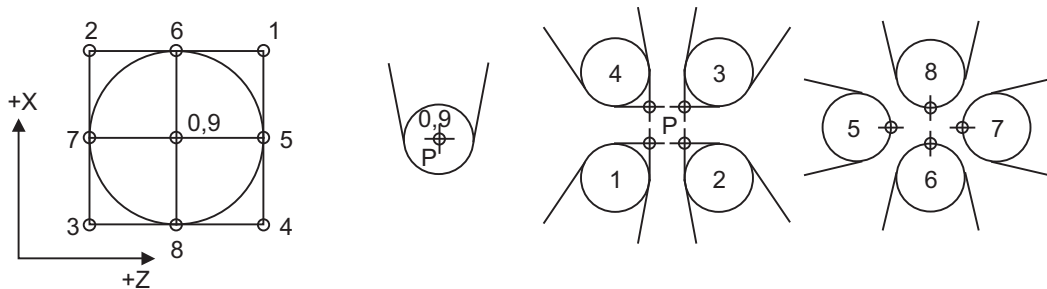
Detailed description

Tool nose point

Because a tool nose is generally rounded, the programmed tool nose position is adjusted to a point P shown in the examples figures below.

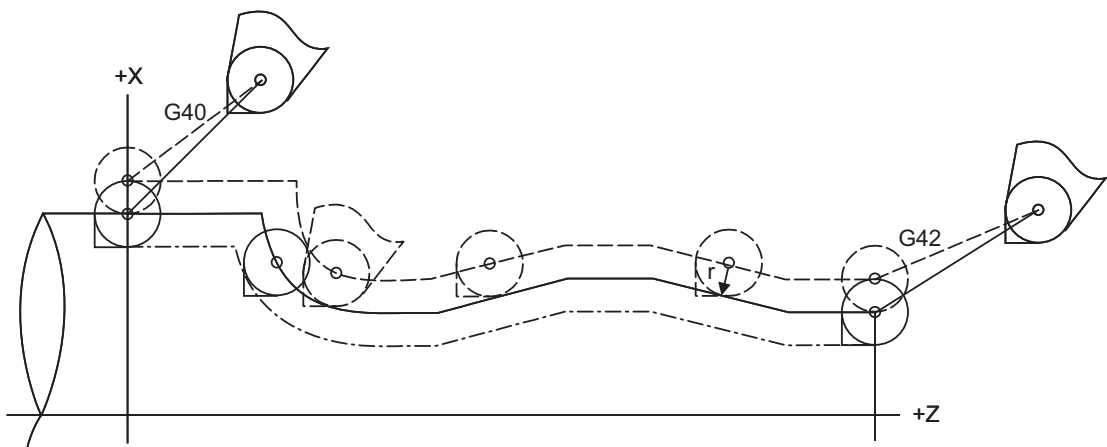
For tool nose radius compensation, select one of the points in the figures below for each tool length numbers and preset the positional relationship.

(Selects 1 to 8 in the G46 mode and 0 to 9 in the G41/G42 mode.)



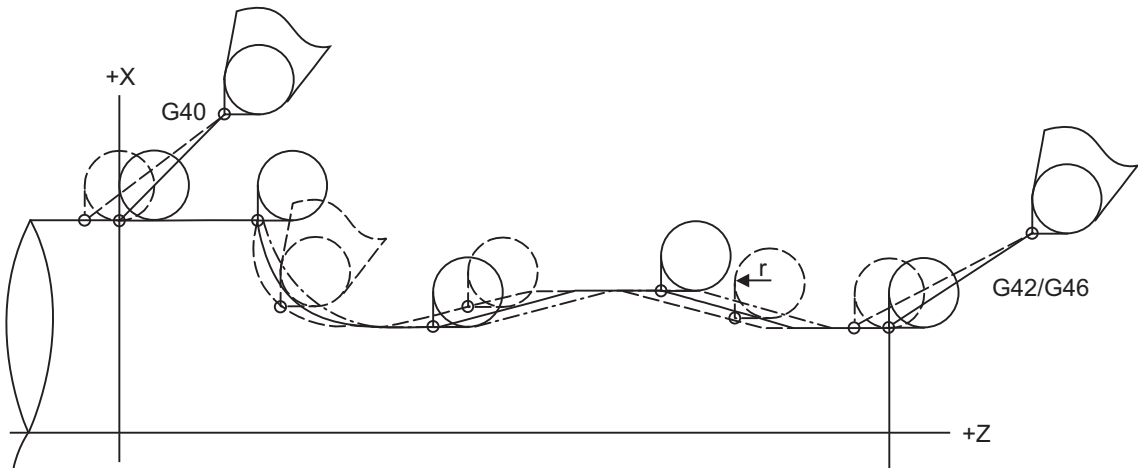
Tool nose point and compensation operation

- (1) When the nose R center is adjusted to the machining start position (Tool nose point 0 or 9)



- Programmed path or machining shape with tool nose radius compensation
- Nose R center path with tool nose radius compensation
- Machining shape with no tool nose radius compensation

(2) When the tool nose point is adjusted to the machining start position (Tool nose point 3)



- Programmed path or machining shape with tool nose radius compensation
- Nose R center path with tool nose radius compensation
- · - · - · - · - Machining shape with no tool nose radius compensation

Compensation direction of G46

The compensation directions of the G41/G42 commands is determined by the G41/G42 codes, while that of the G46 command is automatically determined in accordance with the following table from the relationship between the tool nose points and the commanded movement vectors.

- (1) If the initial movement vector (including G0) corresponds to a "x" mark in the table when starting the tool nose radius compensation, the compensation direction cannot be specified. It will be determined by the next movement vector. When the direction cannot be determined even after reading 5 blocks ahead, program error (P156) will occur.
- (2) When the compensation direction is reversed during tool nose radius compensation, program error (P157) will occur except when it is commanded in G00 block. Even if directions differ before and after the G28, G30 or G53 blocks, an error will not occur because compensation will be temporarily canceled. Using a parameter (#8106 G46 NO REV-ERR), the same compensation direction can be maintained.

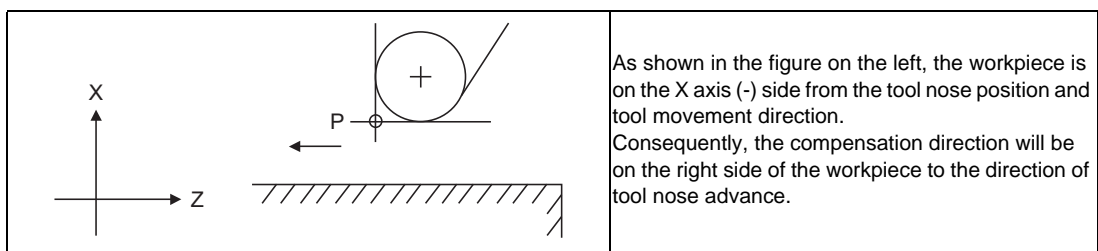
- (3) When the compensation direction during tool nose radius compensation corresponds to a "x" in the table below, the previous direction will be resumed.

[How to determine the compensation direction by the movement vectors and tool nose point in command G46]

		Compensation direction of tool nose									
		Tool nose points									
		①	②	③	④	⑤	⑥	⑦	⑧		
Direction of tool nose advance	Movement vector (tool nose points 1 to 4)		Right	Right	Left	Left	x	Right	x	Left	→
			x	Right	x	Left	Left	Right	Right	Left	
			Left	Right	Right	Left	Left	x	Right	x	↑
			Left	x	Right	x	Left	Left	Right	Right	
	Movement vector (tool nose points 5 to 8)		Left	Left	Right	Right	x	Left	x	Right	←
			x	Left	x	Right	Right	Left	Left	Right	
			Right	Left	Left	Right	Right	x	Left	x	↓
			Right	x	Left	x	Right	Right	Left	Left	

- (Note 1) "x" marks in the table indicate that the compensation direction cannot be determined by the movement vector (tool nose points).
- (Note 2) The " ↗ " mark denotes a movement vector in the 45° direction. (The other movement vectors are based on this.)
- (Note 3) The " ◊ " mark denotes a movement vector with a range larger than 45° and smaller than 135°. (The other movement vectors are based on this.)

(Example)When tool nose point 3, movement vector in the Z axis (-) direction (when movement vector is <-)



12.4.2 Tool Nose Radius Compensation Operations



Detailed description

Tool nose radius compensation cancel mode

The tool nose radius compensation cancel mode is established by any of the following conditions.

- (1) After the power has been turned on
- (2) After the reset button on the setting and display unit has been pressed
- (3) After the M02 or M30 command with reset function has been executed
- (4) After the tool radius compensation cancel command (G40) has been executed
- (5) After tool No. 0 has been selected (T00 has been executed)

The offset vectors are zero in the compensation cancel mode, and the tool nose point path coincides with the programmed path.

Programs including tool nose radius compensation must be terminated in the compensation cancel mode.

Tool nose radius compensation start (start-up)

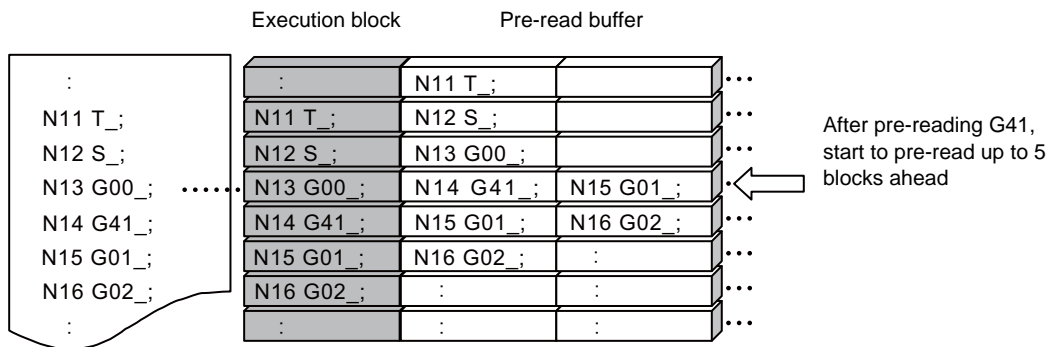
Tool nose radius compensation starts when all the following conditions are satisfied in the compensation cancel mode.

- (1) The movement command is issued after G41, G42 or G46.
- (2) The movement command is not a circular command.

Before starting a compensation, 2 to 5 blocks are pre-read for the intersection point calculation regardless of single block operation or continuous operation. (Two blocks are pre-read if there is a movement command, up to 5 blocks are pre-read if not)

Similarly, during compensation mode, up to five blocks are pre-read for compensation calculation.

[Control mode transition diagram]



There are two ways of starting the compensation operation: type A and type B.

The type to be used is selected by setting of parameter "#1229 set01/bit2".

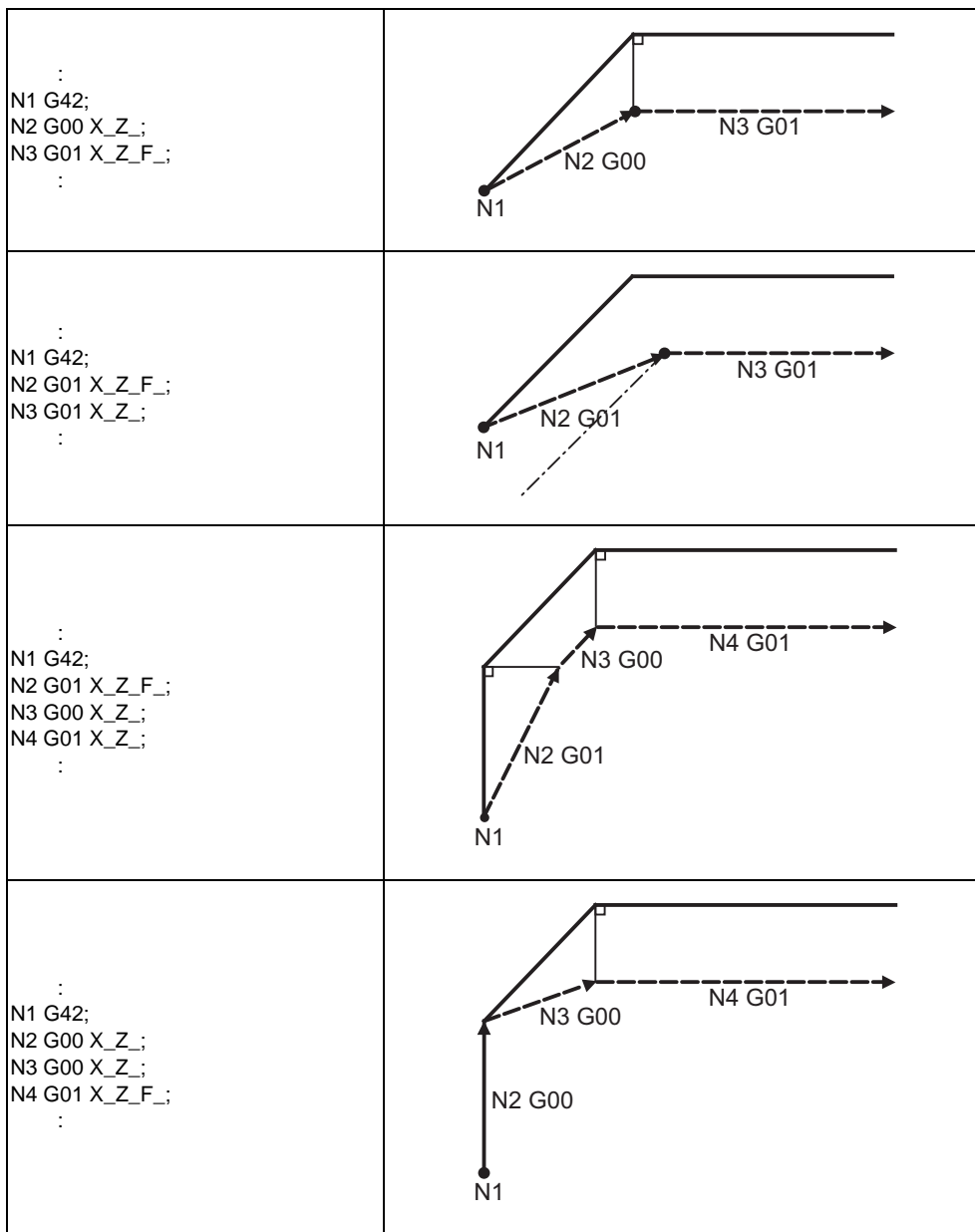
This type is used in common with the compensation cancel type.

#1229 set01/bit2 Nose R comp type B	Type	Explanation
0	Type A	When starting up/canceling a command block with tool nose radius compensation and radius compensation, type A will not conduct intersection operation processing to the block and, instead, convert it to an offset vector which is vertical to the command vector.
1	Type B	When starting up/canceling a command block with tool nose radius compensation and radius compensation, Type B will conduct intersection operation processing to the command block and the next block.

Start operation for tool nose radius compensation

When starting tool nose radius compensation, the tool will not move as much as the specified compensation amount by either G41, G42 or G46 command alone. Tool nose radius compensation cannot be applied to the G00 command. It can be applied from G01, G02 or G03 command. Note that even if there is an axis command, the tool nose radius compensation will not be applied unless there is movement.

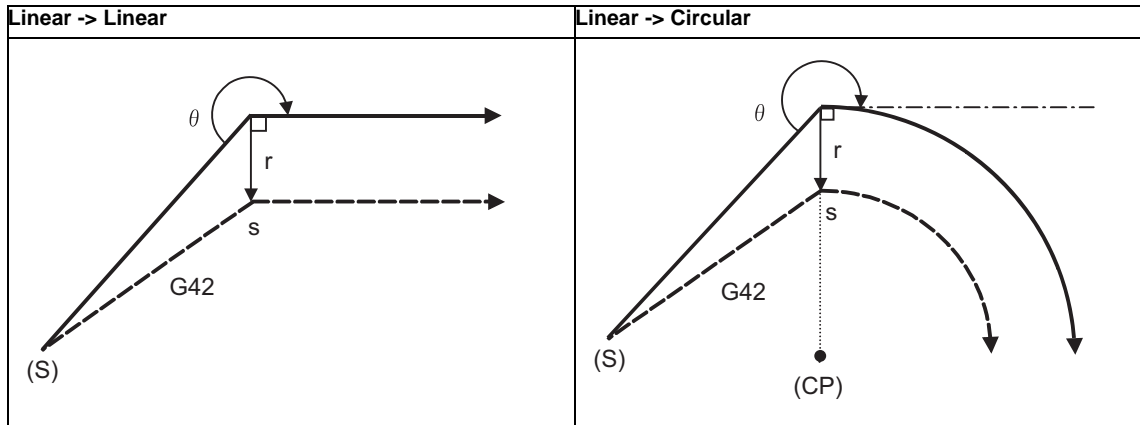
(1) When an independent G41/G42/G46 command is issued at an inside corner



— Program path

- - - - - Nose R center path

(2) When a G41/G42/G46 command is issued at an inside corner, in the same block as a movement command



(S) Start point (CP) Circular radius r : Nose R radius s : Stop point with single block

—— Program path

----- Nose R center path

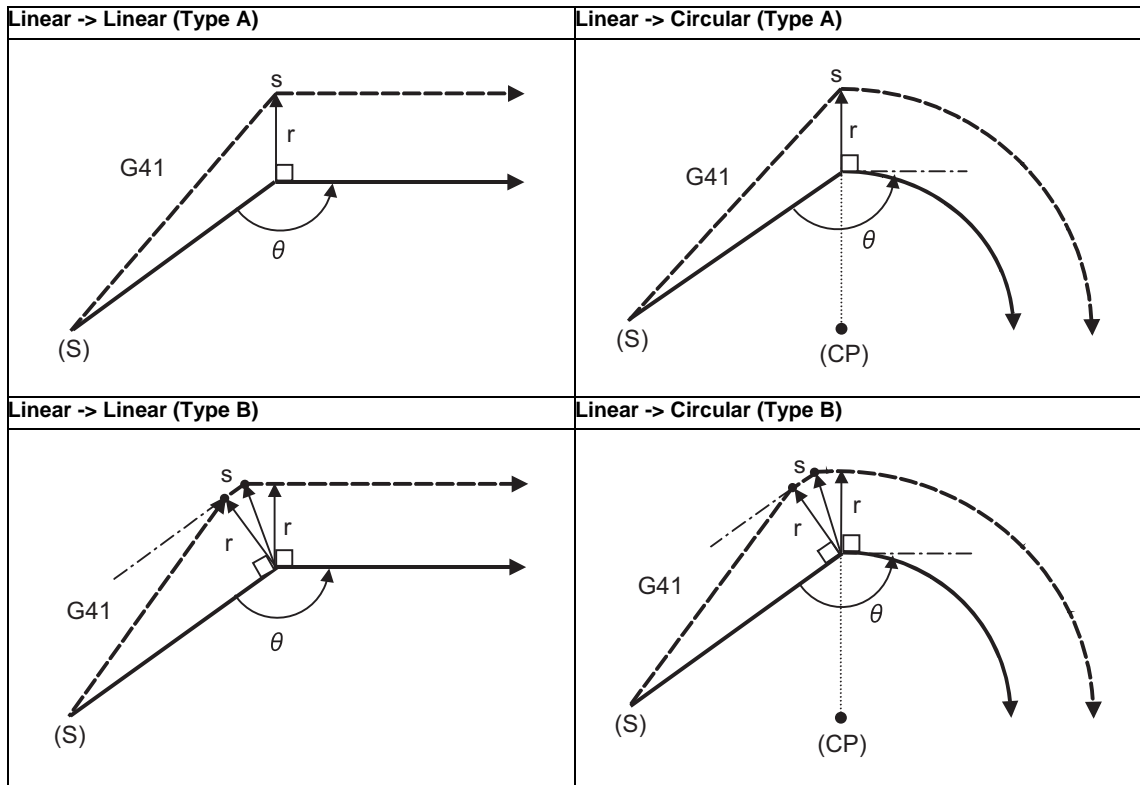
(3) When an independent G41/G42/G46 command is issued at an outside corner (obtuse angle)

	Type A	Type B
<pre> : N1 G41; N2 G00 X_Z_; N3 G01 X_Z_F_; : </pre>		
<pre> : N1 G41; N2 G01 X_Z_F_; N3 G01 X_Z_; : </pre>		
<pre> : N1 G41; N2 G01 X_Z_F_; N3 G00 X_Z_; N4 G01 X_Z_; : </pre>		
<pre> : N1 G41; N2 G00 X_Z_; N3 G00 X_Z_; N4 G01 X_Z_F_; : </pre>		

— Program path

- - - - - Nose R center path

(4) When a G41/G42/G46 command is issued at an outside corner (obtuse angle), in the same block as a movement command [$90^\circ \leq \theta < 180^\circ$]



(S) Start point (CP) Circular radius r : Nose R radius s : Stop point with single block

—— Program path

----- Nose R center path

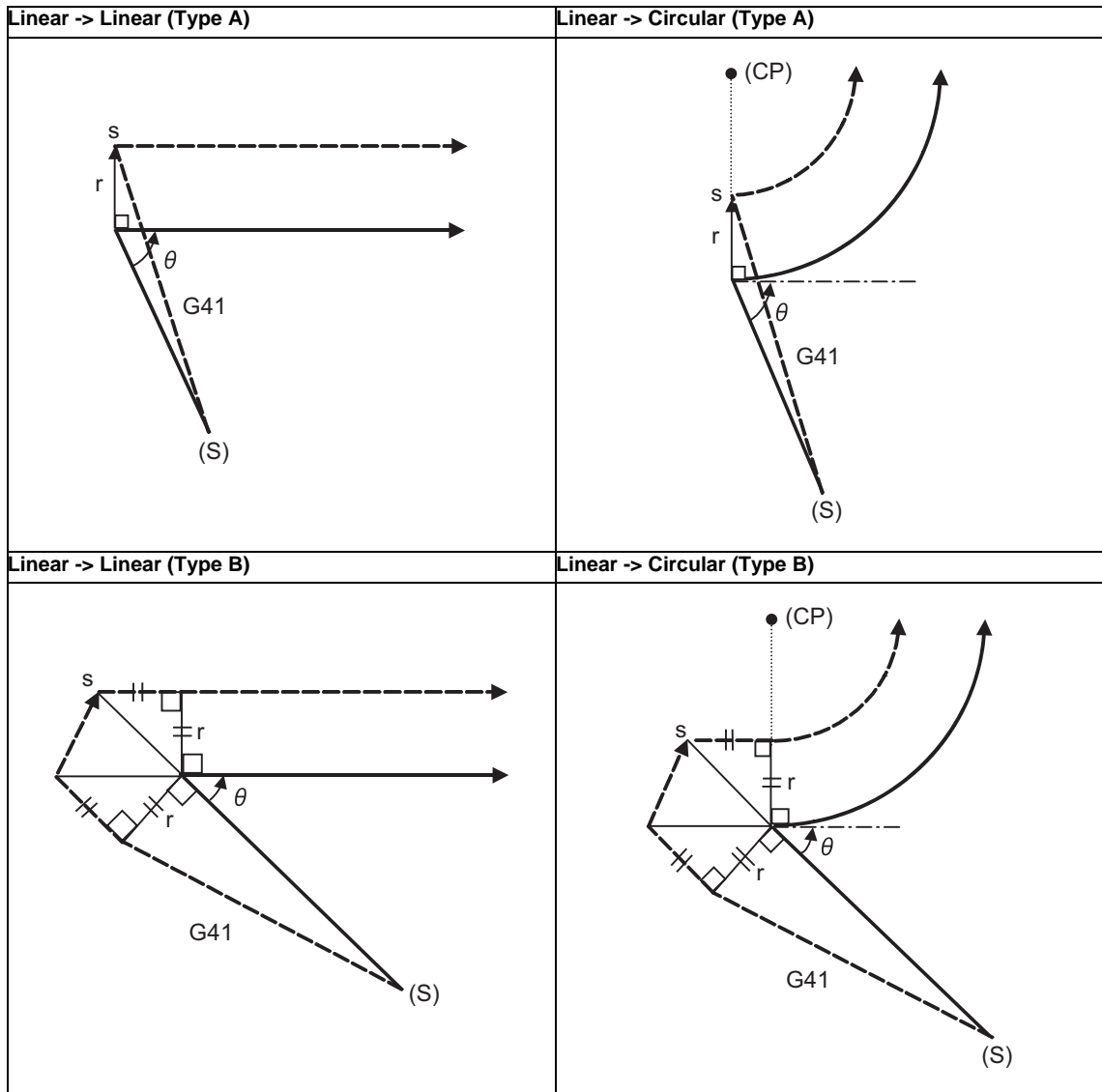
(5) When an independent G41/G42/G46 command is issued at an outside corner (acute angle)

	Type A	Type B
: N1 G41; N2 G00 X_Z_; N3 G01 X_Z_F_; :		
: N1 G41; N2 G01 X_Z_F_; N3 G01 X_Z_; :		
: N1 G41; N2 G01 X_Z_F_; N3 G00 X_Z_; N4 G01 X_Z_; :		
: N1 G41; N2 G00 X_Z_; N3 G00 X_Z_; N4 G01 X_Z_F_; :		

— Program path

- - - - - Nose R center path

(6) When a G41/G42/G46 command is issued at an outside corner (acute angle), in the same block as a movement command [$\theta < 90^\circ$]



(S) Start point (CP) Circular radius r : Nose R radius s : Stop point with single block

—— Program path - - - - Nose R 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

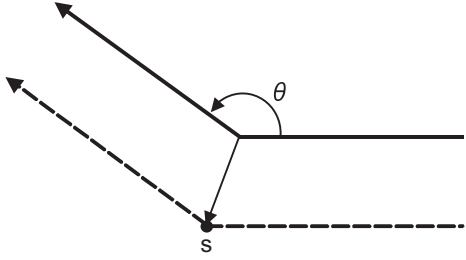
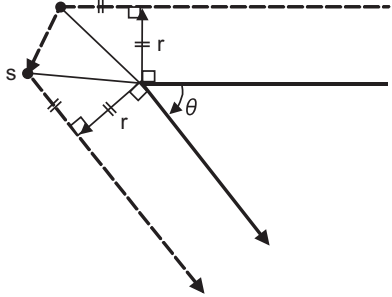
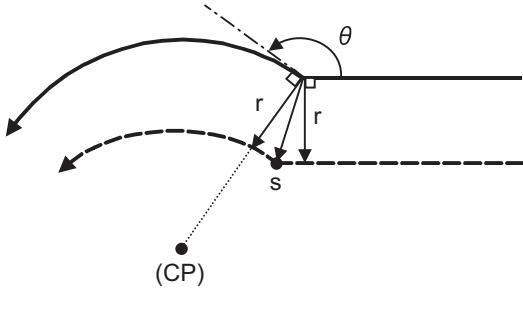
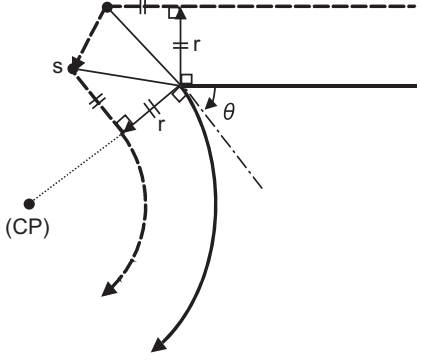
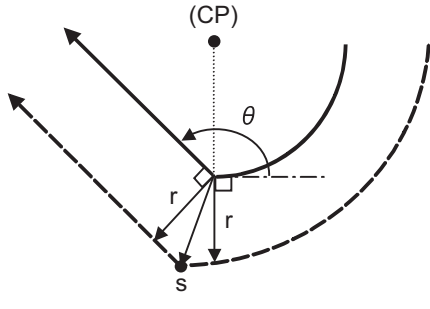
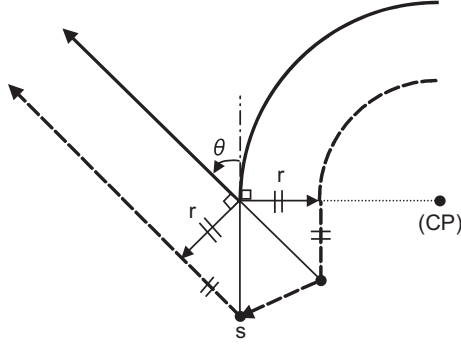
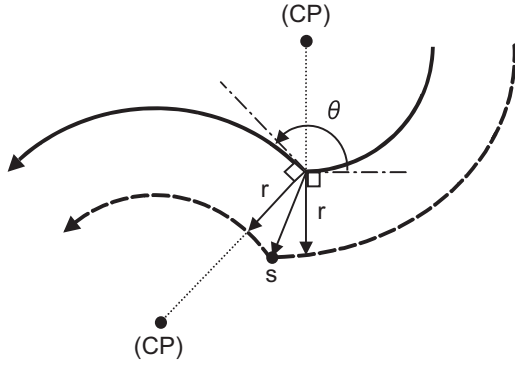
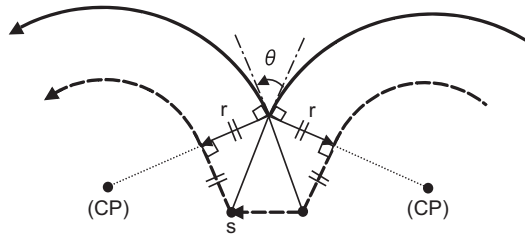
Calculate the tool center path from the straight line/circular arc to perform compensation to the program path (G00, G01, G02, G03).

Even if the same compensation command (G41, G42, G46) is issued in a tool nose radius compensation (G41, G42, G46) 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 is issued during tool nose radius compensation, pre-reading will be prohibited.

(1) Machining an outside corner

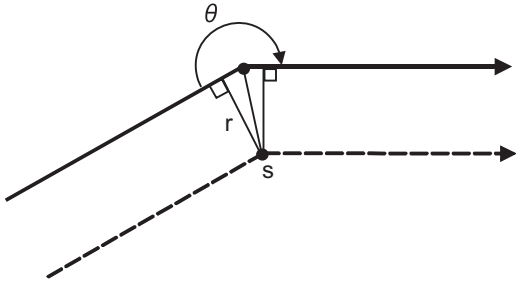
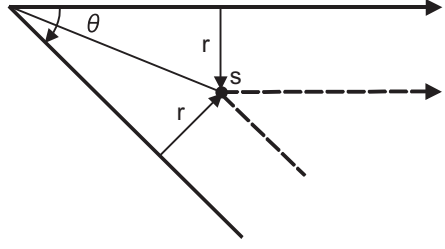
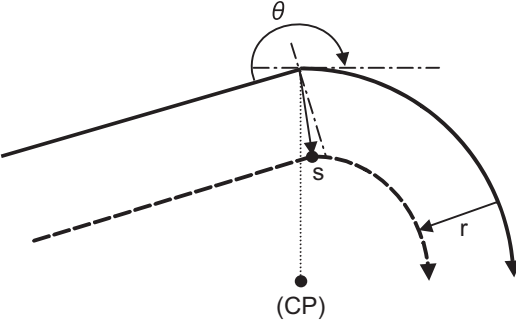
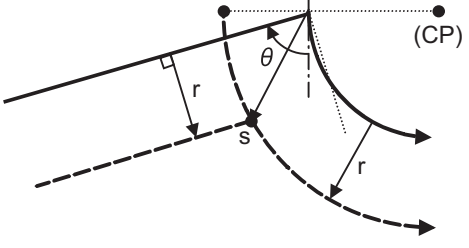
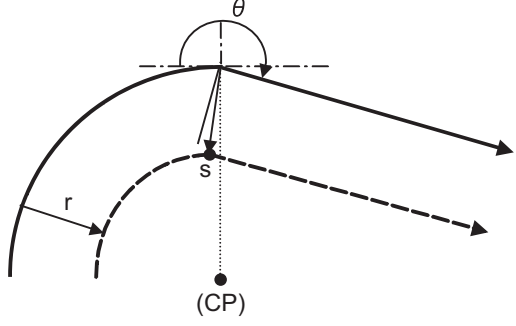
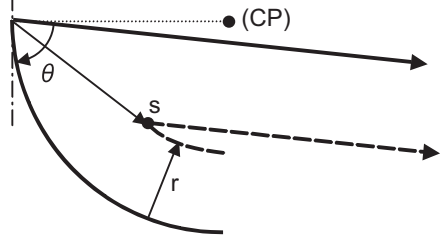
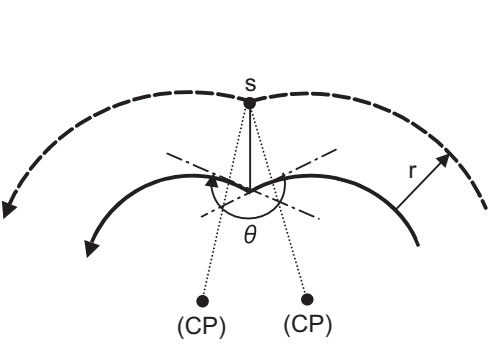
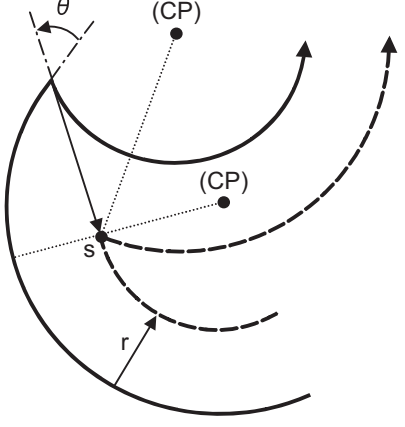
<p>Linear -> Linear ($90^\circ \leq \theta < 180^\circ$)</p> 	<p>Linear -> Linear ($0^\circ < \theta < 90^\circ$)</p> 
<p>Linear -> Linear ($90^\circ \leq \theta < 180^\circ$)</p> 	<p>Linear -> Linear ($0^\circ < \theta < 90^\circ$)</p> 
<p>Circular -> Linear ($90^\circ \leq \theta < 180^\circ$)</p> 	<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) Circular radius r : Nose R radius s : Stop point with single block

— Program path

---- Nose R center path

(2) Machining an inside corner

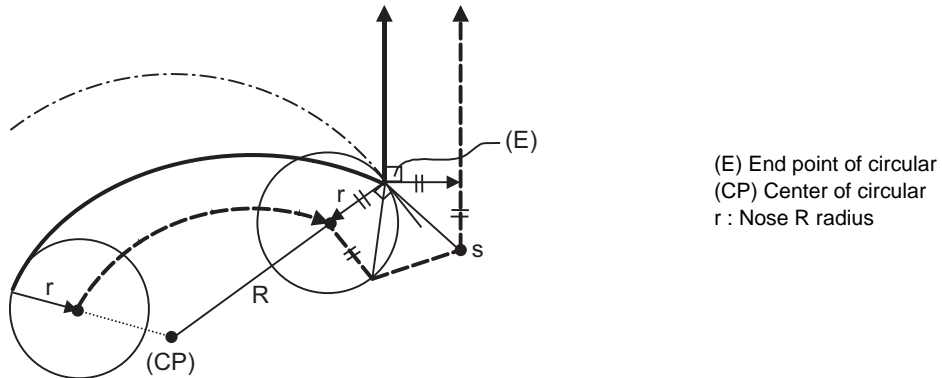
Linear -> Linear (Obtuse angle)	Linear -> Linear (Acute angle)
	
Linear -> Circular (Obtuse angle)	Linear -&to Circular (Acute angle)
	
Circular -> Linear (Obtuse angle)	Circular -> Linear (Acute angle)
	
Circular -> Circular (Obtuse angle)	Circular -> Circular (Acute angle)
	

(CP) Circular radius r : Nose R radius s : Stop point with single block

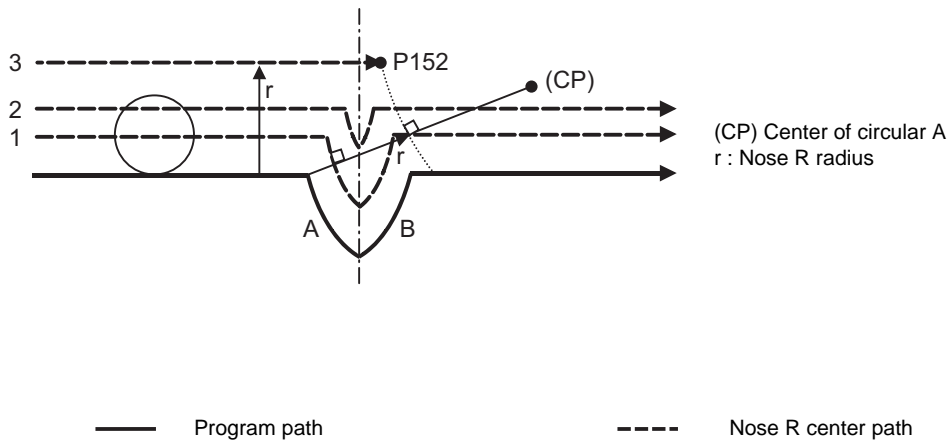
—— Program path

----- Nose R center path

- (3) When the circular end point is not on the circular
 If the error after performing the compensation is within the parameter (#1084 RadErr) range, circular's start point to end point will be interpolated as a spiral circular.



- (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) will be displayed and the tool stops at the end point of the previous block.
 In the pattern 1 and 2 of this figure, machining is possible because nose R radius r is small.
 In pattern 3, nose R radius r is so large that an intersection does not exist and program error (P152) will occur.



Tool nose R compensation cancel

Tool nose R compensation mode will be canceled when any of the following conditions is met. However, the movement command must be a command which is not a circular command. If the compensation is canceled by a circular command, program error (P151) will occur.

- (1) The G40 command has been executed.
- (2) Tool No. T00 is executed.

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

Tool Nose Radius Compensation Operations

Tool nose radius compensation cancel command results as follows.

- (1) If the command before G40 is G00 after the tool nose radius compensation is completed, the tool nose radius compensation is temporarily stopped, and the tool nose radius compensation is canceled in that state by an independent G40 command.
- (2) If there is an interpolation command before G40 after the tool nose radius compensation is completed, the tool nose radius compensation is not canceled by the independent G40 command, so the nose R center can stop in a vertical position. Instead, the tool nose radius compensation is canceled by the first axis movement command after G40. The nose radius compensation will not be canceled by the axis command if there is no axis movement. If there is no axis movement command after G40, and the program finishes by M02, etc., the tool nose radius compensation will remain valid. Resetting will cancel the tool nose radius compensation, but the operation will not be canceled.
- (3) When an independent T00 command is issued in a block, a tool nose radius cancel mode will result in that block, and the axis will move to the tool nose radius cancel position.

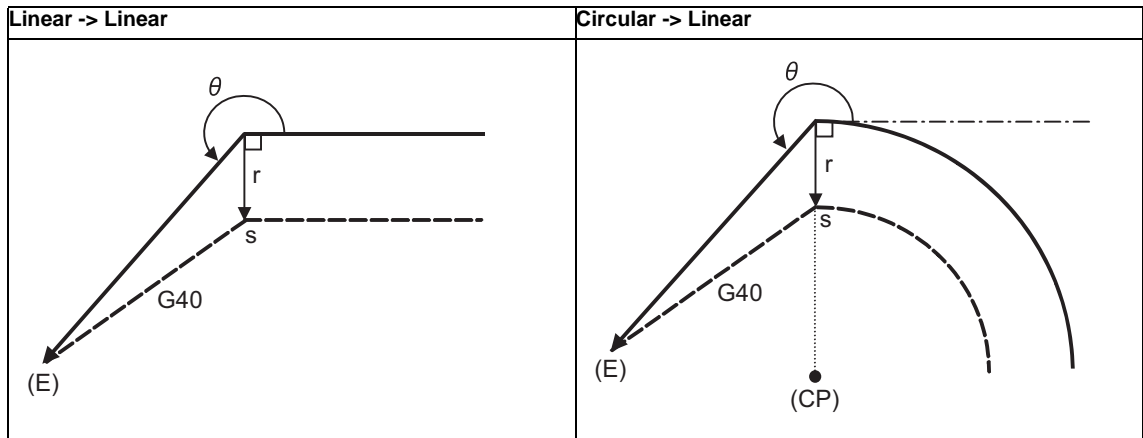
(4) Relation of an inside corner/outside corner and cancel
 (a)-1 When an independent G40 command is issued at an inside corner

<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G40; : </pre>	
<pre> : N1 G01 X_Z_F_; N2 G01 X_Z_; N3 G40; N4 M05; N5 G00 X_Z_; : </pre>	
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G01 X_Z_; N4 G40; N5 M05; N6 G01 X_Z_; : </pre>	
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G00 X_Z_; N4 G40; : </pre>	

— Program path

- - - - - Nose R center path

(a)-2 When a G40 command is issued at an inside corner in the same block as a movement command



(E) End point (CP) Center of circular r : Nose R radius s : Stop point with single block

—— Program path

----- Nose R center path

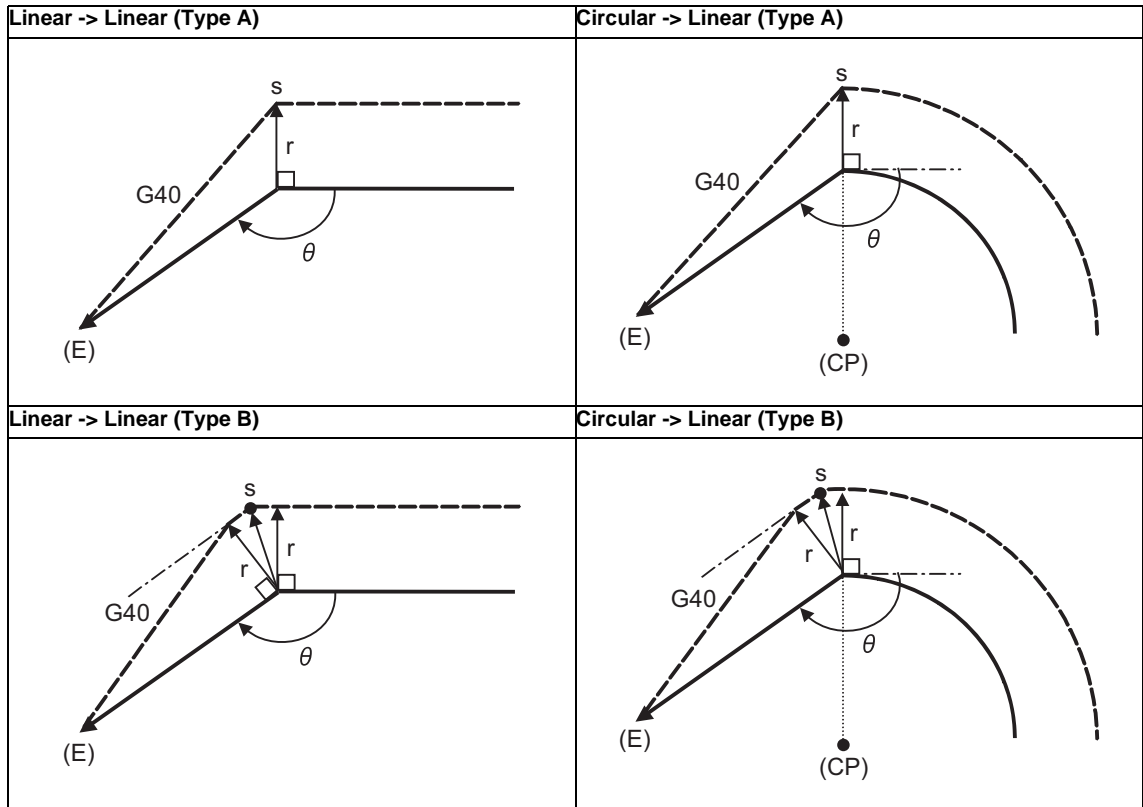
(b)-1 When an independent G40 command is issued at an outside corner (obtuse angle)

	Type A	Type B
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G40; : </pre>		
<pre> : N1 G01 X_Z_F_; N2 G01 X_Z_; N3 G40; N4 M05; N5 G00 X_Z_; : </pre>		
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G01 X_Z_; N4 G40; N5 M05; N6 G01 X_Z_; : </pre>		
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G00 X_Z_; N4 G40; : </pre>		

— Program path

--- Nose R center path

(b)-2 When a G40 command is issued at an outside corner (obtuse angle), in the same block as a movement command



(E) End point (CP) Center of circular r : Nose R radius s : Stop point with single block

—— Program path

----- Nose R center path

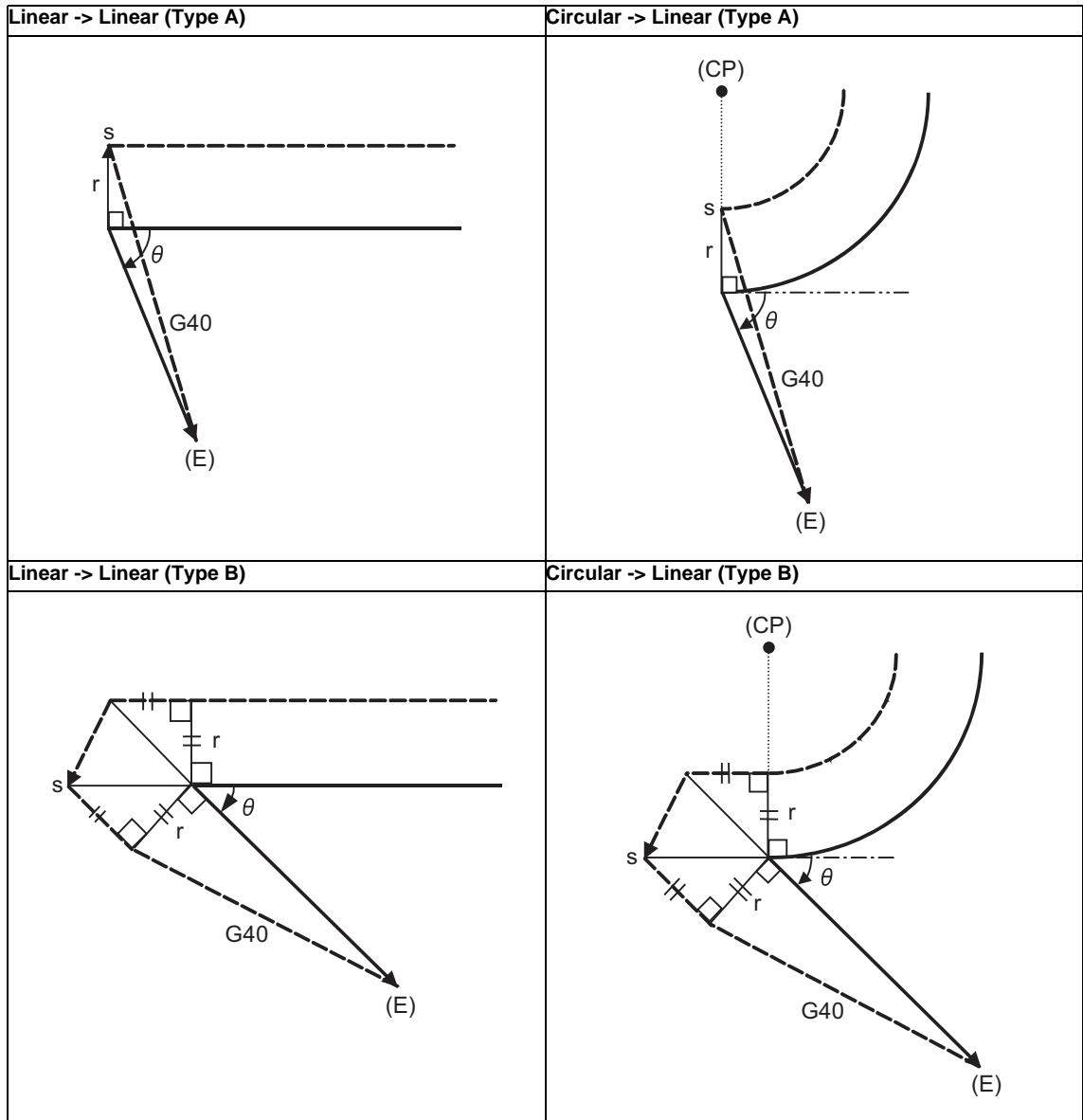
(c)-1 When an independent G40 command is issued at an outside corner (acute angle)

	Type A	Type B
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G40; : </pre>		
<pre> : N1 G01 X_Z_F_; N2 G01 X_Z_; N3 G40; N4 M05; N5 G00 X_Z_; : </pre>		
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G01 X_Z_; N4 G40; N5 M05; N6 G01 X_Z_; : </pre>		
<pre> : N1 G01 X_Z_F_; N2 G00 X_Z_; N3 G00 X_Z_; N4 G40; : </pre>		

— Program path

---- Nose R center path

(c)-2 When a G40 command is issued at an outside corner (acute angle), in the same block as a movement command



(E) End point (CP) Center of circular r : Nose R radius s : Stop point with single block

—— Program path

---- Nose R center path

12.4.3 Other Operations during Tool Nose Radius Compensation



Detailed description

Changing the compensation direction during tool nose radius compensation

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

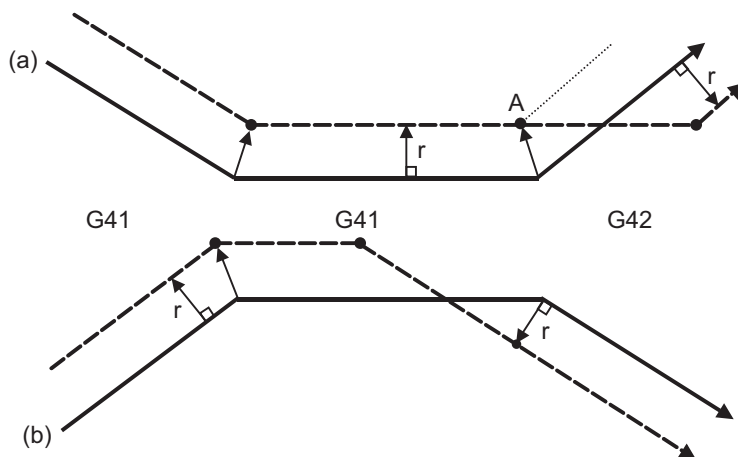
G code	Compensation direction
G41	Left-side compensation
G42	Right-side compensation

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.

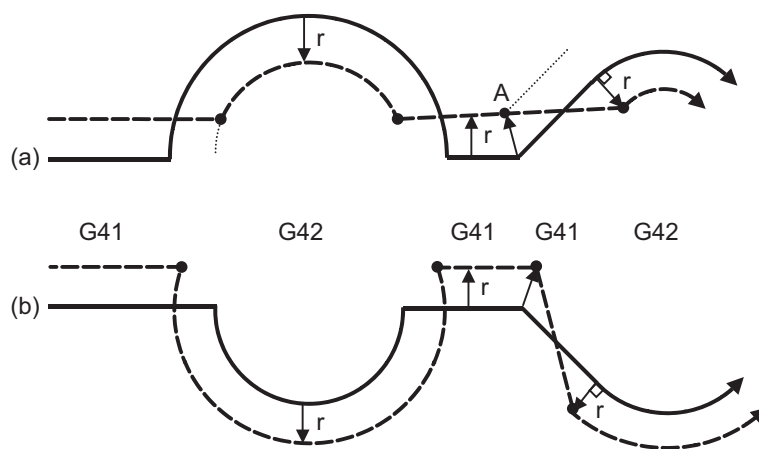
(1) Linear -> Linear

- (a) When there is an intersection (A) at the change of compensation direction
- (b) When there is no intersection at the change of compensation direction



(2) Linear <--> Circular

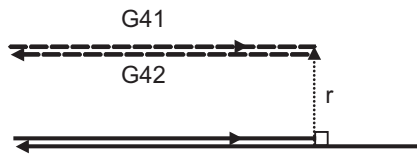
- (a) When there is an intersection (A) at the change of compensation direction
- (b) When there is no intersection at the change of compensation direction



— Program path

--- Nose R center path

(3) Linear return

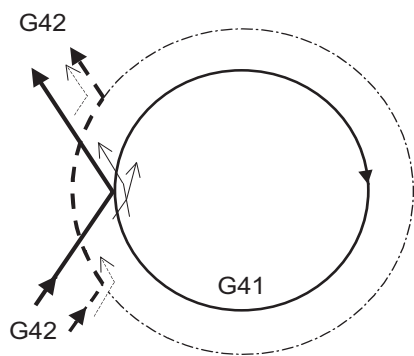


(4) Arc exceeding 360° due to compensation method

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

a. With compensation direction selection based on G41/G42

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



```

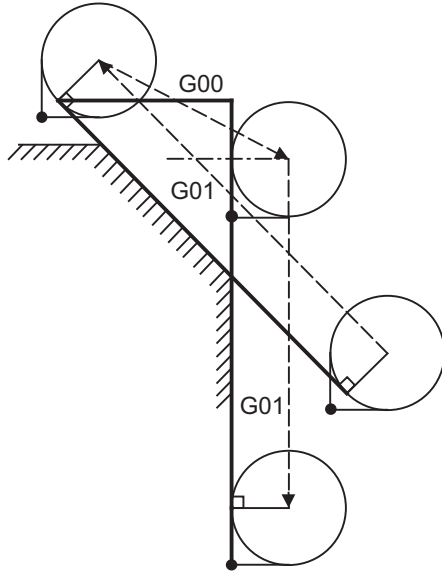
:
G42 G01 X_ Z_;
G41 G02 X_ Z_ I_ K_;
G42 G01 X_ Z_;
:
    
```

—— Program path
 - - - - Uncut section

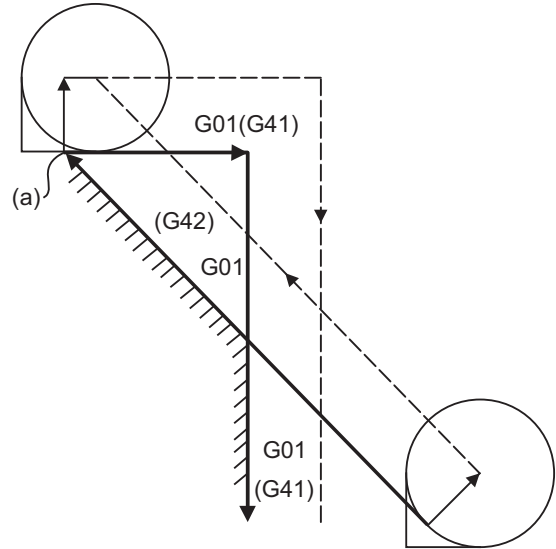
----- Nose R center path

Tool nose radius compensation of path closed by G46/G41/G42

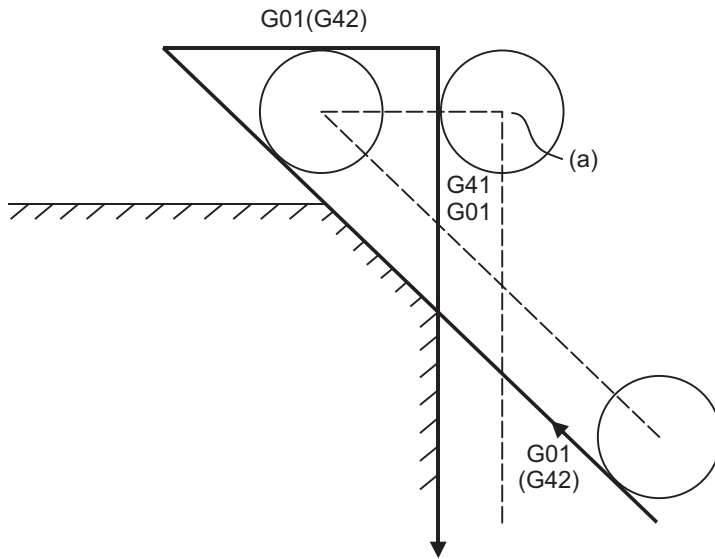
(1) G46 command operation



(2) G42 -> G41 command operation (When commanding G41 at (a))



(3) G42 -> G41 command operation (When commanding G41 at (a))



— Program path

--- Nose R center path

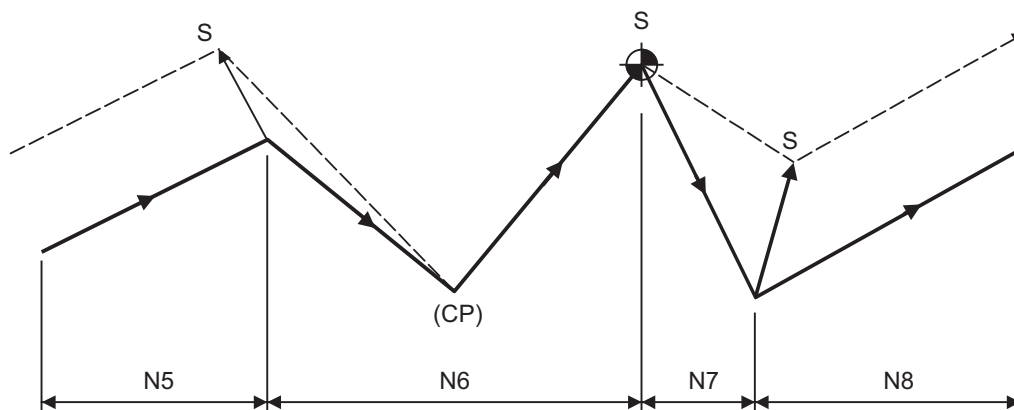
Command for eliminating compensation vectors temporarily

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 point return command

Compensation vector temporarily becomes 0 at the intermediate point (Reference point if there is no intermediate point).



```
(G41) :
N5 G01 U 30. W 60. ;
N6 G28 U-40. W 50. ;
N7   U-60. W 30. ;
N8   U 40. W 70. ;
:
```

(CP) Intermediate Point

(2) The compensation vector will be eliminated temporarily with the G53 command (Basic machine coordinate system selection).

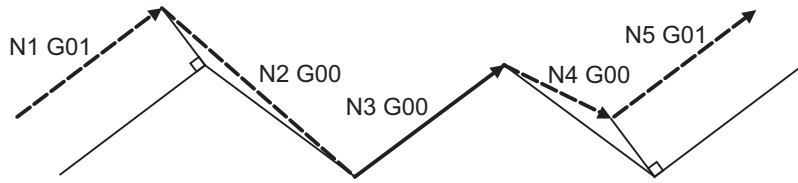
(Note 1) The compensation vectors do not change with the coordinate system setting (G92) command.

(3) Positioning (G00) commands

Tool nose radius compensation is temporarily canceled with G00 commands.

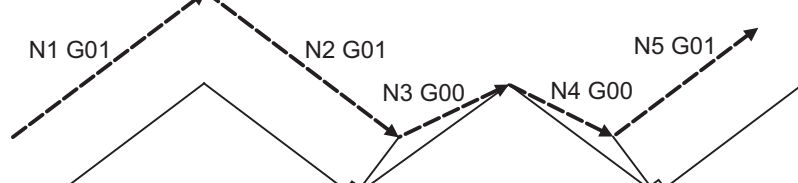
```

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G00 X_Z_;
N5 G01 X_Z_;
:
    
```



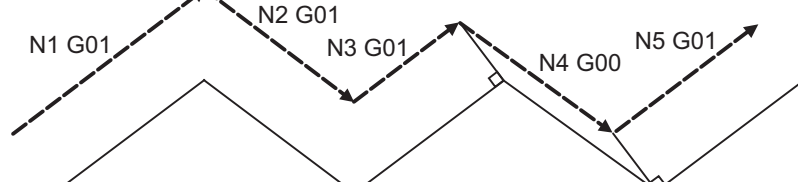
```

:
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G00 X_Z_;
N4 G00 X_Z_;
N5 G01 X_Z_;
:
    
```



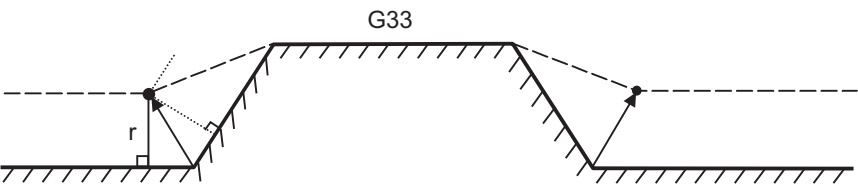
```

:
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G01 X_Z_;
N4 G00 X_Z_;
N5 G01 X_Z_;
:
    
```



(4) G33 thread cutting command

Tool nose radius compensation will not be applied to the G33 block.



(5) Compound type fixed cycle for turning machining

When a compound type fixed cycle for turning machining I command (G70, G71, G72, G73) is issued, the tool nose radius compensation will temporarily be canceled.

The finished shape to which tool nose radius compensation has been applied is cut with the compensation cancel state, and upon completion, operation will automatically return to the compensation mode.

Blocks without movement

The following blocks are known as blocks without movement.

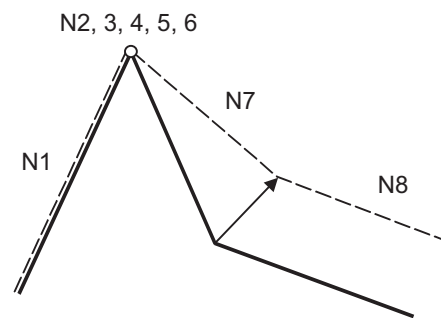
M03 ;	M command
S12 ;	S command
T0101 ;	T command
G04X500 ;	Dwell
G10P01R50 ;	Compensation amount setting
G92X600. Z500. ;	Coordinate system setting
Y40. ;	Movement outside the compensation plane
G00 ;	G code only
U0 ;	Zero movement amount

- (1) When commanded at the compensation start

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

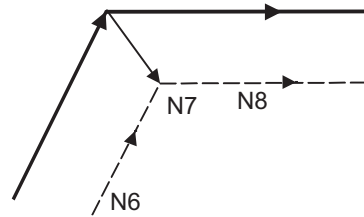
```
N1 U60.W30.T0101 ;
N2 G41 ;
N3 G4 X1000 ;
N4 F100 ;
N5 S500 ;
N6 M3 ;
N7 U- 50.W20. ;
N8 U- 20.W50. ;
```

} Block without movement



- (2) When command is issued during 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 U200. W100. ;
N7 G04X P1000 ;    ... Block without movement
N8 W200. ;
```



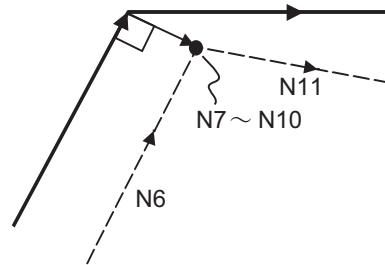
Block N7 is executed at N7 in the figure.

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.

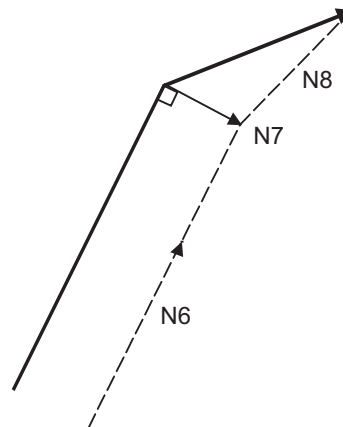
```
N6 U200. W100. ;
N7 G4 X1000 ;
N8 F100 ;
N9 S500 ;
N10 M4 ;
N11 W100. ;
```

} Block without movement



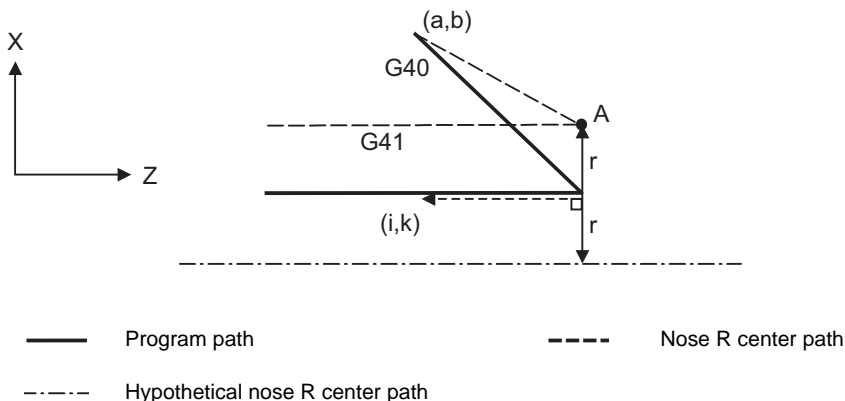
- (3) When commanded together with compensation cancel
 Only the compensation vectors are canceled when a block without movement is commanded together with the G40 command.

```
N6 U200. W100. ;
N7 G40 M5 ;
N8 U50. W100. ;
```



When I, J, K are commanded in G40

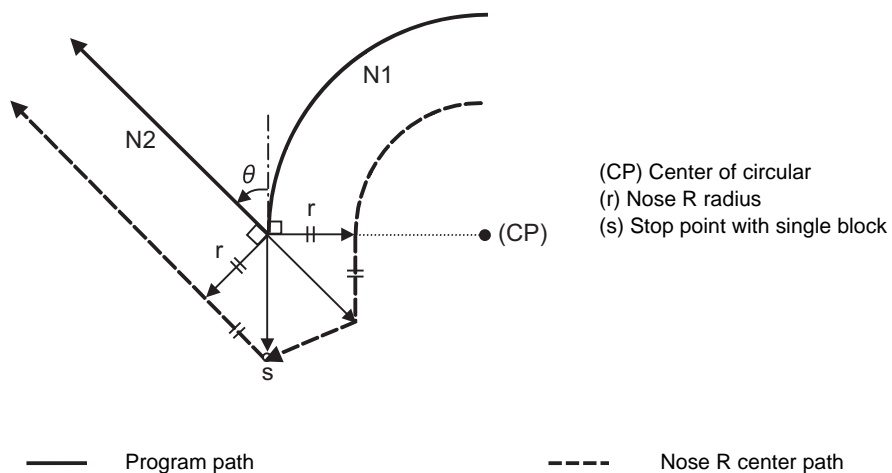
A perpendicular vector is created in the block before G40.



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 those vectors. This action is called corner movement. When the vectors do not coincide, the tool moves in order to machine the corner.

In the single block mode operation, the previous block and corner movement are executed in a single block and the remaining movement and following block are executed in a single block in the next operation.



12.4.4 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

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

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



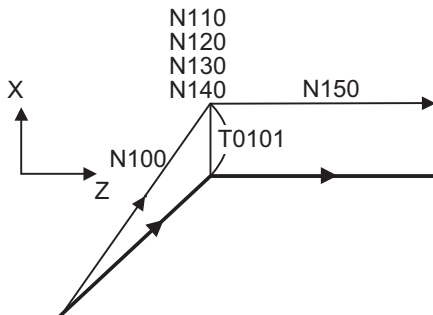
Detailed description

I, K type vectors (G18 X-Z plane selection)

This section describes the new I, K type vectors (G18 plane) created by this command. (Similar descriptions apply to vector I, J for the G17 plane and to J, K for the G19 plane.)

As shown in the following figures, I, K type vectors create compensation vectors which are perpendicular to the direction designated by I, K and equivalent to the compensation amount, without the intersection point calculation of the programmed path. The I, K 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).

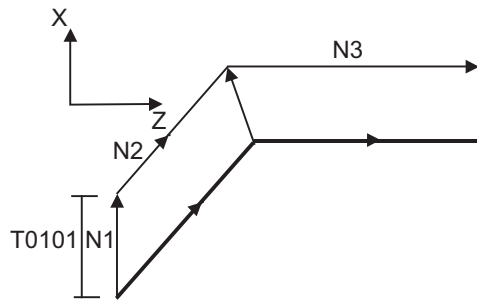
- (1) When I, K is commanded at compensation start



```
(G40)
:
N100 G41 U100. W100.K150.T0101 ;
N110 G04 X1000 ;
N120 G01 F1000 ;
N130 S500 ;
N140 M03 ;
N150 Z150. ;
:
```

- ← Program path
- ← Nose R center path

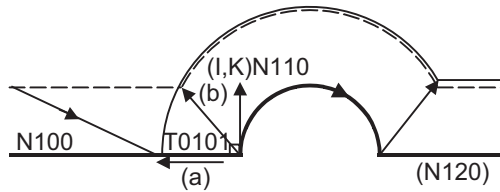
(2) When there are no movement commands at the compensation start.



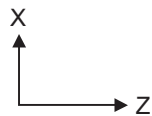
```
(G40)
:
N1 G41 K150. T0101 ;
N2 U100. W100. ;
N3 W150. ;
:
```

← Program path
 ← Nose R center path

(3) When I, K has been commanded in the mode (G18 plane)



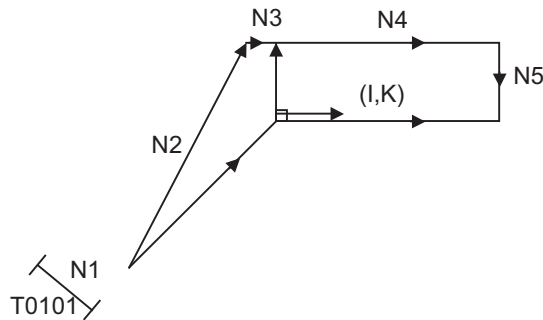
```
:
(G18 G41)
N100 G41 T0101 G01 W150. I50. ;
N110 G02 W100. I50. ;
N120 G01 W100. ;
:
```



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

← Program path
 ← Nose R center path
 - - - - - Path after intersection point calculation

(4) When I, K has been commanded in a block without movement



```
N1 G41 T0101 G01 F1000 ;
N2 U100. W100. ;
N3 G41 K50. ;
N4 W150. ;
N5 G40 ;
```

Offset vector direction

(1) In G41 mode

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

(Example 1) With K100.		(Example 2) With K-100.	
	(0, 100) IK direction		(0, -100) IK direction
	Offset vector direction		Offset vector direction

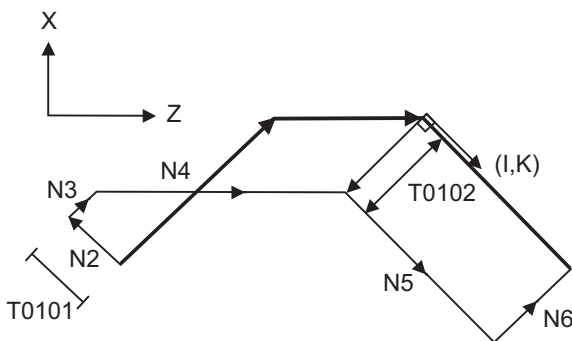
(2) In G42 mode

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

(Example 1) With K100.		(Example 2) With K-100.	
	(0, 100) IK direction		(0, -100) IK direction
	Offset vector direction		Offset vector direction

Selection of offset modal

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



```

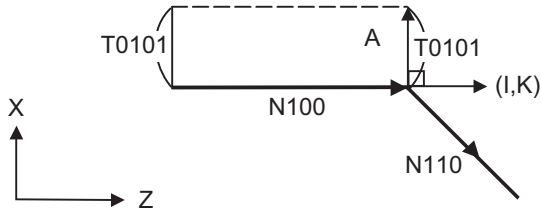
N1 G28 X0Z0 ;
N2 G41 T0101 F1000 ;
N3 G01 U100. W100. ;
N4 G42 W100. I - 100. K100.T0102 ;
N5 U- 100. W100. ;
N6 G40 ;
N7 M02 ;
%
```

Compensation amount for offset vectors

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

<Example 1>

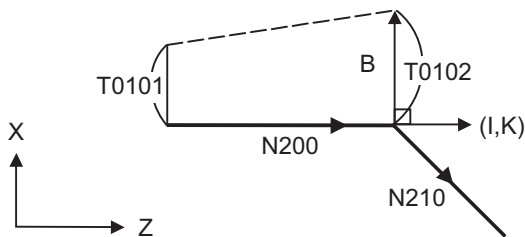
Vector A is the compensation amount registered in tool offset No. modal 1 of the N100 block.



```
(G41 T0101)
:
N100 G41 W150. K50. ;
N110 U- 100. W100. ;
:
```

<Example 2>

Vector B is the compensation amount registered in tool offset No. modal 2 of the N200 block.

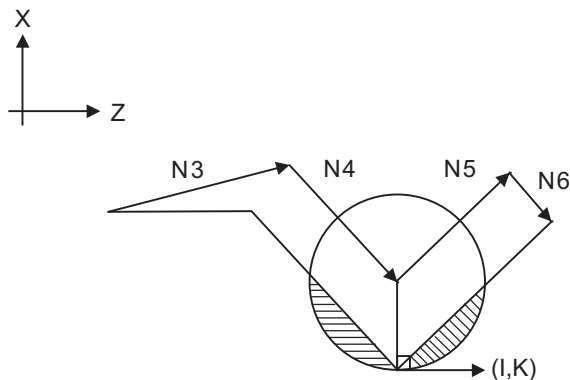


```
(G41 T0101)
:
N200 G41 W150. K50. T0102 ;
N210 U - 100. W100. ;
:
```



Precautions

- Issue the I, K type vector in a linear mode (G0, G1). If it is in an arc mode at the start of compensation, program error (P151) will occur.
 When it is in the offset mode as well as in the arc mode, I, K will be designated at the center of the circular.
- When the I, K 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 X0Z0 ;
N2 G41 T0101 F1000 ;
N3 W100. ;
N4 G41 U-100. W100. K10. ;
N5 U100. W100. ;
N6 G40 ;
N7 M02 ;
```

- 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,K (J)	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, K type vector No insertion block

12.4.5 Interrupts during Tool Nose Radius Compensation



Detailed description

MDI interruption

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

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

Automatic operation

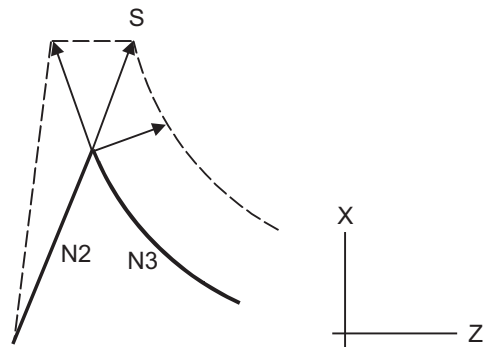
N1 G41 T0101;

N2 U50. W20. ;

N3 G3 U-40. W40. R70. ;

MDI interruption

<--- S1000 M3;



(2) Interrupt with movement

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

With linear interrupt

Automatic operation

N1 G41 T0101;

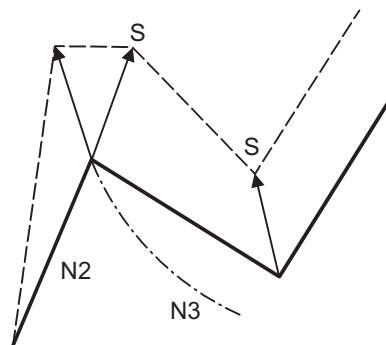
N2 U50. W20. ;

N3 G3 U-40. W40. R70. ;

MDI interruption

<--- U-30. W50. ;

U50. W30. ;



With circular interruption

Automatic operation

N1 G41 T0101;

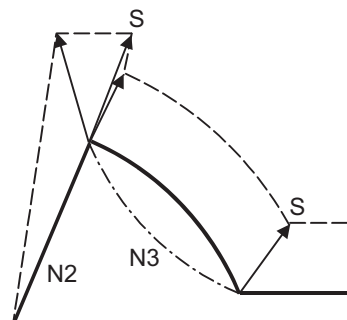
N2 U50. W20. ;

N3 G3 U-40. W40. R70. ;

MDI interruption

<--- G2 U-40. W40. R70. ;

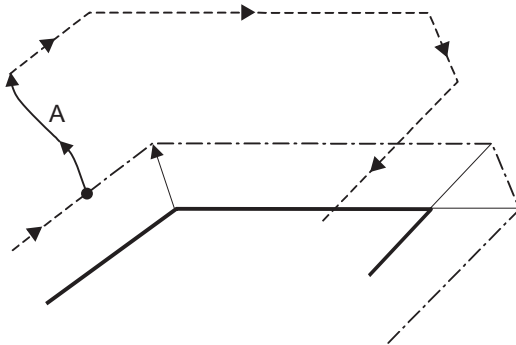
G1 W40. ;



Manual interruption

(1) Interrupt with manual absolute OFF.

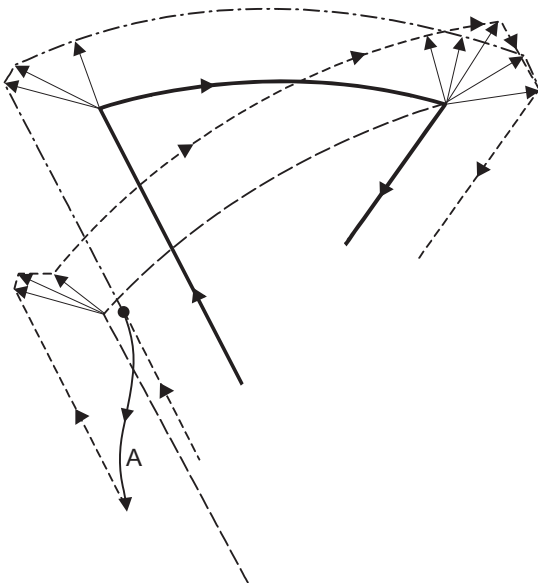
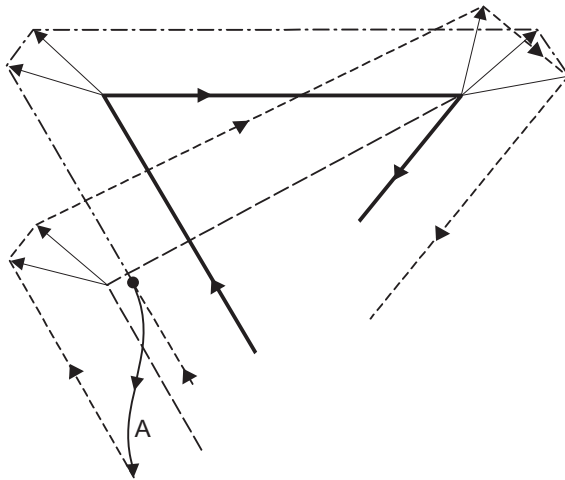
The tool path will deviate from the compensated path by the interrupt amount.



- 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.



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

12.4.6 General Precautions for Tool Nose Radius Compensation



Precautions

Assigning the compensation amounts

- (1) The compensation amount is normally assigned by designating the No. of the compensation amount by the last 1 or 2 digits of the T code. Depending on the machine specifications, the high-order digits may be used. The T code will remain valid once designated until another T code is subsequently commanded. Besides being used to designate the compensation amounts for tool nose radius compensation, the T codes are also used to designate the compensation amounts for tool length 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.

Errors during tool nose radius compensation

- (1) An error will occur when any of the following commands is programmed during tool nose radius compensation.
G17, G18, G19 ("P112" when a plane different from the one used during the compensation is commanded)
G31 (P608)
G74,G75,G76 (P155)
G81 to G89 (P155)
- (2) A program error (P158) will occur when a tool nose point other than 1 to 8 is designated during the G46 mode.
- (3) A program error (P156) will occur when the compensation direction is not determined by the movement vector of the initial cutting command even when the tool nose radius compensation operation has started in the G46 mode and 5 blocks have been pre-read.
- (4) A program error (P151) will occur when a circular command is issued in the first or last block of the tool nose radius compensation.
- (5) A program error (P157) will occur when the compensation direction is reversed in the G46 mode. A parameter can be set to move the tool in the same compensation direction. (Control parameter "#8106 G46 NO REV-ERR")
- (6) A program error (P152) will occur during tool nose radius compensation when the intersection point of single block skip in the interference block processing, cannot be calculated.
- (7) A program error will occur when there is an error in one of the pre-read blocks during tool nose radius compensation.
- (8) A program error (P153) will occur when an interference occurs under no interference avoidance conditions during tool nose radius compensation.
- (9) A program error (P150) will occur when a tool nose radius compensation command is issued even though the tool nose radius compensation specification is not provided.

12.4.7 Interference Check



Function and purpose

A tool, whose tool nose has been compensated under the tool nose radius compensation function by the usual two-block pre-read, may sometimes cut into the workpiece.

This is known as interference, and interference check is the function which prevents this from occurring.

The table below shows the three functions of interference check and each can be selected for use by parameter.

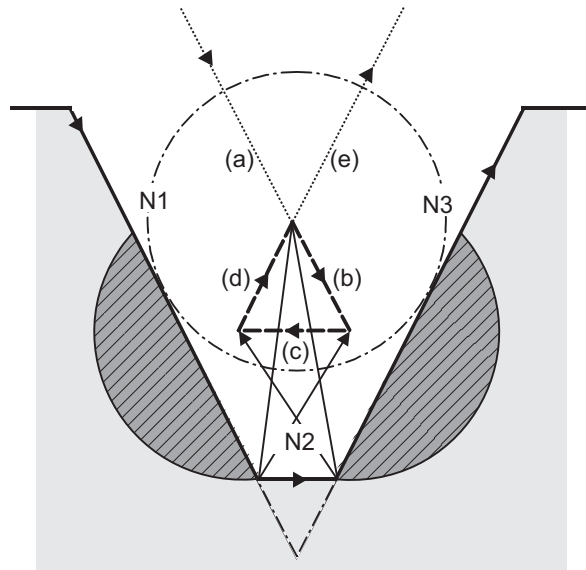
	Function	Parameter		Operation
		#8102 COLL. ALM OFF	#8103 COLL. CHK OFF	
(1)	Interference check alarm function	0	0	Operation stops with a program error before executing a block which will cause cutting.
(2)	Interference check avoidance function	1	0	The tool path is changed to prevent cutting from occurring.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



Detailed description

(Example)

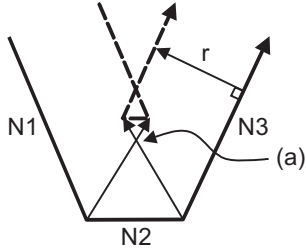
```
(G41)
N1 G91 G1 X100. Z50.;
N2 Z70;
N3 X300. Z120.;
```



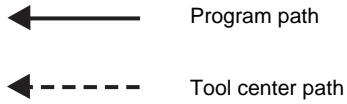
- (1) With alarm function
An alarm is given when N1 is executed. The buffer correction function can thus be used to change N1 to the following, enabling machining to continue: N1 G1 X-100. Z-20. ;
- (2) With avoidance function
The intersection of N1 and N3 is calculated to create interference avoidance vectors.
Tool nose R center path is (a) -> (e).
- (3) With interference check invalid function
The tool passes while cutting the N1 and N3 line.

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.

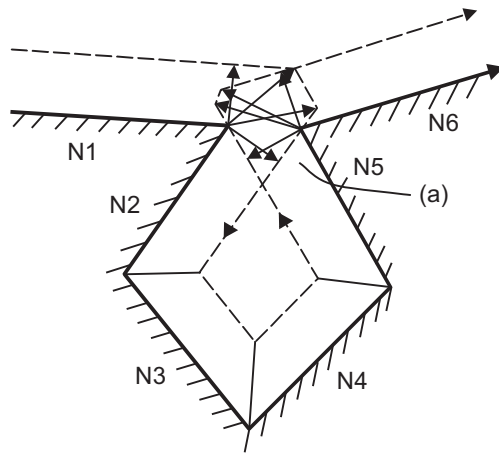


(r) Compensation amount (a) Vectors intersect

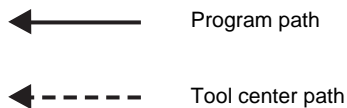


When interference check cannot be executed

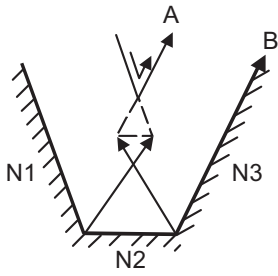
- (1) When three of the movement command blocks cannot be pre-read
 (When there are three or more blocks in the five pre-read blocks that do not have movement)
- (2) When there is an interference following the fourth movement block



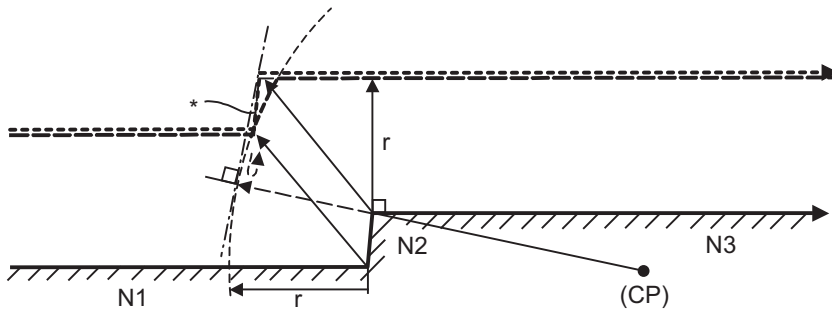
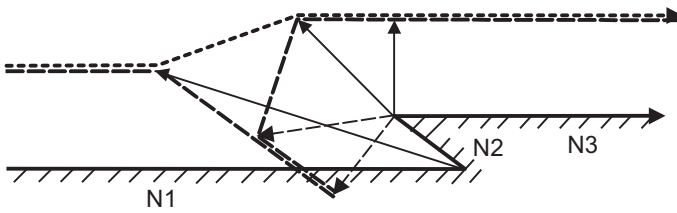
(a) Interference check is not possible



Operation when interference avoidance function is valid

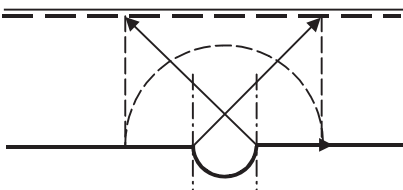


A: Nose R center path B: Program path



- ← Program path
- ← - - - - - Nose R center path when interference check is invalid
- ← - - - - - Tool center path when interference is avoided (*: Linear movement)
- ← Valid vector
- ← - - - - - Invalid vector

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

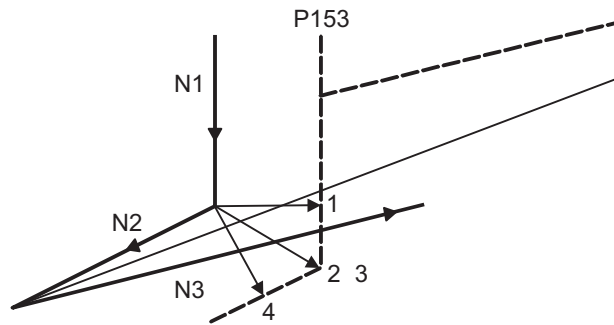


- ← Program path
- ← - - - - - Nose R center path
- ← - - - - - Tool center path when interference is avoided

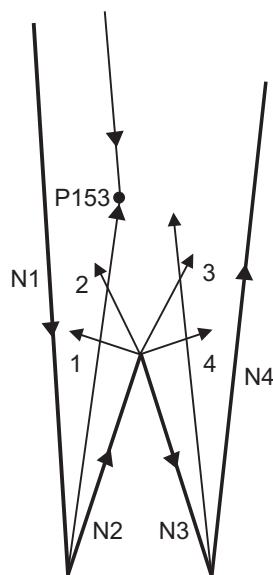
Interference check alarm

The interference check alarm occurs under the following conditions.

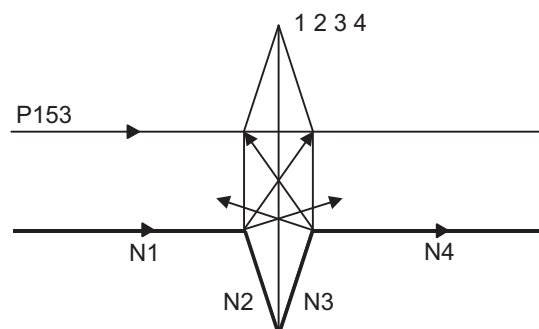
- (1) When the interference check alarm function has been selected
When all the vectors at the end of its own block have been deleted.
When, as shown in the figure below, vectors 1 through 4 at the end point of the N1 block have all been deleted, program error (P153) will occur prior to N1 execution.



- (2) When the interference check avoidance function has been selected
(Ex. 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.



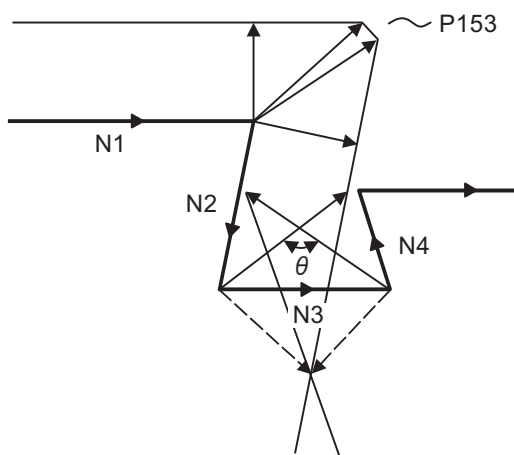
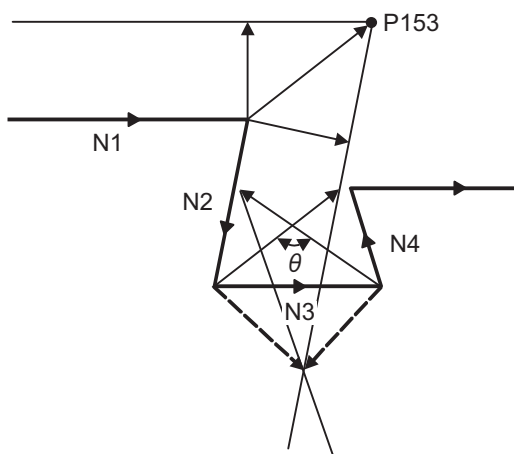
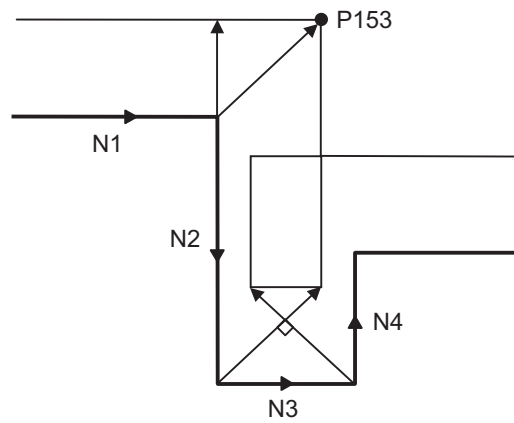
In the case shown in the figure below, the tool will move in the reverse direction at N2.
Program error (P153) now occurs before executing N1 and the operation stops.



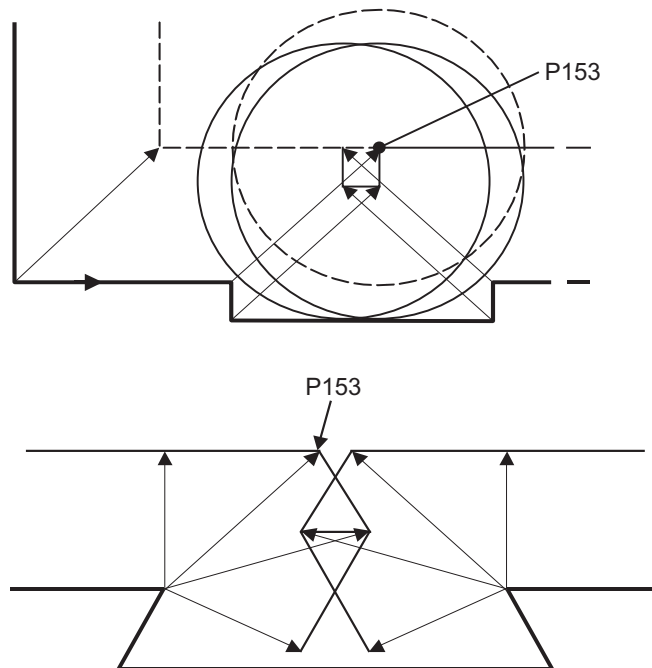
(Ex. 2) When avoidance vectors cannot be created

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.



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



12.5 Compensation Data Input by Program ; G10 L2/L10/L11, G11



Function and purpose

The amount of tool compensation and workpiece offset can be set or changed by the G10 command. When commanded with absolute value (X, Z, R), the compensation amounts serve as the new amounts; when commanded with incremental values (U, W, C), the new compensation amounts are equivalent to the commanded amounts plus the current compensation amount settings.



Command format

G10 L2 P__ X__ (U__)Z__ (W__) ; ... Workpiece offset input (L2)

P	Compensation No.
X	X axis compensation amount (absolute)
U	X axis compensation amount (incremental)
Z	Z axis compensation amount (absolute)
W	Z axis compensation amount (incremental)

G10 L10 P__ X__ (U__)Z__ (W__)R__ (C__)Q__ ; ... Tool length compensation input (L10)

P	Compensation No.
X	X axis compensation amount (absolute)
U	X axis compensation amount (incremental)
Z	Z axis compensation amount (absolute)
W	Z axis compensation amount (incremental)
R	Nose R compensation amount (absolute)
C	Nose R compensation amount (incremental)
Q	hypothetical tool nose point

G10 L11 P__ X__ (U__)Z__ (W__)R__ (C__)Q__ ; ... Tool nose wear compensation input command (L11)

P	Compensation No.
X	X axis compensation amount (absolute)
U	X axis compensation amount (incremental)
Z	Z axis compensation amount (absolute)
W	Z axis compensation amount (incremental)
R	Nose R compensation amount (absolute)
C	Nose R compensation amount (incremental)
Q	Hypothetical tool nose point

(Note) The order of the addresses in a block must be as shown above.

When there is no L command with tool length compensation input (L10) or tool nose wear compensation input (L11).

Tool length compensation input command : P = 10000 + Compensation No.

Tool nose wear compensation input command : P = Compensation No.

G11; ... Compensation input cancel



Detailed description

(1) The following table shows the compensation Nos. and the setting ranges of the hypothetical tool nose points.

Address	Meaning of address	Setting range		
		L2	L10	L11
P	Compensation No.	0: External workpiece offset	When L command is present 1 to Max. number of tool compensation sets	When L command is/is not present 1 to Max. number of tool compensation sets
		1: G54 workpiece offset		
		2: G55 workpiece offset	When L command is not present 10001 to 10000 + Max. number of tool compensation sets	
		3: G56 workpiece offset		
		4: G57 workpiece offset		
		5: G58 workpiece offset		
	6: G59 workpiece offset			
Q	Hypothetical tool nose point		0 to 9	

(Note 1) The maximum number of tool compensation sets for P (compensation No.) with tool compensation input (L10 or L11) is up to a total of 80 with the addition of options.

(The number of sets will differ according to the model so check the specifications.)

(2) 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.

Input unit	Tool length offset amount		Tool wear offset amount	
	Metric system	Inch system	Metric system	Inch system
IS-B	± 99999.999 (mm)	± 9999.9999 (inch)	± 99999.999 (mm)	± 9999.9999 (inch)
IS-C	± 9999.9999 (mm)	± 999.99999 (inch)	± 999.9999 (mm)	± 999.99999 (inch)



Precautions

- (1) Compensation amount setting range check
The maximum value of the wear compensation amount and the maximum additional value for the wear compensation input check respectively take precedence for a single-time compensation amount in the maximum value and incremental value command of the wear compensation amount, and when an amount greater than these values has been commanded, program error (P35) will occur.
- (2) G10 is an unmodal command and is valid only in the commanded block.
- (3) Compensation input can be performed similarly for the 3rd axis but even when the C axis has been designated as the 3rd axis, address C is handled as an incremental command value of the nose R in the L10 or L11 command.
- (4) If an illegal L No. or compensation No. is commanded, the program errors (P172 and P170) will occur respectively.
- (5) When the P command in the workpiece offset input is a command besides 0 to 6, or the P command is omitted, it will be handled as the currently selected workpiece offset input.
- (6) A program error (P35) will occur when the compensation amount exceeds the setting range.
- (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) X, Z and U, W are input together in a single block but when an address that commands the same compensation input (X,U or Z, W) is commanded, the address which is input last is valid.
- (9) Compensation will be input even if one address following G10L(2/10/11) P_ is commanded. A program error (P33) will occur when not even a single command has been assigned.

(Example) G10 L10 P3 Z50. ;

↓

[Tool length data]

#	Z
3	50.000

Input as per left.

- (10) Decimal point is valid for compensation amount.
- (11) G40 to G42 are ignored when they have been commanded in the same block as G10.
- (12) Do not command G10 in the same block as the fixed cycles and subprogram call commands. This will cause malfunctioning and program errors.
- (13) Normally, a program error (P45) occurs when G10/G11 and a fixed cycle are commanded in a same block. When the parameter "#1241 set13/bit0 No G-CODE COMB. Error" is ON, the program error can be avoided but the fixed cycle command will be ignored.
- (14) For the multiple C axis system, both C axis workpiece offset are rewritten with the workpiece offset input.
- (15) An additional axis is input correspond to the axis selected by the additional axis tool offset selection parameter "#1520 Tch34". The axis name is changed by the parameter.

12.6 Tool Life Management II ; G10 L3, G11



Function and purpose

Tool life management divides the tools being used into several groups, and manages the life (usage time, number of uses) of the tools in each group. When it comes to the end of life, a similar spare tool in the same group will be selected in order. This tool life management function with spare tools allows unmanned operation over a long time.

- | | |
|----------------------------------|--|
| (1) Number of life-managed tools | 1-part system : Maximum 80 tools
2-part system/3-part system : Maximum 40 tools/part system |
| (2) Number of groups | 1-part system : Maximum 80 groups
2-part system/3-part system : Maximum 40 groups/part system |
| (3) Group No. | 1 to 9999 |
| (4) Number of tools in one group | Maximum 16 tools |
| (5) Life time | 0 to 999999 minutes (approx. 16667 hours) |
| (6) Number of lives | 0 to 999999 times |

The tool life management data can be set from the NC program or from the Tool Life Management screen. Refer to the Instruction Manual for the method of setting from the Tool Life Management screen. When using the NC program, register the data with the same method as programmable compensation data input.



Command format

```
G10 L3 ;
P__ L__ N__ ; (First group)
T__ ;
T__ ;
P__ L__ N__ ; (Next group)
T__ ;
T__ ; ... Life management data registration start
```

P	Group No., (1 to 9999)
L	Life per tool (0 to 999999 minutes, or 0 to 999999 times)
N	Method (0:Time management, 1: Number of times management)
T	Tool No. The spare tools are selected in the order of the tool Nos. registered here. (Tool No. 1 to 999999. Compensation No. 1 to 80) Tn follows the specifications.

```
G11 ; ... End of life management data registration
```



Program example

(1) Format

⋮ T □□□□ 99 ;	Start use of □□□□ group tool
⋮ ⋮ ⋮	
⋮ T □□□□ 88 ;	Cancel □□□□ group tool compensation (Equivalent to T Δ Δ 00: Δ Δ is No. of the tool being used)
⋮ ⋮ ⋮	
M02(M30) ;	End of machining program

(2) Actual example

⋮ T0199 ;	Start use of group 01 tool
⋮ ⋮ ⋮	
⋮ T0188 ;	Cancel group 01 tool compensation. If the No. of the tool being used is 17, this is equivalent to T1700.
⋮ ⋮ ⋮	
⋮ T0609 ;	Selects tool No. 06 and compensation No. 09. * Life management is not carried out for tool 06.
⋮ ⋮ ⋮	
⋮ T0600 ;	Cancel of group 06 tool compensation.
⋮ ⋮ ⋮	
⋮ T0299 ;	Start use of group 02 tool
⋮ ⋮ ⋮	
⋮ T0199 ;	Start use of group 01 tool If the selected tool has several compensation Nos., the second compensation No. will be selected.



Operation example

Example of tool selection operation (When one tool has several compensation Nos.)

- (1) To use several compensation Nos. with one tool, select the next compensation No. for each T □□□□ 99 command.
- (2) If T □□□□ 99 is commanded for more times than the number of registered compensation Nos., the last compensation No. will be selected, and the operation will continue. (Refer to following.)

Registration to group 1	Program	Tool selection
T1701	T0199 ;	Equivalent to T1701
	⋮	
T1702	T0199 ;	Equivalent to T1702
	⋮	
T1703	T0199 ;	Equivalent to T1703
	⋮	
T2104	T0199 ;	Equivalent to T1703
	⋮	
(Group 1)	⋮	: (Hereafter, same until tool 17 reaches the end of life.)

- (3) If the above program is executed after resetting with M02/M30, or by resetting with external reset, the head compensation No will again be selected.



Precautions

- (1) The tool life data is registered by executing the above program in the memory or MDI mode.
- (2) When the above program is executed, all data (group No., tool No., life data) registered previously will be deleted. The registered data is held even if the power is turned OFF.
- (3) The group No. designated with P does not have to be consecutive, but it should be set in ascending order if possible. Because Nos will be displayed in ascending order on the screen, this will make monitoring easier. The group No. cannot be commanded in duplicate.
- (4) If the life data (L_) is omitted, the life data for that group will be "0". If N_ which specifies method is omitted, the method for that group will follow the basic specification parameter "#1106 Tcount".
- (5) Programming with a sequence No. is not possible between G10 L3 and G11.
- (6) If the usage data count valid signal (TCEF) is ON, G10 L3 cannot be commanded.
(P177 Tool life count active)

12.6.1 Counting the Tool Life



Function and purpose

The tool life can be counted with the time-count type or number of uses-count type. The count method and timing for the number of uses-count type can be changed to type 2 with the parameter setting (#1277 ext13/bit0).

If the usage data is equivalent to or more than the life data as a result of the count up, a spare tool in the group will be selected with the next group selection command (T □□□□ 99), and the newly selected tool will be counted.

When all tools in the group have reached their lives and a spare tool cannot be selected, the count will continue.



Detailed description

Counting the time of uses when the time-count type method is selected

In the cutting mode (G01, G02, G03, G31, G33, etc.), the length of time in which the tool is used is counted with 100ms units.

The time is not counted during dwell, machine lock, miscellaneous function lock or dry run. Whether to count or not during single block can be changed by setting the parameter.

- (Note)
- The maximum value for the life is 999999 minutes.
 - The data is displayed in a minute unit on the Tool Life Management screen.

When the number of uses-count type method is selected: Type 1 (#1277 ext13/bit0: 0)

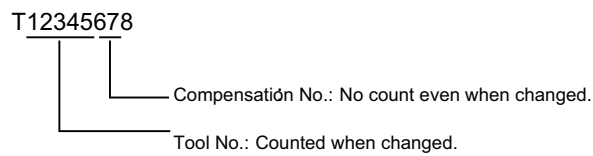
The number of uses is counted when the No. of the tool being used is changed with the tool selection command (T □□□□ 99) and when the program is in the cutting mode (excluding machine lock, miscellaneous function lock and dry run states).

The use is not counted if the cutting mode is not entered even once after the No. changes.

Whether to count or not during single block can be changed with the parameter setting.

- (Note)
- The maximum value for the life is 999999 minutes.
 - If only the compensation No. of the tool being used is changed, counting will not be conducted.

Example: When T code of tool being used is T12345678



<<Operation example 1>>

```

T0199          (1)
:
:
T0299
:
:
T0199          (2)
:
:
T0299
:
:
T0199          (3)
```

Group 01 has been used three times.

<<Operation example 2>>

```

T0199          (1)
:
:
T0299
:
:
T0199
```

Group 01 has been used once.

* The number of uses is for one program execution. If the program is executed again after resetting, it will be counted.

When the number of uses-count type method is selected: Type 2 (#1277 ext13/bit0: 1)

- (1) The groups, used for cutting between the start to reset of the machining program, is added with only "1". The count is made at the reset.
- (2) If recount M is commanded, the group used up to that point will be added with "1" to the counter.

(Note 1) A count is not made in the machine lock, miscellaneous function lock or dry run states.

(Note 2) During single block, select whether to count or not, with the parameter.

(Note 3) The maximum value for the life is 999999 times.

Program Support Functions

13.1 Fixed Cycles for Turning Machining



Function and purpose

When performing rough cutting and other cuttings by turning machining, fixed cycles are effective in simplifying machining programs. The whole commands can be performed in a single block, which normally requires several blocks. The types of fixed cycles for turning machining are listed below.

G code	Function
G77	Longitudinal cutting cycle
G78	Thread cutting cycle
G79	Face cutting cycle



Detailed description

- (1) Fixed cycle commands are modal G codes. They are valid until another command in the same modal group or a cancel command is issued.
The following G code cancel commands are available.
 - G00, G01, G02, G03
 - G09,
 - G10,G11
 - G27, G28, G29, G30
 - G30,
 - G33,G34
 - G37,
 - G92,
 - G52,G53
 - G65,
- (2) The fixed cycle call becomes the movement command block call.
By the movement command block call, the fixed cycle macro subprogram is called only when there is an axis movement command during the fixed cycle mode. It is executed until the fixed cycle is canceled.
- (3) A manual interruption can be applied while a fixed cycle for turning machining (G77 to G79) is being executed. Upon completion of the interrupt, however, the tool must be returned to the position where the manual interruption was applied and then the fixed cycle for turning machining should be resumed.
If it is resumed without returning the tool, all subsequent operations will deviate from the original path by the manual interruption amount.

13.1.1 Longitudinal Cutting Cycle ; G77



Function and purpose

The longitudinal cutting cycle performs continuous straight and taper cutting in the longitudinal direction.



Command format

G77 X/U__ Z/W__ F__ ; ... (straight cutting)

X/U	X axis end point coordinates
Z/W	Z axis end point coordinates
F	Feedrate

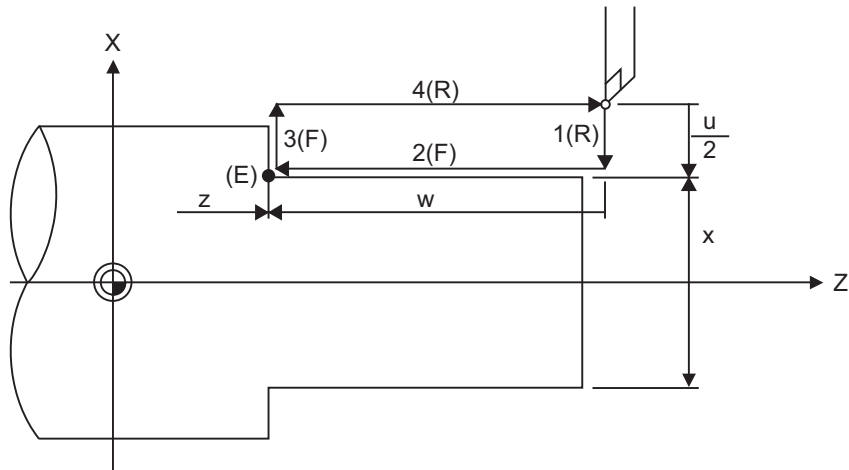
G77 X/U__ Z/W__ R__ F__ ; ... (Taper cutting)

X/U	X axis end point coordinates
Z/W	Z axis end point coordinates
R	Taper depth (radius designation, incremental value, sign required)
F	Feedrate



Detailed description

Straight cutting

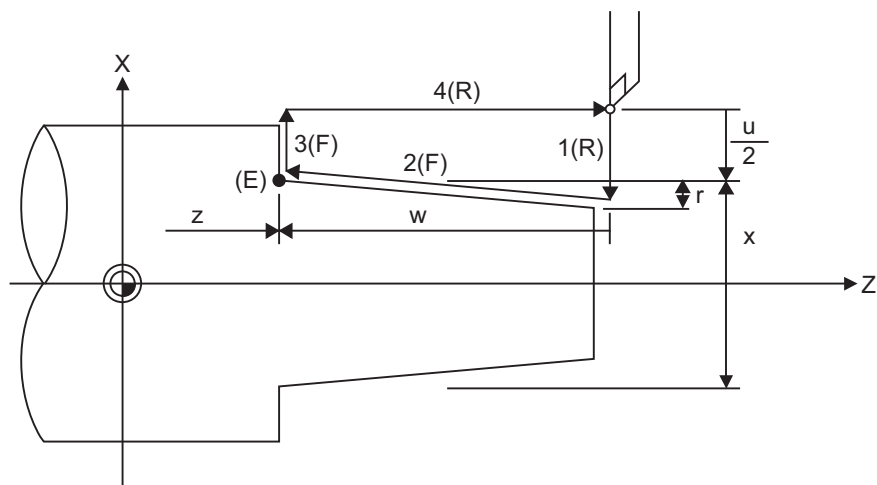


(R) Rapid traverse

(F) Cutting feed

(E) End point coordinates

Taper cutting



(R) Rapid traverse

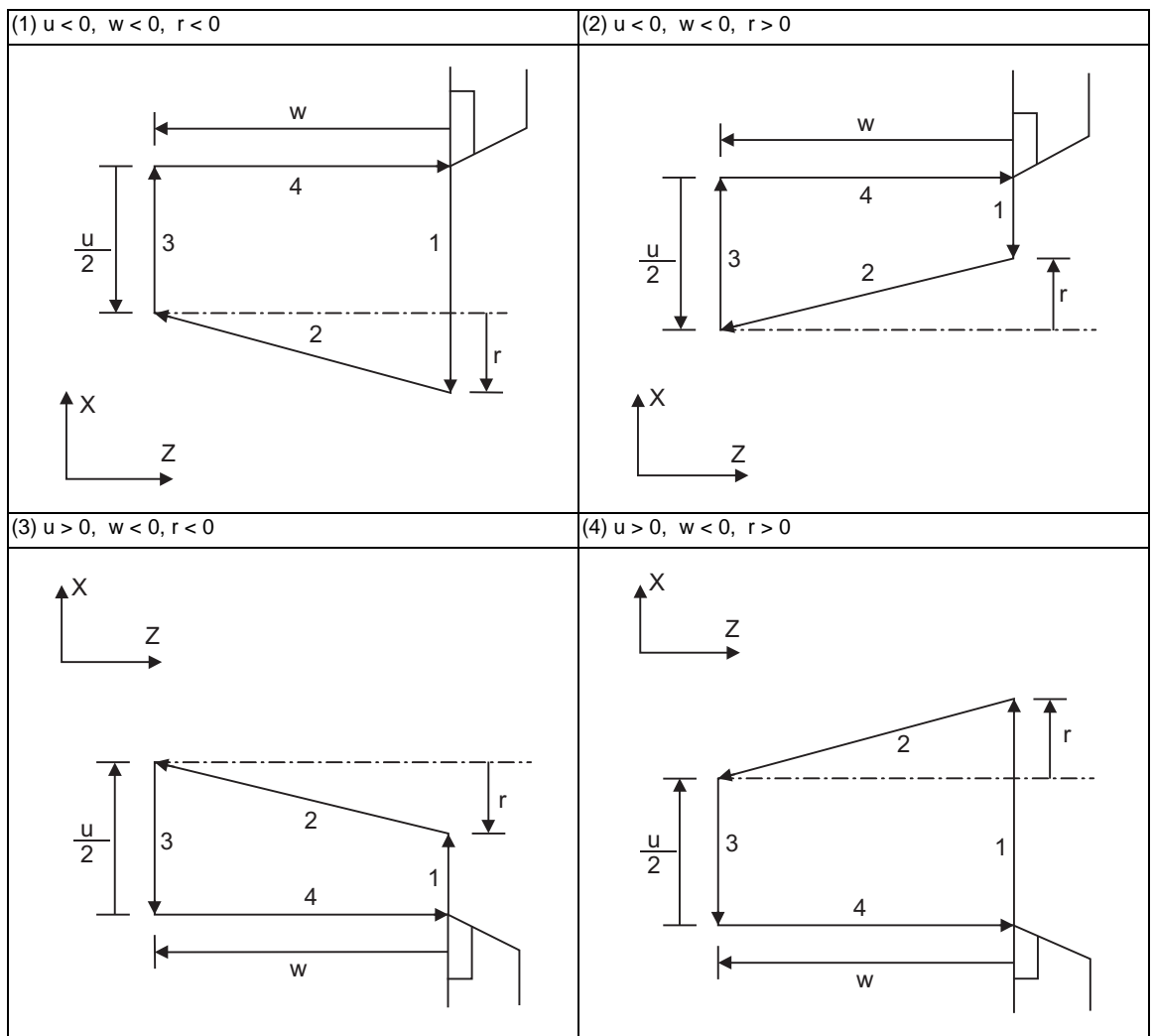
(F) Cutting feed

(E) End point coordinates

r: Taper depth (radius designation, incremental value, sign required)

With a single block, the tool stops at the end points of operations 1, 2, 3 and 4.

Depending on signs of u , w and r , the following shapes are created.



Program error (P191) will occur in (2) and (3) unless the following condition is satisfied.

$$|u/2| \geq |r|$$

13.1.2 Thread Cutting Cycle ; G78



Function and purpose

Thread cutting cycle is a fixed cycle which performs straight and taper thread cutting.



Command format

G78 X/U__ Z/W__ F/E__ ; ... (straight thread cutting)

X/U	X axis end point coordinates
Z/W	Z axis end point coordinates
F/E	Lead of long axis (axis which moves most) direction

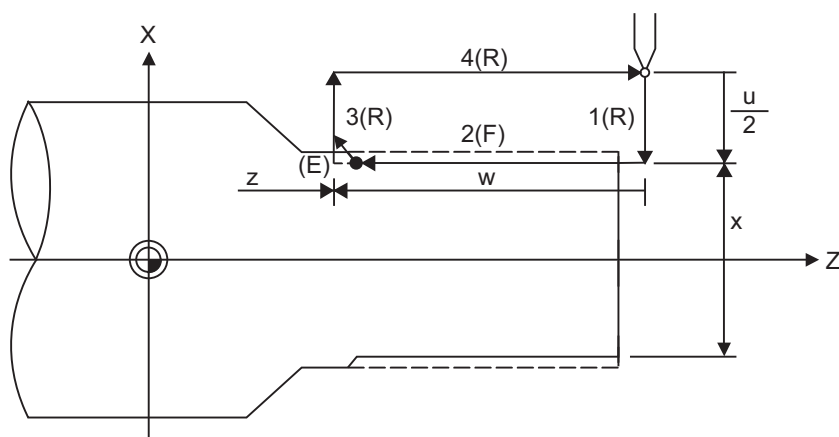
G78 X/U__ Z/W__ R__ F/E__ ; ... (taper thread cutting)

X/U	X axis end point coordinates
Z/W	Z axis end point coordinates
R	Taper depth (radius designation, incremental value, sign required)
F/E	Lead of long axis (axis which moves most) direction



Detailed description

Straight thread cutting



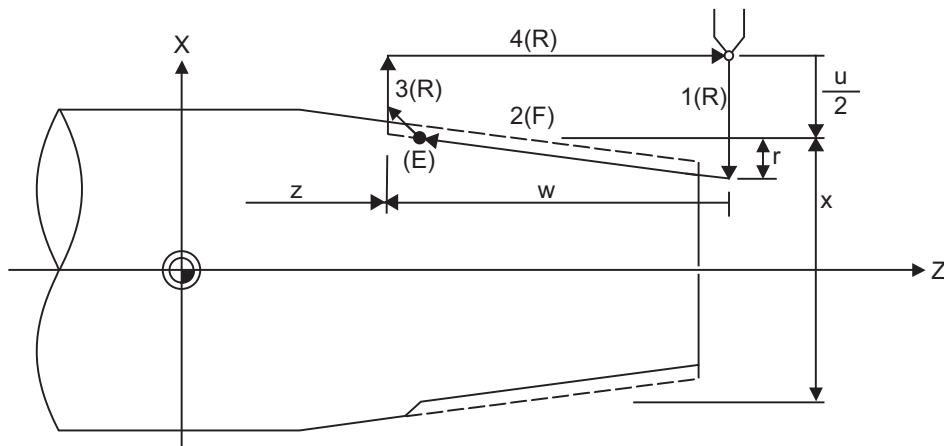
(R) Rapid traverse

(F) Thread cutting cycle

(E) End point coordinates

With a single block, the tool stops at the end points of operations 1, 3 and 4.

Taper thread cutting



(R) Rapid traverse

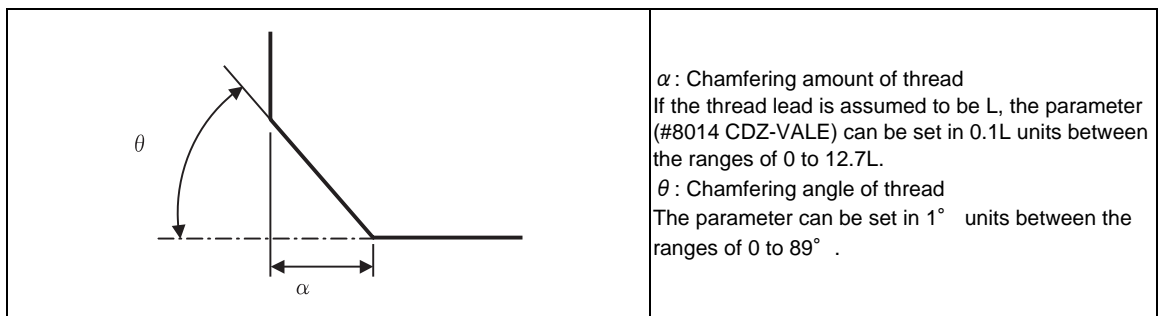
(F) Thread cutting cycle

(E) End point coordinates

r: Taper depth (radius designation, incremental value, sign required)

With a single block, the tool stops at the end points of operations 1, 3 and 4.

(1) Details for chamfering

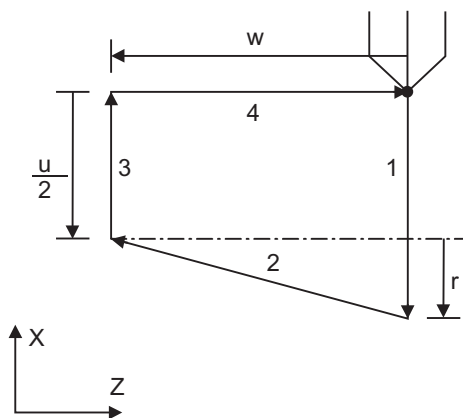


(2) When the feed hold function is applied during a thread cutting cycle, automatic operation will stop if it is applied when thread cutting is not being executed or when cutting command is issued but the axis is yet to move. If thread cutting is proceeding when the function is applied, the operation stops at the next movement completion position (completion of operation 3) of the thread cutting.

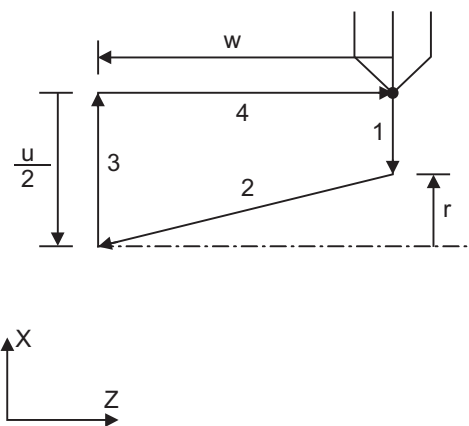
(3) The dry run valid/invalid status does not change during thread cutting.

(4) Depending on signs of u , w and r , the following shapes are created.

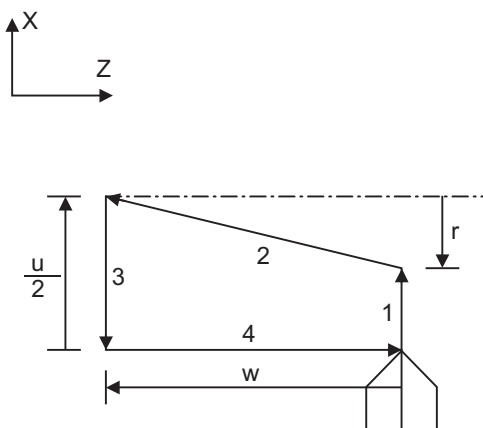
(a) $u < 0, w < 0, r < 0$



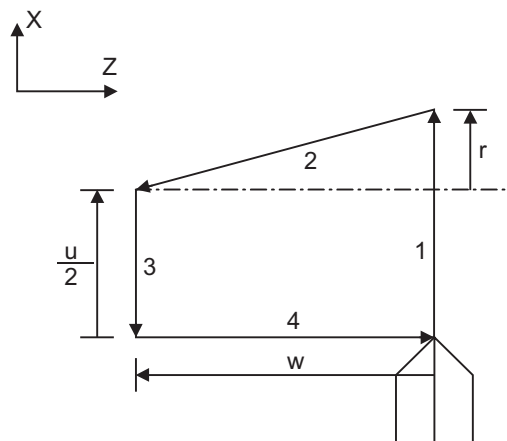
(b) $u < 0, w < 0, r > 0$



(c) $u > 0, w < 0, r < 0$



(d) $u > 0, w < 0, r > 0$



Program error (P191) will occur in (b) and (c) unless the following condition is satisfied.

$$|u/2| \geq |r|$$

13.1.3 Face Cutting Cycle ; G79



Function and purpose

The face cutting cycle performs continuous straight and taper cutting in the face direction.



Command format

G79 X/U__ Z/W__ F__ ; ... (straight cutting)

X/U	X axis end point coordinates
Z/W	Z axis end point coordinates
F	Feedrate

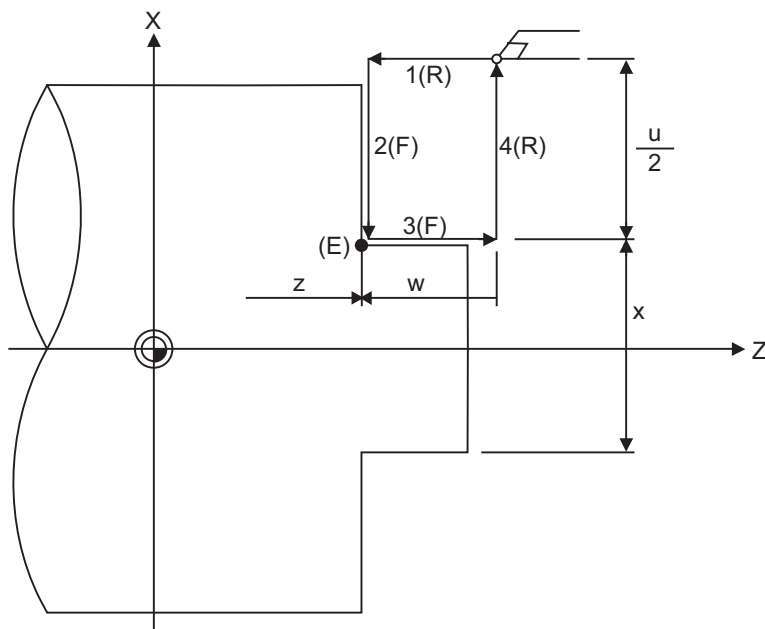
G79 X/U__ Z/W__ R__ F__ ; ... (taper cutting)

X/U	X axis end point coordinates
Z/W	Z axis end point coordinates
R	Taper depth (radius designation, incremental value, sign required)
F	Feedrate



Detailed description

Straight cutting

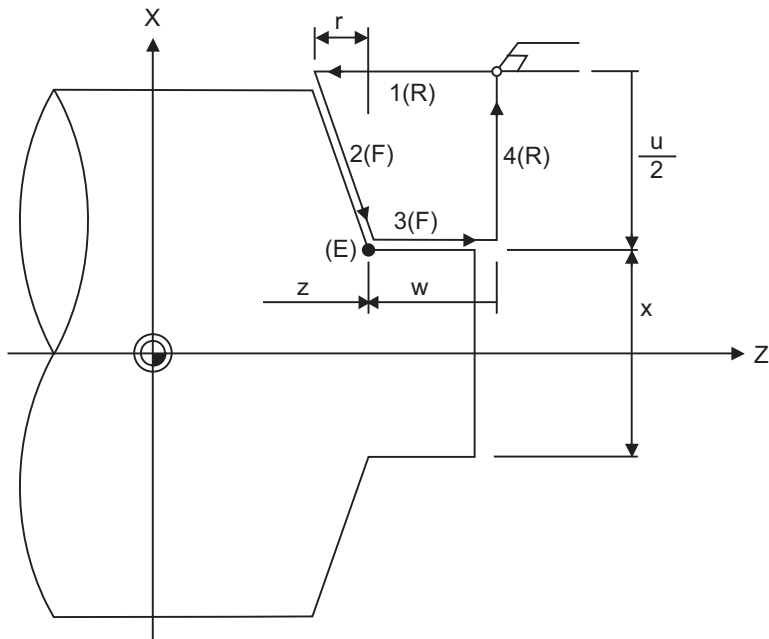


(R) Rapid traverse

(F) Cutting feed

(E) End point coordinates

Taper cutting



(R) Rapid traverse

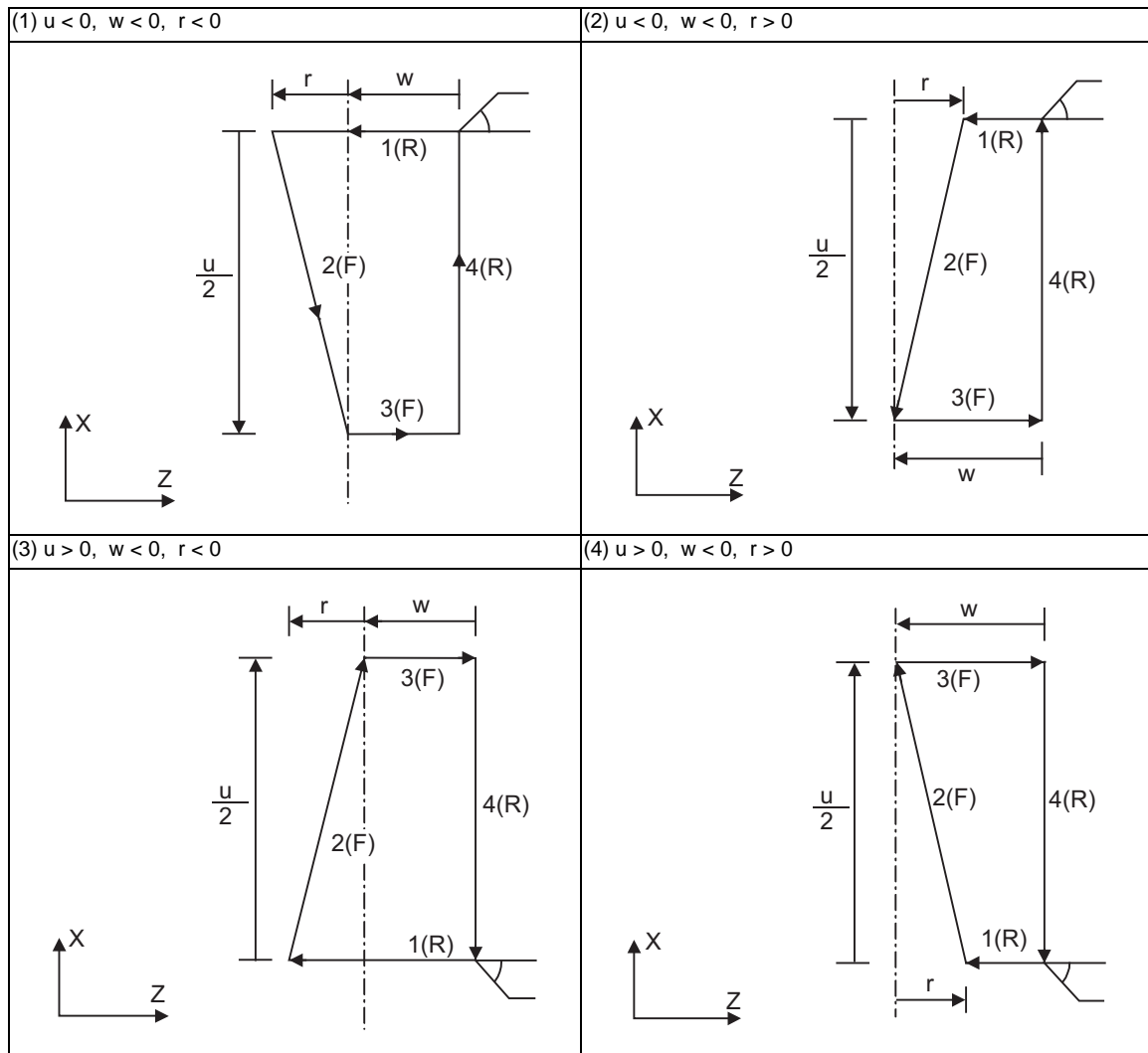
(F) Cutting feed

(E) End point coordinates

r: Taper depth (radius designation, incremental value, sign required)

With a single block, the tool stops at the end points of operations 1, 2, 3 and 4.

Depending on signs of u, w and r, the following shapes are created.



Program error (P191) will occur in (2) and (3) unless the following condition is satisfied.

$$|w| \geq |r|$$

13.2 Compound Type Fixed Cycle for Turning Machining



Function and purpose

This function enables to perform a prepared fixed cycle by commanding a program in a block.

The types of fixed cycles are listed below.

G code	Function	
G70	Finishing cycle	Compound type fixed cycle for turning machining I
G71	Longitudinal rough cutting cycle (finished shape chamfering)	
G72	Face rough cutting cycle (finished shape chamfering)	
G73	Formed Material Rough Cutting Cycle	
G74	Face Cut-Off Cycle	Compound type fixed cycle for turning machining II
G75	Longitudinal cut-off cycle	
G76	Compound thread cutting cycle	

The compound type fixed cycle for turning machining I (G70 to G73) cannot be used if the finished shape program is not registered in the memory.

13.2.1 Longitudinal Rough Cutting Cycle ; G71



Function and purpose

This function calls the finished shape program and, while automatically calculating the tool path, performs rough cutting in the longitudinal direction.



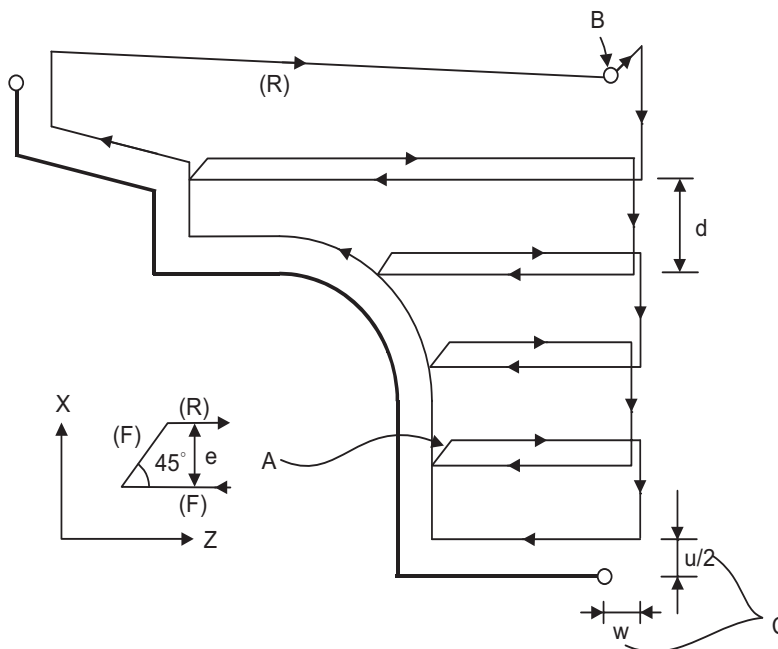
Command format

G71 Ud Re ;
G71 Aa Pp Qq Uu Ww Ff Ss Tt ; ... Longitudinal Rough Cutting Cycle

This fixed cycle requires two blocks.

However, when using a value set by a parameter, the first block can be omitted.

Ud	Cut amount (cut amount without P, Q commands)(modal)
Re	Retract amount (modal)
Aa	Finished shape program No. (Program being executed when omitted)
Pp	Finished shape start sequence No. (Head of program when omitted)
Qq	Finished shape end sequence No. (To the end of program when omitted) If M99 is commanded before Q, the program will end at M99 even if Q is commanded.
Uu	X axis direction finishing allowance (diameter or radius designation).
Ww	Z axis direction finishing allowance
Ff Ss Tt	Cutting speed Spindle command Tool command The F, S and T commands in the finished shape program are ignored, and the value in the rough cutting cycle command or a value prior to it is validated.



A: Detail of retract operation

B: Cycle command point

C: Finishing allowance

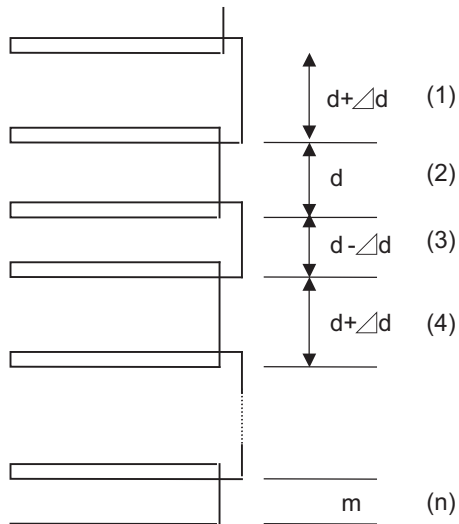
(R): Rapid traverse

(F): Cutting feed

(Note) The U command applied to the finishing allowance when it is in the same block as the A, P and Q commands.

Cut amount: Ud

The cut amount is designated by "d". However, it is possible to change the cut amount with each cutting pass by setting the cut change amount (Δd) using a parameter (#8017 G71 DELTA-D). Program error (P204) results when the amount of one cut commanded in the program is deeper than the cutting depth of the final shape, in other words, when "d" is less than " Δd ".

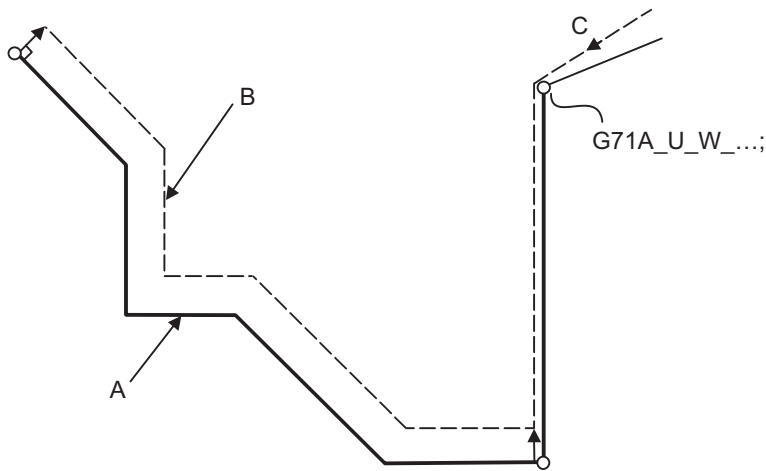


(1) to (n) : 1st cutting to last cutting m : Remainder

Tool nose R compensation

When this cycle is commanded with the tool nose R compensation mode still in force, tool nose R compensation is applied to the finished shape program covered by this cycle and the cycle is executed for this shape.

However, when this cycle is commanded with the tool nose R compensation mode still in force, the mode is temporarily canceled immediately before the cycle and it starts up with the finished shape program, the end block of this program serves as the pre-read prohibit block, and then compensation is applied to the shape and executed.

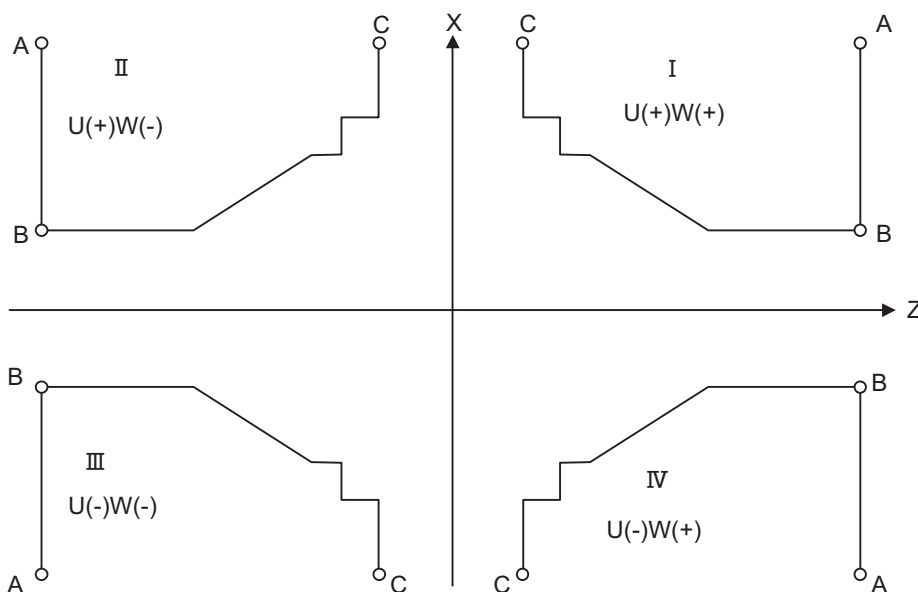


A: Finished shape program
C: Temporary cancellation

B: Shape with tool nose R compensation applied

Others

- (1) After the cutting, the remainder is made the cut amount. However, if this amount is less than the value set by parameter (#8016 G71 MINIMUM), finishing rough cutting is executed and the workpiece is not cut.
- (2) Finishing allowance direction
The finishing allowance direction is determined by the shape as follows. A → B → C apply for the finishing program.





Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.2 Face Rough Cutting Cycle ; G72



Function and purpose

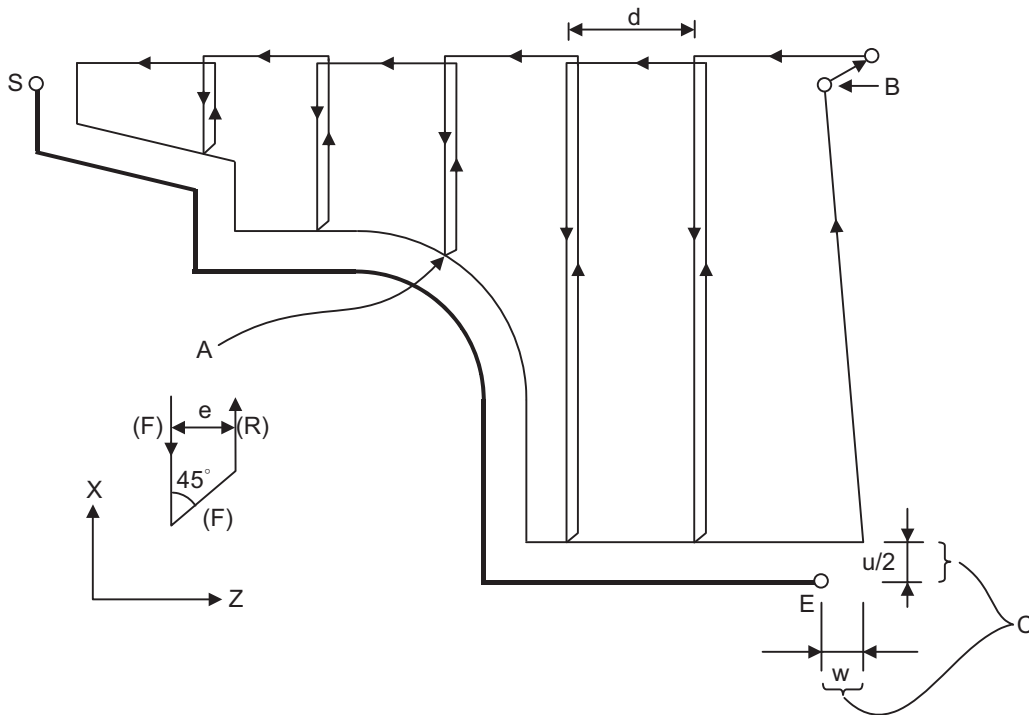
This function calls the finished shape program and, while automatically calculating the tool path, performs rough cutting in the face direction.



Command format

G72 Wd Re ;
G72 Ae Pp Qq Uu Ww Ff Ss Tt ; ... Face Rough Cutting Cycle

Wd	Cut amount (cut amount without P, Q commands)(modal)
Re	Retract amount (modal)
Aa	Finished shape program No. (Program being executed when omitted)
Pp	Finished shape start sequence No. (Head of program when omitted)
Qq	Finished shape end sequence No. (To the end of program when omitted) If M99 is commanded before Q, the program will end at M99 even if Q is commanded.
Uu	X axis direction finishing allowance (diameter or radius designation)
Ww	Z axis direction finishing allowance
Ff	Cutting speed
Ss	Spindle command
Tt	Tool command
	The F, S and T commands in the finished shape program are ignored, and the value in the rough cutting cycle command or a value prior to it is validated.



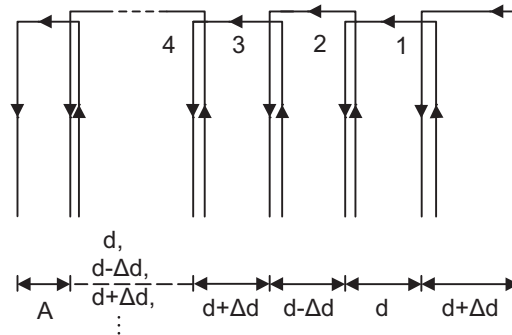
A: Detail of retract operation B: Cycle command point C: Finishing allowance
 (R): Rapid traverse (F): Cutting feed

(Note) The U command applied to the finishing allowance when it is in the same block as the A, P and Q commands.

Cut amount

The cut amount is designated by "d". However, it is possible to change the cut amount with each cutting pass by setting the cut change amount (Δd) using a parameter (#8017 G71 DELTA-D).

1st cutting pass $d + \Delta d$
 2nd cutting pass d
 3rd cutting pass $d - \Delta d$
 4th cutting pass $d + \Delta d$
 :
 : $d, d - \Delta d, d + \Delta d$
 : are repeated



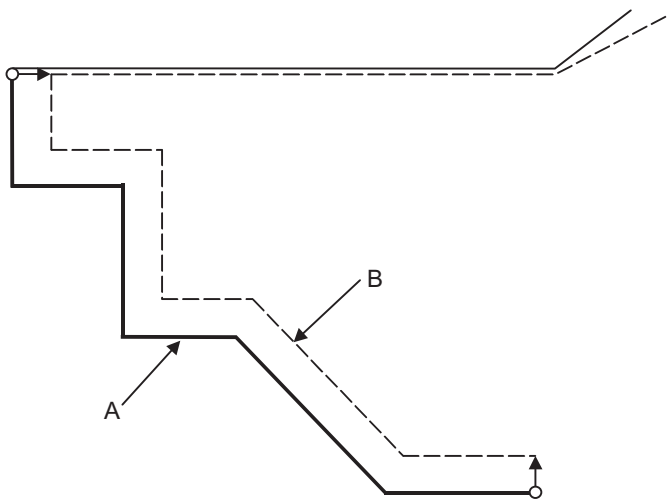
Final cutting pass Remainder

A: Remainder

Tool nose R compensation

When this cycle is commanded with the tool nose R compensation mode still in force, tool nose R compensation is applied to the finished shape program covered by this cycle and the cycle is executed for this shape.

However, when this cycle is commanded with the tool nose R compensation mode still in force, the mode is temporarily canceled immediately before the cycle and it starts up with the finished shape program, the end block of this program serves as the pre-read prohibit block, and then compensation is applied to the shape and executed.



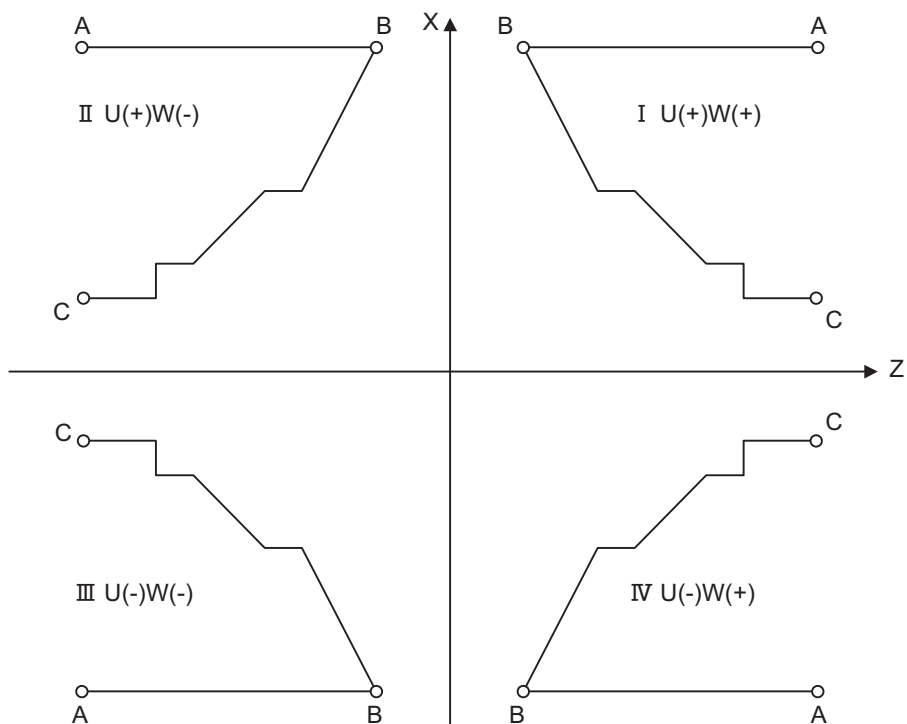
A: Finished shape program

B: Shape with tool nose R compensation applied

Others

- (1) After the cutting, the remainder is made the cut amount. However, if this amount is less than the value set by parameter (#8016 G71 MINIMUM), this cycle is not executed.
- (2) Finishing allowance direction

The finishing allowance direction is determined by the shape as follows.



Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.3 Formed Material Rough Cutting Cycle ; G73



Function and purpose

This function calls the finished shape program, automatically calculates the tool path and performs rough cutting while cutting the workpiece into the finished shape.

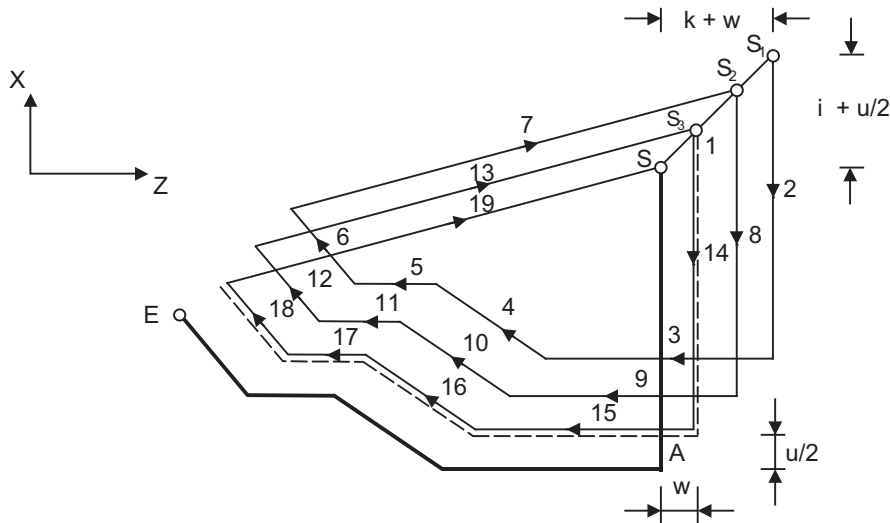


Command format

G73 Ui Wk Rd ;
G73 Aa Pp Qq Uu Ww Ff Ss Tt ; ... Formed Material Rough Cutting Cycle

This fixed cycle requires two blocks.

Ui	X-axis direction cutting allowance i	- Cutting allowances when P, Q commands are not issued
Wk	Z-axis direction cutting allowance k	- Modal data
Rd	Number of divisions d	- Sign is ignored - Radius designation applies to the cutting allowance.
Aa	Finished shape program No.	(program being executed when omitted)
Pp	Finished shape start sequence No.	(program head when omitted)
Qq	Finished shape end sequence No.	(up to end of program when omitted) If M99 is commanded before Qq, the program will end at M99.
Uu	X-axis direction finishing allowance u	- Cutting allowance when P, Q commands are issued.
Ww	Z-axis direction finishing allowance w	- Sign is ignored - Diameter/Radius designation changes in accordance with the parameters (#1019 dia). - The shift direction is determined by the shape. For details, refer to the "finishing allowance direction :Uu, Ww " for G71.
Ff	Cutting feedrate (F function)	The F, S and T commands in the finished shape program are ignored, and the value in the rough cutting cycle command or the value before the command will be valid.
Ss	Spindle speed (S function)	
Tt	Tool command (T function)	



S : G73 cycle command point A : Finished shape start block E : Finished shape end block

(Note) With a single block, operation stops at the end point of each block.

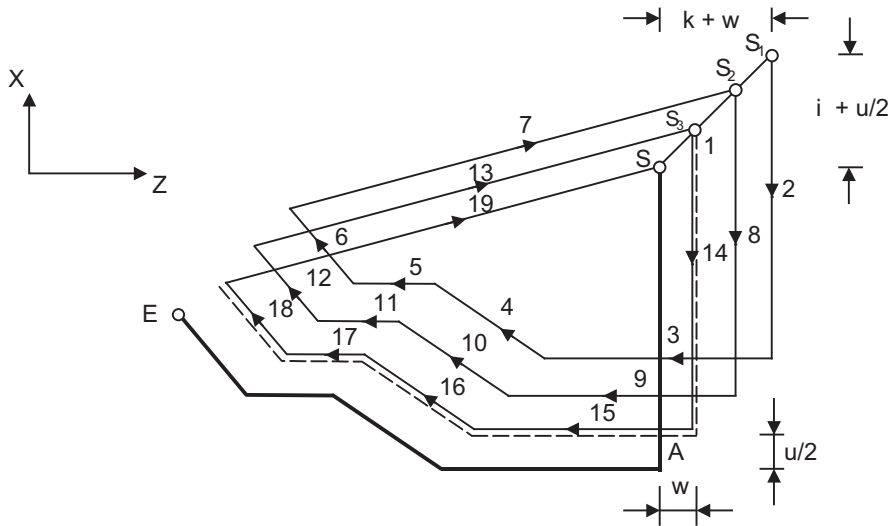


Detailed description

Finished shape

In the program, S -> A -> E in the figure below are commanded.

The section between A and E must be a shape with monotonous changes in both the X axis and Z axis directions.



S : G73 cycle command point

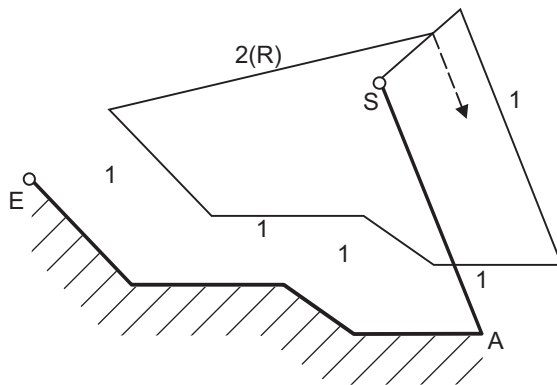
A : Finished shape start block

E : Finished shape end block

(Note) With a single block, operation stops at the end point of each block.

1 cycle configuration

1 cycle is configured as shown below.



1 : Machining with shape profiling (based on shape program)

2 : Return to the next command point (rapid traverse)

S : G73 cycle command point

A : Finished shape start block

E : Finished shape end block

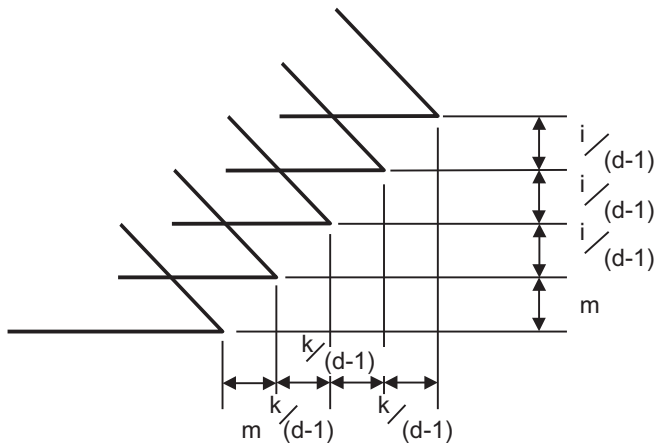
Cut amount

The cut amount is calculated by dividing the cutting allowances (i, k) by the number of divisions (d-1).

X axis direction $i/(d-1)$

Z axis direction $k/(d-1)$

When the allowance is not divisible, chamfering will be performed and adjustment will be made at the final pass.



m : Remainder

Tool nose R compensation

When this cycle is commanded with the tool nose R compensation mode still in force, the compensation is temporarily canceled immediately before this cycle and started at the head block of the finished shape program. So the compensation is applied to the finished shape program covered by this cycle and this cycle is executed for the compensated shape.

Cutting direction

The shift direction for the cutting is determined by the shape in the finishing program, as shown in the table below.

	1	2	3	4
Drawing				
Initial X axis	- direction	-	+	+
Complete Z axis	- direction	+	+	-
X axis cutting	+ direction	+	-	-
Z axis cutting	+ direction	-	-	+

S : G73 cycle command point

A : Finished shape start block

E : Finished shape end block



Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.4 Finishing Cycle ; G70



Function and purpose

After rough cutting have been carried out by the G71 to G73 commands, finishing cutting can be performed by the following command.



Command format

G70 A__ P__ Q__ ; ... Finishing cycle

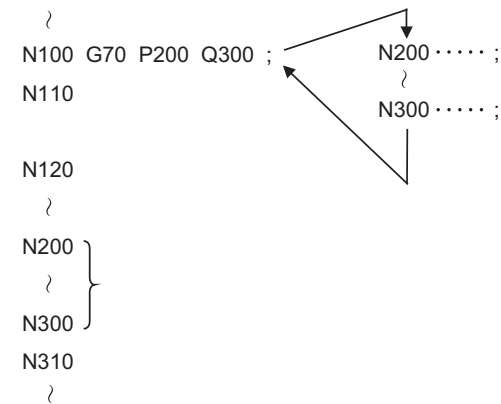
A	Finished shape program No. (Program being executed when omitted)
P	Finished shape start sequence No. (Head of program when omitted)
Q	Finished shape end sequence No. (To the end of program when omitted) If M99 is commanded before Q, the program will end at M99.



Detailed description

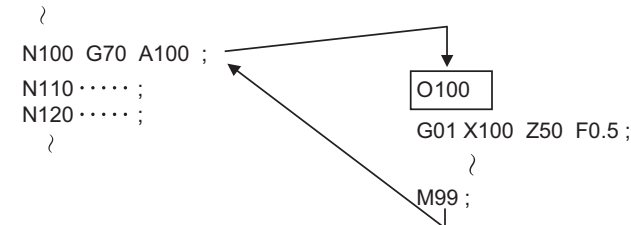
- (1) The F, S and T commands in the finished shape program are valid during the finishing cycle.
- (2) When the G70 cycle is completed, the tool returns to the start point by rapid traverse and the next block is read.

(Example 1) When a sequence No. is designated



N200-N300 ... Finished shape program

(Example 2) When a program No. is designated



In both Example 1 and Example 2, after the N100 cycle is executed, the N110 block will be executed next.



Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.5 Face Cut-Off Cycle ; G74



Function and purpose

The G74 fixed cycle automatically performs grooving in the face direction of the workpiece by commanding the coordinates of the groove end point, cut amount, cutter shift amount and cutter escape at the bottom of the cut. The machining program commands are as follows.



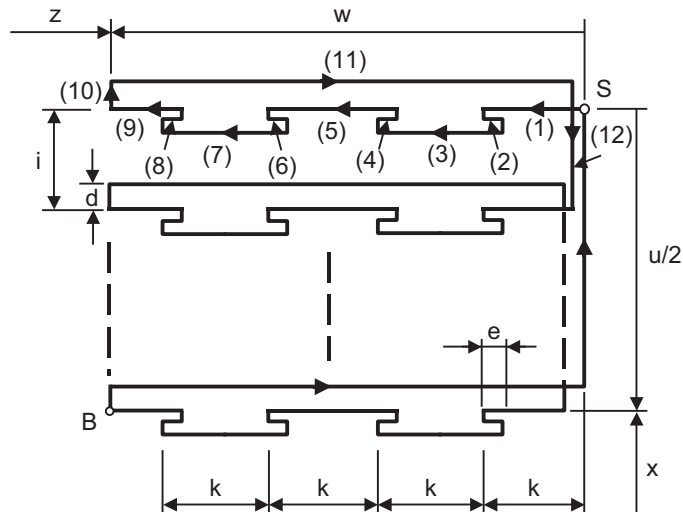
Command format

G74 Re ;
G74 X(U)x Z(W)z Pi Qk Rd Ff ; ... Face Cut-Off Cycle

This fixed cycle requires two blocks.

However, when using a value set by a parameter, the first block can be omitted.

Re	Return amount (no X/U, P commands) (modal)	
X/Ux	B point X coordinate (absolute/incremental value)	
Z/Wz	B point Z coordinate (absolute/incremental value)	
Pi	Tool shift amount (radius designation, incremental value, sign not required)	
Qk	Cut amount (radius designation, incremental value, sign not required)	
Rd	Escape at the bottom of the cut	When there is no sign, the tool escapes even at the bottom of the first cut. When a - sign is attached, the tool escapes from the second cut without escaping the first cut.
Ff	Feedrate	



S : Start point

B : End point

Operations (9) and (12) immediately before the final cycle are executed with the remainder amount.

Operations (2), (4), (6), (8), (10), (11) and (12) are executed at the rapid traverse rate.

With a single block, operation stops at each block.



Detailed description

- (1) When X/U and P are omitted or when the values of "x" and "i" are zero, operation will apply to the Z axis only. Note that when there is an Rd command and no sign, the tool will escape at the bottom of the cut.
- (2) When X/U or Z/W command is not issued, it will be handled as a parameter setting command (G74Re). Even when G74 Pi Qk Rd ; is commanded, Rd is regarded as Re, and the return amount will be set.
- (3) The escape direction does not change whether - sign is attached to the Rd command or not.
- (4) A program error (P204) will occur in the following cases.
 - (a) When " i " is zero or P is not commanded even though X/U is commanded.
 - (b) When tool shift amount " i " is larger than the "x" movement amount.
 - (c) When the escape amount "d" is larger than the shift amount " i ".
 - (d) When the return amount "e" is larger than the cut amount "k".
 - (e) When the cut in amount "k" is larger than the hole depth "w".



Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.6 Longitudinal Cut-off Cycle ; G75



Function and purpose

The G75 fixed cycle automatically performs grooving in the longitudinal direction of the workpiece by commanding the coordinates of the groove end point, cut amount, cutter shift amount and cutter escape at the bottom of the cut.



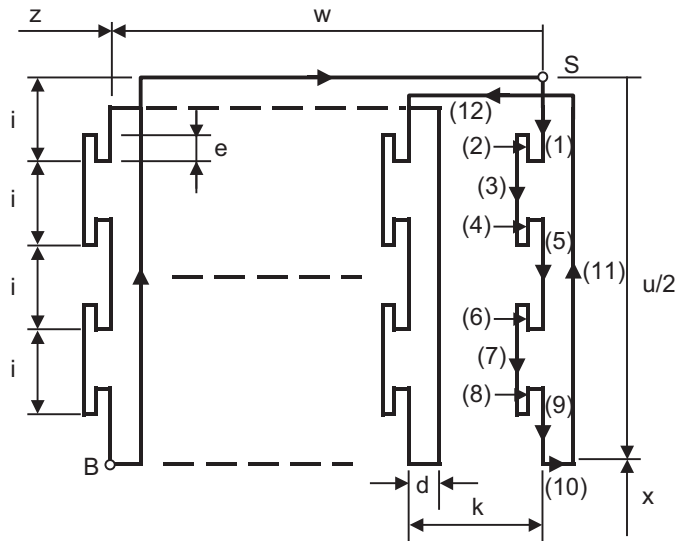
Command format

G75 Re ;
G75 X/(U)x Z/(W)z Pi Qk Rd Ff ; ... Longitudinal cut-off cycle

This fixed cycle requires two blocks.

However, when using a value set by a parameter, the first block can be omitted.

Re	Return amount (no X/U, P commands) (modal)	
X/Ux	B point X coordinate (absolute/incremental value)	
Z/Wz	B point Z coordinate (absolute/incremental value)	
Pi	Cut amount (radius designation, incremental value, sign not required)	
Qk	Tool shift amount (radius designation, incremental value, sign not required)	
Rd	Escape at the bottom of the cut	When there is no sign, the tool escapes even at the bottom of the first cut. When a - sign is attached, the tool escapes from the second cut without escaping the first cut.
Ff	Feedrate	



S : Start point

B : End point

Operations (9) and (12) immediately before the final cycle are executed with the remainder amount.

Operations (2), (4), (6), (8), (10), (11) and (12) are executed at the rapid traverse rate.

With a single block, operation stops at each block.



Detailed description

- (1) When Z/W and Q are omitted or when the values of "z" and "k" are zero, operation will apply to the X axis only (slotting). Note that when there is an Rd command and no sign, the tool will escape at the bottom of the cut.
- (2) When X/U or Z/W command is not issued, it will be handled as a parameter setting command (G74Re). Even when G75 Pi Qk Rd ; is commanded, Rd is regarded as Re, and the return amount will be set.
- (3) The escape direction does not change whether - sign is attached to the Rd command or not.
- (4) A program error (P204) will occur in the following cases.
 - (a) When "k" is zero or Q is not commanded even though Z/W is commanded.
 - (b) When tool shift amount "k" is larger than the "z" movement amount.
 - (c) When the escape amount "d" is larger than the shift amount "k".
 - (d) When the return amount "e" is larger than the cut amount "i".
 - (e) When the cut in amount "i" is larger than the hole depth "u/2".



Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.7 Compound Thread Cutting Cycle ; G76



Function and purpose

The G76 fixed cycle enables to cut the workpiece at a desired angle by designating the thread cutting start point and end point, and it automatically performs cutting so that the cutting cross section (cutting torque) per cutting pass is constant.

Various longitudinal direction threads can be cut by bearing in mind the command value for the thread end point coordinate and taper height component.



Command format

G76 Pmra Rd ;
G76 X/U Z/W Ri Pk QΔd FI ; ... Compound thread cutting cycle

This fixed cycle requires two blocks.

Address		Significance
P	m	Number of cutting passes for finishing: 00 to 99 (times) (modal)
	r	Chamfering amount: 00 to 99 (0.1mm/rev) (modal) ... Reversible parameter "#8014 CDZ-VALE" is also available for setting The chamfering width based on thread lead "λ" is designated by a 2-digit integer without decimal point between the ranges from 0.0 to 9.9.
	a	Tool nose angle (thread angle): 00 to 99 (°) (modal) The angle from 0° to 99° is assigned in 1° units. "m", "r" and "a" are commanded in succession during address P. (Example) When m=5, r=1.5 and a=0°, P is 051500 (P051500). The leading and trailing zeroes cannot be omitted.
R	d	Finishing allowance 0 to 9999 (μm) (modal)
X/U		X-axis end point coordinate of thread The X coordinate of the end point for the thread is commanded by an absolute or incremental value.
Z/W		Z-axis end point coordinate of thread The Z coordinate of the end point for the thread is commanded by an absolute or incremental value.
R	i	Taper height component (radius value) for thread Straight thread when "i" is zero
P	k	Thread height This is commanded by a positive radius value.
Q	Δd	Cut amount The cut amount of the first cutting pass is commanded by a positive radius value.
F	l	Thread lead

(Note 1) A reversible parameter enables to use parameter setting value without issuing a program command and also, the value can be changed by the program command.

(Note 2) The two G76 commands above cannot be assembled in a block.

The data commanded by P and R are automatically identified according to the presence or absence of the X/U and Z/W axis addresses.

(Note 3) The above "r" modal data can be rewritten by the program commands as well as using parameter (#8014 CDZ-VALE) settings.

(Note 4) The chamfering amount designation is valid even for thread cutting fixed cycles.

(Note 5) A program error (P204) will occur in the following cases.

- When "a" is outside the rated value
- When both or one of the X and Z commands is not issued, or when the start and end point coordinates coincides in both or one of the X and Z commands.
- When the thread is larger than the movement of the X axis to the thread bottom

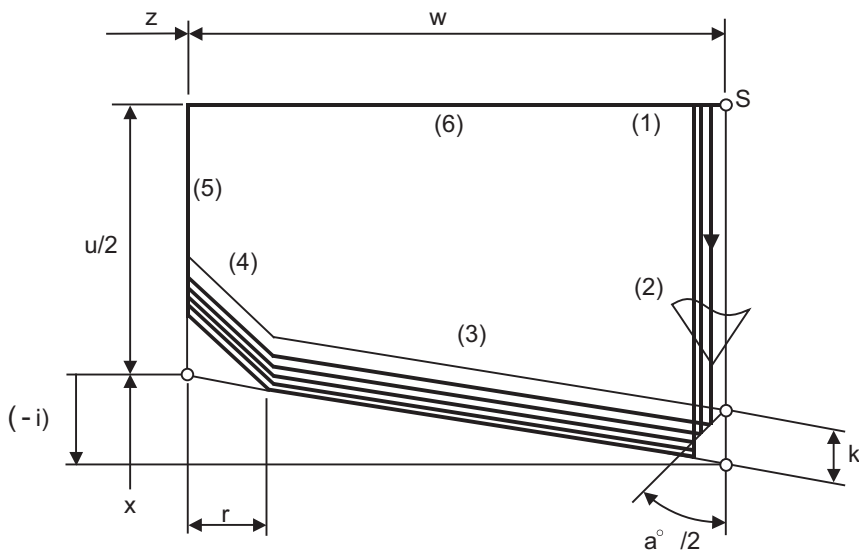
(Note 6) The precautions for the thread cutting command (G33) and thread cutting cycle (G78) should be observed.



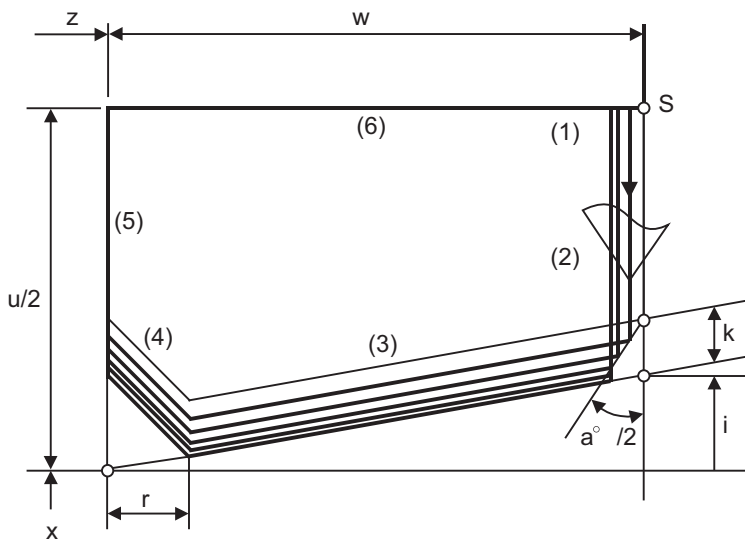
Detailed description

1 cycle configuration

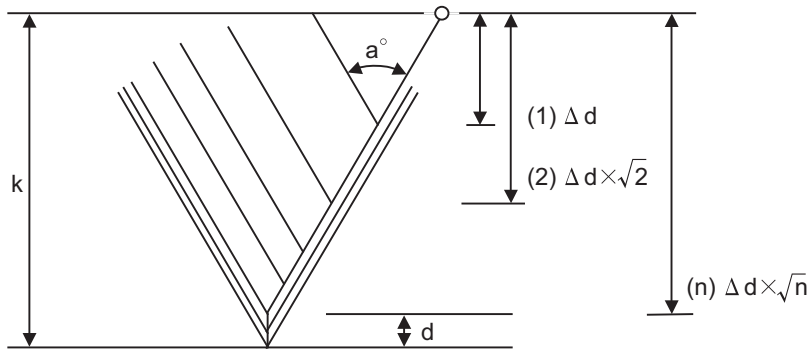
The tool moves at rapid traverse for operations (1), (2), (5) and (6) in the cycle and at the cutting feed based on the F designation for operations (3) and (4) during 1 cycle.



When Ri is negative



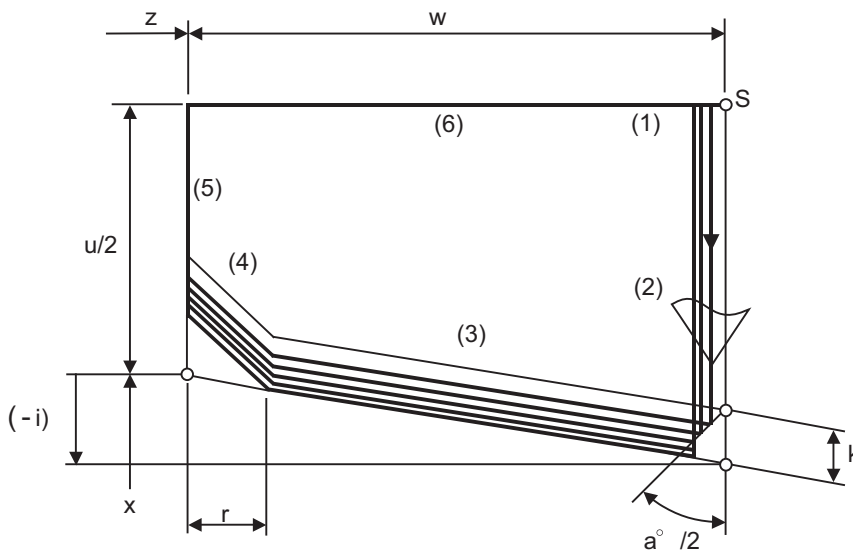
When Ri is positive



k : Thread height
 d : Finishing allowance (cut "m" times)
 (1) to (n) : 1st cutting to nth cutting

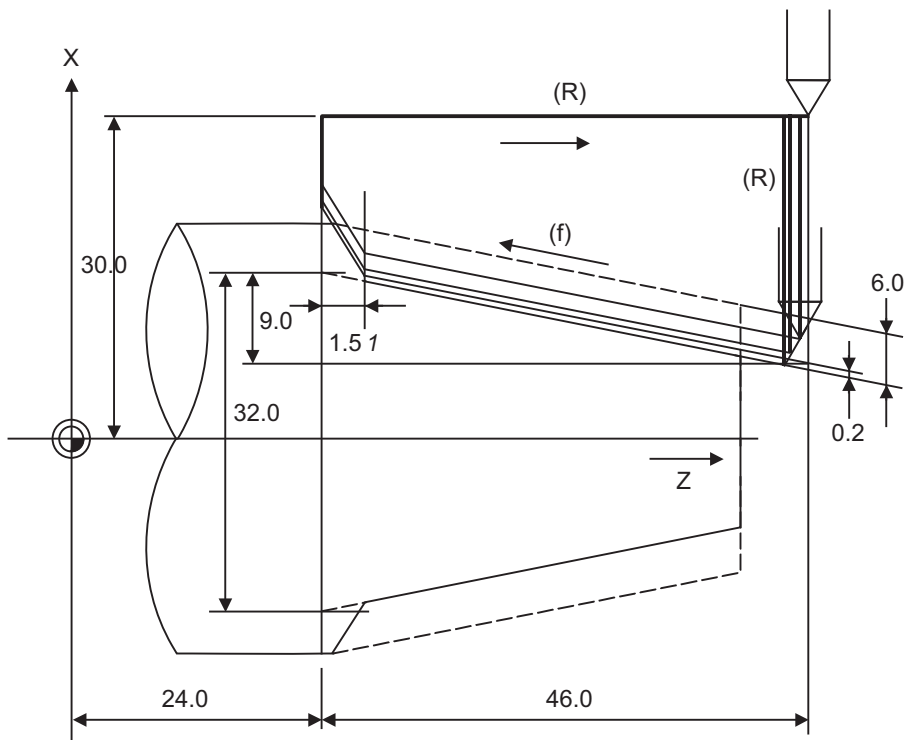
Interrupt operation

- (1) When the feed hold button is pressed during thread cutting, an automatic operation will stop upon completion of a block without thread cutting. (The automatic operation pause lamp turns on immediately and it goes off when automatic operation stops.)
 If feed hold is applied when thread cutting is not executed, or when the thread cutting command is issued but the axis is yet to move, the automatic operation pause lamp will turn on, and the automatic operation will pause.
- (2) The tool stops upon completion of operations (1), (4) and (5) in the figure, if the following operations are conducted during the G76 command execution.
 - When the automatic operation mode is switched to another automatic operation mode
 - When automatic operation is changed to manual operation
 - When single block operation is conducted



- (3) The dry run valid/invalid status does not change when G76 is being executed.

Program example



(R) Rapid traverse

(f) Cutting feed

G76 P011560 R0.2 ;

G76 U-28.0 W-46.0 R-9.0 P6.0 F4.0 ;

Precautions

Refer to "Precautions for Compound Type Fixed Cycle for Turning Machining (G70 to G76)"

13.2.8 Precautions for Compound Type Fixed Cycle for Turning Machining; G70 to G76



Precautions

- (1) Command all required parameters in a compound type fixed cycle for turning machining command block.
- (2) Provided that the finished shape program is registered in the memory, compound type fixed cycle for turning machining I commands can be executed in the memory or MDI.
- (3) When executing G70 to G73 command, ensure that the sequence No. of the finished shape program which is designated with P and Q is not duplicated in that program.
- (4) The finished shape program specified by P and Q in the G71 to G73 blocks should be prepared so that all the commands including those for corner chamfering, corner rounding and the automatic insertion blocks based on tool nose radius compensation, are created within 50 blocks. If this number is exceeded, program error (P202) will occur.
- (5) The finished shape program which is designated by the G71 to G73 blocks should be a program with monotonous changes (increases only or reductions only) for both the X and Z axes.
- (6) Blocks without movement in the finished shape program are ignored.
- (7) N, F, S, M, and T commands in the finished shape program are ignored.
- (8) When any of the following commands exists in a finished shape program, program error (P201) will occur.
 - (a) Commands related to reference position return (G27, G28, G29, G30)
 - (b) Thread cutting (G33)
 - (c) Fixed cycles
 - (d) Skip functions (G31, G37)
- (9) If subprogram call or macro call command exists in the finished shape program, these commands will also be executed.
- (10) Except for thread cutting cycles, operation stops at the end (start) point of each block in the single block mode.
- (11) Note that, depending on whether the sequence No. or program No. is designated, the next block after the completion of the G71, G72 or G73 command will differ.

(a) When a sequence No. is designated	(b) When a program No. is designated
<p>The next block is the one designated by Q.</p> <pre style="margin-left: 40px;"> } N100 G71P200 Q500 U_W_... ; N200 N300 } N400 } N500 } N600 </pre> <p>Operation moves to the N600 block upon completion of the cycle.</p>	<p>The next block is the next block of the cycle command.</p> <pre style="margin-left: 40px;"> } N100 G71A100 U_W_... ; N200 N300 N400 </pre> <div style="border: 1px solid black; padding: 5px; margin-left: 100px; width: fit-content;"> <p>O100</p> <p>N10X100.Z50. ;</p> <p>N20 :</p> <p> :</p> <p> :</p> </div> <p>Operation moves to the N200 block upon completion of the cycle.</p>

- (12) The next block after the completion of the G70 command is the next block of the command block.

```
:  
N100 .... ;  
N200 .... ;  
N300 .... ;  
N400 .... ;  
N500 .... ;  
:  
N1000 G70 P200 Q500 ; (or G70A100 ; )  
N1100 .... ;  
:
```

Operation moves to the N1100 block upon completion of the G70 command.

- (13) It is possible to apply a manual interruption while a compound type fixed cycle for turning machining command (G70 to G76) is being executed. However, upon completion of the interrupt, the tool must first be returned to the position where the interrupt was applied and then the compound type fixed cycle for turning machining must be restarted.
If it is restarted without the tool having been returned, all subsequent movements will deviate by an amount equivalent to the manual interruption amount.
- (14) Compound type fixed cycle for turning machining commands are unmodal. So they must be issued every time they are required.
- (15) Program error (P203) will occur with the G71 and G72 commands when there is no further movement of the Z axis in the second block or the Z axis has moved in the opposite direction because of tool nose radius compensation.

13.3 Fixed Cycle for Drilling



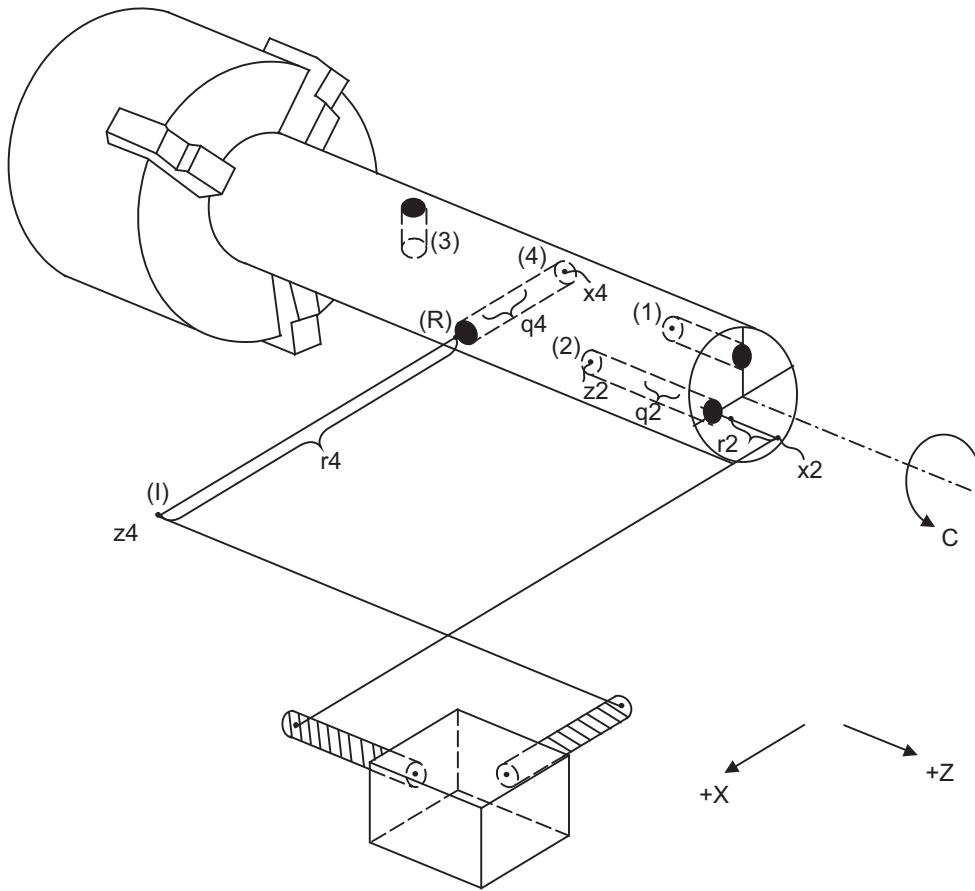
Function and purpose

These fixed cycles are used to perform prepared working sequences of machining programs such as positioning, hole drilling, boring and tapping in a block. When performing a same machining repeatedly, it can be executed by commanding only the axis position. The types of fixed cycles are listed below.

G code	Hole drilling axis	Drilling Hole drilling start	Operation at hole bottom	Return operation	Application
G80	-	-	-	-	Cancel
G83	Z	Cutting feed Intermittent feed	In-position check Dwell	Rapid traverse	Face Deep Hole Drilling Cycle I
G84	Z	Cutting feed	In-position check Dwell Spindle reverse rotation	Cutting feed	Face Tapping Cycle
G85	Z	Cutting feed	In-position check Dwell	Cutting feed	Face Boring Cycle
G87	X	Cutting feed Intermittent feed	In-position check Dwell	Rapid traverse	Deep Hole Drilling Cycle I
G88	X	Cutting feed	In-position check Dwell Spindle reverse rotation	Cutting feed	Longitudinal tapping cycle
G89	X	Cutting feed	In-position check Dwell	Cutting feed	Longitudinal boring cycle
G83.2	Z/X	Cutting feed Intermittent feed	In-position check Dwell	Rapid traverse	Deep hole drilling cycle 2

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

The hole drilling axes and the positioning for the fixed cycle for drilling are shown in the outline drawing below.



(I) Initial point

(R) R point

- (1) G83 Xx1 Cc1 Zz1 Rr1 Qq1 Pp1 Ff1 Kk1 ; Face deep hole drilling cycle
- (2) G83 Xx2 Cc2 Zz2 Rr2 Qq2 Pp2 Ff2 Kk2 ;
- (3) G87 Zz3 Cc3 Xx3 Rr3 Qq3 Pp3 Ff3 Kk3 ; Longitudinal deep hole drilling cycle
- (4) G87 Zz4 Cc4 Xx4 Rr4 Qq4 Pp4 Ff4 Kk4 ;

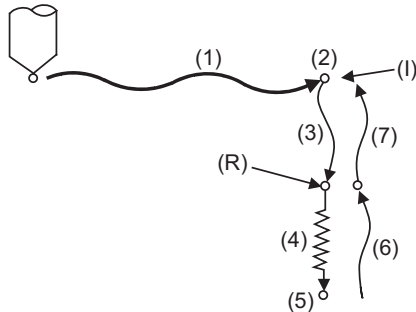
During the hole drilling cycle, the C axis (spindle) is clamped so that it does not move.
 Commands M03, M04 and M05 (forward rotation, reverse rotation and stop) operate to the rotary tools.



Detailed description

Basic operations of fixed cycle for drilling

The actual operation consists of the following seven movements.

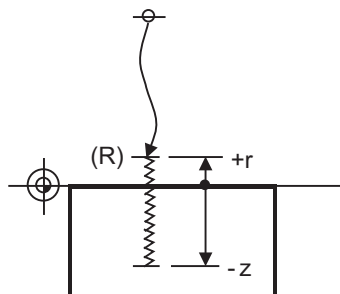


(I) Initial point (R) R point

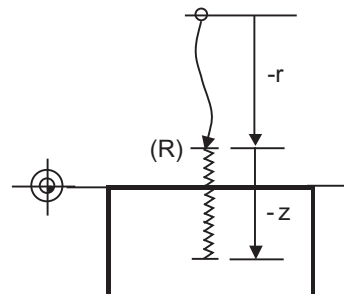
- (1) This denotes the positioning (by rapid traverse) to the X (Z) and C axis initial point.
If "I" (Positioning axis in-position width) is designated, in-position check will be carried out after the block is completed.
 - (2) This will be output if the M code for C-axis clamping is issued.
 - (3) This denotes the positioning (by rapid traverse) to the R point.
 - (4) Hole machining is conducted by cutting feed.
If "J" (Hole drilling axis in-position width) is designated, in-position check will be carried out after the block completes. Note that in case of deep drilling cycle 1 or 2, in-position check will not be carried out for hole drillings in the program. It will be carried out at designated hole bottom position (the last hole drilling).
 - (5) This operation takes place at the hole bottom position and it differs according to the fixed cycle modes including the rotary tool reverse rotation (M04), rotary tool forward rotation (M03) and dwell. .
 - (6) This denotes the returning to the R point.
 - (7) This denotes the returning to the initial point by rapid traverse.
(Operations 6 and 7 may become a single operation depending on the fixed cycle mode.)
- (Note 1) Whether the fixed cycle is to be completed at operation 6 or 7 can be selected by G98/G99 commands. (Refer to "Initial point and R point level return; G98, G99")

Difference between absolute value command and incremental value command

For absolute value



For incremental value



(R) R point

13.3.1 Face Deep Hole Drilling Cycle 1 (Longitudinal deep hole drilling cycle 1) ; G83 (G87)



Command format

G83(G87) X/U(Z/W)___ C/H___ Z/W(X/U)___ Rr Qq Pp Ff Kk Mm ;

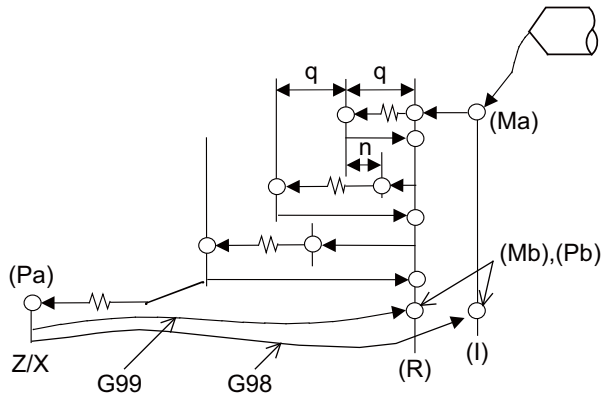
G83(G87)	G83 Face deep hole drilling cycle 1 mode G87 Longitudinal deep hole drilling cycle 1 mode
X/U(Z/W) C/H	Designation of hole position initial point (absolute/incremental value) ... Data for positioning X(Z) and C axes
Z/W(X/U)	Designation of hole bottom position (absolute/incremental value from R point)
Rr	Designation of R point (incremental value from initial point) (sign ignored)
Qq	Designation of cut amount for each cutting passs with G83 (G87). Always incremental value, radius value (sign ignored)
Pp	Designation of dwell time at hole bottom point. Relationship between time and designated value is same as for G04 designation.
Ff	Designation of feed rate for cutting feed
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Mm	Designation of miscellaneous command

- (Note 1) For the longitudinal deep hole drilling cycle 1(G87), designate Z/W to the hole position initial point and X/U to the hole bottom position.
- (Note 2) The designation of the hole position initial point is unmodal. When G83(G87) command is to be executed continuously, designate them block by block.
- (Note 3) Q command is unmodal. Designate them block by block.
- (Note 4) K command is unmodal. When K command is not issued, it is regarded as K1. When K0 is designated, the hole machining data are stored in the memory but no holes will be machined.



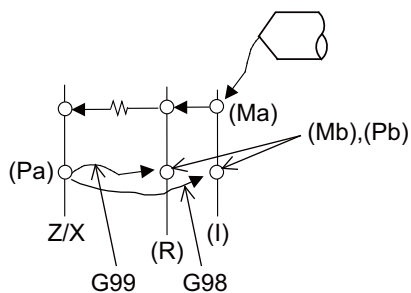
Detailed description

When the Q command is issued (deep hole drilling)



- (1) Return amount "d" is set by the parameter (#8013 G83 n). The tool returns at rapid traverse.
- (2) (Ma) ... The M code (Mm) is output when there is a C-axis clamping M code command (Mm).
- (3) (Mb) ... The C-axis unclamping M code (C-axis clamp M code + 1 = Mm + 1) is output when there is a C-axis clamping M code command (Mm).
- (4) (Pa) ... Dwell is performed for the duration equivalent to the time designated by P.
- (5)(Pb) ' ... After the C-axis unclamping M code (Mm + 2) is output, dwell is performed for the duration equivalent to the time set by the parameter (#1184 clmp_D).

When the Q command is not present (drilling)



G83 (G87) X(z) C Z(x) R r Pp Ff Kk Mm ;

See "When the Q command is present (deep hole drilling)" for details on (Ma),(Mb),(Pa),(Pb).



Precautions

Refer to "Precautions When Using a Fixed Cycle for Drilling".

13.3.2 Face Tapping Cycle (Longitudinal tapping cycle) ; G84 (G88)



Command format

G84 X/U__ C/H__ Z/W__ Rr1 Pp Ff Kk Ss1 ,Ss2 ,Rr2 Mm ; ... Face tapping cycle

G84	G84 Face tapping cycle mode
X/U C/H	Designation of hole position initial point (absolute/incremental value) ... Data for positioning X and C axes
Z/W	Designation of hole bottom position (absolute/incremental value from R point)
Rr1	Designation of R point (incremental value from initial point) (sign ignored)
Pp	Designation of dwell time at hole bottom point. Relationship between time and designated value is same as for G04 designation.
Ff	Synchronous tapping mode: Designation of drilling axis feed amount (tapping pitch) per spindle rotation Asynchronous tapping mode: Designation of feedrate for cutting feed
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Ss1	Designation of spindle rotation speed
,Ss2	Designation of spindle rotation speed at return (Valid only during synchronous tapping mode. This setting is ignored during other modes.)
,Rr2	Synchronization method selection (r2 = 1: synchronous, r2 = 0: asynchronous) (When omitted, the mode will follow the setting of parameter "#1229/bit4 Synchronous tap".)
Mm	Designation of miscellaneous command

G88 Z/W__ C/H__ X/U__ Rr1 Pp Ff Kk Ss1 ,Ss2 ,Rr2 Mm ; ... Longitudinal tapping cycle

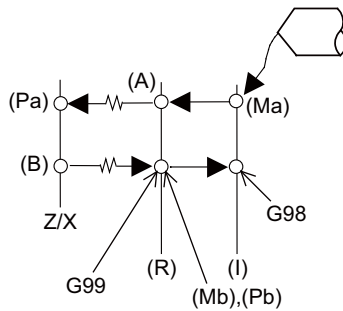
G88	G88 Longitudinal tapping cycle mode
Z/W C/H	Designation of hole bottom position (absolute/incremental value from R point) ... Data for positioning Z and C axes
X/U	Designation of hole bottom position (absolute/incremental value from R point)
Rr1 to Mm	Others are same as face tapping cycle. Designation of R point (incremental value from initial point) (sign ignored)

(Note 1) The designation of the hole position initial point is unmodal. When tapping cycle command is to be executed continuously, designate them block by block.

(Note 2) K command is unmodal. When K command is not issued, it is regarded as K1. When K0 is designated, the hole machining data are stored in the memory but no holes will be machined.



Detailed description



- (1) See "Face Deep Hole Drilling Cycle 1; G83" for details on (Ma),(Mb),(Pa),(Pb).
- (2) When G84(G88) is executed, the override will be canceled and the override will automatically be set to 100%.
- (3) 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(G88) execution, the block stops after returning operation is completed.
- (4) During single block operation, the axis will not stop at the turning point of tapping cycle.
- (5) During the G84 (G88) modal, the "Tapping" NC output signal will be output.
- (6) During the G84 (G88) synchronous tapping modal, the M3, M4, S code, etc. will not be output.
- (7) The tool stops at the R point and the rotary tool forward rotation signal is output. (A)
- (8) The rotation of the rotary tool is reversed at the hole bottom and tapping is performed. (B)
- (9) The fixed cycle subprograms should be edited if the rotary tool stop (M05) command is required before the rotary tool reverse (M04) or forward rotation (M03) signal is output.
- (10) Specify start position for synchronous tapping by "#3106 zrn_typ/bit4".
 - 0: Zero point return 1: Deceleration stop (the position where the synchronous tap is commanded)
 Note that the axis will not return to the zero point during a tapping return even if "#3106 zrn_typ/bit4" is set to "0".

Synchronous/asynchronous tap selection

- (1) Selecting with a program command

Tap cycle ",R0/1" command

G84 (G88) Xx1 Cc1 Zz1 Rr1 Pp1 Ff1 Kk1 Dd1 Ss1 ,Ss2 ,Rr2 Mm1 ;

When r2 = 1, the synchronous tapping mode will be applied, and when r2 = 0, the asynchronous tapping mode will be applied.

- (2) Selecting with parameters

Basic specification parameters

#	Item		Details	Setting range
1229	set01	bit4	0: Handles the tapping cycles as the tapping cycles with a floating tap chuck. (asynchronous tapping) 1: Handles the tapping cycles as the tapping cycles without a floating tap chuck.(Synchronous tapping)	0/1

The tapping command will be synchronous tapping cycle when this parameter is turned on.

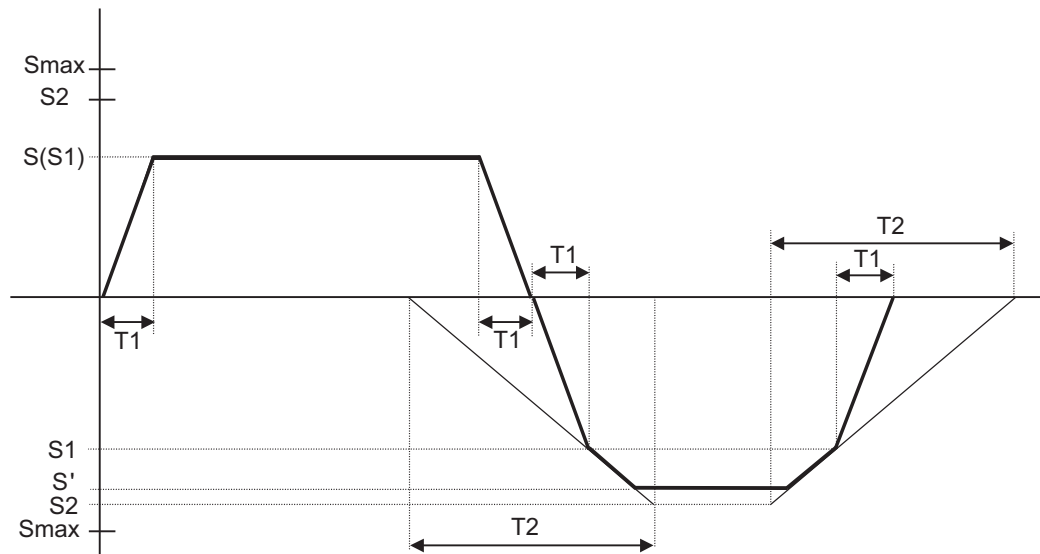
Canceling synchronous tapping selection

To cancel the synchronous tapping selection, command reset, G80 (fixed cycle for drilling cancel), 01 group G code or other fixed cycle G codes.

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 tapping rotation speed < spindle rotation speed during return \leq 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)

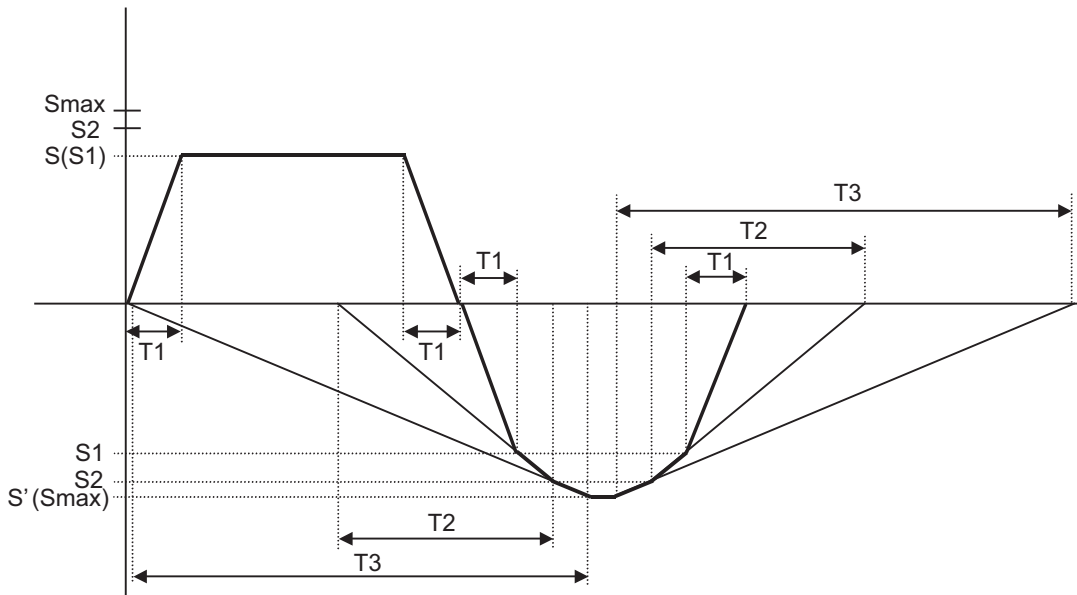
S2 : Synchronous tapping changeover spindle rotation speed 2 (spindle basic specification parameters #3037 to #3040)

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)



Precautions

Refer to "Precautions When Using a Fixed Cycle for Drilling".

13.3.3 Face Boring Cycle (Longitudinal boring cycle) ; G85 (G89)



Command format

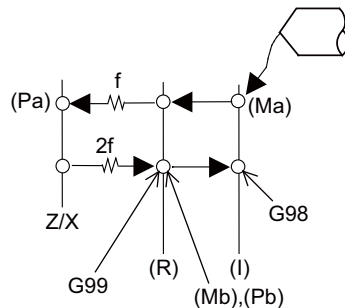
G85 (G89) X/U(Z/W)___ C/H___ Z/W(X/U)___ Rr Pp Ff Kk Mm ;

G85(G89)	G85 Face boring cycle mode G89 Longitudinal boring cycle mode
X/U(Z/W) C/H	Designation of hole position initial point (absolute/incremental value)
Z/W(X/U)	Designation of hole bottom position (absolute/incremental value from R point)
Rr	Designation of R point (incremental value from initial point) (sign ignored)
Pp	Designation of dwell time at hole bottom point. Relationship between time and designated value is same as for G04 designation.
Ff	Designation of feedrate for cutting feed
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Mm	Designation of miscellaneous command

- (Note 1) For the longitudinal boring cycle (G89), designate Z/W to the hole position initial point and X/U to the hole bottom position.
- (Note 2) The designation of the hole position initial point is unmodal. When G85(G89) command is to be executed continuously, designate them block by block.
- (Note 3) Q command is unmodal. Designate them block by block.
- (Note 4) K command is unmodal. When K command is not issued, it is regarded as K1. When K0 is designated, the hole machining data are stored in the memory but no holes will be machined.



Detailed description



- (1) See "Face Deep Hole Drilling Cycle 1; G83" for details on (Ma),(Mb),(Pa),(Pb).
- (2) The tool returns to the R point at a cutting feed rate which is double the designated feed rate command. However, it does not exceed the maximum cutting feed rate.



Precautions

Refer to "Precautions When Using a Fixed Cycle for Drilling".

13.3.4 Deep Hole Drilling Cycle 2 ; G83.2



Function and purpose

The deep hole drilling cycle 2 drills deep holes in the X-axis or Z-axis direction by commanding the X or Z coordinate of the end point and the cut amount at cutting feed.



Command format

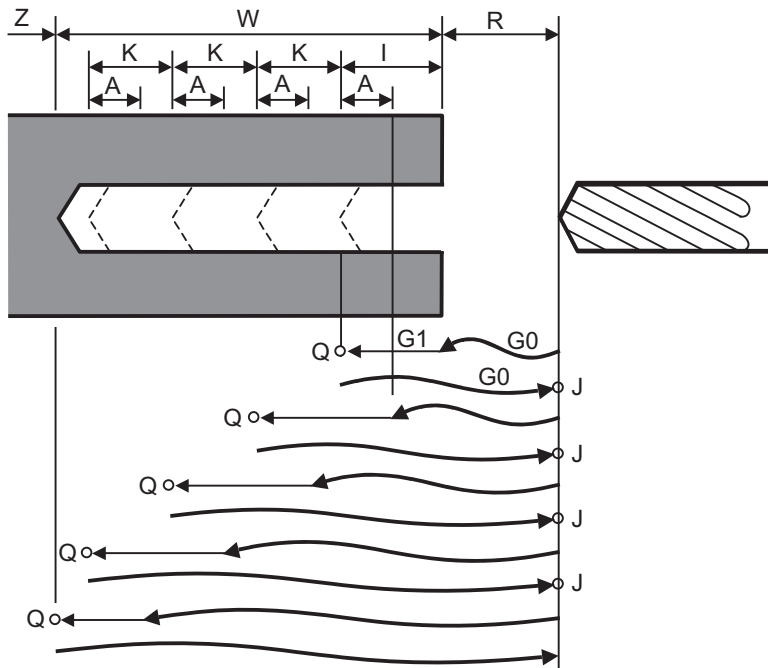
G83.2 W/Z/U/X__ R__ I__ K__ A__ Q__ J__ F__ ; ... Deep hole drilling cycle 2

W/Z/U/X	Incremental value from hole drilling start point/coordinates of hole bottom (sign valid)
R	Incremental value from present position up to hole drilling start point (sign ignored) (always radius value with incremental value.)
I	Cut amount of first cutting pass (sign ignored) (always radius value with incremental value.)
K	Cut amount of second and subsequent cutting passes (sign ignored) (always radius value with incremental value.)
A	Drill stop safety distance for second and subsequent cutting passes (sign ignored) (always radial value with incremental value.)
Q	Dwell time at cut point (sign ignored, decimal point invalid)
J	Dwell time at return point (sign ignored, decimal point invalid)
F	Cutting feedrate

(Note 1) When A command is not issued, the parameter "#8013 G83 n" setting value is used.

(Note 2) If the cut amount for either the first cutting pass (address "I") or the second and subsequent passes (address "K") is not designated (including a command value = 0), the one designated will be used as I = K= command value.

If both commands are not designated, hole drilling will be performed down to the hole bottom at once.



Q : Dwell time at cut point

J : Dwell time at return point

With single block operation, block stops upon completion of the deep hole drilling cycle 2 commands.



Detailed description

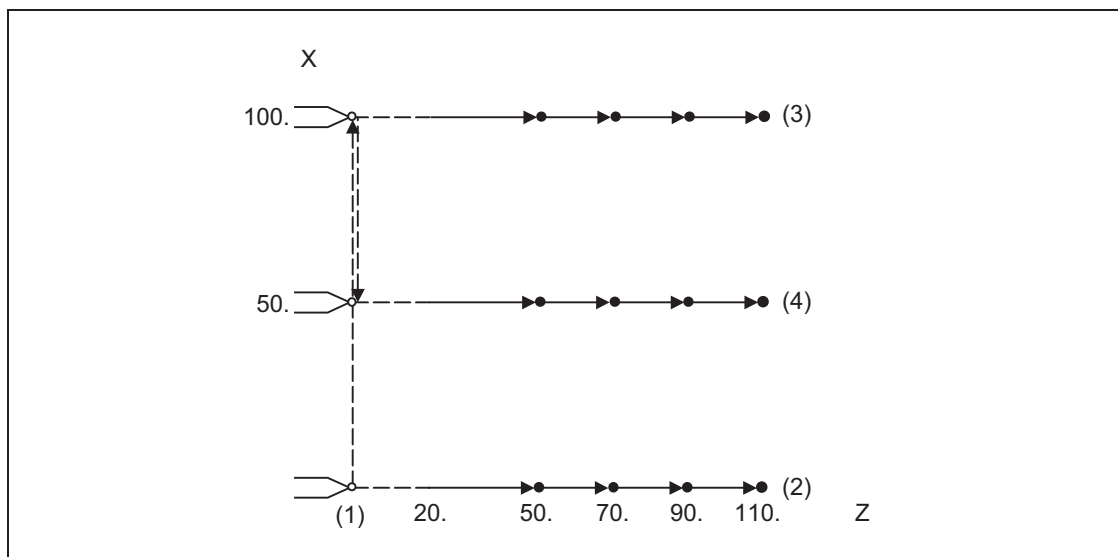
- (1) When the axis address of the hole drilling axes is commanded several times in a block, the last address will be valid.
- (2) A program error (P33) will occur in the following commands.
 - (a) When both hole drilling axis X (command address X or U) and hole drilling axis Z (command address Z or W) are commanded.
 - (b) When an axis other than X or Z (command addresses except X, U, Z and W) is commanded.
- (3) When the feed hold button is pressed while the deep hole drilling cycle 2 is being executed, the automatic operation stops and the remainder of the cycle will be executed when automatic operation is restarted.
- (4) When an interruption is made by manual operation while the automatic operation is stopped (manual ABS switch ON), the operation in the deep hole drilling cycle 2 modal is shifted from the automatic operation restart by an amount equivalent to the movement caused by the interrupt.



Program example

(when deep hole drilling cycle 2 is used as a modal command)

G28 XZ;	
G0 X0. Z0.;	...(1)
G83.2 Z110.R20.I30.K20.A5.Q1000 J500 F300. ;	...(2)
X100.;	...(3)
X50.;	...(4)
M02 ;	



Precautions

Refer to "Precautions When Using a Fixed Cycle for Drilling".

13.3.5 Fixed Cycle for Drilling Cancel; G80



Detailed description

This cancels the fixed cycle for drilling. The hole machining mode and hole machining data are both canceled.

13.3.6 Precautions When Using a Fixed Cycle for Drilling



Precautions

- (1) When G84 or G88 fixed cycle is commanded, the rotary tool must be rotated to the specified direction beforehand by miscellaneous functions (M3, M4).
- (2) If there is data for the basic axis, additional axis or R in the block during the fixed cycle mode, the hole drilling operation will be executed. If there is not data, the hole drilling operation will not be executed. Note that even with X axis data, if that block is a dwell (G04) time command, the hole drilling will not be executed.
- (3) Command hole machining data (Q, P (A,I,K,Q,J for G83.2)) in the block in which hole drilling is carried out (block containing data for the basic axis, additional axis or R).
The modal data is not updated if these data are commanded in blocks without hole drilling operation.

- (4) The F modal may change when a reset is applied during execution of G85 (G89).

- (5) The fixed cycle for drilling can also be canceled by 01 group G codes in addition to G80. If these are commanded in the same block as the fixed cycle, the fixed cycle will be ignored.

m = 01 group code n = fixed cycle for drilling code

Gm Gn X C Z R Q P K F ;

Gm: Executed

Gn: Ignored

X C Z: Executed

R Q P K: Ignored

F: Memorized

(Example)

G01 G83 X100. C30. Z50. R-10. Q10. P1 F100. ;

G83 G01 X100. C30. Z50. R-10. Q10. P1 F100. ;

In both cases, G01 X100. C30. Z50. F100 is executed.

- (6) When miscellaneous functions are commanded in the same block as fixed cycle commands, they are output simultaneously with the initial positioning.
Note that when the M code of the C axis clamp set in the parameters (#1183 clmp_M) is commanded in the same block, the M code will be output after positioning.
After returning to the return point (G98 mode: initial point/G99 mode: R point) after hole drilling, the M code of the C axis unclamp (clamp M + 1) will be output, and the axis dwells for the time set in the parameter (#1184 clmp_D).
When the number of rotations is designated, the above control will be carried out only for the first rotation, except for M codes of the C axis clamp. C axis clamp/unclamp M commands work in modal, and are output at every rotation until canceled by the fixed cycle cancel command.
- (7) When tool length offset command (T function) is issued in the fixed cycle for drilling, they are executed according to the tool length offset function.
- (8) A program error (P155) will occur when a fixed cycle for drilling is commanded during tool nose R compensation.
- (9) For the G code list 1, the initial point level return is fixed. The return level cannot be changed with a G98/G99 command. Note that a deferent function will be executed if G98/G99 is commanded.
- (10) As shown below, in a block where the movement direction of either axis reverses, 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. ;

13.3.7 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	G90 G83 X100. Z-50. R25. F1000 ;	<p>Initial level return is executed.</p>	<p>R point level return is executed.</p>
Two or more executions	G90 G83 X100. Z-50. R25. L5 F1000 ;	<p>Initial level return is executed for all times.</p>	

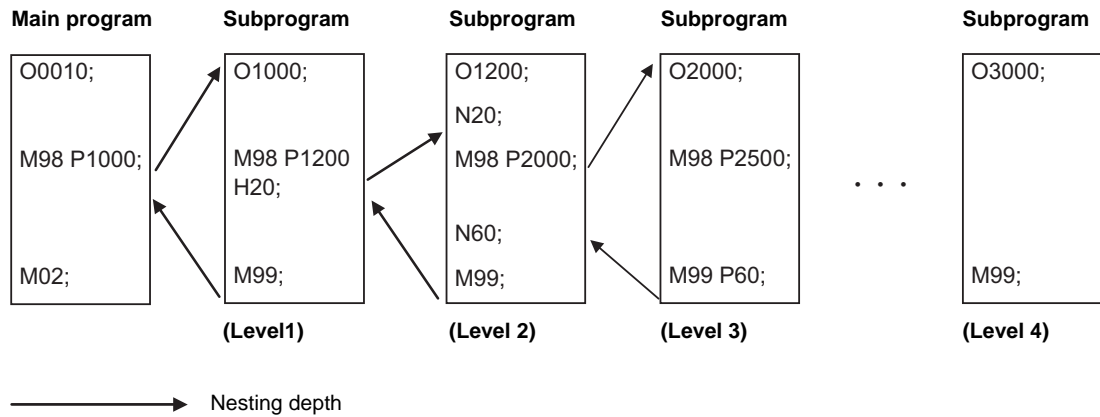
13.4 Subprogram Control; M98, M99

13.4.1 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 editing functions, subprogram control functions and fixed cycle functions.

	Case 1	Case 2	Case 3	Case 4
1. Subprogram control	No	Yes	Yes	No
2. Fixed cycles	No	No	Yes	Yes
Function				
1. Memory mode	○	○	○	○
2. Subprogram call	×	○	○	×
3. Subprogram variable designation (Note 2)	×	○	○	×
4. Subprogram nesting level call (Note 3)	×	○	○	×
5. Fixed cycles	×	×	○	○
6. Editing subprogram for fixed cycle	×	×	○	○

(Note 1) ○ denotes available functions and × denotes unavailable functions.

(Note 2) Variables cannot be transferred with the M98 command but variable commands in subprograms can be used provided that the variable command option is available.

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



Command format

M98 P__ H__ L__ ; ... Subprogram call

P	Program No. of subprogram to be called (own program if omitted) Note that P can be omitted only during memory mode and MDI mode. (Max. 8 digits)
H	Sequence No. in subprogram to be called (head block if omitted) (Max. 5 digits)
L	Number of subprogram repetitions (When omitted, this is interpreted as L1, and is not executed when L0.) (1 to 9999 times depending on the 4-digit value) For instance, For instance, M98 P1 L3 ; is equivalent to the following: M98 P1 ; M98 P1 ; M98 P1 ;

M99 P__ H__ Q__ R__ L__ ; ... Return to main program from subprogra

P	Sequence number of return destination (return to the block that follows the calling block if omitted)
H	Program number of return destination (return to the main program at calling if omitted)
Q	Sequence number to start searching of return destination (the block that follows the calling block will be handled as the search start position if omitted)
R	Sequence number to finish searching of return destination (the block that precedes the calling block will be handled as the search finish position if omitted)
L	Number of times after repetition number has been changed ("-1" if omitted)



Detailed description

Creating and registering subprograms

Subprograms have the same format as machining programs for normal memory mode, except that the subprogram completion instruction M99 (P_) ; must be registered as an independent block in the last block.

O***** ; : ; : ;	Program No. as subprogram No.
..... : ; : ;	Main body of subprogram
M99 ;	Subprogram return command
%(EOR)	Registration completion code

- (1) The above program is registered by editing operations at the setting and display unit. For further details, refer to the section on "program editing" in the Instruction Manual.
- (2) Only those subprogram Nos. ranging from 1 to 99999999 designated by the optional specifications can be used.
- (3) Main 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.)

Registration example

; O O O O O ; : : M99 ; % ;	Subprogram A
O Δ Δ Δ Δ ; : : M99 ; % ;	Subprogram B
O**** ; : : M99 ; % ;	Subprogram C

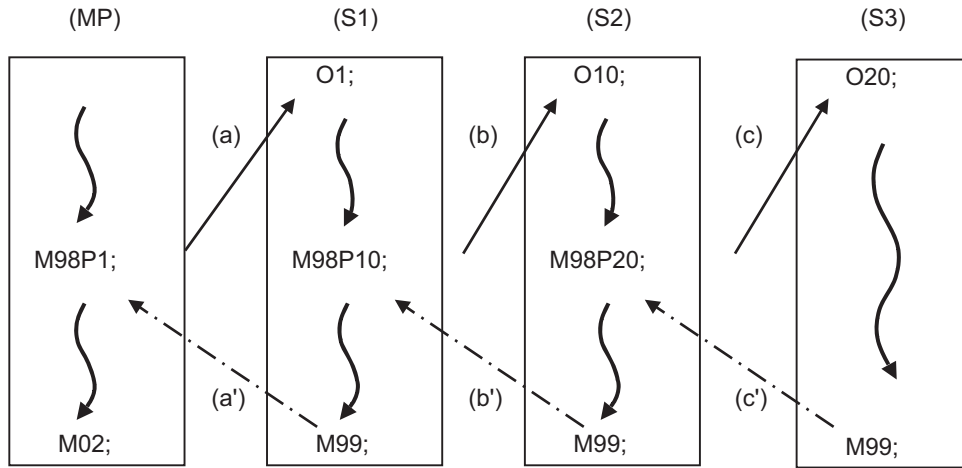
- (4) Main programs can be executed during memory or MDI mode but subprograms must be in the memory mode.
- (5) Besides the M98 command, subprogram nesting is subject to the following commands:
 - G65 : Macro call
 - G66 : Modal call
 - G66.1 : Modal call
 - G code call
 - Miscellaneous function call
 - 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 I1 times by programming M98 Pp1 L1;



Program example

Program example 1

When there are 3 subprogram calls (known as 3 nesting levels)



(M) Main program

(S1) Subprogram 1

(S2) Subprogram 2

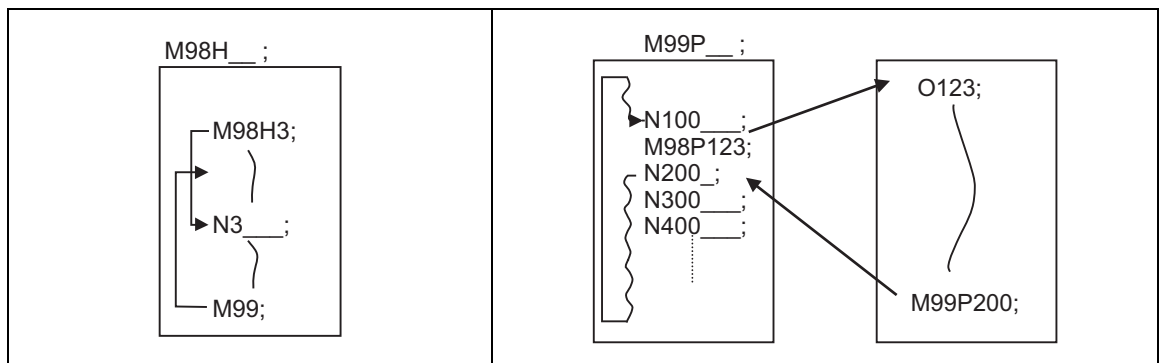
(S3) Subprogram 3

Sequence of execution : (a)-(b)-(c)-(c')-(b')-(a')

- (1) For nesting, the M98 and M99 commands should always be paired off on a 1:1 basis; (a)' for (a), (b)' for (b), etc.
- (2) Modal information is rewritten in the order of execution sequence without distinction between main programs and subprograms. Therefore, after calling a subprogram, attention must be paid to the modal data status when programming.

Program example 2

The M98 H_ ; M99 P_ ; commands designate the sequence Nos. in a program with a call instruction.





Precautions

- (1) Program error (P232) will occur when the designated P (program No.) cannot be found.
- (2) The M98 P_ ; M99 ; block does not perform a single block stop. If any address except O, N, P, L or H is used, single block stop can be executed. (With "X100. M98 P100 ;", the operation branches to O100 after X100. is executed.)
- (3) When M99 is commanded by the main program, operation returns to the head. (This is same for MDI.)
- (4) Note that it takes time to search when the sequence No. is designated by M99 P_ ;.

13.5 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.

Variable set option		No.		Function
Common variables		Variables common to all part systems	Variables for each part system	- Can be used in common throughout main, sub and macro programs.
1st part system	100 sets	500 to 549	100 to 149	
	200 sets	500 to 599	100 to 199	
	300 sets	500 to 699	100 to 199	
	600 sets	500 to 999	100 to 199	
Multi-part system (n = number of part systems)	50 + 50 sets	500 to 549	100 to 149 * n	
	100 + 100 sets	500 to 599	100 to 199 * n	
	200 + 100 sets	500 to 699	100 to 199 * n	
	500 + 100 sets	500 to 999	100 to 199 * n	
Local variables		1 to 33		
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) Variable names can be set for #500 to #519.

(Note 4) Variable names can be used for address "O" and "N".

Variable quotations

Variables can be used for all addresses except O, N and / (slash).

- (1) When the variable value is used directly:
X#1 Value of #1 is used as the X value.
- (2) When the complement of the variable value is used:
X-#2 Value with the #2 sign changed is used as the X value.
- (3) When defining variables:
#3 = #5 Variable #3 uses the equivalent value of variable #5.
#1 = 1000 Variable #1 uses the equivalent value 1000 (which is treated as 1000.).
- (4) When defining the variable arithmetic formula:
#1 = #3 + #2 - 100 Value of the operation result of #3 + #2 - 100. is used as the #1 value.
X [#1 + #3 + 1000] Value of the operation result of #1 + #3 + 1000 is used as the X value.

(Note 1) A variable cannot be defined in the same block as an address. It must be defined in a separate block.

Incorrect	Correct
X#1 = #3 + 100 ;	#1 = #3 + 100 ; X#1 ;

- (Note 2) Up to five sets of square parentheses [] may be used.
#543 = -[[[[[#120]/2+15.]*3-#100]/#520+#125+#128]*#130+#132]
- (Note 3) There are no restrictions on the number of characters and number of variables for variable definition.
- (Note 4) The variable values should be within the range of 0 to ±99999999.
If this range is exceeded, the arithmetic operations may not be conducted properly.
- (Note 5) The variable definitions become valid when definitions are made.
#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.

13.6 User Macro

13.6.1 User Macro

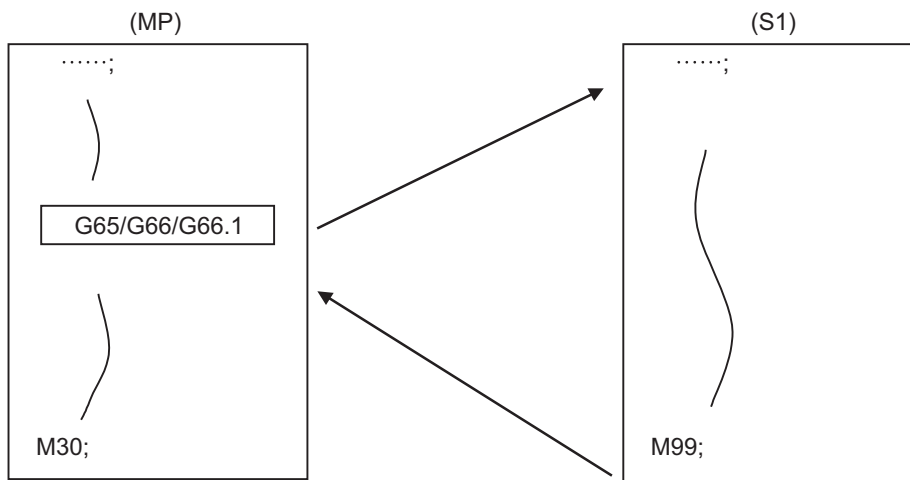


Function and purpose

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.

13.6.2 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.

13.6.2.1 Simple Macro Calls ; G65



Function and purpose



M99 is used to terminate the user macro subprogram.

(MP) Main program

(S1) Subprogram



Command format

G65 P__ L__ argument ; ... Simple macro calls

P	Program No.
L	Number of repetitions



Detailed description

When the argument must be transferred as a local variable to a user macro subprogram, the actual value should be designated after the address.

In this case, regardless of the address, a sign and decimal point can be used in the argument. There are 2 ways in which arguments are designated.

Argument designation I

Format : A_ B_ C_ ... X_ Y_ Z_

- (1) Arguments can be designated using any address except G, L, N, O and P.
- (2) I, J and K must be designated in alphabetical order.
 I_ J_ K_ Correct
 J_ I_ K_ Incorrect
- (3) Except for I, J and K, there is no need for designation in alphabetical order.
- (4) Addresses which do not need to be designated can be omitted.
- (5) The following table shows the correspondence between the addresses which can be designated by argument designation I and the variable numbers in the user macro main body.

Address and variable No. correspondence		Addresses available for call instructions	
Argument designation I address	Variable in macro	G65,G66	G66.1
A	#1	○	○
B	#2	○	○
C	#3	○	○
D	#7	○	○
E	#8	○	○
F	#9	○	○
G	#10	×	× *
H	#11	○	○
I	#4	○	○
J	#5	○	○
K	#6	○	○
L	#12	×	× *
M	#13	○	○
N	#14	×	× *
O	#15	×	×
P	#16	×	× *
Q	#17	○	○
R	#18	○	○
S	#19	○	○
T	#20	○	○
U	#21	○	○
V	#22	○	○
W	#23	○	○
X	#24	○	○
Y	#25	○	○
Z	#26	○	○

○ : Can be used
 × : Cannot be used
 * : Can be used while G66.1 command is modal

Argument designation II

Format : A__ B__ C__ I__ J__ K__ I__ J__ K__ ...

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

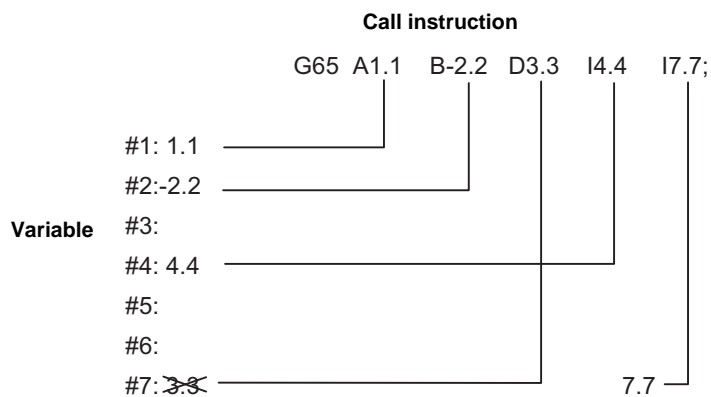
Argument specification II address	Variable in macro		Argument specification II address	Variable in macro
A	#1		J5	#17
B	#2		K5	#18
C	#3		I6	#19
I1	#4		J6	#20
J1	#5		K6	#21
K1	#6		I7	#22
I2	#7		J7	#23
J2	#8		K7	#24
K2	#9		I8	#25
I3	#10		J8	#26
J3	#11		K8	#27
K3	#12		I9	#28
I4	#13		J9	#29
J4	#14		K9	#30
K4	#15		I10	#31
I5	#16		J10	#32
			K10	#33

(Note 1) Subscripts 1 to 10 for I, J, and K indicate the order of the specified command sets. They are not required to specify instructions.

Using arguments designations I and II together

If addresses corresponding to the same variable are commanded when both types I and II are used to designate arguments, the latter address will become valid.

(Example 1)

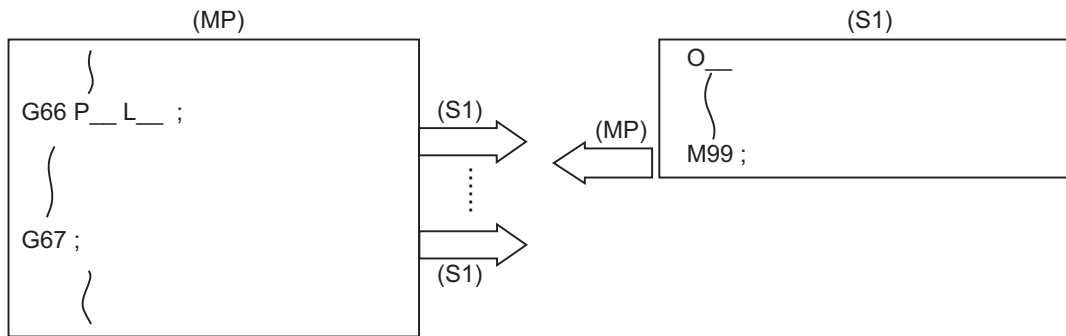


In the above example, I7.7 argument is valid when both arguments D3.3 and I7.7 are commanded for the #7 variable.

13.6.2.2 Modal Call A (Movement Command Call) ; G66



Function and purpose



(MP) Main program

(S1) Subprogram

When the block with a movement command is commanded between G66 and G67, the movement command is first executed and then the designated user macro subprogram is executed. A number of user macro subprograms are designated with "L".

The argument is the same as for a simple call.



Command format

G66 P__ L__ argument ; ... Macro modal call A

P	Program No.
L	Number of repetitions



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.

13.6.2.3 Modal Call B (for each block) ; G66.1



Function and purpose

The specified user macro subprogram is called unconditionally for each command block which is assigned between G66.1 and G67 and the subprogram will be repeated for the number of times specified in L. The argument is the same as for a simple call.



Command format

G66.1 P__ L__ argument ; ... Modal call B

P	Program No.
L	Number of repetitions



Detailed description

- (1) In the G66.1 mode, everything except the O, N and G codes in the various command blocks which are read are handled as the argument without being executed. Any G code designated last or any N code commanded after anything except O and N will function as the argument.
- (2) All significant blocks in the G66.1 mode are handled as when G65P__ is assigned at the head of a block. (Example 1)
 In "G66.1 P1000 ;" mode,
 N100 G01 G90 X100. Z100. F400 R1000 ; is same as
 N100 G65 P1000 G01 G90 X100. Z200. F400 R1000 ;
- (Note 1) The call is performed even in the G66.1 command block in the G66.1 mode and the correspondence between the argument address and the variable number is the same as for G65 (simple call).
- (3) The range of the G and N command values which can be used anew as variables in the G66.1 mode is subject to the restrictions as normal NC command values.
- (4) Program number O, sequence numbers N and modal G codes are updated as modal information.

13.6.2.4 G Code Macro Call



Function and purpose

User macro subprogram with prescribed program numbers can be called merely by issuing the G code command.



Command format

G argument ; ... G code macro call**

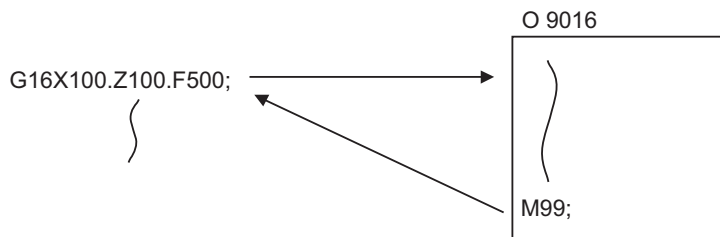
G**	G code for macro call
-----	-----------------------



Detailed description

- (1) The above instruction functions in the same way as the instructions below, however, the correspondence between M codes and instructions can be set by parameters.
 - a : M98 P △△△△ ;
 - b : G65 P △△△△ Argument ;
 - c : G66 P △△△△ Argument ;
 - d : G66.1 P △△△△ Argument ;

When the parameters corresponding to c and d above are set, issue the cancel command (G67) either in the user macro or after the call code has been commanded so as to cancel the modal call.
- (2) The correspondence between the "*" which conducts the macro call and the macro program number P △△△△ to be called is set by parameters.
- (3) Up to 10 G codes from G100 to G999 can be used with this instruction. (G00 to G99 can also be used with parameter "#1081 Gmac_P").
- (4) These commands cannot be issued during a user macro subprogram which has been called by a G code.



13.6.2.5 Miscellaneous Command Macro Call (for M, S, T, B Code Macro Call)



Function and purpose

The user macro subprogram of the specified program number can be called merely by issuing an M (or S, T, B) code. (Registered M code and all S, T and B codes.)



Command format

M** ; (or S** ;, T** ;, B** ;) ... Miscellaneous command macro call	
M**	M code for macro call (or S, T, B code)



Detailed description

- (1) The above instruction functions in the same way as the instructions below, however, the correspondence between M codes and instructions can be set by parameters. (Same for S, T and B codes)

a:M98 P**** ; b:G65 P**** M** ;	M98, M** are not output.
c:G66 P**** M** ;	
d:G66.1 P**** M** ;	

When the parameters corresponding to c and d above are set, issue the cancel command (G67) either in the user macro or after the call code has been commanded so as to cancel the modal call.

- (2) The correspondence between the "M**" which conducts the macro call and the macro program number P**** to be called is set by parameters. Up to 10 M codes from M00 to M95 can be registered. Note that the codes to be registered should exclude those basically required for the machine and M0, M1, M2, M30 and M96 to M99.
- (3) As with M98, it is displayed on the screen display of the setting and display unit but the M codes and MF are not output.
- (4) Even if the registered miscellaneous commands above are issued in a user macro subprogram which are called by an M code, it will not be regarded as a macro call and will be handled as a normal miscellaneous command. (Same for S, T and B codes)
- (5) All S, T and B codes call the subprograms in the prescribed program numbers of the corresponding S, T and B functions.
- (6) Up to 10 M codes can be set. However, if not using up 10 codes, set the parameters as shown below.

[MACRO FILE (1)]			
	<CODE>	<TYPE>	<PROGRAM-No.>
M[01]	20	0	8000
M[02]	21	0	8001
M[03]	9999	0	19999999
M[04]	9999	0	19999999
M[05]	9999	0	19999999
	:	:	:
	:	:	:
M[10]	9999	0	19999999

-----For M20 commands, set to call O8000 in type 0 (M98 type)

-----For M21 commands, set to call O8001 in type 0 (M98 type)

-----Set parameters that will not be used as shown at the left.

13.6.2.6 Detailed Description for Macro Call Instruction



Detailed description

Differences between M98 and G65 commands

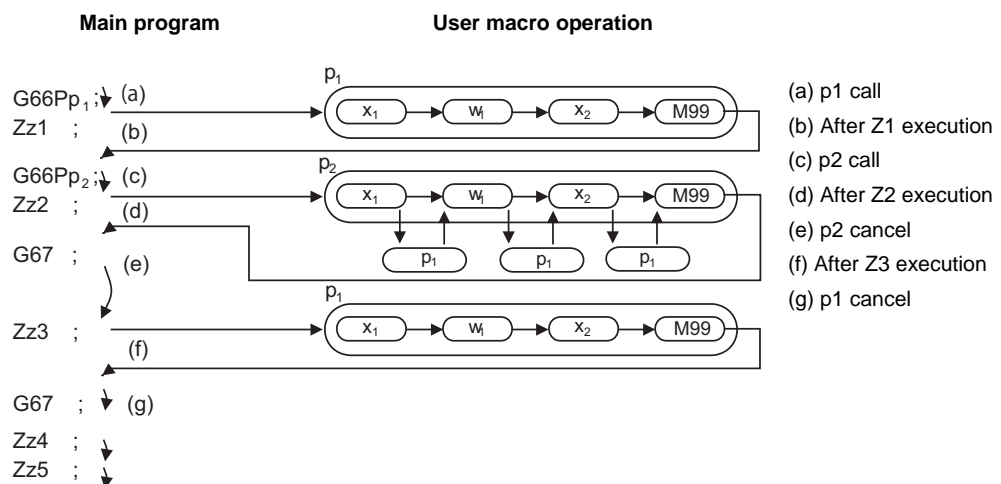
- (1) The argument can be designated for G65 but not for M98.
- (2) The sequence number can be designated for M98 but no for G65, G66 and G66.1.
- (3) M98 executes subprograms after all the commands except M, P, H and L in the M98 block are executed, but G65 branches directly to the subprogram without any further operation.
- (4) When any address except O, N, P, H or L is included in the M98 block, the single block stop will be conducted, but not for the G65.
- (5) The level of the M98 local variables is fixed but it varies in accordance with the nesting depth for G65. ("#1" before and after M98, for instance, has the same significance, but they have different significance in G65.)
- (6) The M98 nesting depth extends up to 8 levels in combination with G65, G66 and G66.1. The G65 nesting depth extends up to only 4 levels in combination with G66 and G66.1.

Macro call command nesting depth

Up to 4 nesting levels are available for macro subprogram calls by simple call or modal call. The argument for a macro call instruction is valid only within the called macro level. Since the nesting depth for macro calls extends up to 4 levels, the argument can be used as a local variable for the programs of each macro call of each level.

- (Note 1) When a G65, G66, G66.1 G code macro call or miscellaneous command macro call is conducted, this is regarded as a nesting level and the level of the local variables is also incremented by one.
- (Note 2) With modal call A, the designated user macro subprogram is called every time a movement command is executed. However, when the G66 command is duplicated, the next user macro subprogram is called to movement commands in the macro every time an axis is moved. User macro subprograms are called from the one commanded last.

(Example 1)



13.6.3 Variable



Function and purpose

Both the variable specifications and user macro specifications are required for the variables which are used with the user macros.

The compensation amounts of the local, common and system variables among the variables for this MELDAS NC system except #33 are retained even when the unit's power is switched off. (Common variables can also be cleared by parameter "#1129 PwrVC1".)



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 Z1000 ;Equivalent to G0 Z1000 ;
- G0 X#1 + 10 Z1000 ; Equivalent to G0 X10 Z1000 ;

(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.

13.6.4 Types of Variables

13.6.4.1 Common Variables



Detailed description

Common variables can be used commonly from any position. Number of the common variables sets depends on the specifications.

Refer to the explanation about Variable Commands for details.

Variable name setting and quotation

Any name (variable name) can be given to common variables #500 to #519. It must be composed of not more than 7 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] ;

[Common variables]		
#500	- 12345.678	A234567
#501	5670.000	DIST
#502	- 156.500	TOOL25
~~~~~		
#518	10.000	NUMBER
Common variable #(502) DATA (-156.5) NAME (TOOL25)		

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

### 13.6.4.2 Local Variables (#1 to #33)



#### Detailed description

Local variables can be defined as an <argument> when a macro subprogram is called, and also used locally within main programs and subprograms. They can be duplicated because there is no relationship between macros. (up to 4 levels)

**G65 P__ L__ <argument> ;**

P	Program No.
L	Number of repetitions

The <argument> is assumed to be Aa1 Bb1 Cc1..... Zz1.

The following table shows the correspondences between the addresses designated by <argument> and the local variable numbers used in the user macro main bodies.

#### [Argument designation I]

Call command		Argument address	Local variable No.	Call command		Argument address	Local variable No.
G65 G66	G66.1			G65 G66	G66.1		
○	○	A	#1	○	○	Q	#17
○	○	B	#2	○	○	R	#18
○	○	C	#3	○	○	S	#19
○	○	D	#7	○	○	T	#20
○	○	E	#8	○	○	U	#21
○	○	F	#9	○	○	V	#22
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.

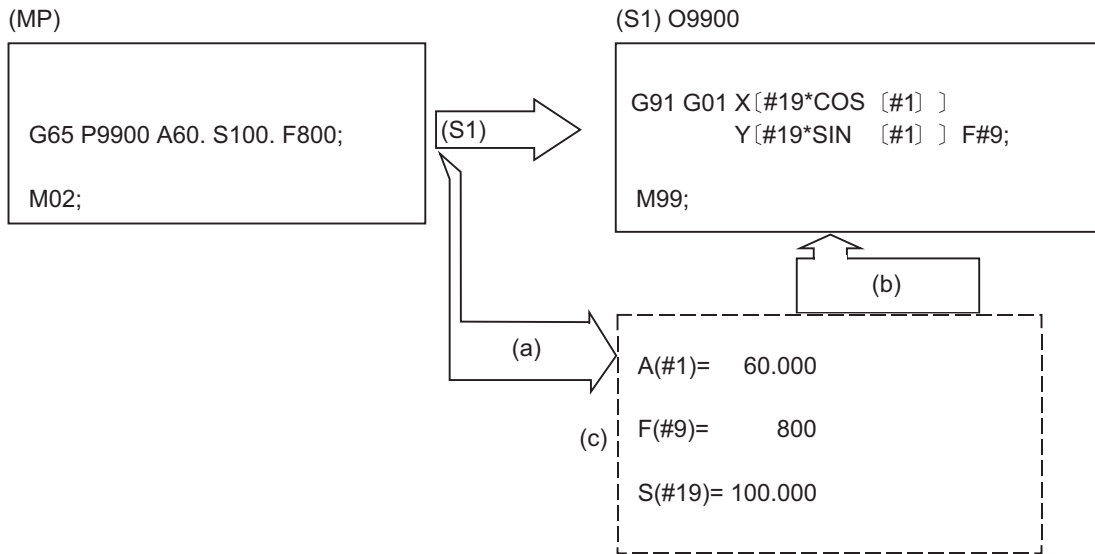
The hyphen (-) mark indicates that there is no corresponding address.

**[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

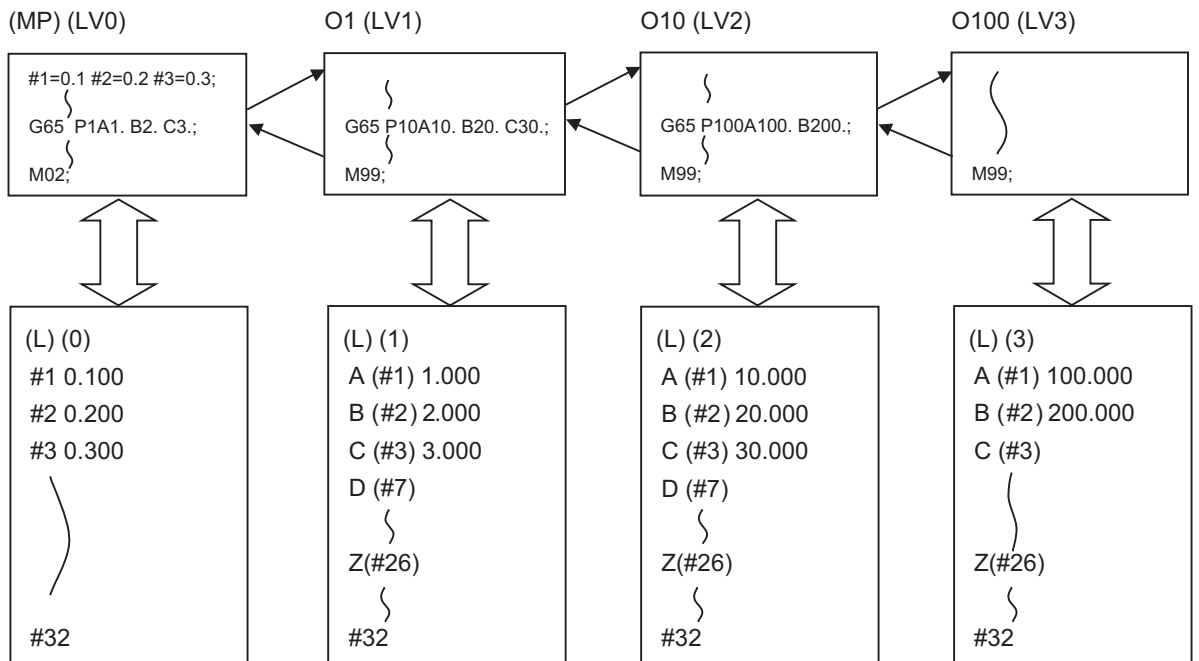
(Note 1) The numbers 1 to 10 accompanying I, J and K indicate the sequence of the commanded sets, and are not required in the actual command.

- (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) 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.

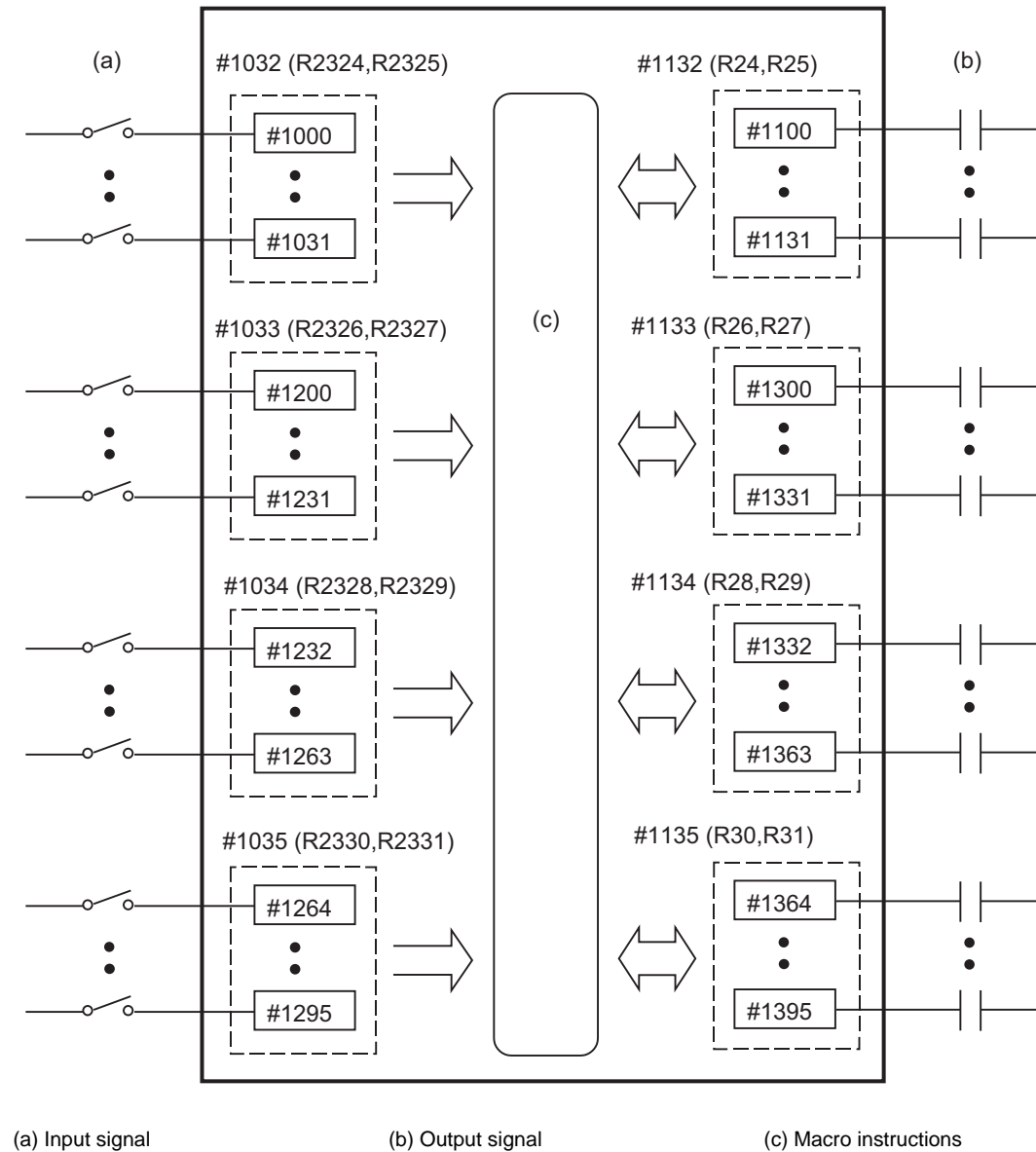
### 13.6.4.3 Macro Interface Inputs/Outputs (#1000 to #1035, #1100 to #1135, #1200 to #1295, #1300 to #1395)



#### Function and purpose

The status of the interface input signals can be ascertained by reading out the values of variable numbers #1000 to #1035, #1200 to #1295.

The interface output signals can be sent by substituting values in variable Nos. #1100 to #1135, #1300 to #1395.







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.

(1) Macro interface common to part systems (input)

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1000	1	Register R2324 bit 0	#1016	1	Register R2325 bit 0
#1001	1	Register R2324 bit 1	#1017	1	Register R2325 bit 1
#1002	1	Register R2324 bit 2	#1018	1	Register R2325 bit 2
#1003	1	Register R2324 bit 3	#1019	1	Register R2325 bit 3
#1004	1	Register R2324 bit 4	#1020	1	Register R2325 bit 4
#1005	1	Register R2324 bit 5	#1021	1	Register R2325 bit 5
#1006	1	Register R2324 bit 6	#1022	1	Register R2325 bit 6
#1007	1	Register R2324 bit 7	#1023	1	Register R2325 bit 7
#1008	1	Register R2324 bit 8	#1024	1	Register R2325 bit 8
#1009	1	Register R2324 bit 9	#1025	1	Register R2325 bit 9
#1010	1	Register R2324 bit 10	#1026	1	Register R2325 bit 10
#1011	1	Register R2324 bit 11	#1027	1	Register R2325 bit 11
#1012	1	Register R2324 bit 12	#1028	1	Register R2325 bit 12
#1013	1	Register R2324 bit 13	#1029	1	Register R2325 bit 13
#1014	1	Register R2324 bit 14	#1030	1	Register R2325 bit 14
#1015	1	Register R2324 bit 15	#1031	1	Register R2325 bit 15

System variable	No. of points	Interface input signal
#1032	32	Register R2324, R2325
#1033	32	Register R2326, R2327
#1034	32	Register R2328, R2329
#1035	32	Register R2330, R2331

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1200	1	Register R2326 bit 0	#1216	1	Register R2327 bit 0
#1201	1	Register R2326 bit 1	#1217	1	Register R2327 bit 1
#1202	1	Register R2326 bit 2	#1218	1	Register R2327 bit 2
#1203	1	Register R2326 bit 3	#1219	1	Register R2327 bit 3
#1204	1	Register R2326 bit 4	#1220	1	Register R2327 bit 4
#1205	1	Register R2326 bit 5	#1221	1	Register R2327 bit 5
#1206	1	Register R2326 bit 6	#1222	1	Register R2327 bit 6
#1207	1	Register R2326 bit 7	#1223	1	Register R2327 bit 7
#1208	1	Register R2326 bit 8	#1224	1	Register R2327 bit 8
#1209	1	Register R2326 bit 9	#1225	1	Register R2327 bit 9
#1210	1	Register R2326 bit 10	#1226	1	Register R2327 bit 10
#1211	1	Register R2326 bit 11	#1227	1	Register R2327 bit 11
#1212	1	Register R2326 bit 12	#1228	1	Register R2327 bit 12
#1213	1	Register R2326 bit 13	#1229	1	Register R2327 bit 13
#1214	1	Register R2326 bit 14	#1230	1	Register R2327 bit 14
#1215	1	Register R2326 bit 15	#1231	1	Register R2327 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1232	1	Register R2328 bit 0	#1248	1	Register R2329 bit 0
#1233	1	Register R2328 bit 1	#1249	1	Register R2329 bit 1
#1234	1	Register R2328 bit 2	#1250	1	Register R2329 bit 2
#1235	1	Register R2328 bit 3	#1251	1	Register R2329 bit 3
#1236	1	Register R2328 bit 4	#1252	1	Register R2329 bit 4
#1237	1	Register R2328 bit 5	#1253	1	Register R2329 bit 5
#1238	1	Register R2328 bit 6	#1254	1	Register R2329 bit 6
#1239	1	Register R2328 bit 7	#1255	1	Register R2329 bit 7
#1240	1	Register R2328 bit 8	#1256	1	Register R2329 bit 8
#1241	1	Register R2328 bit 9	#1257	1	Register R2329 bit 9
#1242	1	Register R2328 bit 10	#1258	1	Register R2329 bit 10
#1243	1	Register R2328 bit 11	#1259	1	Register R2329 bit 11
#1244	1	Register R2328 bit 12	#1260	1	Register R2329 bit 12
#1245	1	Register R2328 bit 13	#1261	1	Register R2329 bit 13
#1246	1	Register R2328 bit 14	#1262	1	Register R2329 bit 14
#1247	1	Register R2328 bit 15	#1263	1	Register R2329 bit 15

System variable	No. of points	Interface input signal	System variable	No. of points	Interface input signal
#1264	1	Register R2330 bit 0	#1280	1	Register R2331 bit 0
#1265	1	Register R2330 bit 1	#1281	1	Register R2331 bit 1
#1266	1	Register R2330 bit 2	#1282	1	Register R2331 bit 2
#1267	1	Register R2330 bit 3	#1283	1	Register R2331 bit 3
#1268	1	Register R2330 bit 4	#1284	1	Register R2331 bit 4
#1269	1	Register R2330 bit 5	#1285	1	Register R2331 bit 5
#1270	1	Register R2330 bit 6	#1286	1	Register R2331 bit 6
#1271	1	Register R2330 bit 7	#1287	1	Register R2331 bit 7
#1272	1	Register R2330 bit 8	#1288	1	Register R2330 bit 8
#1273	1	Register R2330 bit 9	#1289	1	Register R2331 bit 9
#1274	1	Register R2330 bit 10	#1290	1	Register R2331 bit 10
#1275	1	Register R2330 bit 11	#1291	1	Register R2331 bit 11
#1276	1	Register R2330 bit 12	#1292	1	Register R2331 bit 12
#1277	1	Register R2330 bit 13	#1293	1	Register R2331 bit 13
#1278	1	Register R2330 bit 14	#1294	1	Register R2331 bit 14
#1279	1	Register R2330 bit 15	#1295	1	Register R2331 bit 15

## (2) Macro interface by part system (input)

System variable	No. of points	Interface input signal						
		\$1	\$2	\$3	\$4	\$5	\$6	\$7
		R2470	R2570	R2670	R2770	R2870	R2970	R3070
#1000	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1001	1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1
#1002	1	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2
#1003	1	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3
#1004	1	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4
#1005	1	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5
#1006	1	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6
#1007	1	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7
#1008	1	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8
#1009	1	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9
#1010	1	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10
#1011	1	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11
#1012	1	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12
#1013	1	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13
#1014	1	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14
#1015	1	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15

System variable	No. of points	Interface input signal						
		\$1	\$2	\$3	\$4	\$5	\$6	\$7
		R2471	R2571	R2671	R2771	R2871	R2971	R3071
#1016	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1017	1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1
#1018	1	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2
#1019	1	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3
#1020	1	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4
#1021	1	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5
#1022	1	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6
#1023	1	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7
#1024	1	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8
#1025	1	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9
#1026	1	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10
#1027	1	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11
#1028	1	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12
#1029	1	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13
#1030	1	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14
#1031	1	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15

System variable	No. of points	Interface input signal						
		\$1	\$2	\$3	\$4	\$5	\$6	\$7
#1032	32	R2470, R2471	R2570, R2571	R2670, R2671	R2770, R2771	R2870, R2871	R2970, R2971	R3070, R3071
#1033	32	R2472, R2473	R2572, R2573	R2672, R2673	R2772, R2773	R2872, R2873	R2972, R2973	R3072, R3073
#1034	32	R2474, R2475	R2574, R2575	R2674, R2675	R2774, R2775	R2874, R2875	R2974, R2975	R3074, R3075
#1035	32	R2476, R2477	R2576, R2577	R2676, R2677	R2776, R2777	R2876, R2877	R2976, R2977	R3076, R3077

**Macro interface output (#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.

(1) Macro interface common to part systems (output)

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1100	1	Register R24 bit 0	#1116	1	Register R25 bit 0
#1101	1	Register R24 bit 1	#1117	1	Register R25 bit 1
#1102	1	Register R24 bit 2	#1118	1	Register R25 bit 2
#1103	1	Register R24 bit 3	#1119	1	Register R25 bit 3
#1104	1	Register R24 bit 4	#1120	1	Register R25 bit 4
#1105	1	Register R24 bit 5	#1121	1	Register R25 bit 5
#1106	1	Register R24 bit 6	#1122	1	Register R25 bit 6
#1107	1	Register R24 bit 7	#1123	1	Register R25 bit 7
#1108	1	Register R24 bit 8	#1124	1	Register R25 bit 8
#1109	1	Register R24 bit 9	#1125	1	Register R25 bit 9
#1110	1	Register R24 bit 10	#1126	1	Register R25 bit 10
#1111	1	Register R24 bit 11	#1127	1	Register R25 bit 11
#1112	1	Register R24 bit 12	#1128	1	Register R25 bit 12
#1113	1	Register R24 bit 13	#1129	1	Register R25 bit 13
#1114	1	Register R24 bit 14	#1130	1	Register R25 bit 14
#1115	1	Register R24 bit 15	#1131	1	Register R25 bit 15

System variable	No. of points	Interface output signal
#1132	32	Register R24, R25
#1133	32	Register R26,R27
#1134	32	Register R28,R29
#1135	32	Register R30,R31

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1300	1	Register R26 bit 0	#1316	1	Register R27 bit 0
#1301	1	Register R26 bit 1	#1317	1	Register R27 bit 1
#1302	1	Register R26 bit 2	#1318	1	Register R27 bit 2
#1303	1	Register R26 bit 3	#1319	1	Register R27 bit 3
#1304	1	Register R26 bit 4	#1320	1	Register R27 bit 4
#1305	1	Register R26 bit 5	#1321	1	Register R27 bit 5
#1306	1	Register R26 bit 6	#1322	1	Register R27 bit 6
#1307	1	Register R26 bit 7	#1323	1	Register R27 bit 7
#1308	1	Register R26 bit 8	#1324	1	Register R27 bit 8
#1309	1	Register R26 bit 9	#1325	1	Register R27 bit 9
#1310	1	Register R26 bit 10	#1326	1	Register R27 bit 10
#1311	1	Register R26 bit 11	#1327	1	Register R27 bit 11
#1312	1	Register R26 bit 12	#1328	1	Register R27 bit 12
#1313	1	Register R26 bit 13	#1329	1	Register R27 bit 13
#1314	1	Register R26 bit 14	#1330	1	Register R27 bit 14
#1315	1	Register R26 bit 15	#1331	1	Register R27 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1332	1	Register R28 bit 0	#1348	1	Register R29 bit 0
#1333	1	Register R28 bit 1	#1349	1	Register R29 bit 1
#1334	1	Register R28 bit 2	#1350	1	Register R29 bit 2
#1335	1	Register R28 bit 3	#1351	1	Register R29 bit 3
#1336	1	Register R28 bit 4	#1352	1	Register R29 bit 4
#1337	1	Register R28 bit 5	#1353	1	Register R29 bit 5
#1338	1	Register R28 bit 6	#1354	1	Register R29 bit 6
#1339	1	Register R28 bit 7	#1355	1	Register R29 bit 7
#1340	1	Register R28 bit 8	#1356	1	Register R29 bit 8
#1341	1	Register R28 bit 9	#1357	1	Register R29 bit 9
#1342	1	Register R28 bit 10	#1358	1	Register R29 bit 10
#1343	1	Register R28 bit 11	#1359	1	Register R29 bit 11
#1344	1	Register R28 bit 12	#1360	1	Register R29 bit 12
#1345	1	Register R28 bit 13	#1361	1	Register R29 bit 13
#1346	1	Register R28 bit 14	#1362	1	Register R29 bit 14
#1347	1	Register R28 bit 15	#1363	1	Register R29 bit 15

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1364	1	Register R30 bit 0	#1380	1	Register R31 bit 0
#1365	1	Register R30 bit 1	#1381	1	Register R31 bit 1
#1366	1	Register R30 bit 2	#1382	1	Register R31 bit 2
#1367	1	Register R30 bit 3	#1383	1	Register R31 bit 3
#1368	1	Register R30 bit 4	#1384	1	Register R31 bit 4
#1369	1	Register R30 bit 5	#1385	1	Register R31 bit 5
#1370	1	Register R30 bit 6	#1386	1	Register R31 bit 6
#1371	1	Register R30 bit 7	#1387	1	Register R31 bit 7
#1372	1	Register R30 bit 8	#1388	1	Register R31 bit 8
#1373	1	Register R30 bit 9	#1389	1	Register R31 bit 9
#1374	1	Register R30 bit 10	#1390	1	Register R31 bit 10
#1375	1	Register R30 bit 11	#1391	1	Register R31 bit 11
#1376	1	Register R30 bit 12	#1392	1	Register R31 bit 12
#1377	1	Register R30 bit 13	#1393	1	Register R31 bit 13
#1378	1	Register R30 bit 14	#1394	1	Register R31 bit 14
#1379	1	Register R30 bit 15	#1395	1	Register R31 bit 15

(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.

## (2) Macro interface by part system (output)

System variable	No. of points	Interface output signal						
		\$1	\$2	\$3	\$4	\$5	\$6	\$7
		R170	R270	R370	R470	R570	R670	R770
#1100	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1101	1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1
#1102	1	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2
#1103	1	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3
#1104	1	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4
#1105	1	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5
#1106	1	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6
#1107	1	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7
#1108	1	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8
#1109	1	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9
#1110	1	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10
#1111	1	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11
#1112	1	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12
#1113	1	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13
#1114	1	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14
#1115	1	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15

System variable	No. of points	Interface output signal						
		\$1	\$2	\$3	\$4	\$5	\$6	\$7
		R171	R271	R371	R471	R571	R671	R771
#1116	1	bit0	bit0	bit0	bit0	bit0	bit0	bit0
#1117	1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1	bit 1
#1118	1	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2	bit 2
#1119	1	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3	bit 3
#1120	1	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4	bit 4
#1121	1	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5	bit 5
#1122	1	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6	bit 6
#1123	1	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7	bit 7
#1124	1	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8	bit 8
#1125	1	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9	bit 9
#1126	1	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10	bit 10
#1127	1	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11	bit 11
#1128	1	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12	bit 12
#1129	1	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13	bit 13
#1130	1	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14	bit 14
#1131	1	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15	bit 15

System variable	No. of points	Interface output signal						
		\$1	\$2	\$3	\$4	\$5	\$6	\$7
#1132	32	R170, R171	R270, R271	R370, R371	R470, R471	R570, R571	R670, R671	R770, R771
#1133	32	R172, R173	R272, R273	R372, R373	R472, R473	R572, R573	R672, R673	R772, R773
#1134	32	R174, R175	R274, R275	R374, R375	R474, R475	R574, R575	R674, R675	R774, R775
#1135	32	R176, R177	R276, R277	R376, R377	R476, R477	R576, R577	R676, R677	R776, R777



13.6.4.4 Tool Compensation



Detailed description

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

Variable No. range	Variable No. range		Details
	#1120 TofVal = 0	#1120 TofVal = 1	
#10001 to #10000 + n	#2001 to #2000 + n	#2701 to #2700 + n	X shape compensation amount
#11001 to #11000 + n	#2701 to #2700 + n	#2001 to #2000 + n	X wear compensation amount
#12001 to #12000 + n			Additional axis shape compensation amount
#13001 to #13000 + n			Additional axis wear compensation amount
#14001 to #14000 + n	#2101 to #2100 + n	#2801 to #2800 + n	Z shape compensation amount
#15001 to #15000 + n	#2801 to #2800 + n	#2101 to #2100 + n	Z wear compensation amount
#16001 to #16000 + n	#2201 to #2200 + n	#2901 to #2900 + n	R shape compensation amount
#17001 to #17000 + n	#2901 to #2900 + n	#2201 to #2200 + n	R wear compensation amount
#18001 to #18000 + n	#2301 to #2300 + n		Nose compensation amount

"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.

The tool compensation data is configured as data with a decimal point in the same way as other variables. When "# 10001=1000;" is programmed, 1000.000 is set in tool compensation data.

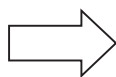
The additional axes' tool compensation can be used for only either the 3rd axis or 4th axis.

Which axis to use is selected by the parameter "#1520 Tchg34".

The variable No. corresponding to the #2000s' shape/wear compensation amount can be changed with the parameter "#1122 TofVal".

Programming example

```
#101=1000;
#10001=#101;
#102=#10001;
```



Common variables

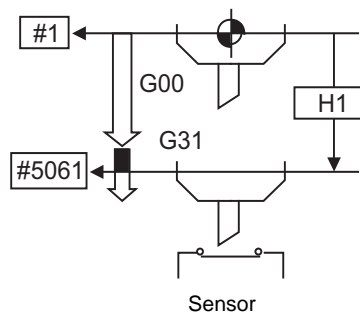
```
#101=1000.0
#102=1000.0
```

Tool compensation data

```
H1=1000.000
```

(Example 1) Calculation and tool offset data setting

- G28X0 T0101; Reference position return
- M06; Tool change (T0101)
- #1=#5001; Start point memory
- G00 X-200. ; Rapid traverse to safe position
- G31 X-50. F100 ; Skip measurement
- #10001=#5061-#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. #5001 is the X axis start point position and #5061 indicates the position at which the skip signal is input while G31 is being executed in the X axis skip coordinates.

### 13.6.4.5 Workpiece Coordinate System Compensation (External Workpiece Coordinate Offset) (#5201 - #532n)



#### Detailed description

By using variable Nos #5201 to #532n, it is possible to read out the workpiece coordinate system compensation data or to substitute values.

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

Coordinate name	1st axis	2nd axis	3rd axis	4th axis	.....	nth axis	Remarks
External workpiece offset	#5201	#5202	#5203	#5204	.....	#520n	External workpiece offset specifications are required.
G54	#5221	#5222	#5223	#5224	.....	#522n	
G55	#5241	#5242	#5243	#5244	.....	#524n	
G56	#5261	#5262	#5263	#5264	.....	#526n	
G57	#5281	#5282	#5283	#5284	.....	#528n	
G58	#5301	#5302	#5303	#5304	.....	#530n	
G59	#5321	#5322	#5323	#5324	.....	#532n	

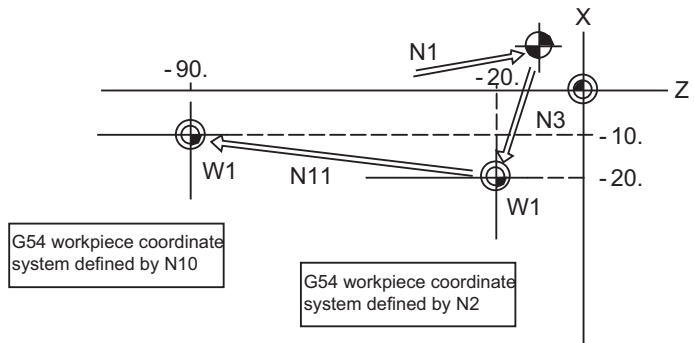
(Example 1)

```

N1 G28 X0 Z0 ;
N2 #5221=-20. #5222=-20. ;
N3 G00 G54 X0 Z0 ;

N10 #5221=-10. #5222=-90. ;
N11 G00 G54 X0Z0 ;

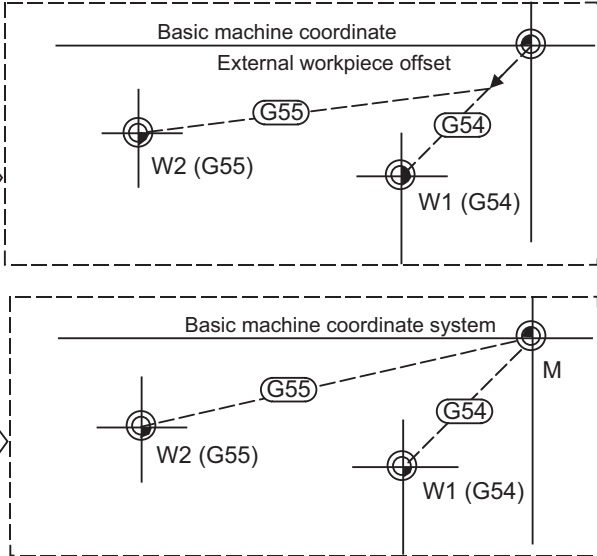
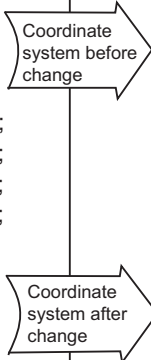
M02 ;
    
```



(Example 2)

```

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



This is an example where the external workpiece compensation values are added to the workpiece coordinate (G54, G55) system compensation values without changing the position of the workpiece coordinate systems.

### 13.6.4.6 NC Alarm (#3000)



#### Detailed description

The NC unit can be forcibly set to the alarm state by using variable No. #3000.

**#3000= 70 (CALL #PROGRAMMER #TEL #530) ;**

70	Alarm No.
CALL #PROGRAMMER #TEL #530	Alarm message

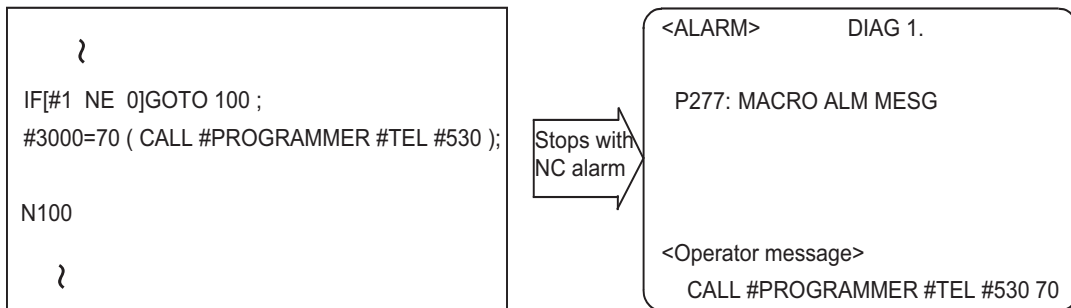
Any alarm number from 1 to 9999 can be specified.

The alarm message must be written in 31 or less characters.

NC alarm 3 signal (program error) is output.

The "P277: MACRO ALM MESG" appears in the <ALARM> column on "DIAG 1." screen and the alarm No. and alarm message "70: (CALL #PROGRAMMER #TEL #530)" will appear in the <Operator message>.

#### Example of program (alarm when #1 = 0)



- (Note 1) Alarm No. does not display 0. Any number exceeding 9999 cannot be displayed.
- (Note 2) The characters following the first alphabet letter in the right member is treated as an alarm message. Therefore, a number cannot be designated as the first character of an alarm message. It is recommended that the alarm messages be enclosed in round parentheses.
- (Note 3) Only the system that the alarm numbers and alarm messages are issued in #3000 can be forcibly set to the alarm state. However, operator messages will be displayed on each screen as they are common system.

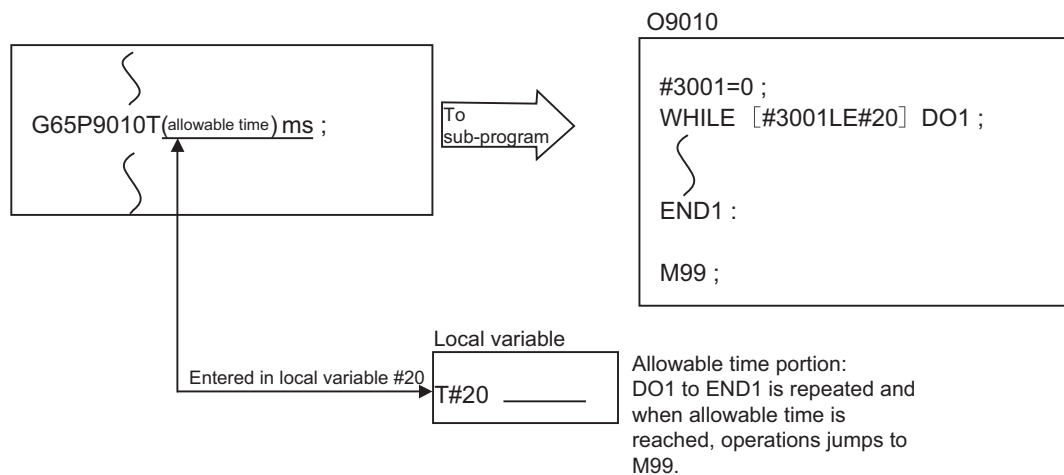
### 13.6.4.7 Integrating Time (#3001, #3002)



#### Detailed description

The integrating time during the power is turned ON or the automatic start is running, can be read or values can be substituted by using variable Nos. #3001 and #3002.

Type	Variable No.	Unit	Contents when power is switched on	Initialization of contents	Count condition
Power ON	3001	1ms	Same as when power is switched off	Substitute values to variables	At all times while power is ON
Automatic start	3002				In-automatic start



### 13.6.4.8 Suppression of Single Block Stop and Miscellaneous Function Finish Signal Waiting (#3003)



#### Detailed description

By substituting the values below in variable No. #3003, it is possible to suppress single block stop in the subsequent blocks or to advance to the next block without waiting for the miscellaneous function (M, S, T, B) finish (FIN) signal.

#3003	Single block stop	Miscellaneous function finish signal
0	Not suppressed	Wait
1	Suppressed	Wait
2	Not suppressed	Not wait
3	Suppressed	Not wait

(Note 1) Variable No. #3003 is set to zero by NC reset.

### 13.6.4.9 Feed Hold, Feedrate Override, G09 Valid/Invalid (#3004)



#### Detailed description

By substituting the values below in variable No. #3004, it is possible to make the feed hold, feedrate override and G09 functions either valid or invalid in the subsequent blocks.

#3004 Contents (value)	Bit 0	Bit 1	Bit 2
	Feed hold	Feedrate override	G09 check
0	Valid	Valid	Valid
1	Invalid	Valid	Valid
2	Valid	Invalid	Valid
3	Invalid	Invalid	Valid
4	Valid	Valid	Invalid
5	Invalid	Valid	Invalid
6	Valid	Invalid	Invalid
7	Invalid	Invalid	Invalid

(Note 1) Variable No. #3004 is set to zero by NC reset.

(Note 2) The functions are valid when the above bits are 0, and invalid when they are 1.

### 13.6.4.10 Message Display and Stop (#3006)



#### Detailed description

By using variable No. #3006, the operation stops after the previous block is executed and, if message display data is commanded, the corresponding message and the stop No. will be indicated on the operator message area.

#3006 = 1( TAKE FIVE );

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.

(Note 1) Only "1" is valid to set the number on #3006. Block Stop is not possible and the messages will not be displayed if the number other than "1" is set.

(Note 2) Block stop is possible only for the system which is issued to set "1" on #3006, however the operator messages are displayed on each part system's alarm screen as they are common to part systems.

### 13.6.4.11 Mirror Image (#3007)



#### Detailed description

By reading variable No. #3007, it is possible to ascertain the status of mirror image of the each axis at the point.

Each axis corresponds to a bit of #3007.

When the bits are 0, the mirror image function is invalid.

When the bits are 1, the mirror image function is valid.

#3007

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
nth axis													4	3	2	1

13.6.4.12 G Command Modals (#4001-#4021, #4201-#4221)



Detailed description

Using variable Nos. #4001 to #4021, it is possible to read the modal commands which have been issued in previous blocks.

Similarly, it is possible to read the modals in the block being executed with variable Nos. #4201 to #4221.

Variable No.		Function	
Pre-read block	Execution block		
#4001	#4201	Interpolation mode	G00 : 0, G01 : 1, G02 : 2, G03 : 3, G33 : 33, G34 : 34
#4002	#4202	Plane selection	G17 : 17, G18 : 18, G19 : 19
#4003	#4203	Absolute/incremental	G90 : 90, G91 : 91
#4004	#4204	Barrier check	G22 : 22, G23 : 23
#4005	#4205	Feed designation	G94 : 94, G95 : 95
#4006	#4206	Inch/metric	G20 : 20, G21 : 21
#4007	#4207	Tool nose R compensation	G40 : 40, G41 : 41, G42 : 42, G46 : G46
#4008	#4208	No variable No.	
#4009	#4209	Fixed cycle	G80 : 80, G70-G79 : 70-79, G83-G85 : 83-85, G83.2 : 83.2, G87-G89 : 87-89
#4010	#4210	Return level	G98 : 98, G99 : 99
#4011	#4211		
#4012	#4212	Workpiece coordinate system	G54-G59 : 54-59
#4013	#4213	Acceleration/deceleration	G61-G64 : 61-64, G61.1 : 61.1
#4014	#4214	Macro modal call	G66 : 66, G66.1 : 66.1, G67 : 67
#4015	#4215		
#4016	#4216	No variable No.	
#4017	#4217	Constant surface speed control	G96 : 96, G97 : 97
#4018	#4218	Balance cut	G14 : 14, G15 : 15
#4019	#4219		
#4020	#4220		
#4021	#4221		

(Example)

```
G28 X0 Z0 ;
G00 X150. Z200 ;
G65 P300 G02 W-30. K-15. F1000 ;
M02 ;
O300
#1 = #4001 ; = -> Group 01 G modal (pre-read) #1 = 2.0
# = #4201 ; = -> Group 01 G modal (now being executed) #2 = 0.0
G#1 W#24 ;
M99 ;
%
```

## 13.6.4.13 Other Modals (#4101 - #4140, #4301 - #4340)



## Detailed description

Using variable Nos. #4101 to #4140, 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 #4340.

Variable No.		Modal information	Variable No.		Modal information
Pre-read	Execution		Pre-read	Execution	
#4101	#4301		#4121	#4321	
#4102	#4302	2nd miscellaneous function (B)	#4122	#4322	
#4103	#4303		#4123	#4323	
#4104	#4304		#4124	#4324	
#4105	#4305		#4125	#4325	
#4106	#4306		#4126	#4326	
#4107	#4307	Tool radius compensation No. (D)	#4127	#4327	
#4108	#4308		#4128	#4328	
#4109	#4309	Feedrate (F)	#4129	#4329	
#4110	#4310		#4130	#4330	Extended workpiece coordinate No. (P) (Note 1)
#4111	#4311	Tool length offset No. (H)	#4131	#4331	
#4112	#4312		#4132	#4332	
#4113	#4313	Miscellaneous function (M)	#4133	#4333	
#4114	#4314	Sequence number (N)	#4134	#4334	
#4115	#4315	Program number (O)	#4135	#4335	
#4116	#4316		#4136	#4336	
#4117	#4317		#4137	#4337	
#4118	#4318		#4138	#4338	
#4119	#4319	Spindle function (S)	#4139	#4339	
#4120	#4320	Tool function (T)	#4140	#4340	

(Note 1) If the workpiece coordinate system is not G54.1, the extended workpiece coordinate No. (P) will be "0".



13.6.4.14 Position Information (#5001 - #5100 + n)

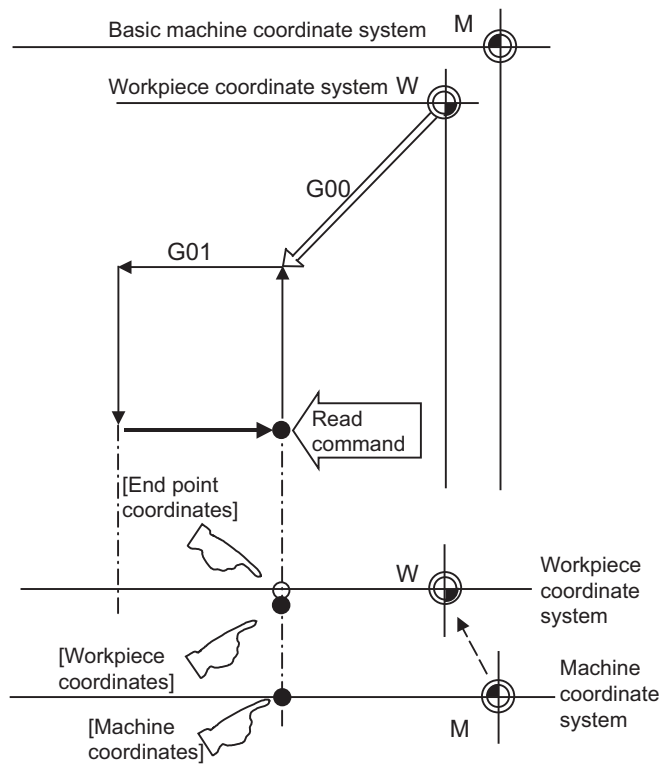


Detailed description

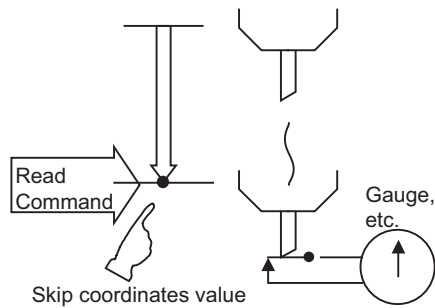
Using variable Nos. #5001 to #5100, it is possible to read the end point coordinates, machine coordinates, workpiece coordinates, skip coordinates, tool position compensation amount and servo deviation amounts in the last block.

Position information	Axis No.						Remarks (reading during movement)
	1	2	3	4	...	n	
End point coordinate of the last block	#5001	#5002	#5003	#5004	...	#5000+n	Yes
Machine coordinate	#5021	#5022	#5023	#5024	...	#5020+n	No
Workpiece coordinate	#5041	#5042	#5043	#5044	...	#5040+n	No
Skip coordinate	#5061	#5062	#5063	#5064	...	#5060+n	Yes
Tool position compensation amount	#5081	#5082	#5083	#5084	...	#5080+n	No
Servo deviation amount	#5101	#5102	#5103	#5104	...	#5100+n	Yes

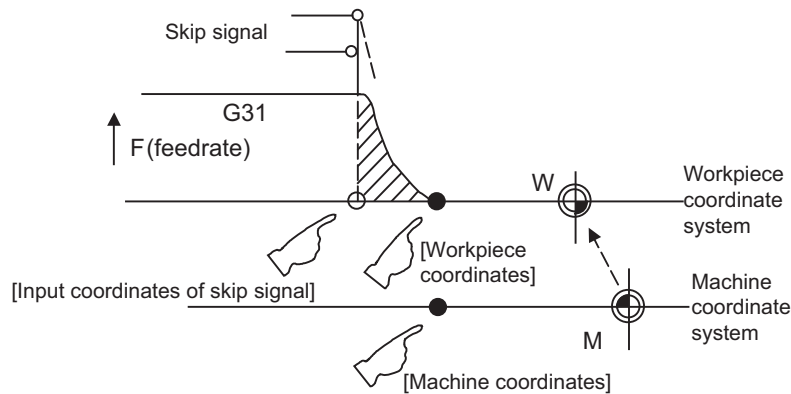
(Note) The number of axes which can be controlled differs according to the specifications. The last digit of the variable No. corresponds to the control axis No.



- (1) The positions of the end point coordinates and skip coordinates are positions in the workpiece coordinate system.
- (2) The end point coordinates, skip coordinates and servo deviation amounts can be read even during movement. However, it must first be checked that movement has stopped before reading the machine coordinates and the workpiece coordinates.
- (3) The skip coordinates indicates the position where the skip signal is turned ON in the G31 block. If the skip signal does not turn ON, they will be the end point position.  
(For further details, refer to the section on Automatic Tool Length Measurement.)



- (4) The end point coordinates indicate the tool nose position regardless of the tool compensation and other such factors. On the other hand, the machine coordinates, workpiece coordinates and skip coordinates indicate the tool reference point position with consideration given to tool compensation.



For "●", check stop and then proceed to read.

For "○", reading is possible during movement.

The skip signal input coordinates value is the position in the workpiece coordinate system. The coordinate value in variable Nos. #5061 to #5064 memorize the moments when the skip input signal during movement was input and so they can be read at any subsequent time.

For further details, refer to the section on "Skip Function".

### 13.6.4.15 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	

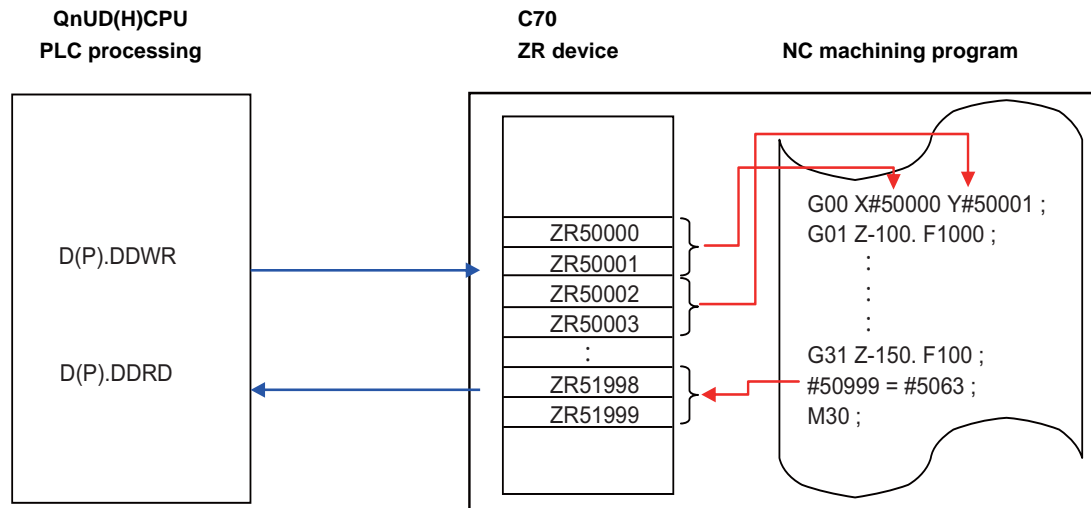
(Note) The number of workpiece machining times must be a positive value.

## 13.6.4.16 ZR device access variable



## Detailed description

By using variable Nos.#50000 to #51199, machining programs can read the ZR device data in NC side that is accessible with PLC I/F command, in the NC side or values can be substituted from QnUD(H)CPU.



ZR device access variable and ZR device no. correspondence table

Variable Nos.	ZR device Nos.
#50000	ZR50000,ZR50001
#50001	ZR50002,ZR50003
#50002	ZR50004,ZR50005
:	:
#5000 + n	ZR50000+2n, ZR50000+2n+1
:	:
#51199	ZR52398,ZR52399

These variables can be decimal points valid or invalid depending on the variable numbers as shown in the table below.

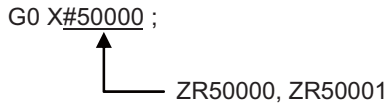
Decimal points Valid/Invalid	Variable Nos.	The number of sets
Valid	#50000 to #50999	1000
Invalid	#51000 to #51199	200

Decimal point position can be varied depending on the parameter "#1003 iunit (input setup units)" and "#1041 I_inch (initial inch)" settings. When setting the values in the ZR device, consider these parameter settings and then decide the decimal point position.

**Readout variables**

- (1) For decimal points valid variable numbers (#50000 to #50999)  
It is used for addresses (X,Y,Z,U,V,W,A,B,C,I,J,K,F,E,Q,R) where a decimal point is valid such as the distances, angles, times and speeds in machining programs.

When a variable is used as below in machining programs, the data set in the device ZR50000 and ZR50001 will be referred.



Example of relationship between the setting value for ZR device and the variable commands

Device	Value	#50000	Parameter	Command
ZR50001	0x0001	0x1e240(hexadecimal) = 123456(decimal)	iunit = B I_inch = 0	X123.456
ZR50000	0xe240			

As the variable #50000 is decimal points valid variable number, it means the same as the decimal points valid variable number command as shown in the "Command" column in the table above.

Input unit "iunit" for the valid decimal point address and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command position when "123456" is set to #50000 by X#50000 (where the X axis is the linear axis) command	X123.456	X12.3456	X12.3456	X1.23456
Command position when "123456" is set to #50000 (where the C axis is the rotary axis) command	C123.456	C12.3456	C12.346 (Note)	C1.2346 (Note)

- (Note) For rotary axis, the displayed number of digits below the decimal point is the same as the metric system when the inch system is selected although the decimal point position of the data is the same as the linear axis.  
Therefore, if "123456" is set to ZR50000,1 and iunit = B the decimal point position is the same as "C12.3456" is commanded when the rotation axis is commanded. Similarly, if iunit = C the decimal point position is the same as "C12.3456" is commanded.

Among the addresses where the decimal point command is valid, the number of digit of below the decimal point for address F is different from other addresses.

The number of digit for below the significant decimal point will be rounded off to the nearest increment.

Input unit "iunit" for address F and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	2 digits below the decimal point	3 digits below the decimal point	3 digits below the decimal point	4 digits below the decimal point
Command speed when "123456" is set to #50010 by F#50010 command	F123.46	F12.346	F12.346	F1.2346

When using these variable numbers for the decimal point invalid addresses (D,H,L,M,N,O,S,T) as shown below, the value which is rounded off below the decimal point in the variable becomes the command value.

N#50010 ;

Input unit "iunit" for the decimal point invalid addresses and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command when "234567" is set to #50010 by F#50010 command	N235	N23	N23	N2

The following table shows the valid range of these variables for the addresses where the decimal point command is valid.

The valid range of these variables for the addresses where the decimal point command is valid

	Movement command (linear)	Movement command (rotation)	Feedrate (Note 1)	Dwell (Note 1)
mm	-99999999 to 99999999	-99999999 to 99999999	-1000000000 to 1000000000	-99999999 to 99999999
inch	-99999999 to 99999999	-99999999 to 99999999	-39370.0787 to 39370.0787	-99999999 to 99999999

(Note 1) "-" will be ignored even "-" is set to Feedrate or Dwell.

(Note 2) The program error (P35) will occur if the command is executed by setting the value, which exceeds the address command range, to these variable numbers.

- (2) For variable number commanded is decimal point invalid (#51000 to #51199)  
 As shown in the example of program below, the data which is set to these variables becomes the command value when these variables are used for the decimal point invalid addresses (D,H,L,M,N,O,S,T) regardless of parameter "#1003 iunit (input setup unit)" or "#1041 I_inch (initial inch)".

S#51000 ;

Input unit "iunit" for the decimal point invalid addresses and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command when "500" is set to #51000 by S#51000 command	S500	S500	S500	S500

Valid range of these variables for decimal point invalid address is within the address command range for variables. The program error (P35) will occur if a value exceeding the command range set to the variable and executed.

If these variables are executed to the decimal point valid address as shown in the example of program below, the command value is expressed as below since these variables will be treated as the data with a decimal point.

X#51000 ;

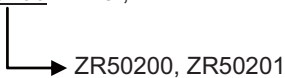
Input unit "iunit" for the decimal point valid addresses and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Command when "500" is set to #51000 by X#51000 command	X500.000	X500.0000	X500.0000	X500.00000

## Substituting in variables

- (1) For variable numbers which are decimal point valid (#50000 to #50999)

When substituting value without decimal point into decimal point valid variable as below, the value will be shifted by the number of the digits in fractional part of the value whose unit is set by the parameter #1003 and be set to ZR devices, regardless of the "#1078 Decpt2(Decimal point type 2)" setting.

#50100 = 123 ;  

 ZR50200, ZR50201

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #50100 by #50100 = 123 command	123000	1230000	1230000	12300000

When the value with decimal point is substituted into the variable as shown below, the value with decimal point will be substituted.

#50101 = 987.654 ;

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #50101 by #50101 = 987.654 command	987654	9876540	9876540	98765400

When a variable with decimal point such as a coordinate variable or a common variable is substituted into the decimal point valid variable as shown below, the coordinate variable or the common variable will be substituted.

#50200 = #5063 (#5063: Skip coordinate)

The table below lists the values which are set to the ZR devices when variable #5063 is substituted into variable #50200.

Substituted value into ZR device when variable is substituted

iunit	Metric system (I_inch = 0) Readout value of #5063	Inch System (I_inch = 1) Readout value of #5063	Substituted value for #50200	Value	Device
B	-123.456	-12.3456	-123456 = 0xfffe1dc0	0xfffe	ZR50401
C	-12.3456	-1.23456		0x1dc0	ZR50400



- (2) For variable numbers with decimal point invalid (#51000 to #51199)

When the value without decimal point is substituted into the decimal point invalid variable as shown below, the substituted value is set to ZR device regardless of parameter "#1003 iunit (input setup unit)", "#1041 I_inch (initial inch)", "#1078 Decpt2 (decimal point type 2)".

#51100 = 123 ;

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #51100 by #51100 = 123 command	123	123	123	123

When the value with decimal point is substituted into the variable, the value which is rounded off decimal point will be substituted.

#51101 = 7.543 ; or #51101 = 7.456 ;

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #51101 by #51101 = 73543 command	8	8	8	8
Substituted value into #51101 by #51101 = 73546 command	7	7	7	7

When a variable with decimal point such as a coordinate variable or a common variable is substituted into the decimal point valid variable as shown below, the value which is rounded off after decimal point will be substituted.

#51102 = #5021 ;

### Using ZR device access variables in user control statement

ZR device access variables can be used in user macro control statement.

Note that the variable data and true/false of the conditions are different between when using decimal point valid variables (#50000 to #50999) and when using decimal point invalid variables (#51000 to #51199).

- (1) For variable number with decimal point valid (#50000 to #50999)

(Example)

```
IF [#50100 EQ 1] GOTO 30 ;
```

```
G00 X100. ;
```

```
N30 ;
```

The table below lists #50100's values that allows the condition of the user macro control statement as shown above to be true.

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
The #50100's value to jump to N30 with control statement #50100 EQ 1.	1000	10000	10000	100000

- (2) For variable number with decimal point invalid (#51000 to #51199)

(Example)

```
IF [#51100 EQ 1] GOTO 30 ;
```

```
G00 X100. ;
```

```
N30 ;
```

The table below lists #51100's values that allow the condition of the user macro control statement as shown above to be true.

Input unit "iunit" and the decimal point position with mm/inch

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
Decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
The #50100's value to jump to N30 with control statement #50100 EQ 1.	1	1	1	1

When using a decimal point invalid variable in user macro control statement, the condition becomes true regardless of the parameter "#1003 iunit (input setup unit)" and "#1041 I_inch(Inicial state(inch))" settings.

**Common variable and substitution between ZR device access values**

(1) For decimal point valid variable (#50000 to #50999)

When a common variable is substituted into a decimal point valid variable, the value is substitute by the setting of parameter "#1003 iunit (input setup unit)", "#1041 I_inch (initial inch)" as show below.

#101 = 123.45678 ;

#50200 = #101 ;

The value of ZR device access variable when a common variable substituted

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
#101 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #101 by #101 = 123.45678 command	123.4568	123.4568	123.4568	123.4568
#50200 decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #50200 by #50200 = #101 command	123457	1234568	1234568	12345678

The number of decimal of the data is treated differently in between ZR devices which is long type, and common variables which is double type.

When the decimal point valid ZR device access variable is substituted into common variable as shown in the table below, the value is substituted by the settings of parameter "#1003 iunit (input setting unit)", "#1041 I_inch (initial inch)".

#50201 = 123.45678 ;

#102 = #50201 ;

Common variable value when ZR device access variable is substituted into common variable.

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
#50201 decimal point position	3 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	5 digits below the decimal point
Substituted value into #502001 by #50201 = 123.45678 command	123457	1234568	1234568	12345678
#102 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #102 by #102 = #50201 command	123.4570	123.4568	123.4568	123.4568

- (2) For decimal point invalid variable (#51000 to #51199)

When a common variable is substituted into the decimal point invalid variable, the value which is rounded off below the decimal point is substituted regardless of parameter "#1003 iunit (input setup unit)", "#1041 I_inch (initial inch)".

#101 = 123.4567 ;

#51100 = #101 ;

ZR device access variable data when a common variable is substituted

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
#101 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #101 by #101 = 123.4567 command	123.4567	123.4567	123.4567	123.4567
#51100 decimal point position	No value below the decimal point	No value below the decimal point	No value below the decimal point	No value below the decimal point
Substituted value into #51100 by #51100 = #101 command	123	123	123	123

#101 = 123.5432 ;

#51100 = #101 ;

ZR device access variable data when a common variable is substituted

	Metric system (I_inch = 0)		Inch System (I_inch = 1)	
	iunit = B	iunit = C	iunit = B	iunit = C
#101 decimal point position	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point	4 digits below the decimal point
Substituted value into #101 by #101 = 123.5432 command	123.5432	123.5432	123.5432	123.5432
#51100 decimal point position	No value below the decimal point	No value below the decimal point	No value below the decimal point	No value below the decimal point
Substituted value into #51100 by #51100 = #101 command	124	124	124	124

13.6.4.17 Tool Life Management (#60000 - #63016)



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 #63016 is designated by substituting a value in this variable No. If a group No. is not designated, the data of the group registered first is read. This is valid until reset.

- (2) Tool life management system variable No. (Read)

#60001 to #63016

# | 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	Method	Refer by registration No.
20	Status	Refer by registration No.
25	Life time/No. of uses	Refer by registration No.
30	Usage time/No. of uses	Refer by registration No.

The group No., method, and life data are common for the group.

| d | e | : Registration No. or data type

Registration No.

1 to 16

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 80
60002	Life current value	For each group (Designate Group No. #60000)	Usage time/No. of uses of tool being used.  Usage data of tool being used (If tool uses several compensation Nos., the total of the usage data for each compensation No.).	0 to 999999 min 0 to 999999 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 16
60004	Number of remaining registered tools		Total number of "usable" tools in the group.  Number of tools registered in designated group, whose ST is 0: Not used.	0 to 16
60005	Execution signal		"1" when this group is used in program being executed.  "1" when tool in designated group is selected.	0/1
60006	Cutting time cumulative value (min)		Indicates the time that this group is used in the program being executed.	
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
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.
61000 +***	Tool No.	Tool No. and compensation No.  Tool No. + compensation No. (When tool No. = 22 and compensation No. = 01, 2201=899H)		0 to 9999
61500 +***	Method	Whether to manage this group's life as time or number of uses.  0 : Time, 1: Number of uses		0/1
62000 +***	Status	Tool usage state  0 : Tool not used 1 : Tool in use 2 : Normal life tool 3 : Tool skip tool		0 to 3
62500 +***	Life time/No. of uses	This group's tool life value		0 to 999999 min 0 to 999999 times
63000 +***	Usage time/No. of uses			0 to 999999 min 0 to 999999 times



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.
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.
P1 LLn NNn ; ..... 1 is the group No., Ln is the life per tool, Nn is the method.
TTn ; ..... Tn is the tool No.
:
G11 ; ..... Registers the life data with the G10 command.
                        (The registered data is deleted.)
#111 = #61001 ; ..... Group 10 does not exist. #111 = -1.
```



## Precautions

- (1) If the tool life management system variable is commanded without designating a group No., the data of the group registered at the head of the registered data will be read.
- (2) If a non-registered group No. is designated and the tool life management system variable is commanded, "-1" will be read as the data.
- (3) If an unused registration No. tool life management system variable is commanded, "-1" will be read as the data.
- (4) Once commanded, the group No. is valid until NC reset.



### 13.6.5 Operation Commands



#### Function and purpose

A variety of operations can be performed between variables.



#### Command format

#i = <formula> ;

<Formula> is a combination of constants, variables, functions and operators.

Constants can be used instead of #j and #k below.

(1) Definition and substitution of variables	#i = #j	Definition, substitution
(2) Addition operation	#i = #j + #k	Addition
	#i = #j - #k	Subtraction
	#i = #j OR #k	Logical sum (at every bit of 32 bits)
	#i = #j XOR #k	Exclusive OR (at every bit of 32 bits)
(3) Multiplication operation	#i = #j * #k	Multiplication
	#i = #j / #k	Division
	#i = #j MOD #k	Remainder
	#i = #j AND #k	Logical product (at every bit of 32 bits)
(4) Functions	#i = SIN [#k]	Sine
	#i = COS [#k]	Cosine
	#i = TAN [#k]	Tangent $\tan \theta$ uses $\sin \theta / \cos \theta$ .
	#i = ASIN [#k]	Arcsine (Note 4)
	#i = ATAN [#k]	Arctangent (ATAN or ATN may be used)
	#i = ACOS [#k]	Arccosine
	#i = SQRT [#k]	Square root (SQRT or SQR may be used)
	#i = ABS [#k]	Absolute value
	#i = BIN [#k]	Conversion from BCD to BIN
	#i = BCD [#k]	Conversion from BIN to BCD
	#i = ROUND[#k]	Rounding off (ROUND or RND may be used)
	#i = FIX [#k]	Discard fractions less than 1
	#i = FUP [#k]	Add for fractions less than 1
	#i = LN [#k]	Natural logarithm
#i = EXP [#k]	Exponent with e (=2.718 ..... ) as bottom	

(Note 1) A value without a decimal point is basically treated as a value with a decimal point at the end (1 = 1.000).

(Note 2) Compensation amounts from #10001 and workpiece coordinate system compensation values from #5201 are handled as data with a decimal point. Consequently, data with a decimal point will be produced even when data without a decimal point have been defined in the variable numbers.  
(Example)

Operation Commands	Common variables after execution
#101 =1000 ; #10001 =#101 ; #102 =#10001 ;	#101 1000.000 #102 1000.000

(Note 3) The <formula> after a function must be enclosed in the square parentheses [ ].

(Note 4) Operation results differ depending on the setting of the parameter "#1273 ext09/bit0".

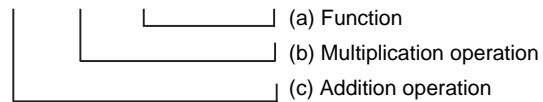


## Detailed description

### Sequence of operations

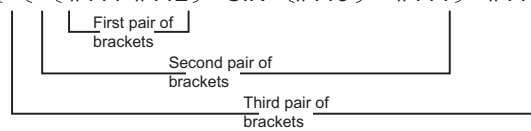
- (1) The sequence of the operations (a) to (c) is performed in the following order; the function, the multiplication operation and the addition operation.

```
#101=#111+#112*SIN [#113]
```



- (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	0.860 0.860 866.025 866.025 866.025 866.025				
			(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	

### Operation accuracy

As shown in the following table errors will be generated when performing operations once and these errors will be accumulated by repeating the operations.

Operation format	Average error	Maximum error	Type of error
a = b + c a = b - c	$2.33 \times 10^{-10}$	$5.32 \times 10^{-10}$	Min. $ \epsilon / b ,  \epsilon / c $
a = b * c	$1.55 \times 10^{-10}$	$4.66 \times 10^{-10}$	Relative error $ \epsilon / a $
a = b / c	$4.66 \times 10^{-10}$	$1.86 \times 10^{-9}$	
a = b	$1.24 \times 10^{-9}$	$3.73 \times 10^{-9}$	
a = SIN[b] a = COS[b]	$5.0 \times 10^{-9}$	$1.0 \times 10^{-8}$	Absolute error $ \epsilon ^\circ$
a = ATAN[b/c]	$1.8 \times 10^{-6}$	$3.6 \times 10^{-6}$	

(Note) SIN/COS is calculated for the function TAN.



### Precautions

(1) Notes on operation accuracy

- Addition and subtraction

It should be noted that when absolute values are used subtractively in addition or subtraction, the relative error cannot be kept below 10.

For instance, it is assumed that the real value produced as the operation calculation result of #10 and #20 are as follows (these value cannot be substituted directly):

#10 = 2345678988888.888

#20 = 2345678901234.567

Performing #10-#20 will not produce #10-#20 = 87654.321. There are 8 decimal digits in the variables and so the values of #10 and #20 will be as follows (strictly speaking, the internal values will differ somewhat from the value below because they are binary numbers):

#10 = 2345679000000.000

#20 = 2345678900000.000

Consequently, #10 . #20 = 100000.000 will generate a large error.

- Logical relation

EQ, NE, GT, LT, GE and LE are basically the same as addition and subtraction and so care should be taken with errors. For instance, to determine whether or not #10 and #20 are equal in the above example:

IF [#10 EQ #20]

It is not always possible to provide proper evaluation because of the above-mentioned error.

Therefore when the error is evaluated as in the following expression:

IF [ABS [#10-#20] LT 200000]

and the difference between #10 and #20 falls within the designated range error, both values should be considered equal.

- Trigonometric functions

Absolute errors are guaranteed with trigonometric functions but since the relative error is not under 10, care should be taken when dividing or multiplying after having used a trigonometric function.

### 13.6.6 Control Commands



#### Function and purpose

The flow of programs can be controlled by IF-GOTO- and WHILE-DO-END.



#### Detailed description

#### Branching

##### IF [conditional expression] GOTO n; (n = sequence number in the program)

When the condition is satisfied, control branches to "n" and when it is not satisfied, the next block is executed. IF [conditional expression] can be omitted and, when it is, control branches to "n" unconditionally.

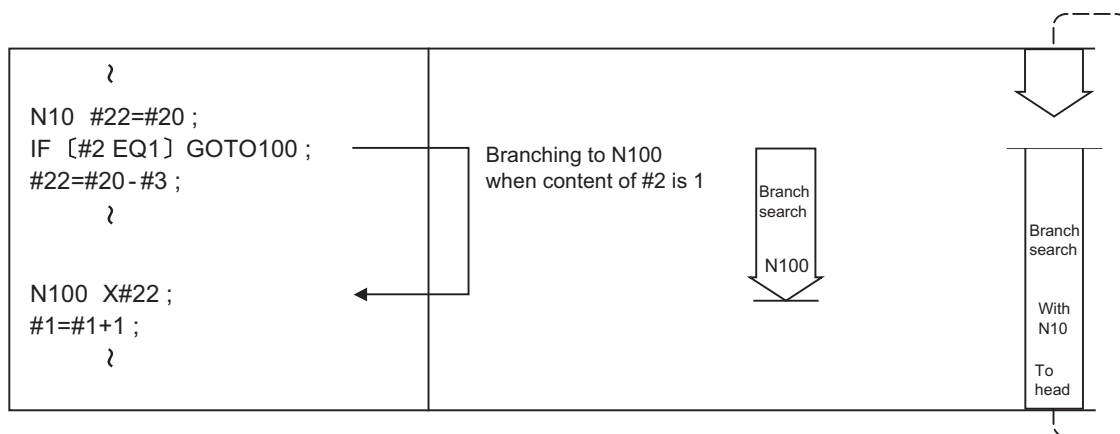
The following types of [conditional expressions] are available.

#i EQ #j	= When #i and #j are equal
#i NE #j	≠ When #i and #j are not equal
#i GT #j	> When #i is greater than #j
#i LT #j	< When #i is less than #j
#i GE #j	>= When #i is #j or more
#i LE #j	<= When #i is #j or less

"n" of "GOTO n" must always be in the same program. If not, program error (P231) will occur. A formula or variable can be used instead of #i, #j and "n".

In the block with sequence number "n" which will be executed after a "GOTO n" command, the sequence number "Nn" must always be at the head of the block. Otherwise, program error (P231) will occur.

If "/" is at the head of the block and "Nn" follows, control can be branched to the sequence number.



(Note 1) When searching the sequence number of the branch destination, the search is conducted up to the end of the program (% code) from the block following IF.....; and if it is not found, it is then conducted from the top of the program to the block before IF.....;. Therefore, branch searches in the opposite direction to the program flow will take longer time compared with branch searches in the forward direction.

(Note 2) EQ and NE should be compared only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.

(Note 3) Make sure that IF statements are not repeated. When a program in which IF statements are repeated (e.g., N10 [#2 EQ1] GOTO10) is executed, it may not operate correctly.

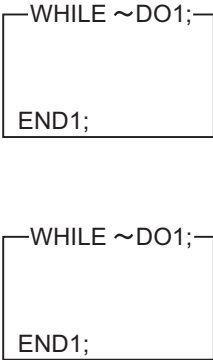
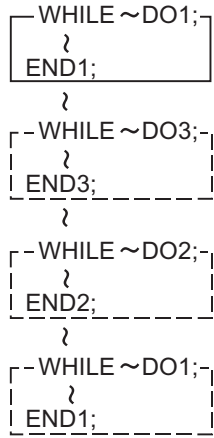
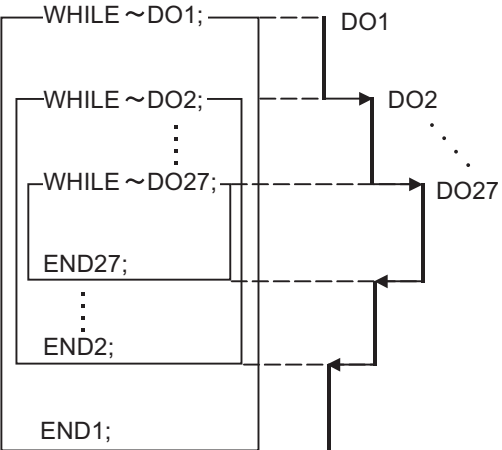
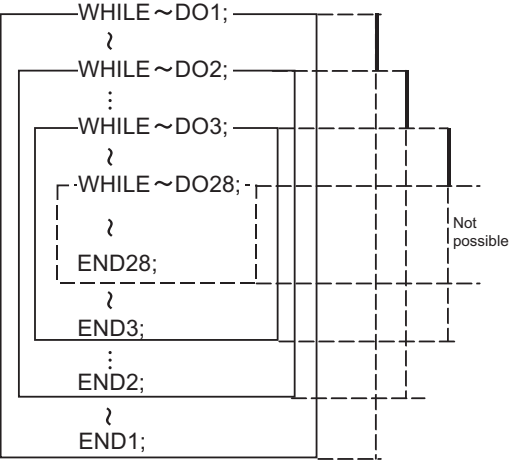
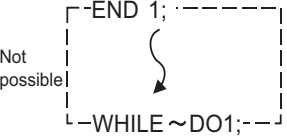
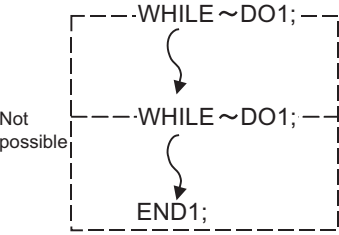
Repetitions

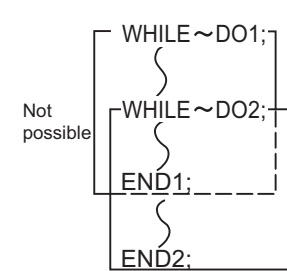
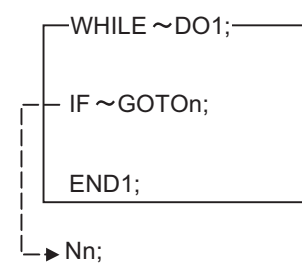
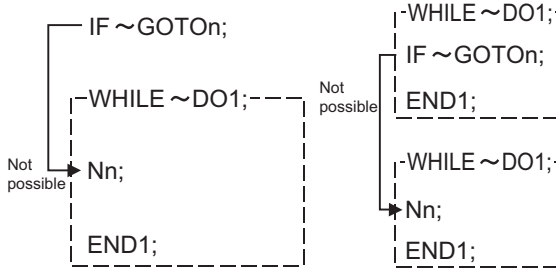
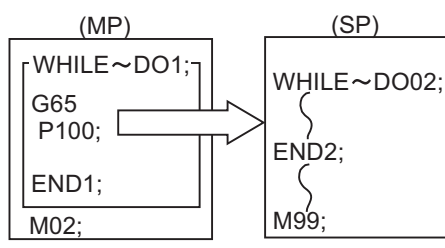
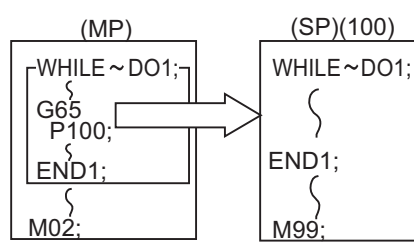
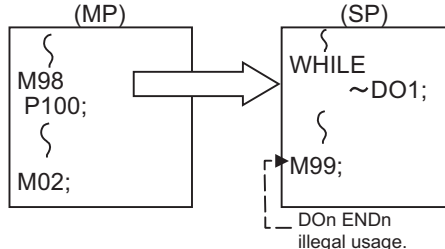
```

WHILE [conditional expression] DOm ; (m =1, 2, 3 ..... 127)
:
END m ;
    
```

While the conditional expression is established, the blocks from the following block to ENDm are repeatedly executed; when it is not established, execution moves to the block following ENDm. DOm may come before WHILE.

"WHILE [conditional expression] DOm" and "ENDm" must be used as a pair. If "WHILE [conditional expression]" is omitted, these blocks will be repeatedly ad infinitum. The repeating identification Nos. range from 1 to 127. (DO1, DO2, DO3, ..... DO127) Up to 27 nesting levels can be used.

<p>(1) Same identification No. can be used any number of times.</p> 	<p>(2) Any number may be used as the WHILE-DOm identification No.</p> 
<p>(3) Up to 27 nesting levels can be used for WHILE-DOm. "m" is any number from 1 to 127 for the nesting depth.</p>  <p>(Note) For nesting, "m" which has been used once cannot be used.</p>	<p>(4) The number of WHILE-DOm nesting levels cannot exceed 27.</p> 
<p>(5) WHILE - DOm must be designated first and ENDm last.</p> 	<p>(6) WHILE - DOm and ENDm must correspond on a 1:1 (pairing) basis in a same program.</p> 

<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.



### 13.6.7 Precautions



#### Precautions

When the user macro commands are employed, it is possible to use the M, S, T and other NC control commands together with the arithmetic, decision, branching and other macro commands for preparing the machining programs. When the former commands are made into executable statements and the latter commands into macro statements, the macro statement processing should be accomplished as quickly as possible in order to minimize the machining time, because such processing is not directly related to machine control.

By setting the parameter "#8101 macro single", the macro statements can be processed in parallel 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 Z0 ;	.....(1)	
N2 G92 X0 Z0 ;	.....(2)	
N3 G00 X-100. Z-100. ;	.....(3)	
N4 #101 = 100. * COS[210.] ;	.....(4)	(4),(5) Macro statements
N5 #103 = 100. * SIN[210.] ;	.....(5)	
N6 G01 X#101 Z#103 F800 ;	.....(6)	

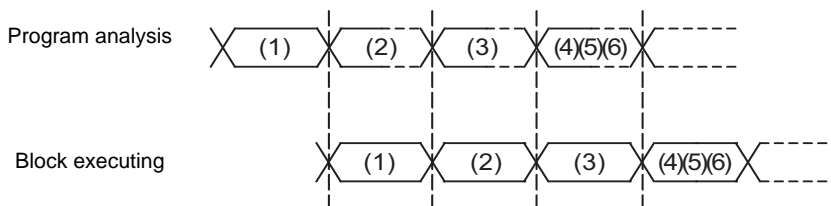
Macro statements are:

- (a) Arithmetic commands (block including =)
- (b) Control commands (block including GOTO, DO-END, etc.)
- (c) Macro call commands (including macro calls based on G codes and cancel commands (G65, G66, G66.1, G67))

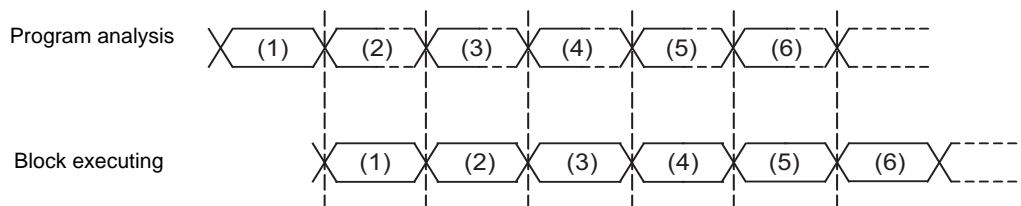
Execution statements refer to statements other than macro statements.

Flow of processing by the Program Example in the previous page

<Macro single OFF>



<Macro single ON>



Machining program display

<Macro single OFF>

<pre>[Executing] N3 G00 X-100. Z-100. ; [Next command] N6 G01 X#101 Z#103                 F800 ;</pre>	<p>N4, N5 and N6 are processed in parallel with the control of the executable statement of N3, N6 is an executable statement and so it is displayed as the next command. If the N4, N5 and N6 analysis is in time during N3 control, the machine movement will be continuously controlled.</p>
--------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<Macro single ON>

<pre>[Executing] N3 G00 X-100. Z-100. ; [Next command] N4 #101=100.                 *COS[210.] ;</pre>	<p>N4 is processed in parallel with the control of the executable statement of N3, and it is displayed as the next command. After N3 is finished, N5 and N6 are analyzed, and then N6 is executed. So the machine control is held on standby during the N5 and N6 analysis time.</p>
--------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 13.7 Mirror Image for Facing Tool Posts ; G68,G69



#### Function and purpose

In a machine in which the base turret and facing turret are integrated, this function is used to cut with the facing turret cutter using a program created with the base turret side.  
The distance between the two turrets is set in the parameters beforehand.



#### Command format

**G68 ; ... Mirror image for facing tool posts ON**

**G69 ; ... Mirror image for facing tool posts cancel**

[T command mirror image for facing tool posts]

The mirror image for facing tool posts can be turned ON and OFF with the T command instead of the G68/G69 command.

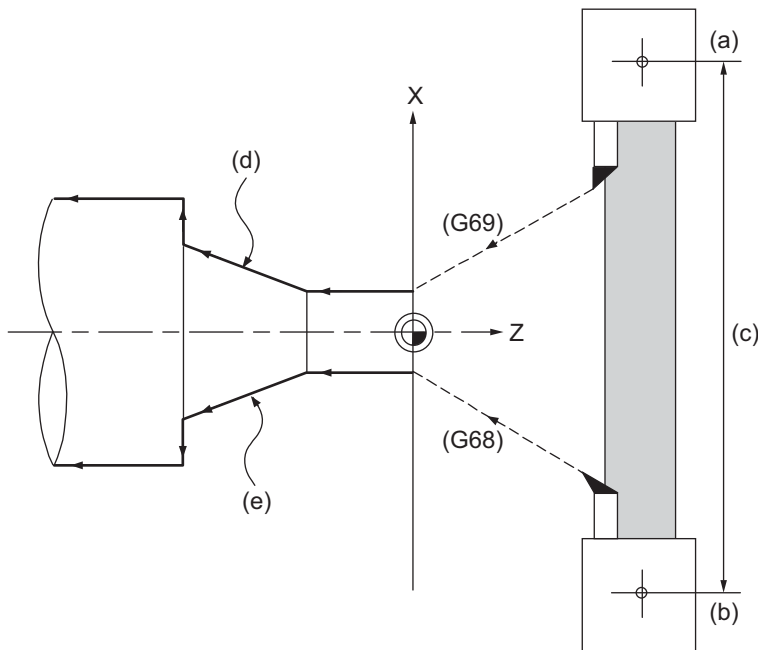
The T command for the G68 mode and the G69 mode is determined for each tool No. with the following base specification parameters.



#### Detailed description

When G68 is commanded, the following program coordinate system is shifted to the doubleturret side of the axis.

The axis movement direction is reversed from the program command. When G69 is commanded, the following program coordinate system will be returned to the base turret side.



(a) Base turret

(c) Turret distance (parameter: radius value)

(e) Facing turret path (Mirror image ON)

(b) Facing turret

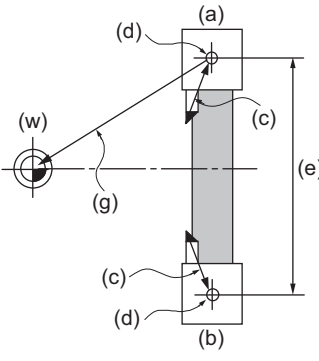
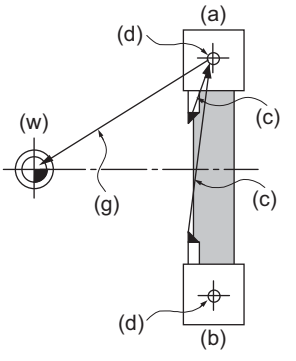
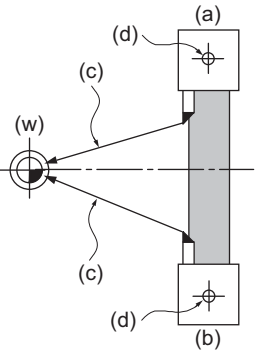
(d) Programmed path

### Tool compensation of facing turret

(1) Tool length offset

The tool length offset amount is the length from the tool nose to the tool length basic point. This also applies for the facing turret. Note that the offset amount setting value differs according to the tool length basic point position as shown below.

Tool length basic point and tool length offset

	Type A	Type B	Type C
Tool length basic point	Each turret basic point	Base turret basic point	Workpiece face center
Workpiece coordinate zero point	Workpiece face center	Workpiece face center	Workpiece face center
Turret distance	Distance between basic points of both turrets (radius value)	0	0
Workpiece offset	Workpiece coordinate zero point - base turret tool length basic point	Workpiece coordinate zero point - base turret tool length basic point	0
Tool length	Tool length basic point - tool nose position	Tool length basic point - tool nose position	Tool length basic point - tool nose position
Outline drawing			

- (a) Base turret
- (b) Facing turret
- (c) Tool length
- (d) Turret basic point
- (e) Turret distance
- (g) Workpiece offset
- (w) Workpiece coordinate zero point

The outline drawing in the table above shows the case when #1118 mirr_A is set to 0. When #1118 mirr_A is set to 1, the sign of the X axis tool length compensation amount for the facing turret will be reversed.

When "mirr_A" is 1, the sign for the X-axis tool length wear amount for the facing tool is reversed and the tool nose point is set to the opposite side (for instance, 2 → 3). When tool length measurement is executed with "mirr_A" as 0, the data will be accepted with "mirr_A" as 0.

(Setting examples)

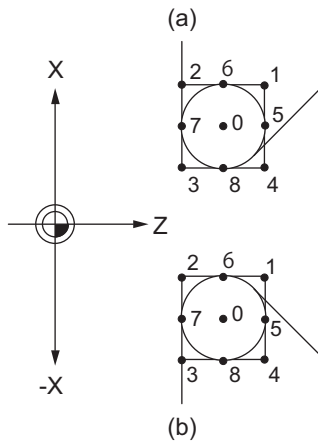
	mirr_A=0		mirr_A=1	
	X	Z	X	Z
Workpiece offset	-120.	-110.	-120.	-110.
Tool length of base turret	80.	35.	80.	35.
Tool wear amount of base turret	-20.	-5.	-20.	-5.
Tool nose point of base turret	3		3	
Tool length of facing turret	150.	40.	120.	40.
Tool wear amount of facing turret	-20.	-5.	-20.	-5.
Tool nose point of facing turret	2		3	
Distance between turret	0		0	

(2) Tool Nose Wear Compensation

The tool nose wear compensation amount is the length from the current tool nose to the original tool nose. The original tool nose is the tool nose when the tool length offset value was set.

(3) Tool nose point with nose R compensation

The tool nose point with nose R compensation is as follows.



(a) Base turret side

(b) Facing turret side

(4) Distance between turrets

The distance between the turrets is the distance from the tool length base point of the facing turret to the tool length base point of the base turret. It is set by parameter only for the X axis.

"0" is set when the tool length base point is common.

Machine parameter (Base specifications parameter) mirofs

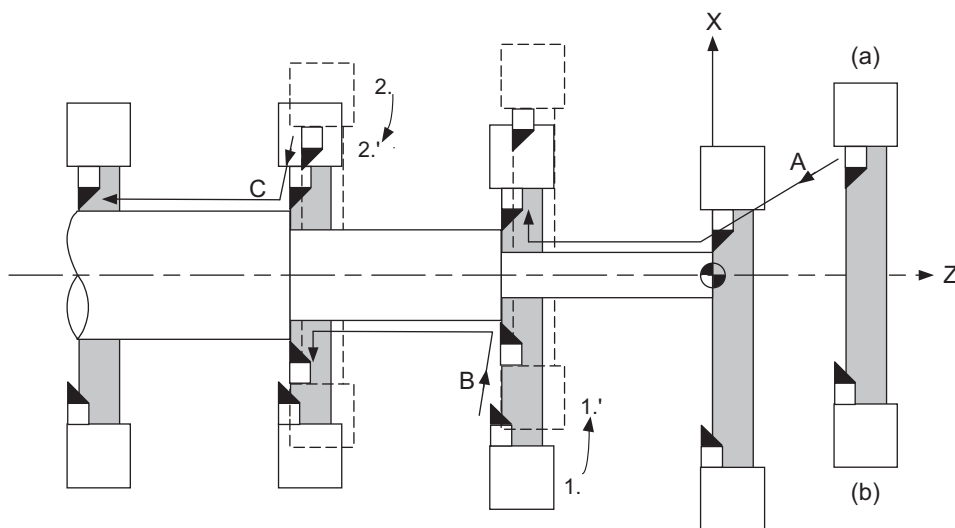
Setting range: 0 to 99999.999 (mm) (radial value)



Program example

(1) Example of operation by absolute value command

T0101 ; G00 X10. Z0. ; G01 Z-40. F400 ; X20. ;	Base turret selection	Machining with base turret .....A
G68 ; T0202 ; G00 X20. Z-40. ; G01 Z-80. F200 ; X30. ;	Mirror image for facing tool posts ON Facing turret selection	Machining with facing turret .....B
G69 ; T0101 ; G00 X30. Z-80. ; G01 Z-120. F400 ;	Mirror image for facing tool posts cancel Base turret selection	Machining with base turret .....C



(a) Base turret

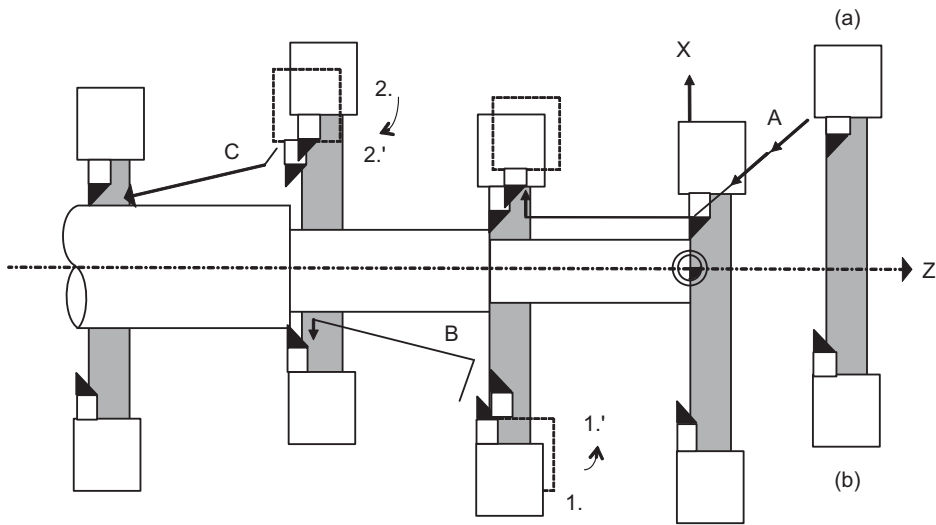
(b) Facing turret

A value determined from the turret distance parameter is added for movement by the first X-axis command issued after the mirror image for facing tool posts is turned ON. In the above operation example, program [1] block causes movement 1 to 1'.

Similarly, a value determined from the turret interval parameter is subtracted for movement by the first X-axis command issued after the mirror image for facing tool posts is canceled. In the above operation example, program [2] block causes movement from 2 to 2'.

(2) Example of operation by incremental value command

T0101; G00 X0. Z0.; G01 Z-40. F400; X20. ;	Selection of base turret	Machining with base turret..... A
G68; T0202; G00 U-10. W0.; G01 X20. Z-80. F200; X30. ;	Double-turret mirror image ON Selection of facing selection.....[1]'	Machining with facing turret..... B
G69; T0101; G00 U-10. W0.; G01 X30. Z-120. F400;	Double-turret mirror image cancel Selection of base turret.....[2]'	Machining with base turret..... C



(a) Base turret

(b) Facing turret

A incremental value command issued after the mirror image for facing tool posts is turned ON causes movement by the amount of the X-axis movement in the opposite direction specified by the program command. Block [1]' made by changing program [1] in "(1) Example of operation by absolute value command" to an incremental command causes the tool to move +10. by reversing .10. In the above operation example, the tool moves from 1 to 1'.

The same applies to an incremental value command issued after the the mirror image for facing tool posts is canceled. Block [2]' made by changing program [2] in "(1) Example of operation by absolute value command" to an incremental command causes the tool to move .10. In the above operation example, the tool moves from 2 to 2'.

## 13.8 Corner Chamfering/Corner Rounding I



### Function and purpose

Chamfering at any angle or corner rounding is performed automatically by adding ",C_" or ",R_" to the end of the block to be commanded first among those command blocks which shape the corner with lines only.

### 13.8.1 Corner Chamfering I ; G01 X_ Z_ ,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_" (or ",I_",",K_",",C_").



### Command format

```
N100 G01 X_ Z_ ,C_ ;
N200 G01 X_ Z_ ;
```

.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

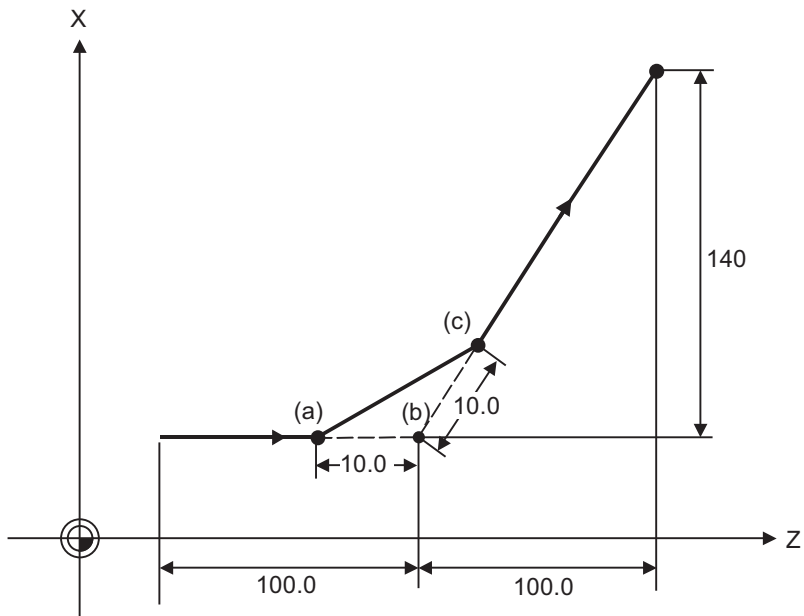
- (1) The start point of the block following the corner chamfering is the hypothetical corner intersection point.
- (2) If there are multiple or duplicate corner chamfering commands in a same block, the last command will be valid.
- (3) When both corner chamfering/corner rounding commands exist in a same block, the latter command will be valid.
- (4) Tool compensation is calculated for the shape which has already been subjected to corner 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 is less than the chamfering amount in the block following the corner chamfering block.
- (8) Program error (P382) will occur when a movement command is not issued in the block following the corner chamfering I command.





Program example

```
G01 W100. ,C10. F100 ;
U280. W100. ;
```



(a) Chamfering start point (b) Hypothetical corner intersection point (c) Chamfering end point

### 13.8.2 Corner Rounding I ; G01 X_ Z_ ,R_



#### Function and purpose

This performs a corner rounding to the both side of the hypothetical corner which would appear as if chamfering is not performed, at the radius of the circular commanded with ",R_" (or "R_").



#### Command format

```
N100 G01 X__ Z__ ,R__ ;  
N200 G01 X__ Z__ ;
```

,R / R	Circular radius of corner rounding
--------	------------------------------------

Corner rounding is performed at the point where N100 and N200 intersect.



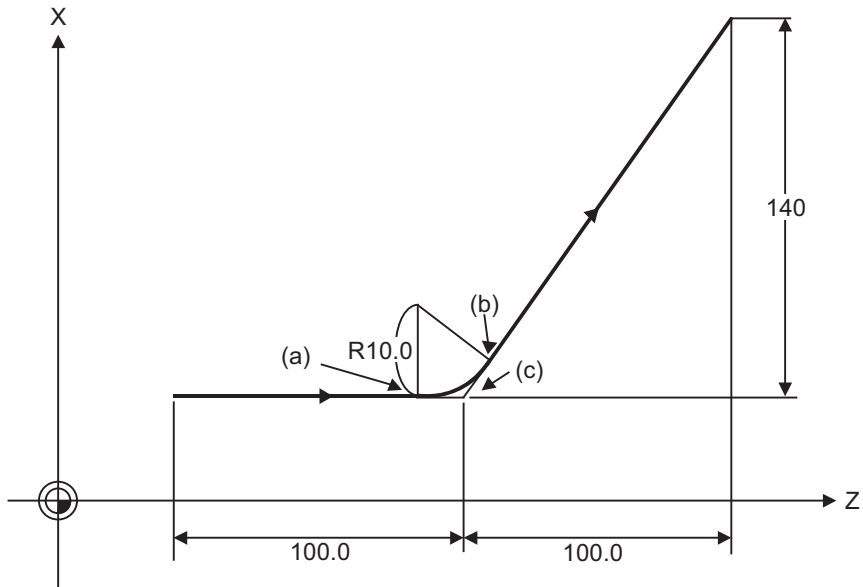
#### Detailed description

- (1) The start point of the block following the corner rounding is the hypothetical corner intersection point.
- (2) The ",R" command will be interpreted as a R command if there is no "," (comma).
- (3) When both the corner chamfering/corner rounding commands exist 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) 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.
- (6) Program error (P383) will occur when the movement amount in the corner rounding block is less than the R value.
- (7) Program error (P384) will occur when the movement amount is less than the R value in the block following the corner rounding.
- (8) Program error (P382) will occur if a movement command is not issued in the block following corner rounding.



Program example

```
G01 W100. ,R10. F100 ;
U280. W100. ;
```



(a) Corner rounding start point

(b) Corner rounding end point

(c) Hypothetical corner intersection point

### 13.8.3 Interrupt during Corner Chamfering/Corner Rounding

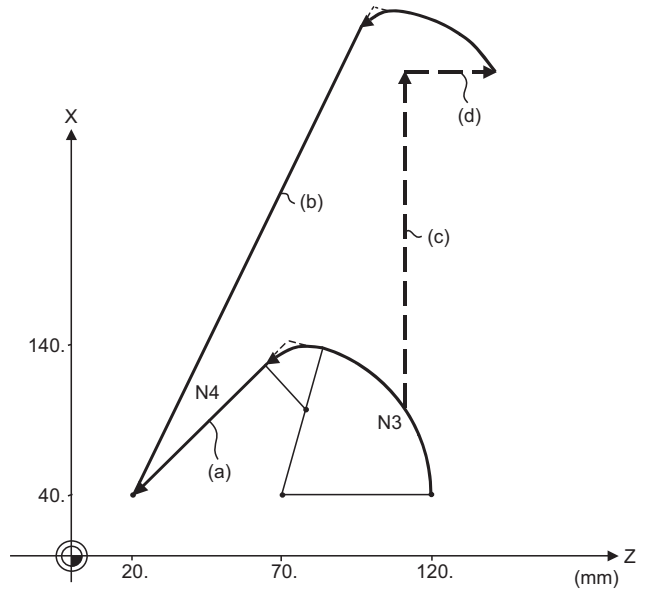


#### Detailed description

- (1) Shown below are the operations of manual interruption during corner chamfering or corner rounding.

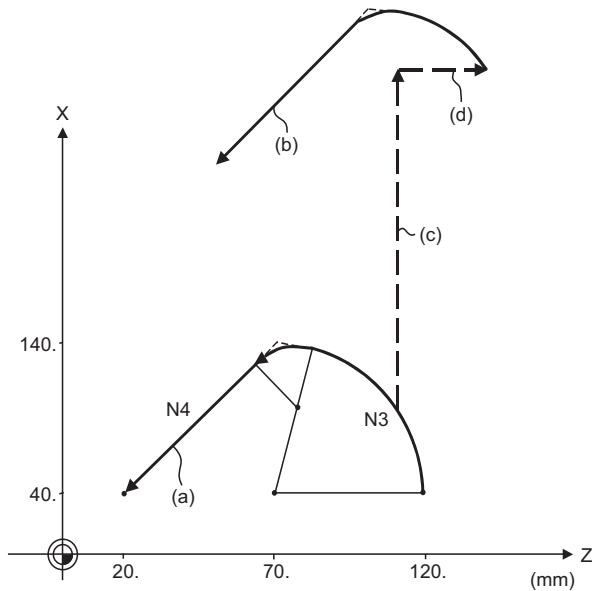
With an absolute value command and manual absolute switch ON.

```
N1 G28 XZ;
N2 G00 X40. Z120.;
N3 G03 X140.Z70. K-50. ,R20. F100 ;
N4 G01 X40. Z20. ;
:
```



With an incremental value command and manual absolute switch OFF

```
N1 G28 XZ;
N2 G00 U40. W120.;
N3 G03 U100. W-50. K-50. ,R20. F100 ;
N4 G01 U-100.W-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.

## 13.9 Corner Chamfering/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.

### 13.9.1 Corner Chamfering II ; G01/G02/G03 X_ Z_ ,C



#### Function and purpose

The corner is chamfered by commanding ",C" (or "I_", "K_", "C_") in the 1st block of two blocks with continuous arcs. For an arc, this will be the chord length.



#### Command format

```
N100 G03 X__ Z__ I__ K__ ,C__ ;
N200 G01 X__ Z__ ;
```

.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) The ",C" command will be interpreted as a C command if there is no "," (comma).
- (4) If there are multiple or duplicate corner chamfering commands in a same block, the last command will be valid.
- (5) When both corner chamfering and corner rounding are commanded in the same block, the latter command will be valid.
- (6) Tool compensation is calculated for the shape which has already been subjected to corner chamfering.
- (7) Program error (P385) will occur when positioning or thread cutting is commanded in the corner chamfering command block or in the next block.
- (8) Program error (P382) will occur when the block following corner chamfering contains a G command other than group 01 or another command.
- (9) Program error (P383) will occur when the movement amount in the block, commanding corner chamfering, is less than the chamfering amount.
- (10) Program error (P384) will occur when the movement amount is less than the chamfering amount in the block following the block commanding corner chamfering.
- (11) Even if a diameter is commanded, it will be handled as a radial command value during corner chamfering.
- (12) Program error (P382) will occur when a movement command is not issued in the block following the corner chamfering II command.



Program example

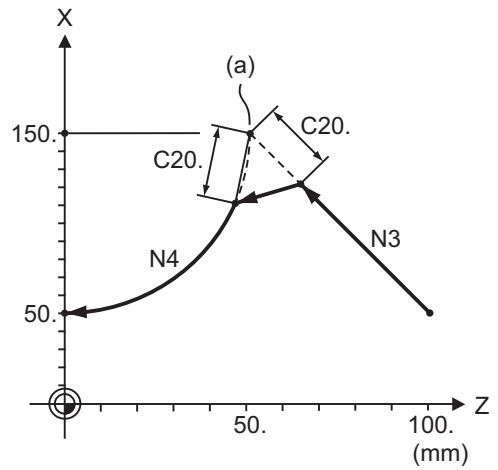
(1) Linear - arc

Absolute value command

```
N1 G28 X Z ;
N2 G00 X50. Z100. ;
N3 G01 X150. Z50. ,C20. F100 ;
N4 G02 X50. Z0 I0 K-50. ;
:
```

Relative value command

```
N1 G28 X Z ;
N2 G00 U25. W100. ;
N3 G01 U50. W-50. ,C20. F100 ;
N4 G02 U-50. W-50. I0 K-50. ;
:
```



(a) Hypothetical corner intersection point

### 13.9.2 Corner Rounding II ; G01/G02/G03 X_ Z_ ,R_



#### Function and purpose

The corner is rounded by commanding ",R_" (or "R_") in the 1st block of two blocks containing continuous arcs.



#### Command format

```
N100 G03 X__ Z__ I__ K__ ,R__;  
N200 G01 X__ Z__ ;
```

,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) The ",R" command will be interpreted as a R command if there is no "," (comma).
- (4) When both corner chamfering and corner rounding are commanded in a same block, the latter command will be valid.
- (5) Tool compensation is calculated for the shape which has already been subjected to corner rounding.
- (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 corner rounding block is less than the R value.
- (9) Program error (P384) will occur when the movement amount is less than the R value in the block following the corner rounding.
- (10) Even if a diameter is commanded, it will be handled as a radial command value during corner chamfering.
- (11) A program error (P382) will occur if a movement command is not issued in the block following corner rounding.



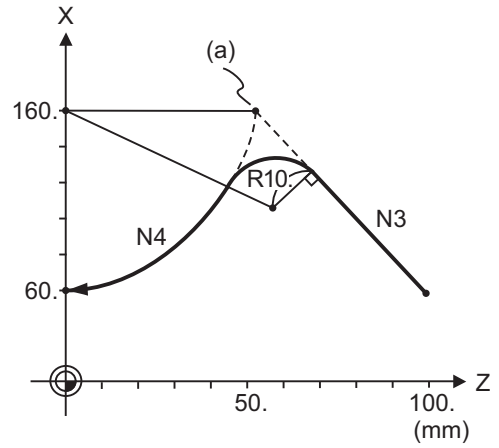


Program example

(1) Linear - arc

```

Absolute value command
N1 G28 X Z ;
N2 G00 X60. Z100. ;
N3 G01 X160. Z50. ,R10. F100 ;
N4 G02 X60. Z0 I0 K-50. ;
:
Relative value command
N1 G28 X Z ;
N2 G00 U30. W100. ;
N3 G01 U50. W-50. ,R10. F100 ;
N4 G02 U-50. W-50. I0 K-50. ;
:
    
```



(a) Hypothetical corner intersection point

### 13.9.3 Interrupt during Corner Chamfering/Corner Rounding

For details, refer to "Corner Chamfering/Corner Rounding I: Interrupt during Corner Chamfering/Corner Rounding".

## 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.

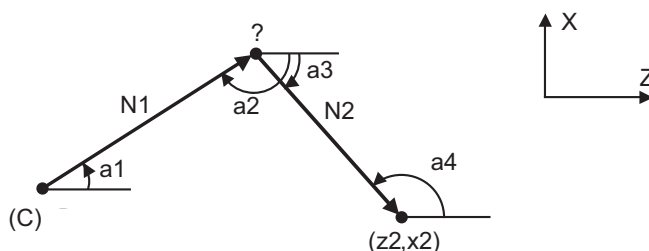
(Note) If the parameter "#1082 Geomet" is set to 0, geometric I will not function.



#### Command format

```
N1 G01 Aa1 (A-a2) Ff1;
N2 Xx2 Zz2 Aa4 (A-a3) Ff2;
```

Aa1, A-a2, A-a3, Aa4	Angle
Ff1, Ff2	Feedrate
Xx2, Zz2	Next block end point coordinates



(C) Present 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 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.
- (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 Zz2 Aa2 ;</p>	
<p>(Example 2)                  N1 Aa1 ,Rr1 ;                  N2 Xx2 Zz2 Aa2 ;</p>	

(2) The geometric command I can be issued after the corner chamfering or corner rounding command.

<p>(Example 3)                  N1 Xx2 Zz2 ,Cc1 ;                  N2 Aa1 ;                  N3 Xx3 Zz3 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 Zz3 Aa3 ;</p>	
--------------------------------------------------------------------------------------------------------------------------	--

### 13.10.2 Geometric IB

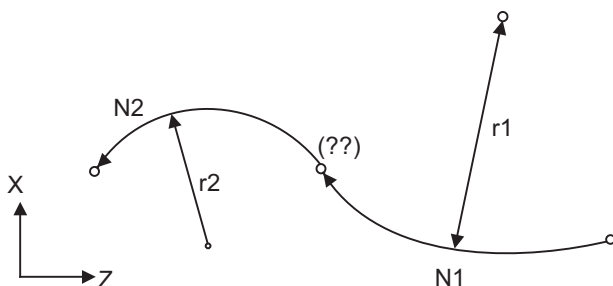


#### Function and purpose

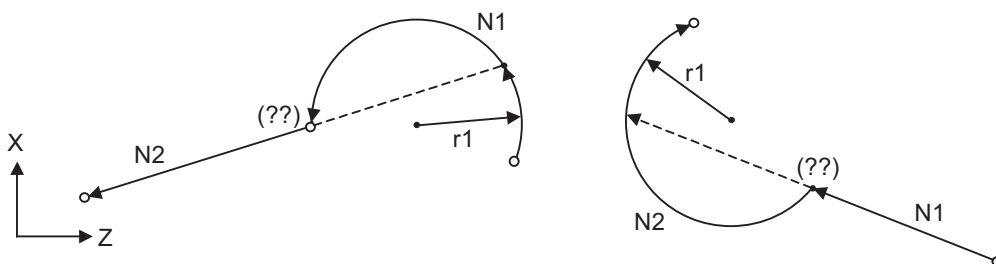
With the geometric IB function, the contact and intersection are calculated by commanding a arc center point or linear angle in the movement commands of two continuous blocks (only blocks with arc commands), instead of commanding the first block end point.

(Note) If the parameter (#1082 Geomet) is not set to 2, geometric IB will not function.

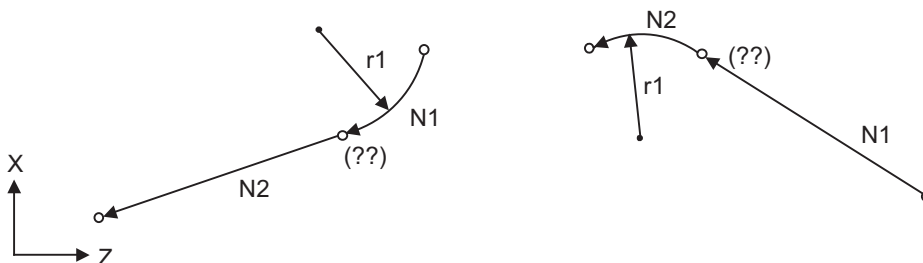
#### Two-arc contact



#### 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 Zz2 Pp2 Qq2 Ff2;

N1 G02(G03) Pp1 Qq1 Ff1;  
N2 G03(G02) Xx2 Zz2 Rr2 Ff2;

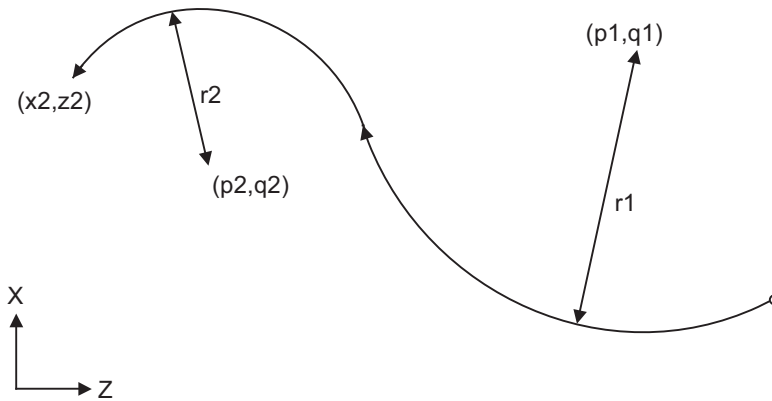
N1 G02(G03) Rr1 Ff1;  
N2 G03(G02) Xx2 Zz2 Pp2 Qq2 Ff2;

P,Q	X and Z 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 K (X and Z axes arc center coordinate incremental value) commands can be issued instead of P and Q.

1st block arc : Radius command incremental amount from the start point to the center

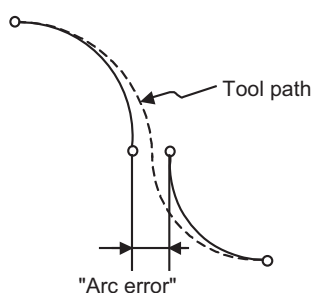
2nd block arc : Radius command incremental amount from the end point to the center



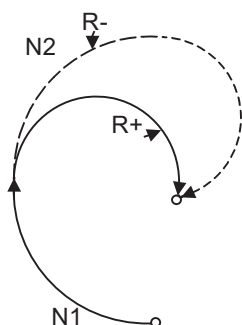


### Detailed description

- (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, K) or P, Q (I, K) 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 K is omitted, the values are regarded as I0 and K0. P and Q cannot be omitted.
- (9) The error range in which the contact is obtained is set in parameter "#1084 RadErr".



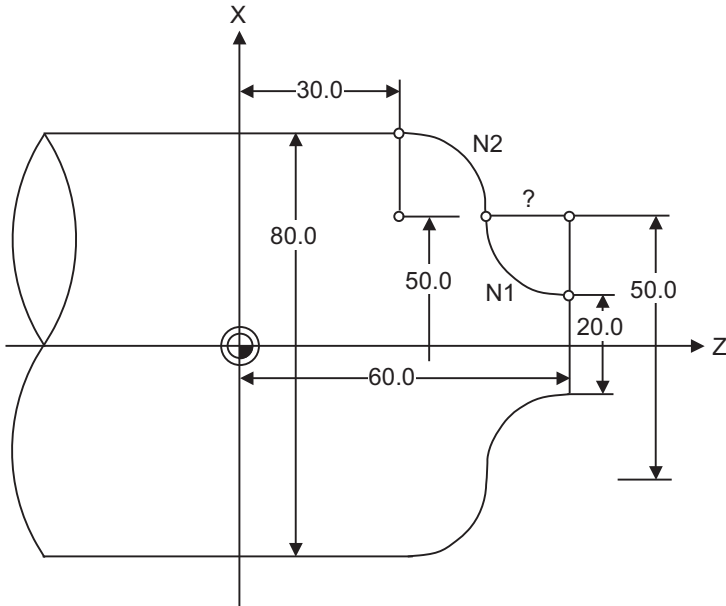
- (10) For an arc block perfect circle command (arc block start point = arc block end point), the R designation arc command finishes immediately, and there is no operation. Thus, use a PQ (IK) designation arc command.
- (11) G codes of the G modal group 1 in the 1st/2nd block can be omitted.
- (12) Addresses 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.





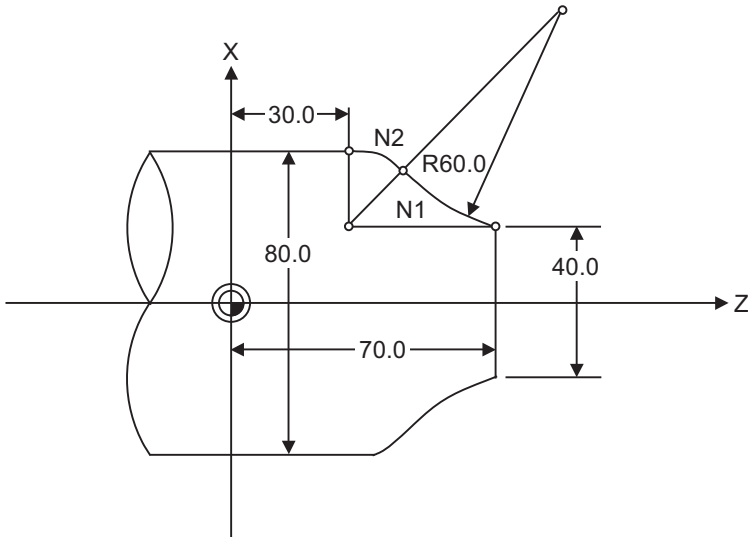
Program example

(1) PQ, PQ command



```
G01 X20.0 Z60.0;
N1 G02 P50.0 Q60.0 F100;
N2 G03 X80.0 Z30.0 P50.0 Q30.0;
(mm)
```

(2) PQ, R command



```
G01 X40.0 Z70.0 F100;
N1 G02 R60.0;
N2 G03 X80.0 Z30.0 P40.0 Q30.0;
(mm)
```



## Relation with other functions

Command	Tool path
Geometric IB + corner chamfering II N1 G09 P_Q_ ; N2 G02 X_Z_R_ ,C_ ; G02 X_Z_R_ ;	
Geometric IB + corner rounding II N1 G03 P_Q_ ; N2 G02 X_Z_R_ ,R_ ; G02 X_Z_R_ ;	
Geometric IB + corner chamfering II N1 G03 P_Q_ ; N2 G02 X_Z_R_ ,C_ ; G01 X_Z_ ;	
Geometric IB + corner rounding II N1 G03 P_Q_ ; N2 G02 X_Z_R_ ,R_ ; G01 X_Z_ ;	



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.



N1 G01 Aa1(A-a2) Ff1 ;  
N2 G02(G03) Xx2 Zz2 Pp2 Qq2 Hh2 Ff2 ;

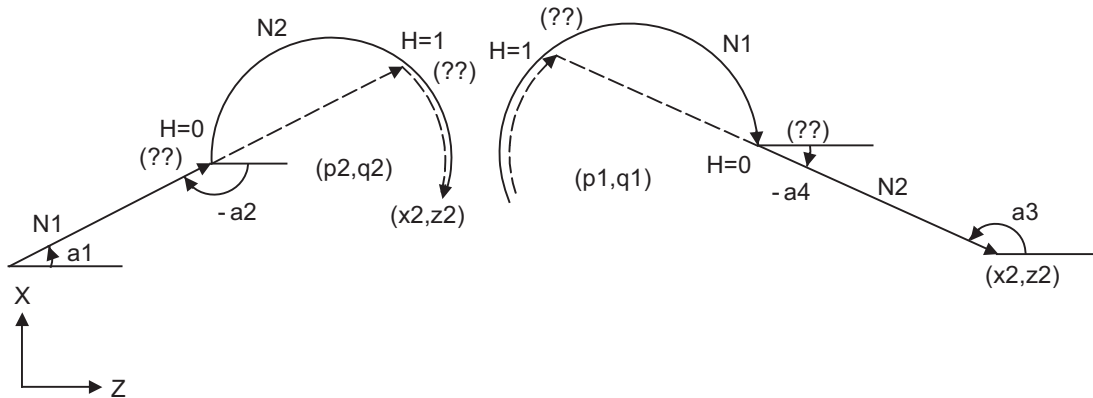
N1 G02(G03) Pp1 Qq1 Hh1 Ff1 ;  
N2 G1 Xx2 Zz2 Aa3 (A-a4) Ff2 ;

A	Linear angle (-360.000° to 360.000° )
P,Q	X and Z axes circular center coordinate absolute value (diameter/radius value command) The center address for the 3rd axis is commanded with A.
H	Selection of linear - arc intersection 0: Intersection of the shorter line 1: Intersection of the longer line

* I and K (X and Z axes arc center coordinate incremental value) commands can be issued instead of P and Q.

1st block arc : Radius command incremental amount from the start point to the center

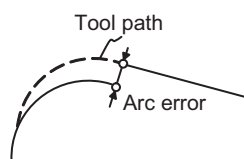
2nd block arc : Radius command incremental amount from the end point to the center



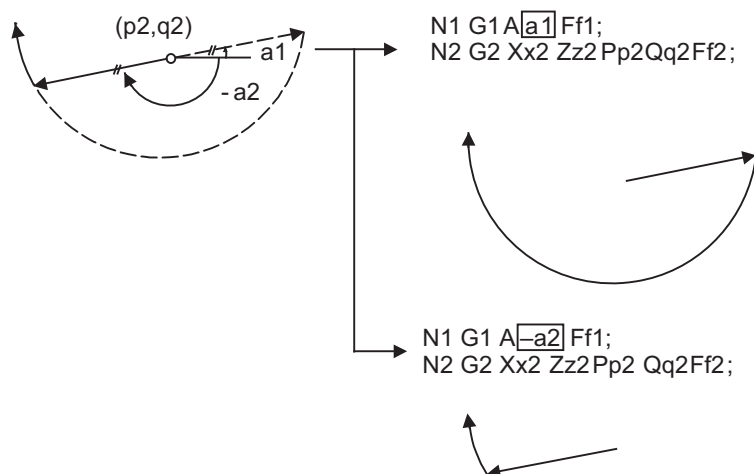


## Detailed description

- (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, K) 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.



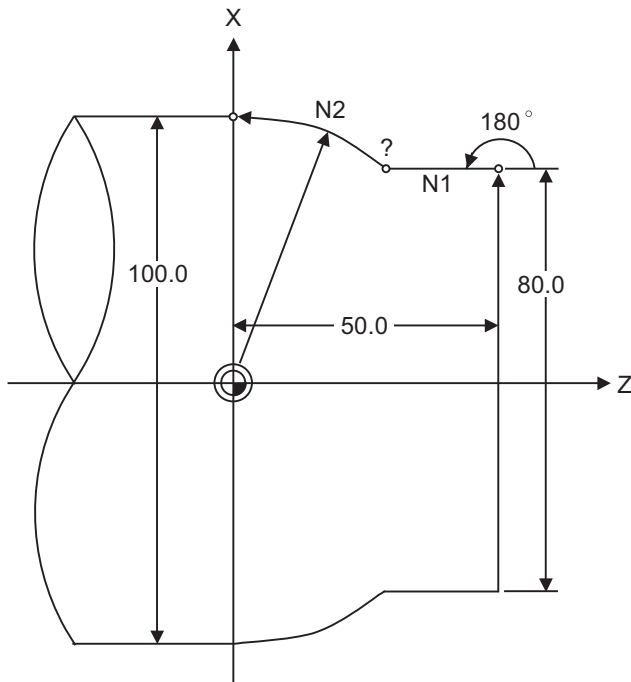
- (7) Single block operation stops at the 1st block.
- (8) When I or K is omitted, the values are regarded as I0 and K0. P and Q cannot be omitted.
- (9) When H is omitted, the value is regarded as H0.
- (10) The linear - arc contact is automatically calculated by designating R instead of P, Q (I, K).
- (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  $\pm 1 \mu\text{m}$  (fractions rounded up).
- (16) In linear - arc intersections, the arc command can only be PQ (IK) 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.



Program example



```
G01 X80.0 Z50.0 F100;
N1 G01 A180.0;
N2 G03 X100.0 Z0 P0 Q0;           (mm)
```



## Relation with other functions

Command	Tool path
Geometric IB + corner chamfering II N1 G01 A_,C_ ; N2 G03 X_Z_P_Q_H_ ;	
Geometric IB + corner rounding II N1 G01 A_,R_ ; N2 G03 X_Z_P_Q_H_ ;	
Geometric IB + corner chamfering II N1 G01 A_ ; N2 G03 X_Z_P_Q_H_ ; G01 X_Z_ ;	
Geometric IB + corner rounding II N1 G01 A_ ; N2 G03 X_Z_P_Q_H_ ; G01 X_Z_ ;	
Geometric IB + corner chamfering N1 G02 P_Q_H_ ; N2 G01 X_Z_A_,C_ ; G01 X_Z_ ;	
Geometric IB + corner rounding II N1 G02 P_Q_H_ ; N2 G01 X_Z_A_,R_ ; G01 X_Z_ ;	

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.

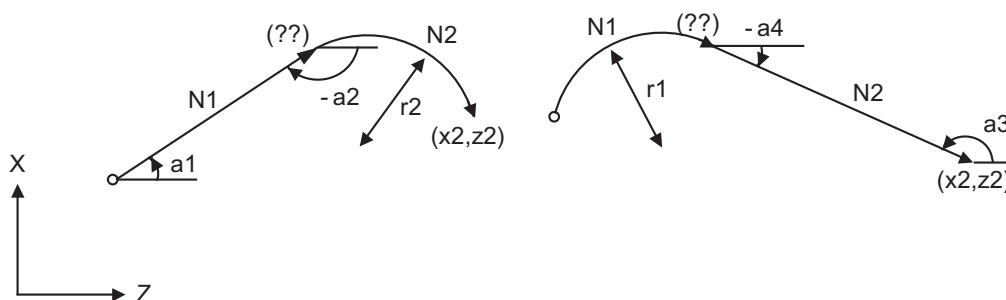


Command format (For G18 plane)

N1 G01 Aa1(A-a2) Ff1;  
N2 G03(G02) Xx2 Zz2 Rr2 Ff2;

N1 G03(G02) Rr1 Ff1;  
N2 G01 Xx2 Zz2 Aa3(A-a4)Ff2;

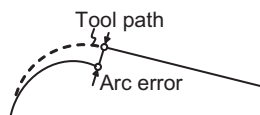
A	Linear angle (-360.000° to 360.000° )
R	Arc radius





### Detailed description

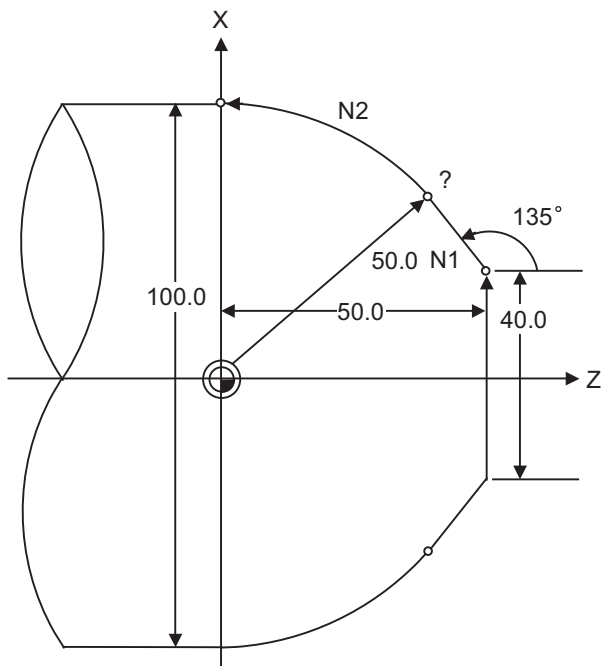
- (1) When the 2nd miscellaneous function address is A, the 2nd miscellaneous function is validated and this function is invalidated.
- (2) Program error (P393) will occur before the 1st block if the 2nd block is not a coordinate absolute value command.
- (3) Program error (P398) will occur before the 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, K).



- (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.



Program example



```
G01 X40.0 Z50.0 F100;
N1 G01 A135.0;
N2 G03 X100.0 Z0.0 R50.0;          (mm)
```



## Relation with other functions

Command	Tool path
Geometric IB + corner chamfering N1 G03 R_ N2 G01 X_Z A_C_ G01 X_Z_	
Geometric IB + corner rounding N1 G03 R_ N2 G01 X_Z A_R_ G01 X_Z_	
Geometric IB + corner chamfering II N1 G01 A_ N2 G02 X_Z R_C_ G01 X_Z_	
Geometric IB + corner rounding II N1 G01 A_ N2 G02 X_Z R_R_ G01 X_Z_	



## 13.11 Programmable Parameter Input ; G10 L70, G11



### Function and purpose

The parameters set from the setting and display unit can be changed in the machining programs.



### Command format

#### G10 L70 ;...Data setting start command

P__ S__ A__ H □ __ ; ..... Bit parameter  
 P__ S__ A__ D__ ; ..... Numerical value parameter  
 P__ S__ A__ <character string> ; ..... Character string parameter  
 P__ S__ A__ ,character string; ..... Character string parameter ...

P	Parameter No.
S	Part system No.
A	Axis No.
H	Data
D	Data
<character string>	
,character string	

#### G11 ; ... Data setting end command

- (Note 1) The sequence of addresses in a block must be as shown above.  
When an address is commanded two or more times, the last command will be valid.
- (Note 2) The part system No. is set in the following manner. "1" for the 1st part system, "2" for 2nd part system, and so forth.  
If the address S is omitted, the part system of the executing program will be applied.  
As for the parameters common to part systems, the command of part system No. will be ignored.
- (Note 3) The axis No. is set in the following manner. "1" for 1st axis, "2" for 2nd axis, and so forth.  
If the address A is omitted, the 1st axis will be applied.  
As for the parameters common to axes, the command of axis No. will be ignored.
- (Note 4) Address H is commanded with the combination of setting data (0 or 1) and the bit designation □ (0 to 7).
- (Note 5) Only the decimal number can be commanded with the address D.  
The value that is smaller than the input setting unit (#1003 iunit) will be round off to the nearest increment.
- (Note 6) Character strings must be with "," or put in angled brackets "<" and ">" when commanded.  
If these brackets are not provided, the program error (P33) will occur.  
Up to 31 characters can be set.
- (Note 7) Command G10 L70, G11 in independent blocks. A program error (P33, P421) will occur if not commanded in independent blocks.
- (Note 8) When the data with a decimal point is commanded without a decimal point, it will be handled as decimal point valid.



## Program example

G10 L70;	
P6401 H71 ;	Sets "1" to "#6401 bit7".
P8204 S1 A2 D1.234 ;	Sets "1.234" to "#8204 of the 1st part system 2nd axis".
P7501 <X> ;	Sets "X" to "#7501".
G11 ;	

## 13.12 Macro Interruption ; M96,M97



### Function and purpose

A user macro interrupt signal (UIT) is input from the machine to interrupt the program being currently executed and instead call another program and execute it. This is called the user macro interrupt function. Use of this function allows the program to operate flexibly enough to meet varying conditions.



### Command format

#### M96 P__ H__ ; ... User macro interruption enable

M96	User macro interruption command
P	Interrupt program No.
H	Interrupt sequence No.

#### M97 ; ... User macro interruption disable

M97	User macro interruption end command
-----	-------------------------------------



### Detailed description

The user macro interrupt function is enabled and disabled by the M96 and M97 commands programmed to make the user macro interrupt signal (UIT) valid or invalid. That is, if an interrupt signal (UIT) is input from the machine side in a user macro interruption enable period from when M96 is issued to when M97 is issued or the NC is reset, a user macro interruption is caused to execute the program specified by P__ instead of the one being executed currently.

Another interrupt signal (UIT) is ignored while one user macro interrupt is being in service. It is also ignored in a user macro interrupt disable state such as after an M97 command is issued or the system is reset.

M96 and M97 are processed internally as user macro interrupt control M codes.

## Interrupt enable conditions

A user macro interruption is enabled only during execution of a program.

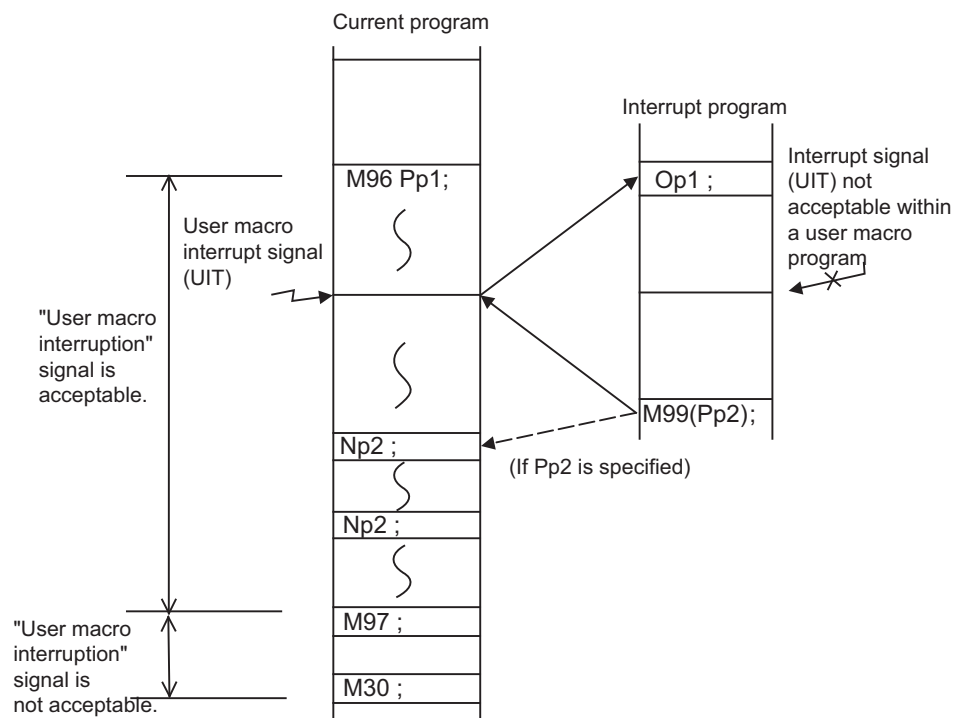
The requirements for the user macro interrupt are as follows:

- (1) A memory operation mode or MDI has been selected.
- (2) The system is running in automatic mode.
- (3) No user macro interruption is being processed.

(Note 1) A macro interruption is disabled in manual operation mode (JOG, STEP, HANDLE, etc.)

## Outline of operation

- (1) When a user macro interrupt signal (UIT) is input after an M96Pp1 ; command is issued by the current program, interrupt program Op1 is executed. When an M99 ; command is issued by the interrupt program, control returns to the main program.
- (2) If M99Pp2 ; is specified, the blocks from the one next to the interrupted block to the last one are searched for the block with sequence number Np2 ;. Control thus returns to the block with sequence number Np2 ; that is found first in the above search.



## Interrupt type

Interrupt types 1 and 2 can be selected by the parameter "#1113 INT_2".

### [Type 1]

- (1) When an interrupt signal (UIT) is input, the system immediately stops moving the tool and interrupts dwell, then permits the interrupt program to run.
- (2) If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are lost. After the interrupt program completes, the main program resumes operation from the block next to the interrupted one.
- (3) If the interrupted program contains no move and miscellaneous (MSTB) commands, it resumes operation, after completion of the interrupt program, from the point in the block where the interrupt was caused.

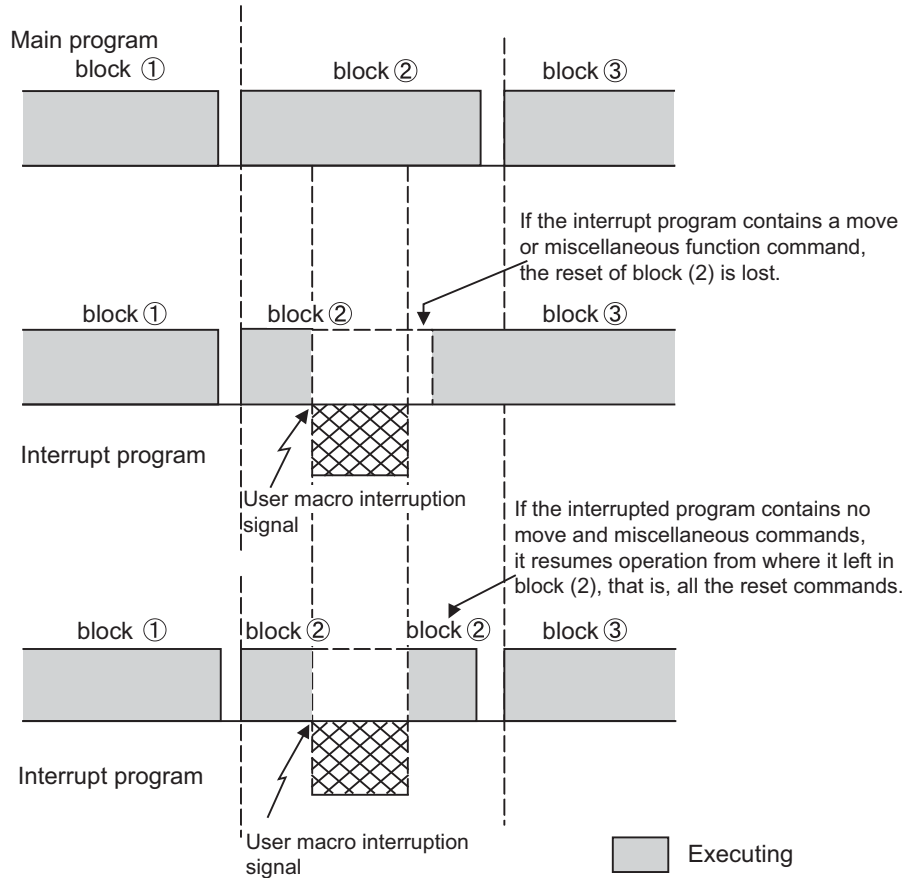
If an interrupt signal (UIT) is input during execution of a miscellaneous function (MSTB) command, the NC system waits for a completion signal (FIN). The system thus executes a move or miscellaneous function command (MSTB) in the interrupt program only after input of FIN.

### [Type 2]

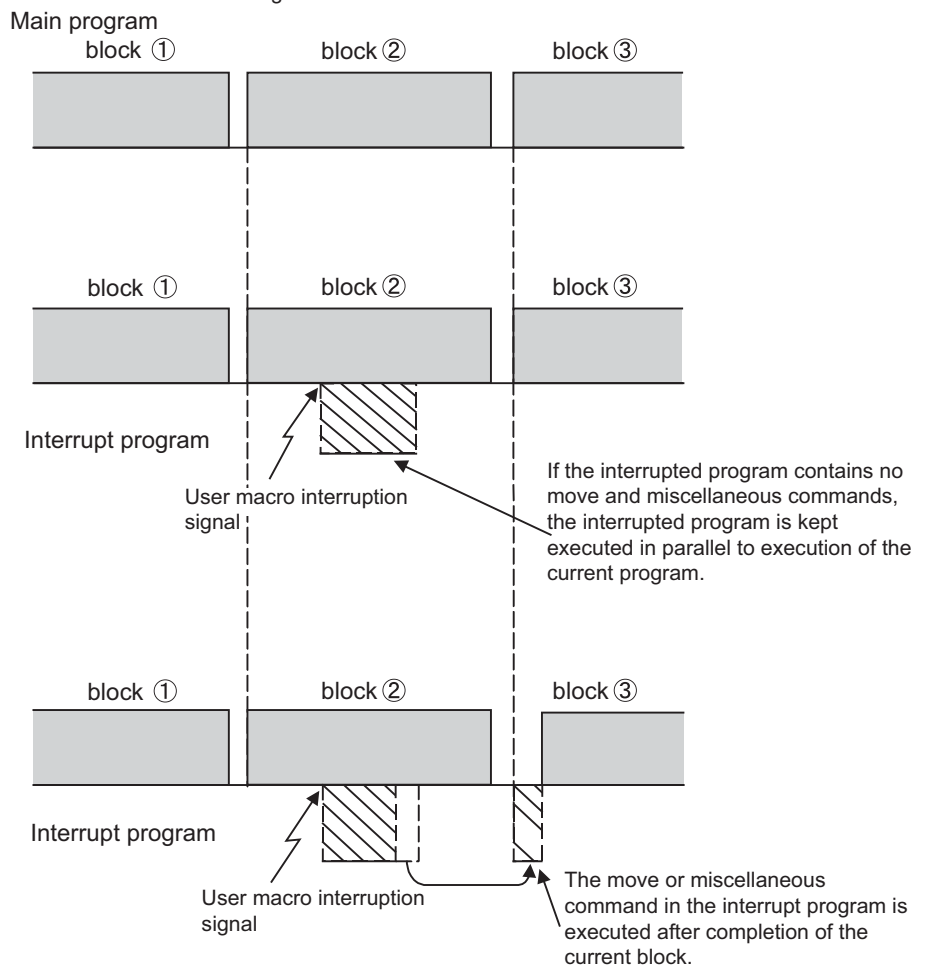
- (1) When an interrupt signal (UIT) is input, the program completes the commands in the current block, then transfers control to the interrupt program.
- (2) If the interrupt program contains no move and miscellaneous function (MSTB) commands, the interrupt program is executed without interrupting execution of the current block.

However, if the interrupt program has not ended even after the execution of the original block is completed, the system may stop machining temporarily.

[Type 1]



[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 "#1229 set01/bit0". Both types of interrupt are added to the calculation of the nest level. The subprograms and user macros called in the interrupt program are also added to the calculation of the nest level.

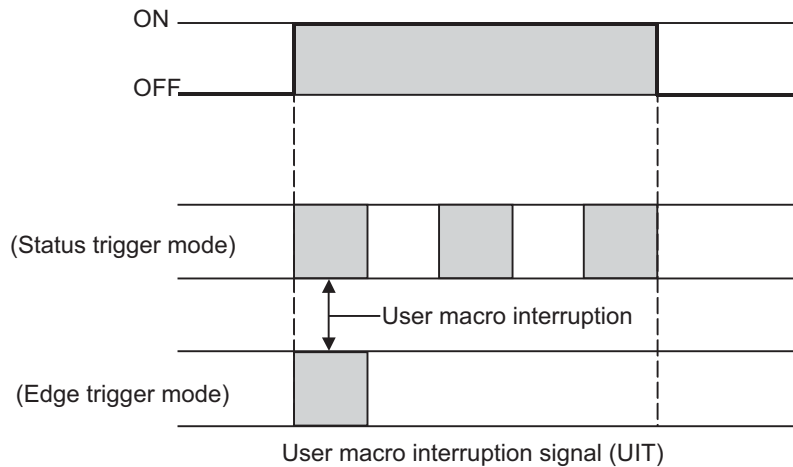
Subprogram type interrupt	The user macro interruption program is called as a subprogram. As with calling by M98, the local variable level remains unchanged before and after an interrupt.
Macro type interruption	The user macro interpretation program is called as a user macro. As with calling by G65, the local variable level changes before and after an interrupt. No arguments in the main program can be passed to the interrupt program.

**Acceptance of user macro interruption signal (UIT)**

A user macro interruption signal (UIT) is accepted in the following two modes: These two modes are selected by a parameter "#1112 S_TRG".

Status trigger mode	The user macro interruption signal (UIT) is accepted as valid when it is ON. If the interrupt signal (UIT) is ON when the user macro interrupt function is enabled by M96, the interrupt program is activated. By keeping the interrupt signal (UIT) ON, the interrupt program can be executed repeatedly.
Edge trigger mode	The user macro interrupt signal (UIT) is accepted as valid at its rising edge, that is, at the instance it turns ON. This mode is useful to execute an interrupt program once.

User macro interruption signal (UIT)



**Returning from user macro interruption**

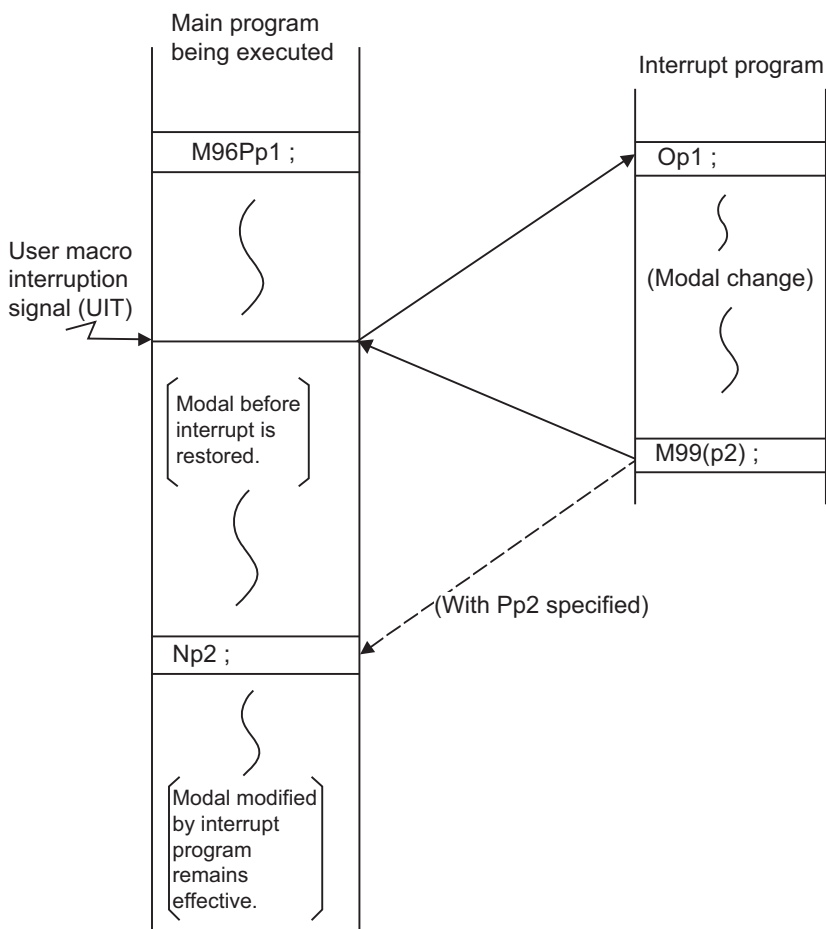
**M99 (P__);**

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.	
#4519	S code	
#4520	T code	

The above system variables are available only in the user macro interrupt program.

If they are used in other programs, program error (P241) will occur.

**M code for control of user macro interruption**

The user macro interruption is controlled by M96 and M97. However, these commands may have been used for other operation. To be prepared for such case, these command functions can be assigned to other M codes.

(This invalidates program compatibility.)

User macro interrupt control with alternate M codes is possible by setting the alternate M code in parameters "#1110 M96_M" and "#1111 M97_M" and by validating the setting by selecting parameter "#1109 subs_M". (M codes 03 to 97 except 30 are available for this purpose.)

If the parameter "#1109 subs_M" used to enable the alternate M codes is not selected, the M96 and M97 codes remain effective for user macro interrupt control.

In either case, the M codes for user macro interrupt control are processed internally and not output to the outside.

## Parameters

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

- (1) Subprogram call validity "#1229 set 01/bit 0"  
 1: Subprogram type user macro interruption  
 0: Macro type user macro interruption
- (2) Status trigger mode validity "#1112 S_TRG"  
 1: Status trigger mode  
 0: Edge trigger mode
- (3) Interrupt type 2 validity "#1113 INT_2"  
 1: The executable statements in the interrupt program are executed after completion of execution of the current block. (Type 2)  
 0: The executable statements in the interrupt program are executed before completion of execution of the current block. (Type 1)
- (4) Validity of alternate M code for user macro interruption control "#1109 subs_M"  
 1: Valid  
 0: Invalid
- (5) Alternate M codes for user macro interruption  
 Interrupt enable M code (equivalent to M96) "#1110 M96_M"  
 Interrupt disable M code (equivalent to M97) "#1111 M97_M"  
 M codes 03 to 97 except 30 are available.



## Precautions

- (1) If the user macro interruption program uses system variables #5001 and after (position information) to read coordinates, the coordinates pre-read in the buffer are used.
- (2) If an interrupt is caused during execution of the tool radius compensation, a sequence No. (M99P__;) must be specified with a command to return from the user macro interrupt program. If no sequence No. is specified, control cannot return to the main program normally.

### 13.13 Tool Change Position Return ; G30.1 - G30.5



#### 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 5: 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	X axis only (-> additional axis)
G30.2	Z axis only (-> additional axis)
G30.3	X axis -> Z axis (-> additional axis)
G30.4	Z axis -> X axis (-> additional axis)
G30.5	X axis - Z axis (-> additional axis)

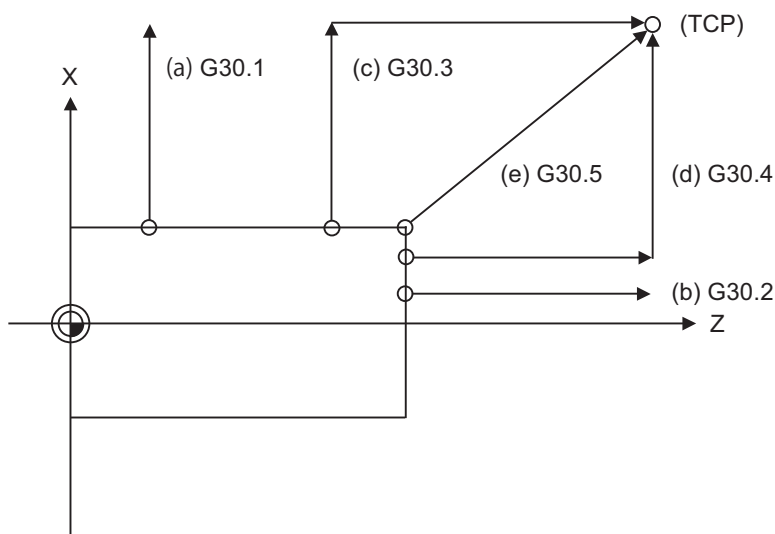
(Note 1) An arrow (->) indicates the order of axes that begin to return. An period ( - ) indicates that the axes begin to return simultaneously. (Example: "Z axis -> X axis" indicates that the Z axis returns to the tool change position, then the X axis does.)

- (1) The tool change position return on/off for the additional axis can be set with parameter "#1092 Tchg_A" for the additional axis.  
 For the order for returning to the tool change position, the axes return after the standard axis completes the return to the tool change position (refer to above table). For specifications having two additional axes, the two additional axes simultaneously return to the tool change position after the standard axis has finished its return to the tool change position.  
 The additional axis alone cannot return to the tool change position.



### Operation example

(1) The following figure shows axes during a tool change position return command.

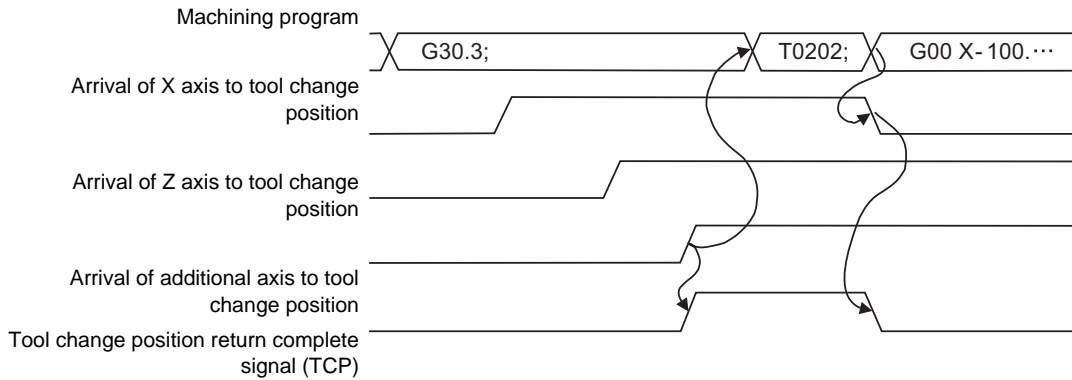


TCP : Tool change position

- (a) G30.1 command: X axis only returns to the tool change position. (If the tool change position return is validated for the additional axis, the additional axis also returns to the tool change position after the X axis reaches the tool change position.)
- (b) G30.2 command: Z axis only returns to the tool change position. (If the tool change position return is validated for the additional axis, the additional axis also returns to the tool change position after the Z axis reaches the tool change position.)
- (c) G30.3 command : X axis returns to the tool change position, then Z axis does the same thing. (If the tool change position return is validated for the additional axis, the additional axis also returns to the tool change position after the X and Z axes reache the tool change position.)
- (d) G30.4 command : Z axis returns to the tool change position, then X axis does the same thing. (If the tool change position return is validated for the additional axis, the additional axis also returns to the tool change position after the Z and X axes reache the tool change position.)
- (e) G30.5 command : X and Z axes return to the tool change position simultaneously. (If the tool change position return is validated for the additional axis, the additional axis also returns to the tool change position after the Z and X axes reache the tool change position.)

- (2) After all necessary tool change position return is completed by a G30.n command, tool change position return complete signal (TCP) is turned ON. When an axis out of those having returned to the tool change position by a G30.n command leaves the tool change position, the TCP signal is turned OFF. (With a G30.3 command, for example, the TCP signal is turned ON when the Z axis has reached the tool change position after the X axis did (after the additional axis did if additional axis tool change position return is valid)). The TCP signal is then turned OFF when the X or Z axis leaves the position. If tool change position return for additional axes is ON with parameter "#1092 Tchg_A", the TCP signal is turned ON when the additional axis or axes have reached the tool change position after the standard axes did. It is then turned OFF when one of the X, Z, and additional axes leaves the position. )

[TCP signal output timing chart] (G30.3 command with tool change position return for additional axes set ON)



- (3) Tool compensation data such as tool length offset and tool nose wear compensation are temporarily canceled by the tool change position return command. The machine moves to the tool change position set in the parameters, but because the tool compensation amount is stored in the memory, it moves by the next movement command to a position with the tool compensation applied.
- (4) This command is executed by dividing blocks for every axis. If this command is issued during single-block operation, therefore, a block stop occurs each time one axis returns to the tool change position. To make the next axis tool change position return, therefore, a cycle start needs to be specified.

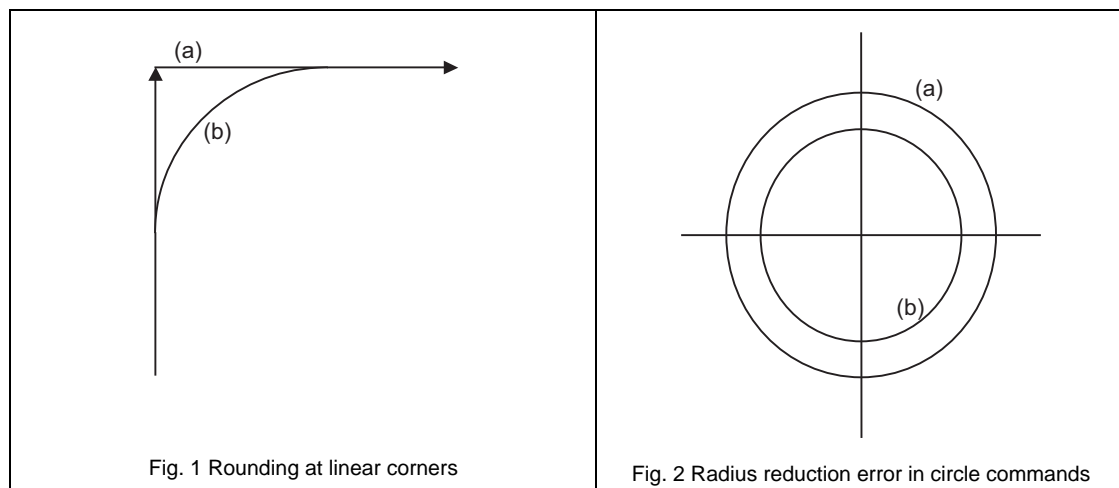
## 13.14 High-accuracy control ; G61.1



### Function and purpose

The following troubles occurred when using normal control:

- (1) Corner rounding occurred at linear and linear-connected corners because the next command movement started before the previous command finished. (Fig. 1)
- (2) When cutting circle commands, an error occurred further inside the commanded path, and the resulting cutting path was smaller than the commanded path. (Fig. 2)



(a) Commanded path (b) Actual path

This function controls the operation so the lag is eliminated in control systems and servo systems. With this function, machining accuracy can be improved, especially during high-speed machining, and machining time can be reduced. The high-accuracy control function is configured of the following functions.

- (1) Pre-interpolation acceleration/deceleration (linear acceleration/deceleration)
- (2) Optimum speed control
- (3) Vector accuracy interpolation
- (4) Feed forward
- (5) Arc entrance/exit speed control
- (6) S-pattern filter control



**Command format**

**G61.1 F__ ; ... High-accuracy control mode ON**

F	Feedrate command
---	------------------

The high-accuracy control mode is validated from the block containing the G61.1 command. This function is valid only for the first part system.

The "G61.1" high-accuracy control mode is canceled with one of the functions of G code group 13.

- G61 (Exact stop check mode)
- G62 (Automatic corner override)
- G63 (Tapping mode)
- G64 (Cutting mode)



**Detailed description**

- (1) The feedrate command F is clamped by the rapid traverse rate or maximum cutting feedrate set with the parameters.
- (2) The modal holding state of the high-accuracy control mode differs according to the combination of the base specification parameter "#1151 rstint" (reset initial) and "#1148 L_G611" (initial high-accuracy).

Parameter		Default state	Resetting		
Reset initial (#1151)	Initial high accuracy (#1148)	Power ON	Reset 1	Reset 2	Reset & rewind
OFF	OFF	OFF	Hold	OFF	
ON			OFF		
OFF	ON	Hold	ON	ON	Hold
ON					

Parameter		Emergency stop	Emergency stop cancel
Reset initial (#1151)	Initial high accuracy (#1148)	Emergency stop switch, External emergency stop	Emergency stop switch, External emergency stop
OFF	OFF	Hold	Hold
ON			OFF
OFF	ON	Hold	Hold
ON			ON

Parameter		Block interruption	Block stop	NC alarm	OT
Reset initial (#1151)	Initial high accuracy (#1148)	Mode changeover (automatic/manual), or Feed hold	Single block	Servo alarm	H/W OT
OFF	OFF	Hold			
ON					
OFF	ON				
ON					

Hold: Modal hold

ON: Switches to high-accuracy mode

As for G61.1, the mode is switched to the high-accuracy mode, even if the other modes (G61 to G64) are valid.

OFF: The status of the high-accuracy mode is OFF.

**Pre-interpolation acceleration/deceleration**

Acceleration/deceleration control is carried out for the movement commands to suppress the impact when the machine starts or stops moving. However, with conventional post-interpolation acceleration/deceleration, the corners at the block seams are rounded, and path errors occur regarding the command shape.

In the high-accuracy control function mode, acceleration/deceleration is carried out before interpolation to solve the above problems. This pre-interpolation acceleration/deceleration enables machining on a machining path that more closely follows the command.

The acceleration/deceleration time can be reduced because constant inclination acceleration/deceleration is carried out.

(1) Basic patterns of acceleration/deceleration control in linear interpolation commands

Acceleration/deceleration waveform pattern	
Normal mode	<p>(F) Speed of each axis (T) Time</p> <p>(a) Because of the constant time constant acceleration/deceleration, the rising edge/falling edge of the waveform becomes more gentle as the command speed becomes slower.</p> <p>(b) The acceleration/deceleration time constant can be independently set for each axis. Linear type, exponential function type, or both can be selected. Note that if the time constant of each axis is not set to the same value, an error will occur in the path course.</p> <p>#2002 clamp : G01 clamp speed                  #2007 G1tL : Linear type acceleration/deceleration time constant                  #2008 G1t1 : Exponential type acceleration/ deceleration time constant</p>
High-accuracy control mode	<p>(f) Combined speed (T) Time</p> <p>(a) Because of the constant inclination type linear acceleration/deceleration, the acceleration/ deceleration time is reduced as the command speed becomes slower.</p> <p>(b) Only one acceleration/deceleration time constant (common for each axis) exists in a system.</p> <p>#2002 clamp : G01 clamp speed                  #1206 G1bF : Target speed                  #1207 G1btL : Acceleration/deceleration time to target speed                  (Note) G1bF and G1btL are values for specifying the inclination of the acceleration/deceleration time. The actual cutting feed maximum speed is clamped by the "#2002 clamp" value.</p>

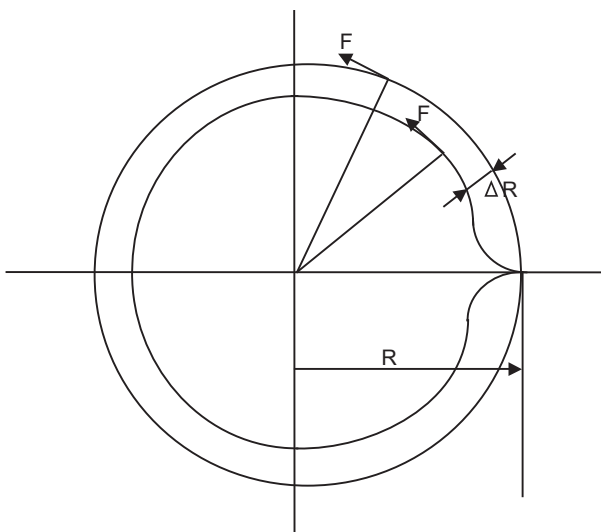


(2) Path control in circular interpolation commands

When commanding circular interpolation with the conventional post-interpolation acceleration/deceleration control method, the path itself that is output from the CNC to the servo runs further inside the commanded path, and the circle radius becomes smaller than that of the commanded circle. This is due to the influence of the smoothing course droop amount for CNC internal acceleration/deceleration.

With the pre-interpolation acceleration/deceleration control method, the path error is eliminated and a circular path faithful to the command results, because interpolation is carried out after the acceleration/deceleration control. Note that the tracking lag due to the position loop control in the servo system is not the target here.

The following shows a comparison of the circle radius reduction error amounts for the conventional post-interpolation acceleration/deceleration control and pre-interpolation acceleration/deceleration control in the high-accuracy control mode.



R : Commanded radius (mm)    ΔR : Radius error (mm)    F : Cutting feedrate (mm/min)

The compensation amount of the circle radius reduction error ( ΔR ) is theoretically calculated as shown in the following table.

Post-interpolation acceleration/deceleration control (normal mode)	Pre-interpolation acceleration/deceleration control (high-accuracy control mode)
Linear acceleration/deceleration $\Delta R = \frac{1}{2R} \left[ \frac{1}{12} T_s^2 + T_p^2 \right] \left( \frac{F}{60} \right)^2$ Exponential function acceleration/deceleration $\Delta R = \frac{1}{2R} \left[ T_s^2 + T_p^2 \right] \left( \frac{F}{60} \right)^2$	Linear acceleration/deceleration $\Delta R = \frac{1}{2R} \left\{ T_p^2 \left[ 1 - K_f^2 \right] \right\} \left( \frac{F}{60} \right)^2$ (a) Because the item $T_s$ can be ignored by using the pre-interpolation acceleration/deceleration control method, the radius reduction error amount can be reduced. (b) Item $T_p$ can be negated by making $K_f = 1$ .

$T_s$  : Acceleration/deceleration time constant in the CNC (s)  
 $T_p$  : Servo system position loop time constant (s)  
 $K_f$  : Feed forward coefficient

### Optimum speed control

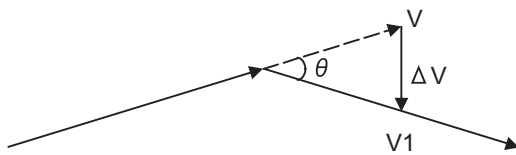
(1) Optimum corner deceleration

By calculating the angle of the seam between blocks, and carrying out acceleration/ deceleration control in which the corner is passed at the optimum speed, highly accurate edge machining can be realized.

When entering in a corner, optimum speed for the corner (optimum corner speed) is calculated from the angle with the next block. The machine decelerates to the speed in advance, and then accelerates back to the command speed after passing the corner.

Corner deceleration is not carried out when blocks are smoothly connected. In this case, the criteria for whether the connection is smooth or not can be designated by the machining parameter "#8020 DCC ANGLE".

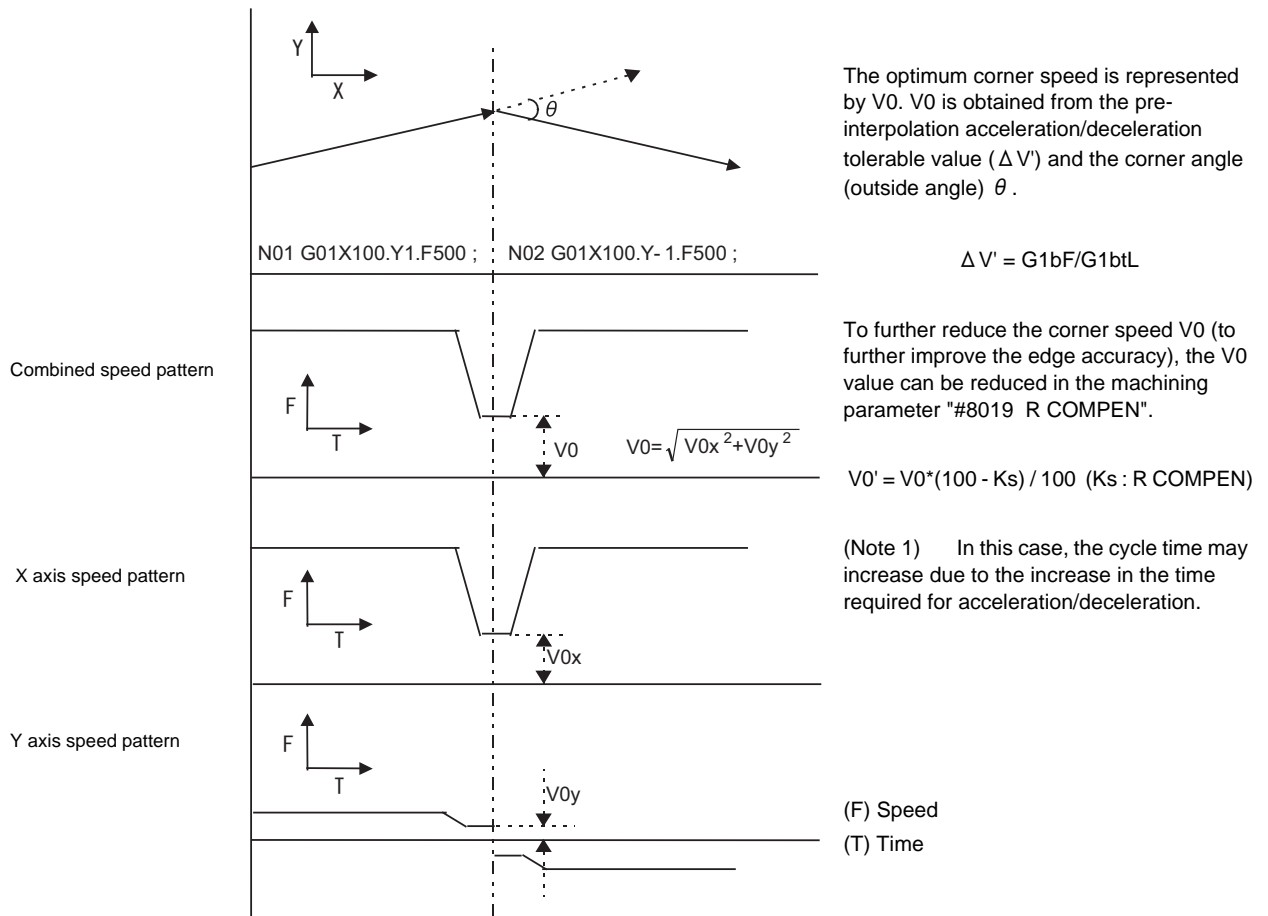
When the corner angle is larger than the parameter "DCC ANGLE" for a linear-linear connection, or for a circle, etc., the acceleration  $\Delta V$  occurs due to the change in the direction of progress after passing the corner at the speed  $V$ .



$V$  : Speed before entering the corner     $\Delta V$  : Speed change at the corner     $V1$  : Speed after passing the corner

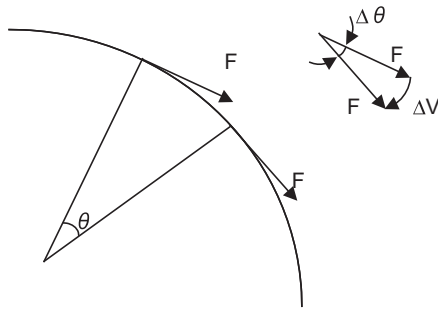
The corner speed  $V$  is controlled so that  $\Delta V$  becomes less than the pre-interpolation acceleration/ deceleration tolerable value set in the parameters ("#1206 G1bF", "#1207 G1btL").

In this case the speed pattern is as follows.



(2) Arc speed clamp

During circular interpolation, even when moving at a constant speed, acceleration is generated as the advance direction constantly changes. When the arc radius is large enough in relation to the commanded speed, control is carried out at the commanded speed. However, when the arc radius is relatively small, the speed is clamped so that the generated acceleration does not exceed the tolerable acceleration/deceleration speed before interpolation, calculated with the parameters. This allows arc cutting to be carried out at an optimum speed for the arc radius.



F : Commanded speed (mm/min)  
 R : Commanded arc radius (mm)  
 $\Delta \theta$  : Angle change per interpolation unit  
 $\Delta V$  : Speed change per interpolation unit  
 The tool is fed with the arc clamp speed  $F'$  so that  $\Delta V$  does not exceed the tolerable acceleration/deceleration speed before interpolation  $\Delta V'$ .

$$F' \leq \sqrt{R * \Delta V' * 60 * 1000} \text{ (mm/min)}$$

$$\Delta V' = \frac{G1bF \text{ (mm/min)}}{G1btL \text{ (ms)}}$$

When the above  $F'$  expression is substituted in the expression for the maximum logical arc radius reduction error amount  $\Delta R$ , explained in the section "Pre-interpolation acceleration/deceleration", the commanded radius  $R$  is eliminated, and  $\Delta R$  does not rely on  $R$ .

$$\Delta R \leq \frac{1}{2R} \left\{ T_p^2 \left[ 1 - K_f^2 \right] \right\} \left[ \frac{F}{60} \right]^2$$

$$\leq \frac{1}{2} \left\{ T_p^2 \left[ 1 - K_f^2 \right] \right\} \left[ \frac{\Delta V' * 1000}{60} \right]$$

$\Delta R$  : Arc radius reduction error amount  
 $T_p$  : Position loop gain time constant of servo system  
 $K_f$  : Feed forward coefficient  
 $F$  : Cutting feedrate

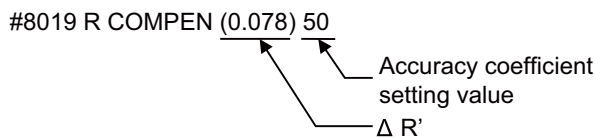
In other words, with the arc command in the high-accuracy control mode, in logical terms regardless of the commanded speed  $F$  or commanded radius  $R$ , machining can be carried out with a radius reduction error amount within a constant value.

To further lower the arc clamp speed (to further improve the roundness), the arc clamp speed can be lowered with the machining parameter "#8019 R COMPEN". In this case, speed control is carried out to improve the maximum arc radius reduction error amount  $\Delta R$  by the set percentage.

$$\Delta R' = \frac{\Delta R * (100 - K_s)}{100} \text{ (mm)}$$

$\Delta R'$  : Maximum arc radius reduction error amount  
 $K_s$  : R COMPEN (%)

After setting the "R COMPEN", the above  $\Delta R'$  will appear on the parameter screen.

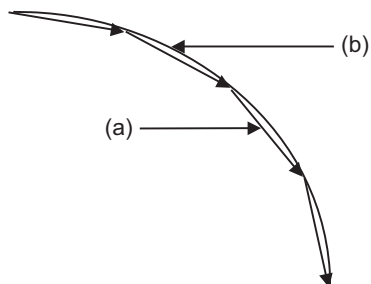


(Note 1) When the "R COMPEN" is set (with a positive value), the arc clamp speed will drop, so in a machining program with many arc commands, the machining time will take longer.

(Note 2) When the "accuracy coefficient" is not set, which means it's set to "0", the arc speed won't be clamped.

### Vector accuracy interpolation

When a fine segment is commanded and the angle between the blocks is extremely small (when not using optimum corner deceleration), interpolation can be carried out more smoothly using the vector accuracy interpolation.

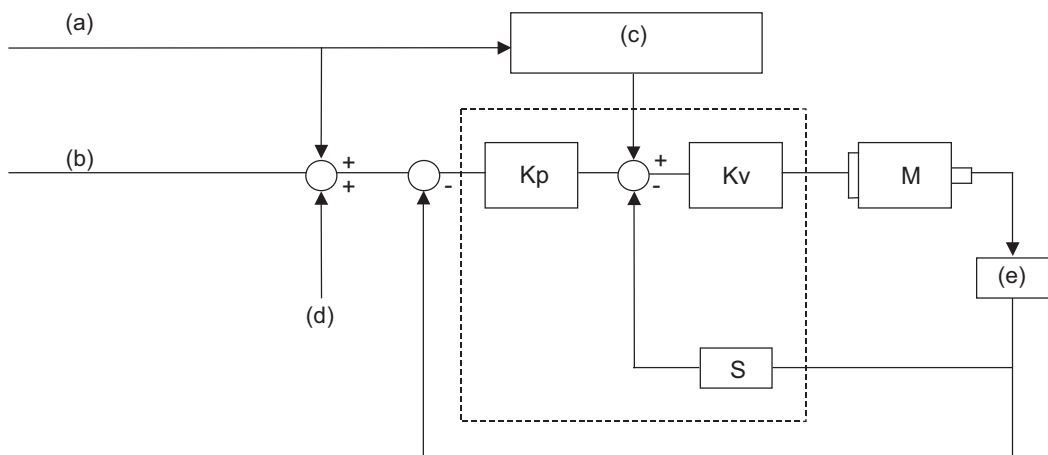


(a) Commanded path (b) Vector accuracy interpolation

### Feed forward control

With this function, the constant speed error caused by the position loop control of the servo system can be greatly reduced. Be noted, though, that, because the feed forward control by its nature induces machine vibration, there are cases when the coefficient cannot be increased.

#### (1) Feed forward control



Kp : Position loop gain    Kv : Speed loop gain    M : Motor    S : Segment

- (a) Command during pre-interpolation acceleration/deceleration
- (b) Command during post-interpolation acceleration/deceleration
- (c) Feed forward control
- (d) Machine error compensation amount
- (e) Detector

(2) Reduction of arc radius reduction error amount using feed forward control

With the high-accuracy control, the arc radius reduction error amount can be greatly reduced by combining the pre-interpolation acceleration/deceleration control method and the feed forward control. The logical radius reduction error amount  $\Delta R$  in the high-accuracy control mode is obtained with the following expression.

$$\Delta R = \frac{1}{2R} T_p^2 \left( 1 - \left( \frac{\text{fwd_g}}{100} \right)^2 \right) \left( \frac{F}{60} \right)^2 \left( 1 - \frac{K_s}{100} \right)$$

R: Arc radius (mm)

F: Cutting feedrate (mm/min)

Tp: Position loop time constant (sec) = the inverse of the position loop gain

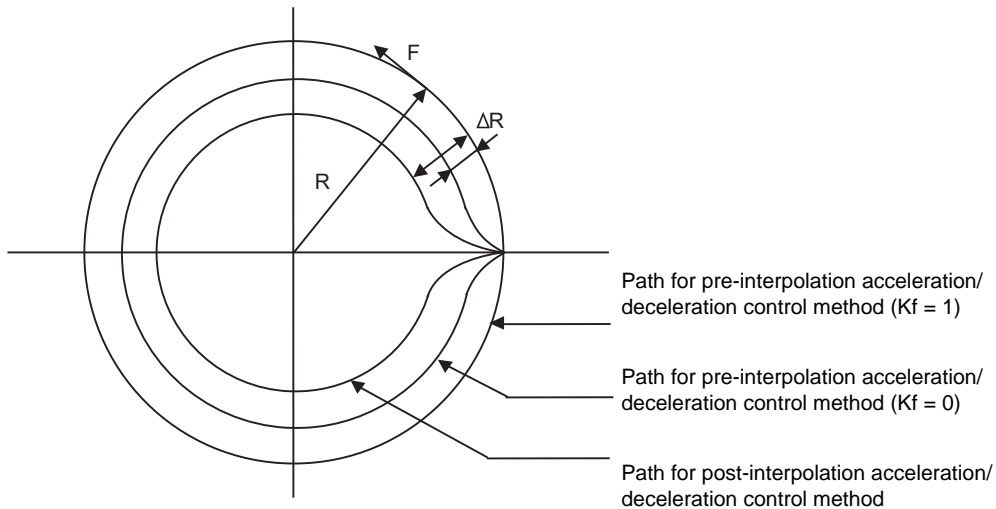
fwd_g: Feed forward gain

Ks: Accuracy coefficient

Values to be used for logical radius reduction error amount calculation:

- (a) Use the value set to axis specification parameter "#2010 fwd_g" for the first axis of the system as the feed forward gain (fwd_g).
- (b) Use the value set to servo parameter "#2203 SV003(PGN)" for the first axis of the system as the position loop gain to calculate the position loop time constant (Tp).

The feed forward gain can be set independently for G00 and G01.



(Note) If the machine vibrates when Kf (feed forward coefficient) is set to 1, Kf must be lowered or the servo system must be adjusted.

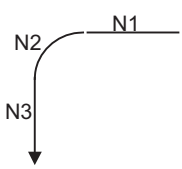
**Arc entrance/exit speed control**

There are cases when the speed fluctuates and the machine vibrates at the joint from the straight line to arc or from the arc to straight line.

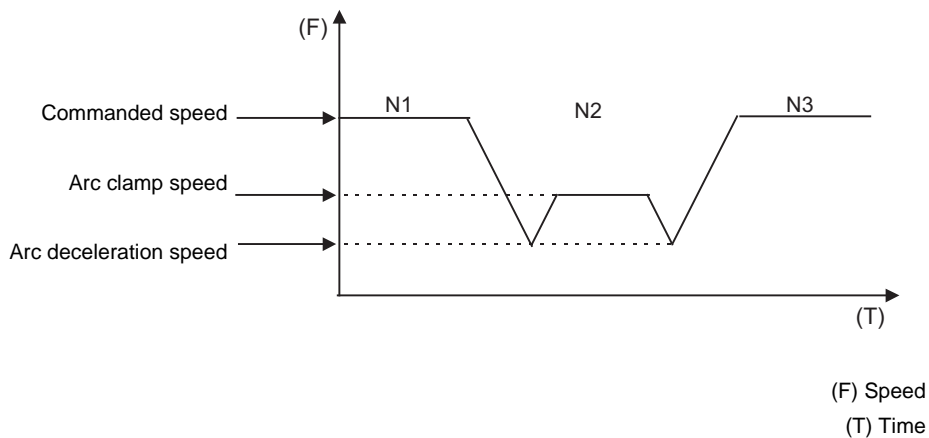
This function decelerates to the deceleration speed before entering the arc and after exiting the arc to reduce the machine vibration. If this is overlapped with corner deceleration, the function with the slower deceleration speed is valid.

The validity of this control can be changed with the base specification parameter "#1149 circft". The deceleration speed is designated with the base specification parameter "#1209 circd".

(Example 1)When not using corner deceleration

<pre>&lt;Program&gt; G61.1 ; . . N1 G01 X-10. F3000 ; N2 G02 X-5. Y-5. J-2.5 ; N3 G01 Y-10. ; . .</pre>	<pre>&lt;Operation&gt;</pre> 
---------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------

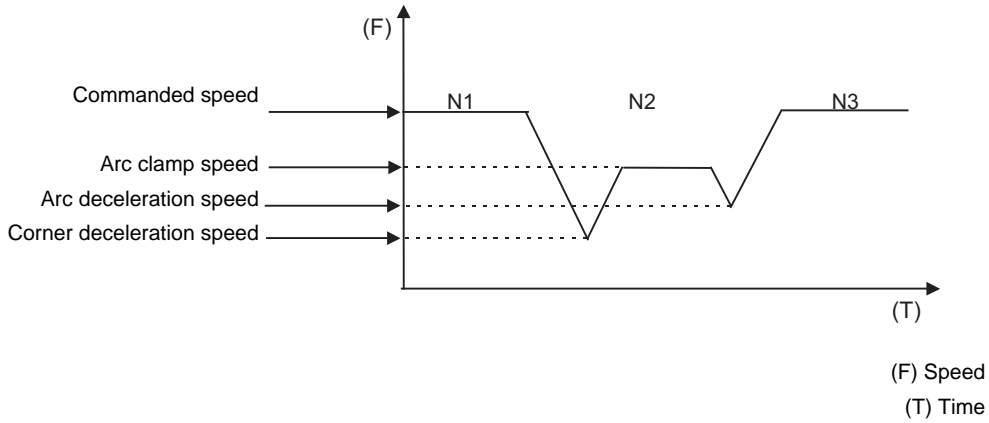
<Speed pattern>



(Example 2) When using corner deceleration

<p>&lt;Program&gt;          G61.1 ;          .          .          N1 G01 X-10. F3000 ;          N2 G02 X5. Y-5. I2.5 ;          N3 G01 X10. ;          .          .</p>	<p>&lt;Operation&gt;</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------

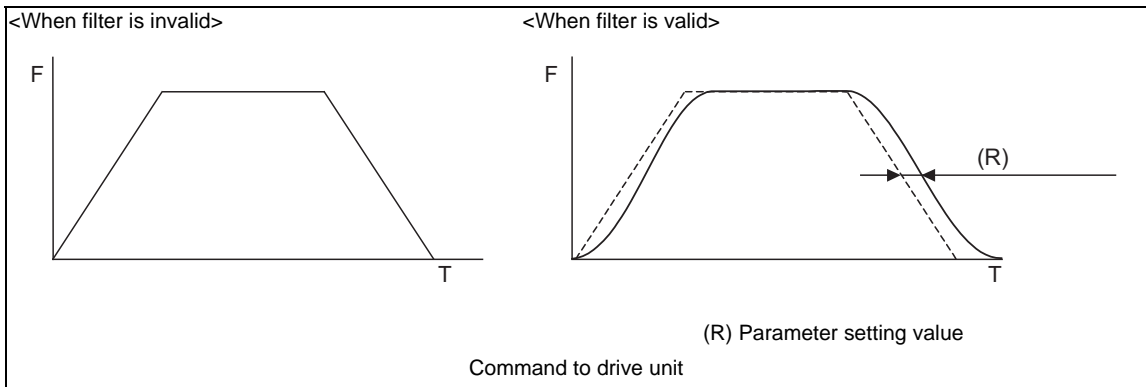
<Speed pattern>



**S-pattern filter control**

This control interpolates the fluctuations in the segments further smoothly, when they are distributed to each axis element with vector accuracy interpolation. With this, the fluctuation amplified by feed forward control is reduced and the effect onto the machine is reduced.

The S-pattern filter can be set in the range of 0 to 200(ms) with the base specification parameters "#1568 SfiltG1" and "#1569 SfiltG0". With "#1570 Sfilt2", the acceleration/deceleration fluctuation can be further smoothed.



**Precautions**

- (1) The "high-accuracy control" specifications are required to use this function  
 If G61.1 is commanded when there are no specifications, a program error (P123) will occur.
- (2) This function may not be usable, depending on the model.

## 13.15 Coordinate Rotation by Program ; G68.1/G69.1



### 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

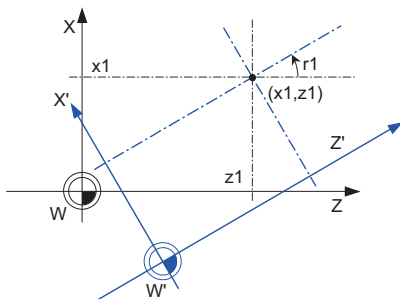
**G68.1 X__ Z__ R__ ; ... Coordinate rotation ON**

X,Z	Rotation center coordinates An axis corresponding to the plane selected from the rotation center coordinates X, Y and Z.
R	Rotation angle Designate the angle from -360° to 360° by the minimum setting unit. The counterclockwise direction on the selected plane is + direction.

**G69.1 ; ... Coordinate rotation cancel**

Select the command plane with G17 to G19.

Command) G68.1 Xx1 Zz1 Rr1 ;



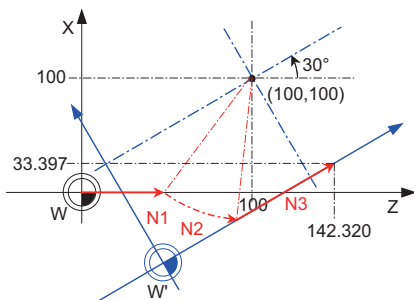
- (1) Command the rotation center coordinates (x1, z1) with an absolute value.
- (2) Rotate the coordinate counterclockwise by the angle designated in rotation angle r1.
- (3) When the minimum setting unit for r1 is 0.001deg, the setting range of rotation angle is -360.000 to 360.000. The setting range of rotation angle or r1 when the minimum setting unit is 0.001 deg is -360.000 to 360.000. When a value out of the range is commanded, a remainder of the value divided by 360° is commanded.  
(Ex.) When 400 is commanded:  
400 divided by 360 is 1, and the remainder is 40. Thus, 40 is the commanded angle.
- (4) Counter displayed values (for POSITION, WORK, and MACHINE) are all expressed in accordance with the prior-to-rotation coordinates. The coordinates after the rotation are displayed in "POSITION(2)" on the servo monitor.

W: Local coordinate system before rotation  
r1: Rotation angle

W': Local coordinate system after the rotation  
(x1, z1) Rotation center

The following is the example of relationship of program command position and the displayed position.

Program example) N1 G00 Z50.  
N2 G68.1 X100. Z100. R30.;  
N3 G00 Z120. ;



- (1) The program command performs positioning on the local coordinates after the rotation.
- (2) The counter display shows the point after the coordinate rotation on the coordinate system before rotation. In this example, the position display when the N3 block is finished is:  
X 33.397  
Z 142.320  
The "POSITION(2)" on the servo monitor indicates:  
X 100.000  
Z 100.000
- (3) G68.1 command does not carry out the actual movement. Therefore, in this example, it moves linearly from the end point of N1 to the end point of N3.

W: Local coordinate system before rotation

W': Local coordinate system after rotation

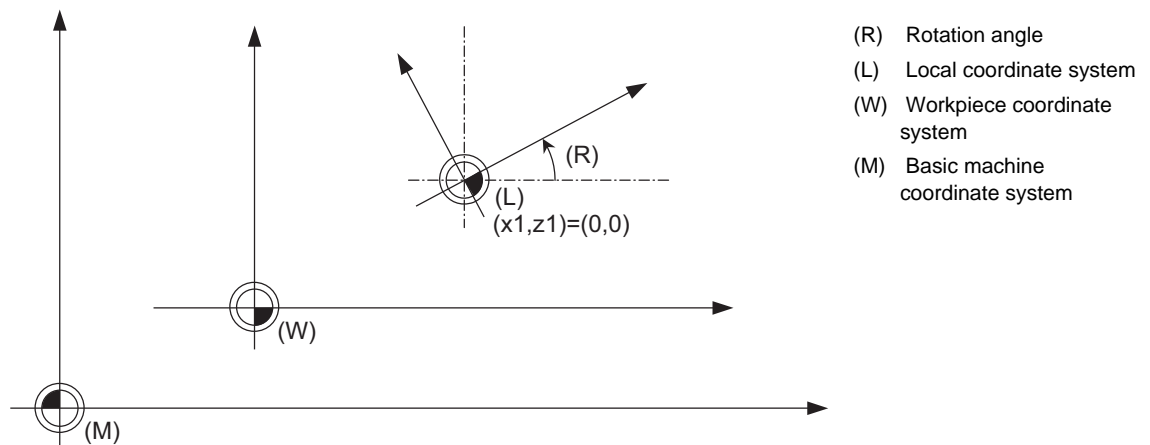


### Detailed description

- (1) G68.1 and G69.1 are the G code of group 16.
- (2) Command the rotation center coordinate (x1, z1) with an absolute value. Even if commanded with an incremental address, it will not be handled as an incremental value.
- (3) If the rotation center coordinates (x1, z1) are omitted, the position where the G68 command was executed will be the rotation center.
- (4) The rotation angle R is commanded with an absolute value. However, it can be commanded with an incremental value if the parameter "#8082 G68.1 R INC" is set.

	G code list 2	G code list 3	
		Absolute value commands (G90)	Incremental value commands (G91)
#8082=0	Absolute value	Absolute value	Incremental value
#8082=1	Incremental value	Incremental value	

- (5) The performance when the rotation angle R is omitted depends on the setting of parameter "#1270 ext06/bit5":
  - 0: Use the previously commanded value (modal value).
  - 1: Use the set value in "#8081 Gcode Rotat".
 If the coordinate rotation mode is canceled by G69.1 command, the modal value will be cleared. If G68.1 is commanded after G69.1 was commanded, the rotation angle becomes 0° by omitting R.  
 The setting value of the parameter is an absolute value regardless of the setting of the parameter "#8082 G68.1 R INC".
- (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.



- (7) The coordinate rotation command during coordinate rotation is processed as the changes of center coordinates and rotation angle.
- (8) If commanding G68.1 or G69.1 without the coordinate rotation specification, a program error (P260) will occur.
- (9) If commanding the plane selection code during the coordinate rotation mode, a program error (P111) will occur.
- (10) The program coordinate rotation function is valid only in the automatic operation mode.
- (11) G68.1 is displayed on the modal information screen during the coordinate rotation mode. When the mode is canceled, the display changes to G69.1. (The modal value for the rotation angle command R is displayed.)

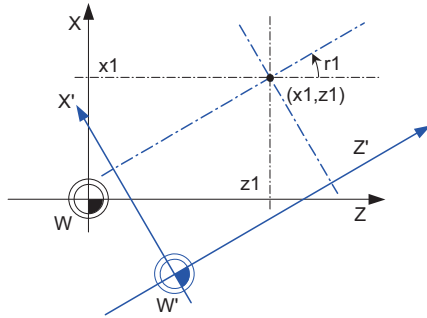
**Coordinate rotation command during coordinate rotation**

The coordinate rotation command during coordinate rotation is processed as the changes of center coordinates and rotation angle.

(1) For absolute command

Command)  
 G68.1 Xx1 Zz1 Rr1 ;  
 G68.1 Xx2 Zz2 Rr2 ;

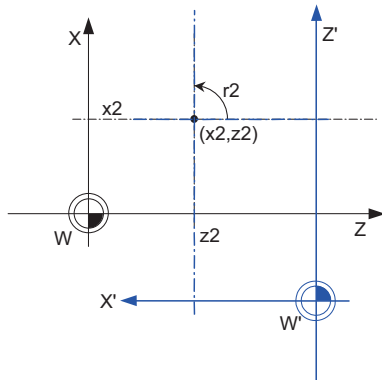
1) G68.1 Xx1 Zz1 Rr1 ;



With spinning around on the center coordinate of the rotation (x1, z1), the rotation takes place in the counterclockwise direction by the angle designated in rotation angle r1.

W: Local coordinate system before the rotation  
 W': Local coordinate system after the rotation  
 r1: Rotation angle  
 (x1, z1) Rotation center

2) G68.1 Xx2 Zz2 Rr2 ;



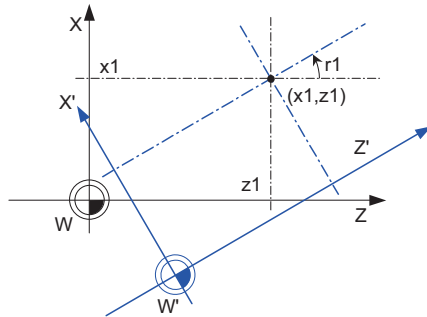
The center coordinate of the rotation switches from (x1, z1) to (x2, z2), and the rotation angle is cleared once. Then the rotation takes place in the counterclockwise direction by the angle designated with r2

W: Local coordinate system before the rotation  
 W': Local coordinate system after the rotation  
 r2: Rotation angle  
 (x2, z2) Rotation center

(2) For incremental command

Command)  
 G68.1 Xx1 Zz1 Rr1 ;  
 G68.1 Ux2 Uz2 Rr2 ;

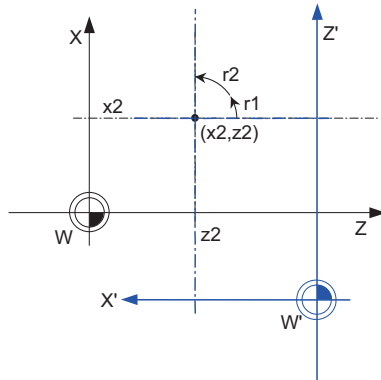
1) G68.1 Xx1 Zz1 Rr1 ;



With spinning around on the center coordinate of the rotation (x1, z1), the rotation takes place in the counterclockwise direction by the angle designated in rotation angle r1.

W: Local coordinate system before the rotation  
 W': Local coordinate system after the rotation  
 r1: Rotation angle  
 (x1, z1) Rotation center

2) G68.1 Ux2 Uz2 Rr2 ;



The center coordinate of the rotation switches from (x1, z1) to (x2, z2). Even if the rotation center coordinate command is the incremental value command, it is handled as the absolute value.

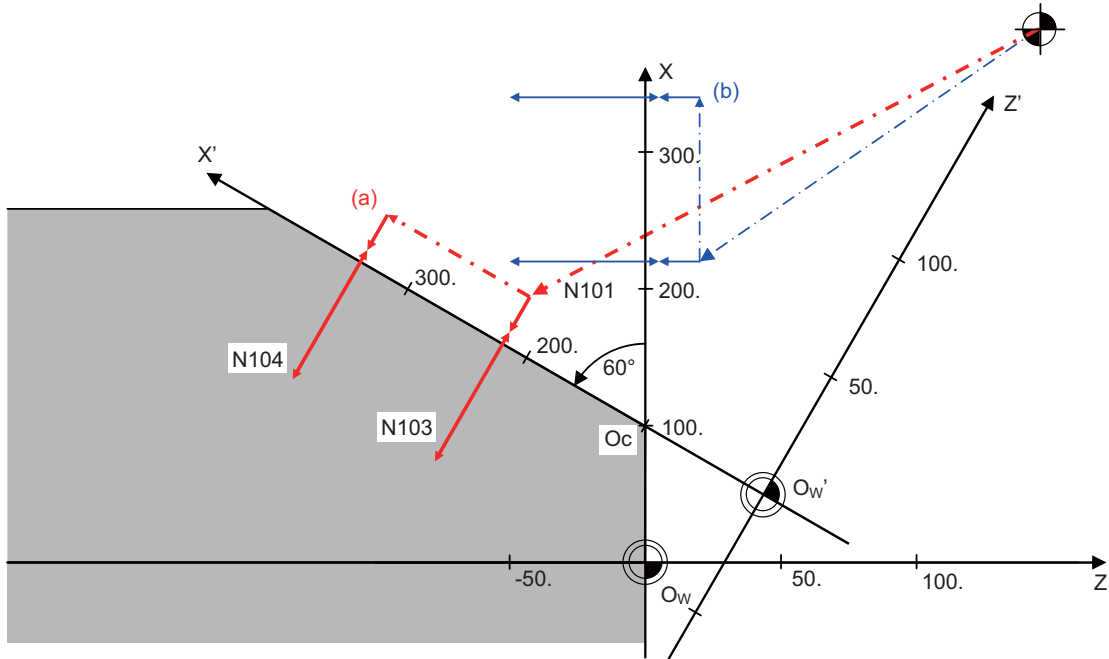
The rotation takes place in the counterclockwise direction by the angle rotated at r1 and another angle commanded at r2.

W: Local coordinate system before the rotation  
 W': Local coordinate system after the rotation  
 r2: Rotation angle  
 (x2, z2) Rotation center



Program example

Program coordinate rotation by absolute command



- (O_c) Rotation center
- (O_w) Workpiece coordinate zero point before rotation
- (O_{w'}) Workpiece coordinate zero point after rotation
- (a) Subprogram path after rotation
- (b) Subprogram path before rotation

Main program

```

N01 G97 G18;           Z-X Plane selection
N02 G91 G28 X0. Z0.;
N03 G54;
N04 G90 T1010;
N05 G68.1 X100. Z0 R60.;   Coordinate rotation ON
N06 M98 H101;           Subprogram execution
N07 G69.1;             Coordinate rotation cancel
N08 M02 ;              End
    
```

Subprogram (Shape programmed with coordinate system before rotation)

```

N101 G00 X220. Z20.;
N102 G94 S2=1000 M3;       2nd spindle (tool spindle) forward
N103 G98 G83 Z-50. R-15. Q-10. F100;
N104 X340. ;
N105 G80;
N106 S2=0 M5;             2nd spindle stop
N107 M99;
    
```



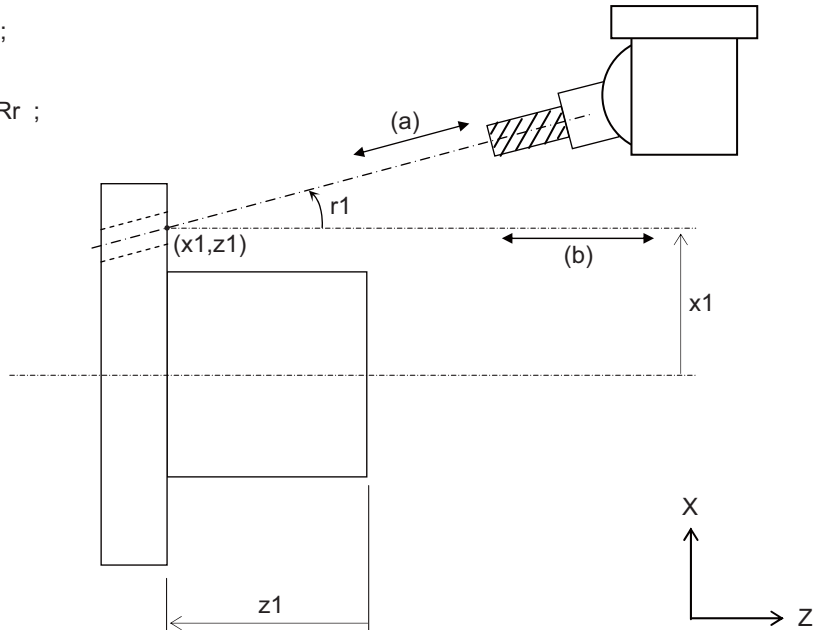
### Relation with other functions

- (1) The tool compensation and tool nose wear compensation during the coordinate rotation mode are carried out in the coordinate system before the coordinate rotation. Therefore, the path after the rotation is exactly the same shape as the one before the rotation. However, when the tool tip point is not located at the center, the tool nose R compensation is performed in the coordinate system after the rotation. Thus the path after the rotation is not the same shape as the one before the rotation.
- (2) The position information that is read by using the variables #5001 to #5100+n (end point coordinates of the previous block, machine coordinates, workpiece coordinates, etc.) are the coordinates before the rotation.
- (3) The coordinates can also be rotated for the parallel axis. Select the plane that contains the parallel axis before issuing the G68.1 command. (The plane which contains the parallel axis cannot be selected in the same block as the G68.1 command.)
- (4) A system variable can read the skip coordinate value when the skip command is issued during the coordinate rotation. The system variable reads the workpiece coordinate value before the rotation. When the multiple axes move for one axis movement command, the skip coordinate value is read by the multiple axes.
- (5) Program error will occur if the following functions are commanded:
  - Thread cutting
  - Variable lead thread cutting
  - Fixed cycles for turning machining
  - Compound type fixed cycle for turning machining
  - 2-part system simultaneous thread cutting cycle
  - User macro modal call B
  - The facing turret mirror image
  - Balance cut
- (6) Program error (P262) will occur if the coordinate rotation is commanded during the following G code modal:
  - Thread cutting
  - Variable lead thread cutting
  - Fixed cycles for turning machining
  - Compound type fixed cycle for turning machining
  - 2-part system simultaneous thread cutting cycle
  - Fixed cycle for drilling
  - User macro
  - The facing turret mirror image
  - Balance cut

(7) Tapping cycle can be carried out during the coordinate rotation mode. This enables the tapping diagonally.

Refer to "Face Tapping Cycle (Longitudinal tapping cycle) ; G84(G88)"

```
G68.1 Xx1 Zz1 Rr1 ;
.
.
G84 Zz2 Rr2 Ff Ss ,Rr ;
.
.
```



- (a): Actual movement direction
- (b): Movement direction with program command
- r1: Rotation angle
- (x1, z1): Rotation center

(7-1) Feedrate/pitch command (F command)

- F command value specified with the machining program is as follows:
  - Asynchronous tap: Feedrate toward the tap cutting direction (diagonally)
  - Synchronous tap: Pitch toward the tap cutting direction (diagonally)

(7-2) Programmable in-position check

- In-position check takes place on two axes as two axes move during tapping diagonally.
- Each axis is checked, and the in-position check is complete when both of two axes come into the commanded in-position width.

(7-3) Tapping return

- Tapping return can be carried out by the tapping return signal (1st part system: Y75C, 2nd part system: Y83C) for diagonal cutting.
- Tap cutting axes (two axes moved during tap cutting) move toward the initial point.

(7-4) Servo gain during the synchronous tapping cycle

When diagonal synchronous tapping is carried out, the setting value of "#2017 tap_g" is the servo gain of two axes which move during tap cutting.

(7-5) Precautions for synchronous tapping cycle

- Set the same value for the servo gain ("#2017 tap_g") of two axes which move during tap cutting.
- When the tapping axis is an inclined axis, and 2 axes are moved during tap cutting, the position loop gain of the non-inclined movement axis is not switched.



## Precautions

- (1) Command G68.1 in independent blocks. If not, a program error (P261) will occur.
- (2) If an axis which is not in the selected plane is commanded to the coordinate value of the rotation center, axes in other than the selected plane moves to the position which is specified by the last G1 modal.
- (3) Always command an absolute value for the movement command immediately after G68.1 and G69.1. If an increment value is commanded, it may not move to the intended position.  
Also command it together with the axis address on the selected plane (for G18 plane, Z-X). If it is omitted, that axis is handled as "no movement command".
- (4) If the manual absolute is ON and interrupted the coordinate rotation axis, then, do not use automatic operation for the following absolute value command.
- (5) The intermediate point during reference point return is the position after the coordinates are rotated.
- (6) If any of the following is commanded during the coordinate rotation mode, the rotation center for the program coordinate rotation will be shifted. (The center will follow the coordinate system.)
  - Workpiece coordinate system offset change
  - Local coordinate system setting (G52)
- (7) If any of the following is commanded during the coordinate rotation mode, the rotation center for program coordinate rotation will not shift. (The center position will be the same looking from the basic machine coordinate system.)
  - Coordinate system setting (G92)
  - Workpiece coordinate system setting and offset (G54 ~ G59)
- (8) If coordinate rotation is executed to the G00 command for only one axis during the coordinate rotation mode, two axes will move. If the parameter "#1086 G0Intp" is set to "1", the interpolation is carried out.
- (9) The normal synchronous tapping only is available during a coordinate rotation on the plane that includes a tapping axis, during which high-speed synchronous tapping is not available.

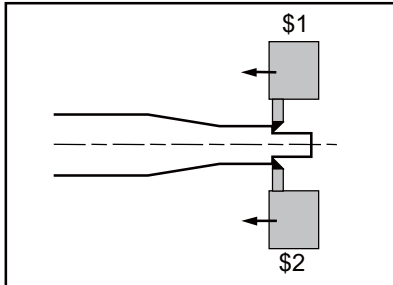


## 13.16 Balance Cut ; G15,G14



### Function and purpose

The timing for starting the operation of the 1st part system turret and 2nd part system turret can be synchronized.



When machining a relatively thin and long workpiece with a lathe, the workpiece could slack, and highly accurate machining may not be possible.

In this case, if the cutters are applied simultaneously from both sides of the workpiece and the workpiece is machined while synchronizing these (balance cut), the slack can be suppressed. Furthermore, the machining time can be shortened by machining with two cutters.

With this function, the movement of two turrets belonging to different part systems can be completely synchronized, so such kind of machining can be done easily.



### Command format

**G15 ; ... Balance cut command ON (modal)**

**G14 ; ... Balance cut command OFF (modal)**



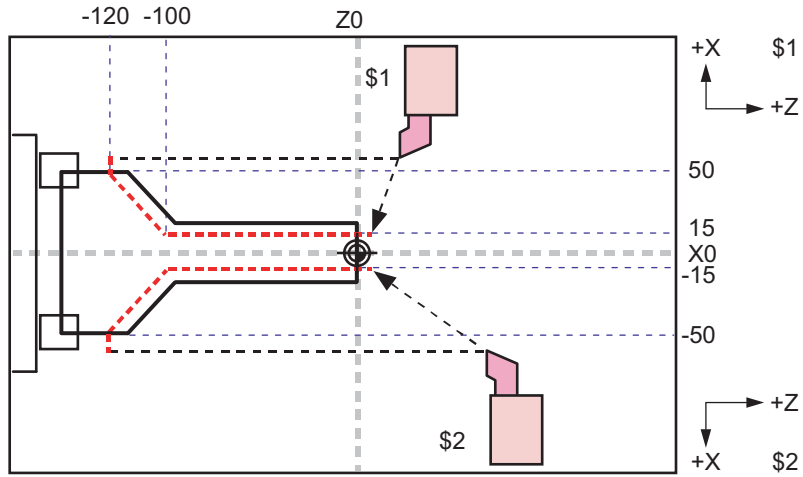
### Detailed description

- (1) G15 and G14 commands are modals. In the CNC's initial state, the G14 balance cut command is OFF.
- (2) When G15 is commanded, the timing synchronization between part systems will be maintained for all blocks until the G14 command is commanded.
- (3) If G15 or G14 is commanded in one part system, movement will not advance until the same G code is commanded in the other part system.
- (4) After G14 is commanded in both part systems, 1st part system and 2nd part system will operate independently.



### Program example

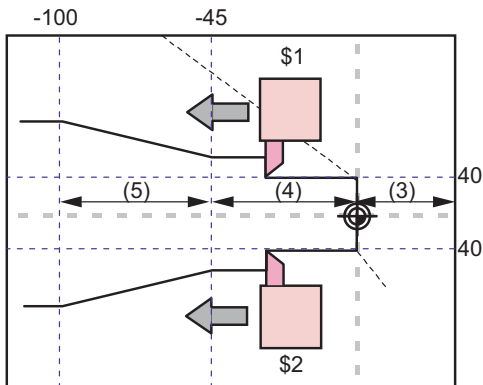
An example of a program for machining with a 1-spindle 2-turret CNC lathe while simultaneously applying the cutters from the top and bottom of a thin long workpiece using balance cut is shown below.



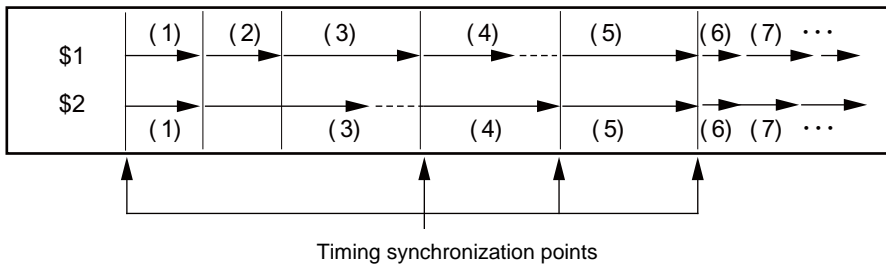
1st part system program (\$1) G28 XZ ; S100 T0101 ; G15 ; G00 X15 Z3 ; G01 Z-100 F0.2 ; X50 Z-120 ; X52 ; G14 ; G28 XZ ; M30 ;	2ndt part system program (\$2) G28 XZ ; T0101 ; G15 ; G00 X15 Z3 ; G01 Z-100 F0.2 ; X50 Z-120 ; X52 ; G14 ; G28 XZ ; M30 ;
--------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------



Operation example



<1st part system>	<2nd part system>	
:	:	
G15	G15	...(1)
S200		...(2)
G00 X40. Z-2.	G00 X40. Z-2.	...(3)
G01 W47. F10.	G01 W47. F5.	...(4)
G01 U40. W55.	G01 X80. Z100. F10.	...(5)
G14	G14	...(6)
G00 X100.	G00 X100.	...(7)
:	:	



- (1) Balance cut is turned ON with the G15 command.
- (2)(3) The S command and rapid traverse command are not timing-synchronized, so the operation waits at the head of (4).
- (4) 1st part system will finish first, but since the next block is a cutting feed command, the operation will wait at the head of (5).
- (5) Cutting will start with 1st system and 2nd system together.
- (6) Balance cut is turned OFF with the G14 command.
- (7) Each part system will operate independently after this.



## Precautions

- (1) A block that does not contain movement such as G4, M, S, T, B command or a variable command is handled as 1 block, and synchronization will be carried out.
- (2) If subprogram call or macro call are carried out while balance cut is ON, the block configuring the subprogram will be handled as one block, and synchronization will be carried out.

## Number of cutting feed blocks in balance cut mode

If G14 is commanded in one part system first and the other part system is in cutting feed, the first part system will enter the waiting state. Operation cannot be advanced to the next blocks in this case. When commanding balance cut, make sure that the same number of cutting feed blocks are set between the 1st part system and 2nd part system mode ON and mode OFF states.

<1st part system>	<2nd part system>
:	:
N20 G15	N20 G15
N30 G00 X40. Z0.	N30 G00 X-40. Z250.
N40 G01 W-30. F1000	N40 G01 W-130. F500
N50 G01 U40. W-70.	N50 G01 X-80. Z50. F1000
<b>N60 G01 W-20.</b>	<b>N60 G14</b>
N70 G14 ;	N70 S200
N80 G01 X120. Z30.	N80 G00 X-100.

The balance cut mode is canceled with G14 on the 2nd part system side first, so 1st part system enters the waiting state.  
 ← The waiting state will also be entered when only one part system has been reset.

## Use with timing synchronization between part systems

If one part system is standby for synchronization with timing synchronization between part systems and the other part system enters the synchronization standby state with the G15 command, both part systems will be in the standby state, and will not shift to the next block. Command so that standby for waiting for G15 and standby for waiting with timing synchronization between part systems do not occur simultaneously.

## Commanding timing synchronization between part systems during balance cut mode

When the timing synchronization between part systems is commanded during the balance cut mode, that block is handled as a command of one block without movement and no timing synchronization will be executed.

## Conditions for alarm with G15 and G14

- (1) A program error (P34) will occur if G15 or G14 is commanded in one part system.
- (2) This function is valid only for part system 1 and part system 2. If commanded for the part system 3 and following, a program error P34 will occur.

## Conditions for ignoring G15 and G14

If G14 is commanded when G15 is not commanded (when balance cut is OFF), the G14 block will be handled as one that has no process.

## 13.17 Timing Synchronization between Part Systems



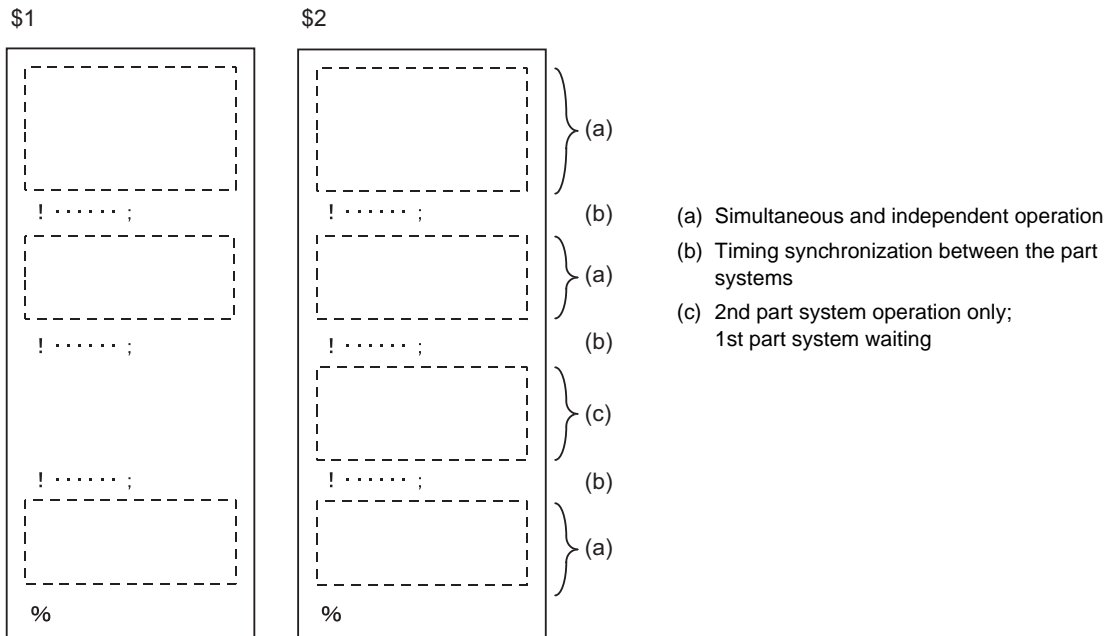
1. When programming a multi-part system, carefully observe the movements caused by other part systems' programs.

### 13.17.1 Timing Synchronization between Part Systems ; !nL



#### Function and purpose

The multi-axis, multi-part system complex control NC system can simultaneously run multiple machining programs independently. The timing synchronization between part systems is used in cases when, at some particular point during operation, the operations of 1st and 2nd part systems are to be synchronized or in cases when the operation of only one part system is required.



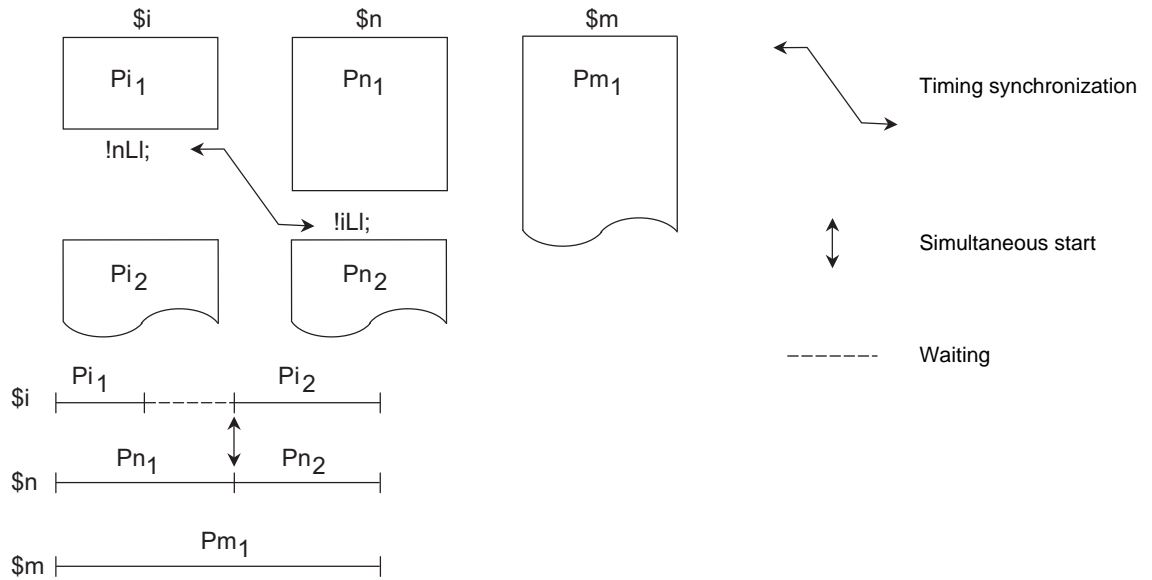
#### Command format

!nL ;	
L	Between-part-systems timing synchronization No. 1 to 9999

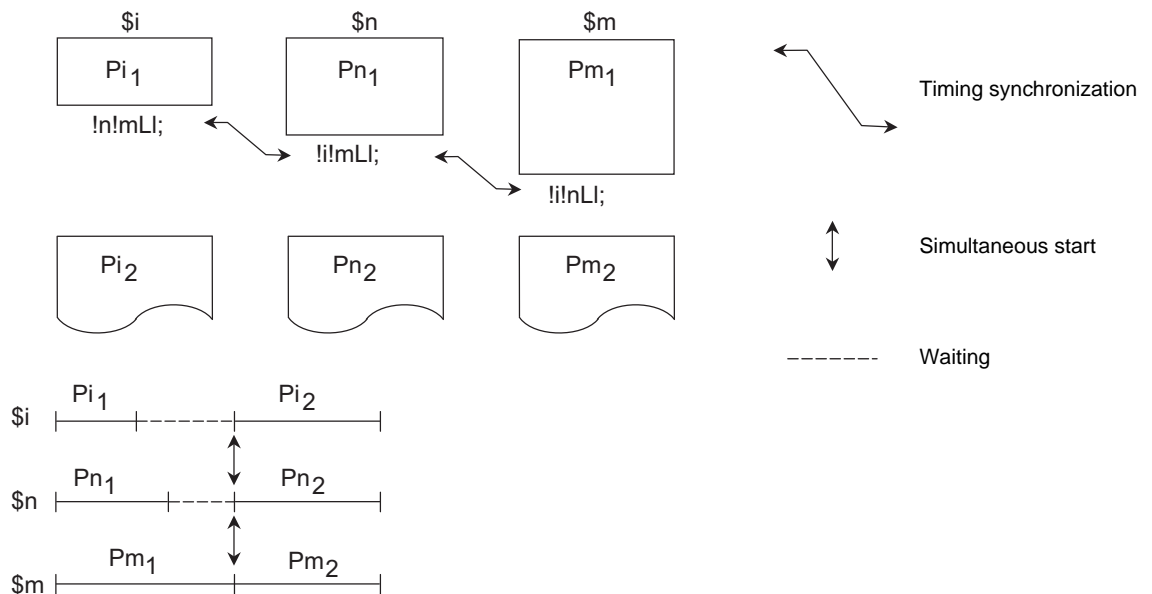


Detailed description

- (1) When the "!nLI" code is issued from the part system "i", the operation of that program will wait until the "!iLI" code is issued from the part system "n".  
 When the "!iLI" code is issued, the programs of both part systems "i" and "n" will start running simultaneously.



- (2) Timing synchronization among three part systems is as follows. When the "!n!mLI" command is issued from the part system "i", the program of part system "i" operation will wait until the "!i!mLI" command is issued from the part system "n" and the "!i!nLI" command is issued from the part system "m".  
 When the between-part-systems timing synchronization commands are all issued, programs of part systems "i", "n" and "m" will start operating simultaneously.

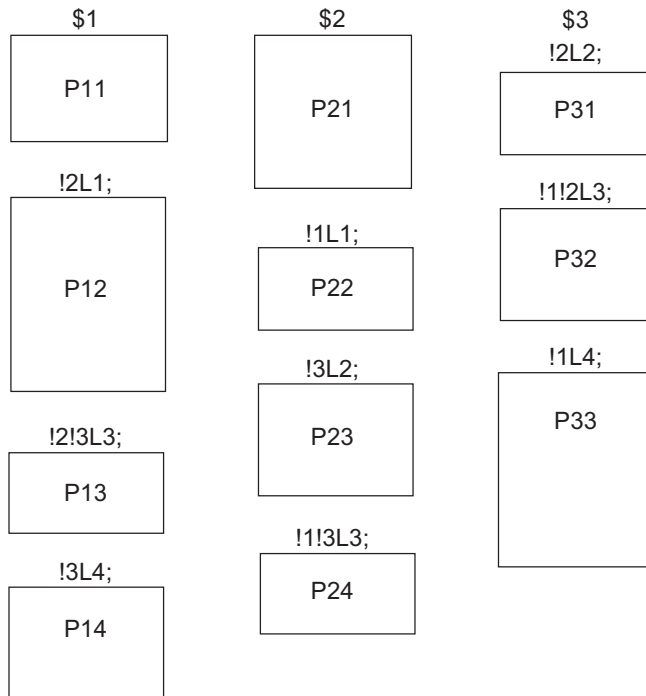


- (3) The between-part-systems timing synchronization command is normally issued in a single block. However, if a movement command or M, S or T command is issued in the same block, whether to enter the waiting state before or after executing the movement command or M, S or T command will depend on the parameter (#1093 Wmvfin).
  - #1093 Wmvfin
  - 0 : Wait before executing movement command.
  - 1 : Wait after executing movement command.
- (4) If there is no movement command in the same block as the between-part-systems timing synchronization command, when the next block movement starts, timing synchronization may not be secured between the part systems. To synchronize the part systems when movement starts after waiting, issue the movement command in the same block as the between-part-systems timing synchronization command.
- (5) Timing synchronization between part systems is executed only when the target part system to be timing-synchronized is operating automatically. If it's not operating automatically, the timing synchronization between part systems will be ignored and operation will advance to the next block.
- (6) The L command is the between-part-systems timing synchronization identification No. The same Nos. are timing-synchronized but, when they are omitted, the Nos. are handled as L0.
- (7) "SYN" will appear in the operation status section during timing synchronization between part systems. The between-part-systems timing synchronization signal will be output to the PLC I/F.
- (8) Program error (P35) occurs when an illegal system number has been issued.
- (9) The timing synchronization command designates the No. of the other part system to be timing-synchronized, and can also be issued along with its own part system No. .  
(Example) Part system "i" command: !i!n!mLl;
- (10) When the part system No. is omitted (when only "!" is issued), part system 1 will be handled as "!2" and part system 2 as "!1". The command using only "!" cannot be used for timing-synchronizing with part system 3 and following.  
If the command using only "!" is used for part system 3 or following, the program error (P33) will occur.



Operation example

Example of timing synchronization between part systems



The above programs are executed as follows:

\$1	P11		P12		P13	P14
		L1			L3	
\$2	P21	P22	P23	P24		L4
			L2		L3	
\$3			P31		P32	P33



### 13.17.2 Start Point Designation Timing Synchronization (Type 1) ; G115



#### Function and purpose

A part system waits for another part system to reach a start point before starting itself.  
The start point can be set in the middle of a block.



#### Command format

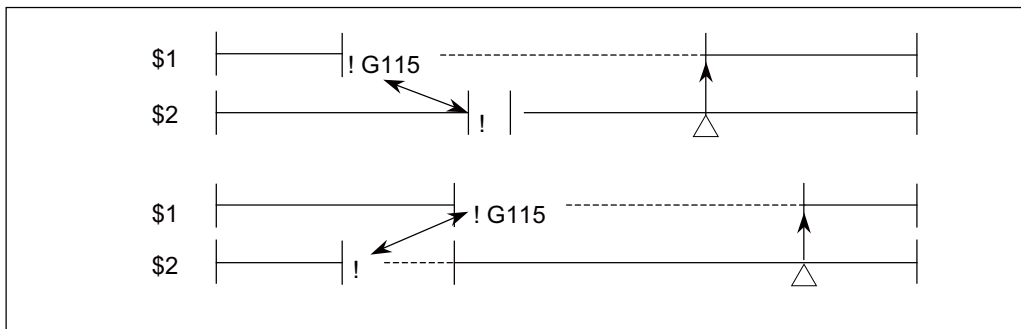
`!nL G115 X_ Z_ ;`

!nL	Timing synchronization command
G115	G command
X Z	Start point (designated by axis name(s) and workpiece coordinate(s))



#### Detailed description

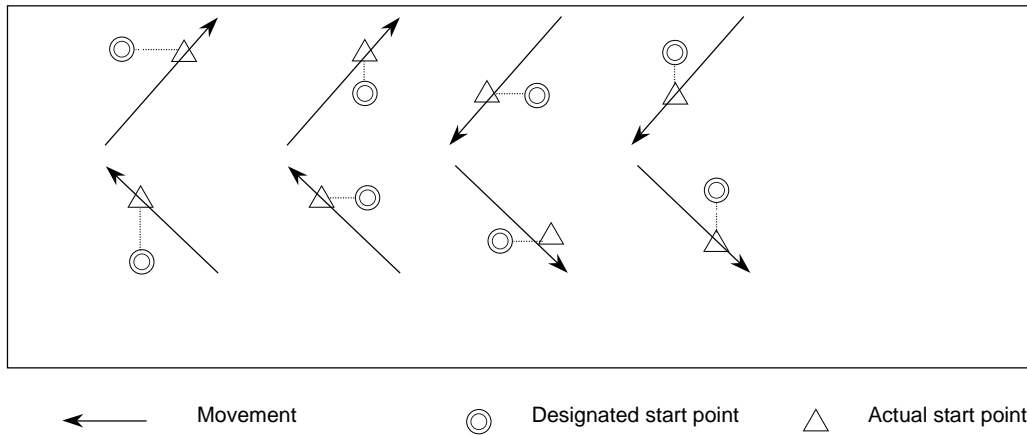
- (1) Designate the start point using the workpiece coordinates of the other part system (ex. \$2).
- (2) The start point check is executed only for the axis designated by G115.  
(Example) !L2 G115 X100. ;  
Once the other part system reaches X100., the own part system (ex. \$1) will start. The other axes are irrelevant in the start point check.
- (3) The other part system starts first when timing synchronization is commanded.
- (4) The own part system waits for the other part system to move and reach the designated start point, and then starts.



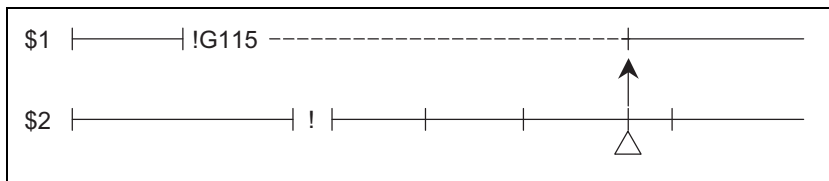
↔ Timing synchronization

△ Designated start point

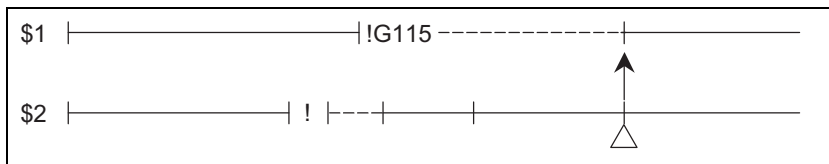
- (5) When the start point designated by G115 is not on the next block movement path of the other part system, the own part system starts once all the designated axes of the other part system reach the designated start point.



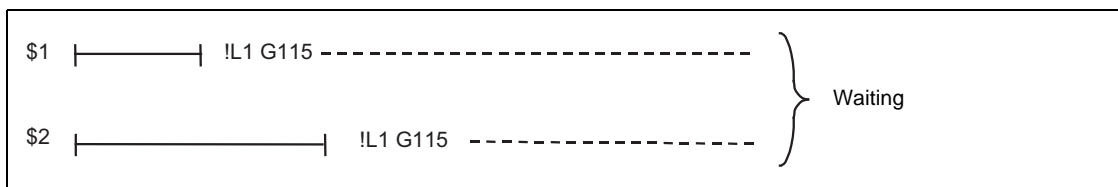
- (6) The parameter setting of the base specification parameter #1229 set01/bit5 determines how to operate when the start point cannot be determined by the next block movement of the other system.  
 (a) When the parameter is ON  
 The own part system waits until the other system reaches the starting point in the next and subsequent blocks.



- (b) When the parameter is OFF  
 The own part system starts upon completion of the next block movement.



- (7) The waiting status continues when the G115 command has been issued by two part systems towards each other.



- (8) A program error (P33) occurs when the G115 command is issued for 3 part systems.  
 (9) The single block stop function does not apply for the G115 block.  
 (10) A program error (P32) will occur if an address other than an axis name is designated in G115 command block.

### 13.17.3 Start Point Designation Timing Synchronization (Type 2) ; G116



#### Function and purpose

A part system makes another part system wait until itself reaches a start point.  
The start point can be set in the middle of a block.



#### Command format

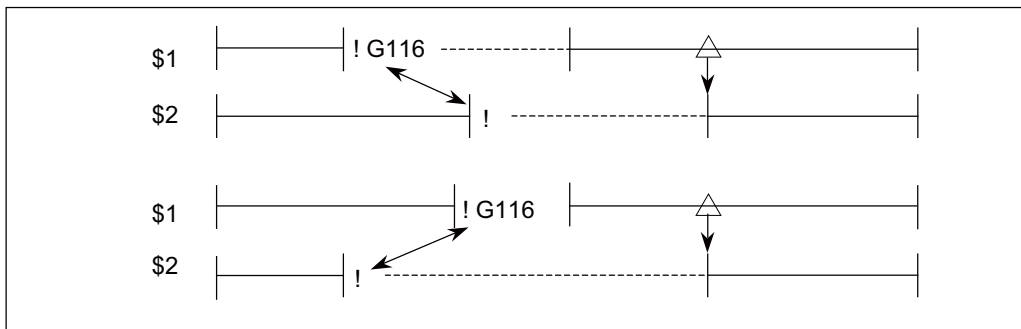
`!nL G116 X_ Z_ ;`

!nL	Timing synchronization command
G116	G command
X Z	Start point (designated by axis name(s) and workpiece coordinate(s))



#### Detailed description

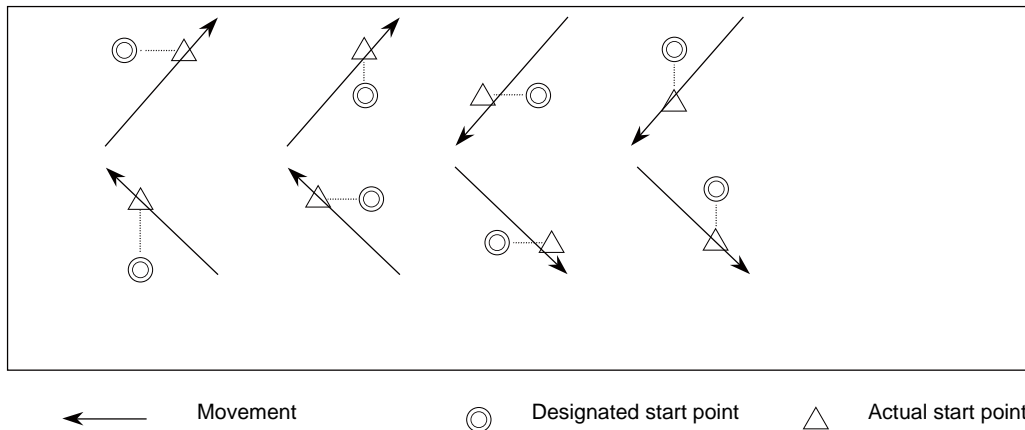
- (1) Designate the start point using the workpiece coordinates of the own part system (ex. \$1).
- (2) The start point check is executed only for the axis designated by G116.  
(Example) !L1 G116 X100. ;  
Once the own part system reaches X100., the other part system (ex. \$2) will start. The other axes are irrelevant in the start point check.
- (3) The own part system starts first when timing synchronization is commanded.
- (4) The other part system waits for the own part system to move and reach the designated start point, and then starts.



←→ Timing synchronization

△ Designated start point

- (5) When the start point designated by G116 is not on the next block movement path of own part system, the other part system starts once all the designated axes of the own part system reach the designated start point.



- (6) The parameter setting of the base specification parameter #1229 set01/bit5 determines how to operate when the start point cannot be determined by the next block movement of the own part system.

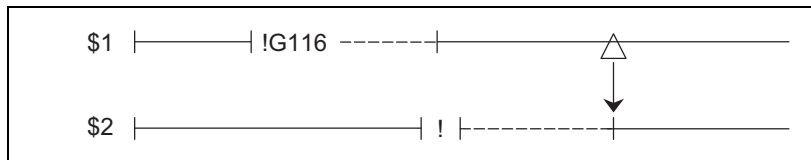
(a) When the parameter is ON

Program error "P33" occurs before the own part system moves.

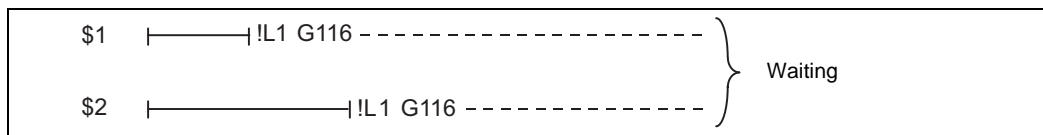


(b) When the parameter is OFF

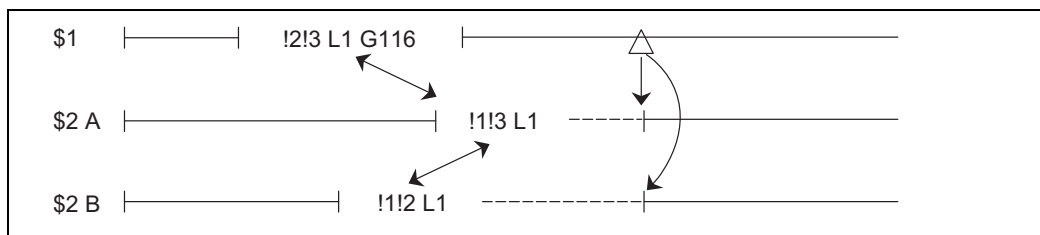
The other part system starts upon completion of the next block movement.



- (7) The waiting status continues when the G116 command has been issued by two part systems towards each other.



- (8) The two target part systems start when the G116 command is issued for 3 part systems.



- (9) The single block stop function does not apply for the G116 block.

- (10) A program error (P32) will occur if an address other than an axis name is designated in G116 command block.

## 13.18 2-part System Simultaneous Thread Cutting Cycle



### Function and purpose

The 2-part system simultaneous thread cutting cycle function allows 1st part system and 2nd part system to perform thread cutting simultaneously for the same spindle.

The 2-part system simultaneous thread cutting cycle has two commands; the command (G76.1) for simultaneously cutting threads in two places, which is known as the "2-part system simultaneous thread cutting cycle I" and the command (G76.2) for simultaneously cutting a thread by two part systems, which is known as the "2-part system simultaneous thread cutting cycle II".

### 13.18.1 2-part System Simultaneous Thread Cutting Cycle Parameter Setting Command ; G76



### Command format

G76 Pmra Q Δdmin Rd ;

Address		Significance
P	m	Number of cutting passes for finishing ... Reversible parameter "#8058 G76 TIMES" is also available for setting. (modal) : 00 to 99 (times)
	r	Chamfering amount ... Reversible parameter "#8014 CDZ-VALE (L system only)" is also available for setting. (modal) : 00 to 99 (0.1mm/rev) This sets the chamfering width based on the thread lead 1 across a range from 0.01 to 9.91 with a 2-digit integer with the decimal point omitted. (00 to 99)
	a	Tool nose angle (thread angle) (modal) : 0 to 99 (°) This selects the angle from 0° to 99° and commands the value in two digits. The angle from 0° to 99° is assigned in 1° units. "m", "r" and "a" are commanded in succession in address P. (Example) When "m" = 5, "r" = 1.5 and "a" = 0° , P is 051500 and the leading and trailing "0" cannot be omitted.
Q	Δdmin	Minimum cut amount (modal) : 0 to 99999 (μm) This is the clamping value for guaranteeing the cut amount of a single cutting pass and it is valid only for rough cutting. Finishing is performed by the cutting allowance designated separately. If the calculated cut amount is smaller than Δdmin, it is clamped by Δdmin.
R	d	Finishing allowance(modal) : 0 to 99999 (μm)

(Note) A reversible parameter enables to use parameter setting value without issuing a program command and also, the value can be changed by the program command.



### Detailed description

- (1) The data is set in machining parameter r: #8014 for each part system.
- (2) Issue the command for each part system.
- (3) If the parameter setting command is omitted, the parameter setting values are used from #8014 setting.  
The minimum cut-in amount ( Δdmin) follows the #1222/bit4 setting.

### 13.18.2 2-part System Simultaneous Thread Cutting Cycle I ; G76.1



#### Command format

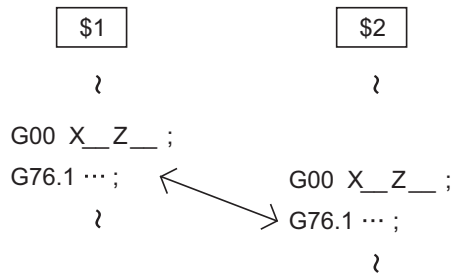
**G76.1 X/U__ Z/W__ R__ P__ Q__ F__ ;**

X/U	X-axis end point coordinates of thread section The X-axis coordinates of the end point at the thread section are commanded with absolute or incremental values.
Z/W	Z-axis end point coordinates of thread section The Z-axis coordinates of the end point at the thread section are commanded with absolute or incremental values.
R	Taper height component (radius value) at thread section ... A straight thread is created when it is 0.
P	Thread height ... This thread height is commanded with a positive radius value.
Q	Cut amount ... The cut amount for the first cutting pass is commanded with a positive radius value.
F	Thread lead

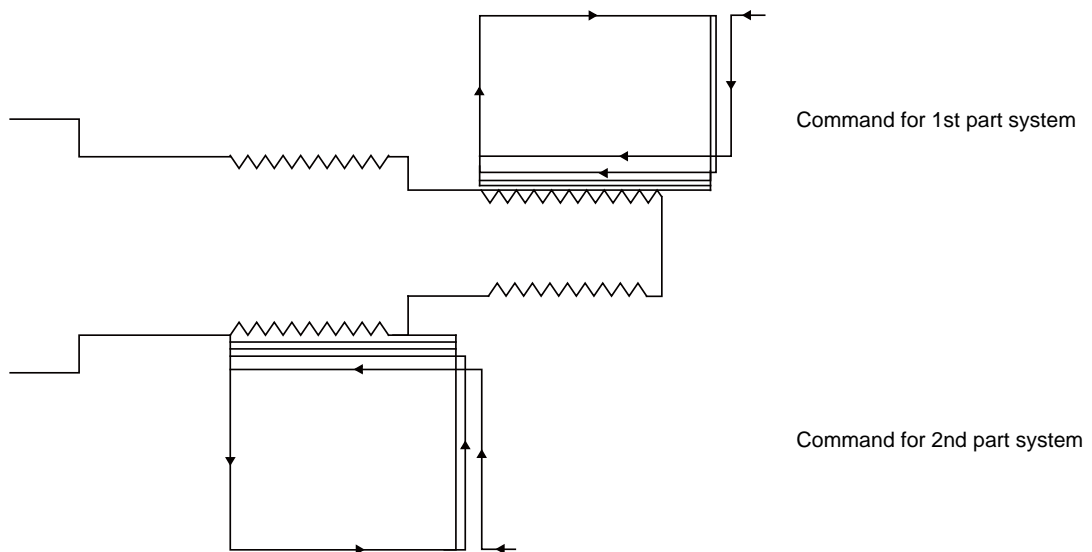


#### Detailed description

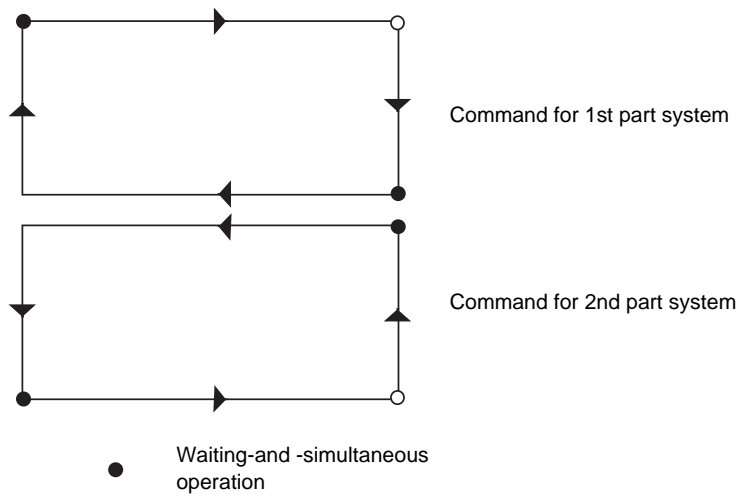
- When G76.1 is issued by 1st part system and 2nd part system, waiting is done until the command is issued to another part system. The thread cutting cycle starts when the commands are aligned properly.



- In the G76.1 cycle, G76.1 is issued simultaneously by 1st part system and 2nd part system, and the thread is cut in waiting at the start and end of thread cutting.



- (3) In one cycle, waiting is done at the start and end of the thread cutting.



- (4) The same precautions for thread cutting command (G33), thread cutting cycle (G78) and compound thread cutting cycle (G76) apply to this cycle.
- (5) As the threads are cut in two places by the G76.1 command, the various commands do not need to be the same. Each of them can be issued independently.

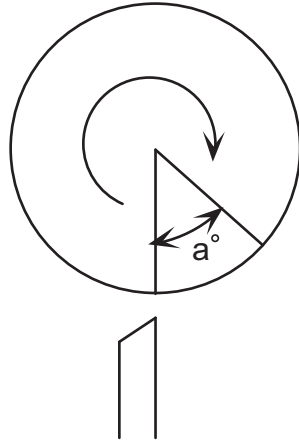
## 13.18.3 2-part System Simultaneous Thread Cutting Cycle II ; G76.2



## Command format

**G76.2 X/U__ Z/W__ R__ P__ Q__ Aa F__ ;**

(1) Thread cutting start shift angle



The thread cutting command starts movement after waiting for the spindle encoder's one rotation synchronization signal. However, the start point can be delayed by a degree amount.

a : Thread cutting start angle

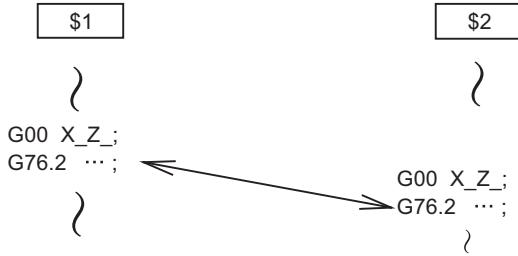
The meanings of the addresses other than A are the same as the 2-part system simultaneous thread cutting cycle I.



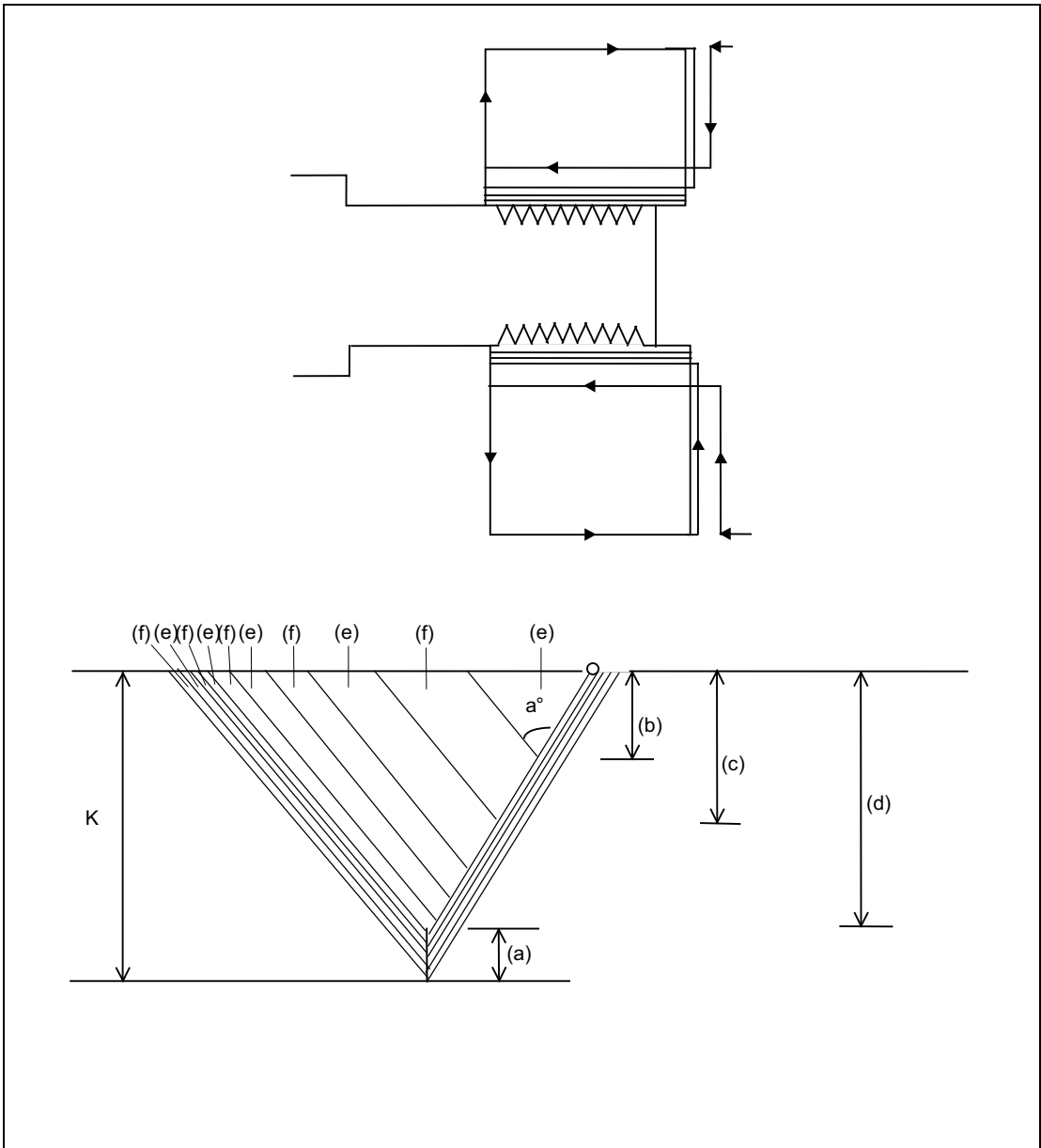


Detailed description

- (1) When G76.2 is issued by 1st part system and 2nd part system, waiting is done until the command is issued to another part system. The thread cutting cycle starts when the commands are aligned properly.

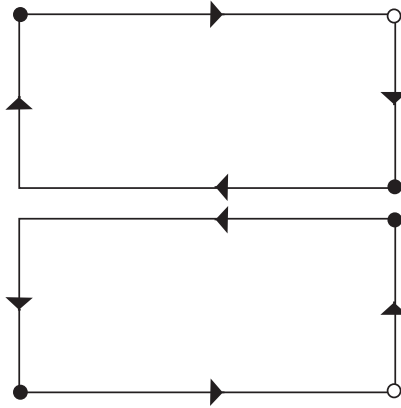


- (2) G76.2 assumes the same thread cutting, and deeply cuts in with the cutting amount using 1st part system and 2nd part system alternately.



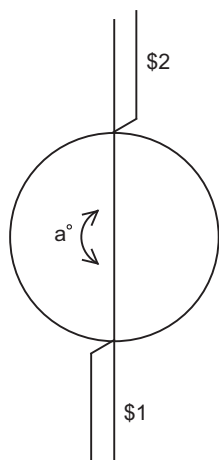
- |                                |                                  |                                  |
|--------------------------------|----------------------------------|----------------------------------|
| (a) Finishing allowance $d$    | (b) $\Delta d$                   | (c) $\Delta d \times \sqrt{2}$   |
| (d) $\Delta d \times \sqrt{n}$ | (e) Cutting with 1st part system | (f) Cutting with 2nd part system |

- (3) In one cycle, waiting is done at the start and end of the thread cutting.



● Waiting-and-simultaneous operation

- (4) The same precautions for thread cutting command (G33), thread cutting cycle (G78) and compound thread cutting cycle (G76) apply to this cycle.
- (5) G76.2 is the same thread cutting, so the various parameters, thread section, taper height, screw thread height, cutting amount and thread lead must be commanded to the same values for 1st part system and 2nd part system.  
Note that the start shift angle can be commanded to match the thread cutting state.
- (6) Thread cutting controls the Z axis position while tracking the spindle encoder rotation. Thus, the relative relation of the spindle position detected by the spindle encoder and the Z axis will change with the following elements.  
(a) Z axis feedrate (spindle rotation speed * screw pitch)  
(b) Cutting feed acceleration/deceleration time constant  
(c) Position loop gain  
Thus, with G76.2 which is same thread cutting, the parameters must be set so that the conditions are the same for 1st part system and 2nd part system.
- (7) Thread cutting start shift angle command



As shown on the left, if the 1st part system and 2nd part system blades oppose at 180 degree, the 1st part system and 2nd part system thread cutting start shift angle difference is set to 180 degree.

(Example)

```
$1
.
.
G76.2 X_ Z_ A0.;
.
.
```

```
$2
.
.
G76.2 X_ Z_ A180.;
.
.
```

\$1 : 1st part system

\$2 : 2nd part system

- (8) When G76.2 and G76.1 are commanded

The part systems, in which each are commanded, will carry out the G76.1 and G76.2 movements.

However, the part system in which G76.2 is commanded will assume that the other part system is using G76.2 when cutting the threads, so the thread grooves will not be guaranteed.

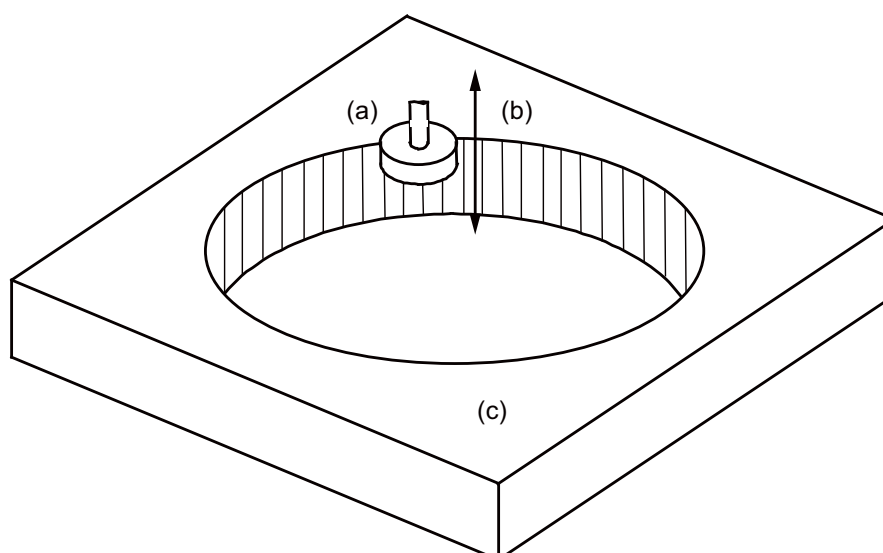
## 13.19 Chopping ; G81.1



### Function and purpose

This function continuously raises and lowers the chopping axis independently of the program operation when workpiece contours are to be cut.

There are two types of command for the chopping function: a command by the machining program and a command by a signal from the PLC.



- (a) Grindstone
- (b) Chopping operation
- (c) Workpiece



### Command format

#### G81.1 Z__ Q__ F__; ... Starting the chopping operation

Z	The upper dead point (Select the chopping axis with commanded axis address)
Q	The distance between the upper dead point and the lower dead point. Command with incremental.
F	The feedrate during chopping (mm/min).

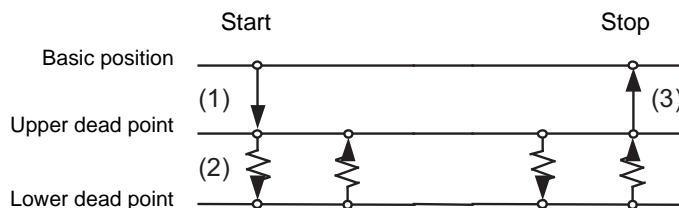
#### G80 ... Cancelling the chopping operation

**Detailed description**

- (1) When "#1323 chopsel (chopping command method)" is set to "0", the program error (P610) will occur.
- (2) Command starting chopping operation (G81.1) in independent blocks. Program error (P33) will occur when commanding G81.1 in the same block with other G command.
- (3) When starting chopping action (G81.1), a command for other address (apart from N command) cannot be issued. If other address is commanded, program error (P33) will occur.
- (4) Only one axis can be designated for the chopping axis. Program error (P33) will occur when designating two or more axes.
- (5) During the chopping operation, movement command does not work. If a movement command is issued, "M01 Operation Error 0151" will occur and all axes will be in the interlock state.
- (6) Cancelling the chopping operation (G80) can cancel the fixed cycle modal.
- (7) The position where starting the chopping operation (G81.1) is assigned will be the base position for chopping.
- (8) Rapid traverse override can be valid for the following movement; travelling from the base position to the upper dead point in starting chopping operation, and traveling from the upper dead point to the base point after the operation. Only when G81.1 commanded, rapid traverse override can be set valid/invalid by external signal. During the chopping operation, the rapid traverse override cannot be switched.
- (9) Chopping override can be set during the chopping operation.  
Chopping override is only valid for chopping axis and it does not affect other axes.  
Also, the axis in chopping operation does not get affected by other override.  
If "0 %" is assigned to the chopping override, "M01 operation error 0150" will occur.
- (10) During the chopping mode, the upper dead point/the lower dead point and feedrate can be changed by commanding G81.1. However, it is not possible to change the chopping axis. Changing chopping axis can cause the program error (P33).
- (11) The chopping operation will be started with G81.1 command, but the machine will not travel to the upper dead point/the lower dead point since the tool offset is done at the initial level of the operation. Until the error amount of command position and feedback position reach the allowable error value, checking with M commands etc. is necessary.
- (12) When speed (F) command is large and stroke of the chopping is short, the chopping operation may be done slower than the command speed.

## Chopping operation

Operation of the chopping axis



### (1) Starting the chopping operation

The chopping mode is entered by issuing the G81.1 command and the chopping operation will be initiated using the current position as a basic position. Chopping is operated after traveling from the base position to the upper dead point with rapid traverse.

### (2) During the chopping operation

The axis travels repeatedly between the upper dead point and the lower dead point by designated number of cycles or the feedrate. During the chopping operation, compensation amount is calculated from the machine operation (feedback position at the motor end) and the compensation is conducted so that the machine movement reaches the upper dead point and the lower dead point. (Refer to Chopping compensation.) Traveling during the chopping operation will be performed by soft acceleration/ deceleration.

### (3) Stopping the chopping operation

The chopping operation is stopped by issuing the G80 command.

After the chopping operation is performed to the upper dead point, the chopping axis will travel to the base point with rapid traverse. The chopping axis will travel to the lower dead point once even during the travel from the upper dead point to the lower dead point. Acceleration/deceleration will be performed by linear acceleration/deceleration for travelling from the basic point to the upper dead point.

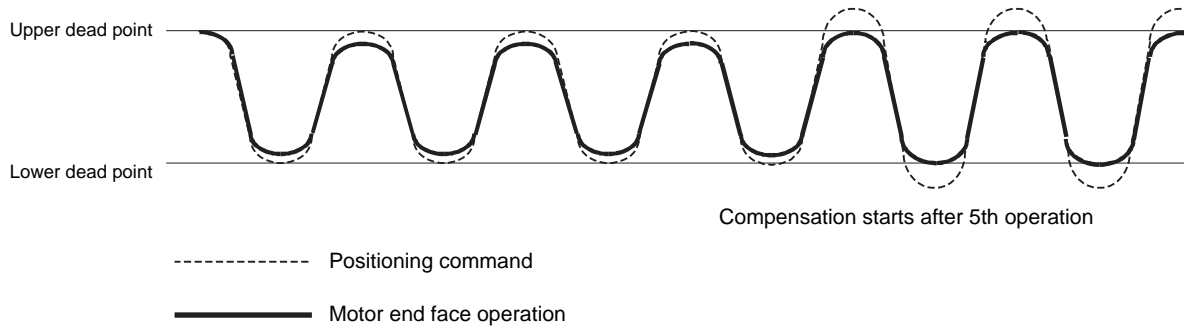
## Interrupt operation during chopping

When the interruption, which affects the chopping axis, occurs during the chopping operation, the chopping axis performs as follow.

Interrupt operation	Chopping performance	Chopping mode
Reset	Travel to the basic position with rapid traverse after travelling to the upper dead point immediately.	Cancel
Feed hold	Performance can be continued without stopping.	Hold
Block stop		
Axis interlock		
Door interlock II	The chopping operation starts again after cancelling.	
Door interlock I	Decelerates and stops.	Cancel
Servo OFF		
Axis detachment		
Stroke end		
Emergency stop	Stops immediately.	
Program error	Travel to the basic position with rapid traverse after travelling to the upper dead point immediately.	Hold

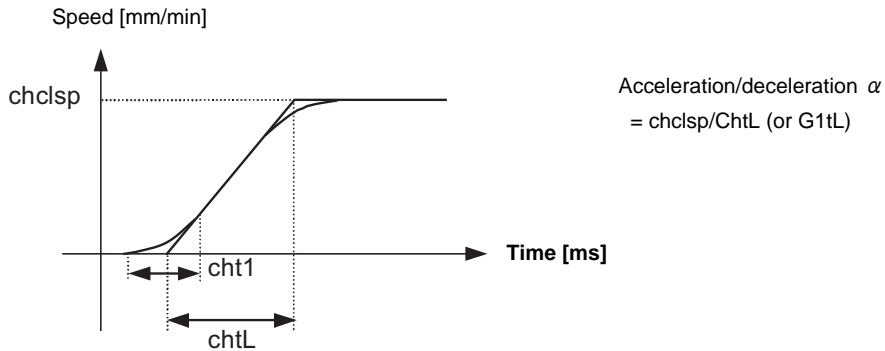
### Chopping compensation operation (Compensation value sequential update type)

This function uses the method that the compensation amount is calculated with the machine operation (feedback position of a motor end face) rather than using in-position check for assigning the position since the positioning command is compensated for high-speed repeated operation. The compensation amount for assigning the position can be calculated from the difference between command position and feedback position every four cycle. Then the compensation amount is added on the positioning command for the next cycle to eliminate the difference between command position and feedback position.



### Chopping feedrate

The feedrate of the chopping axis will be clamped at the clamp speed (#2081 chclsp) of the chopping axis. When 0 is set to the chopping clamp speed, it will be clamped at the G1 clamp speed (#2002 clamp). The acceleration/deceleration time constant can be set by chopping acceleration/deceleration time constant (#2141 chtL). When 0 is set to the chopping axis acceleration/deceleration time constant, the linear acceleration/deceleration time constant (#2007 G1tL) will be executed.





### Program example

(1) Machining condition

Chopping axis : Z axis

Basic point coordinate : (Machine) -20.0, the upper dead point : -25.0, the lower dead point : -45.0

Chopping speed : 1000[mm/min]

(2) Program

O1000();	
N0010 G54;	
N0020 G28 X0 Y0 Z0;	
N0030 G00 U10. V10.;	
N0040 G53 W-20.;	Z-20.0 of the machine coordinate is the basic point
N0050 G81.1 Z-5. Q-20. F1000;	Command Z axis to the chopping axis
N0060 M70;	Waiting to be finished chopping compensation
N0070 G00 X-10. Y-10.;	
N0080 G01 X50. F300;	
N0090 G01 Y50.;	
N0100 G01 X-50.;	
N0110 G01 Y-50.;	
N0120 G00 X10. Y10.	
N0130 G80;	Chopping axis returns to the basic point
N0140 G28 Z0.;	
N0150 G28 X0 Y0;	
N0160 M30;	





## Coordinate System Setting Functions

## 14.1 Coordinate Words and Control Axes



### Function and purpose

In the case of a lathe, axis names (coordinate words) and directions are defined as follows.

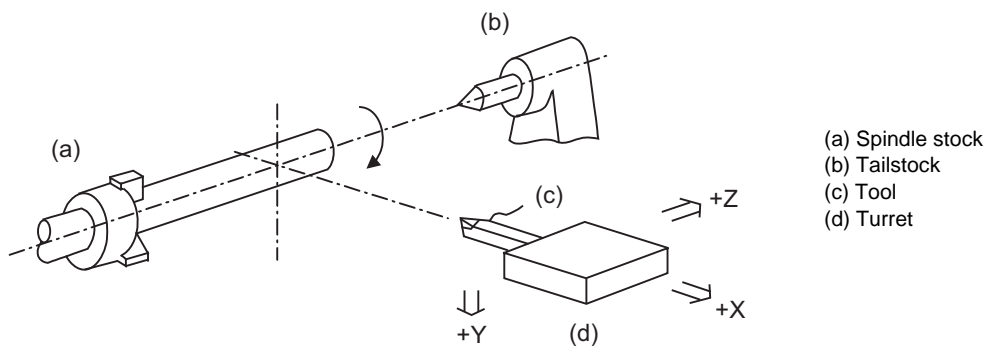
The axis at right angles to the spindle

Axis name: X axis

The axis parallel to the spindle

Axis name: Z axis

### Coordinate axes and polarities

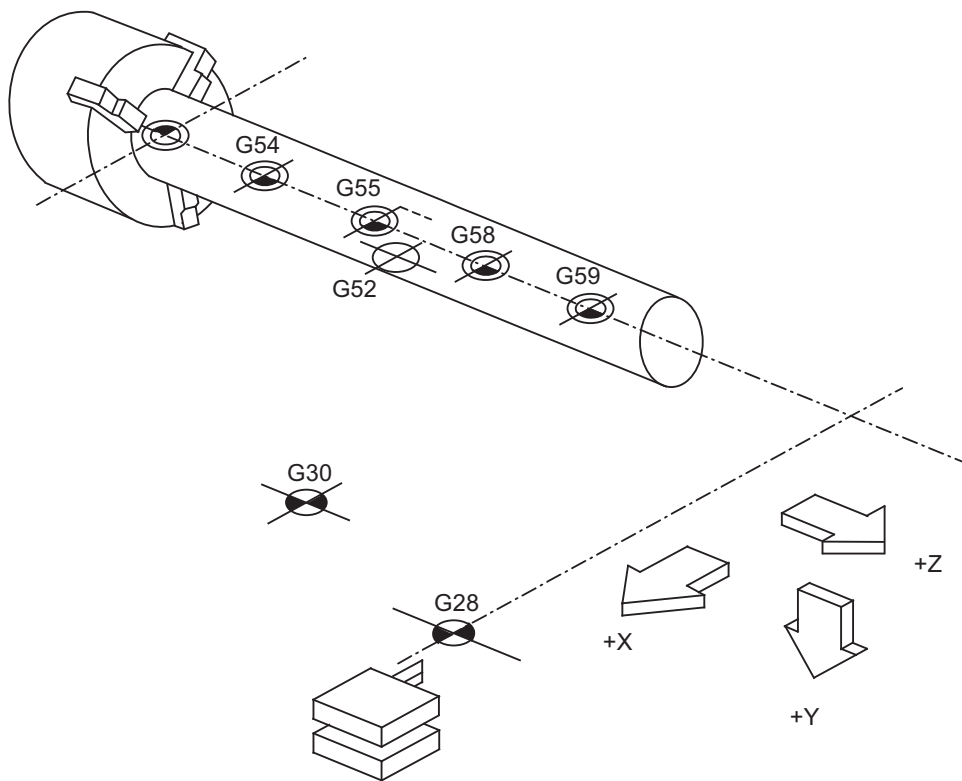






Since coordinates based on the right hand rule are used with a lathe, in the above figure, the positive direction of the Y axis which is at right angles to the X-Z plane is downward.

Note that a circular on the X-Z plane is expressed as clockwise or counterclockwise as seen from the forward direction of the Y axis.

(Refer to the section on circular interpolation.)

Relationship between coordinates



-  Reference position
-  Basic machine coordinate
-  Workpiece coordinate zero points
-  Local coordinate zero point

## 14.2 Basic Machine, Workpiece and Local Coordinate Systems



### Function and purpose

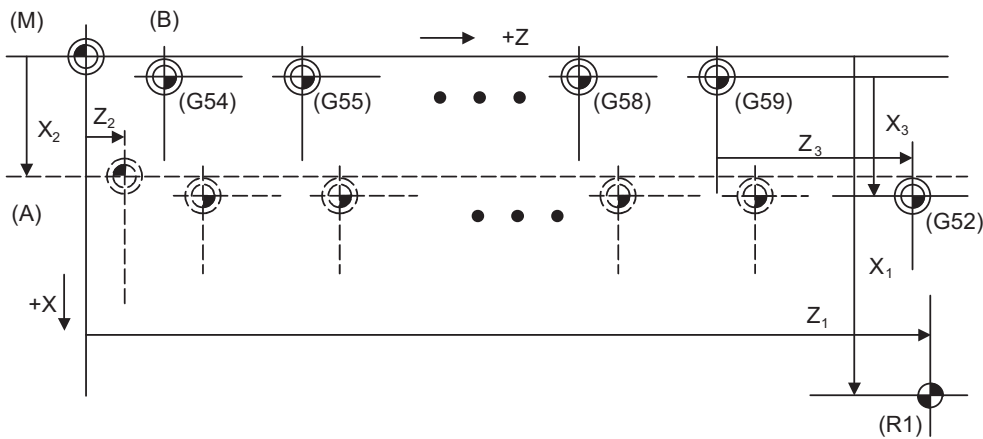
The basic machine coordinate system is fixed in the machine and it denotes that position which is determined inherently by the machine.

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).



- |       |                                                    |       |                               |
|-------|----------------------------------------------------|-------|-------------------------------|
| (A)   | Hypothetical machine coordinate system (G92 shift) | (B)   | Machine zero point            |
| (G54) | Workpiece coordinate system 1                      | (G55) | Workpiece coordinate system 2 |
| (G58) | Workpiece coordinate system 5                      | (G59) | Workpiece coordinate system 6 |
| (G52) | Local coordinate system                            | (R1)  | 1st reference position        |
| (M)   | Basic machine 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.

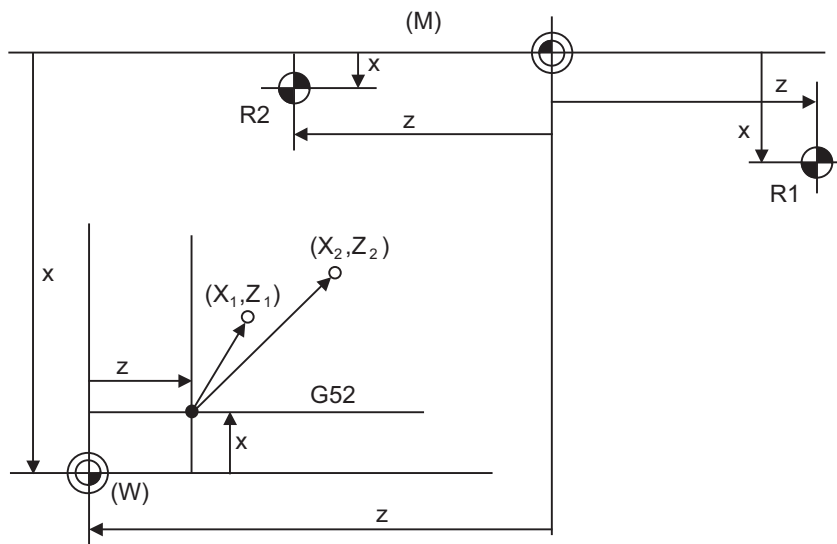
## 14.3 Machine Zero Point and 2nd Reference Position (Zero point)



### 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 reference position (zero point) relates to the position of the coordinates which have been set beforehand by parameter from the zero point of the basic machine coordinate system.



(M) Basic machine coordinate system (G52) Local coordinate system

(R1) 1st reference position

(R2) 2nd reference position

(W) Workpiece coordinate systems (G54 to G59)

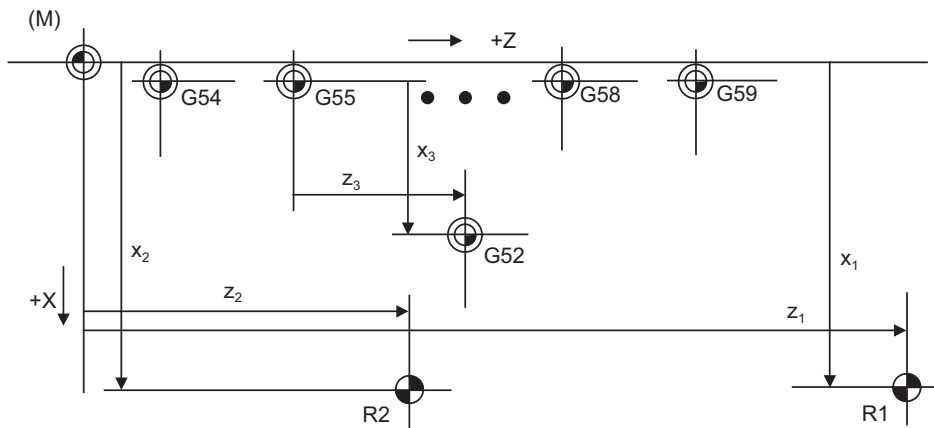
## 14.4 Automatic Coordinate System Setting



### Function and purpose

This function creates each coordinate system according to the parameter values input beforehand from the setting and display unit when the reference position is reached with the first manual reference position return or dog-type reference position return when the NC power is turned ON.

The actual machining program is programmed over the coordinate systems which have been set above.



- |                                     |                                     |                                     |
|-------------------------------------|-------------------------------------|-------------------------------------|
| (M) Basic machine coordinate system | (R1) 1st reference position         | (R2) 2nd reference position         |
| (G54) Workpiece coordinate system 1 | (G55) Workpiece coordinate system 2 | (G58) Workpiece coordinate system 5 |
| (G59) Workpiece coordinate system 6 | (G52) Local coordinate system       |                                     |



### 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)  
The Local coordinate system (G52) is canceled.
- (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 tool is moved to the position commanded on the basic machine coordinate system, using the G53 command, the feed mode command (G01 or G00) and the coordinate command that follow them.



### Command format

```
G53 G00 X__ Z__ α __;
```

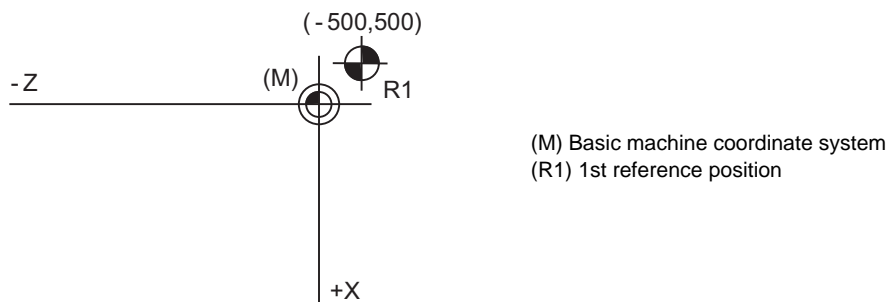
```
G53 G00 U__ W__ β __;
```

$\alpha$	Additional axis
$\beta$	Incremental command of the additional axis



### Detailed description

- (1) When the power is turned ON, the basic machine coordinate system is automatically set as referenced to the reference position (zero point) return position, which is determined by the automatic or manual reference position (zero point) return.
- (2) The basic machine coordinate system is not changed by the G92 command.
- (3) The G53 command is valid only in the designated block.
- (4) When an incremental value is issued (U, W,  $\beta$ ), the axis will move with the incremental value within the selected coordinate system.
- (5) The 1st reference position coordinate value indicates the distance from the basic machine coordinate system zero point to the reference position (zero point) return position.



1st reference position coordinate value: X = -500 and Z=+500



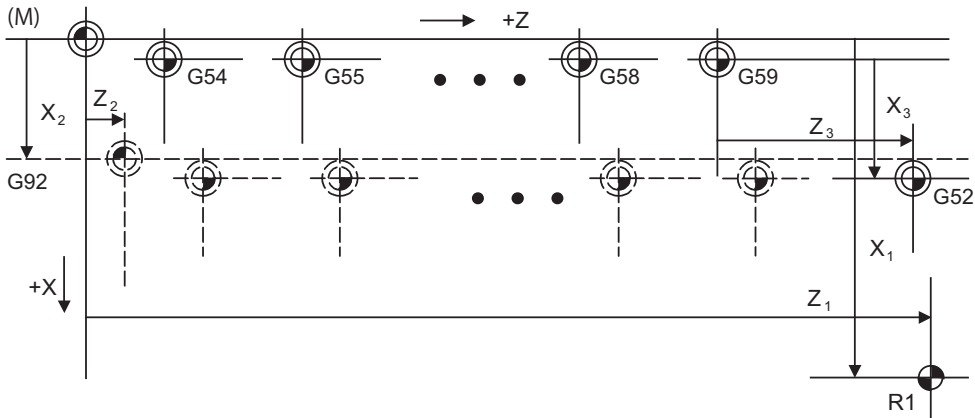
## 14.6 Coordinate System Setting ; G92



### Function and purpose

This function places the tool at the desired position, and the coordinate system is set by assigning the coordinate system setting command G92 at that position.

This system can be set as desired though normally the X and Y axes are set so that the workpiece center serves as the zero point and the Z axis is set so that the workpiece end serves as the zero point.



- (M) Basic machine coordinate system      (G92) Hypothetical machine coordinate system (shifted by G92)  
 (G54) Workpiece coordinate system 1      (G55) Workpiece coordinate system 2      (G58) Workpiece coordinate system 5  
 (G59) Workpiece coordinate system 6      (G52) Local coordinate system      (R1) 1st reference position



### Command format

**G92 Xx2 Zz2 α α 2;**

α α	Additional axis
-----	-----------------



### Detailed description

- (1) The basic machine coordinate system is shifted by the G92 command, the hypothetical machine coordinate system is created, and at the same time all workpiece coordinate systems 1 to 6 are also shifted.
- (2) When G92 and S or Q are assigned, the spindle clamp rotation speed is set. (Refer to the section on setting the spindle clamp rotation speed.)

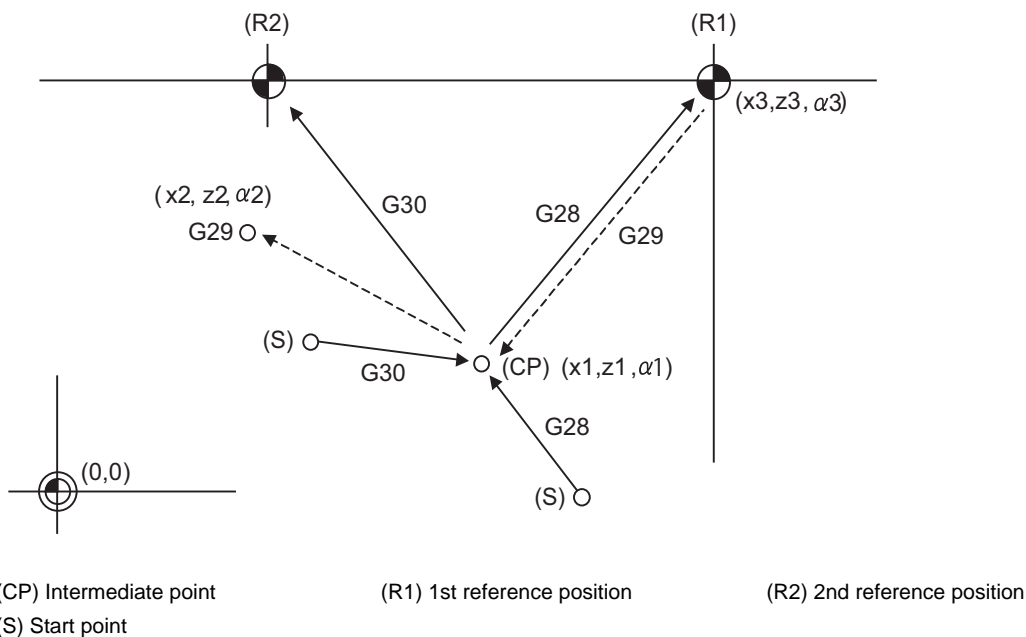
## 14.7 Reference Position (Zero point) Return ; G28,G29



### 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 at the commanded position.



### Command format

<b>G28 Xx1 Zz1 α α 1; ... Automatic reference position return</b>	
<b>G29 Xx2 Zz2 α α 2; ... Start point return</b>	
α α 1/ α α 2	Additional axis



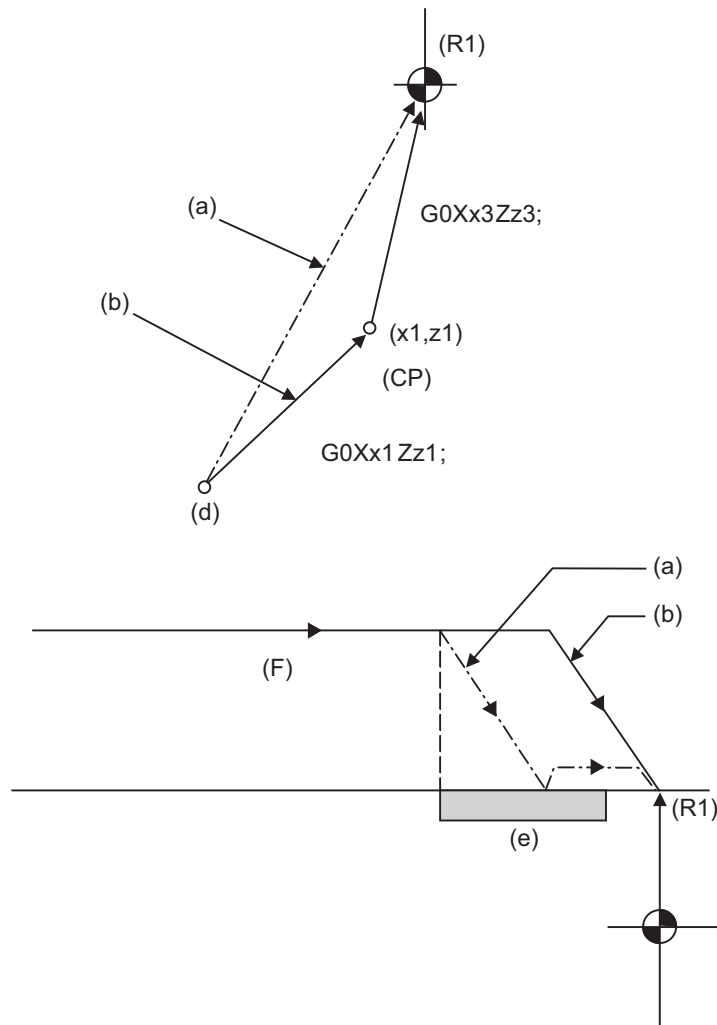
## Detailed description

- (1) The G28 command is equivalent to the following:  
 $G00 Xx1 Zz1 \alpha \alpha 1 ;$   
 $G00 Xx3 Zz3 \alpha \alpha 3 ;$   
 In this case, x3, z3 and  $\alpha 3$  are the reference position coordinates and they are set by parameters "#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. For the second and subsequent returns, the return is made at high speed to the reference position (zero point) which was stored at the first time.
- (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:  
 $G00 Xx1 Zz1 \alpha \alpha 1 ;$   
 $G00 Xx2 Zz2 \alpha \alpha 2 ;$   
 The rapid traverse (non-interpolation type) independent for each axis takes place.  
 In this case, x1, z1 and  $\alpha 1$  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) The intermediate point coordinate values (x1, z1,  $\alpha 1$ ) of the positioning point are assigned by absolute/incremental value commands.
- (7) 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.
- (8) The tool offset will be temporarily canceled during reference position return unless it is already canceled, and the intermediate point will be the compensated position.
- (9) The intermediate point can be ignored by parameter "#1091 Mpoint" setting.
- (10) 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.
- (11) 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.



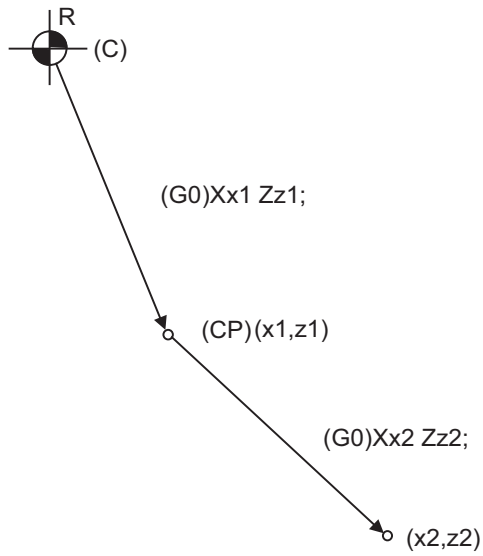
Program example

(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                          | (R1) Reference position (#1)      |
| (CP) Intermediate point                          |                                   |

(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)

:

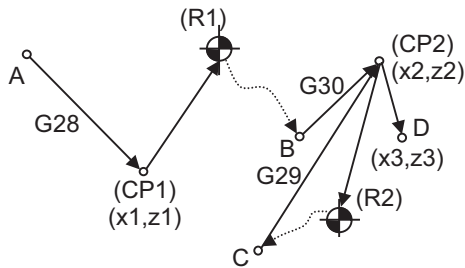
G30 Xx2 Zz2 ;

: (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)

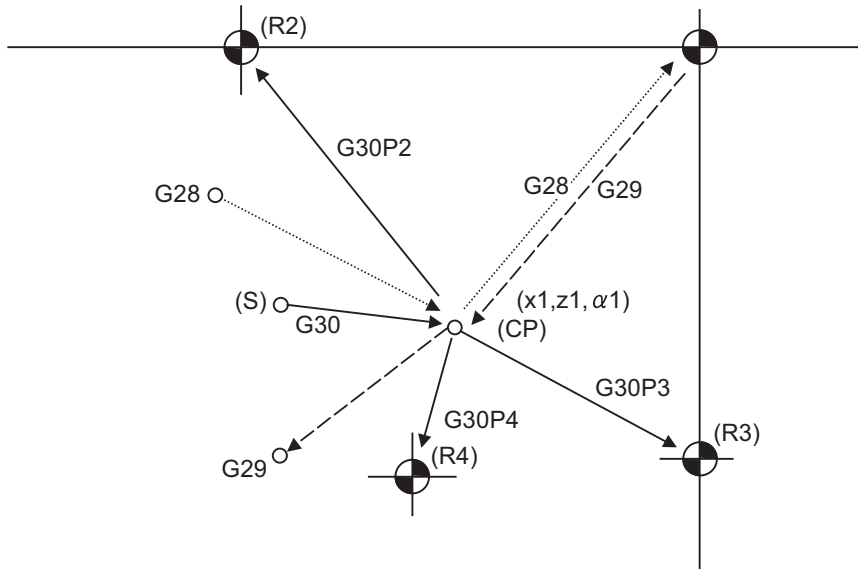
(R2) 2nd reference position (#2)

## 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 Zz1 α α 1;**

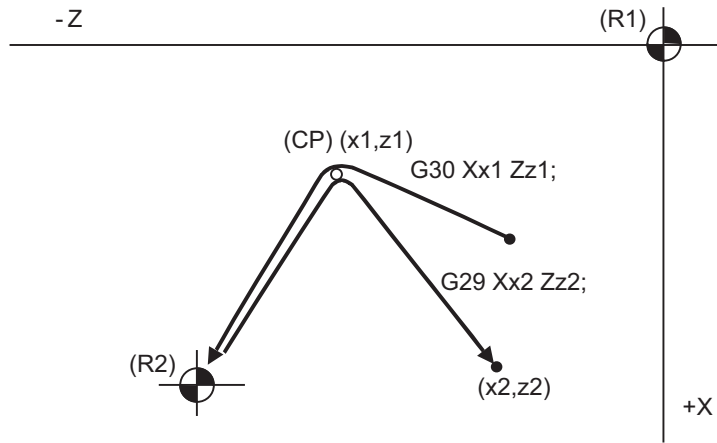
α α 1

Additional axis



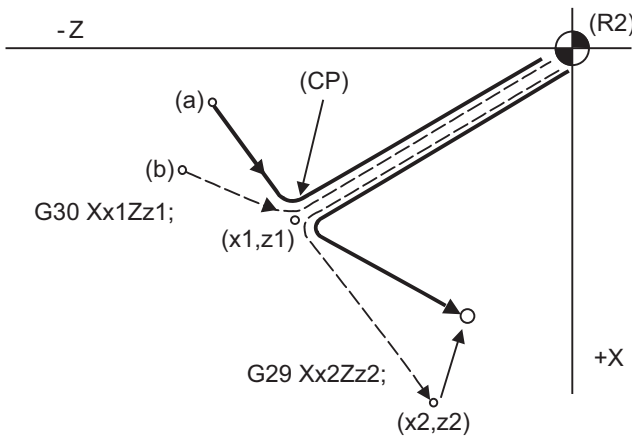
Detailed description

- (1) The 2nd, 3rd, or 4th reference position return is specified by P2, P3, or P4.  
A command without P or with other designation method will return the tool to the 2nd reference position.
- (2) In the 2nd, 3rd, or 4th reference position return mode, as in the 1st reference position return mode, the tool returns to the 2nd, 3rd, or 4th reference position via the intermediate point specified by G30.
- (3) The 2nd, 3rd, and 4th reference position coordinates refer to the positions specific to the machine, and these can be checked with the setting and display unit.
- (4) If G29 is commanded after completion of returning to the 2nd, 3rd, and 4th reference position, the intermediate position used last is used as the intermediate position for returning by G29.



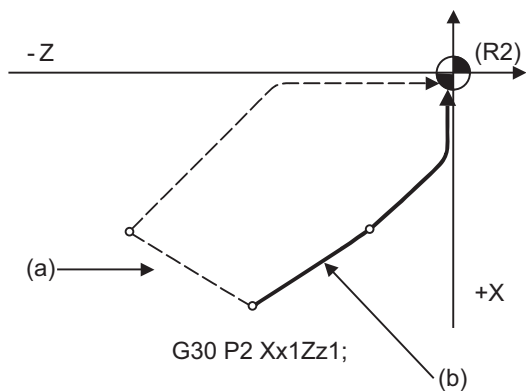
(CP) Intermediate point                      (R1) 1st reference position                      (R2) 2nd reference position

- (5) With reference position return on a plane during compensation, the tool moves without tool nose radius compensation (zero compensation) from the intermediate point as far as the reference position. With a subsequent G29 command, the tool move without tool nose radius compensation from the reference position to the intermediate point and it moves with such compensation until the G29 command from the intermediate point.



(a) Tool nose center path                      (b) Program path  
(CP) Intermediate point                      (R1) 1st reference position                      (R2) 2nd reference position

- (6) The tool length offset amount for the axis involved is temporarily canceled after the 2nd, 3rd and 4th reference position return.
- (7) With second, third and fourth reference (zero) point returns in the machine lock status, control from the intermediate point to the reference (zero) point will be ignored. When the designated axis reaches as far as the intermediate point, the next block will be executed.
- (8) With second, third and fourth reference position returns in the mirror image mode, mirror image will be valid from the start point to the intermediate point and the tool will move in the opposite direction to that of the command. However, mirror image is ignored from the intermediate point to the reference position and the tool moves to the reference position.



(a) X-axis mirror image

(b) No mirror image

(R2) 2nd reference position



## 14.9 Reference Position Check ; G27



### Function and purpose

This command first positions the tool at the position assigned by the program and then, if that positioning point is the 1st reference position, it outputs the reference position arrival signal to the machine in the same way as with the G28 command. Therefore, when a machining program is prepared so that the tool will depart from the 1st reference position and return to the 1st reference position, it is possible to check whether the tool has returned to the reference position after the program has been run.



### Command format

**G27 X__ Z__  $\alpha$ __ P__ ; ... Check command**

X Z $\alpha$	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.
- (3) An alarm will occur if the reference position is not reached after the command is completed.

## 14.10 Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1)



### Function and purpose

- (1) The workpiece coordinate systems facilitate the programming on the workpiece, serving the reference position of the machining workpiece as the zero point.
- (2) These commands enable the tool to move to the positions in the workpiece coordinate system. There are 6 workpiece coordinate systems, which are used by the programmer for programming (G54 to G59).
- (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 "current position of the tool" includes the compensation amounts for tool nose R, tool length and tool position compensation.)
- (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 "current position of the tool" includes the compensation amounts for tool nose R, tool length and tool position compensation.) (G54,G92)



### Command format

**G54 to G59 ... Workpiece coordinate system selection**

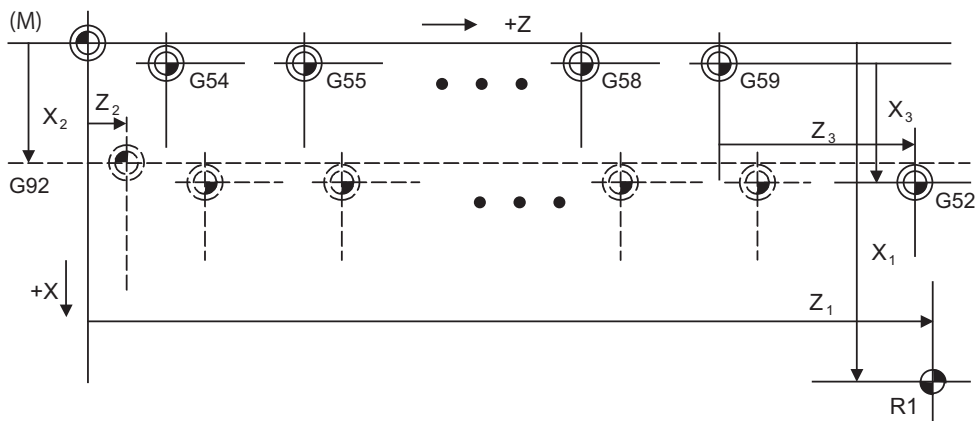
**(G54 to G59) G92 X__ Z__  $\alpha$  __ ; ... Workpiece coordinate system setting**

$\alpha$	Additional axis
----------	-----------------



**Detailed description**

- (1) With any of the G54 through G59 commands, the nose radius offset amounts for the commanded axes will not be canceled even if workpiece coordinate system selection is commanded.
- (2) The G54 workpiece coordinate system is selected when the power is turned ON.
- (3) Commands G54 through G59 are modal commands (group 12).
- (4) The coordinate system will move with G92 in a workpiece coordinate system.
- (5) The offset setting amount in a workpiece coordinate system denotes the distance from the basic machine coordinate system zero point.



- (M) Basic machine coordinate system      (G92) Hypothetical machine coordinate system (shifted by G92)
- (G54) Workpiece coordinate system 1      (G55) Workpiece coordinate system 2
- (G58) Workpiece coordinate system 5      (G59) Workpiece coordinate system 6
- (G52) Local coordinate system              (R1) 1st reference position

- (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 Zz1.)

**[Handling when L or P is omitted]**

G10 L2 Pn Xx 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 Zz ;	Set the offset amount in the currently selected workpiece coordinate system.
G10 L20 Xx 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 Zz ; G10 Xx Zz ;	L10 (tool offset) will be judged if there is no L value.

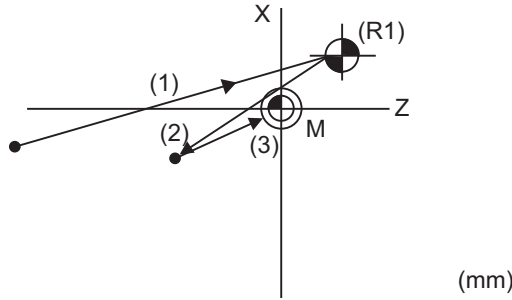




Program example

(Example 1)

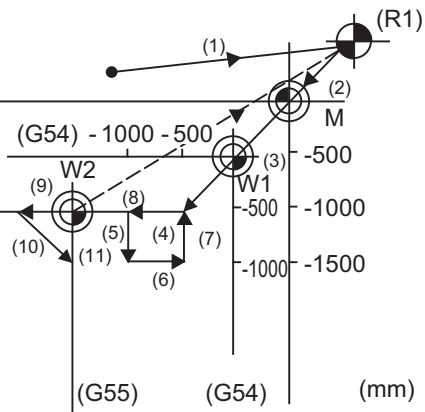
- (1) G28 X0Z0;
- (2) G53 X-500 Z-1000;
- (3) G53 X0Z0;



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.

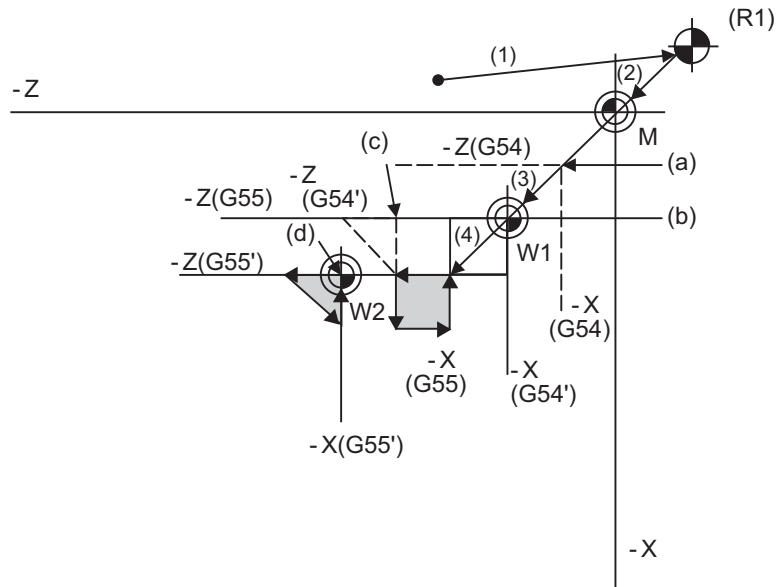
(Example 2)

- (1) G28X0Z0;
- (2) G00G53X0Z0;
- (3) G54X - 500Z - 500;
- (4) G01W - 500F 100;
- (5) U - 500;
- (6) W+500;
- (7) U+500;
- (8) G00G55X0Z0;
- (9) G01Z - 500 F200;
- (10) X - 500 Z0;
- (11) G28X0Z0;



(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 Z0
- (2) G00 G53 X0 Z0 ; (Not required when there is no basic machine coordinate system offset.)
- (3) G54 X-500 Z-500 ; Amount by which workpiece coordinate system deviates
- (4) G92 X0 Z0 ; 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.

### 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.



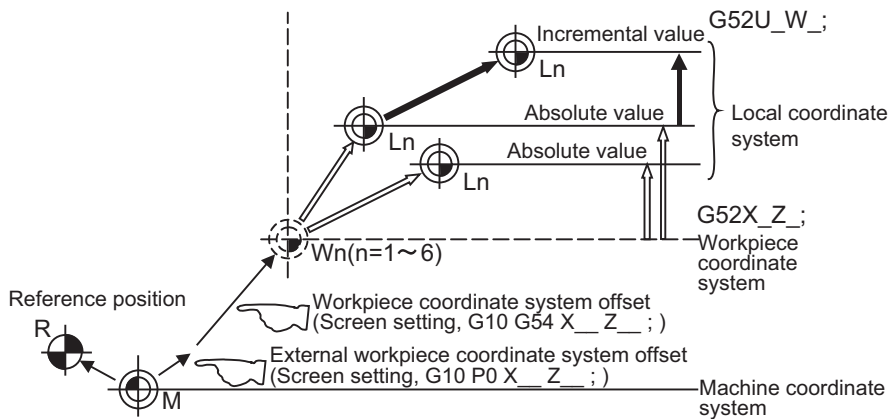
#### Command format

G54(G54 to G59) G52 X__ 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 Z0;.
- (4) Coordinate commands in the absolute value 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.

## 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	
#8213 setting value	0	1	2	-
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  $0^\circ$  to  $359.999^\circ$ .  
The workpiece coordinate position is displayed in the range of 0 to  $\pm 99999.999^\circ$ .
- (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  $360^\circ$  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  $359.999^\circ$ , the system will start up in the range of 0 to  $359.999^\circ$  when the power is turned ON again.

	<b>Program</b>	<b>Workpiece</b>	<b>Machine</b>	<b>Relative position</b>
	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			
	<b>Workpiece</b>	<b>Machine</b>		
	225.000	225.000		



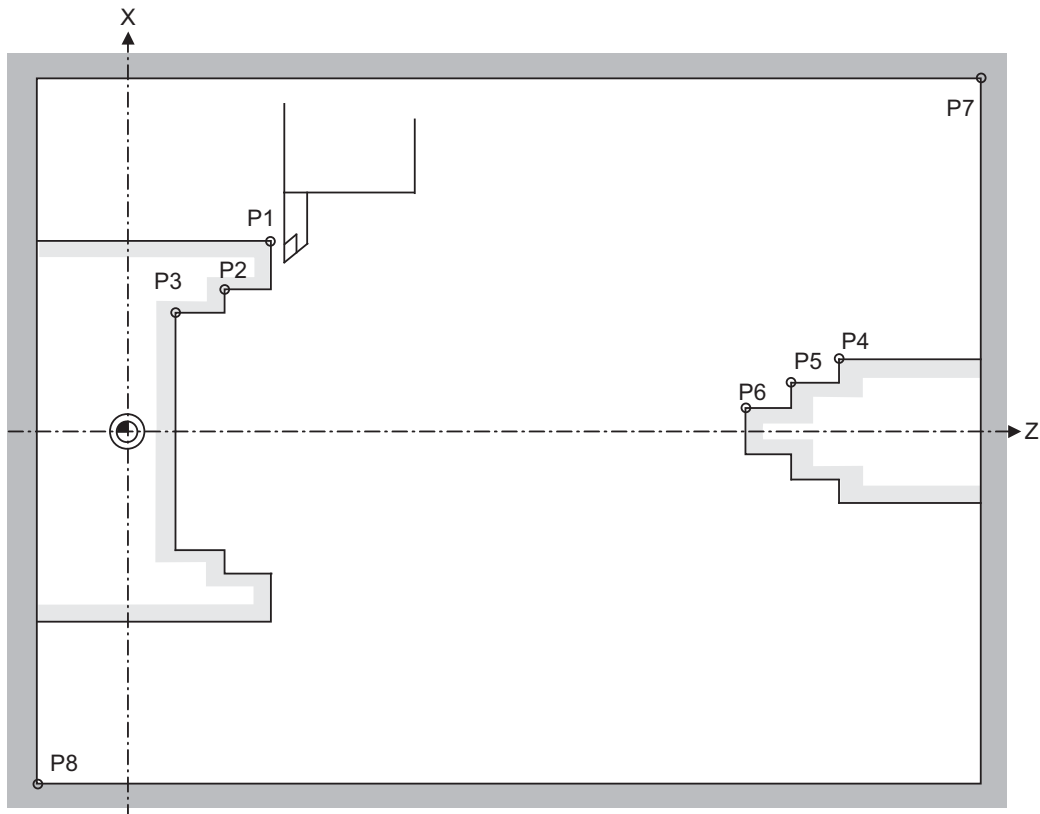
## Protection Function

## 15.1 Chuck Barrier/Tailstock Barrier ; G22,G23



### Function and purpose

By limiting the tool nose movement range, the chuck barrier and tailstock barrier prevent collision with the chuck and tailstock due to programming errors. If movement is commanded which exceeds the region set by the parameters, the tool will automatically stop at the barrier boundary.



P1,P2,P3 : Chuck barrier  
 P4,P5,P6 : Tailstock barrier  
 P7,P8 : Stored stroke limit



### Command format

**G22 ; ... Barriers valid**

**G23 ; ... Barriers invalid**

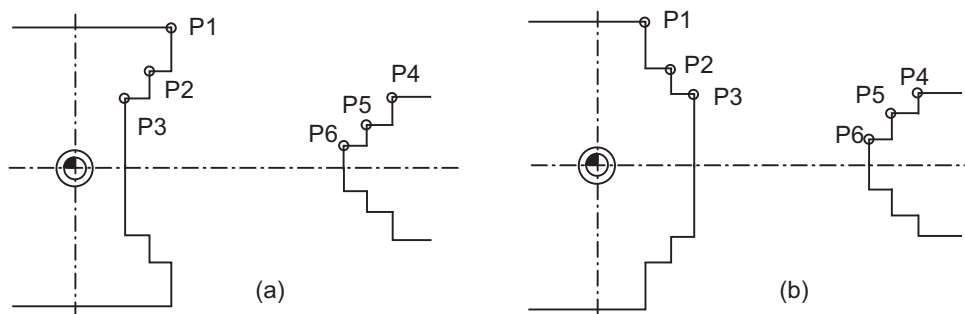
Command G22 and G23 in independent blocks.



### Detailed description

- (1) An alarm will appear at the same time as the machine stops because it was about to exceed the set region.  
Reset to cancel this alarm.
- (2) This function is also valid during machine lock.
- (3) This function is validated when all axes in which chuck barrier and tailstock barrier are set have finished their reference position returns.
- (4) When there is a stored stroke check function, and the stored stroke limit region is set, the chuck barrier/tailstock barrier function is validated simultaneously with the stored stroke check function.

### Setting when using G22 and G23



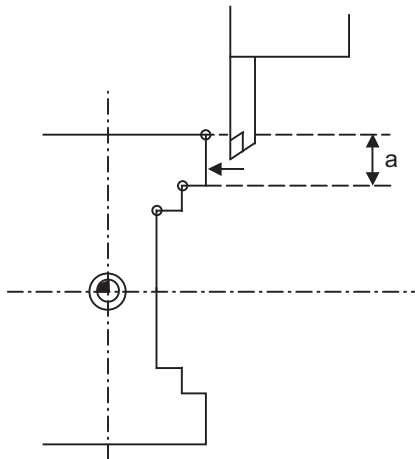
- (1) Three points can be input as parameters for both the chuck barrier and tailstock barrier. Set them in the machine coordinate system.  
Points P1, P2 and P3 (parameters "#8301 P1" to "#8303 P3") are for the chuck barrier. Points P4, P5 and P6 (parameters "#8304 P4" to "#8306 P6") are for the tailstock barrier.
- (2) The barrier region should be a symmetric shape regarding the Z axis. When the X axis coordinates of barrier point P_n are a negative value, reverse the sign to the positive side, then convert and check.  
The absolute value of each barrier point's X axis coordinates must be set as follows.  
 $P1 \geq P2 \geq P3, P4 \geq P5 \geq P6$   
(Note that the Z axis coordinates do not have to follow this setting.)



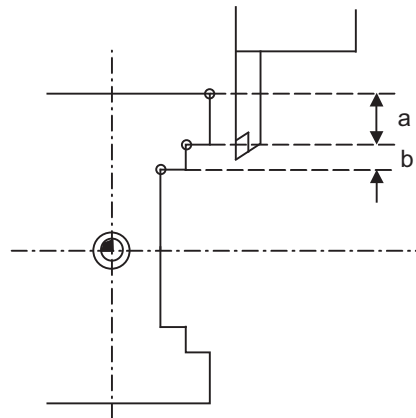
### Precautions

- (1) There is only one checkpoint from the tool regarding the chuck barrier/tailstock barrier. Therefore, the following cautions must be observed.  
In the following examples, when the barrier points are set to be checked by the hypothetical tool nose point and the tool moves in the direction of the arrow in the drawing, the following situation may occur. In Example 1, there is a checkpoint in the range "a", so the tool will automatically stop at the barrier boundary. However, in Example 2 there is a checkpoint in the range "b", so the chuck and tool may collide in the range "a".

(Example 1)

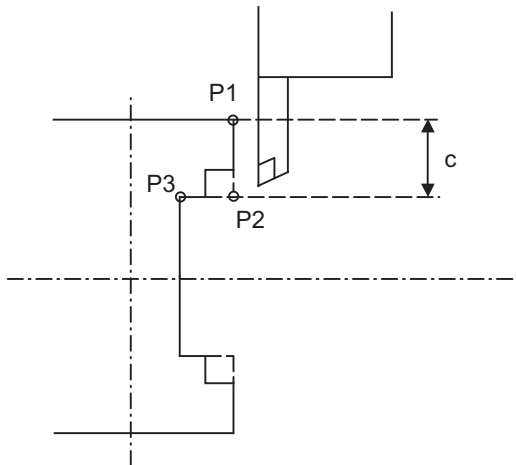


(Example 2)



To avoid this, Example 3 is given. In this example, if the barrier points P1, P2 and P3 are set and the checkpoint is set in range "c", the tool can be stopped at the barrier boundary.

(Example 3)



- (2) When the tool enters the barrier region and an alarm occurs, the tool may move in the opposite direction from which it came, once the alarm is canceled by resetting.
- (3) There is no barrier region for axes without a reference position return function. Thus, there is no barrier alarm for that axis.
- (4) When the tool enters a canceled barrier region, and that barrier is then validated, an alarm will occur immediately if the tool is moved.  
In this case, after canceling the alarm with reset and then invalidate the barrier (G23) before escaping or change the value set for each barrier point.
- (5) The soft limit is valid even if the barrier is invalid (G23).

## Measurement Support Functions



## 16.1 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 wear compensation amount.



### Command format

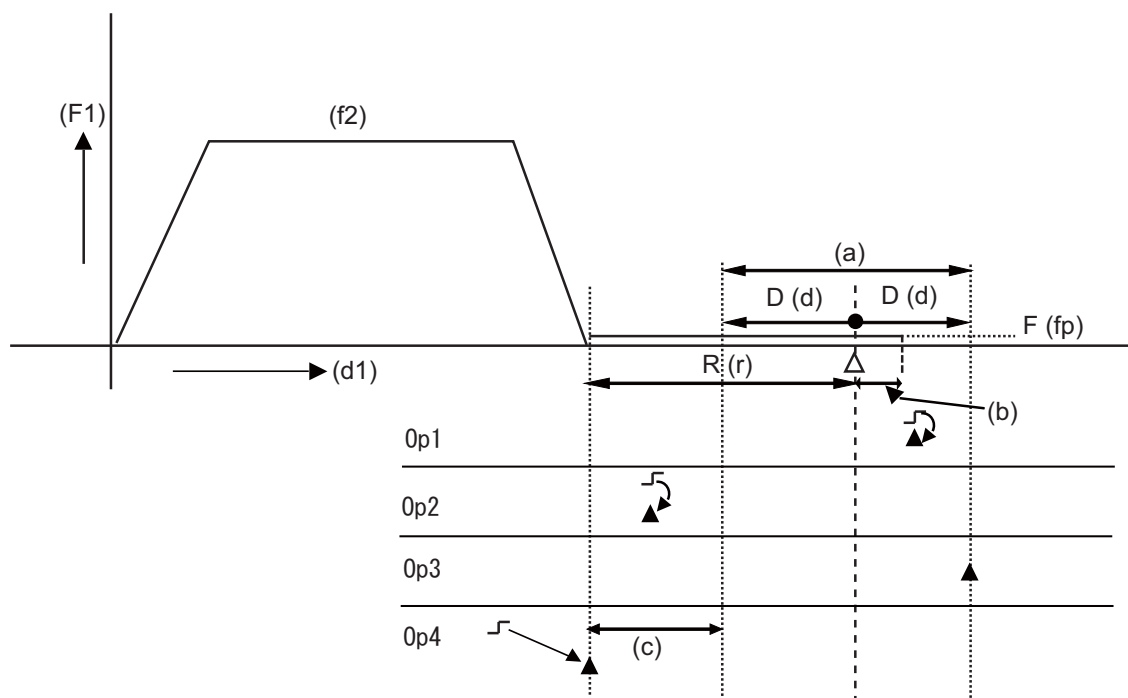
#### G37 $\alpha$ _ R _ D _ F _ ; ... Automatic tool length measurement command

$\alpha$	Measuring axis address and coordinates of measurement position ----- X, Z
R	This commands the distance between the measurement position and point where the movement is to start at the measuring speed. (Radius value fixed, incremental value)
D	This commands the range within which the tool is to stop. (Radius value fixed, incremental value)
F	This commands the measuring feedrate. When R_, D_ or F_ is omitted, the value set in the parameter is used instead. <Parameter> ("AUTO TLM" on machining parameter screen) - #8004 SPEED (measuring feedrate): 0 to 60000 [mm/min] - #8005 ZONE r: 0 to 99999.999 [mm] - #8006 ZONE d: 0 to 99999.999 [mm]



### Detailed description

#### (1) Operation with G37 command



Op1 : Normal completion as it is measurement within the allowable range.

Op2 : Alarm stop (P607) as it is outside of the measurement allowable range.

Op3 : Alarm stop (P607) as the sensor is not detected.

Op4 : Alarm stop (P607) as it is outside of the measurement allowable range. However if there is no (c) area, normal completion will occur.

(a) Measurement allowable range

(b) Compensation amount

(d1) Distance

(F1) Speed

(f2) Rapid traverse rate

(d) Measurement range

(r) Deceleration range

△ Measuring position

▲ Stop point

┌ Sensor output

(2) The sensor signal (measuring position arrival signal) is used in common with the skip signal.

(3) The feedrate will be 1mm/min if the F command and parameter measurement speed are 0.

(4) During the synchronous feed mode, the axis will move at the synchronous feedrate [mm/rev].

(5) An updated offset amount is valid unless it is assigned from the following T command of the G37 command.

(6) 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.

$$\text{Maximum measuring error [mm]} = \text{Measuring speed [mm/min]} * 1/60 * 0.2 \text{ [ms]}/1000$$

(7) 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).



## 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 T code is commanded in the G37 block. If the last one digit or last two digits is 0, the (4) error will occur.
- (4) Program error (P606) will occur when T code is not commanded prior to the G37 block. If the last one digit or last two digits is 0, the (P606) error will occur even if T is commanded.
- (5) Program error (P607) will occur when the sensor signal was input outside the allowable measuring range or when the sensor signal was not detected even upon arrival at the end point.  
If the sensor signal stays ON during operation 3 in the above example, it will be judged as a normal measurement even when the (b) region is not present.
- (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 (6) 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 (6) 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) Always cancel nose R compensation before commanding G37.



## 16.2 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/U__ Z/W__ F__ ;**

X, Z, U, W	Axis coordinate value; they are commanded with the absolute or the incremental values.
F	Feedrate (mm/min)



### Detailed description

- (1) If Ff is assigned as the feedrate in the same block as the G31 command block, command feed f will apply; if not assigned, the value set in the parameter "#1174 Skip_F" will serve as the feedrate. In either case, the F modal will not be updated.
- (2) The machine will not automatically accelerate and decelerate with the G31 block. However, setting the base specification parameter "#21101 add01/bit3" to "1" allows the automatic acceleration/deceleration valid. In such case, the acceleration/deceleration will apply following to the cutting feed acceleration/deceleration pattern set with the axis specification parameter "#2003 smgst". Since the deceleration at skip signal input follows the cutting feed acceleration/deceleration pattern mentioned above, the coasting amount from the skip signal input to stop may be larger than the normal specifications (when automatic acceleration/deceleration is invalid)
- (3) The stop conditions (feed hold and stroke end) are valid.
- (4) The G31 command is unmodal and it needs to be commanded each time.
- (5) 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.
- (6) When the G31 command is issued during tool nose radius compensation, the program error (P608) will occur.
- (7) When there is no F command in the G31 command and the parameter speed is also zero, the program error (P603) will occur.
- (8) 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.
- (9) With the normal specifications, override and dry run are invalid during execution of G31 block. However, setting the base specification parameter "#21101 add01/bit3" to "1" allows the override and dry run.
- (10) Signal input contact can be selected, depending on the parameter "#1258 set30/bit0 skip I/F changeover" setting.
  - 0: A contact (Skip operation will be performed as the signal turns ON.)
  - 1: B contact (Skip operation will be performed as the signal turns OFF.)

### Readout of skip coordinates

The coordinate positions for which the skip signal is input are stored in the system variables #5061 (1st axis) to #506n (n-th axis), so these can be used in the user macros.

```

:
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 to start deceleration to 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)

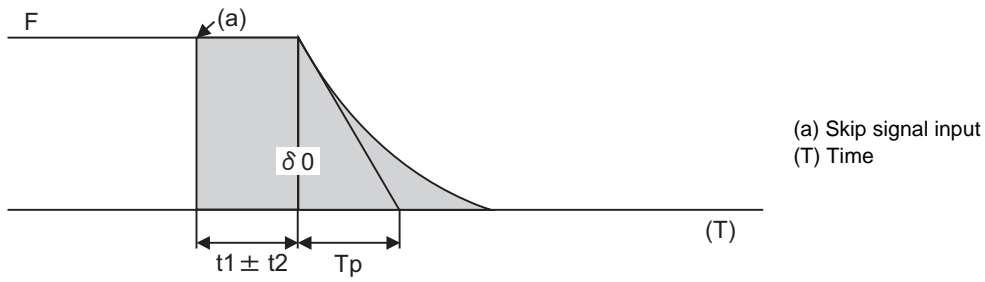
$T_p$  : Position loop time constant (s) = (position loop gain)⁻¹

$t_1$  : Response delay time (s) = (time taken from the detection to the arrival of the skip signal at the controller via PC)

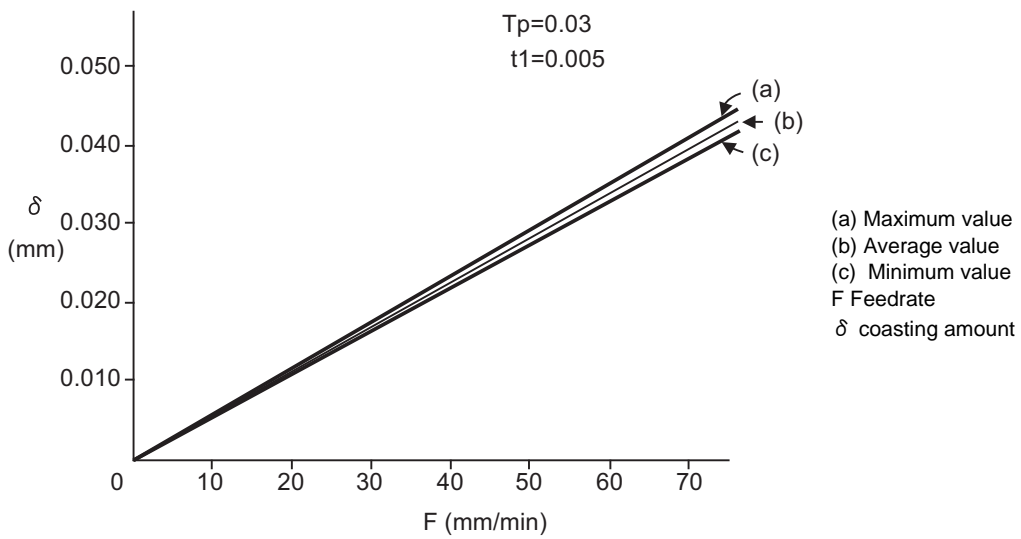
$t_2$  : Response error time 0.001 (s)

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.



(Note) When the base specification parameter "#21101 add01/bit3" is set to "1", the automatic acceleration/deceleration becomes valid for the deceleration at skip signal input. Thus, the coasting amount from the skip signal input to stop may be larger than when the automatic acceleration/deceleration is invalid.

### Readout error of skip coordinates mm

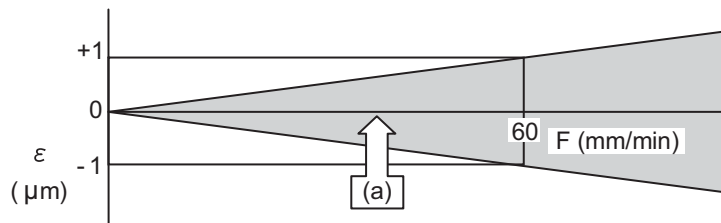
#### (1) Skip signal input coordinate readout

The coasting amount based on the position loop time constant  $T_p$  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.

$$\varepsilon = \pm (F/60) * t_2$$

$\varepsilon$  : 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;

```

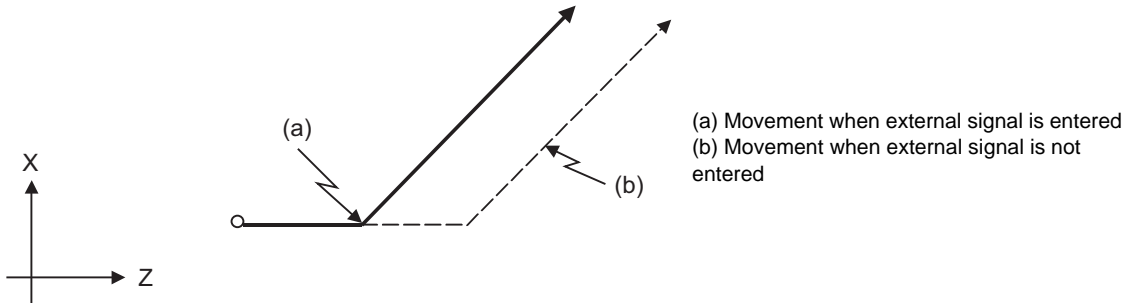




Operation example

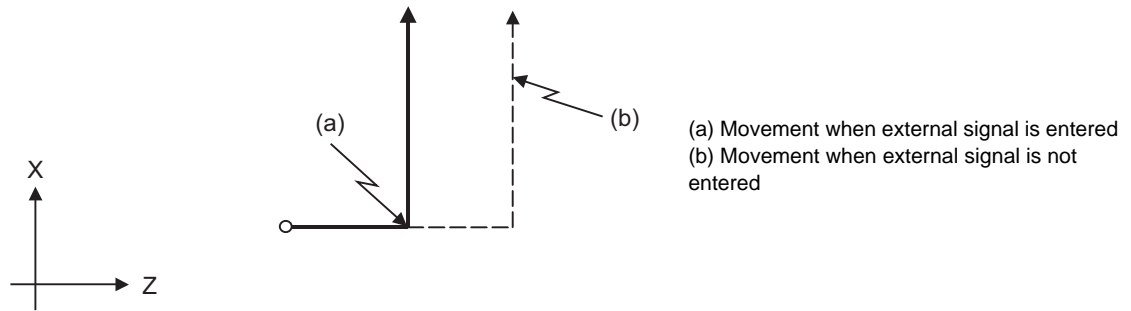
(Example 1) When the next block is an incremental value command

```
G31 Z1000 F100;
G01 U2000 W1000;
```



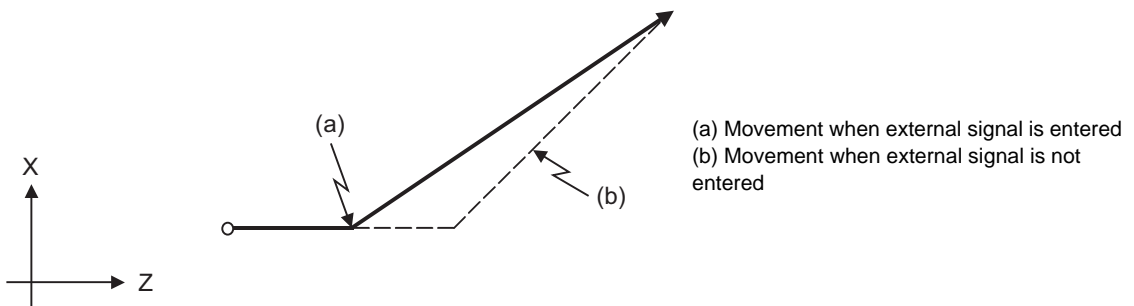
(Example 2) When the next block is a one axis movement command with absolute value

```
G31 Z1000 F100;
G01 X1000;
```



(Example 3) When the next block is a two axes movement command with absolute value

```
G31 Z1000 F100;
G01 X1000 Z2000;
```



## 16.3 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__ Z__  $\alpha$ __ F__ ;**

X Z $\alpha$	Target coordinates
F	Feedrate (mm/min)

Same with G31.2 and G31.3; Ff is not required with G04.

As with the G31 command, this command executes linear interpolation and when the preset skip signal conditions have been met, the machine is stopped, the remaining commands are canceled, and the next block is executed.



### Detailed description

- (1) Feedrate G31.1 set with the parameter corresponds to "#1176 skip1f", G31.2 corresponds to "#1178 skip2f", and G31.3 corresponds to "#1180 skip3f".
- (2) A command is skipped if it meets the specified skip signal condition.
- (3) The 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.

The high-speed skip signal is specified by a skip condition parameter, and the PLC skip signal is specified by a PLC skip condition parameter. The parameter to be used is determined by the G code being used.

G code	Skip condition parameter	PLC skip condition parameter
G04	#1173 dwlskp	#21050 plcdwlskp
G31.1	#1175 skip1	#21051 plcskip1
G31.2	#1177 skip2	#21052 plcskip2
G31.3	#1179 skip3	#21053 plcskip3

Skip condition parameter setting (dicimal)	Valid high-speed skip signal			
	4	3	2	1
1	x	x	x	○
2	x	x	○	x
3	x	x	○	○
4	x	○	x	x
5	x	○	x	○
6	x	○	○	x
7	x	○	○	○
8	○	x	x	x
9	○	x	x	○
10	○	x	○	x
11	○	x	○	○
12	○	○	x	x
13	○	○	x	○
14	○	○	○	x
15	○	○	○	○

(Skip when " ○ " signal is input.)

PLC skip condition parameter setting (hexadecimal)	Valid PLC skip signal									
	32	31	30	29	...	4	3	2	1	
00000001					...				○	
00000002					...			○		
00000003					...			○	○	
00000004					...		○			
00000005					...		○		○	
00000006					...		○	○		
00000007					...		○	○	○	
00000008					...	○				
00000009					...	○			○	
0000000A					...	○		○		
0000000B					...	○		○	○	
0000000C					...	○	○			
0000000D					...	○	○		○	
0000000E					...	○	○	○		
0000000F					...	○	○	○	○	
:	:	:	:	:	:	:	:	:	:	
:	:	:	:	:	:	:	:	:	:	
80000000	○									
:	:	:	:	:	:	:	:	:	:	
:	:	:	:	:	:	:	:	:	:	
FFFFFFFE	○	○	○	○	...	○	○	○		
FFFFFFF	○	○	○	○	...	○	○	○	○	

(Skip when " ○ " signal is input.)

- (4) The PLC skip condition parameters (#21050 plcdwlskp, #21051 plcskip1 to #21053 plcskip3) can be input with character strings using programmable parameter input function.
- (5) Other commands work the same as the G31 (skip function) command.



### Operation example

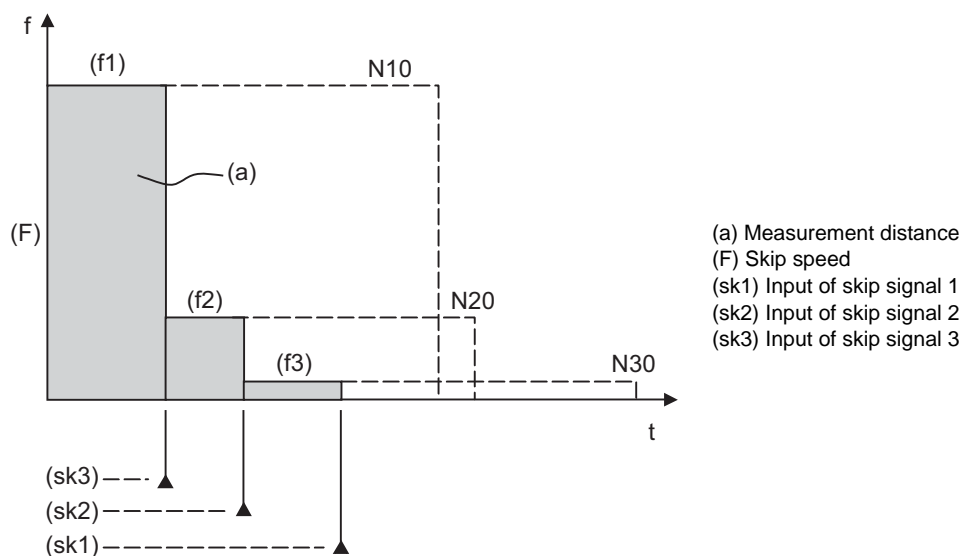
- (1) The multi-step skip function enables the following control, thereby improving measurement accuracy and shortening the time required for measurement.

[Parameter settings]

Skip condition	Skip speed
G31.1 : 7	20.0mm/min (f1)
G31.2 : 3	5.0mm/min (f2)
G31.3 : 1	1.0mm/min (f3)

[Program example]

```
N10 G31.1 X200.0 ;
N20 G31.2 X40.0 ;
N30 G31.3 X1.0 ;
```



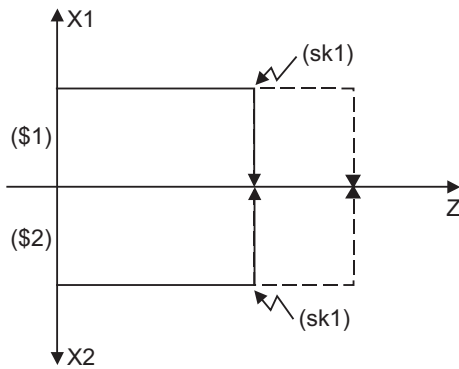
- (Note 1) If skip signal 1 is input before skip signal 2 in the above operation, N20 is skipped at that point and N30 is also ignored.
- (2) If a skip signal with the condition set during G04 (dwell) is input, the remaining dwell time is canceled and the following block is executed.

## 16.4 Multi-step Skip Function 2 ; G31 P/L

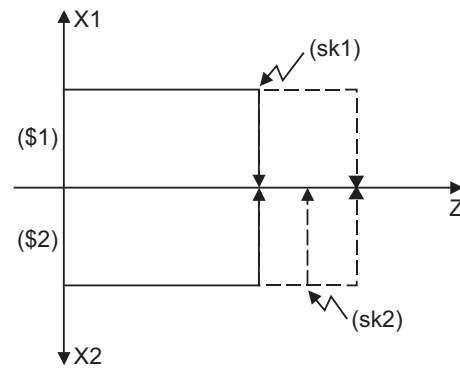


### Function and purpose

During linear interpolation by the skip command (G31), operation can be skipped according to the conditions of the skip signal parameter Pp or Ll. (P is for high-speed skip signal designation and L for PLC skip signal.) If multi-step skip commands are issued simultaneously in different part systems as shown in the left figure, both part systems perform skip operation simultaneously if the input skip signals are the same, or they perform skip operation separately if the input skip signals are different as shown in the right figure. The skip operation is the same as a normal skip command (G31 without P/L command).



[Same skip signals input in both 1st and 2nd part systems]



[Different skip signals input in 1st and 2nd part systems]

(\$1) 1st part system

(\$2) 2nd part system

(sk1) Skip signal 1

(sk2) Skip signal 2

If the skip condition specified by the parameter "#1173 dwlskp" (indicating external skip signals 1 to 4) is met during execution of a dwell command (G04), the remaining dwell time is canceled and the following block is executed.



### Command format

**G31 X__ Z__ α__ P/L__ F__ ;**

X Z α	Target coordinates
P/L	Skip signal command
F	Feedrate (mm/min)



### Detailed description

- (1) The skip speed is specified by program command or parameter (#1174 skip_F). Note that the F modal is not updated.
- (2) The high-speed skip signal is specified by skip signal command P, and the PLC skip signal is specified by skip signal command L. The command range of "P" is from 1 to 15. The command range of "L" is from 1 to 32. If a value outside of the range is commanded, program error (P35) will occur.

<Signal command P>

The bits 0 to 3 of the P-command-value correspond to high-speed skip signals 1 to 4, respectively.

High-speed skip signal command P	Valid high-speed skip signal			
	4	3	2	1
1	x	x	x	○
2	x	x	○	x
3	x	x	○	○
4	x	○	x	x
5	x	○	x	○
6	x	○	○	x
7	x	○	○	○
8	○	x	x	x
9	○	x	x	○
10	○	x	○	x
11	○	x	○	○
12	○	○	x	x
13	○	○	x	○
14	○	○	○	x
15	○	○	○	○

(Skip when "○" signal is input.)

<Signal command L>

The value to be set to L command corresponds to the PLC skip signal No. to be designated.

PLC skip signal command L	Valid PLC skip signal									
	32	31	30	29	...	4	3	2	1	
1	x	x	x	x	...	x	x	x	○	
2	x	x	x	x	...	x	x	○	x	
3	x	x	x	x	...	x	○	x	x	
4	x	x	x	x	...	○	x	x	x	
:					:	:	:	:	:	
:					:	:	:	:	:	
29	x	x	x	○	...	x	x	x	x	
30	x	x	○	x	...	x	x	x	x	
31	x	○	x	x	...	x	x	x	x	
32	○	x	x	x	...	x	x	x	x	

(Skip when "○" signal is input.)

- (3) The specified skip signal command is a logical sum of the skip signals.  
 (Example) G31 X100. P5 F100 ;  
 Operation is skipped if skip signal 1 or 3 is input.
- (4) If skip signal parameter P or L is not specified, it works as a skip function, not a multi-step skip function, is commanded with the high-speed skip signal 1 as the valid skip signal. If speed parameter Ff is not specified, the skip speed set by the parameter "#1174 skip_F" will apply.

[Relations between skip and multi-step skip]

Skip specifications	×		○	
	Condition	Speed	Condition	Speed
G31 X100 ; (Without P and F)	Program error (P601)		Skip 1	#1174 skip_F
G31 X100 P5 ; (Without F)	Program error (P602)		Command value	#1174 skip_F
G31 X100 F100 ; (Without P)	Program error (P601)		Skip 1	Command value
G31 X100 P5 F100 ;	Program error (P602)		Command value	Command value

- (5) If skip specification is effective and P is specified as an axis address, skip signal parameter P will be given a priority. The axis address "P" will be ignored.  
 (Example) G31 X100. P500 F100 ;  
 This is regarded as a skip signal. (The program error (P35) will occur.)
- (6) It is allowed to issue P and L commands together in one block.
- (7) Other than above, the same detailed description as "Skip function; G31" applies.

## 16.5 Programmable Current Limitation ; G10 L14 ;



### Function and purpose

This function allows the current limit value of the NC axis to be changed to a desired value in the program, and is used for the workpiece stopper, etc.

The commanded current limit value is designated with a ratio of the limit current to the rated current.



### Command format

**G10 L14 Xn ;**

L14	Current limit value setting (+ side/- side)
X	Axis address
n	Current limit value (%) Setting range: 1 to 300



### Precautions

- (1) If the current limit value is reached when the current limit is valid, the current limit reached signal is output.
- (2) The following two modes can be used with external signals as the operation after the current limit is reached. The external signal determines which mode applies.
 

[Normal mode]  
The movement command is executed in the current state.  
During automatic operation, the movement command is executed until the end, and then move to the next block with the droops still accumulated.

[Interlock mode]  
During the occurrence of the droops, it enters to the internal interlock state and the next movement will not be carried out.  
During automatic operation, the operation stops at the corresponding block, and the next block is not moved to.  
During manual operation, the following same direction commands are ignored.
- (3) The position droop generated by the current limit can be canceled when the current limit changeover signal of external signals is canceled. (Note that the axis must not be moving.)
- (4) The setting range of the current limit value is 1% to 300%. Commands that exceed this range will cause a program error (P35).
- (5) If a decimal point is designated with the G10 command, only the integer will be valid.  
Example) G10 L14 X10.123 ; The current limit value will be set to 10%.
- (6) For the axis name "C", the current limit value cannot be set from the program (G10 command).  
To set from the program, set the axis address with an incremental axis name, or set the axis name to one other than "C".





# Appendix 1

---

## Order of G Function Command Priority

Appendix 1 Order of G Function Command Priority

(Command in a separate block when possible)

○ indicates that both commands are executed simultaneously

Commanded G code	G Group		
	04 G22, G23	07 G40 to G42 G46	09 G70 to G79 G84 to G89
G04 Dwell	G04 is executed. G22 and G23 are ignored. (Note)	G04 is executed. G40 to G42 are ignored. (Note)	G04 is executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)
G10, G11 Programmable data setting	G10 and G11 are executed. G22 and G23 are ignored. (Note)	G10 and G11 are executed. G40 to G42 are ignored. (Note)	G10 and G11 are executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)
G27 to G30 Reference position compare/ return	G27 to G30 are executed. G22 and G23 are ignored. (Note)	G27 to G30 are executed. G40 to G42 are ignored. (Note)	G27 to G30 are executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)
G37 Automatic tool length measurement	G37 is executed. G22 and G23 are ignored. (Note)	G37 is executed. G40 to G42 are ignored. (Note)	G37 is executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)
G52 Local coordinate system	G52 is executed. G22 and G23 are ignored. (Note)	G52 is executed. G40 to G42 are ignored. (Note)	G52 is executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)
G53 Machine coordinate system	G53 is executed. G22 and G23 are ignored. (Note)	G53 is executed. G40 to G42 are ignored. (Note)	G53 is executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)
G65 Macro call	○	○	G65 is executed. G70 to G79 are ignored. G84 to G89 are ignored. (Note)

(Note) A program error (P45) will occur if they are commanded in the same block.

This error can be avoided by the parameter "#1241 set13/bit0 No G-CODE COMB. Error", but, be aware that one of the G commands is ignored.

# Appendix 2

---

## Program Errors

(Note) Program error messages are displayed in abbreviation on the screen.

**P10 EXCS. AXIS. No.****Details**

The number of axis addresses commanded in a block exceeds the specifications.

**Remedy**

- Divide the alarm block command into two.
- Check the specifications.

**P11 AXIS ADR. ERROR****Details**

The axis address commanded by the program does not match any of the ones set by the parameter.

**Remedy**

- Correct the axis names in the program.

**P20 DIVISION ERROR****Details**

The issued axis command cannot be divided by the command unit.

**Remedy**

- Correct the program.

**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****Details**

The number of holes per character on the paper tape is even for EIA code and odd for ISO code.

**Remedy**

- Check the paper tape.
- Check the tape puncher and tape reader.

**P31 PARITY V****Details**

The number of characters per block on the paper tape is odd.

**Remedy**

- Make the number of characters per block on the paper tape even.
- Set the parameter parity V selection OFF.

**P32 ADDRESS. ERROR****Details**

An address not listed in the specifications has been used.

**Remedy**

- Correct the program address.
- Correct the parameter settings.
- Check the specifications.

**P33 FORMAT ERROR****Details**

The command format in the program is not correct.

**Remedy**

- Correct the program.

**P34 G-CODE ERROR****Details**

The commanded G code is not in the specifications.  
An illegal G code was commanded during the coordinate rotation command.

**Remedy**

- Correct the G code address in the program.

**Details**

G51.2 or G50.2 was commanded when "#1501 polyax (Rotational tool axis number)" was set to "0".  
G51.2 or G50.2 was commanded when the tool axis was set to the linear axis ("#1017 rot (Rotational axis)" is set to "0").

**Remedy**

- Correct the parameter settings.

**P35 CMD-VALUE OVER****Details**

The setting range for the addresses has been exceeded.  
The program coordinates overflowed because commands to the linear type rotary axis accumulated in one direction.

**Remedy**

- Correct the program.

**P36 PROGRAM END ERR****Details**

"EOR" has been read during memory mode.

**Remedy**

- Enter the M02 and M30 command at the end of the program.
- Enter the M99 command at the end of the subprogram.

**P37 PROG. No. ZERO****Details**

"0" has been specified for program or sequence No.

**Remedy**

- Designate program Nos. within a range from 1 to 99999999.
- Designate sequence Nos. within a range from 1 to 99999.
- Add M02 or M03 to the end of the program running in FTP operation.

**P39 NO SPEC ERR****Details**

- A non-specified G code was commanded.
- The selected operation mode is out of specifications.

**Remedy**

- Check the specifications.

**P45 G-CODE COMB.****Details**

The combination of G codes in a block is inappropriate.  
A part of unmodal G codes and modal G codes cannot be commanded in a same block.

**Remedy**

Correct the combination of G codes.  
Separate the incompatible G codes into different blocks.

**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.

**P60 OVER CMP. LENG.**

**Details**

The commanded movement distance is excessive (over  $2^{31}$ ).

**Remedy**

- Correct the command range for the axis address.

**P62 F-CMD. NOTHING**

**Details**

- No feed rate command has been issued.
- There is no F command in the cylindrical interpolation or polar coordinate interpolation immediately after the G95 mode is commanded.

**Remedy**

- The default movement modal command at power ON is G01. This causes the machine to move without a G01 command if a movement command is issued in the program, and an alarm results. Use an F command to specify the feed rate.
- Specify F with a thread lead command.

**P65 No G05P3 SPEC**

**Details**

**Remedy**

- Check whether the specifications are provided for the high-speed mode III.

**P70 ARC ERROR**

**Details**

- There is an error in the arc start and end points as well as in the arc center.
- The difference of the involute curve through the start point and the end point is large.
- When arc was commanded, one of the two axes configuring the arc plane was a scaling valid axis.

**Remedy**

- Correct the numerical values of the addresses that specify the start and end points, arc center as well as the radius in the program.
- Correct the "+" and "-" directions of the address numerical values.
- Check for the scaling valid axis.

**P71 ARC CENTER**

**Details**

- An arc center cannot be obtained in R-specified circular interpolation.
- A curvature center of the involute curve cannot be obtained.

**Remedy**

- Correct the numerical values of the addresses in the program.
- Correct the start and end points if they are inside of the base circle for involute interpolation. When carrying out tool radius compensation, make sure that the start and end points after compensation will not be inside of the base circle for involute interpolation.
- Correct the start and end points if they are at an even distance from the center of the base circle for involute interpolation.

**P72 NO HELICAL SPEC****Details**

A helical command has been issued though it is out of specifications.

**Remedy**

- Check whether the specifications are provided for the helical cutting.
- An Axis 3 command has been issued by the circular interpolation command. If there is no helical specification, move the linear axis to the next block.

**P90 NO THREAD SPEC****Details**

A thread cutting command was issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P93 SCREW PITCH ERR****Details**

An illegal thread lead (thread pitch) was specified at the thread cutting command.

**Remedy**

- Correct the thread lead for the thread cutting command.

**P111 PLANE CHG (CR)****Details**

Plane selection commands (G17, G18, G19) were issued during a coordinate rotation (G68) was being commanded.

**Remedy**

- Always command G69 (coordinate rotation cancel) after the G68 command, and then issue a plane selection command.

**P112 PLANE CHG (CC)****Details**

- Plane selection commands (G17, G18, G19) were issued while tool radius compensation (G41, G42) and nose R compensation (G41, G42, G46) commands were being issued.
- Plane selection commands were issued after completing nose R compensation commands when there were no further axis movement commands after G40, and compensation has not been cancelled.

**Remedy**

- Issue plane selection commands after completing (axis movement commands issued after G40 cancel command) tool radius compensation and nose R compensation commands.

**P113 ILLEGAL PLANE****Details**

The circular command axis does not correspond to the selected plane.

**Remedy**

- Select a correct plane before issuing a circular command.

**P122 NO AUTO C-OVR****Details**

An auto corner override command (G62) was issued though it is out of specifications.

**Remedy**

- Check the specifications.
- Delete the G62 command from the program.

**P130 2nd AUX. ADDR****Details**

The 2nd miscellaneous function address, commanded in the program, differs from the address set in the parameters.

**Remedy**

- Correct the 2nd miscellaneous function address in the program.



**P131 NO G96 SPEC**

**Details**

A constant surface speed control command (G96) was issued though it is out of specifications.

**Remedy**

- Check the specifications.
- Issue a rotation speed command (G97) instead of the constant surface speed control command (G96).

**P132 SPINDLE S = 0**

**Details**

No spindle rotation speed command has been issued.

**Remedy**

- Correct the program.

**P133 G96 P-No. ERR**

**Details**

The illegal No. was specified for the constant surface speed control axis.

**Remedy**

- Correct the parameter settings and program that specify the constant surface speed control axis.

**P134 G96 Clamp Err.**

**Details**

The constant surface speed control command (G96) was issued without commanding the spindle speed clamp (G92/G50).

**Remedy**

Press the reset key and carry out the remedy below.

- Check the program.
- Issue the G92/G50 command before the G96 command.
- Command the constant surface speed cancel (G97) to switch to the rotation speed command.

**P150 NO C-CMP SPEC**

**Details**

- Tool radius compensation commands (G41 and G42) were issued though they are out of specifications.
- Nose R compensation commands (G41, G42, and G46) were issued though they are out of specifications.

**Remedy**

- Check the specifications.

**P151 G2, 3 CMP. ERR**

**Details**

A compensation command (G40, G41, G42, G43, G44, or G46) has been issued in the arc modal (G02 or G03).

**Remedy**

- Issue the linear command (G01) or rapid traverse command (G00) in the compensation command block or cancel block.  
(Set the modal to linear interpolation.)

**P152 I.S.P NOTHING**

**Details**

In interference block processing during execution of a tool radius compensation (G41 or G42) or nose R compensation (G41, G42, or G46) command, the intersection point after one block is skipped cannot be determined.

**Remedy**

- Correct the program.

**P153 I.F ERROR****Details**

An interference error has occurred while the tool radius compensation command (G41 or G42) or nose R compensation command (G41, G42 or G46) was being executed.

**Remedy**

- Correct the program.

**P155 F-CYC ERR (CC)****Details**

A fixed cycle command has been issued in the radius compensation mode.

**Remedy**

- Issue a radius compensation cancel command (G40) to cancel the radius compensation mode that has been applied since the fixed cycle command was issued.

**P156 BOUND DIRECT****Details**

A shift vector with undefined compensation direction was found at the start of G46 nose R compensation.

**Remedy**

- Change the vector to that which has the defined compensation direction.
- Change the tool to that which has a different tip point No.

**P157 SIDE REVERSED****Details**

During G46 nose R compensation, the compensation direction is reversed.

**Remedy**

- Change the G command to that which allows the reversed compensation direction (G00, G28, G30, G33, or G53).
- Change the tool to that which has a different tip point No.
- Enable "#8106 G46 NO REV-ERR".

**P158 ILLEGAL TIP P.****Details**

An illegal tip point No. (other than 1 to 8) was found during G46 nose R compensation.

**Remedy**

- Correct the tip point No.

**P170 NO CORR. NO.****Details**

No compensation No. (DOO, TOO or HOO) command was given when the radius compensation (G41, G42, G43 or G46) command was issued. Otherwise, the compensation No. is larger than the number of sets in the specifications.

**Remedy**

- Add the compensation No. command to the compensation command block.
- Check the number of sets for the tool compensation Nos. and correct the compensation No. command to be within the number of sets.

**P171 NO G10 SPEC****Details**

Compensation data input by program (G10) was commanded though it is out of specifications.

**Remedy**

- Check the specifications.

**P172 G10 L-No. ERR****Details**

An address of G10 command is not correct.

**Remedy**

- Correct the address L No. of the G10 command.

**P173 G10 P-No. ERR**

**Details**

The compensation No. at the G10 command is not within the permitted number of sets in the specifications.

**Remedy**

- Check the number of sets for the tool compensation Nos. and correct the address P designation to be within the number of sets.

**P174 NO G11 SPEC**

**Details**

Compensation data input by program cancel (G11) was commanded though there is no specification of compensation data input by program.

**Remedy**

- Check the specifications.

**P177 LIFE COUNT ACT**

**Details**

Registration of tool life management data with G10 was attempted when the "usage data count valid" signal was ON.

**Remedy**

- The tool life management data cannot be registered during the usage data count. Turn the "usage data count valid" signal OFF.

**P178 LIFE DATA OVER**

**Details**

The number of registration groups, total number of registered tools or the number of registrations per group exceeded the range in the specifications.

**Remedy**

- Correct the number of registrations.

**P179 GROUP NO. ILL.**

**Details**

- A duplicate group No. was found at the registration of the tool life management data with G10.
- A group No. that was not registered was designated during the T****99 command.
- An M code command, which must be issued as a single command, coexists in the same block as that of another M code command.
- The M code commands set in the same group exist in the same block.

**Remedy**

- Register the tool life data once for one group: commanding with a duplicate group No. is not allowed.
- Correct to the group No.

**P180 NO BORING CYC.**

**Details**

A fixed cycle command (G72 - G89) was issued though it is out of specifications.

**Remedy**

- Check the specifications.
- Correct the program.

**P181 NO S-CMD (TAP)**

**Details**

Spindle rotation speed (S) has not been commanded in synchronous tapping.

**Remedy**

- Command the spindle rotation speed (S) in synchronous tapping.
- When "#8125 Check Scode in G84" is set to "1", enter the S command in the same block where the synchronous tapping command is issued.

**P182 SYN TAP ERROR****Details**

- Connection to the main spindle unit was not established.
- The synchronous tapping was attempted with the spindle not serially connected under the multiple-spindle control I.

**Remedy**

- Check connection to the main spindle.
- Check that the main spindle encoder exists.
- Set 1 to the parameter #3024 (sout).

**P183 PTC/THD No.****Details**

The pitch or number of threads has not been commanded in the tap cycle of a fixed cycle for drilling command.

**Remedy**

- Specify the pitch data and the number of threads by F or E command.

**P184 NO PTC/THD CMD****Details**

- The pitch or the number of threads per inch is illegal in the tap cycle of the fixed cycle for drilling command.
- The pitch is too small for the spindle rotation speed.
- The thread number is too large for the spindle rotation speed.

**Remedy**

- Correct the pitch or the number of threads per inch.

**P187 Tap SP clamp 0****Details**

The external spindle speed clamp signal was turned ON without setting the tapping spindle's external spindle speed when commanding the synchronous tapping.

**Remedy**

- Set the external spindle speed clamp speed parameter.
- Turn the external spindle speed clamp signal OFF.

**P190 NO CUTTING CYC****Details**

A lathe cutting cycle command was issued though it is out of specifications.

**Remedy**

- Check the specification.
- Delete the lathe cutting cycle command.

**P191 TAPER LENG ERR****Details**

In the lathe cutting cycle, the specified length of taper section is illegal.

**Remedy**

- Set the smaller radius value than the axis travel amount in the lathe cycle command.

**P192 CHAMFERING ERR****Details**

Chamfering in the thread cutting cycle is illegal.

**Remedy**

- Set a chamfering amount not exceeding the cycle.

**P200 NO MRC CYC SPC**

**Details**

The compound type fixed cycle for turning machining I (G70 to G73) was commanded though it is out of specifications.

**Remedy**

- Check the specifications.

**P201 PROG. ERR (MRC)**

**Details**

- The subprogram, called with a compound type fixed cycle for turning machining I command, has at least one of the following commands: reference position return command (G27, G28, G29, G30); thread cutting (G33, G34); fixed cycle skip-function (G31, G31.n).
- An arc command was found in the first movement block of the finished shape program in compound type fixed cycle for turning machining I.

**Remedy**

- Delete G27, G28, G29, G30, G31, G33, G34, and fixed cycle G codes from the subprogram called with the compound type fixed cycle for turning machining I commands (G70 to G73).
- Delete G02 and G03 from the first movement block of the finished shape program in compound type fixed cycle for turning machining I.

**P202 BLOCK OVR (MRC)**

**Details**

The number of blocks in the shape program of the compound type fixed cycle for turning machining I is over 50 or 200 (the maximum number differs according to the model).

**Remedy**

- Set a 50/200 or less value for the number of blocks in the shape program called by the compound type fixed cycle for turning machining I commands (G70 to G73). (The maximum number differs according to the model).

**P203 CONF. ERR (MRC)**

**Details**

A proper shape will not obtained by executing the shape program for the compound type fixed cycle for turning machining I (G70 to G73).

**Remedy**

- Correct the shape program for the compound type fixed cycle for turning machining I (G70 to G73).

**P204 VALUE ERR (MRC)**

**Details**

A command value of the compound type fixed cycle for turning machining (G70 to G76) is illegal.

**Remedy**

- Correct the command value of the compound type fixed cycle for turning machining (G70 to G76).

**P210 NO PAT CYC SPC**

**Details**

A compound type fixed cycle for turning machining II (G74 to G76) command was commanded though it is out of specifications.

**Remedy**

- Check the specifications.

**P220 NO SPECIAL CYC**

**Details**

There are no special fixed cycle specifications.

**Remedy**

- Check the specifications.

**P221 NO HOLE (S-CYC)****Details**

"0" has been specified for the number of holes in special fixed cycle mode.

**Remedy**

- Correct the program.

**P222 G36 ANGLE ERR****Details**

A G36 command specifies "0" for angle intervals.

**Remedy**

- Correct the program.

**P223 G12 G13 R ERR****Details**

The radius value specified with a G12 or G13 command is below the compensation amount.

**Remedy**

- Correct the program.

**P224 NO G12, G13 SPC****Details**

There are no circular cutting specifications.

**Remedy**

- Check the specifications.

**P230 NESTING OVER****Details**

Over 8 times of subprogram calls have been done in succession from a subprogram.

- A M198 command was found in the program in the data server.
- The program in the IC card has been called more than once (the program in the IC card can be called only once during nested).

**Remedy**

- Correct the program so that the number of subprogram calls does not exceed 8 times.

**P231 NO N-NUMBER****Details**

The sequence No., commanded at the return from the subprogram or by GOTO in the subprogram call, was not set.

**Remedy**

- Specify the sequence Nos. in the call block of the subprogram.

**P232 NO PROGRAM No.****Details**

- The machining program has not been found when the machining program is called.
- The file name of the program registered in IC card is not corresponding to O No.

**Remedy**

- Enter the machining program.
- Check the subprogram storage destination parameters.
- Ensure that the external device (including IC card) that contains the file is mounted.

**P241 NO VARI NUMBER****Details**

The variable No. commanded is out of the range specified in the specifications.

**Remedy**

- Check the specifications.
- Correct the program variable No.

**P242 EQL. SYM. MSSG.****Details**

The "=" sign has not been commanded when a variable is defined.

**Remedy**

- Designate the "=" sign in the variable definition of the program.

**P243 VARIABLE ERR.****Details**

An invalid variable has been specified in the left or right side of an operation expression.

**Remedy**

- Correct the program.

**P260 NO COOD-RT SPC****Details**

A coordinate rotation command was issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P261 G-CODE COMB****Details**

Another G code or a T command has been issued in the block of coordinate rotation command.

**Remedy**

- Correct the program.

**P262 Modal Err****Details**

A coordinate rotation command has been issued during modal in which coordinate rotation is not allowed.

**Remedy**

- Correct the program.

**P270 NO MACRO SPEC****Details**

A macro specification was commanded though it is out of specifications.

**Remedy**

- Check the specifications.

**P271 NO MACRO INT.****Details**

A macro interruption command has been issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P272 MACRO ILL.****Details**

An executable statement and a macro statement exist together in the same block.

**Remedy**

- Place the executable statement and macro statement in separate blocks in the program.

**P273 MACRO OVERCALL****Details**

The number of macro call nests exceeded the limit imposed by the specifications.

**Remedy**

- Correct the program so that the macro calls do not exceed the limit imposed by the specifications.

**P275 MACRO ARG. EX.****Details**

The number of argument sets in the macro call argument type II has exceeded the limit.

**Remedy**

- Correct the program.

**P276 CALL CANCEL****Details**

A G67 command was issued though it was not during the G66 command modal.

**Remedy**

- Correct the program.
- Issue G66 command before G67 command, which is a call cancel command.

**P277 MACRO ALM MMSG****Details**

An alarm command has been issued in #3000.

**Remedy**

- Refer to the operator messages on the diagnosis screen.
- Refer to the instruction manual issued by the machine tool builder.

**P280 EXC. [ , ]****Details**

Over five times have the parentheses "[" or "]" been used in a single block.

**Remedy**

- Correct the program so that the number of "[" or "]" is five or less.

**P281 [ , ] ILLEGAL****Details**

A single block does not have the same number of commanded parentheses "[" as that of "]".

**Remedy**

- Correct the program so that "[" and "]" parentheses are paired up properly.

**P282 CALC. IMPOSS.****Details**

The arithmetic formula is incorrect.

**Remedy**

- Correct the formula in the program.

**P283 DIVIDE BY ZERO****Details**

The denominator of the division is zero.

**Remedy**

- Correct the program so that the denominator for division in the formula is not zero.

**P290 IF SNT. ERROR****Details**

There is an error in the "IF[<conditional>]GOTO(" statement.

**Remedy**

- Correct the program.

**P291 WHILE SNT. ERR****Details**

There is an error in the "WHILE[<conditional>]DO(-END(" statement.

**Remedy**

- Correct the program.



**P292 SETVN SNT. ERR****Details**

There is an error in the "SETVN(" statement when the variable name setting was made.

**Remedy**

- Correct the program.
- The number of characters in the variable name of the SETVN statement must be 7 or less.

**P293 DO-END EXCESS****Details**

The number of DO-END nesting levels in the "WHILE[<conditional>]DO(-END(" statement has exceeded 27.

**Remedy**

- Correct the program so that the nesting levels of the DO-END statement does not exceed 27.

**P294 DO-END MMC.****Details**

The DOs and ENDS are not paired off properly.

**Remedy**

- Correct the program so that the DOs and ENDS are paired off properly.

**P295 WHILE/GOTO TPE****Details**

There is a WHILE or GOTO statement on the tape during FTP operation.

**Remedy**

- Apply memory mode operation instead of FTP operation that does not allow the execution of the program with a WHILE or GOTO statement.

**P296 NO ADR (MACRO)****Details**

A required address has not been specified in the user macro.

**Remedy**

- Correct the program.

**P297 ADR-A ERR.****Details**

The user macro does not use address A as a variable.

**Remedy**

- Correct the program.

**P298 PTR OP (MACRO)****Details**

User macro G200, G201, or G202 was specified during tape or MDI mode.

**Remedy**

- Correct the program.

**P300 VAR. NAME ERROR****Details**

The variable names have not been commanded properly.

**Remedy**

- Correct the variable names in the program.

**P301 VAR. NAME DUPLI****Details**

A duplicate variable name was found.

**Remedy**

- Correct the program so that no duplicate name exists.

**P360 NO PROG.MIRR.****Details**

A mirror image (G50.1 or G51.1) command has been issued though the programmable mirror image specifications are not provided.

**Remedy**

- Check the specifications.

**P380 NO CORNER R/C****Details**

The corner R/C was issued though it is out of specifications.

**Remedy**

- Check the specifications.
- Delete the corner chamfering/corner rounding command in the program.

**P381 NO ARC R/C SPC****Details**

Corner chamfering II or corner rounding II was commanded in the arc interpolation block though it is out of specifications.

**Remedy**

- Check the specifications.

**P382 CORNER NO MOVE****Details**

The block next to corner chamfering/ corner rounding is not a travel command.

**Remedy**

- Replace the block succeeding the corner chamfering/ corner rounding command by G01 command.

**P383 CORNER SHORT****Details**

The travel distance in the corner chamfering/corner rounding command was shorter than the value in the corner chamfering/corner rounding command.

**Remedy**

- Set the smaller value for the corner chamfering/corner rounding than the travel distance.

**P384 CORNER SHORT****Details**

The travel distance in the following block in the corner chamfering/corner rounding command was shorter than the value in the corner chamfering/corner rounding command.

**Remedy**

- Set the smaller value for the corner chamfering/corner rounding than the travel distance in the following block.

**P385 G0 G33 IN CONR****Details**

A block with corner chamfering/corner rounding was given during G00 or G33 modal.

**Remedy**

- Correct the program.

**P390 NO GEOMETRIC****Details**

A geometric command was issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P391 NO GEOMETRIC 2****Details**

There are no geometric IB specifications.

**Remedy**

- Check the specifications.

**P392 LES AGL (GEOMT)****Details**

The angular difference between the geometric line and line is 1° or less.

**Remedy**

- Correct the geometric angle.

**P393 INC ERR (GEOMT)****Details**

The second geometric block has a command with an incremental value.

**Remedy**

- Issue a command with an absolute value in the second geometric block.

**P394 NO G01 (GEOMT)****Details**

The second geometric block contains no linear command.

**Remedy**

- Issue the G01 command.

**P395 NO ADRS (GEOMT)****Details**

The geometric format is invalid.

**Remedy**

- Correct the program.

**P396 PL CHG. (GEOMT)****Details**

A plane switching command was issued during geometric command processing.

**Remedy**

- Complete the plane switching command before geometric command processing.

**P397 ARC ERR (GEOMT)****Details**

In geometric IB, the circular arc end point does not contact or cross the next block start point.

**Remedy**

- Correct the geometric circular arc command and the preceding and following commands.

**P398 NO GEOMETRIC1B****Details**

A geometric command was issued though the geometric IB specifications are not provided.

**Remedy**

- Check the specifications.

**P420 NO PARAM IN****Details**

Parameter input by program (G10) was commanded though it is out of specifications.

**Remedy**

- Check the specifications.

**P421 PRAM. IN ERROR****Details**

- The specified parameter No. or set data is illegal.
- An illegal G command address was input in parameter input mode.
- A parameter input command was issued during fixed cycle modal or nose R compensation.
- G10L50, G10L70, G11 were not commanded in independent blocks.

**Remedy**

- Correct the program.

**P430 AXIS NOT RET.****Details**

- A command was issued to move an axis, which has not returned to the reference position, away from that reference position.
- A command was issued to an axis removal axis.

**Remedy**

- Execute reference position return manually.
- Disable the axis removal on the axis for which the command was issued.

**P431 NO 2ndREF. SPC****Details**

A command for second, third or fourth reference position return was issued though there are no such command specifications.

**Remedy**

- Check the specifications.

**P434 COLLATION ERR****Details**

One of the axes did not return to the reference position when the reference position check command (G27) was executed.

**Remedy**

- Correct the program.

**P435 G27/M ERROR****Details**

An M command was issued simultaneously in the G27 command block.

**Remedy**

- Place the M code command, which cannot be issued in a G27 command block, in separate block from G27 command block.

**P436 G29/M ERROR****Details**

An M command was issued simultaneously in the G29 command block.

**Remedy**

- Place the M code command, which cannot be issued in a G29 command block, in separate block from G29 command block.

**P438 NOT USE (G52)****Details**

A local coordinate system command was issued during execution of the G54.1 command.

**Remedy**

- Correct the program.

**P450 NO CHUCK BARR.****Details**

The chuck barrier on command (G22) was specified although the chuck barrier is out of specifications.

**Remedy**

- Check the specifications.

**P460 TAPE I/O ERROR**

**Details**

An error has occurred in the tape reader. Otherwise an error has occurred in the printer during macro printing.

**Remedy**

- Check the power and cable of the connected devices.
- Correct the I/O device parameters.

**P461 FILE I/O ERROR**

**Details**

- A file of the machining program cannot be read.

**Remedy**

- In memory mode, the programs stored in memory may have been destroyed. Output all of the programs and tool data and then format the system.

**P600 NO AUTO TLM.**

**Details**

An automatic tool length measurement command (G37) was issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P601 NO SKIP SPEC.**

**Details**

A skip command (G31) was issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P602 NO MULTI SKIP**

**Details**

A multiple skip command (G31.1, G31.2 or G31.3) was issued though it is out of specifications.

**Remedy**

- Check the specifications.

**P603 SKIP SPEED 0**

**Details**

The skip speed is "0".

**Remedy**

- Specify the skip speed.

**P604 TLM ILL. AXIS command**

**Details**

No axis was specified in the automatic tool length measurement block. Otherwise, two or more axes were specified.

**Remedy**

- Specify only one axis.

**P605 T-CMD IN BLOCK**

**Details**

The T code is in the same block as the automatic tool length measurement block.

**Remedy**

- Specify the T code before the automatic tool length measurement block.

**P606 NO T-CMD BEFOR**

**Details**

The T code was not yet specified in automatic tool length measurement.

**Remedy**

- Specify the T code before the automatic tool length measurement block.

**P607 TLM ILL. 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 ERROR (CC)****Details**

A skip command was issued during radius compensation processing.

**Remedy**

- Issue a radius compensation cancel (G40) command or remove the skip command.

**P609 NO PLC SKIP****Details**

PLC skip has been commanded (L to G31) while PLC skip is out of specifications.

**Remedy**

- Check the specifications.

**P610 ILLEGAL PARA.****Details**

- G114.1 was commanded when the spindle synchronization with PLC I/F command was selected.
- Spindle synchronization was commanded to a spindle that is not connected serially.

**Remedy**

- Check the program.
- Check the argument of G114.1 command.
- Check the state of spindle connection.

**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.

**P990 PREPRO S/W ERR****Details**

Combining commands that required pre-reading (nose R offset, corner chamfering/corner rounding, geometric I, geometric IB, and compound type fixed cycle for turning machining) resulted in eight or more pre-read blocks.

**Remedy**

- Delete some or all of the combinations of commands that require pre-reading.

# Index

## Numbers

2nd, 3rd, and 4th Reference Position (Zero point) Return ; G30.....	425
2-part System Simultaneous Thread Cutting Cycle I ; G76.1.....	401
2-part System Simultaneous Thread Cutting Cycle II ; G76.2.....	403
2-part System Simultaneous Thread Cutting Cycle Parameter Setting Command ; G76.....	400
2-part System Simultaneous Thread Cutting Cycle .....	400

## A

Absolute/Incremental Value Commands ; G90,G91 .....	26
Automatic Acceleration/Deceleration .....	80
Automatic Coordinate System Setting .....	418
Automatic Corner Override ; G62.....	93
Automatic Tool Length Measurement ; G37 .....	444

## B

Balance Cut ; G15,G14.....	388
Basic Machine Coordinate System Selection ; G53.....	419
Basic Machine, Workpiece and Local Coordinate Systems.....	416

## C

Chopping ; G81.1 .....	407
Chuck Barrier/Tailstock Barrier ; G22,G23.....	440
Circular Interpolation ; G02,G03 .....	46
Common Variables .....	279
Compensation Data Input by Program ; G10 L2/L10/ L11, G11.....	198
Compound Thread Cutting Cycle ; G76.....	237
Compound Type Fixed Cycle for Turning Machining .....	218
Constant Lead Thread Cutting ; G33 .....	54
Constant Surface Speed Control ; G96,G97.....	111
Continuous Thread Cutting ; G33 .....	60
Control Commands .....	321
Coordinate Rotation by Program ; G68.1/G69.1 ...	379
Coordinate System for Rotary Axis .....	435
Coordinate System Setting ; G92.....	420
Coordinate Systems and Coordinate Zero Point Symbols .....	4
Coordinate Words and Control Axes .....	2
Coordinate Words and Control Axes .....	414
Corner Chamfering I ; G01 X_ Z_ ,C_.....	331
Corner Chamfering II ; G01/G02/G03 X_ Z_ ,C_..	337
Corner Chamfering/Corner Rounding I .....	331
Corner Chamfering/Corner Rounding II .....	336
Corner Rounding I ; G01 X_ Z_ ,R_.....	333
Corner Rounding II ; G01/G02/G03 X_ Z_ ,R_....	339
Counting the Tool Life .....	204
Cutting Feedrate .....	69
Cutting Mode ; G64.....	100

## D

Deceleration Check .....	89
Decimal Point Input .....	30
Deep Hole Drilling Cycle 2 ; G83.2.....	254
Detailed Description for Macro Call Instruction ...	276
Dwell (Time Designation) ; G04.....	102

## E

Exact Stop Check ; G09 .....	84
Exact Stop Check Mode ; G61 .....	88

## F

F1-digit Feed.....	70
Face Boring Cycle (Longitudinal boring cycle) ; G85 (G89).....	253
Face Cut-Off Cycle ; G74 .....	233
Face Cutting Cycle ; G79.....	215
Face Deep Hole Drilling Cycle 1 (Longitudinal deep hole drilling cycle 1) ; G83 (G87) .....	246
Face Rough Cutting Cycle ; G72.....	224
Face Tapping Cycle (Longitudinal tapping cycle) ; G84 (G88) .....	248
Feed Hold, Feedrate Override, G09 Valid/Invalid (#3004) .....	296
Feed Per Minute/Feed Per Revolution (Asynchronous Feed/Synchronous Feed) ; G94,G95.....	72
Feedrate Designation and Effects on Control Axes .....	74
Finishing Cycle ; G70.....	232
Fixed Cycle for Drilling .....	243
Fixed Cycle for Drilling Cancel; G80 .....	256
Fixed Cycles for Turning Machining .....	208
Formed Material Rough Cutting Cycle ; G73.....	228

## G

G code .....	16
G code Lists .....	16
G Code Macro Call .....	274
G Command Modals (#4001-#4021, #4201-#4221) .....	298
G41/G42 Commands and I, J, K Designation .....	184
General Precautions for Tool Nose Radius Compensation .....	191
Geometric .....	341
Geometric I ; G01 A_.....	341
Geometric IB .....	343
Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 R_ H_ .....	352
Geometric IB (Automatic calculation of linear - arc intersection) ; G01 A_ , G02/G03 P_ Q_ H_ .....	348
Geometric IB (Automatic calculation of two-arc contact) ; G02/G03 P_ Q_ /R_.....	344

## H

Helical Interpolation ; G17,G18,G19 and G02,G03	63
High-accuracy control ; G61.1 .....	369



I	Inch Thread Cutting ; G33 .....	58	Other Operations during Tool Nose Radius Compensation .....	176
	Inch/Metric Conversion ; G20,G21 .....	29		
	Initial Point and R Point Level Return ; G98,G99.....	258	P	
	Input Setting Unit .....	6	Plane Selection ; G17,G18,G19.....	52
	Integrating Time (#3001, #3002) .....	295	Position Information (#5001 - #5100 + n) .....	300
	Interference Check .....	192	Positioning (Rapid Traverse) ; G00 .....	36
	Interrupt during Corner Chamfering/Corner Rounding .....	335	Precautions .....	324
	Interrupt during Corner Chamfering/Corner Rounding .....	340	Precautions Before Starting Machining .....	21
	Interrupts during Tool Nose Radius Compensation .....	188	Precautions for Compound Type Fixed Cycle for Turning Machining; G70 to G76 .....	241
			Precautions for Using Spindle Synchronization Control .....	137
L			Precautions When Using a Fixed Cycle for Drilling .....	257
	Linear Interpolation ; G01 .....	43	Pre-read Buffers .....	24
	Local Coordinate System Setting ; G52.....	434	Program Format .....	8
	Local Variables (#1 to #33) .....	280	Program/sequence/block numbers; O, N .....	12
	Longitudinal Cut-off Cycle ; G75.....	235	Programmable Current Limitation ; G10 L14 ;.....	459
	Longitudinal Cutting Cycle ; G77 .....	209	Programmable Parameter Input ; G10 L70, G11 ..	356
	Longitudinal Rough Cutting Cycle ; G71.....	219		
M			R	
	Machine Zero Point and 2nd Reference Position (Zero point) .....	417	R Specification Circular Interpolation ; G02,G03.....	50
	Macro Call Instruction .....	269	Radius/Diameter Designation .....	28
	Macro Interface Inputs/Outputs (#1000 to #1035, #1100 to #1135, #1200 to #1295, #1300 to #1395) .....	283	Rapid Traverse Constant Inclination Acceleration/Deceleration .....	81
	Macro Interruption ; M96,M97.....	358	Rapid Traverse Rate .....	68
	Message Display and Stop (#3006) .....	296	Reference Position (Zero point) Return ; G28,G29 .....	421
	Mirror Image (#3007) .....	297	Reference Position Check ; G27 .....	428
	Mirror Image for Facing Tool Posts ; G68,G69.....	326		
	Miscellaneous Command Macro Call (for M, S, T, B Code Macro Call) .....	275	S	
	Miscellaneous Functions (M8-digits) .....	106	Secondary Miscellaneous Functions (A8-digits, B8-digits or C8-digits) .....	108
	Modal Call A (Movement Command Call) ; G66..	272	Simple Macro Calls ; G65.....	269
	Modal Call B (for each block) ; G66.1 .....	273	Skip Function ; G31 .....	448
	Modal, unmodal .....	16	Speed Clamp.....	83
	Multiple-spindle Control I (spindle control command) ; S ○ = .....	141	Spindle Clamp Speed Setting ; G92.....	113
	Multiple-spindle Control I (spindle selection command) ; G43.1,G44.1.....	142	Spindle Functions.....	110
	Multiple-spindle Control .....	140	Spindle Synchronization Control I ; G114.1 .....	121
	Multi-step Skip Function 1 ; G31.n , G04.....	453	Spindle Synchronization Control II.....	132
	Multi-step Skip Function 2 ; G31 P/L .....	456	Spindle Synchronization.....	120
			Spindle/C Axis Control .....	115
N			Start Point Designation Timing Synchronization (Type 1) ; G115.....	396
	NC Alarm (#3000) .....	294	Start Point Designation Timing Synchronization (Type 2) ; G116.....	398
	Number of Workpiece Machining Times (#3901, #3902) .....	302	Subprogram Call ; M98,M99 .....	259
			Subprogram Control; M98, M99 .....	259
O			Suppression of Single Block Stop and Miscellaneous Function Finish Signal Waiting (#3003) .....	295
	Operation Commands .....	316		
	Optional Block Skip .....	13	T	
	Optional Block Skip Addition ; /n .....	14	Table of G Code Lists .....	17
	Optional Block Skip; / .....	13	Tapping Mode ; G63.....	99
	Other Modals (#4101 - #4140, #4301 - #4340) ...	299	Thread Cutting Cycle ; G78.....	212
			Thread Cutting Mode.....	79
			Thread Cutting.....	54
			Timing Synchronization between Part Systems ; InL.....	392

Timing Synchronization between Part Systems.....	392
Tool Change Position Return ; G30.1 - G30.5 .....	366
Tool Compensation .....	292
Tool Compensation Start .....	149
Tool Compensation .....	148
Tool Functions (T8-digit BCD) .....	146
Tool Length Compensation .....	150
Tool Life Management (#60000 - #63016) .....	312
Tool Life Management II ; G10 L3, G11 .....	201
Tool Nose Point and Compensation Directions ...	155
Tool Nose R Compensation ; G40,G41,G42,G46 .....	153
Tool Nose Radius Compensation Operations .....	158
Tool Nose Wear Compensation .....	152
Types of Variables .....	279

## U

User Macro .....	268
------------------	-----

## V

Variable Commands .....	265
Variable Lead Thread Cutting ; G34 .....	61
Variable .....	277

## W

Workpiece Coordinate System Compensation (External Workpiece Coordinate Offset) (#5201 - #532n) .....	293
Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1).....	429

## Z

ZR device access variable .....	303
---------------------------------	-----



## Revision History

Date of revision	Manual No.	Revision details
Nov. 2006	IB(NA)1500275-A	First edition created.
Jan. 2007	IB(NA)1500275-B	Second edition created..
Mar. 2010	IB(NA)1500275-C	Third edition created. Corrections are made corresponding to C70 S/W version B2. Following chapter is added. -13.20 Chopping Following chapter is revised. -13.10 Parameter Input by Program; G10, G11 Following chapter is deleted -13.6.7 External output commands -Appendix 1. Parameter Input by Program N No. Correspondence Table Mistakes were corrected.
Jul. 2010	IB(NA)1500275-D	Fourth edition created. Reviewed "Precautions for Safety". Corrected the items below. - 10.5 Constant Surface Speed Control; G96, G97 - 10.6 Spindle Clamp Speed Setting; G92 Mistakes were corrected.
Mar. 2011	IB(NA)1500275-E	Fifth edition created. Corrections are made corresponding to C70 S/W version C5. Following chapters are added - 6.7 Helical Interpolation - 7.8 Rapid Traverse Constant Inclination Acceleration/Deceleration - 7.12 Deceleration Check - 14.12 Coordinate System for Rotary Axis Following chapters are deleted - 3.1 Tape codes - 3.3 Program address check function - 3.4 Tape memory format - 3.7 Parity H/V - 10.1 Spindle functions (S2-digits BCD) - 10.2 Spindle functions (S6-digits Analog) Structure of the following chapters are changed. -3 Program Formats ("Data Formats" in previous version) -6.1 Positioning (Rapid Traverse) ;G00 - 13.3 Fixed Cycle for Drilling Following chapters are revised. - 13.6.4 Types of Variables - 13.6.5 Operation Commands Mistakes were corrected.
Sep. 2012	IB(NA)1500275-F	Sixth edition created. Corrections were made corresponding to C70 S/W version D5. Following sections were added and the subsequent sections were accordingly re-numbered: - 13.14 High-accuracy Control ; G61.1 - 13.15 Coordinate Rotation by Program ; G68.1/G69.1 Following section was deleted and the subsequent sections were accordingly re-numbered: - 13.6.4.15 External Workpiece Coordinate System Compensation (#2501, #2601) Following sections were revised: - 3.4.3 Table of G Code Lists - 5.4 Decimal Point Input - 6.7 Helical Interpolation ; G17,G18,G19 and G02,G03 - 13.3.2 Face Tapping Cycle (Longitudinal tapping cycle) ; G84 (G88) - 13.6.4.13 Other Modals (#4101-#4120, #4301-#4320) - 13.17 Timing Synchronization between Part Systems (formerly "Waiting-and-simultaneous Operation")

(Continued on the following page)

Date of revision	Manual No.	Revision details
Sep. 2012	IB(NA)1500275-F	<p style="text-align: right;">(Continued from the following page)</p> <ul style="list-style-type: none"> <li>- 13.17.1 Timing Synchronization between Part Systems ; !nL (formerly "Waiting-and-simultaneous Operation ; !nL")</li> <li>- 13.17.2 Start Point Designation Timing Synchronization (Type 1) ; G115</li> <li>- 13.17.3 Start Point Designation Timing Synchronization (Type 2) ; G116</li> <li>- 16.3 Multi-step Skip Function 1 ; G31.n, G04</li> <li>- 16.4 Multi-step Skip Function 2 ; G31 P/L</li> </ul> <p>"Handling of our product" was added.  Mistakes were corrected.</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: +1-269-673-4092

### Ohio Service Satellite

LIMA, OHIO 45901, U.S.A.  
TEL: +1-847-478-2500 / FAX: +1-847-478-2650  
CLEVELAND, OHIO 44114, 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

### East Region Service Center

200 COTTONTAIL LANE SOMERSET, NEW JERSEY 08873, U.S.A.  
TEL: +1-732-560-4500 / FAX: +1-732-560-4531

### Pennsylvania Service Satellite

ERIE, PENNSYLVANIA 16510, U.S.A.  
TEL: +1-814-897-7820 / FAX: +1-814-987-7820

### South Region Service Center

1845 SATTELITE BOULEVARD STE. 450, DULUTH, GEORGIA 30097, U.S.A.  
TEL: +1-678-985-4529 / FAX: +1-678-258-4519

### Texas Service Satellites

GRAPEVINE, TEXAS 76051, U.S.A.  
TEL: +1-817-251-7468 / FAX: +1-817-416-5000  
HOUSTON, TEXAS 77001, U.S.A.  
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

### Florida Service Satellite

WEST MELBOURNE, FLORIDA 32904, U.S.A.  
TEL: +1-321-610-4436 / FAX: +1-321-610-4437

### 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 City Service Center

MARIANO ESCOBEDO 69 TLALNEPANTLA, 54030 EDO. DE MEXICO  
TEL: +52-55-9171-7662 / FAX: +52-55-9171-7649

### Monterrey Service Satellite

MONTERREY, N.L., 64720, MEXICO  
TEL: +52-81-8365-4171 / FAX: +52-81-8365-4171

## BRAZIL

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. (EUROPE FA CENTER)

GOTHAER STRASSE 10, 40880 RATINGEN, GERMANY  
TEL: +49-2102-486-0 / FAX: +49-2102-486-5910

### Germany Service Center

KURZE STRASSE, 40, 70794 FILDERSTADT-BONLANDEN, GERMANY  
TEL: +49-711-770598-121 / FAX: +49-711-770598-141

### France Service Center

25, BOULEVARD DES BOUVETS, 92741 NANTERRE CEDEX FRANCE  
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

### France (Lyon) Service Satellite

120, ALLEE JACQUES MONOD 69800 SAINT PRIEST FRANCE  
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

### Italy Service Center

VIALE COLLEONI 7-PALAZZO SIRIO CENTRO DIREZIONALE COLLEONI,  
20864 AGRATE BRIANZA MILANO ITALY  
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

### Italy (Padova) Service Satellite

VIA SAVELLI 24 - 35129 PADOVA ITALY  
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

### U.K. Service Center

TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.  
TEL: +44-1707-282-846 / FAX: +44-1707-27-8992

### Spain Service Center

CTRA. DE RUBI, 76-80-APDO. 420  
08190 SAINT CUGAT DEL VALLES, BARCELONA SPAIN  
TEL: +34-935-65-2236 / FAX: +34-935-69-1579

### Poland Service Center

UL. KRAKOWSKA 50, 32-083 BALICE, POLAND  
TEL: +48-12-630-4700 / FAX: +48-12-630-4701

### Turkey Service Center

ŞERIFALI MAH. NUTUK SOK. NO.5 34775  
ÜMRANIYE / ISTANBUL, TURKEY  
TEL: +90-216-526-3990 / FAX: +90-216-526-3995

### Czech Republic Service Center

TECHNOLOGICKA 3746, 708 00 OSTRAVA-PUSTKOVEC, CZECH REPUBLIC  
TEL: +420-59-5691-185 / FAX: +420-59-5691-199

### Russia Service Center

213, B.NOVODIMITROVSKAYA STR., 14/2, 127015 MOSCOW, RUSSIA  
TEL: +7-495-748-0191 / FAX: +7-495-748-0192

### Sweden Service Center

STRANDKULLEN, 718 91 FRÖVI, SWEDEN  
TEL: +46-581-700-20 / FAX: +46-581-700-75

### Bulgaria Service Center

4 ANDREJ LJAPCHEV BLVD, 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

Nezavisimosti pr.177, 220125 Minsk, Belarus  
TEL: +375-17-393-1177 / FAX: +375-17-393-0081

### South Africa Service Center

P.O. BOX 9234, EDLEEN, KEMPTON PARK GAUTENG, 1625 SOUTH AFRICA  
TEL: +27-11-394-8512 / FAX: +27-11-394-8513

**ASEAN****MITSUBISHI ELECTRIC ASIA PTE. LTD. (ASEAN FA CENTER)**

**Singapore Service Center**  
307 ALEXANDRA ROAD #05-01/02 MITSUBISHI ELECTRIC BUILDING SINGAPORE 159943  
TEL: +65-6473-2308 / FAX: +65-6476-7439

**Indonesia Service Center**  
THE PLAZZA OFFICE TOWER, 28TH FLOOR, J.L.M.H. THAMRIN KAV.28-30, JAKARTA, INDONESIA  
TEL: +62-21-2992-2333 / FAX: +62-21-2992-2555

**Malaysia (KL) Service Center**  
60, JALAN USJ 10/1B 47620 UEP SUBANG JAYA SELANGOR DARUL EHSAN, MALAYSIA  
TEL: +60-3-5631-7605 / FAX: +60-3-5631-7636

**Malaysia (Johor Baru) Service Center**  
NO. 16, JALAN SHAH BANDAR 1, TAMAN UNGKU TUN AMINAH, 81300 SKUDAI, JOHOR MALAYSIA  
TEL: +60-7-557-8218 / FAX: +60-7-557-3404

**Vietnam (Ho Chi Minh) Service Center**  
UNIT 2408-11, 24TH FLOOR, SAIGON TRADE CENTER, 37 TON DUC THANG STREET,  
DISTRICT 1, HO CHI MINH CITY, VIETNAM  
TEL: +84-8-3910 5945 / FAX: +84-8-3910 5947

**Vietnam (Hanoi) Service Center**  
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

**Philippines Service Center**  
UNIT NO.411, ALABAMG CORPORATE CENTER KM 25, WEST SERVICE ROAD  
SOUTH SUPERHIGHWAY, ALABAMG MUNTINLUPA METRO MANILA, PHILIPPINES 1771  
TEL: +63-2-807-2416 / FAX: +63-2-807-2417

**MITSUBISHI ELECTRIC AUTOMATION (THAILAND) CO., LTD. (THAILAND FA CENTER)**

BANG-CHAN INDUSTRIAL ESTATE NO.111 SOI SERITHAI 54  
T.KANNAYAO, A.KANNAYAO, BANGKOK 10230, THAILAND  
TEL: +66-2906-8255 / FAX: +66-2906-3239

**Thailand Service Center**  
898/19,20,21,22 S.V. CITY BUILDING OFFICE TOWER 1, FLOOR 7  
RAMA III RD., BANGPONGPANG, YANNAWA, BANGKOK 10120, THAILAND  
TEL: +66-2-682-6522 / FAX: +66-2-682-9750

**INDIA****MITSUBISHI ELECTRIC 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: +86-21-2308-2830

**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: +86-10-6518-3907  
**China (Beijing) Service Dealer**

**China (Tianjin) Service Center**  
B-2 801/802, YOUYI BUILDING, NO.50 YOUYI ROAD, HEXI DISTRICT,  
TIANJIN 300061, CHINA  
TEL: +86-22-2813-1015 / FAX: +86-22-2813-1017  
**China (Shenyang) Service Satellite**  
**China (Changchun) Service Satellite**

**China (Chengdu) Service Center**  
ROOM 407-408, OFFICE TOWER AT SHANGRI-LA CENTER, NO. 9 BINJIANG DONG ROAD,  
JINJIANG DISTRICT, CHENGDU, SICHUAN 610021, CHINA  
TEL: +86-28-8446-8030 / FAX: +86-28-8446-8630

**China (Shenzhen) Service Center**  
ROOM 2512-2516, 25/F., GREAT CHINA INTERNATIONAL EXCHANGE SQUARE, JINTIAN RD.S.,  
FUTIAN DISTRICT, SHENZHEN 518034, CHINA  
TEL: +86-755-2399-8272 / FAX: +86-755-8218-4776  
**China (Xiamen) Service Dealer**  
**China (Dongguan) Service Dealer**

**KOREA****MITSUBISHI ELECTRIC AUTOMATION KOREA CO., LTD. (KOREA FA CENTER)**

**Korea Service Center**  
1480-6, GAYANG-DONG, GANGSEO-GU, SEOUL 157-200, KOREA  
TEL: +82-2-3660-9602 / FAX: +82-2-3664-8668

**Korea Taegu Service Satellite**  
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**  
NO.8-1, GONG YEH 16TH RD., TAICHUNG INDUSTRIAL PARK, SITUN DIST.,  
TAICHUNG CITY 407, TAIWAN R.O.C.  
TEL: +886-4-2359-0688 / FAX: +886-4-2359-0689

**Taiwan (Taipei) Service Center**  
10F, NO.88, SEC.6, CHUNG-SHAN N. RD., SHI LIN DIST., TAIPEI CITY 111, TAIWAN R.O.C.  
TEL: +886-2-2833-5430 / FAX: +886-2-2833-5433

**Taiwan (Tainan) Service Center**  
11F-1, NO.30, ZHONGZHENG S. ROAD, YONGKANG DISTRICT, TAINAN CITY 710, 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 2006-2012 MITSUBISHI ELECTRIC CORPORATION  
ALL RIGHTS RESERVED

# MITSUBISHI CNC



MODEL	C70
MODEL CODE	100-007
Manual No.	IB-1500275