Changes for the Better

MITSUBISHI CNC

Programming Manual (Lathe System)

E70 Series

Introduction

This manual is a guide for using the MITSUBISHI CNC E70 Series.
Programming is described in this manual, so read this manual thoroughly before starting programming. Thoroughly study the "Precautions for Safety" on the following page to ensure safe use of this NC unit.

Details described in this manual

⚠ CAUTION

⚠ **For items described as "Restrictions" or "Usable State" in this manual, the instruction manual issued by the machine tool builder takes precedence over this manual.**

⚠ **Items not described in this manual must be interpreted as "not possible".**

⚠ **This manual is written on the assumption that all option functions are added.**
Refer to the specifications issued by the machine tool builder before starting use.

⚠ **Refer to the Instruction Manual issued by each machine tool builder for details on each machine tool.**

⚠ **Some screens and functions may differ depending on the NC system (or its version), and some functions may not be possible. Please confirm the specifications before use.**

General precautions

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

Precautions for Safety

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

⚠ 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 following sings indicate prohibition and compulsory.

🛇	**This sign indicates prohibited behavior (must not do).** **For example, 🛇 indicates "Keep fire away".**

●	**This sign indicated a thing that is pompously (must do).** **For example, ⏚ indicates "it must be grounded".**

The meaning of each pictorial sing is as follows.

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

For Safe Use

Mitsubishi CNC is designed and manufactured solely for applications to machine tools to be used for industrial purposes.
Do not use this product in any applications other than those specified above, especially those which are substantially influential on the public interest or which are expected to have significant influence on human lives or properties.

⚠ DANGER

Not applicable in this manual.

⚠ WARNING

1. Items related to operation

⚠ **If the operation start position is set in a block which is in the middle of the program and the program is started, the program before the set block is not executed. Please confirm that G and F modal and coordinate values are appropriate. If there are coordinate system shift commands or M, S, T and B commands before the block set as the start position, carry out the required commands using the MDI, etc. If the program is run from the set block without carrying out these operations, there is a danger of interference with the machine or of machine operation at an unexpected speed, which may result in breakage of tools or machine tool or may cause damage to the operators.**

⚠ **Under the constant surface speed control (during G96 modal), if the axis targeted for the constant surface speed control (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.**

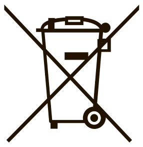
2. Items related to operation

- ⚠ **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.**
- ⚠ **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.**

Disposal



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

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

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

Trademarks

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

Ethernet is a registered trademark of Xerox Corporation in the United States and/or other countries.
Microsoft® and Windows® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.
CompactFlash and CF are either trademarks or registered trademarks of SanDisk Corporation in the United States and/or other countries.
UNIX is a registered trademark of The Open Group in the United States and/or other countries.
Intel® and Pentium® are either trademarks or registered trademarks of Intel Corporation in the United States and/or other countries.
Other company and product names that appear in this manual are trademarks or registered trademarks of the respective companies.

本製品の取扱いについて

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

Handling of our product

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

본 제품의 취급에 대해서

(한국어 /Korean)
이 기기는 업무용 (A 급) 전자파적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며 가정외의 지역에서 사용하는 것을 목적으로 합니다 .

CONTENTS

1

Control Axes

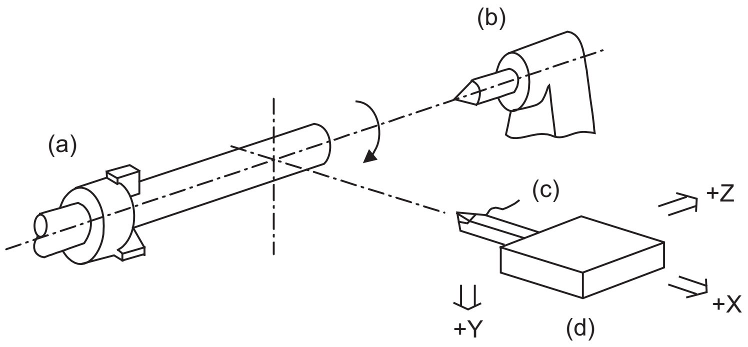
1.1 Coordinate Words and Control Axes

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

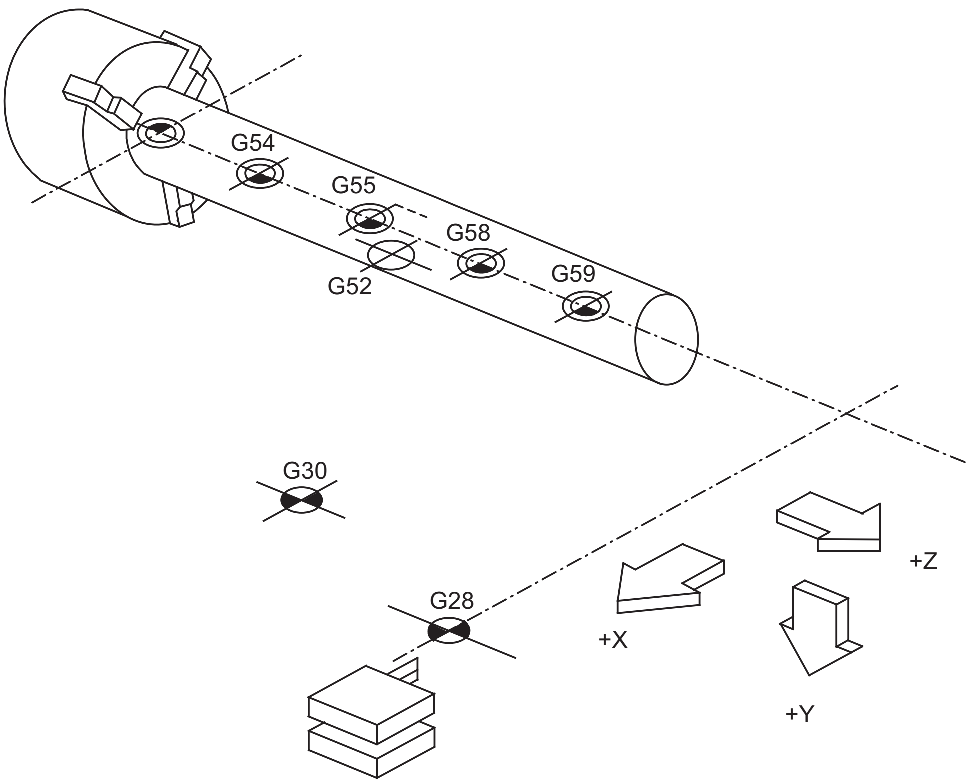
Coordinate axes and polarities



(a) Spindle stock
(b) Tailstock
(c) Tool
(d) Turret

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



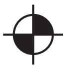
⊕ Reference position

⊕ Basic machine coordinate

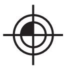
⊕ Workpiece coordinate zero points

⊕ Local coordinate zero point

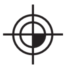
1.2 Coordinate Systems and Coordinate Zero Point Symbols



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

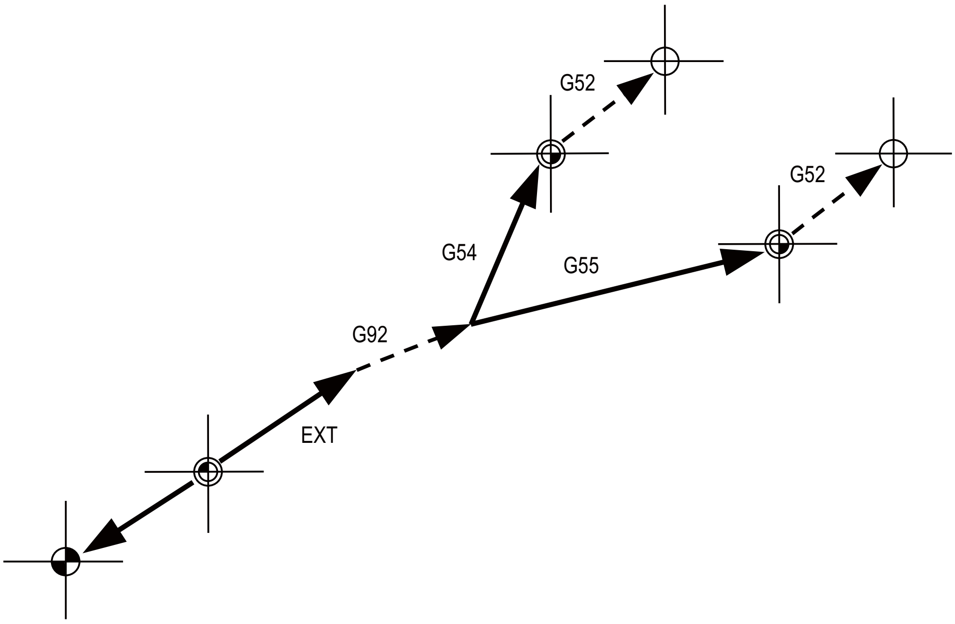


Basic machine coordinate zero point:
A position specific to machine



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, etc.



◑ Reference position

◉ Basic machine coordinate zero point

◉ Workpiece coordinate zero points

⊕ Local coordinate zero point

→ Offset set by a parameter

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

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

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

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

2

Least Command Increments

2.1 Input Setting Unit

Function and purpose

The input setting units are the units of setting data including tool compensation amounts and workpiece coordinates compensation.
The program command units are the units of movement amounts in programs.
These are expressed with mm, inch or degree (°).

Detailed description

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

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

Precautions

(1) Inch/metric changeover can be handled by either a parameter screen (#1041 I_inch: valid only when the power is turned ON) or G commands (G20 or G21).
However, the changeover by a G command applies only to the program command units, and not to the input setting units. Consequently, the tool 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.

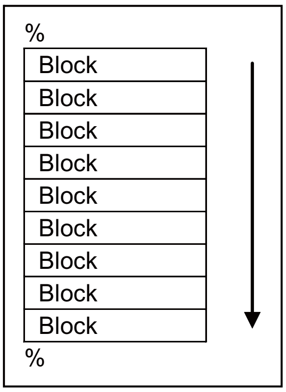
(3) When performing a circular interpolation between the axes whose program command units are different, the center command (I, J, K) and the radius command (R) are designated by the input setting units. (Use a decimal point to avoid confusion.)

3

Program Formats

3.1 Program Format

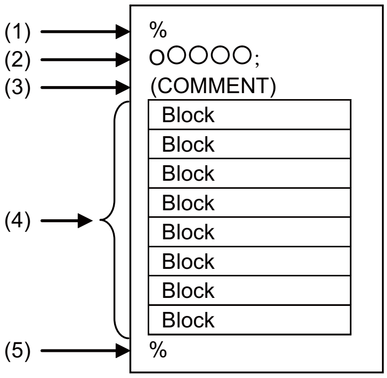
A collection of commands assigned to an NC to move a machine is called "program".
A program is a collection of units called "block" which specifies a sequence of machine tool operations.
Blocks are written in the order of the actual movement of a tool.
A block is a collection of units called "word" which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



Detailed description

Program

A program format looks as follows.



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

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

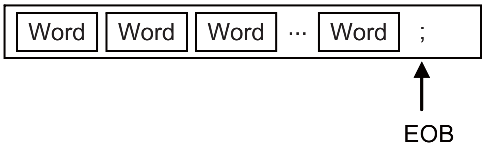
(3) Comment
Data between control out "(" and control in ")" is ignored.
Information including program names and comments can be written in.

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

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

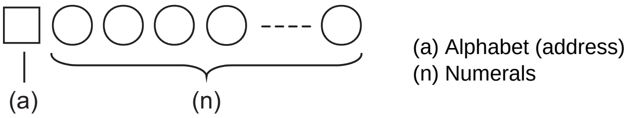
Block and word

[Block]



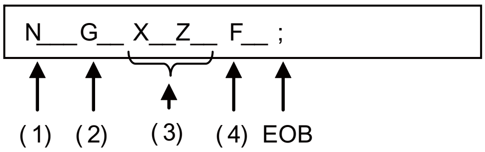
A block is a least command increment, consisting of words.
It contains the information which is required for a tool machine to execute a specific operation. One block unit constitutes a complete command.
The end of each block is marked with an End of Block (EOB, expressed as ";" for the sake of convenience).

[Word]



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

The major contents of a word are described below.



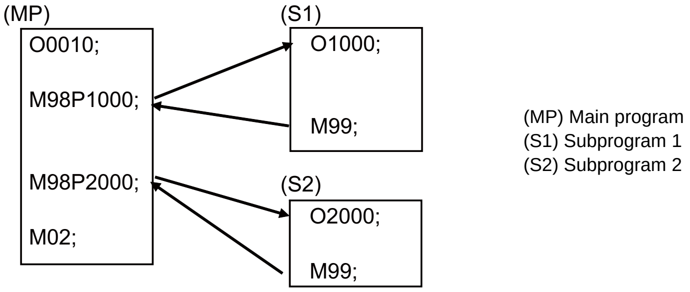
(1) Sequence No.
A "sequence No." consists of the address "N" followed by numbers of up to 6 digits (Normally 3 or 4 digits). It is used as an index when searching a necessary block in a program (as branch destination and etc.).
It does not affect the operation of a tool machine.

(2) Preparatory functions (G code, G function)
"Preparatory function (G code, G function)" consists of the address G followed by numbers of 2 or 3 digits (it may include 1 digit after the decimal point). G codes are mainly used to designate functions, such as axis movements and setting of coordinate systems. For example, G00 executes a positioning and G01 executes a linear interpolation.
There are 4 types of G code systems: 2, 3, 6 and 7. 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.

3.2 File Format

Function and purpose

Program file can be created using NC edit screen and PC.
It can be input/output between NC memory and an external I/O device. For the details of input/output method, refer to the instruction manual.
Program file format differs depending on the device which creates the program.

Detailed description

Devices available for input/output

Devices which can input/output program files are as follows.
- NC memory
- Serial
- Memory card (front IC card)
- USB memory
- Ethernet

Program file format

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

(1) NC memory (Creates program on NC)

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

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

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

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

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

(3) External device (serial)

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

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

3.3 Optional Block Skip

3.3.1 Optional Block Skip; /

Function and purpose

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

Detailed description

Provided that the optional block skip switch is ON, sections of a machining program from a "/" to the end of the block are ignored. They are executed if the switch is OFF.
Parity check is valid regardless of whether the optional block skip switch is ON or OFF.
When, for instance, all blocks are to be executed for one workpiece but specific blocks are not to be executedfor another workpiece, one machining program can be used to machine different parts by inserting the "/" into those specific blocks.

Program example

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

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

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

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

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

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

(3) When the parameter "#1274 ext10/bit4" is set to "1":
When a "/" is placed in a bracketed expression or when an expression that includes a "/" is on the right side of an equation, the "/" is interpreted as a division instruction.
As for a "/" that appears in any other contexts, the section of the block following the "/" will be skipped if the optional skip signal is ON, and the "/" itself will be ignored if the optional skip signal is OFF.

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

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

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

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

Precautions

(1) When parameter "#1274 ext10/bit4" = "0" and parameter "#1226 aux10/bit1" = "0", put the "/" code for optional block skip at the beginning of a block. If it is placed inside the block, it is assumed as a user macro, a division instruction.
(Example)
N20 G1 X25. /Z25. ; ....NG (User macro, a division instruction; a program error results.)
/N20 G1 X25. Z25. ; ....OK
When parameter "#1274 ext10/bit4" = "0" and parameter "#1226 aux10/bit1" = "1", a "/" placed in the middle of a block functions as a starting point of the optional skip. To use a "/" as a division instruction, bracket (enclose in square brackets) the necessary section.

(2) A space immediately followed by a "/" at the very beginning of a block is always regarded as equal to a "/" at the head of a block regardless of the value set in parameter "#1226 aux10/bit1".

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

(4) This function is valid even during a sequence No. search.

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

3.4 G code

3.4.1 Modal, unmodal

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

3.4.2 G code Lists

G codes include the four G code lists 2, 3, 6 and 7.

cmdtyp	G code lists
3	2
4	3
7	6
8	7

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
Standard		Special				
2	3	6	7			
ΔG00	ΔG00	ΔG00	ΔG00	01	Positioning	6.1
ΔG01	ΔG01	ΔG01	ΔG01	01	Linear interpolation	6.2
G02	G02	G02	G02	01	Circular interpolation CW/Helical interpolation CW	6.3 6.4 6.7
G03	G03	G03	G03	01	Circular interpolation CCW/Helical interpolation CCW	6.3 6.4 6.7
G04	G04	G04	G04	00	Dwell	8.1
G09	G09	G09	G09	00	Exact stop check	7.11
G10	G10	G10	G10	00	Programmable parameter input, Programmable compensation data input, Tool life management data registration	12.5 12.6 13.11
G11	G11	G11	G11	00	Programmable parameter input cancel, Programmable compensation data input cancel, Tool life management data registration cancel	12.5 12.6 13.11
ΔG17	ΔG17	ΔG17	ΔG17	02	Plane selection X-Y	6.5
ΔG18	ΔG18	ΔG18	ΔG18	02	Plane selection Z-X	6.5
ΔG19	ΔG19	ΔG19	ΔG19	02	Plane selection Y-Z	6.5
ΔG20	ΔG20	ΔG20	ΔG20	06	Inch command	5.3
ΔG21	ΔG21	ΔG21	ΔG21	06	Metric command	5.3
G22	G22			04	Barrier check ON	15.1
*G23	*G23			04	Barrier check OFF	15.1
		G22	G22	00	Soft limit ON	15.2
		G23	G23	00	Soft limit OFF	15.2
G27	G27	G27	G27	00	Reference position check	14.9
G28	G28	G28	G28	00	Automatic reference position return	14.7
G29	G29	G29	G29	00	Start position return	14.7
G30	G30	G30	G30	00	2nd, 3rd and 4th reference position return	14.8
G30.1	G30.1	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	G31	G31	00	Skip/Multi-step skip function 2	16.2 16.4
G31.1	G31.1	G31.1	G31.1	00	Multi-step skip function 1-1	16.3
G31.2	G31.2	G31.2	G31.2	00	Multi-step skip function 1-2	16.3
G31.3	G31.3	G31.3	G31.3	00	Multi-step skip function 1-3	16.3
G32	G33	G32	G33	01	Thread cutting	6.6.1 6.6.2
G34	G34	G34	G34	01	Variable lead thread cutting	6.6.4
G37	G37	G36/G37 G37.1 G37.2	G36/G37 G37.1 G37.2	00	Automatic Tool Length Measurement	16.1
*G40	*G40	*G40	*G40	07	Tool nose radius compensation cancel	12.4
G41	G41	G41	G41	07	Tool nose radius compensation left	12.4
G42	G42	G42	G42	07	Tool nose radius compensation right	12.4

G code lists				Group	Function	Section
Standard		Special				
2	3	6	7			
G46	G46	G46	G46	07	Tool nose radius compensation (automatic direction identification) ON	12.4
G43.1	G43.1	G43.1	G43.1	20	1st spindle control mode ON	10.5.2
G44.1	G44.1	G44.1	G44.1	20	Selected spindle control mode	10.5.2
G47.1	G47.1	G47.1	G47.1	20	All spindles simultaneous control mode ON	10.5.2
G50	G92	G50	G92	00	Spindle Clamp Speed Setting Coordinate system setting	10.3 14.6
G52	G52	G52	G52	00	Local coordinate system setting	14.11
G53	G53	G53	G53	00	Basic Machine Coordinate System Selection	14.5
*G54	*G54	*G54	*G54	12	Workpiece coordinate system selection 1	14.10
G55	G55	G55	G55	12	Workpiece coordinate system selection 2	14.10
G56	G56	G56	G56	12	Workpiece coordinate system selection 3	14.10
G57	G57	G57	G57	12	Workpiece coordinate system selection 4	14.10
G58	G58	G58	G58	12	Workpiece coordinate system selection 5	14.10
G59	G59	G59	G59	12	Workpiece coordinate system selection 6	14.10
G54.1	G54.1	G54.1	G54.1	12	Work coordinate system selection 48 sets expanded	14.10
G61	G61	G61	G61	13	Exact stop check mode	7.12
G62	G62	G62	G62	13	Automatic Corner Override	7.14
G63	G63	G63	G63	13/19	Tapping mode	7.15
*G64	*G64	*G64	*G64	13/19	Cutting mode	7.16
G65	G65	G65	G65	00	User macro call	13.6.1
G66	G66	G66	G66	14	User macro modal call A	13.6.1
G66.1	G66.1	G66.1	G66.1	14	User macro modal call B	13.6.1
*G67	*G67	*G67	*G67	14	User macro modal call cancel	13.6.1
G70	G70	G70	G70	09	Finishing cycle	13.2.4
G71	G71	G71	G71	09	Longitudinal rough cutting cycle	13.2.1
G72	G72	G72	G72	09	Face rough cutting cycle	13.2.2
G73	G73	G73	G73	09	Formed material rough cutting cycle	13.2.3
G74	G74	G74	G74	09	Face cut-off cycle	13.2.5
G75	G75	G75	G75	09	Longitudinal cut-off cycle	13.2.6
G76	G76	G76	G76	09	Compound thread cutting cycle	13.2.7
G90	G77	G90	G77	09	Longitudinal cutting fixed cycle	13.1.1
G92	G78	G92	G78	09	Thread cutting fixed cycle	13.1.2
G94	G79	G94	G79	09	Face cutting fixed cycle	13.1.3
*G80	*G80	*G80	*G80	09	Fixed cycle for drilling cancel	13.3 13.3.5
G79	G83.2	G79	G83.2	09	Deep hole drilling cycle 2	13.3.4
G83	G83	G83	G83	09	Deep hole drilling cycle (Z axis)	13.3
G84	G84	G84	G84	09	Tapping cycle (Z axis)	13.3 13.3.2
G85	G85	G85	G85	09	Boring cycle (Z axis)	13.3 13.3.3
G87	G87	G87	G87	09	Deep hole drilling cycle (X axis)	13.3 13.3.1
G88	G88	G88	G88	09	Tapping cycle (X axis)	13.3 13.3.2

G code lists				Group	Function	Section
Standard		Special				
2	3	6	7			
G89	G89	G89	G89	09	Boring cycle (X axis)	13.3 13.3.3
G84.1	G84.1	G84.1	G84.1	09	Reverse tapping cycle (Z axis)	13.3.2
G88.1	G88.1	G88.1	G88.1	09	Reverse tapping cycle (X axis)	13.3.2
G50.3	G92.1	G50.3	G92.1	00	Workpiece coordinate preset	14.12
ΔG96	ΔG96	ΔG96	ΔG96	17	Constant surface speed control ON	10.2
ΔG97	ΔG97	ΔG97	ΔG97	17	Constant surface speed control OFF	10.2
ΔG98	ΔG94	ΔG98	ΔG94	05	Feed per minute (asynchronous feed)	7.4
ΔG99	ΔG95	ΔG99	ΔG95	05	Feed per revolution (synchronous feed)	7.4
-	ΔG90	-	ΔG90	03	Absolute value command	5.1
-	ΔG91	-	ΔG91	03	Incremental value command	5.1
-	*G98	-	*G98	10	Fixed cycle initial level return	13.3 13.3.7
-	G99	-	G99	10	Fixed cycle R point level return	13.3 13.3.7

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) If two or more G codes from the same group are commanded in a block, the last G code will be valid.

(6) 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.

(7) 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.

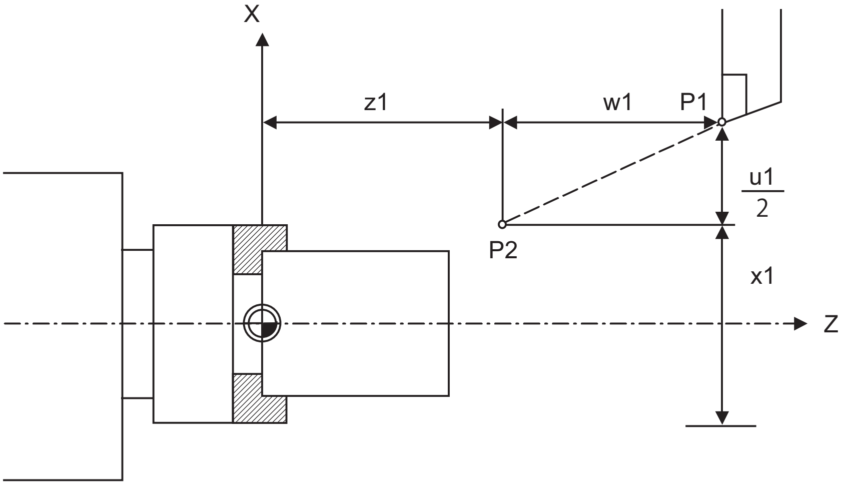
5

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 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 Zz1 ;
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

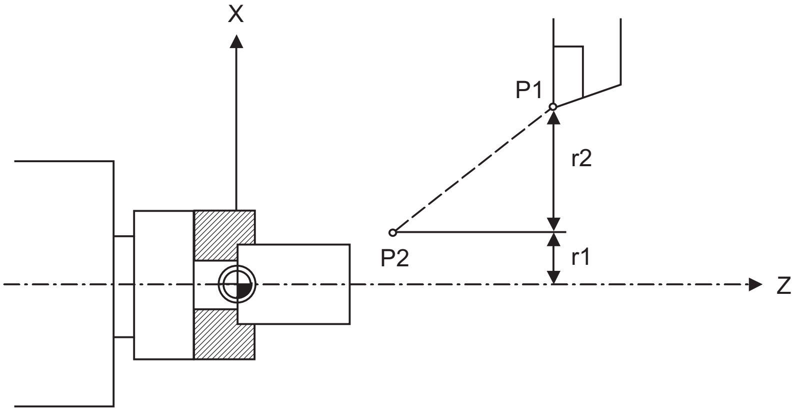
Precautions

(1) Designation of a circular radius (R) and center (I,J,K) is always conducted with incremental values.
(2) When parameter "#1076 AbsInc" is 1, and H is used for the incremental command address, address H of blocks M98 will be handled as the parameter of 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	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.
X = r1	X = 2 * r1	U = r2	U = 2 * r2	

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.

Output unit, command unit and setting unit

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

NC axis

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

PLC axis

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

Precautions

(1) The parameter and tool data will be input/output with the unit set by "#1041 I_inch".
If "#1041 I_inch" is not found in the parameter input data, the unit will follow the unit currently set to NC.

(2) The unit of read/write used in PLC window is fixed to metric unit regardless of a parameter and G20/G21 command modal.

5.4 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 (rotary)	Feedrate	Dwell
Input unit [mm]	-99999.999 to 99999.999	-99999.999 to 99999.999	0.001 to 10000000.000	0 to 99999.999
Input unit [inch]	-9999.9999 to 9999.9999		0.0001 to 1000000.0000	

(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.

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 No.	
	Invalid	Parameter input by program Axis No.	
	Valid	Deep hole drilling cycle (2) Safety distance	
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	Tool compensation data input by program 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	
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 Nos. in subprograms	
	Invalid	Parameter input by program bit type data	
	Invalid	Selection of linear - arc intersection (geometric)	
I	Valid	Circular center coordinates	
	Valid	Nose R compensation vector components	
	Valid	Deep hole drilling cycle (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 vector components	
	Invalid	Deep hole drilling cycle (2) Dwell at return point	
	Valid	Hole drilling cycle G1 in-position width	,J
K	Valid	Circular center coordinates	
	Valid	Nose R compensation vector components	
	Invalid	Hole drilling 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)	
L	Invalid	Subprogram Number of repetitions	
	Invalid	Tool compensation data input by program Type selection	L2 L10 L11
	Invalid	Parameter input by program selection	L70
	Invalid	Parameter input by program two-word type data	4byte
	Invalid	Tool life data	
M	Invalid	Miscellaneous function codes	
N	Invalid	Sequence Nos.	
	Invalid	Parameter input by program data No.	
O	Invalid	Program No.	

Address	Decimal Point Command	Application	Remarks
P	Invalid	Dwell time	
	Invalid	Subprogram call Program No.	
	Invalid	2nd, 3rd and 4th reference position No.	
	Invalid	Constant surface speed control, axis No.	
	Invalid	MRC finishing shape start sequence No.	
	Valid	Cut-off cycle shift amount/cut amount	
	Invalid	Compound thread cutting cycle number of cutting passes, chamfering, tool nose angle	
	Valid	Compound thread cutting cycle Thread height	
	Invalid	Tool compensation data input by program input compensation No.	
	Invalid	Parameter input by program section No.	
	Valid	Coordinate position data	
	Invalid	Skip signal command	
	Valid	Arc center coordinates (absolute value) (geometric)	
	Invalid	Subprogram return destination sequence No.	
	Invalid	Extended workpiece coordinate system No.	
	Invalid	Tool life data group No.	
Q	Invalid	Minimum spindle clamp rotation speed	
	Invalid	MRC finishing shape end	
	Valid	Cut-off cycle Cut amount/shift amount	
	Valid	Compound thread cutting cycle Minimum cut amount	
	Valid	Compound thread cutting cycle First cut amount	
	Valid	Deep hole drilling cycle 1 Cut amount of each pass	
	Invalid	Tool compensation data input by program Hypothetical tool nose point No.	
	Invalid	Deep hole drilling cycle (2) Dwell at cut point	
	Valid	Arc center coordinates (absolute value) (geometric)	
	Valid	Thread cutting start shift angle	
	Invalid	Tool life data management method	
R	Valid	R-designated arc radius	
	Valid	Arc radius of corner rounding	,R
	Valid	Automatic tool length measurement, deceleration range r	
	Valid	MRC longitudinal/face escape amount	
	Invalid	MRC shaping division No.	
	Valid	Cut-off cycle, return amount	
	Valid	Cut-off cycle, escape amount	
	Valid	Compound thread cutting cycle Finishing allowance	
	Valid	Compound thread cutting cycle turning cycle, taper difference	
	Valid	Hole drilling cycle/deep hole drilling cycle (2) Distance to R point	
	Valid	Tool compensation data input by program Nose R compensation	
	Valid	Coordinate position data	
	Valid	Rough cutting cycle (longitudinal) (face) pull amount	
	Invalid	Synchronous tap/ asynchronous tap changeover	,R
	Valid	Synchronous spindle phase shift amount	
	Valid	Rotation angle	
	Invalid	Skip acceleration/deceleration time constant	
S	Invalid	Spindle function codes	
	Invalid	Maximum spindle clamp rotation speed	
	Invalid	Constant surface speed control, surface speed	
	Invalid	Parameter input by program word type data	2byte

Address	Decimal Point Command	Application	Remarks
T	Invalid	Tool function codes	
U	Valid	Coordinate position data	
	Valid	Tool compensation data input by program	
	Valid	Rough cutting cycle (longitudinal) Cut amount	
	Valid	Dwell	
V	Valid	Coordinate position data	
	Valid	Tool compensation data input by program	
W	Valid	Coordinate position data	
	Valid	Tool compensation data input by program	
	Valid	Rough cutting cycle (longitudinal) Cut amount	
X	Valid	Coordinate position data	
	Valid	Dwell	
	Valid	Tool compensation data input by program	
Y	Valid	Coordinate position data	
	Valid	Tool compensation data input by program	
Z	Valid	Coordinate position data	
	Valid	Tool compensation data input by program	

(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 = 1 μ m	When 1 = 10 μ m	When 1 = 1mm
G0 X123.45 (decimal points are all mm points)	X123.450 mm	X123.450 mm	X123.450 mm
G0 X12345	X12.345 mm (last digit is 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. This is valid only in the commanded block. A block that does not contain this address will follow the parameter "#1193 inpos" settings. 1 to 999999

The command addresses are valid for all additional axes.

Detailed description

(1) Positioning will be performed at the rapid traverse rate set in the parameter "#2001 rapid".

(2) G00 command belongs to the 01 group and is modal. When G00 command is successively issued, the following blocks can be specified only by the coordinate words.

(3) In the G00 mode, acceleration and deceleration are always carried out at the start point and end point of the block. Before advancing to the next block, a commanded deceleration or an in-position check is conducted at the end point to confirm that the movement is completed for all the moving axes.

(4) G functions (G83 to G89) in the 09 group are cancelled (G80) by the G00 command.

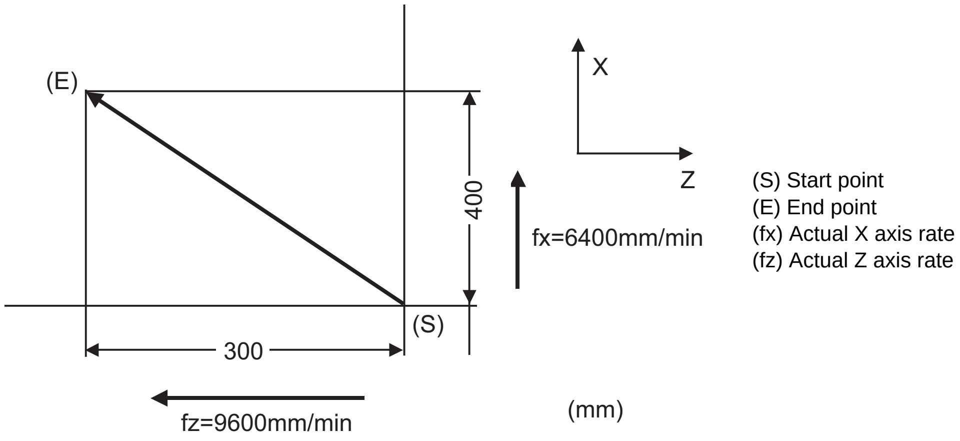
⚠ **CAUTION**

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

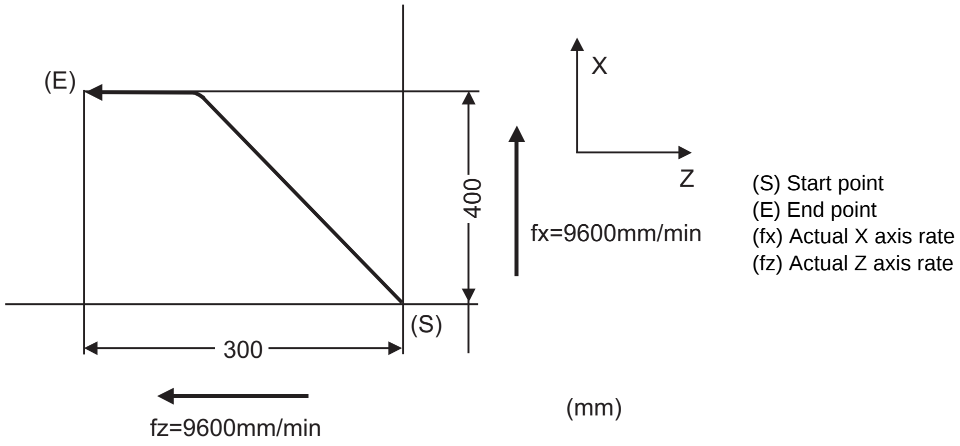
Tool path

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.

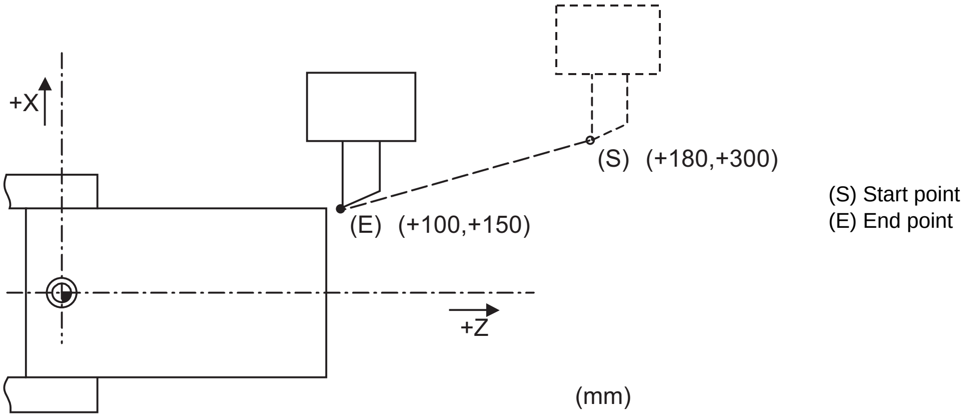
(1) Linear path: When the parameter "#1086 G0Intp" is set to "0"
In positioning, a tool follows the shortest path which connects the start point and the end point. The positioning speed is automatically calculated so that the shortest distribution time is obtained in order that the commanded speeds for 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")	

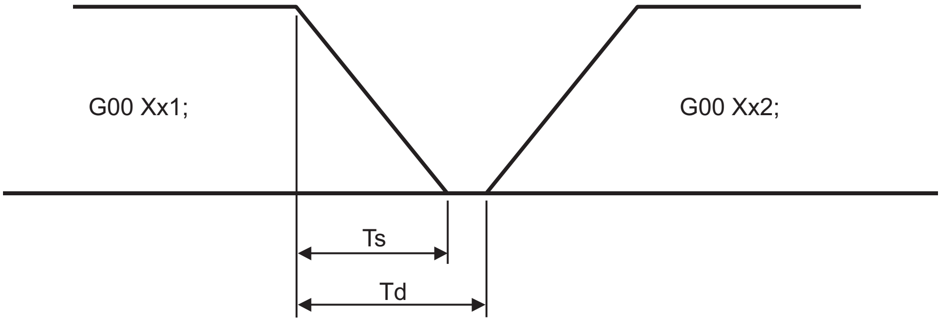
Cutting feedrate (G01)		**#1193 inpos**	
		0	**1**
,I command	No	Commanded deceleration method (Commanded deceleration check which varies according to the type of acceleration/deceleration, set in "#2003 smgst" bit7-4)	In-position check method (In-position check by "#2078 G1inps", "#2224 SV024")
	Yes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

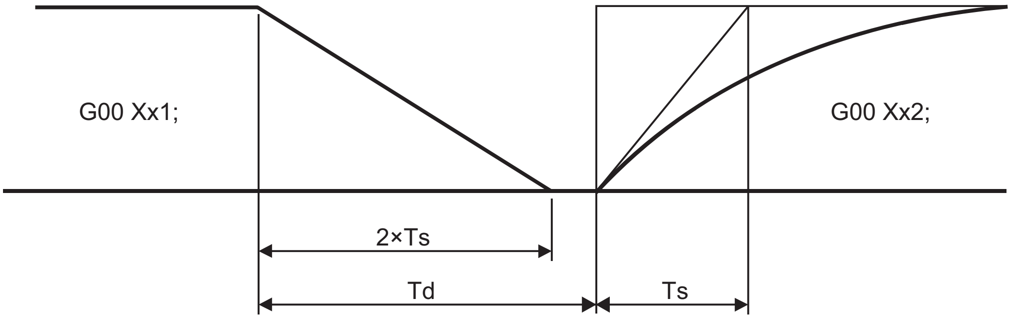
Upon completion of the rapid traverse (G00), the next block will be executed after the deceleration check time (Td) has elapsed.
The deceleration check time (Td) is as follows, depending on the acceleration/deceleration type set in the parameter "#2003 smgst".

(1) Linear acceleration/linear deceleration



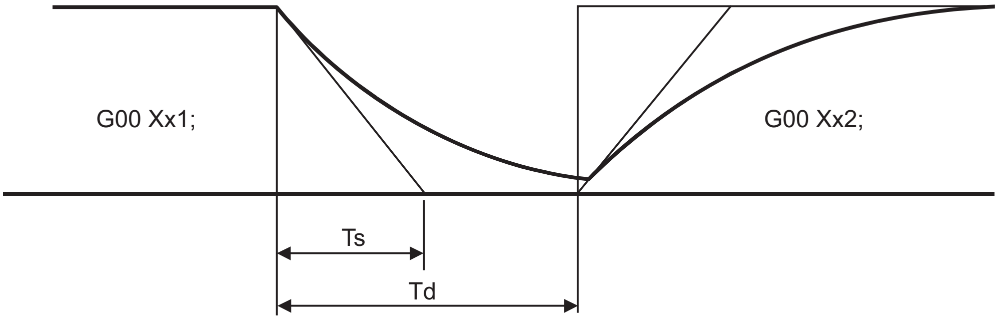
(Ts) Acceleration/deceleration time constant
(Td) Deceleration check time: Td = Ts + (0 to 7ms)

(2) Exponential acceleration/linear deceleration



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

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



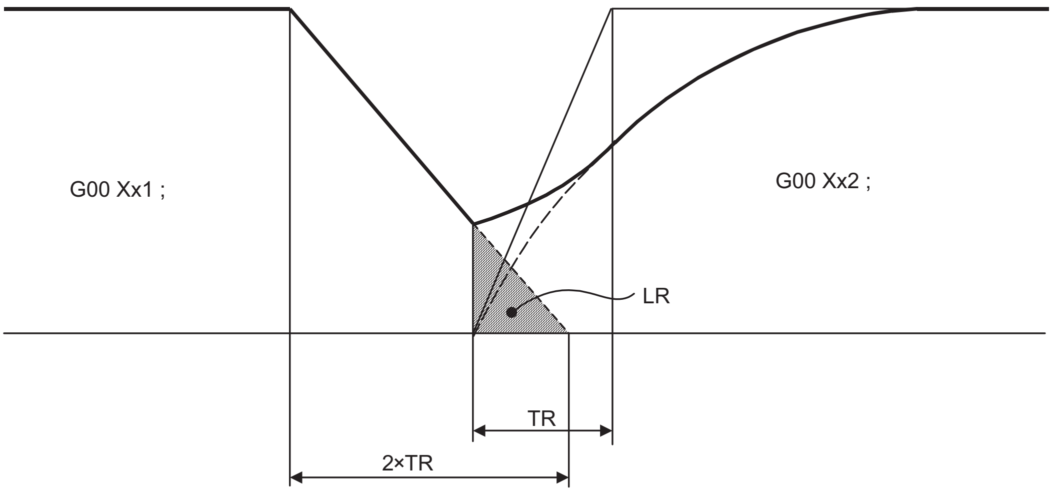
(Ts) Acceleration/deceleration time constant
(Td) Deceleration check time: Td = 2 × Ts + (0 to 7ms)

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

In-position check method when "inpos" = 1

Upon completion of the rapid traverse (G00), the next block will be executed after confirming that the remaining distances for each axis are below the fixed amounts.
The confirmation of the remaining distance should be done with the imposition width.
The bigger one of the servo parameter "#2224 SV024" or G0 in-position width "#2077 G0inps" (For G01, in-position width "#2078 G1inps"), will be adapted as the in-position width.

G0 deceleration pattern

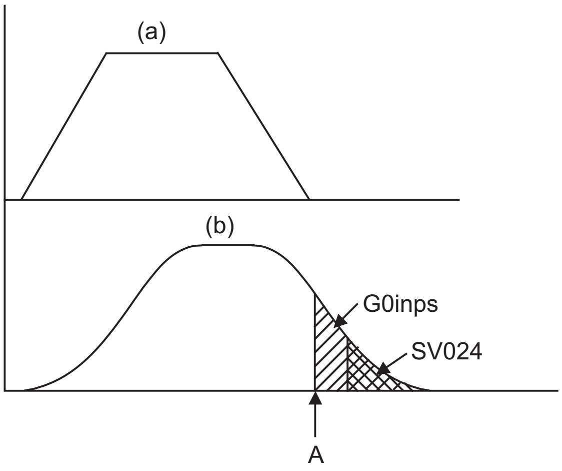


(TR) Rapid traverse acceleration and deceleration time constant
(LR) In-position width

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

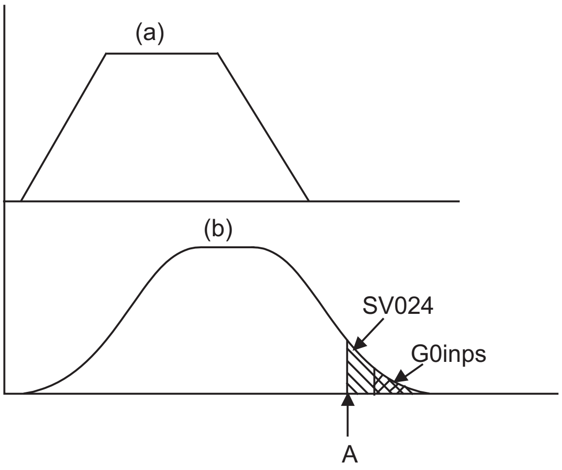
The purpose of the rapid traverse deceleration check is to minimize the positioning time. The bigger the setting value for the in-position width, the shorter the time is, but the remaining distance of the previous block at the start of the next block also becomes larger, and this could become an obstacle in the actual processing work.
The check for the remaining distance is done at set intervals. Accordingly, it may not be possible to get the effect of time reduction for positioning as in-position width setting value.

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



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

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



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

Programmable in-position width command

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

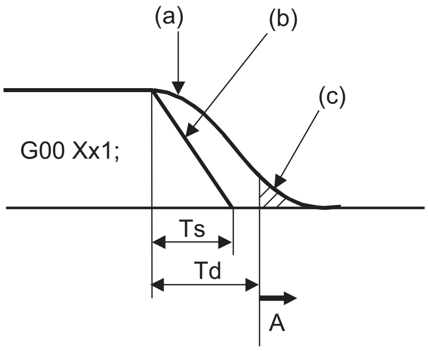
G00 X_ Z_ ,I_ ;	
X,Z	Positioning coordinate value of each axis
,I	In-position width

Execution of the next block starts after confirming that the position error amount of the positioning (rapid traverse: G00) command block is less than the in-position width issued in this command.
The bigger one of in-position width (SV024, G0inps (For G01, G1inps)) with parameter or in-position width specified by program will be adapted as the in-position width.
When there are several movement axes, the system confirms that the position error amount of each movement axis is less than the in-position width issued in this command before executing the next block.

The differences of In-position check

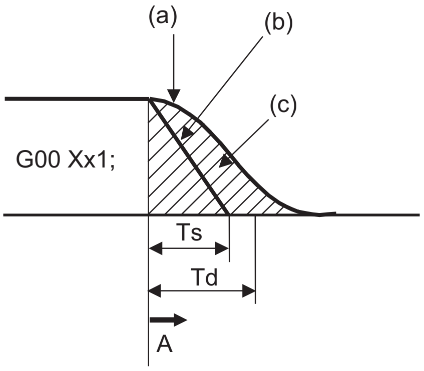
The differences between the in-position check with parameter and with programmable command are as follows:

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



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

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



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

6.2 Linear Interpolation ; G01

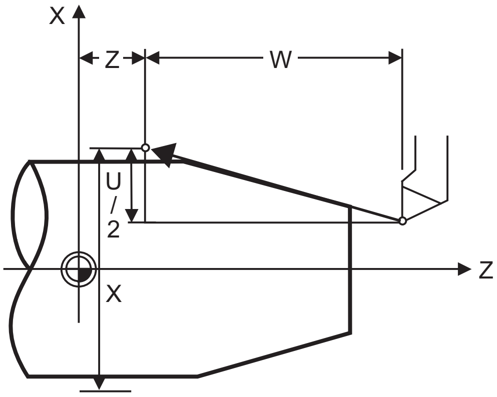
Function and purpose

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

Command format

G01 X__/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 block that does not contain this address will follow the parameter "#1193 inpos" settings. 1 to 999999



Detailed description

(1) G01 command is a modal command in the 01 group. When G01 command is issued in succession, it can only be issued with coordinate words in subsequent blocks.
(2) The feedrate for a rotary axis is commanded by °/min (decimal point position unit). (F300=300°/min)
(3) The G functions (G70 to G89) in the 09 group are cancelled (G80) by the G01 command.

Programmable in-position width command for linear interpolation

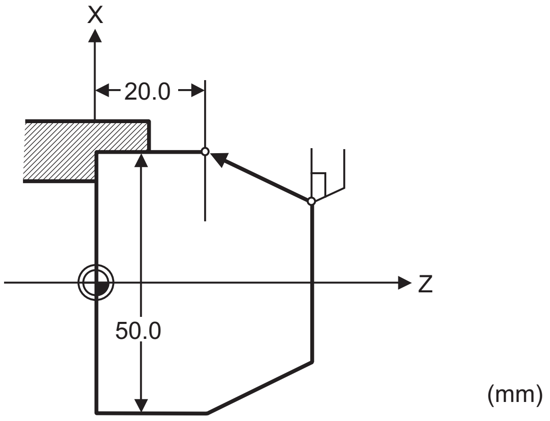
This command commands the in-position width for the linear interpolation command from the machining program.

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.

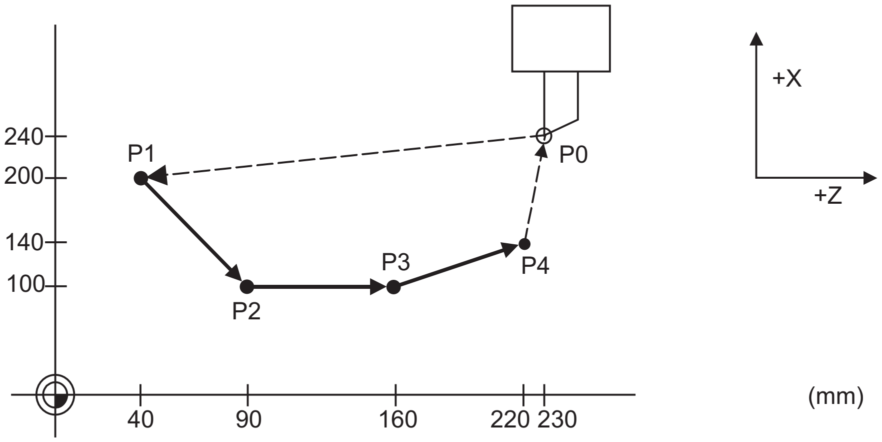
Program example

(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

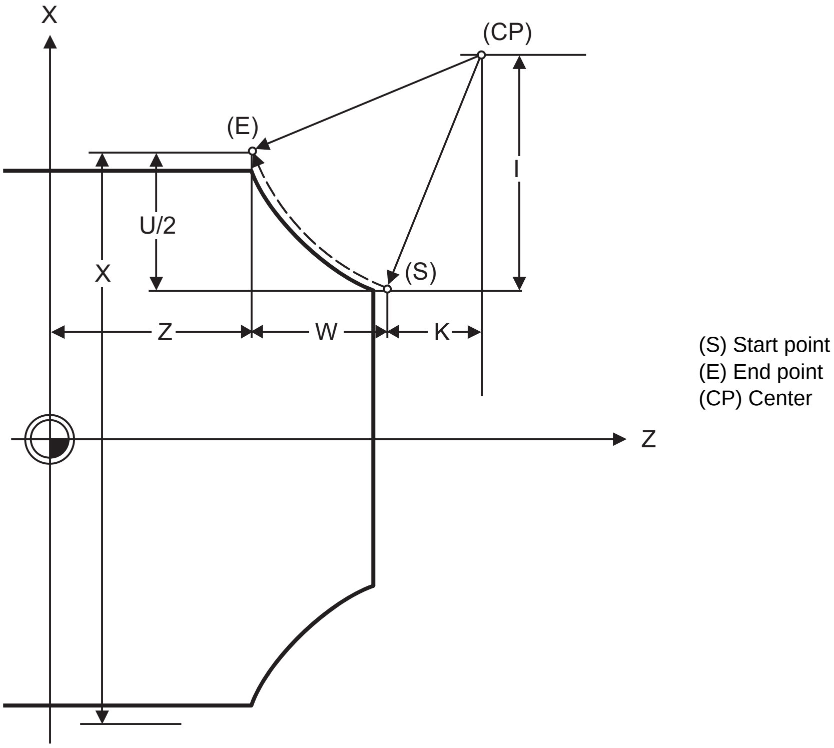
G02 X__/U__ Z__/W__ I__ K__ F__ ; ... Circular interpolation : Clockwise (CW)

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

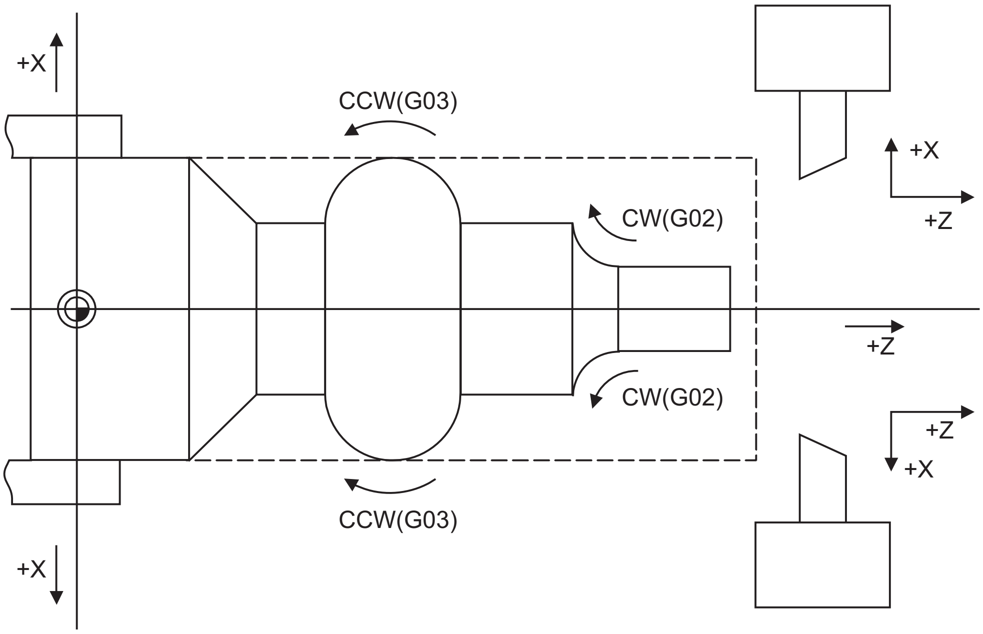
X/U	Circular end point coordinates, X axis (X is the absolute value of workpiece coordinate system, U is the incremental value from present position)
Z/W	Circular end point coordinates, Z axis (Z is the absolute value of workpiece coordinate system, W is the incremental value from present position)
I	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 a program command unit. Caution is required for the circular command of an axis for which the program command unit (#1015 cunit) differs. Command with a decimal point to avoid confusion.



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



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

(4) The following information is needed for circular interpolation.
(a) Rotation direction : Clockwise (G02) or counterclockwise (G03)
(b) Circular end point coordinates : Given by addresses X, Z, U, W
(c) Circular center coordinates : Given by addresses I, K (incremental value commands)
(d) Feedrate : Given by address F

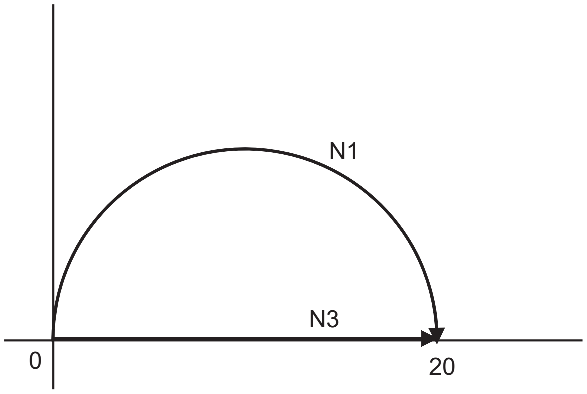
(5) A program error results when I, K or R is not commanded.
Consideration must be given to the sign for I and K since I is the distance in the X-axis direction to the arc center from the start point and K in the Z-axis direction.

(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.

Change into linear interpolation command

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

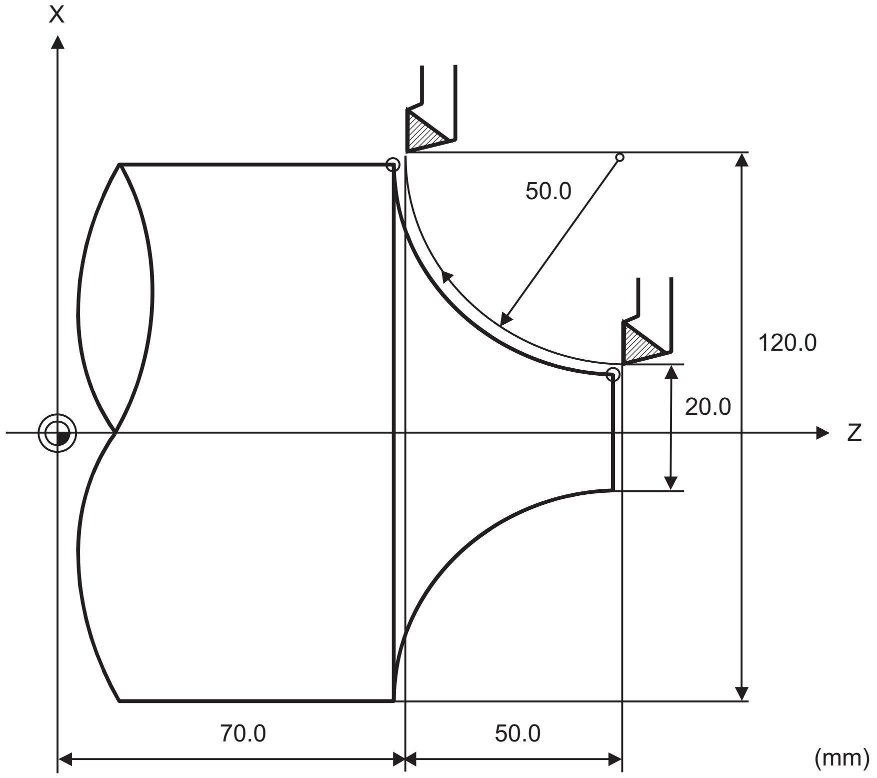
(Example)The parameter "#11029 Arc to G1 no Cent (Change command from arc to linear when no arc center designation)" = "1"



```
G90 X0 Y0 ;
N1 G02 X20. I10. F500 ;     ... (a)
N2 G00 X0 ;
N3 G02 X20. F500 ;          ... (b)
M02 ;
```

(a) The circular interpolation (G02) is executed because there is a center command.
(b) The linear interpolation (G01) is executed because there is no center and radius command.

Program example



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

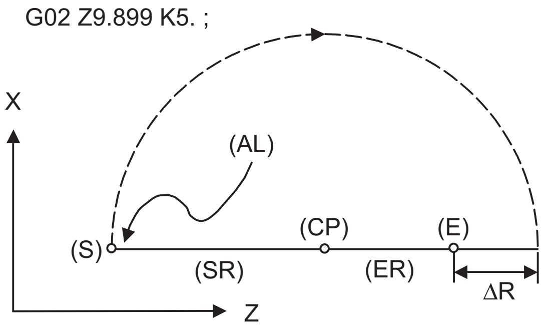
Precautions

(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 :

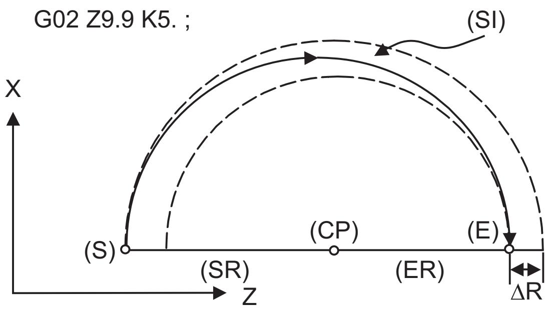
(a) Program error (P70) results at the circular start point when error ΔR is greater than parameter "#1084 RadErr".



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

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

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



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

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

6.4 R Specification Circular Interpolation ; G02,G03

Function and purpose

Along with the conventional circular interpolation commands based on the circular center coordinate (I, 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)

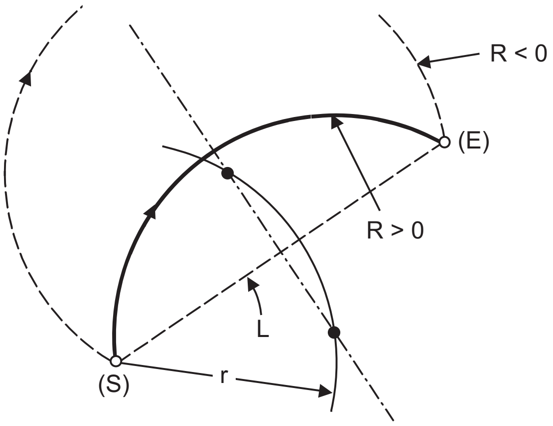
G03 X/U__ Z/W__ R__ F__ ; ... R specification circular interpolation Counterclockwise (CCW)

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 input command unit differs. Command with a decimal point to avoid confusion.

Detailed description

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



(S) Start point
(E) End point

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

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

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

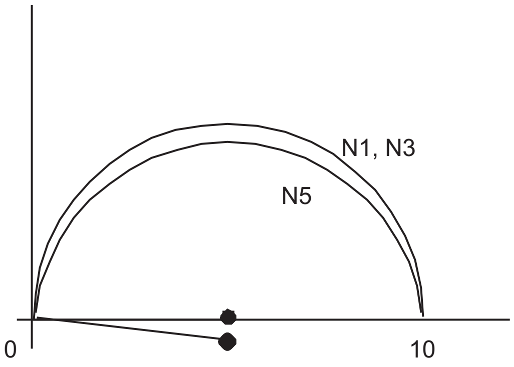
Where L is the line from the start point to the end point. If an R specification and I, K specification are given at the same time in the same block, the circular command with the R specification takes precedence. In the case of a full-circle command (where the start and end points coincide), an R specification 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.

Circular center coordinate compensation

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

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

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



```
G90 X0 Y0 ;
N1 G02 X10. R5.000;
N2 G0 X0;                    ...(a)
N3 G02 X10. R5.001;
N4 G0 X0;                    ...(b)
N5 G02 X10. R5.002;
N6 G0 X0;
M02 ;
```

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

Calculation error margin compensation allowance value: 0.002 mm
Segment connecting the start and end paints: 10.000
N3: Radius × 2 = 10.002 "Error 0.002 -> Compensate"
N5: Radius × 2 = 10.004 "Error 0.004 -> Do not compensate"
Therefore, this example is shown in the above figure.

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

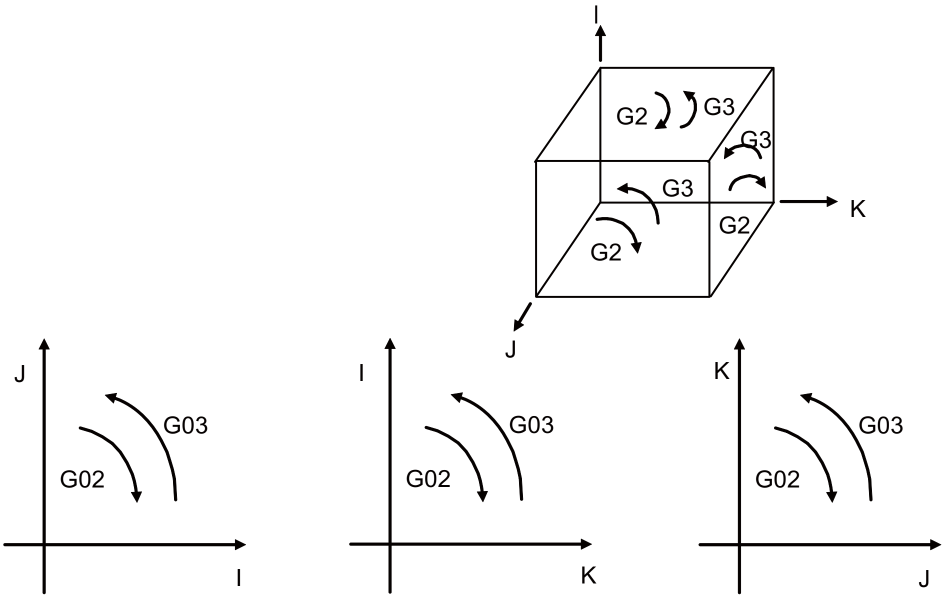
G17; ... I-J plane selection

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.



G17 (I-J)   G18 (K-I)   G19 (J-K)

Parameter entry

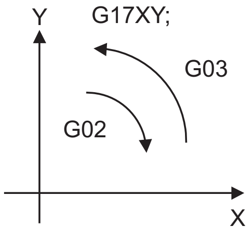
	#1026 to 1028 base_I,J,K	#1029 to 1031 aux_I,J,K	Basic axes and parallel axes can be entered in the parameters. The same axis name can be entered in duplication, but when it is assigned in duplication, the plane is determined by plane selection system (5). The axis which is not registered as the control axis cannot be set.
I	X	Y	
J	Y		
K	Z		

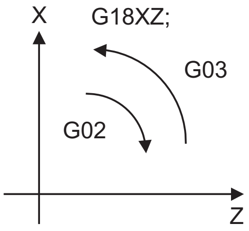
Table 1 Examples of plane selection parameter entry

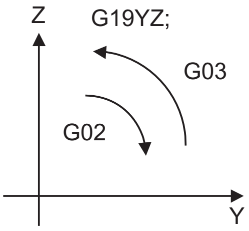
Plane selection system

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

(5) When the basic axes or their parallel axes are duplicated and assigned in the same block as the plane selection G code (G17, G18, G19), the plane is determined in the order of basic axes, and then parallel axes.
G18 XYZ ; The Z-X plane is selected. Therefore, the Y movement is unrelated to the selected plane.

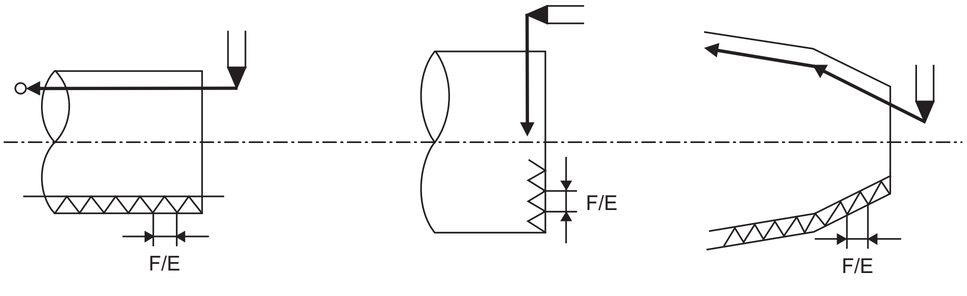
(Note 1) When the "2" in the parameter "#1025 I_plane" is kept ON, the G18 plane is selected when the power is turned ON or when the system is reset.

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.



Straight thread   Scrolled thread   Continuous thread

Command format

Normal lead thread cutting

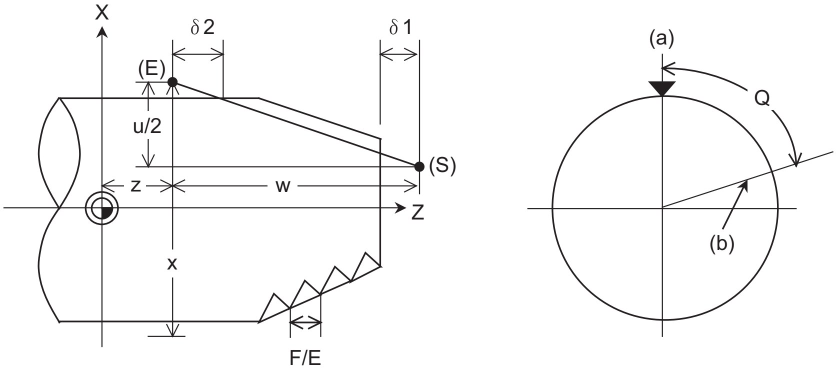
```
G33  Z/W__ X/U__ F__ Q__ ;
```

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°

Precision lead thread cutting

```
G33  Z/W__ X/U__ E__ Q__ ;
```

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°



δ1 > Illegal lead at start of thread cutting

δ2 > Illegal lead at end of thread cutting

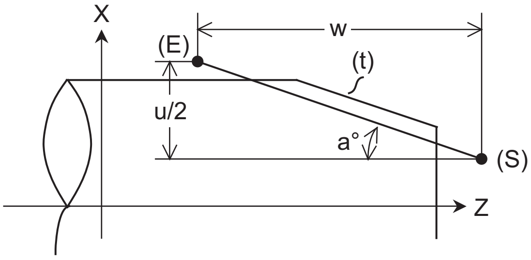
(S) Start point (E) End point

(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 "#8156 Fine thread cut E" is set to "1" for precision lead designation.)

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



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

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

Thread cutting metric input

Input setting unit	**B (0.000mm)**		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Least Command Increments	1(=0.0001) (1.=1.0000)	1(=0.00001) (1.=1.00000)	1(=1.00) (1.=1.00)
Command range	0.0001 - 999.9999	0.00001 - 999.99999	0.03 - 999.99

Input setting unit	**C (0.0001mm)**		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Least Command Increments	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)	1(=1.000) (1.=1.000)
Command range	0.00001 - 999.99999	0.000001 - 999.999999	0.026 - 222807.017

Thread cutting inch input

Input setting unit	**B (0.0001inch)**		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Least Command Increments	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)	1(=1.000) (1.=1.000)
Command range	0.00001 - 39.37007	0.000001 - 39.370078	0.025 - 9999.999

Input setting unit	**C (0.00001inch)**		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Least Command Increments	1(=0.000001) (1.=1.000000)	1(=0.0000001) (1.=1.0000000)	1(=1.0000) (1.=1.0000)
Command range	0.000001 - 39.370078	0.0000001 - 39.3700787	0.0254 - 9999.9999

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

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

(4) The spindle rotation speed should be kept constant throughout from the rough cutting until the finishing.

(5) If the feed hold function is employed during thread cutting to stop the feed, the thread ridges will lose their shape. For this reason, feed hold does not function during thread cutting. Note that this is valid from the time the thread cutting command is executed to the time the axis moves.
If the feed hold switch is pressed during thread cutting, block stop will occur at the end point of the block following the block in which thread cutting is completed (no longer G33 mode).

(6) The converted cutting feedrate is compared with the cutting feed clamp rate when thread cutting starts, and if it is found to exceed the clamp rate, an operation error will occur.

(7) In order to protect the lead during thread cutting, a cutting feedrate which has been converted may sometimes exceed the cutting feed clamp rate.

(8) An illegal lead is normally produced at the start of the thread and at the end of the cutting because of servo system delay and other such factors.
Therefore, it is necessary to command a thread length which is determined by adding the illegal lead lengths δ1 and δ2 to the required thread length.

(9) The spindle rotation speed is subject to the following restriction :
1 <= R <= Maximum feed rate/Thread lead
Where R <= Tolerable speed of encoder (r/min)
R: Spindle rotation speed (r/min)
Thread lead = mm or inches
Maximum feedrate= mm/min or inch/mm (this is subject to the restrictions imposed by the machine specifications.)

(10) A program error (P97) may occur when the result of the expression (9) is R<1 because the thread lead is very large to the highest cutting feedrate.

(11) Dry run is valid for thread cutting but the feedrate based on dry run is not synchronized with the spindle rotation.
The dry run signal is checked at the start of thread cutting and any switching during thread cutting is ignored.

(12) Synchronous feed applies for the thread cutting commands even with an asynchronous feed command (G94).

(13) Spindle override and cutting feed override are invalid and the speeds are fixed to 100% during thread cutting.

(14) When a thread cutting command is programmed during nose R compensation, the compensation is temporarily canceled and the thread cutting is executed.

(15) When the mode is switched to another automatic mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops.

(16) When the mode is switched to the manual mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops. In the case of a single block, the following block which does not contain a thread cutting command (G33 mode is cancelled) is first executed and then the automatic operation stops. Note that automatic operation is stopped until the G33 command axis starts moving.

(17) The thread cutting command waits for the single rotation synchronization signal of the rotary encoder and starts movement.

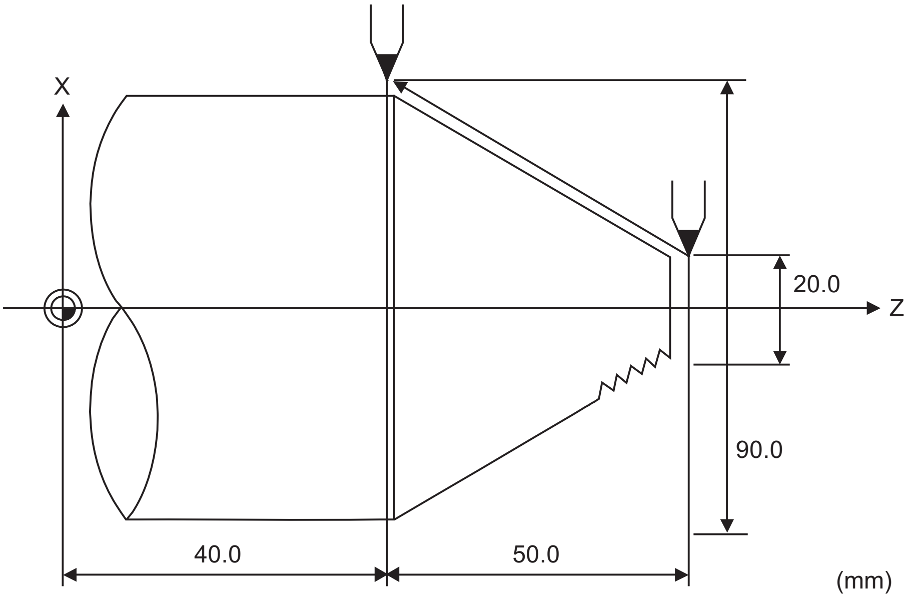
(18) The thread cutting start shift angle is not modal. If there is no Q command with G33, this will be handled as "Q0".

(19) The automatic handle interrupt/interruption is valid during thread cutting.

(20) If a value exceeding 360.000 is command in G33 Q, a program error (P35) will occur.

(21) G33 cuts one row with one cycle. To cut two rows, change the Q value, and issue the same command.

Program example



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

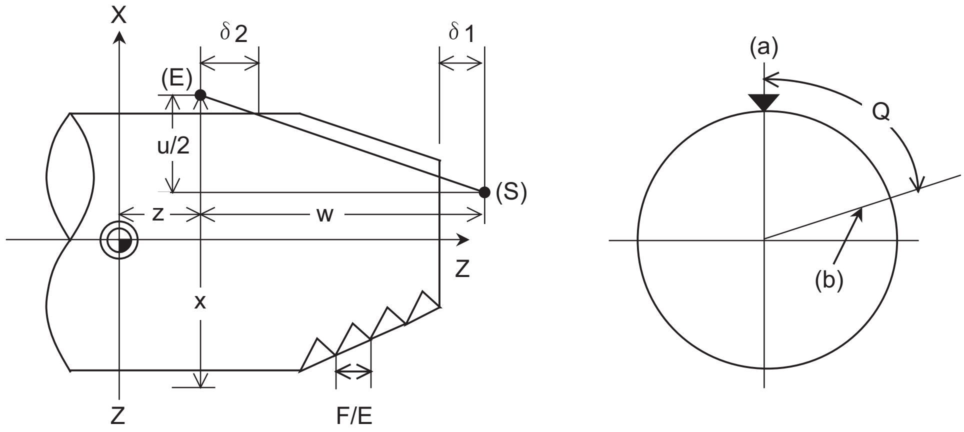
Function and purpose

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 and tapered 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°

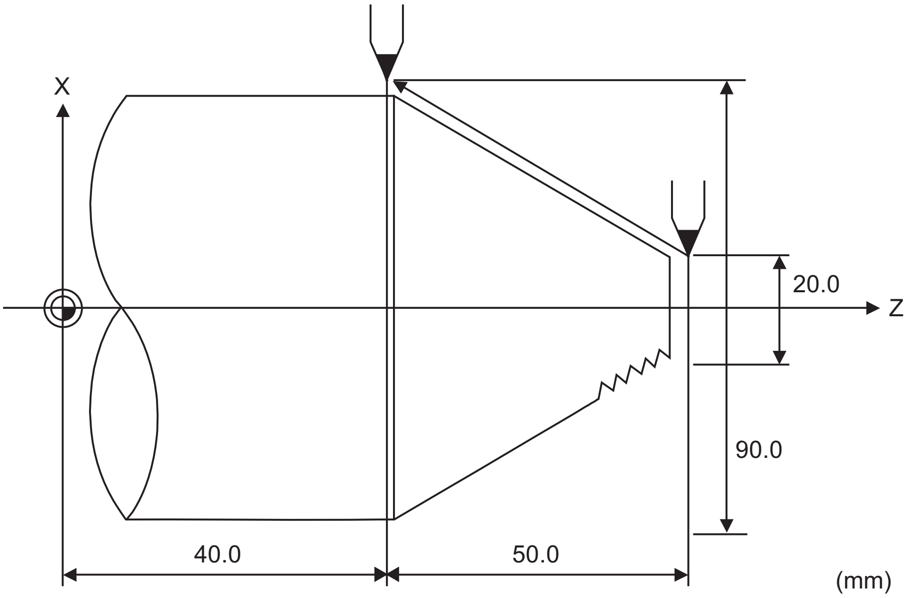


δ1 > Illegal lead at start of thread cutting
δ2 > Illegal lead at end of thread cutting
(S) Start point
(E) End point
(a) One-rotation synchronization signal
(b) Thread cutting start position

Detailed description

(1) The number of ridges in the long axis direction is assigned as the number of ridges per inch.
(2) The E code is also used to assign the precision lead length, and whether the number of ridges or precision lead length is to be designated can be selected by parameter setting. (The number of ridges is designated by setting the parameter "#8156 Fine thread cut E" to "0".)
(3) The E command value should be set within the lead value range when converted to lead.
(4) See Section "Constant lead thread cutting" for other details.

Program example

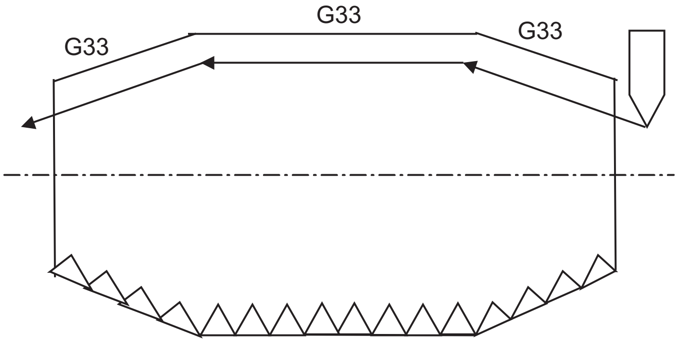


G33 X90.0 Z40.0 E12.0 ;	Absolute value command
G33 U70.0 W-50.0 E12.0 ;	Incremental value command

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.



Command format

G33 Zz1/Ww1 Xx1/Uu1 Ff1/Ee1 Qq1 ; ... Continuous thread cutting

Zzn,Wwn,Xxn,Uun	Thread end point
Ffn/Een	Lead of long axis (axis which moves most) direction
Qqn	Thread cutting start shift angle, 0.001 to 360.000°

Detailed description

(1) The first thread cutting block in the continuous thread cutting command waits for the spindle's single rotation synchronization signal before starting thread cutting. From the second and following blocks, movement starts without waiting for the spindle's single rotation synchronization command.
Thus, the thread cutting start shift angle (Q) can be commanded only in the first block.

(2) G33 command can be omitted from the second and following blocks.

(3) When commanding continuous thread cutting, command the thread cutting commands in successive blocks. If a command other than thread cutting is issued, continuous thread cutting will not take place. Note that if a command that does not involve axis movement (G4 dwell command, MST command, etc.) is commanded between the thread cutting command blocks, whether to wait for the spindle's single rotation synchronization signal after the 2nd block can be selected with the parameters.

# No.	Item	Details	Setting range
1270	ext06/bit6	Set the continuous thread cutting Z phase wait operation. 0: If there is no movement command (MST command, etc.) between the thread cutting blocks, the 2nd block thread cutting waits for the spindle's single rotation synchronization signal before starting movement. 1: Even if there is no movement command (MST command, etc.) between the thread cutting blocks, the 2nd block thread cutting starts movement without waiting for the spindle's single rotation synchronization signal.	0 / 1

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

6.6.4 Variable Lead Thread Cutting ; G34

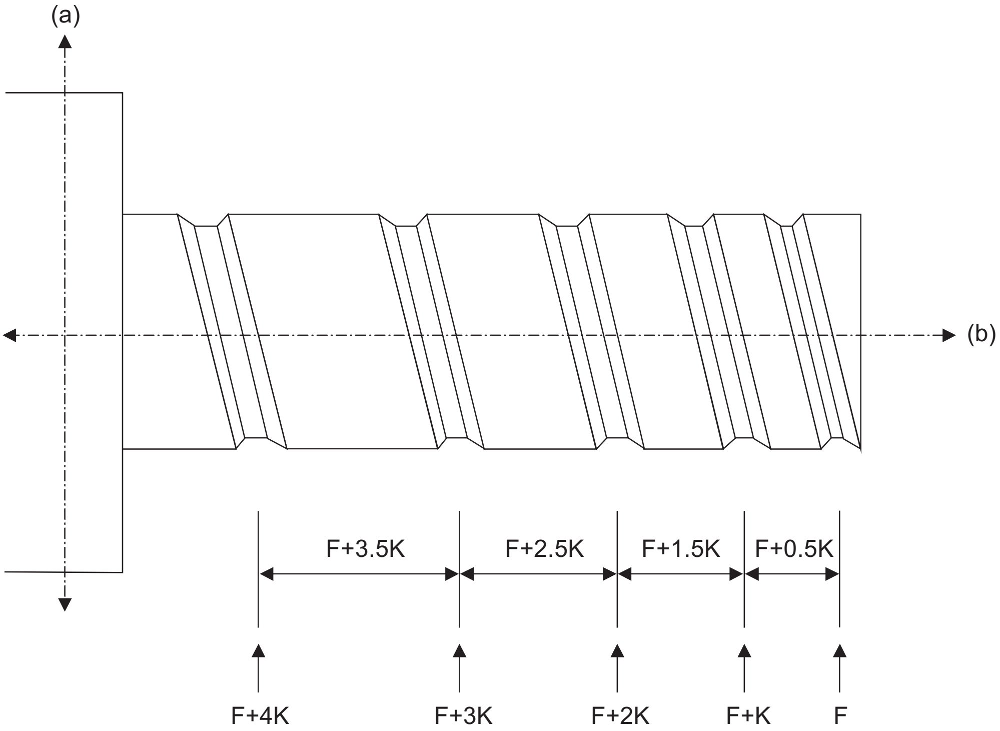
Function and purpose

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 setting unit	B (0.001mm)	
Command address	F (mm/rev)	E (mm/rev)
Least Command Increments	1(=0.0001) (1.=1.0000)	1(=0.00001) (1.=1.00000)
Command range	0.0001 - 999.9999	0.00001 - 999.99999

Input setting unit	C (0.0001mm)	
Command address	F (mm/rev)	E (mm/rev)
Least Command Increments	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)
Command range	0.00001 - 999.99999	0.000001 - 999.999999

Input setting unit	B/C
Command address	K(n*mm/rev) n:Number of pitches Same as F or E (signed)
Least Command Increments	
Command range	

Thread cutting inch input

Input setting unit	B (0.0001inch)	
Command address	F (inch/rev)	E (inch/rev)
Least Command Increments	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)
Command range	0.00001 - 39.37007	0.000001 - 39.370078

Input setting unit	C (0.00001inch)	
Command address	F (inch/rev)	E (inch/rev)
Least Command Increments	1(=0.000001) (1.=1.000000)	1(=0.0000001) (1.=1.0000000)
Command range	0.000001 - 39.370078	0.0000001 - 39.3700787

Input setting unit	B/C
Command address	K(n*inch/rev) n:Number of pitches Same as F or E (signed)
Least Command Increments	
Command range	

(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.

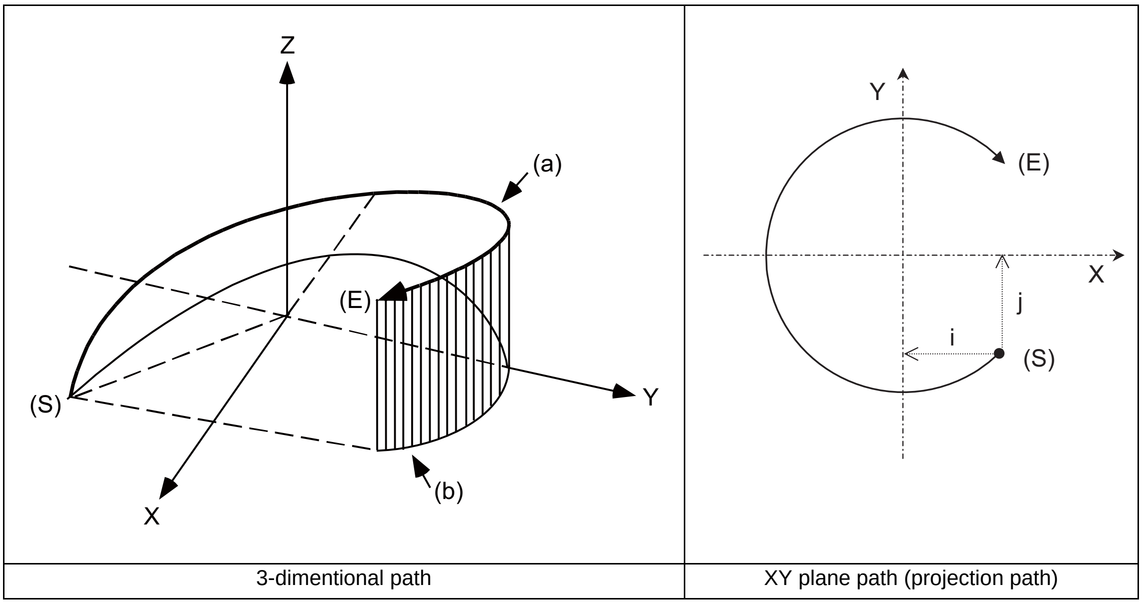
Error No.	Details	Remedy
P 93	Illegal pitch value (1) An invalid value is specified for F/E or K in a thread cutting command. (2) The last lead goes outside of the F/E command range.	The last lead goes outside of the F/E command range. LL : Last lead NP : Number of pitches $LL = \sqrt{(F^2 + 2KZ)}$ $NP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

(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.



3-dimentional path | XY plane path (projection path)

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

Command format

G17/G18/G19 G02/G03 X/U__ Y/V__ Z/W__ I__ J__ F__ (R__ F__); ... Helical interpolation

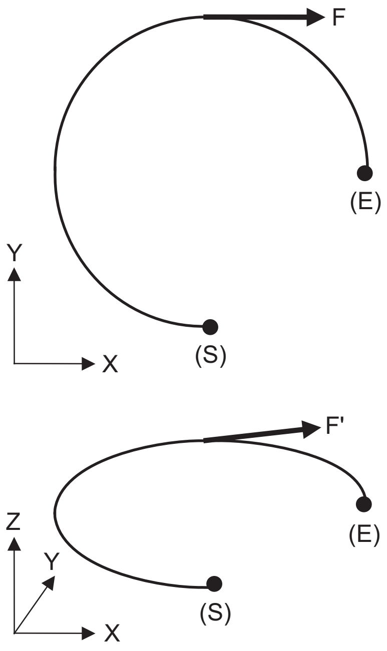
G17/G18/G19	Arc plane (G17: X-Y plane, G18: Z-X plane, G19: Y-Z plane)
G02/G03	Arc rotation direction (G02: clockwise, G03: counterclockwise)
X/U, Y/V	Arc end point coordinates
Z/W	Linear axis end point coordinates
I, J	Arc center coordinates
R	Arc radius
F	Feedrate

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

Detailed description

Speed designation during the helical interpolation

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



(S) Start point
(E) End point

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

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

Arc plane element speed designation selection

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

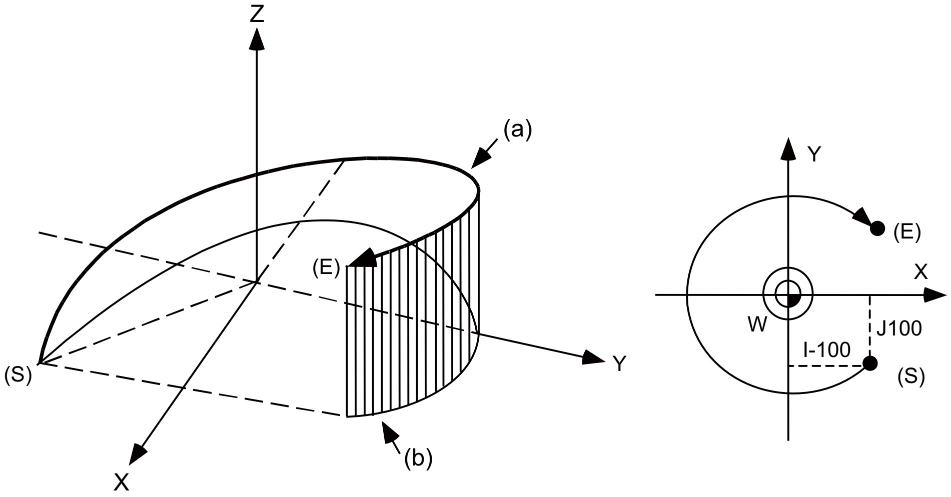
G17 G91 G02 X10. Y10. Z-4. I10. F100 ;	Helical interpolation at speed at which arc plane element is F100
G01 X20. ;	Linear interpolation at F100
G02 X10. Y-10. Z4. J10. ;	Helical interpolation at speed at which arc plane element is F100
G01 Y-40. F120;	Linear interpolation at F120
G02 X-10. Y-10. Z-4.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

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

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

Program example

```
G17 G02 X100. Y100. Z100. I-100. J100. F120 ;
```



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

The left drawing shows the process as an exploded view, and the right drawing shows the arc plane from directly above.
At the start of the block, the axis centers at the point -100mm in the X axis direction and 100mm in the Y axis direction from the workpiece coordinates (start point), and starts cutting at the feedrate 120mm/min while rotating.

Precautions and restrictions

(1) When executing helical interpolation, command another linear axis (several axis can be commanded) that does not contain the circular interpolation command and arc axis.

(2) Up to the number of simultaneous contouring control axes can be commanded simultaneously.

(3) A command exceeding one rotation cannot be issued. (It is complied with the circular interpolation command specifications.)

(4) Command the feedrate as the composite speed for each axis.

(5) With helical interpolation, the axis that configures the plane is the circular interpolation axis, and the other axes are the linear interpolation axes.

(6) The movement of the linear interpolation axis is stopped and only the circular interpolation axis operates during the corner chamfering or corner R commands.

(7) Refer to the circular interpolation (G02,G03) for other precautions.

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 10000000 mm/min. The upper limit is subject to the restrictions limited by the machine specifications. Refer to the specifications manual of the machine for the rapid traverse rate settings. 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 Feed per minute (asynchronous feed)
Metric input

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	

Inch input

Feedrate		
G1 X100. Z100. F200 ;	2.0inch/min	
G1 X100. Z100. F200. ;	200.0inch/min	
G1 X100. Z100. F123.4 ;	123.4inch/min	
G1 X100. Z100. F56.789 ;	56.789inch/min	

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

Command mode	Feedrate command range	Remarks
mm/min	0.001 to 10,000,000	
inch/min	0.0001 to 1000000	
° /min	0.001 to 10,000,000	

(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, but the acceleration/deceleration method follows the rapid traverse setting.)
When F1 to F5 is assigned, the feedrate set to correspond to the command serves as the command value.

Detailed description

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

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

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

K : Operation constant
(This is the number of FM divisions, and is the calculated constant of the increment/decrement speed per scale of the manual handle pulse generator.)
This is set with the base specification parameter "#1507 F1_K".

FM : This is the clamp speed of F1 to F5
This is set with the base specification parameter "#1506 F1_FM".

P : The number of manual handle pulses

Set the corresponding speed of F1 to F5 with the base specification parameters "#1185 spd_F1" to "#1189 spd_F5" respectively.
The increase/reduction range is from "0" to the set value of the parameter "#1506 F1_FM".
The error "M01 operation error 0104" will occur when the feedrate is 0.

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

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

(4) The F1-digit command is valid in a G01, G02, or G03 modal.
(5) The F1-digit command can also be used for fixed cycle.
(6) The number of manual handle pulses is 1 pulse per scale unit regardless of the scaling factor.
(7) During a F1-digit command, the F1-digit number and F1-digit command signal are output as the PLC signals.

F1-digit and G commands

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

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

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

Example of arithmetic constant K setting

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

FM is made 15000 mm/min:

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

Therefore, K is 1500.

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

The conditions for validation of the speed change with the handle

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

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

Conversion of the F1-digit feedrate changing with handle operation

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

Precautions

(1) If F1-digit feedrate changing valid signal is turned ON when F1-digit feed is commanded and changing feedrate with the manual handle is valid and applied to the following conditions, it is clamped at the upper limit of F1-digit feedrate.
F1-digit feedrate > Upper limit of F1-digit feedrate

(2) When the F1-digit feedrate is 0 with the handle operation during the F1-digit feedrate command, the alarm "0104 F1-digit feed rate zero" will occur. This alarm can be eliminated when the F1-digit feedrate is set in larger value than 0 with the handle operation.

(3) When K is 0, the speed is not changed.

(4) F1 to F5 are invalid in the G00 mode and the rapid traverse rate is established instead.

(5) If F0 is used in the G02 or G03 mode, the program error (P121) will occur. The error will be eliminated if the F0 command is rewritten.

(6) When F1. to F5. (with decimal point) are assigned, the 1mm/min to 5mm/min (direct numerical value command) are established instead of the F1-digit feed command.

(7) When the commands are used with inch units, one-tenth of the feedrate set correspond to F1 to F5 serves at the assigned speed inch/min.

(8) When the commands are used with the millimeter or degree units, the feedrate set to correspond to F1 to F5 serves as the assigned speed mm (°)/mm.

(9) Even if the F1-digit feed is commanded during feed per revolution (G95), it is executed as a normal F command (direct numerical value command).

(10) When both the F1-digit feedrate changing and the manual speed command are present, the manual speed command will have the priority.

(11) In the synchronous tapping command, the speed cannot be changed with the handle.

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

Function and purpose

Feed per minute (asynchronous feed)

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

Feed per revolution (synchronous feed)

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

Command format

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

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

Detailed description

G94/G95 commands are modal commands.
(Ex.) After the G95 command is assigned, the G95 command is valid until the G94 command is next assigned.

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

Metric input

Input Setting unit	B(0.001mm)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)
Minimum command unit	1(=1.000) (1.=1.000)	1(=0.0001) (1.=1.00)
Command range	0.001 - 1000000.000	0.0001 - 999.9999

Input Setting unit	C(0.0001mm)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)
Minimum command unit	1(=1.0000) (1.=1.0000)	1(=0.00001) (1.=1.00)
Command range	0.0001 - 1000000.0000	0.00001 - 999.99999

Inch input

Input Setting unit	B(0.0001inch)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)
Minimum command unit	1(=0.0100) (1.=1.0000)	1(=0.000001) (1.=1.000)
Command range	0.0001 - 100000.0000	0.00001 - 99.99999

Input Setting unit	C(0.00001inch)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)
Minimum command unit	1(=0.01000) (1.=1.00000)	1(=0.0000001) (1.=1.000)
Command range	0.00001 - 100000.00000	0.000001 - 99.999999

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

FC = F × N × OVR ...... (Formula 1)
FC : Effective rate (mm/min, inch/min)
F : Commanded feedrate (mm/rev, inch/rev)
N : Spindle rotation speed (r/min)
OVR: Cutting feed override

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

Precautions

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

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

(3) If the spindle rotation speed is zero when feed per revolution is executed, the error "M01 operation error 0105" occurs.

(4) Feedrate during the machine lock becomes the command speed.

(5) Under dry run conditions, feed per minute applies and movement results at the manual feedrate (mm/min or inch/min).

(6) 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

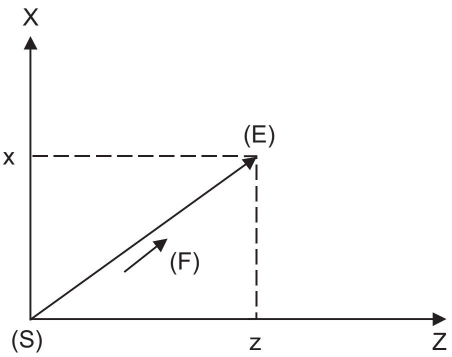
It has already been mentioned that a machine has a number of control axes. These control axes can be divided into linear axes which control linear movement and rotary axes which control rotary movement. The feedrate is designed to assign the displacement speed of these axes, and the effect exerted on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



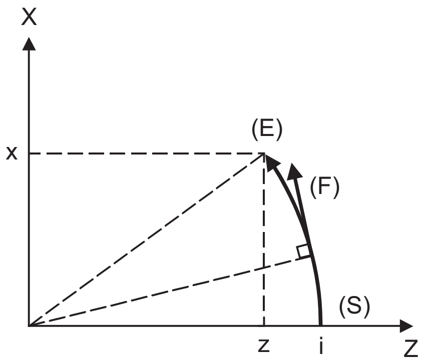
$$fx = f \times \frac{x}{\sqrt{x^2 + z^2}}$$

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

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



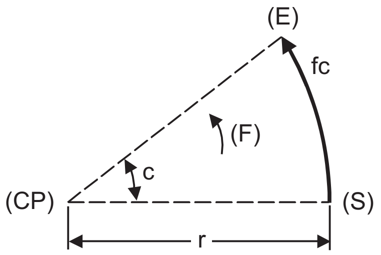
(S) Tool start point
(E) Tool end point
(F) Speed in this direction is "f".

(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" :

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

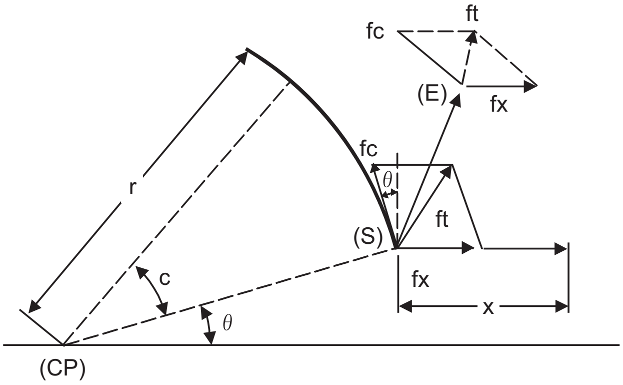
Therefore, the feedrate to be designated in the program must be :

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

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

The controller proceeds in exactly the same way whether linear or rotary axes are to be controlled.
When a rotary axis is to be controlled, the numerical value assigned by the coordinate word (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 values is "c":



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

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

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

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

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

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

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

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

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

...... (4)

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

...... (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:

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

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

...... (6)

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

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

...... (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 setting unit	**B(0.001mm)**		
Command Address	F(mm/rev)	F(mm/rev)	E(ridges/inch)
Minimum command unit	1(=0.0001) (1.=1.0000)	1(=0.00001) (1.=1.00000)	1(=1.00) (1.=1.00)
Command range	0.0001 - 999.9999	0.00001 - 999.99999	0.03 - 999.99

Input setting unit	**C(0.0001mm)**		
Command Address	F(mm/rev)	E(mm/rev)	E(ridges/inch)
Minimum command unit	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)	1(=1.000) (1.=1.000)
Command range	0.00001 - 999.99999	0.000001 - 999.999999	0.026 - 222807.017

Thread cutting inch input

Input setting unit	**B(0.0001inch)**		
Command Address	F(inch/rev)	E(inch/rev)	E(ridges/inch)
Minimum command unit	1(=0.00001) (1.=1.00000)	1(=0.000001) (1.=1.000000)	1(=1.000) (1.=1.000)
Command range	0.00001 - 39.37007	0.000001 - 39.370078	0.025 - 9999.999

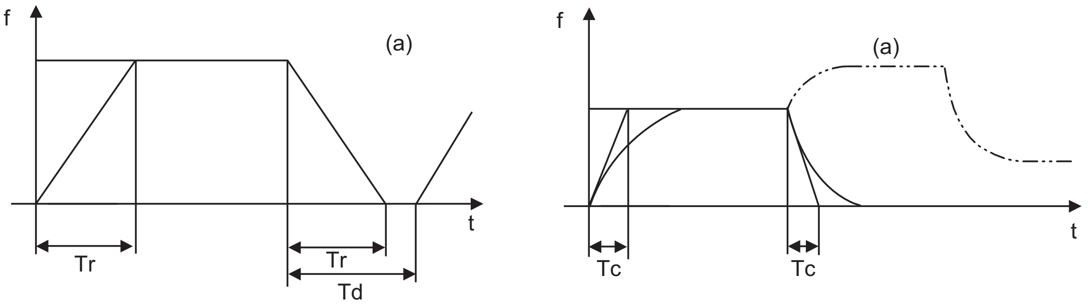
Input setting unit	**C(0.00001inch)**		
Command Address	F(inch/rev)	E(inch/rev)	E(ridges/inch)
Minimum command unit	1(=0.000001) (1.=1.000000)	1(=0.0000001) (1.=1.0000000)	1(=1.0000) (1.=1.0000)
Command range	0.000001 - 39.370078	0.0000001 - 39.3700787	0.0254 - 9999.9999

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 unit 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 unit 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

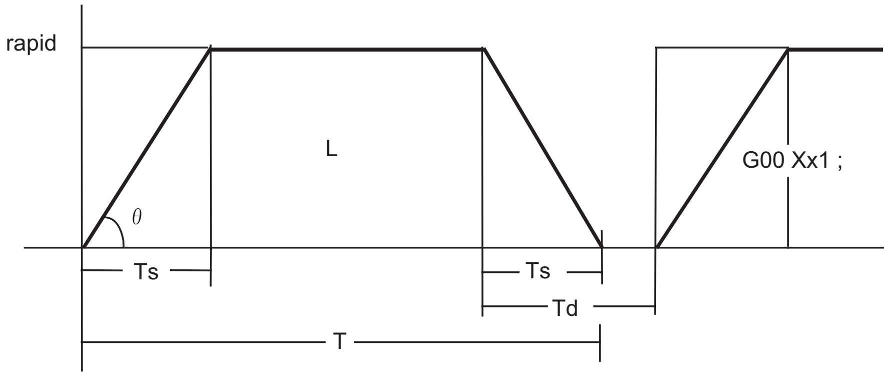
This function performs acceleration and deceleration at a constant inclination during linear acceleration/ deceleration in the rapid traverse mode. Compared to the method of acceleration /deceleration after interpolation, the constant inclination acceleration/deceleration method makes for improved cycle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



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

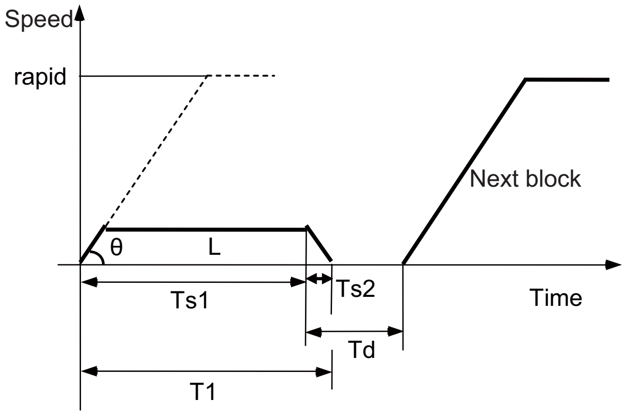
$$Td = Ts + (0 \sim 14\text{ms})$$

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

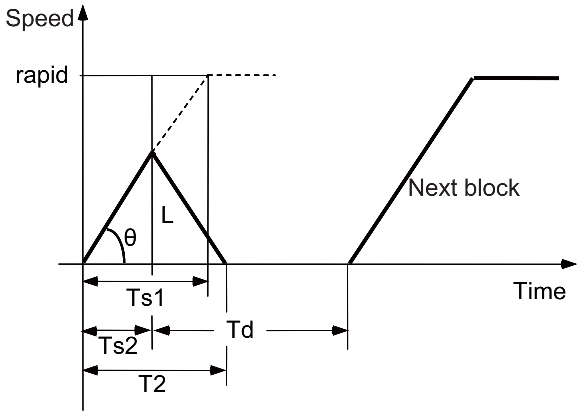
rapid : Rapid traverse rate
Ts : Acceleration/deceleration time constant
Td : Command deceleration check time
θ : Acceleration/deceleration inclination
T : Interpolation time
L : Interpolation distance

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

In case of time-constant acceleration/deceleration:



In case of inclination-constant acceleration/deceleration:



$$T1 = Ts1 + Ts2$$

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

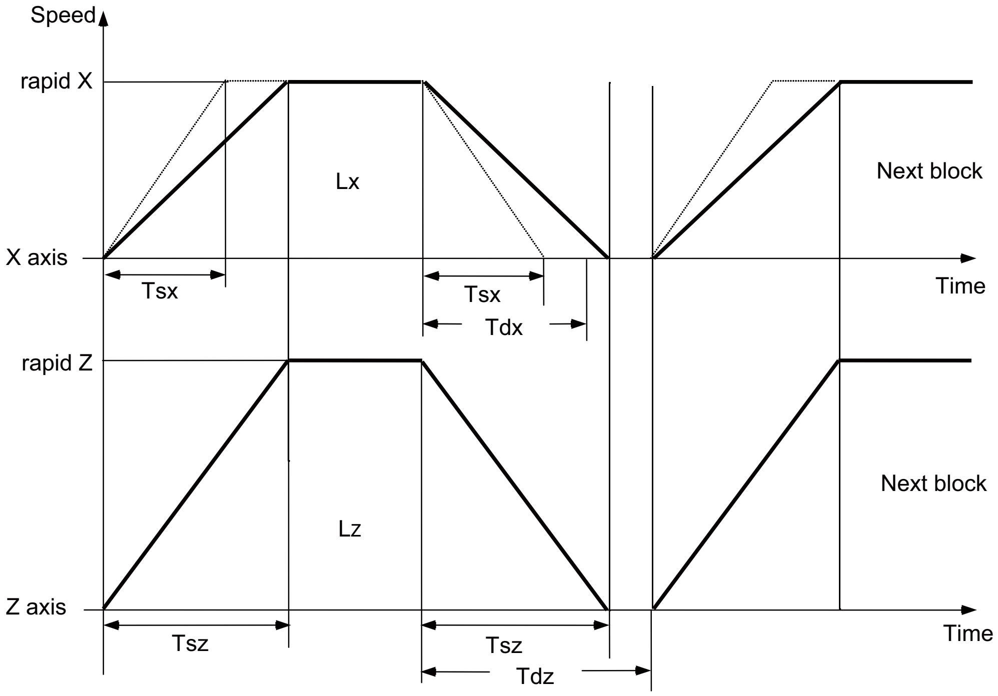
$$Td = T2/2 + (0 \text{ to } 14 \text{ ms})$$

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

rapid : Rapid traverse rate (Axis specification parameter #2001 rapid)
Ts1 : Acceleration/deceleration time (Axis specification parameter #2004 G0tL)
Ts2 : Acceleration/deceleration time to reach the maximum speed
Td : Command deceleration check time
θ : Acceleration/deceleration inclination
T1 : Interpolation time (time-constant acceleration/deceleration)
T2 : Interpolation time (inclination-constant acceleration/deceleration)
L : Interpolation distance

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

[2-axis simultaneous interpolation (When linear interpolation is used, Tsx < Tsz, Lx ≠ Lz)]

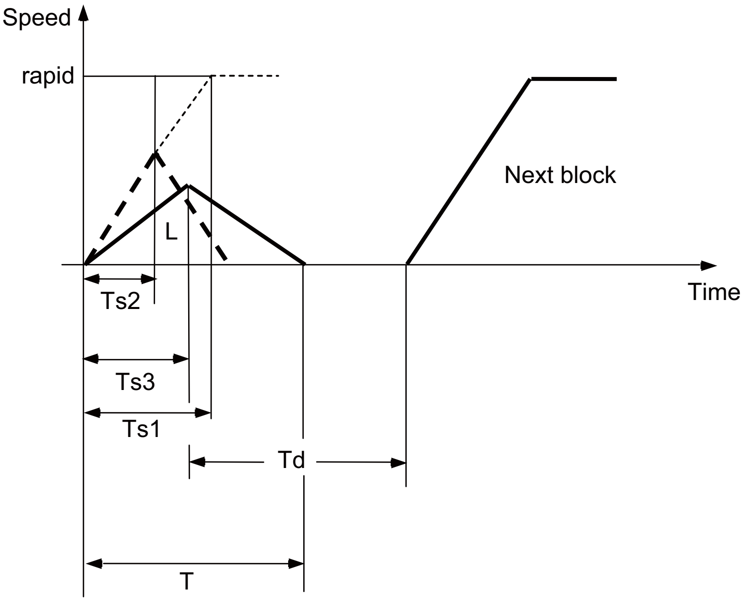


When Tsz is greater than Tsx,
Tdz is also greater than Tdx, and
Td = Tdz in this block.

Tsx : X axis acceleration/deceleration time
Tsz : Z axis acceleration/deceleration time
Tdx : X axis commanded deceleration check time
Tdz : Z axis commanded deceleration check time
Lx : X axis interpolation distance
Lz : Z axis interpolation distance

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

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



$$T = 2 \times Ts2$$

$$Td = T/_2 + (0 \text{ to } 14 \text{ ms})$$

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

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

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

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

(6) The program format of G0 (rapid traverse command) when rapid traverse constant inclination acceleration/deceleration are executed is the same as when this function is invalid (time constant acceleration/deceleration).

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

Precautions

(1) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, and "#1219 aux03 bit7: Time constant setting changeover for soft acceleration/deceleration" is set to "1: Accelerating time is obtained by G0tL+G0t1 (G1tL+G1t1)", acceleration/deceleration speed is adjusted to prevent the sum of the 1st step and 2nd step acceleration/deceleration times from going below the minimum time constant for constant-inclination acceleration/deceleration.

(2) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, if the acceleration/deceleration time is shorter than G0tL (or G1tL), the 2nd step time constant will be reduced by the same rate as the 1st step time constant.

(3) If a commanded travel distance in a block is small, acceleration/deceleration time becomes quite short when the constant-inclination acceleration/deceleration method is enabled. Although this does contribute to reducing the cycle time, this can also be a cause of machine vibrations. In such case set the minimum time constant for constant-inclination acceleration/deceleration in parameter "#2198 G0tMin". By this setting, the acceleration/deceleration speed is controlled so that the acceleration/deceleration time should not fall below the preset value.

7.9 Cutting Feed Constant Inclination Acceleration/Deceleration

Function and purpose

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

Detailed description

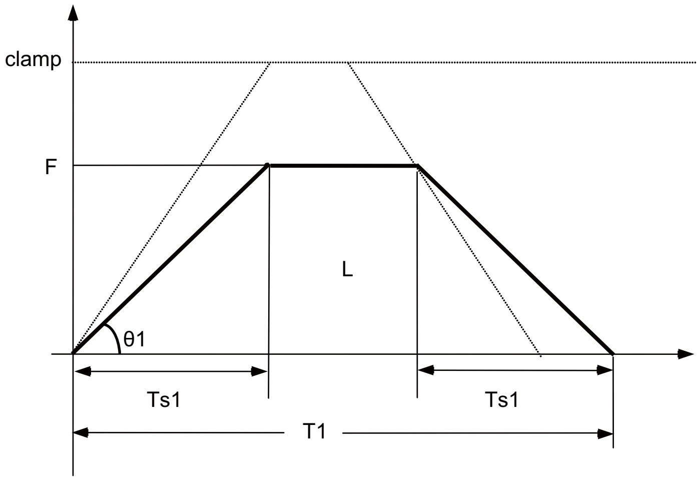
(1) Cutting feed constant inclination acceleration/deceleration function is effective only when the commanded cutting feed acceleration/deceleration mode is linear method or soft method in a linear interpolation command.

(2) The program format of linear interpolation when cutting feed constant inclination acceleration/ deceleration is executed is the same as when this function is invalid (time constant acceleration/ deceleration).

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

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

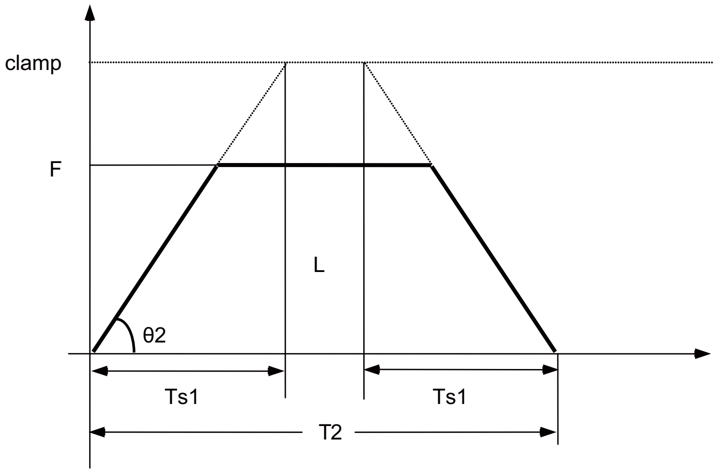
In case of time-constant acceleration/deceleration:



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

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

In case of inclination-constant acceleration/deceleration:



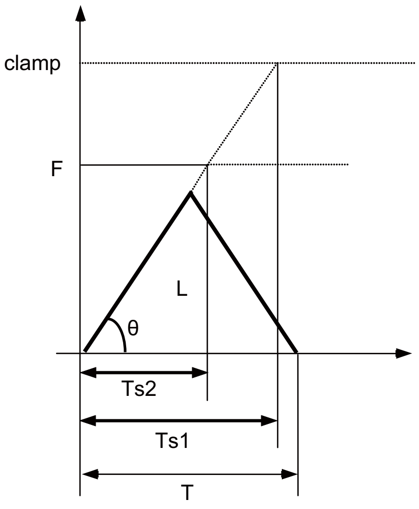
$$T2 = \frac{L}{F} + \frac{Ts1 \times F}{clamp}$$

$$\theta 2 = \tan^{-1}\left(\frac{clamp}{Ts1}\right)$$

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")
F : Cutting feed rate
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")
θ 1 : Acceleration/deceleration inclination (time-constant acceleration/deceleration)
θ 2 : Acceleration/deceleration inclination (inclination-constant acceleration/deceleration)
T1 : Interpolation time (time-constant acceleration/deceleration)
T2 : Interpolation time (inclination-constant acceleration/deceleration)
L : Interpolation distance

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

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

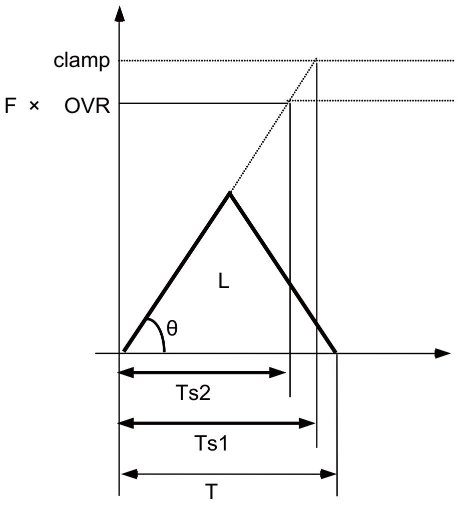


$$T = 2 \times \sqrt{Ts2 \times L / clamp}$$

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

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")
F : Cutting feed rate
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")
Ts2 : Acceleration/deceleration time to reach the cutting feed rate
θ : Acceleration/deceleration inclination
T : Interpolation time
L : Interpolation distance

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



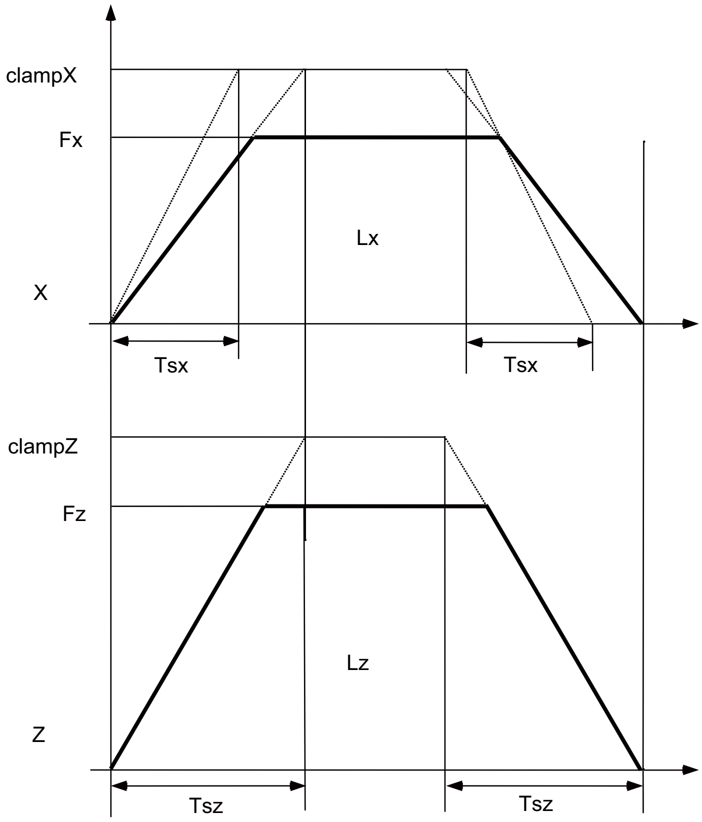
$$T = 2 \times \sqrt{Ts2 \times L \times OVR / clamp}$$

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

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")
F : Cutting feed rate
OVR : Maximum override value for cutting feed inclination-constant acceleration/deceleration (Base Specifications Parameter "#1367 G1AccOVRMax")
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")
Ts2 : Acceleration/deceleration time to reach the cutting feed rate
θ : Acceleration/deceleration inclination

[2-axis simultaneous interpolation (When Tsx < Tsz, Lx ≠ Lz)]

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

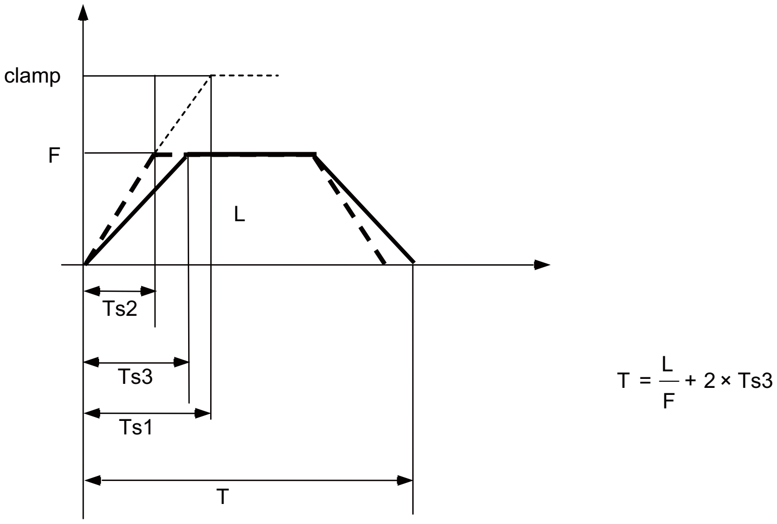


Tsx : X axis acceleration/deceleration time
Tsz : Z axis acceleration/deceleration time
Lx : X axis interpolation distance
Lz : Z axis interpolation distance
Fx : X axis feed rate
Fz : Z axis feed rate

When Tsx < Tsz, the acceleration/deceleration time of the block (Ts) will be equal to the acceleration/ deceleration time Tsz (Z axis acceleration/deceleration time).

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

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

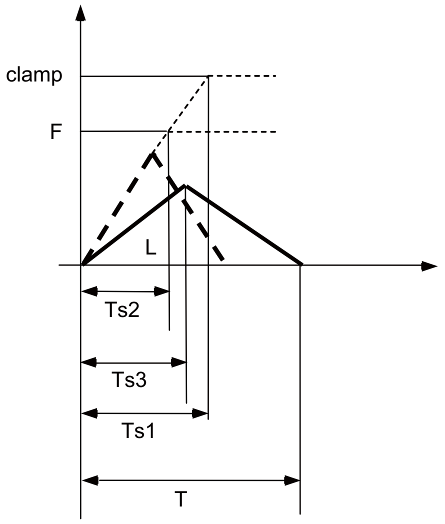


$$T = \frac{L}{F} + 2 \times Ts3$$

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")
F : Cutting feed rate
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")
Ts2 : Acceleration/deceleration time to reach the cutting feed rate
Ts3 : Minimum time for inclination-constant acceleration/deceleration (Axis specification parameter "#2199 G1tMin")
T : Interpolation time
L : Interpolation distance

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

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



T = 2 × Ts3

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")
F : Cutting feed rate
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")
Ts2 : Acceleration/deceleration time to reach the cutting feed rate
Ts3 : Minimum time for inclination-constant acceleration/deceleration (Axis specification parameter "#2199 G1tMin")
T : Interpolation time
L : Interpolation distance

Precautions

(1) If a value greater than 100 (%) is designated for cutting feed override under cutting feed inclination-constant acceleration/deceleration control, the acceleration/deceleration inclination becomes steeper as the feedrate increases. To use the cutting feed override function at a rate higher than 100%, set the parameter "#1367 G1AccOVRMax" accordingly. When the setting of this parameter is between 0 and 99, the override value is handled as 100% even if the specified cutting feed override is over 100%.

(2) If there are one or more NC control axes that are set to soft acceleration/deceleration for G1, the parameter "#1367 G1AccOVRMax" setting will be ignored and the cutting feed override value is handled as 100%.

(3) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, and "#1219 aux03 bit7: Time constant setting changeover for soft acceleration/deceleration" is set to "1: Accelerating time is obtained by G0tL+G0t1 (G1tL+G1t1)", acceleration/deceleration speed is adjusted to prevent the sum of the 1st step and 2nd step acceleration/deceleration times from going below the minimum time constant for inclination-constant acceleration/deceleration.

(4) When "#2003 smgst Acceleration and deceleration modes" is set to the soft acceleration/deceleration, if the acceleration/deceleration is shorter than G0tL (or G1tL), the 2nd step time constant will be reduced by the same rate as the 1st step time constant.

(5) If the commanded travel distance in the block is small or the commanded linear interpolation (G1) feed rate is low, acceleration/deceleration time becomes quite short when the inclination-constant acceleration/deceleration method is enabled. Although this does contribute to reduce the cycle time, this can also be a cause of machine vibrations. Such vibrations can be prevented by setting the minimum time constant for inclination-constant acceleration/deceleration in parameter "#2198 G0tMin". By this setting, the acceleration/deceleration speed is controlled so that the acceleration/deceleration time should not fall below the preset value.

7.10 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 synchronous feed and thread cutting.

7.11 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.
A deceleration check is performed when the G09 (exact stop check) command has been designated in the same block. The G09 command is unmodal.
Either the deceleration check time or in-position state is selected with the parameter. (Refer to the section "Deceleration check")
The in-position width is set into parameter the servo parameter "#2224 sv024" or "#2077 G0inps", "#2078 G1inps".

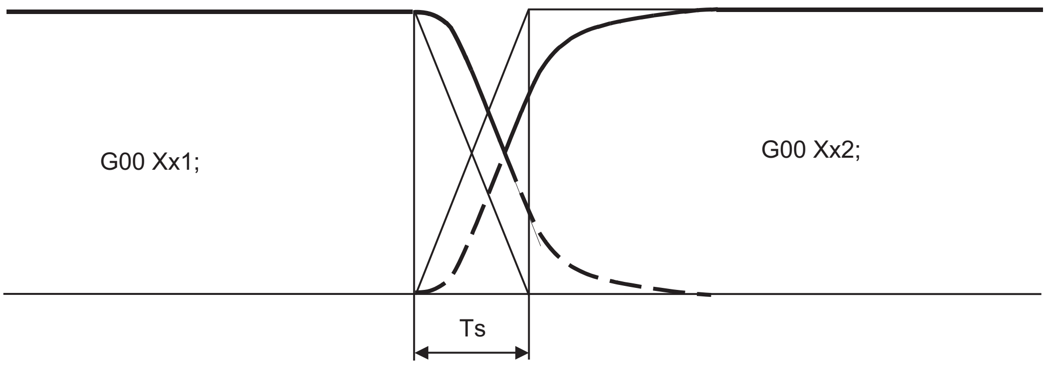
Command format

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

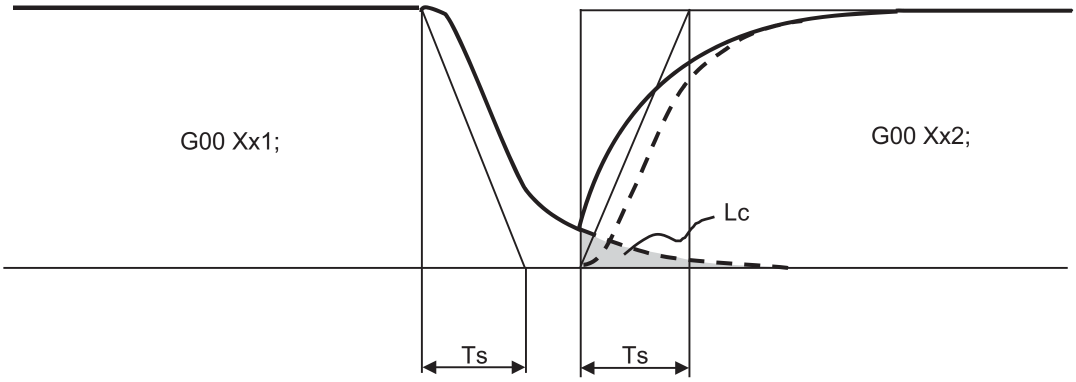
The exact stop check command G09 has an effect only with the cutting command (G01 - G03) in its particular block.

Detailed description

[With continuous cutting feed]



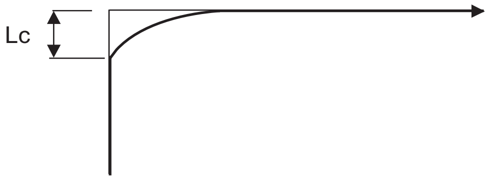
[With cutting feed in-position check]



Ts : Cutting feed acceleration/deceleration time constant

Lc : In-position width

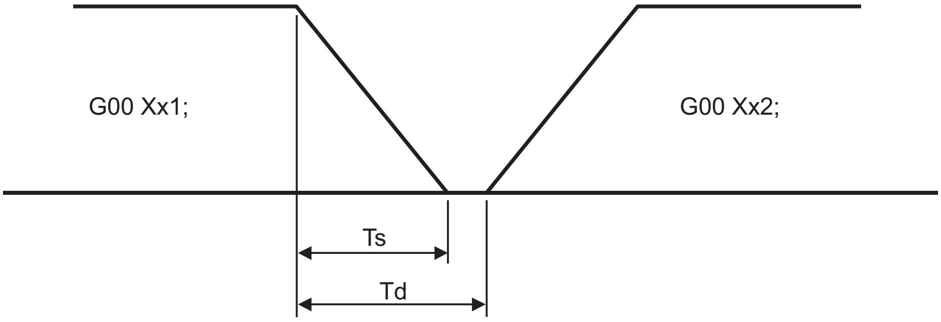
As shown in the figure below, the remaining distance (shaded area in the above figure) of the previous block when the next block is started can be set into the servo parameter "#2224 sv024" as the in-position width "Lc". The in-position width is designed to reduce the roundness at the workpiece corners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



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

With deceleration check

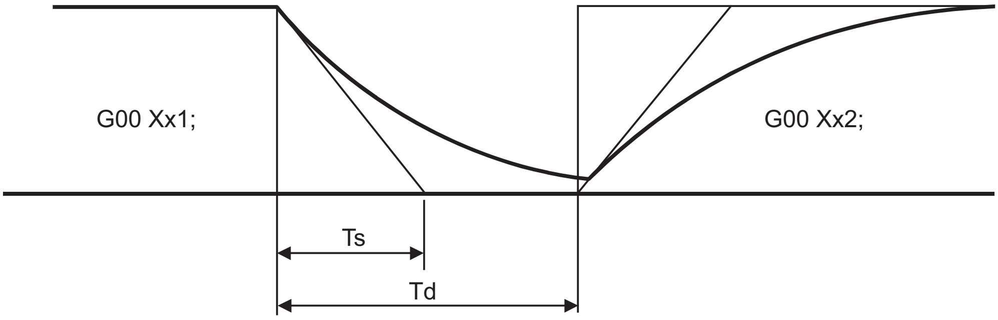
(1) With linear acceleration/deceleration



TS: Acceleration/deceleration time constant

Td: Deceleration check time Td = Ts + (0 to 7ms)

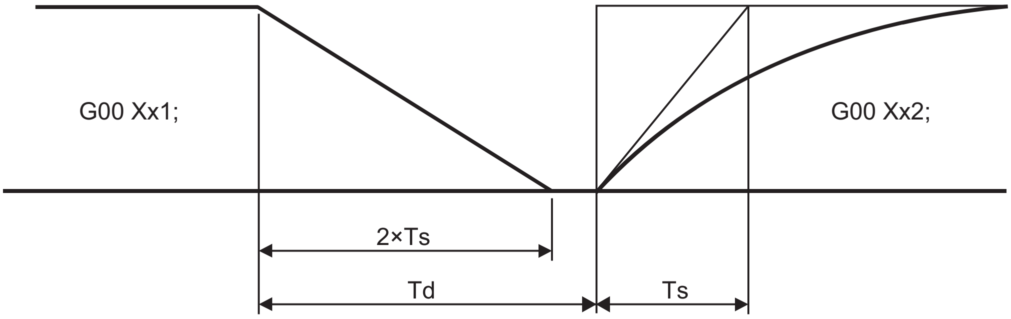
(2) With exponential acceleration/deceleration



TS: Acceleration/deceleration time constant

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

(3) With exponential acceleration/linear deceleration



TS: Acceleration/deceleration time constant

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

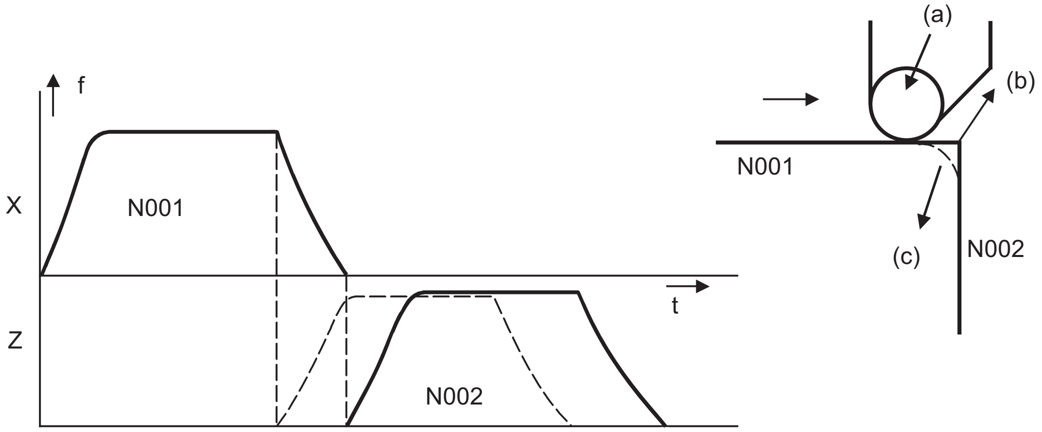
The time required for the deceleration check during cutting feed is the longest among the cutting feed deceleration check times of each axis determined by the cutting feed acceleration/deceleration time constants and by the cutting feed acceleration/ deceleration mode of the axes commanded simultaneously.

(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

7.12 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.

7.13 Deceleration Check

Function and purpose

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

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

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

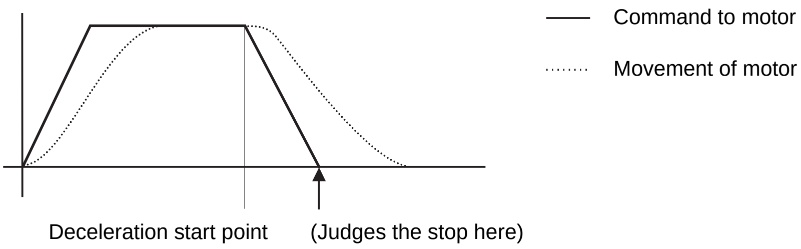
With this function, the deceleration check in the reverse direction of G1 -> G0 or G1 -> G1 can be changed by changing the parameter setting.

Detailed description

Types of deceleration check

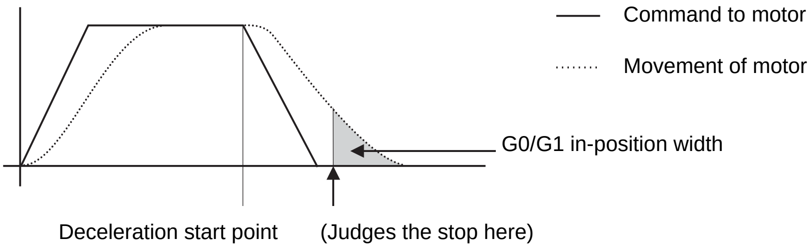
(1) Command speed check

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



(2) In-position check

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



Designating deceleration check

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

(1) Deceleration check specification type 1 ("#1306 InpsTyp" = 0)
The G0 and G1 deceleration check method can be selected with the base specification parameter deceleration check method 1 (#1193 inpos) and deceleration check method 2 (#1223 AUX07/BIT-1).

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

(Note 1) XX expresses all commands.

(2) Deceleration check specification type 2 ("#1306 InpsTyp" = 1)
Rapid traverse and cutting feed in-position are designated with the "#1193 inpos" parameter.

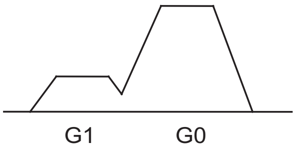
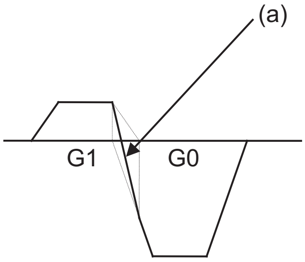
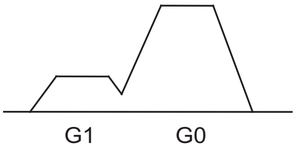
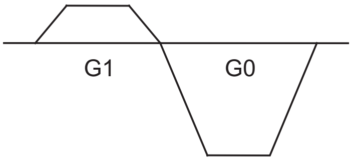
Parameter	Command block		
#1193 inpos	G0	G1+G9	G1
0	Command deceleration check	Command deceleration check	No deceleration check
1	In-position check	In-position check	No deceleration check

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

7.13.1 G1 -> G0 Deceleration Check

Detailed description

In G1 -> G0 continuous blocks, the parameter "#1502 G0Ipfg" can be changed to change the deceleration check in the reverse direction.

	Same direction	Reverse direction
G0Ipfg:0		 (a) The acceleration rate is excessive due to the composite speed of G1 and G0.
G0Ipfg:1		 Command deceleration

Program example

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

(1)

G91 G1 X100. Z100. F4000 ; G0 X-100. Z120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.

(2)

G91 G1 X100. Z-100. F4000 ; G0 X80. Z100. ;	A deceleration check is carried out, because the Z axis moves in the reverse direction in the program.

(3)

G90 G1 X100. Z100. F4000 ; G0 X80. Z100. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program. (When the program start position is X0 Z0)

(4)

G91 G1 X100. Z100. F4000 ; G0 X100. Z100. ;	A deceleration check is not carried out, because both the X axis and the Z axis move in the same direction in the program.

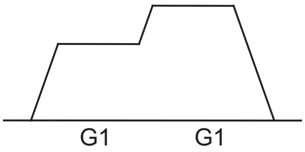
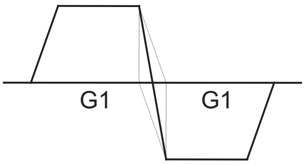
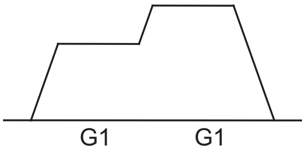
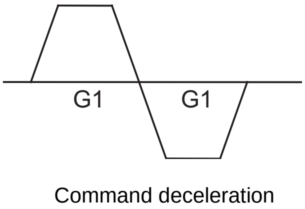
(5)

G91 G1 X100. Z80. F4000 ; G0 X80. ;	A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Z axis movement command in the program.

7.13.2 G1 -> G1 Deceleration Check

Detailed description

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

	Same direction	**Reverse direction**
G1lpfg:0		
G1lpfg:1		

Program example

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

(1)

G91 G1 X100. Z100. F4000 ; G1 X-100. Z120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.

(2)

G91 G1 X100. Z-100. F4000 ; G1 X80. Z100. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.

(3)

G90 G1 X100. Z100. F4000 ; G1 X80. Z120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program. (When the program start position is X0 Z0)

(4)

G91 G1 X100. Z100. F4000 ; G1 X100. Z100. ;	A deceleration check is not carried out, because both the X axis and the Z axis move in the same direction in the program.

(5)

G91 G1 X100. Z80. F4000 ; G1 X80. ;	A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Z axis movement command in the program.

7.14 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.

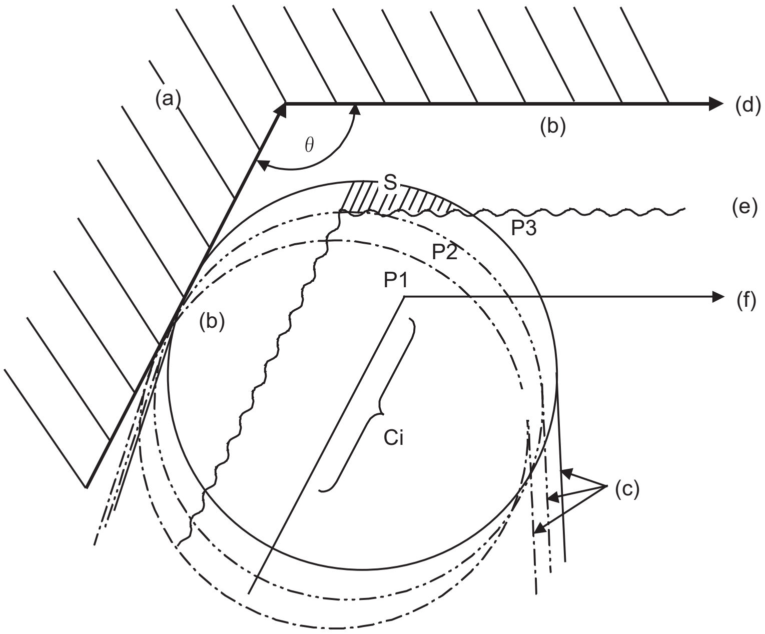
Command format

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.

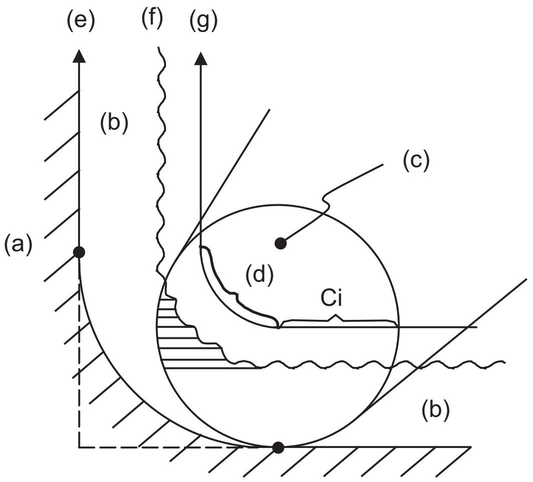
(2) When automatic corner override is to be applied :
When the inside corner angle θ in the above figure is less than the angle set in the parameter, the override set into the parameter is automatically applied in the deceleration range Ci.

[Parameter setting]

The following parameters are set into the machining parameters. Refer to the Instruction Manual for details on the setting method.

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

Automatic corner R

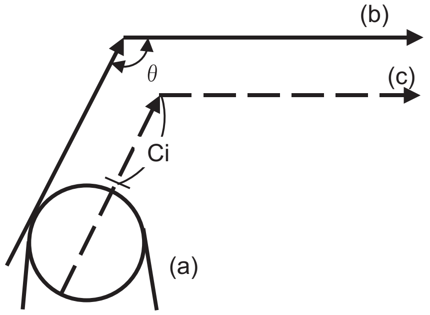


(a) Workpiece	(b) Machining allowance
(c) Corner R center	(d) Corner R section
(e) Programmed path	(f) Workpiece surface shape
(g) Nose R center path	

(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

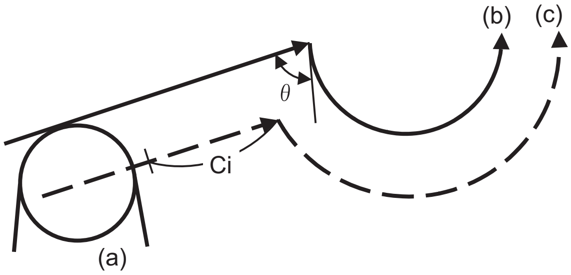
(1) Linear - linear corner



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

The override set in the parameter is applied at Ci.

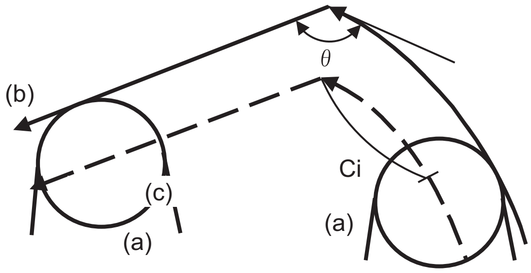
(2) Linear - arc (outside offset) corner



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

The override set in the parameter is applied at Ci.

(3) Arc (inside offset) - linear corner

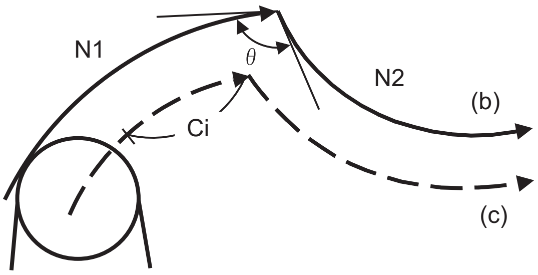


(a) Tool
(b) Program
(c) Nose R center

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 synchronous feedrate.
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

Precautions

(1) Automatic corner override is valid only in the G01, G02, and G03 modes; it is not effective in the G00 mode. When switching from the G00 mode to the G01 (or G02 or G03) mode at a corner (or vice versa), automatic corner override will not be applied at that corner in the G00 block.

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

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

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

(5) Automatic corner override will not be applied when intersection calculation cannot be executed. Intersection calculation cannot be executed in the following case.
-When the movement command block does not continue for four or more times.

(6) The deceleration range with an arc command is the length of the arc.

(7) The inside corner angle, as set by parameter, is the angle on the programmed path.

(8) Automatic corner override will not be applied when the maximum angle in the parameter is set to 0 or 180.

(9) Automatic corner override will not be applied when the override in the parameter is set to 0 or 100.

7.15 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

7.16 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

8

Dwell

8.1 Dwell (Time Designation) ; G04

Function and purpose

The machine movement is temporarily stopped by the program command to make the waiting time state. Therefore, the start of the next block can be delayed. The waiting time state can be canceled by inputting the skip signal.

Command format

G04 X/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 designating the dwell time with P, the availability of the decimal point command can be selected with the parameter (#8112). When the decimal point command is invalid in the parameter setting, the command below the decimal point issued with P is ignored.

(3) When the decimal point command is valid or invalid, the dwell time command range is as follows.

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

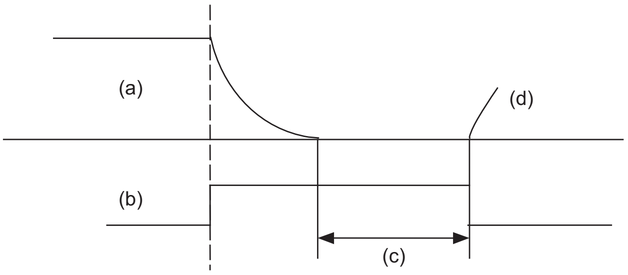
(4) The dwell time setting unit applied when there is no decimal point can be made 1s by setting 1 in the parameter "#1078 Decpt2". This is effect only for X, U and P for which the decimal command is valid.

(5) When a cutting command is in the previous block, the dwell command starts calculating the dwell time after the machine has decelerated and stopped. When it is commanded in the same block as an M, S, T or B command, the calculation starts simultaneously.

(6) The dwell is valid during the interlock.

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

(8) The dwell can be canceled by setting the parameter “#1173 dwlskp” beforehand. If the set skip signal is input during the dwell time, the remaining time is discarded, and the following block will be executed.



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

Program example

Command	Dwell time [s]			
	#1078 Decpt2 = 0		#1078 Decpt2 = 1	
	G04P DECIMAL PNT-N	G04P DECIMAL PNT-P	G04P DECIMAL PNT-N	G04P DECIMAL PNT-P
G04 X500 ;	0.5		500	
G04 X5000 ;	5		5000	
G04 X5. ;	5		5	
G04 X#100 ;	1000		1000	
G04 U500 ;	0.5		500	
G04 U5000 ;	5		5000	
G04 U5. ;	5		5	
G04 U#100 ;	1000		1000	
G04 P5000 ;	5		5	5000
G04 P12.345 ;	0.012	12.345	0.012	12.345
G04 P#100 ;	1	1000	1	1000

(Note 1) The above examples are the results under the following conditions.
- Input setting unit 0.001mm or 0.0001inch
- #100 = 1000 ;

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

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

Precautions and restrictions

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

9

Miscellaneous Functions

9.1 Miscellaneous Functions (M8-digits)

Function and purpose

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

Detailed description

These functions are designated by an 8-digit number (0 to 99999999) following the address M with this controller, and up to 4 groups can be commanded in a single block.
The number of M commands that can be issued within the same block is determined by the parameter setting (#12005 Mfig).
(Example) G00 Xx Mm1 Mm2 Mm3 Mm4 ;

When five or more commands are issued in a block, only the last four will be valid.
Whether to BCD output or binary output the second miscellaneous function can be selected by a parameter.
The eight commands of M00, M01, M02, M30, M96, M97, M98 and M99 are used as auxiliary commands for specific objectives and so they cannot be used as general auxiliary commands. Therefore, 92 miscellaneous functions are available.
Reference should be made to the instructions issued by the machine manufacturer for the actual correspondence between the functions and numerical values.
When the M00, M01, M02, and M30 functions are used, the next block is not read into the pre-read buffer due to pre-read inhibiting.
If the M function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which sequence applies.

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

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

Program stop : M00

When the tape reader has read this function, it stops reading the next block. As far as the NC system's functions are concerned, only the tape reading is stopped. Whether such machine functions as the spindle rotation and coolant supply are stopped or not differs according to the machine in question.
Re-start is enabled by pressing the automatic start button on the machine operation board.
Whether resetting can be initiated by M00 depends on the machine specifications.

Optional stop : M01

If the tape reader reads the M01 command when the optional stop switch on the machine operation board is ON, it will stop and performs the same operation as the M00.
If the optional stop switch is OFF, the M01 command is ignored.

(Example)	The state and operation of optional stop switch
N10 G00 X1000 ;	Stops at N11 when switch is ON
N11 M01 ;	Next command (N12) is executed without stopping at N11 when switch is OFF
N12 G01 X2000 Z3000 F600 ;	
:	

Program end : M02 or M30

This command is normally used in the final block for completing the machining, and so it is primarily used for tape rewinding. Whether the tape is actually rewound or not depends on the machine specifications.
Depending on the machine specifications, the system is reset by the M02 or M30 command upon completion of tape rewinding and any other commands issued in the same block.
(Although the contents of the command position display counter are not cleared by this reset action, the modal commands and compensation amounts are canceled.)
The next operation stops when the rewinding operation is completed (the in-automatic operation lamp goes off). To restart the unit, the automatic start button must be pressed or similar steps must be taken.

(Note 1) Independent signals are also output respectively for the M00, M01, M02 and M30 commands and these outputs are each reset by pressing the reset key.
(Note 2) M02 or M30 can be assigned by manual data input (MDI).
At this time, commands can be issued simultaneously with other commands just as with the tape.

Macro interruption; M96, M97

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

Subprogram call/completion : M98, M99

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

Internal processing with M00/M01/M02/M30 commands

Internal processing suspends pre-reading when the M00, M01, M02 or M30 command has been read. Other tape rewinding operations and the initialization of modals by resetting differ according the machine specifications.

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

Function and purpose

These serve to assign the indexing table positioning and etc. In this controller, they are assigned by an 8-digit number from 0 to 99999999 following address A, B or C. The machine maker determines which codes correspond to which positions.

Detailed description

Select the address A, B or C that is used for the second miscellaneous function by the 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. The number of M commands to be issued within a same block is determined by the parameter setting (#12011 Bfig).
Whether to BCD output or binary output the 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.
- Linear angle commands (,A can be used.)
- Geometric command
- Deep hole drilling cycle 2 commands

10

Spindle Functions

10.1 Spindle Functions

Function and purpose

These functions are assigned with an 8-digit (0 to 99999999) number following the address S, and one group can be assigned in one block.
The output signal is a 32-bit binary data with sign and start signal.
Processing and completion sequences are required for all S commands.

10.2 Constant Surface Speed Control ; G96,G97

Function and purpose

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

Command format

G96 S__ P__ ; ... Constant surface speed ON

S	Surface speed
P	Constant surface speed control axis

G97 ; ... Constant surface speed cancel

Detailed description

(1) The constant surface speed control axis is set by parameter "#1181 G96_ax".
0: Fixed at 1st axis (P command invalid)
1: 1st axis
2: 2nd axis
3: 3rd axis

(2) When the above-mentioned parameter is not zero, the constant surface speed control axis can be assigned by address P.
(Example) G96_ax = 1

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

(3) Example of selection program and operation

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

(4) The spindle subject to control is determined with the following.
For multiple-spindle control I (#1300 ext36 bit0 = 0), the spindle is determined by the spindle selection command in the G group 20.
For multiple-spindle control II (#1300 ext36 bit0 = 1), the spindle is determined by the spindle selection signal (SWS) from the PLC. .

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
Q	Minimum clamp rotation speed

Detailed description

(1) Besides this command, parameters can be used to set the rotation speed range up to 4 stages in 1 r/min units to accommodate gear selection between the spindle and spindle motor. The lowest upper limit and highest lower limit are valid among the rotation speed ranges based on the parameters and based on "G92 S_ Q_ ;".

(2) Set in the parameters "#1146 Sclamp" and "#1227 aux11/bit5" whether to carry out rotation speed clamp only in the constant surface speed mode or even when the constant surface speed is canceled.

(Note 1) G92S command and rotation speed clamp operation

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

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

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

Precautions

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

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

⚠ WARNING

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

10.4 Spindle/C Axis Control

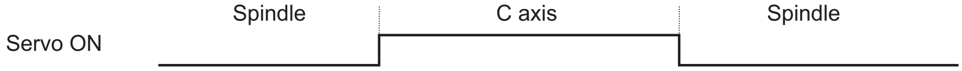
Function and purpose

This function enables a spindle (MDS-D or later) 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.

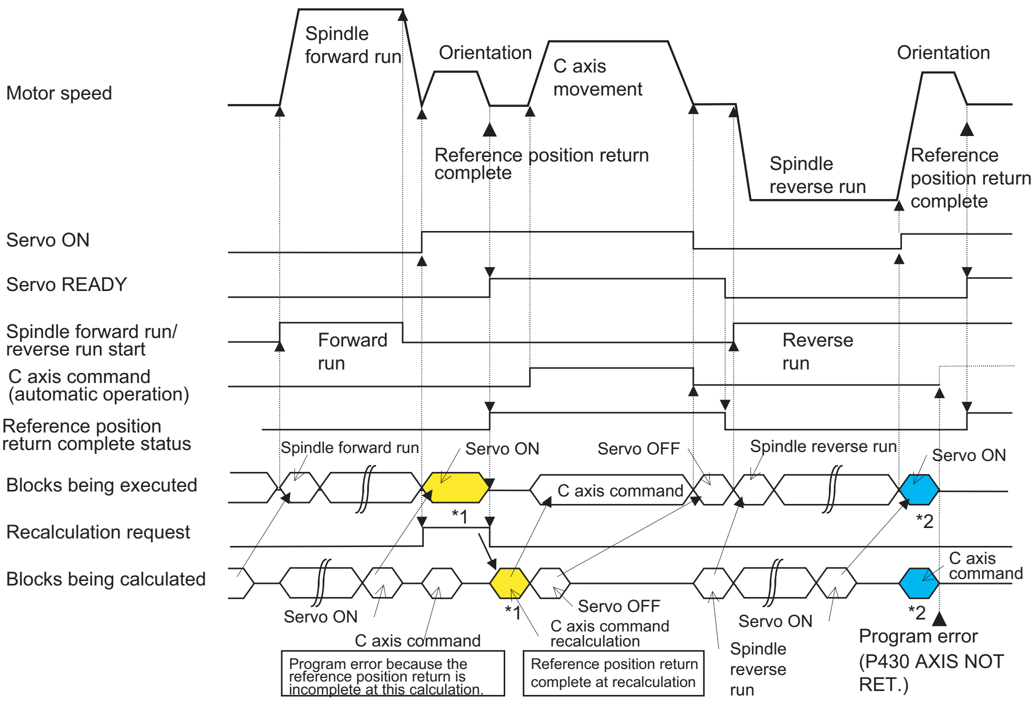


At servo OFF ...... Spindle (C axis control not possible)
At servo ON ...... C axis (spindle control not possible)
The C axis is in a reference position return incomplete state.

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

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

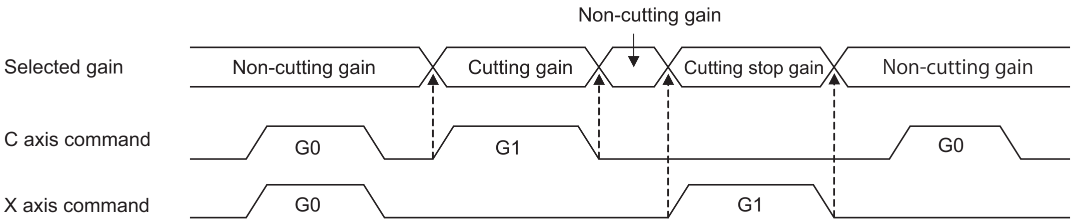
Changeover timing chart example


Motor speed
Spindle forward run
Orientation
C axis movement
Orientation
Reference position return complete
Spindle reverse run
Reference position return complete
Servo ON
Servo READY
Spindle forward run/ reverse run start
Forward run
Reverse run
C axis command (automatic operation)
Reference position return complete status
Spindle forward run
Servo ON
Servo OFF
Spindle reverse run
Servo ON
Blocks being executed
C axis command
*1
*2
Recalculation request
Blocks being calculated
C axis command
Servo ON
*1
Servo OFF
C axis command recalculation
Servo ON
*2
C axis command
Spindle reverse run
Program error (P430 AXIS NOT RET.)
Program error because the reference position return is incomplete at this calculation.
Reference position return complete at recalculation

(Note 1) For axis commands, the reference position return complete is checked at calculation. Thus, when the C axis servo ON command and C axis command are continuous, the program error (P430) will occur as shown above in *2.
In response to this kind of situation, the following two processes must be carried out on user PLC, as shown above in *1.
- Input the recalculation request signal with a servo ON command.
- Wait for the completion of the servo ON command until the C axis enters a servo READY state.

C axis gain

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



(Note 1) The cutting feed of other part systems does not affect the C axis gain selection.
(Note 2) There are 1st to 3rd cutting gains, which are selected with the ladder.

Deceleration check in movement including spindle/C-axis

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

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

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

(Note 2) XX expresses all commands.

Precautions and restrictions

(1) The program error (P430) will occur if a C axis command is issued during servo OFF or during orientation.

(2) Do not execute a servo OFF during a C axis command.
The remaining C axis commands will be cleared at servo ON.
(If servo OFF is executed during C axis control, the feed will stop and spindle control will occur.)

(3) If servo ON is executed during spindle rotation, the rotation will stop and C axis control will occur.

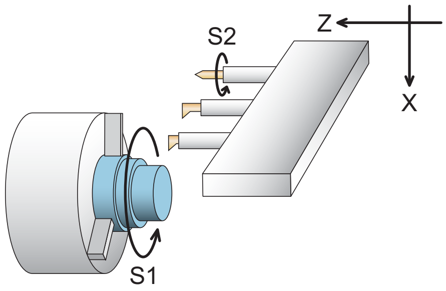
(4) Dog-type reference position return is not possible for the C axis.
Set the reference position return to the orientation method in the parameters (Spindle base specifications parameters "#3106 zrn_typ/bit8"= 0), or set the axis to "Axis without reference position (zero point)" (Zero point return parameters "#2031 noref: 1").

10.5 Multiple-spindle Control

Function and purpose

Multiple spindle control is a function used to control the 2nd spindle in a machine tool that has 2 spindles. The spindle control method differs between multiple-spindle control I and multiple-spindle control II. Whether to use multiple-spindle control I or multiple-spindle control II depends on the parameter (#1300 ext36/bit0) setting.

Multiple-spindle control I: (ext36/bit0 = 0)	Control following the spindle selection command (G43.1, etc.) and spindle control command ([S*****;] or [S ○ =*****;]), etc.
Multiple-spindle control II: (ext36/bit0 = 1)	Control following the external signal (spindle command selection signal, spindle selection signal) and spindle control command ([S***** ;] only), etc. The spindle selection command [S ○ =*****;] cannot be used.



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

Function and purpose

In addition to the S***** command, the 1st and 2nd spindles can be identified and commanded with the S ○ = ***** command.

Command format

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

○	Designate the spindle No. with one numeric character (1: 1st spindle, 2: 2nd spindle). A variable can be designated.
*****	Rotation speed or constant surface speed command. A variable can be designated.

(Note 1) A program error (P35) will occur if the ○ value is not 1 to 2.

(Note 2) A program error (P33) will occur if G47.1 is issued in a modal.

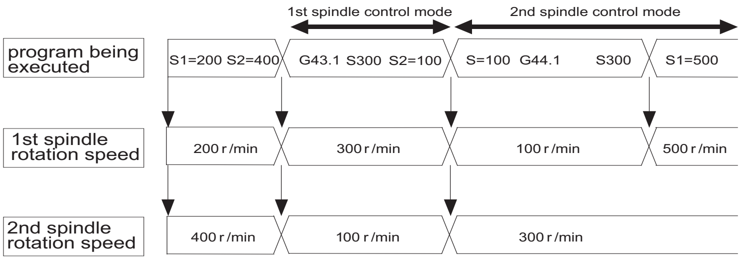
Detailed description

(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



10.5.2 Multiple-spindle Control I (spindle selection command) ; G43.1,G44.1,G47.1

Function and purpose

The target of following S commands (S****) can be switched between the 1st spindle and the 2nd spindle by issuing the spindle selection command (G43.1, etc. [G group 20]).

Command format

G43.1 ; ... 1st spindle control mode ON

G44.1 ; ... Selected spindle control mode ON (Selected spindle No. is set with SnG44.1).

G47.1 ; ... All spindles simultaneous control mode ON

Detailed description

(1) The selected spindle No. is set with the parameter (#1534 SnG44.1).

(2) The spindle selection commands are modal G codes.

(3) A program error (P33) will occur if the spindle selection command is issued while the multiple-spindle control II mode is valid.

(4) The spindle control mode entered when the power is turned ON or reset can be set with the parameters. The state when the power is turned ON or reset is as follows.

G group 20 modal state	Set with setup parameter "#1199 Sselect". 0: G43.1 1: G44.1 2: G47.1
G44.1 spindle No.	Set with setup parameter "#1534 SnG44.1". 0: 2nd spindle 1: 1st spindle 2: 2nd spindle

(5) If the S command is issued in the same block as the spindle selection command, it will be valid for the spindle switched to with the spindle selection command.

(Example)		
G43.1 S100 ;	->	Command 100 rotations for 1st spindle.
S200 G44.1 S300 ; (SnG44.1 = 2)	->	After commanding 200 rotations for 1st spindle, command 300 rotations for 2nd spindle.

(6) If a spindle that does not exist is set, the 2nd spindle will be used. Note that if there is only one spindle, the 1st spindle will be used.

Relation with other functions

The following functions change after the spindle selection command.

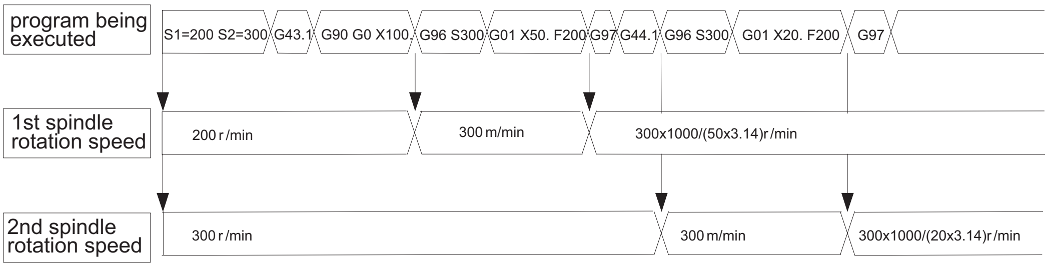
(1) S command (S*****)
The S command in G97 (rotation speed command)/G96 (constant surface speed command) will be the command for the spindle designated with the spindle selection command.

(2) Spindle clamp speed command
The spindle clamp speed command issued with G92 S_Q_ will also follow the spindle selection command mode.

(3) Per rotation command (synchronous feed)
Even if F is commanded in the G95 mode, the per rotation feedrate for the spindle designated with the spindle selection command will be applied.

(4) Constant surface speed control spindle changeover
The constant surface speed control will also follow the spindle selection command mode.
If spindle other than the current mode is commanded with S ○ =*****, the spindle rotation speed command designated with ○ will have the priority.
(Example)
S2 = **** in G43 system mode .... 2nd spindle rotation speed command
S1 = **** in G44 system 2nd spindle selection mode .... 1st spindle rotation speed command
Note that the constant surface speed command in the constant surface speed control (G96) will be modal information independent from the rotation speed command.

When G44.1 spindle No. is 2



10.5.3 Multiple-spindle Control II

Function and purpose

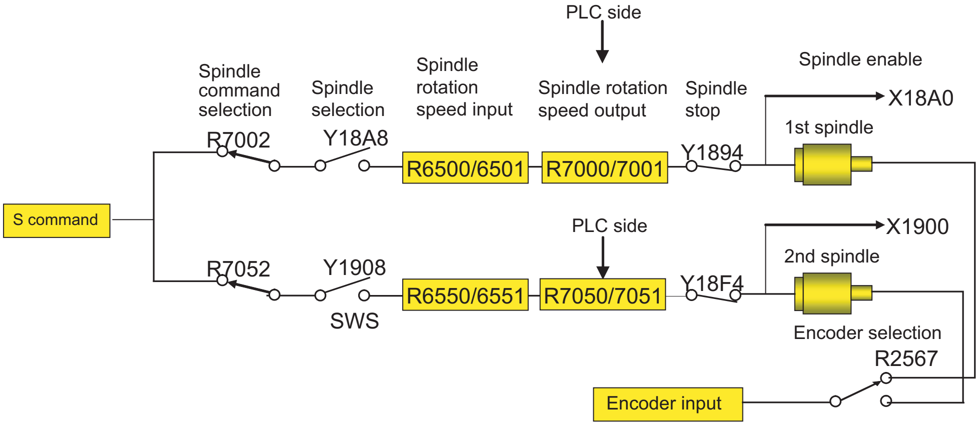
Multiple-spindle control II is a function that designates which spindle to select with the signals from PLC. The command is issued to the spindle with one S command.

Detailed description

(1) Spindle command selection, spindle selection

The S command to the spindle is output as the rotation speed command to the selected spindle when the spindle selection signal (SWS) from the PLC turns ON. The selected spindle rotates at the output rotation speed. The spindle whose selection is canceled when the spindle selection signal (SWS) turns OFF maintains the speed at which it was rotating at before being canceled. This allows each axis to be simultaneously rotated at differing rotation speeds.

The spindle command selection signal is used to determine which part system each spindle receives the S command from.



(Note) Refer to the PLC Interface Manual for details on each signal.

Relation with other functions

(1) Spindle clamp speed setting (G92)
This is valid only on the spindle selected with the spindle selection signal (SWS).
The spindle not selected with the spindle selection signal (SWS) maintains the speed at which it was rotating at before being canceled.
(The spindle clamp speed is maintained with the G92 command.)

(2) Constant surface speed control
Constant surface speed control can be applied on all spindles.
The spindle rotation speed is automatically controlled during constant surface speed control, so when machining with constant surface speed, the spindle selection signal (SWS) for that spindle must be left ON.
The spindle not selected with the spindle selection signal (SWS) maintains the speed at which it was rotating at before being canceled.

(3) Thread cutting/synchronous feed
The threads are cut with the spindle selected with the spindle selection signal (SWS). The encoder feedback selected with the encoder selection signal is used.

(4) Synchronous tapping
The synchronous tapping spindle is selected with the spindle selection signal (SWS).
Select the synchronous tapping spindle before issuing the synchronous tapping command. Do not change the synchronous tapping spindle selection signal during the synchronous tapping mode.
If a C axis mode command is issued to the synchronous tapping spindle, the "M01 operation error 1026" will occur. When the C axis command is canceled, the error will be canceled and machining will resume.

(5) Asynchronous tapping
The asynchronous tapping spindle is selected with the spindle selection signal (SWS).
Select the asynchronous tapping spindle before issuing the tapping command. Input a calculation request to change the asynchronous tapping spindle selection. Do not change the asynchronous tapping spindle selection signal during the asynchronous tapping mode.

(6) Tapping retract
The tapping retract spindle is selected with the spindle selection signal (SWS).
Select the spindle for which the tapping cycle execution is stopped before turning the tapping retract signal ON. If tapping retract is executed when a different spindle is selected, the "M01 operation error 1032" will occur. Do not change the spindle selection signal during tapping retract.

Restrictions

(1) The S manual numerical value command is invalid when multiple-spindle control II is valid.

(2) Setup parameter "#1199 Sselect" is invalid when multiple-spindle control II is valid.

(3) The spindle control mode changeover G code cannot be used when multiple-spindle control II is valid. A program error (P34) will occur.

(4) The "S1=***" and "S2=***" commands are invalid when multiple-spindle control II is valid. The program error (P33) will occur.

(5) The spindle gear shift command output signal (GR1/GR2) is not output when multiple-spindle control II is valid.

11

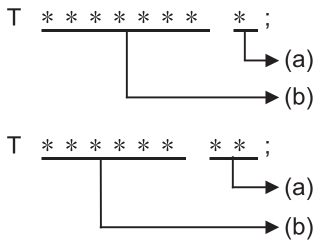
Tool Functions

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 T1digt". 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



(a) Tool compensation No.
(b) Tool No.

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.

12

Tool Compensation Functions

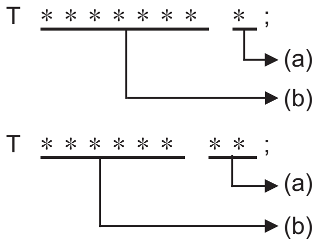
12.1 Tool Compensation

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. And there are two ways to issue the commands: to designate both compensations by the last 1 or 2 digits of the T command or to designate the tool nose wear compensation by the last 1 or 2 digits of the T command and the tool length compensation by the tool No. Parameter "#1098 TLno." is used to switch between them. Parameter "#1097 T1digt" is also used to switch between the last 1 or 2 digits for the compensation.
One set of T commands can be issued in one block.

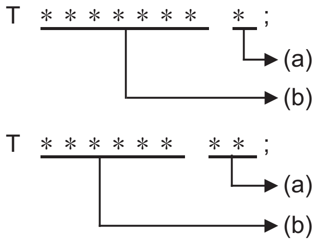
Command format

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



(a) Tool length compensation + tool nose wear compensation tool No.
(b) Tool No.

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



(a) Tool nose wear compensation
(b) Tool No. + tool length compensation

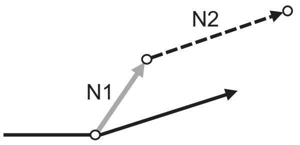
The lower two digits of the tool No. are the tool length compensation No.

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



```
N1 T0101 ;
N2 X100. Z200. ;
```

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.

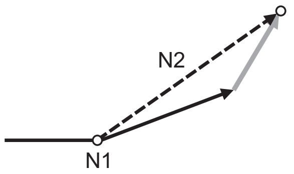
(Note 2) When performing compensation with T command execution, the path is compensated as a linear movement in a circular modal.

(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
G11 : Parameter input by program cancel
G92 : Coordinate system setting
G92.1 : Workpiece coordinate system pre-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).

(Note 5) If G53 command is issued in the same block as the T command, the compensation amount for the axis with no movement commanded will employ the amount in the previous block.

(2) Compensation with movement command



```
N1 T0101 ;
N2 X100. Z200. ;
```

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.)

12.1.2 Expanded Method at Starting Tool Compensation

Function and purpose

By setting the parameter "#1100 Tmove", the compensation operation when T is commanded is selected whether that is carried out when the T command is executed or carried out with superimposed on the movement command.
Also, depending on the setting of the parameter, it is possible to compensate only the wear amount when the T command is executed and to compensate the tool length by superimposing on the movement command.

Detailed description

To compensate only the wear amount when the T command is executed and to compensate the tool length by superimposing on the movement command, set the "#1100 Tmove" to 2.
When 2 is set, the length compensation will follow the setting 1 (superimposed on movement command), and the wear compensation will follow the setting 0 (move when T is commanded).
The compensation operation for each "#1100 Tmove" setting is as shown below.

Length compensation amount: 7.000mm Wear compensation amount: 0.500mm

Machining program example	**Machine value**		
	#1100 Tmove = 0	#1100 Tmove = 1	#1100 Tmove = 2
G28 X.; G0 T116 ; G0 X100. ; :	0.000 7.500 107.500	0.000 0.000 107.500	0.000 0.500 107.500

(Note 1) When performing wear 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
G11 : Parameter input by program cancel
G92 : Coordinate system setting
G92.1 : Workpiece coordinate system pre-setting

(Note 2) After the compensation amount is temporarily canceled, the compensation amount will return for each movement command axis.
The movement at this time will follow the parameter "#1101 Tabsmv" for both length compensation and wear compensation.
G command involving temporary cancel: G28, G30, G53

(Note 3) If G53 command is issued in the same block as the T command, the compensation amount for the axis with no movement commanded will employ the amount in the previous block.

(1) Relation with "#1101 Tabsmv" (tool compensation amount superimposition during absolute command)
When "#1100 Tmove" is set to 2, the compensation operation will follow the "#1101 Tabsmv" setting.

(2) Parameters

# No.	Item	Details	Setting range
1100	Tmove	Designate the timing to perform tool length compensation and wear compensation. 0: Compensates when T command is executed. 1: Superimposes and compensates with the movement command in the block where the T command is located. If there is no movement command in the same block, compensation will be executed after the movement command is superimposed in the next movement command block. 2: Compensates wear amount when the T command is executed. Superimposes and compensates a tool length compensation amount with the movement command in the same block. If there is no movement command in the same block, compensation will be executed after the movement command is superimposed in the next movement command block.	0 to 2
1101	Tabsmv	Designate the type of movement command superimposed when Tmove is set to 1 or 2. 0: Compensates regardless of the movement command type. 1: Compensates only at the movement command in the absolute value command.	0 / 1

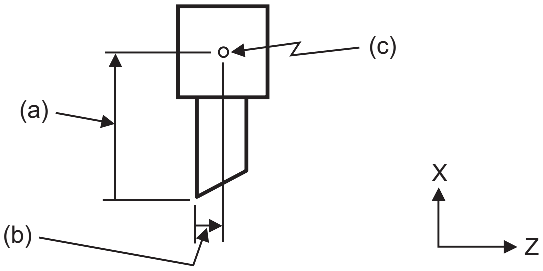
12.2 Tool Length Compensation

Detailed description

Tool length compensation amount setting

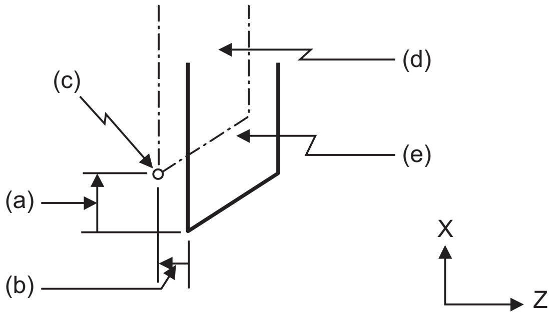
This function compensates the tool length with respect to the programmed basic position. This position may generally be set to either the center position of the turret or the tool nose position of the basic tool.

(1) Center position of turret



(a) X-axis tool length compensation amount
(b) Z-axis tool length compensation amount
(c) Basic position (basic point)

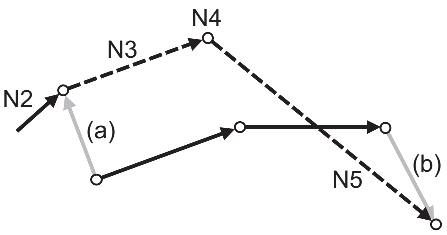
(2) Tool nose position of basic tool



(a) X-axis tool length compensation amount
(b) Z-axis tool length compensation amount
(c) Basic position
(d) Basic tool
(e) Tool used for machining

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.



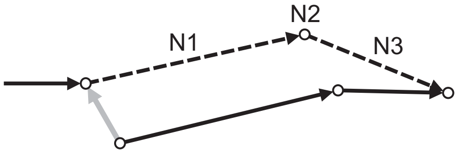
```
N1 T0100 ;
N2 G1 X10.0 Z10.0 F100 ;
N3 G1 X13.0 Z15.0 F20.0 ;
N4 T0200 ;
N5 G1 X13.0 Z20.0 F25.0 ;
```

─── Programmed machining path
- - - Compensated tool path
─── (a) Compensation amount (old)
─── (b) Compensation amount (new)

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.

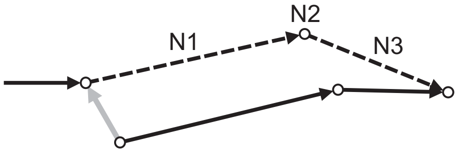


N1 X10.0 Z10.0 F10 ;
N2 T0000 ;
N3 G1 X10.0 Z20.0 ;

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.

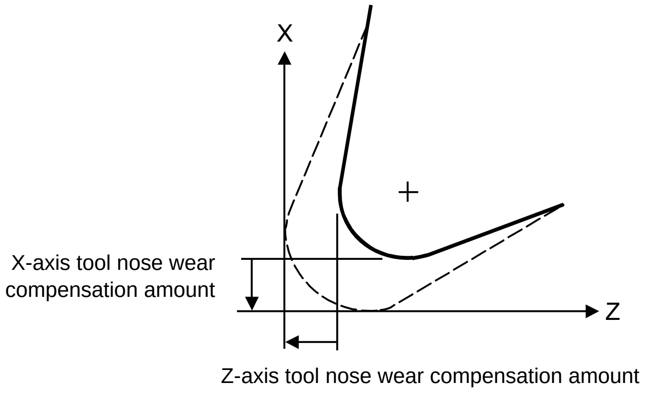
(4) The tool length compensation and tool nose wear compensation amounts are cleared by resetting and by emergency stop. They can be retained by parameter "#1099 Treset".

12.3 Tool Nose Wear Compensation

Detailed description

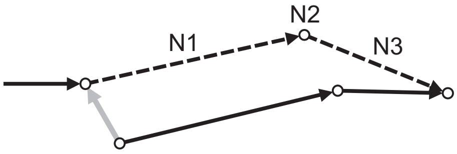
Tool nose wear compensation amount setting

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


X
Z
X-axis tool nose wear compensation amount
Z-axis tool nose wear compensation amount

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.

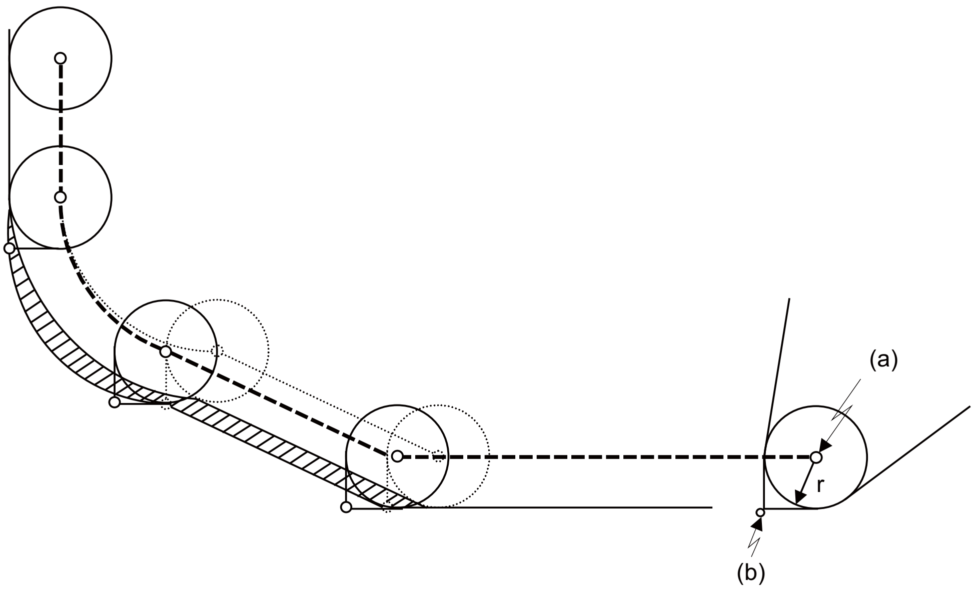
(3) Even if the compensation amount of the compensation No. currently selected by MDI is changed during automatic operation, the changed compensation amount will not be valid unless a T command with the same No. is executed again.

(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

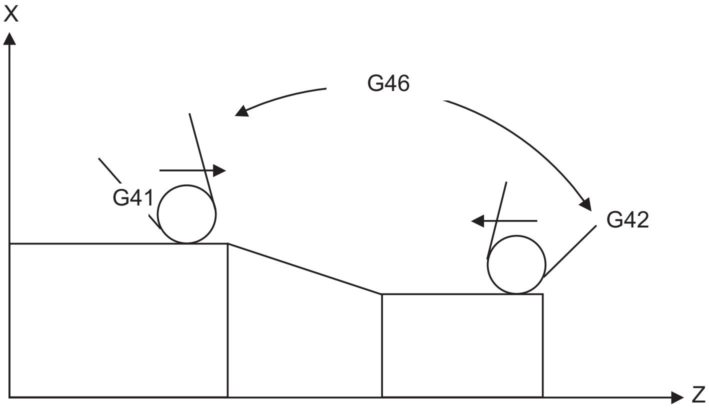
G42(Xx/Uu Zz/Ww) ; ... Tool nose radius compensation right

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

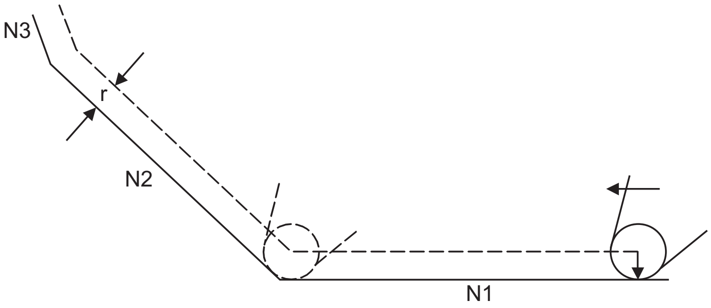
X_/U_	X axis end point coordinate (X is the absolute value of workpiece coordinate system, U is the incremental value from current position)
Z_/W_	Z axis end point coordinate (Z is the absolute value of workpiece coordinate system, W is the incremental value from current position)

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.

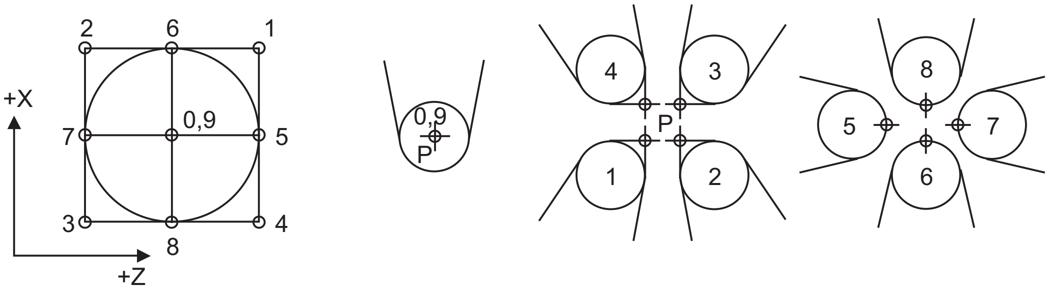
(7) The compensation plane, movement axes and next advance direction vector follow the plane selection command designated by G17, G18 or G19.
G17 ... X-Y plane X, Y, I, J
G18 ... Z-X plane Z, X, K, I
G19 ... Y-Z plane Y, Z, J, K

12.4.1 Tool Nose Point and Compensation Directions

Detailed description

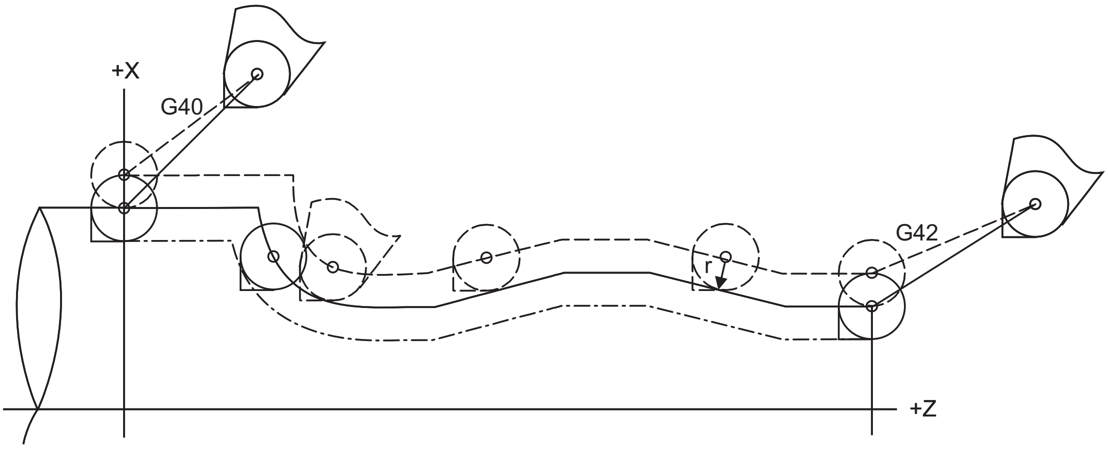
Tool nose point

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.)



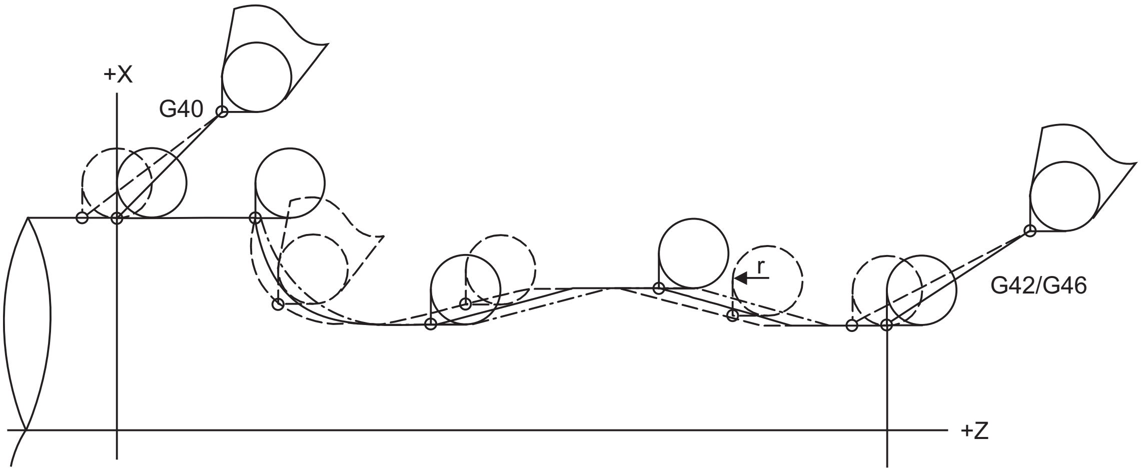
Tool nose point and compensation operation

(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

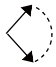
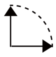
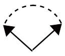
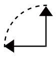
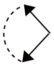
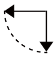
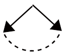
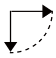
The compensation directions of the G41/G42 commands is determined by the G41/G42 codes, while that of the G46 command is automatically determined in accordance with the following table from the relationship between the tool nose points and the commanded movement vectors.

(1) If the initial movement vector (including G0) corresponds to a "×" 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 between 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 "×" 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]

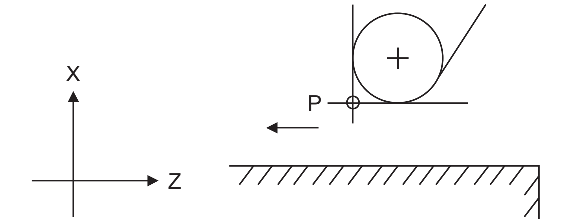
			Compensation direction of tool nose										
			Tool nose points										
			1	2	3	4	5	6	7	8			
Direction of tool nose advance	Movement vector (tool nose points 1 to 4)		Right	Right	Left	Left	×	Right	×	Left	→	Movement vector (tool nose points 5 to 8)	Direction of tool nose advance
		↗	×	Right	×	Left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

(Note 1) "×" 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 <-)



As shown in the figure on the left, the workpiece is on the X axis (-) side from the tool nose position and tool movement direction.
Consequently, the compensation direction will be on the right side of the workpiece to the direction of tool nose advance.

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.

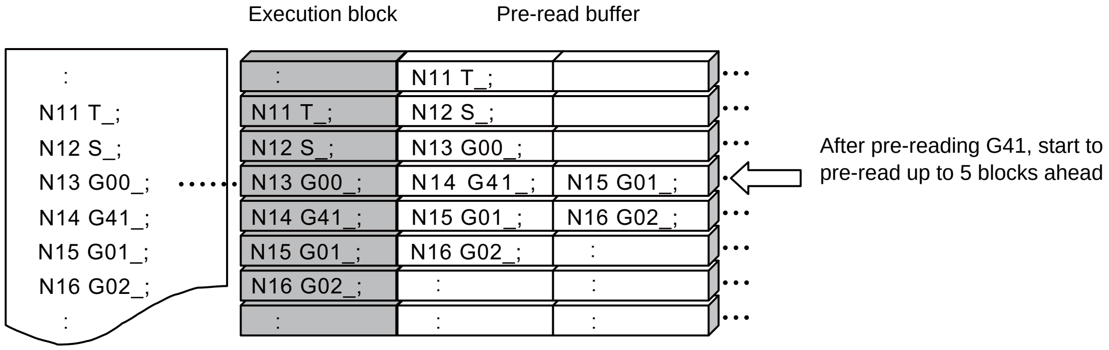
Tool nose radius compensation start (start-up)

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]



```
   :
N11 T_;
N12 S_;
N13 G00_;
N14 G41_;
N15 G01_;
N16 G02_;
   :
```

Execution block	Pre-read buffer	
:	N11 T_;	
N11 T_;	N12 S_;	
N12 S_;	N13 G00_;	
N13 G00_;	N14 G41_;	N15 G01_;
N14 G41_;	N15 G01_;	N16 G02_;
N15 G01_;	N16 G02_;	:
N16 G02_;	:	:
:	:	:

After pre-reading G41, start to pre-read up to 5 blocks ahead

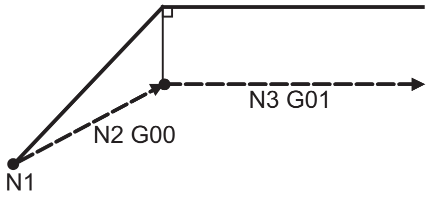
There are two ways of starting the compensation operation: type A and type B.
The type to be used is selected by setting of parameter "#8157 tool nose comp type B".
This type is used in common with the compensation cancel type.

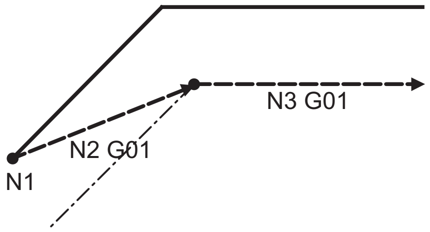
#8157 Nose R comp type B	Type	Explanation
0	Type A	When starting up/canceling a command block with tool nose radius compensation and radius compensation, type A will not conduct intersection operation processing to the block and, instead, convert it to an offset vector which is vertical to the command 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.

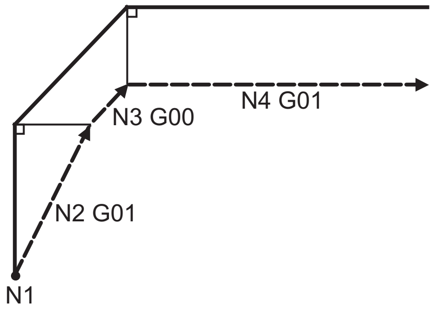
Start operation for tool nose radius compensation

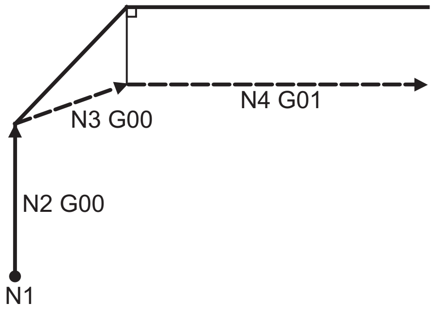
When starting tool nose radius compensation, the tool will not move as much as the specified compensation amount by either G41, G42 or G46 command alone. Tool nose radius compensation cannot be applied to the G00 command. It can be applied from G01, G02 or G03 command. Note that even if there is an axis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

:
N1 G42;
N2 G00 X_Z_;
N3 G01 X_Z_F_;
:
	

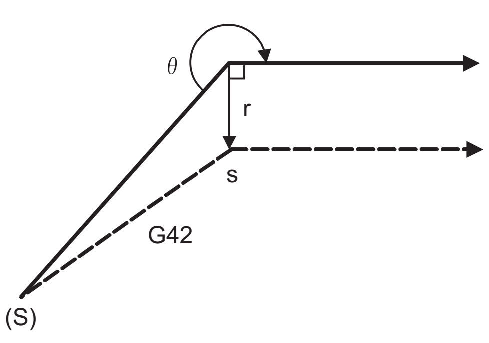
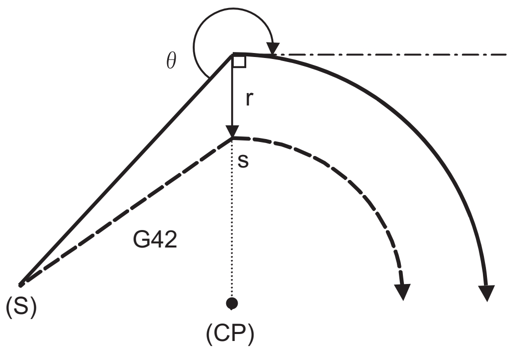
:
N1 G42;
N2 G01 X_Z_F_;
N3 G01 X_Z_;
:
	

:
N1 G42;
N2 G01 X_Z_F_;
N3 G00 X_Z_;
N4 G01 X_Z_;
:
	

:
N1 G42;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G01 X_Z_F_;
:
	

—— 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

Linear -> Linear	**Linear -> Circular**
	

(S) Start point

(CP) Circular radius

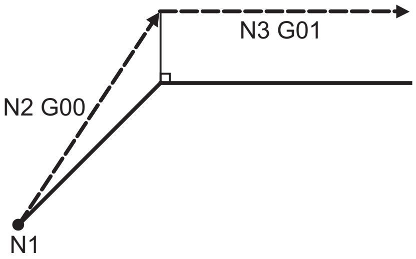
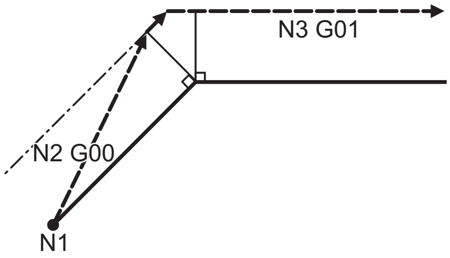
r : Nose R radius

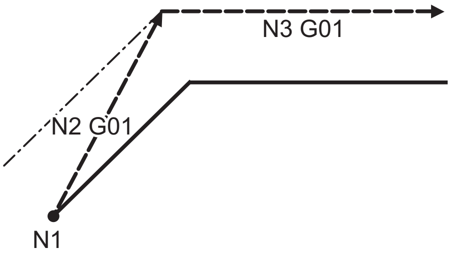
s : Stop point with single block

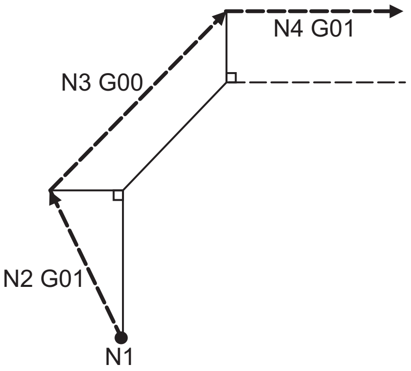
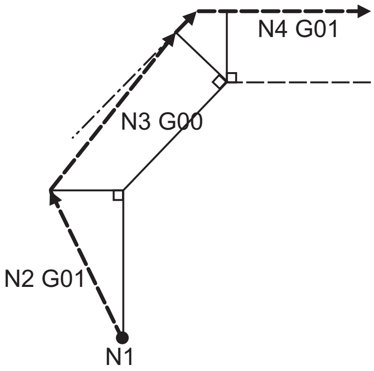
——— Program path

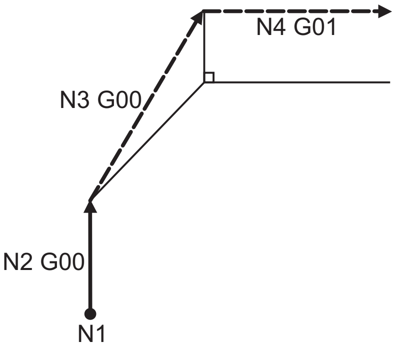
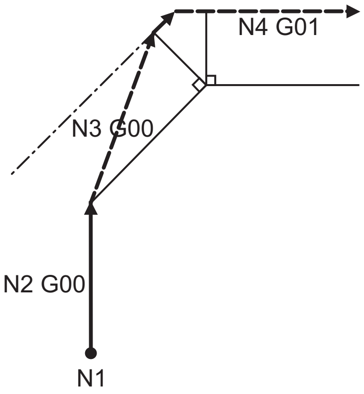
- - - - Nose R center path

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

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

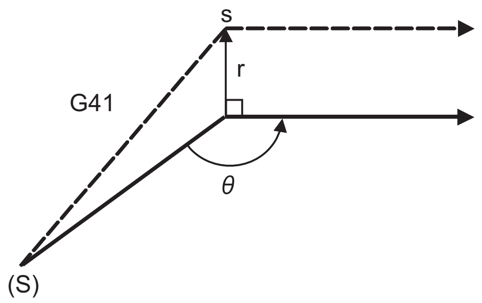
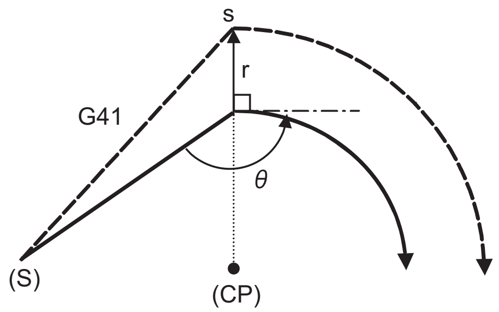
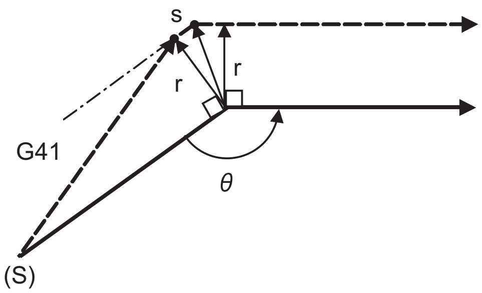
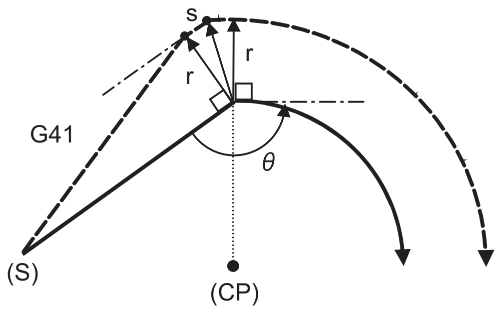
:
N1 G41;
N2 G01 X_Z_F_;
N3 G01 X_Z_;
:
		

:
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

(4) When a G41/G42/G46 command is issued at an outside corner (obtuse angle), in the same block as a movement command [90° <= θ < 180°]

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

(S) Start point

(CP) Circular radius

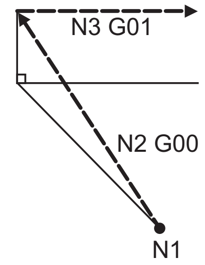
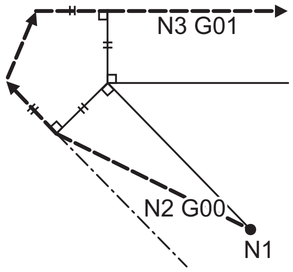
r : Nose R radius

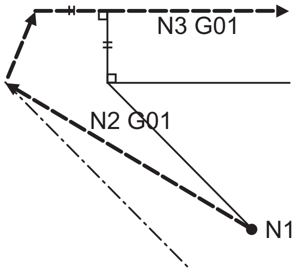
s : Stop point with single block

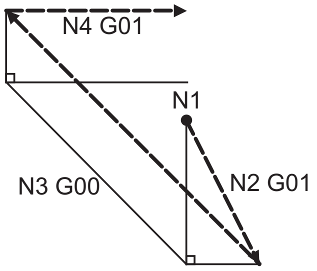
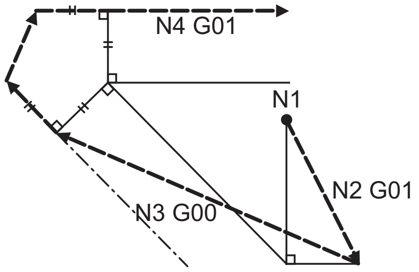
—— Program path ---- Nose R center path

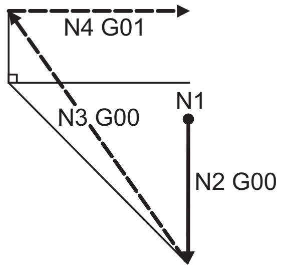
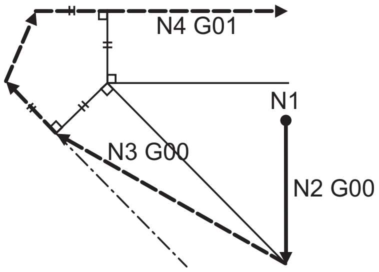
(5) When an independent G41/G42/G46 command is issued at an outside corner (acute angle)

	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_;
:
		

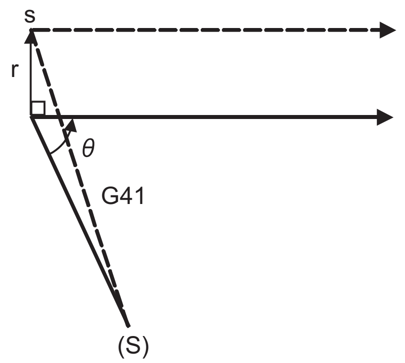
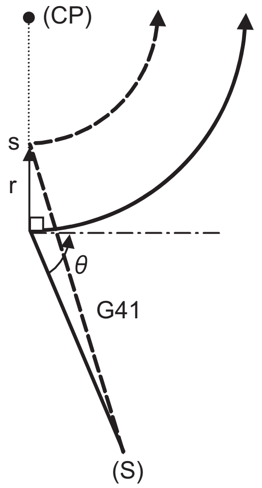
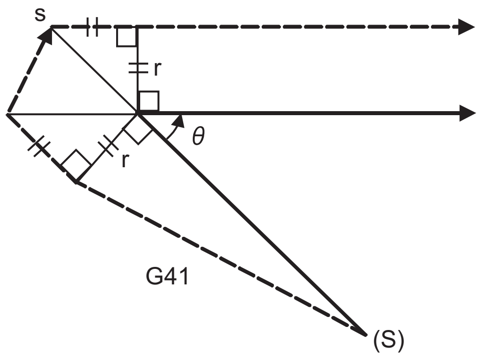
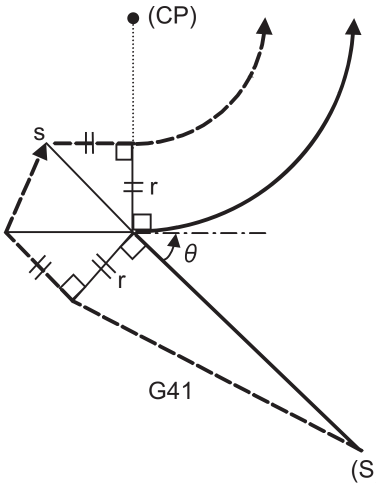
:
N1 G41;
N2 G01 X_Z_F_;
N3 G00 X_Z_;
N4 G01 X_Z_;
:
		

:
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 [θ < 90°]

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

(S) Start point

(CP) Circular radius

r : Nose R radius

s : Stop point with single block

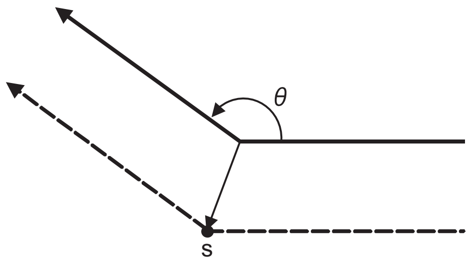
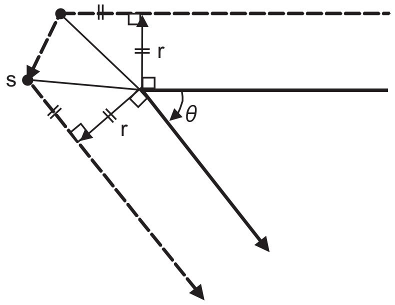
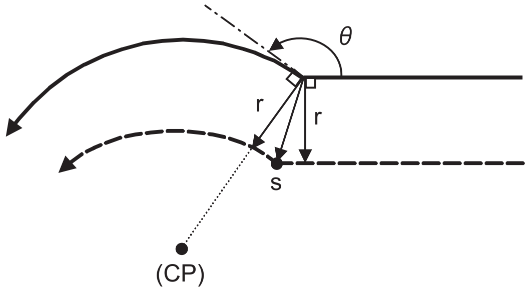
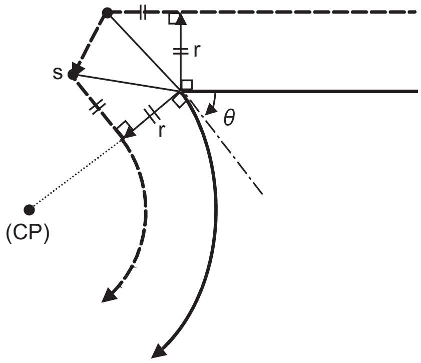
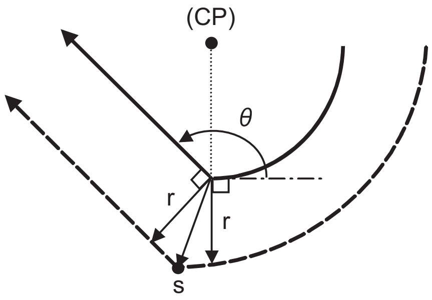
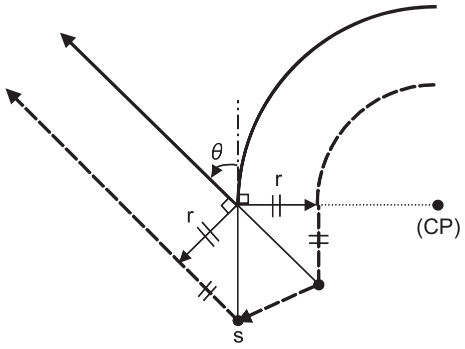
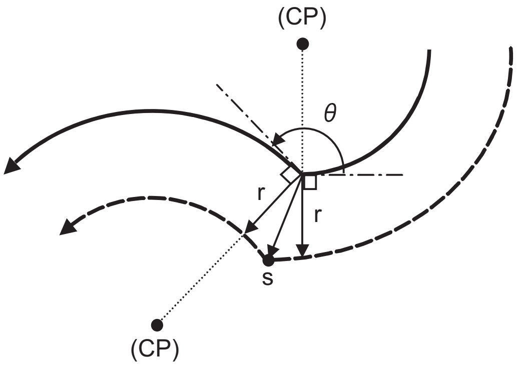
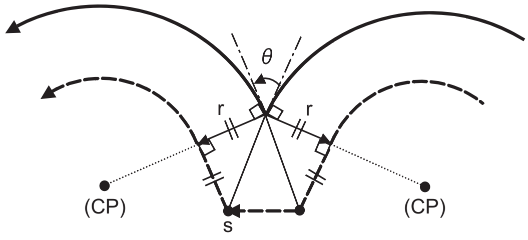
—— Program path   ---- Nose R center path

(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

Linear -> Linear (90° <= θ < 180°)	**Linear -> Linear (0° < θ < 90°)**
	
Linear -> Linear (90° <= θ < 180°)	**Linear -> Linear (0° < θ < 90°)**
	
Circular -> Linear (90° <= θ < 180°)	**Circular -> Linear (0° < θ < 90°)**
	
Circular -> Circular (90° <= θ < 180°)	**Circular -> Circular (0° < θ < 90°)**
	

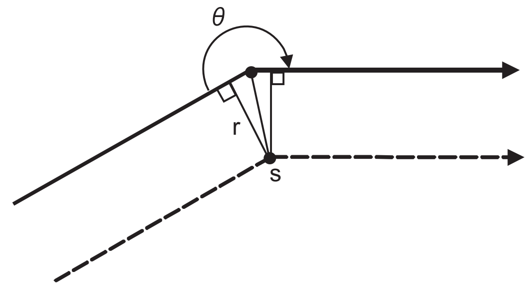
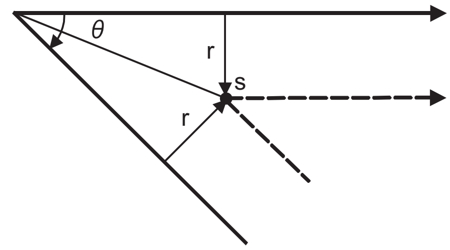
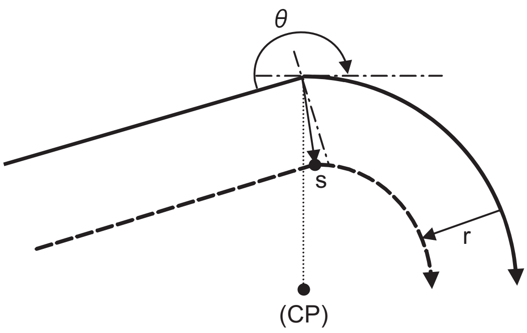
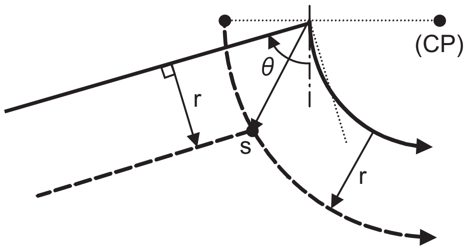
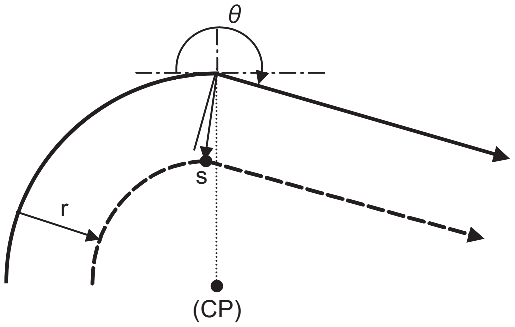
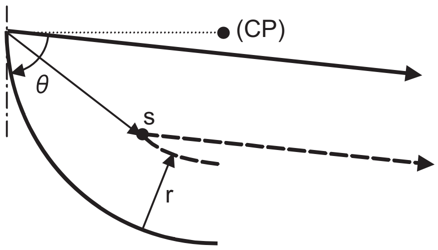
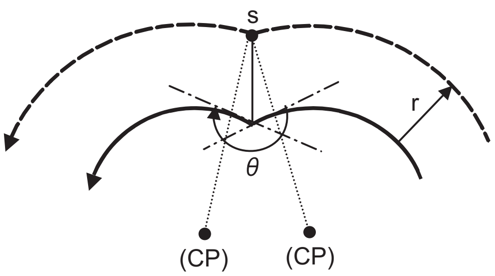
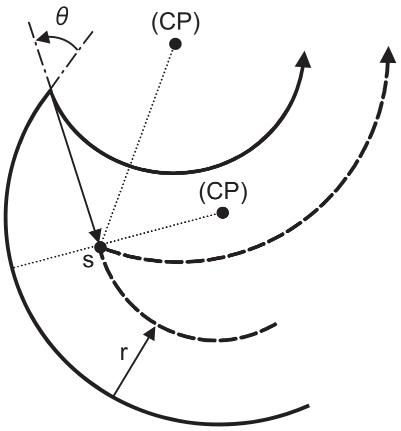
(CP) Circular radius

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

(CP) Circular radius

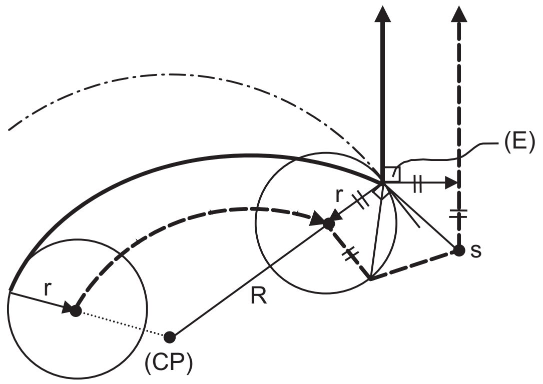
r : Nose R radius

s : Stop point with single block

—— Program path

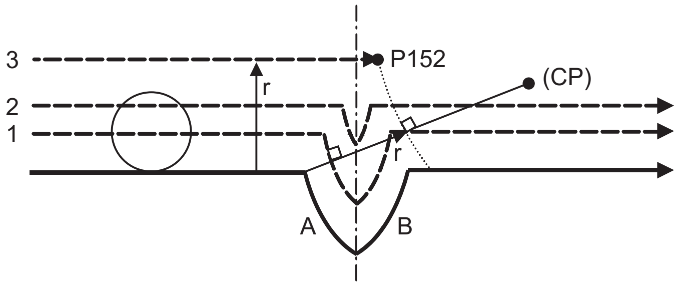
---- Nose R center path

(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.



(E) End point of circular
(CP) Center of circular
r : Nose R radius

(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.



(CP) Center of circular A
r : Nose R radius

—— Program path

---- Nose R center path

Tool nose R compensation cancel

Tool nose R compensation mode will be canceled when any of the following conditions is met.
However, this must be any movement command except a circular command.
If the compensation is canceled by a circular command, program error (P151) will occur.

(1) The G40 command has been executed.
(2) 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.

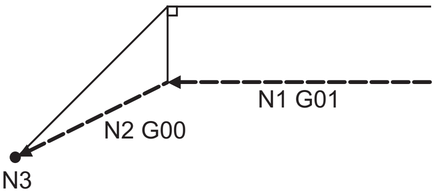
Tool Nose Radius Compensation Operations

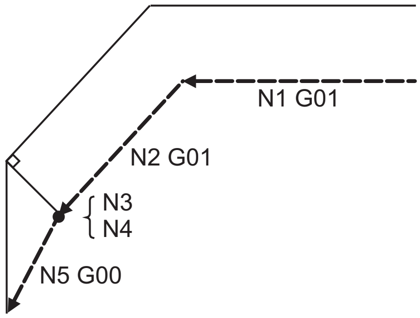
Tool nose radius compensation cancel command results as follows.

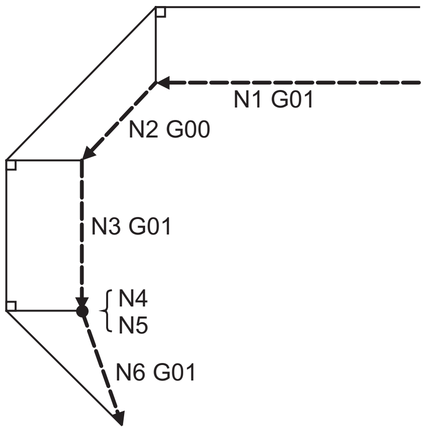
(1) If the command before G40 is G00 after the tool nose radius compensation is completed, the tool nose radius compensation is temporarily stopped, and the tool nose radius compensation is canceled in that state by an independent G40 command.
(2) If there is an interpolation command before G40 after 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.

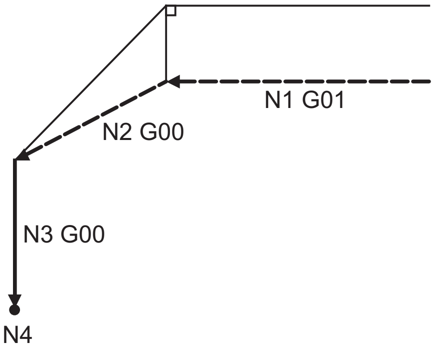
(4) Relation of an inside corner/outside corner and cancel

(a)-1 When an independent G40 command is issued at an inside corner

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G40;
:
	

:
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G40;
N4 M05;
N5 G00 X_Z_;
:
	

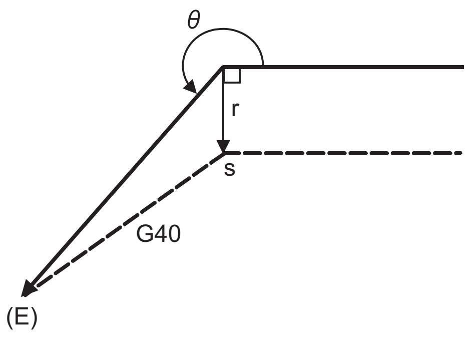
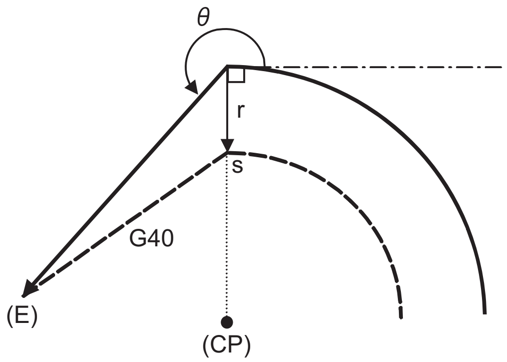
:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G01 X_Z_;
N4 G40;
N5 M05;
N6 G01 X_Z_;
:
	

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G40;
:
	

——— 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

Linear -> Linear	**Circular -> Linear**
	

(E) End point

(CP) Center of circular

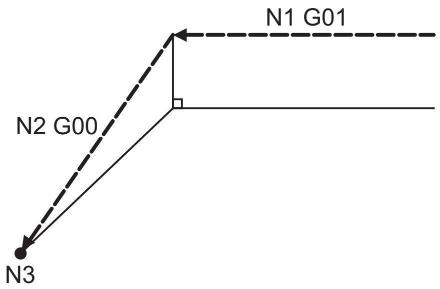
r : Nose R radius

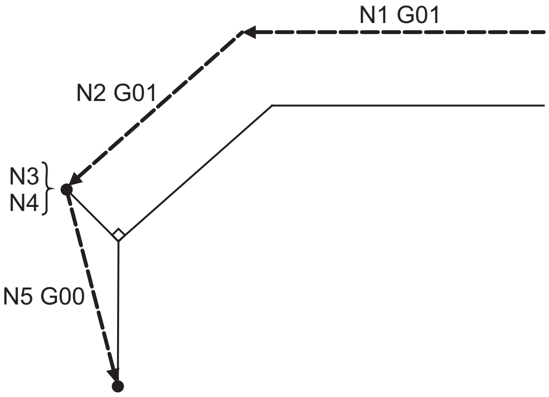
s : Stop point with single block

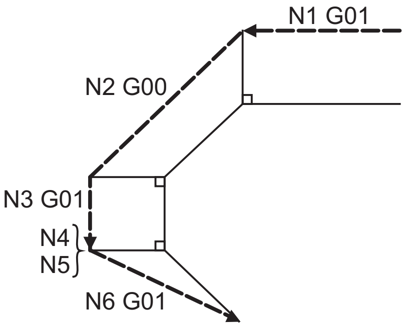
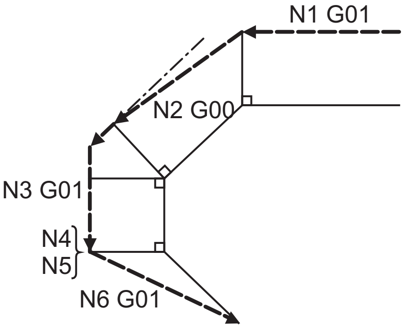
—— Program path ---- Nose R center path

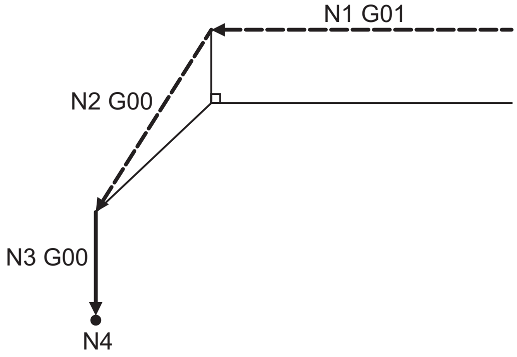
(b)-1 When an independent G40 command is issued at an outside corner (obtuse angle)

	Type A	Type B

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G40;
:
		

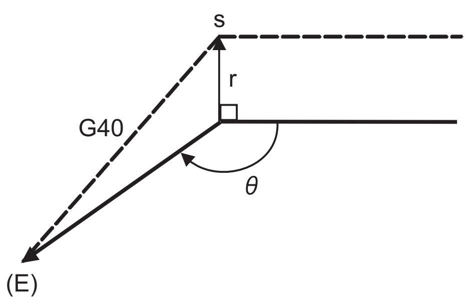
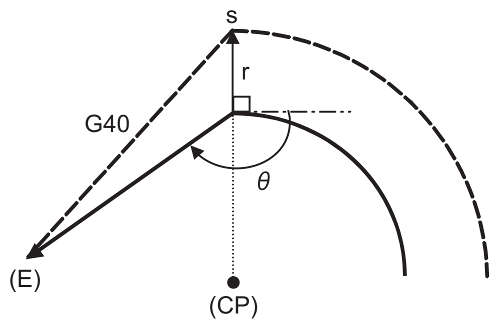
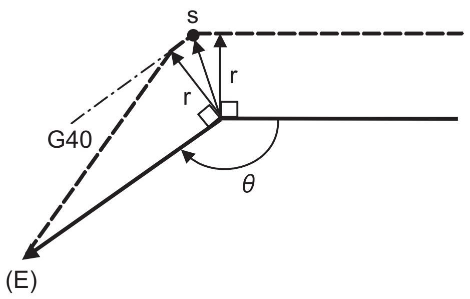
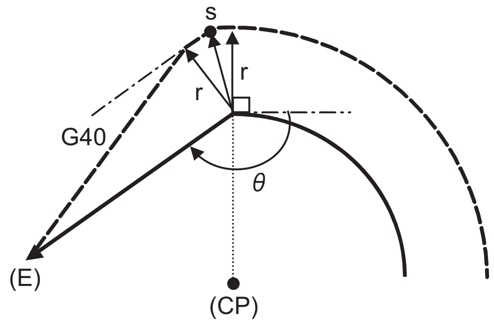
:
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G40;
N4 M05;
N5 G00 X_Z_;
:
		

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G01 X_Z_;
N4 G40;
N5 M05;
N6 G01 X_Z_;
:
		

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G40;
:
		

—— 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

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

(E) End point

(CP) Center of circular

r : Nose R radius

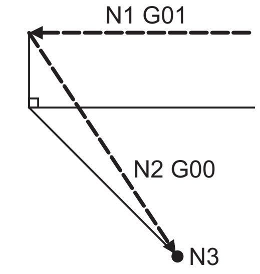
s : Stop point with single block

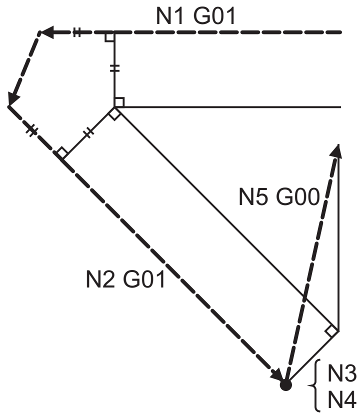
—— Program path

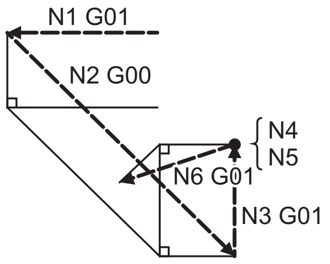
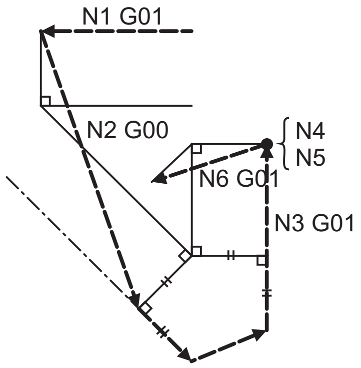
---- Nose R center path

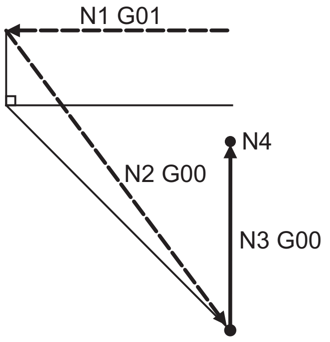
(c)-1 When an independent G40 command is issued at an outside corner (acute angle)

	Type A	Type B

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G40;
:
		

:
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G40;
N4 M05;
N5 G00 X_Z_;
:
		

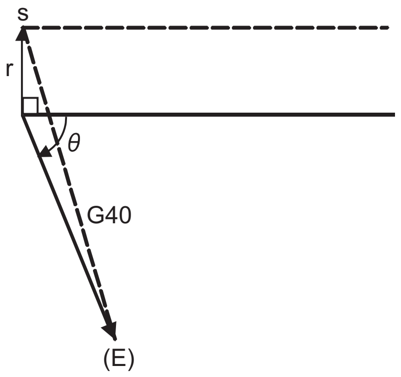
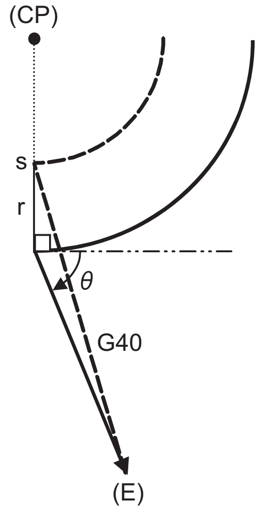
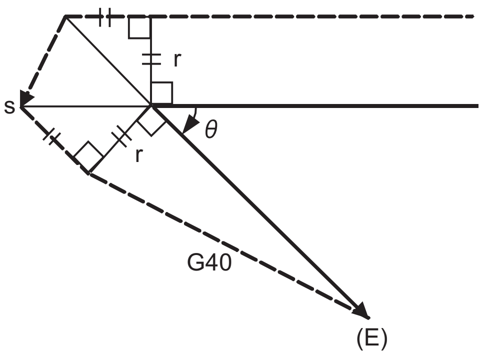
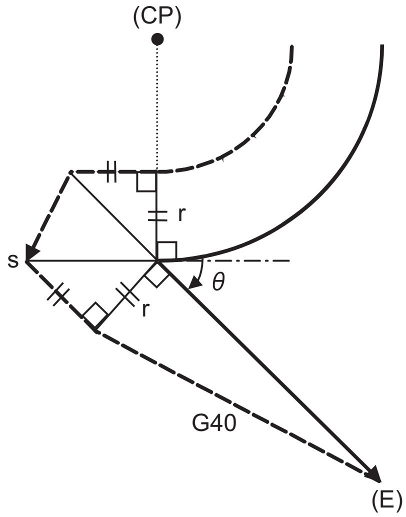
:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G01 X_Z_;
N4 G40;
N5 M05;
N6 G01 X_Z_;
:
		

:
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G40;
:
		

—— 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

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

(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

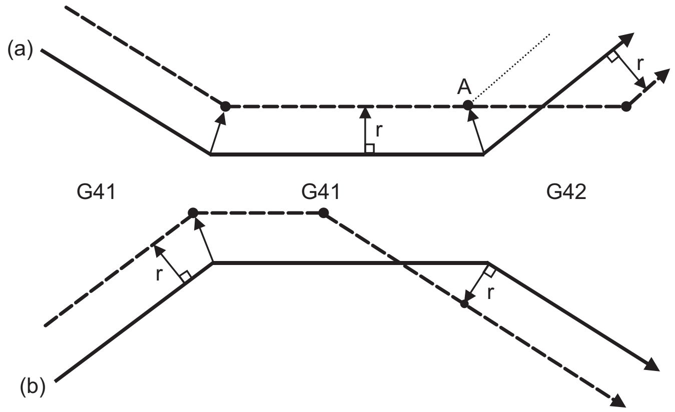
Changing the compensation direction during tool nose radius compensation

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

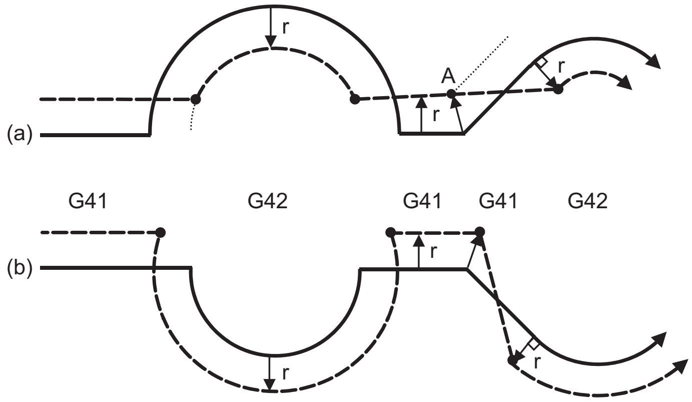
G code	Compensation direction
G41	Left-side compensation
G42	Right-side compensation

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

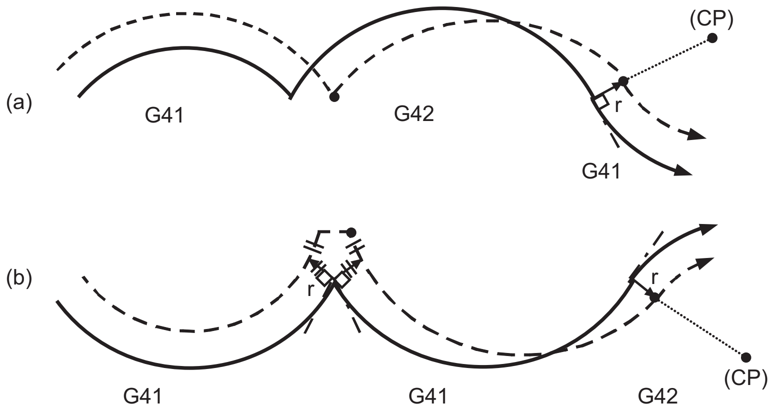


—— Program path ---- Nose R center path

(3) Circular -> Circular

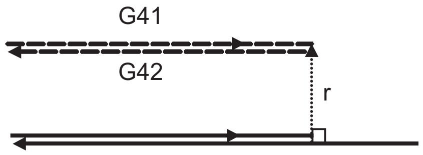
(a) When there is an intersection at the change of compensation direction

(b) When there is no intersection at the change of compensation direction



(CP) Center of circular

(4) Linear return

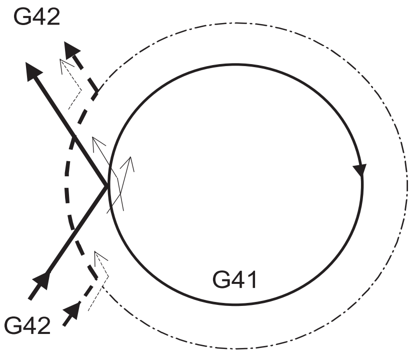


(5) Arc exceeding 360° due to compensation method

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_;
:
```

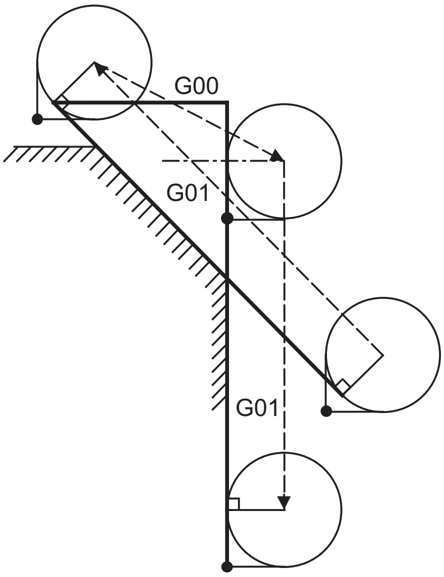
—— Program path

---- Nose R center path

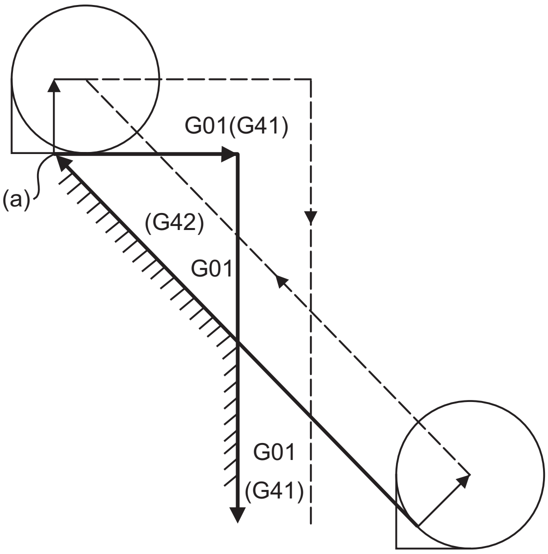
–·–·– Uncut section

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

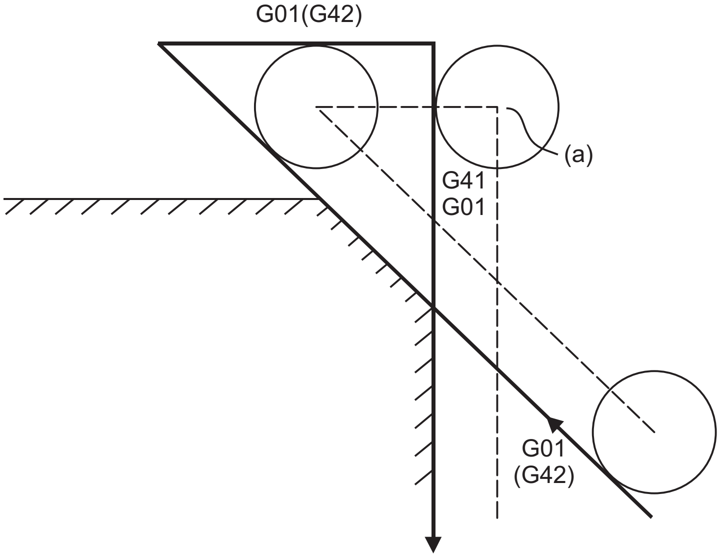
(1) G46 command operation



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



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



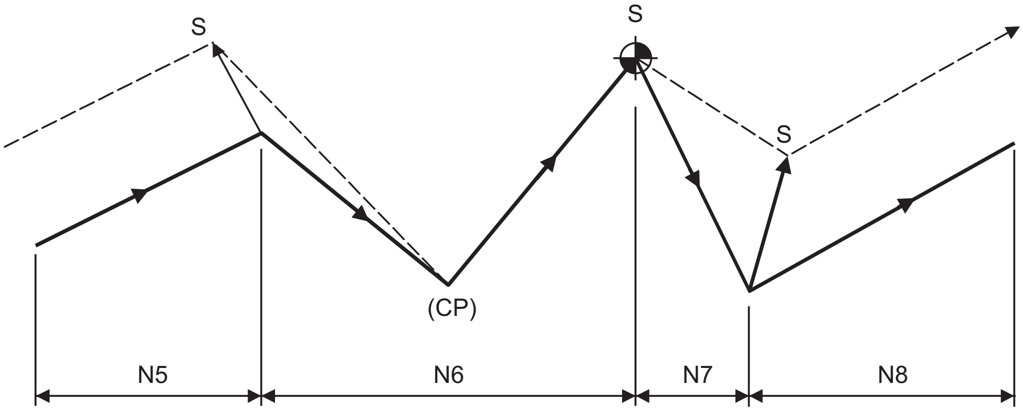
—— Program path

---- 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

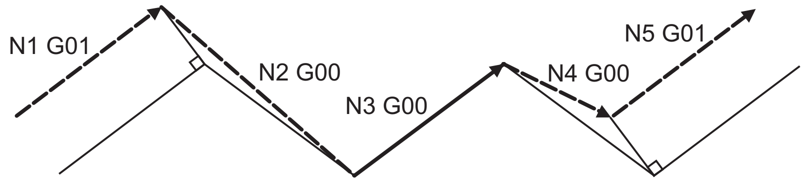
(2) The compensation vector will be eliminated temporarily with the G53 command (Basic machine coordinate system selection).

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

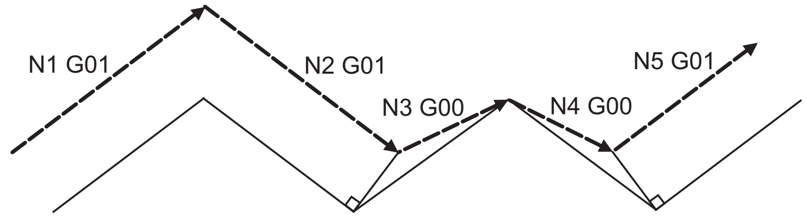
(3) Positioning (G00) commands

Tool nose radius compensation is temporarily canceled with G00 commands.

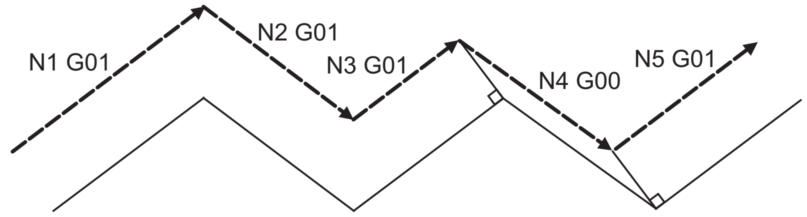
```
      :
N1 G01 X_Z_F_;
N2 G00 X_Z_;
N3 G00 X_Z_;
N4 G00 X_Z_;
N5 G01 X_Z_;
      :
```



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

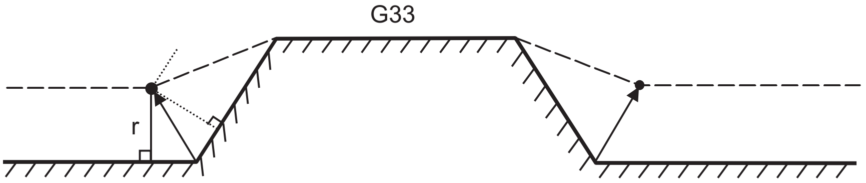


```
      :
N1 G01 X_Z_F_;
N2 G01 X_Z_;
N3 G01 X_Z_;
N4 G00 X_Z_;
N5 G01 X_Z_;
      :
```



(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.

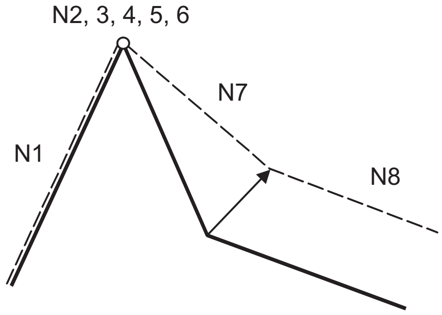
Blocks without movement

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

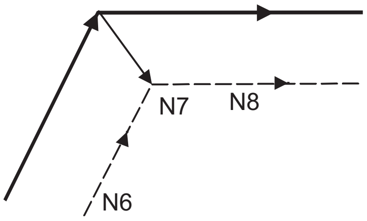
(1) When commanded at the compensation start
Compensation vector cannot be created when there are four or more successive blocks without movement, or when pre-read prohibiting M command is issued.

```
N1 U60.W30.T0101 ;
N2 G41 ;           }
N3 G4 X1000 ;      }
N4 F100 ;          }  Block without movement
N5 S500 ;          }
N6 M3 ;            }
N7 U- 50.W20. ;
N8 U- 20.W50. ;
```



(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.

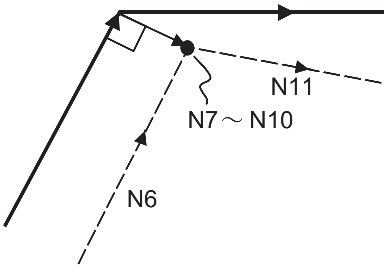
```
N6 U200. W100. ;
N7 G04X P1000 ;          ...  Block without movement
N8 W200. ;
```



Block N7 is executed at N7 in the figure.

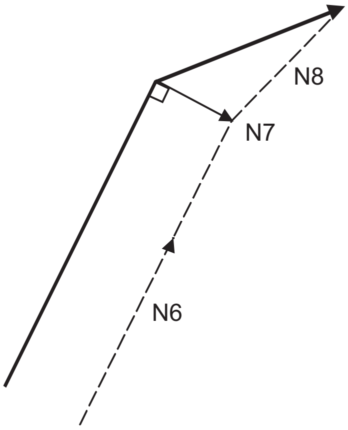
Compensation vector will be created perpendicularly to the end point of the previous block when there are four or more successive blocks without movement, or when pre-read prohibiting M command is issued.
In this case, a cut may occur.

```
N6 U200. W100. ;
N7 G4 X1000 ;    }
N8 F100 ;        }
N9 S500 ;        }  Block without movement
N10 M4 ;         }
N11 W100. ;
```



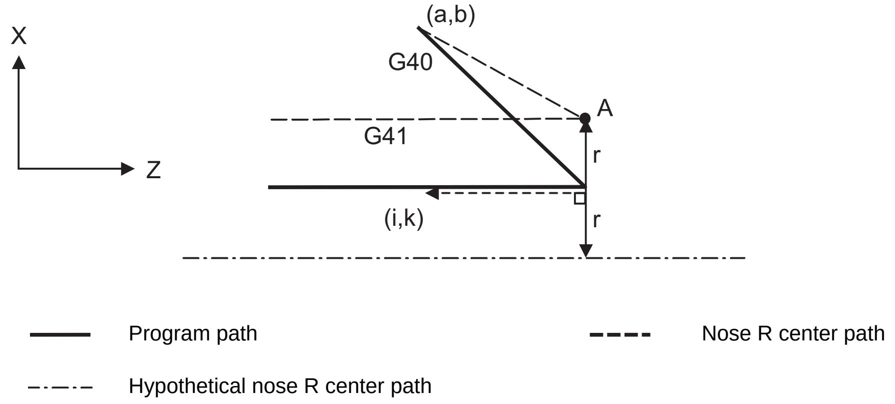
(3) When commanded together with compensation cancel
Only the compensation vectors are canceled when a block without movement is commanded together with the G40 command.

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



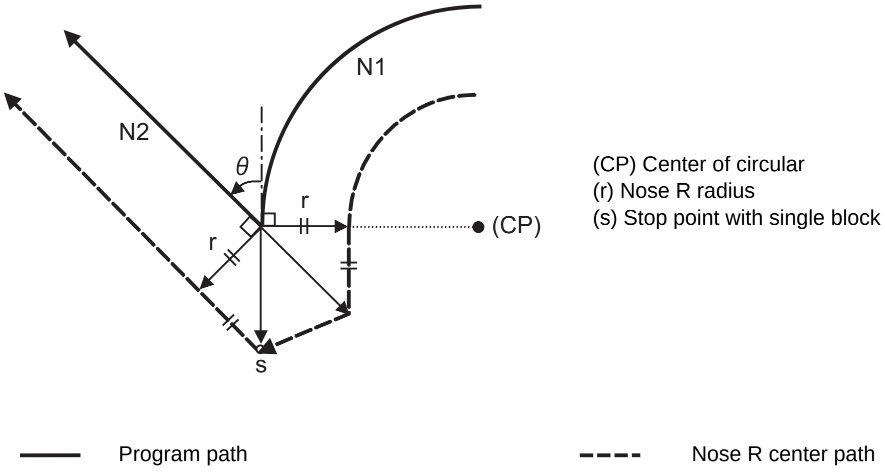
When I, J, K are commanded in G40

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__ ;

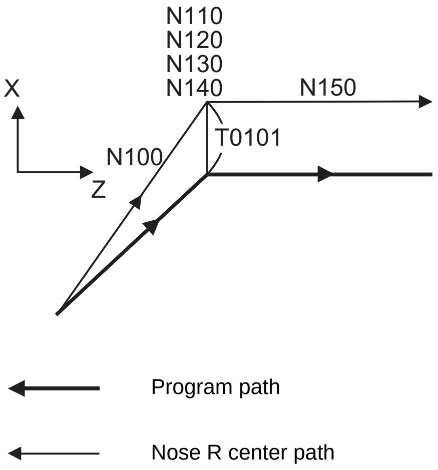
Assign a linear command (G00, G01) in a movement mode.

Detailed description

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

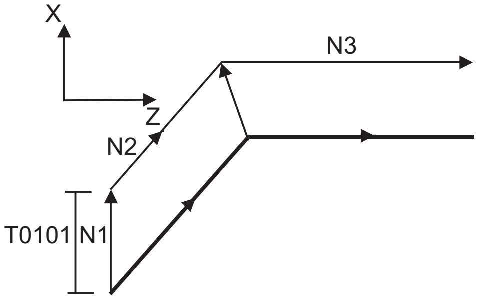
This section describes the new I, K type vectors (G18 plane) created by this command. (Similar descriptions apply to vector I, J for the G17 plane and to J, K for the G19 plane.)
As shown in the following figures, I, K type vectors create compensation vectors which are perpendicular to the direction 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. ;
    :
```

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

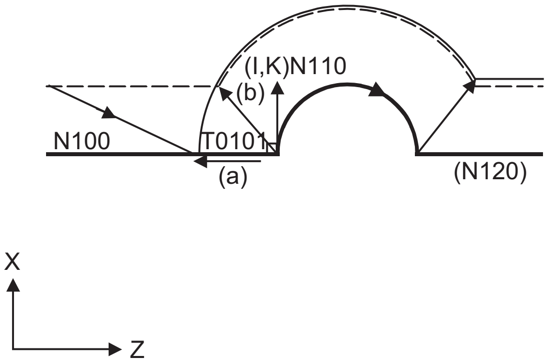


```
 (G40)
   :
N1 G41 K150. T0101 ;
N2 U100. W100. ;
N3 W150. ;
    :
```

◀━━━ 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

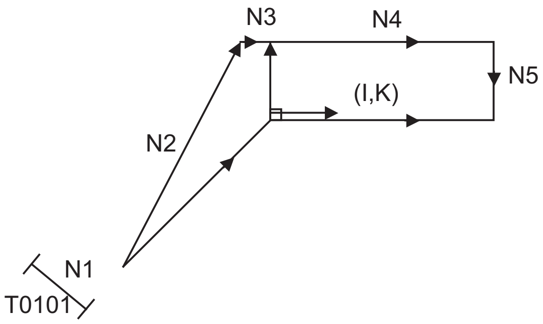
(b) Intersection point calculation type vector

◀━━━ Program path

◀─── Nose R center path

– – – – – – Path after intersection point calculation

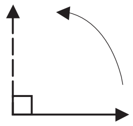
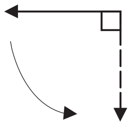
(4) When I, K has been commanded in a block without movement



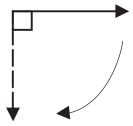
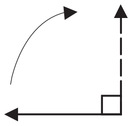
```
N1 G41 T0101 G01 F1000 ;
N2 U100. W100. ;
N3 G41 K50. ;
N4 W150. ;
N5 G40 ;
```

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).

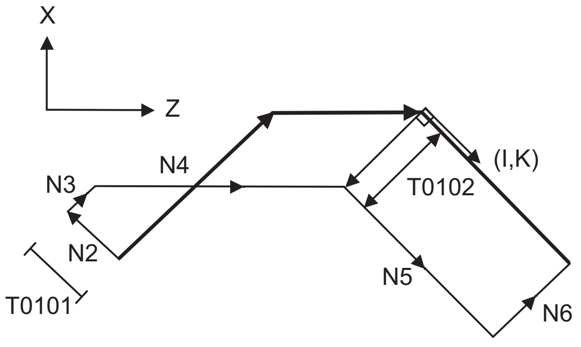
<table>
<tr><td colspan="2">(Example 1) With K100.</td><td colspan="2">(Example 2) With K-100.</td></tr>
<tr><td colspan="2"></td><td colspan="2"></td></tr>
<tr><td>←――――</td><td>(0, 100) IK direction</td><td>←――――</td><td>(0, -100) IK direction</td></tr>
<tr><td>←- - - - -</td><td>Offset vector direction</td><td>←- - - - -</td><td>Offset vector direction</td></tr>
</table>

(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).

<table>
<tr><td colspan="2">(Example 1) With K100.</td><td colspan="2">(Example 2) With K-100.</td></tr>
<tr><td colspan="2"></td><td colspan="2"></td></tr>
<tr><td>←――――</td><td>(0, 100) IK direction</td><td>←――――</td><td>(0, -100) IK direction</td></tr>
<tr><td>←- - - - -</td><td>Offset vector direction</td><td>←- - - - -</td><td>Offset vector direction</td></tr>
</table>

Selection of offset modal

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



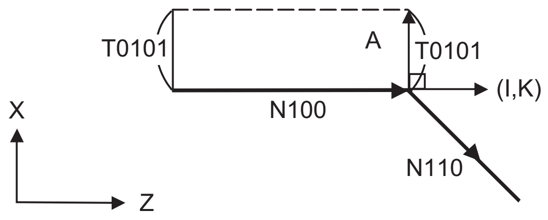
```
N1 G28 X0Z0 ;
N2 G41 T0101 F1000 ;
N3 G01 U100. W100. ;
N4 G42 W100. I - 100. K100.T0102 ;
N5 U- 100. W100. ;
N6 G40 ;
N7 M02 ;
%
```

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>

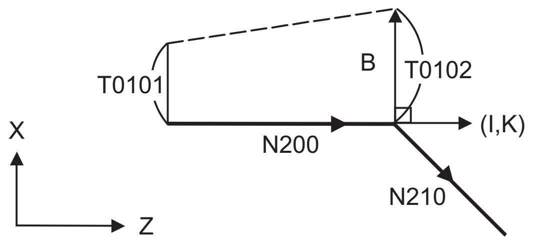
Vector A is the compensation amount registered in tool offset No. modal 1 of the N100 block.



```
  (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.

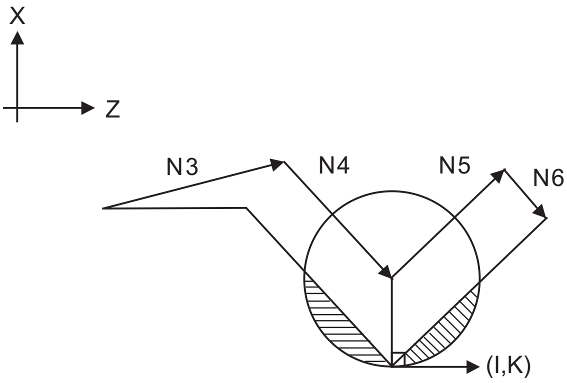


```
  (G41 T0101)
     :
N200 G41 W150. K50. T0102 ;
N210 U - 100. W100. ;
      :
```

Precautions

(1) 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.

(2) 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 ;
```

(3) 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

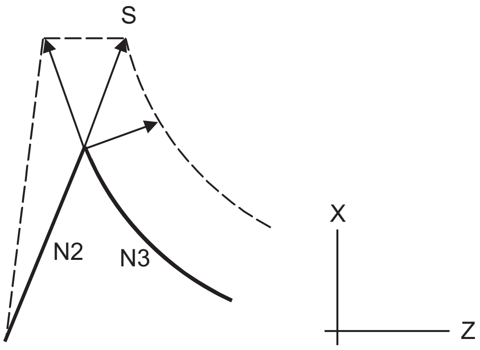
Detailed description

MDI interruption

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

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

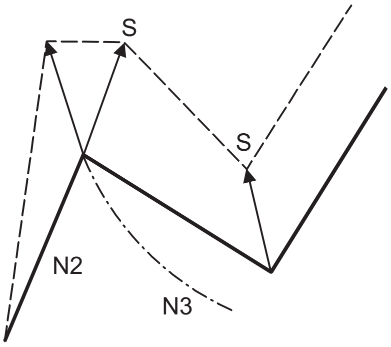
```
Automatic operation              MDI interruption
N1 G41 T0101;
N2 U50. W20. ;
                           <---  S1000 M3;
N3 G3 U-40. W40. R70. ;
```



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

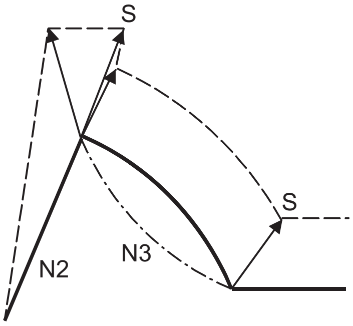
With linear interrupt

```
Automatic operation                  MDI interruption
N1 G41 T0101;
N2 U50. W20. ;
                               <---  U-30. W50. ;
                                     U50. W30. ;
N3 G3 U-40. W40. R70. ;
```



With circular interruption

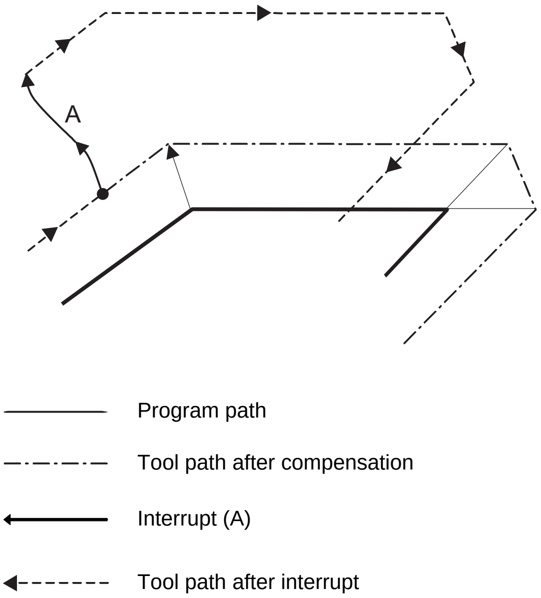
```
Automatic operation                  MDI interruption
N1 G41 T0101;
N2 U50. W20. ;
                               <---  G2 U-40. W40. R70. ;
                                     G1 W40. ;
N3 G3 U-40. W40. R70. ;
```



Manual interruption

(1) Interrupt with manual absolute OFF.

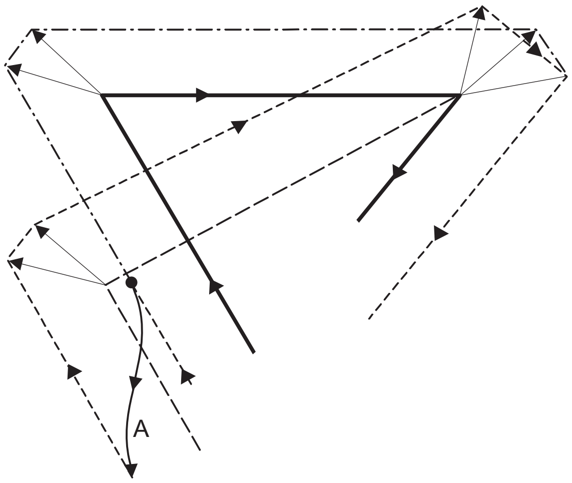
The tool path will deviate from the compensated path by the interrupt amount.



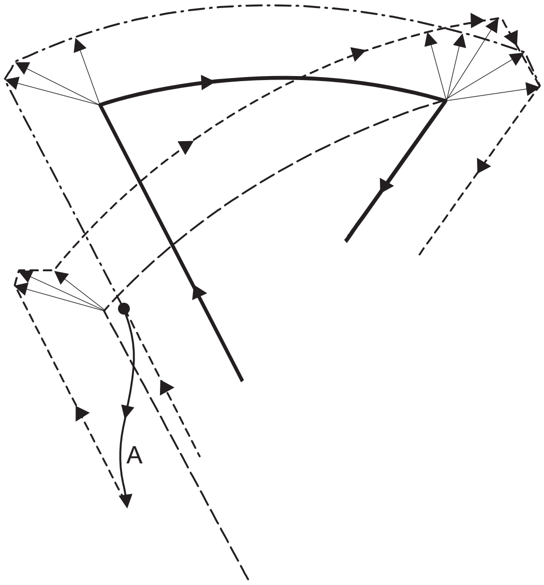
(2) Interrupt with manual absolute ON.

In the incremental value mode, the same operation will be performed as the manual absolute OFF.
In the absolute value mode, however, the tool returns to its original path at the end point of the block following the interrupted block, as shown in the figure.

[Line-Line-Line]



[Line-arc-Line]



——— 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

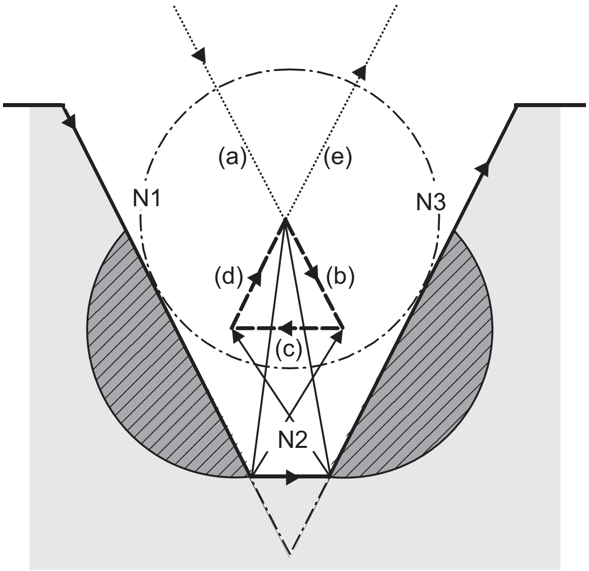
A tool, whose tool nose has been compensated under the tool nose radius compensation function by the usual two-block pre-read, may sometimes cut into the workpiece.
This is known as interference, and interference check is the function which prevents this from occurring.
The table below shows the three 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(P153) before executing a block which will cause cutting.
(2)	Interference check avoidance function	1	0	The tool path is changed to prevent cutting from occurring. Operation stops with a program error(P153) when it is unable to change.
(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 G1 X100. Z50.;
N2 Z70;
N3 X300. Z120.;
```

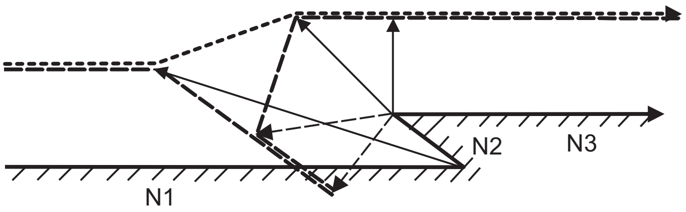


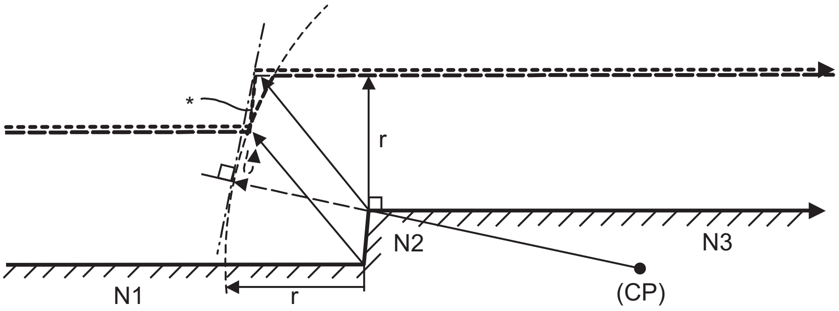
(1) With alarm function
An alarm is given when N1 is executed. The buffer correction function can thus be used to change N1 to the following, enabling machining to continue: N1 G1 X-100. Z-20. ;

(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.
Tool nose R center path is (a)->(b)->(c)->(d)->(e) .

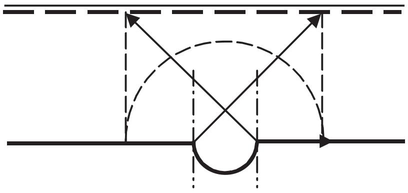
Operation when interference avoidance function is valid





◀━━━━	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.

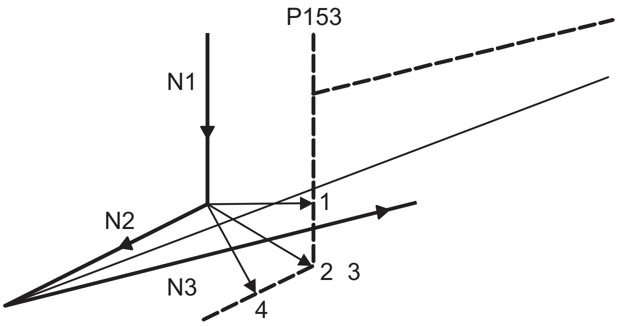


◀━━━━	Program path
◀– – – –	Nose R center path
◀┅┅┅┅	Tool center path when interference is avoided

Interference check alarm

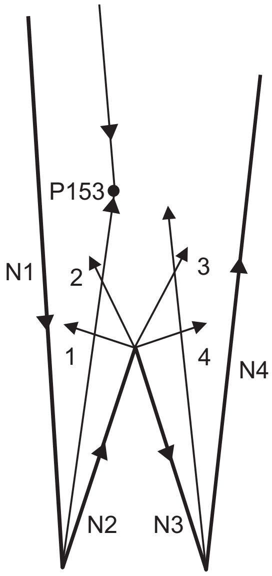
The interference check alarm occurs under the following conditions.

(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.

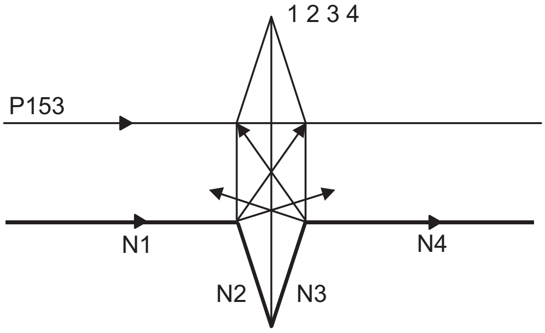


(2) When the interference check avoidance function has been selected

(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.



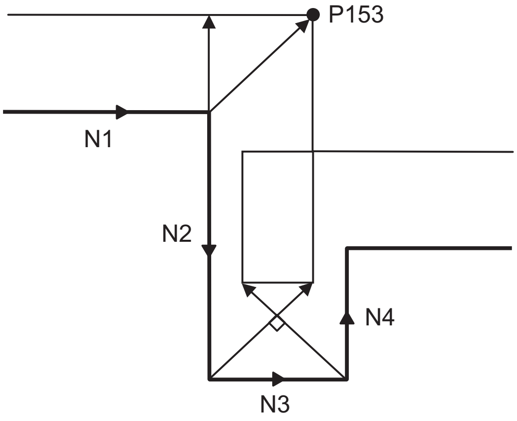
In the case shown in the figure below, the tool will move in the reverse direction at N2.
Program error (P153) now occurs before executing N1and the operation stops.

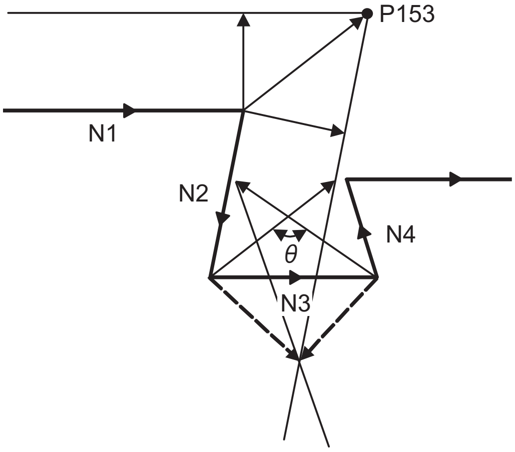


(Ex. 2) When avoidance vectors cannot be created

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.

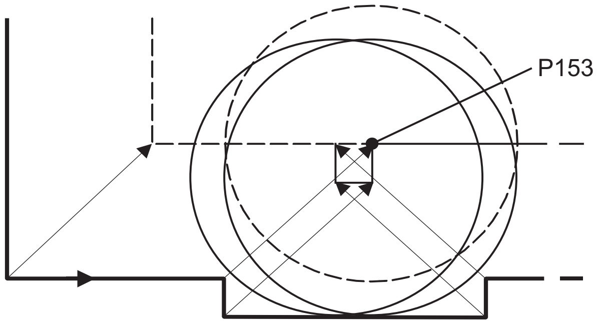
Program error (P153) will occur at the N1 end point when the vector intersecting angle is more than 90° and the operation will stop.

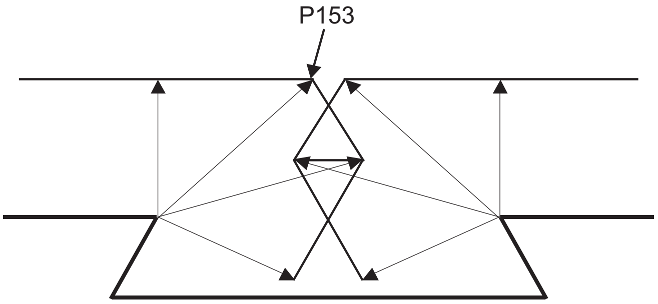




(Ex. 3) When the program advance direction and the advance direction after compensation are reversed

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.

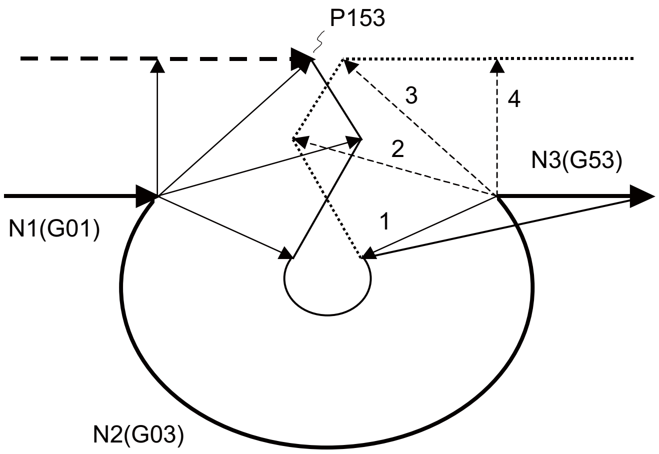




(Ex. 4) When the end point vector at the end point of the most recent block to temporarily eliminate the compensation vector interferes

Interference check will be executed even at the end point of the most recent block to eliminate the compensation vector temporarily in a similar way to check the interference at the end vector point when the compensation vector will not be eliminated. Therefore, it may be regarded as an interference even though there is no interference. If regarded as an interference, program error (P153) will occur.

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



◀━━━━ Program path

◀- - - - Nose R center path

──── Nose R center path when the interference check is invalid

········ Nose R center path when the interference check is invalid in N3 due to a command (G01 etc.) not to eliminate compensation vectors.

◀──── Valid vector

◀– – – – – Invalid vector (Invalid, however, subject to interference check)

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

Workpiece offset input

```
G10  L2  P__ X__ (U__ )Z__ (W__ )  ;
```

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)

Tool length compensation input

```
G10  L10  P__ X__ (U__ )Z__ (W__ )R__ (C__ )Q__  ;
```

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

Tool nose wear compensation input command

```
G10  L11  P__ X__ (U__ )Z__ (W__ )R__ (C__ )Q__  ;
```

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

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.

Compensation input cancel

```
G11;
```

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 not present 10001 to 10000 + 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		
		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.

(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.

Setting	Tool length compensation amount		Wear compensation amount	
	Metric system	Inch system	Metric system	Inch system
#1004=B	± 9999.999 (mm)	± 999.9999 (inch)	± 9999.999 (mm)	± 999.9999 (inch)
#1004=C	± 9999.9999 (mm)	± 999.99999 (inch)	± 9999.9999 (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 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) 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.

(8) 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.

(9) Decimal point is valid for compensation amount.

(10) G40 to G42 are ignored when they have been commanded in the same block as G10.

(11) Do not command G10 in the same block as the fixed cycles and subprogram call commands. This will cause malfunctioning and program errors.

(12) When T command has been issued in the same block as G10 with a parameter "#1100 Tmove" set to "0", the compensation will be accomplished in the following block.

(13) For the multiple C axis system, both C axis workpiece offset are rewritten with the workpiece offset input.

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
Maximum 80 tools
(2) Number of groups
Maximum 80 groups
(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

Life management data registration start

```
G10 L3 ;
P__ L__ N__ R__ ;   (First group)
T__ ;
T__ ;
P__ L__ N__ ;   (Next group)
T__ ;
T__ ;
```

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)
R	Remaining life (0 to 999999 min. or 0 to 999999 times) (Note) Only when the life prediction function is enabled ("#1277 ext13/bit1"="1"), the remaining life can be set with address R. The R command without the life prediction function causes the program error (P33).
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.

End of life management data registration

```
G11 ;
```

Program example

(1) Format

: T □□□□ 99 ;	Start use of □□□□ group tool
: :	
T □□□□ 88 ; : :	Cancel □□□□ group tool compensation (Equivalent to TΔΔ00: ΔΔ 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) If the remaining life data (R_) is omitted, the remaining life data for that group will be "0". When the remaining life data is "0", the "Life prediction" signal is not output for the designated tool. If N_ which specifies method is omitted, the method for that group will follow the basic specification parameter "#1106 Tcount". If the remaining life data (R_) is issued in larger value than life value (R_>L_), a program error (P35) will occur.
If the data L_ is omitted and the remaining life data (R_) is issued, a program error (P33) will occur.
(6) Programming with a sequence No. is not possible between G10 L3 and G11.
(7) If the usage data count valid signal (YC8A) is ON, G10 L3 cannot be commanded.
(Program error(P177))

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.

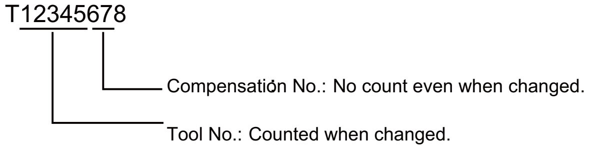
(Note) - The maximum value for the life is 999999 minutes.
- The data is displayed in a minute unit on the Tool Life Management screen.

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


T12345678
Compensation No.: No count even when changed.
Tool No.: Counted when changed.

<<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.

Re-selecting same tool group

When tool data of a tool with different compensation Nos. exists, the re-count M codes enable selecting the first compensation No.
Use the base specification parameter "#1108 TlrectM" to set the re-count M code.

〔Example of re-count M〕
When the re-count M code parameter "#1108 TlrectM" is "35":

(1) Tool selection when re-count M is not used:

Register 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	
	:		
	T0199 ;	Equivalent to T1703	
	:		
	T0199 ;	Equivalent to T1703	
		:	
(Group 1)		(Hereafter, the same until tool No. 17 reaches its life)	

(2) Tool selection when re-count M is used:

Register to group 1	Program	Tool selection	
T1701	T0199 ;	Equivalent to T1701	
	:		
T1702	T0199 ;	Equivalent to T1702	
	:		
T1703	T0199 ;	Equivalent to T1703	
	:		
T2104	M35 ;	Equivalent to T0188	→ Re-count M command
			(As MF is ON, FIN is returned)
	T0199 ;	Equivalent to T1701	
	:		
	T0199 ;	Equivalent to T1702	
	:		
	T0199 ;	Equivalent to T1703	
		:	
(Group 1)		(Hereafter, the same until tool No. 17 reaches its life)	

(Note 1) In the case of "number of uses-count type" life management, the usage data is counted up by one when cutting is carried out after re-count M code is commanded.
Therefore, in the case of the above example, count up is only once in a), but count up is made once again after M35 in b).
In the case of "time-count type" life management, the length of time during cutting is counted, thus the re-count M code command does not affect the usage data.

(Note 2) Among the settings of #1108 TlrectM, the setting 0 disables re-count M code, and 1, 2 and 30 are not allowed to use.

13

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
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

Straight cutting

G77 X/U__ Z/W__ F__ ;

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

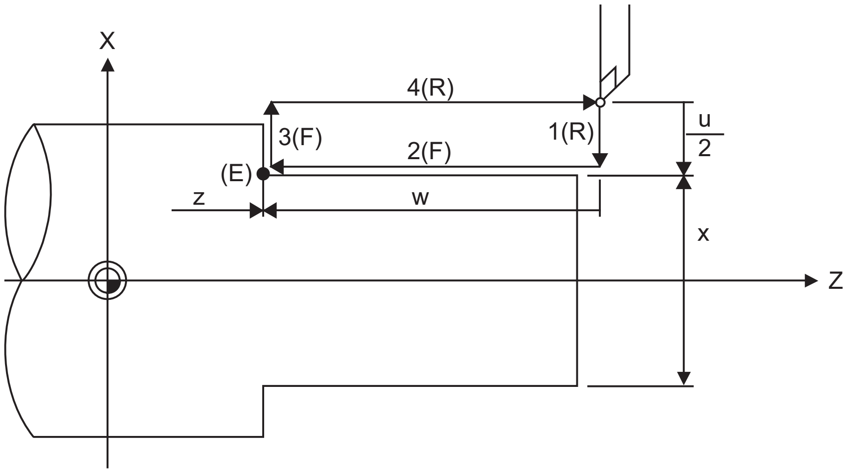
Taper cutting

G77 X/U__ Z/W__ R__ F__ ;

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

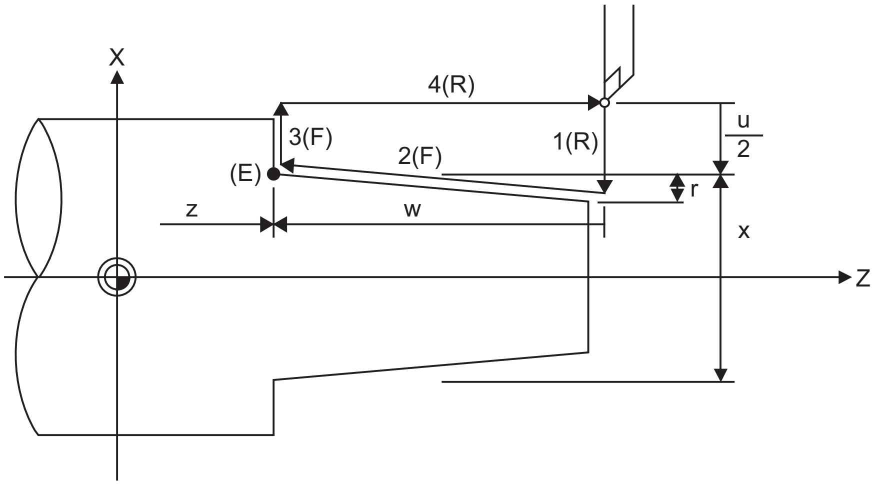
Detailed description

Straight cutting



(R) Rapid traverse
(F) Cutting feed
(E) End point coordinates

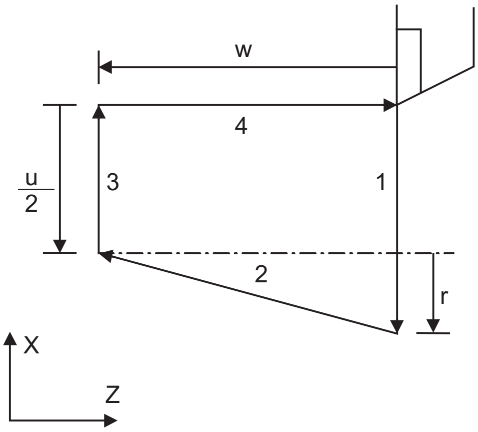
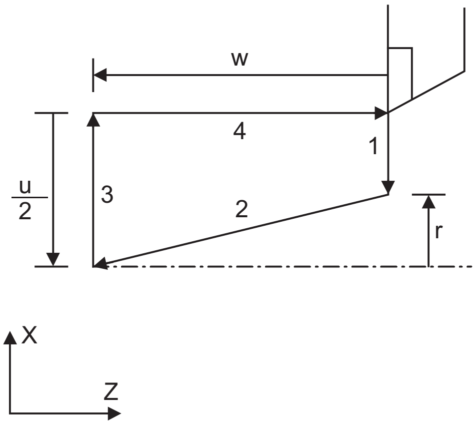
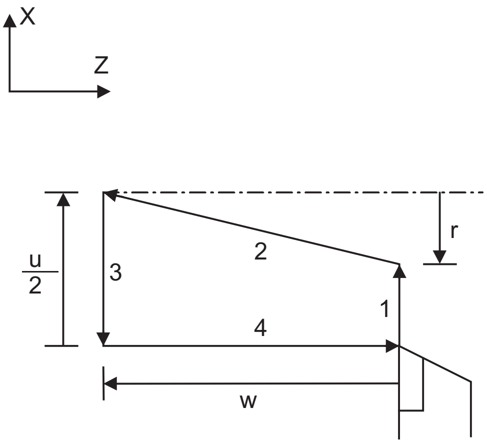
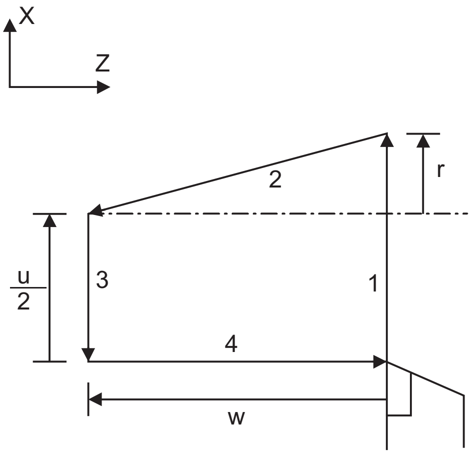
Taper cutting



(R) Rapid traverse
(F) Cutting feed
(E) End point coordinates

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.

(1) u < 0, w < 0, r < 0	(2) u < 0, w < 0, r > 0
	
(3) u > 0, w < 0, r < 0	(4) u > 0, w < 0, r > 0
	

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

$| u / 2 | \geqq | r |$

13.1.2 Thread Cutting Cycle ; G78

Function and purpose

Thread cutting cycle is a fixed cycle which performs straight and taper thread cutting.

(Note) The operation of thread cutting is same as the thread cutting command(G33).

Command format

Straight thread cutting

G78 X/U__ Z/W__ F/E__ Q__ ;

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
Q	Thread cutting start shift angle

Taper thread cutting

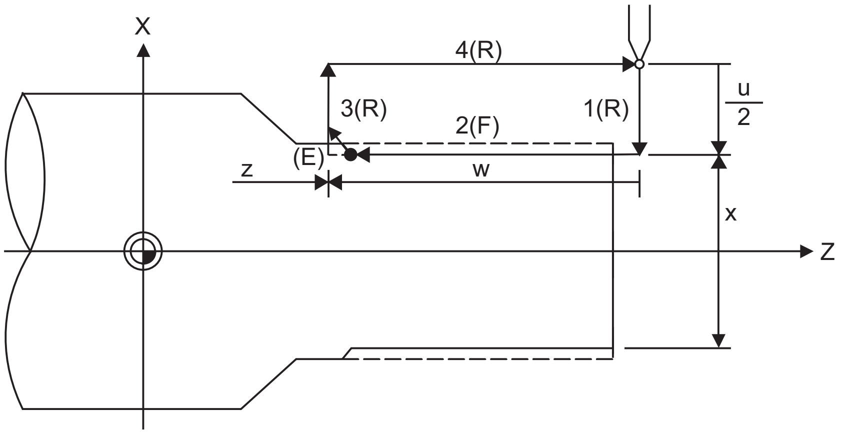
G78 X/U__ Z/W__ R__ F/E__ Q__ ;

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
Q	Thread cutting start shift angle

Detailed description

Straight thread cutting

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



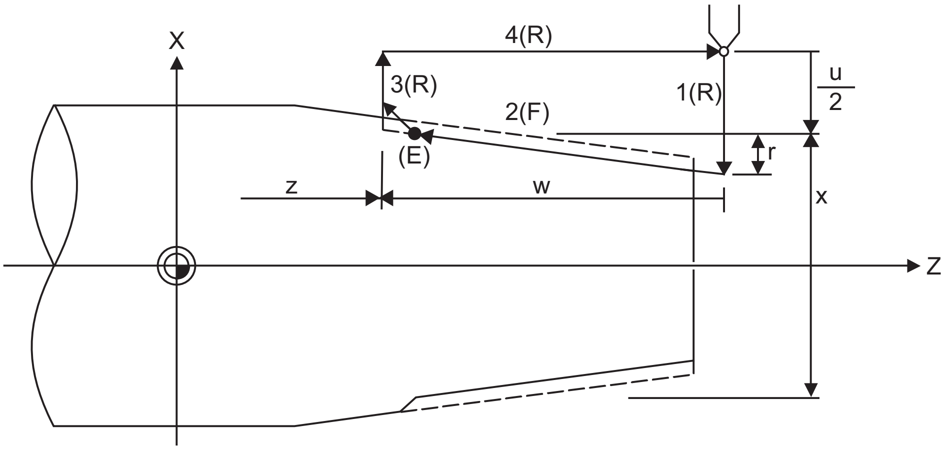
(R) Rapid traverse

(F) Thread cutting cycle

(E) End point coordinates

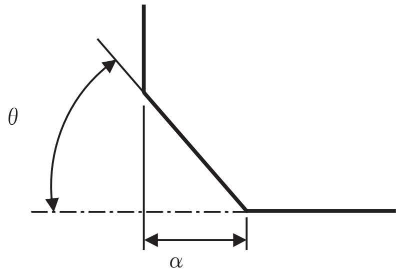
Taper thread cutting

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



(R) Rapid traverse
(F) Thread cutting cycle
(E) End point coordinates

(1) Details for chamfering

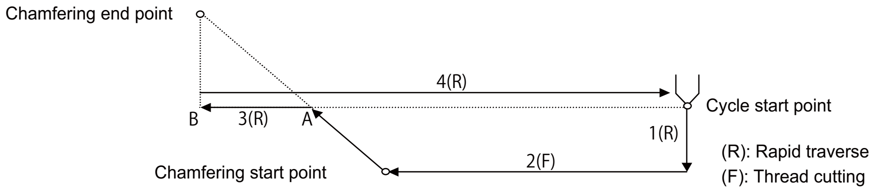
	α : Chamfering amount of thread cutting Chamfering amount is set in #8014. The setting range is 0 to 12.7. It can be set in 0.1L units. θ : Chamfering angle of thread Chamfering angle is set in #8015. The setting range is 0 to 89°. It can be set in 1° units.

If chamfering amount of thread cutting up is bigger than a thread lead length, a program error(P192) will occur before starting thread cutting.
Set a parameter to select the operation in thread cutting cycle when a result of the chamfering cutting up exceeds the cycle start point.

<#1270 ext06/bit4 = 0>
A program error(P192) will occur before thread cutting is started.

<#1270 ext06/bit4 = 1>
A program error(P192) will not occur.
After the cutting up is stopped and the thread cutting block ends at the cycle start point (A in the figure), the axis moves to the end coordinate (B in the figure) at rapid traverse feed.

Cutting up operation when #1270 ext06/bit4 is set to “1”

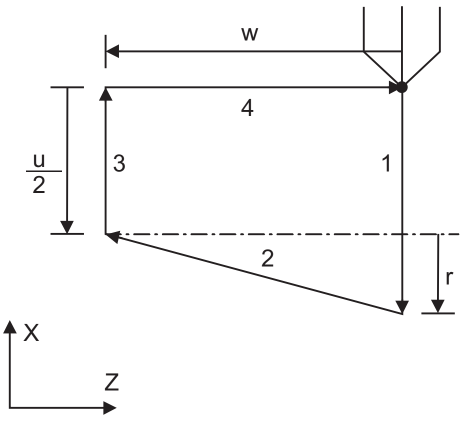


(2) When the feed hold function is applied during a thread cutting cycle, automatic operation will stop if it is applied when thread cutting is not being executed or when cutting command is issued but the axis is yet to move. If thread cutting is proceeding when the function is applied, the operation stops at the next movement completion position (completion of operation 3) of the thread cutting.

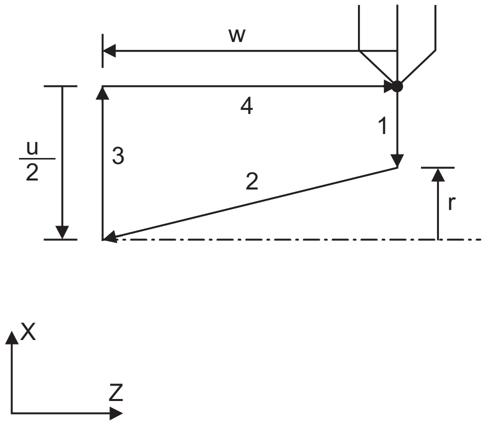
(3) The dry run valid/invalid status does not change during thread cutting.

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

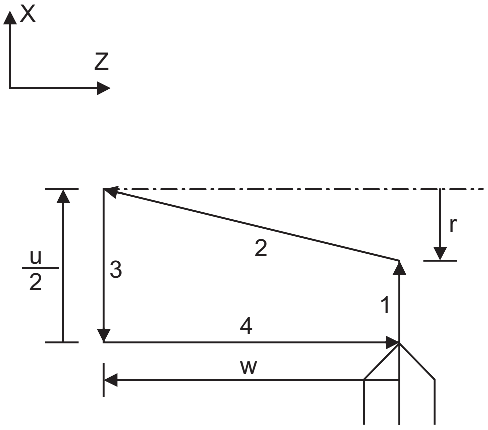
(a) $u < 0$, $w < 0$, $r < 0$



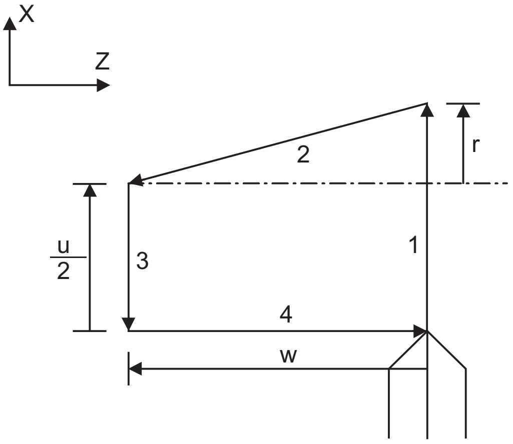
(b) $u < 0$, $w < 0$, $r > 0$



(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.

$$|u / 2| \geqq |r|$$

(Note 1) The thread cutting start shift angle is not modal. If there is no Q command with G78, this will be handled as "Q0".

(Note 2) If a value exceeding 360.000 is command in G78 Q, this will be handled as "Q360.000".

(Note 3) G78 cuts one row with one cycle. To cut two rows, change the Q value, and issue the same command.

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

Straight cutting

G79 X/U__ Z/W__ F__ ;

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

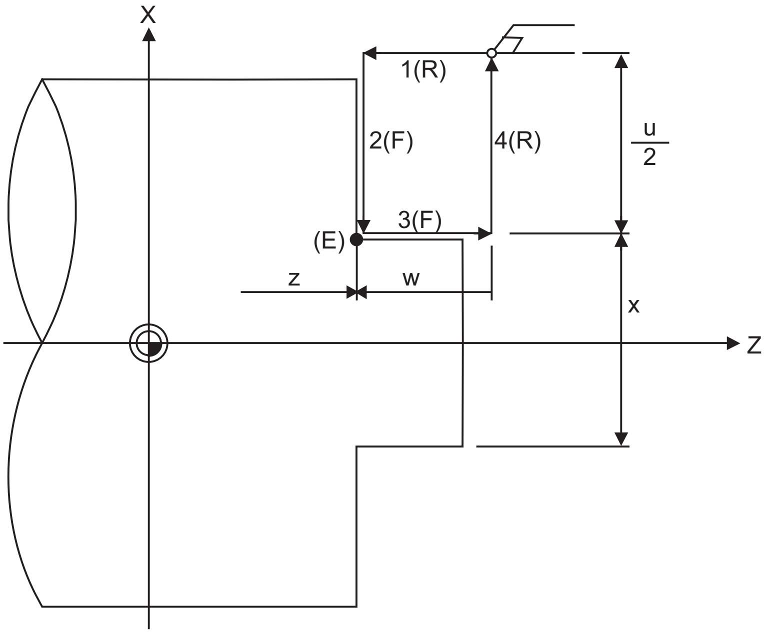
Taper cutting

G79 X/U__ Z/W__ R__ F__ ;

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

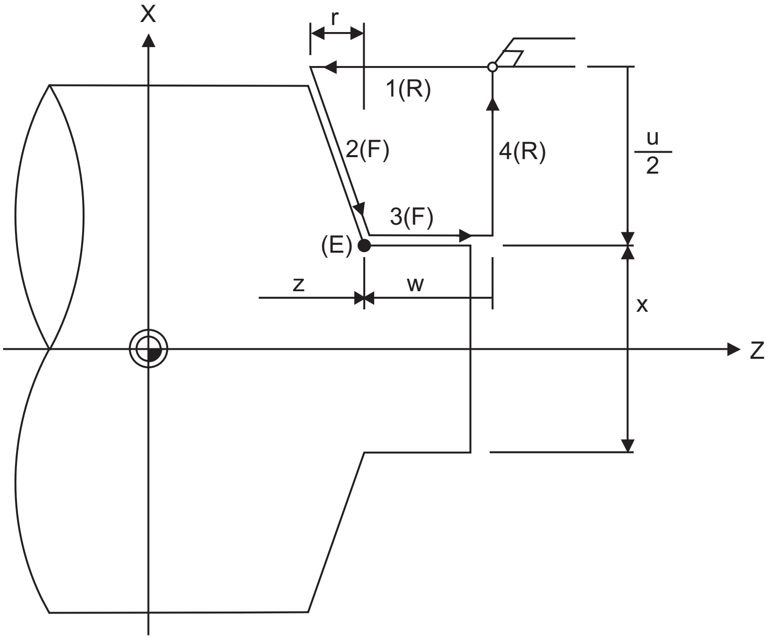


(R) Rapid traverse

(F) Cutting feed

(E) End point coordinates

Taper cutting



(R) Rapid traverse

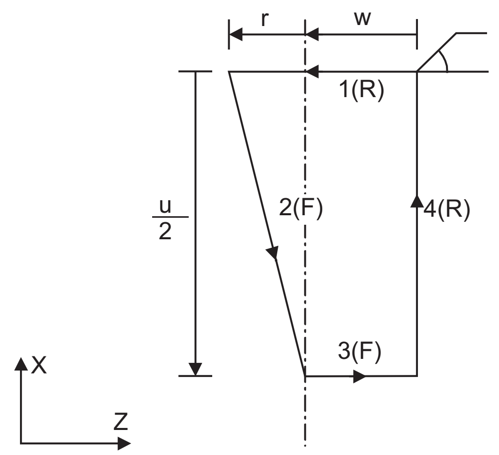
(F) Cutting feed

(E) End point coordinates

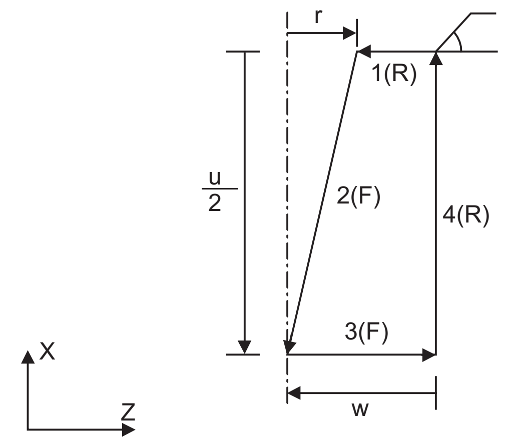
With a single block, the tool stops at the end points of operations 1, 2, 3 and 4.

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

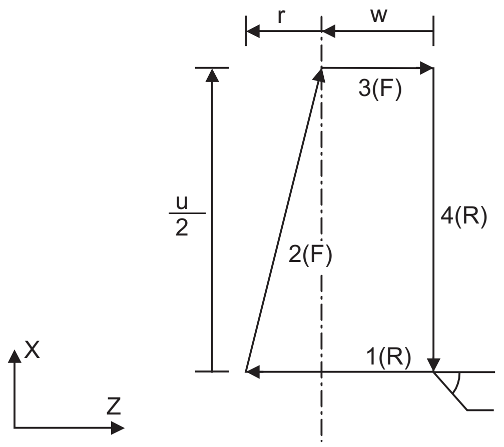
(1) u < 0, w < 0, r < 0



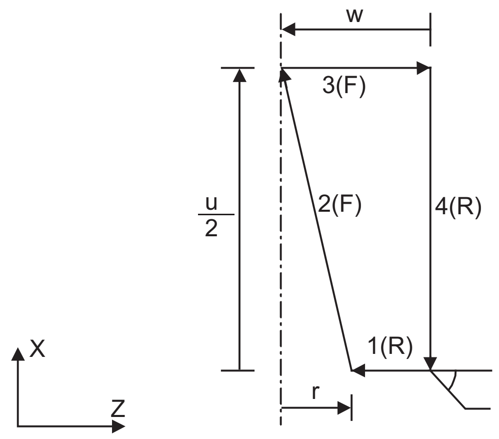
(2) u < 0, w < 0, r > 0



(3) u > 0, w < 0, r < 0



(4) u > 0, w < 0, r > 0



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

| w | ≧ | 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

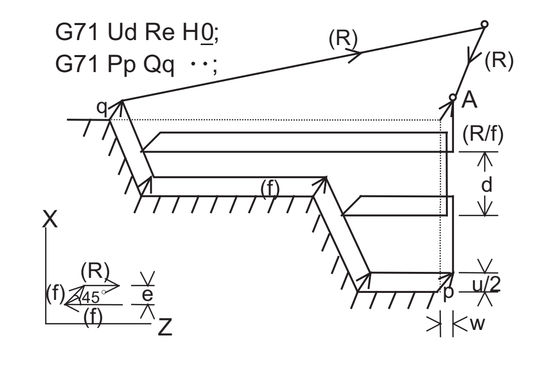
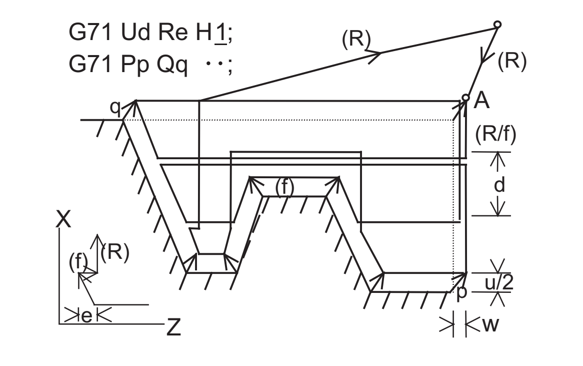
This function calls the finished shape program and, while automatically calculating the tool path, performs rough cutting in the longitudinal direction.

Command format

G71 Ud Re Hh ;
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 (modal) ... Reversible parameter "#8051 G71 THICK" is also available for setting Unit: μm/ 1/10000 inch radius value command
Re	Retract amount (modal) ... Reversible parameter "#8052 G71 PULL UP" is also available for setting Unit: μm/ 1/10000 inch radius value command
Hh	Pocket machining (modal) ....Reversible parameter "#8110 G71/G72 POCKET" is also available for setting 0: No pocket machining 1: Pocket machining
Aa	Finished shape program No. (Program being executed when omitted) Use a parameter to specify a 4- or 8-digit program No. starting with O.
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 (When omitted, X axis direction finishing allowance is handled as 0.) Unit: μm/ Follows 1/10000ich diameter/radius value command (#1019 dia).
Ww	Z axis direction finishing allowance (When omitted, Z axis direction finishing allowance is handled as 0.) Unit: μm/ 1/10000 inch radius value command
Ff	Cutting speed (When omitted, the cutting speed (modal) just before G71 will be applied)
Ss,Tt	Spindle command, tool command

Example of a finished shape without pocket	Example of a finished shape with pocket
	

(R) Rapid traverse (f) Cutting feed A : Rough cutting start point

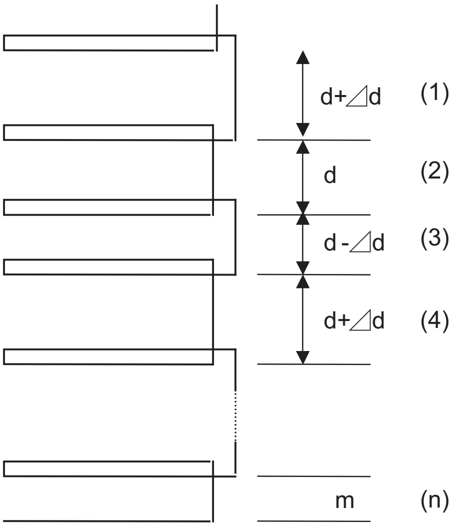
(Note) A reversible parameter enables to use parameter setting value without issuing a program command and also, the value can be changed by the program command.

Cut amount: Ud

(1) Designate the cut amount by Ud or parameter "#8051 G71 THICK".
Also, the amount can be changed for each cutting by setting the cut change amount (⊿d) using the parameter.

(2) Using the following parameters, select the response when a cut amount commanded in the program is deeper than the cutting depth of the finished shape.
Cut amount designation (#1271 ext07/bit7)
<ext07/bit7=0>
If a cut amount commanded in a program is deeper than the cutting depth of the finished shape (d < Δd), a program error (P204) occurred.
<ext07/bit7=1>
A program error will not occur even if a designated cut amount is deeper than the cutting depth of the finished shape, and the command will be executed as one cut.
Note that a cut amount must be between 0 and 99.999mm. A program error will occur if a value exceeds this range.

(3) At the end of the cutting, remaining part will be the cut amount, but if this part is smaller than the value set in the parameter, finishing rough cutting will be executed instead of cutting.
- Cut amount (d):
  [Machining parameter] "#8051 G71 THICK" 0 to 99.999mm
- Cutting change amount (Δd):
  [Machining parameter] "#8017 G71 DELTA-D" 0 to 99.999mm
- Minimum cut amount for final cut:
  [Machining parameter] "#8016 G71 MINIMUM" 0 to 99.999mm

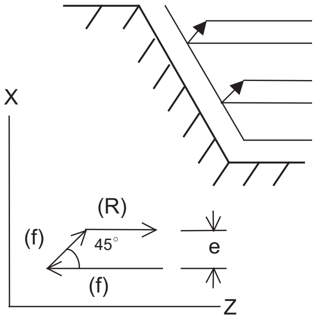
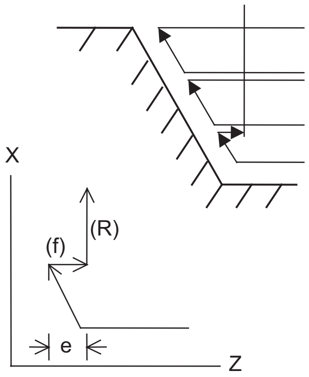
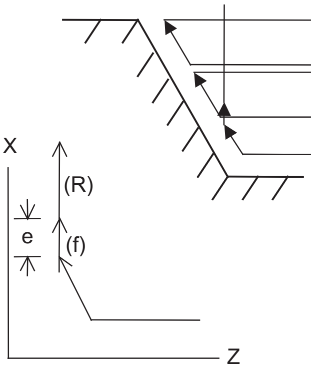


(1) to (n) : 1st cutting to last cutting
m : Remainder

Cutting method and retract amount: Re

(1) Designate the retract amount by Re or the parameter "#8052 G71 PULL UP" (0 to 99.999mm).

(2) The cutting method differs according to whether pocket machining is ON or OFF.
Pocket machining OFF.............Retracts in 45° direction in respect to workpiece
Pocket machining ON ....Traces shape
When the pocket machining is ON, cutting direction differs depending on "#1272 ext08/bit0" setting.
Cut-in amount designation [Base specification parameter] "#1272 ext08/bit0"
0 : The pull up direction is Z axis direction when pocket machining is ON
1 : The pull up direction is X axis direction when pocket machining is ON
However, when "#1272 ext08/bit0" is "1", pocket machining will be performed only when the first movement block after the finished shape start block, is designated to X and Z axes.

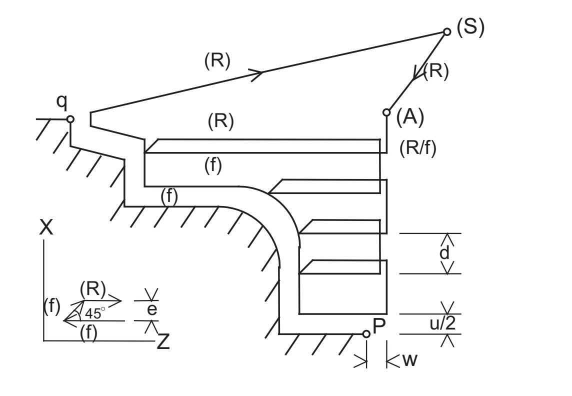
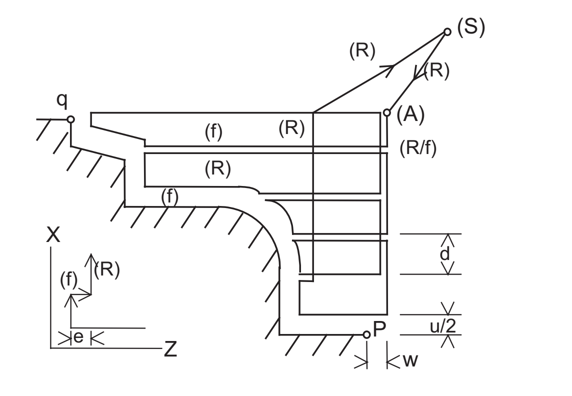
#1272 ext08	<When pocket machining is OFF>	<When pocket machining is ON>
bit0 = 0		
bit0 = 1		

Pocket machining: Hh

(1) Pocket machining ON/OFF can be set by Hh and a parameter.
#8110 G71/G72 POCKET
0: No pocket machining
1: Pocket machining
However, when "#1272 ext08/bit0" is "1", pocket machining will be performed only when the first movement block after the finished shape start block, is designated to X and Z axes.

Finished shape start block and finished shape end block:Aa,Pp,Qq

Designate the finished shape start block and finished shape end block by Aa,Pp,Qq.
If calling a subprogram numbered with O is enabled, a program number starting with O and specified by A command value is called.
When A command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of the command value by adding leading zeros.
In the following cases, a subprogram numbered without O and specified by A command value is called even when calling a subprogram numbered with O is enabled.
- When the digit number of A command value is greater than that of the parameter "#8129 subprogram number selection".
- When a specified subprogram number starting with O does not exist.
When A command is omitted, program command P, Q being executed, will be applied.
And when A command is omitted, the program will be executed in the order of G71 and then the program following Qq (finished shape end sequence No.)
A file name can be designated instead of address A by enclosing the file name in brackets <>. (The file name can have up to 32 characters including the extension.)
The maximum number of finished shape blocks is 200 including the blocks inserted with corner chamfer, corner R command and nose R command. A program error (P202) will occur if the numbers of blocks exceed the limit.

<When pocket machining is OFF>	<When pocket machining is ON>
	

(R) Rapid traverse (f) Cutting feed (S) Cycle command point
(A) Rough cutting start point

<When creating a finished shape program in a machining program other than the one currently being executed>	<When creating a finished shape program in a program currently being executed>
G71 A__ P__ Q__ A: Finished shape program No. P: Finished shape start sequence No. (From head of finished shape program when omitted) Q: Finished shape end sequence No. (To end of finished shape program or M99 when omitted)	G71 P__ Q__ P: Finished shape start sequence No. Q: Finished shape end sequence No.

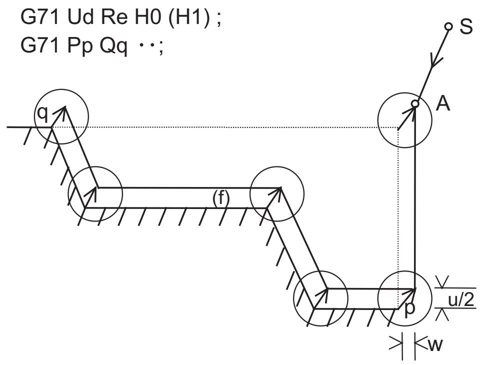
Program currently being executed
G0 X80.0 Z75.0 T0101; <- (a)
G71 U10. R3.; <- (b)
G71 A100 P10 Q20 U3. W1.5 F500 S1500; <- (c)
G70 A100 P10 Q20; <- (d)
 	
Shape program
O100
G28 XZ;
N10 G0 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
:
G1 Z42.0;
G1 X50.0 Z35.0;
N20 G1 X60.0;
N30 G0 X13.0 Z68.0;
 	
Program execution order (a)(b)(c)(d)
(N10-N20...Finished shape program)
	Program currently being executed
	G0 X80.0 Z75.0 T0101; <- (a)
	G71 U10. R3.; <- (b)
	G71 P10 Q20 U3. W1.5 F500 S1500; <- (c)
	N10 G0 X15.0 Z65.0;
	G1 Z55. F450;
	G1 X30.0;
	G3 X40.0 Z50.0 R5.0;
	G1 Z42.0;
	G1 X50.0;
	G1 X55.0 Z35.0;
	N20 G1 X60.0;
	G70 A100 P10 Q20; <- (d)
	
	Program execution order (a)(b)(c)(d)
	(N10-N20...Finished shape program)

	* G70 command: Finishing cycle The workpiece is finished following the finishing program, and then the axis returns to the G70 command position.

finishing allowance: Uu,Ww

When the finishing allowance is designated, the Uu/Ww part will be left uncut from the finished shape.
X axis finishing allowance ...The finishing allowance is left uncut in the rough cutting start point direction.

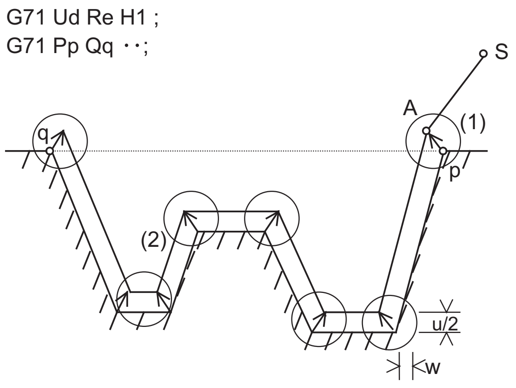
<Finishing allowance when pocket machining is OFF or when machining an open section when pocket machining is ON (finished shape start block X axis is at base)>



S : Cycle command point
A : Rough cutting start point
p : Finished shape start block
q : Finished shape end block

Z axis finishing allowance ...The finishing allowance is left uncut in the rough cutting start point direction. Note that if down cutting (block commanded in direction in which X axis faces hole base) is executed when pocket machining is ON and if cutting is started from the middle of the longitudinal direction, the Z axis finishing allowance direction will be reversed.

<Finishing allowance of mid-section when pocket machining is ON
(Finished shape start block X axis = finished shape end block X axis)>



S : Cycle command point
A : Rough cutting start point
p : Finished shape start block
q : Finished shape end block

The block that starts cutting from the middle of the longitudinal direction will have a reversed Z axis finishing allowance direction. (1)

(Note) During tool nose R compensation, programming must be done so that the X axis positions of p and q after the tool nose R compensation will be the same value. (Refer to "Cutting shape" (5).)

The block that executes down cutting has a reversed Z axis finishing allowance direction. (2)

Detailed description

Validity of pocket machining

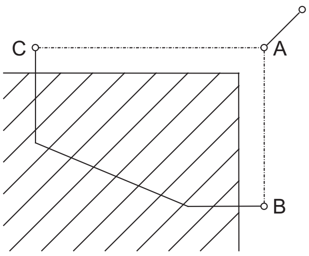
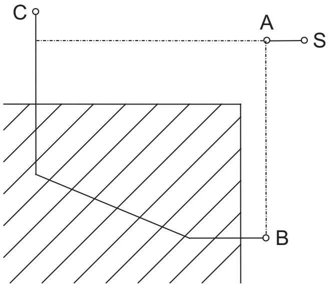
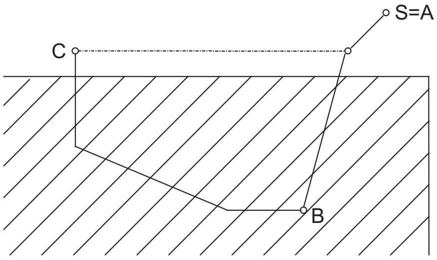
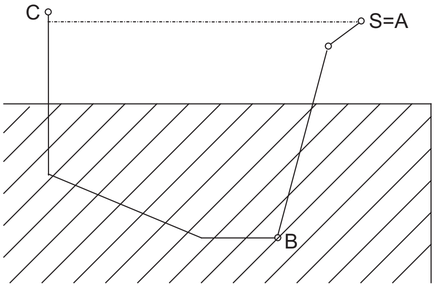
It is judged that there is a pocket section when down cutting (finished shape block in which previous movement block has no X axis movement command, or the X axis moves in the reverse direction of the hole base and then moves toward the hole base) is issued between the block following the finished shape start block and the end block.

(Note 1) During tool nose radius compensation, the validity of the pocket section is judged in respect to the finished shape after tool nose radius compensation.

(Note 2) If the finished shape has a pocket, program the pocket in a way that causes no interference between the tool and workpiece.

Machining of open section and machining of mid-section

The rough cutting cycle includes machining with one end of the workpiece is open, and machining which starts at a mid-section of a workpiece.
Machining of open section ....Possible when pocket machining is either OFF or ON
Machining of mid-section ....Possible only when pocket machining is ON
(Entire rough cutting cycle is handled as one pocket.)

	#1271 ext07/bit5 = 0	#1271 ext07/bit5 = 1
Open section		
Mid-section		

S : Cycle command point
A : Rough cutting start point
B : Finished shape start block
C : Finished shape end block

(Note) #1271 ext07/bit5: Selection of cut start position
0: Determined by finished shape program
1: Determined by cycle start point

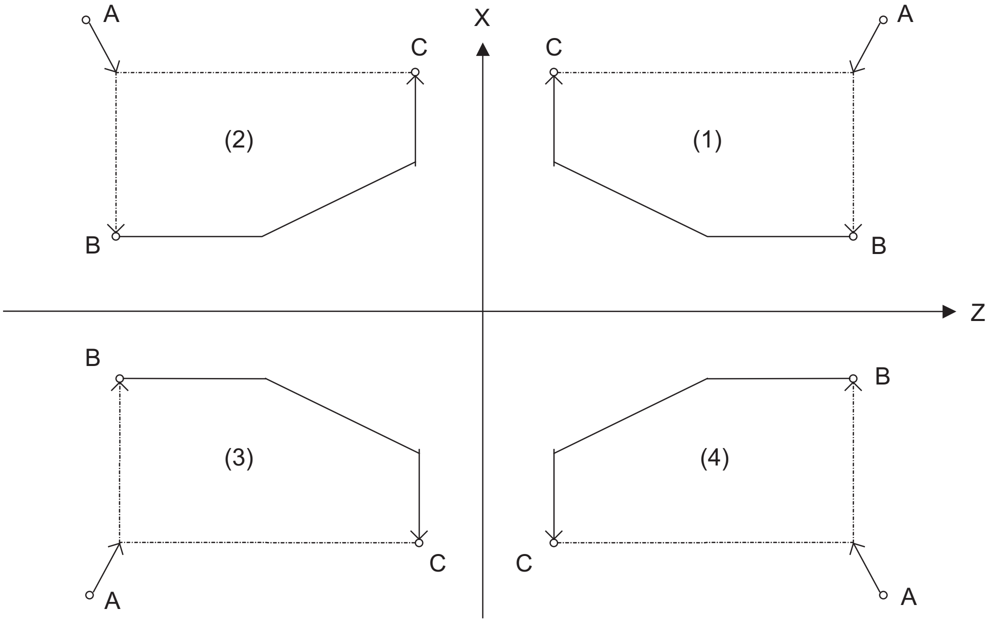
Rough cutting direction

<Rough cutting direction when pocket machining is OFF>

[Automatically determine according to finished shape (#1273 ext09/bit2=0)]
The rough cutting direction is determined in the following manner according to the finished shape.

C (finished shape end block) X axis > B (finished shape start block) X axis	(1) or (2)
C (finished shape end block) X axis < B (finished shape start block) X axis	(3) or (4)
C (finished shape end block) Z axis > B (finished shape start block) Z axis	(2) or (3)
C (finished shape end block) Z axis < B (finished shape start block) Z axis	(1) or (4)

A program error (P203) will occur if C (finished shape end block) X axis = B (finished shape start block) X axis or C (finished shape end block) Z axis = B (finished shape start block) Z axis.



A : G71 cycle command point
B : Finished shape start block
C : Finished shape end block

The finished shape program is A → B → C.

[Designate with program (#1273 ext09/bit2=1)]
The finishing allowance direction follows the sign of the finishing allowance designated in the program.

<Rough cutting direction when pocket machining is ON>

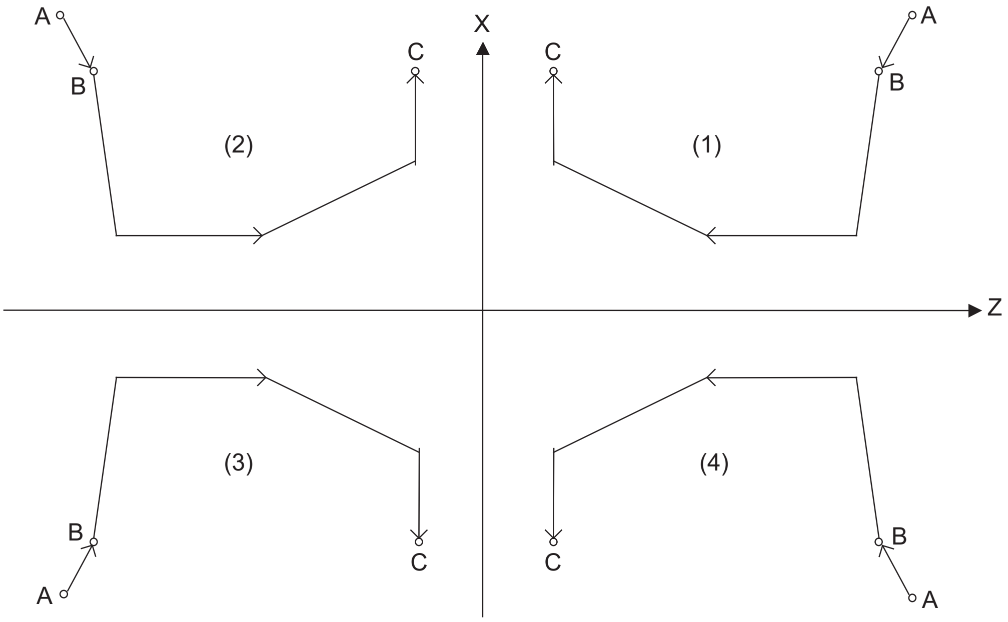
Select one of the followings.

[Automatically determine according to finished shape (#1273 ext09/bit2=0)]
The rough cutting direction is determined in the following manner according to the finished shape.

A (G71 cycle command point) X axis > B (finished shape start block) X axis, and B (finished shape start block) X axis <= C (finished shape end block) X axis	(1) or (2)
A (G71 cycle command point) X axis < B (finished shape start block) X axis, and B (finished shape start block) X axis >= C (finished shape end block) X axis	(3) or (4)
C (finished shape end block) Z axis > B (finished shape start block) Z axis	(2) or (3)
C (finished shape end block) Z axis < B (finished shape start block) Z axis	(1) or (4)

(Note 1) A program error (P203) will occur if C (finished shape end block) Z axis = B (finished shape start block) Z axis

(Note 2) When setting the B point X axis as the base, the method is the same as the "<Rough cutting direction when pocket machining is OFF>".



A : G71 cycle command point
B : Finished shape start block
C : Finished shape end block

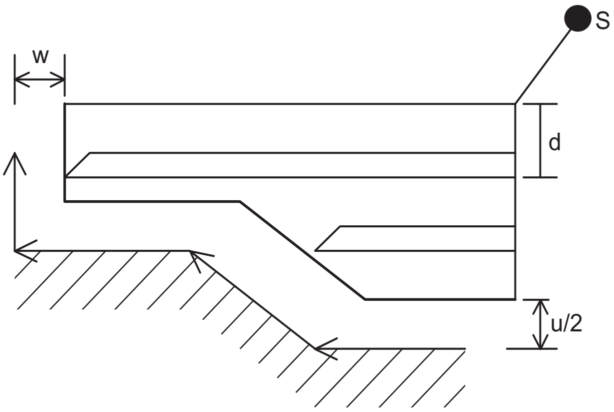
The finished shape program is A → B → C.

[Designate with program (#1273 ext09/bit2=1)]
The finishing allowance direction follows the sign of the finishing allowance designated in the program.

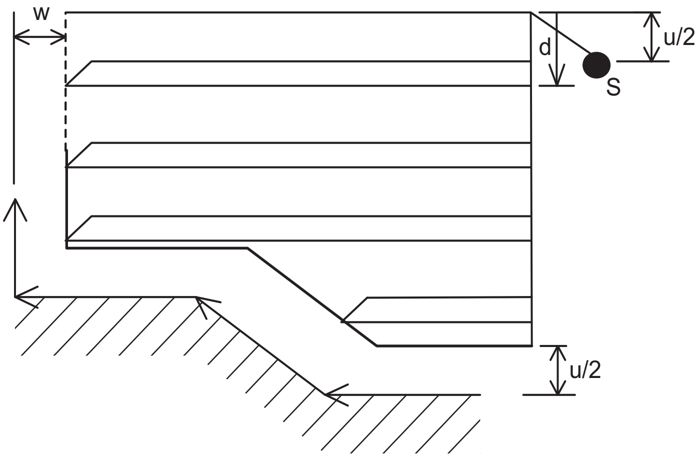
Finished shape

Selection of cut start position (#1271 ext07/bit5)
The cut start position is calculated from the final position of the finished shape program, but this can be changed to the cycle start point.

<ext07/bit5 = 0>



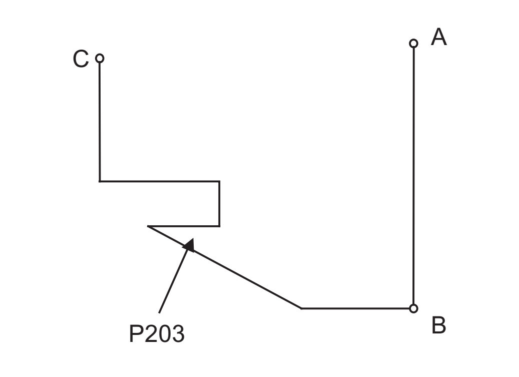
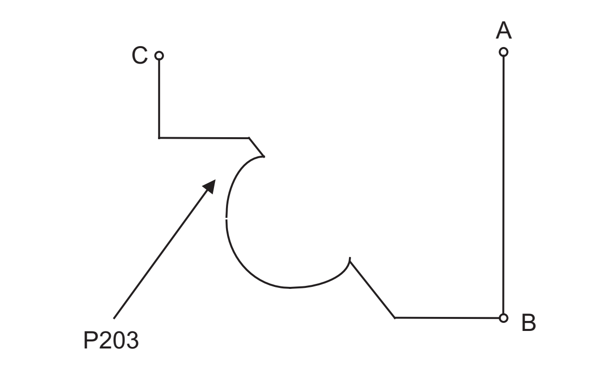
<ext07/bit5 = 1>



S : Cycle start point
u/2, w : Finishing allowance
d : First cut

<Finished shape in Z axis direction when pocket machining is OFF>

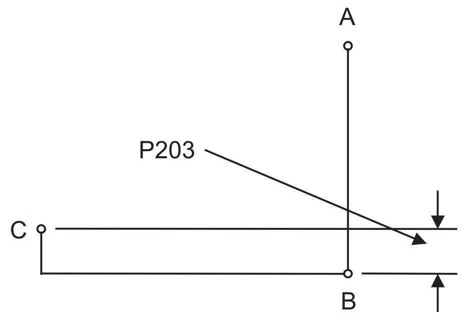
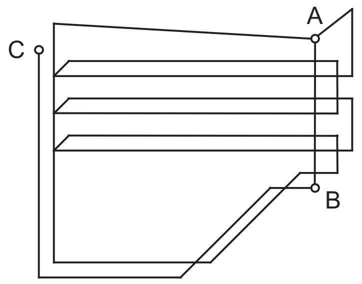
The Z axis direction finishing allowance must always change monotonously (only increment, or only decrement).
A program error (P203) will occur if the shape is illegal.
However, the program error can be avoided in the case of a minute inversion by setting a tolerance error range.

<Example of illegal Z axis direction when pocket machining is OFF>	
Z axis direction reverses (program error (P203))	Z axis direction reverses at middle of arc (program error (P203))
	

The program error does not occur and machining continues as long as a reversed travel distance of Z axis is within the tolerance error range (#8060 G71 ERR).

<Finished shape in X axis direction when pocket machining is OFF>

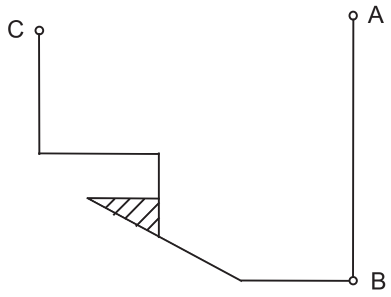
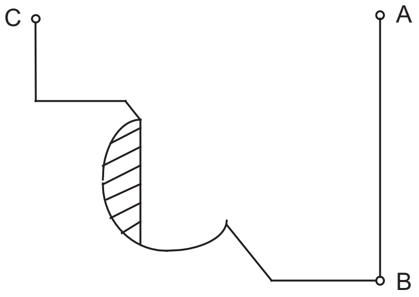
As a principle, the X axis direction finishing allowance must change monotonously (increment or decrement).
At sections that do not change monotonously, the rough cutting cycle will not be executed at the indent section (pocket), but the finishing allowance will be traced as commanded.
When programming, be aware that if the pocket's X axis depth is not less than the cut amount, the cutting load will increase at the pocket section.

<Example of illegal X axis direction when pocket machining is OFF>	
Smaller than cut amount d (Program error (P203))	There is a pocket when pocket machining is OFF. (An error will not occur, but the pocket section will be rough cut with one cut.)
	

A : G71 cycle command point
B : Finished shape start block
C : Finished shape end block

<Finished shape in Z axis direction when pocket machining is ON>

The Z axis direction finishing allowance must always change monotonously (only increment, or only decrement).
Sections that do not change monotonously will have a cover.

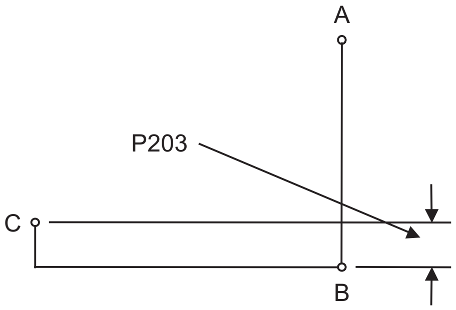
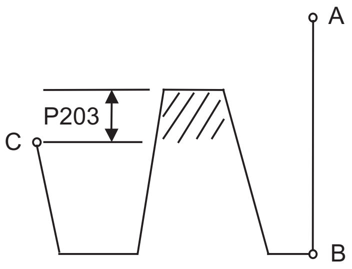
<Example of section where Z axis direction does not change monotonously when pocket machining is ON>	
The Z axis direction reverses, so the finished shape is created with a cover at the shaded section. (The shaded section is not cut)	The Z axis direction reverses in the middle of the arc, so the finished shape is created with a cover at the shaded section. (The shaded section is not cut)
	

A : Cycle command point
B : Finished shape start block
C : Finished shape end block

The shape is changed (cover) even when the reversed travel distance of Z axis is within the tolerable error range (#8060 G71 ERR).

<Finished shape in X axis direction when pocket machining is ON>

As a principle, the finished shape in the X axis direction must change monotonously (increment or decrement). The direction must reverse only at the indent shape (pocket).
During pocket machining, the block in finishing, must not exceed the C (finished shape end block) X axis position.
If it does, a program error (P203) will occur.

<Example of illegal X axis direction when pocket machining is ON>	
Smaller than cut amount d (Program error (P203))	Block in finishing is higher than finished end point block (Program error (P203))
	

A : Cycle command point
B : Finished shape start block
C : Finished shape end block

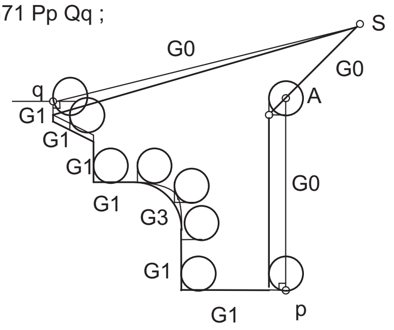
Finished shape during tool nose R compensation

Selection of tool nose R compensation (#1271 ext07/bit6)

<ext07/bit6=0>
If there is a G71 command during tool nose R compensation, the G71 cycle command point will be at the position where tool nose R compensation is temporarily canceled. The finished shape will be created while tool nose R compensation is applied.

<Example of tool nose No. 3 (pocket machining OFF) ... Machining of open section>

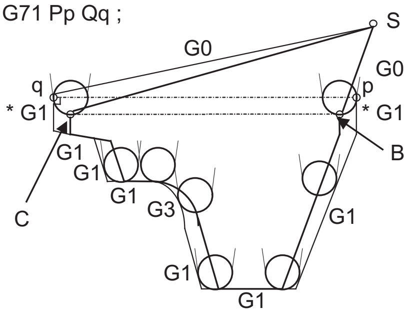
```
G71 Ud Re H0 ;
G71 Pp Qq ;
```



S : Cycle command point A : Rough cutting start point

<Example of tool nose No. 8 (pocket machining ON) ... Machining of mid-section>

```
G71 Ud Re H1 ;
G71 Pp Qq ;
```



S : Cycle command point
B : Finished shape start block after nose R compensation
C : Finished shape end block after nose R compensation

(Note) To machine a mid-section while pocket machining is ON, programming must be done so that the X axis positions of the finished shape start block and finished shape end block are the same after tool nose R compensation is applied.

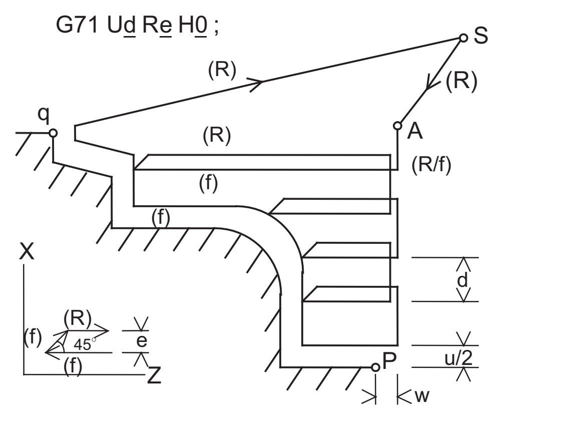
Precautions for machining mid-section
The mid-section will be machined when the X axis positions of the finished shape start block and finished shape end block are the same after nose R compensation.
Programming must be done so that only the X axis moves (more than nose R diameter) in the block with * mark in the above figure.

<ext07/bit6=1>
Rough cutting will be executed according to the finished shape program without the tool nose R compensation.

Program example

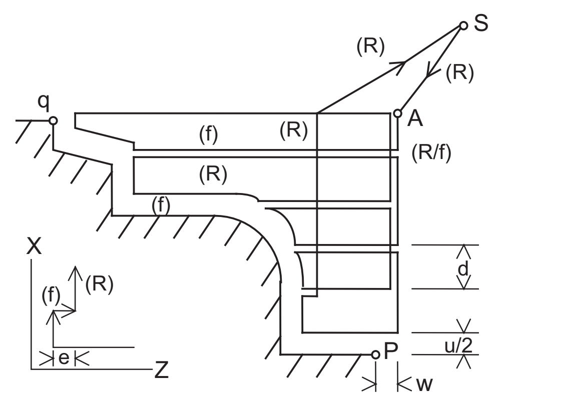
Machining of open section (example with pocket machining OFF)

```
G0 X80.0 Z75.0 T0101;                  <- (1)
G71 U10. R3. H0;                       <- (2)
G71 P10 Q20 U3.W1.5 F500 S1500;  <- (3)
N10 G0 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
G1 Z42.0;
G1 X50.0;
G1 X55.0 Z35.0;
N20 G1 X60.0;
G70 P10 Q20;                           <- (4)

Program execution order (1)(2)(3)(4)
(N10-N20...Finished shape program)
```



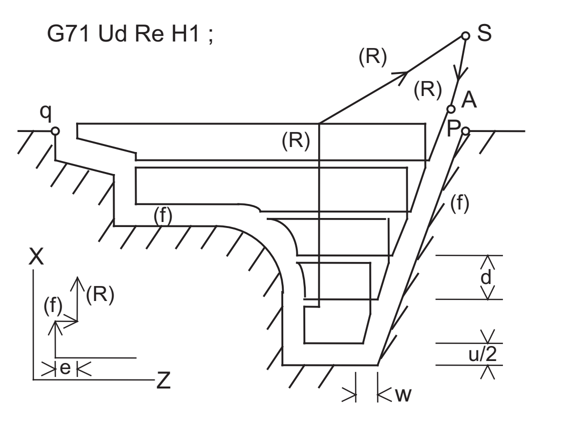
Machining of open section (example with pocket machining ON)

```
G0 X80.0 Z75.0 T0101;                 <- (1)
G71 U10. R3. H1;                      <- (2)
G71 P10 Q20 U3.W1.5 F500 S1500; <- (3)
N10 G0 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
G1 Z42.0;
G1 X50.0;
G1 X55.0 Z35.0;
N20 G1 X60.0;
G70 P10 Q20;                          <- (4)

Program execution order (1)(2)(3)(4)
(N10-N20...Finished shape program)
```



Machining of mid-section (example with pocket machining ON)

```
G0 X80.0 Z75.0 T0101;                 <- (1)
G71 U10. R3. H1;                      <- (2)
G71 P10 Q20 U3.W1.5 F500 S1500; <- (3)
N10 G0 X60.0 Z73.0;
G1 X15.0 Z65.0;
G1 Z55. F450;
G1 X30.0;
G3 X40.0 Z50.0 R5.0;
G1 Z42.0;
G1 X50.0;
G1 X55.0 Z35.0;
N20 G1 X60.0;
G70 P10 Q20;                          <- (4)

Program execution order (1)(2)(3)(4)
(N10-N20...Finished shape program)
```



S : Cycle command point	A : Rough cutting start point
(R) Rapid traverse	(f) Cutting feed

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

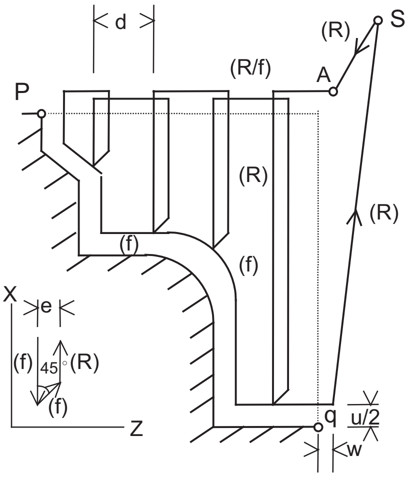
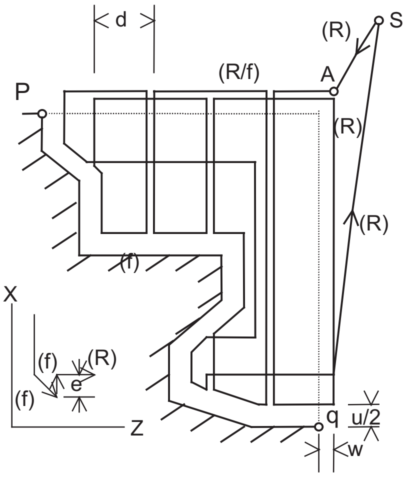
This function calls the finished shape program and, while automatically calculating the tool path, performs rough cutting in the face direction.

Command format

G72 Wd Re Hh ;
G72 Ae Pp Qq Uu Ww Ff Ss Tt ; ... Face Rough Cutting Cycle

This fixed cycle requires two blocks.
However, when using a value set by a parameter, the first block can be omitted.

Wd	Cut amount (modal) ...Reversible parameter "#8051 G71 THICK" is also available for setting Unit: μm/ 1/10000 inch radius value command
Re	Retract amount (modal) ...Reversible parameter "#8052 G71 PULL UP" is also available for setting Unit: μm/ 1/10000 inch radius value command
Hh	Pocket machining (modal) ...Reversible parameter "#8110 G71/G72 POCKET" is also available for setting 0: No pocket machining 1: No pocket machining
Aa	Finished shape program No. (Program being executed when omitted) Use a parameter to specify a 4- or 8-digit program No. starting with O.
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 (When omitted, X axis direction finishing allowance is handled as 0.) Unit: μm/ Follows 1/10000ich diameter/radius value command (#1019 dia).
Ww	Z axis direction finishing allowance (When omitted, Z axis direction finishing allowance is handled as 0.) Unit: μm/ 1/10000 inch radius value command
Ff	Cutting speed (When omitted, the cutting speed (modal) just before G72 will be applied)
Ss,Tt	Spindle command, tool command

Example of a finished shape without pocket	Example of a finished shape with pocket
G72 Wd Re H0; G72 Pp Qq ‥;  d (R) S (R/f) A P (R) (R) (f) (f) X e (f) 45° (R) (f) Z q u/2 w	G72 Wd Re H1; G72 Pp Qq‥ ;  d (R) S (R/f) A P (R) (R) (f) X (f) (R) e (f) Z q u/2 w

S : Cycle command point

A : Rough cutting start point

(R) Rapid traverse

(f) Cutting feed

(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

Refer to "Longitudinal Rough Cutting Cycle" for details of the functions.

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

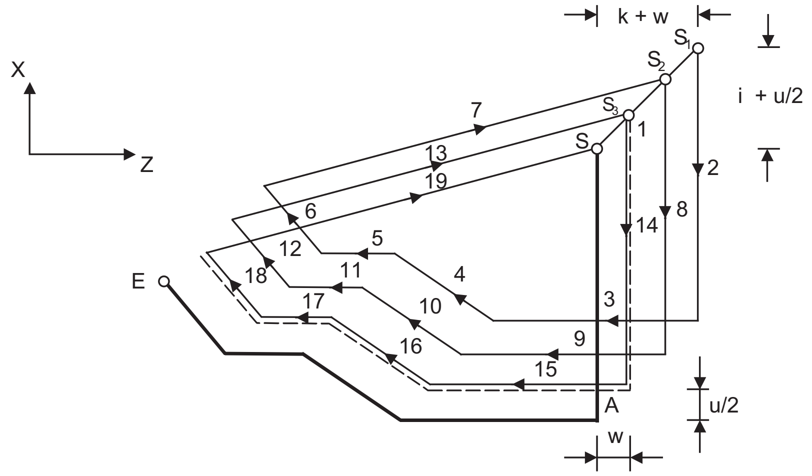
This function calls the finished shape program, automatically calculates the tool path and performs rough cutting while cutting the workpiece into the finished shape.

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 Wk Rd	X-axis direction cutting allowance i Z-axis direction cutting allowance k Number of divisions d	- Cutting allowances when P, Q commands are not issued - Modal data - Sign is ignored - Radius designation applies to the cutting allowance.
Aa	Finished shape program No.	(program being executed when omitted) A file name can be designated instead of address A by enclosing the file name in brackets <>. (The file name can have up to 32 characters including the extension.)
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 Ww	X-axis direction finishing allowance u Z-axis direction finishing allowance w	- Cutting allowance when P, Q commands are issued. - 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.
FfSsTt	Cutting feedrate (F function) Spindle speed (S function) Tool command (T 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.



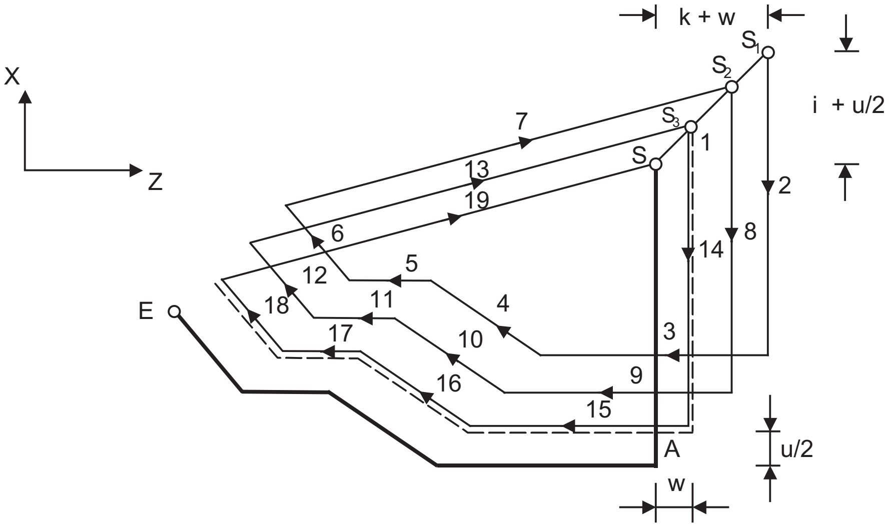
S : G73 cycle command point
A : Finished shape start block
E : Finished shape end block

(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.

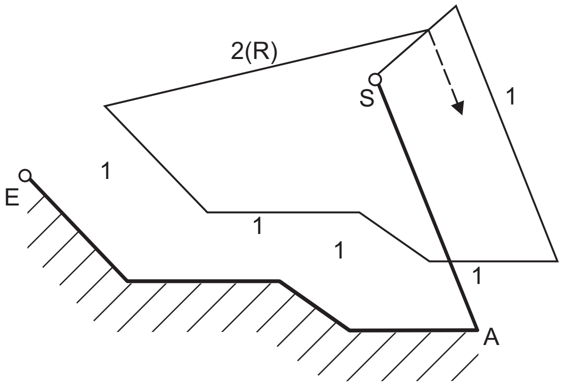


S : G73 cycle command point
A : Finished shape start block
E : Finished shape end block

(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

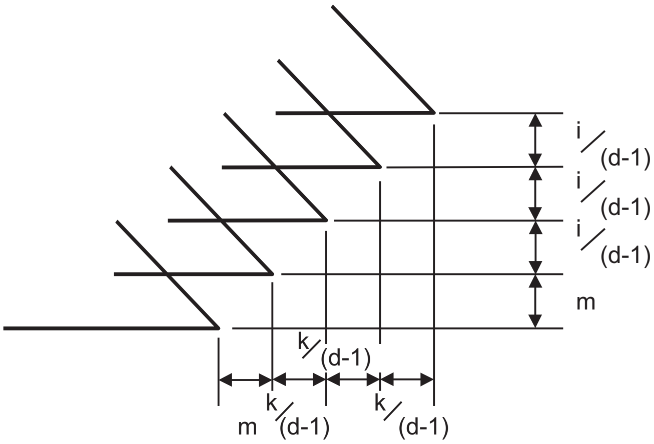
Cut amount

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

<#1271 ext07/bit6 = 0>

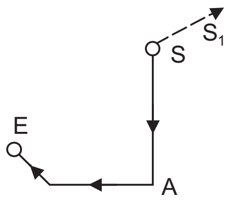
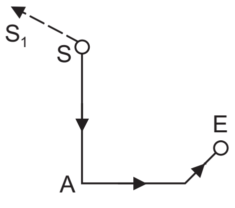
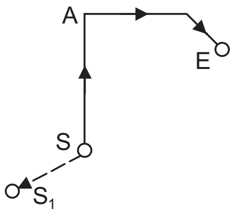
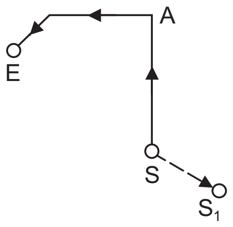
When this cycle is commanded with the tool nose R compensation mode still in force, the compensation is temporarily canceled immediately before this cycle and started at the head block of the finished shape program. So the compensation is applied to the finished shape program covered by this cycle and this cycle is executed for the compensated shape.

<#1271 ext07/bit6 = 1>

Like G71 and G72 cycle, rough cutting is carried out to the shape which tool nose radius compensation is not carried out.

Cutting direction

Determined according to finish shape (#1273 ext09/bit2=0)
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

Determined according to commanded finish allowance/cutting allowance (ext09/bit2=1)
The shift direction for cutting follows the sign of the cutting allowance, etc., designated in the program.

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

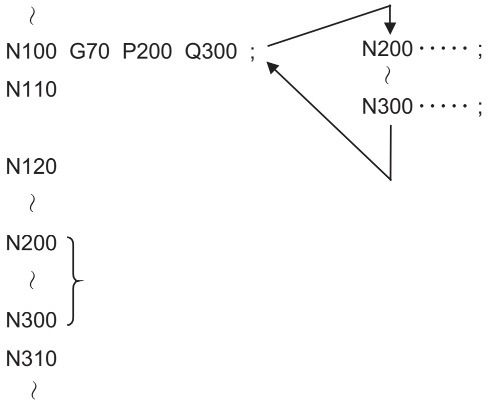
A	Finished shape program No. (Program being executed when omitted)
P	Finished shape start sequence No. (Head of program when omitted)
Q	Finished shape end sequence No. (To the end of program when omitted) If M99 is commanded before Q, the program will end at M99.

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.

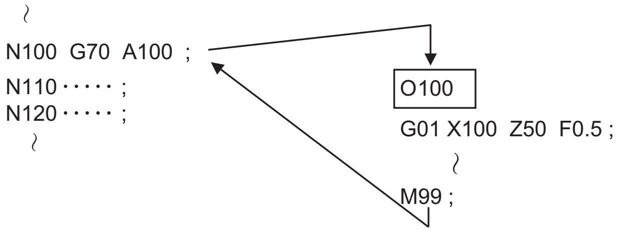
(Example 1) When a sequence No. is designated



```
  ~
N100  G70  P200  Q300  ;        N200 · · · · ·  ;
N110                               ~
                                N300 · · · · ·  ;

N120
  ~
N200 ┐
  ~  ├
N300 ┘
N310
  ~
```

N200-N300 ... Finished shape program

(Example 2) When a program No. is designated



```
  ~
N100  G70  A100  ;
N110 · · · · ·  ;                 O100
N120 · · · · ·  ;                 G01 X100  Z50  F0.5 ;
  ~                                    ~
                                  M99 ;
```

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.

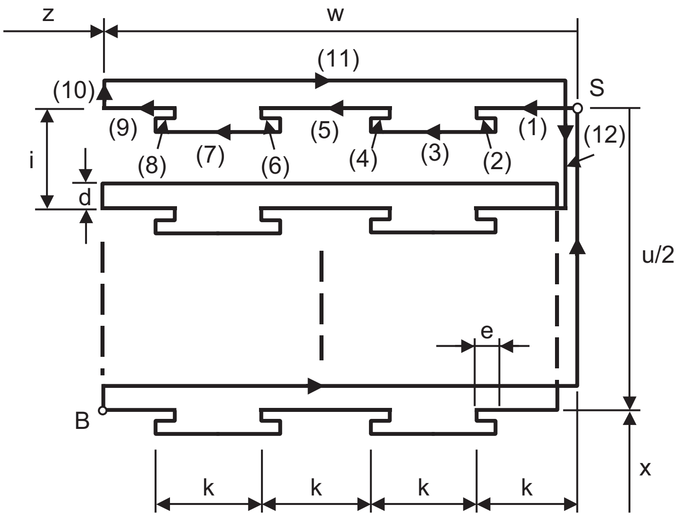
Command format

```
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) ... Reversible parameter "#8056 G74 RETRACT" is also available for setting
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

(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.



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, Re setting value will be the return amount. 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

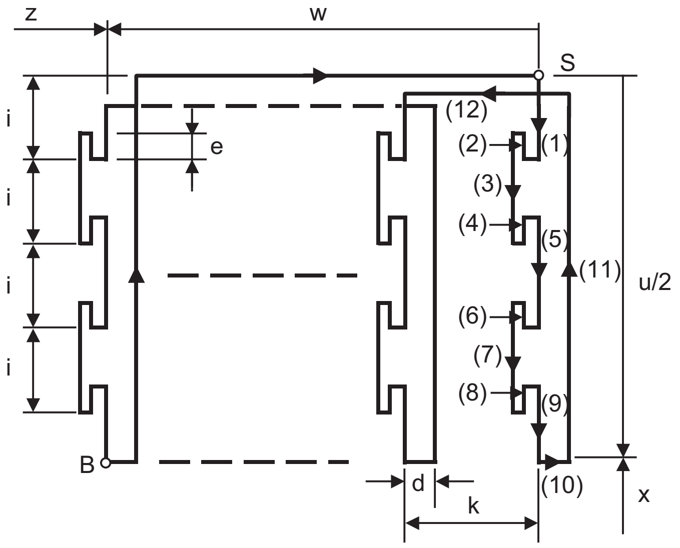
The G75 fixed cycle automatically performs grooving in the longitudinal direction of the workpiece by commanding the coordinates of the groove end point, cut amount, cutter shift amount and cutter escape at the bottom of the cut.

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) ... Reversible parameter "#8056 G74 RETRACT" is also available for setting
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.

(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) 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, Re setting value will be the return amount. 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.
With compound thread cutting cycle,chamfering can be operated. Refer to "Thread cutting cycle ; G78" for details on chamfering.
(Note) The operation of thread cutting is same as the thread cutting command(G33).

Command format

```
G76  Pmra  QΔdmin  Rd ;
G76  X/U  Z/W  Ri  Pk  QΔd  Fl ; ... Compound thread cutting cycle
```

This fixed cycle requires two blocks.
However, if "#1222 aux06/bit5" = "1" and "#1265 ext01/bit0" = "0" the 1st block can be omitted.
If the 1st block is omitted, the value set with the parameter is used.

Address		Significance
P	m	Number of cutting passes for finishing: 00 to 99 (times) (modal) (Reversible parameter "#8056 G76 TIMES" is also available for setting)
	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) (Reversible parameter "#8059 G76 ANGLE" is also available for setting) 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.
Q	Δdmin	Minimum cut amount If the calculated cut amount is smaller than Δdmin, it is clamped by Δdmin. (When omitted, operations differ depending on #1222/bit4 setting) (Note 6)
R	d	Finishing allowance 0 to 9999 (μm) (modal) (Reversible parameter "#8057 G76 LAST-D" is also available for setting)
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, Q 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.

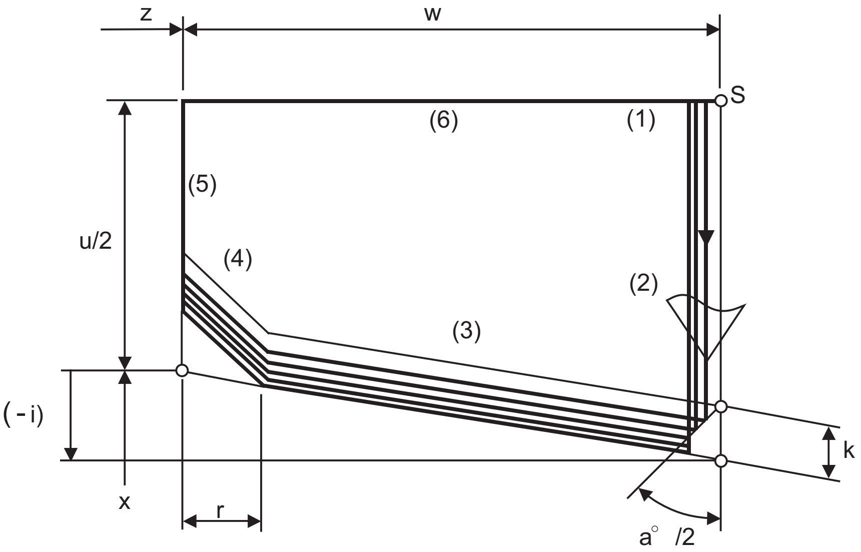
(Note 5) A program error (P204) will occur in the following cases.
- When "a" is outside the rated value
- When both or one of the X and Z commands is not issued, or when the start and end point coordinates coincides in both or one of the X and Z commands.
- When the thread is larger than the movement of the X axis to the thread bottom

(Note 6) The precautions for the thread cutting cycle (G78) should be observed.

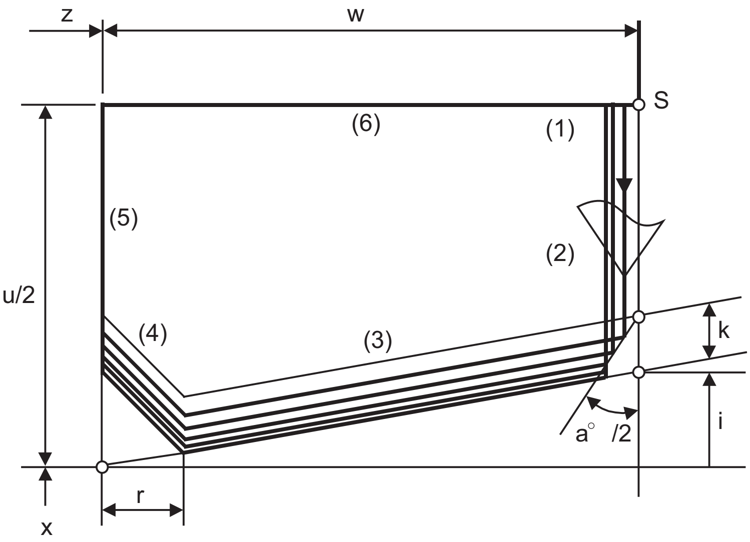
Detailed description

1 cycle configuration

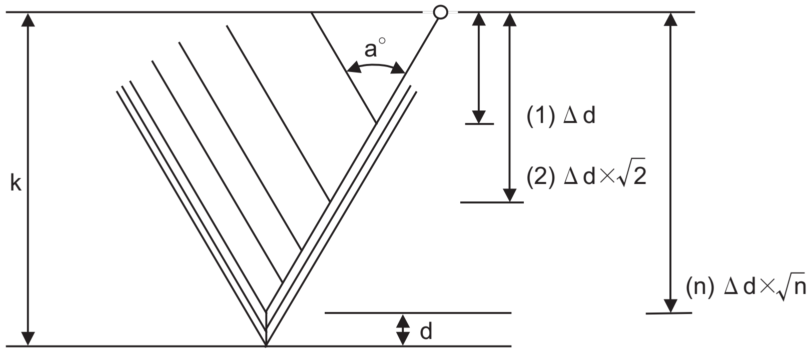
The tool moves at rapid traverse for operations (1), (2), (5) and (6) in the cycle and at the cutting feed based on the F designation for operations (3) and (4) during 1 cycle.



When Ri is negative



When Ri is positive

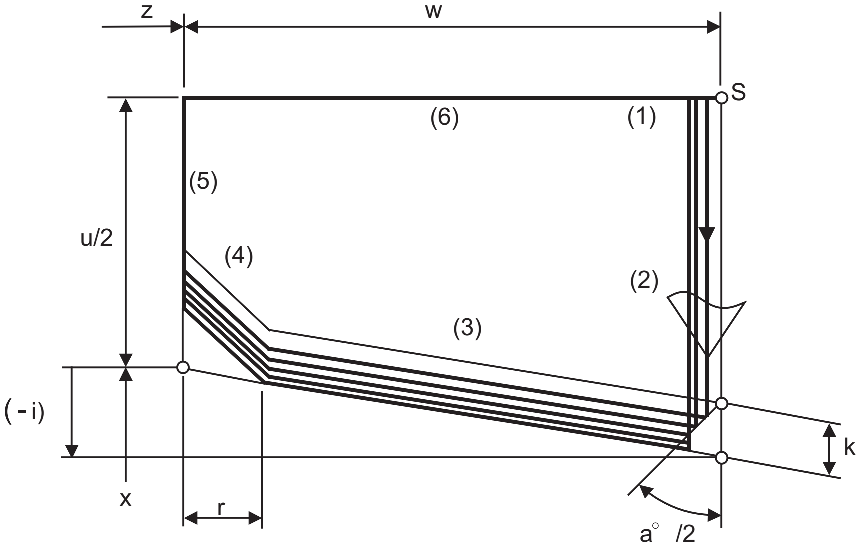


k : Thread height
d : Finishing allowance (cut "m" times)
(1) to (n) : 1st cutting to nth cutting

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.

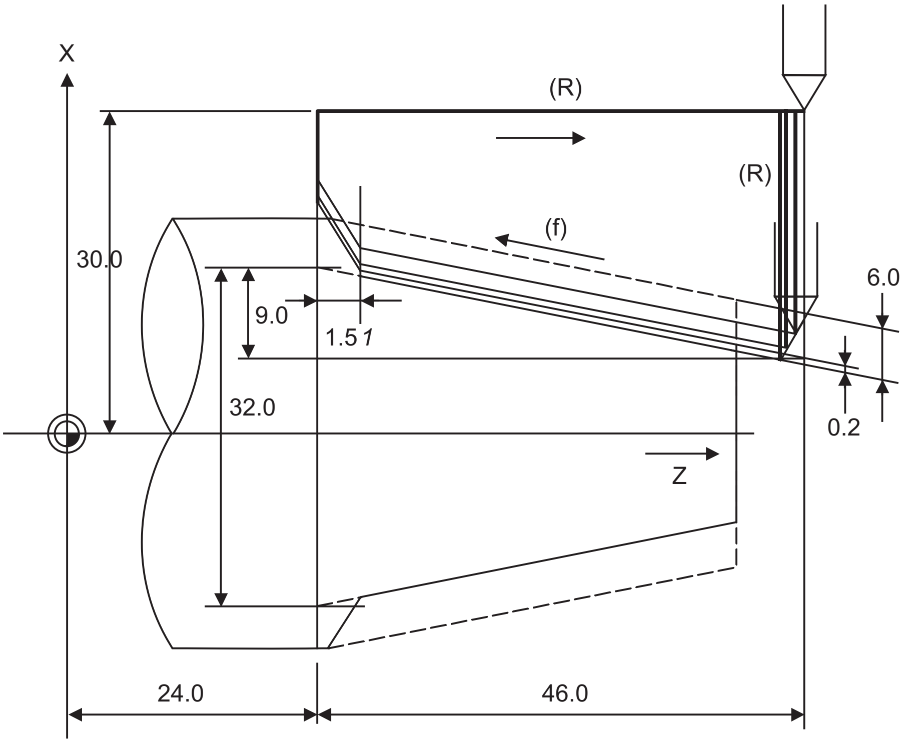
(2) The tool stops upon completion of operations (1), (4) and (5) in the figure, if the following operations are conducted during the G76 command execution.
- When the automatic operation mode is switched to another automatic operation mode
- When automatic operation is changed to manual operation
- When single block operation is conducted



When Ri is negative

(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 Q3.5 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, MDI or tape mode.

(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 200 blocks. If this number is exceeded, program error (P202) will occur.
If the cutting start position is the cycle start position (#1271 ext07/bit5 =1), the finished shape program can have up to 199 blocks. (When the cycle start point is located out of the finished shape program's final position.)

(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.

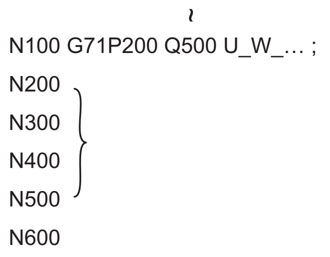
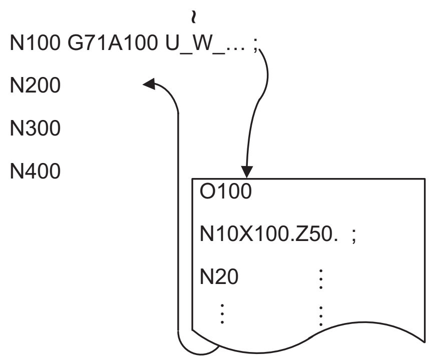
(7) N, F, S, M, and T commands in the finished shape program are ignored.

(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
The next block is the one designated by Q.  ≀ N100 G71P200 Q500 U_W_… ; N200 N300 N400 N500 N600	The next block is the next block of the cycle command.  ≀ N100 G71A100 U_W_… ; N200 N300 N400 O100 N10X100.Z50. ; N20 ⋮ ⋮ ⋮
Operation moves to the N600 block upon completion of the cycle.	Operation moves to the N200 block upon completion of the cycle.

(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.

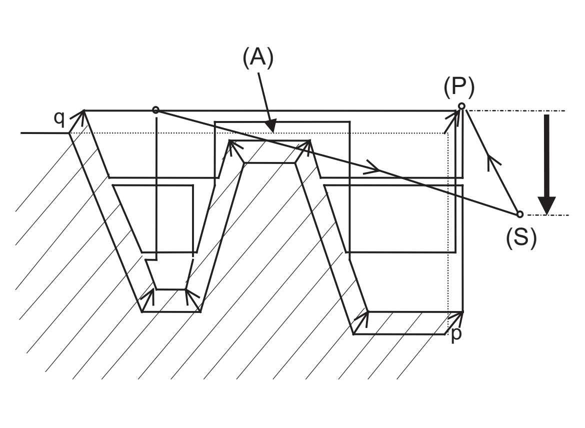
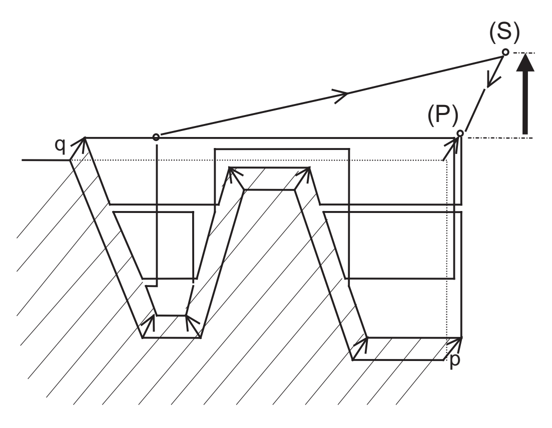
(15) Program error (P203) will occur with the G71 and G72 commands when there is no further movement of the Z axis in the second block or the Z axis has moved in the opposite direction because of tool nose radius compensation.

(16) A program error (P204) will occur if the finished shape program is issued in a same program with G70 to G73 and both P and Q are not designated.

Note that if A, P and Q are all not designated in G71 to G73, this will be interpreted as the first block of the compound type fixed cycle for turning machining I, and an error will not occur.

(17) Set the cycle start point in the final block of the compound type fixed cycle for turning machining I so that the tool retracts from the workpiece cutting face. This is necessary to prevent the cutter from interfering with the convex sections of the workpiece.

The figure below shows a poor example. The position (A) is located higher than the cycle start point and an interference will occur in the final block.

Poor example	Good example
	

(S) Cycle command point

(18) The common variable settings are ignored in the finished shape program. The setting values before compound type fixed cycle for turning machining I command will be valid.

(19) If there is a circular command in the first movement block for the finished shape when pocket machining is OFF, a program error (P203) will occur.

13.3 Fixed Cycle for Drilling

Function and purpose

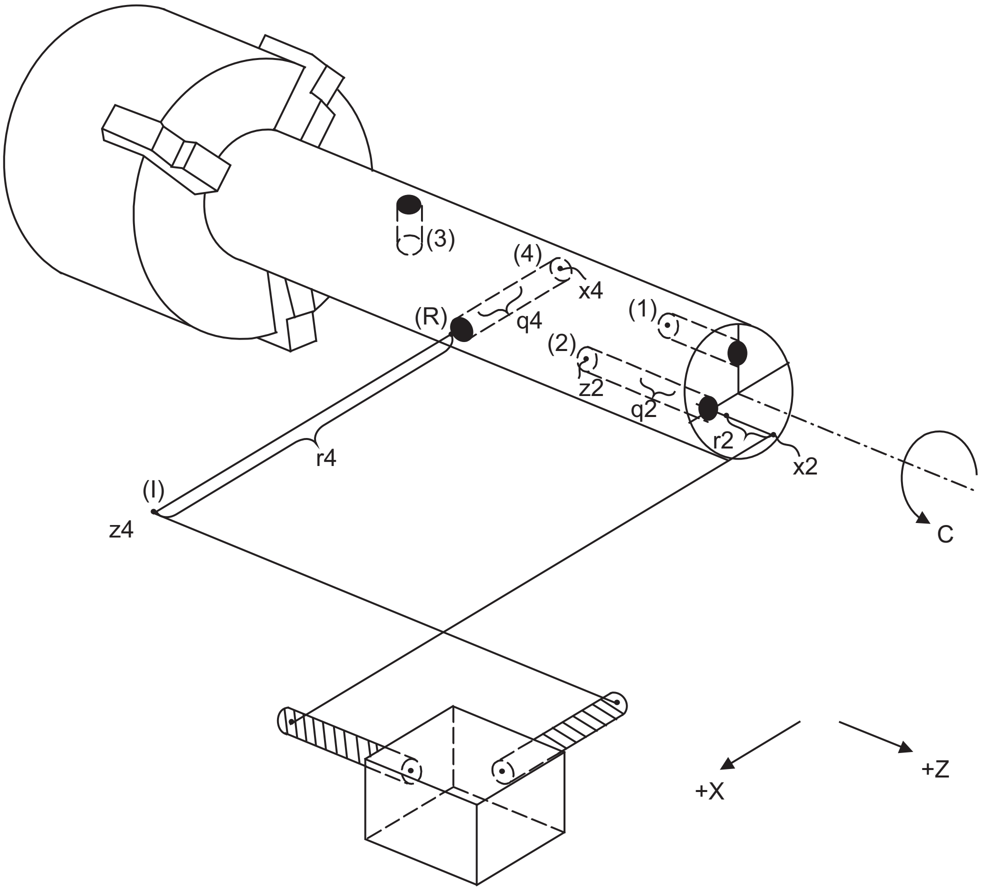
These fixed cycles are used to perform prepared working sequences of machining programs such as positioning, hole drilling, boring and tapping in a block. When performing a same machining repeatedly, it can be executed by commanding only the axis position. The types of fixed cycles are listed below.

(Note) The X axis is designated as the hole drilling axis by longitudinal hole drilling fixed cycle. But he Y axis can also be designated as the hole drilling axis by using the longitudinal hole drilling axis selection function. "Type II" in the table explains details for using the longitudinal hole drilling axis selection function.

G code	Hole drilling axis	Drilling Hole drilling start	Operation at hole bottom	Return operation	Retract at high speed	Application	Type II	
							Longitudinal hole drilling axis selection signal ON/OFF	Hole drilling axis
G80	-	-	-	-	-	Cancel	-	-
G83	Z	Cutting feed Intermittent feed	In-position check, Dwell	Rapid traverse	Possible	Face Deep Hole Drilling Cycle I	ON	Z
							OFF	
G84 (G84.1)	Z	Cutting feed	In-position check, Dwell, Spindle reverse rotation	Cutting feed	-	Face Tapping Cycle (Face Reverse Tapping Cycle)	ON	Z
							OFF	
G85	Z	Cutting feed	In-position check, Dwell	Cutting feed	-	Face Boring Cycle	ON	Z
							OFF	
G87	X	Cutting feed Intermittent feed	In-position check, Dwell	Rapid traverse	Possible	Deep Hole Drilling Cycle I	ON	Y
							OFF	X
G88 (G88.1)	X	Cutting feed	In-position check, Dwell, Spindle reverse rotation	Cutting feed	-	Longitudinal tapping cycle (Longitudinal reverse tapping cycle)	ON	Y
							OFF	X
G89	X	Cutting feed	In-position check, Dwell	Cutting feed	-	Longitudinal boring cycle	ON	Y
							OFF	X
G83.2	Z/X	Cutting feed Intermittent feed	In-position check, Dwell	Rapid traverse	Possible	Deep hole drilling cycle 2	ON	Z/X
							OFF	

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

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

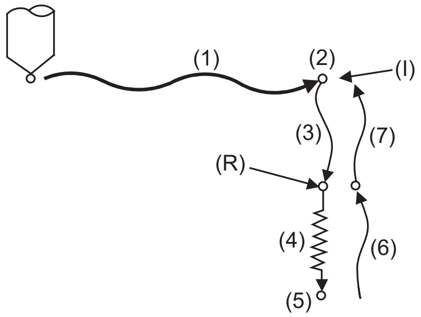
(1) G83 Xx1 Cc1 Zz1 Rr1 Qq1 Pp1 Ff1 Kk1 ; Face deep hole drilling cycle
(2) G83 Xx2 Cc2 Zz2 Rr2 Qq2 Pp2 Ff2 Kk2 ;
(3) G87 Zz3 Cc3 Xx3 Rr3 Qq3 Pp3 Ff3 Kk3 ; Longitudinal deep hole drilling cycle
(4) G87 Zz4 Cc4 Xx4 Rr4 Qq4 Pp4 Ff4 Kk4 ;

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.

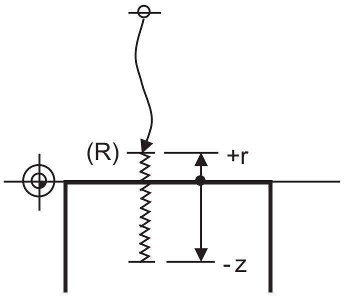
(7) This denotes the returning to the initial point by rapid traverse.
(Operations 6 and 7 may become a single operation depending on the fixed cycle mode.)

(Note 1) In the synchronous tapping command, the position of an in-position check will deffir depending on the parameter setting.
(Refer to "Tapping Cycle; G84 (Face)/G88 (Longitudinal), Reverse Tapping Cycle; G84.1 (Face)/ G88.1 (Longitudinal)"(General format))

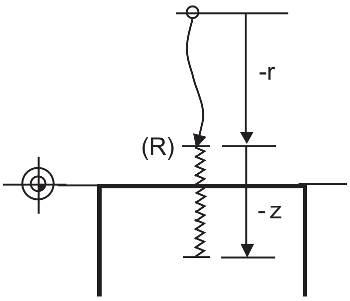
(Note 2) Whether the fixed cycle is to be completed at operation 6 or 7 can be selected by G98/G99 commands. (Refer to "Initial point and R point level return; G98, G99")

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

Face deep hole drilling cycle 1

G83 X/U__ C/H__ Z/W__ Rr Qq Pp Ff Kk Mm ;

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) (modal)
Rr	Designation of R point (incremental value from initial point) (sign ignored) (modal)
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. (modal) Relationship between time and designated value is same as for G04 designation.
Ff	Designation of feed rate for cutting feed (modal)
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Mm	Designation of miscellaneous command

Longitudinal deep hole drilling cycle 1

G87 Z/W__ C/H__ X/U__ Rr Qq Pp Ff Kk Mm ;

Z/W C/H	Designation of hole position initial point (absolute/incremental value) ... Data for positioning Z and C axes
X/U	Designation of hole bottom position (absolute/incremental value from R point) (modal)
Rr	Same as face deep hole drilling cycle 1
Qq	Same as face deep hole drilling cycle 1
Pp	Same as face deep hole drilling cycle 1
Ff	Same as face deep hole drilling cycle 1
Kk	Same as face deep hole drilling cycle 1
Mm	Same as face deep hole drilling cycle 1

Precautions

(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.

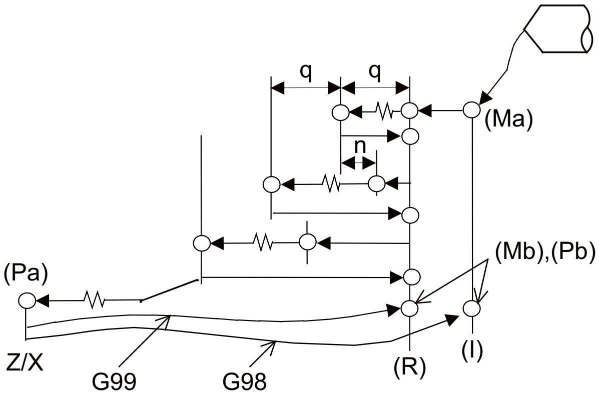
(2) The designation of the hole position initial point is unmodal. When G83(G87) command is to be executed continuously, designate them block by block.

(3) Q command is unmodal. Designate them block by block.

(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.

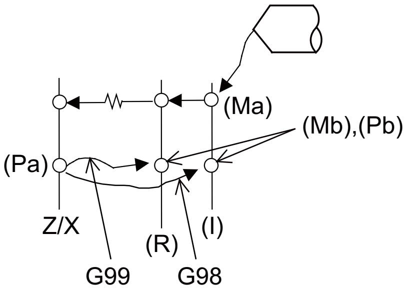
(2) (Ma) ... The M code (Mm) is output when there is a C-axis clamping M code command (Mm).

(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 F f   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) / Face Reverse Tapping Cycle (Longitudinal reverse tapping cycle) ; G84 (G88) / G84.1 (G88.1)

Command format

Face tapping cycle

G84(G84.1) X/U__ C/H__ Z/W__ Rr1 Pp Ff Kk Dd Ss1 ,Ss2 ,Rr2 Mm ;

G84(G84.1)	G84 Face tapping cycle mode G84.1 Face tapping cycle mode (the tap rotation direction is reversed)
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) (modal)
Rr1	Designation of R point (incremental value from initial point) (sign ignored) (modal)
Pp	Designation of dwell time at hole bottom point. Relationship between time and designated value is same as for G04 designation. (modal)
Ff	During synchronous tapping: Designation of drilling axis feed amount (tapping pitch) per spindle revolution (modal) During asynchronous tapping: Designation of the feedrate for cutting feed (modal)
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Dd	Designation of spindle used in tapping cycle (modal) (Setting range: 1 to No. of spindles)
Ss1	Designation of spindle rotation speed (During synchronous tapping, the S command in the command format with a spindle No. (example: "S2 = 2000") is ignored.)
,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 "#8159 Synchronous tap".)
Mm	Designation of miscellaneous command

Longitudinal tapping cycle

G88(G88.1) Z/W__ C/H__ X/U__ Rr1 Pp Ff Kk Dd Ss1 ,Ss2 ,Rr2 Mm ;

G88(G88.1)	G88 Longitudinal tapping cycle mode G88.1 Longitudinal reverse tapping cycle mode (the tap rotation direction is reversed)
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) (modal)
Rr1	Same as face tapping cycle.
Pp	Same as face tapping cycle.
Ff	Same as face tapping cycle.
Kk	Same as face tapping cycle.
Dd	Same as face tapping cycle.
Ss1	Same as face tapping cycle.
,Ss2	Same as face tapping cycle.
,Rr2	Same as face tapping cycle.
Mm	Same as face tapping cycle.

Precautions

(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.

(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.

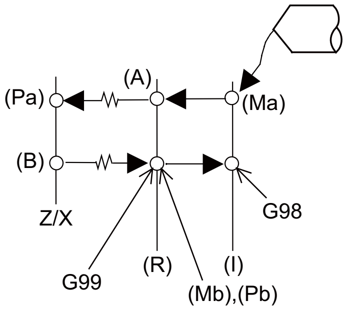
(3) Set a minus value to D command when designating the reverse tapping cycle by G84/G88 with "#1309 Gtype" setting.

(4) When multiple-spindle control II is valid, the spindle is selected from the PLC so do not issue a D command. A program error (P32) will occur if commanded.

(5) When D command is omitted, the spindle designated with the spindle selection command will be selected.

(6) ,S command is held as modal information.
When the value is smaller than the spindle rotation speed (S command), the value of the spindle speed is valid even at the return.
When the spindle speed at the return is not "0", the value of tap return override (#1172 tapovr) is invalid.

Detailed description



(Ma) The M code (Mm) is output when there is a C-axis clamping M code command (Mm).

(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).

(Pa) Dwell is performed for the duration of time commanded by P.

(Pb) After the C-axis unclamping M code (Mm + 2) is output, dwell is performed for the duration of time set by the parameter (#1184 clmp_D).

(R) R point

(I) Initial point

With the reverse tapping cycle G84.1 (G88.1), the tapping rotation direction is reversed.

(1) When G84(G88) is executed, the override will be canceled and the override will automatically be set to 100%.
(2) 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.
(3) During single block operation, the axis will not stop at the turning point of tapping cycle.
(4) During the G84 (G88) modal, the "Tapping" NC output signal will be output.
(5) During the G84 (G88) synchronous tapping modal, the M3, M4, S code, etc. will not be output.
(6) The tool stops at the R point and the rotary tool forward rotation signal is output. (A)
(7) The rotation of the rotary tool is reversed at the hole bottom and tapping is performed. (B)
(8) 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.
(9) Whether to set the reverse tapping cycle G codes to G84.1/G88.1 or G84/G88 (address D value is minus) is determined according to the basis specification parameter "#1309 Gtype".
(10) When it is interrupted by such as the emergency stop during the tapping cycle, a tool can be taken out from the workpiece after performing the tap retract.

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
User parameters

#	Item	Details	Setting range
8159	Synchronous tap	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.

(3) Selecting with M function
Basic specification parameters

#	Item		Details	Setting range
1272(PR)	ext08	bit1	M-function synchronous tapping cycle valid.	0 : Invalid 1 : Valid

Synchronous tapping can be selected with the M function when this parameter is ON.
This parameter requires the CNC to be turned OFF after the settings. Turn the power OFF and ON to enable the parameter settings.

Basic specification parameters

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

The synchronous tapping mode is selected with the miscellaneous function code set with this parameter.
The M function can be commanded just before or in the same block as the tapping command.

(Note1) Do not use M00, 01, 02, 30, 96, 97, 98 and 99.

(Example)

M29 ; (M command synchronous tapping)
G84 Z50. R20. F2.; or G84 Z50. R20. F2. M29 ;

The selection of synchronous and asynchronous tapping by the three items above will follow the combination shown below.

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

× : Does not command
○ : Commands

A : Asynchronous tap
S : Synchronous tapping in-position

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.

Feed per minute command of the synchronous tapping

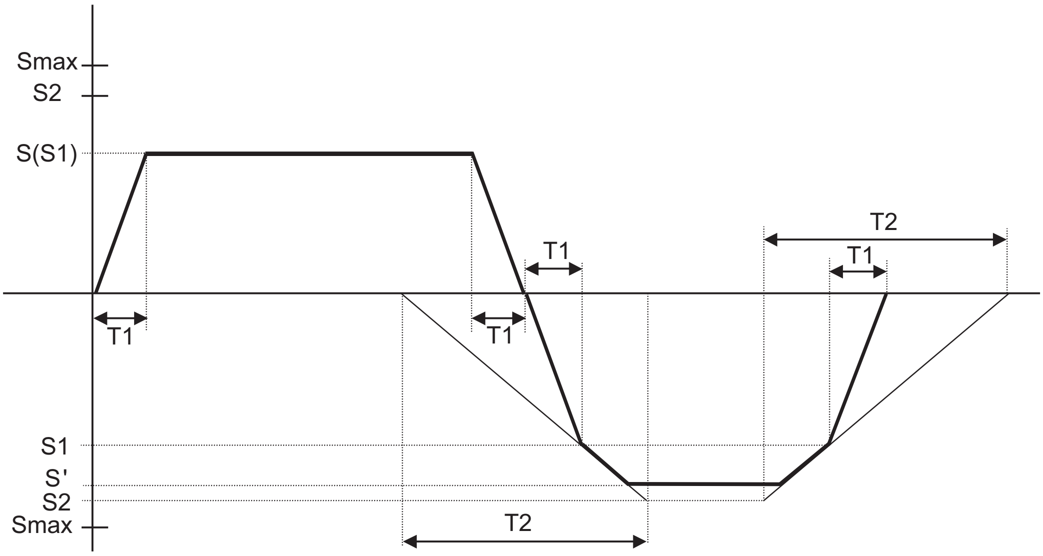
Enable the feed per minute command of the synchronous tapping by the setting of parameter "#1268 ext04/ bit2". When this parameter is valid, G94 and G95 modal will be applied.

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

Spindle acceleration/deceleration pattern during synchronous tapping

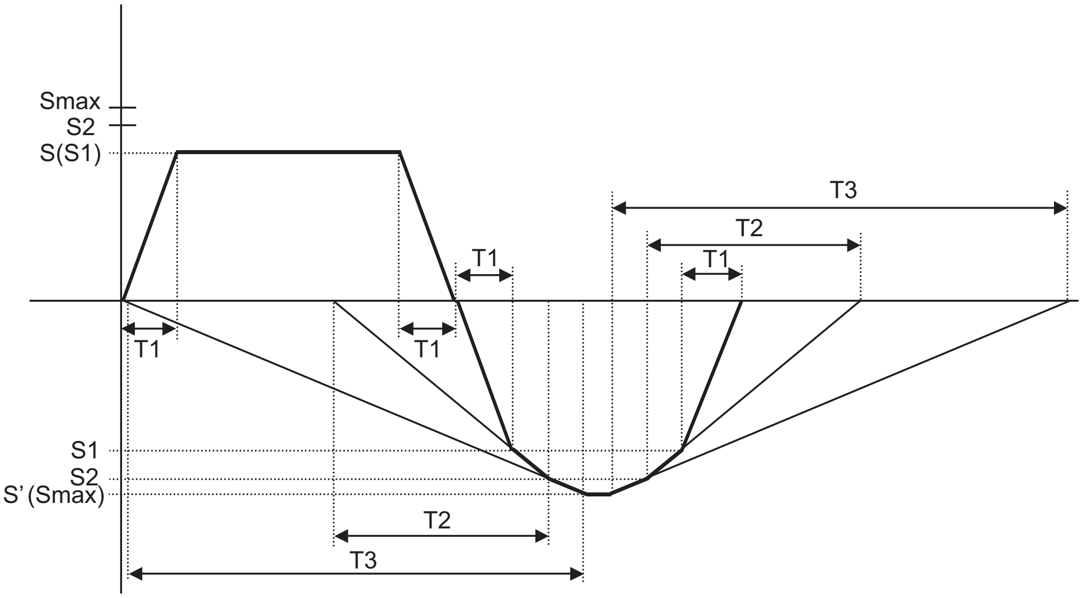
This function enables to make spindle acceleration/deceleration pattern closer to that of the speed loop by dividing the spindle and drilling axis acceleration/deceleration pattern into up to three stages during synchronous tapping. The acceleration/deceleration pattern can be set up to three stages 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 ≦ 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)

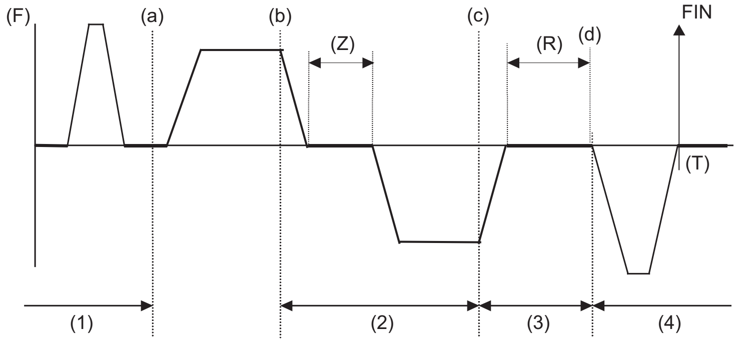
In-position check during synchronous tapping

Synchronous tapping in-position check settings

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

(Note 1) The I point refers to the initial point.

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



(Z) Hole bottom (R) R point (F): Feed rate (T): Time

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

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

R point: In-position check by the G1inps
I point: In-position check by the G0inps
Hole bottom: In-position check by the Tapinps

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

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

(Note 1) The I point refers to the initial point.

(Note 2) Note that vibration or deterioration in accuracy may occur when invalidating the in-position check at R point.

Precautions

Refer to "Precautions When Using a Fixed Cycle for Drilling".

13.3.3 Face Boring Cycle (Longitudinal boring cycle) ; G85 (G89)

Command format

Face boring cycle mode

```
G85 X/U__ C/H__ Z/W__ Rr Pp Ff Kk Mm ;
```

X/U C/H	Designation of hole position initial point (absolute/incremental value)
Z/W	Designation of hole bottom position (absolute/incremental value from R point) (modal)
Rr	Designation of R point (incremental value from initial point) (sign ignored) (modal)
Pp	Designation of dwell time at hole bottom point. Relationship between time and designated value is same as for G04 designation. (modal)
Ff	Designation of feedrate for cutting feed (modal)
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Mm	Designation of miscellaneous command

Longitudinal boring cycle mode

```
G89 Z/W__ C/H__ X/U__ Rr Pp Ff Kk Mm ;
```

Z/W C/H	Designation of hole position initial point (absolute/incremental value)
X/U	Designation of hole bottom position (absolute/incremental value from R point) (modal)
Rr	Designation of R point (incremental value from initial point) (sign ignored) (modal)
Pp	Designation of dwell time at hole bottom point. Relationship between time and designated value is same as for G04 designation. (modal)
Ff	Designation of feedrate for cutting feed (modal)
Kk	Designation of number of repetitions, 0 to 9999 (standard value = 1)
Mm	Designation of miscellaneous command

Precautions

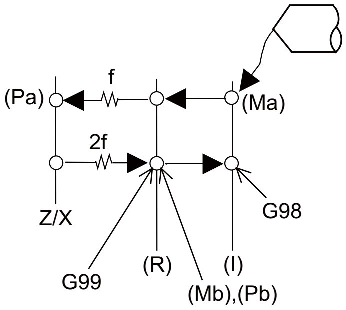
(1) For the longitudinal boring cycle (G89), designate Z/W to the hole position initial point and X/U to the hole bottom position.

(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.

(3) Q command is unmodal. Designate them block by block.

(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.

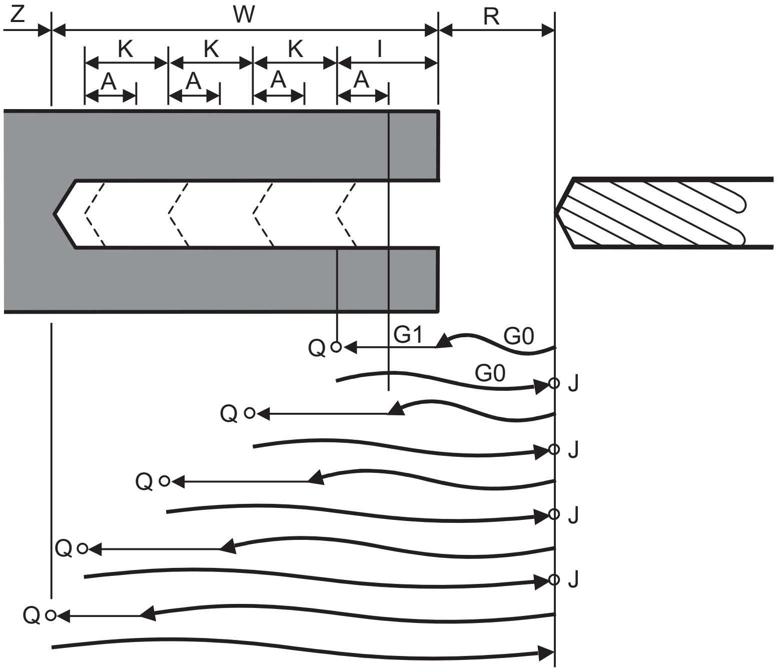
Command format

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.

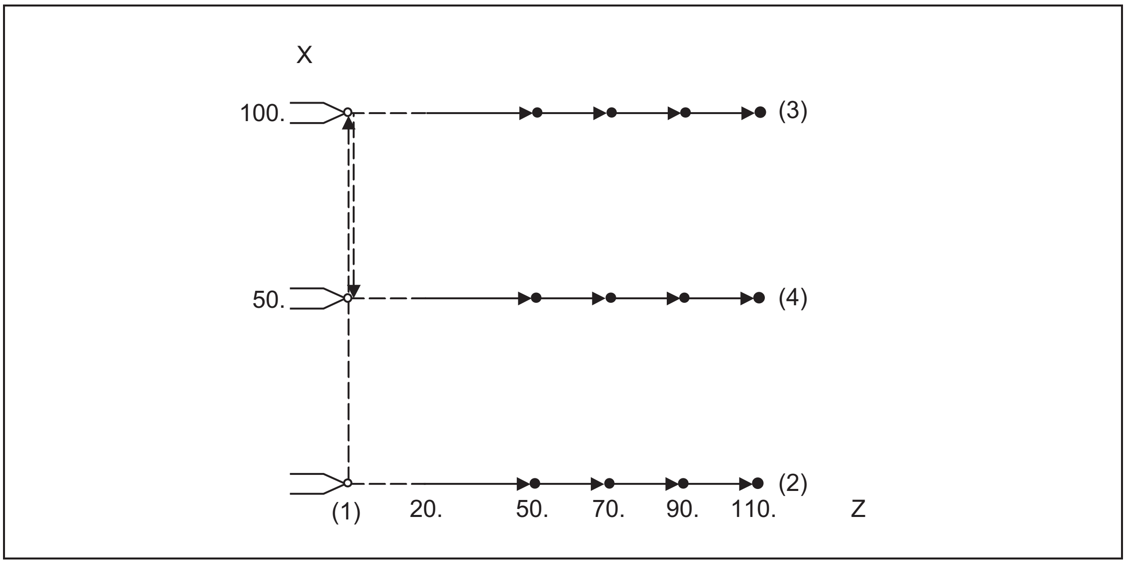
(3) When the feed hold button is pressed while the deep hole drilling cycle 2 is being executed, the automatic operation stops and the remainder of the cycle will be executed when automatic operation is restarted.

(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 lists 2,4,6 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.

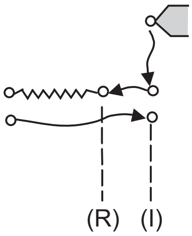
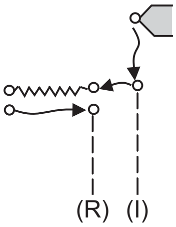
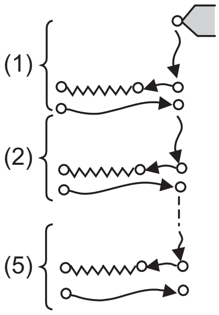
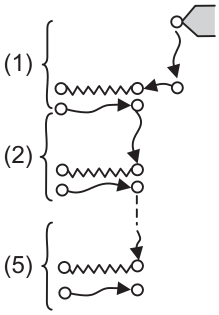
Command format

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 ;	 Initial level return is executed.	 R point level return is executed.
Two or more executions	G90 G83 X100. Z-50. R25. L5 F1000 ;	 Initial level return is executed for all times.	

13.3.8 Setting of Workpiece Coordinates in Fixed Cycle Mode

Function and purpose

The designated axis moves in the workpiece coordinate system set for the axis.
The Z axis becomes valid from the R point positioning after positioning is completed or from Z axis movement.

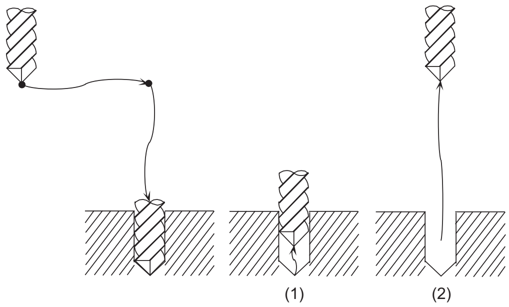
(Note) When the workpiece coordinates change, re-program the addresses Z and R, even if the values are the same.

```
(Example)G54 Xx1 Zz1;
         G83 Xx1 Zz2 Rr2 ;
         G55 Xx3 Zz2 Rr2 ;  ... Re-command even if Z and R are the same as the previous value.
         Xx4 ;
         Xx5 ;
```

13.3.9 Drilling Cycle High-Speed Retract

Function and purpose

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



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

Command format

The command format is the same as fixed cycle.

Detailed description

(1) When "#8123 H-spd retract ON" is ON, the tool is retracted from the hole bottom at high speed using the lost motion compensation function.
Set the lost motion compensation type 2 or 3 to the servo parameter. Then set the following parameters to adjust the retract amount.
- #2170 Lmc1QR (Lost motion compensation gain 1 for high-speed retract)
(correspond to "#2216 SV016 LMC1 Lost motion compensation 1")
- #2171 Lmc2QR (lost motion compensation gain 2 for high-speed retract)
(correspond to "#2241 SV041 LMC2 Lost motion compensation 2")
Set the following parameters for lost motion compensation timing or lost motion compensation 3 spring constant viscous coefficient, apart from the ordinary lost motion compensations.
- #2172 LmcdQR (Lost motion compensation timing for high-speed retract)
(correspond to "#2239 SV039 LMCD Lost motion compensation timing)
- #2173 LmckQR (Lost motion compensation 3 spring constant for high-speed retract)
(correspond to "#2285 SV085 LMCk Lost motion compensation 3 spring constant")
- #2174 LmccQR (Lost motion compensation 3 viscous coefficient for high-speed retract)
(correspond to "#2286 SV086 LMCc Lost motion compensation 3 viscous coefficient")

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

(3) When the inclined axis control is valid and also, the hole drilling axis is an inclined axis or the basic axis of an inclined axis, a high-speed retract will not be performed. The drill moves in the ordinary rapid traverse.

13.3.10 Acceleration/Deceleration Mode Change in Hole Drilling Cycle

Function and purpose

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

Command format

The command formats are the same as those of the fixed cycles G83, G87, and G83.2.

Detailed description

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

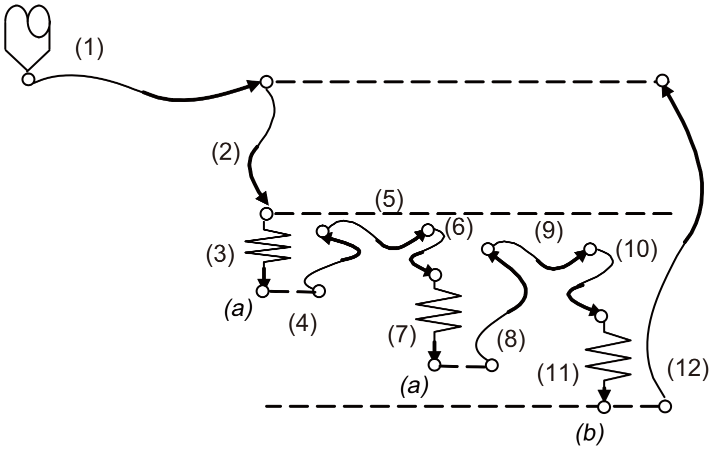
(1) Acceleration/deceleration mode will be either linear or soft method. (Unless soft acceleration/deceleration is applied, the linear method will always be applied.)

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

Operation example

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

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



(a) cut point
(b) hole bottom

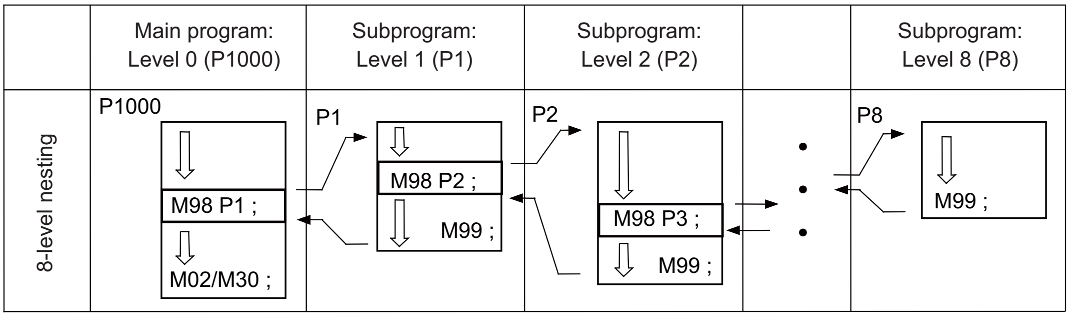
#19417		G83	G87	G83.2
0	(a)Cut point	Perform no deceleration check.		
	(b)Hole bottom	Perform no deceleration check.		
1	(a)Cut point	Perform command deceleration check.		
	(b)Hole bottom	Perform command deceleration check.		
2	(a)Cut point	Perform command deceleration check.		Perform in-position check (sv024).
	(b)Hole bottom	Perform in-position check (sv024).		

13.4 Subprogram Control; M98, M99, M198

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

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

(Note 1) ○ denotes available functions and × denotes unavailable functions.

(Note 2) Variables cannot be transferred with the M98 command but variable commands in subprograms can be used.

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

Command format

Subprogram call

```
M98  P__  H__  L__  ,D__  ;
```

```
M98  <file name>  H__  L__  ,D__  ;
```

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

Return to main program from subprogram

```
M99  P__  ;
```

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

Detailed description

Creating and registering subprograms

Subprograms have the same format as machining programs for normal memory mode, except that the subprogram completion instruction M99 (P_) ; must be registered as an independent block in the last block.

O******** ;	Program No. as subprogram No.
.... ; : .... ;	Main body of subprogram
M99 ;	Subprogram return command
%(EOR)	Registration completion code

(1) The above program is registered by editing operations at the setting and display unit. For further details, refer to the section on "program editing" in the Instruction Manual.

(2) Only those subprogram Nos. ranging from 1 to 99999999 designated by the optional specifications can be used. When there are no program Nos. on the tape, they are registered as the setting No. for "program input."

(3) Up to 8 nesting levels can be used for calling programs from subprograms and program error (P230) will occur if this number is exceeded.

(4) Main programs and subprograms are registered in the order they were read without distinction. Therefore, main programs and subprograms should not be given the same Nos. (If they are, error "E11" will be displayed at registration.)

(5) Main programs can be executed during memory, tape or MDI mode but subprograms must be in the memory mode.

(6) 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

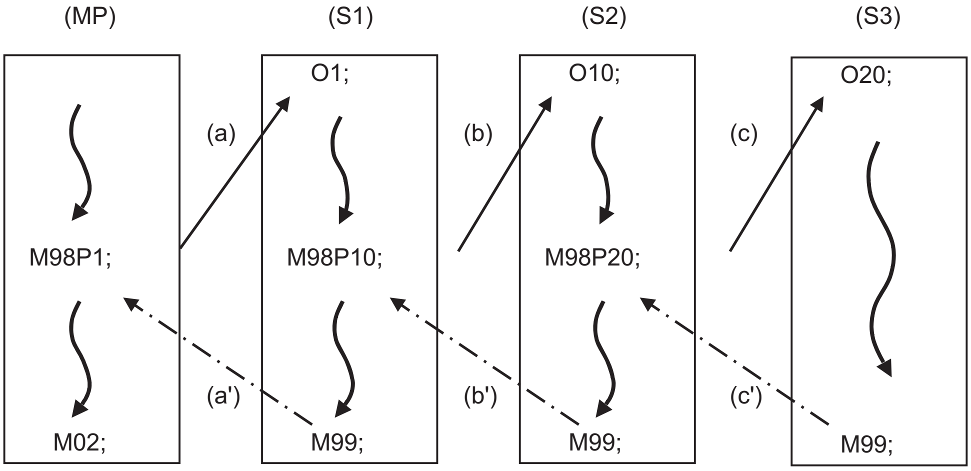
(7) Subprogram nesting is not subject to the following commands which can be called even beyond the 8th nesting level.
- Fixed cycles
- Pattern cycles

(8) To repeatedly use the subprogram, it can be repeated l1 times by programming M98 Pp1 Ll1;.

Program example

Program example 1

When there are 3 subprogram calls (known as 3 nesting levels)



(M) Main program

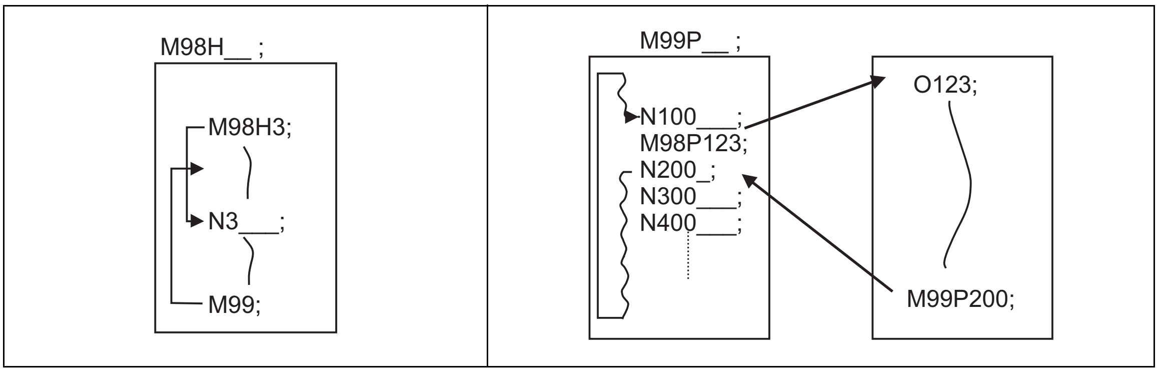
(S1) Subprogram 1
(S2) Subprogram 2
(S3) Subprogram 3

Sequence of execution : (a)-(b)-(c)-(c')-(b')-(a')

(1) For nesting, the M98 and M99 commands should always be paired off on a 1:1 basis; (a)' for (a), (b)' for (b), etc.
(2) Modal information is rewritten in the order of execution sequence without distinction between main programs and subprograms. Therefore, after calling a subprogram, attention must be paid to the modal data status when programming.

Program example 2

The M98 H_ ; M99 P_ ; commands designate the sequence Nos. in a program with a call instruction.



Precautions

(1) Program error (P232) will occur when the designated P (program No.) cannot be found.

(2) The M98 P_ ; M99 ; block does not perform a single block stop. If any address except O, N, P, L or H is used, single block stop can be executed. (With "X100. M98 P100 ;, the operation branches to O100 after X100. is executed.)

(3) When M99 is commanded by the main program, operation returns to the head. (This is same for MDI.)

(4) Branching from tape mode to the subprogram with M98 P_ is possible, but the return destination sequence No. cannot be designated with M99 P_ ;. (P_ is ignored.)

(5) Note that it takes time to search when the sequence No. is designated by M99 P_ ;.

(6) When using a file name for the subprogram, specify the file name with 32 characters or less, including the extension. If a file name exceeding 32 characters is specified, a program error (P232) will occur.

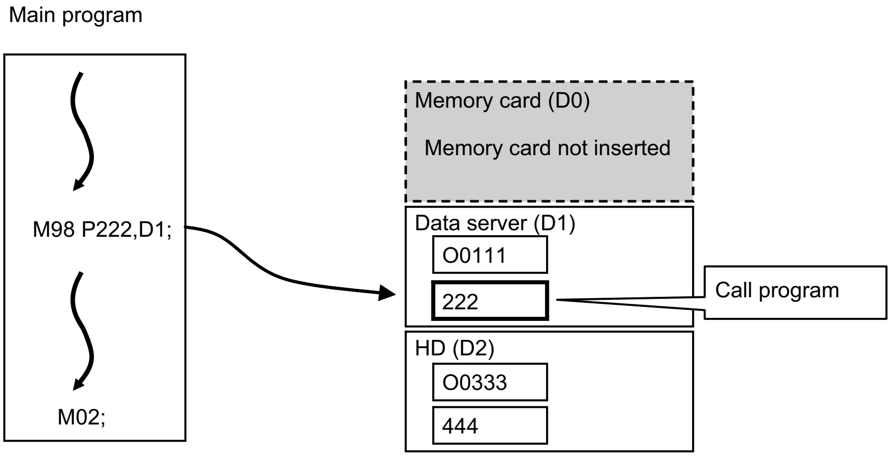
(7) All the programs are registered as files. For example, when calling a file "0100" as a subprogram, "0100" cannot be searched with M98P100 or M98P0100. When numerical values are specified after P, 0 is ignored. In this case, it is regarded that the program No. (file) "100" is specified. To call a program like "0100" , specify the file name using the M98<0100> format.

(8) A subprogram added O No. is searched with the parameter setting (#8129="1" or "2") which calls a subprogram with O No. as priority.
If a subprogram with O No. is not found, a subprogram with a name specified with the P command is searched.

(Note) To prevent any unintended program call, avoid using such program names that may be confused. (For example, 123, O0123 and O00000123 can be considered identical.)

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

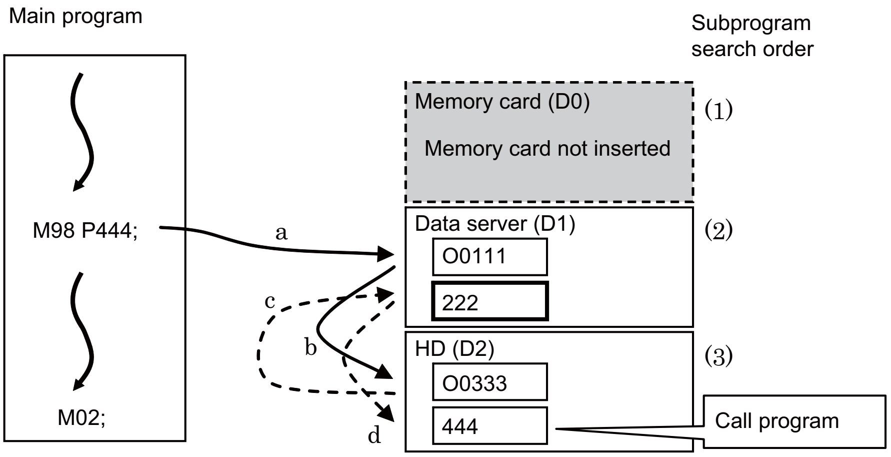
(a) With designation of device No.
Only the designated devices are subject to search.



［Parameter setting］

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

(b) Without designation of device No.
A subprogram with O No. is searched according to the settings of #8890(D0 in order of subprogram search) to #8894(D4 in order of subprogram search). (Refer to the solid line arrows a and b.)
If a subprogram with O No. is not found, subprograms with a name designated with the P command are searched in order of the subprogram search setting. (Refer to the broken lines c,d.)
If none of the designated subprogram storage locations are subject to search, memories are searched.
 * If any device or directory designated as the subprogram storage location is not found due to a reason such as absence, poor contact and contact failure of a memory card, the said device or directory will be excluded from the search target.



［Parameter setting］

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

13.4.2 Subprogram Call ; M198

Function and purpose

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

Command format

Subprogram call

```
M198  P__  L__  ;
```

```
M198 <File name> L__ ;
```

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

Return to main program from subprogram

```
M99 ;
```

Detailed description

(1) The subprogram can be called with the M198 command once in the subprogram nest. The subprogram can be called only from the memory or MDI program.

(2) The section from the head of the program to the first LF (line feed code, 0x0A hexadecimal) is invalid, and is not run or displayed. Note that if the head starts with a 0 No., the program will be valid from the head.

(3) Refer to "Subprogram call ; M98,M99" for <File name> and calling the subprogram with O No.

13.5 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

#* = ○○○○○○○○ ;**

#* = [formula] ;**

Detailed description

Variable expressions

		Example
#m	m = value consisting of 0 to 9	#100
# [f]	f = one of the followings in the formula	# [-#120]
	Numerical value m	123
	Variable	#543
	Formula Operator Formula	#110+#119
	- (minus) formula	-#120
	[Formula]	[#119]
	Function [formula]	SIN [#110]

(Note 1) The 4 standard operators are +, -, * and /.

(Note 2) Functions cannot be used unless the user macro specifications are available.

(Note 3) Error (P241) will occur when a variable No. is negative.

(Note 4) Examples of incorrect variable expressions are given below.

Incorrect	Correct
#6/2	# [6/2] (#6/2 is regarded as [#6] /2)
#--5	# [-[-5]]
#- [#1]	# [-#1]

Types of variables

The following table gives the types of variables.

Type	No.		Function
Common variables	Common variables 1	Common variables 2	- Can be used in common throughout main, sub and macro programs.
200 sets	500 to 599	100 to 199	

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

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

(Note 3) When inputting the common variable data, if the following illegal variable No. data exist in the input file, the illegal variable No. data is ignored and only the correct common variable data will be input.
- Variable data which is not common variables such as local variables (#1 to #33) and system variables (from #1000), etc.
- Variable data with which conditions of number of common variable sets do not match.

Variable quotations

Variables can be used for all addresses except O, N and / (slash).

(1) When the variable value is used directly:
X#1 .... Value of #1 is used as the X value.

(2) When the complement of the variable value is used:
X-#2 .... Value with the #2 sign changed is used as the X value.

(3) When defining variables:
#3 = #5 .... Variable #3 uses the equivalent value of variable #5.
#1 = 1000 .... Variable #1 uses the equivalent value 1000 (which is treated as 1000.).

(4) When defining the variable arithmetic formula:
#1 = #3 + #2 - 100 .... Value of the operation result of #3 + #2 - 100. is used as the #1 value.
X［#1 + #3 + 1000］........ Value of the operation result of #1 + #3 + 1000 is used as the X value.

(Note 1) A variable cannot be defined in the same block as an address. It must be defined in a separate block.

Incorrect	Correct
X#1 = #3 + 100 ;	#1 = #3 + 100 ; X#1 ;

(Note 2) Up to five sets of square parentheses [ ] may be used.
#543 = -[[[[[#120]/2+15.]*3-#100]/#520+#125+#128]*#130+#132]

(Note 3) There are no restrictions on the number of characters and number of variables for variable definition.

(Note 4) The variable values should be within the range of 0 to ±99999999.
If this range is exceeded, the arithmetic operations may not be conducted properly.

(Note 5) The variable definitions become valid when definitions are made.
#1 = 100 ; .... #1 = 100
#1 = 200 #2 = #1 + 200 ; ..... #1 = 200, #2 = 400
#3 = #1 + 300 ; .... #3 = 500

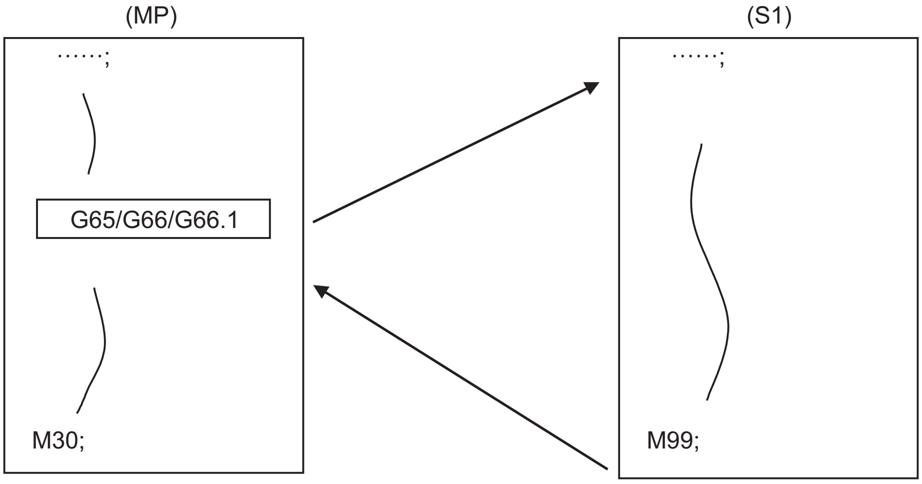
(Note 6) Variable quotations are always regarded as having a decimal point at the end.
When #100 = 10
X#100 ; is treated as X10.

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.

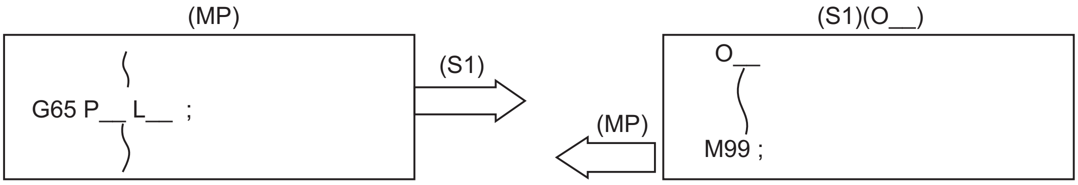
13.6.2 Macro Call Instruction

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

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

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

Detailed description

(1) When the argument must be transferred as a local variable to a user macro subprogram, the actual value should be designated after the address.
In this case, regardless of the address, a sign and decimal point can be used in the argument. There are 2 ways in which arguments are designated.

Argument designation I

Format : A_ B_ C_ ... X_ Y_ Z_

(a) Arguments can be designated using any address except G, L, N, O and P.

(b) I, J and K must be designated in alphabetical order.
I_ J_ K_ ...... Correct
J_ I_ K_ ...... Incorrect

(c) Except for I, J and K, there is no need for designation in alphabetical order.

(d) Addresses which do not need to be designated can be omitted.

(e) The following table shows the correspondence between the addresses which can be designated by argument designation I and the variable numbers in the user macro main body.

Address and variable No. correspondence		Addresses available for call instructions	
Argument designation I address	Variable in macro	G65,G66	G66.1
A	#1	○	○
B	#2	○	○
C	#3	○	○
D	#7	○	○
E	#8	○	○
F	#9	○	○
G	#10	×	× *
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__ ...

(a) In addition to address A, B and C, up to 10 groups of arguments with I, J, K serving as 1 group can be designated.

(b) When the same address is duplicated, designate the addresses in the specified order.

(c) Addresses which do not need to be designated can be omitted.

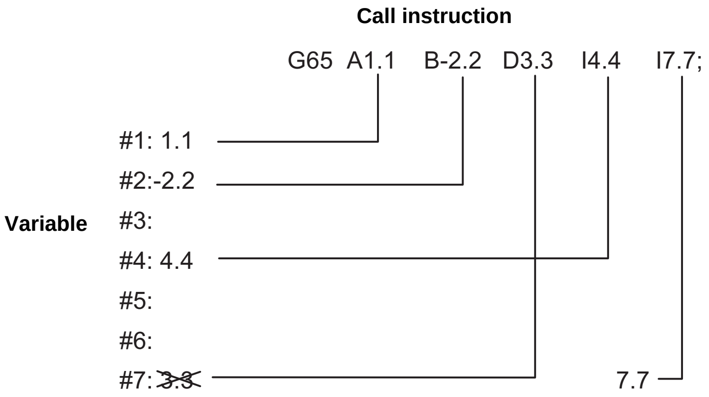
(d) The following table shows the correspondence between the addresses which can be designated by argument designation II and the variable numbers in the user macro main body.

Argument specification II address	Variable in macro		Argument specification II address	Variable in macro
A	#1		J5	#17
B	#2		K5	#18
C	#3		I6	#19
I1	#4		J6	#20
J1	#5		K6	#21
K1	#6		I7	#22
I2	#7		J7	#23
J2	#8		K7	#24
K2	#9		I8	#25
I3	#10		J8	#26
J3	#11		K8	#27
K3	#12		I9	#28
I4	#13		J9	#29
J4	#14		K9	#30
K4	#15		I10	#31
I5	#16		J10	#32
			K10	#33

(Note 1) Subscripts 1 to 10 for I, J, and K indicate the order of the specified command sets. They are not required to specify instructions.

Using arguments designations I and II together

If addresses corresponding to the same variable are commanded when both types I and II are used to designate arguments, the latter address will become valid.
(Example 1)



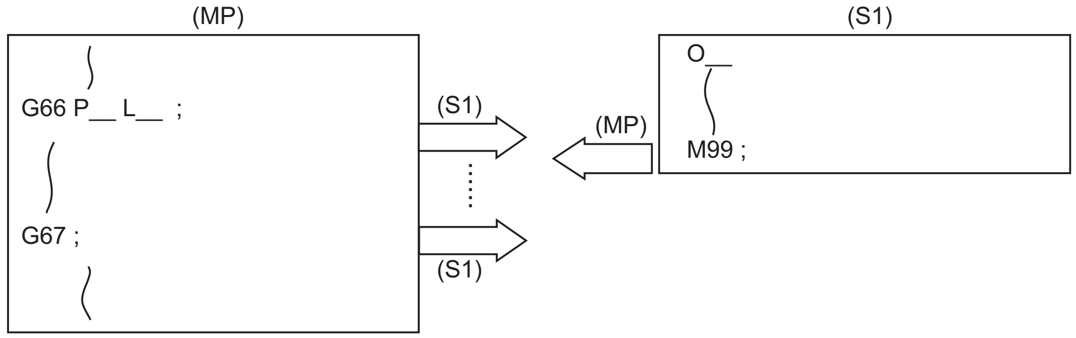
In the above example, I7.7 argument is valid when both arguments D3.3 and I7.7 are commanded for the #7 variable.

(2) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.
(Example)
When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "G65 P12" command.

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

13.6.2.2 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

Modal call A

G66 P__ L__ argument ;

G66 <File name> L__ argument ;

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

Modal call end

G67 ;

Detailed description

(1) When the G66 command is entered, the specified user macro program will be called after the movement command in a block with the movement commands has been executed, until the G67 (cancel) command is entered.

(2) The G66 and G67 commands must be paired in a same program.
A program error will occur when G67 is issued without G66.

(3) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.
(Example)
When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "G66 P12" command.

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

13.6.2.3 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

Modal call B

```
G66.1 P__ L__ argument ;
```

```
G66.1 <File name> L__ argument ;
```

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

Modal call end

```
G67 ;
```

Detailed description

(1) In the G66.1 mode, everything except the O, N and G codes in the various command blocks which are read are handled as the argument without being executed. Any G code designated last or any N code commanded after anything except O and N will function as the argument.

(2) All significant blocks in the G66.1 mode are handled as when G65P__ is assigned at the head of a block.
(Example 1)
In "G66.1 P1000 ; " mode,
N100 G01 G90 X100. 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.

(5) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.
(Example)
When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "G66.1 P12" command.

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

13.6.2.4 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. G codes from G01 to G99 can also be made available for use by parameter setting "#1081 Gmac_P" even though they are used in the system.

(4) These commands cannot be issued in a program which has been called by a G code macro. If issued in such a program, they will be handled as ordinary G commands.

13.6.2.5 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 M9999 can be registered.
Note that the codes to be registered should exclude those basically required for the machine and M0, M1, M2, M30, M96 to M99, and M198.

(3) As with M98, it is displayed on the screen display of the setting and display unit but the M codes and MF are not output.

(4) Even if the registered miscellaneous commands above are issued in a user macro subprogram which are called by an M code, it will not be regarded as a macro call and will be handled as a normal miscellaneous command. (Same for S, T and B codes)

(5) All S, T and B codes call the subprograms in the prescribed program numbers of the corresponding S, T and B functions.

(6) Up to 10 M codes can be set.

(Note) If a value 1 to 3 is set to "#7002 M[01] Type", a macro call equivalent to G65, G66 or G66.1 will be executed. In this case the alphabet before M,S,T,B code macro is not treated as an argument.

(Preparation)

Register M06 to M code macro.

(Command)

M06 T02 ... The value of T is treated as variable #20 in macro. At the same time a value will be inserted to T code.

T02 M06 ... A value will not be input to variable #20. A value will be input to T code.

13.6.2.6 Detailed Description for Macro Call Instruction

Detailed description

Differences between M98 and G65 commands

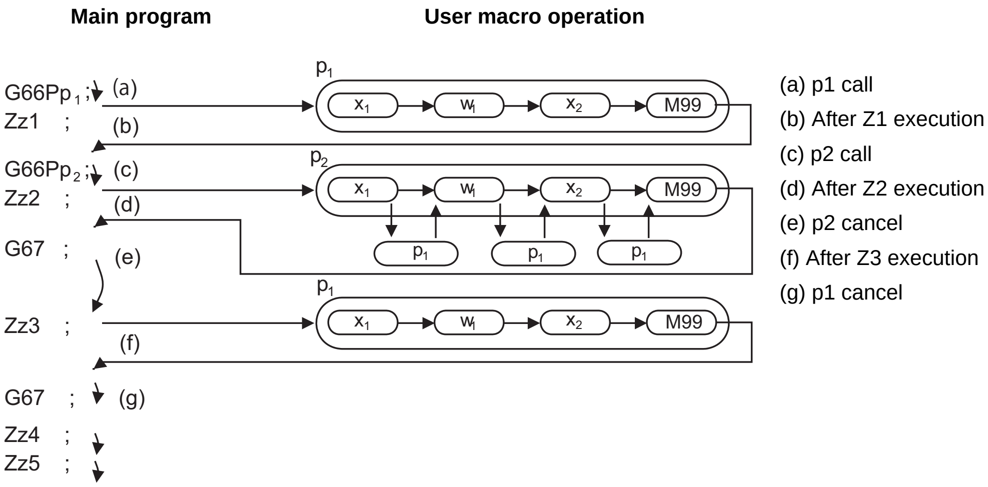
(1) The argument can be designated for G65 but not for M98.
(2) The sequence number can be designated for M98 but no for G65, G66 and G66.1.
(3) M98 executes subprograms after all the commands except M, P, H and L in the M98 block are executed, but G65 branches directly to the subprogram without any 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.

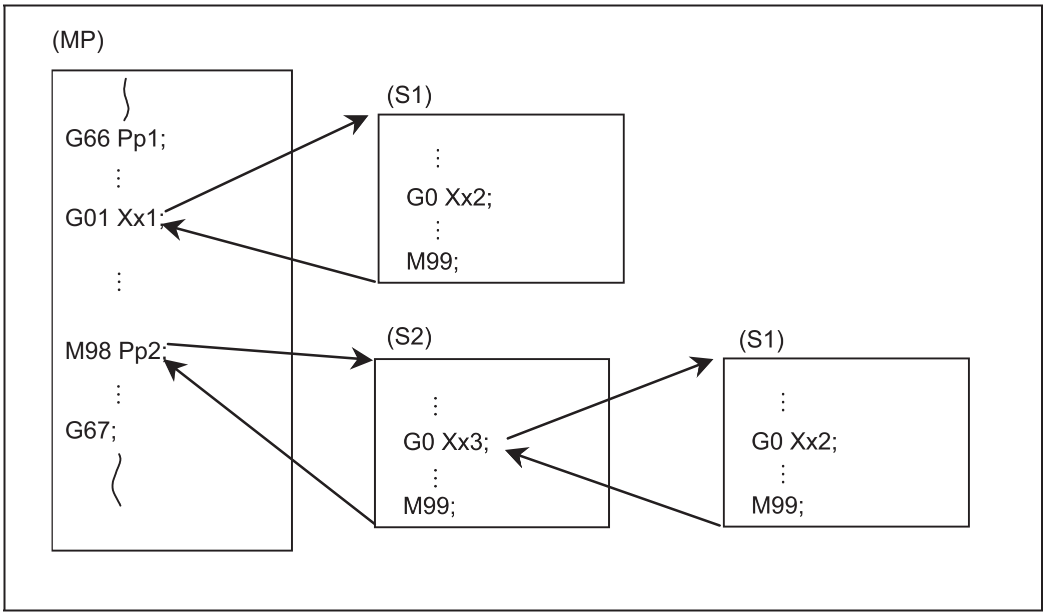
(Note 1) When a G65, G66, G66.1 G code macro call or miscellaneous command macro call is conducted, this is regarded as a nesting level and the level of the local variables is also incremented by one.
(Note 2) With modal call A, the designated user macro subprogram is called every time a movement command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



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

(Example 2)



(MP) Main program

(S1),(S2) Subprogram

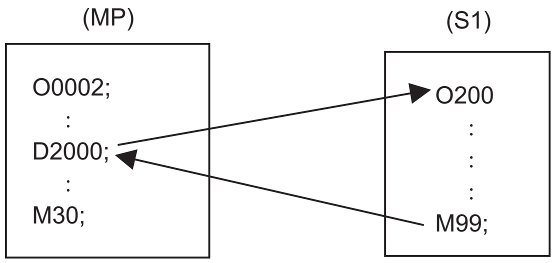
When the program numbers of p1 and p2 are same, the program numbers of (S1) and (S2) will be same.

13.6.3 ASCII Code Macro

Function and purpose

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

(Execution example 1)　M98 type



```
O0002;
   :
D2000;
   :
M30;
```

```
O200
   :
   :
   :
M99;
```

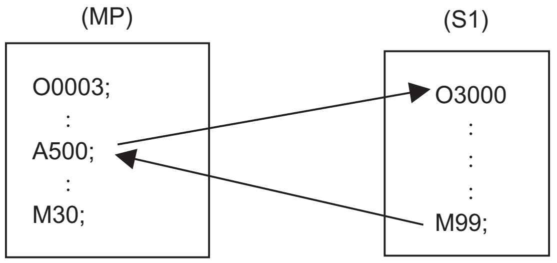
(MP) Main program
(S1)　Subprogram

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

Parameter

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

(Execution example 2)　G65 type



```
O0003;
   :
A500;
   :
M30;
```

```
O3000
   :
   :
   :
M99;
```

(MP) Main program
(S1)　Subprogram

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

Parameter

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

Command format

□ ** ; ... Designates the address and code**

□	ASCII code for calling out a macro (one character)
****	Value or expression output to variable (Setting range: ± 999999.9999)

Detailed description

(1) The command above performs same operations as commands listed below. The correspondence of commands is set for each ASCII code with the parameters.

0 : M98 P****;
1 : G65 P**** <Argument>;
2 : G66 P**** <Argument>;
3 : G66.1 P**** <Argument>;

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

(2) The ASCII code for calling the macro and the program No. P**** to be called are set with the parameters. Up to two ASCII codes can be registered.

(3) The code section is output to the variables, but the output destination differs according to the call type and address.

(a) For M98 type
The code section is output to a common variable and the variable No. is set with a parameter.
When corresponding to the first address (parameter #7401), the section is output to the common variable which is indicated by the first variable No. (parameter #7404).

(b) For G65/G66/G66.1 type
The code section is output to a local variable. The variable No. differs according to the address, and corresponds to the following table.

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

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

Program example

The figure below describes s a program that controls the tailstock with address A and machines a long thin workpiece.

#7411 (ASCII [02] Valid/Invalid)	1 (Valid)
#7412 (ASCII [02] Code)	A
#7413 (ASCII [02] Type)	1 (G65 type)
#7414 (ASCII [02] Program No.)	500
#7415 (ASCII [02] Variable)	100 (Not used)

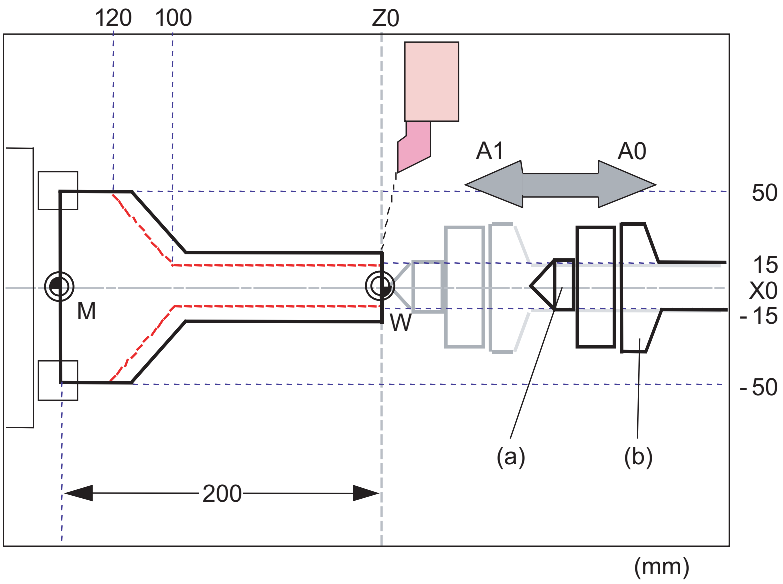
Main program

```
G28 XZ;
A1;                 Call O500
G00 X20. Z0.;
G01 X15. F100;
G01 Z100. F200;
X50 Z120. ;
A0;                 Call O500
M30 ;
```

Subprogram

```
O500;
IF[#1 EQ 0]GOTO 10;
G53 G00 A-205.;       Tailstock advance
G53 G01 A-200. F100;
GOTO N20;
N10 M26;
G53 G0 A-400.;        Tailstock retract
N20 M99;
```

Address A is converted into G65 P500.



(a) Tailstock spindle

(b) Tailstock A spindle

Press with A1
Pull with A0

Precautions

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

A macro cannot be called with an ASCII code from a macro-called program with an ASCII code.
The other patterns are shown below.
If it is judged that a macro cannot be called, the command will be handled as a normal command.

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

Nest level of macro call command

Up to 4 nesting levels are available for macro subprogram calls using simple call (G65) and modal call (G66/ G66.1).
The macro call command's argument is valid only in the called macro level.
Since the macro call nest level is four, the argument can be used in the program as a local variable for each macro call.

Nest level of macro call command

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

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

The following commands can be issued regardless of nesting.

(7) Fixed cycle
(8) Macro interruption

Order of command priority

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

(1) M98,M99 (subprogram call command)
M00 (program stop command), M01 (optional stop command)
M02,M30,M198,M199 (end command)
M96,M97 (macro interruption command)
(2) When corresponding to ASCII code macro command
(3) Used as normal command
"S", "T" and "B" will also overlap with the miscellaneous command macro call. However, if these commands do not correspond to the identification, the ASCII code macro will be executed.
If the other addresses do not correspond to the ASCII code macro command, they will be identified as normal commands. If the command to be used, overlaps with an ASCII code macro command, it must be commanded in the macro-called program with the ASCII code.
Note that there are cases where the command will be unconditionally handled as a normal command, as explained in (5) below.

Conditions where the address set is handled as a normal command

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

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

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

13.6.4 Variable

Function and purpose

Both the variable specifications and user macro specifications are required for the variables which are used with the user macros.
The compensation amounts of the local, common and system variables among the variables for this NC system except #33 are retained even when the unit's power is switched off. (Common variables can also be cleared by parameter "#1129 PwrVC1".)

Detailed description

Use of multiple variable

When the user macro specifications are applied, variable Nos. can be turned into variables (multiple uses of variables) or replaced by <formula>.
Only one of the four basic arithmetic rule (+, -, * , /) operations can be conducted with <formula>.
(Example 1) Multiple uses of variables

#1=10 #10=20 #20=30 ;	
#5=# [#[#1]] ;	# [# [#1]] = # [#10] from #1 = 10. # [#10] = #20 from #10 = 20. Therefore, #5 = #20 or #5 = 30.

#1=10 #10 =20 #20=30 #5=1000;	
#[#[#1]]=#5;	# [# [#1]] = # [#10] from #1 = 10. # [#10] = #20 from #10 = 20. Therefore, #20 = #5 or #20 = 1000.

(Example 2) Example of multiple designations of variables

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

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

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

Undefined variables

When applying the user macro specifications, variables which have not been used even once after the power was switched on or local variables which were not specified by the G65, G66 or G66.1 commands, can be used as <Blank>. Also, variables can forcibly be set to <Blank>.
Variable #0 is always used as the <Blank> and cannot be defined in the left-side member.

(1) Arithmetic expressions

```
#1 = #0 ; ...........................#1 = <Blank>
#2 = #0 + 1;      .......... #2 = 1
#3 = 1 + #0;      .......... #3 = 1
#4 = #0 * 10;   .......... #4 = 0
#5 = #0 + #0;   .......... #5 = 0
```

Note that <Blank> in an arithmetic expression is handled in the same way as 0.
<Blank> + <Blank> = 0
<Blank> + <Constant> = Constant
<Constant> + <Blank> = Constant

(2) Variable quotations
When only the undefined variables are quoted, they are ignored including the address itself.
When #1 = <Blank>

```
G0 X#1 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.

13.6.5 Types of Variables

13.6.5.1 Common Variables

Detailed description

Common variables can be used commonly from any position. Number of the common variables sets depends on the specifications.
Refer to the explanation about Variable Commands for details.

Variable name setting and quotation

Any name (variable name) can be given to common variables #500 to #519. It must be composed of not more than 7 alphanumerics and it must begin with a letter. Do not use "#" in variable names. It causes an alarm when the program is executed.

SETVNn [NAME1,NAME2, ...] ;

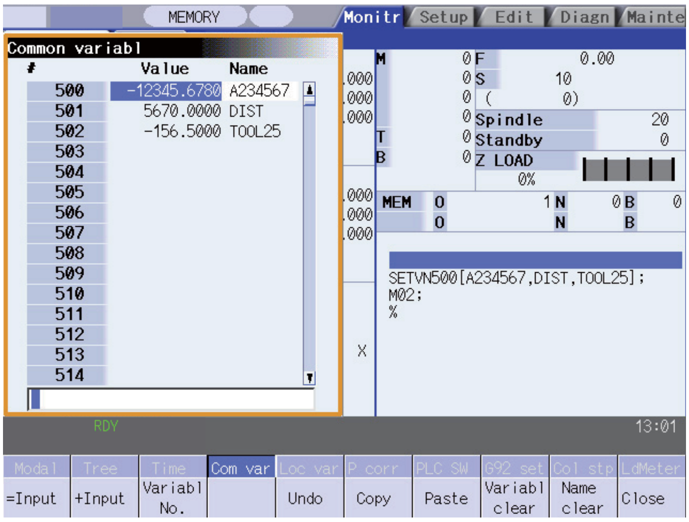
n	Head No. of variable to be named
NAME1	#n name (variable name)
NAME2	#n + 1 name (variable name)

Variable names are separated by a comma (,).

(1) Once variable names have been set, they will not be cleared even when the power is turned off.

(2) Variables in programs can be quoted by their variable names. In this case, the variables should be enclosed in square parentheses [ ].
(Example 1) G01X [#POINT1] ;

(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 variabl

#	Value	Name
500	-12345.6780	A234567
501	5670.0000	DIST
502	-156.5000	TOOL25
503		
504		
505		
506		
507		
508		
509		
510		
511		
512		
513		
514		

```
SETVN500[A234567,DIST,TOOL25];
M02;
%
```

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

13.6.5.2 Local Variables (#1 to #33)

Detailed description

Local variables can be defined as an <argument> when a macro subprogram is called, and also used locally within main programs and subprograms. They can be duplicated because there is no relationship between macros. (up to 4 levels)

G65 P__ L__ <argument> ;

P	Program No.
L	Number of repetitions

The <argument> is assumed to be Aa1 Bb1 Cc1...... Zz1.
The following table shows the correspondences between the addresses designated by <argument> and the local variable numbers used in the user macro main bodies.

[Argument designation I]

Call command		Argument address	Local variable No.		Call command		Argument address	Local variable No.
G65 G66	G66.1				G65 G66	G66.1		
○	○	A	#1		○	○	Q	#17
○	○	B	#2		○	○	R	#18
○	○	C	#3		○	○	S	#19
○	○	D	#7		○	○	T	#20
○	○	E	#8		○	○	U	#21
○	○	F	#9		○	○	V	#22
×	× *	G	#10		○	○	W	#23
○	○	H	#11		○	○	X	#24
○	○	I	#4		○	○	Y	#25
○	○	J	#5		○	○	Z	#26
○	○	K	#6				-	#27
×	× *	L	#12				-	#28
○	○	M	#13				-	#29
×	× *	N	#14				-	#30
×	×	O	#15				-	#31
×	× *	P	#16				-	#32
							-	#33

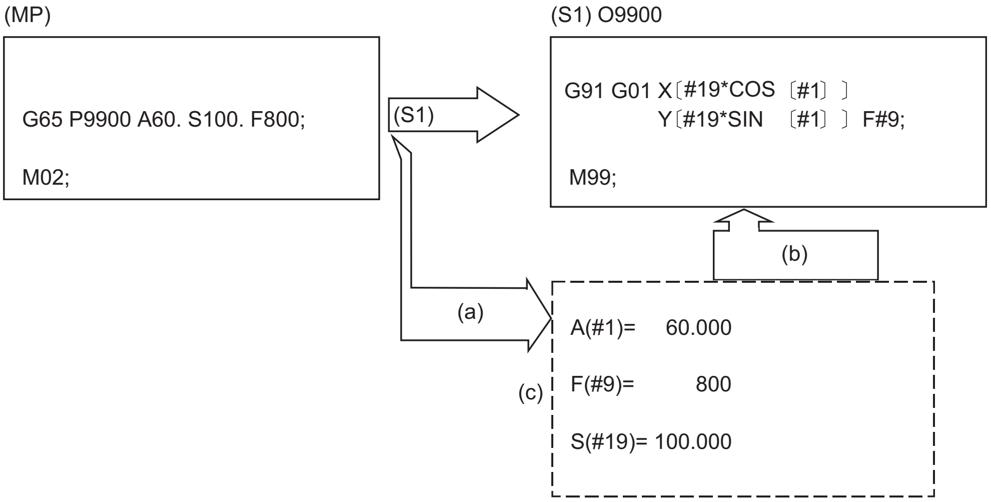
"×" in the above table denotes argument addresses which cannot be used. However, provided that the G66.1 mode has been established, an argument address denoted by the asterisk can be added for use.
The hyphen (-) mark indicates that there is no corresponding address.

[Argument designation II]

Argument designation II address	Variable in macro		Argument designation II address	Variable in macro
A	#1		J5	#17
B	#2		K5	#18
C	#3		I6	#19
I1	#4		J6	#20
J1	#5		K6	#21
K1	#6		I7	#22
I2	#7		J7	#23
J2	#8		K7	#24
K2	#9		I8	#25
I3	#10		J8	#26
J3	#11		K8	#27
K3	#12		I9	#28
I4	#13		J9	#29
J4	#14		K9	#30
K4	#15		I10	#31
I5	#16		J10	#32
			K10	#33

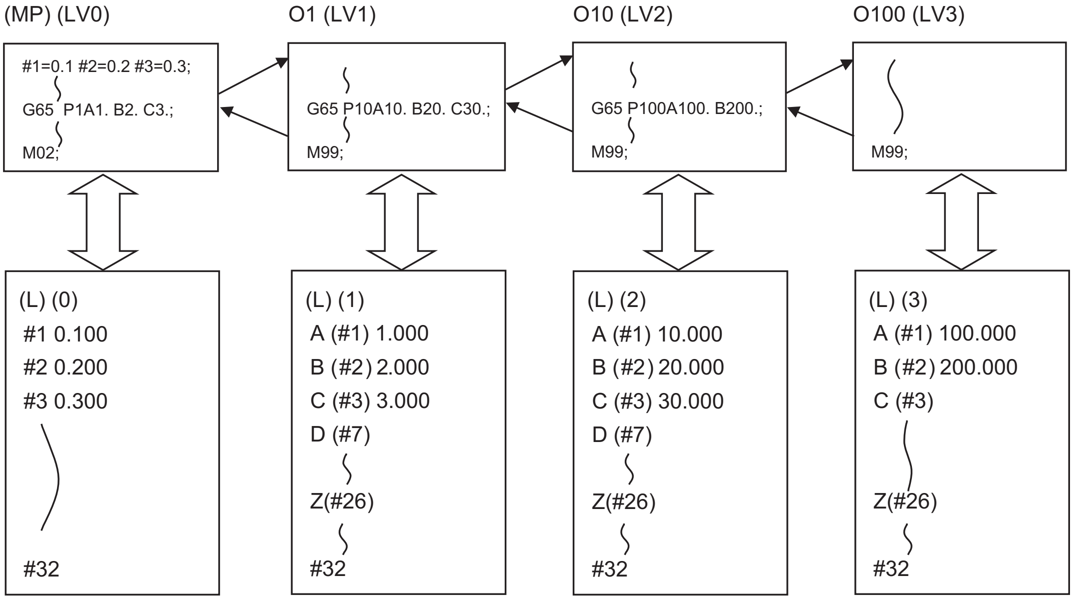
(Note 1) The numbers 1 to 10 accompanying I, J and K indicate the sequence of the commanded sets, and are not required in the actual command.

(1) Local variables in subprograms can be defined by means of the <argument> designation during macro call. (Local variables can be used freely in those subprograms.)



(MP) Main program
(S1) Subprogram
(a) Local variables set by argument
(b) Refer to the local variables and control the movement, etc.
(c) Local variable data table

(2) 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.



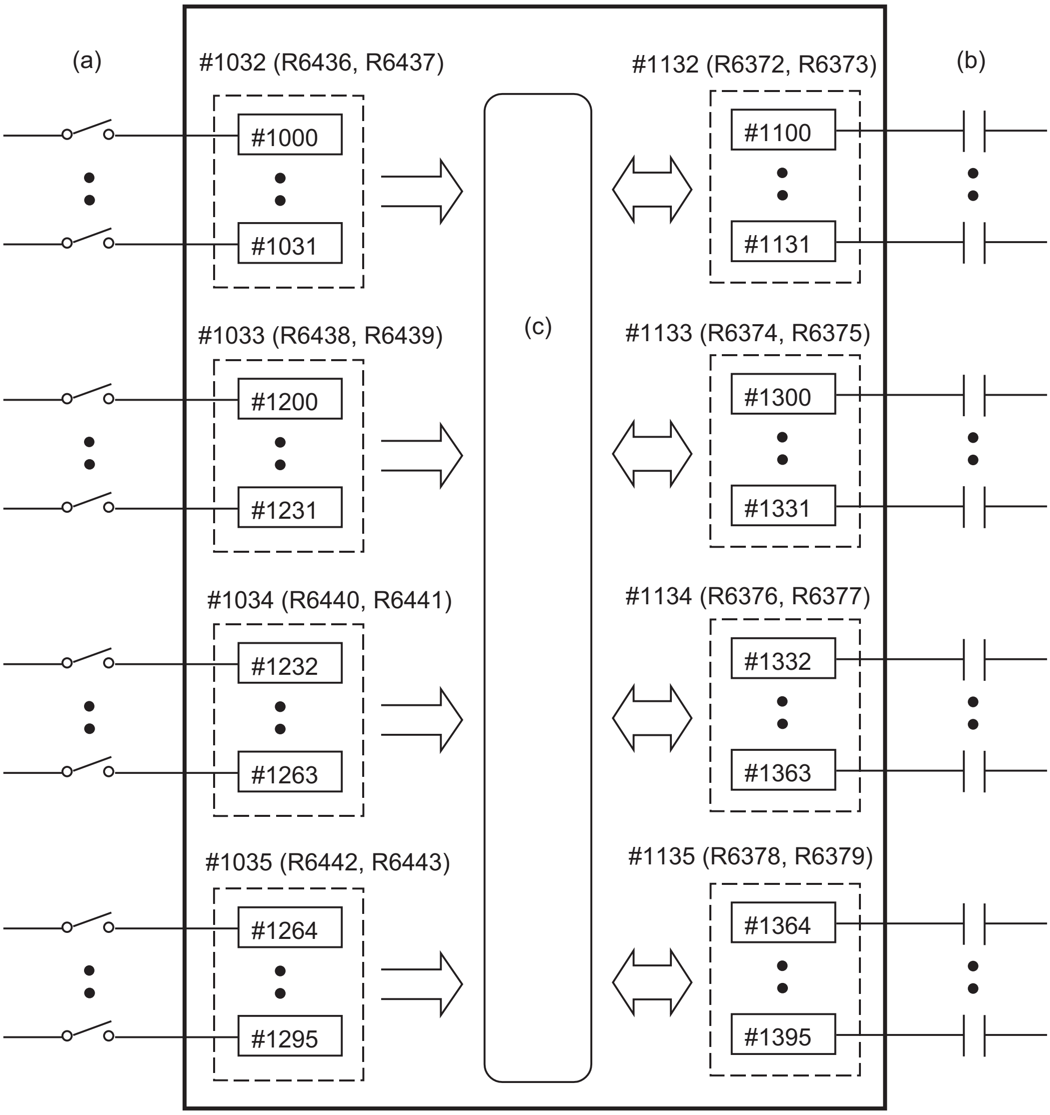
(MP) Main program
(L) Local variables
(LV0 - 3) Macro level 0 - 3

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

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

Function and purpose

The status of the interface input signals can be ascertained by reading out the values of variable numbers #1000 to #1035, #1200 to #1295.
The interface output signals can be sent by substituting values in variable Nos. #1100 to #1135, #1300 to #1395.



(a) Input signal
(b) Output signal
(c) Macro instructions

Detailed description

Macro interface inputs (#1000 to #1035, #1200 to #1295) : PLC -> NC

A variable value which has been read out can be only 1 or 0 (1:contact closed, 0:contact open). All the input signals from #1000 to #1031 can be read at once by reading out the value of variable No. #1032. Similarly, the input signals #1200 to #1231, #1232 to #1263, and #1264 to #1295 can be read by reading the values of the variable Nos. #1033 to #1035.
Variable Nos. #1000 to #1035, #1200 to #1295 are for readout only, and nothing can be placed in the left side member of their operation formula.

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

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

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

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

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

Macro interface outputs (#1100 to #1135, #1300 to #1395) : NC -> PLC

Output signals can only be 0 or 1.
All the output Nos. from #1100 to #1131 can be sent at once by substituting a value in variable No. #1132.
Similarly, the output signals #1300 to #1311, #1332 to #1363, and #1364 to #1395 can be sent by substituting values to the variable Nos. #1133 to #1135. (2^0 to 2^{31})
The status of the writing and output signals can be read in order to compensate the #1100 to #1135, #1300 to #1395 output signals.
Output here refers to the output from the NC side.

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

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

System variable	No. of points	Interface output signal	System variable	No. of points	Interface output signal
#1300	1	Register R6374 bit 0	#1316	1	Register R6375 bit 0
#1301	1	Register R6374 bit 1	#1317	1	Register R6375 bit1
#1302	1	Register R6374 bit 2	#1318	1	Register R6375 bit 2
#1303	1	Register R6374 bit 3	#1319	1	Register R6375 bit 3
#1304	1	Register R6374 bit 4	#1320	1	Register R6375 bit 4
#1305	1	Register R6374 bit 5	#1321	1	Register R6375 bit 5
#1306	1	Register R6374 bit 6	#1322	1	Register R6375 bit 6
#1307	1	Register R6374 bit 7	#1323	1	Register R6375 bit 7
#1308	1	Register R6374 bit 8	#1324	1	Register R6375 bit 8
#1309	1	Register R6374 bit 9	#1325	1	Register R6375 bit 9
#1310	1	Register R6374 bit 10	#1326	1	Register R6375 bit 10
#1311	1	Register R6374 bit 11	#1327	1	Register R6375 bit 11
#1312	1	Register R6374 bit 12	#1328	1	Register R6375 bit 12
#1313	1	Register R6374 bit 13	#1329	1	Register R6375 bit 13
#1314	1	Register R6374 bit 14	#1330	1	Register R6375 bit 14
#1315	1	Register R6374 bit 15	#1331	1	Register R6375 bit 15

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

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

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

(Note 2) The following applies when any number except 1 or 0 is substituted into #1100 to #1131, #1300 to #1395.
<Blank> is treated as 0.
Any number except 0 and <Blank> is treated as 1.
Any value less than 0.00000001 is indefinite.

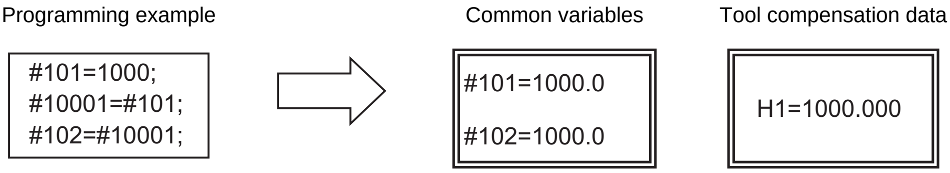
13.6.5.4 Tool Compensation

Detailed description

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

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 variable No. corresponding to the #2000s' shape/wear compensation amount can be changed with the parameter "#1120 TofVal".



Programming example

```
#101=1000;
#10001=#101;
#102=#10001;
```

Common variables

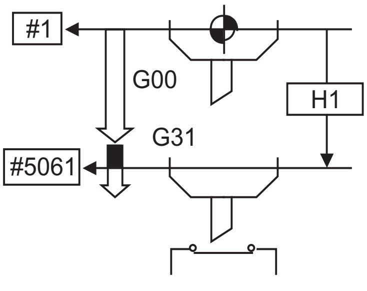
```
#101=1000.0

#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



Sensor

(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.

13.6.5.5 Workpiece Coordinate System Compensation (#5201 - #532n)

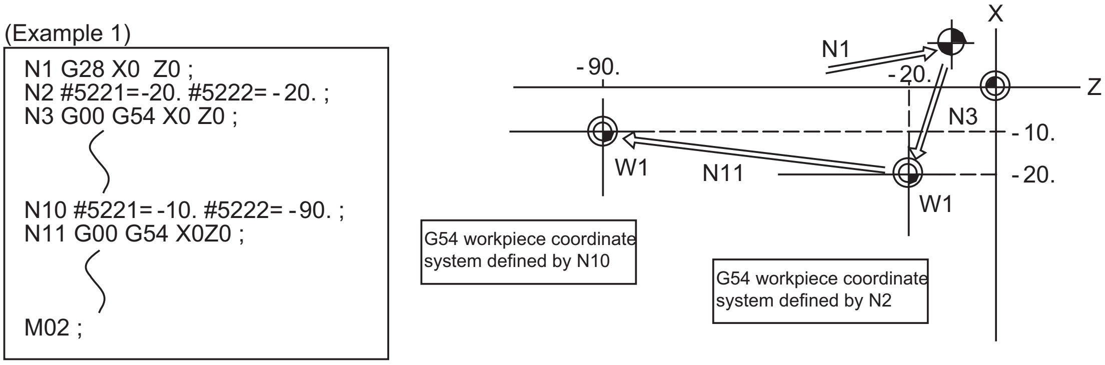
Detailed description

By using variable Nos #5201 to #532n, it is possible to read out the workpiece coordinate system compensation data or to substitute values.

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

Coordinate name	1st axis	2nd axis	3rd axis
External workpiece offset	#5201	#5202	#5203
G54	#5221	#5222	#5223
G55	#5241	#5242	#5243
G56	#5261	#5262	#5263
G57	#5281	#5282	#5283
G58	#5301	#5302	#5303
G59	#5321	#5322	#5323

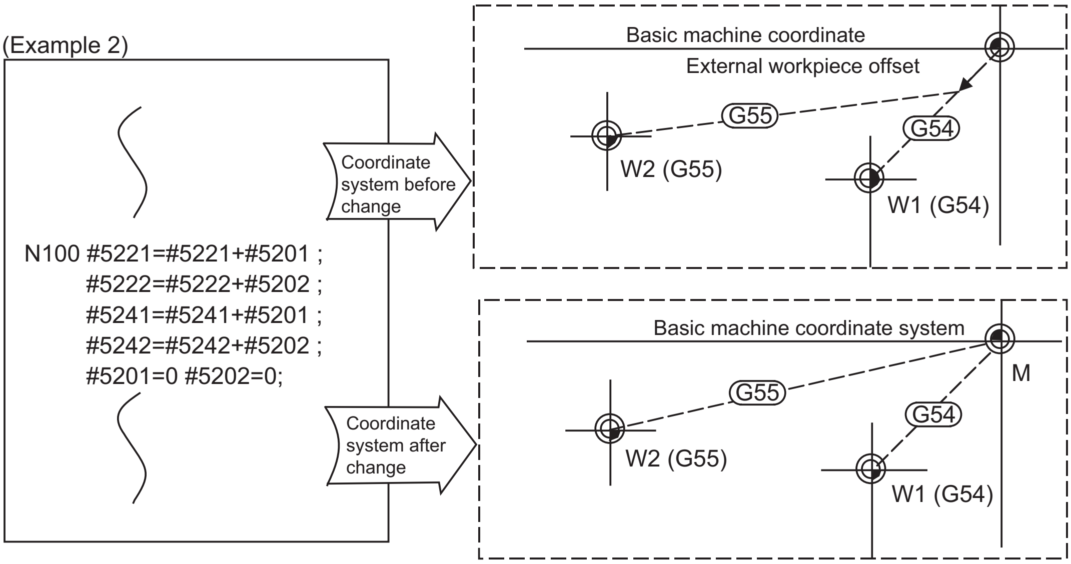
(Example 1)



```
N1 G28 X0  Z0 ;
N2 #5221=-20. #5222= - 20. ;
N3 G00 G54 X0 Z0 ;

N10 #5221= -10. #5222= -90. ;
N11 G00 G54 X0Z0 ;

M02 ;
```

X
Z
N1
-90.
-20.
N3
-10.
-20.
W1
N11
W1
G54 workpiece coordinate system defined by N10
G54 workpiece coordinate system defined by N2

(Example 2)



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

Coordinate system before change
Basic machine coordinate
External workpiece offset
G55
G54
W2 (G55)
W1 (G54)
Coordinate system after change
Basic machine coordinate system
M
G55
G54
W2 (G55)
W1 (G54)

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.

13.6.5.6 NC Alarm (#3000)

Detailed description

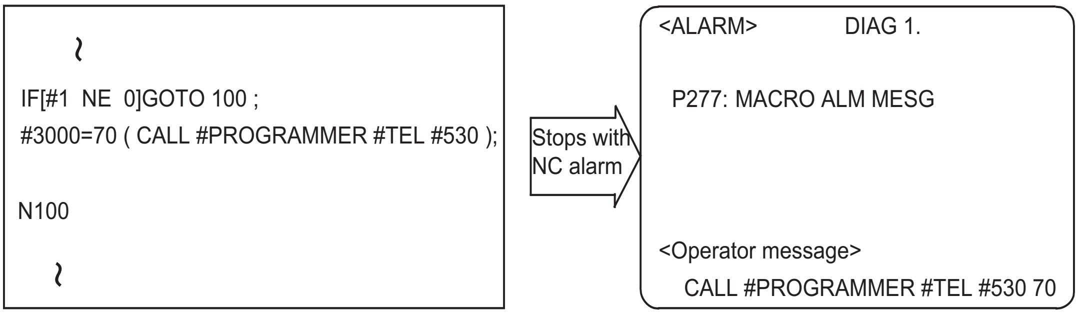
The NC unit can be forcibly set to the alarm state by using variable No. #3000.

#3000= 70 (CALL #PROGRAMMER #TEL #530) ;

70	Alarm No.
CALL #PROGRAMMER #TEL #530	Alarm message

Any alarm number from 1 to 9999 can be specified.
The alarm message must be written in 31 or less characters.
NC alarm 3 signal (program error) is output.
The "P277: MACRO ALM MESG" appears in the <ALARM> column on "DIAG 1." screen and the alarm message " (CALL #PROGRAMMER #TEL #530)" and the alarm No. (70) will appear in the <Operator massage>.

Example of program (alarm when #1 = 0)



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

N100
    ≀
```

```
<ALARM>            DIAG 1.

  P277: MACRO ALM MESG



<Operator message>
   CALL #PROGRAMMER #TEL #530 70
```

(Note 1) If zero or any number greater than 9999 is specified for the alarm No., the number will be invalid and it will not display. However, the operation will be in the alarm status, and the specified alarm message will appear.

(Note 2) Specify the alarm message by enclosing it in round parentheses after the alarm number. If there is any character string between the number and the alarm message enclosed in round parentheses, the alarm message will be invalid and it will not display.
However, the operation will be in the alarm status, and the specified alarm No. will appear.

(Note 3) When 32 or more characters are specified for the alarm message, characters after the 32nd character will not display.

(Note 4) A space included in an alarm message character string is ignored, and characters after the 32nd character will not display. To split the character string insert a character such as "." (period).

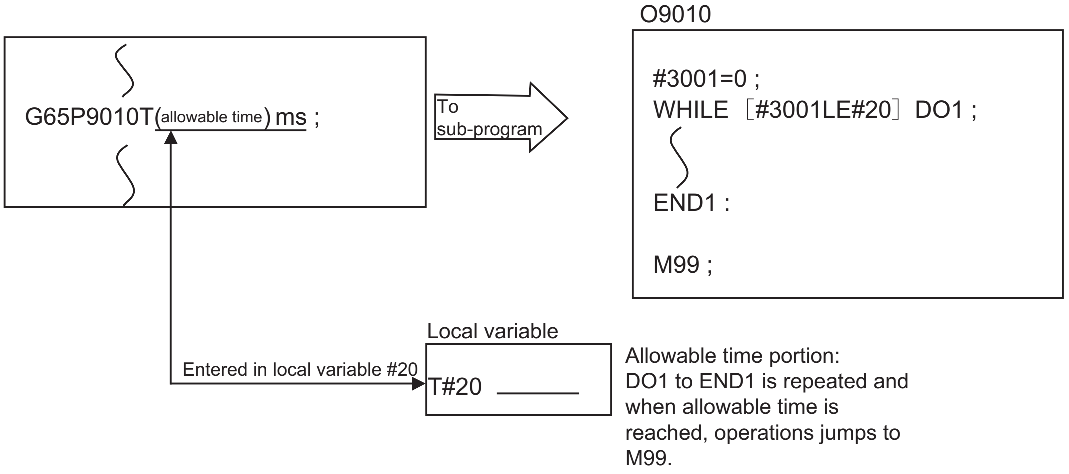
13.6.5.7 Integrating Time (#3001, #3002)

Detailed description

The integrating time during the power is turned ON or the automatic start is running, can be read or values can be substituted by using variable Nos. #3001 and #3002.

Type	Variable No.	Unit	Contents when power is switched on	Initialization of contents	Count condition
Power ON	3001	1ms	Same as when power is switched off	Substitute values to variables	At all times while power is ON
Automatic start	3002				In-automatic start

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


G65P9010T(allowable time) ms ;
To
sub-program
O9010
Local variable
Entered in local variable #20
T#20
Allowable time portion:
DO1 to END1 is repeated and
when allowable time is
reached, operations jumps to
M99.

```
#3001=0 ;
WHILE ［#3001LE#20］ DO1 ;

END1 :

M99 ;
```

13.6.5.8 Suppression of Single Block Stop and Miscellaneous Function Finish Signal Waiting (#3003)

Detailed description

By substituting the values below in variable No. #3003, it is possible to suppress single block stop in the subsequent blocks or to advance to the next block without waiting for the miscellaneous function (M, S, T, B) finish (FIN) signal.

#3003	Single block stop	Miscellaneous function finish signal
0	Not suppressed	Wait
1	Suppressed	Wait
2	Not suppressed	Not wait
3	Suppressed	Not wait

(Note 1) Variable No. #3003 is set to zero by NC reset.

13.6.5.9 Feed Hold, Feedrate Override, G09 Valid/Invalid (#3004)

Detailed description

By substituting the values below in variable No. #3004, it is possible to make the feed hold, feedrate override and G09 functions either valid or invalid in the subsequent blocks.

Contents (value)	#3004		
	Bit 0	Bit 1	Bit 2
	Feed hold	Feedrate override	G09 check
0	Valid	Valid	Valid
1	Invalid	Valid	Valid
2	Valid	Invalid	Valid
3	Invalid	Invalid	Valid
4	Valid	Valid	Invalid
5	Invalid	Valid	Invalid
6	Valid	Invalid	Invalid
7	Invalid	Invalid	Invalid

(Note 1) Variable No. #3004 is set to zero by NC reset.

(Note 2) The functions are valid when the above bits are 0, and invalid when they are 1.

(Note 3) When the feed hold is set to invalid with #3004, the following will occur when the feed hold switch is pressed.
- During thread cutting, block stop will be carried out at the end of the next block of the block where thread cutting is completed.
- During tapping with tap cycle, block stop will be carried out after the operation of R point return.
- In the case other than above, block stop will be carried out after the termination of the currently executing block.

13.6.5.10 Message Display and Stop (#3006)

Detailed description

By using variable No. #3006, the operation stops after the previous block is executed and, if message display data is commanded, the corresponding message and the stop No. will be indicated on the operator message area.

```
#3006 =  1( TAKE FIVE );
```

1 to 9999	Stop No. (When Nos. other than 1 - 9999 are set, the command will be invalidated.)
TAKE FIVE	Message (Nothing will be displayed if no message is designated.)

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

13.6.5.11 G Command Modals (#4001-#4021, #4201-#4221)

Detailed description

Using variable Nos. #4001 to #4021, it is possible to read the modal commands which have been issued in previous blocks.
Similarly, it is possible to read the modals in the block being executed with variable Nos. #4201 to #4221.

Variable No.		Function	
Pre-read block	Execution block		
#4001	#4201	Interpolation mode	G00 : 0, G01 : 1,G 02 : 2, G03 : 3, G33 : 33
#4002	#4202	Plane selection	G17 : 17, G18 : 18, G19 : 19
#4003	#4203	Absolute/incremental	G90 : 90, G91 : 91
#4004	#4204	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, G54.1:54.1
#4013	#4213	Acceleration/deceleration	G61-G64 : 61-64
#4014	#4214	Macro modal call	G66 : 66, G66.1 : 66.1, G67 : 67
#4015	#4215		
#4016	#4216	No variable No.	
#4017	#4217	Constant surface speed control	G96 : 96, G97 : 97
#4018	#4218		
#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 ;
%
```

13.6.5.12 Other Modals (#4101 - #4120, #4301 - #4320)

Detailed description

Using variable Nos. #4101 to #4120, it is possible to read the modal commands which have been issued in previous blocks.
Similarly, it is possible to read the modals in the block being executed with variable Nos. #4301 to #4320.

Variable No.		Modal information	Variable No.		Modal information
Pre-read	Execution		Pre-read	Execution	
#4101	#4301		#4111	#4311	
#4102	#4302	2nd miscellaneous function B	#4112	#4312	
#4103	#4303		#4113	#4313	Miscellaneous function M
#4104	#4304		#4114	#4314	Sequence number N
#4105	#4305		#4115	#4315	Program number O (Note1)
#4106	#4306		#4116	#4316	
#4107	#4307		#4117	#4317	
#4108	#4308		#4118	#4318	
#4109	#4309	Feedrate F	#4119	#4319	Spindle function S
#4110	#4310		#4120	#4320	Tool function T
			#4130	#4330	Extended workpiece coordinate system No. P

(Note 1) The programs are registered as files. When the program No. (file name) is read with #4115, #4315, the character string will be converted to a value.
(Example 1) The file name "123" is the character string 0×31, 0×32, 0×33, so the value will be (0×31-0×30)*100 + (0×32-0×30)*10 + (0×33-0×30) = 123.0. Note that if the file name contains characters other than numbers, it will be "blank".
(Example 2) If the file name is "123ABC", it contains characters other than numbers; so, the result will be "blank".

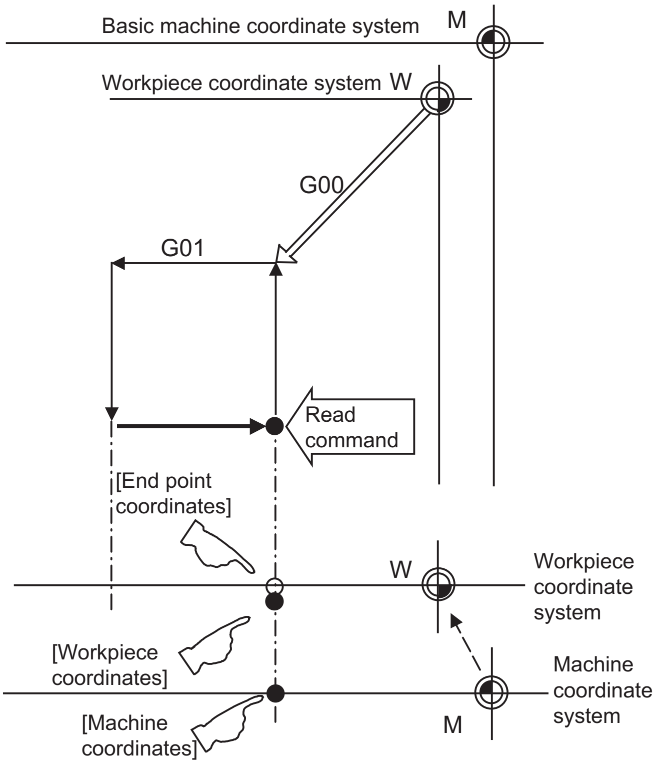
13.6.5.13 Position Information (#5001 - #5140 + n)

Detailed description

Using variable Nos. #5001 to #5104, it is possible to read the end point coordinates, machine coordinates, workpiece coordinates, skip coordinates and servo deviation amounts in the last block.

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

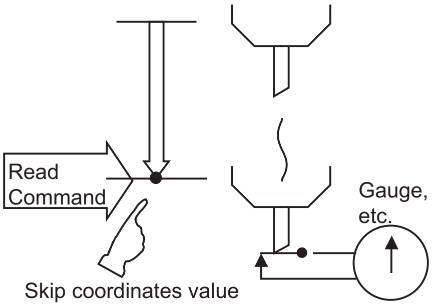
(Note) The number of axes which can be controlled differs according to the specifications.
The last digit of the variable No. corresponds to the control axis No.



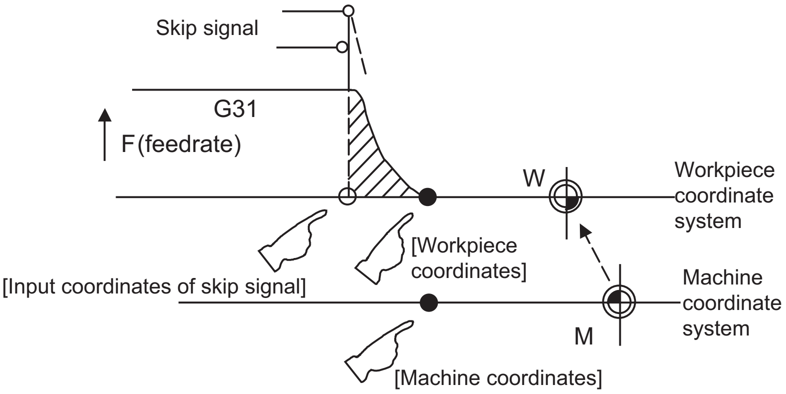
(1) The positions of the end point coordinates and skip coordinates are positions in the workpiece coordinate system.

(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 #5060+n memorize the moments when the skip input signal during movement was input and so they can be read at any subsequent time.
For further details, refer to the section on "Skip Function".

13.6.5.14 External Workpiece Coordinate System Compensation (#2501, #2601)

Detailed description

The workpiece coordinate system compensation amount can be read using variables #2501 and #2601.
By substituting a value in these variable Nos., the workpiece coordinate system compensation amount can be changed.

Axis No.	External workpiece coordinate compensation
1	#2501
2	#2601

13.6.5.15 Number of Workpiece Machining Times (#3901, #3902)

Detailed description

The number of workpiece machining times can be read using variables #3901 and #3902.
By substituting a value in these variable Nos., the number of workpiece machining times can be changed.

Type	Variable No.	Data setting range
Number of workpiece machining times	#3901	0 to 999999
Maximum workpiece value	#3902	

(Note) The number of workpiece machining times must be a positive value.

13.6.5.16 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.	1 to 9999
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 from G10 to G11.
P10 LLn NNn ;      .....  10 is the group No., Ln is the life per tool, Nn is the method.
TTn ;              .....  Tn is the tool No.
  :
G11 ;              .....  Registers the life data with the G10 command.
#111 = #61001 ;    .....  Reads the group 10, #1 tool No.
G10 L3 ;           .....  Starts the life management data registration.
                          The life data other than group 10 is registered from G10 to G11.
P1 LLn NNn ;       .....  1 is the group No., Ln is the life per tool, Nn is the method.
TTn ;              .....  Tn is the tool No.
  :
G11 ;              .....  Registers the life data with the G10 command.
                          (The registered data is deleted.)
#111 = #61001 ;    .....  Group 10 does not exist.  #111 = -1.
```

Precautions

(1) If the tool life management system variable is commanded without designating a group No., the data of the group registered at the head of the registered data will be read.

(2) If a non-registered group No. is designated and the tool life management system variable is commanded, "-1" will be read as the data.

(3) If an unused registration No. tool life management system variable is commanded, "-1" will be read as the data.

(4) Once commanded, the group No. is valid until NC reset.

(5) When the tool life management I specifications are provided, the tool life management system variable specifications are not provided. A program error (P241) will occur if commanded.

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

Function and purpose

Parameter data can be read in with the system variables.

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

Detailed description

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

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

Parameter No. designation (#100000)

The parameter to be read in is designated by substituting the parameter No. in this system variable.
If the parameters are read without designating this No., the parameters will be read in the same manner as if the minimum parameter No. (#1) is designated. Once designated, the setting is held until the parameter No. is designated again or until it is reset.
A program error (P39) will occur if a nonexistent parameter No. is set.

System reserve (#100001)

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

Axis/spindle No. designation (#100002)

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

The axis/spindle No. of the parameter to be read in is designated by substituting an index value in this system variable. This designation will be ignored when reading in parameters that are neither for a specific axis nor spindle.
The spindle parameter's index value is not affected by the part system designation.
If the parameters are read without designating this number, the parameters will be read in the same manner as when the index value 1 (1st axis/1st spindle in the designated part system) is designated.
Once designated, the setting is held until the index value is designated again or until it is reset.
A program error (P39) will occur if a nonexistent axis/spindle No. is set.

(2) Index values

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

Reading the parameters (#100010)

The designated parameter data is read with this system variable.
Data to be read as follows, depending on the parameter type.

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

Program example

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

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

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

[Conditions]

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

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

(3) To read each parameter for axis and spindle

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

Precautions

(1) The number of part systems, axes and spindles is set at the maximum number specified by the model.

(2) The inch/metric changeover function for the setting and display is valid for the readout data.

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

Function and purpose

PLC data can be read in with the system variables.

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

(Note 1) The readable devices are limited.

Detailed description

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

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

Device designation (#100100)

(1) System variable for device designation

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

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

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

(2) Device designation value

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

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

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

Device No. designation (#100101)

The device to be read in is designated by substituting the device No. in this system variable.
Convert a device expressed as a hexadecimal into a decimal when designating.
If the data is read without designating this number, the data will be read in the same manner as when the minimum device No. (0) is designated. Once designated, the setting is held until the device No. is designated again or until it is reset.
A program error (P39) will occur if a nonexistent device No. is set.

Number of bytes designation (#100102)

(1) System variable for number of bytes designation
The reading size is designated by substituting the number of bytes designation value in this system variable.
If the data is read without designating this number, the data will be read in the same manner as when the minimum device designation value (0: M device) is designated. Once designated, the setting is held until the number of bytes is designated again or until it is reset.
A program error (P39) will occur if a number of bytes that does not exist in the specifications is set.

(2) Number of bytes designation value

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

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

Bit designation (#100103)

(1) System variable for bit designation
The bit to be read in is designated by substituting the bit designation value in this system variable.
This designation is valid only when reading the bits for a 16-bit device, and is invalid for the others.
If the data is read without designating this number, the data will be read in the same manner as if the minimum bit designation value (0: bit 0) is designated. Once designated, the setting is held until the bit is designated again or until it is reset.
A program error (P39) will occur if a nonexistent bit is set.

(2) Bit designation value

Bit designation value	Read in bit
0	Bit 0
1	Bit 1
:	:
15	Bit 15

Reading PLC data (#100110)

The data for the designated device is read in with this system variable.
Refer to the table for number of bytes designation for details on the range of data read in.

Program example

To read a bit device

```
#100100 = 0;        ...........    Designates [M device].
#100101 = 0;        ...........    Designates [Device No. 0].
#100102 = 0;        ...........    Designates [Bit].
#100 = #100110;     ...........    Reads M0 (one bit).
#100102 = 1;        ...........    Designates [1 bytes].
#101 = #100110;     ...........    Reads M0 to M7 (8 bits).
                                   (If M7 to M0 is 0001 0010, this will be #102 = 18 (0x12).)
#100102 = 102;      ...........    Designates [Signed two bytes].
#102 = #100110;     ...........    Reads M0 to M15 (16 bits).
                                   (If M15 to M0 is 1111 1110 1101 1100, this will be #102 = -292 (0xFEDC).)
#100102 = 4;        ...........    Designates [4 bytes].
#104 = #100110;     ...........    Reads M0 to M31 (32 bits).
                                   (If M31 to M0 is 0001 0010 0011 0100 0101 0110 0111 1000,
                                   #104 = 305419896 (0x12345678).)
```

(2) To read a word device

```
#100100 = 1;        ...........    Designates [D device].
#100101 = 0;        ...........    Designates [Device No. 0].
#100102 = 0;        ...........    Designates [Bit].
#100103 = 1;        ...........    Designates [Bit 1].
#100 = #100110;     ...........    Reads the D0 bit 1.
                                   (If D0 = 0x0102, #101 =1.)
#100102 = 1;        ...........    Designates [1 bytes].
#101 = #100110;     ...........    Reads the low-order byte of D0.
                                   (If D0 = 0x0102, #101 =2.)
#100102 = 2;        ...........    Designates [2 bytes].
#102 = #100110;     ...........    Reads D0.  (If D0 = 0x0102, #102 =258.)
#100102 = 104;      ...........    Designates [Signed four bytes].
#104 = #100110;     ...........    Reads D0 and D1.
                                   (If D0 = 0xFFFE and D1 = 0xFFFF, #104 =-2.)
```

Precautions

(1) As the PLC data is read asynchronously from the ladder execution, the data is not necessarily the one which was gained when the program was executed. Be careful when reading devices which are changing.

(2) If reading of a nonexistent device is attempted by designating the device No. and number of bytes, the 0 value will be read in only for the nonexistent section.

13.6.5.19 Time Reading Variables (#3001, #3002, #3011, #3012)

Detailed description

The following operations can be carried out using the system variable extension for the user macro time.

(1) By adding time information system variable #3011 and #3012, the current date (#3011) and current time (#3012) can be read and written.

(2) By adding parameter #1273/bit1, the unit (millisecond unit/hour unit) of the cumulative time during system variable #3002 automatic start, can be changed.

Variable No.	Details
#3001	The cumulative time during power ON can be read and the value can be substituted. The unit is millisecond.
#3002	The cumulative time during automatic start can be read and the value can be substituted. The unit can be changed between millisecond and hour with parameter #1273/bit1.
#3011	The current date can be read and written. YYYY/MM/DD is read as a YYYYMMDD value. Command range for year/month/date setting Year (YYYY) : 2000 to 2099 Month (MM) : 1 to 12 Date (DD) : 1 to maximum number of days in month
#3012	The current time can be read and written. HH/MM/SS is read as a value "HHMMSS". When a value "HHMMSS" is written in, it will be set as HH/MM/DD. Command range for hour/minute/second setting Hour (HH) : 0 to 23 (24-hour system) Minute (MM) : 0 to 59 Second (SS) : 0 to 59

(3) The cumulative time returns to 0 at approx. 2.44×10^{11} milliseconds (approx. 7.7 years).

(4) If a negative value or a value exceeding 244335917226 milliseconds (67871.08811851 hours for #3002 time designation) is set for the cumulative time, a program error (P35) will occur.

(5) If a value exceeding the command range is set for the date or time, a program error (P35) will occur.

(6) Always set the month/date/hour/minute/second as a two-digit value when setting the date and time.
If the value only has one digit, always add 0. (February 14, 2001 => #3001= 20010214 ;, etc.)

Program example

Example of use (#3011, #3012)

(Example 1) To read the current date (February 14, 2001) in common variable #100
#100 = #3011 ; (20010214 is inserted in #100)

(Example 2) To write current time (18 hours, 13 minutes, 6 seconds) into system variable #3012
#3012 = 181306 ; (The command value cumulative time #2: time is set to 18:13:06.)

(Example 3) By setting the following program example, the machining start/end time (year/month/date/hour/minute/second) can be viewed.

```
#100=#3011 ;    => Machining start year/month/date
#101=#3012 ;    => Machining start hour/minute/second
G28 X0 Y0 Z0 ;

G92 ;
G0 X50. ;
     :
     :
     :
#102=#3011 ;    .... Machining end year/month/date
#103=#3012 ;    ....  Machining end hour/minute/second
M30 ;
```

Precautions

Limits and precautions for using time reading variable

(1) #3011 reads the date as an eight-digit value, so the difference between the two dates read in will not be the difference of days.

(2) #3012 reads the time as a six-digit value, so the difference between the two times read in will not be the difference of hours.

13.6.5.20 R Device Access Variables (#50000 - #50749, #51000 - #51749, #52000 - #52749)

Function and purpose

By using variable Nos. #50000 to #50749, #51000 to #51749, #52000 to #52749, it is possible to read data (R8300 to R9799, R18300 to R19799, R28300 to R29799) and substitute value in the R device user backup area.

Variable No.	R device	
#50000	R8300,R8301	User backup area (1500 points)
#50001	R8302,R8303	
:		
#50749	R9798,R9799	

Variable No.	R device	
#51000	R18300,R18301	User backup area (1500 points)
#51001	R18302,R18303	
:		
#51749	R19798,R19799	

Variable No.	R device	
#52000	R28300,R28301	User backup area (1500 points)
#52001	R28302,R28303	
:		
#52749	R29798,R29799	

Detailed description

These variables read and write the two words of R device.
Data range of these variables is -2147483648 to 2147483647.
Depending on the setting of the PLC bit selection parameter "#6455/ bit0 to 2", these variables can be changed between decimal point valid or invalid for each user backup area.
The position of the decimal point when decimal point valid is selected, varies according to the parameters "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch	#1003 iunit	
	B	C
0 : Metric	Three digits after the decimal point	Four digits after the decimal point
1 : Inch	Four digits after the decimal point	Five digits after the decimal point

These variables are retained even when the power is off.

Access from a machining program to R device

[Reading variables]

When the variable #50000 is used in a machining program as shown below, the data set in device R8300 and R8301 will be referred.

	Device	Value	#50000
G0 X#50000 ; ↑ └── R8300,R8301	**R8301**	0x0001	0x1e240 (hexadecimal) = 123456 (decimal)
	R8300	0xe240	

(1) When decimal point invalid is selected

Regardless of the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch),
the data set in the R device will be the command value.
In case of the above example, the command value will be "X123456.".

(2) When decimal point valid is selected

The data set in the R device will be read as a data with a decimal point.
The position of the decimal point will be as follows, according to the settings of the parameters "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch	#1003 iunit	
	B	C
0: Metric	X123.456	X12.3456
1: Inch	X12.3456	X1.23456

[Substitution into variables]

When substituting a value to the variable #50000 in a machining program as shown below, data will be set in the device R8302 and R8303.

#50001 = 123 ;
└→ R8302,R8303

(1) When decimal point invalid is selected

Regardless of the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), substituted value will be set in the R device.

#50001	Device	Value
123 (decimal) = 0x7b (hexadecimal)	**R8303**	0x0000
	R8302	0x007b

When a value with a decimal point is substituted to a variable like "#50001 = 123.456 ;", the numbers after the decimal point will be truncated and "123" will be set.

(2) When decimal point valid is selected
According to the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), values which are shifted for the number of decimals will be set in the R device, as shown below.

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50001		123000 (decimal) = 0x1e078 (hexadecimal)	1230000 (decimal) = 0x12c4b0 (hexadecimal)
Device	R8303	0x0001	0x0012
	R8302	0xe078	0xc4b0

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50001		1230000 (decimal) = 0x12c4b0 (hexadecimal)	12300000 (decimal) = 0xbbaee0 (hexadecimal)
Device	R8303	0x0012	0x00bb
	R8302	0xc4b0	0xaee0

When a value with a decimal point is substituted to a variable like "#50001 = 123.456 ;", the value will directly be set.

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50001		123456 (decimal) = 0x1e240 (hexadecimal)	1234560 (decimal) = 0x12d680 (hexadecimal)
Device	R8303	0x0001	0x0012
	R8302	0xe240	0xd680

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50001		1234560 (decimal) = 0x12d680 (hexadecimal)	12345600 (decimal) = 0xbc6100 (hexadecimal)
Device	R8303	0x0012	0x00bc
	R8302	0xd680	0x6100

If the number of decimals of the substituted data exceeds the number of significant figures, the value will be rounded off to the number of significant figures and will be set.
When "#50001 = 123.4567899 ;".

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50001		123457 (decimal) = 0x1e241 (hexadecimal)	1234568 (decimal) = 0x12d688 (hexadecimal)
Device	R8303	0x0001	0x0012
	R8302	0xe241	0xd688

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50001		1234568 (decimal) = 0x12d688 (hexadecimal)	12345679 (decimal) = 0xbc614f (hexadecimal)
Device	R8303	0x0012	0x00bc
	R8302	0xd688	0x614f

Use of R device access variables in control command

These variables can be used in control command.
However, note that the variable value and the condition of true/false differ between decimal point valid variables and invalid variables.

```
IF [#50003 EQ 1] GOTO 30 ;
G00 X100  ;
N30
```

(1) When decimal point invalid is selected
Regardless of the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), R device value of #50003 whose condition is true, will be "1".

#50003	Device	Value
1 (decimal) = 0x01 (hexadecimal)	**R8307**	0x0000
	R8306	0x0001

(2) When decimal point valid is selected
The condition is true when #50003 is "1". So the R device value of #50003 will be as follows depending on the setting of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch		**0 : Metric**	
#1003 iunit		**B**	**C**
#50003		1000 (decimal) = 0x3e8 (hexadecimal)	10000 (decimal) = 0x2710 (hexadecimal)
Device	R8307	0x0000	0x0000
	R8306	0x03e8	0x2710

#1041 I_inch		**1 : Inch**	
#1003 iunit		**B**	**C**
#50003		10000 (decimal) = 0x2710 (hexadecimal)	100000 (decimal) = 0x186a0 (hexadecimal)
Device	R8307	0x0000	0x0001
	R8306	0x2710	0x86a0

Substitution between R device access variable and other variables.

[Substitution into R device access variables]

Common variables and coordinates variables can be substituted to the R device access variables.

(Example 1) Common variables

```
#101 = -123.456 ;
#50004 = #101 ;
```

(Example 2) #5063 : Skip coordinates #5063

```
#50004 = #5063 ;
```

(1) When decimal point invalid is selected
Regardless of the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), the value which is rounded off will be set.
- When the common variable and coordinate variable in the above example are "-123.456"

#50004	Device	Value
-123 (decimal) = 0xffffff85 (hexadecimal)	R8309	0xffff
	R8308	0x0085

(2) When decimal point valid is selected
Substitution will be as follows according to the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch		0 : Metric	
#1003 iunit		B	C
#50004		-123456 (decimal) = 0xfffe1dc0 (hexadecimal)	-1234560 (decimal) = 0xffed2980 (hexadecimal)
Device	R8309	0xfffe	0xffed
	R8308	0x1dc0	0x2980

#1041 I_inch		1 : Inch	
#1003 iunit		B	C
#50004		-1234560 (decimal) = 0xffed2980 (hexadecimal)	-12345600 (decimal) = 0xff439f00 (hexadecimal)
Device	R8309	0xffed	0xff43
	R8308	0x2980	0x9f00

[Substitution of R device access variables]

```
#50005 = 123.456789 ;
#102 = #50005
```

(1) When decimal point invalid is selected
Regardless of the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch), #102 will be "123".

(2) When decimal point valid is selected
Substitution will be as follows according to the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch).

#1041 I_inch	**0: Metric**	
#1003 iunit	**B**	**C**
#102	123.4570	123.4568

#1041 I_inch	**1 : Inch**	
#1003 iunit	**B**	**C**
#102	123.4568	123.4568

Precautions

(1) The position of a decimal point changes depending on the settings of the parameter "#1003 iunit"(inupt setting unit) and "#1041 I_inch"(initial inch). Fix the decimal point position while considering these parameter settings when setting a number to an R device.

(2) These variables do not handle <Blank>. If #0<Blank> is substituted, it will be converted into "0". Therefore, when comparing this variable after substituting #0 #0<Blank> and #0 #0<Blank> with a conditional expression (EQ), it will not be formed.

(3) A program error (P35) will occur when a value exceeding an effective range is substituted into these variables.

(4) When these values are used as decimal point invalid, the settings of "#1078 Decpt2" (Decimal point type 2) will not be applied.

(5) When a graphic is being checked, writing into R device will not be executed even if a value is substituted into these variables.
During a graphic check, the R device value being currently set is read.

13.6.6 Operation Commands

Function and purpose

A variety of operations can be performed between variables.

Command format

#i = <formula> ;

<Formula> is a combination of constants, variables, functions and operators.
Constants can be used instead of #j and #k below.

(1) Definition and substitution of variables	#i = #j	Definition, substitution
(2) Addition operation	#i = #j + #k	Addition
	#i = #j - #k	Subtraction
	#i = #j OR #k	Logical sum (at every bit of 32 bits)
	#i = #j XOR #k	Exclusive OR (at every bit of 32 bits)
(3) Multiplication operation	#i = #j * #k	Multiplication
	#i = #j / #k	Division
	#i = #j MOD #k	Remainder
	#i = #j AND #k	Logical product (at every bit of 32 bits)
(4) Functions	#i = SIN [#k]	Sine
	#i = COS [#k]	Cosine
	#i = TAN [#k]	Tangent tan θ uses sin θ /cos θ .
	#i = ASIN [#k]	Arcsine
	#i = ATAN [#k]	Arctangent (ATAN or ATN may be used)
	#i = ACOS [#k]	Arccosine
	#i = SQRT [#k]	Square root (SQRT or SQR may be used)
	#i = ABS [#k]	Absolute value
	#i = BIN [#k]	Conversion from BCD to BIN
	#i = BCD [#k]	Conversion from BIN to BCD
	#i = ROUND[#k]	Rounding off (ROUND or RND may be used)
	#i = FIX [#k]	Discard fractions less than 1
	#i = FUP [#k]	Add for fractions less than 1
	#i = LN [#k]	Natural logarithm
	#i = EXP [#k]	Exponent with e (=2.718 .....) as bottom

(Note 1) A value without a decimal point is basically treated as a value with a decimal point at the end (1 = 1.000).

(Note 2) Compensation amounts from #10001 and workpiece coordinate system compensation values from #5201 are handled as data with a decimal point. Consequently, data with a decimal point will be produced even when data without a decimal point have been defined in the variable numbers.
(Example)

Operation Commands	Common variables after execution
#101 =1000 ; #10001 =#101 ; #102 =#10001 ;	#101 1000.000 #102 1000.000

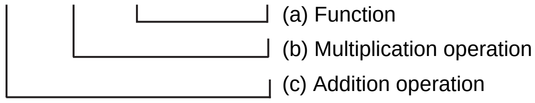
(Note 3) The <formula> after a function must be enclosed in the square parentheses [ ].

Detailed description

Sequence of operations

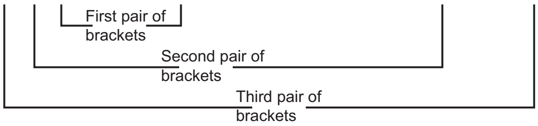
(1) The sequence of the operations (a) to (c) is performed in the following order; the function, the multiplication operation and the addition operation.

#101=#111+#112*SIN〔#113〕



(2) The part to be given priority in the operation sequence should be enclosed in square parentheses [ ]. Up to 5 pairs of such parentheses, including those for the functions, may be used.

#101=SQRT〔〔〔#111-#112〕*SIN〔#113〕+#114〕*#115〕;



Examples of operation commands

(1) Main program and argument designation	G65 P100 A10 B20. ; #101 = 100.000 #102 = 200.000 ;	#1 10.000 #2 20.000 #101 100.000 #102 200.000		
(2) Definition and substitution =	#1 = 1000 #2 = 1000.	#1 1000.000 #2 1000.000		
	#3 = #101 #4 = #102	#3 100.000 #4 200.000	From common variables	
	#5 = #10001 (#10001 = -10.)	#5 -10.000	From compensation amount	
(3) Addition and subtraction + -	#11 = #1 + 1000 #12 = #2 - 50. #13 = #101 + #1 #14 = #10001 - 3. (#10001 = -10.) #15 = #10001 + #102	#11 2000.000 #12 950.000 #13 1100.000 #14 -13.000 #15 190.000		
(4) Multiplication and division * /	#21 = 100 * 100 #22 = 100. * 100 #23 = 100 * 100. #24 = 100. * 100. #25 = 100 / 100 #26 = 100. / 100 #27 = 100 / 100. #28 = 100. / 100. #29 = #10001 * #101 (#10001 = -10.) #30 = #10001 / #102	#21 10000.000 #22 10000.000 #23 10000.000 #24 10000.000 #25 1.000 #26 1.000 #27 1.000 #28 1.000 #29 -1000.000 #30 -0.050		
(5) Remainder MOD	#19 = 48 #20 = 9 #31 = #19 MOD #20	#19/#20 = 48/9 = 5 with 3 over #31 = 3		
(6) Logical sum OR	#3 = 100 #4 = #3 OR 14	#3 = 01100100 (binary) 14 = 00001110 (binary)		
		#4 = 01101110 = 110		
(7) Exclusive OR XOR	#3 = 100 #4 = #3 XOR 14	#3 = 01100100 (binary) 14 = 00001110 (binary)		
		#4 = 01101010 = 106		
(8) Logical product AND	#9 = 100 #10 = #9 AND 15	#9 = 01100100 (binary) 15 = 00001111 (binary)		
		#10 = 00000100 = 4		
(9) Sin SIN	#501 = SIN [60] #502 = SIN [60.] #503 = 1000 * SIN [60] #504 = 1000 * SIN [60.] #505 = 1000. * SIN [60] #506 = 1000. * SIN [60.] (Note) SIN [60] is equivalent to SIN [60.]	#501 #502 #503 #504 #505 #506	0.866 0.866 866.025 866.025 866.025 866.025	
(10) Cosine COS	#541 = COS [45] #542 = COS [45.] #543 = 1000 * COS [45] #544 = 1000 * COS [45.] #545 = 1000. * COS [45] #546 = 1000. * COS [45.] (Note) COS [45] is equivalent to COS [45.]	#541 #542 #543 #544 #545 #546	0.707 0.707 707.107 707.107 707.107 707.107	
(11) Tangent TAN	#551 = TAN [60] #552 = TAN [60.] #553 = 1000 * TAN [60] #554 = 1000 * TAN [60.] #555 = 1000. * TAN [60] #556 = 1000. * TAN [60.] (Note) TAN [60] is equivalent to TAN [60.]	#551 #552 #553 #554 #555 #556	1.732 1.732 1732.051 1732.051 1732.051 1732.051	

(12) Arcsine ASIN	#531 = ASIN [100.500 / 201.] #532 = ASIN [100.500 / 201] #533 = ASIN [0.500] #534 = ASIN [-0.500]	#531 #532 #533 #534	30.000 30.000 30.000 -30.000	
		(Note) When #1273/bit 0 is set to 1, #534 will be 330 ° .		
(13) Arctangent ATN or ATAN	#561 = ATAN [173205 / 100000] #562 = ATAN [173205 / 100000.] #563 = ATAN [173.205 / 100] #564 = ATAN [173.205 / 100.] #565 = ATAN [1.73205]	#561 #562 #563 #564 #565	60.000 60.000 60.000 60.000 60.000	
(14) Arccosine ACOS	#521 = ACOS [100 / 141.421] #522 = ACOS [100. / 141.421]	#521 #522	45.000 45.000	
(15) Square root SQR or SQRT	#571 = SQRT [1000] #572 = SQRT [1000.] #573 = SQRT [10. * 10. + 20. * 20] (Note) In order to increase the accuracy, proceed with the operation inside parentheses as much as possible.	#571 #572 #573	31.623 31.623 22.360	
(16) Absolute value ABS	#576 = -1000 #577 = ABS [#576] #3 = 70. #4 = -50. #580 = ABS [#4 - #3]	#576 #577 #580	-1000.000 1000.000 120.000	
(17) BIN, BCD	#1 = 100 #11 = BIN [#1] #12 = BCD [#1]	#11 #12	64 256	
(18) Rounding off RND or ROUND	#21 = ROUND [14 / 3] #22 = ROUND [14. / 3] #23 = ROUND [14 / 3.] #24 = ROUND [14. / 3.] #25 = ROUND [-14 / 3] #26 = ROUND [-14. / 3] #27 = ROUND [-14 / 3.] #28 = ROUND [-14. / 3.]	#21 #22 #23 #24 #25 #26 #27 #28	5 5 5 5 -5 -5 -5 -5	
(19) Discarding fractions below decimal point FIX	#21 = FIX [14 / 3] #22 = FIX [14. / 3] #23 = FIX [14 / 3.] #24 = FIX [14. / 3.] #25 = FIX [-14 / 3] #26 = FIX [-14. / 3] #27 = FIX [-14 / 3.] #28 = FIX [-14. / 3.]	#21 #22 #23 #24 #25 #26 #27 #28	4.000 4.000 4.000 4.000 -4.000 -4.000 -4.000 -4.000	
(20) Adding fractions less than 1 FUP	#21 = FUP [14 / 3] #22 = FUP [14. / 3] #23 = FUP [14 / 3.] #24 = FUP [14. / 3.] #25 = FUP [-14 / 3] #26 = FUP [-14. / 3] #27 = FUP [-14 / 3.] #28 = FUP [-14. / 3.]	#21 #22 #23 #24 #25 #26 #27 #28	5.000 5.000 5.000 5.000 -5.000 -5.000 -5.000 -5.000	
(21) Natural logarithms LN	#10 = LN [5] #102 = LN [0.5] #103 = LN [-5]	#101 #102 Error	1.609 -0.693 "P282"	
(22) Exponents EXP	#104 = EXP [2] #105 = EXP [1] #106 = EXP [-2]	#104 #105 #106	7.389 2.718 0.135	

Precautions

(1) Notes on logical relation

EQ, NE, GT, LT, GE and LE conduct the same calculation as addition and subtraction. Pay careful attention to errors. For example, to determine whether or not #10 and #20 are equal in the following example, it is not always possible to judge correctly because of the error.

```
IF [#10 EQ #20]
```

Therefore when the difference between #10 and #20 falls within the designated error range, both values should be considered equal.

```
IF [ABS [#10 - #20] LT 200000]
```

13.6.7 Control Commands

Function and purpose

The flow of programs can be controlled by IF-GOTO- and WHILE-DO-END.

Detailed description

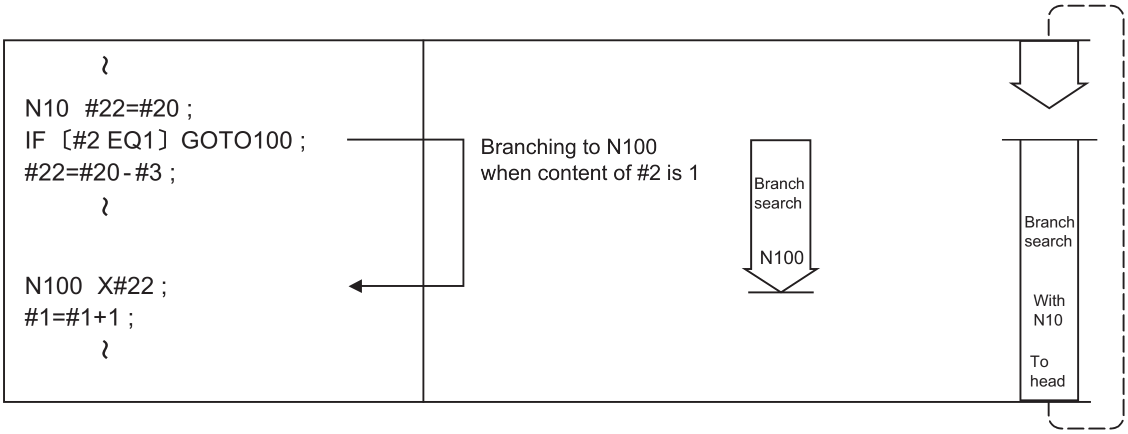
Branching

IF [conditional expression] GOTO n; (n = sequence number in the program)

When the condition is satisfied, control branches to "n" and when it is not satisfied, the next block is executed.
IF [conditional expression] can be omitted and, when it is, control branches to "n" unconditionally.
The following types of [conditional expressions] are available.

#i EQ #j	= When #i and #j are equal
#i NE #j	≠ When #i and #j are not equal
#i GT #j	> When #i is greater than #j
#i LT #j	< When #i is less than #j
#i GE #j	>= When #i is #j or more
#i LE #j	<= When #i is #j or less

"n" of "GOTO n" must always be in the same program. If not, program error (P231) will occur. A formula or variable can be used instead of #i, #j and "n".
In the block with sequence number "n" which will be executed after a "GOTO n" command, the sequence number "Nn" must always be at the head of the block. Otherwise, program error (P231) will occur.
If "/" is at the head of the block and "Nn" follows, control can be branched to the sequence number.



```
        ~
N10  #22=#20 ;
IF〔#2 EQ1〕GOTO100 ;
#22=#20-#3 ;
        ~

N100  X#22 ;
#1=#1+1 ;
        ~
```

(Note 1) When searching the sequence number of the branch destination, the search is conducted up to the end of the program (% code) from the block following IF............; and if it is not found, it is then conducted from the top of the program to the block before IF............;. Therefore, branch searches in the opposite direction to the program flow will take longer time compared with branch searches in the forward direction.

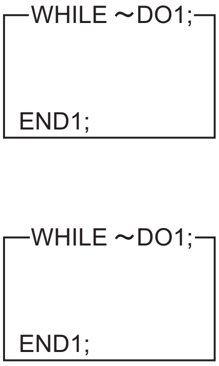
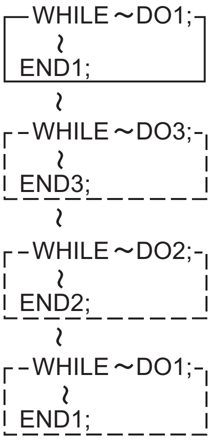
(Note 2) EQ and NE should be compared only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.

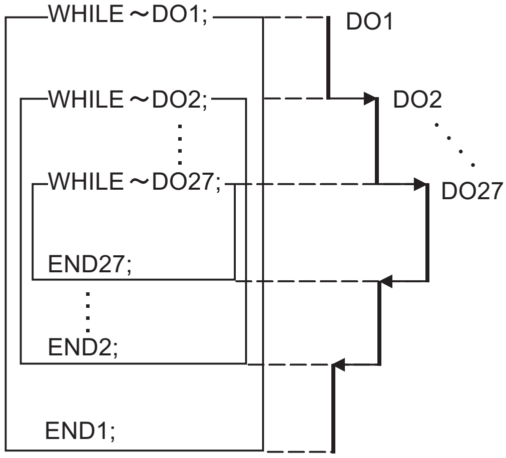
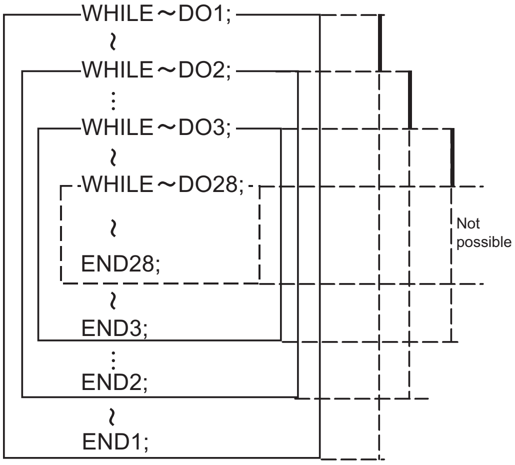
Repetitions

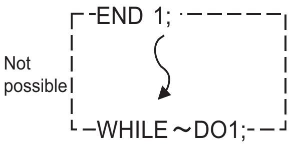
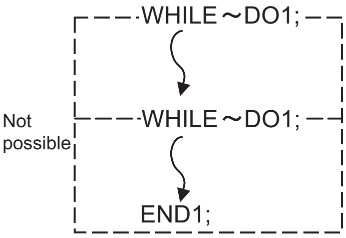
```
WHILE [conditional expression] DOm ; (m =1, 2, 3 ..... 127)
   :
END m ;
```

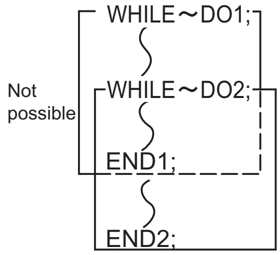
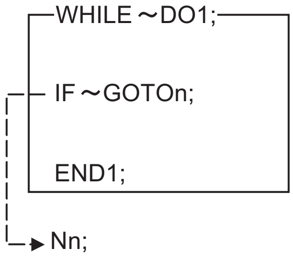
While the conditional expression is established, the blocks from the following block to ENDm are repeatedly executed; when it is not established, execution moves to the block following ENDm. DOm may come before WHILE.

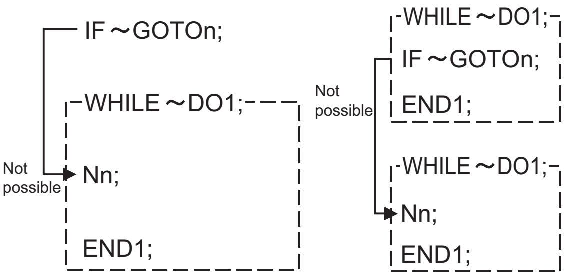
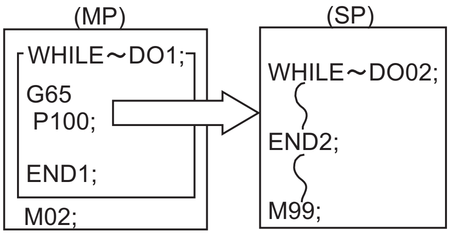
"WHILE [conditional expression] DOm" and "ENDm" must be used as a pair. If "WHILE [conditional expression]" is omitted, these blocks will be repeatedly ad infinitum. The repeating identification Nos. range from 1 to 127. (DO1, DO2, DO3, ... DO127) Up to 27 nesting levels can be used.

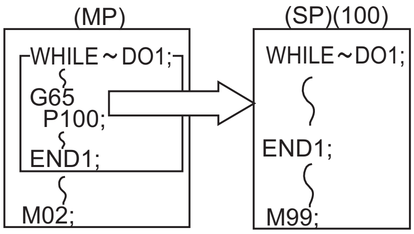
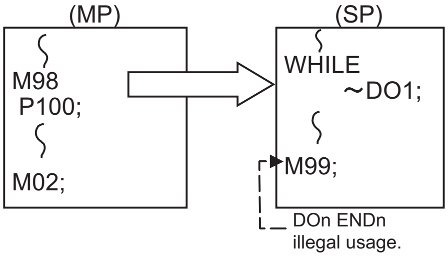
(1) Same identification No. can be used any number of times.

 	
	(2) Any number may be used as the WHILE-DOm identification No.
	
	 WHILE～DO1;
	END1;
	WHILE～DO3;
	END3;
	WHILE～DO2;
	END2;
	WHILE～DO1;
	END1;

(3) Up to 27 nesting levels can be used for WHILE-DOm. "m" is any number from 1 to 127 for the nesting depth.

 WHILE～DO1;
DO1
WHILE～DO2;
DO2
WHILE～DO27;
DO27
END27;
END2;
END1;
(Note) For nesting, "m" which has been used once cannot be used.
	(4) The number of WHILE-DOm nesting levels cannot exceed 27.
	
	 WHILE～DO1;
	WHILE～DO2;
	WHILE～DO3;
	WHILE～DO28;
	END28;
	END3;
	END2;
	END1;
	Not possible

(5) WHILE - DOm must be designated first and ENDm last.  	(6) WHILE - DOm and ENDm must correspond on a 1:1 (pairing) basis in a same program. 

(7) Two WHILE - DOm's must not overlap.  	(8) Branching externally out of the WHILE - DOm range, is possible. 

(9) No branching into WHILE - DOm, is possible.

 IF～GOTOn;
WHILE～DO1;
Not possible
Nn;
END1;
WHILE～DO1;
IF～GOTOn;
Not possible
END1;
WHILE～DO1;
Nn;
END1;
	(10) Subprograms can be called by M98, G65 or G66 between WHILE - DOm's.
	
	 (MP)
	WHILE～DO1;
	G65
	P100;
	END1;
	M02;
	(SP)
	WHILE～DO02;
	END2;
	M99;

(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.

 (MP)
WHILE～DO1;
G65
P100;
END1;
M02;
(SP)(100)
WHILE～DO1;
END1;
M99;
	(12) A program error will occur in M99 if WHILE and END are not paired in the subprogram (including macro subprogram).
	
	 (MP)
	M98
	P100;
	M02;
	(SP)
	WHILE
	～DO1;
	M99;
	DOn ENDn
	illegal usage.

(MP) Main program (SP) Subprogram

(Note) Even if a fixed cycle containing WHILE is called, the nesting level will be counted up.

13.6.8 External Output Commands ; POPEN, PCLOS, DPRNT

Function and purpose

Besides the standard user macro commands, the following macro instructions are also available as external output commands. They are designed to output the variable values or characters to external devices. The data output port can be chosen from RS-232C interface or memory card.

Command format

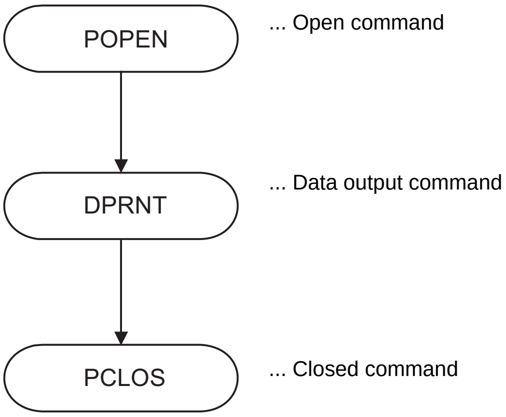
POPEN ... Open command

PCLOS ... Closed command

DPRNT ... Data output command

POPEN	**For preparing the data outputs**
PCLOS	**For terminating the data outputs**
DPRNT	**For character output and digit-by-digit variable numerical output**

Command sequence



Detailed description

Open command : POPEN

(1) The command is issued before the series of data output commands.

(2) The DC2 control code and % code are output from the NC system to the external output device.

(3) Once POPEN; is issued, it will remain valid until PCLOS; is issued.

Close command : PCLOS

(1) This command is issued when all the data outputs are completed.

(2) The DC4 control code and % code are output from the NC system to the external output device.

(3) This command is used together with the open command and it should not be issued unless the open mode has been established.

(4) Issue the close command at the end of the program even when the operation is suspended by resetting or some other operation during data output.

Data output command : DPRNT

DPRNT [l1#v1 [d1 c1] l2#v2 [d2 c2] ......] ;

l1	Character string	
v1	Variable No.	
d1	Significant digits above decimal point	c + d <= 8
c1	Significant digits after decimal point	

(1) The character output and decimal output of the variable values are done with ISO codes.

(2) The commanded character string is output as it is by the ISO code.
Alphanumerics (A to Z, 0 to 9) and special characters (+, -, *, /) can be used.
Note that asterisk (*) is output as a space code.

(3) The required significant digits above and below the decimal point of the variable values are each commanded within square parentheses. As a result, the commanded number of digits of variable values are output in ISO code in decimal notation from the high-order digits including the decimal point. In this case, trailing zeroes are not omitted.

(4) Leading zeroes are omitted.
The omitted leading zero can be replaced by a space by the setting of a parameter.
This can justify the last digit of the data output to the printer.

(5) Linefeed (LF) code will be output to the end of the output data.
And by setting the parameter "#9112 to #9512 DEV0 - 4 CR OUTPUT" to "1", (CR) code will be written in just before EOB (LF) code.

Data output port

(1) The output port can be selected by the parameter "#9007 MACRO PRINT PORT".

(2) When the port is a memory card, the file name of the port can be designated by the parameter "#9054 MACRO PRINT FILE".

(3) When the port is a memory card, the port directory is fixed to root directory.

Example of use

<Parameter setting>

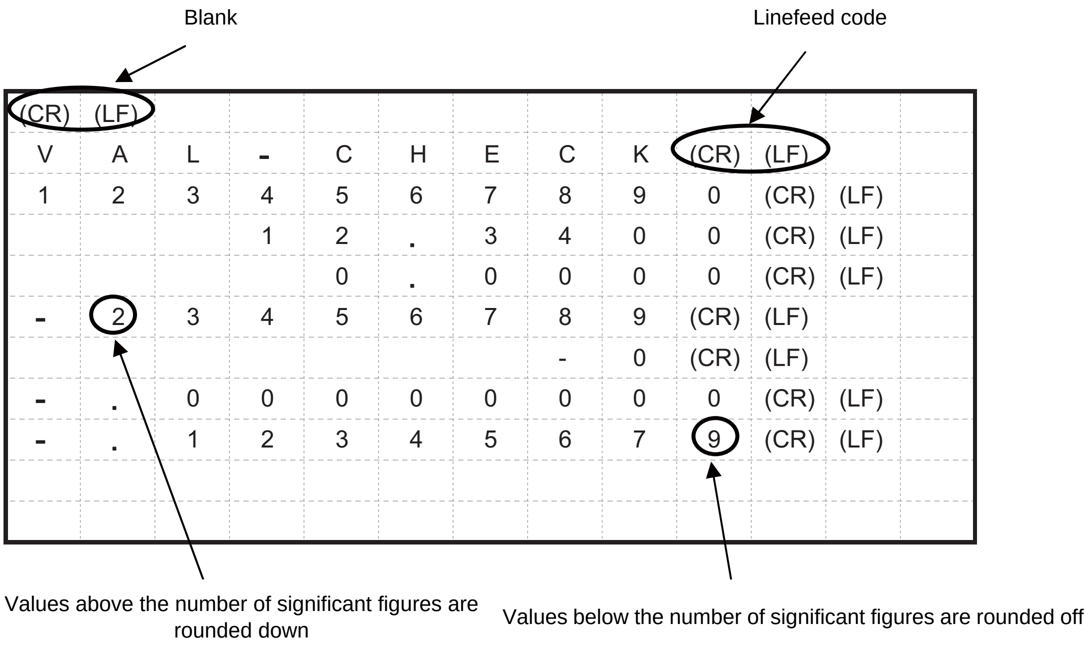
#1127 DPRINT (DPRINT alignment)	= 1 (Align the minimum digit and output)
#9007 MACRO PRINT PORT	= 9 (Output to a memory card by an external output command)
#9008 MACRO PRINT DEV.	= 0 (Device 0 is selected for an external output command)
#9054 MACRO PRINT FILE	= DPRNT_OUT (File name to store output data of an external output command)
#9112 DEV0 CR OUTPUT	= 1 (Insert the CR code just before the LF code)

<Machining program>

```
#1=12.34;
#2=#0
#100=-123456789.;
#500=-0.123456789;
POPEN;
DPRNT[]; (Note)
DPRNT[VAL-CHECK];
DPRNT[1234567890];
DPRNT[#1[44]];
DPRNT[#2[44]];
DPRNT[#100[80]];
DPRNT[#500[80]];
DPRNT[#100[08]];
DPRNT[#500[08]];
PCLOS;
M30;
%
```

(Note) Designate a blank line when opening an output file on an edit screen. If not, it will be regarded that no information is provided in the head line of the file on the NC edit screen.

<Output file (file name = DPRNT_OUT)>


Blank
Linefeed code
Values above the number of significant figures are rounded down
Values below the number of significant figures are rounded off

```
(CR)  (LF)
V     A     L     -     C     H     E     C     K     (CR)  (LF)
1     2     3     4     5     6     7     8     9     0     (CR)  (LF)
                  1     2     .     3     4     0     0     (CR)  (LF)
                        0     .     0     0     0     0     (CR)  (LF)
-     2     3     4     5     6     7     8     9     (CR)  (LF)
                                          -     0     (CR)  (LF)
-     .     0     0     0     0     0     0     0     0     (CR)  (LF)
-     .     1     2     3     4     5     6     7     9     (CR)  (LF)
```

Precautions

(1) An external output command during restart-searching is ignored.
After using the restart-search type 2 to restart-search between a POPEN command and a PCLOS command, execute a POPEN command by such as an MDI interruption before restarting the program.

(2) An external output command during graphic check is ignored.

(3) A program error (P460) will occur if an external output command is issued when the output device is unable to output due to a lack of connection, a low free space etc.

(4) The NC automatically conducts a closing processing when it is reset between a POPEN command and a PCLOS command. So, execute a POPEN command by such as an MDI interruption before executing the rest of the machining program.

(5) If a program error occurs between a POPEN command and a PCLOS command, NC will not automatically conduct a closing processing. So, there is no need to execute a POPEN command by such as an MDI interruption before executing the rest of the machining program.

(6) If a program error occurs to the output port due to the setting of the memory card, execute an NC reset and close the output file before demounting the card.

(7) When the output port is a memory card, the output file may be destroyed if the card is dismounted or the power is turned off without issuing a PCLOS command or NC reset after a POPEN command is issued.

(8) When the data is output to a memory card, the maximum number of files that can be created is determined by the FAT16 format.

13.6.9 Precautions

Precautions

When the user macro commands are employed, it is possible to use the M, S, T and other NC control commands together with the arithmetic, decision, branching and other macro commands for preparing the machining programs. When the former commands are made into executable statements and the latter commands into macro statements, the macro statement processing should be accomplished as quickly as possible in order to minimize the machining time, because such processing is not directly related to machine control.
By setting the parameter "#8101 macro single", the macro statements can be processed 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)
```

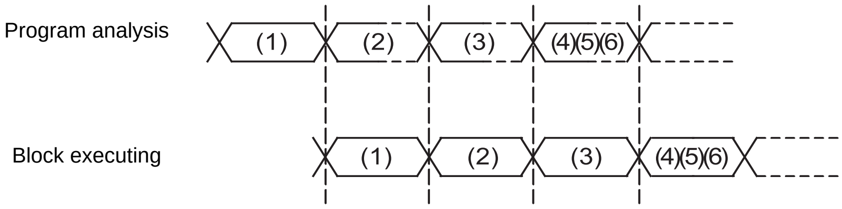
Macro statements are:

(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)

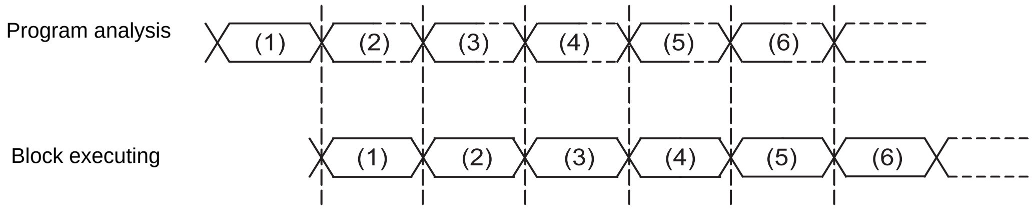
Execution statements refer to statements other than macro statements.

Flow of processing by the Program Example in the previous page

<Macro single OFF>
N4, N5 and N6 are processed in parallel with the control of the executable statement of N3. If the N4, N5 and N6 analysis is in time during N3 control, the machine movement will be continuously controlled.



<Macro single ON>
N4 is processed in parallel with the control of the executable statement of N3. After N3 is finished, N5 and N6 are analyzed, and then N6 is executed. So the machine control is held on standby during the N5 and N6 analysis time.



13.7 Corner Chamfering I/Corner Rounding I

Function and purpose

Chamfering at any angle or corner rounding is performed automatically by adding ",C_" or ",R_" to the end of the block to be commanded first among those command blocks which shape the corner with lines only.
By setting the parameters, "I_", "K_", "C_" can be set for chamfering instead of ",C_", and "R_" can be set for rounding instead of ",R_".

13.7.1 Corner Chamfering I ; G01 X_ Z_ ,C_/I_/K_/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__ (or I__ / K__ / C__) ;
N200 G01 X__ Z__ ;
```

,C / I / K / 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 the basic specification parameter "#1272 ext08/bit6" is "0", the ",C" command will be interpreted as a C command if there is no "," (comma).

(3) If there are multiple or duplicate corner chamfering commands in a same block, the last command will be valid.

(4) When both corner chamfering/corner rounding commands exist 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 chamfering.

(6) 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.

(7) Program error (P383) will occur when the movement amount in the corner chamfering block is less than the chamfering amount.

(8) Program error (P384) will occur when the movement amount is less than the chamfering amount in the block following the corner chamfering block.

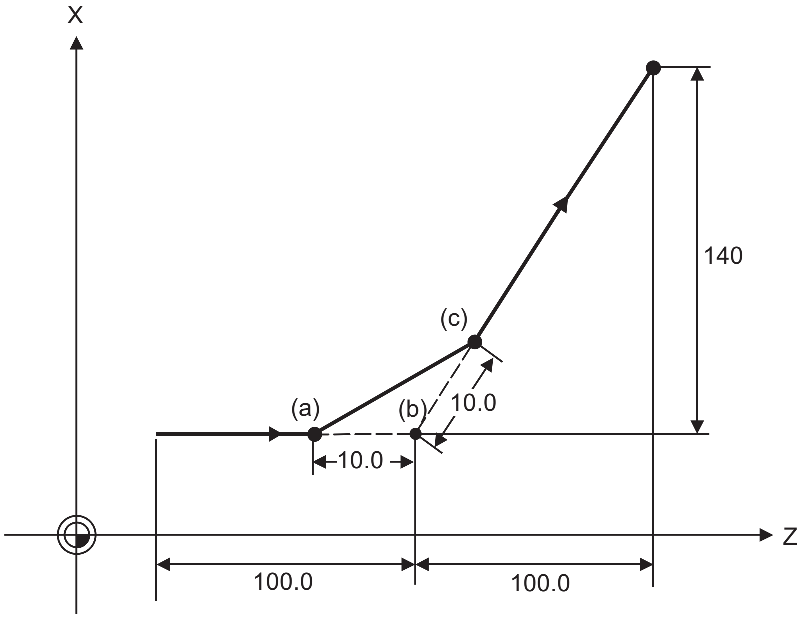
(9) Program error (P382) will occur when a movement command is not issued in the block following the corner chamfering I command.

(10) If "C" is used as the axis name or the 2nd miscellaneous function, corner chamfering cannot be commanded with "C".

(11) Corner chamfering cannot be commanded with "I" or "K" in a circular command block. "I" and "K" are the circular center commands.

Program example

```
G01 W100. ,C10. F100 ;
U280. W100. ;
```



(a) Chamfering start point

(b) Hypothetical corner intersection point

(c) Chamfering end point

Precautions

(1) Corner chamfering and corner rounding can be commanded with "I", "K", "R" only when the 1st block of the corner chamfering/corner rounding command is linear.

(2) Corner chamfering with "I", "K", and corner rounding with "R" can be commanded when the 1st block of the corner chamfering/corner rounding command is linear and the 2nd block is circular. "I", "K" in the 2nd block are the circular center commands.
N100 G01 Xx Zz Ii ;.................. Ii corner chamfering length
N200 G02 Xx Zz Ii Kk ;............. Ii, Kk circular center command

(3) If ",C_", ",R_" or "I_", "K_", "C_", "R_" are commanded in a same block, ",C_", ",R_" will have the priority.

13.7.2 Corner Rounding I ; G01 X_ Z_ ,R_/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__ (or 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) If the basic specification parameter "#1272 ext08/bit6" is "0", 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.

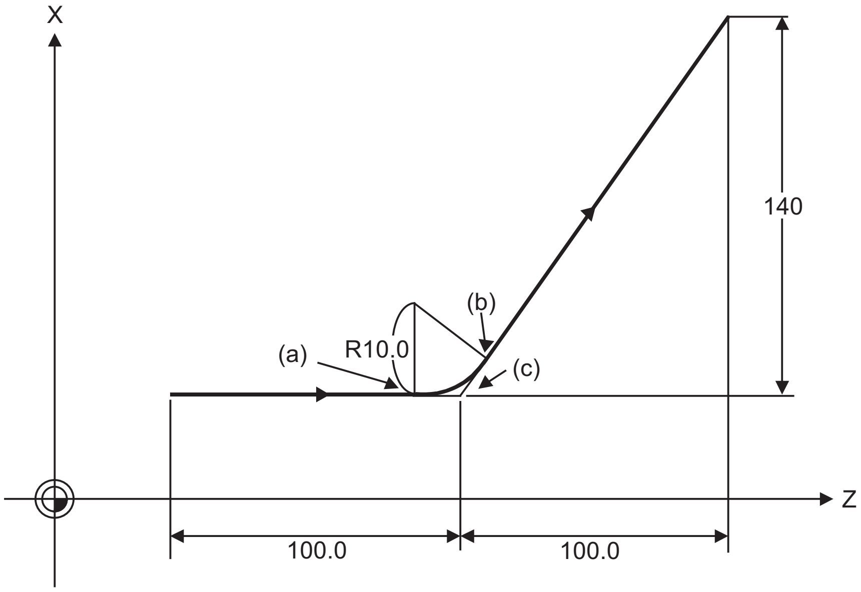
(7) Program error (P384) will occur when the movement amount is less than the R value in the block following the corner rounding.

(8) Program error (P382) will occur if a movement command is not issued in the block following corner rounding.

(9) Corner rounding cannot be commanded with "R" in a circular command block. "R" is the circular radius command.

Program example

```
G01 W100. ,R10. F100 ;
U280. W100. ;
```



(a) Corner rounding start point

(b) Corner rounding end point

(c) Hypothetical corner intersection point

Precautions

(1) Corner chamfering and corner rounding can be commanded with "I", "K", "R" only when the 1st block of the corner chamfering/corner rounding command is linear.

(2) Corner chamfering with "I", "K", and corner rounding with "R" can be commanded when the 1st block of the corner chamfering/corner rounding command is linear and the 2nd block is circular. "I", "K" in the 2nd block are the circular center commands.
N100 G01 Xx Zz Ii ;.................. Ii corner chamfering length
N200 G02 Xx Zz Ii Kk ;............. Ii, Kk circular center command

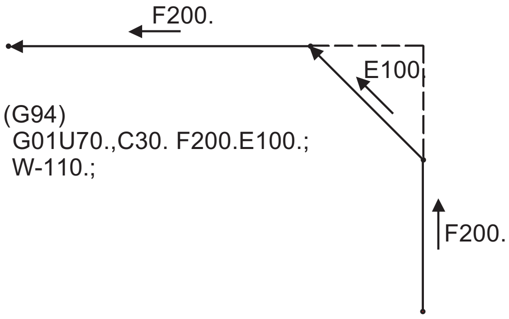
(3) If ",C_", ",R_" or "I_", "K_", "C_", "R_" are commanded in a same block, ",C_", ",R_" will have the priority.

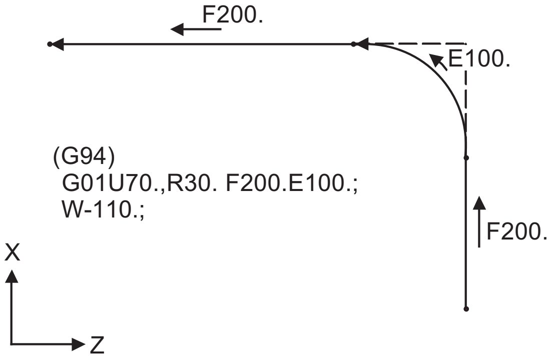
13.7.3 Corner Chamfering Expansion/Corner Rounding Expansion

Function and purpose

Using an E command, the feedrate can be designated for the corner chamfering and corner rounding section. In this way, the corner section can be cut into a correct shape.

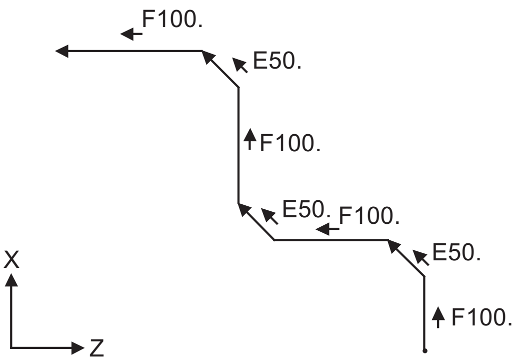
Example





Detailed description

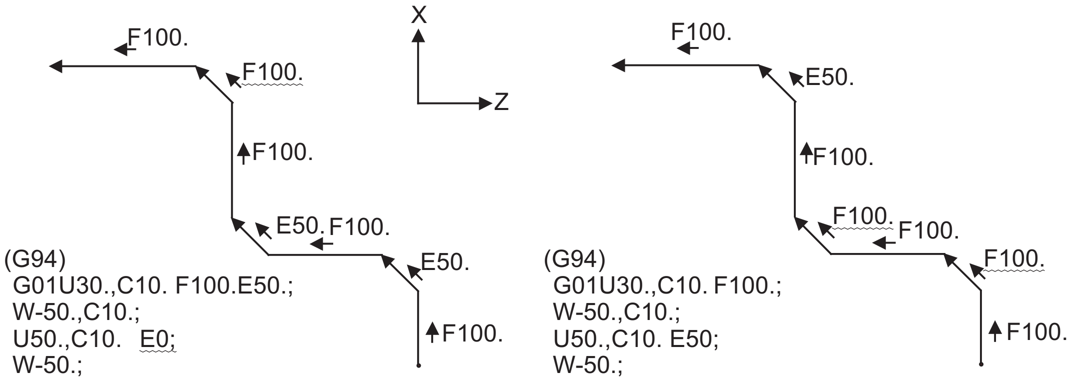
(1) The E command is modal. It is also valid for the feed in the next corner chamfering/corner rounding section.
Example



```
(G94)
 G01U30.,C10. F100.E50.;
 W-50.,C10.;
 U50.,C10.,
 W-50.;
```

(2) E command modal has separate asynchronous feedrate modal and synchronous feedrate modal functions.
Which one is validated depends on the asynchronous/synchronous mode (G94/G95).

(3) When the E command is 0, or when there has not been an E command up to now, the corner chamfering/corner rounding section feedrate will be the same as the F command feedrate.
Example



```
(G94)
 G01U30.,C10. F100.E50.;
 W-50.,C10.;
 U50.,C10.   E0;
 W-50.;
```

```
(G94)
 G01U30.,C10. F100.;
 W-50.,C10.;
 U50.,C10. E50;
 W-50.;
```

(4) E command modal is not cleared even if the reset button is pressed.
It is cleared when the power is turned OFF. (In the same manner as F commands.)

(5) All E commands except those shown below are at the corner chamfering/corner rounding section feedrate.
- E commands during thread cutting modal
- E commands during thread cutting cycle modal

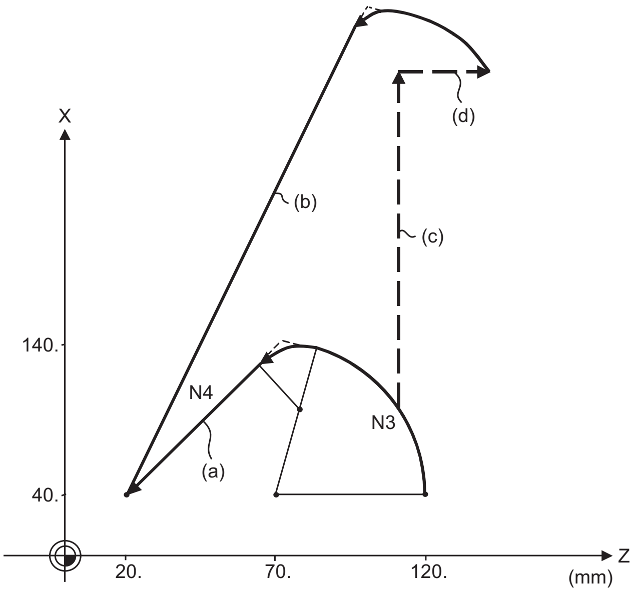
13.7.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding

Detailed description

(1) Shown below are the operations of manual interruption during corner chamfering or corner rounding.

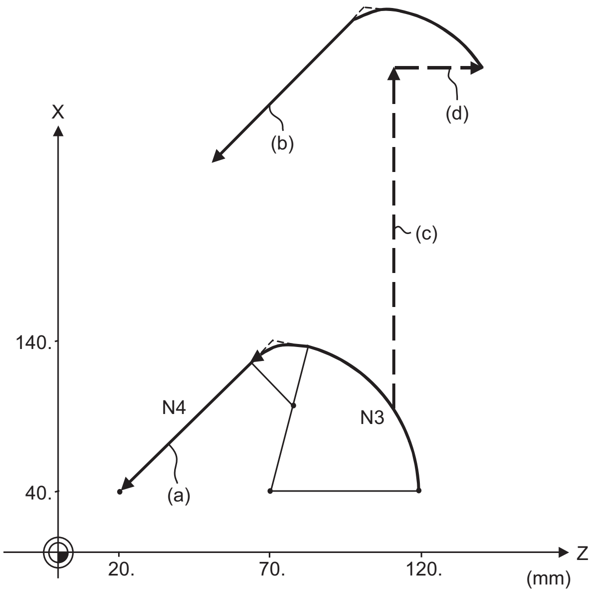
With an absolute value command and manual absolute switch ON.

```
N1 G28 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.

13.8 Corner Chamfering II / Corner Rounding II

Function and purpose

Corner chamfering and corner rounding can be performed by adding ",C" or ",R" to the end of the block which is commanded first among the block that forms a corner with continuous arbitrary angle lines or arcs.

By setting the parameters, "I_", "K_", "C_" can be set for chamfering instead of ",C_", and "R_" can be set for rounding instead of ",R_". Corner chamfering and corner rounding can both be commanded with an absolute value or incremental value.

13.8.1 Corner Chamfering II ; G01/G02/G03 X_ Z_ ,C_/I_/K_/C_

Function and purpose

The corner is chamfered by commanding ",C" (or "I_", "K_", "C_") in the 1st block of the two continuous blocks containing an arc. For an arc, this will be the chord length.

Command format

```
N100 G03 X__  Z__  I__  K__  ,C__ (or C__) ;
N200 G01 X__  Z__ ;
```

,C / 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 start point of the block following the corner chamfering is the hypothetical corner intersection point.

(2) If the basic specification parameter "#1272 ext08/bit6" is "0", the ",C" command will be interpreted as a C command if there is no "," (comma).

(3) If there are multiple or duplicate corner chamfering commands in a same block, the last command will be valid.

(4) When both corner chamfering and corner rounding are commanded in the same block, the latter command will be valid.

(5) Tool compensation is calculated for the shape which has already been subjected to corner chamfering.

(6) Program error (P385) will occur when positioning or thread cutting is commanded in the corner chamfering command block or in the next block.

(7) Program error (P382) will occur when the block following corner chamfering contains a G command other than group 01 or another command.

(8) Program error (P383) will occur when the movement amount in the block, commanding corner chamfering, is less than the chamfering amount.

(9) Program error (P384) will occur when the movement amount is less than the chamfering amount in the block following the block commanding corner chamfering.

(10) Even if a diameter is commanded, it will be handled as a radial command value during corner chamfering.

(11) Program error (P382) will occur when a movement command is not issued in the block following the corner chamfering II command.

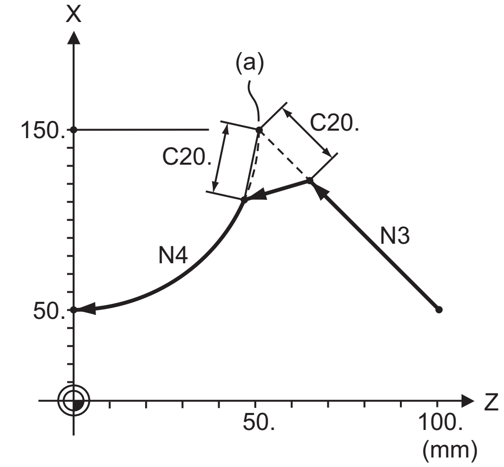
(12) If "C" is used as the axis name or the 2nd miscellaneous function, corner chamfering cannot be commanded with "C".

(13) Corner chamfering cannot be commanded with "I" or "K" in a circular command block. "I" and "K" are the circular center commands.

Program example

(1) Linear - arc

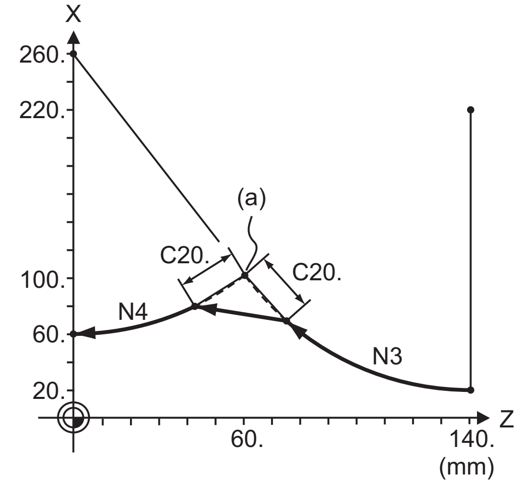
```
Absolute value command
   N1 G28 X Z ;
   N2 G00 X50. Z100. ;
   N3 G01 X150. Z50. ,C20. F100 ;
   N4 G02 X50. Z0 I0 K-50. ;
         :
Relative value command
   N1 G28 X Z ;
   N2 G00 U25. W100. ;
   N3 G01 U50. W-50. ,C20. F100 ;
   N4 G02 U-50. W-50. I0 K-50. ;
         :
```



(a) Hypothetical corner intersection point

(2) Arc - arc

```
Absolute value command
   N1 G28 X Z;
   N2 G00 X20. Z140. ;
   N3 G02 X100. Z60. I100. K0. ,C20. F100 ;
   N4 X60. Z0 I80. K-60. ;
         :
Relative value command
   N1 G28 X Z ;
   N2 G00 U10. W140. ;
   N3 G02 U40. W-80. R100. ,C20. F100 ;
   N4 U-20. W-60. I80. K-60. ;
         :
```



(a) Hypothetical corner intersection point

Precautions

(1) Corner chamfering and corner rounding can be commanded with "I", "K", "R" only when the 1st block of the corner chamfering/corner rounding command is linear.

(2) Corner chamfering with "I", "K", and corner rounding with "R" can be commanded when the 1st block of the corner chamfering/corner rounding command is linear and the 2nd block is circular. "I", "K" in the 2nd block are the circular center commands.

N100 G01 X__ Z__ I__ ; .... I corner chamfering length
N200 G02 X__ Z__ I__ K__ ; .... I,K circular center command

(3) If ",C_", ",R_" or "I_", "K_", "C_", "R_" are commanded in a same block, ",C_", ",R_" will have the priority.

13.8.2 Corner Rounding II ; G01/G02/G03 X_ Z_ ,R_/R_

Function and purpose

The corner is rounded by commanding ",R_" (or "R_") in the 1st block of the two continuous blocks containing an arc.

Command format

```
N100 G03  X__  Z__  I__  K__  ,R__ (or R__) ;
N200 G01  X__  Z__  ;
```

,R / R	Arc 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) If the basic specification parameter "#1272 ext08/bit6" is "0", the ",R" command will be interpreted as a R command if there is no "," (comma).

(3) When both corner chamfering and corner rounding are commanded in a same block, the latter command will be valid.

(4) Tool compensation is calculated for the shape which has already been subjected to corner rounding.

(5) Program error (P385) will occur when positioning or thread cutting is commanded in the corner chamfering command block or in the next block.

(6) Program error (P382) will occur when the block following corner chamfering contains a G command other than group 01 or another command.

(7) Program error (P383) will occur when the movement amount in the corner rounding block is less than the R value.

(8) Program error (P384) will occur when the movement amount is less than the R value in the block following the corner rounding.

(9) Even if a diameter is commanded, it will be handled as a radial command value during corner chamfering.

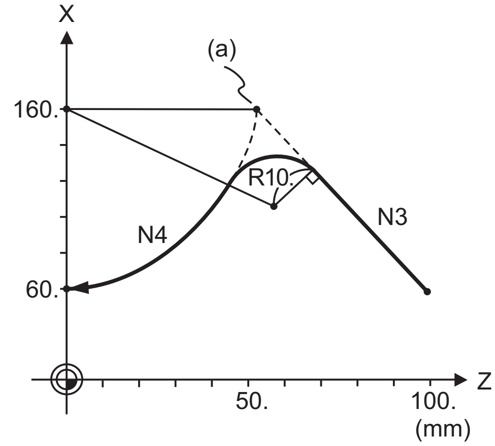
(10) A program error (P382) will occur if a movement command is not issued in the block following corner rounding.

(11) Corner rounding cannot be commanded with "R" in a circular command block. "R" is the circular radius command.

Program example

(1) Linear - arc

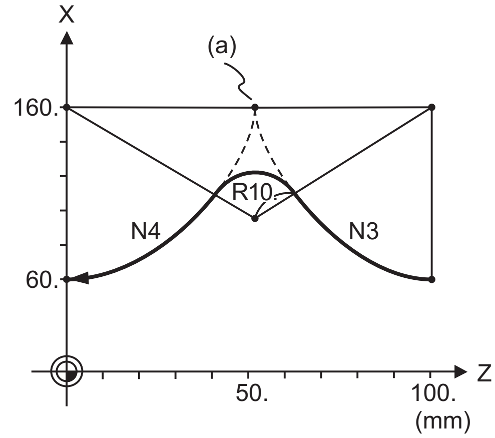
```
Absolute value command
   N1 G28 X Z ;
   N2 G00 X60.  Z100. ;
   N3 G01 X160. Z50. ,R10. F100 ;
   N4 G02 X60.  Z0 I0 K-50. ;
         :
Relative value command
   N1 G28 X Z ;
   N2 G00 U30. W100. ;
   N3 G01 U50. W-50. ,R10. F100 ;
   N4 G02 U-50. W-50. I0 K-50. ;
         :
```



(a) Hypothetical corner intersection point

(2) Arc - arc

```
Absolute value command
   N1 G28 X Z;
   N2 G00 X60. Z100. ;
   N3 G02 X160. Z50. R60 ,R10. F100 ;
   N4 X60. Z0  R50. ;
         :
Relative value command
   N1 G28 X Z ;
   N2 G00 U30. W100. ;
   N3 G02 U50. W-50. I50. K0 ,R10. F100 ;
   N4 U-50. W-50. I0. K-50. ;
         :
```



(a) Hypothetical corner intersection point

Precautions

(1) Corner chamfering and corner rounding can be commanded with "I", "K", "R" only when the 1st block of the corner chamfering/corner rounding command is linear.

(2) Corner chamfering with "I", "K", and corner rounding with "R" can be commanded when the 1st block of the corner chamfering/corner rounding command is linear and the 2nd block is circular. "I", "K" in the 2nd block are the circular center commands.

N100 G01 X__ Z__ I__ ; .... I corner chamfering length
N200 G02 X__ Z__ I__ K__ ; .... I,K circular center command

(3) If ",C_", ",R_" or "I_", "K_", "C_", "R_" are commanded in a same block, ",C_", ",R_" will have the priority.

13.8.3 Corner Chamfering Expansion/Corner Rounding Expansion

For details, refer to "Corner Chamfering I / Corner Rounding I: Corner Chamfering Expansion / Corner Rounding Expansion".

13.8.4 Interrupt during Corner Chamfering/Interrupt during Corner Rounding

For details, refer to "Corner Chamfering I / Corner Rounding I: Interrupt during Corner Chamfering Interrupt during / Corner Rounding".

13.9 Linear Angle Command ; G01 X_/Z_ A_/,A_

Function and purpose

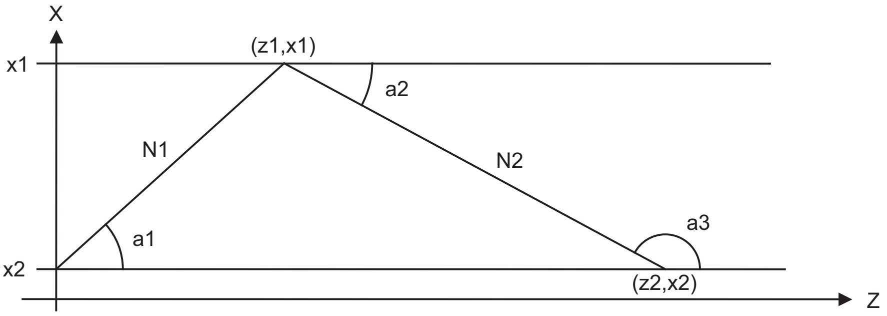
The end point coordinates are automatically calculated by commanding the linear angle and one of the end point coordinate axes.

Command format

N1 G01 Xx1(Zz1) Aa1;
N2 G01 Xx2(Zz2) A-a2; (A-a2 can also be set as Aa 3.)

N1 G01 Xx1(Zz1) ,Aa1;
N2 G01 Xx2(Zz2) ,A-a2;

This designates the angle and the X or Z axis coordinates.
Select the command plane with G17 to G19.



Detailed description

(1) As seen from the + direction of the horizontal axis of the selected plane, the counterclockwise (CCW) direction is considered to be + and the clockwise direction (CW) -.
(2) Either of the axes on the selected plane is commanded for the end point.
(3) The angle is ignored when the angle and the coordinates of both axes are commanded.
(4) When only the angle has been commanded, this is treated as a geometric command.
(5) The angle of either the start point (a1) or end point (a2) may be used.
(6) This function is valid only for the G01 command; it is not valid for other interpolation or positioning commands.
(7) The range of slope "a" is between -360.000 and 360.000.
When a value outside this range is commanded, it will be divided by 360 (degrees) and the remainder will be commanded.
(Example) If 400 is commanded, 40° (remainder of 400/360) will become the command angle.
(8) If an address A is used for the axis name or the 2nd miscellaneous function, use ",A" as the angle.
(9) If "A" and ",A" are commanded in a same block, ",A" will be interpreted as the angle.

13.10 Geometric

13.10.1 Geometric I ; G01 A_

Function and purpose

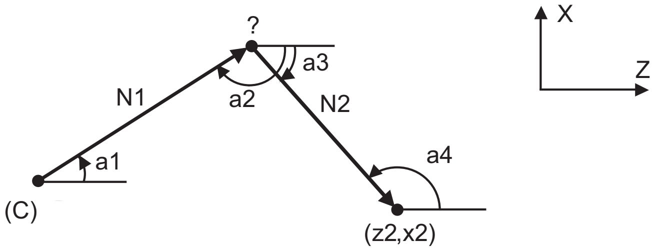
When it is difficult to calculate the intersection point of two straight lines in a continuous linear interpolation command, the end point of the first straight line will be automatically calculated inside the CNC and the movement command will be controlled, provided that the slope of the first straight line as well as the end point coordinates and slope of the second straight line are commanded.

(Note) If the parameter "#1082 Geomet" is set to 0, geometric I will not function.

Command format

```
N1 G01 Aa1 (A-a2) Ff1;
N2 Xx2 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.

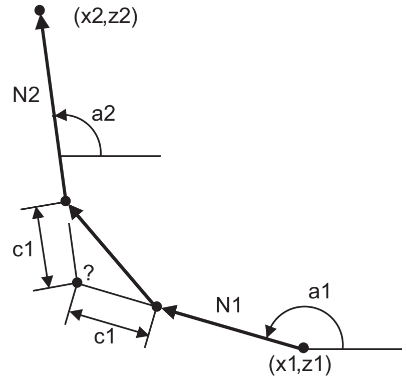
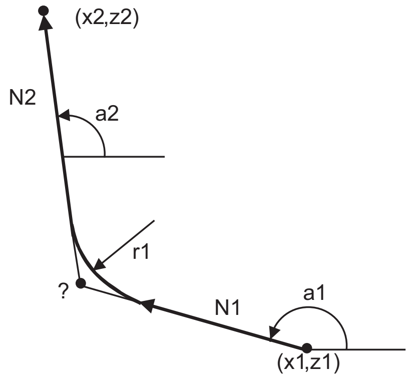
(9) This function is ignored when address A is used for the axis name or as the 2nd miscellaneous function.

(10) Single block stop is possible at the end point of the 1st block.

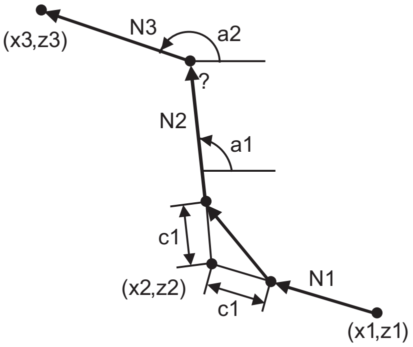
(11) Program error (P394) will occur when the 1st and 2nd blocks do not contain the G01 or G33 command.

Relation with other functions

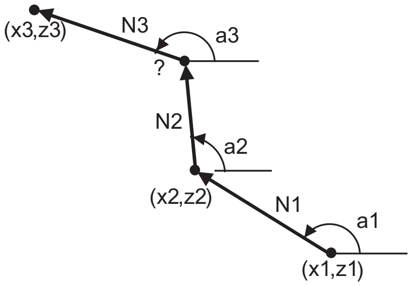
(1) Corner chamfering and corner rounding can be commanded after the angle command in the 1st block.

(Example 1) N1 Aa1 ,Cc1 ; N2 Xx2 Zz2 Aa2 ;	
(Example 2) N1 Aa1 ,Rr1 ; N2 Xx2 Zz2 Aa2 ;	

(2) The geometric command I can be issued after the corner chamfering or corner rounding command.

(Example 3) N1 Xx2 Zz2 ,Cc1 ; N2 Aa1 ; N3 Xx3 Zz3 Aa2 ;	 (x3,z3) N3 a2 ? N2 a1 c1 (x2,z2) c1 N1 (x1,z1)

(3) The geometric command I can be issued after the linear angle command.

(Example 4) N1 Xx2 Aa1 ; N2 Aa2 ; N3 Xx3 Zz3 Aa3 ;	 (x3,z3) N3 a3 ? N2 a2 (x2,z2) N1 a1 (x1,z1)

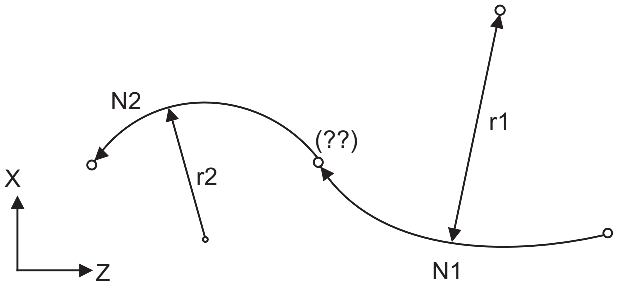
13.10.2 Geometric IB

Function and purpose

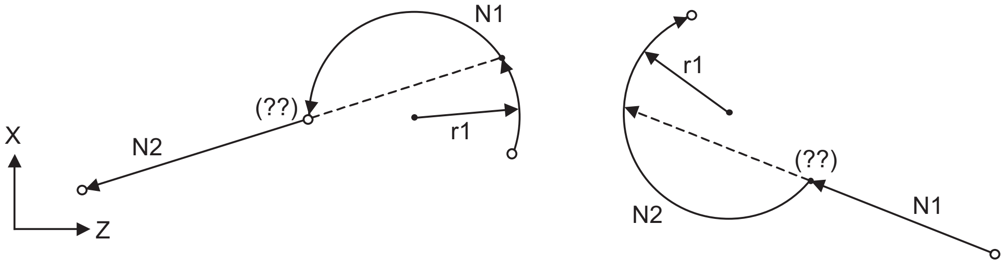
With the geometric IB function, the contact and intersection are calculated by commanding a arc center point or linear angle in the movement commands of two continuous blocks (only blocks with arc commands), instead of commanding the first block end point.

(Note) If the parameter (#1082 Geomet) is not set to 2, geometric IB will not function.

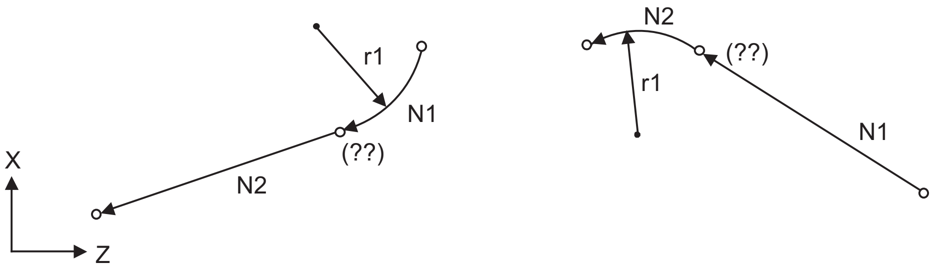
Two-arc contact



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;
```

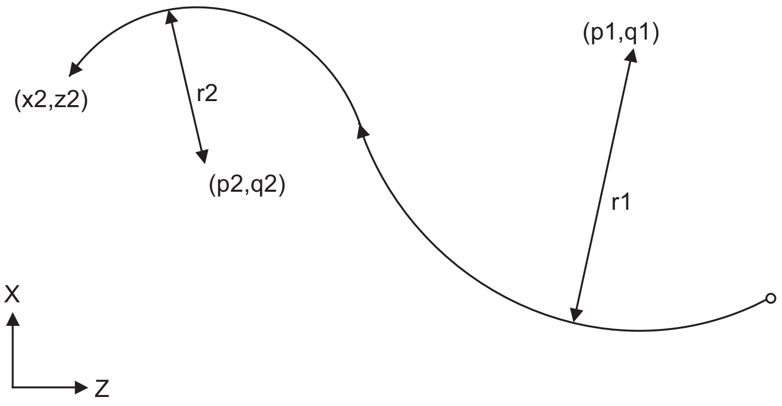
```
N1 G02(G03) Rr1 Ff1;
N2 G03(G02) Xx2 Zz2 Pp2 Qq2 Ff2;
```

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.

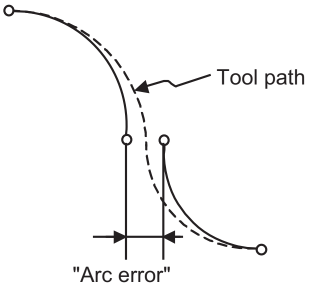
(5) Program error (P397) will occur before the 1st block if two arcs that do not contact are commanded.

(6) The contact calculation accuracy is ±1 μ 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".

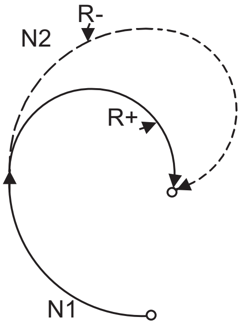


(10) For an arc block perfect circle command (arc block start point = arc block end point), the R designation arc command finishes immediately, and there is no operation. Thus, use a PQ (IK) designation arc command.

(11) G codes of the G modal group 1 in the 1st/2nd block can be omitted.

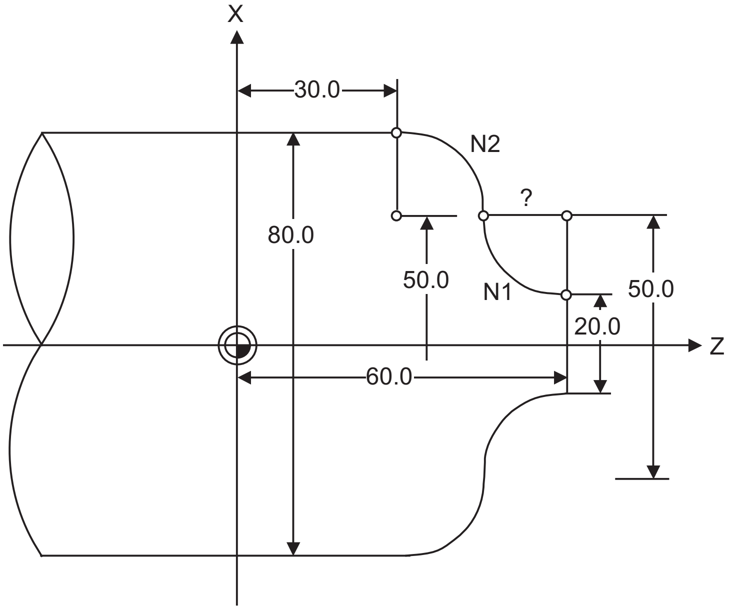
(12) Addresses being used as axis names cannot be used as command addresses for arc center coordinates or arc radius.

(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

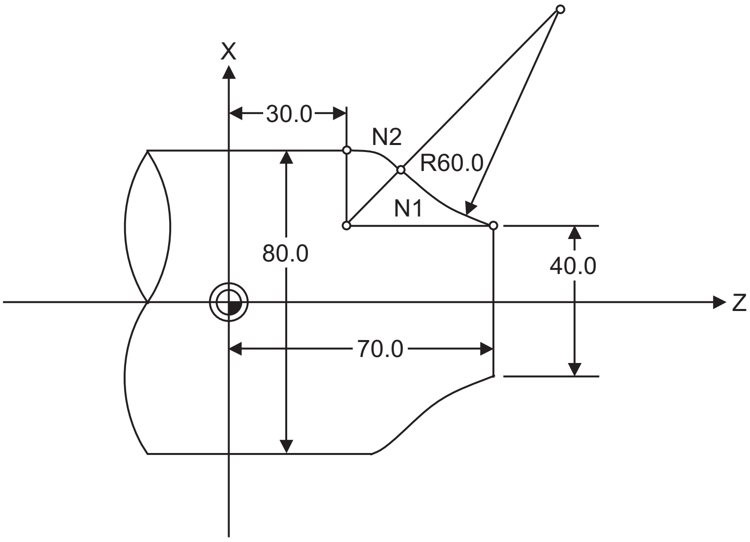
(1) PQ, PQ command



```
   G01 X20.0 Z60.0;
N1 G02 P50.0 Q60.0 F100;
N2 G03 X80.0 Z30.0 P50.0 Q30.0;
```

(mm)

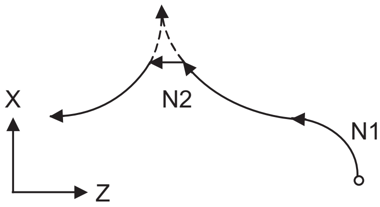
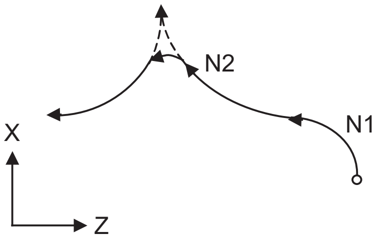
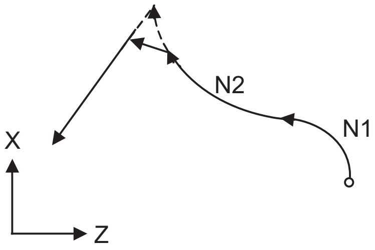
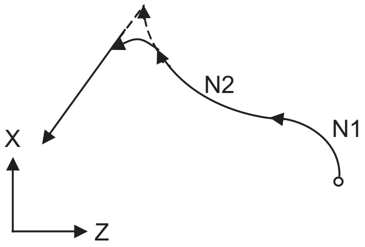
(2) PQ, R command



```
   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.

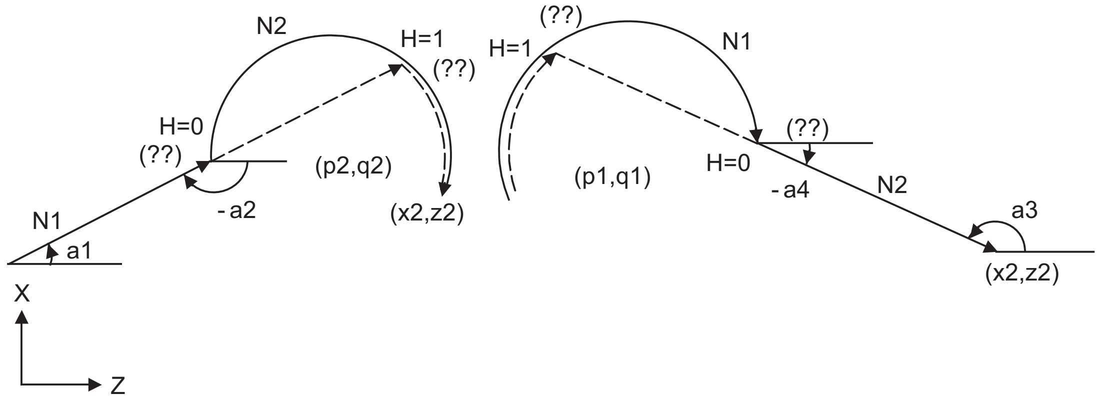
Command format

```
N1 G01 Aa1(A-a2) Ff1 ;
N2 G02(G03) Xx2 Zz2 Pp2 Qq2 Hh2 Ff2 ;
```

```
N1 G02(G03) Pp1 Qq1 Hh1 (,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 (,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.

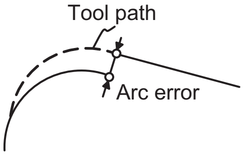
(2) Program error (P393) will occur before the 1st block if the 2nd block is not a coordinate absolute value command.

(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.

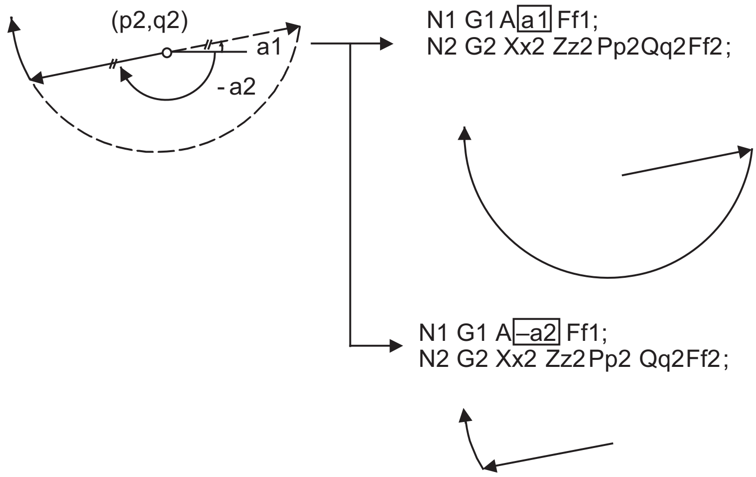
(10) The linear - arc contact is automatically calculated by designating R instead of P, Q (I, K).

(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.



```
N1 G1 A a1 Ff1;
N2 G2 Xx2 Zz2 Pp2Qq2Ff2;
```

```
N1 G1 A–a2 Ff1;
N2 G2 Xx2 Zz2 Pp2 Qq2Ff2;
```

(15) The intersect calculation accuracy is ±1 μ m (fractions rounded up).

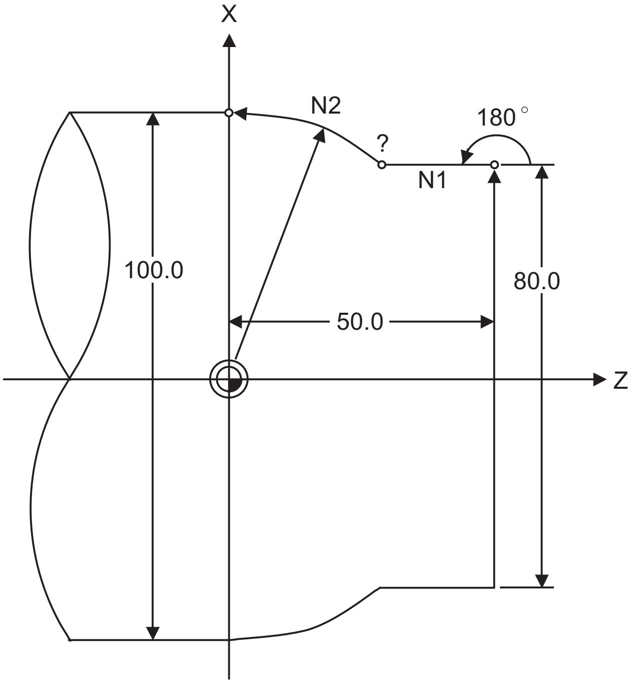
(16) In linear - arc intersections, the arc command can only be PQ (IK) command. When the arc block start point and arc block end point are the same point, the arc is a perfect circle.

(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.

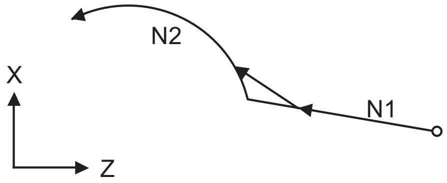
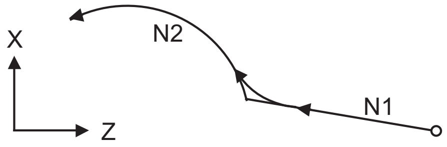
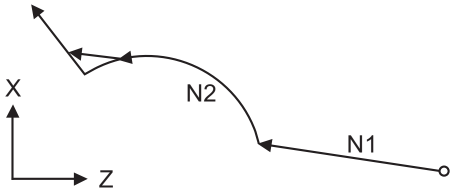
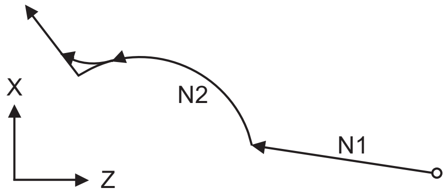
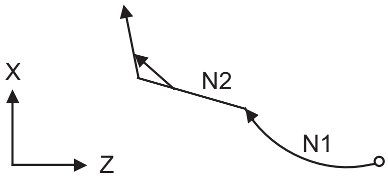
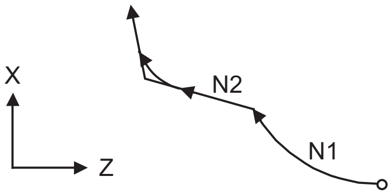
Program example



```
   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

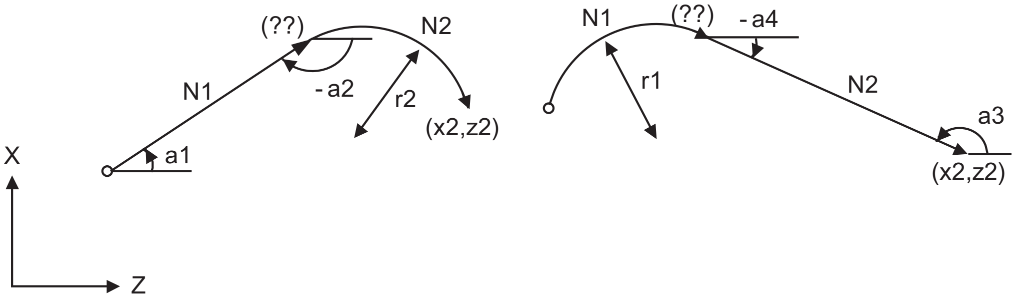
When the intersection of a shape in which a line and arc intersect is not indicated in the drawing, it can be automatically calculated by commanding the following program.

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.

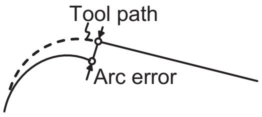
(4) Program error (P396) will occur before the 1st block if another plane selection command (G17 to G19) is issued in the 2nd block.

(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 ±1 μ m (fractions rounded up).

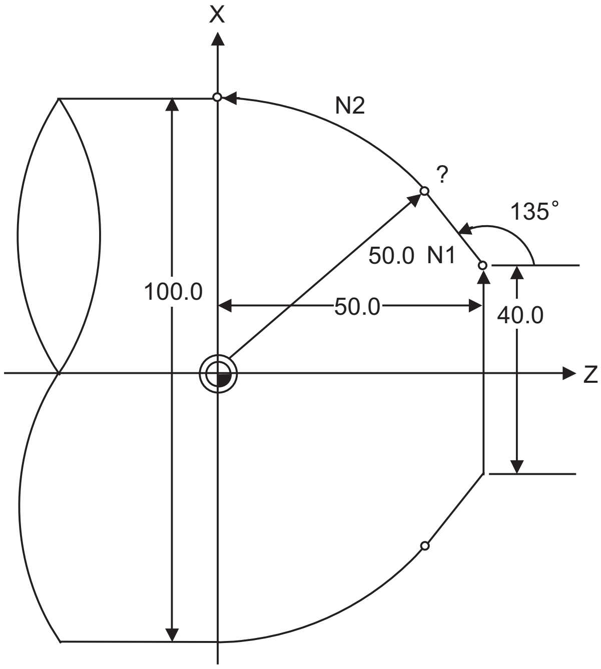
(13) In linear - arc contact, the arc command can only be an R command. Thus, when the arc block start point = arc block end point, the arc command finishes immediately, and there will be no operation.(Perfect circle command is impossible.)

(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.

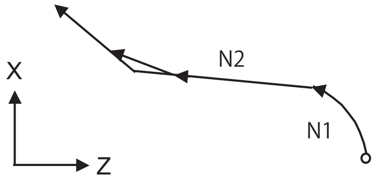
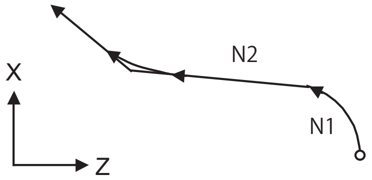
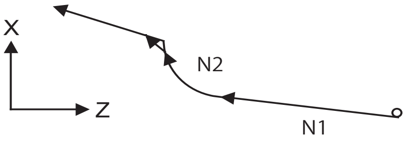
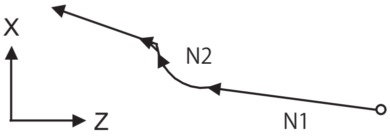
Program example



```
    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 Parameter Input by Program ; G10 L70/L100, G11

Function and purpose

The parameters set from the setting and display unit can be changed in the machining programs.

G10 L70 ... For commanding data with decimal point, and character string data.
The data's command range conforms to the parameter setting range described in Setup Manual.

Command format

G10 L70 ;...Data setting start command

P__ S__ A__ H □ __ ; .... Bit parameter

P__ S__ A__ D__ ; .... Numerical value parameter

P__ S__ A__ <character string> ; ...... Character string parameter ...

P	Parameter No.
S	Part system No.
A	Axis No.
H	Data
D	Data
character string	Data

G11 ; ... Data setting end command

(Note 1) The sequence of addresses in a block must be as shown above.
When an address is commanded two or more times, the last command will be valid.

(Note 2) Set "1" to the part system No.. It is allowed to omit the address S.

(Note 3) The axis No. is set in the following manner. "1" for 1st axis, "2" for 2nd axis, and so forth.
If the address A is omitted, the 1st axis will be applied.
As for the parameters common to axes, the command of axis No. will be ignored.

(Note 4) Address H is commanded with the combination of setting data (0 or 1) and the bit designation □ (0 to 7).

(Note 5) Only the decimal number can be commanded with the address D.
The value that is smaller than the input setting increment (#1003 iunit) will be round off to the nearest increment.

(Note 6) The character string must be put in angled brackets "<" and ">".
If these brackets are not provided, the program error (P33) will occur.
Up to 63 characters can be set.

(Note 7) Command G10 L70, G11 in independent blocks. A program error (P33, P421) will occur if not commanded in independent blocks.

(Note 8) The following data cannot be changed with the G10 L70 command.
Tool compensation data, workpierce coordinate data, PLC switch, and PLC axis parameter.

(Note 9) The settings of the parameters with (PR) in the parameter list will be enabled after the power is turned OFF and ON. Refer to the parameter list in your manual.

Program example

(1)When G10 L70 command

G10 L70;	
P6401 H71 ;	Sets "1" to "#6401 bit7".
P8204 S1 A2 D1.234 ;	Sets "1.234" to "#8204 of the 2nd axis".
P8621 <X> ;	Sets "X" to "#8621".
G11 ;	

13.12 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

User macro interruption enable

```
M96  P__  H__  ;
```

```
M96  <File name> H__  ;
```

P	Interrupt program No. Use a parameter to read out a 4- or 8-digit interrupt program No. starting with O.
<File name>	File name A file name can be specified instead of a program No. In this case, enclose the file name with brackets <>. (The file name can have up to 32 characters including the extension.)
H	Interrupt sequence No.

User macro interruption disable

```
M97  ;
```

Detailed description

(1) The user macro interrupt function is enabled and disabled by the M96 and M97 commands programmed to make the user macro interrupt signal (UIT) valid or invalid. That is, if an interrupt signal (UIT) is input from the machine side in a user macro interruption enable period from when M96 is issued to when M97 is issued or the NC is reset, a user macro interruption is caused to execute the program specified by P__ instead of the one being executed currently.

(2) Another interrupt signal (UIT) is ignored while one user macro interrupt is being in service. It is also ignored in a user macro interrupt disable state such as after an M97 command is issued or the system is reset.

(3) M96 and M97 are processed internally as user macro interrupt control M codes.

(4) If calling a subprogram numbered with O is enabled, a program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command value by adding leading zeros.
(Example)
When parameter "#8129 subprogram number selection"="1", call the subprogram "O0012" with "M96 P12" command.

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

Interrupt enable conditions

A user macro interruption is enabled only during execution of a program.
The requirements for the user macro interrupt are as follows:

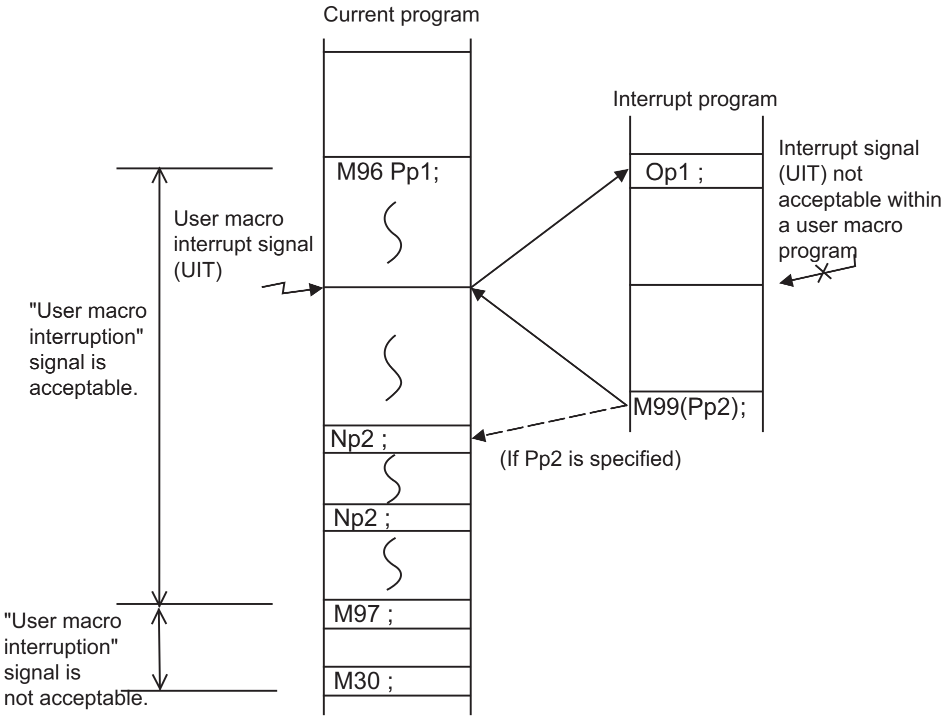
(1) An automatic operation mode or MDI has been selected.
(2) The system is running in automatic mode.
(3) No user macro interruption is being processed.

(Note 1) A macro interruption is disabled in manual operation mode (JOG, STEP, HANDLE, etc.)

Outline of operation

(1) When a user macro interrupt signal (UIT) is input after an M96Pp1 ; command is issued by the current program, interrupt program Op1 is executed. When an M99; command is issued by the interrupt program, control returns to the main program.

(2) If M99Pp2 ; is specified, the blocks from the one next to the interrupted block to the last one are searched for the block with sequence number Np2 ;. Control thus returns to the block with sequence number Np2 that is found first in the above search.


Current program
Interrupt program
M96 Pp1;
Op1 ;
Interrupt signal (UIT) not acceptable within a user macro program
User macro interrupt signal (UIT)
"User macro interruption" signal is acceptable.
M99(Pp2);
Np2 ;
(If Pp2 is specified)
Np2 ;
M97 ;
"User macro interruption" signal is not acceptable.
M30 ;

Interrupt type

Interrupt types 1 and 2 can be selected by the parameter "#1113 INT_2".

[Type 1]

(1) When an interrupt signal (UIT) is input, the system immediately stops moving the tool and interrupts dwell, then permits the interrupt program to run.

(2) If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are lost. After the interrupt program completes, the main program resumes operation from the block next to the interrupted one.

(3) If the interrupted program contains no move and miscellaneous (MSTB) commands, it resumes operation, after completion of the interrupt program, from the point in the block where the interrupt was caused.

If an interrupt signal (UIT) is input during execution of a miscellaneous function (MSTB) command, the NC system waits for a completion signal (FIN). The system thus executes a move or miscellaneous function command (MSTB) in the interrupt program only after input of FIN.

[Type 2]

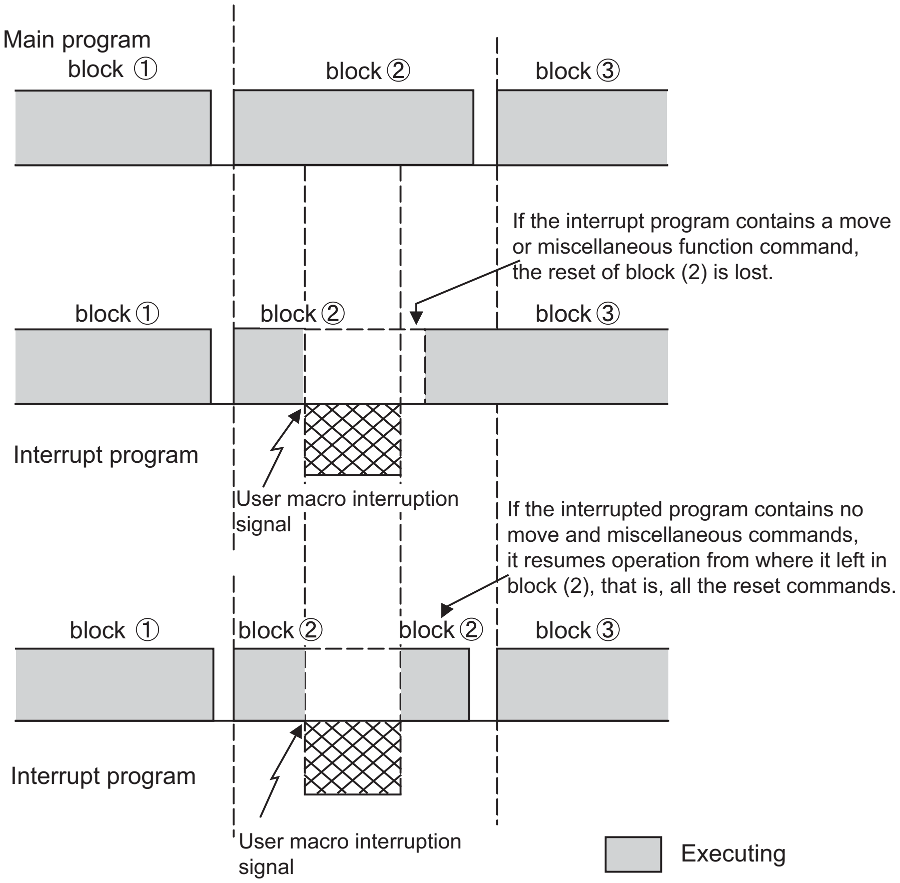
(1) When an interrupt signal (UIT) is input, the interrupt program will be executed without interrupting execution of the current block in parallel with the executing block.

(2) If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are completed, then, these commands will be executed.

(3) If the interrupt program contains no move and miscellaneous function (MSTB) commands, the interrupt program is executed without interrupting execution of the current block.

However, if the interrupt program has not ended even after the execution of the original block is completed, the system may stop machining temporarily.

[Type 1]

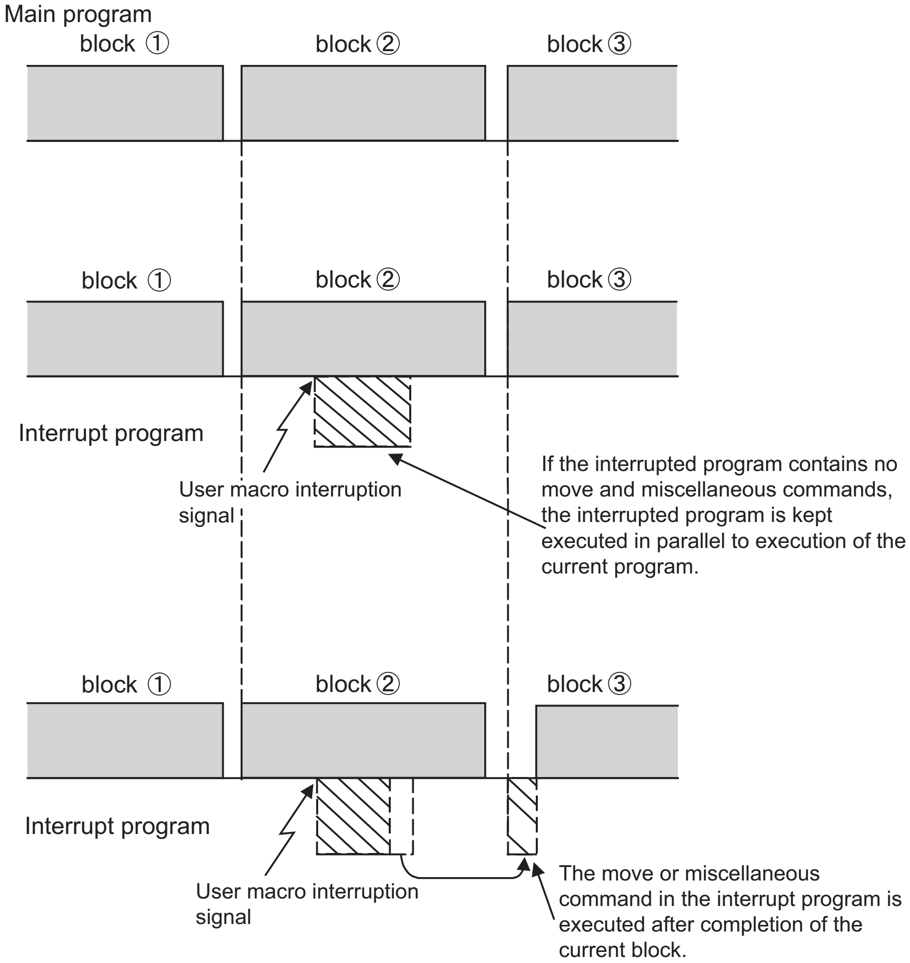


If the interrupt program contains a move or miscellaneous function command, the reset of block (2) is lost.

If the interrupted program contains no move and miscellaneous commands, it resumes operation from where it left in block (2), that is, all the reset commands.

▭ Executing

[Type 2]



If the interrupted program contains no move and miscellaneous commands, the interrupted program is kept executed in parallel to execution of the current program.

The move or miscellaneous command in the interrupt program is executed after completion of the current block.

Calling method

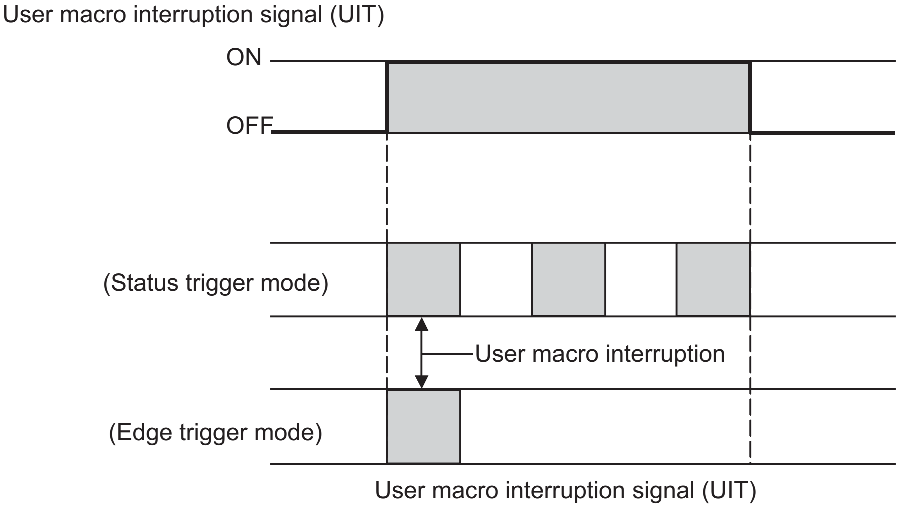
User macro interruption is classified into the following two types depending on the way an interrupt program is called. These two types of interrupt are selected by parameter "#8155 Sub-pro interrupt".
Both types of interrupt are added to the calculation of the nest level. The subprograms and user macros called in the interrupt program are also added to the calculation of the nest level.

Subprogram type interrupt	The user macro interruption program is called as a subprogram. As with calling by M98, the local variable level remains unchanged before and after an interrupt.
Macro type interruption	The user macro interpretation program is called as a user macro. As with calling by G65, the local variable level changes before and after an interrupt. No arguments in the main program can be passed to the interrupt program.

Acceptance of user macro interruption signal (UIT)

A user macro interruption signal (UIT) is accepted in the following two modes: These two modes are selected by a parameter "#1112 S_TRG".

Status trigger mode	The user macro interruption signal (UIT) is accepted as valid when it is ON. If the interrupt signal (UIT) is ON when the user macro interrupt function is enabled by M96, the interrupt program is activated. By keeping the interrupt signal (UIT) ON, the interrupt program can be executed repeatedly.
Edge trigger mode	The user macro interrupt signal (UIT) is accepted as valid at its rising edge, that is, at the instance it turns ON. This mode is useful to execute an interrupt program once.


User macro interruption signal (UIT)
ON
OFF
(Status trigger mode)
User macro interruption
(Edge trigger mode)
User macro interruption signal (UIT)

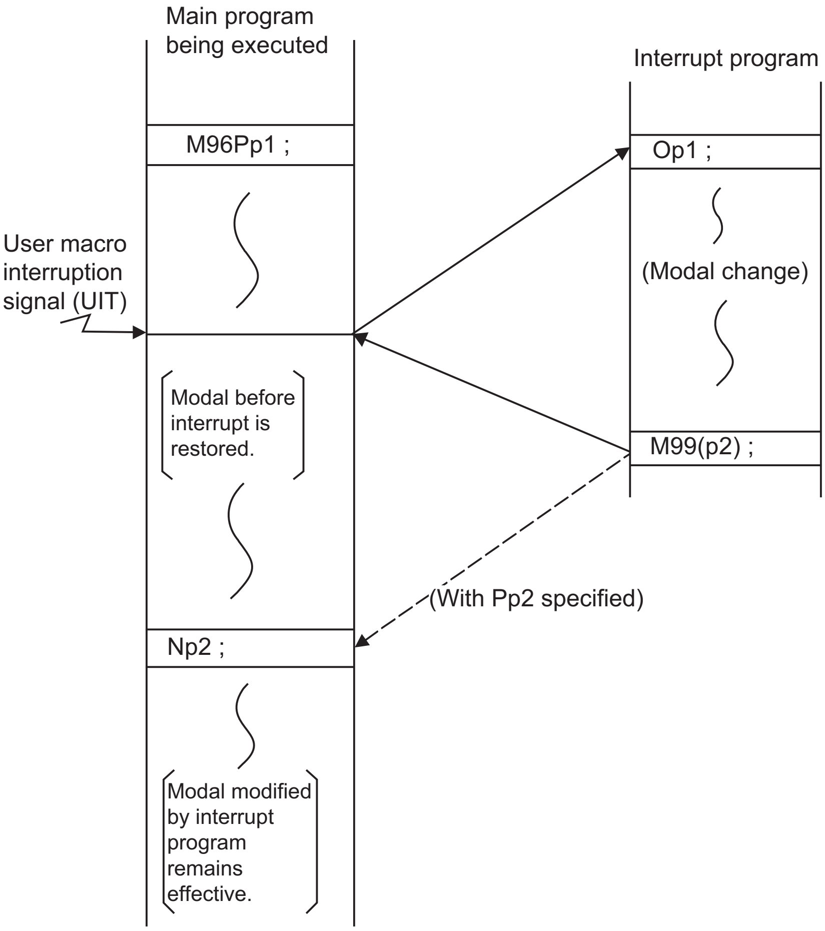
Returning from user macro interruption

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.


Main program being executed
Interrupt program
M96Pp1 ;
Op1 ;
User macro interruption signal (UIT)
(Modal change)
Modal before interrupt is restored.
M99(p2) ;
(With Pp2 specified)
Np2 ;
Modal modified by interrupt program remains effective.

Modal information affected by user macro interruption

Modal information variables (#4401 to #4520)

Modal information when control passes to the user macro interruption program can be known by reading system variables #4401 to #4520.
The unit specified with a command applies.

System variable	Modal information	
#4401 : #4421	G code (group01) : G code (group21)	Some groups are not used.
#4507	D code	
#4509	F code	
#4511	H code	
#4513	M code	
#4514	Sequence No.	
#4515	Program No. (Note 1)	
#4519	S code	
#4520	T code	

The above system variables are available only in the user macro interrupt program.
If they are used in other programs, program error (P241) will occur.

(Note 1) The programs are registered as files. When the program No. (file name) is read with #4515, the character string will be converted to a value.
(Example 1)
The file name "123" is the character string 0x31, 0x32, 0x33, so the value will be (0x31-0x30)*100 + (0x32-0x30)*10 + (0x33-0x30) = 123.0.
Note that if the file name contains characters other than numbers, it will be "blank".
(Example 2)
If the file name is "123ABC", it contains characters other than numbers, so the result will be "blank".

M code for control of user macro interruption

The user macro interruption is controlled by M96 and M97. However, these commands may have been used for other operation. To be prepared for such case, these command functions can be assigned to other M codes.
(This invalidates program compatibility.)

User macro interrupt control with alternate M codes is possible by setting the alternate M code in parameters "#1110 M96_M" and "#1111 M97_M" and by validating the setting by selecting parameter "#1109 subs_M".
(M codes 03 to 97 except 30 are available for this purpose.)

If the parameter "#1109 subs_M" used to enable the alternate M codes is not selected, the M96 and M97 codes remain effective for user macro interrupt control.
In either case, the M codes for user macro interrupt control are processed internally and not output to the outside.

Parameters

(1) Subprogram call validity "#8155 Sub-pro interrupt"
1: Subprogram type user macro interruption
0: Macro type user macro interruption

(2) Status trigger mode validity "#1112 S_TRG"
1: Status trigger mode
0: Edge trigger mode

(3) Interrupt type 2 validity "#1113 INT_2"
1: The executable statements in the interrupt program are executed after completion of execution of the current block. (Type 2)
0: The executable statements in the interrupt program are executed before completion of execution of the current block. (Type 1)

(4) Validity of alternate M code for user macro interruption control "#1109 subs_M"
1: Valid
0: Invalid

(5) Alternate M codes for user macro interruption
Interrupt enable M code (equivalent to M96) "#1110 M96_M"
Interrupt disable M code (equivalent to M97) "#1111 M97_M"

(6) Subprogram number selection "#8129 subprogram number selection"
Select a subprogram number to be called preferentially under subprogram control.
0: Commanded program number
1: 4-digits program number beginning with O No.
2: 8-digits program number beginning with O No.

Precautions

(1) If the user macro interruption program uses system variables #5001 and after (position information) to read coordinates, the coordinates pre-read in the buffer are used.

(2) If an interrupt is caused during execution of the nose R 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.

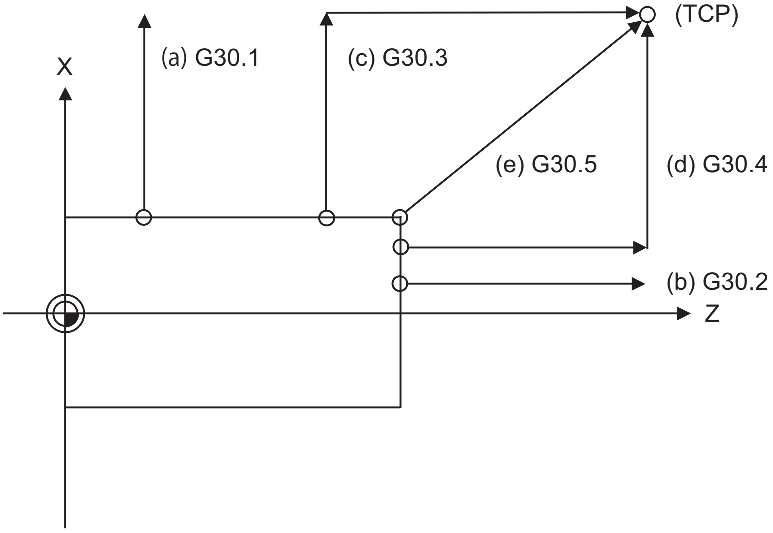
Command	Return order
G30.1	X axis only (-> additional axis)
G30.2	Z axis only (-> additional axis)
G30.3	X axis -> Z axis (-> additional axis)
G30.4	Z axis -> X axis (-> additional axis)
G30.5	X axis - Z axis (-> additional axis)

(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.

(2) If the axis address is commanded in the same block as the tool change position return command, a program error (P33) will occur.

Operation example



TCP : Tool change position

(1) 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.)

(2) 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.)

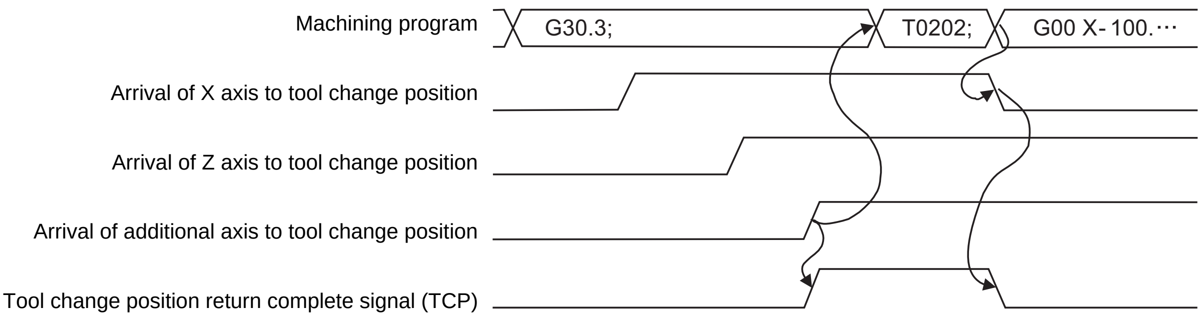
(3) G30.3 command : X axis returns to the tool change position, then Z axis does the same thing. (If the tool change position return is validated for the additional axis, the additional axis also returns to the tool change position after the X and Z axes reache the tool change position.)

(4) 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.)

(5) 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.)

(6) After all necessary tool change position return is completed by a G30.n command, tool change position return complete signal TCP (XC93) 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)


Machining program
G30.3;
T0202;
G00 X- 100.···
Arrival of X axis to tool change position
Arrival of Z axis to tool change position
Arrival of additional axis to tool change position
Tool change position return complete signal (TCP)

(7) 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.

(8) 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.

14

Coordinate System Setting Functions

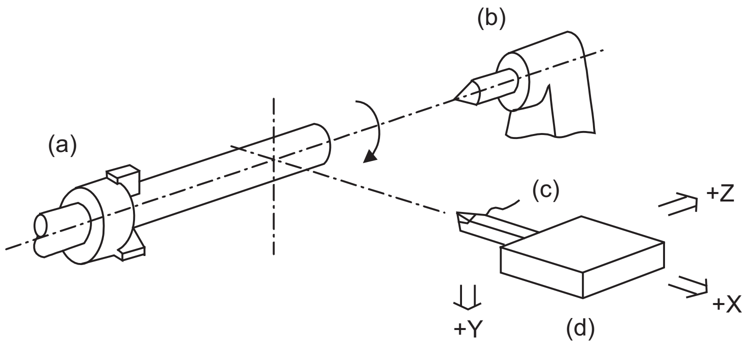
14.1 Coordinate Words and Control Axes

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

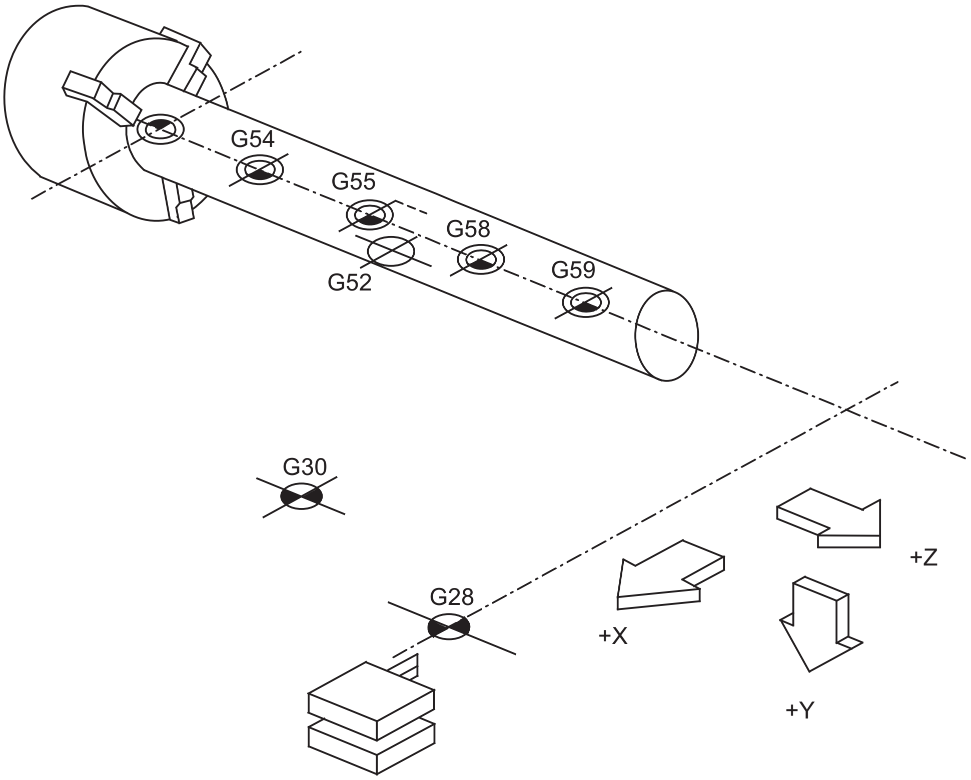
Coordinate axes and polarities



(a) Spindle stock
(b) Tailstock
(c) Tool
(d) Turret

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

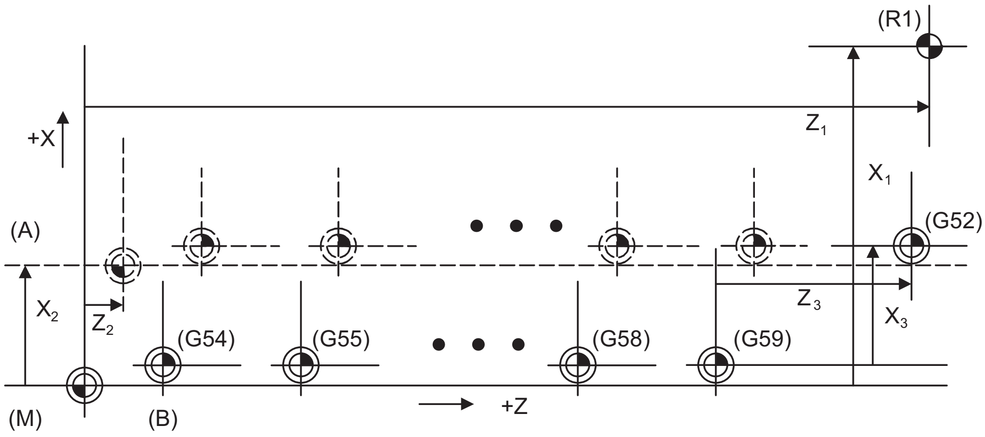
◎ Workpiece coordinate zero points

⊕ 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

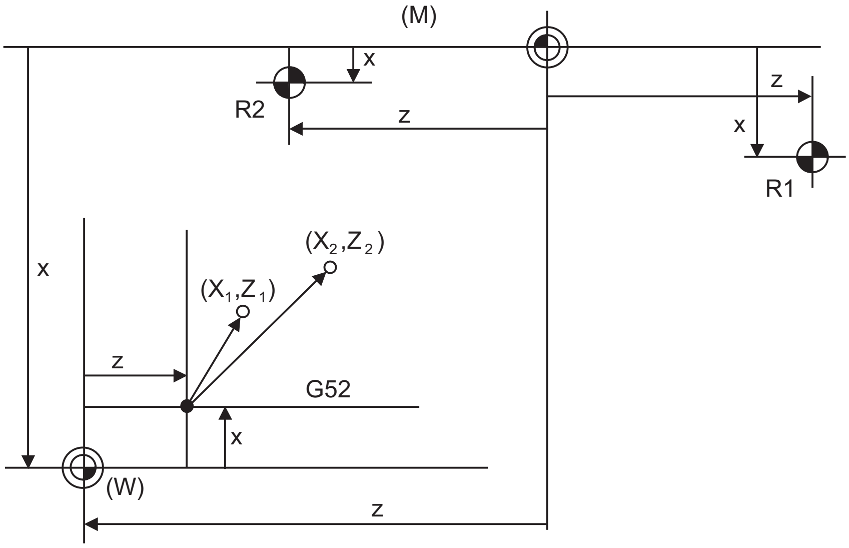
The local coordinate systems (G52) are valid on the coordinate systems designated by workpiece coordinate systems 1 to 6.
The hypothetical machine coordinate system can be set on the basic machine coordinate system using a G92 command. At this time, the workpiece coordinate system 1 to 6 is also simultaneously shifted.
Also refer to "Coordinate Systems and Coordinate Zero Point symbols".

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.

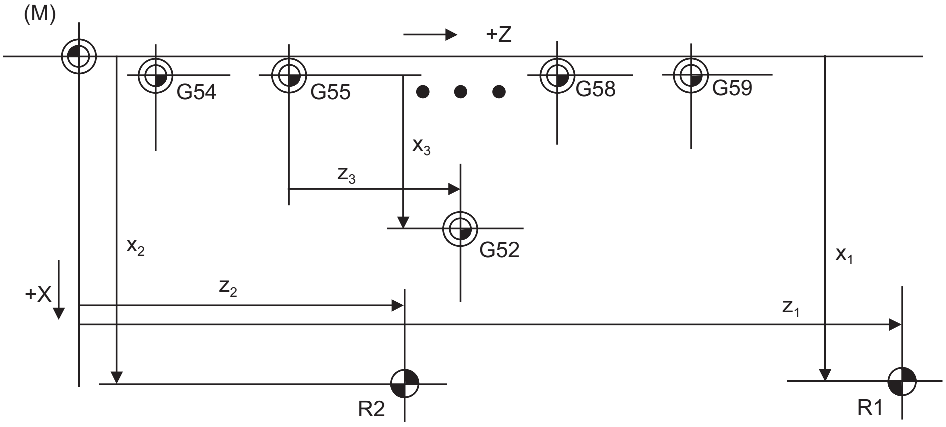


(M) Basic machine coordinate system
(G52) Local coordinate system
(R1) 1st reference position
(R2) 2nd reference position
(W) Workpiece coordinate systems (G54 to G59)

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 first manual reference position return or the reference position is reached with the dog-type reference position return when the NC power is turned ON.
The actual machining program is programmed over the coordinate systems which have been set above.



(M) Basic machine coordinate system
(R1) 1st reference position
(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)

(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__;

α	Additional axis
β	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.

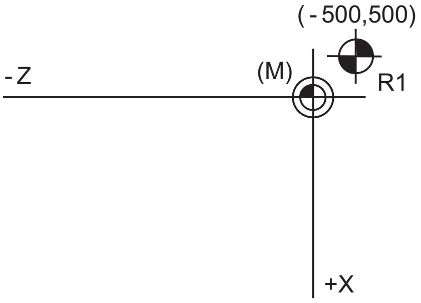
(2) The basic machine coordinate system is not changed by the G92 command.

(3) The G53 command is valid only in the designated block.

(4) When an incremental value is issued (U, W, β), 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.

(6) The G53 command will move with cutting feedrate or rapid traverse following command modal.



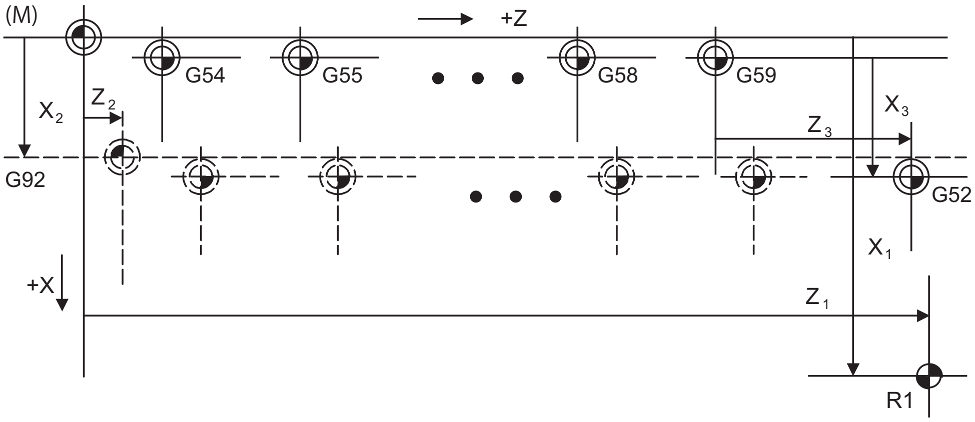
(M) Basic machine coordinate system
(R1) 1st reference position

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 α2;

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.)

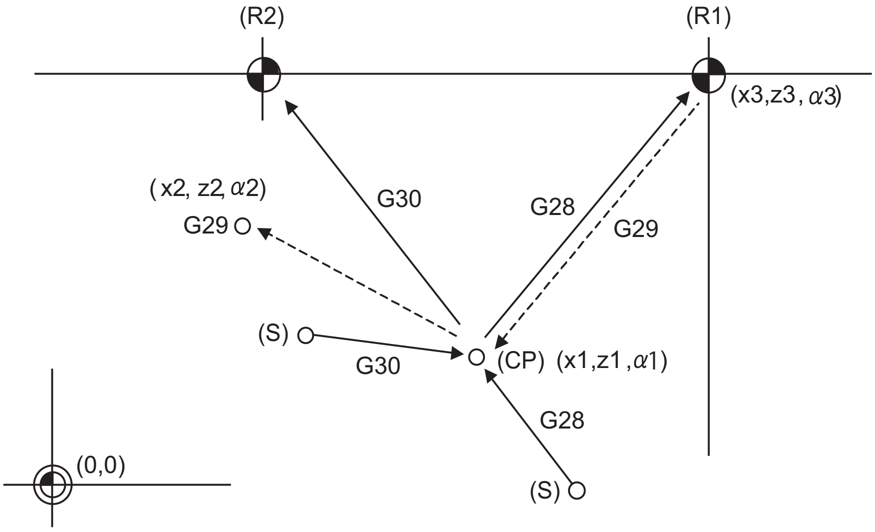
Precautions

(1) If the parameter "#1279 ext15/bit5" is set to "1", the coordinate systems setting (G92) shift amount is cleared when the axis reaches to the manual reference position.

14.7 Reference Position (Zero point) Return ; G28,G29

Function and purpose

After the commanded axes have been positioned by G0, they are returned respectively at rapid traverse to the first reference position when G28 is commanded.
By commanding G29, the axes are first positioned independently at high speed to the G28 or G30 intermediate point and then positioned by G0 at the commanded position.



(CP) Intermediate point
(R1) 1st reference position
(R2) 2nd reference position
(S) Start point

Command format

G28 Xx1 Zz1 α α1; ... Automatic reference position return

X, Z, α	Coordinate value of the intermediate point (α is an additional axis)

G29 Xx2 Zz2 α α2; ... Start point return

X, Z, α	Coordinate value of the end point (α is an additional axis)

Detailed description

(1) The G28 command is equivalent to the following:

G00 Xx1 Zz1 α α1 ;
G00 Xx3 Zz3 α α3 ;

In this case, x3, z3 and α3 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 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 α α1 ;
G00 Xx2 Zz2 α α2 ;

The rapid traverse (non-interpolation type) independent for each axis takes place.

In this case, x1, z1 and α1 are the coordinate value of the G28 or G30 intermediate point.

(5) Program error (P430) occurs when G29 is executed without executing automatic reference position (zero point) return (G28) after the power has been turned ON.

(6) The intermediate point coordinate values (x1, z1, α1) 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 Ignore middle point" 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) Do not issue T commands in a block where G29 exits. Program error (P29) occurs if the workpiece compensation amounts differ between the current and the previous blocks.

(12) When G28/G29/G30 is commanded in single block mode, if "#1279 ext15/bit6 Enable single block stop at middle point" is set to "1", single block stop at middle point will be performed; single block stop at middle point will not be performed if set to "0".

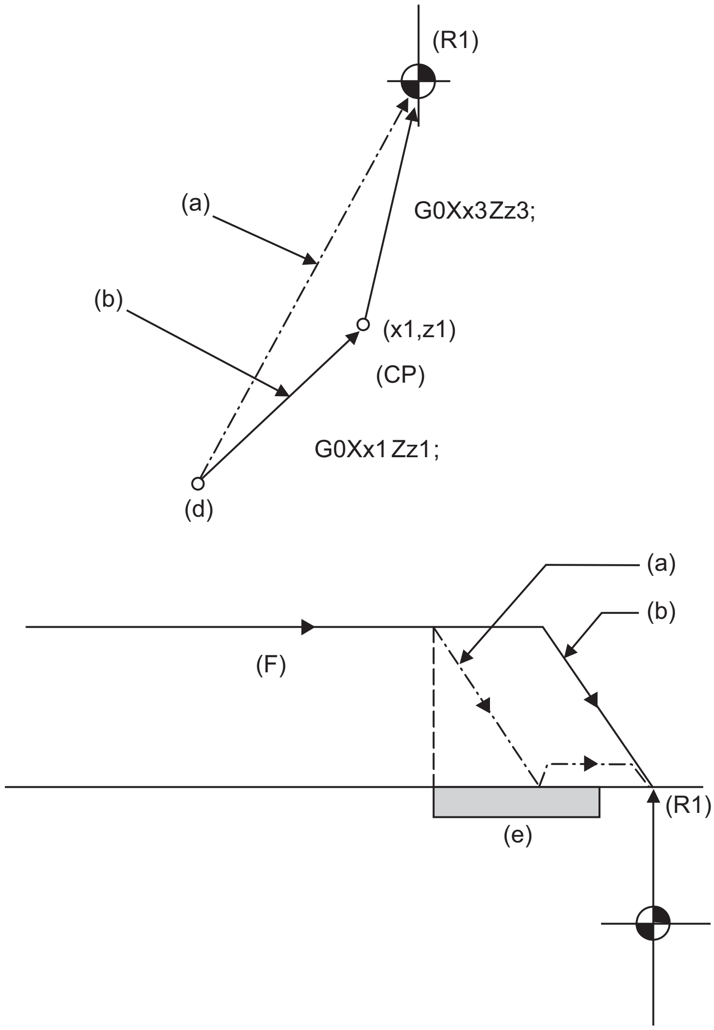
(13) If the mode is switched to MDI mode or reference point return mode while in a single block stop at middle point, "M01 Operation Error 0013" occurs.

(14) If the NC is reset while in a single block stop at middle point, the interim position for G29 start position return will not be updated.

(15) If a miscellaneous function is commanded in the same block as G28/G29, the miscellaneous function completion waiting point will be the end of commanded movement, instead of the interim position.

Program example

(Example 1) G28 Xx1 Zz1 ;



(a) 1st operation after power has been turned ON

(b) 2nd and subsequent operations

(d) Return start position

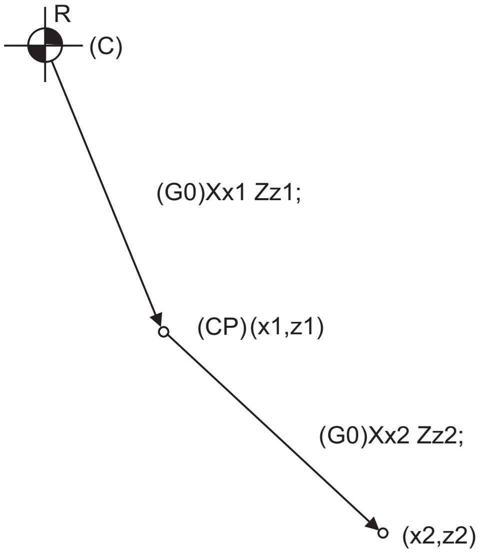
(e) Near-point dog

(F) Rapid traverse rate

(CP) Intermediate point

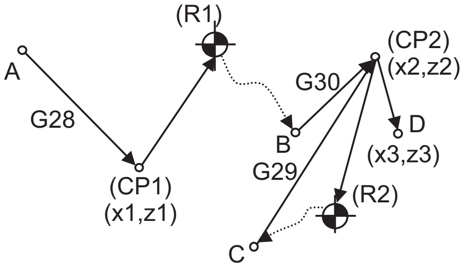
(R1) Reference position (#1)

(Example 2)G29 Xx2, Zz2 ;



(C) Current position (CP) G28, G30 Intermediate point

```
(Example 3)G28  Xx1  Zz1  ;
              :      (From point A to 1st reference position)
              :
             G30  Xx2  Zz2  ;
              :      (From point B to 2nd reference position)
              :
             G29  Xx3  Zz3  ;
                     (From point C to point D)
```



(CP1) Old intermediate point

(CP2) New intermediate point

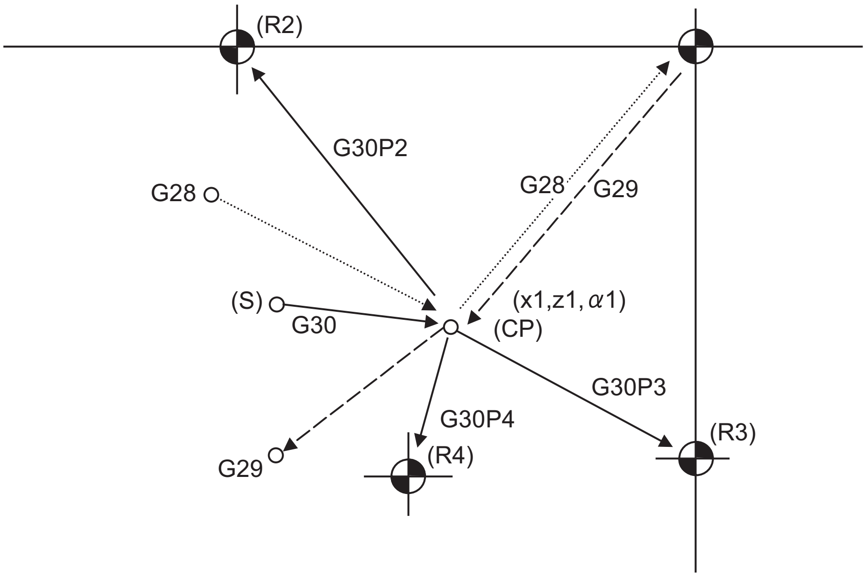
(R1) Reference position (#1)

(R2) 2nd reference position (#2)

14.8 2nd, 3rd, and 4th Reference Position (Zero point) Return ; G30

Function and purpose

The tool can return to the second, third, or fourth reference position by specifying G30 P2 (P3 or P4).



(S) Start point
(CP) Intermediate point
(R2) 2nd reference position
(R3) 3rd reference position
(R4) 4th reference position

Command format

G30 P2(P3,P4)Xx1 Zz1 α α1;

X, Z, α	Coordinate value of the intermediate point (α is an additional axis)
P	Reference position No. P2: 2nd reference position return P3: 3rd reference position return P4: 4th reference position return

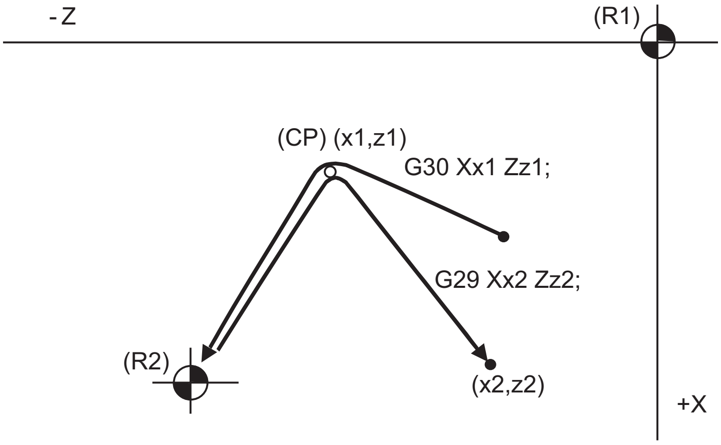
Detailed description

(1) The 2nd, 3rd, or 4th reference position return is specified by P2, P3, or P4.
A command without P or with other designation method will return the tool to the 2nd reference position.

(2) In the 2nd, 3rd, or 4th reference position return mode, as in the 1st reference position return mode, the tool returns to the 2nd, 3rd, or 4th reference position via the intermediate point specified by G30.

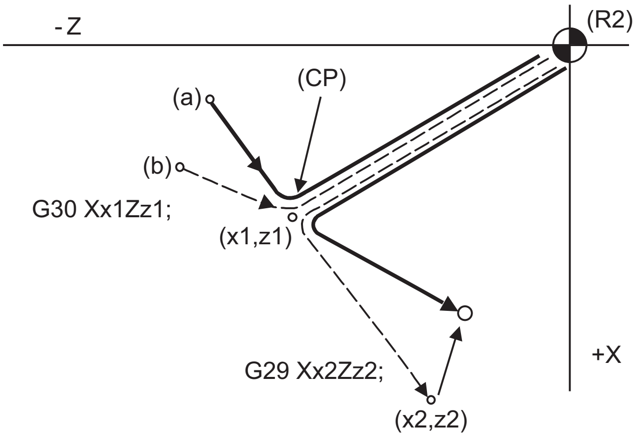
(3) The 2nd, 3rd, and 4th reference position coordinates refer to the positions specific to the machine, and these can be checked with the setting and display unit.

(4) If G29 is commanded after completion of returning to the 2nd, 3rd, and 4th reference position, the intermediate position used last is used as the intermediate position for returning by G29.



(CP) Intermediate point
(R1) 1st reference position (R2) 2nd reference position

(5) With reference position return on a plane during compensation, the tool moves without tool 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.



(CP) Intermediate point
(a) Tool nose center path (b) Program path
(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 position returns in the machine lock status, control from the intermediate point to the reference position will be ignored. When the designated axis reaches as far as the intermediate point, the next block will be executed.

(8) If the 2nd, 3rd or 4th reference position is changed while G30 zero point return operation is in pause due to an interlock, "M01 Operation Error" occurs.

(9) When G28/G29/G30 is commanded in single block mode, if "#1279 ext15/bit6 Enable single block stop at middle point" is set to "1", single block stop at middle point will be performed; single block stop at middle point will not be performed if set to "0".

(10) If the mode is switched to MDI mode or reference point return mode while in a single block stop at middle point, "M01 Operation Error 0013" occurs.

(11) If the NC is reset while in a single block stop at middle point, the interim position for G29 start position return will not be updated.

(12) If a miscellaneous function is commanded in the same block as G30, the miscellaneous function completion waiting point will be the end of commanded movement, instead of the interim position.

14.9 Reference Position Check ; G27

Function and purpose

This command first positions the tool at the position assigned by the program and then, if that positioning point is the 1st reference position, it outputs the reference position arrival signal to the machine in the same way as with the G28 command. Therefore, when a machining program is prepared so that the tool will depart from the 1st reference position and return to the 1st reference position, it is possible to check whether the tool has returned to the reference position after the program has been run.

Command format

G27 X__ Z__ α__ P__ ; ... Check command

X Z α	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 48 sets of added workpiece coordinate systems, as well as 6 workpiece coordinate systems, which are used by the programmer for programming (G54 to G59).

(3) Among the workpiece coordinate systems currently selected by these commands, any workpiece coordinate system with coordinates which have been commanded by the current position of the tool is reset. (The "current position of the tool" includes the compensation amounts for tool nose R and tool length 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 and tool length compensation.)　(G54,G92)

Command format

G54 to G59 ... Workpiece coordinate system selection

(G54 to G59) G92 X__ Z__ α__ ; ... Workpiece coordinate system setting

α	Additional axis

G54.1 Pn ; ... Workpiece coordinate system selection (P1 to P48)

G54.1 Pn ;
G92 X__ Z__ ; ... Workpiece coordinate system setting (P1 to P48)

G10 L20 Pn X__ Z__ ; ... Workpiece coordinate system offset amount setting (P1 to P48)

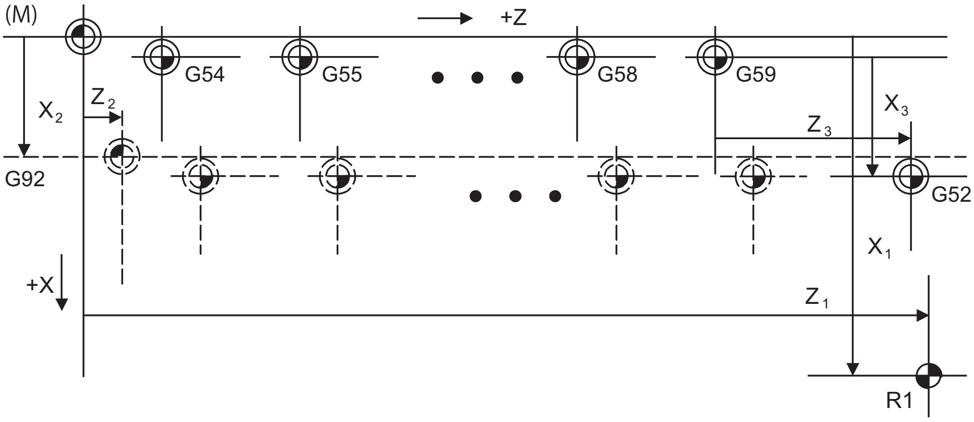
When the designated extended workpiece coordinate system offset amount is rewritten

G10 G54.1 Pn X__ Z__ ; ... Workpiece coordinate system offset amount setting (P1 to P48)

When the extended workpiece coordinate system is selected, and the offset amount is rewritten

Detailed description

(1) With any of the G54 through G59 commands, 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

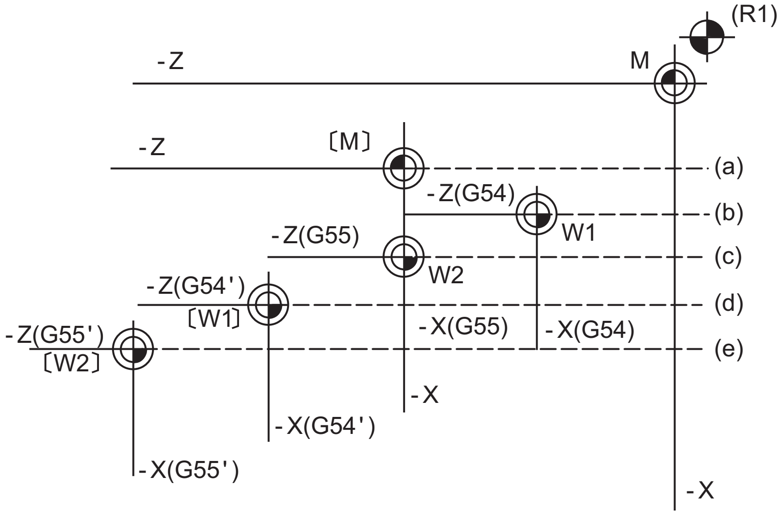
(6) The offset settings of workpiece coordinate systems can be changed any number of times. (They can also be changed by G10 L2 Pp1 Xx1 Zz1.)

[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. When in G54.1 modal, the program error (P33) will occur.
G10 L20 Pn Xx Zz ;	n=1 to 48 : Set the offset amount in the designated workpiece coordinate system. Others : The program error (P35) will occur.
G10 L20 Xx 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 ; G10 G54.1 Xx Yy Zz ;	L10 (tool offset) will be judged if there is no L value.

(7) A new workpiece coordinate system 1 is set by issuing the G92 command in the G54 (workpiece coordinate system 1) mode. At the same time, the other workpiece coordinate systems 2 to 6 (G55 to G59) will move in parallel and new workpiece coordinate systems 2 to 6 will be set.

(8) A hypothetical machine coordinate system is formed at the position which deviates from the new workpiece reference position (zero point) by an amount equivalent to the workpiece coordinate system offset amount.



After the power has been switched on, the hypothetical machine coordinate system is matched with the basic machine coordinate system by the first automatic (G28) or manual reference position (zero point) return.

(R1) Reference position 1
(a) Hypothetical machine coordinate system based on G92
(b) Old workpiece 1 (G54) coordinate system
(c) Old workpiece 2 (G55) coordinate system
(d) New workpiece 1 (G54) coordinate system
(e) New workpiece 2 (G55) coordinate system

(9) By setting the hypothetical machine coordinate system, the new workpiece coordinate system will be set at a position which deviates from that hypothetical machine coordinate system by an amount equivalent to the workpiece coordinate system offset amount.

(10) When the first automatic (G28) or manual reference position (zero point) return is completed after the power has been turned ON, the basic machine coordinate system and workpiece coordinate systems are set automatically in accordance with the parameter settings.

(11) If G54 X- ; is commanded after the reference position return (both automatic or manual) executed after the power is turned ON, the program error (P62) will occur. (A speed command is required as the movement will be controlled with the G01 speed.)

(12) Do not command a G code for which a P code is used in the same block as G54.1 or G10L20. If a G code is commanded, a P code is used for a prior G command or the program error occurs(P33).

(13) The local coordinate system cannot be used during G54.1 modal. The program error (P438) will occur when the G52 command is executed during G54.1 modal.

(14) A new workpiece coordinate system P1 can be set by commanding G92 in the G54.1 P1 mode. However, the workpiece coordinate system of the other workpiece coordinate systems G54 to G59, G54.1, and P2 to P48 will move in parallel with it, and a new workpiece coordinate system will be set.

(15) The offset amount of the extended workpiece coordinate system is assigned to the variable number as shown in Table 1.
[Variable Nos. of the extended workpiece coordinate offset system]

	1st axis to nth axis			**1st axis to nth axis**
P1	#7001 to #700n		P25	#7481 to #748n
P2	#7021 to #702n		P26	#7501 to #750n
P3	#7041 to #704n		P27	#7521 to #752n
P4	#7061 to #706n		P28	#7541 to #754n
P5	#7081 to #708n		P29	#7561 to #756n
P6	#7101 to #710n		P30	#7581 to #758n
P7	#7121 to #712n		P31	#7601 to #760n
P8	#7141 to #714n		P32	#7621 to #762n
P9	#7161 to #716n		P33	#7641 to #764n
P10	#7181 to #718n		P34	#7661 to #766n
P11	#7201 to #720n		P35	#7681 to #768n
P12	#7221 to #722n		P36	#7701 to #770n
P13	#7241 to #724n		P37	#7721 to #772n
P14	#7261 to #726n		P38	#7741 to #774n
P15	#7281 to #728n		P39	#7761 to #776n
P16	#7301 to #730n		P40	#7781 to #778n
P17	#7321 to #732n		P41	#7801 to #780n
P18	#7341 to #734n		P42	#7821 to #782n
P19	#7361 to #736n		P43	#7841 to #784n
P20	#7381 to #738n		P44	#7861 to #786n
P21	#7401 to #740n		P45	#7881 to #788n
P22	#7421 to #742n		P46	#7901 to #790n
P23	#7441 to #744n		P47	#7921 to #792n
P24	#7461 to #746n		P48	#7941 to #794n

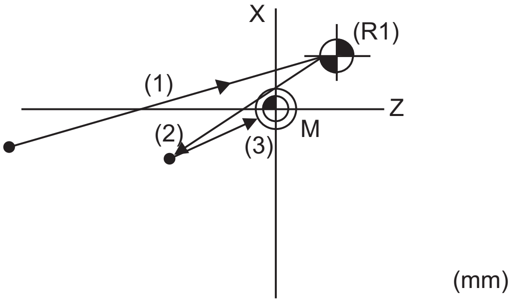
⚠ CAUTION

1. If the workpiece coordinate system offset amount is changed during single block stop, the new setting will be valid from the next block.

Program example

(Example 1)

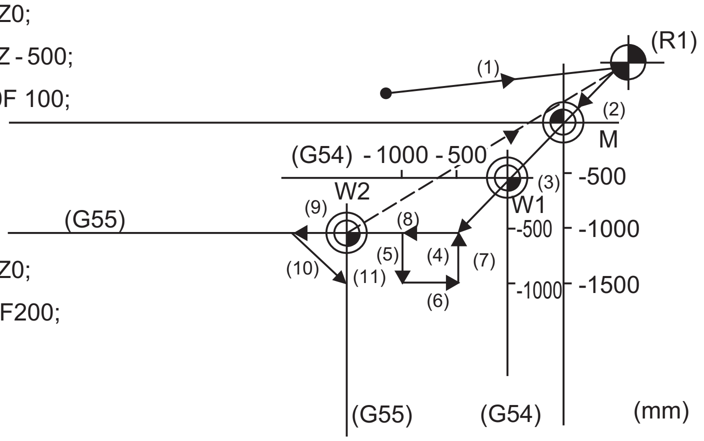
(1) G28 X0Z0;
(2) G53 X- 500 Z- 1000;
(3) G53 X0Z0;



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.

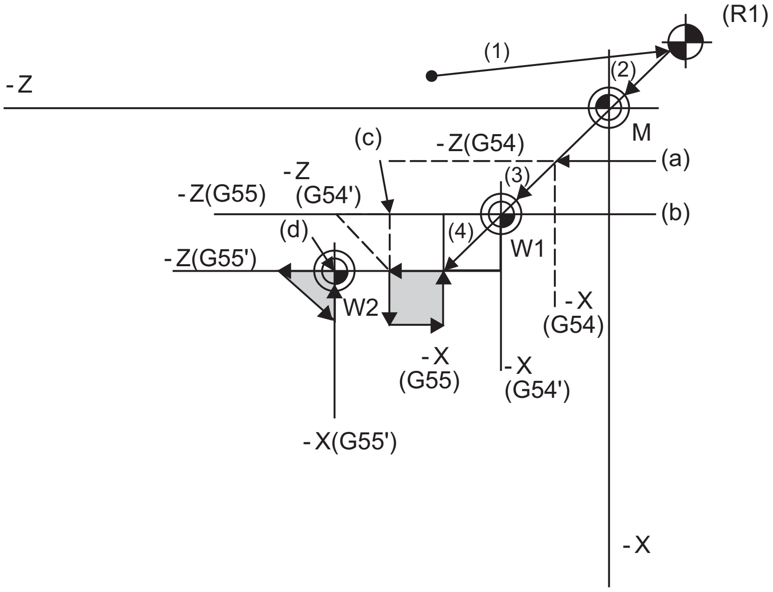
(Example 2)

(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__ ;
```

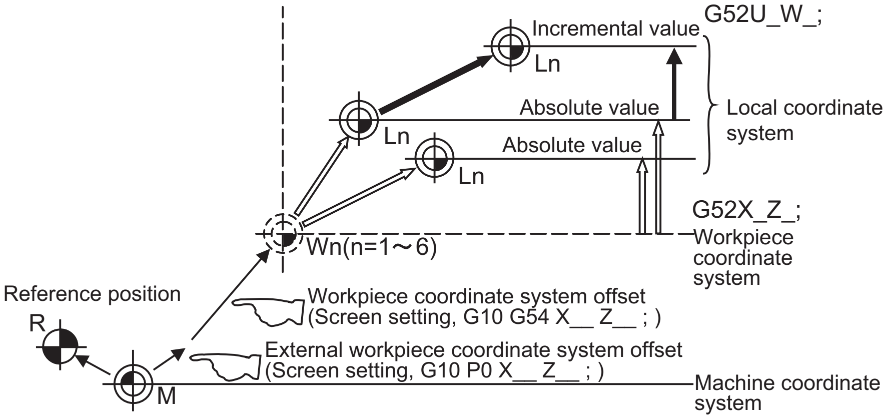
Detailed description

(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.


Incremental value
G52U_W_;
Ln
Absolute value
Local coordinate system
Ln
Absolute value
Ln
G52X_Z_;
Wn(n=1∼6)
Workpiece coordinate system
Reference position
R
Workpiece coordinate system offset
(Screen setting, G10 G54 X__ Z__ ;)
External workpiece coordinate system offset
(Screen setting, G10 P0 X__ Z__ ;)
M
Machine coordinate system

(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 Workpiece Coordinate System Preset ; G92.1

Function and purpose

This function presets the workpiece coordinate system shifted with the program command during manual operation to the workpiece coordinate system offset from the machine zero point by the workpiece coordinate offset amount by the program command (G92.1).
The workpiece coordinate system, which is set when the following type of operation or program command is executed, will be shifted from the machine coordinate system.

- When manual interrupt is executed while manual absolute is OFF
- When movement command is issued in machine lock state
- When axis is moved with handle interrupt
- When local coordinate system is set with G52
- Shifting the workpiece coordinate system with G92

This function presets the shifted workpiece coordinate system to the workpiece coordinate system offset from the machine zero point by the workpiece coordinate offset amount. This takes place in the same manner as manual reference position return. Whether to preset the relative coordinates or not can be selected with the parameters.

Command format

```
G92.1    X0 Z0 α0;
(G50.3)
```

α0	Additional axis

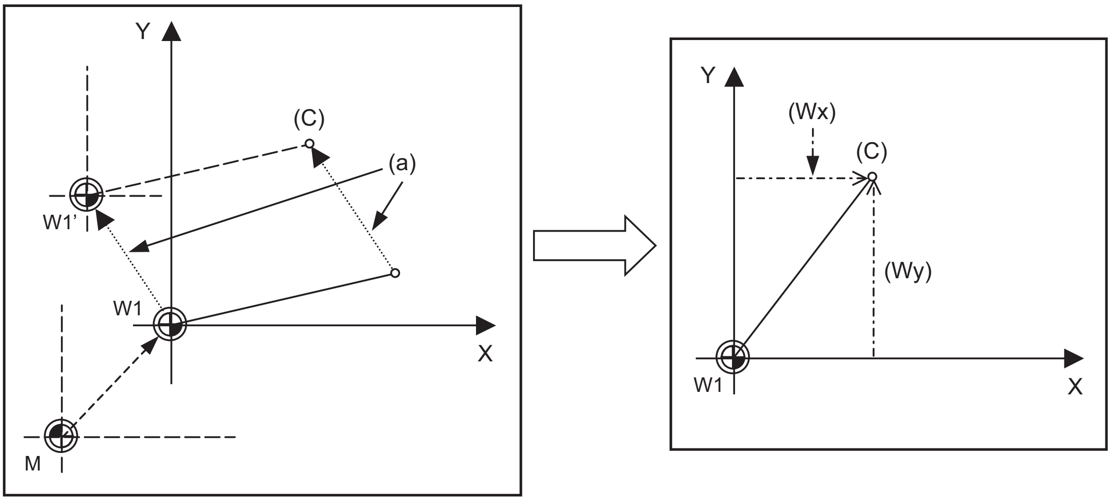
Detailed description

(1) Command the address of the axis to be preset. The axis will not be preset unless commanded.

(2) A program error (P35) will occur if a value other than "0" is commanded.

(3) Depending on the G code list, the G code will be "G50.3".

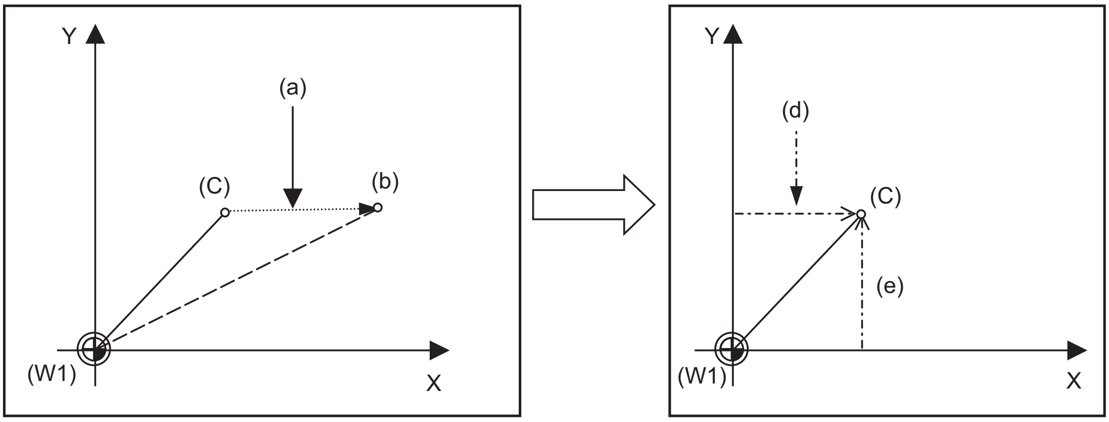
(4) When manual operation is carried out when manual absolute is set to OFF, or if the axis is moved with handle interrupt.



(a) Manual movement amount
(C) Current position
(Wx) Workpiece coordinate x after preset
(Wy) Workpiece coordinate y after preset

If manual operation is carried out when manual absolute is set to OFF, or if the axis is moved with handle interrupt, the workpiece coordinate system will be shifted by the manual movement amount.
This function returns the shifted workpiece coordinate zero point W1' to the original workpiece coordinate zero point W1, and sets the distance from W1 to the current position as the workpiece coordinate system's current position.

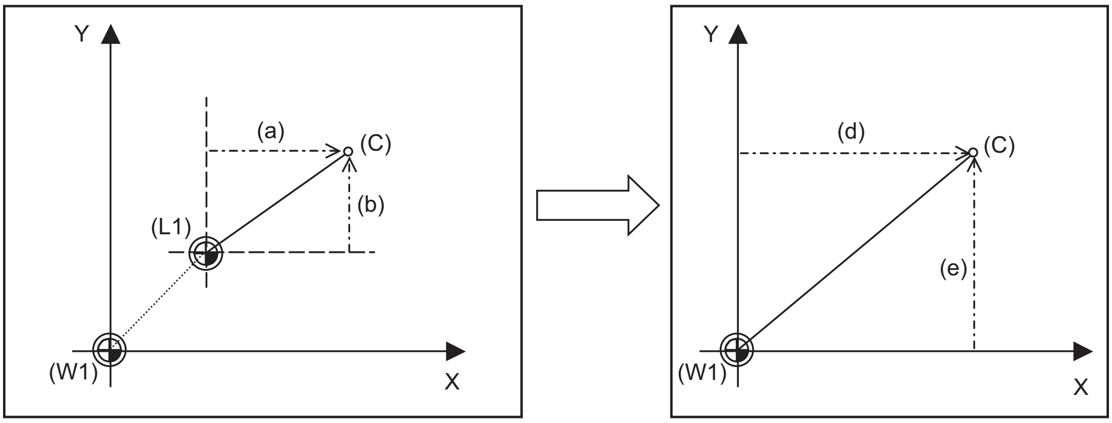
(5) When movement command is issued in machine lock state



(a) Movement amount during machine lock
(b) Workpiece coordinate system coordinate value
(C) Current position
(d) Workpiece coordinate x after preset
(e) Workpiece coordinate y after preset
(W1) Workpiece coordinate zero point

If the movement command is issued in the machine lock state, the current position will not move, and only the workpiece coordinates will move.
This function returns the moved workpiece coordinates to the original current position, and sets the distance from W1 to the current position as the workpiece coordinate system's current position.

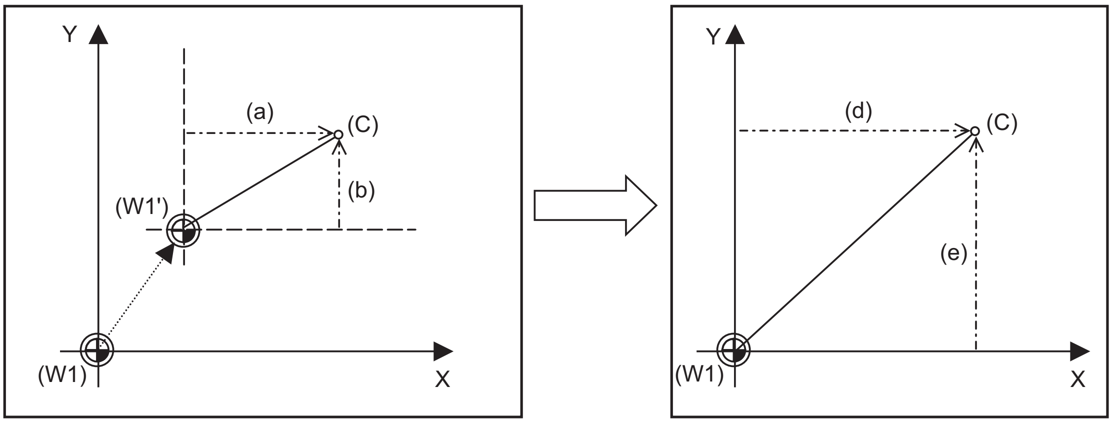
(6) Setting local coordinate system with G52



(a) Local coordinates x
(b) Local coordinates y
(C) Current position
(d) Workpiece coordinate x after preset
(e) Workpiece coordinate y after preset
(L1) Local coordinate zero point
(W1) Workpiece coordinate zero point

The local coordinate system is set with the G52 command, and the program commands, etc., are issued with the local coordinate system.
With this function, the set local coordinate system is canceled, and the program commands, etc., use the workpiece coordinate system which has W1 as the zero point. The canceled local coordinate system is only the selected workpiece coordinate system.

(7) Shifting the workpiece coordinate system with G92

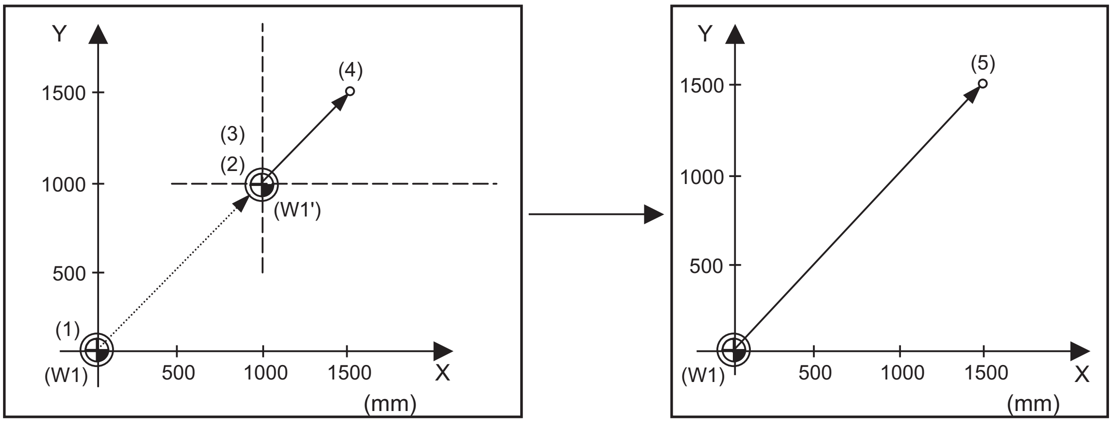


(a) Local coordinates x (b) Local coordinates y
(C) Current position
(d) Workpiece coordinate x after preset
(e) Workpiece coordinate y after preset
(W1) Workpiece coordinate zero point
(W1') Workpiece zero point after G92 command

The workpiece coordinate system shifts with the G92 command, and the distance between W1' and the current position is set as the current position of the workpiece coordinate system.
This function returns the shifted workpiece coordinate zero point to W1, and sets the distance from W1 to the current position as the workpiece coordinate system's present position. This is valid for all workpiece coordinate systems.

Program example

The workpiece coordinate system shifted with G92 is preset with G92.1.



(W1) Workpiece coordinate zero point
(W1') Workpiece zero point after G92 command

(Example)

```
G28 X0 Y0 ;          ... (1)
G00 G90 X1. Y1. ;    ... (2)
G92 X0 Y0            ... (3)
G00 X500 Y500 ;      ... (4)
G92.1 X0 Y0 ;        ... (5)
```

Precautions

(1) Cancel nose R compensation and tool length compensation before using this function. If this function is executed without canceling them, the workpiece coordinates will be at a position obtained by subtracting the workpiece coordinate offset amount from the machine value. Thus, the compensation vector will be temporarily canceled.

(2) This function cannot be executed while the program is being resumed.

14.13 Coordinate System for Rotary Axis

Function and purpose

The axis designated as the rotary axis with the parameters is controlled with the rotary axis' coordinate system.
The rotary axis includes the rotating type (short-cut valid/invalid) and linear type (workpiece coordinate position linear type and all coordinate position linear type).
The workpiece coordinate position range is 0 to 359.999° for the rotating type, and 0 to ± 99999.999° for the linear type.
The machine coordinate value and relative position differ according to the parameters.
The rotary axis is commanded with a degree (°) unit regardless of the inch or metric designation.
The rotary axis type can be set with the parameter "#8213 rotation axis type" for each axis.

	Rotary axis				**Linear axis**
	Rotating type rotary axis		**Linear type rotary axis**		
	Short-cut invalid	**Short-cut valid**	**Workpiece coordinate position linear type**	**All-coordinate position linear type**	
#8213 setting value	**0**	**1**	**2**	**3**	**-**
Workpiece coordinate position	Displayed in the range of 0° to 359.999° .		Displayed in the range of 0° to ± 99999.999° .		
Machine coordinate position/relative position	Displayed in the range of 0° to 359.999° .			Displayed in the range of 0° to ± 99999.999° .	
ABS command	The incremental amount from the end point to the current position is divided by 360 degrees, and the axis moves by the remainder amount according to the sign.	Moves with a short-cut to the end point.	In the same manner as the normal linear axis, it moves according to the sign by the amount obtained by subtracting the current position from the end point (without rounding up to 360 degrees).		
INC command	Moves in the direction of the commanded sign by the commanded incremental amount starting at the current position.				
Reference position return	Depends on the absolute command or the incremental command during the movement to the intermediate point.				
	Returns with movement within 360 degrees.			Moves and returns in the R point direction for the difference from the current position to the R point.	

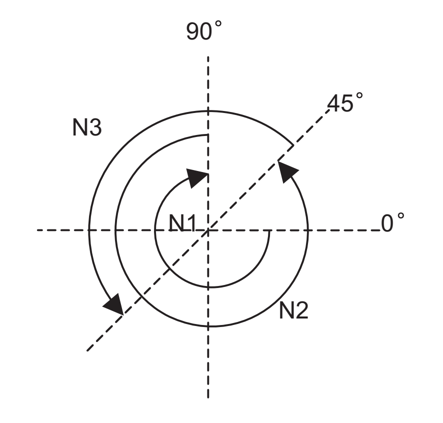
Operation example

Examples of differences in the operation and counter displays according to the type of rotation coordinate are given below.
(The workpiece offset is set as 0°.)

Rotary type (short-cut invalid)

(1) The machine coordinate position, workpiece coordinate position and relative position are displayed in the range of 0° to 359.999°.

(2) For the absolute position command, the axis moves according to the sign by the remainder amount obtained by dividing by 360°.

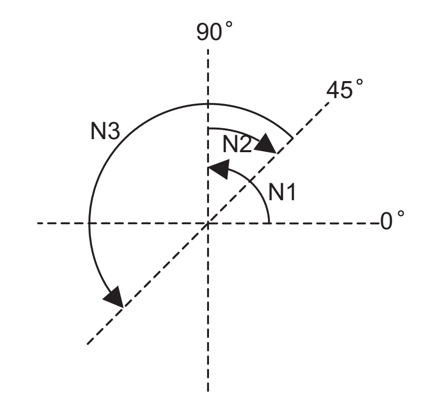


Program	Workpiece	Machine
G28 C0.		
N1 G90 C-270.	90.000	90.000
N2 C405.	45.000	45.000
N3 G91 C180.	225.000	225.000

Rotary type (short-cut valid)

(1) The machine coordinate position, workpiece coordinate position and relative position are displayed in the range of 0° to 359.999°.

(2) For the absolute position command, the axis rotates to the direction having less amount of movement to the end point.



Program	Workpiece	Machine
G28 C0.		
N1 G90 C-270.	90.000	90.000
N2 C405.	45.000	45.000
N3 G91 C180.	225.000	225.000

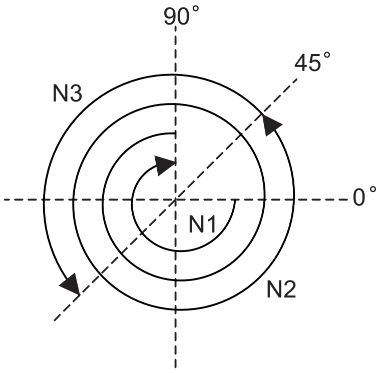
Linear type (workpiece coordinate position linear type)

(1) The coordinate position counter other than the workpiece coordinate position is displayed in the range of 0° to 359.999°.
The workpiece coordinate position is displayed in the range of 0 to ±99999.999°.

(2) The movement is the same as the linear axis.

(3) During reference position return, the axis moves in the same manner as the linear axis until the intermediate point is reached. The axis returns with a rotation within 360° from the intermediate point to the reference position.

(4) During absolute position detection, even if the workpiece coordinate position is not within the range of 0 to 359.999°, the system will start up in the range of 0 to 359.999° when the power is turned ON again.



Program	Workpiece	Machine	Relative position
G28 C0.			
N1 G90 C-270.	-270.000	90.000	90.000
N2 C405.	405.000	45.000	45.000
N3 G91 C180.	585.000	225.000	225.000
	After the power is turned ON again		
	Workpiece	Machine	
	225.000	225.000	

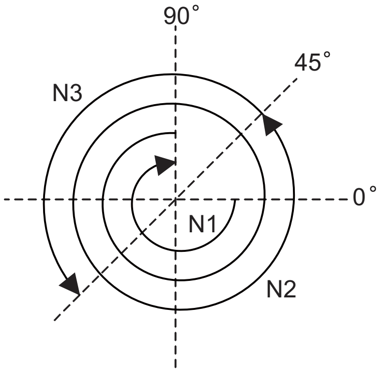
Linear type (all coordinate position linear type)

(1) The workpiece coordinate position counter is displayed in the range of 0 to ±99999.999°.

(2) The movement is the same as the linear axis.

(3) During reference position return, the axis moves in the same manner as the linear axis until the intermediate point is reached.
The axis rotates by the difference from the intermediate point to the reference position and returns to the reference position.

(4) During absolute position detection, the system starts up at the position where the power was turned OFF when the power is turned ON again.



Program	Workpiece	Machine	Relative position
G28 C0.			
N1 G90 C-270.	-270.000	-270.000	-270.000
N2 C405.	405.000	405.000	405.000
N3 G91 C180.	585.000	585.000	585.000
	After the power is turned ON again		
	Workpiece	Machine	
	585.000	585.000	

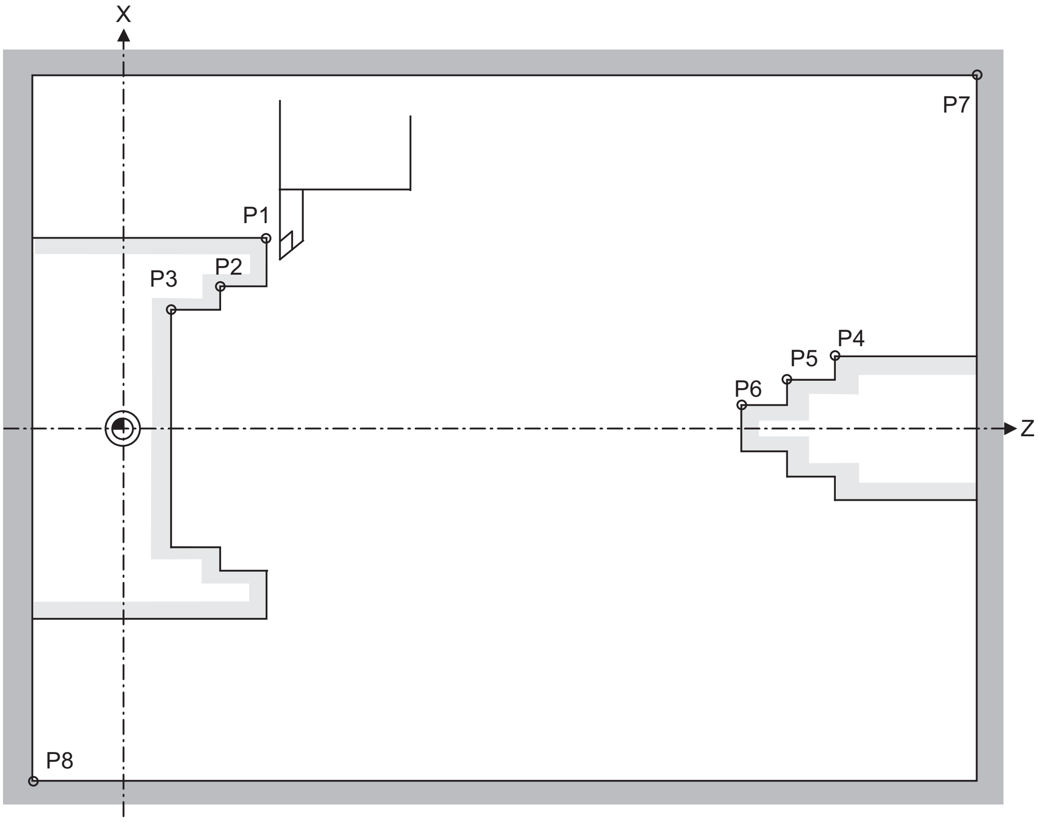
15

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.

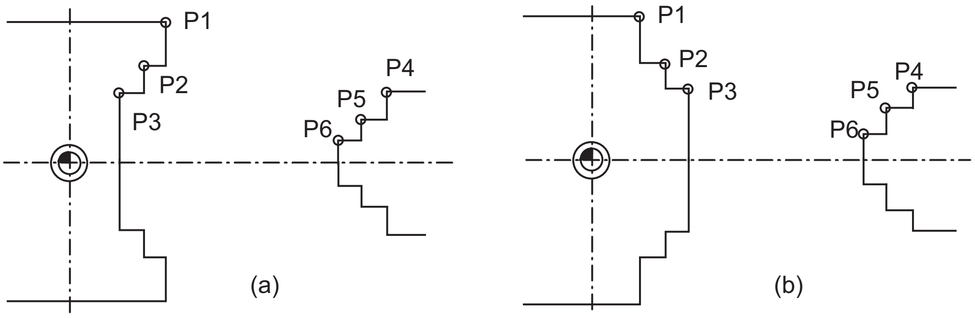
(2) This function is also valid during machine lock.

(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.

(5) A detailed barrier can be set on the left and right by using the PLC external signal.

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_ 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 >= P2 >= P3,   P4 >= P5 >= P6
(Note that the Z axis coordinates do not have to follow this setting.)

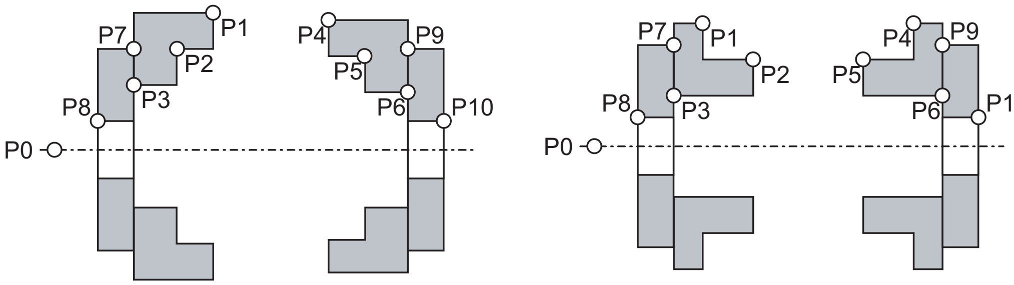
Setting when using the PLC external signal input

Set the coordinate values for each point between "#8300 P0" to "#8314 P10".
P0 is the basic X coordinate of the chuck and tailstock barrier. Set the workpiece center coordinate in the basic machine coordinate system as a radius value.
The coordinate value from the workpiece center (P0) is set as a radius value for the P1 to P10 X axes. The Z axis is set as a basic machine coordinate system coordinate.
The barrier region is symmetrical in respect to P0.
The X axis coordinate of each point must be set in the following manner.
P1 >= P2 >= P3, P4 >= P5 >= P6,
P7 >= P8, P9 >= P10
If the P8 Z axis coordinate must be less than P1 to P3, and the P10 Z axis coordinate must be more than P4 to P6.

The positions of the each point between P0 and P10 in the chuck/tailstock are as follow.

(1) Setting the chuck

[For external claw] **[For internal claw]**

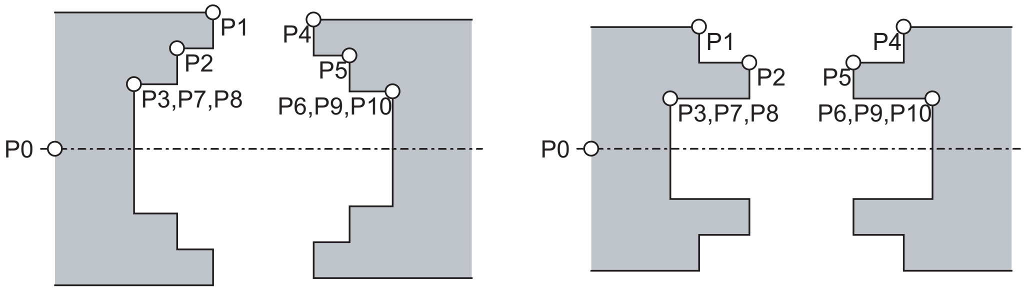


P0 : Workpiece rotation center coordinate
P1,P2,P3 : Chuck section (left) P4,P5,P6 : Chuck section (right)
P7,P8 : Spindle section (left) P9,P10 : Spindle section (right)

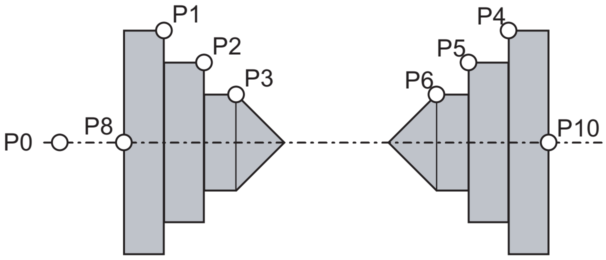
- Set the X axis coordinates so that P1 > P2 > P3 is established.
- The P3 and P7 in Z coordinates and the P6 and P9 in Z coordinates shall be equal.
- If there is no spindle inner diameter, set the P8 X coordinate to the same value as P0.

If a spindle section is not to be set, set the P3, P7 and P8 values to the same values, and the P6, P9 and P10 values to the same values. The barrier range will be as shown below in this case.

[For external claw] **[For internal claw]**



(2) Setting the tailstock

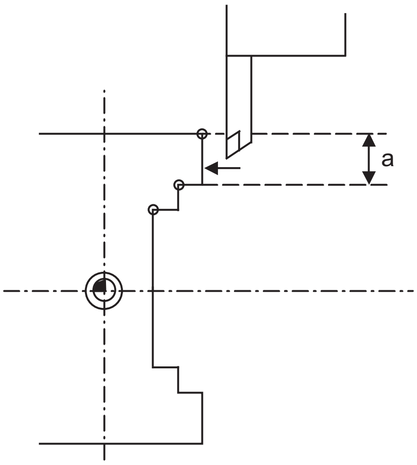


- P7 and P9 are not used.
- The tailstock end section angle is set with the parameters.
  "#8318 STOCK ANGLE (L)"
  "#8319 STOCK ANGLE (R)"
- If "0" is set, the angle will be 90°. (Default)
- Set the P8 X coordinate to the same value as P0. (No spindle inner diameter)

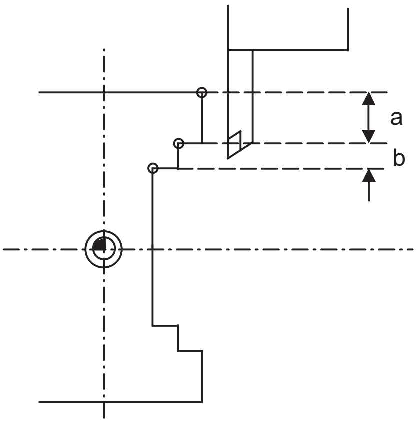
Precautions

(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".

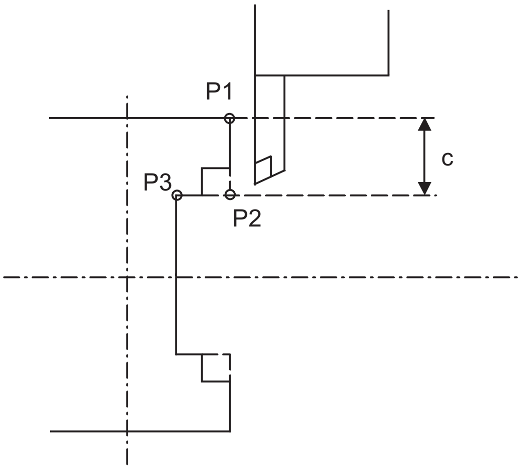
(Example 1)



(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).

15.2 Stored Stroke Limit ; G22,G23

Function and purpose

The stored stroke limit is a function that can prevent collision of the tools by setting the tool entry prohibited ranges.
The stored stroke limit II function can change the tool entry prohibited ranges (parameter) and command its function's ON/OFF by the program.

Command format

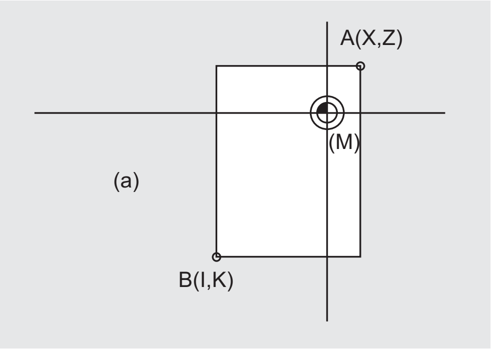
G22 X__ Z__ C__ I__ J__ K__ ; ... Stored stroke limit II function parameter change, function ON (all axes)

G23; ... Stored stroke limit II function OFF (all axes)

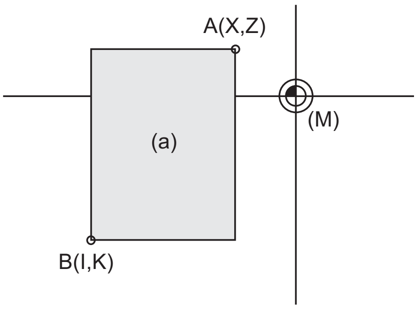
X, Z, C (1st to 3rd axis name)	Stored stroke limit + side designation
I, J, K (Plane selection axis name)	Stored stroke limit - side designation

Detailed description

Command G22 and G23 in independent blocks.
Whether to set the prohibited range on the inside or outside follows the parameter settings (#8204 or #8205).



[When prohibited range is on outside]



[When prohibited range is on inside]

(a) Prohibited range
(M) Machine coordinate system

Precautions

(1) If the maximum value and minimum value of the stored stroke limit's prohibited range are set to the same value, the following will occur.
(a) When the maximum value and minimum value are set to 0, if the outside is the prohibited range, the entire range will be prohibited. If the inside is the prohibited range, the entire range will be the moveable range.
(b) If data other than 0 is set for the maximum value and minimum value, the entire range will be the moveable range.

(2) The settings (G22, G23) with the stored stroke limit II program command are valid at the G code list 6 and 7 (cmdtyp = 7, 8).

(3) G22 and G23 are unmodal, and are valid only in the commanded block.

16

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.

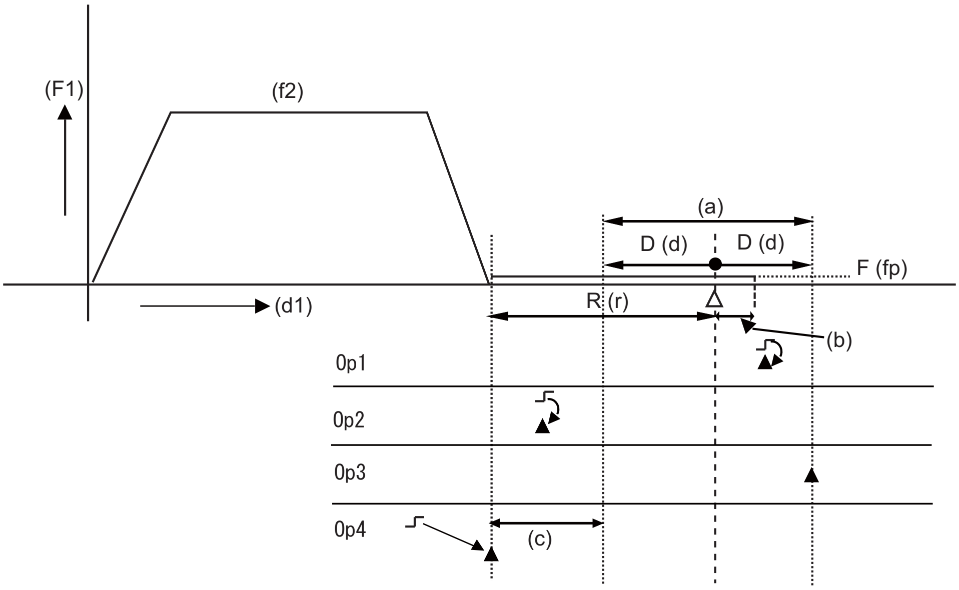
Command format

G37 α__ R__ D__ F__ ; ... Automatic tool length measurement command

α	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) Feedrate
(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.
Maximum measuring error [mm] = Measuring speed [mm/min] * 1/60 * 0.2 [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.
Maximum overtravel [mm] = Measuring speed [mm/min] * 1/60 * 1/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) 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.

(6) After making a manual interrupt while in movement at the measurement speed, be sure to return to the pre-interruption position first and then restart.

(7) The data commanded in G37 or the parameter setting data must meet the following conditions:
| Measurement point start point | > R command or parameter r > D command or parameter d

(8) When the D address and parameter d in (7) above are zero, the operation will be completed normally only when the commanded measurement point and sensor signal detection point coincide. Otherwise, program error (P607) will occur.

(9) When the R and D addresses as well as parameters r and d in (7) above are all zero, program error (P607) will occur regardless of whether the sensor signal is present or not after the tool has been positioned at the commanded measurement point.

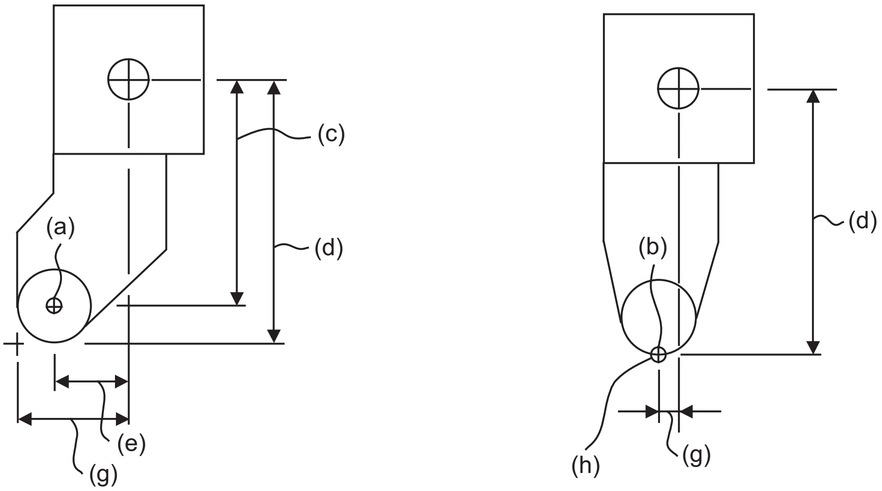
(10) When the measurement allowable range is larger than the measurement command distance, it becomes the measurement allowable range for all axes.

(11) When the measurement speed movement distance is larger than the measurement command distance, all axes move at the measurement speed.

(12) When the measurement allowable range is larger than the measurement speed movement distance, the axis moves in the measurement allowable range at the measurement speed.

(13) Always cancel nose R compensation before commanding G37.

(14) Calculate the tool length offset amount without regard for the nose R value and tool nose point No.
To set the tool nose point No. to 0, subtract the nose R value from the measured tool length offset amount.
When the tool nose point No. (tool nose shape) is 5, 6, 7, or 8, measure the tool length at the tool tip.



(a) Tool nose point 0
(b) Tool nose point 8
(c) X axis tool length offset value with nose R value subtracted
(d) Measured X axis tool length offset amount
(e) Z axis tool length offset value with nose R value subtracted
(g) Measured Z axis tool length offset amount
(h) Tip of tool nose

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__ R__ F__ ;

X, Z, U, W	Axis coordinate value; they are commanded with the absolute or the incremental values.
R	Acceleration/deceleration command R0: Acceleration/deceleration time constant=0 (No automatic acceleration/deceleration after interpolation.) R1: Acceleration/deceleration time constant valid. Accelerate/decelerate with the time constant set with the parameters "#2102 skip_tL" and "#2103 skip_t1". R0 is applied when it is omitted.
F	Feedrate (mm/min)

Detailed description

(1) If Ff is commanded as the feedrate in the same block as the G31 command block, command feed f will apply; if not 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.

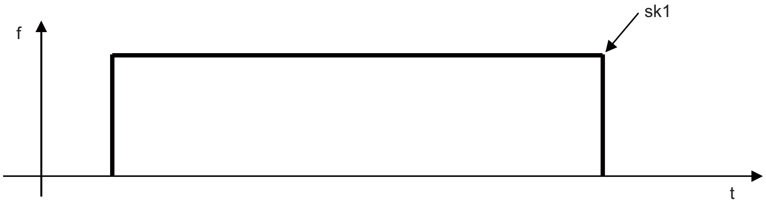
(2) The maximum speed of G31 command is determined by the machine specification.

(3) When R0 is commanded or the R command is omitted, the step acceleration/deceleration will be applied to G31 block after the interpolation without performing the automatic acceleration/deceleration.

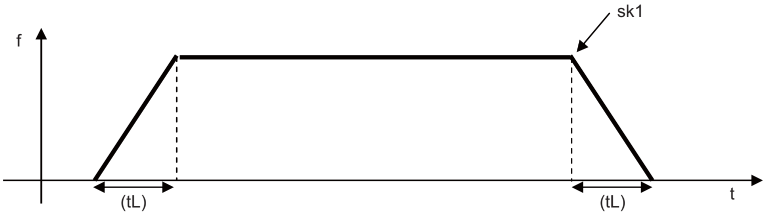
When R1 is commanded, the automatic acceleration/deceleration will be performed according to the cutting feed acceleration/deceleration mode set by the parameter "#2003 smgst" with the time constant set by the parameter "#2102 skip_tL" and "#2103 skip_t1".

Even if G1 constant inclination acceleration/deceleration (the parameter "#1201 G1_acc" is set to "1") is valid, the time constant acceleration and deceleration will be performed.

(4) When the R1 is commanded with the acceleration and deceleration command, the automatic acceleration and deceleration will be performed after the interpolation even if the skip single is input. Note that if the value of the parameter "#2102 skip_tL" and #2103 skip_t1" are large, it will not stop immediately.
Acceleration/deceleration when R0 is commanded or R is omitted



Acceleration/deceleration when R1 is commanded



(sk1) Skip signal (tL) Skip time constant

(5) Command the acceleration/deceleration command (R0/R01) per G31 command. If not, "the acceleration/deceleration time constant=0 (R0)" will be applied and no automatic acceleration/deceleration will be performed after the interpolation.
(6) Override is invalid with the G31 command and it is fixed at 100%. Dry run is also invalid. The stop conditions (feed hold, interlock, override zero and stroke end) are valid. External deceleration is also valid.
(7) The G31 command is unmodal and it needs to be commanded each time.
(8) If the skip command is input at the start of the G31 command, the G31 command will be completed immediately.
When a skip signal has not been input until the completion of the G31 block, the G31 command will also be completed upon completion of the movement commands.
(9) When the G31 command is issued during tool nose radius compensation, the program error (P608) will occur.
(10) When there is no F command in the G31 command and the parameter speed is also zero, the program error (P603) will occur.
(11) With machine lock or with the Z axis cancel switch ON when only the Z axis is commanded, the skip signal will be ignored and execution will continue as far as the end of the block.

Readout of skip coordinates

The coordinate positions for which the skip signal is input are stored in the system variables #5061 (1st axis) to #506n (n-th axis), so these can be used in the user macros.

```
:
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.

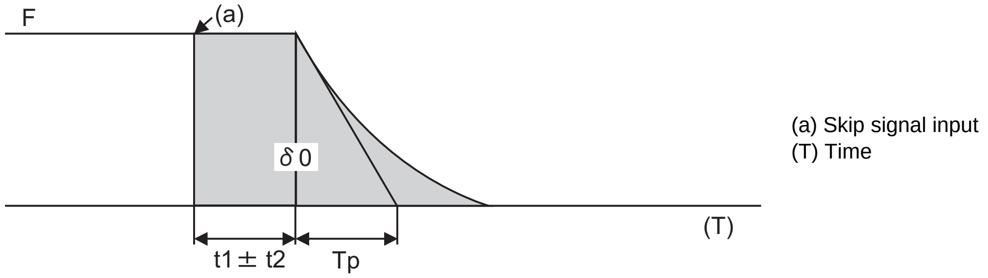
$$\delta 0 = \frac{F}{60} \times Tp + \frac{F}{60} \times (t1 \pm t2)$$

$$= \underbrace{\frac{F}{60} \times (Tp+t1)}_{\delta 1} \pm \underbrace{\frac{F}{60} \times t2}_{\delta 2}$$

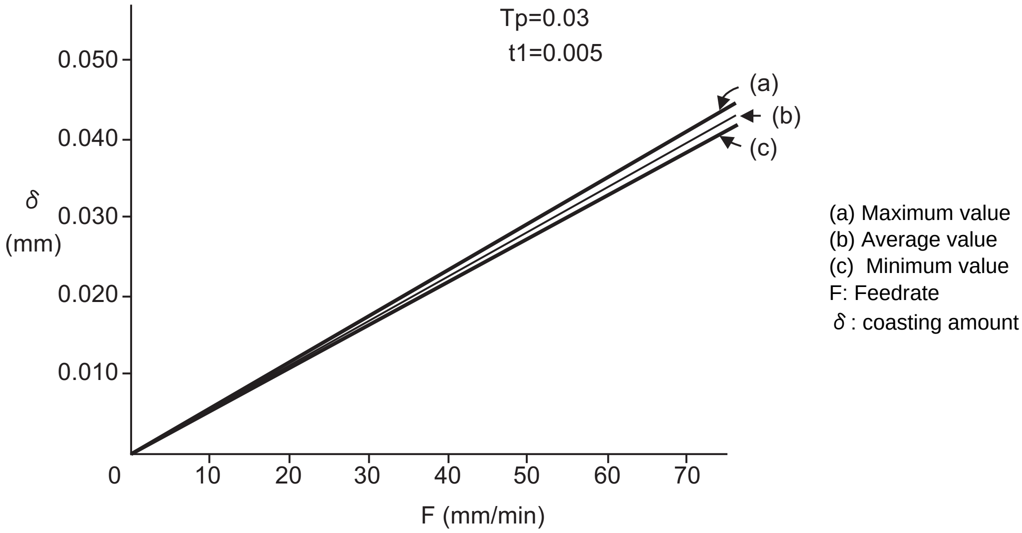
$\delta 0$	: Coasting amount (mm)
F	: G31 skip speed (mm/min)
Tp	: Position loop time constant (s) = (position loop gain)$^{-1}$
t1	: Response delay time (s) = (time taken from the detection to the arrival of the skip signal at the controller via PC)
t2	: Response error time 0.001 (s)

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 Tp is 30ms and t1 is 5ms is shown in the following figure.

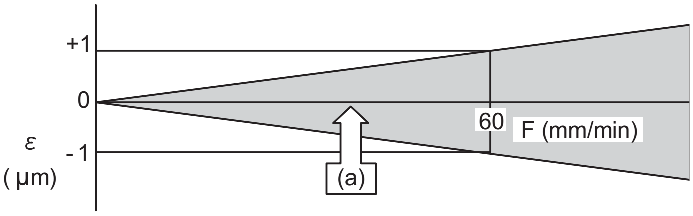


Readout error of skip coordinates mm

(1) Skip signal input coordinate readout
The coasting amount based on the position loop time constant Tp and cutting feed time constant Ts is not included in the skip signal input coordinate values.
Therefore, the workpiece coordinate values applying when the skip signal is input can be readout within the error range in the following formula as the skip signal input coordinate values. However, coasting based on response delay time t1 results in a measurement error and so compensation must be provided.

$$\varepsilon = \pm (F/60) * t\,2$$

ε : Readout error
F : Feedrate
t2 : Response error time 0.001 (s)
(a) Measurement value



Readout error of skip signal input coordinates
Readout error with a 60mm/min feedrate is as shown below and the measurement value is within readout error range of ±1 μ m:

ε = ± (60/60) * 0.001 = ± 0.001 (mm)

(2) Readout of other coordinates
The readout coordinate values include the coasting amount. Therefore, when coordinate values at the time of skip signal input is required, reference should be made to the section on the G31 coasting amount to compensate the coordinate value. As in the case of (1), the coasting amount based on the delay error time t2 cannot be calculated, and this generates a measuring error.

Examples of compensating for coasting

(1) Compensating for skip signal input coordinates

```
:
G31 X100.F100 ;         Skip command
G04;                    Machine stop check
#101=#5061 ;            Skip signal input coordinate readout
#102=#110*#111/60 ;     Coasting based on response delay time
#105=#101-#102 ;        Skip signal input coordinates
:
```

#110 = Skip feedrate;
#111 = Response delay time t1;

(2) Compensating for workpiece coordinates

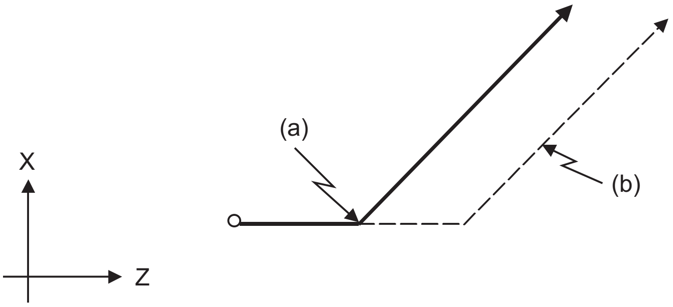
```
:
G31 X100.F100 ;           Skip command
G04;                      Machine stop check
#101=#5061 ;              Skip signal input coordinate readout
#102=#110*#111/60 ;       Coasting based on response delay time
#103=#110*#112/60 ;       Coasting based on position loop time constant
#105=#101-#102-#103 ;     Skip signal input coordinates
:
```

#110 = Skiop feedrate;
#111 = Response delay time t1;
#112 = Position loop time constant Tp;

Operation example

(Example 1)When the next block is an incremental value command

```
G31 Z1000 F100;
G01 U2000 W1000;
```

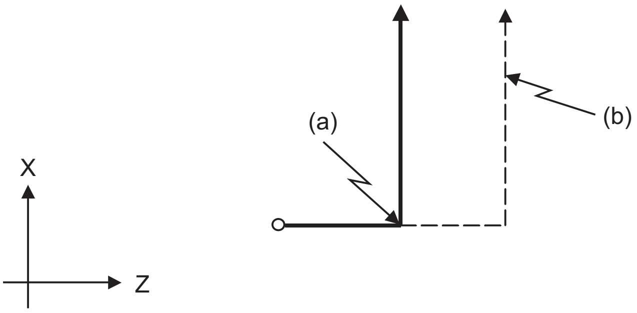


(a) Movement when external signal is entered

(b) Movement when external signal is not entered

(Example 2)When the next block is a one axis movement command with absolute value

```
G31 Z1000 F100;
G01 X1000;
```

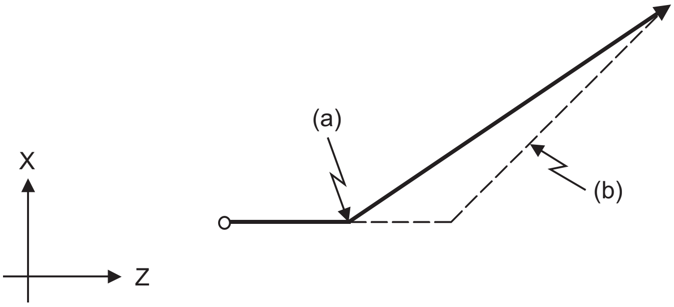


(a) Movement when external signal is entered

(b) Movement when external signal is not entered

(Example 3)When the next block is a two axes movement command with absolute value

```
G31 Z1000 F100;
G01 X1000 Z2000;
```



(a) Movement when external signal is entered

(b) Movement when external signal is not entered

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__ α__ R__ F__ ;

X Z α	Target coordinates
R	Acceleration/deceleration command R0: Acceleration/deceleration time constant=0 (No automatic acceleration/deceleration after interpolation.) R1: Acceleration/deceleration time constant valid. Accelerate/decelerate with the time constant set with the parameters "#2102 skip_tL" and "#2103 skip_t1". R0 is applied when it is omitted.
F	Feedrate (mm/min)

Same with G31.2 and G31.3; Ff is not required with G04.
As with the G31 command, this command executes linear interpolation and when the preset skip signal conditions have been met, the machine is stopped, the remaining commands are canceled, and the next block is executed.

Detailed description

(1) Command the skip speed by the program command or the parameter. Feedrate G31.1 set with the parameter corresponds to "#1176 skip1f", G31.2 corresponds to "#1178 skip2f", and G31.3 corresponds to "#1180 skip3f". However, F modal will not be updated in each case.
(2) A command is skipped if it meets the specified skip signal condition.
(3) The feedrates corresponding to the G31.1, G31.2, and G31.3 commands can be set by parameters.
(4) The skip conditions (logical sum of skip signals which have been set) corresponding to the G31.1, G31.2, G31.3 and G04 commands can be set by parameters.

Parameter setting	Valid skip signal		
	1	2	3
1	○		
2		○	
3	○	○	
4			○
5	○		○
6		○	○
7	○	○	○

(5) Other commands work the same as the G31 (skip function) command.

Operation example

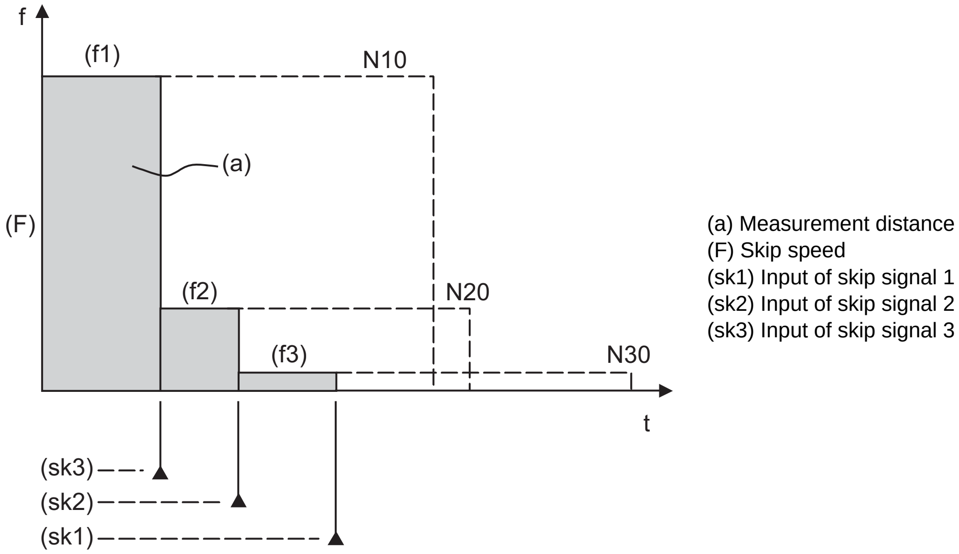
(1) The multi-step skip function enables the following control, thereby improving measurement accuracy and shortening the time required for measurement.

[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)

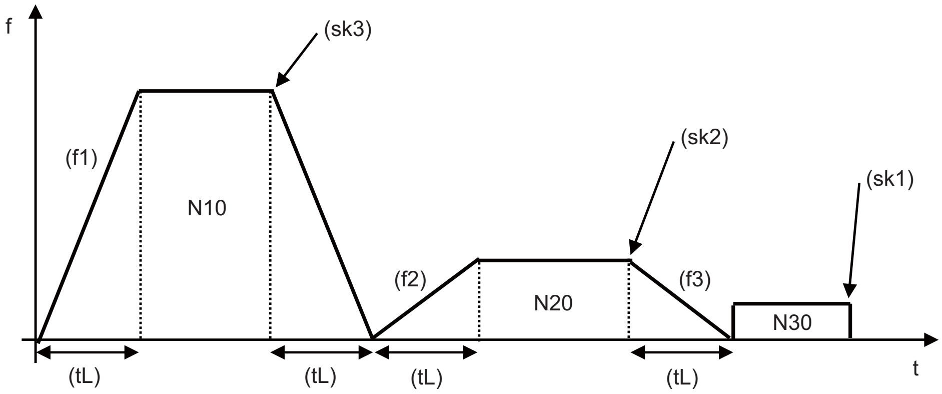
[Program example]

```
N10 G31.1 X200.0 ;
N20 G31.2 X40.0 ;
N30 G31.3 X1.0 ;
```



(a) Measurement distance
(F) Skip speed
(sk1) Input of skip signal 1
(sk2) Input of skip signal 2
(sk3) Input of skip signal 3

(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.



(sk1) Skip signal (tL) Skip time constant

(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

Function and purpose

During linear interpolation by the skip command (G31), operation can be skipped according to the conditions of the skip signal parameter Pp.
If the skip condition specified by the parameter "#1173 dwlskp" (indicating external skip signals 1 to 4) is met during execution of a dwell command (G04), the remaining dwell time is canceled and the following block is executed.

Command format

G31 X__ Z__ α__ P__ R__ F__ ;

X Z α	Target coordinates
P	Skip signal command
R	Acceleration/deceleration command R0: Acceleration/deceleration time constant=0 (No automatic acceleration/deceleration after interpolation.) R1: Acceleration/deceleration time constant valid. Accelerate/decelerate with the time constant set with the parameters "#2102 skip_tL" and "#2103 skip_t1". R0 is applied when it is omitted.
F	Feedrate (mm/min)

Detailed description

(1) The skip speed is specified by program command or parameter. The feedrate by the parameter is set by "#1174 skip_F". Note that the F modal is not updated in each case.

(2) The skip signal is specified by skip signal command p. The command range of "p" is from 1 to 255. If outside the range is commanded, program error (P35) will occur.

Skip signal command P	Valid skip signal							
	8	7	6	5	4	3	2	1
1								○
2							○	
3							○	○
4						○		
5						○		○
6						○	○	
7						○	○	○
8					○			
: : :								
253	○	○	○	○	○	○		○
254	○	○	○	○	○	○	○	
255	○	○	○	○	○	○	○	○

(3) The specified skip signal command is a logical sum of the skip signals.
(Example) G31 X100. P5 F100 ;
Operation is skipped if skip signal 1 or 3 is input.

(4) If skip signal parameter Pp is not specified, it works as a skip function (G31), not as a multi-step skip function. If speed parameter Ff is not specified, the skip speed set by the parameter "#1174 skip_F" will apply.

[Relations between skip and multi-step skip]

Skip specifications	×		○	
	Condition	Speed	Condition	Speed
G31 X100 ; (Without P and F)	Program error (P601)		Skip 1	#1174 skip_F
G31 X100 P5 ; (Without F)	Program error (P602)		Command value	#1174 skip_F
G31 X100 F100 ; (Without P)	Program error (P601)		Skip 1	Command value
G31 X100 P5 F100 ;	Program error (P602)		Command value	Command value

(5) If skip specification is effective and P is specified as an axis address, skip signal parameter P will be given a priority. The axis address "P" will be ignored.
(Example) G31 X100. P500 F100 ;
This is regarded as a skip signal. (The program error (P35) will occur.)

(6) Other than above, the same detailed description as "Skip function; G31" applies.

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. "#2214 SVO14(current limit value in special control)" can be changed. The commanded current limit value is designated with a ratio of the limit current to the rated current.

Command format

G10 L14 Xn ;

L14	Current limit value setting (+ side/- side)
X	Axis address
n	Current limit value (%) Setting range: 1 to 999

Precautions

(1) If the current limit value is reached when the current limit is valid, the current limit reached signal is output.

(2) The following two modes can be used with external signals as the operation after the current limit is reached. The external signal determines which mode applies.

[Normal mode]
The movement command is executed in the current state.
During automatic operation, the movement command is executed until the end, and then move to the next block with the droops still accumulated.

[Interlock mode]
During the occurrence of the droops, it enters to the internal interlock state and the next movement will not be carried out.
During automatic operation, the operation stops at the corresponding block, and the next block is not moved to.
During manual operation, the following same direction commands are ignored.

(3) The position droop generated by the current limit can be canceled when the current limit changeover signal of external signals is canceled. (Note that the axis must not be moving.)

(4) The setting range of the current limit value is 1% to 999%. Commands that exceed this range will cause a program error (P35).

(5) If a decimal point is designated with the G10 command, only the integer will be valid.
Example) G10 L14 X10.123 ; The current limit value will be set to 10%.

(6) For the axis name "C", the current limit value cannot be set from the program (G10 command).
To set from the program, set the axis address with an incremental axis name, or set the axis name to one other than "C".

Appendix 1

Program Errors

These alarms occur during automatic operation, and the causes of these alarms are mainly program errors which occur, for instance, when mistakes have been made in the preparation of the machining programs or when programs which conform to the specification have not been prepared.

(Note) This alarm description is common for M700V/M70V/E70 Series.
It is written on the assumption that all option functions are added.
Confirm with the specifications issued by the machine tool builder before starting use.

P10 No. of simultaneous axes over

Details

The number of axis addresses commanded in a block is exceeds the specifications.

Remedy

- Divide the alarm block command into two.
- Check the specifications.

P11 Illegal axis address

Details

The axis address commanded by the program does not match any of the ones set by the parameter.

Remedy

- Correct the axis names in the program.

P20 Division error

Details

The issued axis command cannot be divided by the command unit.

Remedy

- Correct the program.

P29 Not accept command

Details

The command has been issued when it is impossible.

- The normal line control command (G40.1, G41.1, G42.1) has been issued during the modal in which the normal line control is not acceptable.
- The command has been issued during the modal in which the 2-part system synchronous thread cutting is not acceptable.

Remedy

- Correct the program.

P30 Parity H error

Details

The number of holes per character on the paper tape is even for EIA code and odd for ISO code.

Remedy

- Check the paper tape.
- Check the tape puncher and tape reader.

P31 Parity V error

Details

The number of characters per block on the paper tape is odd.

Remedy

- Make the number of characters per block on the paper tape even.
- Set the parameter parity V selection OFF.

P32 Illegal address

Details

An address not listed in the specifications has been used.

Remedy

- Correct the program address.
- Correct the parameter settings.
- Check the specifications.

P33 Format error

Details

The command format in the program is not correct.

Remedy

- Correct the program.

P34 Illegal G code

Details

The commanded G code is not in the specifications.
An illegal G code was commanded during the coordinate rotation command.

Remedy

- Correct the G code address in the program.

Details

G51.2 or G50.2 was commanded when "#1501 polyax (Rotational tool axis number)" was set to "0".
G51.2 or G50.2 was commanded when the tool axis was set to the linear axis ("#1017 rot (Rotational axis)" is set to"0").

Remedy

- Correct the parameter settings.

P35 Setting value range over

Details

The setting range for the addresses has been exceeded.

Remedy

- Correct the program.

P36 Program end error

Details

"EOR" has been read during tape and memory mode.

Remedy

- Enter the M02 and M30 command at the end of the program.
- Enter the M99 command at the end of the subprogram.

P37 O, N number zero

Details

"0" has been specified for program or sequence No.

Remedy

- Designate program Nos. within a range from 1 to 99999999.
- Designate sequence Nos. within a range from 1 to 99999.

P38 No spec: Add. Op block skip

Details

"/n" has been issued while the optional block skip addition is not in the specifications.

Remedy

- Check the specifications.

P39 No specifications

Details

- A non-specified G code was commanded.
- The selected operation mode is out of specifications.

Remedy

- Check the specifications.

P45 G-CODE COMB.

Details

The combination of G codes in a block is inappropriate.
A part of unmodal G codes and modal G codes cannot be commanded in a same block.

Remedy

Correct the combination of G codes.
Separate the incompatible G codes into different blocks.

P48 Restart pos return incomplete

Details

A travel command was issued before the execution of the block that had been restart-searched.

Remedy

- Carry out program restart again.
  Travel command cannot be executed before the execution of the block that has been restart-searched.

P49 Invalid restart search

Details

- Restart search was attempted for the 3-dimensional circular interpolation.
- Restart search was attempted for the mixed control (cross axis control) command (G110).
- Restart search was attempted during the cylindrical interpolation, polar coordinate interpolation, or tool tip center control.
- Restart search was attempted from a block (G68.2) during the inclined surface machining mode or from the inclined surface machining mode cancel command block (G69).
- Restart search was attempted to the program after direct command mode.

Remedy

- Correct the program.
- Correct the restart search position.

P50 No spec: Inch/Metric change

Details

Inch/Metric changeover (G20/G21) command was issued while the function is out of specifications.

Remedy

- Check the specifications.

P60 Compensation length over

Details

The commanded movement distance is excessive (over 2^{31}).

Remedy

- Correct the command range for the axis address.

P61 No spec: Unidirectional posit.

Details

Unidirectional positioning (G60) was commanded while the function is out of specifications.

Remedy

- Check the specifications.

P62 No F command

Details

- No feed rate command has been issued.
- There is no F command in the cylindrical interpolation or polar coordinate interpolation immediately after the G95 mode is commanded.

Remedy

- The default movement modal command at power ON is G01. This causes the machine to move without a G01 command if a movement command is issued in the program, and an alarm results. Use an F command to specify the feed rate.
- Specify F with a thread lead command.

P63 No spec: High-speed machining

Details

High-speed machining cancel (G5P0) was commanded while the high-speed machining is out of specifications.

Remedy

- Check the specifications.

P65 No spec: High speed mode 3

Details

Remedy

- Check whether the specifications are provided for the high-speed mode III.

P70 Arc end point deviation large

Details

- There is an error in the arc start and end points as well as in the arc center.
- The difference of the involute curve through the start point and the end point is large.
- When arc was commanded, one of the two axes configuring the arc plane was a scaling valid axis.

Remedy

- Correct the numerical values of the addresses that specify the start and end points, arc center as well as the radius in the program.
- Correct the "+" and "-" directions of the address numerical values.
- Check for the scaling valid axis.

P71 Arc center error

Details

- An arc center cannot be obtained in R-specified circular interpolation.
- A curvature center of the involute curve cannot be obtained.

Remedy

- Correct the numerical values of the addresses in the program.
- Correct the start and end points if they are inside of the base circle for involute interpolation. When carrying out tool radius compensation, make sure that the start and end points after compensation will not be inside of the base circle for involute interpolation.
- Correct the start and end points if they are at an even distance from the center of the base circle for involute interpolation.

P72 No spec: Herical cutting

Details

A helical command has been issued though it is out of specifications.

Remedy

- Check whether the specifications are provided for the helical cutting.
- An Axis 3 command has been issued by the circular interpolation command. If there is no helical specification, move the linear axis to the next block.

P73 No spec: Spiral cutting

Details

A spiral command was issued though it is out of specifications.

Remedy

- Issue the G02.1 and G03.1 commands for circular interpolation.
- Check whether the specifications are provided for the spiral cutting.

P74 Can't calculate 3DIM arc

Details

The 3-dimension circular cannot be obtained because the end block was not specified during 3-dimension circular interpolation supplementary modal.
The 3-dimension circular cannot be obtained due to an interruption during 3-dimension circular interpolation supplementary modal.

Remedy

- Correct the program.

P75 3DIM arc illegal

Details

An illegal G code was issued during 3-dimension circular interpolation modal.
Otherwise, 3-dimension circular interpolation command was issued during a modal for which a 3-dimension circular interpolation command cannot be issued.

Remedy

- Correct the program.

P76 No spec: 3DIM arc interpolat

Details

G02.4/G03.4 was commanded though there is no 3-dimension circular interpolation specification.

Remedy

- Check the specifications.

P80 No spec: Hypoth ax interpolat

Details

Hypothetical axis interpolation (G07) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P90 No spec: Thread cutting

Details

A thread cutting command was issued though it is out of specifications.

Remedy

- Check the specifications.

P91 No spec: Var lead threading

Details

Variable lead thread cutting (G34) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P93 Illegal pitch vaule

Details

An illegal thread lead (thread pitch) was specified at the thread cutting command.

Remedy

- Correct the thread lead for the thread cutting command.

P100 No spec: Cylindric interpolat

Details

A cylindrical interpolation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P110 Plane select during figure rot

Details

Plane selection (G17/G18/G19) was commanded during figure rotation.

Remedy

- Correct the machining program.

P111 Plane selected while coord rot

Details

Plane selection commands (G17, G18, G19) were issued during a coordinate rotation was being commanded.

Remedy

- Always command coordinate rotation cancel after the coordinate rotation command, and then issue a plane selection command.

P112 Plane selected while R compen

Details

- Plane selection commands (G17, G18, G19) were issued while tool radius compensation (G41, G42) and nose R compensation (G41, G42, G46) commands were being issued.
- Plane selection commands were issued after completing nose R compensation commands when there were no further axis movement commands after G40, and compensation has not been cancelled.

Remedy

- Issue plane selection commands after completing (axis movement commands issued after G40 cancel command) tool radius compensation and nose R compensation commands.

P113 Illegal plane select

Details

The circular command axis does not correspond to the selected plane.

Remedy

- Select a correct plane before issuing a circular command.

P120 No spec: Feed per rotation

Details

Feed per rotation (G95) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P121 F0 command during arc modal

Details

F0 (F 1-digit feed) was commanded during the arc modal (G02/G03).

Remedy

- Correct the machining program.

P122 No spec: Auto corner override

Details

An auto corner override command (G62) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Delete the G62 command from the program.

P123 No spec: High-accuracy control

Details

High-accuracy control command was issued though it is out of specifications.

Remedy

- Check the specifications.

P124 No spec: Inverse time feed

Details

- The inverse time option is not provided.

Remedy

- Check the specifications.

P125 G93 mode error

Details

- The issued G code command is illegal during G93 mode.
- G93 command was issued during a modal for which inverse time feed cannot be performed.

Remedy

- Correct the program.

P126 Invalid cmnd in high-accuracy

Details

An illegal command was issued during the high-accuracy control mode.

- A G code group 13 command was issued during the high-accuracy control mode.
- Milling, cylindrical interpolation or pole coordinate interpolation was commanded during the high-accuracy control mode.

Remedy

- Correct the program.

P127 No spec: SSS Control

Details

The SSS control valid parameter has been set although there is no SSS control specification.

Remedy

- Check the specifications. Set "#8090 SSS ON" to "0" when there is no SSS control specification.

P128 Machin condtn select I disable

Details

Machining condition selection I was commanded during the mode where the selection command is unavailable.

Remedy

- Check the program and see whether any unavailable mode is included during machining condition selection I. If any, cancel that mode.

P130 2nd M function code illegal

Details

The 2nd miscellaneous function address, commanded in the program, differs from the address set in the parameters.

Remedy

- Correct the 2nd miscellaneous function address in the program.

P131 No spec: Cnst surface ctrl G96

Details

A constant surface speed control command (G96) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Issue a rotation speed command (G97) instead of the constant surface speed control command (G96).

P132 Spindle rotation speed S=0

Details

No spindle rotation speed command has been issued.

Remedy

- Correct the program.

P133 Illegal P-No. G96

Details

The illegal No. was specified for the constant surface speed control axis.

Remedy

- Correct the parameter settings and program that specify the constant surface speed control axis.

P134 G96 Clamp Err.

Details

The constant surface speed control command (G96) was issued without commanding the spindle speed clamp (G92/G50).

Remedy

Press the reset key and carry out the remedy below.

- Check the program.
- Issue the G92/G50 command before the G96 command.
- Command the constant surface speed cancel (G97) to switch to the rotation speed command.

P140 No spec: Pos compen cmd

Details

The position compensation command (G45 to G48) is out of specifications.

Remedy

- Check the specifications.

P141 Pos compen during rotation

Details

Position compensation was commanded during the figure rotation or coordinate rotation command.

Remedy

- Correct the program.

P142 Pos compen invalid arc

Details

Position compensation cannot be executed with the issued arc command.

Remedy

- Correct the program.

P150 No spec: Nose R compensation

Details

- Tool radius compensation commands (G41 and G42) were issued though they are out of specifications.
- Nose R compensation commands (G41, G42, and G46) were issued though they are out of specifications.

Remedy

- Check the specifications.

P151 Radius compen during arc mode

Details

A compensation command (G40, G41, G42, G43, G44, or G46) has been issued in the arc modal (G02 or G03).

Remedy

- Issue the linear command (G01) or rapid traverse command (G00) in the compensation command block or cancel block.
  (Set the modal to linear interpolation.)

P152 No intersection

Details

- In interference block processing during execution of a tool radius compensation (G41 or G42) or nose R compensation (G41, G42, or G46) command, the intersection point after one block is skipped cannot be determined.
- The compensation amount cannot be calculated during the tool radius compensation for 5-axis machining (G41.2,G42.2).

Remedy

- Correct the program.

P153 Compensation interference

Details

An interference error has occurred while the tool radius compensation command (G41 or G42) or nose R compensation command (G41, G42 or G46) was being executed.

Remedy

- Correct the program.

P154 No spec: 3D compensation

Details

A three-dimensional compensation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P155 Fixed cyc exec during compen

Details

A fixed cycle command has been issued in the radius compensation mode.

Remedy

- Issue a radius compensation cancel command (G40) to cancel the radius compensation mode that has been applied since the fixed cycle command was issued.

P156 R compen direction not defined

Details

A shift vector with undefined compensation direction was found at the start of G46 nose R compensation.

Remedy

- Change the vector to that which has the defined compensation direction.
- Change the tool to that which has a different tip point No.

P157 R compen direction changed

Details

During G46 nose R compensation, the compensation direction is reversed.

Remedy

- Change the G command to that which allows the reversed compensation direction (G00, G28, G30, G33, or G53).
- Change the tool to that which has a different tip point No.
- Enable "#8106 G46 NO REV-ERR".

P158 Illegal tip point

Details

An illegal tip point No. (other than 1 to 8) was found during G46 nose R compensation.

Remedy

- Correct the tip point No.

P159 C-CMP REMAINS

Details

One of the followings was commanded while the compensation has not been canceled (the compensation amount remained).

(1) Automatic tool nose R compensation (G143)

(2) Radius compensation (G145)

(3) Plane selection (G17 to G19)

(4) Skip (G31 or G31.1/G31.2/G31.3)

(5) Fixed cycle for drilling (G81 to G89)

(6) Compound type fixed cycle II (G74 to G76)

Remedy

- Cancel the compensation (with compensation amount "0") before commanding.
- Command the G00 move block before the block with a command among (1) to (6).

P160 G53 CMP. ERR

Details

- G53 was commanded during nose R compensation (by G41/G42/G46).
- G53 was commanded to the block where the nose R compensation mode is changed (with G40/ G41/G42/G46).
- G53 was commanded while nose R compensation amount has not been cleared.

Remedy

- Correct the program.
- When issuing G53 after G40 command, move the compensation plane axis by G00/G01/G02/G03 command before issuing G53.

P161 No spec: 5ax tool R compensate

Details

Tool radius compensation for 5-axis machining is not included in the specifications.

Remedy

- Check the specifications.

P162 Disable Cmd in 5ax tool R comp

Details

A command (G or T command, etc) was issued during tool radius compensation for 5-axis machining, although it is disabled during the compensation.

Remedy

- Cancel the tool radius compensation for 5-axis machining.

P163 5 ax tool R comp is disable

Details

Tool radius compensation for 5-axis machining was commanded in a mode where the command is disabled.

Remedy

- Cancel the mode that disables the command.

P170 No offset number

Details

- No compensation No. (DOO, TOO or HOO) command was given when the radius compensation (G41, G42, G43 or G46) command was issued. Otherwise, the compensation No. is larger than the number of sets in the specifications.
- When setting the L system tool life management II, the tool group management program was executed with the tool life management disabled.

Remedy

- Add the compensation No. command to the compensation command block.
- Check the number of sets for the tool compensation Nos. and correct the compensation No. command to be within the number of sets.
- When setting the L system tool life management II ("#1096 T_Ltyp"="2"), enable the tool life management ("#1103 T_Life"="1") to execute a tool group management program.

P171 No spec:Comp input by prog G10

Details

Compensation data input by program (G10) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P172 G10 L number error

Details

An address of G10 command is not correct.

Remedy

- Correct the address L No. of the G10 command.

P173 G10 P number error

Details

The compensation No. at the G10 command is not within the permitted number of sets in the specifications.

Remedy

- Check the number of sets for the tool compensation Nos. and correct the address P designation to be within the number of sets.

P174 No spec:Comp input by prog G11

Details

Compensation data input by program cancel (G11) was commanded though there is no specification of compensation data input by program.

Remedy

- Check the specifications.

P177 Tool life count active

Details

Registration of tool life management data with G10 was attempted when the "usage data count valid" signal was ON.

Remedy

- The tool life management data cannot be registered during the usage data count. Turn the "usage data count valid" signal OFF.

P178 Tool life data entry over

Details

The number of registration groups, total number of registered tools or the number of registrations per group exceeded the range in the specifications.

Remedy

- Correct the number of registrations.

P179 Illegal group No.

Details

- A duplicate group No. was found at the registration of the tool life management data with G10.
- A group No. that was not registered was designated during the T****99 command.
- An M code command, which must be issued as a single command, coexists in the same block as that of another M code command.
- The M code commands set in the same group exist in the same block.

Remedy

- Register the tool life data once for one group: commanding with a duplicate group No. is not allowed.
- Correct to the group No.

P180 No spec: Drilling cycle

Details

A fixed cycle command (G72 - G89) was issued though it is out of specifications.

Remedy

- Check the specifications.
- Correct the program.

P181 No spindle command (Tap cycle)

Details

Spindle rotation speed (S) has not been commanded in synchronous tapping.

Remedy

- Command the spindle rotation speed (S) in synchronous tapping.
- When "#8125 Check Scode in G84" is set to "1", enter the S command in the same block where the synchronous tapping command is issued.

P182 Synchronous tap error

Details

- Connection to the main spindle unit was not established.
- The synchronous tapping was attempted with the spindle not serially connected under the multiple-spindle control I.

Remedy

- Check connection to the main spindle.
- Check that the main spindle encoder exists.
- Set 1 to the parameter #3024 (sout).

P183 No pitch/thread number

Details

The pitch or number of threads has not been commanded in the tap cycle of a fixed cycle for drilling command.

Remedy

- Specify the pitch data and the number of threads by F or E command.

P184 Pitch/thread number error

Details

- The pitch or the number of threads per inch is illegal in the tap cycle of the fixed cycle for drilling command.
- The pitch is too small for the spindle rotation speed.
- The thread number is too large for the spindle rotation speed.

Remedy

- Correct the pitch or the number of threads per inch.

P185 No spec: Sync tapping cycle

Details

Synchronous tapping cycle (G84/G74) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P186 Illegal S cmnd in synchro tap

Details

S command was issued during synchronous tapping modal.

Remedy

- Cancel the synchronous tapping before issuing the S command.

P190 No spec: Turning cycle

Details

A lathe cutting cycle command was issued though it is out of specifications.

Remedy

- Check the specification.
- Delete the lathe cutting cycle command.

P191 Taper length error

Details

In the lathe cutting cycle, the specified length of taper section is illegal.

Remedy

- Set the smaller radius value than the axis travel amount in the lathe cycle command.

P192 Chamfering error

Details

Chamfering in the thread cutting cycle is illegal.

Remedy

- Set a chamfering amount not exceeding the cycle.

P200 No spec: MRC cycle

Details

The compound type fixed cycle for turning machining I (G70 to G73) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P201 Program error (MRC)

Details

- The subprogram, called with a compound type fixed cycle for turning machining I command, has at least one of the following commands: reference position return command (G27, G28, G29, G30); thread cutting (G33, G34); fixed cycle skip-function (G31, G31.n).
- An arc command was found in the first movement block of the finished shape program in compound type fixed cycle for turning machining I.

Remedy

- Delete G27, G28, G29, G30, G31, G33, G34, and fixed cycle G codes from the subprogram called with the compound type fixed cycle for turning machining I commands (G70 to G73).
- Delete G02 and G03 from the first movement block of the finished shape program in compound type fixed cycle for turning machining I.

P202 Block over (MRC)

Details

The number of blocks in the shape program of the compound type fixed cycle for turning machining I is over 50 or 200 (the maximum number differs according to the model).

Remedy

- Set a 50/200 or less value for the number of blocks in the shape program called by the compound type fixed cycle for turning machining I commands (G70 to G73). (The maximum number differs according to the model).

P203 D cmnd figure error (MRC)

Details

A proper shape will not obtained by executing the shape program for the compound type fixed cycle for turning machining I (G70 to G73).

Remedy

- Correct the shape program for the compound type fixed cycle for turning machining I (G70 to G73).

P204 E cmnd fixed cycle error

Details

A command value of the compound type fixed cycle for turning machining (G70 to G76) is illegal.

Remedy

- Correct the command value of the compound type fixed cycle for turning machining (G70 to G76).

P210 No spec: Pattern cycle

Details

A compound type fixed cycle for turning machining II (G74 to G76) command was commanded though it is out of specifications.

Remedy

- Check the specifications.

P220 No spec: Special fixed cycle

Details

There are no special fixed cycle specifications.

Remedy

- Check the specifications.

P221 No. of special fixed holes = 0

Details

"0" has been specified for the number of holes in special fixed cycle mode.

Remedy

- Correct the program.

P222 G36 angle error

Details

A G36 command specifies "0" for angle intervals.

Remedy

- Correct the program.

P223 G12/G13 radius error

Details

The radius value specified with a G12 or G13 command is below the compensation amount.

Remedy

- Correct the program.

P224 No spec: Circular (G12/G13)

Details

There are no circular cutting specifications.

Remedy

- Check the specifications.

P230 Subprogram nesting over

Details

Over 8 times of subprogram calls have been done in succession from a subprogram.

- A M198 command was found in the program in the data server.
- The program in the IC card has been called more than once (the program in the IC card can be called only once during nested).

Remedy

- Correct the program so that the number of subprogram calls does not exceed 8 times.

P231 No sequence No.

Details

The sequence No., commanded at the return from the subprogram or by GOTO in the subprogram call, was not set.

Remedy

- Specify the sequence Nos. in the call block of the subprogram.

P232 No program No.

Details

- The machining program has not been found when the machining program is called.
- The file name of the program registered in IC card is not corresponding to O No.

Remedy

- Enter the machining program.
- Check the subprogram storage destination parameters.
- Ensure that the external device (including IC card) that contains the file is mounted.

P235 Program editing

Details

Operation was attempted for the file under program editing.

Remedy

- Execute the program again after completion of program editing.

P240 No spec: Variable commands

Details

A variable command (with #) was issued though it is out of specifications.

Remedy

- Check the specifications.

P241 No variable No.

Details

The variable No. commanded is out of the range specified in the specifications.

Remedy

- Check the specifications.
- Correct the program variable No.

P242 = not defined at vrble set

Details

The "=" sign has not been commanded when a variable is defined.

Remedy

- Designate the "=" sign in the variable definition of the program.

P243 Can't use variables

Details

An invalid variable has been specified in the left or right side of an operation expression.

Remedy

- Correct the program.

P244 Invalid set date or time

Details

Date or time was set earlier than current date or time in the system variables (#3011, #3012) when the system lock was valid.

Remedy

- Date or time cannot be changed.
- Correct the program.

P250 No spec: Figure rotation

Details

Figure rotation (M98 I_J_P_H_L_) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P251 Figure rotation overlapped

Details

Figure rotation command was issued during figure rotation.

Remedy

- Correct the program.

P252 Coord rotate in fig. rotation

Details

A coordinate rotation related command (G68, G69) was issued during figure rotation.

Remedy

- Correct the program.

P260 No spec: Coordinates rotation

Details

A coordinate rotation command was issued though it is out of specifications.

Remedy

- Check the specifications.

P261 G code illegal (Coord rot)

Details

Another G code or a T command has been issued in the block of coordinate rotation command.

Remedy

- Correct the program.

P262 Illegal modal (Coord rot)

Details

A coordinate rotation command has been issued during modal in which coordinate rotation is not allowed.

Remedy

- Correct the program.

P270 No spec: User macro

Details

A macro specification was commanded though it is out of specifications.

Remedy

- Check the specifications.

P271 No spec: Macro interrupt

Details

A macro interruption command has been issued though it is out of specifications.

Remedy

- Check the specifications.

P272 NC and macro texts in a block

Details

An executable statement and a macro statement exist together in the same block.

Remedy

- Place the executable statement and macro statement in separate blocks in the program.

P273 Macro call nesting over

Details

The number of macro call nests exceeded the limit imposed by the specifications.

Remedy

- Correct the program so that the macro calls do not exceed the limit imposed by the specifications.

P275 Macro argument over

Details

The number of argument sets in the macro call argument type II has exceeded the limit.

Remedy

- Correct the program.

P276 Illegal G67 command

Details

A G67 command was issued though it was not during the G66 command modal.

Remedy

- Correct the program.
- Issue G66 command before G67 command, which is a call cancel command.

P277 Macro alarm message

Details

An alarm command has been issued in #3000.

Remedy

- Refer to the operator messages on the diagnosis screen.
- Refer to the instruction manual issued by the machine tool builder.

P280 Brackets [ ] nesting over

Details

Over five times have the parentheses "[" or "]" been used in a single block.

Remedy

- Correct the program so that the number of "[" or "]" is five or less.

P281 Brackets [ ] not paired

Details

A single block does not have the same number of commanded parentheses "[" as that of "]".

Remedy

- Correct the program so that "[" and "]" parentheses are paired up properly.

P282 Calculation impossible

Details

The arithmetic formula is incorrect.

Remedy

- Correct the formula in the program.

P283 Divided by zero

Details

The denominator of the division is zero.

Remedy

- Correct the program so that the denominator for division in the formula is not zero.

P288 IF EXCESS

Details

Over ten times of IF statement nesting have been done.

Remedy

Correct the program so that the number of IF statement nesting does not exceed ten.

P289 IF-ENDIF MMC.

Details

An IF statement is not ended with ENDIF.
THEN/ELSE has been commanded while there is no IF command.

Remedy

- Correct the program so that IF statements are ended with ENDIFs.
- Put IF[condition] before THEN/ELSE command.

P290 IF sentence error

Details

There is an error in the "IF[<conditional>]GOTO(" statement.

Remedy

- Correct the program.

P291 WHILE sentence error

Details

There is an error in the "WHILE[<conditional>]DO(-END(" statement.

Remedy

- Correct the program.

P292 SETVN sentence error

Details

There is an error in the "SETVN(" statement when the variable name setting was made.

Remedy

- Correct the program.
- The number of characters in the variable name of the SETVN statement must be 7 or less.

P293 DO-END nesting over

Details

The number of DO-END nesting levels in the "WHILE[<conditional>]DO(-END(" statement has exceeded 27.

Remedy

- Correct the program so that the nesting levels of the DO-END statement does not exceed 27.

P294 DO and END not paired

Details

The DOs and ENDs are not paired off properly.

Remedy

- Correct the program so that the DOs and ENDs are paired off properly.

P295 WHILE/GOTO in tape

Details

There is a WHILE or GOTO statement on the tape during tape operation.

Remedy

- Apply memory mode operation instead of tape mode that does not allow the execution of the program with a WHILE or GOTO statement.

P296 No address (macro)

Details

A required address has not been specified in the user macro.

Remedy

- Correct the program.

P297 Address-A error

Details

The user macro does not use address A as a variable.

Remedy

- Correct the program.

P298 G200-G202 cmnd in tape

Details

User macro G200, G201, or G202 was specified during tape or MDI mode.

Remedy

- Correct the program.

P300 Variable name illegal

Details

The variable names have not been commanded properly.

Remedy

- Correct the variable names in the program.

P301 Variable name duplicated

Details

A duplicate variable name was found.

Remedy

- Correct the program so that no duplicate name exists.

P310 Not use GMSTB macro code

Details

G, M, S, T, or B macro code was called during fixed cycle.

Remedy

- Correct the program.
- Correct the parameter settings.

P350 No spec: Scaling command

Details

The scaling command (G50, G51) was issued though it is out of specifications.

Remedy

- Check the specifications.

P360 No spec: Program mirror

Details

A mirror image (G50.1 or G51.1) command has been issued though the programmable mirror image specifications are not provided.

Remedy

- Check the specifications.

P370 No spec: Facing t-post MR

Details

The facing turret mirror image specifications are not provided.

Remedy

- Check the specifications.

P371 Facing t-post MR illegal

Details

- Mirror image for facing tool posts was commanded to an axis in external mirror image or parameter mirror image.
- The commanded mirror image for facing tool posts enables the mirror image for a rotary axis.

Remedy

- Correct the program.
- Correct the parameter settings.

P380 No spec: Corner R/C

Details

The corner R/C was issued though it is out of specifications.

Remedy

- Check the specifications.
- Delete the corner chamfering/corner rounding command in the program.

P381 No spec: Arc R/C

Details

Corner chamfering II or corner rounding II was commanded in the arc interpolation block though it is out of specifications.

Remedy

- Check the specifications.

P382 No corner movement

Details

The block next to corner chamfering/ corner rounding is not a travel command.

Remedy

- Replace the block succeeding the corner chamfering/ corner rounding command by G01 command.

P383 Corner movement short

Details

The travel distance in the corner chamfering/corner rounding command was shorter than the value in the corner chamfering/corner rounding command.

Remedy

- Set the smaller value for the corner chamfering/corner rounding than the travel distance.

P384 Corner next movement short

Details

The travel distance in the following block in the corner chamfering/corner rounding command was shorter than the value in the corner chamfering/corner rounding command.

Remedy

- Set the smaller value for the corner chamfering/corner rounding than the travel distance in the following block.

P385 Corner during G00/G33

Details

A block with corner chamfering/corner rounding was given during G00 or G33 modal.

Remedy

- Correct the program.

P390 No spec: Geometric

Details

A geometric command was issued though it is out of specifications.

Remedy

- Check the specifications.

P391 No spec: Geometric arc

Details

There are no geometric IB specifications.

Remedy

- Check the specifications.

P392 Angle < 1 degree (GEOMT)

Details

The angular difference between the geometric line and line is 1° or less.

Remedy

- Correct the geometric angle.

P393 Inc value in 2nd block (GEOMT)

Details

The second geometric block has a command with an incremental value.

Remedy

- Issue a command with an absolute value in the second geometric block.

P394 No linear move command (GEOMT)

Details

The second geometric block contains no linear command.

Remedy

- Issue the G01 command.

P395 Illegal address (GEOMT)

Details

The geometric format is invalid.

Remedy

- Correct the program.

P396 Plane selected in GEOMT ctrl

Details

A plane switching command was issued during geometric command processing.

Remedy

- Complete the plane switching command before geometric command processing.

P397 Arc error (GEOMT)

Details

In geometric IB, the circular arc end point does not contact or cross the next block start point.

Remedy

- Correct the geometric circular arc command and the preceding and following commands.

P398 No spec: Geometric1B

Details

A geometric command was issued though the geometric IB specifications are not provided.

Remedy

- Check the specifications.

P411 Illegal modal G111

Details

- G111 was issued during milling mode.
- G111 was issued during nose R compensation mode.
- G111 was issued during constant surface speed.
- G111 was issued during mixed control (cross axis control).
- G111 was issued during fixed cycle.
- G111 was issued during polar coordinate interpolation.
- G111 was issued during cylindrical interpolation mode.

Remedy

- Before commanding G111, cancel the following commands.
- Milling mode
- Nose R compensation
- Constant surface speed
- Mixed control (cross axis control)
- Fixed cycle
- Polar coordinate interpolation
- Cylindrical interpolation

P412 No spec: Axis name switch

Details

Axis name switch (G111) was issued though it is out of specifications.

Remedy

- Check the specifications.

P420 No spec: Para input by program

Details

Parameter input by program (G10) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P421 Parameter input error

Details

- The specified parameter No. or set data is illegal.
- An illegal G command address was input in parameter input mode.
- A parameter input command was issued during fixed cycle modal or nose R compensation.
- G10L70, G11 were not commanded in independent blocks.

Remedy

- Correct the program.

P430 R-pnt return incomplete

Details

- A command was issued to move an axis, which has not returned to the reference position, away from that reference position.
- A command was issued to an axis removal axis.

Remedy

- Execute reference position return manually.
- Disable the axis removal on the axis for which the command was issued.

P431 No spec: 2,3,4th R-point ret

Details

A command for second, third or fourth reference position return was issued though there are no such command specifications.

Remedy

- Check the specifications.

P432 No spec: Start position return

Details

Start position return (G29) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P433 No spec: R-position check

Details

Reference position check (G27) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P434 Compare error

Details

One of the axes did not return to the reference position when the reference position check command (G27) was executed.

Remedy

- Correct the program.

P435 G27 and M commands in a block

Details

An M command was issued simultaneously in the G27 command block.

Remedy

- Place the M code command, which cannot be issued in a G27 command block, in separate block from G27 command block.

P436 G29 and M commands in a block

Details

An M command was issued simultaneously in the G29 command block.

Remedy

- Place the M code command, which cannot be issued in a G29 command block, in separate block from G29 command block.

P438 G52 invalid during G54.1

Details

A local coordinate system command was issued during execution of the G54.1 command.

Remedy

- Correct the program.

P450 No spec: Chuck barrier

Details

The chuck barrier on command (G22) was specified although the chuck barrier is out of specifications.

Remedy

- Check the specifications.

P451 No spec: Stroke chk bef travel

Details

Stroke check before travel (G22/G23) was commanded though it is out of specifications.

Remedy

- Check the specifications.

P452 Limit before travel exists

Details

An illegal command, which places the axis travel start/end point in the prohibited area or moves the axis through the prohibited area, was detected when Stroke check before travel (G22) was commanded.

Remedy

- Correct the coordinate values of the axis address commanded in the program.

P460 Tape I/O error

Details

An error has occurred in the tape reader. Otherwise an error has occurred in the printer during macro printing.

Remedy

- Check the power and cable of the connected devices.
- Correct the I/O device parameters.

P461 File I/O error

Details

- A file of the machining program cannot be read.
- IC card has not been inserted.

Remedy

- In memory mode, the programs stored in memory may have been destroyed. Output all of the programs and tool data and then format the system.
- Ensure that the external device (including an IC card, etc) that contains the file is mounted.
- Correct the parameter settings for HD operation or IC card operation.

P462 Computer link commu error

Details

A communication error occurred during the BTR operation.

Remedy

- "L01 Computer link error" is displayed simultaneously. Take the remedy corresponding to the error No.

P480 No spec: Milling

Details

- Milling was commanded though it is out of specifications.
- Polar coordinate interpolation was commanded though it is out of specifications.

Remedy

- Check the specifications.

P481 Illegal G code (mill)

Details

- An illegal G code was used during the milling mode.
- An illegal G code was used during cylindrical interpolation or polar coordinate interpolation.
- The G07.1 command was issued during the tool radius compensation.

Remedy

- Correct the program.

P482 Illegal axis (mill)

Details

- A rotary axis was commanded during the milling mode.
- Milling was executed though an illegal value was set for the milling axis No.
- Cylindrical interpolation or polar coordinate interpolation was commanded during mirror image.
- Cylindrical interpolation or polar coordinate interpolation was commanded before the tool compensation was completed after the T command.
- G07.1 was commanded when cylindrical interpolation was not possible (there is no rotary axis, or external mirror image is ON).
- An axis other than a cylindrical coordinate system axis was commanded during cylindrical interpolation.

Remedy

- Correct the machining program, parameters and PLC interface signals.

P484 R-pnt ret incomplete (mill)

Details

- Movement was commanded to an axis that had not completed reference position return during the milling mode.
- Movement was commanded to an axis that had not completed reference position return during cylindrical interpolation or polar coordinate interpolation.

Remedy

- Carry out manual reference position return.

P485 Illegal modal (mill)

Details

- The milling mode was turned ON during nose R compensation or constant surface speed control.
- A T command was issued during the milling mode.
- The mode was switched from milling to cutting during tool compensation.
- Cylindrical interpolation or polar coordinate interpolation was commanded during the constant surface speed control mode (G96).
- The command unacceptable in the cylindrical interpolation was issued.
- A T command was issued during the cylindrical interpolation or polar coordinate interpolation mode.
- A movement command was issued when the plane was not selected just before or after the G07.1 command.
- A plane selection command was issued during the polar coordinate interpolation mode.
- Cylindrical interpolation or polar coordinate interpolation was commanded during tool radius compensation.
- The G16 plane in which the radius value of a cylinder is "0" was specified.
- A cylindrical interpolation or polar coordinate interpolation command was issued during coordinate rotation by program.

Remedy

- Correct the program.
- Issue G40 or G97 before issuing G12.1.
- Issue a T command before issuing G12.1.
- Issue G40 before issuing G13.1.
- Specify the radius value of a cylinder other than "0", or specify the X axis's current value other than "0" before issuing G12.1/G16.

P486 Milling error

Details

- The milling command was issued during the mirror image (when parameter or external input is turned ON).
- Polar coordinate interpolation, cylindrical interpolation or milling interpolation was commanded during mirror image for facing tool posts.
- The start command of the cylindrical interpolation or polar coordinate interpolation was issued during the normal line control.

Remedy

- Correct the program.

P501 Cross (G110) impossible

Details

Mixed control (cross axis control) command (G110) was issued in the following modes.

- During nose R compensation mode
- During pole coordinate interpolation mode
- During cylindrical interpolation mode
- During balance cut mode
- During fixed cycle machining mode
- During facing turret mirror image
- During constant surface speed control mode
- During hobbing mode
- During axis name switch

Remedy

- Correct the program.

P503 Illegal G110 axis

Details

- The commanded axis does not exist.
- The mixed control (cross axis control) (G110) was commanded to the axis for which the mixed control (cross axis control) is disabled.
- The number of axes included in the mixed control (cross axis control) (G110) command is exceeding the maximum number of axes per part system.

Remedy

- Correct the program.

P511 Synchronization M code error

Details

- Two or more synchronization M codes were commanded in the same block.
- The synchronization M code and "!" code were commanded in the same block.
- Synchronization with the M code was commanded in 3rd part system or more. (Synchronization with the M code is valid only in 1st part system or 2nd part system.)

Remedy

- Correct the program.

P520 Control axis superimposition/Designated axis illegal

Details

An axis which was impossible to superimpose was designated as a master axis or superimposing axis.

Remedy

Correct the program.

P521 Illegal synchronization axis

Details

The axis specified as a basic or synchronous axis of synchronization across part systems cannot be synchronized.

Remedy

- Correct the program.

P530 DEC. POINT ERR

Details

A decimal point was added to the address where the decimal point command is not allowed. ("#1274 ext10/bit0 (Type of address enabling/disabling decimal point command)")

Remedy

Do not add any decimal point to the addresses where the decimal point is not allowed.

P544 No spec: Wk instl err cmp

Details

The workpiece installation error compensation function is out of the specifications.

Remedy

- Check the specifications.

P545 Invld cmd in wk instl err cmp

Details

During workpiece installation error compensation, a command impossible to issue (such as G command) was issued.

Remedy

- Check the program. If you wish to issue a command impossible to issue (such as G command) during workpiece installation error compensation, cancel workpiece installation error compensation once.

P546 Wk instl err cmp cmd invalid

Details

- Workpiece installation error compensation was commanded in a G modal in which commanding it is not allowed.
- An illegal G command was issued in the block that has a workpiece installation error compensation command.

Remedy

- Check the program. Also check the G modals which were issued at commanding the workpiece installation error compensation, and cancel illegal ones.
- Issue the G command in a separate block.

P547 Illegal wk instl err cmp cmd

Details

A command in which the rotary axis's travel distance exceeds 180 degrees was issued.

Remedy

- Divide the travel command so that the rotary axis's travel distance per block is less than 180 degrees.

P550 No spec: G06.2(NURBS)

Details

There is no NURBS interpolation option.

Remedy

- Check the specifications.

P551 G06.2 knot error

Details

The knot (k) command value is smaller than the value for the previous block.

Remedy

- Correct the program.
- Specify the knot by monotone increment.

P552 Start point of 1st G06.2 err

Details

The block end point immediately before the G06.2 command and the G06.2 first block command value do not match.

Remedy

- Match the G06.2 first block coordinate command value with the previous block end point.

P554 Invld manual interrupt in G6.2

Details

Manual interruption was executed in a block that applies the G06.2 mode.

Remedy

- Execute the manual interruption in the block that does not apply the G06.2 mode.

P555 Invalid restart during G06.2

Details

Restart was attempted from the block that applies G06.2 mode.

Remedy

- Restart from the block that does not apply the G06.2 mode.

P600 No spec: Auto TLM

Details

An automatic tool length measurement command (G37) was issued though it is out of specifications.

Remedy

- Check the specifications.

P601 No spec: Skip

Details

A skip command (G31) was issued though it is out of specifications.

Remedy

- Check the specifications.

P602 No spec: Multi skip

Details

A multiple skip command (G31.1, G31.2, G31.3 or G31 Pn) was issued though it is out of specifications.

Remedy

- Check the specifications.

P603 Skip speed 0

Details

The skip speed is "0".

Remedy

- Specify the skip speed.

P604 TLM illegal axis

Details

No axis was specified in the automatic tool length measurement block. Otherwise, two or more axes were specified.

Remedy

- Specify only one axis.

P605 T & TLM command in a block

Details

The T code is in the same block as the automatic tool length measurement block.

Remedy

- Specify the T code before the automatic tool length measurement block.

P606 T cmnd not found before TLM

Details

The T code was not yet specified in automatic tool length measurement.

Remedy

- Specify the T code before the automatic tool length measurement block.

P607 TLM illegal signal

Details

The measurement position arrival signal turned ON before the area specified by the D command or "#8006 ZONE d". Otherwise, the signal remained OFF to the end.

Remedy

- Correct the program.

P608 Skip during radius compen

Details

A skip command was issued during radius compensation processing.

Remedy

- Issue a radius compensation cancel (G40) command or remove the skip command.

P610 Illegal parameter

Details

- The parameter setting is not correct.
  - G114.1 was commanded while the spindle synchronization was selected with the PLC interface signal.
  - G110 was commanded while the mixed control (cross axis control) was selected with the PLC interface signal.
  - G125 was commanded while the control axis synchronization across part systems was selected with the PLC interface signal.
  - G126 was commanded while the control axis superimposition was selected with the PLC interface signal.

Remedy

- Correct the settings of "#1514 expLinax (Exponential function interpolation linear axis)" and "#1515 expRotax (Exponential function interpolation rotary axis)".
  - Correct the program.
  - Correct the parameter settings.

P611 No spec: Exponential function

Details

There is no specification for the exponential interpolation.

Remedy

- Check the specifications.

P612 Exponential function error

Details

A travel command for exponential interpolation was issued during mirror image for facing tool posts.

Remedy

- Correct the program.

P700 Illegal command value

Details

Spindle synchronization was commanded to a spindle that is not connected serially.

Remedy

- Correct the program.
- Correct the parameter settings.

P900 No spec: Normal line control

Details

A normal line control command (G40.1, G41.1, or G42.1) was issued though it is out of specifications.

Remedy

- Check the specifications.

P901 Normal line control axis G92

Details

A coordinate system preset command (G92) was issued to a normal line control axis during normal line control.

Remedy

- Correct the program.

P902 Normal line control axis error

Details

- The normal line control axis was set to a linear axis.
- The normal line control axis was set to the linear type rotary axis II axis.
- The normal line control axis has not been set.
- The normal line control axis is the same as the plane selection axis.

Remedy

- Correct the normal line control axis setting.

P903 Plane chg in Normal line ctrl

Details

The plane selection command (G17, G18, or G19) was issued during normal line control.

Remedy

- Delete the plane selection command (G17, G18, or G19) from the program of the normal line control.

P920 No spec: 3D coord conv

Details

There is no specification for 3-dimensional coordinate conversion.

Remedy

- Check the specifications.

P921 Illegal G code at 3D coord

Details

The commanded G code cannot be performed during 3-dimensional coordinate conversion modal.

Remedy

- Refer to "MITSUBISHI CNC 700/70 Series Programming Manual (Machining Center System)" for usable G commands.
- When the parameter "#8158 Init const sur spd" is enabled, disable the parameter or issue the constant surface speed control cancel (G97) command.

P922 Illegal mode at 3D coord

Details

A 3-dimensional coordinate conversion command was issued during a modal for which 3-dimensional coordinate conversion cannot be performed.

Remedy

- Refer to "MITSUBISHI CNC 700/70 Series Programming Manual (Machining Center System)" for usable G commands.

P923 Illegal addr in 3D coord blk

Details

A G code and G68 was commanded in a block though the G code cannot be commanded with G68.

Remedy

- Refer to "MITSUBISHI CNC 700/70 Series Programming Manual (Machining Center System)" for usable G commands.

P930 No spec: Tool axis compen

Details

A tool length compensation along the tool axis command was issued though it is out of specifications.

Remedy

- Check the specifications.

P931 Executing tool axis compen

Details

There is a G code that cannot be commanded during tool length compensation along the tool axis.

Remedy

- Correct the program.

P932 Rot axis parameter error

Details

There is an illegal linear axis name or rotary axis name set in the rotary axis configuration parameters.
There is an illegal setting in the parameter concerning the configuration of the inclined surface machining axis.

Remedy

- Set the correct value and turn the power ON again.

P940 No spec: Tool tip control

Details

There is no specification for tool tip center control.

Remedy

- Check the specifications.

P941 Invalid T tip control command

Details

A tool tip center control command was issued during a modal for which a tool tip center control command cannot be issued.

Remedy

- Correct the program.

P942 Invalid cmnd during T tip ctrl

Details

A G code that cannot be commanded was issued during tool tip center control.

Remedy

- Correct the program.

P943 Tool posture command illegal

Details

In tool tip center control type 1, if the signs at the tool-side rotary axis or table base-side rotary axis start and finish points differ, a tool base-side rotary axis or table workpiece-side rotary axis rotation exists for the same block, and does not pass a singular point.
In tool tip center control type 2, the posture vector command is incorrect.

Remedy

- Correct the program.

P950 No spec: Tilt face machining

Details

Inclined surface machining option is not supported.

Remedy

- Check the specifications.

P951 Ill cmd in tilt face machining

Details

A forbidden command (G command, etc) was issued during inclined surface machining.

Remedy

- Check the program. If you want to execute a command (G command, etc) that is unavailable during inclined surface machining, cancel the inclined surface machining.

P952 Inclined face cut prohibited

Details

Inclined surface machining was commanded during the mode where the machining is unavailable.
Inclined surface machining was commanded during interruption.

Remedy

- Check the program and see whether any unavailable mode is included during inclined surface machining command. If any, cancel that mode.

P953 Tool axis dir cntrl prohibited

Details

Tool axis direction control was commanded during the mode where the control is unavailable.

Remedy

- Check the program and see whether any unavailable mode is included during tool axis direction control. If any, cancel that mode.

P954 Inclined face command error

Details

The address to issue the inclined surface machining command is incorrect.

Remedy

- Check the program.

P955 Inclined face coord illegal

Details

Impossible to define an inclined surface with the values you specified.

Remedy

- Check the program.

P956 G68.2P10 surface not defined

Details

The coordinate system for the machining surface selected with G68.2P10 has not been defined.

Remedy

- Set the machining surface so that the coordinate system can be defined.

P957 Tool axis dir ctrl cmp amt 0

Details

When the tool axis direction control type 2 (G53.6) was commanded, a tool length compensation No. whose compensation amount is 0 was commanded.

Remedy

- Correct the program. Set the tool length compensation amount, or command a tool length compensation No. whose compensation amount is not 0.

P960 No spec: Direct command mode

Details

G05 P4 was commanded while direct command mode option is OFF.

Remedy

- Check the specifications.

P961 Invalid during dir cmnd mode

Details

- A G code other than G05 P0 was commanded in direct command mode.
- A sequence No. command, F code command, MSTB command or variable command was issued.
- A corner chamfering command or corner R command was issued.
- A travel command was issued to an axis that had not been command in the G05 P4 block.

Remedy

- Check the program.

P962 Dir cmnd mode cmnd invalid

Details

G05 P4 was commanded in a modal where direct command mode is not available.

Remedy

- Check the program.

P963 Illegal direct cmnd mode cmnd

Details

The commanded coordinate value was beyond the maximum travel distance in direct command mode.

Remedy

- Correct the coordinate value in direct command mode.

P990 PREPRO error

Details

Combining commands that required pre-reading (nose R offset, corner chamfering/corner rounding, geometric I, geometric IB, and compound type fixed cycle for turning machining) resulted in eight or more pre-read blocks.

Remedy

- Delete some or all of the combinations of commands that require pre-reading.

Appendix 2

Order of G Function Command Priority

When commanded in the same block, ○ indicates that the both commands are executed simultaneously.
Command in a separate block when possible.

Commanded G code	**G group**			
	04 **G22, G23**	**07** **G40 to G42** **G46**	**09** **G70 to G79** **G80** **G81 to G89**	**11** **G50.2, G51.2**
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. G80 is executed. G81 to G89 are ignored. (Note)	G04 is executed. G50.2 and G51.2 are ignored. (Note)
G09 Exact stop check	○	○	G09 is executed. G70 to G79 are ignored. G80 is executed. G81 to G89 are executed. (Note)	○
G10, G11 Program 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. G80 is executed. G81 to G89 are ignored. (Note)	○
(System 6 and 7) G22, G23 Soft limit	○	○	G22 and G23 are executed. G70 to G79 are ignored. G80 is executed. G81 to G89 are executed. (Note)	○
G27 to G30 Reference position check/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. G80 is executed. G81 to G89 are ignored. (Note)	G27 to G30 are executed. G50.2 and G51.2 are ignored. (Note)
G30.1 to G30.6 Tool change position return	○	○	G30.1 to G30.6 are executed. G70 to G79 are ignored. G80 is executed. G81 to G89 are ignored. (Note)	○
G31, G31.1 to G31.3 Skip/multi-step skip	○	○	G31, G31.1 to G31.5 are executed. G70 to G79 are ignored. G80 is executed. G81 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. G80 is executed. G81 to G89 are ignored. (Note)	G37 is executed. G50.2 and G51.2 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. G80 is executed. G81 to G89 are executed. (Note)	G52 is executed. G50.2 and G51.2 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. G80 is executed. G81 to G89 are ignored. (Note)	G53 is executed. G50.2 and G51.2 are ignored. (Note)
G65 User macro calls	○	○	G65 is executed. G70 to G79 are ignored. G80 is executed. G81 to G89 are executed. (Note)	○

Commanded G code	**G group**			
	04 **G22, G23**	**07** **G40 to G42** **G46**	**09** **G70 to G79** **G80** **G81 to G89**	**11** **G50.2, G51.2**
G92 Spindle clamp speed setting/ coordinate system setting	○	○	G92 is executed. G70 to G79 are ignored. G80 is executed. G81 to G89 are ignored. (Note)	○
G92.1(G50.3) Workpiece coordinate system preset	○	○	G92.1 is executed. G70 to G79 are ignored. G80 is executed. G81 to G89 are ignored. (Note)	○

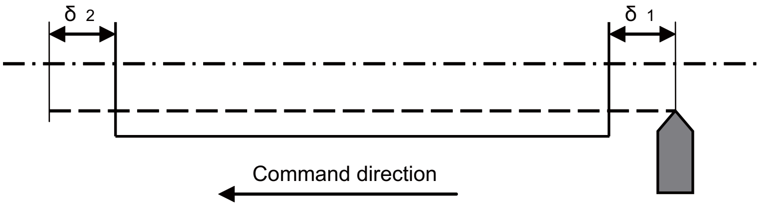
(Note) A program error (P45) will occur if they are commanded in the same block.
This error can be avoided by setting the parameter "#1241 bit0 (No G-CODE COMB. Error)" to "1", but be aware that one of the G commands is ignored.

Appendix 3

Supplementary Explanation for Incomplete Thread Area of Thread Control

Function and purpose

The delay caused by the automatic acceleration/deceleration and delay caused by the position loop in the servo system create an illegal pitch near the start and end points of thread cutting. When programming steps must be taken to assign thread cutting commands which include a margin for the approach distance 1 and for the length of the area 2 where the thread is incomplete during chamfering, as shown in the figure below.



δ1 : Approach distance

δ2 : Area where thread is incomplete during chamfering

Detailed description

Approach distance [δ 1]

(1) When Ts is not equal to zero

$$\delta_1 = \frac{F}{60} t1 - \frac{F}{60} (Ts + Tp - \frac{Tp^2 e^{-\frac{t1}{Tp}} - Ts^2 e^{-\frac{t1}{Tp}}}{Tp - Ts}) \text{ (mm)}$$

Where F : Thread cutting speed (mm/min)

t1 : Time taken until pitch error reaches allowable limit "a" (s)

Ts : Acceleration/deceleration time constant (s)

Tp : Position loop time constant (s) = 1/Position loop gain(PGN1)

During the SHG control, a position loop gain should be calculated by multiplying $\sqrt{\ }$ 2.

If "p" is the pitch and "ΔP" is the pitch error, then allowable limit "a" will be:

$$a = \frac{1}{Tp - Ts} (Tp\, e^{-\frac{t1}{Tp}} - Ts\, e^{-\frac{t1}{Ts}})$$

(2) When Ts is equal to zero

$$\delta_1 = \frac{F}{60} t1 - \frac{F}{60} \left(Tp - Tp\, e^{\frac{T1}{Tp}} \right) \quad (mm)$$

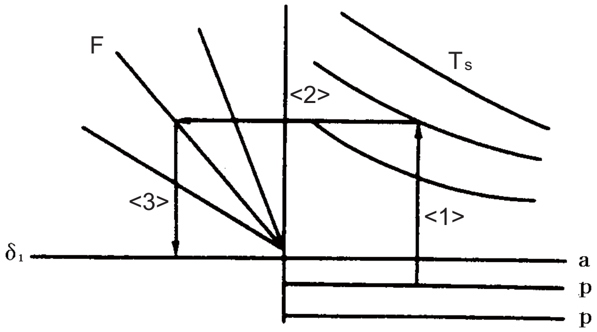
$$a = e^{-\frac{t1}{Tp}}$$

Since the calculation of approach distance δ1 is a complicated procedure, δ1 is normally determined from the chart on the next page. This chart is used as follows.

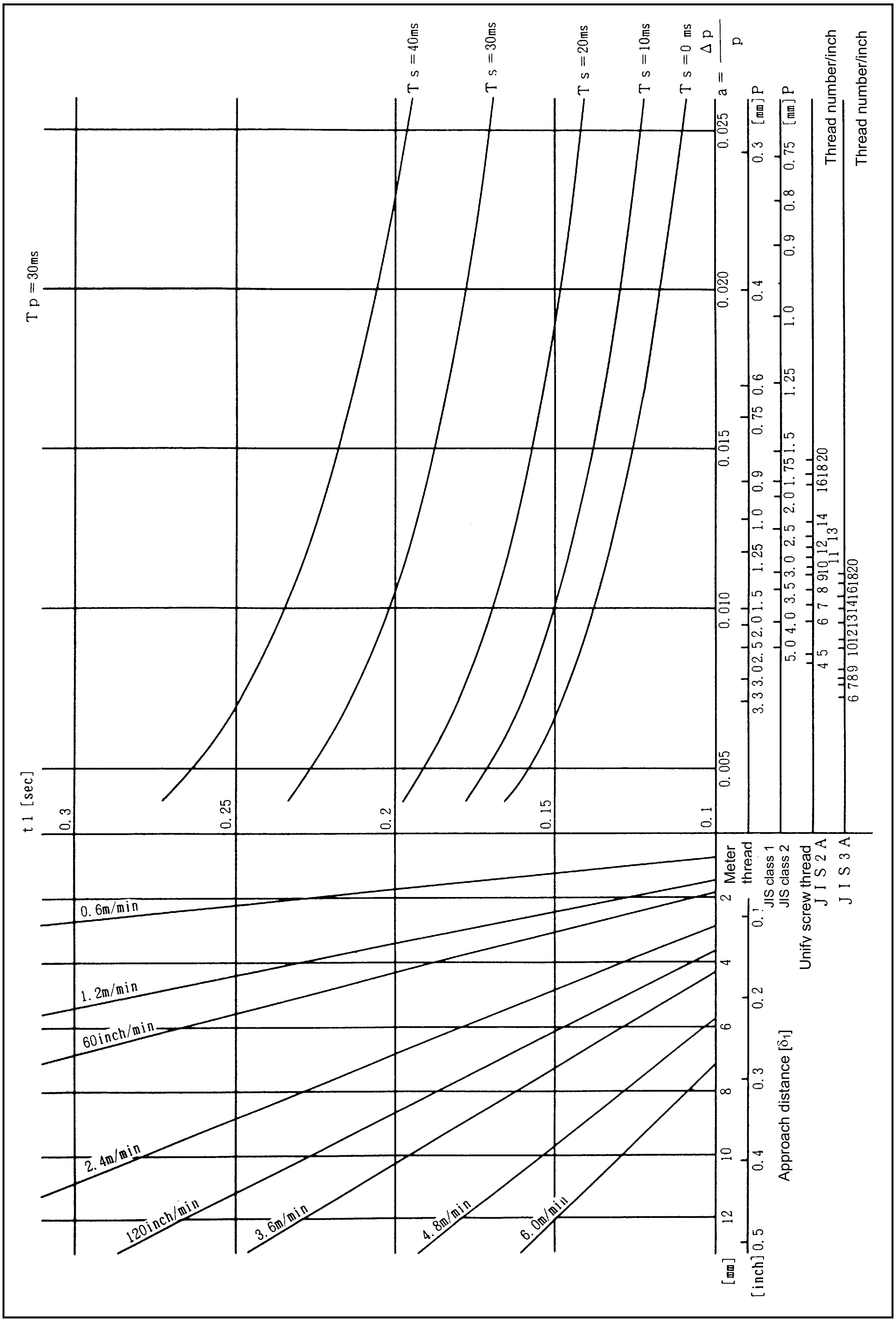
(a) Find the position on the p axis scale represents by the thread grade and pitch [P], and follow the perpendicular <1> drawn upward to find the point where it intersects with the curve of acceleration/ deceleration time constant [Ts].

(b) Follow horizontal line <2> and find where it intersects with the thread cutting speed [F].

(c) Follow perpendicular <3> and find approach distance [δ1] on the scale at the point where it intersects with the δ1 axis.



Approach distance δ1 calculation chart (when position loop gain = 33)



Length of area where thread is incomplete during chamfering [δ 2]

$$\delta_2 = (Ts + Tp)\frac{F}{60} \quad \text{(mm)}$$

Where F : Thread cutting speed (mm/min)

Ts : Acceleration/deceleration time constant (s)

Tp : Position loop time constant (s) = 1/ position loop gain (PGN1)

During the SHG control, a position loop gain should be calculated by multiplying √ 2.

(Note) When proceeding with chamfering during a thread cutting cycle, the length of the area where the thread is incomplete is equivalent to the value produced by adding δ 2 determined by the above formula to the chamfering pitch set by parameter.

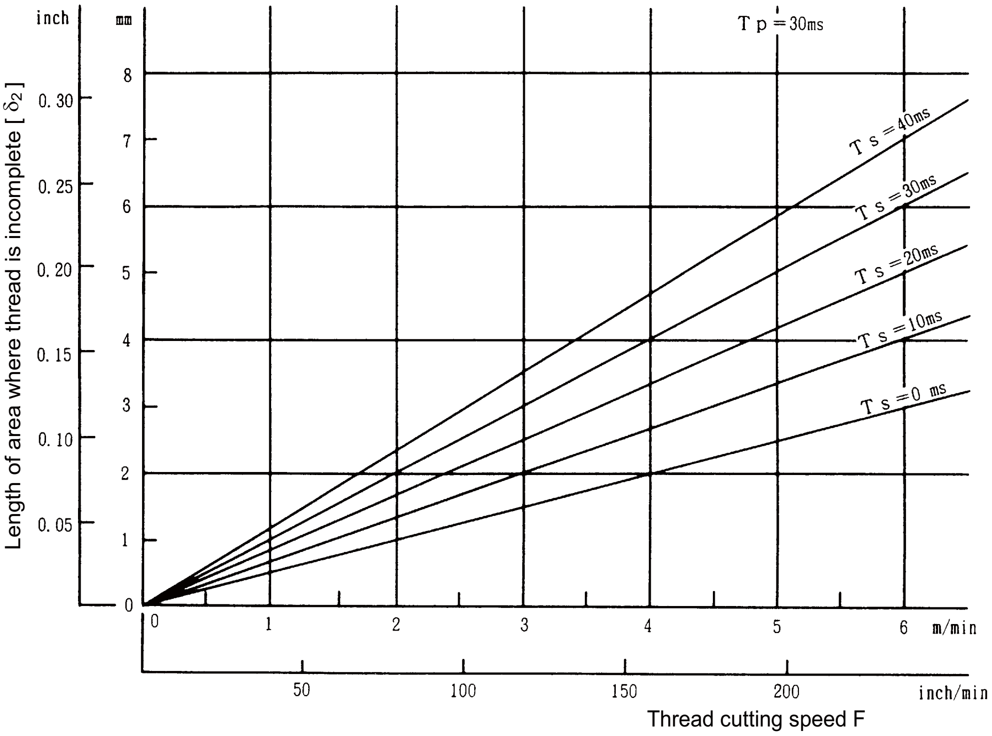


Chart to calculate length of area where thread is incomplete [δ 2]

Index

Revision History

Date of revision	Manual No.	Revision details
Mar. 2013	IB(NA)1501193-A	First edition created.
Mar. 2014	IB(NA)1501193-B	- Corrections and additions are made corresponding to S/W version K1. [Added] 13.4.2 Subprogram Call ; M198 Appendix 3. Supplementary explanation for incomplete thread area of thread control [Changed the chapter titles] 13.4 Subprogram Call ; M98,M99,M198 [Changed] 3.1 Program Format 6.6.1 Constant Lead Thread Cutting ; G33 6.6.2 Inch Thread Cutting ; G33 7.2 Cutting Feedrate 7.4 Feed Per Minute/Feed Per Revolution (Asynchronous Feed/Synchronous Feed) ; G94,G95 7.7 Automatic Acceleration/Deceleration 7.8 Rapid Traverse Constant Inclination Acceleration/Deceleration 12.4.5 Interrupts during Tool Nose Radius Compensation 12.4.7 Interference Check 12.5 Compensation Data Input by Program ; G10 L2/L10/L11, G11 12.6 Tool Life Management II ; G10 L3, G11 13.1.2 Thread Cutting Cycle ; G78 13.2.1 Longitudinal Rough Cutting Cycle ; G71 13.2.2 Face Rough Cutting Cycle ; G72 13.2.7 Compound Thread Cutting Cycle ; G76 13.3.1 Face Deep Hole Drilling Cycle 1 (Longitudinal deep hole drilling cycle 1) ; G83 (G87) 13.3.2 Face Tapping Cycle (Longitudinal tapping cycle) / Face Reverse Tapping Cycle (Longitudinal reverse tapping cycle) ; G84 (G88) / G84.1 (G88.1) 13.3.3 Face Boring Cycle (Longitudinal boring cycle) ; G85 (G89) 13.4.1 Subprogram Call ; M98,M99 13.6.2.1 Simple Macro Calls ; G65 13.6.2.2 Modal Call A (Movement Command Call) ; G66 13.6.2.3 Modal Call B (for each block) ; G66.1 13.6.2.5 Miscellaneous Command Macro Call (for M, S, T, B Code Macro Call) 13.6.5.6 NC Alarm (#3000) 13.6.5.12 Other Modals (#4101 - #4120, #4301 - #4320) 13.11 Programmable Parameter Input ; G10 L70, G11 13.12 Macro Interruption ; M96,M97 14.10 Workpiece Coordinate System Setting and Offset ; G54 to G59 (G54.1) 16.5 Programmable Current Limitation ; G10 L14 ; - Corrected the mistakes.

Global Service Network

AMERICA

MITSUBISHI ELECTRIC AUTOMATION INC. (AMERICA FA CENTER)

Central Region Service Center
500 CORPORATE WOODS PARKWAY, VERNON HILLS, ILLINOIS 60061, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

Michigan Service Satellite
ALLEGAN, MICHIGAN 49010, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

Ohio Service Satellite
LIMA, OHIO 45801, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650
CINCINATTI, OHIO 45201, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

Minnesota Service Satellite
ROGERS, MINNESOTA 55374, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650

West Region Service Center
16900 VALLEY VIEW AVE., LAMIRADA, CALIFORNIA 90638, U.S.A.
TEL: +1-714-699-2625 / FAX: +1-847-478-2650

Northern CA Satellite
SARATOGA, CALIFORNIA 95070, U.S.A.
TEL: +1-714-699-2625 / FAX: +1-847-478-2650

Pennsylvania Service Satellite
PITTSBURG, PENNSYLVANIA 15644, U.S.A.
TEL: +1-732-560-4500 / FAX: +1-732-560-4531

Connecticut Service Satellite
TORRINGTON, CONNECTICUT 06790, U.S.A.
TEL: +1-732-560-4500 / FAX: +1-732-560-4531

South Region Service Center
1845 SATTELITE BOULEVARD STE. 450, DULUTH, GEORGIA 30097, U.S.A.
TEL +1-678-258-4529 / FAX +1-678-258-4519

Texas Service Satellites
GRAPEVINE, TEXAS 76051, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519
HOUSTON, TEXAS 77001, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Tennessee Service Satellite
Nashville, Tennessee, 37201, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Florida Service Satellite
WEST MELBOURNE, FLORIDA 32904, U.S.A.
TEL: +1-678-258-4529 / FAX: +1-678-258-4519

Canada Region Service Center
4299 14TH AVENUE MARKHAM, ONTARIO L3R OJ2, CANADA
TEL: +1-905-475-7728 / FAX: +1-905-475-7935

Canada Service Satellite
EDMONTON, ALBERTA T5A 0A1, CANADA
TEL: +1-905-475-7728 FAX: +1-905-475-7935

Mexico Region Service Center
MARIANO ESCOBEDO 69 TLALNEPANTLA, 54030 EDO. DE MEXICO
TEL: +52-55-3067-7500 / FAX: +52-55-9171-7649

Monterrey Service Satellite
MONTERREY, N.L., 64720, MEXICO
TEL: +52-81-8365-4171

BRAZIL

MELCO CNC do Brasil Comércio e Serviços S.A

Brazil Region Service Center
ACESSO JOSE SARTORELLI, KM 2.1 CEP 18550-000, BOITUVA-SP, BRAZIL
TEL: +55-15-3363-9900 / FAX: +55-15-3363-9911

EUROPE

MITSUBISHI ELECTRIC EUROPE B.V.

GOTHAER STRASSE 10, 40880 RATINGEN, GERMANY
TEL: +49-2102-486-0 / FAX: +49-2102-486-5910

Germany Service Center
KURZE STRASSE. 40, 70794 FILDERSTADT-BONLANDEN, GERMANY
TEL: + 49-711-770598-123 / FAX: +49-711-770598-141

France Service Center DEPARTEMENT CONTROLE NUMERIQUE
25, BOULEVARD DES BOUVETS, 92741 NANTERRE CEDEX FRANCE
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

France (Lyon) Service Satellite DEPARTEMENT CONTROLE NUMERIQUE
120, ALLEE JACQUES MONOD 69800 SAINT PRIEST FRANCE
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

Italy Service Center
VIALE COLLEONI, 7 - CENTRO DIREZIONALE COLLEONI PALAZZO SIRIO INGRESSO 1
20864 AGRATE BRIANZA (MB), ITALY
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

Italy (Padova) Service Satellite
VIA G. SAVELLI, 24 - 35129 PADOVA, ITALY
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

U.K. Branch
TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.
TEL: +49-2102-486-0 / FAX: +49-2102-486-5910

Spain Service Center
CTRA. DE RUBI, 76-80-APDO. 420
08173 SAINT CUGAT DEL VALLES, BARCELONA SPAIN
TEL: +34-935-65-2236 / FAX: +34-935-89-1579

Poland Service Center
UL.KRAKOWSKA 50, 32-083 BALICE, POLAND
TEL: +48-12-630-4700 / FAX: +48-12-630-4701

Mitsubishi Electric Turkey A.Ş Ümraniye Şubesi
Turkey Service Center
ŞERIFALI MAH. NUTUK SOK. NO.5 34775
ÜMRANIYE, ISTANBUL, TURKEY
TEL: +90-216-526-3990 / FAX: +90-216-526-3995

Czech Republic Service Center
KAFKOVA 1853/3, 702 00 OSTRAVA 2, CZECH REPUBLIC
TEL: +420-59-5691-185 / FAX: +420-59-5691-199

Russia Service Center
213, B.NOVODMITROVSKAYA STR., 14/2, 127015 MOSCOW, RUSSIA
TEL: +7-495-748-0191 / FAX: +7-495-748-0192

MITSUBISHI ELECTRIC EUROPE B.V. (SCANDINAVIA)
Sweden Service Center
HAMMARBACKEN 14 191 49 SOLLENTUNA, SWEDEN
TEL: +46-8-6251000 / FAX: +46-8-966877

Bulgaria Service Center
4 A.LYAPCHEV BOUL., POB 21, BG-1756 SOFIA, BULGARIA
TEL: +359-2-8176009 / FAX: +359-2-9744061

Ukraine (Kharkov) Service Center
APTEKARSKIY LANE 9-A, OFFICE 3, 61001 KHARKOV, UKRAINE
TEL: +380-57-732-7774 / FAX: +380-57-731-8721

Ukraine (Kiev) Service Center
4-B, M. RASKOVOYI STR., 02660 KIEV, UKRAINE
TEL: +380-44-494-3355 / FAX: +380-44-494-3366

Belarus Service Center
OFFICE 9, NEZAVISIMOSTI PR.177, 220125 MINSK, BELARUS
TEL: +375-17-393-1177 / FAX: +375-17-393-0081

South Africa Service Center
5 ALBATROSS STREET, RHODESFIELD, KEMPTON PARK 1619, GAUTENG, SOUTH AFRICA
TEL: +27-11-394-8512 / FAX: +27-11-394-8513

ASEAN

MITSUBISHI ELECTRIC ASIA PTE. LTD. (ASEAN FA CENTER)

Singapore Service Center
307 ALEXANDRA ROAD #05-01/02 MITSUBISHI ELECTRIC BUILDING SINGAPORE 159943
TEL: +65-6473-2308 / FAX: +65-6476-7439

Malaysia (KL) Service Center
60, JALAN USJ 10 /1B 47620 UEP SUBANG JAYA SELANGOR DARUL EHSAN, MALAYSIA
TEL: +60-3-5631-7605 / FAX: +60-3-5631-7636

Malaysia (Johor Baru) Service Center
17 & 17A, JALAN IMPIAN EMAS 5/5, TAMAN IMPIAN EMAS, 81300 SKUDAI, JOHOR MALAYSIA.
TEL: +60-7-557-8218 / FAX: +60-7-557-3404

Philippines Service Center
UNIT NO.411, 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

VIETNAM

MITSUBISHI ELECTRIC VIETNAM CO.,LTD

Vietnam (Ho Chi Minh) Service Center
UNIT 01-04, 10TH FLOOR, VINCOM CENTER 72 LE THANH TON STREET, DISTRICT 1,
HO CHI MINH CITY, VIETNAM
TEL: +84-8-3910 5945 / FAX: +84-8-3910 5946

Vietnam (Hanoi) Service Satellite
SUITE 9-05, 9TH FLOOR, HANOI CENTRAL OFFICE BUILDING, 44B LY THUONG KIET STREET,
HOAN KIEM DISTRICT, HANOI CITY, VIETNAM
TEL: +84-4-3937-8075 / FAX: +84-4-3937-8076

INDONESIA

PT. MITSUBISHI ELECTRIC INDONESIA

Indonesia Service Center
GEDUNG JAYA 11TH FLOOR, JL. MH. THAMRIN NO.12, JAKARTA PUSAT 10340, INDONESIA
TEL: +62-21-3192-6461 / FAX: +62-21-3192-3942

THAILAND

MITSUBISHI ELECTRIC FACTORY AUTOMATION (THAILAND) CO.,LTD

Thailand Service Center
12TH FLOOR, SV.CITY BUILDING, OFFICE TOWER 1, NO. 896/19 AND 20 RAMA 3 ROAD,
KWAENG BANGPONGPANG, KHET YANNAWA, BANGKOK 10120,THAILAND
TEL: +66-2-682-6522-31 / FAX: +66-2-682-6020

INDIA

MITSUBISHI ELECTRIC INDIA PVT. LTD.

India Service Center
2nd FLOOR, TOWER A & B, DLF CYBER GREENS, DLF CYBER CITY,
DLF PHASE-III, GURGAON 122 002, HARYANA, INDIA
TEL: +91-124-4630 300 / FAX: +91-124-4630 399
Ludhiana satellite office
Jamshedpur satellite office

India (Pune) Service Center
EMERALD HOUSE, EL-3, J-BLOCK, MIDC BHOSARI. PUNE – 411 026, MAHARASHTRA, INDIA
TEL: +91-20-2710 2000 / FAX: +91-20-2710 2100
Baroda satellite office
Mumbai satellite office

India (Bangalore) Service Center
PRESTIGE EMERALD, 6TH FLOOR, MUNICIPAL NO. 2,
LAVELLE ROAD, BANGALORE - 560 043, KAMATAKA, INDIA
TEL: +91-80-4020-1600 / FAX: +91-80-4020-1699
Chennai satellite office
Coimbatore satellite office

OCEANIA

MITSUBISHI ELECTRIC AUSTRALIA LTD.

Australia Service Center
348 VICTORIA ROAD, RYDALMERE, N.S.W. 2116 AUSTRALIA
TEL: +61-2-9684-7269 / FAX: +61-2-9684-7245

CHINA

MITSUBISHI ELECTRIC AUTOMATION (CHINA) LTD. (CHINA FA CENTER)

China (Shanghai) Service Center
1-3,5-10,18-23/F, NO.1386 HONG QIAO ROAD, CHANG NING QU,
SHANGHAI 200336, CHINA
TEL: +86-21-2322-3030 / FAX: +86-21-2308-3000
China (Ningbo) Service Dealer
China (Wuxi) Service Dealer
China (Jinan) Service Dealer
China (Hangzhou) Service Dealer
China (Wuhan) Service Satellite

China (Beijing) Service Center
9/F, OFFICE TOWER 1, HENDERSON CENTER, 18 JIANGUOMENNEI DAJIE,
DONGCHENG DISTRICT, BEIJING 100005, CHINA
TEL: +86-10-6518-8830 / FAX: +86-10-6518-8030
China (Beijing) Service Dealer

China (Tianjin) Service Center
UNIT 2003, TIANJIN CITY TOWER, NO 35 YOUYI ROAD, HEXI DISTRICT,
TIANJIN 300061, CHINA
TEL: +86-22-2813-1015 / FAX: +86-22-2813-1017
China (Shenyang) Service Satellite
China (Changchun) Service Satellite

China (Chengdu) Service Center
ROOM 407-408, OFFICE TOWER AT SHANGRI-LA CENTER, NO. 9 BINJIANG DONG ROAD,
JINJIANG DISTRICT, CHENGDU, SICHUAN 610021, CHINA
TEL: +86-28-8446-8030 / FAX: +86-28-8446-8630

China (Shenzhen) Service Center
ROOM 2512-2516, 25/F., GREAT CHINA INTERNATIONAL EXCHANGE SQUARE, JINTIAN RD.S.,
FUTIAN DISTRICT, SHENZHEN 518034, CHINA
TEL: +86-755-2399-8272 / FAX: +86-755-8218-4776
China (Xiamen) Service Dealer
China (Dongguan) Service Dealer

KOREA

MITSUBISHI ELECTRIC AUTOMATION KOREA CO., LTD. (KOREA FA CENTER)

Korea Service Center
1480-6, GAYANG-DONG, GANGSEO-GU, SEOUL 157-200, KOREA
TEL: +82-2-3660-9602 / FAX: +82-2-3664-8668

Korea Taegu Service Satellite
4F KT BUILDING, 1630 SANGYEOK-DONG, BUK-KU, DAEGU 702-835, KOREA
TEL: +82-53-382-7400 / FAX: +82-53-382-7411

TAIWAN

MITSUBISHI ELECTRIC TAIWAN CO., LTD. (TAIWAN FA CENTER)

Taiwan (Taichung) Service Center (Central Area)
NO.8-1, INDUSTRIAL 16TH RD., TAICHUNG INDUSTRIAL PARK, SITUN DIST.,
TAICHUNG CITY 40768, TAIWAN R.O.C.
TEL: +886-4-2359-0688 / FAX: +886-4-2359-0689

Taiwan (Taipei) Service Center (North Area)
10F, NO.88, SEC.6, CHUNG-SHAN N. RD., SHI LIN DIST., TAIPEI CITY 11155, TAIWAN R.O.C.
TEL: +886-2-2833-5430 / FAX: +886-2-2833-5433

Taiwan (Tainan) Service Center (South Area)
11F-1., NO.30, ZHONGZHENG S. ROAD, YONGKANG DISTRICT, TAINAN CITY 71067, TAIWAN, R.O.C
TEL: +886-6-252-5030 / FAX: +886-6-252-5031

Notice

Every effort has been made to keep up with software and hardware revisions in the contents described in this manual. However, please understand that in some unavoidable cases simultaneous revision is not possible. Please contact your Mitsubishi Electric dealer with any questions or comments regarding the use of this product.

Duplication Prohibited

MITSUBISHI CNC

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE : TOKYO BUILDING,2-7-3 MARUNOUCHI,CHIYODA-KU,TOKYO 100-8310,JAPAN

MODEL	E70 Series
MODEL CODE	100-340
Manual No.	IB-1501193

IB(NA)1501193(ENG)-B(1403)MEE
Made in JAPAN

Specifications are subject to change without notice.