MITSUBISHI ELECTRIC

Changes for the Better

MITSUBISHI CNC

Programming Manual Machining Center System (1/2)

M800/M80/C80 Series

IB-1501277(ENG)-D

Introduction

This manual describes how to carry out MITSUBISHI CNC programming.
Supported models are as follows:

Supported models	Abbreviations in this manual
M800W series	M800 series, M800, M8
M800S series	
M80W series	M80 series, M80, M8
M80 series	
C80 series	C80

This manual describes programming, therefore, read this manual thoroughly before using this NC system.
To ensure safe use of this NC system, thoroughly study the "Precautions for Safety" on the following page before using this NC system.
Be sure to always keep this manual on hand so that users can refer to it at any time.

Details described in this manual

The description concerning "Signals" in the main text refers to information transmission between a machine and PLC or between NC and PLC.
The method for controlling the signals (ON/OFF) differs depending on the machine. Refer to the manual issued by the machine tool builder (MTB).
Some parameters can be used by end-users and some parameters are set by the MTB according to the specifications.
End-users may not be able to set or change some of the parameters described as "... can be set with the parameter #XXXX" in the main text. Confirm the specifications for your machine with the manual issued by the MTB.

⚠ CAUTION

⚠ For items described as "Restrictions" or "Usable State" in this manual, the instruction manual issued by the machine tool builder (MTB) 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 the applicable functions are included. Some of them, however, may not be available for your NC system. Refer to the specifications issued by the machine tool builder before use.

⚠ Refer to the Instruction Manual issued by the MTB for details regarding each machine tool.

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

General precautions
(1) Refer to the following documents for details handling

MITSUBISHI CNC M800/M80 Series Instruction Manual IB-1501274
MITSUBISHI CNC C80 Series Instruction Manual IB-1501453

(2) Refer to the following documents for details on programming

MITSUBISHI CNC M800/M80/C80 Series Programming Manual

Lathe System (1/2) IB-1501275
Lathe System (2/2) IB-1501276
Machining Center System (1/2) IB-1501277
Machining Center System (2/2) IB-1501278

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 the applicable functions are included. Some of them, however, may not be available for your NC system.
Refer to the specifications issued by the machine tool builder before use.

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

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

2. Items related to operation

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

⚠ If the workpiece coordinate system offset amount is changed during single block stop, the new setting will be valid from the next block.

⚠ Turn the mirror image ON and OFF at the mirror image center.

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

⚠ Do not make the synchronized spindle rotation command OFF with one workpiece chucked by the reference spindle and synchronized spindle during the spindle synchronization.
Failure to observe this may cause the synchronized spindle stop, and hazardous situation.

3. Items related to programming

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

⚠ ";" "EOB" and "%" "EOR" are expressions used for explanation. The actual codes are: For ISO: "CR, LF", or "LF" and "%".
Programs created on the Edit screen are stored in the NC memory in a "CR, LF" format, but programs created with external devices such as the FLD or RS-232C may be stored in an "LF" format.
The actual codes for EIA are: "EOB (End of Block)" and "EOR (End of Record)".

⚠ When creating the machining program, select the appropriate machining conditions, and make sure that the performance, capacity and limits of the machine and NC are not exceeded. The examples do not consider the machining conditions.

⃠ Do not change fixed cycle programs without the prior approval of the machine tool builder.

⚠ When programming the multi-part system, take special care to the movements of the programs for other part systems.

Disposal

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

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

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

Trademarks

MELDAS, MELSEC, EZSocket, EZMotion, iQ Platform, MELSEC iQ-R, 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®, Windows®, SQL Server® and Access® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.
SD logo and SDHC logo are either registered trademarks or trademarks of LLC.
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.
MODBUS® is either a trademark or a registered trademark of Schneider Electric USA, Inc. or the affiliated companies in Japan and/or other countries.
EtherNet/IP is a trademark of Open DeviceNet Vendor Association,Inc.
PROFIBUS-DP is a trademark of Profibus International.
Oracle® is a registered trademark of Oracle Corporation, the subsidiaries, or the affiliated companies 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

Chapter 1 - 14 : Refer to Programming Manual (Machining Center System) (1/2)
Chapter 15 and later : Refer to Programming Manual (Machining Center System) (2/2)

1

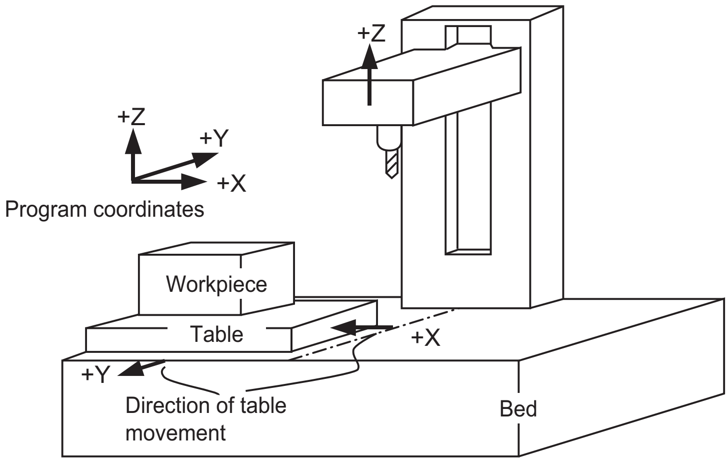
Control Axes

1.1 Coordinate Words and Control Axes

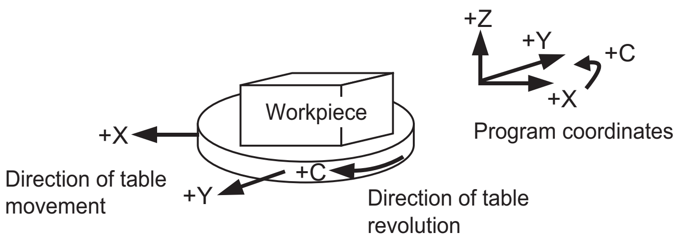
Function and purpose

The number of control axes is set to "3" in the standard specifications; however, up to eight axes can be controlled if an additional axis is added. To specify each machining direction, use alphabetical coordinate words that are pre-defined appropriately.

X-Y table



X-Y and rotating table



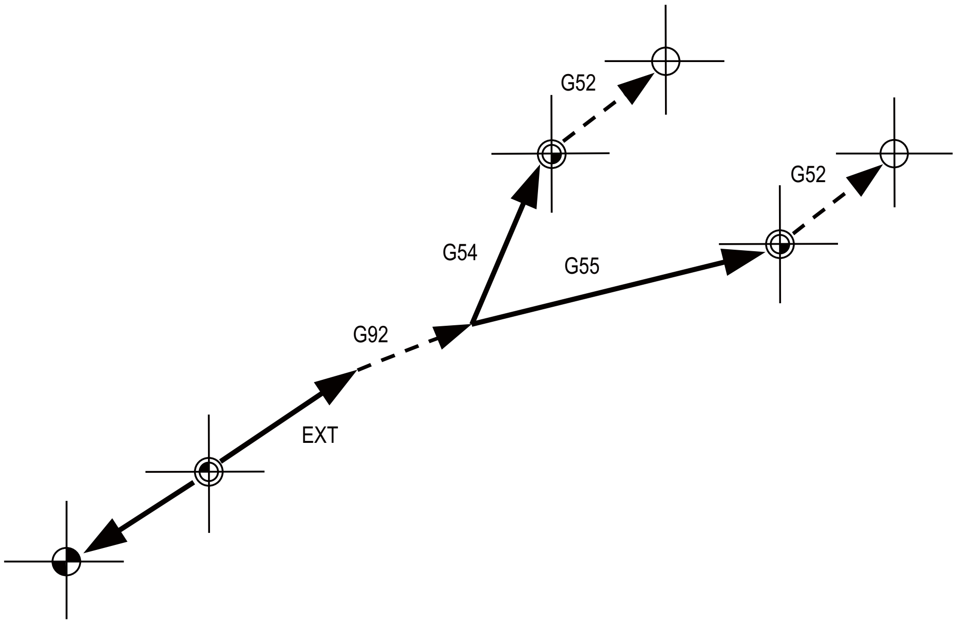
1.2 Coordinate Systems and Coordinate Zero Point Symbols

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



Reference position

Basic machine coordinate zero point

Workpiece coordinate zero points

Local coordinate zero point

Offset set by a parameter

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

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

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

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

2

Minimum Command Unit

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. (This depends on the MTB specifications.)

	Parameter		Linear axis		Rotary axis (°)
			Metric	Inch	
Input setting unit	#1003 iunit	= B	0.001	0.0001	0.001
		= C	0.0001	0.00001	0.0001
		= D	0.00001	0.000001	0.00001
		= E	0.000001	0.0000001	0.000001
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.)

2.2 Input Command Increment Tenfold

Function and purpose

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

Detailed description

(1) When running a machining program already created with a 10µm input command increment with a CNC unit for which the command increment is set to 1µm and this function's parameter value is set to "10", this function enables the same machining as the original program.

(2) When running a machining program already created with a 1µm input command increment with a CNC unit for which the command increment is set to 0.1µm and this function's parameter value is set to "10", this function enables the same machining as the original program.

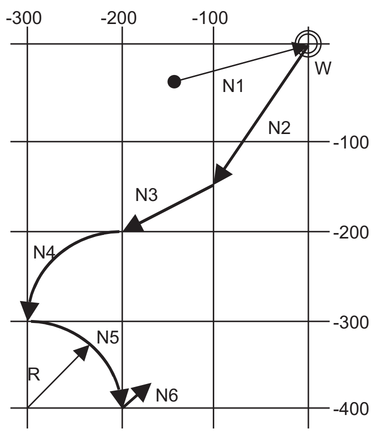
(3) This function cannot be used for the dwell function G04_X_(P_);.

(4) This function cannot be used for the compensation amount of the tool compensation input.

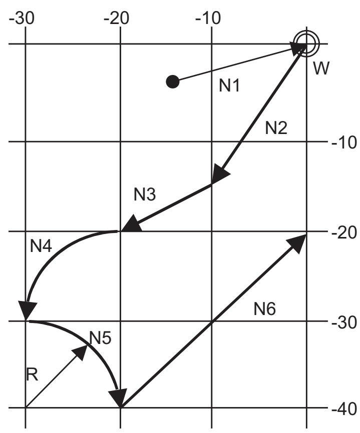
(5) This function can be used when decimal point type I is valid, but cannot be used when decimal point type II is valid.

(6) This function cannot be used for a tool shape setting command (in G10L100 format).

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



UNIT*10 ON



UNIT*10 OFF

2.3 Indexing Increment

Function and purpose

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

Detailed description

When the indexing increment (parameter) which limits the command value is set, the rotary axis can only be positioned with that indexing increment. If a program other than the indexing increment setting value is commanded, a program error (P20) will occur.
The indexing position will not be checked when the parameter is set to 0.

(Example) When the indexing increment setting value is 2 degrees, the machine coordinate position at the end point can only be commanded with the 2-degree increment.
G90 G01 C102.000 ; ... Moves to the 102 degree angle.
G90 G01 C101.000; Program error
G90 G01 C102 ; ... Moves to the 102 degree angle. (Decimal point type II)

The following axis specification parameter is used. (This depends on the MTB specifications.)

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

Precautions

(1) When the indexing increment is set, positioning will be conducted in degree unit.

(2) The indexing position is checked with the rotary axis, and is not checked with other axes.

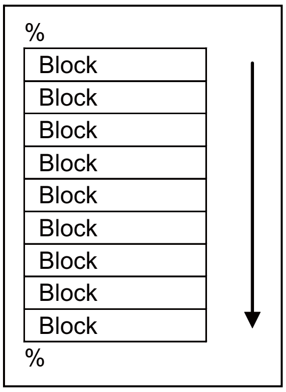
(3) When the indexing increment is set to 2 degrees, the rotary axis is set to the B axis, and the B axis is moved with JOG to the 1.234 position, an indexing error will occur if "G90B5." or "G91B2." is commanded.

3

Program Formats

3.1 Program Format

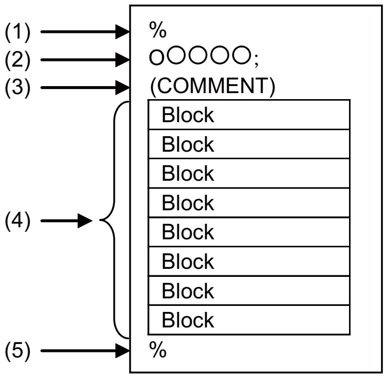
A collection of commands assigned to an NC to move a machine is called "program".
A program is a collection of units called "block" which specifies a sequence of machine tool operations.
Blocks are written in the order of the actual movement of a tool.
A block is a collection of units called "word" which 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.

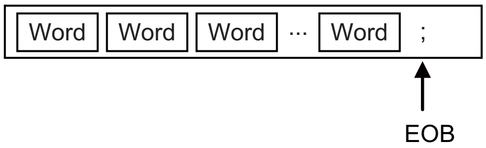
(3) Comment
Data between control out "(" and control in ")" is ignored.
Information including program names and comments can be written in.

(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.

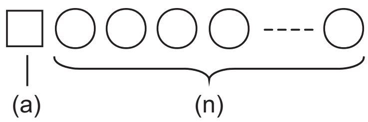
Block and word

[Block]



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

[Word]



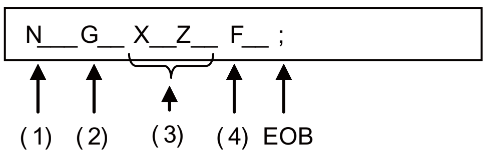
(a) Alphabet (address)
(n) Number

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

(1) Leading zeros can be omitted from numerals.

The major contents of a word are described below.



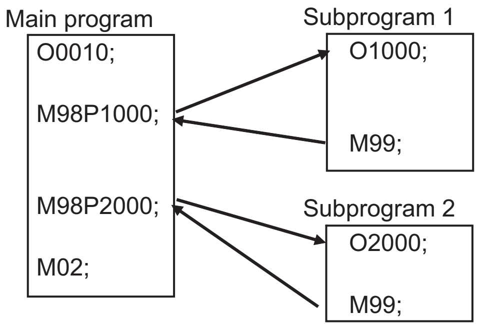
(1) Sequence No.
A "sequence No." consists of the address "N" followed by numbers of up to 8 digits (Normally 3 or 4 digits). It is used as an index when searching a necessary block in a program (as branch destination and etc.).
It does not affect the operation of a tool machine.

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

(3) Coordinate words
"Coordinate words" specify the coordinate positions and movement amounts of machine tool 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 "14.1 Subprogram Control; M98, M99, M198" for details of subprogram execution.

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. Hard discs stored in NC unit are regarded as 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.

External data input/output interface	M800W	M800S	M80	C80
NC memory	○	○	○	○
Serial	○	○	○	-
SD card in control unit	○	-	-	-
Front-side SD card	○	○	○	○ (*1)
Ethernet	○	○	○	○
Display unit-side data server	○	○	○	-
Front-side USB memory	○	○	○	○ (*2)

(*1) GOT back-side SD card

(*2) GOT front-side USB memory

Program file format

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

(1) NC memory (Creates program on NC)

```
(COMMENT) ;
G28 XYZ ;
   :
   :
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 SD card and 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 (29 characters for a multi-part system program).
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 (29 characters for a multi-part system program).
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, a section of a machining program from a "/" to the end of the block are ignored. They are executed if the switch is OFF.
Parity check is valid regardless of whether the optional block skip switch is ON or OFF.
When, for instance, all blocks are to be executed for one workpiece but specific blocks are not to be executed for another workpiece, one machining program can be used to machine different parts by inserting the "/" into those specific blocks.

Program example

(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 whether or not the optional block skip signal state is 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 the parameter "#1274 ext10/bit4" is set to "0" and parameter "#1226 aux10/bit1" is set to "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 G01 X25. /Z25. ; NG (User macro, a division instruction; a program error results.)
/N20 G01 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 formula containing a slash code.

(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 number search.

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

3.3.2 Optional Block Skip Addition ; /n

Function and purpose

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

Detailed description

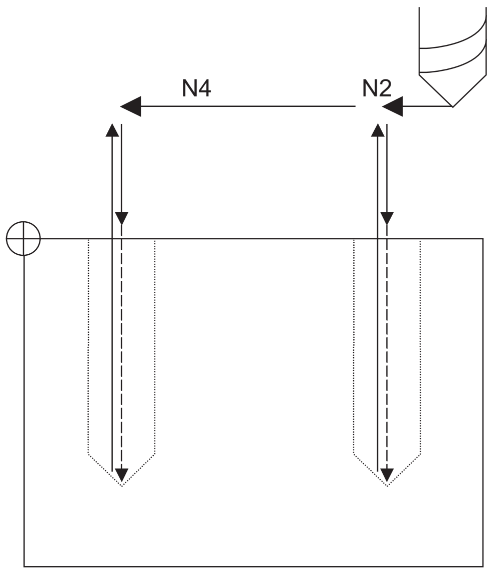
The block with "/n" (slash) code is skipped when the "/n" is programmed to the head of the block and the optional block skip n signal is turned ON. For a block with the "/n" code inside the block (not at the head of the block), the program is operated according to the value of the parameter "#1226 aux10/bit1"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

Program example

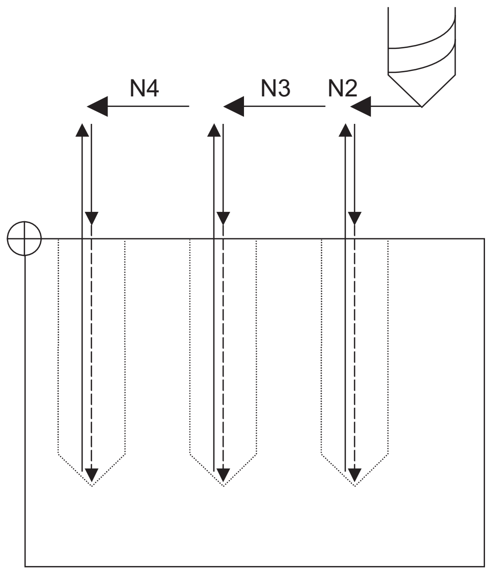
(1) When the 2 parts like the figure below are machined, the following program is used. When the optional block skip 5 signal is ON, the part 1 is created. When the optional block skip 5 signal is OFF, the part 2 is created.

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

Part 1
Optional block skip 5 signal ON



Part 2
Optional block skip 5 signal OFF



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

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

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

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

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

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

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

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

3.4 G Codes

3.4.1 Modal, Unmodal

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

3.4.2 G Code Lists

G code	Group	Function	Section
Δ 00	01	Positioning	6.1
0.5	28	Rapid traverse block overlap	7.13.1
Δ 01	01	Linear interpolation	6.2
02	01	Circular interpolation CW R-specified circular interpolation CW Helical interpolation CW Spiral/Conical interpolation CW (type2)	6.3 6.4 6.7 6.14
03	01	Circular interpolation CCW R-specified circular interpolation CCW Helical interpolation CCW Spiral/Conical interpolation CCW (type2)	6.3 6.4 6.7 6.14
02.1	01	Spiral/Conical interpolation CW (type1)	6.14
03.1	01	Spiral/Conical interpolation CCW (type1)	6.14
02.3	01	Exponential function interpolation positive rotation	6.12
03.3	01	Exponential function interpolation negative rotation	6.12
02.4	01	3-dimensional circular interpolation CW	6.15
03.4	01	3-dimensional circular interpolation CCW	6.15
04	00	Dwell (Time-based designation)	8.1
05	00	High-speed machining mode High-speed high-accuracy control II/III	17.1 17.3
05.1	00	High-speed high-accuracy control I Spline interpolation	17.3 17.4
06.2	01	NURBS interpolation	6.16
07	00	Hypothetical axis interpolation	6.17
07.1 107	21	Cylindrical interpolation	6.9
08	00	High-accuracy control	17.2
09	00	Exact stop check	7.10
10	00	Data input by program (Parameter input, Compensation input, Tool shape input, R-Navi data input) Tool life management data input	15.9 15.11
11	00	Data input by program cancel (Parameter input, Compensation input, Tool shape input, R-Navi data input) Tool life management data input	15.9 15.11
12	00	Circular cutting CW	6.10
13	00	Circular cutting CCW	6.10
12.1 112	21	Polar coordinate interpolation ON	6.11

G code	Group	Function	Section
* 13.1 113	21	Polar coordinate interpolation cancel	6.11
14			
* 15	18	Polar coordinate command OFF	6.13
16	18	Polar coordinate command ON	6.13
Δ 17	02	X-Y plane selection	6.5
Δ 18	02	Z-X plane selection	6.5
Δ 19	02	Y-Z plane selection	6.5
Δ 20	06	Inch command	5.2
Δ 21	06	Metric command	5.2
22	04	Stroke check before travel ON	20.1
23	04	Stroke check before travel cancel	20.1
24			
25			
26			
27	00	Reference position check	19.15
28	00	Reference position return	19.12
29	00	Start position return	19.12
30	00	2nd to 4th reference position return	19.13
30.1	00	Tool change position return 1	19.14
30.2	00	Tool change position return 2	19.14
30.3	00	Tool change position return 3	19.14
30.4	00	Tool change position return 4	19.14
30.5	00	Tool change position return 5	19.14
30.6	00	Tool change position return 6	19.14
31	00	Skip/Speed change skip Multi-step skip 2	21.2 21.4
31.1	00	Multi-step skip 1-1	21.3
31.2	00	Multi-step skip 1-2	21.3
31.3	00	Multi-step skip 1-3	21.3
32			
33	01	Thread cutting	6.6
34	00	Special fixed cycle (bolt hole circle)	13.2.1
35	00	Special fixed cycle (line at angle)	13.2.2
36	00	Special fixed cycle (arc)	13.2.3
37	00	Automatic tool length measurement	21.1
37.1	00	Special fixed cycle (grid)	13.2.4
38	00	Tool radius compensation vector designation	12.3
39	00	Tool radius compensation corner arc	12.3
* 40	07	Tool radius compensation (Tool nose radius compensation) cancel 3-dimensional tool radius compensation cancel 3-dimensional tool radius compensation (Tool's vertical-direction compensation) cancel	12.3 12.5 18.5
41	07	Tool radius compensation (Tool nose radius compensation) left 3-dimensional tool radius compensation left	12.3 12.5
42	07	Tool radius compensation (Tool nose radius compensation) right 3-dimensional tool radius compensation right	12.3 12.5
* 40.1 150	15	Normal line control cancel	15.7
41.1 151	15	Normal line control left ON	15.7

G code	Group	Function	Section
42.1 152	15	Normal line control right ON	15.7
41.2	07	3-dimensional tool radius compensation (Tool's vertical-direction compensation) (left)	18.5
42.2	07	3-dimensional tool radius compensation (Tool's vertical-direction compensation) (right)	18.5
43	08	Tool length compensation (+)	12.2
44	08	Tool length compensation (-)	12.2
43.1	08	Tool length compensation along the tool axis ON	
43.4	08	Tool center point control type1 ON	18.3
43.5	08	Tool center point control type2 ON	18.3
43.7	08	Tool position compensation start	18.1
45	00	Tool position offset (extension)	12.6
46	00	Tool position offset (reduction)	12.6
47	00	Tool position offset (double elongation)	12.6
48	00	Tool position offset (double contraction)	12.6
* 49	08	Tool length compensation cancel Tool length compensation along the tool axis Tool center point control cancel Tool position compensation cancel	12.2 18.1 18.3 18.1
* 50	11	Scaling cancel	19.11
51	11	Scaling ON	19.11
* 50.1	19	Mirror image by G code cancel	15.6
51.1	19	Mirror image by G code ON	15.6
52	00	Local coordinate system setting	19.5
53	00	Basic machine coordinate system selection	19.3
53.1	00	Tool axis direction control (type 1)	18.4.8
53.6	00	Tool axis direction control (type 2)	18.4.8
* 54	12	Workpiece coordinate system 1 selection	19.6
55	12	Workpiece coordinate system 2 selection	19.6
56	12	Workpiece coordinate system 3 selection	19.6
57	12	Workpiece coordinate system 4 selection	19.6
58	12	Workpiece coordinate system 5 selection	19.6
59	12	Workpiece coordinate system 6 selection	19.6
54.1	12	Extended workpiece coordinate system selection	19.6
54.4	27	Workpiece installation error compensation	-
60	00 (01)	Unidirectional positioning	6.8.1
61	13	Exact stop check mode	7.11
61.1	13	High-accuracy control ON	17.2
61.2	13	High-accuracy spline	17.6
61.4	13	Spline interpolation 2	17.5
62	13	Automatic corner override	7.14.1
63	13	Tapping mode	7.15
* 64	13	Cutting mode	7.16
65	00	User macro simple call	14.4.1
66	14	User macro modal call A	14.4.2
66.1	14	User macro modal call B	14.4.3
* 67	14	User macro modal call cancel	14.3
68	16	Coordinate rotation by program ON 3-dimensional coordinate conversion mode ON	19.9 19.8

G code	Group	Function	Section
68.2	16	Inclined surface machining command R-Navi data input (Selecting the registered machining surface)	18.4 15.9.5
68.3	16	Inclined surface machining command (Based on tool axis direction)	18.4
* 69	16	Coordinate rotation by program cancel 3-dimensional coordinate conversion mode OFF Inclined surface machining cancel R-Navi data input (Canceling the selected machining surface)	19.9 19.8 18.4 15.9.5
70	09	User fixed cycle	
71	09	User fixed cycle	
72	09	User fixed cycle	
73	09	Fixed cycle (step)	13.1.10
74	09	Fixed cycle (reverse tap)	13.1.11
75	09	Fixed cycle (circle cutting cycle)	13.1.12
76	09	Fixed cycle (Fine boring)	13.1.13
77	09	User fixed cycle	
78	09	User fixed cycle	
79	09	User fixed cycle	
* 80	09	Fixed cycle cancel	13.1
81	09	Fixed cycle (drill/spot drill)	13.1.1
82	09	Fixed cycle (drill/counter boring)	13.1.2
83	09	Fixed cycle (deep drilling/small-diameter deep-hole drilling)	13.1.3
84	09	Fixed cycle (tapping)	13.1.4
85	09	Fixed cycle (boring)	13.1.5
86	09	Fixed cycle (boring)	13.1.6
87	09	Fixed cycle (back boring)	13.1.7
88	09	Fixed cycle (boring)	13.1.8
89	09	Fixed cycle (boring)	13.1.9
Δ 90	03	Absolute value command	5.1
Δ 91	03	Incremental value command	5.1
92	00	Coordinate system setting Spindle clamp speed setting	19.4 10.3
92.1	00	Workpiece coordinate system preset	19.7
93	05	Inverse time feed	7.5
Δ 94	05	Feed per minute (asynchronous feed)	7.4
Δ 95	05	Feed per revolution (synchronous feed)	7.4
Δ 96	17	Constant surface speed control ON	10.2
Δ 97	17	Constant surface speed control OFF	10.2
* 98	10	Fixed cycle Initial level return	13.1.15
99	10	Fixed cycle (R point level return)	13.1.15
100 to 225	00	User macro (G code call) Max. 10	14.4.4
120.1	00	Machining condition selection I	17.7
121	00	Machining condition selection I cancel	17.7
122	00	Activate sub part system I	16.3.1
127	00	Prohibit manual arbitrary reverse run	15.8
140	00	Arbitrary axis exchange command	16.2.1
141	00	Arbitrary axis exchange return command	16.2.1
142	00	Arbitrary axis exchange reference axis arrange return command	16.2.1
145	00	Cancel sub part systems	16.3
160	00	Torque limitation skip	21.6
186	00	Interference check III interfering object data enable command	20.2

Precautions

(1) Codes marked with * are codes that must be or are selected in the initial state.
The codes marked with Δ are codes that should be or are selected in the initial state by the parameters.

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

(3) This G code list is a list of conventional G codes. Depending on the machine, movements that differ from the conventional G commands may be included when called by the G code macro. Refer to the Instruction Manual issued by the MTB.

(4) Whether the modal is initialized or not depends on each reset input.

(a) "Reset 1"
The modal is initialized when the reset initialization parameter (#1151 rstinit) is ON. (This depends on the MTB specifications.)

(b) "Reset 2" and "Reset & rewind"
The modal is initialized when the signal is input.

(c) Reset at emergency stop release
Conforms to "Reset 1".

(d) When modal is automatically reset at the start of individual functions such as reference position return
Follows "Reset & rewind".

⚠ CAUTION

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

3.5 Precautions Before Starting Machining

⚠ CAUTION

⚠ **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 take into account the machining conditions.**

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

4

Pre-read Buffer

4.1 Pre-read Buffer

Function and purpose

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

Detailed description

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

(1) The data of 1 block is stored in this buffer.

(2) When comments and the optional block skip function is ON, the data extending from the "/" (slash) code up to the EOB code are not read into the pre-read buffer.

(3) The pre-read buffer contents are cleared with resetting.

(4) When the single block function is ON during continuous operation, the pre-read buffer stores the next block's data and then stops operation.

(5) The way to prohibit the M command which operates the external controls from pre-reading, and to make it to recalculate, is as follows:
Identify the M command which operates the external controls by a PLC, and turn on the "recalculation request" on PLC output signal. (When the "recalculation request" is turned ON, the block that has been pre-read is recalculated.)
These operations depend on the MTB specifications.

Precautions

(1) Depending on whether the program is executed continuously or by single blocks, the timing of the validation/invalidation of the PLC signals, including optional block skip, will differ.

(2) If the PLC signal such as optional block skip is turned ON/OFF with the M command, the PLC signal operation will not be effective for the program pre-read with the buffer register.

5

Position Commands

5.1 Position Command Methods ; G90,G91

Function and purpose

By using the G90 and G91 commands, it is possible to execute the next coordinate commands using absolute values or incremental values.
The R-designated circle radius and the center of the circle determined by I, J, K are always incremental value commands.

Command format

G90/G91 X__ Y__ Z__ α__ ;

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

Detailed description

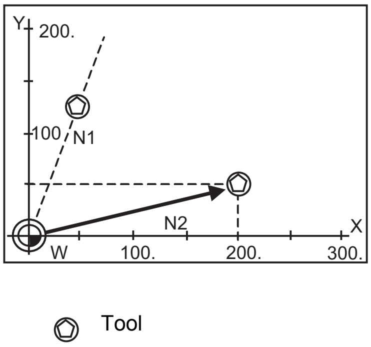
(1) Regardless of the current position, in the absolute value mode, it is possible to move to the position of the workpiece coordinate system that was designated in the program.

N1 G90 G00 X0 Y0 ;

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

N2 G90 G01 X200. Y50. F100 ;
N2 G91 G01 X200. Y50. F100 ;

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



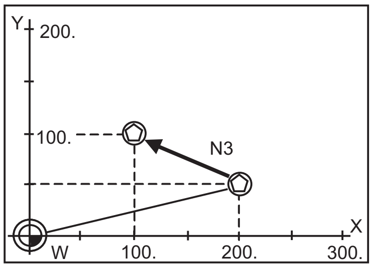
(2) For the next block, the last G90/G91 command that was given becomes the modal.

(G90) N3 X100. Y100. ;

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

(G91) N3 X-100. Y50. ;

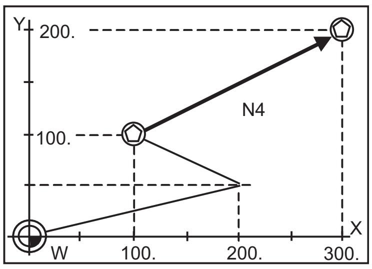
The X axis moves to -100.mm and the Y axis to +50.0mm as an incremental value, and as a result X moves to 100.mm and Y to 100.mm.



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

N4 G90 X300. G91 Y100. ;

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



(4) When the power is turned ON, it is possible to select whether you want absolute value commands or incremental value commands with the #1073 I_Absm parameter.

(5) Even when commanding with the manual data input (MDI), it will be treated as a modal from that block.

5.2 Inch/Metric Conversion ; G20,G21

Function and purpose

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

Command format

Inch command

```
G20;
```

Metric command

```
G21;
```

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".
These parameter settings depend on the MTB specifications.

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.

(3) A program error (P33) will occur if G20/G21 command is issued in the same block as following G codes. Command in a separate block.
- G05 (High-speed machining mode)
- G7.1 (Cylindrical Interpolation)
- G12.1 (Polar coordinate interpolation)

5.3 Decimal Point Input

Function and purpose

This function enables 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.
Whether to apply minimum input command increment (type I) or zero point (type II) to the least significant digit of data without a decimal point depends on the MTB specifications (parameter "#1078 Decpt2").

Detailed description

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

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

Command	Command unit	Type I	Type II
X1;	cunit=10000	1000 (μm, 10^{-4}inch, 10^{-3}°)	1 (mm, inch, °)
	cunit= 1000	100	1
	cunit= 100	10	1
	cunit= 10	1	1

(3) The valid addresses for the decimal points are X, Y, Z, U, V, W, A, B, C, I, J, K, E, F, P, Q, and R. However, P is valid only during scaling. For details, refer to the list.

(4) In decimal point command, the valid range of command value is as shown below. (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	

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

(6) While the smallest decimal point command is validated, the smallest unit for a command without a decimal point designation is the smallest command input unit set in the specifications (1μm, 10μm, etc.) or mm can be selected. This selection can be made with parameter "#1078 Decpt2".

(7) Decimal point commands for decimal point invalid addresses are processed as integer data only and everything after 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.

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

Decimal point input I, II and decimal point command validity

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

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

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

-Addresses used, validity of decimal point commands-

Address	Decimal point command	Usage	Remarks
A	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function codes	
	Valid	Angle data	
	Invalid	Data settings, axis numbers (G10)	
	Invalid	Program No.	
	Invalid	R-Navi data input by program: Coordinate axis selection	
B	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function codes	
	Invalid	Sub part system I: Identification No.	
	Invalid	R-Navi data input by program: Coordinate axis direction setting	
C	Valid	Coordinate position data	
	Invalid	Revolving table	
	Invalid	Miscellaneous function codes	
	Valid	Corner chamfering amount	,C
	Invalid	Tool shape input by program: Tool color	
	Invalid	R-Navi data input by program: Basic coordinate system	
	Invalid	R-Navi data input by program: Coordinate axis direction setting	
D	Invalid	Compensation numbers (tool position, tool radius)	
	Valid	Automatic tool length measurement: Deceleration distance d	
	Invalid	Data setting: Byte type data	
	Invalid	Subprogram storing device number	,D
	Invalid	Sub part system I: Synchronous control designation	
	Valid	Droop skip value	
	Valid	Tool shape input by program: Shape data 1	
	Invalid	R-Navi data input by program: Machining registration No.	
E	Valid	Inch thread: number of ridges, precision thread: lead	
	Invalid	R-Navi data input by program: Coordinate axis direction setting	
	Valid	Synchronous tap: Cutting feedrate (Number of screw threads)	

Address	Decimal point command	Usage	Remarks
F	Valid	Cutting feedrate, automatic tool length measurement speed	
	Valid	Thread lead	
	Valid	Number of Z axis pitch in synchronous tap	
	Valid	Rapid Traverse Rate	,F
	Invalid	R-Navi data input by program: Workpiece shape	
	Invalid	R-Navi data input by program: Coordinate axis direction setting	
G	Valid	Preparatory function code	
H	Invalid	Tool length compensation No.	
	Invalid	Sequence Nos. in subprograms	
	Invalid	Parameter input by program: Bit type data	
	Invalid	Reference spindle selection	
	Invalid	Sub part system I: Reset type	
	Valid	Tool shape input by program: Shape data 2	
	Invalid	R-Navi data input by program: Coordinate axis direction setting	
	Invalid	Tool position compensation: Compensation No.	
I	Valid	Coordinates for arc center and center of figure rotation	
	Valid	Tool radius compensation vector components	
	Valid	Hole pitch in the special fixed cycle	
	Valid	Circle radius of cut circle (increase amount)	
	Valid	G0/G1 in-position width, Hole drilling cycle: G0 in-position width	,I
	Valid	Stroke check before travel: Lower limit coordinates	
	Valid	Tool shape input by program: Shape data 3	
	Valid	R-Navi data input by program: Workpiece shift	
	Invalid	R-Navi data input by program: Coordinate axis direction setting	
J	Invalid	Coordinates for arc center and center of figure rotation	
	Valid	Tool radius compensation vector components	
	Valid	Special fixed cycle's hole pitch or angle	
	Valid	G0/G1 in-position width, Hole drilling cycle: G1 in-position width	,J
	Valid	Stroke check before travel: Lower limit coordinates	
	Valid	Tool shape input by program: Shape data 4	
	Valid	R-Navi data input by program: Workpiece shift	
K	Valid	Coordinates for arc center and center of figure rotation	
	Valid	Tool radius compensation vector components	
	Invalid	Number of holes of the special fixed cycle	
	Invalid	Hole drilling cycle, sub part system I: Number of repetitions	
	Valid	Stroke check before travel: Lower limit coordinates	
	Valid	Spline interpolation 2: Tolerance	
	Invalid	Tool shape input by program: Tool type	
	Valid	R-Navi data input by program: Workpiece shift	
L	Invalid	Number of fixed cycle and subprogram repetitions	
	Invalid	Tool compensation data input by program/workpiece offset input: type selection	L2,L20,L10, L11,L12, L13
	Invalid	Parameter input by program: data setting selection	L70
	Invalid	Parameter input by program: 2-word type data	4 byte
	Invalid	R-Navi data input by program: Start setting workpiece data	L110
	Invalid	R-Navi data input by program: Start setting machining surface data	L111
	Invalid	Timing synchronization number	
	Invalid	Tool life data	

Address	Decimal point command	Usage	Remarks
M	Invalid	Miscellaneous function codes	
	Invalid	R-Navi data input by program: Coordinate axis direction designation method	
N	Invalid	Sequence No.	
	Invalid	Parameter input by program: data numbers	
O	Invalid	Program No.	
P	Invalid/Valid	Dwell time	Parameter
	Invalid	Subprogram program call: program No.	
	Invalid/Valid	Dwell at tap cycle hole base	Parameter
	Invalid	Number of holes of the special fixed cycle	
	Invalid	Amount of helical pitch	
	Invalid	Offset number (G10)	
	Invalid	Constant surface speed control axis number	
	Invalid	Parameter input by program: broad classification number	
	Invalid	Tool compensation data input by program/workpiece offset input: Compensation No.	
	Invalid	Tool shape input by program: Data numbers	
	Invalid	Multi-step skip function 2 signal command	
	Invalid	Subprogram return destination sequence No.	
	Invalid	2nd, 3rd, 4th reference position return number	
	Valid	Scaling factor	
	Invalid	High-speed mode type	
	Invalid	High-accuracy control mode: Start/End	
	Invalid	Extended workpiece coordinate system No, external workpiece coordinate system offset compensation No.	
	Invalid	Tool life data: Group No.	
	Invalid	Machining purpose	
	Invalid	Sub part system I: Start sequence No.	
	Invalid	R-Navi data input by program: Machining surface registration	
	Invalid	R-Navi data input by program: Coordinate axis direction axis designation	
Q	Valid	Cut amount of deep hole drill cycle	
	Valid	Shift amount of back boring	
	Valid	Shift amount of fine boring	
	Invalid	Minimum spindle clamp speed	
	Valid	Thread cutting start shift angle	
	Invalid	Tool life data management method	
	Invalid	Machining condition	
	Invalid	Sub part system I: End sequence No.	
	Invalid	Droop skip value	
	Invalid	R-Navi data input by program: Workpiece registration No.	

Address	Decimal point command	Usage	Remarks
R	Valid	R-point in the fixed cycle	
	Valid	R-specified arc radius	
	Valid	Corner R arc radius	,R
	Valid	Offset amount (G10)	
	Invalid	Synchronous tap/asynchronous tap changeover	,R
	Valid	Synchronous tap: Designation of R point position (absolute or incremental value)	
	Valid	Automatic tool length measurement: deceleration distance r	
	Valid	Rotation angle	
	Invalid	Skip acceleration/deceleration time constant	
	Valid	Tool compensation data input by program/workpiece offset input: Compensation amount	
	Invalid	R-Navi data input by program: Marked point No.	
S	Invalid	Spindle function codes	
	Invalid	Maximum spindle clamp speed	
	Invalid	Constant surface speed control or constant surface speed cancel: Surface speed	
	Invalid	Parameter input by program: word type data	2 byte
	Valid	Synchronous tap: Designation of spindle rotation speed at the return	
T	Invalid	Tool function codes	
U	Valid	Coordinate position data	
V	Valid	Coordinate position data	
W	Valid	Coordinate position data	
X	Valid	Coordinate position data	
	Valid	Dwell time	
	Valid	R-Navi data input by program: Workpiece size	
	Valid	R-Navi data input by program: Feature coordinate origin	
Y	Valid	Coordinate position data	
	Valid	R-Navi data input by program: Workpiece size	
	Valid	R-Navi data input by program: Feature coordinate origin	
Z	Valid	Coordinate position data	
	Valid	R-Navi data input by program: Workpiece size	
	Valid	R-Navi data input by program: Feature coordinate origin	

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 µm	When 1 = 10 µm	1 = 1 mm
G00 X123.45 (decimal points are all mm points)	X123.450 mm	X123.450 mm	X123.450 mm
G00 X12345	X12.345 mm (last digit is 1µm unit)	X123.450 mm	X12345.000 mm
#111=123 #112=5.55 X#111 Y#112	X123.000 mm Y5.550 mm	X123.000 mm Y5.550 mm	X123.000 mm Y5.550 mm
#113=#111+#112 (addition)	#113=128.550	#113=128.550	#113=128.550
#114=#111-#112 (subtraction)	#114=117.450	#114=117.450	#114=117.450
#115=#111*#112 (multiplication)	#115=682.650	#115=682.650	#115=682.650
#116=#111/#112 #117=#112/#111 (division)	#116=22.162 #117=0.045	#116=22.162 #117=0.045	#116=22.162 #117=0.045

Precautions

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

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

Positioning (Rapid Traverse)

G00 X__ Y__ Z__α__ ,I__ ,F__;

X, Y, Z, α	Coordinate values (α is the additional axis.) An absolute position or incremental position is indicated based on the state of G90/G91 at that time.
,I	In-position width. (1 to 999999) This address is valid only in the commanded block. A block that does not contain this address will follow the parameter "#1193 inpos" settings. For details, refer to "7.12 Deceleration Check".
,F	Specifies the rapid traverse rate of the movement initiated by a G00 command, the movement in the G00 mode, and the movement during the fixed cycle for drilling. The range is equal to the range of the feed per minute F command (mm/min, inch/min) in the G01 mode. Switching inch/mm is invalid for rotary axes. For details, refer to "7.1.2 G00 Feedrate Command (,F Command)".

The command addresses are valid for all additional axes.

Detailed description

(1) The rapid traverse speed varies depending on the MTB specifications (parameter "#2001 rapid").
When the "G00 feedrate designation (,F command)" function is enabled and an ",F" command is included in the same block as for the G00 command, positioning is carried out at the feedrate specified by the ",F" command. If this function is invalid or an ",F" command is not designated, positioning is carried out at the feedrate specified in parameter "#2001 rapid".

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

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

(4) G functions (G72 to G89) in the 09 group are canceled (G80) by the G00 command.

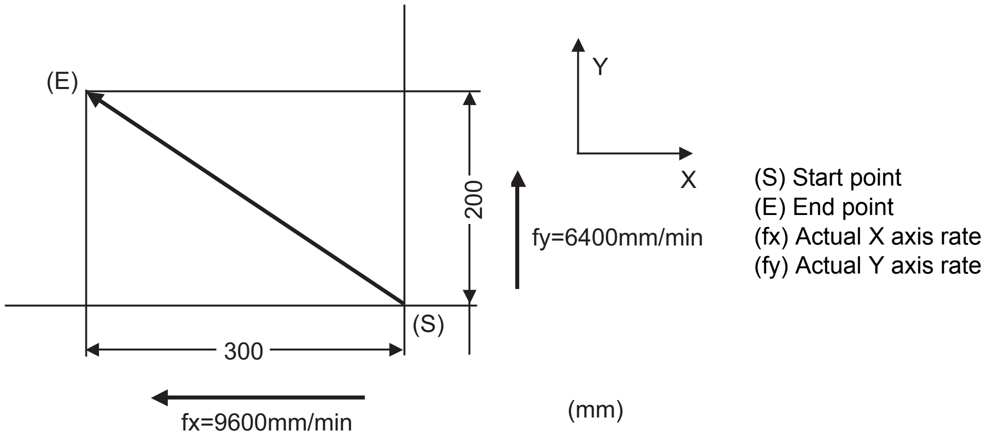
⚠ **CAUTION**

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

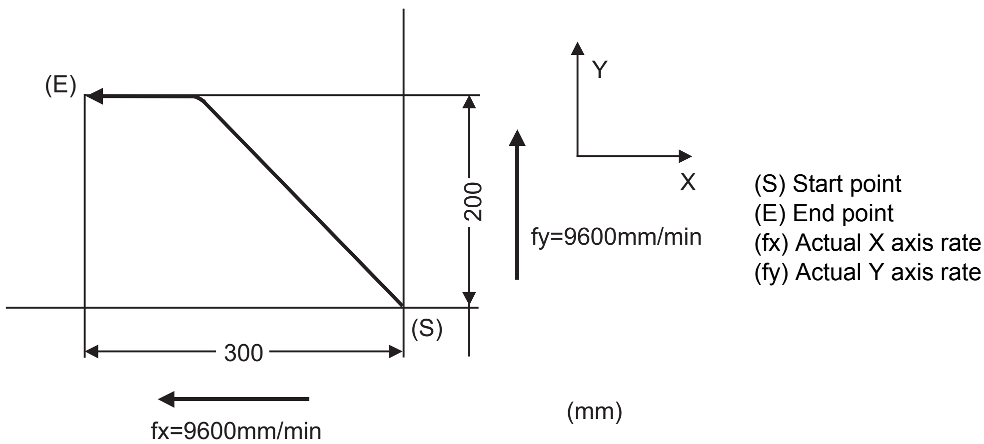
Tool path

Whether the tool moves along a linear or non-linear path varies depending on the MTB specifications (parameter "#1086 G0Intp").
The positioning time does not change according to the path.

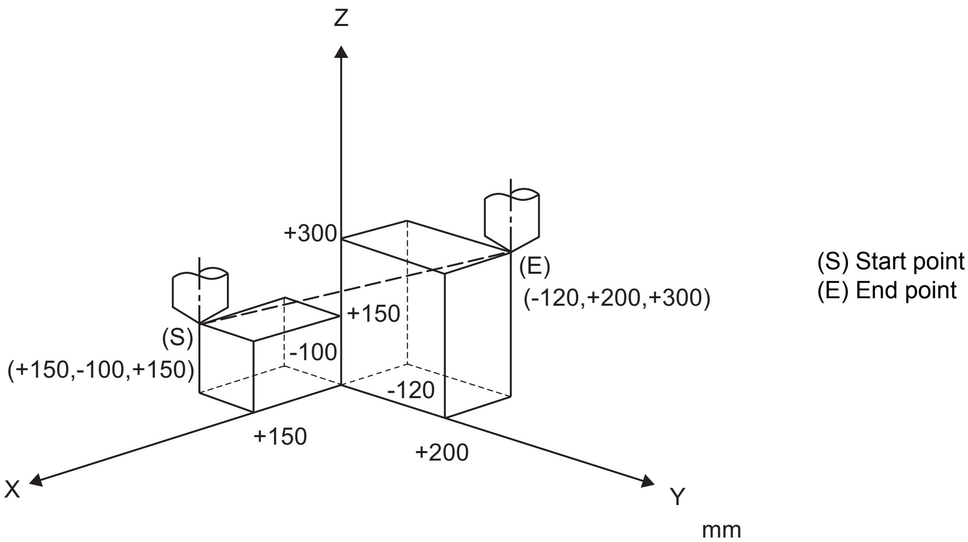
(1) Linear path (When parameter "#1086 G0Intp" is set to "0")
In the positioning process, a tool follows the shortest path that connects the start point and the end point. The positioning speed is automatically calculated so that the shortest distribution time is obtained in order that the commanded speeds for each axis do not exceed the rapid traverse rate.
When, for instance, the X axis and Y axis rapid traverse rates are both 9600 mm/min and when programmed as follows, the tool will follow the path shown in the figure below.
G91 G00 X-300000 Y200000; (With an input setting unit of 0.001 mm)



(2) Non-linear path (When parameter "#1086 G0Intp" is set to "1")
In positioning, the tool will move along the path from the start point to the end point at the rapid traverse rate of each axis.
When, for instance, the X axis and Y axis rapid traverse rates are both 9600 mm/min and when programmed as follows, the tool will follow the path shown in the figure below.
G91 G00 X-300000 Y200000; (With an input setting unit of 0.001 mm)



Program example




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

Precautions for deceleration check

There are three methods of carrying out a deceleration check: the command deceleration check method, the smoothing check method, and the in-position check method. The method used for rapid traverse or cutting feed varies depends on the MTB specification (combination of parameters "#1306 InpsTyp", "#1389 G1SmthChk", "#1223 aux07/bit1", and "#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)
A block without an in-position width command is processed using the deceleration check method based on the MTB specifications (parameter "#1193 inpos").
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 that 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 that 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")	

Refer to "7.12 Deceleration Check" for the deceleration check method.

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

Linear interpolation

G01 X__ Y__ Z__ α__ F__ ,I__ ;

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

Detailed description

(1) G01 command is a modal command in the 01 group. When G01 command is successively issued, the following blocks can be specified only by the coordinate words. If there is no command, a program error (P62) will occur.

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

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

Programmable in-position width command for linear interpolation

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

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

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

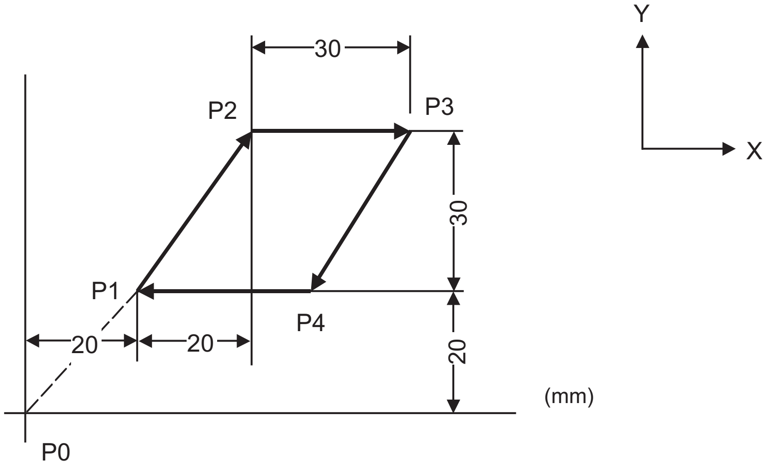
- When the error detection switch is ON.
- When G09 (exact stop check) is commanded in the same block.
- When G61 (exact stop check mode) is selected.

Note

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

Program example

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



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

6.3 Circular Interpolation ; G02,G03

Function and purpose

These commands serve to move the tool along a circular.

Command format

Circular interpolation : Clockwise (CW)

```
G02 X__ Y__ I__ J__ F__ ;
```

Circular interpolation : Counterclockwise (CCW)

```
G03 X__ Y__ I__ J__ F__ ;
```

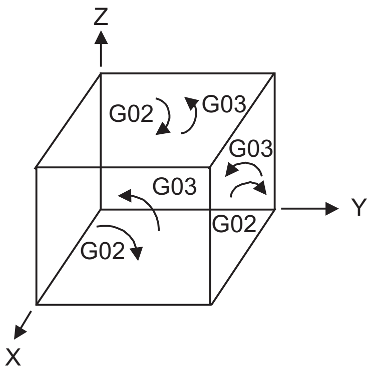
X,Y	Arc end point coordinates
I,J	Arc center coordinates
F	Feedrate

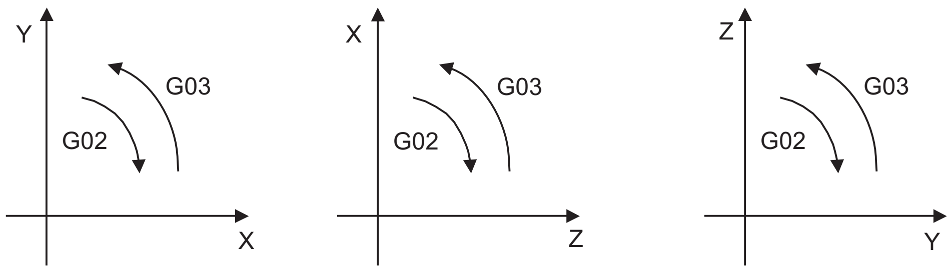
Detailed description

(1) For the arc command, the arc end point coordinates are assigned with addresses X, Y (or Z, or parallel axis X, Y, Z), and the arc center coordinate value is assigned with addresses I, J (or K).
Either an absolute value or incremental value can be used for the arc end point coordinate value command, but the arc center coordinate value must always be commanded with an incremental value from the start point.
The arc center coordinate must be commanded in the input setting 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.

(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 CW (Clockwise)
G03 CCW (Counterclockwise)

(3) Select the XY plane, ZX plane or YZ plane to draw an arc on it, using the plane selection G code.





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

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

(5) The following information is needed for circular interpolation.

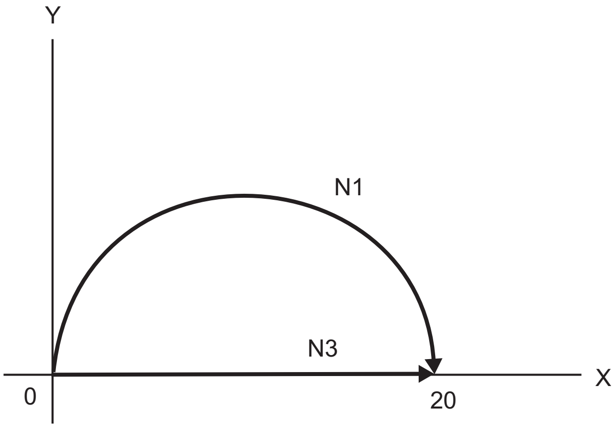
(a) Plane selection	Is there an arc parallel to one of the XY, ZX or YZ planes?
(b) Rotation direction	Clockwise (G02) or counterclockwise (G03)
(c) Arc end point coordinates	Set by addresses X, Y, Z.
(d) Arc center coordinates	Set by addresses I, J, K. (incremental value commands)
(e) Feedrate	Set by address F

(6) 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.

Change into linear interpolation command

Program error (P33) will occur in general use when the center and radius are not designated at circular command. Depending on the MTB specifications, the linear interpolation can be carried out up to the end point coordinates only in a block with no center coordinates or radius specified (parameter "#11029 Arc to G1 no Cent"). However, a modal is the circular modal.
This function is not applied to a circular command by a geometric function.

(Example) #11029 = "1"

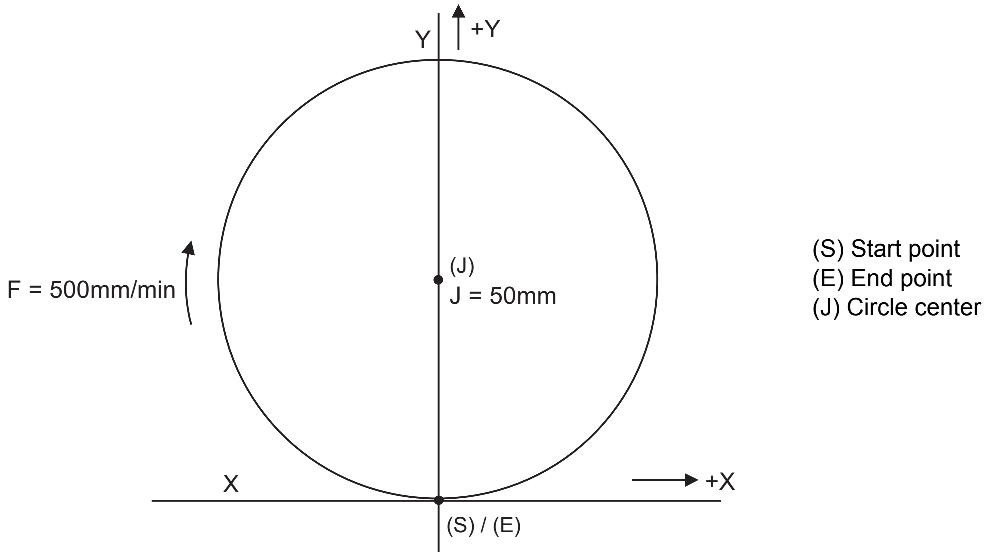



```
G90 X0 Y0 ;
N1 G02 X20. I10. F500 ;     ... (a)
N2 G00 X0 ;
N3 G02 X20. F500 ;          ... (b)
M02 ;
```

(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.

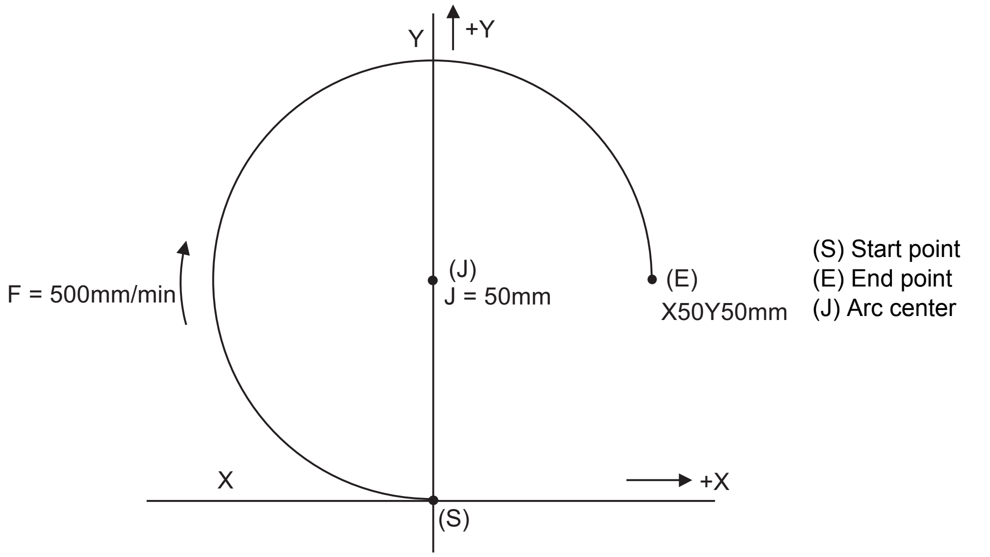
Program example

(Example 1)



G02 J50. F500;	Circle command

(Example 2)



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

Precautions

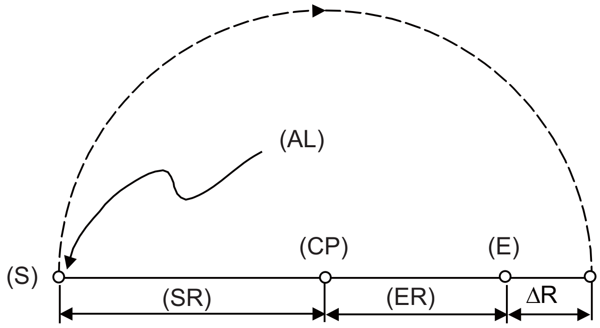
(1) The terms "clockwise" (G02) and "counterclockwise" (G03) used for circular operations are defined as a case where, in a right-hand coordinate system, the negative direction is viewed from the positive direction of the coordinate axis which is at right angles to the plane in question.

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

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

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

(G91) G02 X9.899 I5. ;

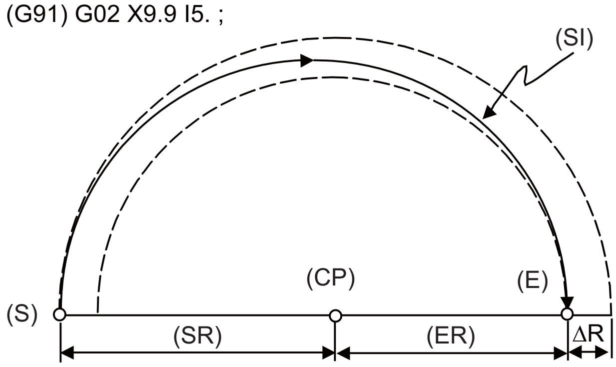


#1084 RadErr parameter value 0.100
Start point radius=5.000
End point radius=4.899
Error ΔR =0.101

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

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

(G91) G02 X9.9 I5. ;

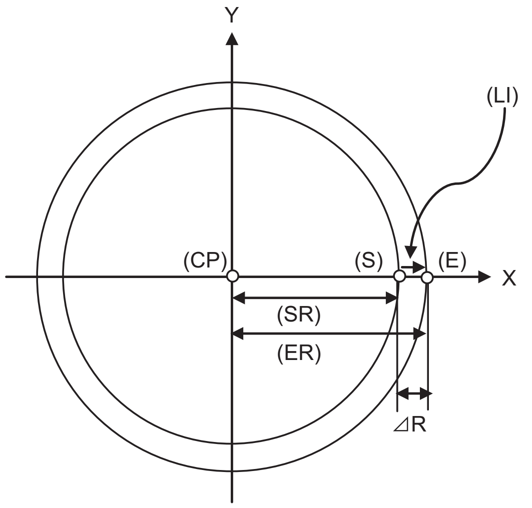


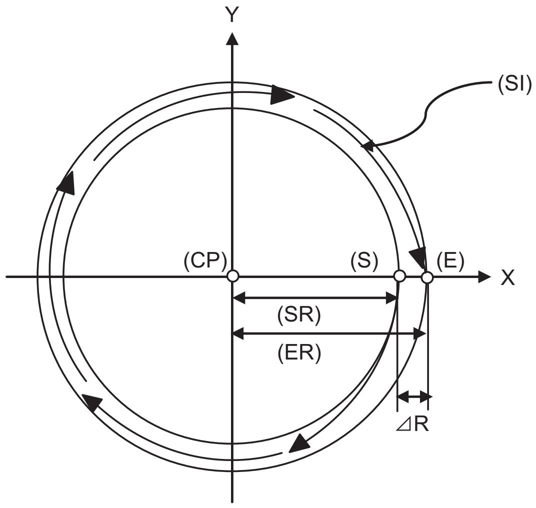
#1084 RadErr parameter value 0.100
Start point radius=5.000
End point radius=4.900
Error ΔR =0.100

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

Also, if "#1084 RadErr" is set to "0", "0.1" is assumed to set.

(c) If the start point radius differs from the end point radius but if the start point angle does not differ from the end point angle, the linear interpolation or spiral interpolation is selected depending on the MTB specifications (parameter "#1278 ext14/bit7").

<table>
<tr><th>#1278 ext14/bit = 0
Linear interpolation</th><th>#1278 ext14/bit7 = 1
Spiral interpolation</th></tr>
<tr><td>G90 G00 X10. Y0.;
G02 X10.01 Y0. I-10.01;</td><td>G90 G00 X10. Y0.;
G02 X10.01 Y0. I-10.01;</td></tr>
<tr><td>


</td><td>


</td></tr>
<tr><td>(CP) Center point
(S) Start point
(E) End point</td><td>(SR) Start point radius
(ER) End point radius
(LI) Linear interpolation
(SI) Spiral interpolation</td></tr>
</table>

6.4 R Specification Circular Interpolation ; G02,G03

Function and purpose

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

Command format

R specification circular interpolation Clockwise (CW)

G02 X__ Y__ R__ F__ ;

R specification circular interpolation Counterclockwise (CCW)

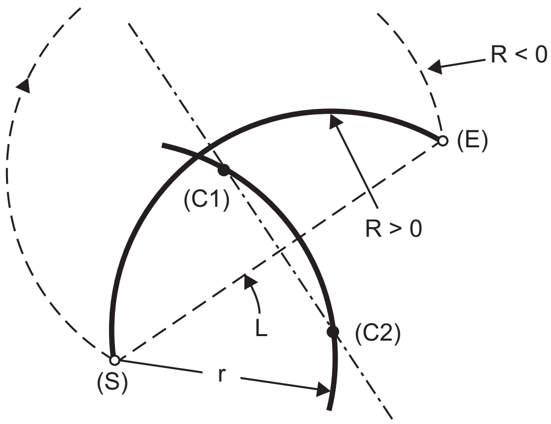
G03 X__ Y__ R__ F__ ;

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

The arc radius must be commanded in the input setting unit. Caution is required for the arc command of an axis for which the input command unit (#1015 cunit) differs. Command with a decimal point to avoid confusion.
A maximum of 6 digits before decimal point can be specified for the radius.

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
(C1): Arc center if R<0
(C2): Arc center if R>0

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

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

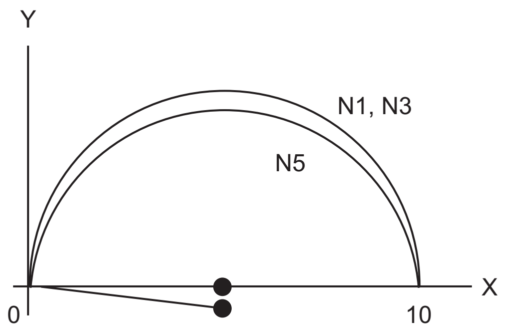
Where L is the line from the start point to the end point. If an R specification and I, J, (K) specification are given at the same time in the same block, the circular command with the R specification takes precedence. In the case of a full-circle command (where the start and end points coincide), an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

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 the segment connecting the start and end points" is compensated for as the circular center.
The setting value depends on the MTB specifications (parameter "#11028 Tolerance Arc Cent" (Tolerable correction value of arc center error)).

(Example) #11028 = "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 G00 X0;
N3 G02 X10. R5.001;     ...(a)
N4 G00 X0;
N5 G02 X10. R5.002;     ...(b)
N6 G00 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 points: 10.000
N3: Radius × 2 = 10.002 "Error 0.002 -> Compensate"
N5: Radius × 2 = 10.004 "Error 0.004 -> Do not compensate"

Program example

(Example 1)

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

(Example 2)

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

(Example 3)

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

(Example 4)

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

Precautions

(1) In the case of a full-circle command (where the start and end points coincide), an R specification circular command will be completed immediately even if it is issued and no operation will be executed. An I, J, K specification circular command should therefore be used in such a case.

(2) 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.

6.5 Plane Selection ; G17,G18,G19

Function and purpose

The plane to which the movement of the tool during the circle interpolation (including helical cutting) and tool radius compensation command belongs is selected.
If the 3 basic axes and the parallel axes corresponding to these basic axes are entered as parameters, the commands can select the plane composed of any 2 axes which are not parallel axes. If a rotary axis is entered as a parallel axis, the commands can select the plane containing the rotary axis.
These commands are used to select:

- Plane that executes circular interpolation (including helical cutting)
- Plane that executes tool radius compensation
- Used to select a plane that executes fixed cycle positioning.

Command format

G17; (J-K plane selection)

G18; (Z-X plane selection)

G19; (Y-Z plane selection)

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

Detailed description

Parameter entry

	#1026-1028 Basic axis I, J, K	#1029-1031 Flat axis I, J, K
I	X	U
J	Y	
K	Z	V

Table 1 Examples of plane selection parameter entry

As shown in the above example, the basic axis and its parallel axis can be registered.
The basic axis can be an axis other than X, Y and Z.
Axes that are not registered are irrelevant to the plane selection.

Plane selection system

In Table 1, characters I, J, K represent the following axes.

I: Horizontal axis for the G17 plane or the vertical axis for the G18 plane

J: Vertical axis for the G17 plane or horizontal axis for the G19 plane

K: Horizontal axis for the G18 plane or vertical axis for the G19 plane

Therefore, the G17, G18 and G19 commands select the following planes.

G17: I-J plane

G18: K-I plane

G19: J-K plane

(1) Axis addresses assigned in the same block as the plane selection (G17, G18, G19) command determine which of the basic axes or parallel axes are to be in the actual plane selected.
For the parameter entry example in Table 1
G17 X__ Y__ ;   X-Y plane
G18 X__ V__ ;   V-X plane
G18 U__ V__ ;   V-U plane
G19 Y__ Z__ ;   Y-Z plane
G19 Y__ V__ ;   Y-V plane

(2) Plane selection is not performed with blocks in which the plane selection G code (G17, G18, G19) is not assigned.
G17 X__ Y__ ;   X-Y plane
Y__ Z__ ;   X-Y plane (No plane change)

(3) If the axis address is omitted in the block where the plane selection G code (G17, G18, G19) is commanded, it is assumed that the axis addresses of the 3 basic axes have been omitted.
For the parameter entry example in Table 1
G17 ;   X-Y plane
G17 U__ ;   U-Y plane
G18 U__ ;   Z-U plane
G18 V__ ;   V-X plane
G19 Y__ ;   Y-Z plane
G19 V__ ;   Y-V plane

(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) The axis command that does not exist in the plane determined by the plane selection G code (G17, G18, G19) is irrelevant to the plane selection.
For the parameter entry example in Table 1
G17 U__Z__ ;
If the above is commanded, the U-Y plane will be selected, and Z will move regardless of the plane.

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

Note

- When the power is turned ON or when the system is reset, the plane set by the parameter "#1025 Initial plane selection" is selected.

6.6 Thread Cutting

6.6.1 Constant Lead Thread Cutting ; G33

Function and purpose

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

Command format

Normal lead thread cutting

G33 Z__(X__ Y__ α__) F__ Q__ ;

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

Precision lead thread cutting

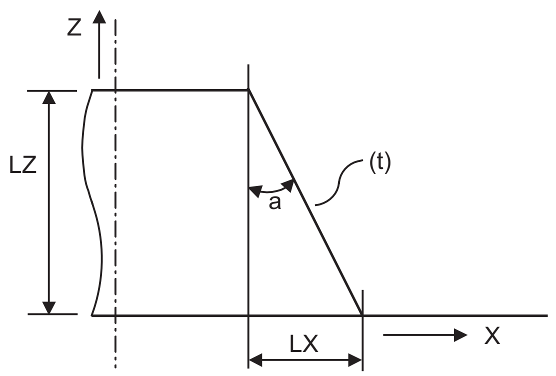
G33 Z__(X__ Y__ α__) E__ Q__ ;

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

Detailed description

(1) The E command is also used for the number of ridges in inch thread cutting, and whether the number of ridges or precision lead is to be designated can be selected by parameter setting. (Parameter "#8156 Fine thread cut E" is set to "1" for precision lead designation.)

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



(t) Tapered thread section
When a < 45°, lead is in LZ direction.
When a > 45°, lead is in LX direction.
When a = 45°, lead can be in either LZ or LX direction.

Thread cutting metric input

Input Setting unit	**B (0.001 mm)**			**C (0.0001 mm)**		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)	F (mm/rev)	E (mm/rev)	E (ridges/inch)
Minimum command unit	1 (=0.001) (1.=1.000)	1 (=0.0001) (1.=1.0000)	1 (=1.00) (1.=1.00)	1 (=0.0001) (1.=1.0000)	1 (=0.00001) (1.=1.00000)	1 (=1.000) (1.=1.000)
Range	0.001 to 999.999	0.0001 to 999.9999	0.03 to 999.99	0.0001 to 999.9999	0.00001 to 999.99999	0.026 to 222807.017

Input Setting unit	**D (0.00001 mm)**			**E (0.000001 mm)**		
Command address	F (mm/rev)	E (mm/rev)	E (ridges/inch)	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 (=1.0000) (1.=1.0000)	1 (=0.000001) (1.=1.000000)	1 (=0.0000001) (1.=1.0000000)	1 (=1.00000) (1.=1.00000)
Range	0.00001 to 999.99999	0.000001 to 999.999999	0.0255 to 224580.0000	0.000001 to 999.999999	0.0000001 to 999.9999999	0.02541 to 224719.00000

Thread cutting inch input

Input Setting unit	**B (0.0001inch)**			**C (0.00001inch)**		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)	F (inch/rev)	E (inch/rev)	E (ridges/inch)
Minimum command unit	1 (=0.0001) (1.=1.0000)	1 (=0.00001) (1.=1.00000)	1 (=1.000) (1.=1.000)	1 (=0.00001) (1.=1.00000)	1 (=0.000001) (1.=1.000000)	1 (=1.0000) (1.=1.0000)
Range	0.0001 to 39.3700	0.00001 to 39.37007	0.025 to 9999.999	0.00001 to 39.37007	0.000001 to 39.370078	0.0255 to 9999.9999

Input Setting unit	**D (0.000001 inch)**			**E (0.0000001 inch)**		
Command address	F (inch/rev)	E (inch/rev)	E (ridges/inch)	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 (=1.00000) (1.=1.00000)	1 (=0.0000001) (1.=1.0000000)	1 (=0.00000001) (1.=1.00000000)	1 (=1.000000) (1.=1.000000)
Range	0.000001 to 39.370078	0.0000001 to 39.3700787	0.02541 to 9999.99999	0.0000001 to 39.3700787	0.00000001 to 39.37007873	0.025401 to 9999.999999

<Note>

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

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

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

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

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

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

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

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

(10) A program error (P93) may occur when the result of the expression (9) is R<1 because the thread lead is very large to the highest cutting feedrate.

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

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

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

(14) When a thread cutting is commanded during tool radius compensation, the compensation is temporarily canceled and the thread cutting is executed.

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

(16) When the mode is switched to the manual mode while G33 is executed, the following block which does not contain a thread cutting command is first executed and then the automatic operation stops. In the case of a single block, the following block which does not contain a thread cutting command (G33 mode is canceled) 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.
Make sure to carry out timing synchronization operation between part systems before issuing a thread cutting command with multiple part systems. For example, when using the 1-spindle specifications with two part systems, if one part system issues a thread cutting command during ongoing thread cutting by another part system, the movement will start without waiting for the rotary encoder single rotation synchronization signal causing an illegal operation.

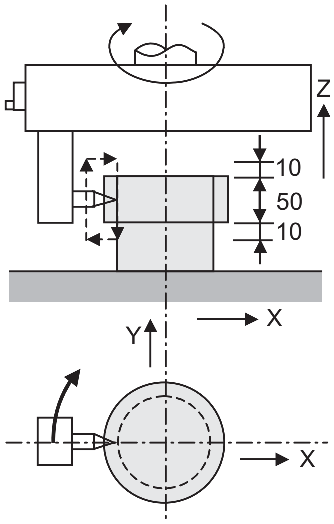
(18) The thread cutting start shift angle is not modal. If there is no Q command with G33, this will be handled as "Q0".

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

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

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

Program example



N110 G90 G00 X-200. Y-200. S50 M3 ;	The spindle center is positioned to the workpiece center, and the spindle rotates in the forward direction.
N111 Z110 ;	
N112 G33; Z40 F6.0;	The first thread cutting is executed. Thread lead = 6.0mm
N113 M19 ;	Spindle orientation is executed with the M19 command.
N114 G00 X-210.;	The tool is evaded in the X axis direction.
N115 Z110. M0 ;	The tool rises to the top of the workpiece, and the program stops with M00. Adjust the tool if required.
N116 X-200. ; M3 ;	Preparation for second thread cutting is done.
N117 G04 X5.0 ;	Command dwell to stabilize the spindle rotation if necessary.
N118 G33 Z40.;	The second thread cutting is executed.

6.6.2 Inch Thread Cutting ; G33

Function and purpose

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

Command format

G33 Z__ (X_ Y_ α_) E__ Q__ ; ... Inch thread cutting

Z (X Y α)	End point of thread cutting
E	Number of ridges per inch in direction of long axis (axis which moves most) (decimal point command can also be assigned)
Q	Thread cutting start shift angle, 0 to 360°

Detailed description

(1)The number of ridges in the long axis direction is assigned as the number of ridges per inch.

(2)The E code is also used to assign the precision lead length, and whether the number of ridges or precision lead length is to be designated can be selected by parameter setting. (The number of ridges is designated by setting the parameter "#8156 Fine thread cut E" to "0".)

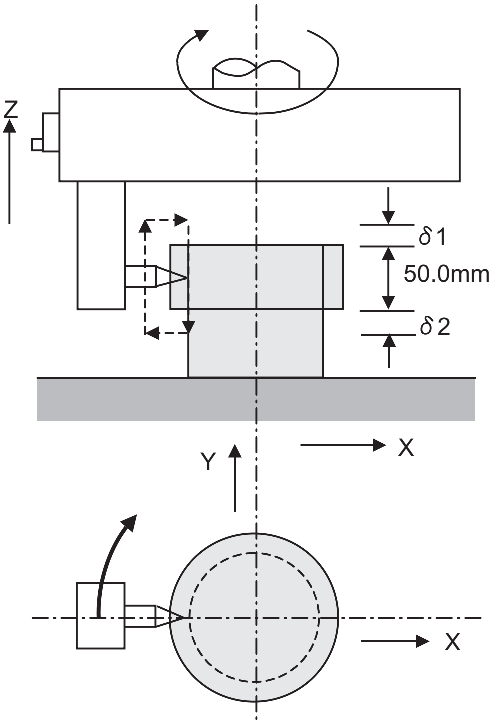
(3)The E command value should be set within the lead value range when converted to lead.

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

Program example

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

When programmed with δ1= 10 mm, δ2=10 mm using metric input

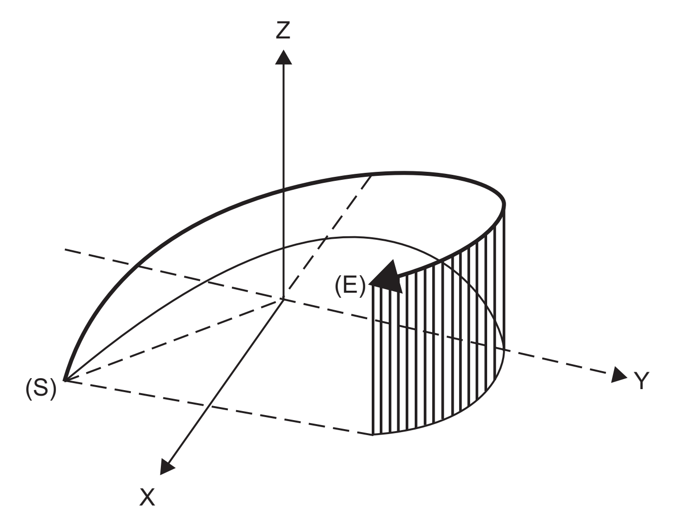


N210 G90 G00 X-200. Y-200. S50 M3 ;	
N211 Z110. ;	
N212 G91 G33 Z-70. E3.0 ;	(First thread cutting)
N213 M19;	
N214 G90 G00 X-210. ;	
N215 Z110. M0 ;	
N216 X-200. ; M3 ;	
N217 G04 X2.0 ;	
N218 G91 G33 Z-70. ;	(Second thread cutting)

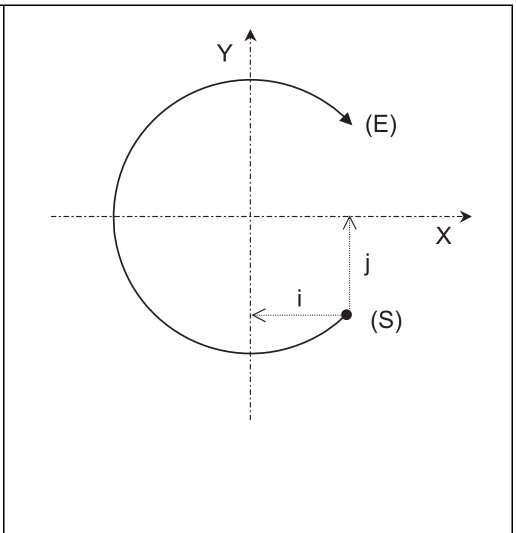
6.7 Helical Interpolation ; G17,G18,G19 and G02,G03

Function and purpose

When this interpolation is performed with 3 orthogonal axes, the tool will travel helically when circular interpolation is executed for any 2 axes and, at the same time, when another 1 axis is synchronized with the rotation of the circular and linear interpolation is executed synchronously with the rotation of the circular arc.
Large diameter threads and 3-dimensional cams can be machined.



3-dimensional path



XY plane path (projection path)

(S) Start point (E) End point

Program command path — XY plane projection path in command program

Command format

Helical interpolation command (Specify arc center)

G17/G18/G19 G02/G03 X__ Y__ Z__ I__ J__ P__ F__ ;

Helical interpolation command (Specify radius (R))

G17/G18/G19 G02/G03 X__ Y__ Z__ R__ F__ ;

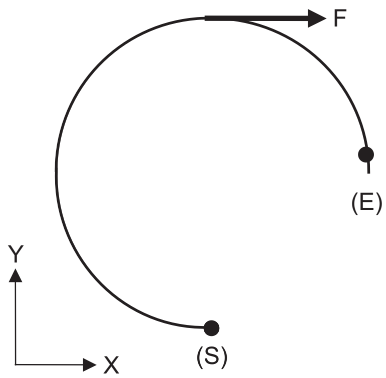
G17/G18/G19	Arc plane (G17: XY plane, G18: ZX plane, G19: YZ plane)
G02/G03	Arc rotation direction (G02: clockwise, G03: counterclockwise)
X, Y	Arc end point coordinates
Z	Linear axis end point coordinates
I, J	Arc center coordinates
P	Number of pitches
R	Circular radius
F	Feedrate

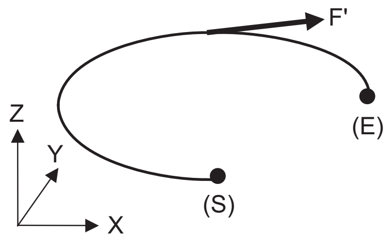
Either an absolute value or incremental value can be used for the arc end point coordinate value command and the linear axis end point coordinate value command, but the arc center coordinate value must always be commanded with an incremental value from the start point.
The arc center coordinate value and arc radius value must be commanded in the input setting unit. Caution is required for the helical interpolation command of an axis for which the input command unit differs.
Command with a decimal point to avoid confusion.

Detailed description

Speed designation during the helical interpolation

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





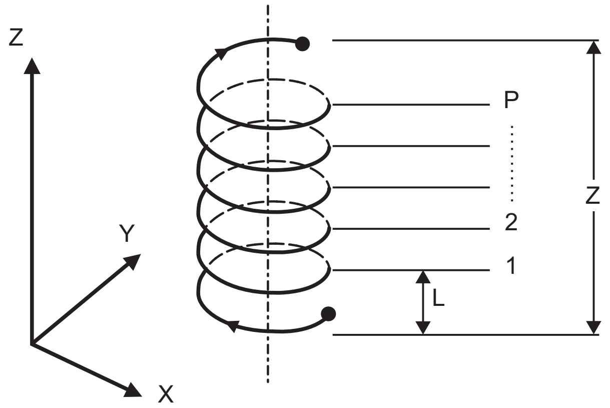
(S) Start point (E) End point

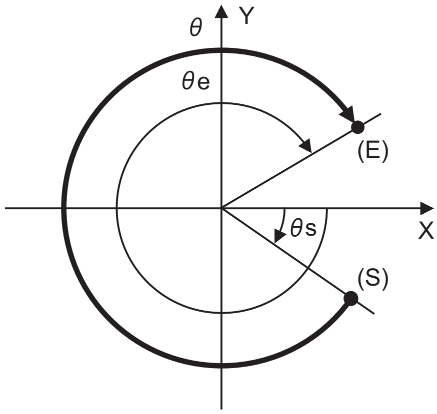
Whether the arc plane element speed designation or the normal speed designation is valid depends on the MTB specifications.

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

Command the feedrate F as the composite speed for each axis.

Normal speed designation





(S) Start point (E) End point

(1) This command should be issued with a linear axis (multiple axes can be commanded) that does not contain a circular axis in the circular interpolation command combined.

(2) For feedrate F, command the X, Y and Z axis composite element directions speed.

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

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

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

xs, ys: Start point coordinates from the arc center xe, ye: End point coordinates from the arc center

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

<Note>

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

(5) Plane selection

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

XY plane circular, Z axis linear
Command the X, Y and Z axis addresses in the G02 (G03) and G17 (plane selection G code) mode.
ZX plane circular, Y axis linear
Command the X, Y and Z axis addresses in the G02 (G03) and G18 (plane selection G code) mode.
YZ plane circular, X axis linear
Command the X, Y and Z axis addresses in the G02 (G03) and G19 (plane selection G code) mode.

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

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

In addition to the basic command methods above, the command methods described in the following program example can be used. Refer to the section "6.5 Plane Selection ; G17,G18,G19" for the arc planes selected with these command methods.

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.

(Example)

G17 G91 G02 X10. Y10. Z-4. I10. F100 ;	Helical interpolation at speed at which arc plane element is F100
G01 X20 ;	Linear interpolation at F100
G02 X10. Y-10. Z4. J-10. ;	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.

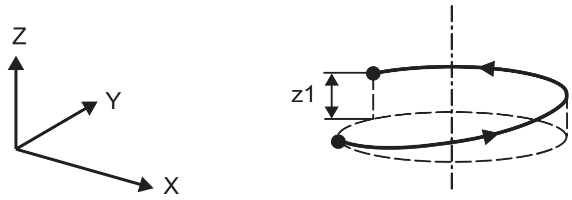
(2) The modal value speed display (FA) indicates the command speed.

(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

(Example 1)

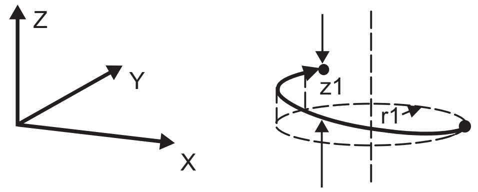


G17;	XY plane
G03 Xx1 Yy1 Zz1 Ii1 Jj1 P0 Ff1;	XY plane arc, Z axis linear

Note

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

(Example 2)

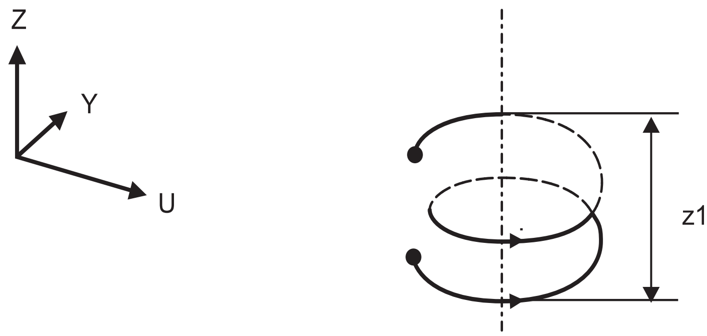


G17;	XY plane
G02 Xx1 Yy1 Zz1 Rr1 Ff1;	XY plane arc, Z axis linear

Note

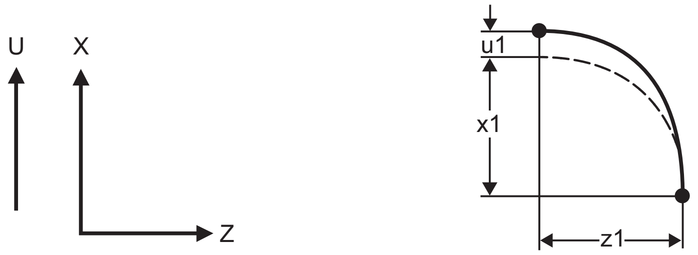
(1) The number of pitches is ignored even when it is commanded.

(Example 3)



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

(Example 4)



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

Note

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

(Example 5)

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

Note

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

6.8 Unidirectional Positioning

The unidirectional positioning function positions the tool at a high degree of precision without backlash error by locating the final tool position from a constant direction.
There are two types of positioning methods: G command method (G60 use method) and axis-based unidirectional positioning method (in which the axis specified by the MTB is always targeted for unidirectional positioning).
For the specifications of the machine you are using, see the specifications or Instruction Manual issued by the MTB.

6.8.1 Unidirectional Positioning ; G60

Function and purpose

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

Command format

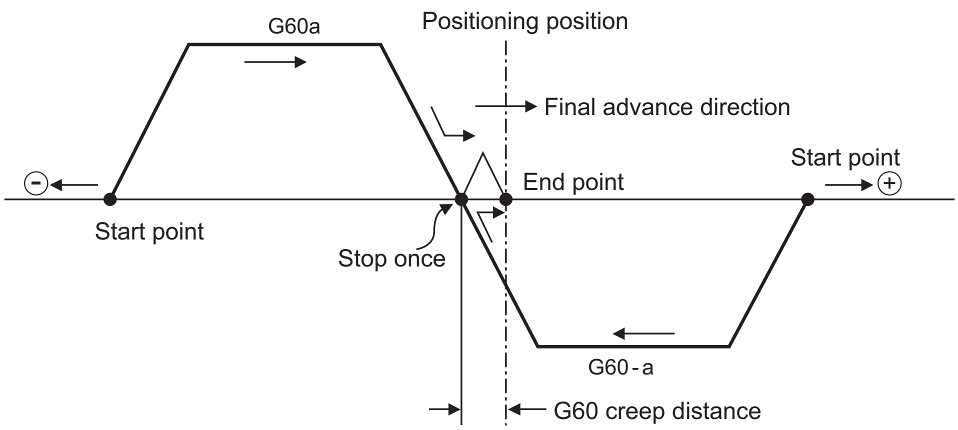
G60 X__ Y__ Z__ α__; ... Unidirectional positioning

α	Additional axis

Detailed description

(1) The creep distance for the final positioning as well as the final positioning direction is set by parameter.

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



(3) The above positioning operation is performed even when Z axis commands have been assigned for Z axis cancel and machine lock. (Display only)

(4) When the mirror image function is ON, the tool will move in the opposite direction as far as the intermediate position due to the mirror image function but the operation within the creep distance during its final advance will not be affected by a mirror image.

(5) The tool moves to the end point at the dry run speed during dry run when the G00 dry run function is valid.

(6) Feed during creep distance movement with final positioning can be stopped by resetting, emergency stop, interlock, feed hold and rapid traverse override zero.
The tool moves over the creep distance at the rapid traverse setting. Rapid traverse override is valid.

(7) Unidirectional positioning is not performed for the drilling axis during drilling fixed cycles.

(8) Unidirectional positioning is not performed for shift amount movements during the fine boring or back boring fixed cycle.

(9) Normal positioning is performed for axes whose creep distance has not been set by parameter.

(10) Unidirectional positioning is always a non-interpolation type of positioning.

(11) When the same position (movement amount of zero) has been commanded, the tool moves back and forth over the creep distance and is positioned at its original position from the final advance direction.

(12) Program error (P61) will occur when the G60 command is assigned with an NC system which has not been provided with this particular specification.

(13) The G60 command is assigned to group 00 (unmodal) in the previous versions; however, it can be operated as the modal G code of group 01 depending on the MTB specifications (parameter "#1271 ext07/bit3").
This omits a step to command G60 for each block.
This G60 command is the same as the previous unmodal G60 command, except it handles the G60 command as a modal.

(14) If the G code of group 01 is commanded in the same block when the G60 command is handled as a modal, the G code commanded next becomes valid.

6.8.2 Axis-based Unidirectional Positioning

Function and purpose

This function carries out unidirectional positioning for each axis for G00 positioning.
The target axis is determined in the MTB specifications (parameter "#2084 G60_ax").
When the unidirectional positioning is commanded, set the last positioning direction and distance to parameter "#8209 G60 shift amount" for each axis.
The example below shows a case in which axis B is set as the unidirectional positioning axis.

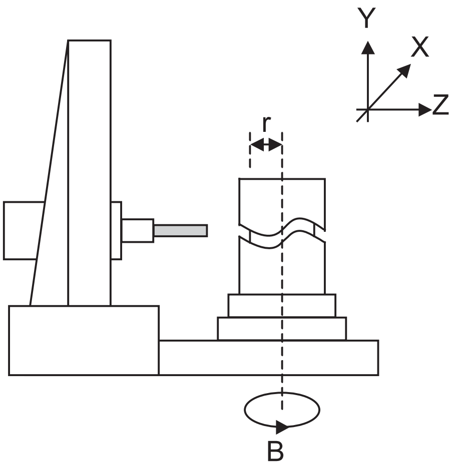
G00 X100.;	...	Normal positioning
G00 B100.;	...	Unidirectional positioning
G00X200.B200.;	...	Axis X: Normal positioning, Axis B: Unidirectional positioning

The axis-based unidirectional positioning is the same as for the G60 command. Refer to "Unidirectional Positioning:G60".

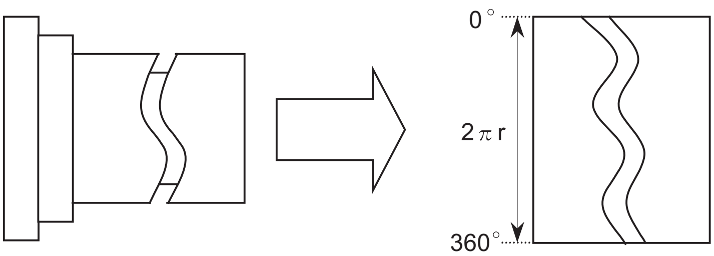
6.9 Cylindrical Interpolation ; G07.1

Function and purpose

This function develops a shape on the side of a cylinder (shape in a cylindrical coordinate system) into a plane. When the developed shape is programmed as the plane coordinates, it will be converted into a linear axis movement and rotary axis movement in the cylindrical coordinates to conduct a contour control when machining.



As programming can be carried out to the developed shape of the side of the cylinder, this is effective for machining cylindrical cams, etc. When programmed with the rotary axis and its orthogonal axis, grooves and other shapes can be machined on the side of the cylinder.



Command format

Cylindrical interpolation mode start/cancel

G07.1 C__ ;

C	Cylinder radius value (When rotary axis name is "C") Radius value ≠ 0: Cylindrical interpolation mode start Radius value = 0: Cylindrical interpolation mode cancel

Detailed description

(1) The coordinate commands in the interval from the start to cancellation of the cylindrical interpolation mode will be the cylindrical coordinate system.

G07.1 C20.;	Cylindrical interpolation mode start (Cylindrical interpolation will start.)
:	(The coordinate commands in this interval will be the cylindrical coordinate system)
G07.1 C0 ;	Cylindrical interpolation mode cancel (Cylindrical interpolation will be canceled.)

(2) G107 can be used instead of G07.1.

(3) Command G07.1 is an independent block. If it is commanded in the same block with other G code, a program error (P33) will occur.

(4) Program the rotary axis with an angle degree.

(5) Linear interpolation or circular interpolation can be commanded during the cylindrical interpolation mode. Note that the plane selection command must be issued just before the G07.1 block.

(6) The coordinate commands can be both an absolute command or incremental command.

(7) Tool radius compensation can be applied on the program command. Cylindrical interpolation will be executed to the path after it has gone through a tool radius compensation.

(8) Command the tangent speed on the developed cylinder by F. F is in mm/min or inch/min unit.

Cylindrical interpolation accuracy

In the cylindrical interpolation mode, the movement amount of the rotary axis commanded with an angle is converted into distance on a circle periphery, and after calculating the linear and circular interpolation between the other axes, the amount is converted into an angle again.
Thus, the actual movement amount may differ from the commanded value such as when the cylinder radius is small. Note that the gap generated by this will not be cumulated.

Related parameter

- #1516 mill_ax (Milling axis name)
- #8111 Milling Radius
- #1267 ext03/bit0 (G code type)
- #1270 ext06/bit7 (Handling of C axis coordinate during cylindrical interpolation)

Plane selection

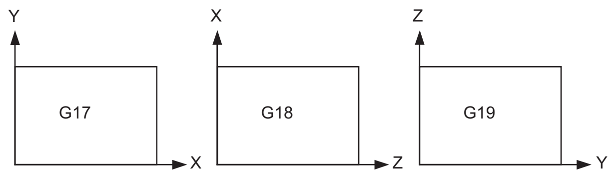
The axis used for cylindrical interpolation must be set with the plane selection command. (*1)
Use parameters (#1029, #1030 and #1031) to set which parallel axis corresponds to the rotary axis.
The circular interpolation and tool radius compensation, etc., can be designated on that plane.
The plane selection command is set immediately before or after the G07.1 command. If a movement command is issued without this command, a program error (P485) will occur.

When the axis address is commanded while selecting the plane, the axis will travel. To prevent the axis from traveling, command the address with the incremental value.

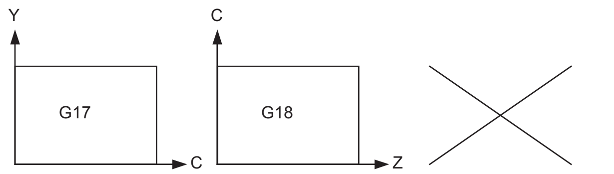
(Example)

G19 Z0. C0. ;	Plane selection command for cylindrical interpolation, and 2-axis command of Z axis and C axis for interpolation
G07.1 C100 ;	Cylindrical interpolation start
:	
G07.1 C0 ; .	Cylindrical interpolation cancel

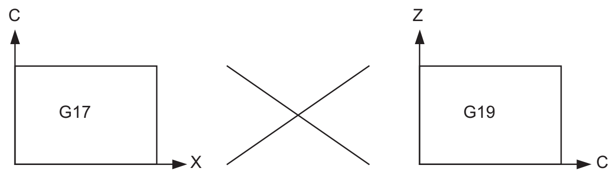
Basic coordinate system
X, Y, Z



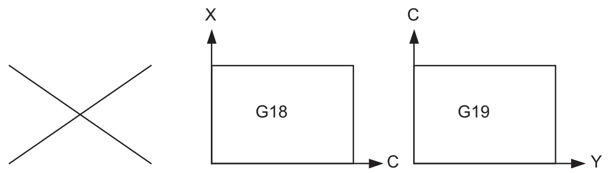
Cylindrical coordinate system
C , Y , Z
(Rotary axis is X axis' parallel axis)
#1029



Cylindrical coordinate system
X , C , Z
(Rotary axis is Y axis' parallel axis)
#1030

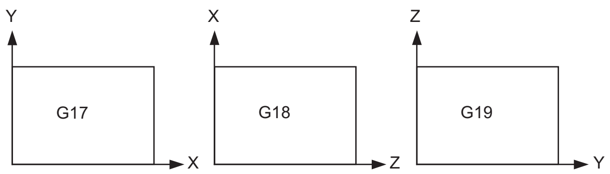


Cylindrical coordinate system
X , Y , C
(Rotary axis is Z axis' parallel axis)
#1031

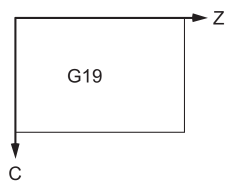


(*1) Depending on the model or version, the Z-C plane (Y-Z cylindrical plane) will be automatically selected with G07.1 and G19.
The circular interpolation and tool radius compensation, etc., can be designated on that plane.

Basic coordinate system
X, Y, Z



Cylindrical coordinate system



Program example

<Program>

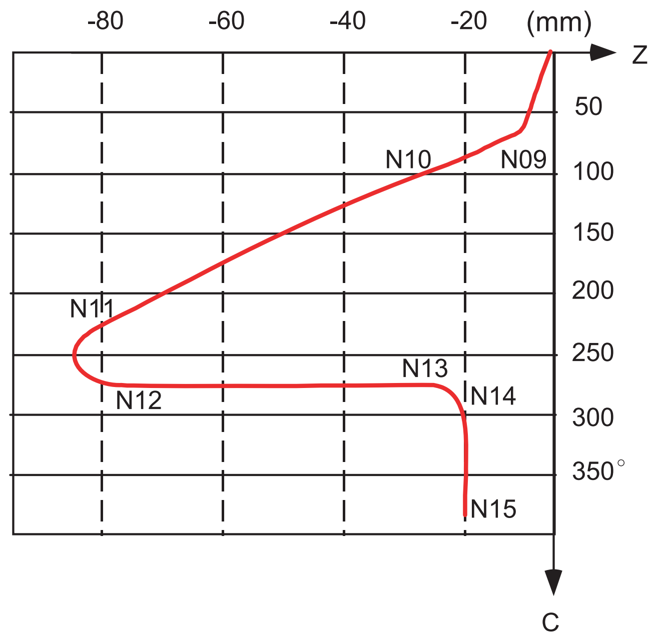
N01 G28 XZC ;	
N02 T020 ;	
N03 G97 S100 M23 ;	
N04 G00 X50. Z0. ;	
N05 G94 G01 X40. F100. ;	
N06 G19 C0 Z0 ;	Plane selection command for cylindrical interpolation and two axes command for interpolation
N07 G07.1 C20. ;	Cylindrical interpolation start
N08 G41 ;	
N09 G01 Z-10. C80. F150 ;	
N10 Z-25. C90. ;	
N11 Z-80. C225. ;	
N12 G03 Z-75.C270. R55. ;	
N13 G01 Z-25 ;	
N14 G02 Z-20.C280. R80. ;	
N15 G01 C360. ;	
N16 G40 ;	
N17 G07.1 C0 ;	Cylindrical interpolation cancel
N18 G01 X50. ;	
N19 G00 X100. Z100. ;	
N20 M25 ;	
N21 M30 ;	

<Parameters>

#1029 aux_I

#1030 aux_J C

#1031 aux_K



Relationship with Other Functions

Circular interpolation

(1) Circular interpolation between the rotary axis and linear axis is possible during the cylindrical interpolation mode.

(2) An R specification command can be issued with circular interpolation. (I, J and K cannot be designated.)

Tool radius compensation

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

(1) Command the plane selection in the same manner as circular interpolation.
When using tool radius compensation, start up/cancel the compensation in the cylindrical interpolation mode.

(2) A program error (P485) will occur if G07.1 is commanded during tool radius compensation.

(3) If the G07.1 command is issued with no movement command after the tool radius compensation is canceled, the position of the axis in the G07.1 command block is interpreted as the position applied after the tool radius compensation is canceled and the following operations are performed.

Cutting feed per minute (asynchronous feed)

(1) The feed per minute (asynchronous) mode is forcibly set when the cylindrical interpolation mode is started.

(2) When the cylindrical interpolation mode is canceled, the feed per revolution (synchronous) will return to the state before the cylindrical interpolation mode was started.

Constant surface speed control

(1) A program error (P485) will occur if G07.1 is commanded in the constant surface speed control mode (G96).

Miscellaneous function

(1) The miscellaneous function (M) and 2nd miscellaneous function can be issued in the cylindrical interpolation mode.

(2) The S command in the cylindrical interpolation mode specifies the rotary tool's rotation speed instead of the spindle rotation speed.

Tool length compensation

(1) Program error (P481) will occur if tool length compensation is performed in the cylindrical interpolation mode.

```
 :
G43 H12 ;          Tool length compensation before cylindrical interpolation -> Valid
G00 X100. Z0. ;
G19 Z C ;
G07.1 C100. ;
 :
G43 H11 ;          Tool length compensation in cylindrical interpolation mode -> Program error
 :
G07.1 C0 ;
```

(2) Complete the tool compensation operation (movement of tool length and wear compensation amount) before executing the cylindrical interpolation.
If the tool compensation operation is not completed when the cylindrical interpolation start command is issued, the followings will occur:

- The machine coordinate will not change even if G07.1 is executed.
- The workpiece coordinate will change to that of the post tool length compensation when G07.1 is executed.

(Even if the cylindrical interpolation is canceled, this workpiece coordinate will not be canceled.)

F command during cylindrical interpolation

As for the F command during cylindrical interpolation mode, whether to use the previous F command depends on the previous mode of the feed per minute command (G94/G98) or feed per rotation command (G95/G99).

(1) When G94 (G98) is commanded just before G07.1
If there is no F command in the cylindrical interpolation, the previous F command feedrate will be used.
After the cylindrical interpolation mode is canceled, the F command feedrate set at the start of the cylindrical interpolation mode or the last F command feedrate set during cylindrical interpolation will continue to be the feedrate.

(2) When G95 (G99) is commanded just before G07.1
The previous F command feedrate cannot be used during cylindrical interpolation, thus a new F command must be issued.
After the cylindrical interpolation mode is canceled, the feedrate will return to the state before the cylindrical interpolation mode was started.

When there is no F command in G07.1

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

When F is commanded in G07.1

Previous mode	With F command	After G07.1 is canceled
G94 (G98)	Commanded F is used	←
G95 (G99)	Commanded F is used (*1)	F just before G07.1 is used
(*1) Moves with the feed per minute command during G07.1.		

Restrictions and precautions

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

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

A program error will occur if a G code other than those listed above is commanded during cylindrical interpolation.

(2) The cylindrical interpolation mode is canceled when the power is turned ON or reset.

(3) A program error (P484) will occur if any axis commanded during cylindrical interpolation has not completed the reference position return.

(4) Tool radius compensation must be canceled before canceling the cylindrical interpolation mode.

(5) When the cylindrical interpolation mode is canceled and switched to the cutting mode, the plane selected before the cylindrical interpolation will be restored.

(6) Program cannot be restarted (program restart) when the block is in the cylindrical interpolation.

(7) A program error (P486) will occur if the cylindrical interpolation command is issued during the mirror image.

(8) When the cylindrical interpolation mode is started or canceled, the deceleration check is performed.

(9) A program error (P481) will occur if the cylindrical interpolation or the polar coordinate interpolation is commanded during the cylindrical interpolation mode.

6.10 Circular Cutting ; G12,G13

Function and purpose

Circular cutting starts the tool from the center of the circle, and cuts the inner circumference of the circle. The tool continues cutting while drawing a circle and returns to the center position.

Command format

G12 I__ D__ F__ ; ... Circular cutting Clockwise (CW)

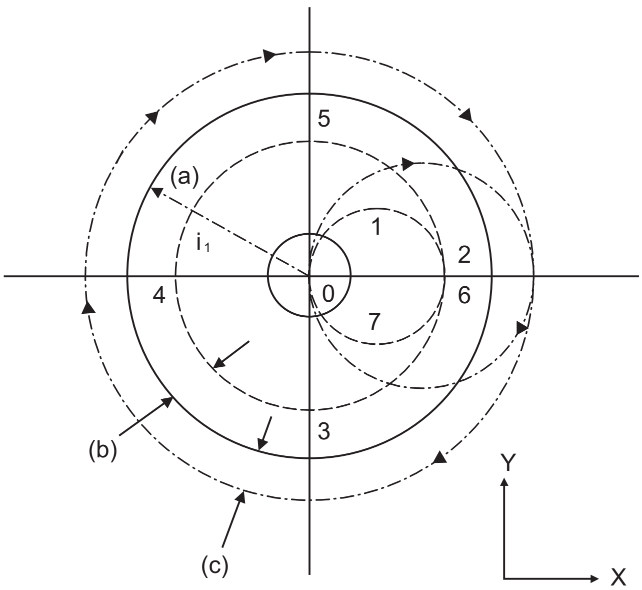
G13 I__ D__ F__ ; ... Circular cutting Counterclockwise (CCW)

I	Radius of circle (incremental value), the sign is ignored
D	Offset No. (The offset No. and offset data are not displayed on the setting and display unit.)
F	Feedrate

Detailed description

(1) The sign + for the offset amount indicates reduction, and - indicates enlargement.

(2) The circle cutting is executed on the plane G17, G18 or G19 currently selected.



————— Compensation amount sign +　　　—·—·—·— Compensation amount sign -

(a) Circle radius　　　(b) d1 offset amount +

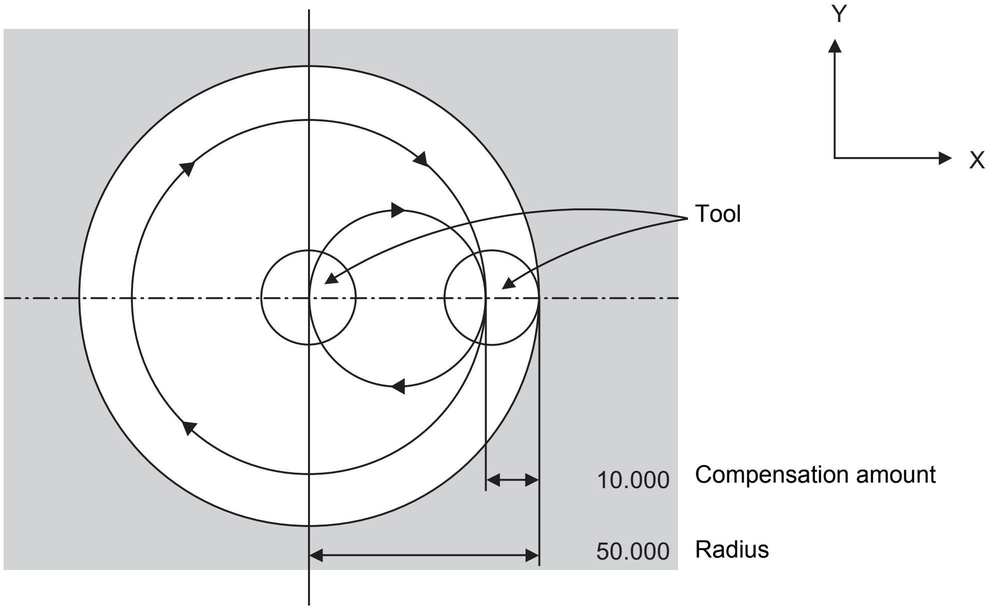
(c) d1 offset amount -

For G12 (tool center path) 0->1->2->3->4->5->6->7->0

For G13 (tool center path) 0->7->6->5->4->3->2->1->0

Program example

(Example 1) G12 I50.000 D01 F100 ; When compensation amount is +10.000mm



Precautions

(1) If the offset No. "D" is not issued or if the offset No. is illegal, the program error (P170) will occur.

(2) If [Radius (I) - offset amount] is 0 or negative, the program error (P223) will occur.

(3) If G12 or G13 is commanded during radius compensation (G41, G42), the radius compensation will be validated on the path after compensated with the D, commanded with G12 or G13.

(4) If an address not included in the format is commanded in the same block as G12 and G13, the program error (P32) will occur.
But when the parameter "#11034 Circular cutting command address check type" is set to "1", it operates as follows;

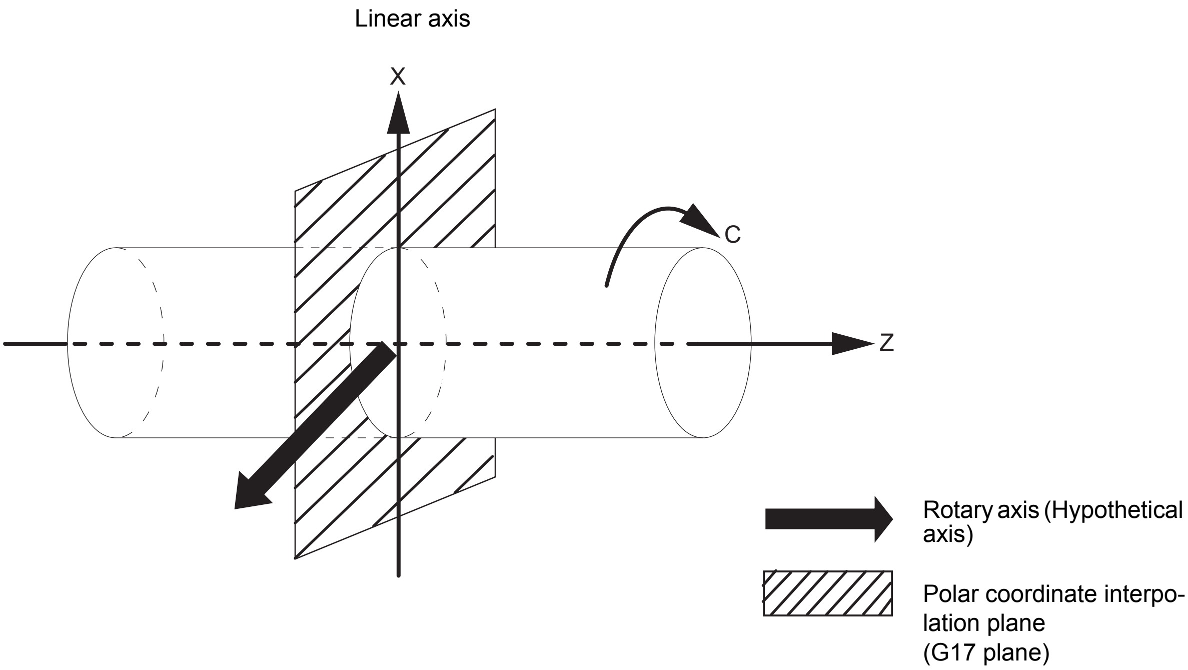
(a) Program error will not occur except for an "H" command.

(b) Only "D","F","I" and "M","S","T","B" will be valid.

6.11 Polar Coordinate Interpolation ; G12.1,G13.1/G112,G113

Function and purpose

This function converts the commands programmed with the orthogonal coordinate axis into linear axis movement (tool movement) and rotary axis movement (workpiece rotation), and controls the contour.
The plane that uses the linear axis as the plane's 1st orthogonal axis, and the intersecting hypothetical axis as the plane's 2nd axis (hereafter "polar coordinate interpolation plane") is selected. Polar coordinate interpolation is carried out on this plane. The workpiece coordinate system zero point is used as the coordinate system zero point during polar coordinate interpolation.



This is effective for cutting a notch in a linear line to the external diameter of the workpiece, for cutting cam shafts, etc.

Command format

Polar coordinate interpolation mode start

G12.1 ;

Polar coordinate interpolation mode cancel

G13.1 ;

Detailed description

(1) The coordinate commands in the interval from the start to cancellation of the polar coordinate interpolation mode will be polar coordinate interpolation.

G12.1 ; Polar coordinate interpolation mode start (Polar coordinate interpolation will start.)
(The coordinate commands in this interval will be the polar coordinate interpolation.)

:

G13.1 ; Polar coordinate interpolation mode cancel (Polar coordinate interpolation is canceled.)

(2) G112 and G113 can be used instead of G12.1 and G13.1.

(3) Command G12.1,G13.1 in an independent block. If it is commanded in the same block with other G code, a program error (P33) will occur.

(4) Linear interpolation or circular interpolation can be commanded during the polar coordinate interpolation mode.

(5) The coordinate commands can be both an absolute command or incremental command.

(6) Tool radius compensation can be applied on the program command. Polar coordinate interpolation will be executed to the path after it has gone through a tool radius compensation.

(7) Command the tangent speed in the polar coordinate interpolation plane (orthogonal coordinate system) by F. F is in mm/min or inch/min unit.

(8) When the G12.1/G13.1 command is issued, the deceleration check is executed.

Plane selection

The linear axis and rotary axis used for polar coordinate interpolation depend on the MTB specifications (parameter #1533).

(1) Determine the deemed plane for carrying out polar coordinate interpolation with the parameter (#1533) of the linear axis used for polar coordinate interpolation.

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

(2) A program error (P485) will occur if the plane selection command (G17 to G19) is issued during the polar coordinate interpolation mode.

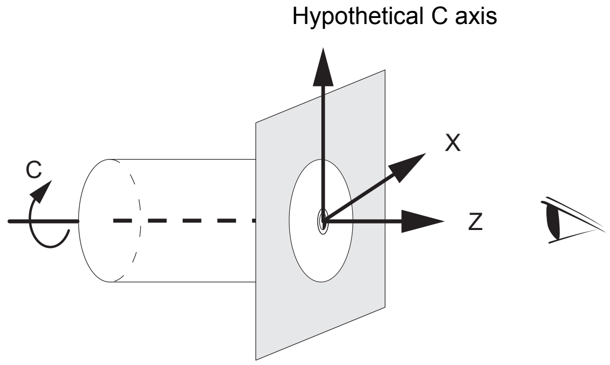
<Note>

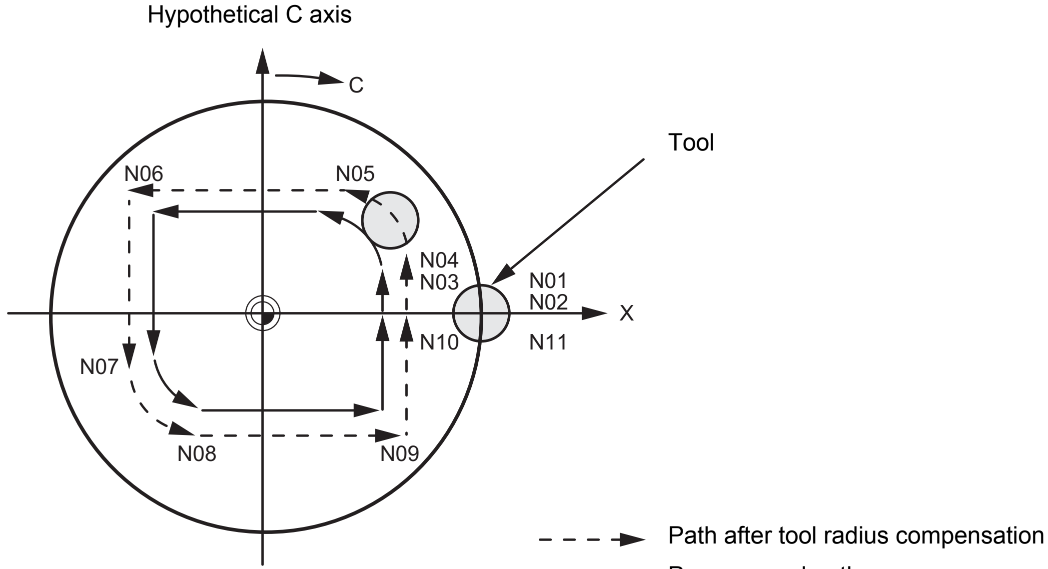
- Depending on the model or version, parameter (#1533) may not be provided. In this case, the operation will be the same as when the parameter (#1533) is blank (no setting).

Related parameter

#1516 mill_ax (Milling axis name)
#1517 mill_c (Milling interpolation hypothetical axis name)
#8111 Milling Radius
#1533 mill_Pax (Polar coordinate linear axis name)

Program example





<Program>

:	
N01 G17 G90 G00 X40.0 C0 Z0;	Setting of start position
N02 G12.1 ;	Polar coordinate interpolation mode: Start
N03 G01 G42 X20.0 F2000;	Actual machining start
N04 C10.0; N05 G03 X10.0 C20.0 R10.0;	
N06 G01 X-20.0;	Shape program
N07 C-10.0;	
N08 G03 X-10.0 C-20.0 I10.0 J0;	(Follows orthogonal coordinate positions on X-C hypothetical axis plane.)
N09 G01 X20.0; N10 C00; N11 G40 X40.0;	
N12 G13.1 ;	Polar coordinate interpolation mode: Cancel
:	
M30 ;	

Relationship with Other Functions

Program commands during polar coordinate interpolation

(1) The program commands in the polar coordinate interpolation mode are issued by the orthogonal coordinate value of the linear axis and rotary axis (hypothetical axis) on the polar coordinate interpolation plane.
The axis address of the rotary axis (C) is specified as the axis address for the plane's 2nd axis (hypothetical axis) command.
The command unit is not deg (degree). The same unit (mm or inch) as used for the command by the axis address of the plane's 1st axis (linear axis) will be used.

(2) The hypothetical axis coordinate value will be set to "0" when G12.1 is commanded. In other words, the position where G12.1 is commanded will be interpreted as angle = 0, and the polar coordinate interpolation will start.

Circular interpolation on polar coordinate plane

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

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

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

Note

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

Tool radius compensation

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

(1) Command the plane selection in the same manner as polar coordinate interpolation.
When conducting tool radius compensation, it must be started up and canceled during the polar coordinate interpolation mode.

(2) A program error (P485) will occur if polar coordinate interpolation is executed during tool radius compensation.

(3) If the G12.1 and G13.1 commands are issued with no movement command after the tool radius compensation is canceled, the position of the axis in the G12.1 and G13.1 commands block is interpreted as the position applied after the tool radius compensation is canceled and the following operations are performed.

Cutting asynchronous feed

(1) The asynchronous mode is forcibly set when the polar coordinate interpolation mode is started.

(2) When the polar coordinate interpolation mode is canceled, the synchronous mode will return to the state before the polar coordinate interpolation mode was started.

(3) A program error (P485) will occur if G12.1 is commanded in the constant surface speed control mode (G96).

Miscellaneous function

(1) The miscellaneous function (M) and 2nd miscellaneous function can be issued in the polar coordinate interpolation mode.

(2) The S command in the polar coordinate interpolation mode specifies the rotary tool's rotation speed instead of the spindle rotation speed.

Tool length compensation

(1) Program error (P481) will occur if tool length compensation is performed in the polar coordinate interpolation mode.

```
:
G43 H12 ;          Tool length compensation before polar coordinate interpolation -> Valid
G00 X100. Z0. ;
G12.1 ;
:
G43 H11 ;          Tool length compensation in polar coordinate interpolation mode -> Pro-
                   gram error
:
G13.1 ;
```

(2) Complete the tool compensation operation (movement of tool length and wear compensation amount) before executing the polar coordinate interpolation.
If the tool compensation operation is not completed when the polar coordinate interpolation start command has been issued, the followings will occur:

- Machine coordinate will not change even if G12.1 is executed.
- When G12.1 is executed, the workpiece coordinate will change to that of the post tool length compensation.

(Even if polar coordinate interpolation is canceled, this workpiece coordinate will not be canceled.)

F command during polar coordinate interpolation

As for the F command during polar coordinate interpolation mode, whether to use the previous F command depends on the previous mode of the feed per minute command (G94/G98) or feed per rotation command (G95/G99).

(1) When G94 (G98) is commanded just before G12.1
If there is no F command in the polar coordinate interpolation, the previous F command feedrate will be used.
After the polar coordinate interpolation mode is canceled, the F command feedrate set at the start of the polar coordinate interpolation mode or the last F command feedrate set during polar coordinate interpolation will continue to be the feedrate.

(2) When G95 (G99) is commanded just before G12.1
The previous F command feedrate cannot be used during polar coordinate interpolation. A new F command must be issued.
The feedrate after the polar coordinate interpolation mode is canceled will return to the state before the polar coordinate interpolation mode was started.

[When there is no F command in G12.1]

Previous mode	No F command	After G13.1
G94 (G98)	Previous F is used	(Same as on the left)
G95 (G99)	Program error (P62)	F just before G12.1 is used

[When F is commanded in G12.1]

Previous mode	With F command	After G13.1
G94 (G98)	Commanded F is used	(Same as on the left)
G95 (G99)	Commanded F is used (*1)	F just before G12.1 is used
(*1) Moves with the feed per minute command during G12.1.		

Hole drilling axis in the fixed cycle for drilling command

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

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

Shift amount in the G76 (fine boring) or G87 (back boring) command

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

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

Restrictions and precautions

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

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

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

(2) Program cannot be restarted (program restart) when the block is in the polar coordinate interpolation.

(3) Before commanding polar coordinate interpolation, set the workpiece coordinate system so that the center of the rotary axis is at the coordinate system zero point. Do not change the coordinate system during the polar coordinate interpolation mode. (G50, G52, G53, relative coordinate reset, G54 to G59, etc.)

(4) The feedrate during polar coordinate interpolation will be the interpolation speed on the polar coordinate interpolation plane (orthogonal coordinate system).
(The relative speed with the tool will vary according to the polar coordinate conversion.)
When passing near the center of the rotary axis on the polar coordinate interpolation plane (orthogonal coordinate system), the rotary axis side feedrate after polar coordinate interpolation will be very high.

(5) The axis movement command outside of the plane during polar coordinate interpolation will move unrelated to the polar coordinate interpolation.

(6) The current position displays during polar coordinate interpolation will all indicate the actual coordinate value. However, the "remaining movement amount" indicates the movement amount on the polar coordinate input plane.

(7) The polar coordinate interpolation mode is canceled when the power is turned ON or reset.

(8) A program error (P484) will occur if any axis commanded during polar coordinate interpolation has not completed the reference position return.

(9) Tool radius compensation must be canceled before canceling the polar coordinate interpolation mode.

(10) When the polar coordinate interpolation mode is canceled and switched to the cutting mode, the plane selected before the polar coordinate interpolation will be restored.

(11) A program error (P486) will occur if the polar coordinate interpolation command is issued during the mirror image.

(12) A program error (P481) will occur if the cylindrical interpolation or the polar coordinate interpolation is commanded during the polar coordinate interpolation mode.

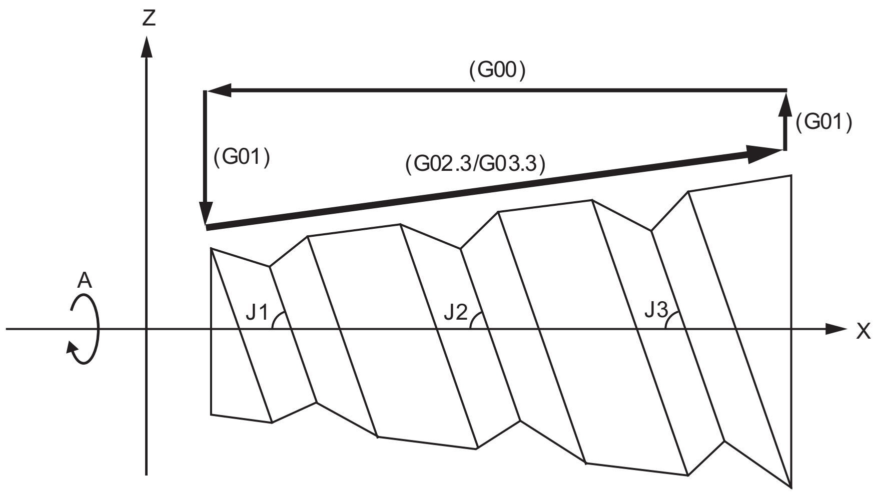
(13) During polar coordinate interpolation, if X axis moveable range is controlled in the plus side, X axis has to be moved to the plus area that includes "0" and above before issuing the polar coordinate interpolation command. If X axis moveable range is controlled in the minus side, X axis has to be moved to the minus area that does not include "0" before issuing the polar coordinate interpolation command.

6.12 Exponential Interpolation ; G02.3,G03.3

Function and purpose

Exponential function interpolation changes the rotary axis into an exponential function shape in respect to the linear axis movement.
At this time, the other axes carry out linear interpolation between the linear axis.
This allows a machining of a taper groove with constant torsion angle (helix angle) (uniform helix machining of taper shape).
This function can be used for slotting or grinding a tool for use in an end mill, etc.

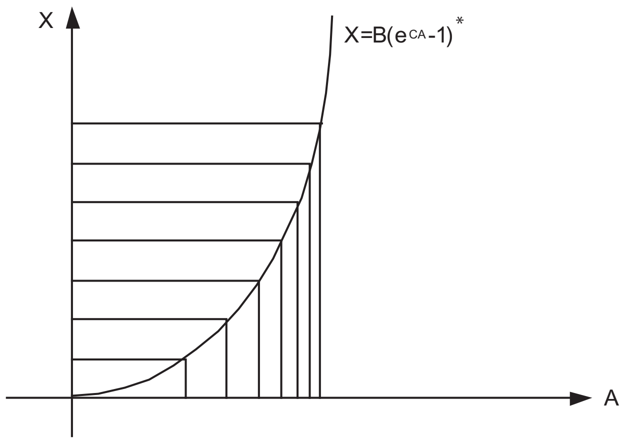
⬩Uniform helix machining of taper shape



A: A axis (rotation axis)
X: X axis (linear axis)

Torsion angle: J1 = J2 = J3

⬩Relation of linear axis and rotary axis



A: A axis (rotation axis)
X: X axis (linear axis)
* : {B, C... constant}

Command format

Forward rotation interpolation (Modal)

G02.3 Xx1 Yy1 Zz1 Ii1 Jj1 Rr1 Ff1 Qq1 Kk1 ;

Backward rotation interpolation (Modal)

G03.3 Xx1 Yy1 Zz1 Ii1 Jj1 Rr1 Ff1 Qq1 Kk1 ;

X	X axis end point (*1)
Y	Y axis end point (*1)
Z	Z axis end point (*1)
I	Angle i1 (*2)
J	Angle j1 (*2)
R	Constant value r1 (*3)
F	Initial feedrate (*4)
Q	Feedrate at end point (*5)
K	Command will be ignored.

(*1) Designate the end point of the linear axis specified by parameter "#1514 expLinax" and the axis that carries out linear interpolation between that axis.
If the end point on of the rotary axis designated with parameter "#1515 expRotax" is specified, linear interpolation without exponential function interpolation will take place. These parameter settings depend on the MTB specifications.

(*2) The command unit is as follows.

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

The command range is -89 to +89°.
A program error (P33) will occur if there is no address I or J command.
A program error (P35) will occur if the address I or J command value is 0.

(*3) The command unit is as follows.

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

The command range is a positive value that does not include 0.
A program error (P33) will occur if there is no address R command.
A program error (P35) will occur if the address R command value is 0.

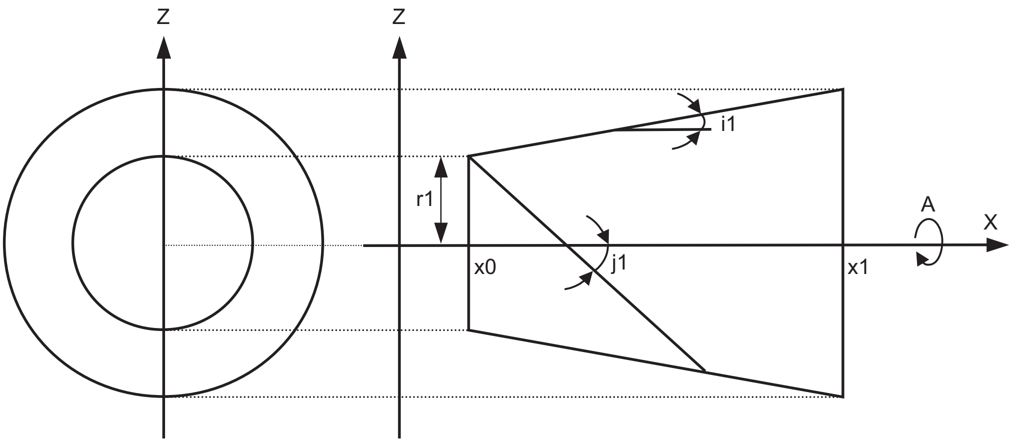
(*4) The command unit and command range is the same as the normal F code. (Command as per minute feed.)
Command the composite feedrate that includes the rotary axis.
The normal F modal value will not change by the address F command.
A program error (P33) will occur if there is no address F command.
A program error (P35) will occur if the address F command value is 0.

(*5) The command unit is as follows.

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

The command unit and command range is the same as the normal F code.
Command the composite feedrate that includes the rotary axis.
The normal F modal value will not change by the address Q command.
The axis will interpolate between the initial speed (F) and end speed (Q) in the CNC according to the linear axis.
If there is no address Q command, interpolation will take place with the same value as the initial feedrate (address F command). (The start point and end point feedrates will be the same.)
A program error (P35) will occur if the address Q command value is 0.

[Example of uniform helix machining of taper shape]



X : X axis (linear axis)   A : A axis (rotation axis)   x0 : linear axis (X axis) start point

Detailed description

Relational expression of exponential function

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

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

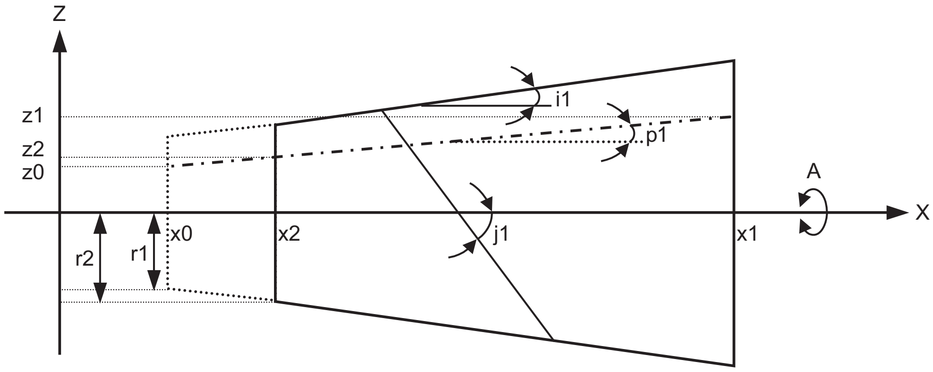
$A(\theta) = (-1)^{\omega} * 360 * \theta / (2\pi)$ Rotary axis (A) movement

where, "D" is as follows.
D = tan(j1) / tan(i1)
During forward rotation (G02.3): ω = 0

During backward rotation (G03.3): ω = 1
θ is the rotation angle (radian) from the rotary axis' start point. The rotary axis' rotation angle (θ) is as follows according to expression (1).
θ = D * 1n{(X * tan(i1) / r1) + 1}

Machining example

[Uniform helix machining of taper shape]



<Relational expression of exponential function in machining example>

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

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

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

where, "D" is as follows.

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

Z(θ)	Absolute value from zero point of Z axis (axis that linearly interpolates with linear axis (X axis))
X(θ)	Absolute value from X axis (linear axis) start point
A(θ)	Absolute value from A axis (rotary axis) start point
r1	Exponential function interpolation constant value (address R command)
r2	Workpiece left edge radius
x2	X axis (linear axis) position at the left edge of the workpiece
x1	X axis (linear axis) end point (address X command)
x0	X axis (linear axis) start point (Set as "x0 ≤ x1" so that workpiece does not interfere with the tool)
z1	End point of Z axis (axis that linearly interpolates between interval with linear axis (X axis)) (address Z command)
z0	Start point of Z axis (axis that linearly interpolates between interval with linear axis (X axis))
i1	Taper gradient angle (address I command)
p1	Slot base gradient angle
j1	Torsion angle (helix angle) (address J command)
ω	Torsion direction (0: Forward rotation, 1: reverse direction)
θ	Workpiece rotation angle (radian)
f1	Initial feedrate (address F command)
q1	Feedrate at end point (address Q command)
k1	Insignificant data (address K command)

According to expressions (1) and (2), expression (3) is obtained.

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

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

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

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

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

The taper gradient angle (i1) and torsion angle (j1) are set by the command address I and J, respectively.

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

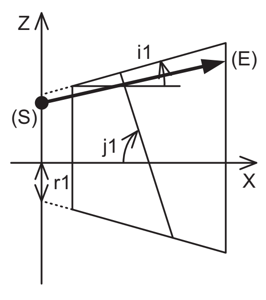
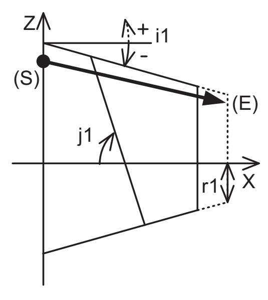
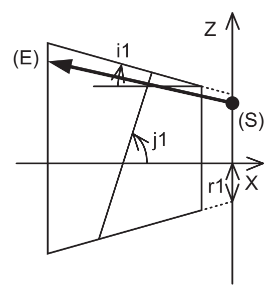
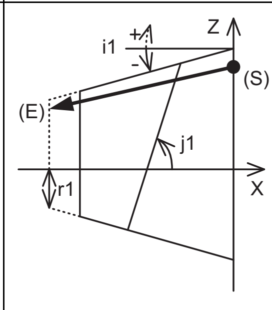
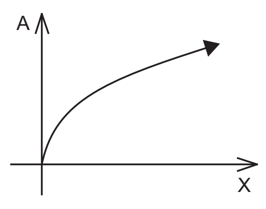
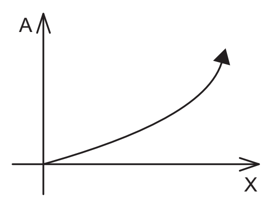
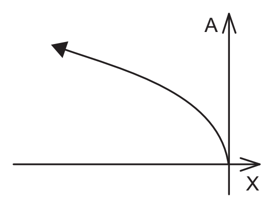
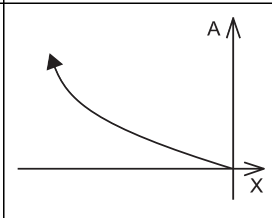
The torsion direction (ω) is changed by the G code. (Forward rotation when G02.3 is commanded, negative rotation when G03.3 is commanded)

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

Command and operation

(1) G2.3 (equivalent to G3.3 if j1 < 0)

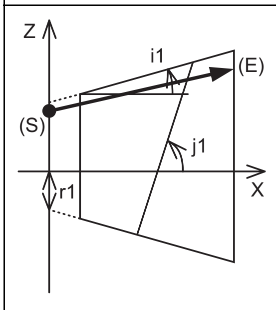
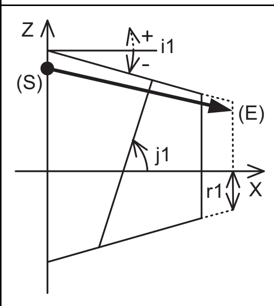
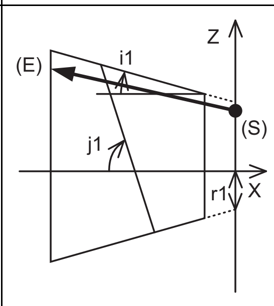
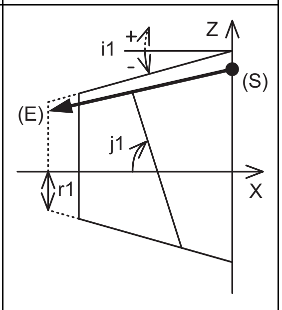
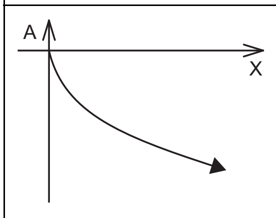
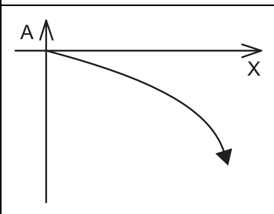
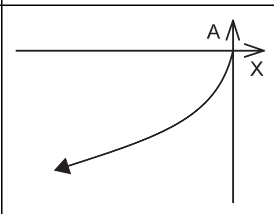
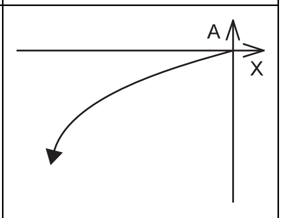
In the conditional figure below, the upper side shows a command, and the lower side shows an operation.

X movement direction > 0		X movement direction < 0	
i1 > 0	**i1 < 0**	**i1 > 0**	**i1 < 0**
			
			
N10 G28XYZC; N20 G91G0 X100. Z100.; N30 G2.3 X100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X100. Z200.; N30 G2.3 X100. Z-100. I-50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z100.; N30 G2.3 X-100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z200.; N30 G2.3 X-100. Z-100. I-50. J80. R105. F500.; N40 M30;

(S) Start point, (E) End point

(2) G3.3 (equivalent to G2.3 if j1 < 0)

In the conditional figure below, the upper side shows a command, and the lower side shows an operation.

X movement direction > 0		X movement direction < 0	
i1 > 0	**i1 < 0**	**i1 > 0**	**i1 < 0**
			
			
N10 G28XYZC; N20 G91G0 X100. Z100.; N30 G3.3 X100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X100. Z200.; N30 G3.3 X100. Z-100. I-50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z100.; N30 G3.3 X-100. Z100. I50. J80. R105. F500.; N40 M30;	N10 G28XYZC; N20 G91G0 X-100. Z200.; N30 G3.3 X-100. Z-100. I-50. J80. R105. F500.; N40 M30;

(S) Start point, (E) End point

Precautions

(1) When G02.3/G03.3 is commanded, interpolation takes place with the exponential function relational expression using the start position of the linear axis and rotary axis as 0.

(2) Linear interpolation will take place in the following cases, even if in the G02.3/G03.3 mode.
The feedrate for linear interpolation will be the F command in that block. (Note that the normal F modal is not updated.)
- The linear axis designated with the parameter (#1514 expLinax) is not commanded, or the movement amount for that axis is 0.
- The rotary axis designated with the parameter (#1515 expRotax) is commanded.

(3) A program error will occur if the following commands are issued during the G02.3/G03.3 mode.
A program error will also occur if G02.3 or G03.3 command is issued in the following modes.
- Tool length compensation (A program error will occur only when the compensation starts at the same time as the movement by exponential function interpolation.
- The tool length compensation will operate normally if it has started before the G02.3/G03.3 mode starts.)
- Tool radius compensation
- High-speed High-accuracy Control
- High-speed machining
- Scaling
- Tool length compensation along the tool axis
- Figure rotation
- Coordinate rotation by program
- Coordinate rotation by parameter
- 3-dimensional coordinate conversion

(4) A program error (P481) will occur if commands are issued during the pole coordinate interpolation, cylindrical interpolation or milling interpolation modes.

(5) Program error (P612) will occur if commands are issued during the scaling or mirror image.

(6) Program error (P34) will occur if commands are issued during the high-speed high-accuracy control II.

(7) G02.3/G03.3 will function with asynchronous feed even during the synchronous feed mode, and the synchronous feed mode will be canceled.

(8) If the parameter "#1515 expRota" setting is the same axis name as the initial C axis, the axis selected with the C axis selection signal will interpolate as the rotary axis.

6.13 Polar Coordinate Command ; G16

Function and purpose

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

Command format

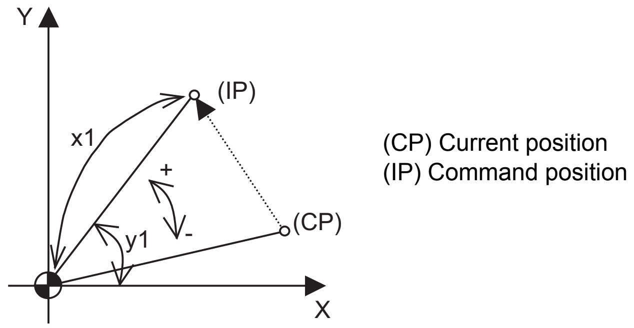
G16 ; ... Polar coordinate command mode ON

G15 ; ... Polar coordinate command mode OFF

Detailed description

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

G1x ;	Plane selection for polar coordinate command (G17/G18/G19)
G16 ;	Polar coordinate command mode ON
G9x G01 Xx1 Yy1 F2000 ; :	Polar coordinate command G9x ; Center selection for polar coordinate command (G90/G91) G90: The workpiece coordinate system zero point is the polar coordinate center. G91: The current position is the polar coordinate center. x1 : 1st axis for the plane : The radius of the polar coordinate commanded y1 : 2nd axis for the plane : The angle of the polar coordinate commanded
G15 ;	Polar coordinate command mode OFF



For G90/G17 (X-Y plane)

(2) The plane selection during the polar coordinate command mode is carried out with G17, G18 and G19.

(3) The polar coordinate command is a modal. The polar coordinate command mode when the power is turned ON is off (G15). Whether to initialize the modal at reset or not can be selected with the parameter (#1210 RstGmd/ bit 11) setting.

(4) During polar coordinate command mode, command the radius with the 1st axis for the selected plane, and the angle with the 2nd axis. For example, when the X-Y plane is selected, command the radius with the address "X", and the angle with the address "Y".

(5) For the angle, the counterclockwise direction of the selected plane is positive and the clockwise direction is negative.

(6) The radius and angle can be commanded with both the absolute value (G90) and incremental value (G91).

(7) When the radius is commanded with the absolute value, command the distance from the zero point in the workpiece coordinate system (note that when the local coordinate system is set, command the distance in the local coordinate system).

(8) When the radius is commanded with the incremental value command, considering the end point of the previous block as the polar coordinate center, command the incremental value from that end point. The angle is commanded with the incremental value of the angle from the previous block.

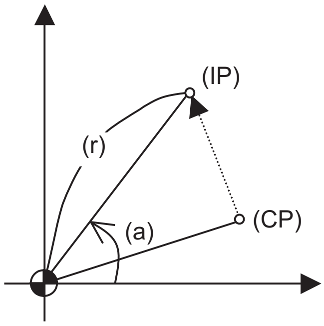
(9) When the radius is commanded with the negative value, the same operation as the command that the radius command value is changed to the absolute value and 180° is added to the angle command value.

Commanded position

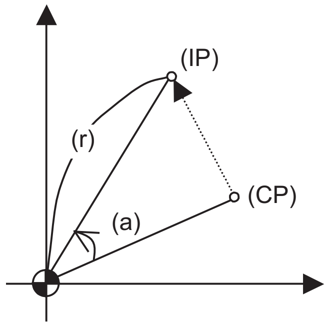
(1) When the zero point in the workpiece coordinate system is applied to the polar coordinate center

The zero point in the workpiece coordinate system is applied to the polar coordinate center by commanding the radius value with the absolute value.

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



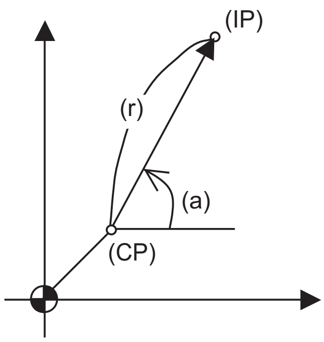
When the angle is the absolute value command



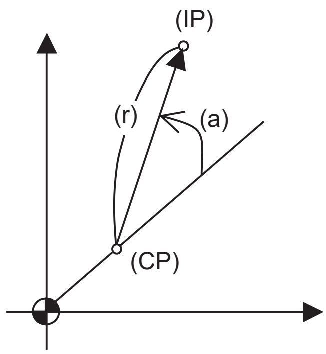
When the angle is the incremental value command

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

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



When the angle is the absolute value command

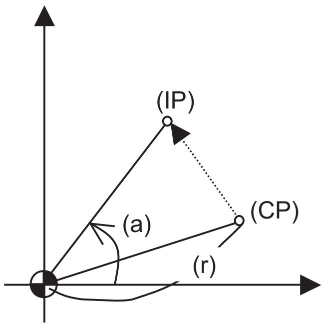


When the angle is the incremental value command

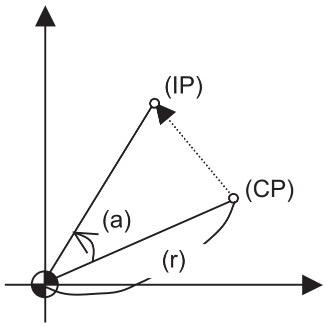
(CP) Current position	(IP) Command position
(a) Angle	(r) Radius

(3) When the radius value command is omitted

When the radius value command is omitted, the zero point in the workpiece coordinate system is applied to the polar coordinate center, and the distance between the polar coordinate center and current position is regarded as the radius. Note that the zero point in the local coordinate system is applied to the polar coordinate center if the local coordinate system (G52) is used.



When the angle is the absolute value command

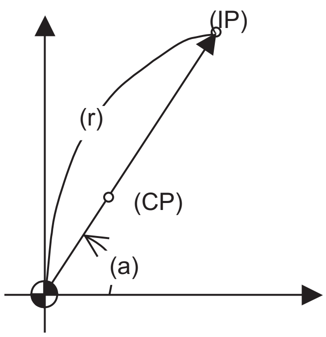


When the angle is the incremental value command

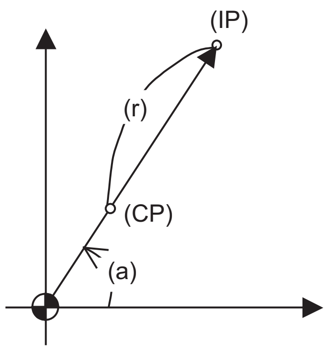
(4) When the angle command is omitted

When the angle command is omitted, the angle of the present position in the workpiece coordinate system is applied to the angle command.
The zero point in the workpiece coordinate system is applied to the polar coordinate center by commanding the radius value with the absolute value. Note that the zero point in the local coordinate system is applied to the polar coordinate center if the local coordinate system (G52) is used.

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



When the angle is the absolute value command



When the angle is the incremental value command

(CP) Current position
(a) Angle

(IP) Command position
(r) Radius

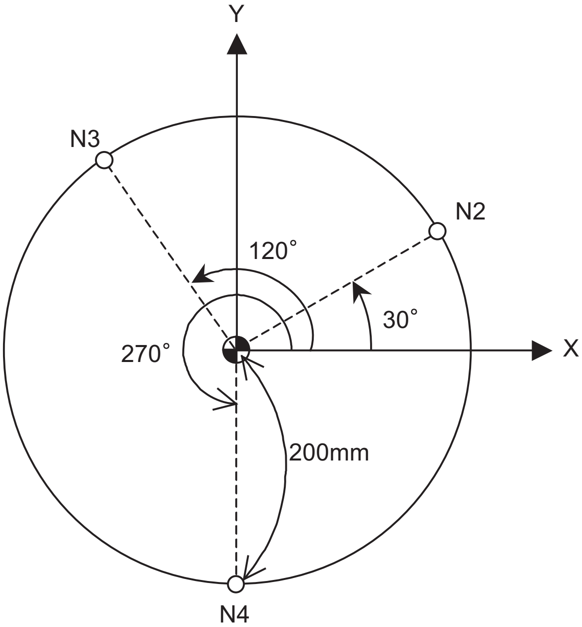
Axis command not interpreted as polar coordinate command

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

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

Program example

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



- The polar coordinate zero point is the zero point in the workpiece coordinate system.- The plane is the X-Y plane.

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

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

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

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

Precautions

(1) If the high-accuracy commands are carried out during the polar coordinate command mode, or if the high-accuracy command is carried out during the following command mode, operations are performed depending on your machine's specifications.
Refer to "High-accuracy Control" and "High-speed High-accuracy Control" for details.

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

6.14 Spiral/Conical Interpolation ; G02.1/G03.1(Type1), G02/G03(Type2)

Function and purpose

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

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

Command format

Spiral/conical interpolation (Type 1: #1272 ext08/bit2=0)

G17 G02.1/G03.1 X__ Y__ I__ J__ P__ F__;

Spiral/conical interpolation (Type 2: #1272 ext08/bit2=1)

G17 G02/G03 X__ Y__ I__ J__ Q__/L__/K__ F__;

G17	Arc plane
G02.1/G03.1(Type1)	Arc rotation direction (Type 1)
G02/G03(Type2)	Arc rotation direction (Type 2)
X Y	End point coordinates (Conical Interpolation when the axis other than arc plane axes is included.)
I J	Arc center
P (Type 1)	Number of pitches (number of spirals) (Type 1)
Q/L/K (Type 2)	Incremental-decremental amount of radius /Number of pitches(Number of spirals)/ Increment-decrement amount of height (Type 2)
F	Feedrate (tool path direction speed)

Circular interpolation operations are carried out at the f1 speed by the commands above.

The path is toward the end point, following a spiral arc path centered at the position designated by distance i (X axis direction) and distance j (Y axis direction) in respect to the start point.

(1) The arc plane is designated by G17, G18 and G19. (Common for type 1 and 2)

G17	XY plane
G18	ZX plane
G19	YZ plane

(2) The arc rotation direction is designated by G02.1(G02) or G03.1(G03). (Common for type 1 and 2)

G02.1/G02	Clockwise (CW)
G03.1/G03	Counterclockwise (CCW)

(3) The end point coordinates are designated with XYZ. (Common for type 1 and 2)
(Decimal point command is possible. Use mm (or inch) as the unit.)
When designation of arc plane axes is omitted, the coordinates of the start point are inherited.
If the axis other than arc plane axes is designated, conical interpolation is applied.

(4) The arc center is designated with IJK. (Common for type 1 and 2)
(Decimal point command is possible. Use mm (or inch) as the unit.)
I : Incremental designation in the X axis direction from the start point
J : Incremental designation in the Y axis direction from the start point
K : Incremental designation in the Z axis direction from the start point
When either 1 axis of arc plane is omitted, the coordinates of the start point are inherited.

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

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

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

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

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

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

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

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

The number of rotations when the designation of increment or decrement amount of height is specified can be calculated with the following expression.
Number of rotations = Height / | Increment/decrement amount of height |
If Q, K and L have been designated at the same time, the order of precedence is Q>K>L.
Decimal point command is possible in the range of the increment or decrement amount of radius and height. Use mm (or inch) as the unit.

Detailed description

(1) The arc rotation direction G02.1 is the same as G02, and G03.1 is the same as G03.

(2) There are no R-designated arcs in spiral interpolation.

(3) Conical cutting, tapered thread-cutting and other such machining operations can be conducted by changing the start point and end point radius and commanding the linear axis simultaneously.

(4) Normally the spiral interpolation is automatically enabled with the arc commands (G02, G03) when the difference between the start point radius and the end point radius is less than the parameter setting value.

(5) The axis combination that can be simultaneously commanded depends on the specifications. The combination within that range is arbitrary.

(6) The feedrate is the constant tangent speed.

(7) Simultaneous control by combining with tool radius compensation (G41, G42) is not possible.

(8) The arc plane always follows G17, G18 and G19. The plane arc control is carried out by G17, G18 and G19, even if designated by two addresses that do not match the plane.

(9) Conical interpolation When an axis designation other than the spiral interpolation plane is simultaneously designated, other axes are also interpolated in synchronization with the spiral interpolation.

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

(a) Items common for type 1 and 2

Setting Items	Command range (unit)	Error
End point coordinates	Range of coordinate command (mm/inch) (Decimal point command is possible.)	• If a value exceeding the command range is issued, a program error (P35) will occur. • If an axis other than one which can be controlled with the command system is commanded, a program error (P33) will occur.
Arc center	Range of coordinate position command (mm/inch) (Decimal point command is possible.)	• If a value exceeding the command range is issued, a program error (P35) will occur. • If an axis other than one which can be controlled with the command system is commanded, a program error (P33) will occur. • If rotation plane axis is not designated completely, a program error (P33) will occur.
Number of pitches	0 to 99	• If a value exceeding the command range is issued, a program error (P35) will occur.
Feedrate	Range of speed command (mm/min, inch/min) (Decimal point command is possible.)	• If a value exceeding the command range is issued, a program error (P35) will occur.

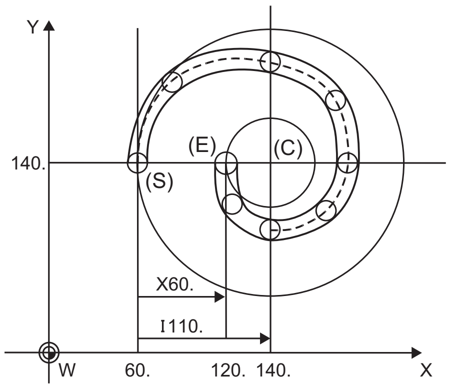
(b) Items for type 2 only

Setting Items	Command range (unit)	Error
Increment or decrement amount of radius	Range of coordinate position command (mm/inch) (Decimal point command is possible.)	•If the sign of designated increment or decrement amount is opposite from that of the difference between the start point radius and the end point radius, a program error (P33) will occur. •If the end point position obtained from the speed and increment or decrement amount is larger than "SpiralEndErr (#8075)", a program error (P70) will occur.
Increment or decrement amount of height	Range of coordinate position command (mm/inch) (Decimal point command is possible.)	•If the sign of designated increment or decrement amount is opposite from that of the movement direction of height, a program error (P33) will occur. •If the end point position obtained from the speed and increment or decrement amount is larger than "SpiralEndErr (#8075)", a program error (P70) will occur.
G02.1/0G3.1		•Program error (P34) will occur if G02.1/G03.1 are used during type 2.

Program example

(Example 1)

```
G91 G17 G01 X60. F500 ;
Y140. ;
G2.1 X60. Y0  I100. P1  F300 ;
G01 X-120.;
G90
G17  G01  X60. F500 ;
Y140. ;
G2.1   X120. Y140. I100. P1   F300 ;
G01 X0 ;
```



(S) Start point   (E) End point   (C) Center

(Example 2)

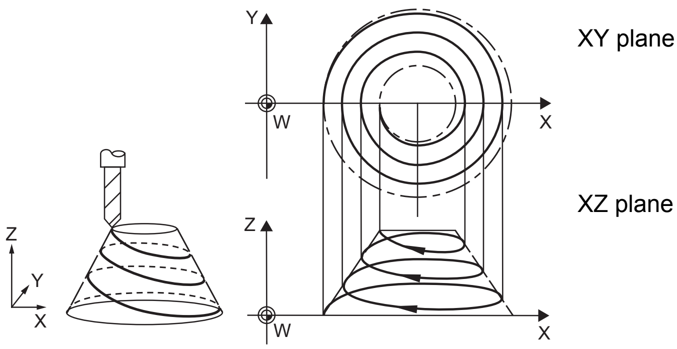
```
G91 G17 G01 X60. F500 ;
Y140. ;
G02.1X60.0 Z100.0 I100. P1  F300 ;
G01X-120  ;
```

-> Because this is the G17 plane, arc control is not carried out by X-Z.

Arc control is carried out by X-Y.

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

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



Relation with other functions

(1) Items common for type 1 and 2

- As the start point and end point are not on the same arc, a normal line control will not be applied correctly.
- If there is no center command when geometric is valid, a program error (P33) will occur.

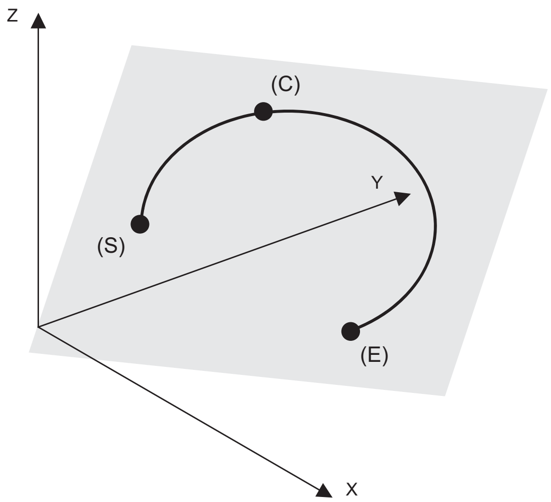
(2) Items for type 2 only

- If the spiral interpolation command is issued during the mirror image, a program error (P34) will occur.
- If the spiral interpolation command is issued during the scaling, a program error (P34) will occur.
- If the spiral interpolation command is issued during the corner chamfering/corner rounding command, a program error (P33) will occur.

6.15 3-dimensional Circular Interpolation ; G02.4,G03.4

Function and purpose

To issue a circular command over a 3-dimensional space, an arbitrary point (intermediate point) must be designated on the arc in addition to the start point (current position) and end point. By using the 3-dimensional circular interpolation command, an arc shape determined by the three points (start point, intermediate point, end point) designated on the 3-dimensional space can be machined.
To use this function, the optional specification is required. If the option is not provided and the 3-dimensional circular interpolation command is issued, a program error (P39) will occur.



(S) Start point (Current position)
(E) End point
(C) Intermediate point

Command format

G02.4 (G03.4)	**Xx1 Yy1 Zz1 αα1 ;**	**Intermediate point designation (1st block)**
	Xx2 Yy2 Zz2 αα2 ;	**End point designation (2nd block)**

G02.4 (G03.4)	3-dimensional circular interpolation command (Cannot designate the rotation direction)
x1, y1, z1	Intermediate point coordinates
x2, y2, z2	End point coordinates
α	Arbitrary axis other than axis used as the reference (X,Y,Z) in 3-dimensional circular interpolation (Cannot be omitted.)

(1) The G02.4 and G03.4. operations are the same. (Cannot designate the rotation direction)

(2) The axes used as the reference in 3-dimensional circular interpolation are the three basic axes set with the parameters.

(3) The X, Y, Z address in the block may be omitted. The intermediate point coordinates omitted in the 1st block become the start point coordinates, and the end point coordinates omitted in the 2nd block become the intermediate point coordinates.

(4) When using the 3-dimensional circular interpolation command, an arbitrary axis can be commanded in addition to the orthogonal coordinate system (X, Y, Z) used as the reference. The arbitrary axis designated in the intermediate point designating block (1st block) will interpolate to the command point when moving from the start point to intermediate point. The arbitrary axis designated in the end point command block (2nd block) will interpolate to the command point when moving from the intermediate point to the end point. The number of arbitrary axes that can be commanded differs according to the number of simultaneous contour control axes. The total of the basic three axes used as the reference of the 3-dimensional circular interpolation and the arbitrary axes commanded simultaneously must be less than the number of simultaneous contour control axes.

Detailed description

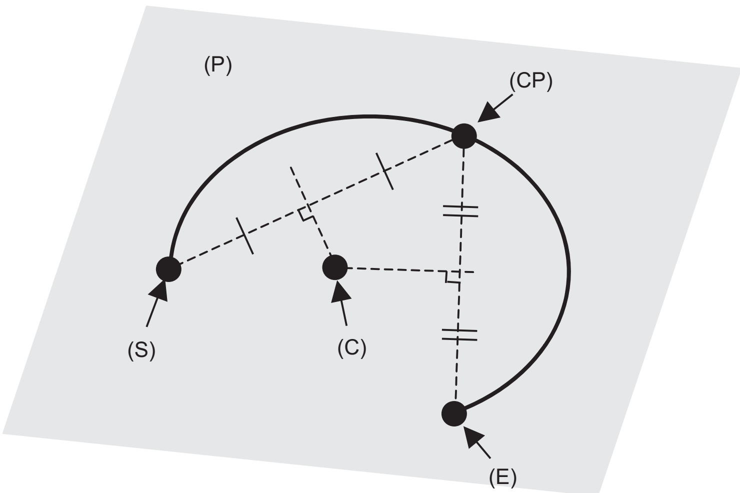
Designating intermediate point and end point

When using the 3-dimensional circular interpolation command, an arc that exists over the 3-dimensional space can be determined by designating the intermediate point and end point in addition to the start point (current position). (Refer to following figure) So according to the command format, it is necessary to designate an intermediate point in the 1st block and an end point in the 2nd block. If only one block is designated, a program error (P74) will occur.

Liner interpolation is applied when the end point match the start point in the 3-dimensional circular interpolation command. (Refer to "When liner interpolation is applied") Thus, a true circle (360-degree rotation) cannot be designated in the 3-dimensional circular interpolation.

In addition, designate the intermediate point in the middle of a start point and an end point. If the intermediate point is near the start point or the end point, arc accuracy may fall.

- Designation of arc in 3-dimensional space



(P) Plane including start point, intermediate point and end point
(S) Start point (Current position)
(E) End point
(CP) Intermediate point
(C) Center

As shown in the above figure, when three points (start point, intermediate point, end point) are specified on 3-dimensional space, arc center coordinates can be obtained. An arc center cannot be obtained if only two points are specified, and a liner interpolation is applied.
If the intermediate point is near the start point or the end point, an error may occur when calculating arc center.

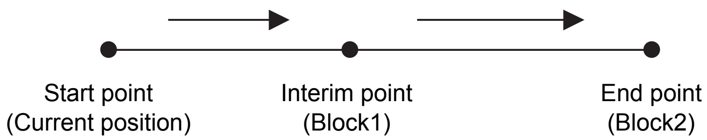
When liner interpolation is applied

In the following case, liner interpolation is applied without executing 3-dimensional circular interpolation.

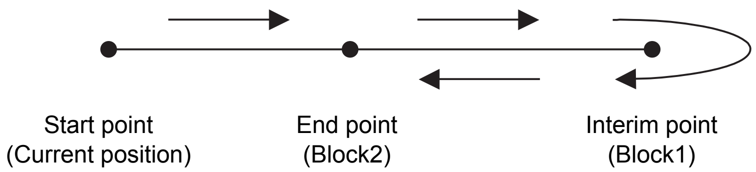
(1) When the start point, intermediate point, and end point are on the same line (refer to the following figure)(If the end point exists between the start point and intermediate point, axes move in the order of start point, intermediate point, and end point.)

(2) When two of the start point, intermediate point and end point match(Linear interpolation is applied even if the end point matches the start point to command true circle.
When the start point matches the end point, axes move in order of the start point, an intermediate point, and an end point.)

- When liner interpolation is applied



When the three points are on the same line, liner interpolation is applied.



Even if the end point exists between the start point and intermediate point, move in the order of start point, intermediate point, and end point.

Modal command

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

Relation with other functions

Commands that cannot be used

G code command which leads to a program error during 3-dimensional circular interpolation modal

G code	Function name	Program error
G05 Pn	High speed machining mode	P34
G05 P10000	High-speed high-accuracy control II	P34
G05.1 Q0/G05.1 Q1	High-speed high-accuracy control I	P34
G07.1	Cylindrical interpolation	P485
G12/G13	Circular cutting CW/CCW	P75
G12.1	Polar coordinate interpolation	P485
G16	Polar coordinate command	P75
G41/G42	Tool radius compensation	P75
G41/G42	3-dimensional tool radius compensation	P75
G41.1/G42.1	Normal line control	P75
G43/G44	Tool length compensation	P75
G51	Scaling	P75
G51.1	Mirror image	P75
G66/G66.1	User Macro	P75
G67	User Macro	P276
G68	Coordinate rotation by program	P75
G68	3-dimensional coordinate conversion	P921
G73/G74/G76/G81/G82 G83/G84/G85/G86/G87 G88/G89	Fixed cycle	P75

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

G code modal	Function name	Program error
G05 Pn	High speed machining mode	P34
G05 P10000	High-speed high-accuracy control II	P34
G05.1 Q1	High-speed high-accuracy control I	P34
G07.1	Cylindrical interpolation	P481
G12.1	Polar coordinate interpolation	P481
G16	Polar coordinate command	P75
G41/G42	Tool radius compensation	P75
G41/G42	3-dimensional tool radius compensation	P75
G41.1/G42.1	Normal line control	P75
G43/G44	Tool length compensation	P75
G51	Scaling	P75
G51.1	Mirror image	P75
G66/G66.1	User Macro	P75
G68	Coordinate rotation by program	P75
G68	3-dimensional coordinate conversion	P922

Functions that cannot be used

If following functions are used in 3-dimensional circular interpolation, alarm will occur.

- Chopping
- Macro interruption
- Mirror image by parameter setting
- Mirror image by external input
- Corner chamfering/corner rounding

Restrictions may be added for other functions. Refer to explanation of each function.

Precautions

(1) If single block is valid and this command is operated, a block stop is carried out at an intermediate point and the end point.

(2) The speed command during 3-dimensional circular interpolation is the tangent speed on arc.

(3) When 3-dimensional circular interpolation is commanded while incremental command is valid, the relative position of the intermediate point in respect to the start point is designated in the intermediate point designation block, and the relative position of the end point in respect to the intermediate point is designated in the intermediate point designation block.

(4) The path of 3-dimensional circular interpolation during graphic check is drawn as linear at each range from start point to intermediate point and from intermediate point to end point.

6.16 NURBS Interpolation ; G06.2

Function and purpose

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

This function operates only in the high-speed high-accuracy control II mode, therefore, the high-speed high-accuracy control II function is also required as the specification.

However, if the curvature is large, the speed is clamped so that the machine's tolerable acceleration rate is not exceeded.

Command format

NURBS interpolation start

```
G06.2  Pp  Kk1 X1 Yy1 Zz1 Rr1 Ff;
           Kk2 Xx2 Yy2 Zz2 Rr2;
           Kk3 Xx3 Yy3 Zz3 Rr3;
           Kk4 Xx4 Yy4 Zz4 Rr4;
             :
           Kkn Xxn Yyn Zzn Rrn;
           Kkn+1;
           Kkn+2;
           Kkn+3;
           Kkn+4;
```

Pp	Set the stage of the NURBS curve. Designate in the same block as G06.2 command. The NURBS curve of the stage p will be (p-1)th curve. When omitted, Pp means the same as P4. (Example) P2: Primary curve (liner)
Kkn	Knot Set the knot for each NURBS interpolation block. Set the same value for the knot in the 1st block to the stage p block. NURBS interpolation is terminated if there is a block exclusively with knot.
Xxn Yyn Zzn	Control point coordinate value. Designate the same coordinate value for the 1st block control point as that designated right before NURBS interpolation.
Rrn	Control point weight. Set the weight of each NURBS interpolation control point.
Ff	Interpolation speed (Can be omitted)

Detailed description

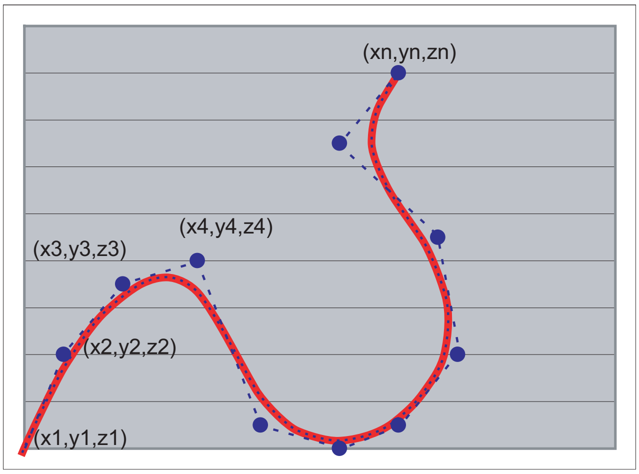
(1) Designate the stage P for the 1st block of NURBS interpolation.

(2) Designate the same coordinate value for the 1st block control point of NURBS interpolation as that designated right before NURBS interpolation.

(3) Designate all axes to be used in the subsequent NURBS interpolation blocks for 1st block of NURBS interpolation.

(4) Set the same value for knot K from the 1st block of NURBS interpolation to setting value block of the stage P.

(5) Command knot K exclusive block of the same number as the setting value of the stage P for terminating NURBS interpolation.
At this time, set the same value for knot K setting.



Passes through control point

NURBS interpolation curve

Note

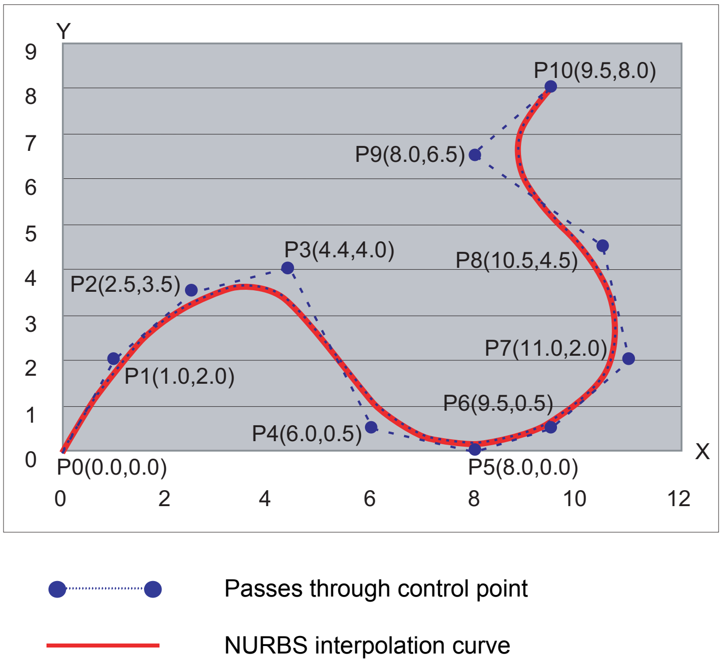
(1) If an exclusive knot is commanded immediately after NURBS interpolation, NURBS interpolation mode is active again.
An exclusive knot that is commanded immediately after NURBS interpolation is the same meaning as following command.
G06.2 Pp Km Xxn Yyn Zzn R1.0

Program example

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

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

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



Relationship with Other Functions

G code/Feed/Miscellaneous functions

All the G code, feedrate and MSTB code cannot be set during NURBS interpolation.
However, when the fixed cycle G code is commanded in the same block where G06.2 is commanded, the fixed cycle G code is ignored.
If a command other than the axis address designated in the 1st block of NURBS interpolation, R and K is commanded, a program error will occur.

Data format

(1) Optional block skip "/"
Cannot be set in the NURBS interpolation 2nd block or after.

(2) Control IN "("and Control OUT ")"
Cannot be set in the NURBS interpolation 2nd block or after.

(3) Local variables and common variables
Can be referred but cannot be set in the NURBS interpolation. Setting the variables causes a program error (P29).

(4) System variables
Cannot be referred nor set in the NURBS interpolation; a program error (P29) will occur.

Interruption/restart

The validity of program interruption/restart is shown below.

Type	During NURBS interpolation
Single block	Valid (*1)
Feed hold	Valid
Resetting	Valid (*2)
Program stop	Invalid
Optional stop	Invalid
Manual interruption	Invalid (*3)
MDI interruption	Invalid
Restart search	Invalid
Macro interruption	Invalid (*4)
PLC interruption	Invalid (*5)

(*1) A single block stop is carried out at the last control points only.
The single block stop is not applied during NURBS interpolation.

(*2) NURBS interpolation mode is canceled with Reset (Reset1/Reset2/Reset&Rewind).

(*3) The operation differs according to the manual absolute signal status.

- When the manual absolute signal OFF, NURBS interpolation is carried out in the state where axis-coordinate system is shifted by the manual absolute movement amount.
- When the manual absolute signal ON
Upon automatic start after manual interruption, a program error (P554) will occur after moving by the remaining distance.
Note that the operation can run continually by returning the axis to the original position after manual interruption.

(*4) "Macro interrupt" signal (UIT) is ignored.

(*5) "PLC interrupt" signal (PIT) is ignored.

Graphic check

NURBS interpolation cannot be applied during graphic check (continuous/step check).
Linear interpolation that connects the control points is applied during graphic check.

High-accuracy Control in 2 part Systems

With the high-accuracy control in 2 part systems specification, NURBS interpolation can be commanded by 1st and 2nd part systems.

Precautions

(1) Target axes for NURBS interpolation are 3 basic axes.

(2) Command the control point for all the axes for which NURBS interpolation is carried out in the 1st block (G06.2 block). A program error (P32) will occur if an axis which was not commanded in the 1st block is commanded in the 2nd block or after.

(3) The first control point (G06.2 block coordinate value) should be commanded as the start point of the NURBS curve. Thus, the start point of the NURBS curve should be commanded to match the end point of the previous block. A program error will occur if the points do not match. (P552)

(4) The command range of the weight is 0.0001 to 99.9999. If "1" is commanded, the resulting command will be equal to "1.0".
If more than 5 digits are commanded after the decimal point, a program error (P33) will occur.

(5) The knot command cannot be omitted, and must be commanded in each block. A program error (P33) will occur if omitted.

(6) As with knot, in the same manner as weight, up to 4 digits can be commanded after the decimal point. Even if the decimal point is omitted, the value will be handled as the one with a decimal point. If "1" is commanded, the result will be the same as "1.0".
If more than 5 digits are commanded after the decimal point, a program error (P33) will occur.

(7) As with knot, command the same or greater value than the previous block. If a smaller value than previous block is set, a program error (P551) will occur.

(8) NURBS interpolation cannot be applied during graphic check (continuous/step check).
Linear interpolation that connects the control points is applied during graphic check.

(9) NURBS interpolation mode is canceled with Reset (Reset1/Reset2/Reset&Rewind).

(10) NURBS interpolation can be commanded in only the following modes. If NURBS interpolation is commanded in other than the following modes, the program error (P29) will occur.

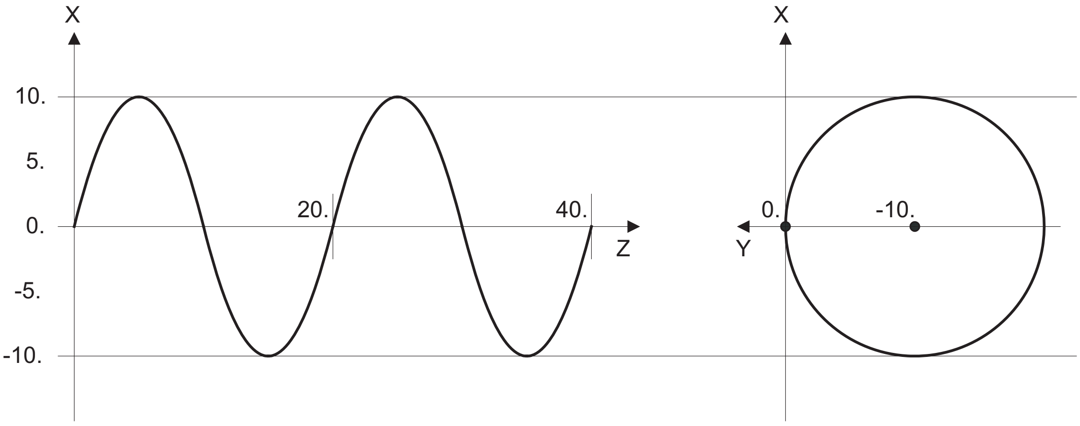
Type	Mode in which NURBS interpolation can be commanded
G group 5	Asynchronous feed (G94)
G group 7	Tool radius compensation cancel (G40)
G group 8	Tool length compensation +/-(G43/G44) Tool length compensation cancel (G49)
G group 9	Fixed cycle cancel (G80)
G group 11	Scaling cancel (G50)
G group 13	High-accuracy control ON (G61.1) Cutting mode (G64)
G group 14	User macro modal call cancel (G67)
G group 15	Normal line control cancel (G40.1)
G group 16	Programmable coordinate rotation mode OFF /3-dimensional coordinate conversion mode OFF (G69.1)
G group 17	Constant surface speed control OFF (G97)
G group 18	Polar coordinate command OFF (G15)
G group 19	G command mirror image cancel (G50.1)
G group 21	Polar coordinate interpolation cancel (G13.1)
-	Not during the coordinate rotation by parameter
-	Not during the mirror image by parameter setting
-	Not during the mirror image by external input

6.17 Hypothetical Axis Interpolation ; G07

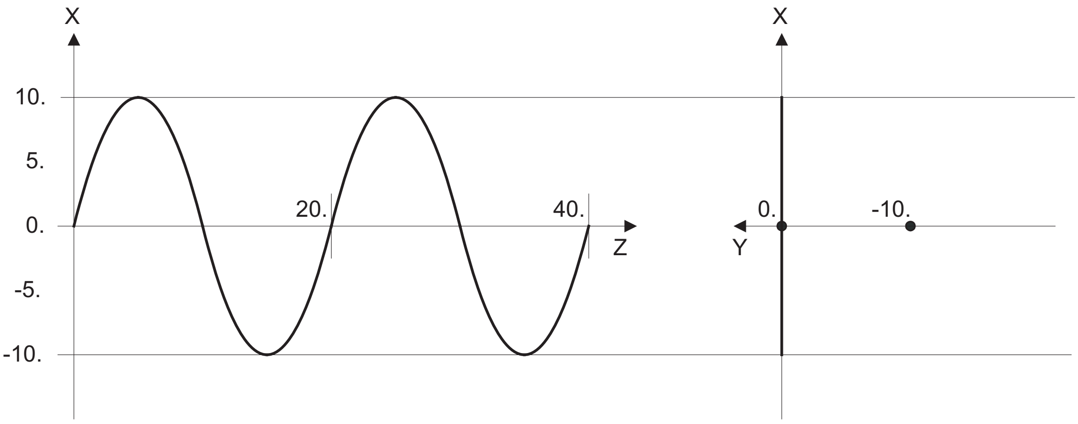
Function and purpose

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

Normal helical interpolation



Helical interpolation in the hypothetical axis interpolation mode



To perform the SIN interpolation on Z-X plane, execute the helical interpolation (Y-X plane: G17 G02) with Y axis, which is designated as the hypothetical axis. The hypothetical axis does not make any actual movement.

Command format

G07 α0 ; ... Hypothetical axis interpolation mode ON

G07 α1 ; ... Hypothetical axis interpolation mode cancel

α	Axis name for which hypothetical axis interpolation is performed.

Detailed description

(1) During “G07 α0 ;” to “G07 α1 ;”, α axis will be the hypothetical axis.

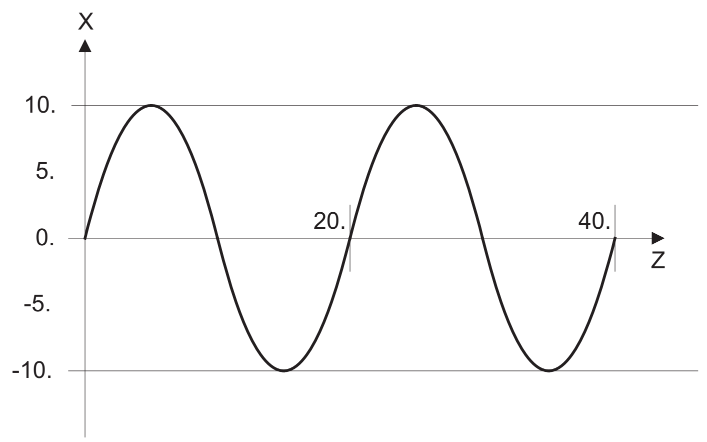
(2) Any axis among the NC axes can be designated as the hypothetical axis.

(3) Multiple axes can be designated as the hypothetical axis.

(4) The number other than “0” (hypothetical axis interpolation mode ON) or “1” (cancel) is commanded, it will be handled as “1” (cancel). However, when only the axis name is designated without a number, it will be handled as “0” (mode ON).

Program example

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



Precautions

(1) Interpolation functions that are used for hypothetical axis interpolation are helical interpolation and spiral interpolation.

(2) Cancel the hypothetical axis interpolation before the high-speed high-accuracy control II (G05P10000) is commanded.

(3) The hypothetical axis interpolation is valid only in the automatic operation. It is invalid in the manual operation mode. Handle interruption is valid even for the hypothetical axis, that is, axis will move by the interrupted amount.

(4) Movement command for the hypothetical axis will be ignored. The feedrate will be distributed in the same manner as actual axis.

(5) The protection functions such as interlock or stored stroke limit are valid for the hypothetical axis.

(6) Even when the hypothetical axis is applied for the hypothetical axis again, no error will occur and the hypothetical mode will be continued.

(7) When the hypothetical axis cancel is commanded to the actual axis, no error will occur and the axis remains as the actual axis.

(8) The hypothetical axis will be canceled by carrying out the reset 2 or reset & rewind.

7

Feed Functions

7.1 Rapid Traverse Rate

7.1.1 Rapid Traverse Rate

Function and purpose

The rapid traverse rate can be set with parameters independently for each axis. The available speed ranges are from 1 mm/min to 10000000 mm/min. The upper limit is subject to the restrictions limited by the machine specifications.
Refer to the specifications manual of the machine for the rapid traverse rate settings.
The feedrate is valid for the G00, G27, G28, G29, G30 and G60 commands.
Two paths are available for positioning: the interpolation type where the area from the start point to the end point is linearly interpolated or the non-interpolation type where movement proceeds at the maximum speed of each axis.
The type is selected with parameter "#1086 G0Intp". The positioning time is the same for each type.

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

- If the value set for the high-accuracy control mode rapid traverse rate is 0, the axis will move at the rapid traverse rate.
- The high-accuracy control mode rapid traverse rate can be set independently for each axis.
- The high-accuracy control mode rapid traverse rate is effective for the following G commands: G00, G27, G28, G29, G30 and G60.
- Override can be applied on the high-accuracy control mode rapid traverse rate using the PLC signal supplied. (The operation of the PLC signal depends on the MTB specifications.)

Note

(1) Rapid traverse override
Override can be applied by a PLC 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% and 100%).
Type2 : Override in 1% steps from 0% to 100%.

7.1.2 G00 Feedrate Command (,F Command)

Function and purpose

Use this function to specify G00 (positioning command) and an axis feedrate in G00 mode.
The speed of tool exchange, axis movement of gantry, etc. can be specified with the machining program so that the mechanical vibration can be suppressed.
Operations other than the feedrate follows the G00 specification.

Command format

Rapid traverse at a feedrate specified with the ",F" command

G00 X_ Z_ (Y_)   ,F1000;

,F	Specifies the rapid traverse rates for G00, movement in G00 mode and the movement during the fixed cycle for drilling. The range is equal to the range of the feed per minute F command (mm/min, inch/min) in the G01 mode. Switching inch/mm is invalid for rotary axes.

Detailed description

(1) ",F" command is in effect only in the block in which it is commanded.

(2) If ",F" is commanded in G00, G27 to G30, G60, G00 mode, a block other than the one that specifies the movement to the initial point of the hole position for the drilling cycle or a block that does not contain a movement command (axis address command), ",F" is ignored.

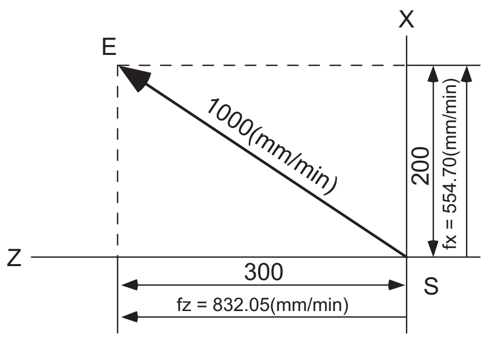
(3) ",F" command in the feed per revolution (G95) mode will also be considered a feed per minute feedrate.

(4) The motion of the ",F" command varies depending on the status of parameter "#1086 G0Intp".

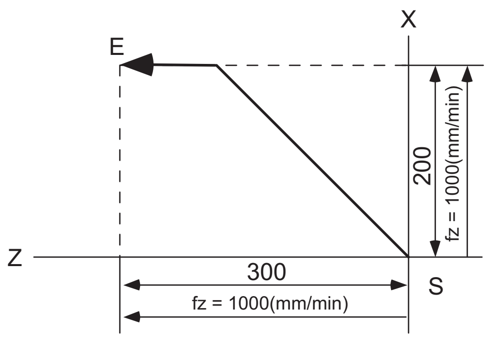
"#1086 G0Intp"	Handling of ",F"command
OFF (see figure shown at below left)	Handled as an interpolation speed.
ON (see figure shown at below right)	Handled as a commanded speed for each axis.

Feedrates when commanding G00 X200. Z300. ,F1000

When "G0 non-interpolation" is OFF



When "G0 non-interpolation" is ON



fx: Actual X axis rate
fz: Actual Z axis rate

(5) When the ",F" command has not been issued, the rapid traverse rate set by the axis specification parameter will be valid. (*1)

(6) The ",F" command is clamped by the rapid traverse rate set by the axis specification parameter. (*1)
Feedrate clamping depends on the setting of parameter "#1086 G0Intp".

"#1086 G0Intp"	Speed clamp
OFF	If it is found that, after converting ",F" command value (interpolation speed) into a speed for each axis, there is an axis for which the programmed feedrate exceeds the rapid traverse rate parameter, the interpolation speed is calculated so that it does not exceed the rapid traverse rate. (*1)
ON	An axis whose ",F" command value (per axis speed) exceeds the rapid traverse rate parameter is clamped to a speed specified by the parameter. (*1) For an axis that does not exceed the rapid traverse rate parameter, the commanded speed is applied.

(*1) The rapid traverse rate parameter depends on the MTB specifications.
Typically, parameter "#2001 rapid" is selected.

Program example

(1) Feedrate command in G00 block and G00 mode (for G00 interpolation)

:	
G00 X100. Z100. ,F1000 ;	The tool moves at the combined feedrate, 1000 (mm/min), of XZ.
X200. Z200. ;	The X and Z axes interpolate at the fastest feedrate that does not exceed the rapid traverse rate parameter for each of these axes.
X300.Z300. ,F2500 ;	The tool moves at the combined feedrate, 2500 (mm/min), of XZ.
:	

(2) Speed command for the movement to the initial point of the hole position for the drilling cycle (for a longitudinal tapping cycle)

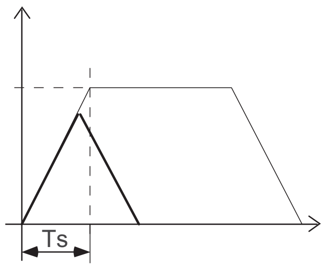
:	
G88 X-20. Z30 R5. F1.D3 S500 ,R1 ,F2000 ;	The tool moves to the initial point (Z30.) of the hole position at 2000 (mm/min). Positioning (G00) during the drilling cycle moves at 2000 (mm/min).
X-20. Z35. R5. ;	The tool moves to the initial point (Z35.) of the hole position at the Z axis rapid traverse rate (parameter setting value).
X-20. Z40. R5.,F3000 ;	The tool moves to the initial point (Z40.) of the hole position at 3000 (mm/min). Positioning (G00) during the drilling cycle moves at 2000 (mm/min).
G80 ;	
:	

Relationship with Other Functions

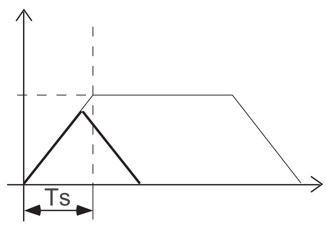
Rapid Traverse Constant Inclination Acceleration/Deceleration

When ",F" is specified, constant inclination acceleration/deceleration control is applied to the feedrate specified by ",F".
The feedrate (vertical axis in the figure below) varies depending on whether or not the ",F" command has been issued.

(1) When the ",F" command has not been issued
Rapid traverse rate set by the parameter



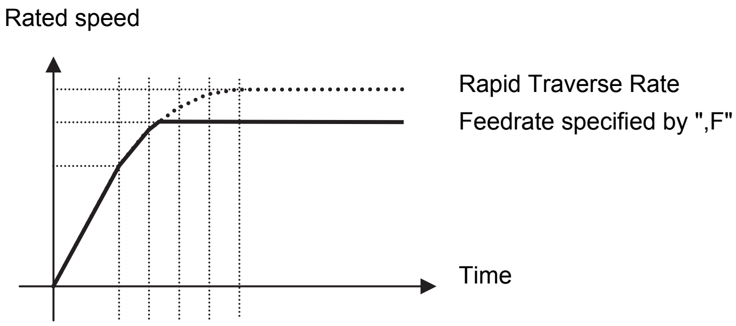
(2) When the ",F" command has been issued
Feedrate specified by the ",F" command



Movement when rapid traverse constant inclination is enabled
Movement when rapid traverse constant inclination is disabled

Rapid Traverse Constant Inclination Multi-step Acceleration/Deceleration

The feedrate specified by ",F" is applied according to acceleration pattern calculated from acceleration rate to "rapid traverse rate", "rated speed", "G00 time constant to rated speed" and "maximum acceleration".



Rapid traverse override

An override for ",F" command

Override cancel

The override cancel for the rapid traverse override is also invalid when ",F" is specified.

Dry run

Dry run is valid when the parameter "#1085 G00Drn" is ON and the rapid traverse is OFF. The axis will move at the manual feedrate that is set. If the manual override valid is turned ON, the cutting feed override also becomes valid.

External deceleration

It is also valid when ",F"is specified.

Programmable in-position check

It is also valid when ",F"is specified.

Tool center point control

",F" command is ineffective in tool center point control.

Special Fixed Cycle

When an ",F" command is specified in the fixed cycle for drilling, the movement between hole positions is carried out at the speed commanded with ",F".
",F" commands in the same block as for the special fixed cycle are ignored.

Unidirectional positioning

When an ",F" command is specified in the same block as G60 (unidirectional positioning), the feedrate specified by ",F" is assumed.

Reference position check, Start point return, Tool change position return

When an ",F" command is specified in the same block as G27 (reference position check), G29 (start point return), and/or G30.n (tool change position return), the feedrate specified by ",F" is assumed.

Reference position return, 2nd to 4th reference position return

When an ",F" command is specified in the same block as G28 (reference position return) and G30 (2nd to 4th reference position return), the feedrate specified by ",F" is assumed.
Axes not subject to high-speed reference position return are returned by the dog-type of in the same way as with the manual type. The feedrate depends on the MTB specifications (parameter "#2025 G28rap").

Precautions

(1) If an ",F" command is specified when there is no specifications for the feedrate specified for G00, a program error (P39) will occur.
(2) ",F" and "F" commands may be specified in the same block. The "F" command is assumed to the feedrate for cutting feed.
(3) Depending on the MTB specifications (parameter "#1100 Tmove"), compensation may be performed on a block that does not contain a move command.
If an ",F" is specified in a tool compensation command (T command) block in which no move command is specified, compensation move is made at the feedrate specified by ",F" only in G00 mode.
(4) If an ",F" is specified in a tool radius compensation cancel command (G40) block in which no move command is specified, tool radius compensation is canceled at the specified feedrate only in G00 mode.
This is the same as when using the tool nose radius compensation instead of tool radius compensation.

7.2 Cutting Feed Rate

Function and purpose

The cutting feedrate is assigned with address F and numerals.
The cutting feedrate is valid for the G01, G02, G03, G02.1 and G03.1 commands.

If the cutting clamp feedrate for the high-accuracy control mode is set, the axis will move at that feedrate during high-accuracy control, high-speed high-accuracy control I/II/III, high-accuracy spline control or SSS control.

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

Examples Feed per minute (asynchronous feed)

	Feedrate	
G01 X100. Y100. F200 ;	200.0mm/min	F200 or F200.000 gives the same rate.
G01 X100. Y100. F123.4 ;	123.4mm/min	
G01 X100. Y100. F56.789 ;	56.789mm/min	

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

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

Note

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

7.3 F1-digit Feed

Function and purpose

By setting the F1-digit feed parameter, the feedrate which has been set to correspond to the 1-digit number following the F address serves as the command value.
When F0 is assigned, the rapid traverse rate is established and the speed is the same as for G00. (G modal does not change, but the acceleration/deceleration method follows the rapid traverse setting.)
When F1 to F5 is assigned, the feedrate set to correspond to the command serves as the command value.
If F1-digit feedrate changing valid signal is turned ON when F1-digit feed is commanded, the feedrate specified by the parameter can be increased or decreased by operation of manual handle. For the changing of F1-digit feedrate with the handle feed, refer to the instruction manual.

Detailed description

(1) To validate the F1-digit feed, the parameter "#8145 Validate F1 digit" or "#1079 f1digt" must be ON.

(2) The feedrates that correspond to F1 to F5 depend on the MTB specifications (parameters "#1185 spd_F1" to "#1189 spd_F5").
The increase/reduction range is from "0" to the set value of the parameter "#1506 F1_FM".
An operation error (M01 0104) will occur when the feedrate is "0".
When F0 is commanded, the acceleration or deceleration method follows the rapid traverse setting. Note that the G modal is not changed.

(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, G03, G02.1 or G03.1 modal.

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

(6) The F1-digit feedrate command can also be used during high-speed high-accuracy control II.
However, a program error (P62) will occur when F0 command is issued.

(7) The F1-digit command is modal.

(8) The number of manual handle pulses is 1 pulse per scale unit regardless of the scaling factor.

(9) During a F1-digit command, the F1-digit number and F1-digit command signal are output as the PLC signals. (Based on the MTB specifications.)

Precautions

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

(2) If F0 is used in the G02, G03, G02.1 or G03.1 mode, the program error (P121) will occur. The error will be eliminated if the F0 command is rewritten.

(3) 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.

(4) 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.

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

(6) 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).

(7) When both the F1-digit feed command and inverse time feed command are present, the inverse time feed command will have priority.
(The inverse time feed function is available only for a machining center system.)

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

(9) 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 or G93 command (inverse time feed) is next assigned.

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

Metric input

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

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

Input Setting unit	**D(0.00001mm)**	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)
Minimum command unit	1 (=1.00000) (1.=1.00000)	1 (=0.00001) (1.=1.00000)
Command range	0.00001 - 1000000.00000	0.00001 - 999.99999

Input Setting unit	E(0.000001mm)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(mm/min)	F(mm/rev)
Minimum command unit	1 (=1.000000) (1.=1.000000)	1 (=0.000001) (1.=1.000000)
Command range	0.000001 - 1000000.000000	0.000001 - 999.999999

Inch input

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

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

Input Setting unit	D(0.000001inch)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)
Minimum command unit	1 (=1.000000) (1.=1.000000)	1 (=0.000001) (1.=1.000000)
Command range	0.000001 - 100000.000000	0.000001 - 999.999999

Input Setting unit	E(0.0000001inch)	
Command Mode	Feed per minute	Feed per revolution
Command Address	F(inch/min)	F(inch/rev)
Minimum command unit	1 (=1.0000000) (1.=1.0000000)	1 (=0.0000001) (1.=1.0000000)
Command range	0.0000001 - 100000.0000000	0.0000001 - 999.9999999

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

FC = F × 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 erro 0105" occurs.

(4) Feedrate in the machine lock status is the command speed.

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

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

(7) Whether feed per minute (G94) or feed per revolution (G95) is to be established when the power is turned ON or when M02 or M30 is executed can be selected by setting parameter "#1074 I_Sync".

7.5 Inverse Time Feed ; G93

Function and purpose

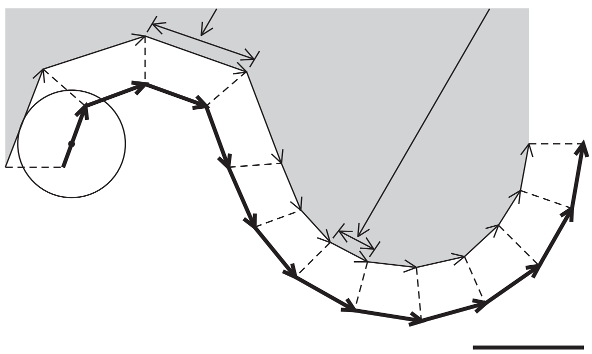
During inside cutting when machining curved shapes with radius compensation applied, the machining speed on the cutting surface becomes faster than the tool center feedrate. Therefore, problems such as reduced accuracy may occur.

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

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

Regular F command

Actual machining speed: High　　Actual machining speed: Low

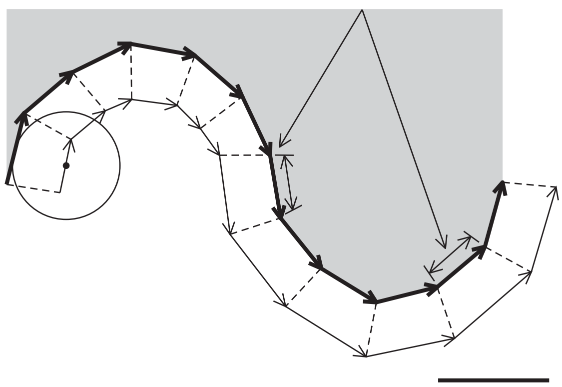


The speed of tool center is commanded, thus the actual speed at the cutting surface may become larger or smaller.

F command

Inverse time feed

The actual machining speed is constant.



The actual speed at the cutting surface is commanded, thus, the speed will be constant and machining speed can be maintained as commanded regardless of the tool radius.

F command

Command format

Inverse time feed

G93 ;

Inverse time feed (G93) is a modal command. Once commanded, it will be valid until feed per minute (G94) or feed per revolution (G95) is commanded.

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

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

Detailed description

(1) Inverse time feed (G93) is a modal command. Once commanded, it is valid until feed per minute (G94) or feed per revolution (G95) is commanded, or until a reset (M02, M30, etc.) is executed.

(2) Command method of F command values in inverse time feed

<table>
<tr><th colspan="2"></th><th>Metric command (G21)</th><th>Inch command (G20)</th></tr>
<tr><td colspan="2">In linear mode (G01)</td><td>$\frac{\text{Cutting point feedrate (mm/min)}}{\text{Line segment length (mm)}}$</td><td>$\frac{\text{Cutting point feedrate (inch/min)}}{\text{Line segment length (inch)}}$</td></tr>
<tr><td colspan="2">In arc mode (G02, G03) (G02.1, G03.1)</td><td>$\frac{\text{Cutting point feedrate (mm/min)}}{\text{Start point arc radius (mm)}}$</td><td>$\frac{\text{Cutting point feedrate (inch/min)}}{\text{Start point arc radius (inch)}}$</td></tr>
<tr><td rowspan="4">Command range</td><td>B</td><td colspan="2">0.001 to 999999.999 (1/min)</td></tr>
<tr><td>C</td><td colspan="2">0.0001 to 999999.9999 (1/min)</td></tr>
<tr><td>D</td><td colspan="2">0.00001 to 999999.99999 (1/min)</td></tr>
<tr><td>E</td><td colspan="2">0.000001 to 999999.999999 (1/min)</td></tr>
</table>

(3) The initial modal after a restart is G94 (feed per minute) or G95 (feed per revolution).

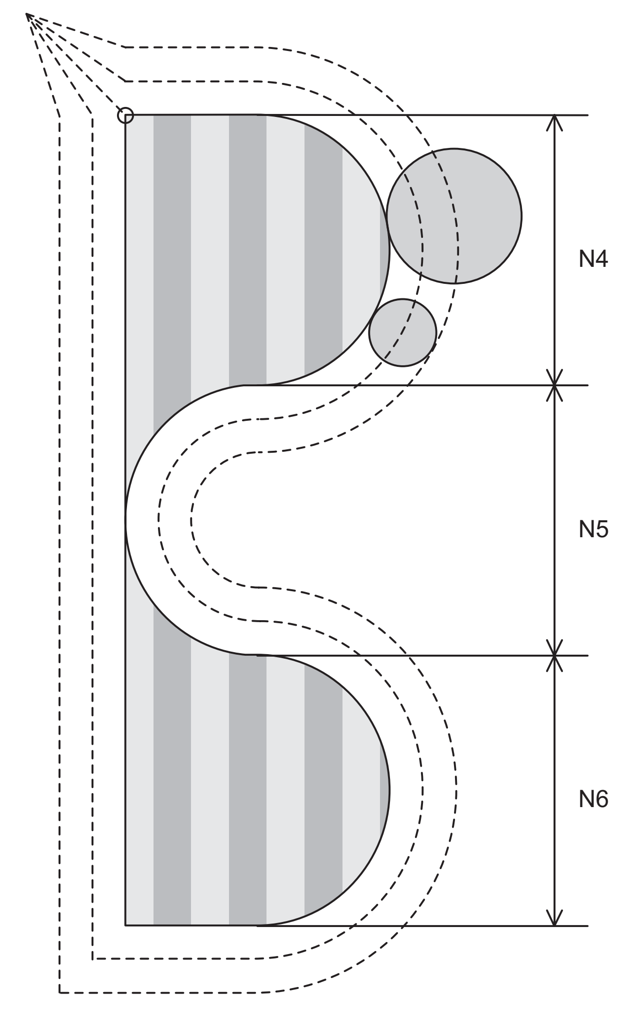
(4) The feedrate of the block inserted in tool radius compensation and corner R/C is the same speed as the feedrate of the block immediately before it.

(5) The feedrate of the block inserted in C axis normal line control (normal line control type II) is the same speed as the feedrate of the movement block after turning.

Program example

When using inverse time feed during tool radius compensation

Feed per minute
N01 G90 G00 X80. Y-80. ;
N02 G01 G41 X80. Y-80. D11 F500 ;
N03 X180. ;
N04 G02 Y-280. R100. ;
N05 G03 Y-480. R100. ;
N06 G02 Y-680. R100. ;
N07 G01 X80. F500 ;
N08 Y-80. ;
N09 G04 X80. Y-80. ;
N10 M02;
Inverse time feed
N01 G90 G00 X80. Y-80. ;
N02 G01 G41 X80. Y-80. D11 F500 ;
N03 X180. ;
N04 G93 G02 Y-280. R100. F5 ;
N05 G03 Y-480. R100. F5 ;
N06 G02 Y-680. R100. F5 ;
N07 G94 G01 X80. F500 ;
N08 Y-80. ;
N09 G04 X80. Y-80. ;
N10 M02;



Comparison between feed per minute and inverse time feed (Assuming that tool radius is 10. [mm]) (Unit: mm/min)

Sequence No.	Feed per minute		Inverse time feed	
	Feedrate of tool center	Feedrate of cutting point	Feedrate of tool center	Feedrate of cutting point
N04	F500	F450	F550	F500
N05	F500	F550	F450	F500
N06	F500	F450	F550	F500
		↓		↓
		The block seam protrudes due to the cutting speed change at the block seam.		The feedrate follows the command regardless of the tool radius.

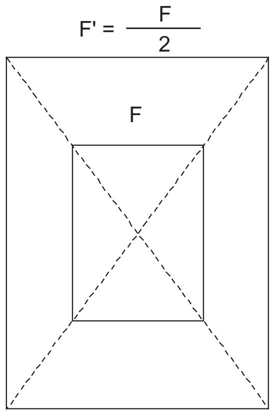
Relation with other functions

(1) Scaling (G51)

When using a scaling function, issue a F command for the shape after scaling. For example, if a double-size scaling is carried out, the machining distance will be doubled.

Thus, if executing a cutting at the same speed as that of before scaling, command the value (F’) calculated by dividing F value by the multiples of scaling.

F = Feedrate (mm/min) / Distance (mm)



Shape after scaling (Double size)

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

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

(3) If the speed calculated in the G93 mode exceeds the speed range at the feed per minute, clamping is performed at the clamp speed set with parameters.

(4) The program error (P125) will occur when the commands below are issued in the inverse time feed (G93) mode.

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

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

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

Precautions

(1) The initial modal after a restart is G94 (feed per minute) or G95 (feed per revolution).

(2) The F command in G93 modal is unmodal. Issue an F command for each block. The program error (P62) will occur in blocks with no F command.

(3) The program error (P62) will occur when F0 is commanded.

(4) An F command is necessary when changing from G93 to G94 or G95. The program error (P62) will occur if there is no F command.

(5) The feed function is clamped at the maximum cutting speed. Consequently, the feed may be slower than the commanded speed.

(6) If an extremely slow speed such as F0.001 is designated, an error will occur in the machining time.

7.6 Feedrate Designation and Effects on Control Axes

Function and purpose

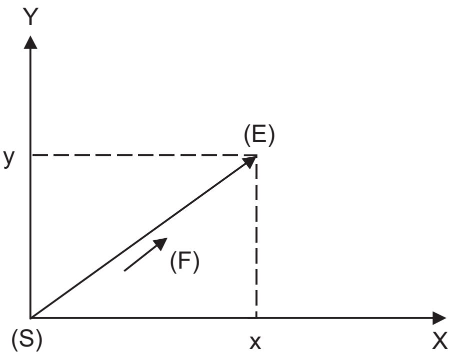
It has already been mentioned that a machine has a number of control axes. These control axes can be divided into linear axes which control linear movement and rotary axes which control rotary movement. The feedrate is designed to assign the displacement speed of these axes, and the effect exerted on the tool movement speed which poses problems during cutting differs according to when control is exercised over the linear axes or when it is exercised over the rotary axes.
The displacement amount for each axis is assigned separately for each axis by a value corresponding to the respective axis. The feedrate is not assigned for each axis but assigned as a single value. Therefore, when two or more axes are to be controlled simultaneously, it is necessary to understand how this will work for each of the axes involved.
The assignment of the feedrate is described with the following related items.

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 Y) are to be controlled:



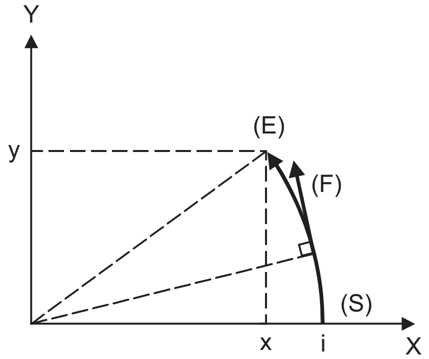
$$fx = f \times \frac{x}{\sqrt{x^2 + y^2}}$$ ... Feedrate for X axis

$$fy = f \times \frac{y}{\sqrt{x^2 + y^2}}$$ ... Feedrate for Y axis

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

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

(Example) When the feedrate is designated as "f" and the linear axes (X and Y) are to be controlled using the circular interpolation function:
The rate in the tool advance direction, or in other words the tangential direction, will be the feedrate designated in the program.



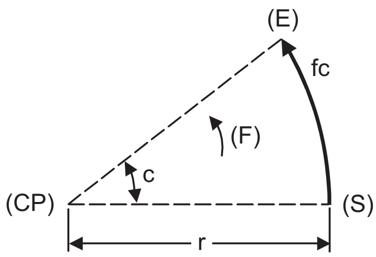
(S) Tool start point
(E) Tool end point
(F) Speed in this direction is "f".

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

When controlling rotary axes

When rotary axes are to be controlled, the designated feedrate functions as the rotary speed of the rotary axes or, in other words, as an angular speed.
Consequently, the cutting feed in the tool advance direction, or in other words the linear speed, varies according to the distance between the center of rotation and the tool.
This distance must be borne in mind when designating the feedrate in the program.

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



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

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

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

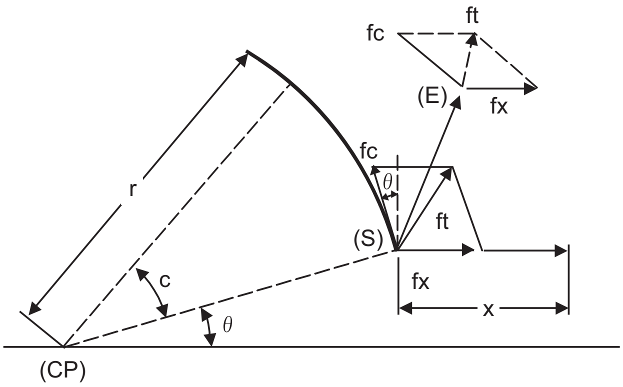
Therefore, the feedrate to be designated in the program must be :

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

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

The controller proceeds in exactly the same way whether linear or rotary axes are to be controlled.
When a rotary axis is to be controlled, the numerical value assigned by the coordinate word (A,B,C) is the angle and the numerical values assigned by the feedrate (F) are all handled as linear speeds. In other words, 1°of the rotary axis is treated as being equivalent to 1mm of the linear axis.
Consequently, when both linear and rotary axes are to be controlled simultaneously, in the components for each axis of the numerical values assigned by F will be the same as previously described "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, 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.
Size and direction vary for ft.

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

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

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

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

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

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

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

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

Where r is the distance (in millimeters) between the center of rotation and the tool and
θ is the angle (in degrees) between the (S) point and the X axis at the center of rotation

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

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

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

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

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

"ft" in formula (6) is the speed at the (S) point and the value of θ changes as the C axis rotates, which means that the value of "ft" will also change. Consequently, in order to keep the cutting feed "ft" as constant as possible the angle of rotation which is designated in one block must be reduced to as low as possible and the extent of the change in the θ value must be minimized.

7.7 Rapid Traverse Constant Inclination Acceleration/Deceleration

Function and purpose

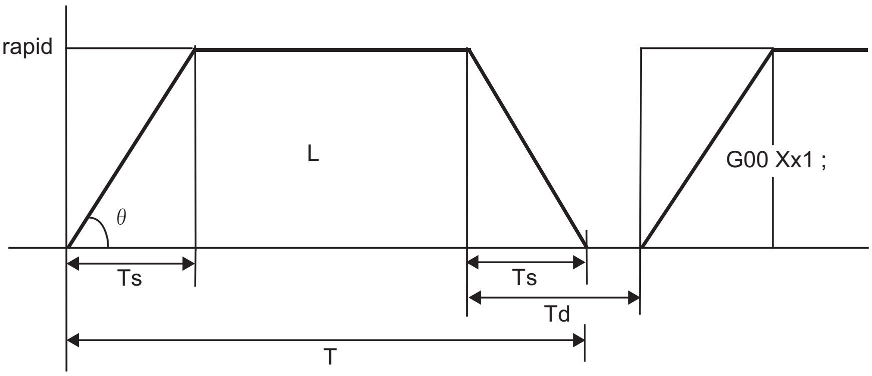
This function performs acceleration and deceleration at a constant inclination during linear acceleration/deceleration in the rapid traverse mode. Compared to the method of acceleration /deceleration after interpolation, the constant inclination acceleration/deceleration method makes for improved cycle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

Detailed description

(1) Rapid traverse constant inclination acceleration/deceleration are valid only for a rapid traverse command. Also, this function is effective only when the rapid traverse command acceleration/deceleration mode is linear acceleration and linear deceleration.

(2) The acceleration/deceleration patterns in the case where rapid traverse constant inclination acceleration/deceleration are performed are as follows.

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



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

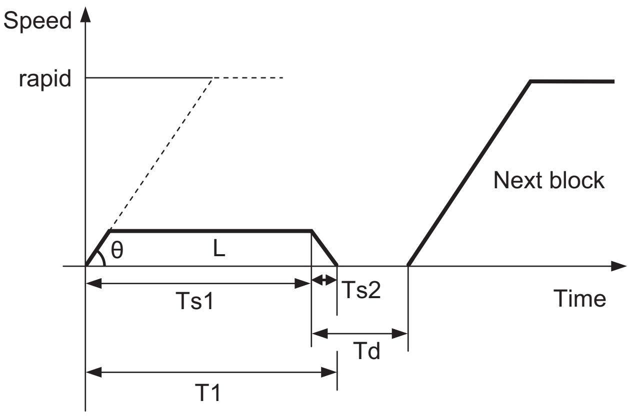
$$Td = Ts + (0 \sim 14ms)$$

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

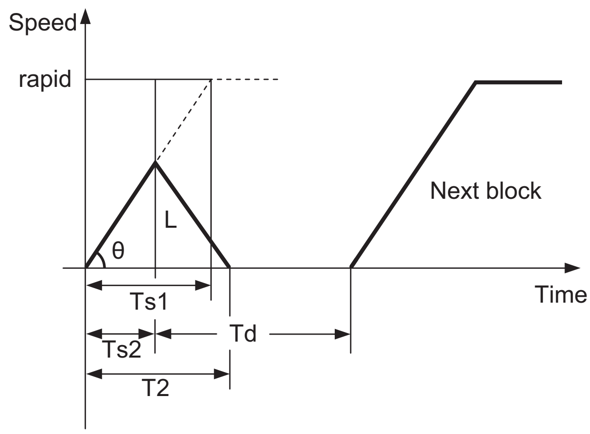
rapid : Rapid Traverse Rate
Ts : Acceleration/deceleration time constant
Td : Command deceleration check time
θ : Acceleration/deceleration inclination
T : Interpolation time
L : Interpolation distance

[When the interpolation distance is smaller than the acceleration/deceleration distance]

In case of time-constant acceleration/deceleration:



In case of inclination-constant acceleration/deceleration:



$$T1 = Ts1 + Ts2$$

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

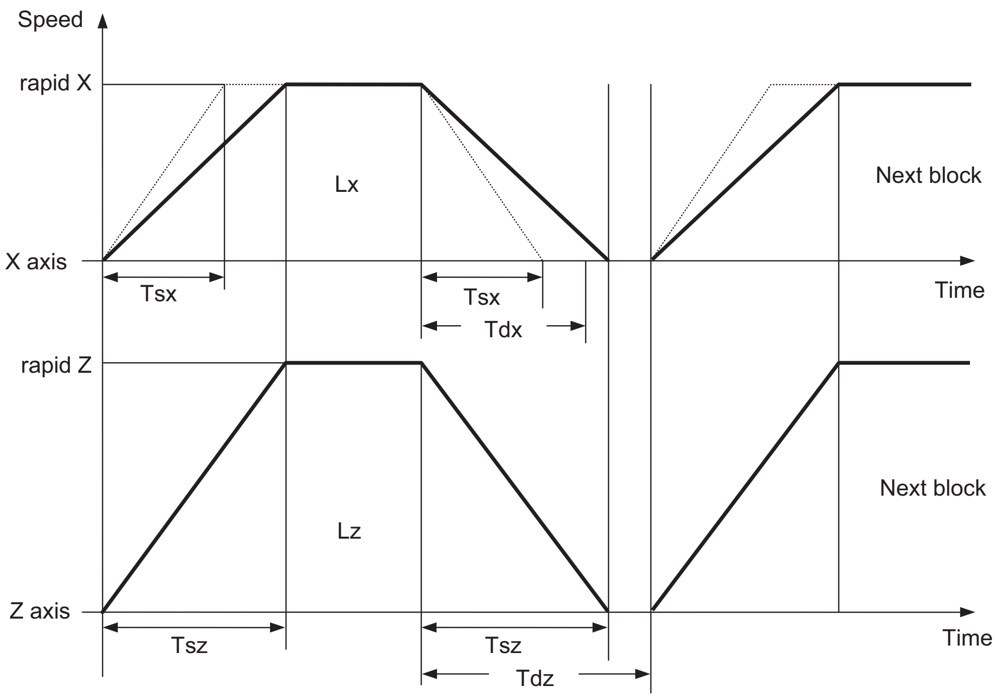
$$Td = T2/2 + (0 \sim 14\ ms)$$

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

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
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)]

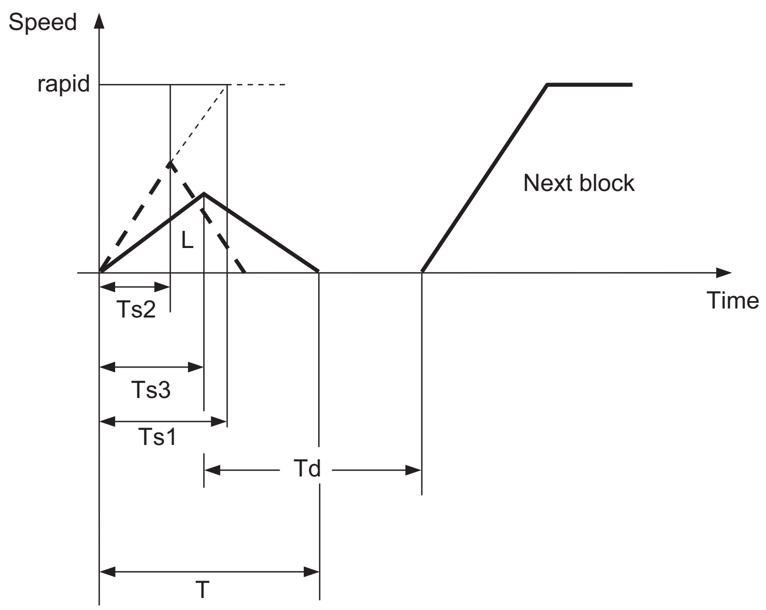


When Tsz is greater than Tsx,
Tdz is also greater than Tdx, and
Td = Tdz in this block.

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]



$T = 2 \times Ts2$

$Td = T/2 + (0 \sim 14\ ms)$

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
T : Interpolation time
L : Interpolation distance

(5) Use the rapid traverse time constant changeover request signal to switch the rapid traverse time constant. The operations via PLC signals and the settings of related parameters depend on the MTB specifications.
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 time 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 G00 (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 G00 (rapid traverse).

Precautions

(1) When "#2003 smgst" (acceleration/deceleration mode) is set to the soft acceleration/deceleration, and "#1219 aux03 bit7" (time constant setting changeover for soft acceleration/deceleration) is set to "1", the 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. In this case, the acceleration time will be "G0tL+G0t1" or "G1tL+G1t1".

(2) When "#2003 smgst" (acceleration/deceleration mode) 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.

(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 reduce the cycle time, this can also be a cause of machine vibrations. In these cases, if the minimum time constant for inclination-constant acceleration/deceleration is set in parameter "#2198 G0tMin", it is possible to perform acceleration/deceleration to prevent the acceleration/deceleration time from being below this setting value. This parameter depends on the MTB specifications.

7.8 Rapid Traverse Constant Inclination Multi-step Acceleration/ Deceleration

Function and purpose

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

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

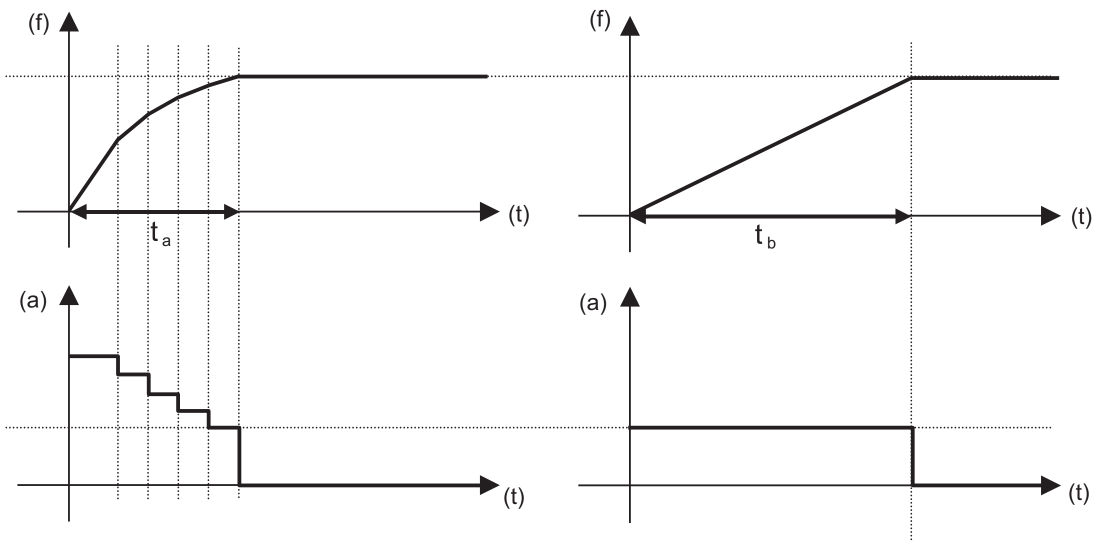
In the rapid traverse constant inclination acceleration/deceleration method, the acceleration is treated as constant because this torque characteristic is not considered. So, It is necessary to use a minimum acceleration within the used speed range. Therefore, the margin of acceleration must be had in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

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

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

[Rapid traverse constant inclination multi-step acceleration/deceleration]

[Rapid traverse constant inclination acceleration/ deceleration]



Number of steps is automatically adjusted by parameter setting.

It was necessary to slow down the acceleration for high speed rotation.

(f) Speed (t) Time (a) Acceleration

Detailed description

Use conditions

(1) The validity of this function depends on the MTB specifications. (Parameter "#1205 G0bdcc")
However, note the following conditions.

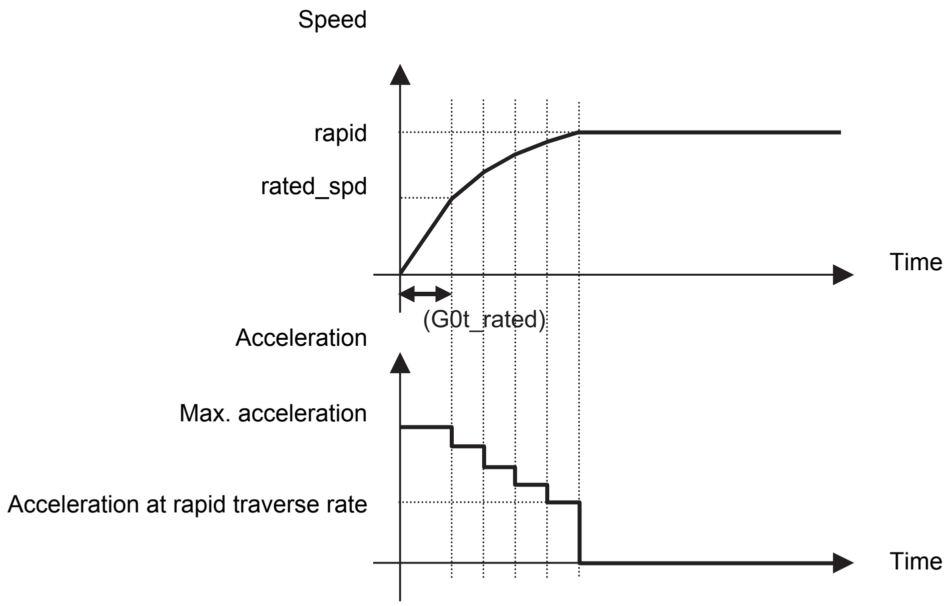
(a) "2" cannot be set to parameter "#1205 G0bdcc" except the 1st part system. When "2" is set to other than 1st part system, an MCP alarm (Y51 0017) will occur.

(b) When there is no specification for the rapid traverse constant inclination acceleration/deceleration, "2" cannot be set to parameter "#1205 G0bdcc". Even if the parameter is set to "2", this function is invalid. A normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.

(c) Even if "2" is set to "#1205 G0bdcc" when G00 non-interpolation type ("#1086 G00Intp" = "1"), this function is invalid. In this case, a normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.

(2)To use this function, the following parameters must be set for each axis.

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



Acceleration rate in proportion to the maximum acceleration rate = Acceleration at rapid traverse rate / Max. acceleration

(3) When either of the following conditions applies, this function is invalid and operates as "rapid traverse constant inclination acceleration/deceleration". For the axis for which the rapid traverse constant inclination multi-step acceleration/deceleration is not necessary, set "0" to "#2151 rated_spd", "#2152 acc_rate" and "#2153 G0t_rated". However, these parameters depend on the MTB specifications.

(a) When "#2151 rated_spd" (rated speed) is "0" or larger than "#2001 rapid" (rapid traverse)

(b) When "#2152 acc_rate" (Acceleration rate in proportion to the maximum acceleration rate) is "0" or "100"

(c) Even if "2" is set to "#1205 G0bdcc" when G00 non-interpolation type ("#1086 G00Intp" = "1"), this function is invalid. In this case, a normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.

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

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

Decision method of steps

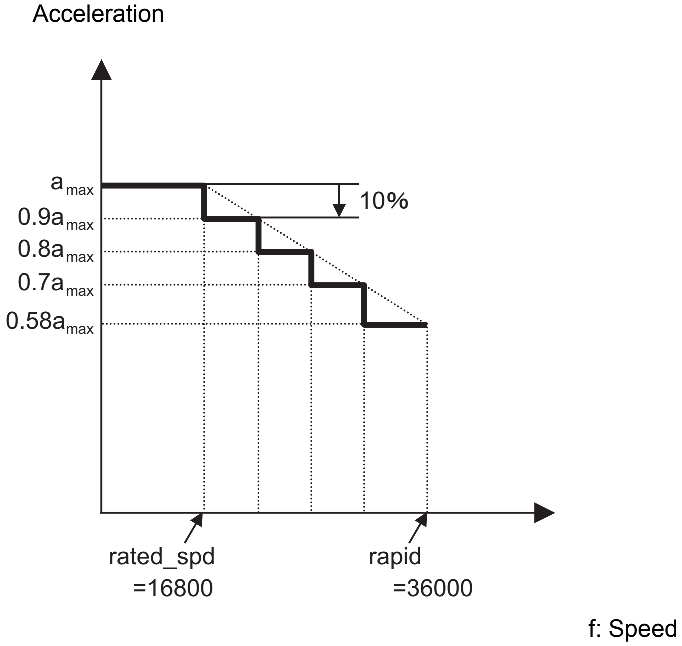
For rapid traverse constant inclination multi-step acceleration/deceleration, the number of steps is automatically adjusted by set parameter.

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

"Step" = (100 - "#2152 acc_rate") / 10 + 1 (Discard fractions less than 1)

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

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

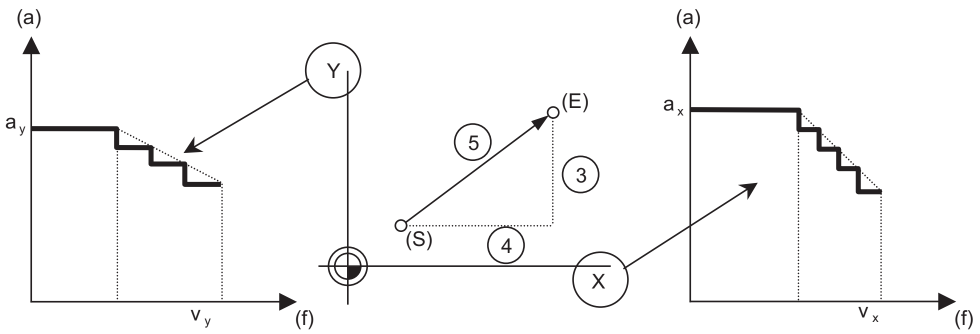


Acceleration pattern at two or more axis interpolation

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

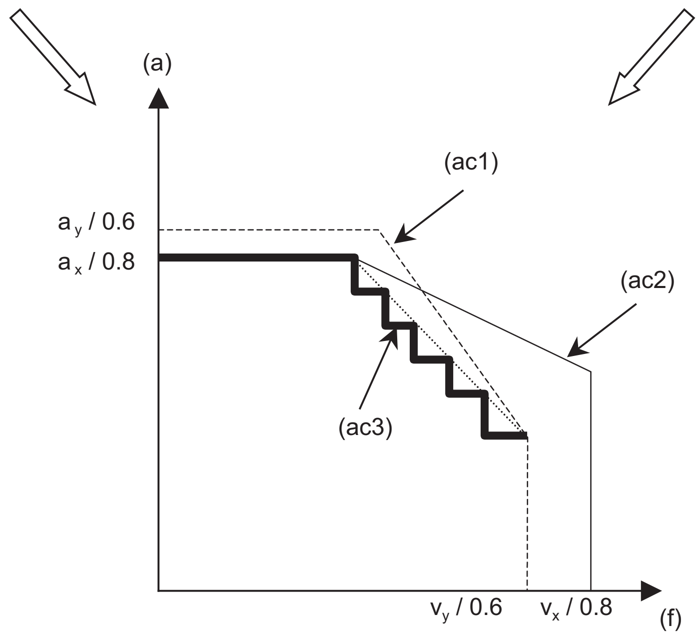
- Interpolation type (#1086 G0Intp = 0) : Moves from the start point to the end point by straight line
- Non-interpolation type (#1086 G0Intp = 1) : Each axis moves separately at the speed of the parameter

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



[Acceleration pattern of Y axis independently]

[Acceleration pattern of X axis independently]



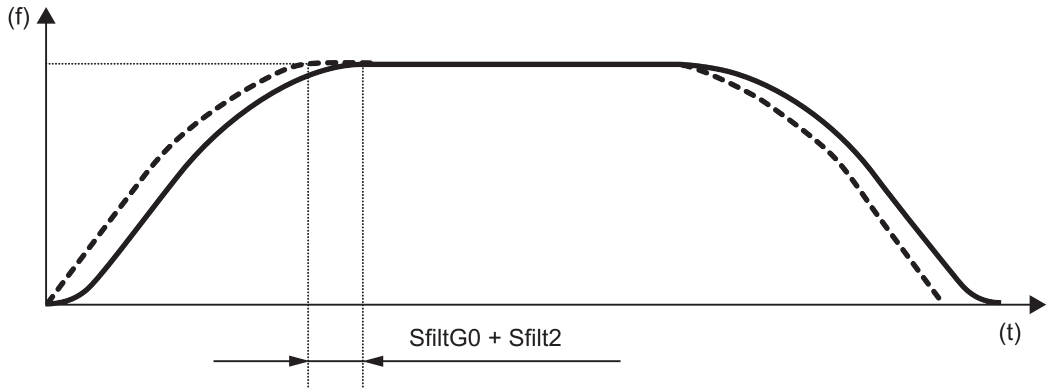
[Acceleration pattern of synthesis direction]

(a) Acceleration (f) Speed
(S) Start point (E) End point
(ac1) Acceleration pattern when the axis moved to synthesis direction at Y axis rapid traverse rate
(ac2) Acceleration pattern when the axis moved to synthesis direction at X axis rapid traverse rate
(ac3) Acceleration pattern of synthesis direction

S-pattern filter control

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

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



(f) Speed (t) Time

No S-pattern filter control S-pattern filter control

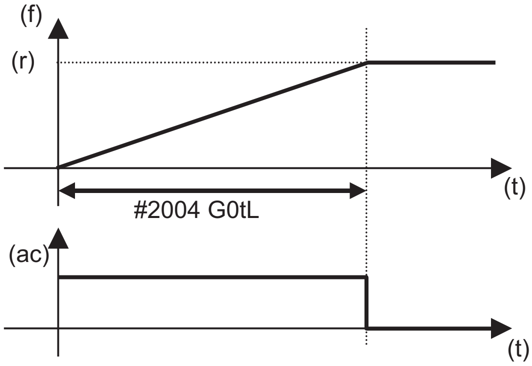
The high-accuracy control mode rapid traverse rate

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

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

(1) When "The high-accuracy control mode rapid traverse rate" > "rapid traverse rate"

This function is invalid and operates as "rapid traverse constant inclination acceleration/deceleration".

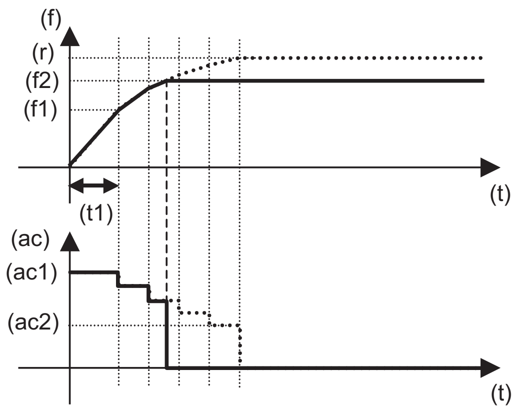


(f) Speed
(t) Time
(r) Rapid traverse rate
(ac) Acceleration

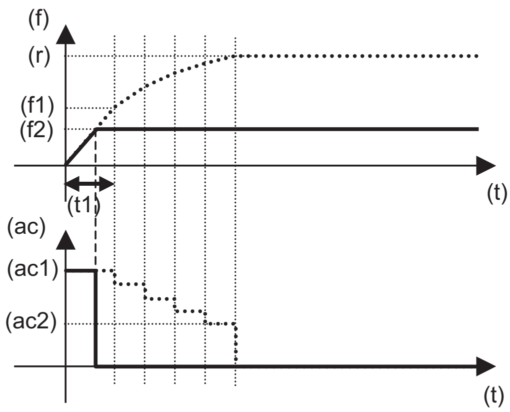
(2) When "The high-accuracy control mode rapid traverse rate" < "rapid traverse rate"

"The high-accuracy control mode rapid traverse rate" is applied according to acceleration pattern calculated from acceleration rate to "rapid traverse", "rated speed", "G00 time constant to rated speed" and "maximum acceleration".

Larger than the rated speed



Smaller than the rated speed

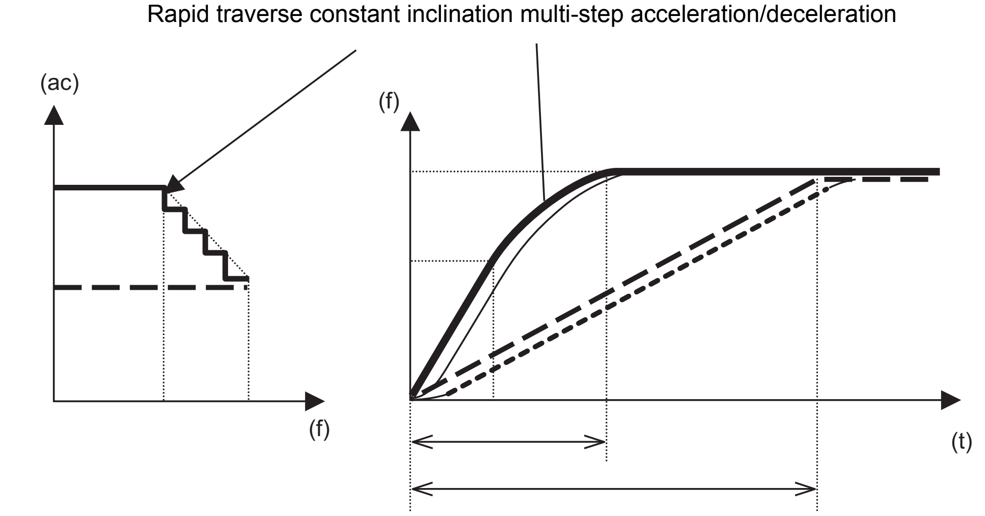


(f) Speed
(f1) Rated speed
(f2) The high-accuracy control mode rapid traverse rate
(t) Time
(t1) Acceleration time to rated speed
(ac) Acceleration
(ac1) Max. Acceleration
(ac2) Acceleration at rapid traverse rate
(r) Rapid traverse rate

Precautions

(1) Rapid traverse constant inclination multi-step acceleration/deceleration are valid only for a rapid traverse command. Note that when the manual rapid traverse, rapid traverse constant inclination multi-step acceleration/deceleration cannot be used.
In this case, a time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied. So, acceleration/deceleration is decided by the following parameters.
- #2001 rapid : Rapid traverse rate
- #2003 smgst : Acceleration/deceleration mode
- #2004 G0tL: G00 time constant (linear)
- #2005 G0t1: G00 time constant (primary delay)

The acceleration time (time constant) is different between the rapid traverse constant inclination multi-step acceleration/deceleration and the manual rapid traverse as shown in figure.



(f) Speed (t) Time (ac) Acceleration

Rapid traverse constant inclination multi-step acceleration/deceleration
Manual rapid traverse (linear)
S-pattern filter control
Soft acceleration/deceleration

(2) Rapid traverse constant inclination multi-step acceleration/deceleration cannot be used in part system excluding 1st part system. However, even if two or more part systems are used, it is possible to use this function in case of the 1st part system.

(3) When there is no specification for the rapid traverse constant inclination acceleration/deceleration, this function is invalid even if "2" is set to the parameter "#1205 G0bdcc". In this case, a normal time constant acceleration/deceleration (acceleration/deceleration after interpolation) is applied.

(4) When G00 non-interpolation type ("#1086 G0Intp" = "1"), rapid traverse constant inclination multi-step acceleration/deceleration cannot be used. It is valid in interpolation mode only.

(5) When the rapid traverse constant inclination multi-step acceleration/deceleration is applied, rapid traverse acceleration/deceleration types ("#2003 smgst" bit0 to bit3) are ignored.

(6) When the rapid traverse constant inclination multi-step acceleration/deceleration is valid, G0 constant inclination ("#1200 G0_acc") cannot be used. Even if G0 constant inclination is valid ("#1200 G0_acc" = "1"), the setting is ignored.

(7) When the rapid traverse constant inclination multi-step acceleration/deceleration is valid, programmable in-position check cannot be used. The in-position width will be ignored even if commanded.

(8) This function cannot be used during the tool center point control.

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.

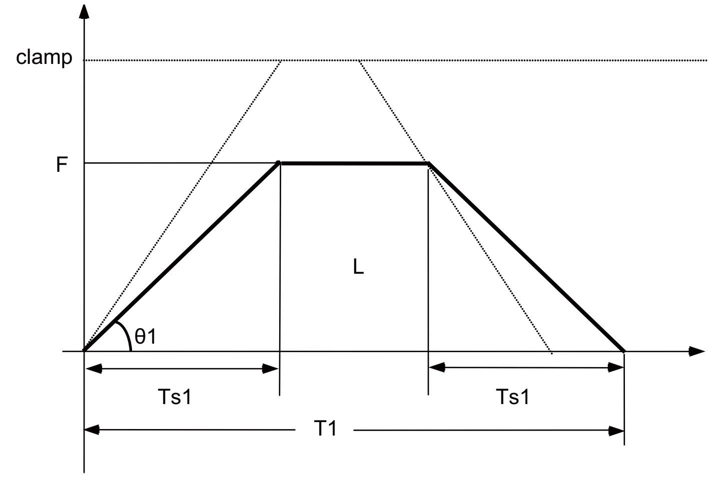
Detailed description

(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 linear interpolation (G01) 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.

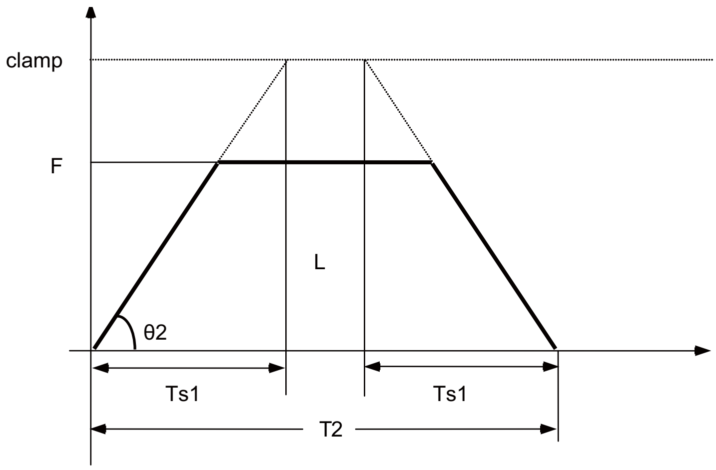
[When the interpolation distance is long enough for the rapid traverse rate to be achieved]In case of time-constant acceleration/deceleration:



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

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

In case of inclination-constant acceleration/deceleration:



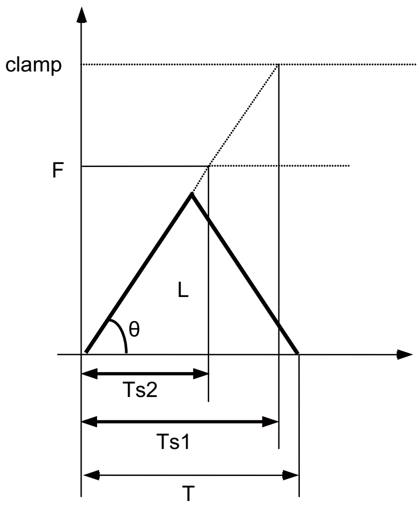
$$T2 = \frac{L}{F} + \frac{Ts1 \times F}{clamp}$$

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

clamp : Maximum cutting feed rate (Axis specification parameter "#2002 clamp")
F: Cutting feed rate
Ts1 : Acceleration/deceleration time (Axis specification parameter "#2007 G1tL")
θ1 : Acceleration/deceleration inclination (time-constant acceleration/deceleration)
θ2 : 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]

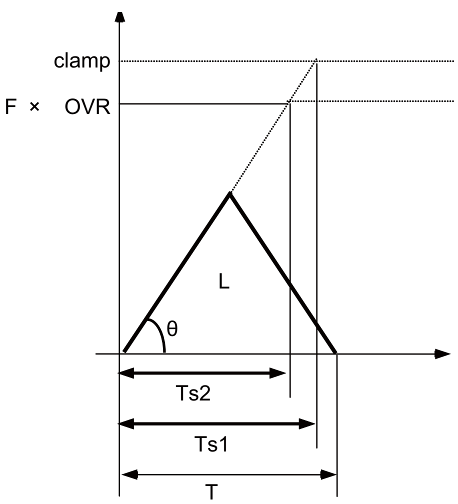


$$T = 2 \times \sqrt{Ts2 \times {}^{L}/_{clamp}}$$

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

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]



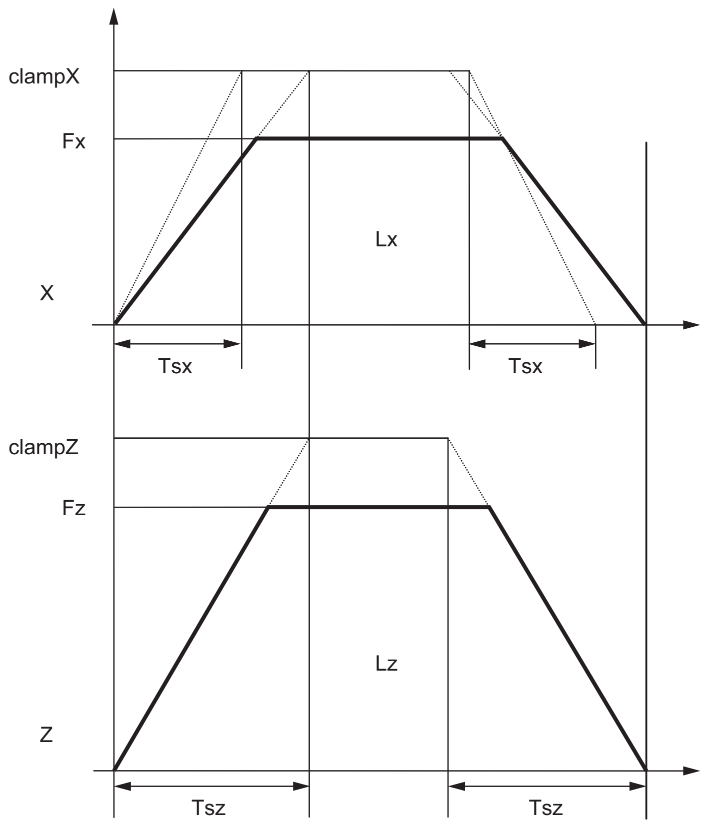
$$T = 2 \times \sqrt{Ts2 \times L \times OVR /_{clamp}}$$

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

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").

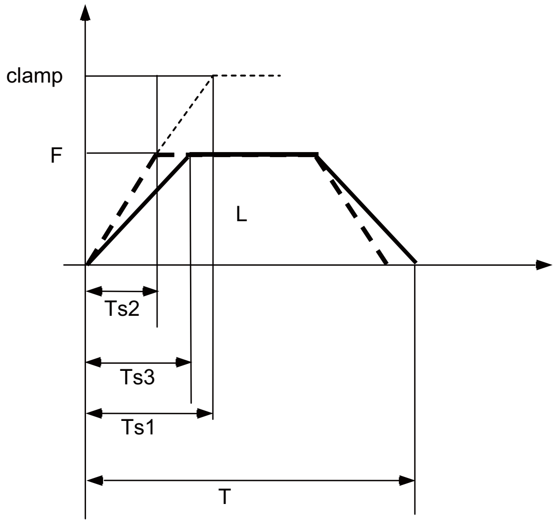


Tsx : X axis acceleration/deceleration time
Tsz : Z axis acceleration/deceleration time
Lx : X axis interpolation distance
Lz : Z axis interpolation distance
Fx : X axis feed rate
Fz : Z axis feed rate

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.

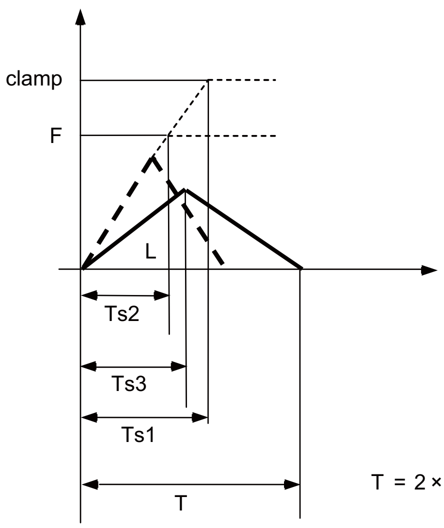


$$T = \frac{L}{F} + 2 \times Ts3$$

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.



$$T = 2 \times Ts3$$

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

Precautions

(1) If a value greater than 100 (%) is designated for cutting feed override under cutting feed inclinationconstant 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. (This parameter depends on the MTB specifications.) When the setting of this parameter is between 0 and 99 for "#1367 G1AccOVRMax", the override value is handled as 100% even if the specified cutting feed override is greater than 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/deceleration mode) is set to the soft acceleration/deceleration, and "#1219 aux03 bit7: Time constant setting changeover for soft acceleration/deceleration" is set to "1": Acceleration 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/deceleration mode) 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 (G01) 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. In these cases, if the minimum time constant for inclination-constant acceleration/deceleration is set in parameter "#2199 G1tMin", it is possible to perform acceleration/deceleration to prevent the acceleration/deceleration time from being below this setting value.
This parameter depends on the MTB specifications.

7.10 Exact Stop Check ; G09

Function and purpose

In order to prevent roundness during corner cutting and machine shock when the tool feedrate changes suddenly, there are times when it is desirable to start the commands in the following block once the in-position state after the machine has decelerated and stopped or the elapsing of the deceleration check time has been checked. The exact stop check function is designed to accomplish this purpose.
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 based on the parameter settings specified by the MTB. (Refer to section "7.12 Deceleration Check".)
The in-position width is set in servo parameter "#2224 sv024", "#2077 G0inps" or "#2078 G1inps". This parameter also depends on the MTB specifications.

Command format

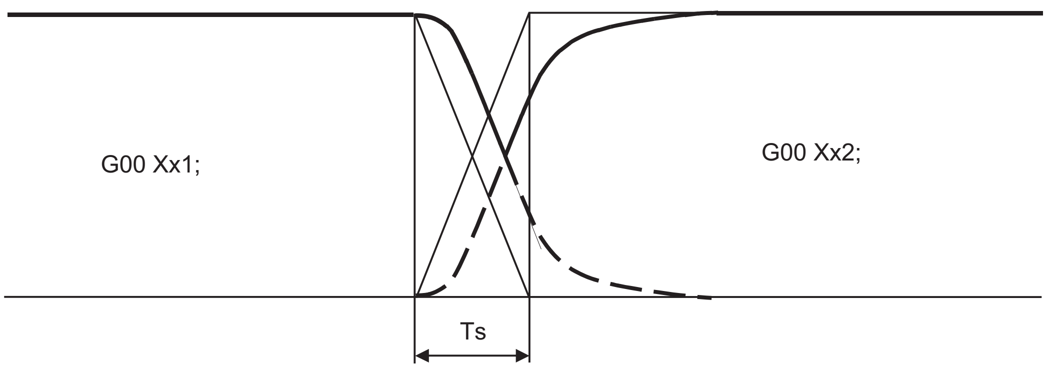
Exact stop check

G09 G01 (G02, G03);

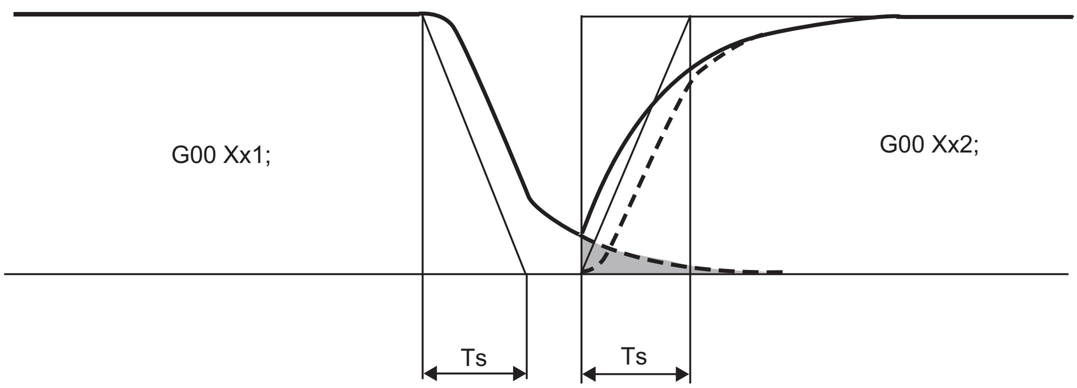
The exact stop check command G09 has an effect only with the cutting command (G01 - G03) in its particular block.

Detailed description

[With continuous cutting feed]



[With cutting feed in-position check]



Ts : Cutting feed acceleration/deceleration time constant

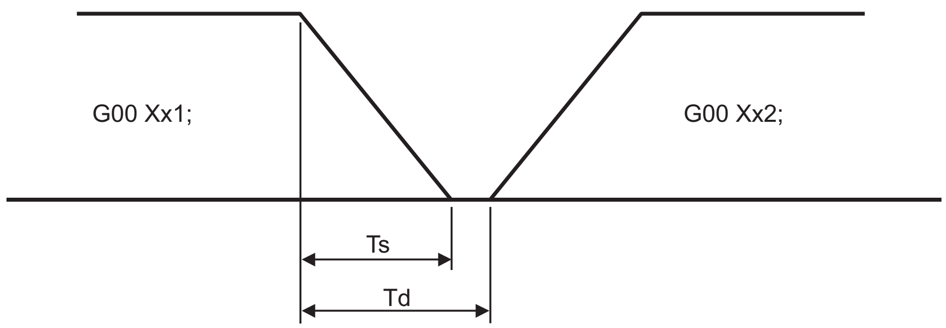
In-position width

The in-position width, as shown in the figure above, is the remaining distance (shaded area in the above figure) of the previous block when the next block is started is set in the servo parameter "#2224 sv024". (This depends on the MTB specifications.)
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.
(The parameter setting depends on the MTB specifications.)

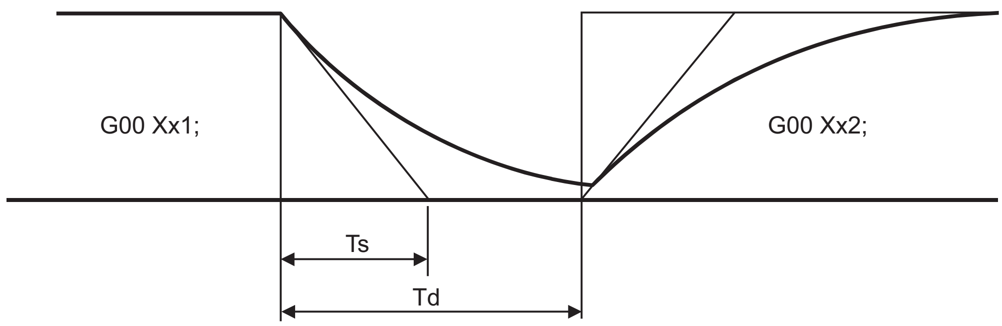
With deceleration check

(1) With linear acceleration/deceleration



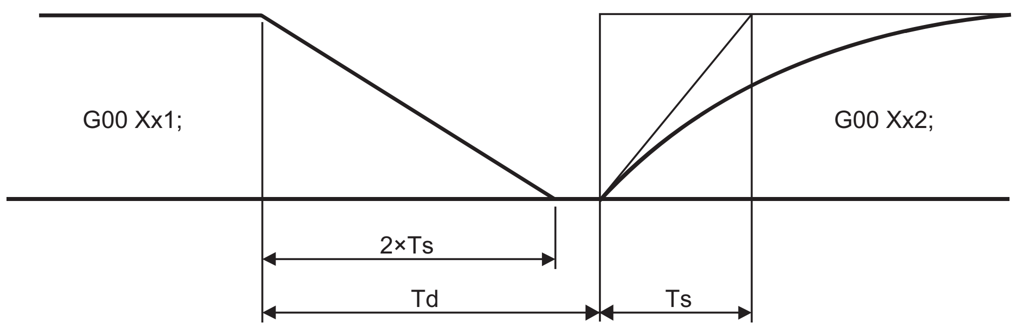
Ts: Acceleration/deceleration time constant   Td: Deceleration check time Td = Ts + α (0 to 10ms)

(2) With exponential acceleration/deceleration



Ts: Acceleration/deceleration time constant   Td: Deceleration check time Td = 2 x Ts + α (0 to 10ms)

(3) With exponential acceleration/linear deceleration



Ts: Acceleration/deceleration time constant   Td: Deceleration check time Td = 2 x Ts + α (0 to 10ms)

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

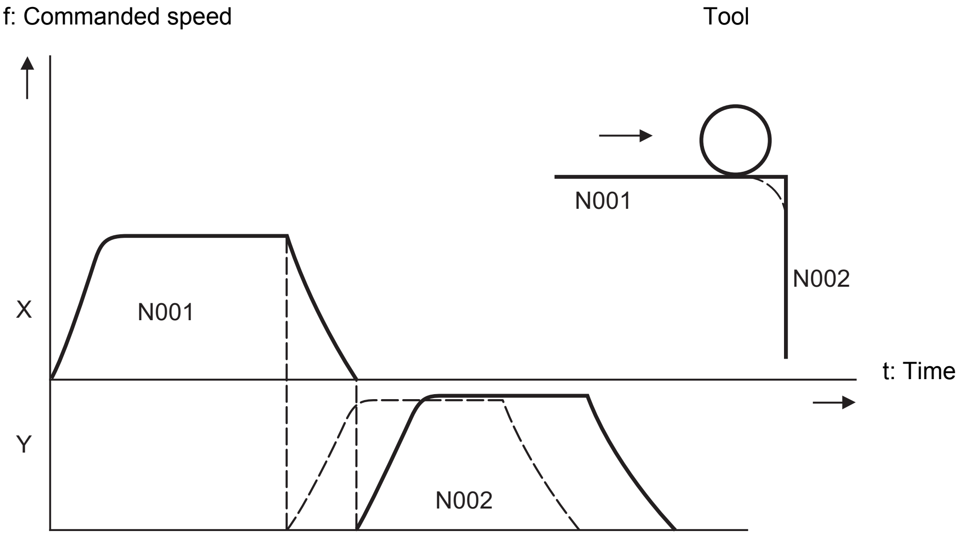
Note

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

Program example

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

[Exact stop check result]



Solid line indicates speed pattern with G09 command

Broken line indicates speed pattern without G09 command

7.11 Exact Stop Check Mode ; G61

Function and purpose

Whereas the G09 exact stop check command checks the in-position status only for the block in which the command has been assigned, the G61 command functions as a modal. This means that deceleration will apply at the end points of each block to all the cutting commands (G01 to G03) subsequent to G61 and that the in-position status will be checked.

The modal command is released by the following commands.

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

Command format

G61 ; ... Exact stop check mode

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

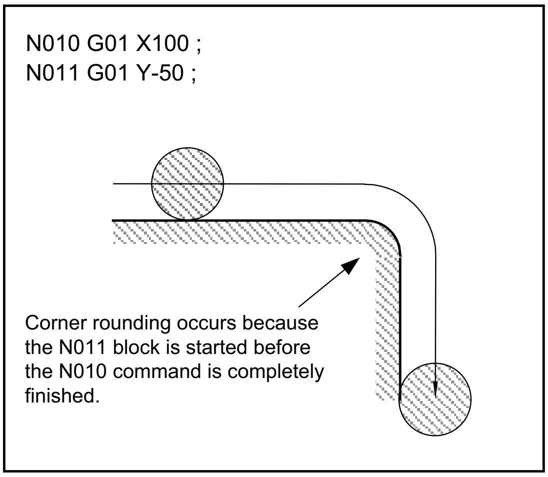
7.12 Deceleration Check

7.12.1 Deceleration Check

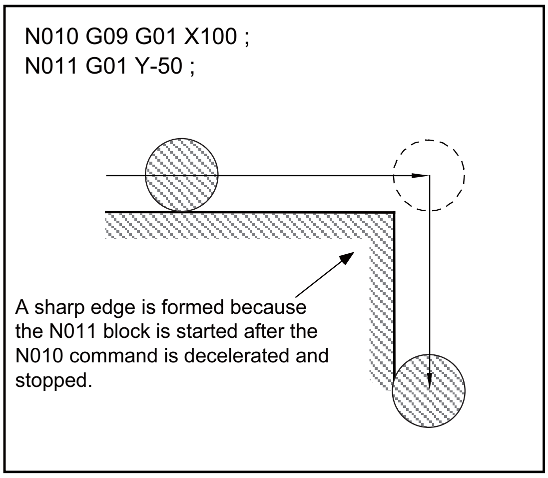
Function and purpose

The deceleration check reduces the machine shock that occurs when the control axis feedrate is suddenly changed and prevents corners from becoming rounded. This is accomplished by decelerating the motor to a stop at axis movement block joints before the next block is executed.

Without deceleration check



With deceleration check



The conditions for executing a deceleration check are described below.

(1) Deceleration check in the rapid traverse mode
In the rapid traverse mode, the deceleration check is always performed when block movement is completed before executing the next block.

(2) Deceleration check in the cutting feed mode
In the cutting feed mode, the deceleration check is performed and the program starts moving the next block when one of the following conditions is satisfied.
(a) When G61 (Exact stop check mode) is selected
(b) When the G09 (Exact stop check) command has been designated in the same block
(c) When the error detect switch (PLC signal) is ON

There are three methods for deceleration check: command deceleration check method, smoothing check method, and in-position check method.
The method that is selected for rapid traverse or cutting feed depends on the MTB specifications (combination of parameters "#1306 InpsTyp", "#1389 G1SmthChk", "#1223 aux07/bit1", and "#1193 inpos").
Depending on the MTB specifications, different deceleration check methods may be used for each feed command during rapid traverse command and cutting feed command (parameter "#1306 InpsTyp").

Detailed description

Behavior for each combination of movement commands

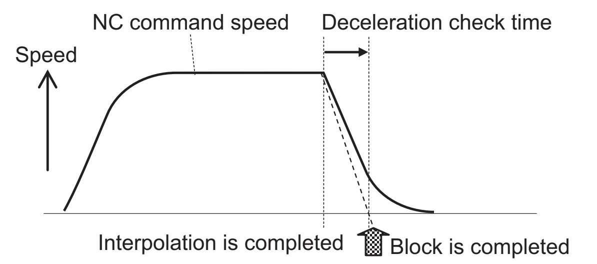
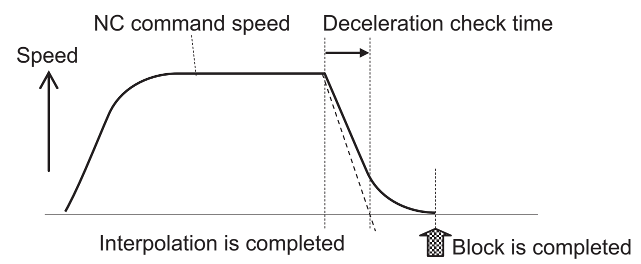
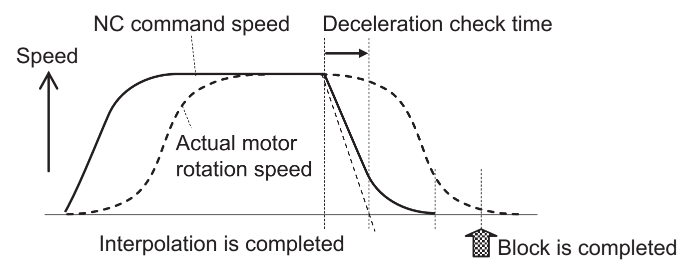
Next block	Current block		
	G00	G01	G00/G01 without movement
G00	○	(○) (*1)(*2)	×
G01	○	(○) (*1)(*3)	×
Others	○	(○) (*1)	×

○ : Deceleration check is performed.

(○): (*1) A deceleration check is performed when the error detection signal is ON or when G09 or G61 is enabled.
(*2) A command deceleration check is performed when G01 => G00 block is specified, "#1502 G0Ipfg" is ON or the movement reverses to the opposite direction.
(*3) A command deceleration check is performed when G01 => G01 block is specified, "#1503 G1Ipfg" is ON or the movement reverses to the opposite direction.
For the deceleration check when movement in the opposite direction is reversed, refer to "7.12.2 Deceleration Check when Movement in The Opposite Direction Is Reversed".
A deceleration check is not performed if the above conditions are not satisfied.

× : Deceleration check is not performed.

Types of deceleration check

(1) Command deceleration check method Deceleration is completed after the deceleration check time (*1) has passed after the interpolation.	
(2) Smoothing check method Deceleration is completed after the deceleration check time (*1) has passed after the interpolation and all axis smoothing has become zero.	
(3) In-position check method The deceleration is completed after the deceleration check time (*1) had passed after the interpolation, all axis smoothing has become zero, and all axes have become in-position.	

(*1) Deceleration check time is automatically calculated from the acceleration/deceleration mode and time constant.

Selecting deceleration checks (MTB specifications)

(1) When a rapid traverse command (G00/G53) block is to be executed

Parameters	Deceleration check method	Conditions of deceleration check
#1193 inpos		
0	Command deceleration check method	Deceleration check time has elapsed.
1	In-position check method	Deceleration check time has elapsed, all axis smoothing has become zero, and all axes have become in-position.
2	Smoothing check method	Deceleration check time has elapsed, and smoothing zero for all axes.

(2) When a cutting command (G01/G02/G03) block is to be executed
When parameter "#1306 InpsTyp" is "0", the following occurs (MTB specifications).

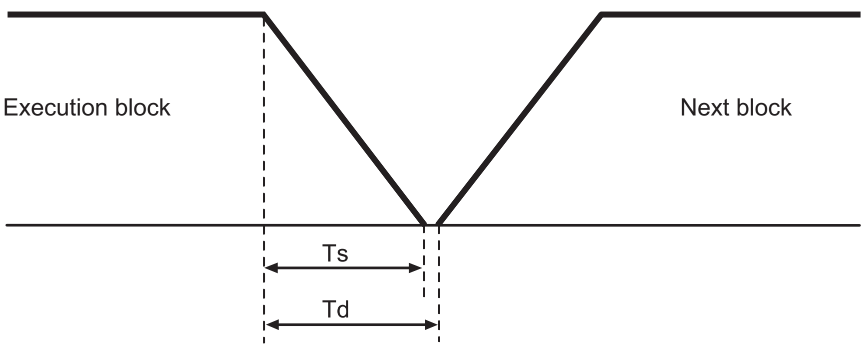
Parameters		Deceleration check method	Conditions of deceleration check
#1389 G1SmthChk	#1223 aux07/bit1		
0	0	Command deceleration check method	Deceleration check time has elapsed.
	1	In-position check method	Deceleration check time has elapsed, all axis smoothing has become zero, and all axes have become in-position.
1	-	Smoothing check method	Deceleration check time has elapsed, and smoothing zero for all axes.

When parameter "#1306 InpsTyp" is "1", the same method as for rapid traverse in (1) is used regardless of the value of parameter "#1389 G1SmthChk".

Command deceleration check method

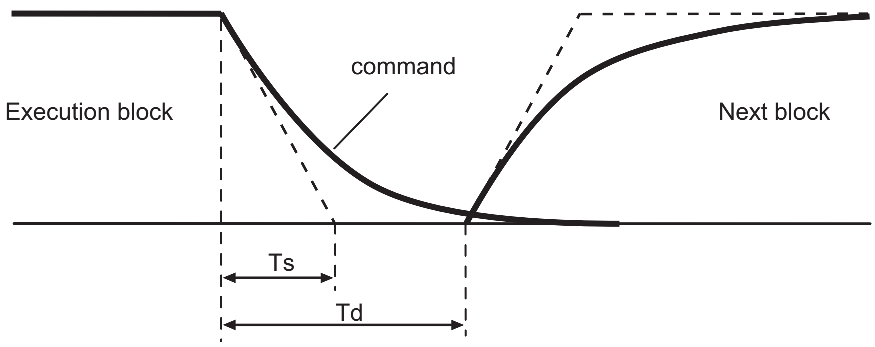
Execution of the next block starts after confirming that the deceleration of the command system is completed upon completion of interpolation for one block.
The following explains an example of transition from the current block (rapid traverse) to the next block.
The time required for the deceleration 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.

(a) For linear acceleration/deceleration



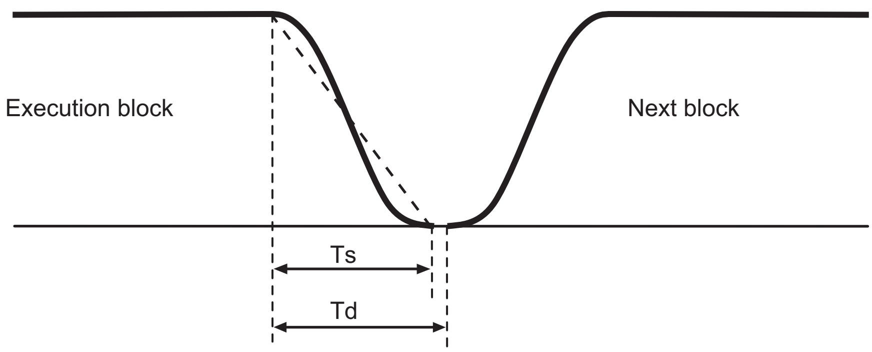
(Ts) Linear acceleration/deceleration time constant
(Td) Deceleration check time: Td = Ts + α (0 to 10 ms)

(b) For exponential acceleration/deceleration



(Ts) Exponential acceleration/deceleration time constant
(Td) Deceleration check time: Td = 2 x Ts + α (0 to 10 ms)

(c) For soft acceleration/deceleration

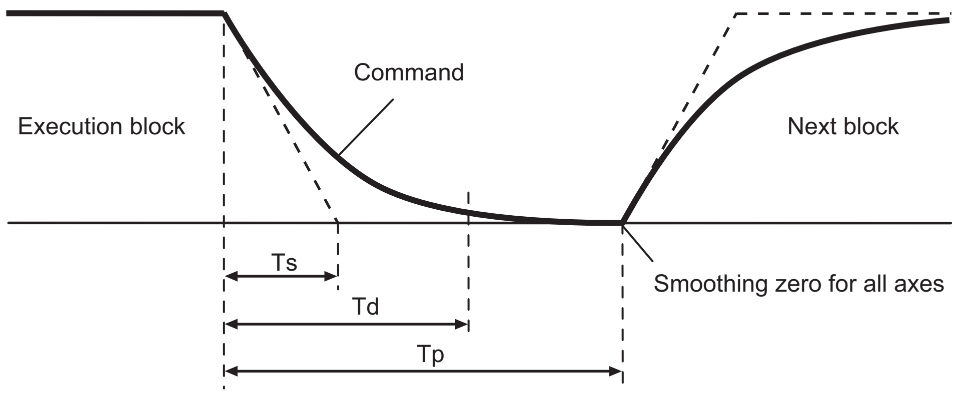


(Ts) Soft acceleration/deceleration time constant
(Td) Deceleration check time: Td = 2 x Ts + α (0 to 10 ms)

Smoothing check method

Execution of the next block starts after the command deceleration check is performed and after confirming that the smoothing for all axes in the part system has reached zero.

For exponential acceleration/deceleration

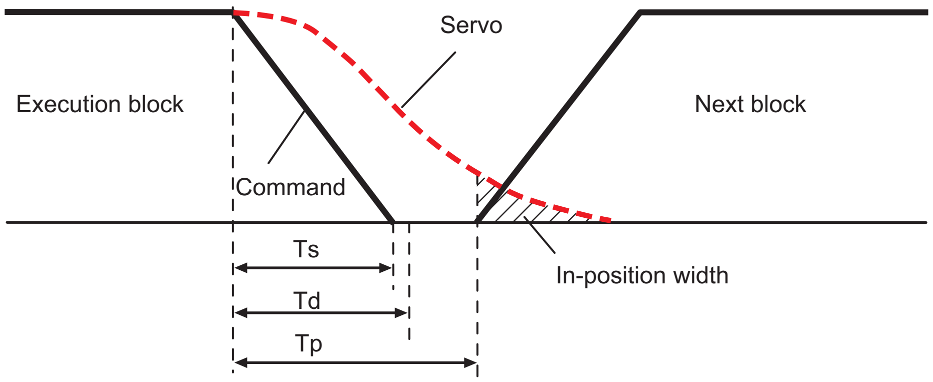


(Ts) Exponential acceleration/deceleration time constant
(Td) Deceleration check time
(Tp) Waiting time for a block to complete

In-position check method

Execution of the next block starts after the command deceleration check is performed and after confirming that the remaining distances for all axes in the part system are below certain values.
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.
(For a rotary axis, the setting value of spindle parameter "#13024 SP024" is assumed to be the in-position width.)

With linear acceleration/deceleration

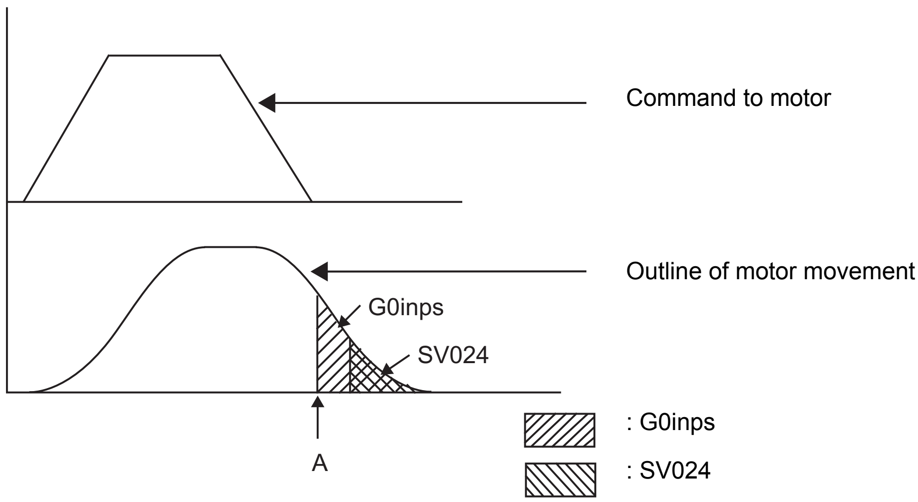


(Ts) Linear acceleration/deceleration time constant
(Td) Deceleration check time
(Tp) Waiting time for a block to complete

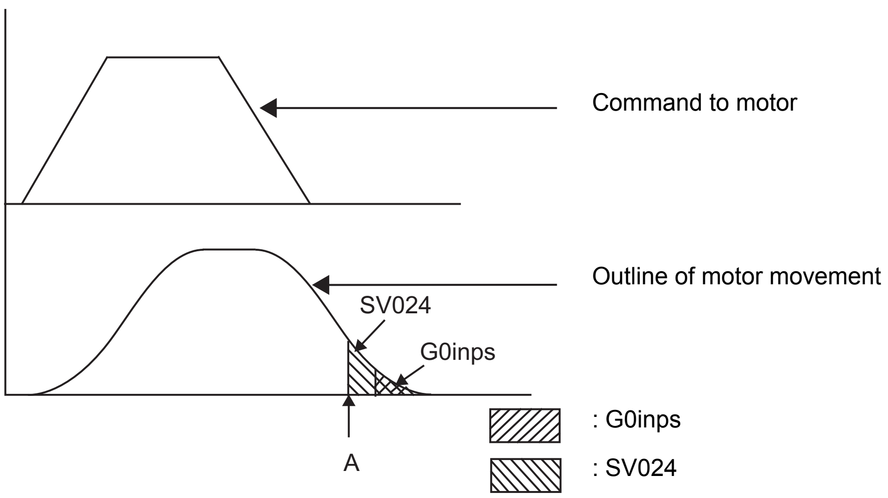
As shown in the figure above, the in-position width is the remaining distance from the previous block at the start of the next block. (Shaded area of the figure above).

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

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



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



Programmable in-position width command

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

G00 X_ Z_(Y_) ,I_ ;

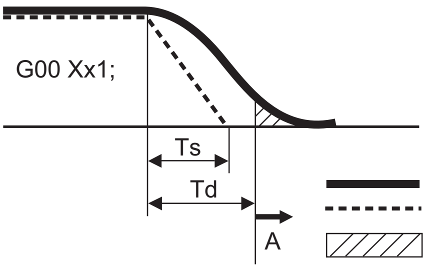
X,Z(,Y_)	Positioning coordinate value of each axis
,I	In-position width (setting range: 1 to 999999)

Execution of the next block starts after confirming that the position error amount in the block in which the deceleration check is to be performed is less than the in-position width.
The in-position width specified by parameter (SV024, G0inps (G1inps for G01)) or the one specified in the program, whichever is greater, will be adapted as the in-position width.
When there are several movement axes, the system confirms that the position error amount of each movement axis in each part system is less than the in-position width issued in this command before executing the next block.
For ",I" command, also refer to "6.1 Positioning (Rapid Traverse) ; G00".

The differences of In-position check

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

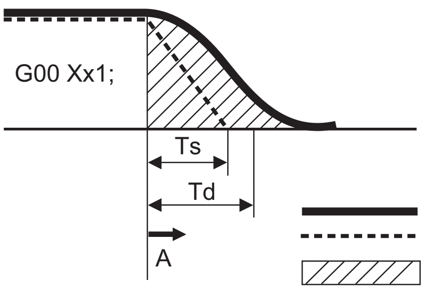
(1) In-position check with parameter
After completing deceleration of the command system ("A" in the figure), the servo system's position error amount and the parameter setting value (in-position width) are compared.



(Ts) Acceleration/deceleration time constant
(Td) Deceleration check time: Td = Ts + α (0 to 10 ms)

Servo machine position
Command
In-position width (Servo system position error amount)

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



(Ts) Acceleration/deceleration time constant
(Td) Deceleration check time: Td = Ts + α (0 to 10 ms)

Servo machine position
Command
In-position width (Servo system position error amount)

Relationship with Other Functions

Tool compensation

The deceleration check acts on the compensated block when tool compensation is performed.

Automatic error detection

The automatic error detection is disabled in a block in which deceleration check is enabled.

High-speed machining mode

When the axis movement reverses to the opposite direction in a G01 → G01 successive block during the high-speed machining mode other than high-speed machining mode I (G05 P1) the commanded deceleration will not take place even if parameter "#1503 G1Ipfg" is set to 1.
Note that the G0Ipfg setting will be followed if the axis direction reverses to the opposite direction in a G01 → G00 successive block.

High-speed simple program check

A deceleration check is performed even when high-speed simple program check is running. During high-speed simple program checking, the deceleration check time is reduced according to the time reduce coefficient.

Precautions

(1) When the in-position check is valid, the parameter for the in-position width "#2224 SV024" must be set. (Based on the MTB specifications.)

(2) This function is disabled for an axis to which automatic machine lock is applied.

(3) If MSTB is commanded in the block that follows a cutting command, the MSTB code is output before deceleration is completed in the cutting command. If an MSTB command must be executed after the completion of axis movement, check the PLC signals (DEN) before executing it. (The behavior depends on the MTB specifications.)

(4) If there is an axis in control axis synchronization/superimposition in the part system for which the in-position check method is specified, deceleration is considered to be completed when all axis smoothing has become zero. (Equivalent to smoothing check method)

(5) If thread cutting commands are specified in succession, a deceleration check is not carried out at block joints.

(6) If the parameter "#1205 G0bdcc" is set to "1", the value set with the parameter "#2224 SV024" (in-position detection width) will be used as the in-position width.
The setting of the parameter "#2077 G0inps" (G0 in-position width) and the programmable in-position check with ",I" address are disabled. These parameters depend on the MTB specifications.

7.12.2 Deceleration Check when Movement in The Opposite Direction Is Reversed

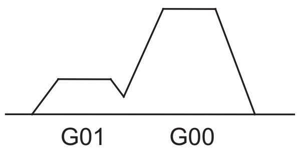
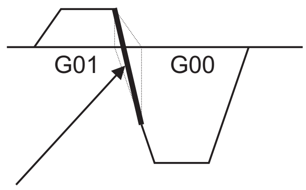
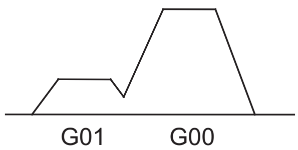
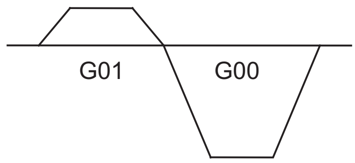
Function and purpose

A deceleration check cannot be designated for G01 -> G00 or G01 -> G01, but it can be designated in the following manner only when the movement reverses to the opposite direction in successive blocks.
A deceleration check can also be executed if even one axis is moving in the opposite direction while several axes are interpolating.
For the relation with other functions and precautions, refer to "Deceleration Check".

Detailed description

Designating deceleration check for G01 -> G00 opposite direction movement reversal

If the axis movement reverses to the opposite direction in a G01 to G00 successive block, the deceleration check for the movement in the opposite direction can be changed with the MTB specifications (parameter "#1502 G0Ipfg").

	Same direction	Opposite direction
G0Ipfg:0		 The acceleration is excessive due to the G01 and G00 composite speed.
G0Ipfg:1		 Command deceleration

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

(1)

G91 G01 X100. Y100. F4000 ; G00 X-100. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.

(2)

G91 G01 X100. Y-100. F4000 ; G00 X80. Y100. ;	A deceleration check is carried out, because the Y axis moves in the reverse direction in the program.

(3)

G90 G01 X100. Y100. F4000 ; G00 X80. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program. (When the program start position is X0 Y0)

(4)

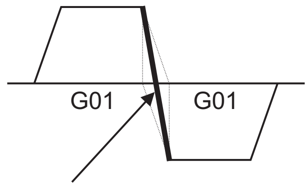
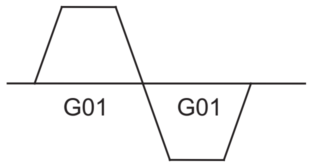
G91 G01 X100. Y100. F4000 ; G00 X100. Y100. ;	A deceleration check is not carried out, because both the X axis and the Y axis move in the same direction in the program.

(5)

G91 G01 X100. Y80. F4000 ; G00 X80. ;	A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Y axis movement command in the program.

Designating deceleration check for G01 -> G01 opposite direction movement reversal

If the axis movement reverses to the opposite direction in a G01 to G01 successive block, the deceleration check for the movement in the opposite direction can be changed with the MTB specifications (parameter "#1503 G1Ipfg").

	Same direction	Opposite direction
G1Ipfg:0		 The acceleration is excessive due to the G01 and G01 composite speed.
G1Ipfg:1		 Command deceleration

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

(1)

G91 G01 X100. Y100. F4000 ; G01 X-100. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program.

(2)

G91 G01 X100. Y-100. F4000 ; G01 X80. Y100. ;	A deceleration check is carried out, because the Y axis moves in the reverse direction in the program.

(3)

G90 G01 X100. Y100. F4000 ; G01 X80. Y120. ;	A deceleration check is carried out, because the X axis moves in the reverse direction in the program. (When the program start position is X0 Y0)

(4)

G91 G01 X100. Y100. F4000 ; G01 X100. Y100. ;	A deceleration check is not carried out, because both the X axis and the Y axis move in the same direction in the program.

(5)

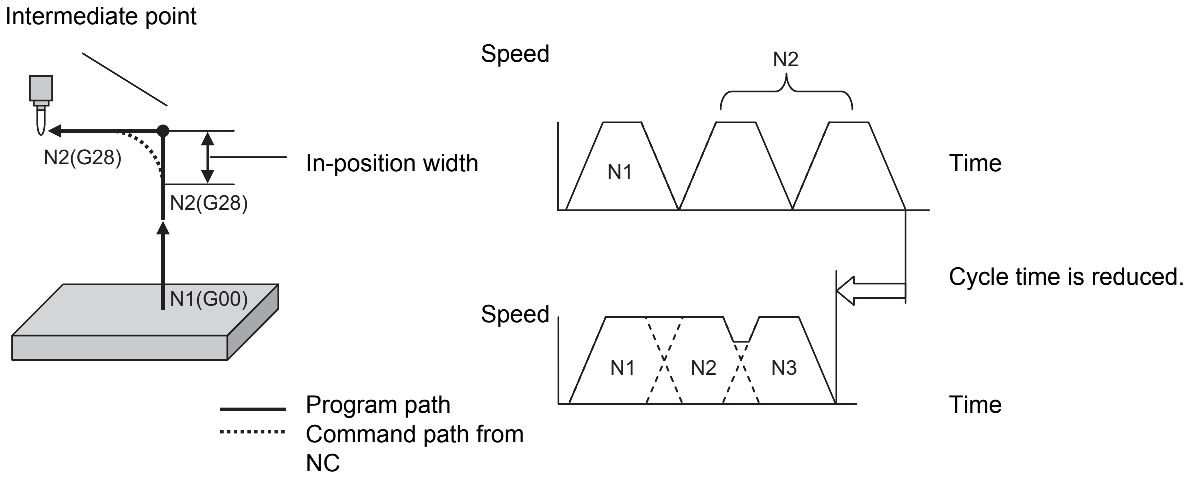
G91 G01 X100. Y80. F4000 ; G01 X80. ;	A deceleration check is not carried out, because the X axis moves in the same direction, and there is no Y axis movement command in the program.

7.13 Rapid Traverse Block Overlap; G0.5 P1

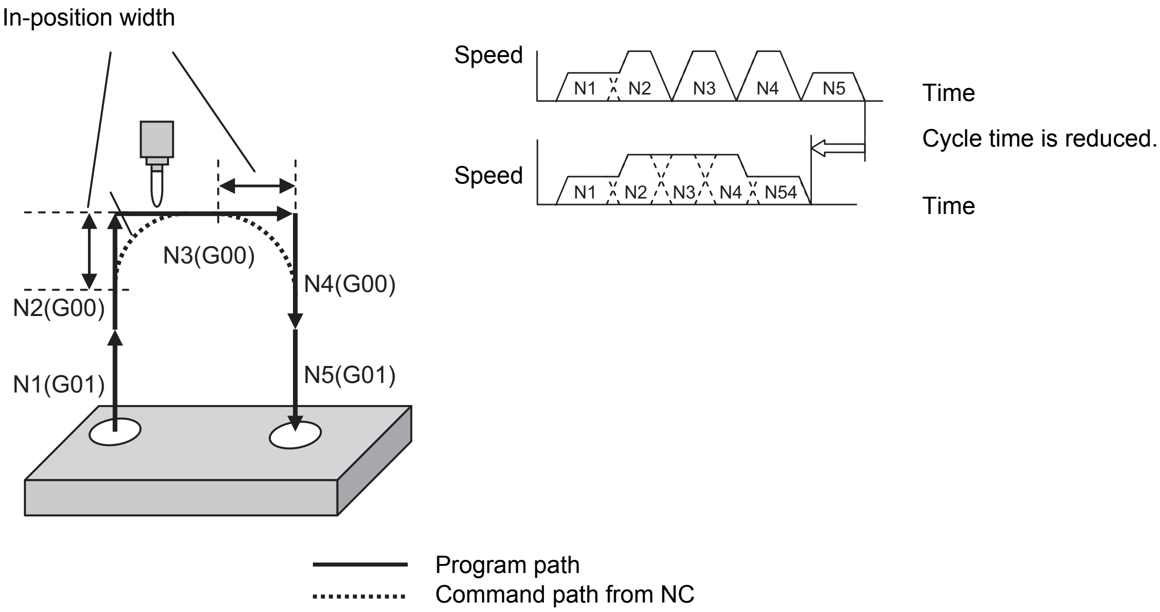
Function and purpose

This function enables the next block to start (overlap) without waiting for positioning (G00) or reference position return (G28/G30).
Consequently, cycle time of machining including operation of positioning (G00) or reference position return (G28/G30) can be reduced.
Adjust the overlap amount according to the command issued by the machining program or with the parameter, and specify it as in-position width for rapid traverse block overlap.
Also, the operation does not decelerate between blocks if the movement command continues in same direction.
The overlap is also valid when G00 is followed by a G01 block, rather than G00 or G28/G30.
It is not invalid when G28 is followed by G00 or G28/G30.
The validity of this function depends on the MTB specifications.

Example of behavior and velocity waveform 1 (example of application of rapid traverse block overlap in tool change motion)



Example of behavior and velocity waveform 2 (example of application of rapid traverse block overlap in continuous drilling motion)



Deceleration check method using in-position width

For a deceleration check method that uses the in-position width for rapid traverse (G00) or reference position return (G28/G30), a function with a higher priority that is enabled will be applied.
If none of the functions is enabled, the command deceleration is carried out.

Function (Deceleration check method)	Enabling conditions	Deceleration Check Enabled behavior	Priority
Programmable in-position	Valid when the in-position width is designated with address ",I" in the same block as G00. (It is valid only for a block in which address ",I" is specified.) (For details, refer to "6.1 Positioning (Rapid Traverse) ; G00" and "7.12 Deceleration Check".)	G00	1
Rapid traverse block overlap (this function)	(1) For G00 Parameter "#1442 G0ol" must be "1" and G00 rapid traverse block overlap must be valid modal code (G0.5P1). (2) For G28/G30 Parameter "#1443 G28ol" must be "1".	G00/G28/G30	2
In-position check by parameter settings	Parameter "#1193 inpos" must be "1". (For details, refer to "7.12 Deceleration Check".)	G00	3

For G00 overlap, refer to "7.13.1 Rapid Traverse Block Overlap for G00; G0.5". For G28/G30 overlap, refer to "7.13.2 Rapid Traverse Block Overlap for G28".

7.13.1 Rapid Traverse Block Overlap for G00; G0.5

Function and purpose

This function enables the next block to start (overlap) without waiting for positioning (G00) or reference position return (G28/G30).
For the rapid block overlap function, also refer to "7.13 Rapid Traverse Block Overlap; G0.5 P1".
G28/G30 can be overlapped when the rapid traverse block overlap for G28 is enabled. For details, refer to "7.13.2 Rapid Traverse Block Overlap for G28".

Command format

Starting rapid traverse block overlap for G00

```
G0.5 P1 J_ K_;
```

P	Starting or canceling the rapid traverse block overlap function (0: Cancel, 1: Start)
J	Liner axis in-position width (0.000 to 1000.000 (mm))
K	Rotary axis in-position width (0.000 to 1000.000 (°))

Canceling rapid traverse block overlap for G00

```
G0.5   P0;
```

Note

(1) A program error (P35) will occur unless this block is independently specified.
(2) This block can be specified simultaneously with an N code (sequence number).
(3) The in-position width at joints between two blocks containing G28/G30 cannot be changed with G0.5P1 command.
(4) G0.5P1 and G0.5P0 are modal.
(5) Address J in G20 must be programmed in inches.
(6) If an address is omitted, the width determined by the MTB specifications becomes valid. (Parameters "#2224 SV024" and "#13024 SP024")
If a value less than the width determined by the MTB is specified, that width becomes valid.
(7) If address J or K is set to "0", the conventional deceleration check is performed.

Detailed description

Enabling conditions

The rapid traverse block overlap function for G00 becomes effective when all of the following conditions are satisfied.

(1) The rapid traverse block overlap for G00 must be enabled.
Refer to the MTB specifications (parameter "#1442 G0ol").

(2) G0.5P1 modal must be active.
To make G0.5P1 modal active:

- Specify a G code (G0.5P1) in which rapid traverse block overlap is enabled in the machining program.
- Set parameter "#12056 I_G0oL G00" to "1" (valid).

N1 G0.5 P1;	Rapid traverse block overlap function: Enabled
N2 G91 G00 X10.;	
N3 G00 X20.;	
N4 G0.5 P0;	Rapid traverse block overlap function: Disabled
:	

Motion subject to rapid traverse block overlap

(1) When the rapid traverse block overlap for G00 is enabled, a G code (positioning (G00) or linear interpolation (G01)) following positioning (G00) may not be subject to rapid traverse block overlap depending on the current control mode or parameter settings that are specified by the MTB. (Parameters "#1086 G0intp" and "#1205 G0bdcc")
For details, refer to the table below.

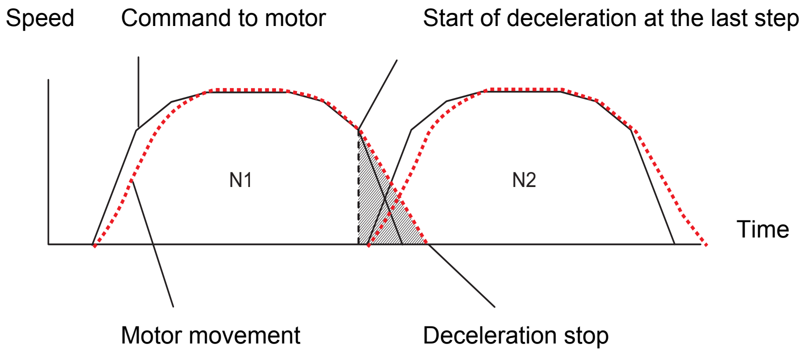
Control mode	Parameters		G code following positioning (G00)	
High-accuracy mode	#1086	#1205	G00	G01
OFF	0	0	○	○
		1	○	×
		2	○	×
	1	0/1/2	○	○
ON	0	0	○	×
		1	○	×
		2	○	×
	1	0/1/2	○	×

○ : Motion subject to rapid traverse block overlap for G00

× : Motion not subject to rapid traverse block overlap for G00

(2) When executing a rapid traverse block overlap in G00 multi-step acceleration/deceleration, the next block (N2 in the following program) will be started after the deceleration at the last step in the execution bock (N1) has started.
The hatched area in the figure below is an area in which the in-position width can be specified.

N1 G91 G00 X10.;
N2 X10.;



(3) When the rapid traverse block overlap for G00 is enabled, this function is valid if positioning (G00) is followed by a fixed cycle, subprogram or macro call command block.
In addition, this function is valid if a fixed cycle, subprogram or macro program contains consecutive move commands to which this function is applied.
(If the in-position width is specified in a fixed cycle command, that value is given priority.)

Adjustment of start position of overlap

The start position of overlap when a rapid traverse block overlap for G00 is executed can be adjusted with the in-position width. The next block is started when the remaining distances of all movement axes in the current movement block are smaller than the in-position width. (Refer to following figure.)
When setting the in-position width with J and K commands, set a value for each linear and rotary axis. Setting the in-position width for axes with parameter settings depend on the MTB specifications (parameter "#2631 G0olinps").

The start position of the next block based on the remaining distance and in-position width for each movement axis is shown below.
This shows an example of when the X axis in-position width is set to 0.5 mm and the Y axis in-position width to 1 mm.

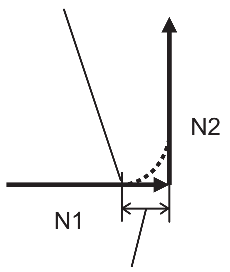
(a) For X axis

Program example

```
N1 G91 G00 X50.;
N2 Y50.;
```

Operation

Start position of N2 block



X axis in-position width = 0.5mm

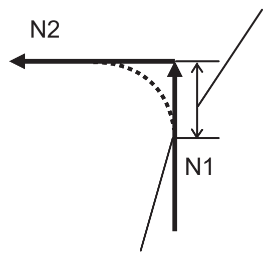
(b) For Y axis

Program example

```
N1 G91 G00 Y50.;
N2 X-50.;
```

Operation

Y axis in-position width = 1mm



Start position of N2 block

(c) For X and Y axes

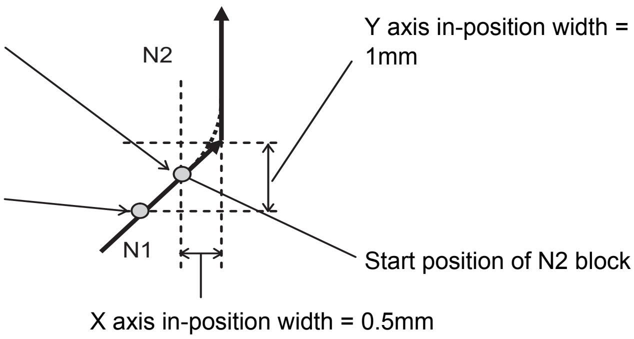
Program example

```
N1 G91 G00 X50. Y50.;
N2 X-50. Y50.;
```

Operation

If the position error amount is smaller than the in-position width for both of the X and Y axes, the next block is started.

If the Y axis position error amount is smaller than the in-position width, but the X axis position error amount is larger than that width, the next block is not started.



The in-position width is determined by the G code address or parameter value.

(1) When specifying the in-position width with a G code, the one specified with address J/K becomes effective. Note that if address J or K is set to "0", the rapid traverse overlap is disabled.

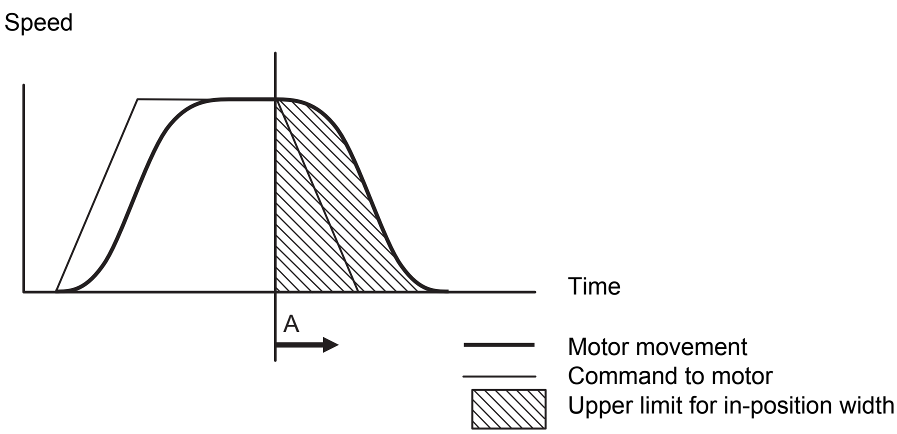
(2) If a command with address J/K is omitted, the in-position width determined for each of positioning and cutting feed by the MTB specifications becomes effective. (Parameters "#2631 G0olinps" and "#2632 G1olinps")

(a) Positioning (G00) - Positioning (G00): Parameter "#2631 G0olinps"

(b) Positioning (G00) - Cutting feed (G01)(high-accuracy mode is OFF): Parameter "#2632 G1olinps"

Upper and lower limits for in-position width

(1) Upper limit for in-position width
When rapid traverse block overlap is enabled, the in-position check is performed after starting deceleration specified in the speed command ("A" in the figure). Thus, the distance from the servo machine position after starting the command deceleration to the commanded position (hatched area in the figure below) is the upper limit for the actual in-position width.



(2) Lower limit for in-position width
The lower limit for the in-position width depends on the MTB specifications (parameters "#2224 SV024" or "#13024 SP024").
The value of this parameter is applied even if a value less than or equal to this parameter is specified as an in-position width.

Compensation for in-position width based on the path

The conventional deceleration check (in-position check method) applies the same in-position width regardless of the path (corner angle). Therefore, an extra speed change occurs and cycle time is increased even though path direction stays almost the same. The rapid traverse block overlap automatically compensates for the in-position width based on the path (corner angle).
However, the in-position width is not compensated for based on the path if a block without a movement command is inserted between the movement commands to be overlapped.

(1) If the angle is greater than 90°, the rapid traverse overlap function is temporarily canceled.

(2) If the angle is less than 90°, the in-position width is compensated for so that it matches the amount of droop at a corner when the corner angle is 90°.

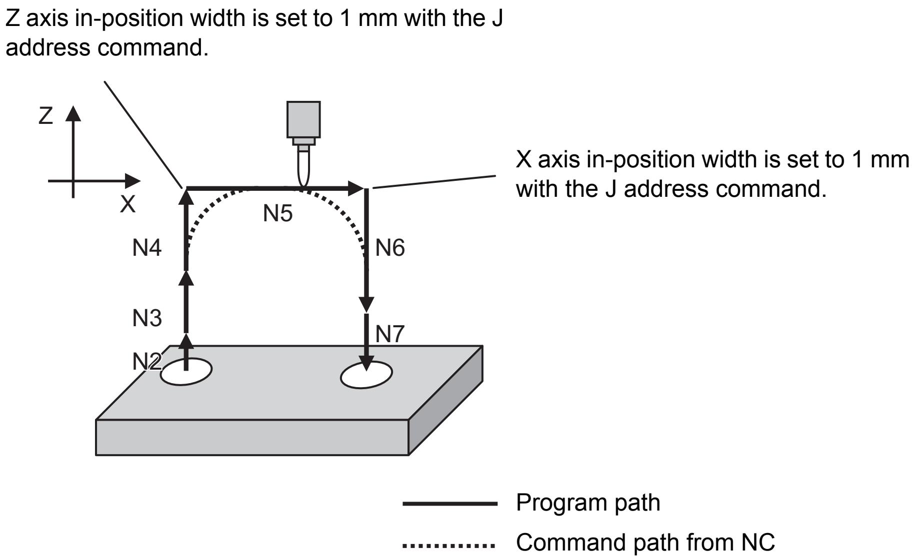
Program example

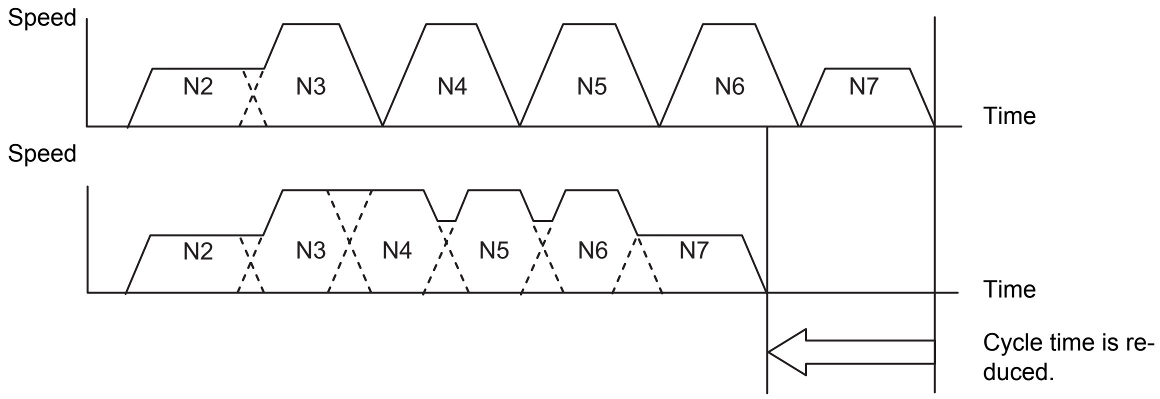
When the in-position width is specified with address J (G0.5P1 J_)

The following are examples of using G00 rapid traverse block overlap in combination with G00 (rapid traverse) and G01 (cutting feed). (When the high-accuracy control mode is OFF)

Parameter setting	X axis	Z axis
#2631(G0olinps)	2mm	1.5mm
#2632(G1olinps)	1 mm	0.5 mm

```
N1 G0.5P1 J1.0;
N2 G91 G01 Z25.;
N3 G00 Z25.;
N4 G00 Z50.;
N5 G00 X125.;
N6 G00 Z-75.;
N7 G01 Z-25.
F1000.;
```

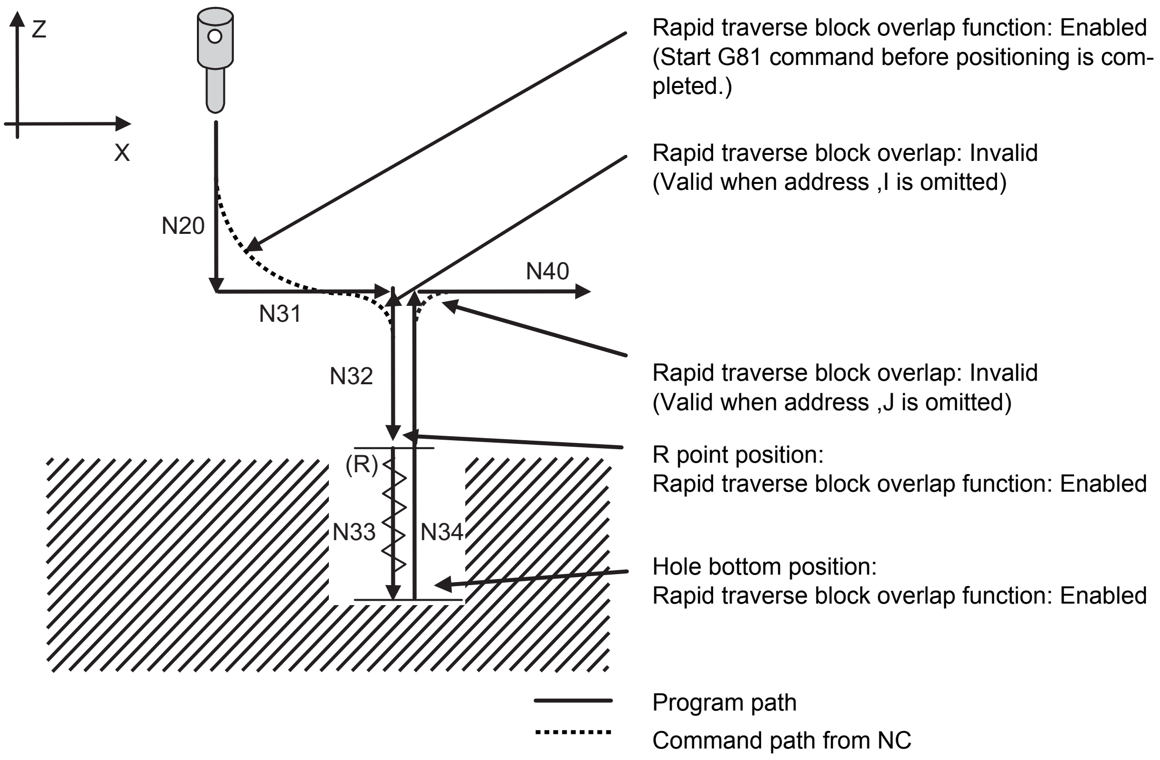




Example behavior in fixed cycle

When specifying G00 (positioning) -> G81 (drilling)

(Main program)	(G81 program)
N10 G0.5 P1 J0.5;	N31 G00 X50. Y0.;
N20 G91 G98 G64 G00 X50.;	N32 G00 Z-25.;
N30 G81 X50. Y0. Z-25. R-25. F1000. L1. ,I2.0 ,J1.0;	N33 G01 Z-25. F1000.;
N40 G00 X50. ;	N34 G00 Z50.;

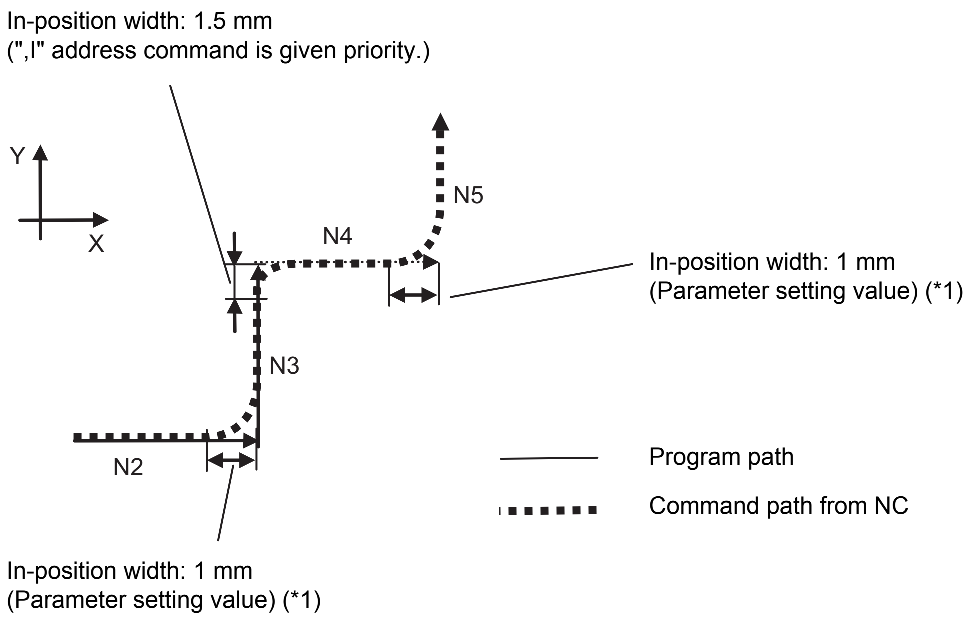


Relationship with Other Functions

Programmable in-position

If an ",I" address command is used to specify the in-position width from the program when the rapid traverse block overlap is enabled, the in-position width of programmable in-position is given priority.
Because the programmable in-position is an unmodal command, the in-position width specified with the rapid traverse block overlap enabled is assumed for commands following ",I" address.
This shows an example of when the X and Y axis in-position widths for G00 are set to 1 mm by parameters.

N1 G0.5 P1;	G0.5 command (for G00)
N2 G91 G00 X50.;	Rapid traverse block overlap for G00: Valid
N3 Y50. ,I1.5;	",I" address command is valid
N4 X50.;	Rapid traverse block overlap for G00: Valid
N5 Y50.;	
:	



(*1) The in-position width is the parameter setting value because the programmable in-position is an unmodal command.

Note

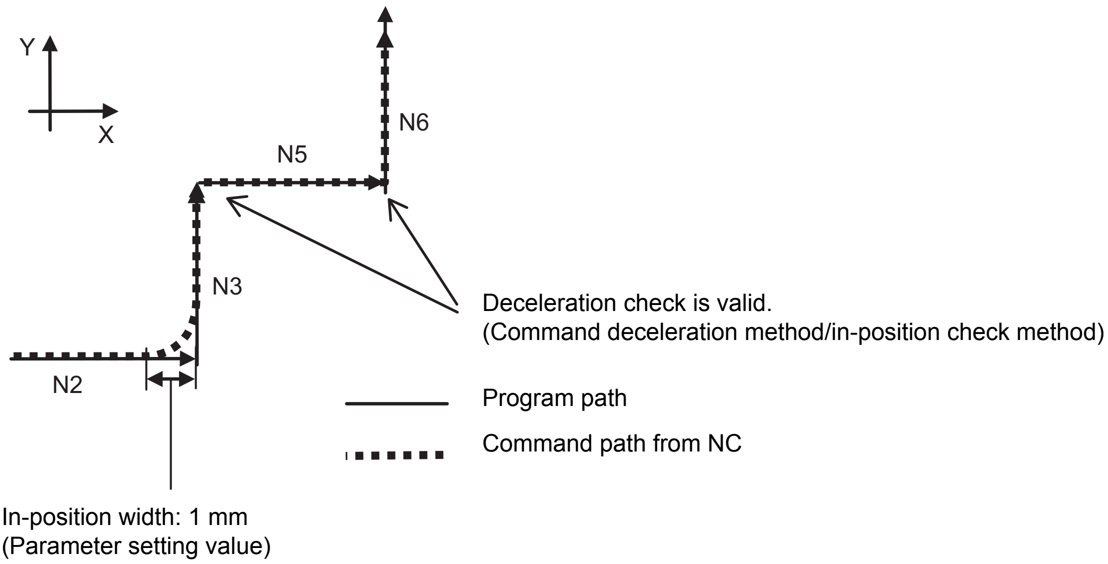
(1) The programmable in-position (",I" command) for G00 pre-interpolation acceleration/deceleration can only be used when the rapid traverse block overlap is enabled.

(2) When G00 is followed by a block without a movement command, a command of address ",I", if specified for G00, is handled as a command specifying a rapid traverse overlap. Therefore, the overlap takes place only when overlapped movements are executed.

Deceleration Check

When the rapid traverse block overlap is enabled, the conventional deceleration check is disabled for the behavior subject to this function.
When the rapid traverse block overlap is disabled, the conventional deceleration check is enabled.
This shows an example of when the X and Y axis in-position widths for G00 are set to 1 mm by parameters.

N1 G0.5 P1;	G0.5 command (for G00)
N2 G91 G00 X50.;	Rapid traverse block overlap for G00 is valid
N3 Y50.;	Deceleration check is valid.
N4 G0.5 P0;	
N5 X50.;	Deceleration check is valid.
N6 Y50.;	Deceleration check is valid.
:	



Precautions

(1) When a block without a movement command is inserted between blocks that are subject to the rapid traverse block overlap, blocks are overlapped if the high-accuracy mode is OFF (they are not overlapped if the mode is ON).
If the high-accuracy mode is OFF, a block without movement that is inserted between a G00 command and G28/30 block is not overlapped when the rapid traverse block overlap for G00 is disabled ("#1442 G0ol" is OFF) and rapid traverse block overlap for G28 is enabled ("#1443 G28ol" is ON).

(2) When a block without a movement command is inserted between blocks that are subject to the rapid traverse block overlap, the in-position width is not compensated for based on the path.

(3) When the high-accuracy control mode is selected or the parameter #1205 is set to "1" or "2", the next block will not be performed until the speed is reduced below the rapid speed (#2001) if the speed at the completion of in-position check is higher than the rapid speed (parameter #2001) of the next block.

(4) Even when the overlap process blocks continue, if one or more axes are moved in reversed direction, the overlap function is temporarily canceled.

7.13.2 Rapid Traverse Block Overlap for G28

Function and purpose

This function enables the next block to start (overlap) without waiting for positioning (G00) or reference position return (G28/G30).
For the rapid block overlap function, also refer to "7.13 Rapid Traverse Block Overlap; G0.5 P1".
G00 can be overlapped when the rapid traverse block overlap for G00 is enabled. For details, refer to "7.13.1 Rapid Traverse Block Overlap for G00; G0.5".

Detailed description

Enabling conditions

The rapid traverse block overlap function for G28 becomes effective when all of the following conditions are satisfied.

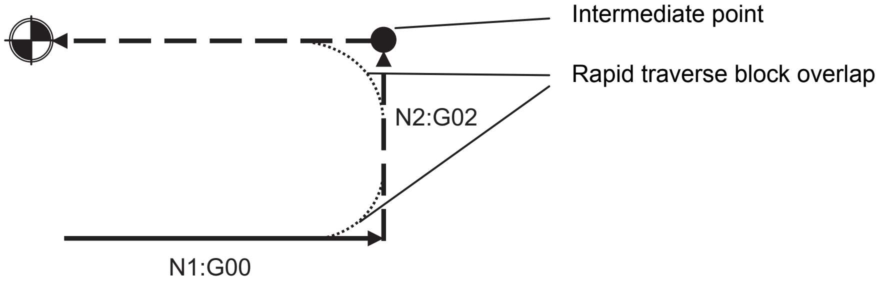
(1) The rapid traverse block overlap for G28 is enabled.
(Refer to the MTB specifications. "#1443 G28ol")

(2) High-speed reference position return is active. (Dog-type is not subject to this.)

(3) When the rapid traverse block overlap for G00 is enabled, a G00 command is followed by G28 or G30 positioning command.

Note

- For G28/G30, whether or not the appropriate block, if its movement is made via an intermediate point, is overlapped depends on the MTB specifications (parameters "#1205 G0bdcc" and "#1086 G0intp").
- If G28/G30 command is followed by another G28/G30, blocks are not overlapped in rapid traverse. (Blocks are not overlapped.



Adjustment of start position of overlap

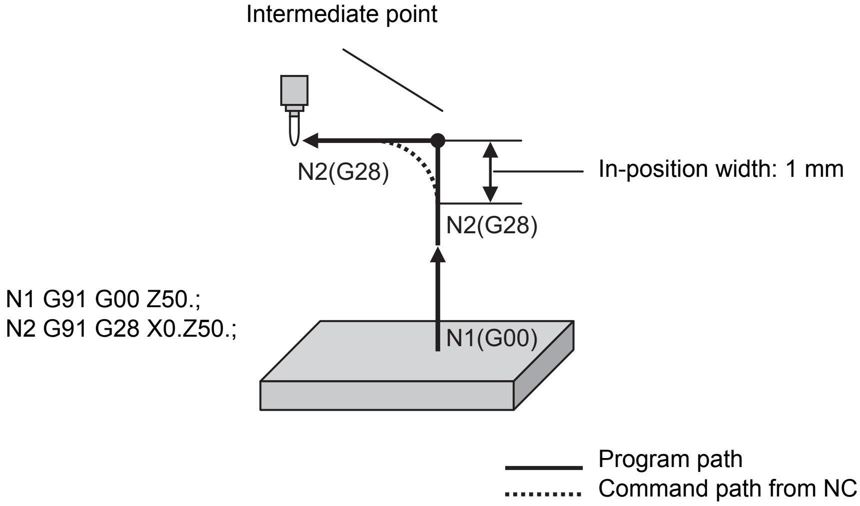
The start position of overlap when a rapid traverse block overlap for G28 is executed can be adjusted with the in-position width. The next block is started when the remaining distances of all movement axes in the current movement block are smaller than the in-position width.
The in-position width depends on the MTB specifications (parameter "#2633 G28olinps").

Program example

The following are examples of using rapid traverse block overlap for G28 in combination with G28/G30 (reference position return) and G00 (rapid traverse).

Parameter setting	X axis	Z axis
#2633 G28olinps	0.5 mm	1 mm

Initial position of axes: X axis = -50 mm; Z axis = -100 mm



Relationship with Other Functions

Refer to "7.13 Rapid Traverse Block Overlap; G0.5 P1".

Precautions

Refer to "7.13 Rapid Traverse Block Overlap; G0.5 P1".

7.14 Automatic Corner Override

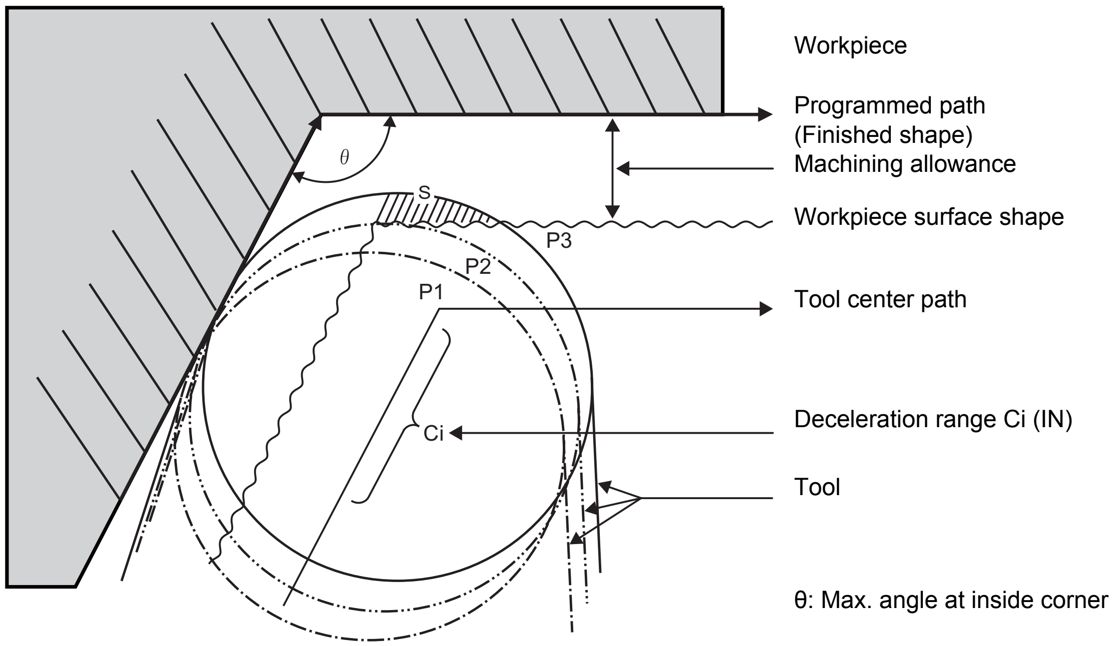
Function and purpose

With tool radius compensation, this function reduces the load during inside cutting of automatic corner R, or during inside corner cutting, by automatically applying override to the feedrate.
There are two types of automatic corner override: Automatic corner override (G62) and inner arc override.
Automatic corner override (G62) is valid until the tool radius compensation cancel (G40), exact stop check mode (G61), high-accuracy control mode (G61.1), tapping mode (G63), or cutting mode (G64) command is issued.
The inner arc override is valid whenever the machine is in the tool radius compensation mode (G41/G42), regardless of the automatic corner override (G62) mode.

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.



[Operation]

(1) If there is no G62 command:
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 compared to P2 and so that tool load increases.

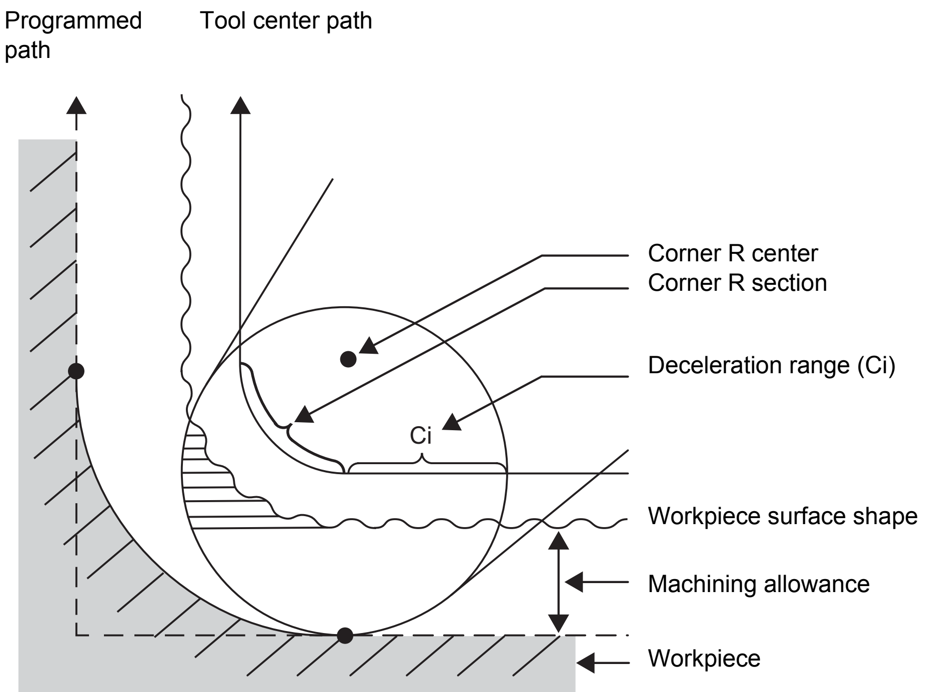
(2) If there is G62 command:
When the inside corner angle θ in the above figure is less than the angle set in the parameter, the override set into the parameter is automatically applied in the deceleration range Ci.

[Parameter setting]

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

#	Parameters	Setting range
#8007	Override	0 to 100 [%]
#8008	MAX ANGLE	0 to 180 [°]
#8009	DSC.ZONE	0 to 99999.999 [mm] or 0 to 3937.000 [inches]

Automatic corner R



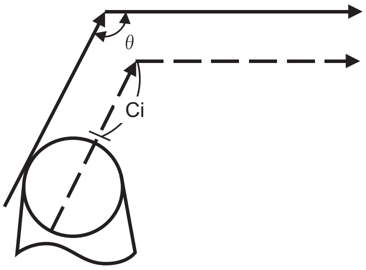
(1) The override set in the parameter is automatically applied at the deceleration range Ci and corner R section for inside offset with automatic corner R. (There is no angle check.)

Application example

The lines in the figure denote:

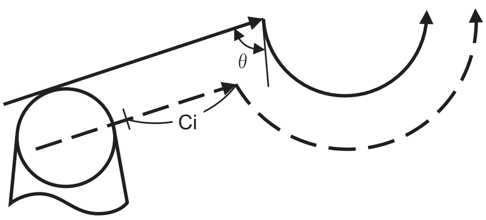
- Program path
- Tool center
- Arc (inside offset)

(1) Linear - linear corner



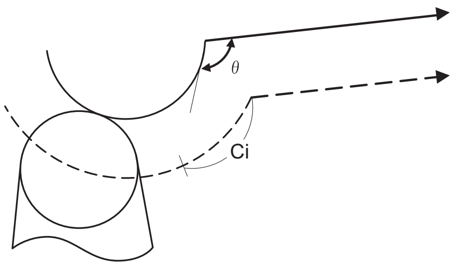
The override set in the parameter (#8007) is applied in the deceleration range Ci.

(2) Linear - arc (outside offset) corner



The override set in the parameter (#8007) is applied in the deceleration range Ci.

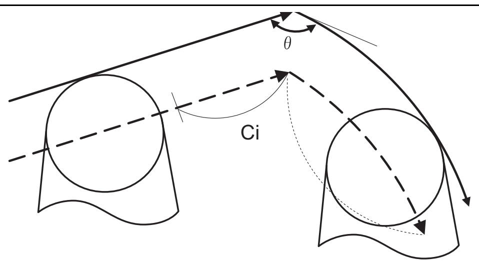
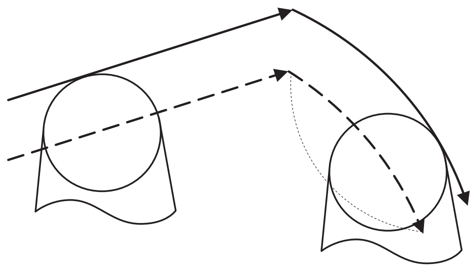
(3) Arc (outside offset) - linear corner



<Note>

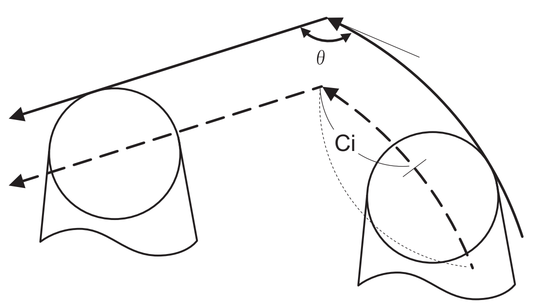
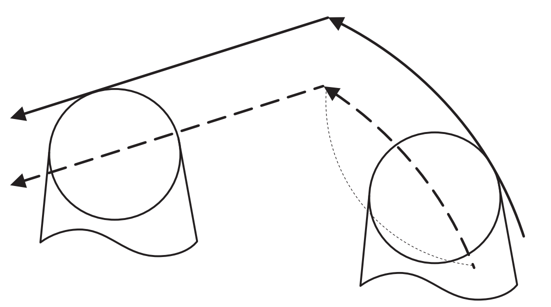
- The deceleration range Ci where the override is applied is the length of the arc with an arc command.

(4) Linear - arc (inside offset) corner

(a) If there is a G62 command:	(b) If there is no G62 command:
 	
For straight lines, the override set in the parameter (#8007) is applied in the deceleration range Ci.	

During cutting of arc (inside offset), an inner arc override is applied.

(5) Arc (inside offset) - linear corner

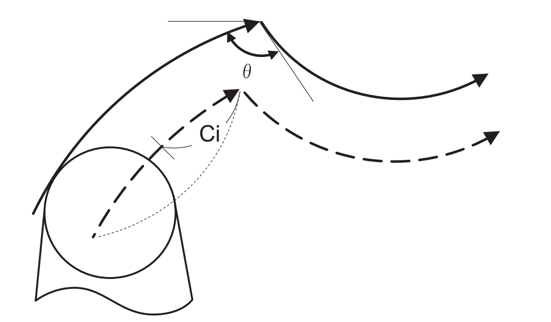
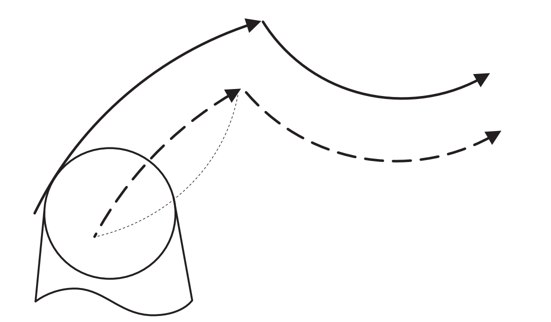
(a) If there is a G62 command:	(b) If there is no G62 command:
	
In addition to the inner arc override, the override set in the parameter (#8007) is applied in the deceleration range Ci. (*1) F (speed command value) x (inner arc override) x (setting for #8007)	

(*1) The deceleration range Ci where the override is applied is the length of the arc with an arc command.

During cutting of arc (inside offset), an inner arc override is applied.

Automatic corner override will not be applied to straight lines.

(6) Arc (inside offset) - arc (outside offset) corner

(a) If there is a G62 command:	(b) If there is no G62 command:
	
In addition to the inner arc override, the override set in the parameter (#8007) is applied in the deceleration range Ci. (*1) F (speed command value) x (inner arc override) x (setting for #8007)	

(*1) The deceleration range Ci where the override is applied is the length of the arc with an arc command.

During cutting of arc (inside offset), an inner arc override is applied.

Automatic corner override will not be applied to straight lines.

Relationship with Other Functions

Function	Operation of automatic corner override (G62)	Operation of inner arc override
F1-digit Feed	Automatic corner override is applied to the F1-digit speed.	Same as on the left.
Cutting feed override	Cutting feed override is applied to automatic corner override.	Same as on the left.
Override cancel	Automatic corner override will not be canceled by override cancel.	Same as on the left.
External deceleration	External deceleration speed will be applied after automatic corner override is applied to the cutting feedrate.	Same as on the left.
Speed clamp	Clamp speed will be applied after automatic corner override is applied to the cutting feedrate.	Same as on the left.
Dry run	Automatic corner override will not be applied.	Same as on the left.
Synchronous feed	Automatic corner override is applied to the synchronous feedrate.	Same as on the left.
Thread cutting	Automatic corner override will not be applied.	Same as on the left.
G31 Skip	Program error occurs with G31 command during tool radius compensation.	Same as on the left.
Machine lock	Automatic corner override is applied even in the machine lock state.	Same as on the left.
Positioning (G00)	Automatic corner override is not applied to the positioning command.	Same as on the left.
Linear interpolation (G01)	Automatic corner override is applied to linear interpolation.	Inner arc override will not be applied to linear interpolation.
Circular Interpolation (G02, G03)	Automatic corner override is applied to circular interpolation.	Same as on the left.
Spiral/conical interpolation (G02.1,G03.1)	Automatic corner override is applied to spiral/conical interpolation.	Same as on the left.
Involute Interpolation (G02.2, G03.2)	Automatic corner override is applied to involute interpolation.	Inner arc override will not be applied to involute interpolation (*1).
High-speed high-accuracy control (G08P1)	If G62 is commanded during high-accuracy control, a program error (P126) will occur and automatic corner override will not be applied.	Inner arc override will not be applied during high-accuracy control.
High-speed high-accuracy control II (G05P10000)	Automatic corner override is applied during high-speed high-accuracy control II.	Same as on the left.
Tool radius compensation cancel (G40)	Automatic corner override will not be applied while the tool radius compensation is being canceled.	Same as on the left.
Feedrate override OFF (#3004 bit1 = ON)	Automatic corner override will not be applied while the feedrate override is invalid.	Inner arc override is applied while the feedrate override is invalid.

(* 1) Involute interpolation provides involute interpolation override (equivalent to inner arc override) to adjust the speed so that the tool center speed does not exceed the lower limit (#1558 IvOMin) of override when tool radius compensation inside is set.
To designate the minimum override value for inner arc cutting in involute interpolation, set the parameter above. (This parameter setting depends on the MTB specifications.)

Precautions

(1) Automatic corner override (G62) is valid only in the G01, G02, and G03 modes; it is not effective in the G00 mode. When switching from the G00 mode to the G01 (or G02 or G03) mode at a corner (or vice versa), automatic corner override will not be applied at that corner in the G00 block.

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

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

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

(5) Automatic corner override will not be applied when intersection calculation cannot be executed.
Intersection calculation cannot be executed in the following case.
- 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) When the parameters are set as shown below, the automatic corner override (G62) or inner arc override is disabled.

(a) Conditions that disable the automatic corner override (G62)
- #8007 (override) is set to 0 or 100
- #8008 (max. angle) is set to 0 or 180
- #8009 (DSC.ZONE) is set to 0

(b) Condition that disables the inner arc override
- #19418 (minimum OVR for inner arc) is set to 0 or 100

(9) The inclined surface machining command is not available during automatic corner override modal.
To perform inclined surface machining, command G64 (cutting mode) in advance, then cancel the modal mode.

7.14.1 Automatic Corner Override ; G62

Command format

G62 ; ... Automatic Corner Override

Automatic corner override (G62) is valid until the nose R compensation cancel (G40), exact stop check mode (G61), high-accuracy control mode (G61.1), tapping mode (G63), or cutting mode (G64) command is issued.
For detailed description, execution example, the relationship with other functions and precautions, refer to "7.15 Automatic Corner Override".

7.14.2 Inner Arc Override

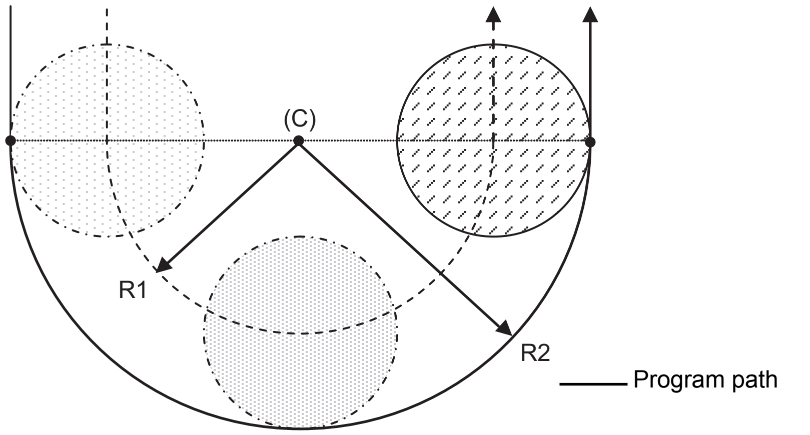
Detailed description

When cutting an arc (inside offset), applying the override given by the following expression to the commanded feedrate (F) causes that feedrate to become the F value for which the feedrate of the programmed path is commanded.

$$F \times \frac{R1}{R2}$$

R1: Radius of tool center path
R2: Radius of program path

The inner arc override is valid whenever the machine is in the tool radius compensation mode (G41/G42), regardless of the automatic corner override (G62) mode.
Inner arc override will not be applied when the machine is in automatic corner R.



(C) Arc center
R1: Radius of tool center path
R2: Radius of program path

If the radius (R1) of tool center path is very small compared to the radius (R2) of program path, R1/R2 is nearly equal to 0, causing the tool feed to stop.
To prevent the tool feed from being stopped, set the parameter "#19418" (minimum OVR for inner arc) and set the tool feedrate to F x (setting for #19418) when the value of R1/R2 is equal to or less than (setting for #19418).
The inner arc override is invalid when the #19418 (minimum OVR for inner arc) is set to 0 or 100.

For detailed description, execution example, the relationship with other functions and precautions, refer to "7.14 Automatic Corner Override".

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), high-accuracy control mode (G61.1), automatic corner override (G62), or cutting mode (G64) command.
The machine is in the cutting mode status when its power is turned ON.

Command format

G63; ... Tapping mode

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

Command format

G64; ... Cutting mode

8

Dwell

8.1 Dwell (Time Designation) ; G04

Function and purpose

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

Command format

Dwell (Time-based designation)

G94 G04 X__ /P__ ;

X/P	Dwell time

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

Detailed description

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

(2) When designating the dwell time with P, the availability of the decimal point command can be selected with the parameter (#8112). When the decimal point command is set to be invalid, the command value below the decimal point with P is ignored.

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

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

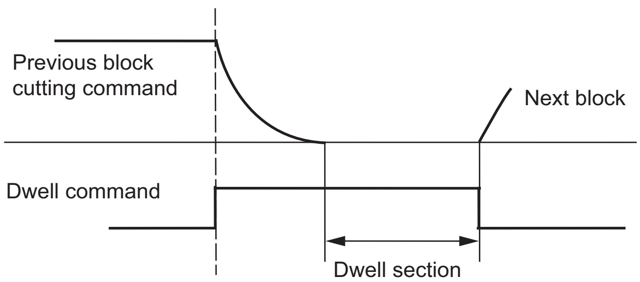
(4) The dwell time setting unit applied when there is no decimal point can be made 1s by setting 1 in the parameter "#1078 Decpt2". This is effective only for X and P for which the decimal command is valid.

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

(6) The dwell is valid during the interlock.

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

(8) Dwell can be canceled depending on the MTB specifications (parameter "#1173 dwlskp"). If the set skip signal is input during the dwell time, the remaining time is discarded, and the following block will be executed.



Program example

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

Note

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

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

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

Precautions and restrictions

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

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, 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 depends on the MTB specifications (parameter "#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 2nd 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 MTB for the actual correspondence between the functions and numerical values.
When the M00, M01, M02, and M30 functions are used, the next block is not read into the pre-read buffer due to pre-read inhibiting.
If the M function is designated in the same block as a movement command, the commands may be executed in either of the following two orders. The machine specifications determine which sequence applies.

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

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

Program stop : M00

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

Optional stop : M01

If the M01 command is read when the optional stop switch on the machine operation board is ON, it will stop reading the next block and perform the same operation as the M00.
If the optional stop switch is OFF, the M01 command is ignored.
(Example)

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

Program end : M02 or M30

This command is normally used in the final block for completing the machining, and so it is primarily used for cueing up the machining program. Whether the program is actually cued up or not depends on the machine specifications. Depending on the machine specifications, the system is reset by the M02 or M30 command upon completion of cueing up the program and any other commands issued in the same block.
(Although the contents of the command position display counter are not cleared by this reset action, the modal commands and compensation amounts are canceled.)
The next operation stops when the cueing up operation is completed (the in-automatic operation lamp goes off).

To restart the unit, the automatic start button must be pressed or similar steps must be taken. When the program is restarted after M02 and M30 are completed, if the first movement command is designated only with a coordinate word, the interpolation mode will function when the program ends. It is recommended that a G function always be designated for the movement command designated first.

Note

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

Macro interruption; M96, M97

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

Subprogram call/completion : M98, M99

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

Internal processing with M00/M01/M02/M30 commands

Internal processing suspends pre-reading when the M00, M01, M02 or M30 command has been read. Other 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 MTB determines which codes correspond to which positions.

Detailed description

The address that is used for the 2nd miscellaneous function (A, B, or C) depends on the MTB specifications (parameter "#1170 M2name"). (Except the address that is used for the axis name.) The 2nd 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 2nd 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 2nd miscellaneous functions.
The table below gives address combinations. It is not possible to use an address that is the same for the axis name of an additional axis and 2nd miscellaneous function.

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

Precautions

(1) When A has been assigned as the 2nd miscellaneous function address, the following commands cannot be used.
- Linear angle commands (,A can be used.)
- Geometric command

(2) When the sub part system control I function is valid, the independent command of address B is ignored.

9.3 Index Table Indexing

Function and purpose

Index table indexing can be carried out by setting the index axis.
The indexing command only requires specifying the indexing angle to the axis set for indexing. It is not necessary to command special M codes for table clamping and unclamping, thus simplifying the program.
There are the following two types for this function. Which type is valid and which axis is set as the indexing axis depend on the MTB specifications (parameters "#1282 ext18/bit3" and "#2076 index_x").
⬩Type A: When the unclamp command signal is turned OFF, the clamp operation is performed.
⬩Type B: When the clamp command signal is turned ON, the clamp operation is performed.

The PLC operation and each signal input/output depend on the MTB specifications.

Command format

G00 B90 ;

B	Index table indexing axis (designated with parameter "#2076 index_x")

Detailed description

Type A operations

(1) The movement command (either absolute or incremental) for the selected axis is executed with the program command.

(2) The unclamp command signal is now output prior to the axis movement.

(3) When the axes are unclamped, the unclamp completion signal is turned ON by the PLC.
(Turn the signal ON after performing required process such as servo ON or the unclamp process.)

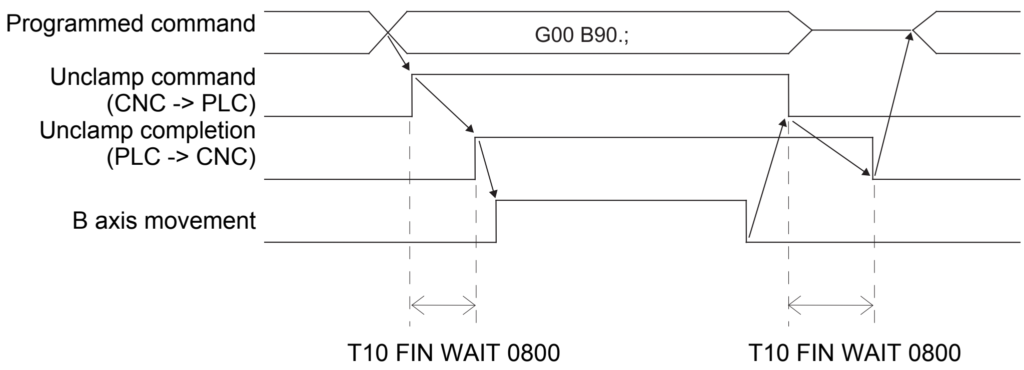
(4) After checking the unclamp completion signal, the designated axis starts moving.

(5) Upon completion of the movement, the unclamp command signal is turned OFF.

(6) Clamp the axes and turn the unclamp completion signal OFF with the PLC.
(Turn the signal OFF after performing required process such as in-position check, servo OFF or the clamp process.)

(7) After checking that the unclamp completion signal is OFF, processing of the next block is initiated.

[Operation time chart]



Type B operations

(1) The movement command (either absolute or incremental) for the selected axis is executed with the program command.

(2) The unclamp command signal is now output prior to the axis movement.

(3) When the axes are unclamped, the unclamp completion signal is turned ON by the PLC.
(Turn the signal ON after performing required process such as servo ON or the unclamp process.)

(4) After checking the unclamp completion signal, turn the unclamp command signal OFF and the designated axis starts moving.

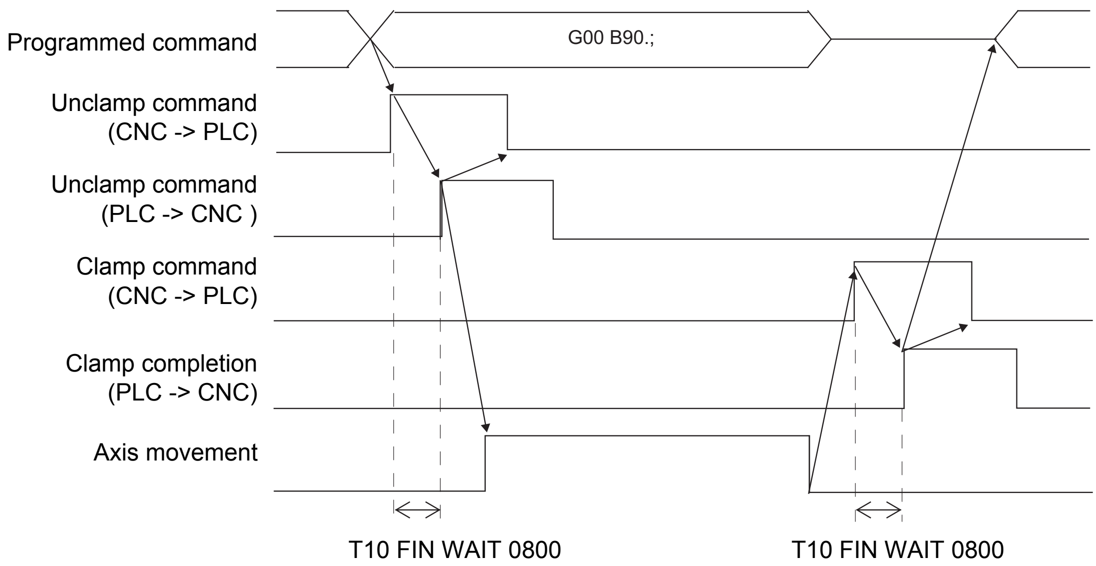
(5) Turn the unclamp completion signal OFF with the PLC.

(6) Upon completion of the movement, the clamp command signal is turned ON.

(7) Clamp the axes and turn the clamp completion signal ON with the PLC.
(Turn the signal OFF after performing required process such as in-position check, servo OFF or the clamp process.)

(8) After checking that the clamp completion signal is ON, turn the clamp command signal OFF and processing of the next block is initiated.

(9) Turn the clamp completion signal OFF with the PLC.



Cutting feed prohibit of index table indexing axes

When the cutting feed of index table indexing axes is prohibited, the cutting feed can be prohibited by issuing a program error (P20) if all of the following conditions are satisfied during automatic operation.

- The indexing axis movement command is issued. (*1)
- The modal of G code group 1 is other than "G00" or "G60".

(*1) If a cutting feed command without axis movement (such as "G01 B0;" during incremental command) is issued, the program error does not occur. Also, the unclamp command is not output.

The cutting feed prohibit function is valid for both type A and type B, and the parameter settings depend on the MTB specifications (Parameter "#2580 index_Gcmd").

Relationship with Other Functions

Index table indexing and other functions

Function	Details
Machine coordinate system selection (G53)	Possible.
Unidirectional positioning	(*1)
Servo ON/OFF signal control	Perform the required process on the PLC.

(*1) The unidirectional positioning function can be used in the machining center system only.

- If parameter "#8209 G60 SHIFT" is not in the indexing unit, a program error (P20) will occur.
- When an axis command that cannot be divided in the command unit is issued, a program error (P20) will occur.
- In a single block operation, the block stop is carried out for the indexing axis at the position specified by parameter #8209, and the clamp or unclamp operation is carried out.

Single block

The clamp and unclamp operations are not executed when the movement commands of the index table indexing axis are successively issued.
Note that the clamp and unclamp operations are executed even when the movement commands are continued during single block operation.
A combination of G codes that executes clamp or unclamp operation with continuous blocks is listed below.
(The unclamp operation is executed before the axis movement of previous block is started, and the clamp operation is executed after the axis movement of the next block.)

(1) Clamp and unclamp operations between continuous blocks

Command	Continuous block	Condition and result
Reference position check (G27)	G00 -> G27	(*1)
	G27 -> G00	(*2)
Start position return (G29)	G00 -> G29	(*1)
	G29 -> G00	(*1)
Tool change position return 1 to 6 Lathe system: G30.1 to G30.5 Machining center system: G30.1 to G30.6	G00 -> G30.1	(*1)
Normal line control cancel (G40.1) (Machining center system only)	G40.1 -> G00	(*1)
Basic machine coordinate system selection (G53)	G00 -> G53	(*3)
	G53 -> G00	
Unidirectional positioning (G60) (Machining center system only)	G00 -> G60	(*1)
	G60 -> G00	(*2)
	G60 -> G60	
Program stop (M00)	M00	(*1)
Optional stop (M01)	M01	(*1)

(*1) The clamp and unclamp operations are executed between blocks.

(*2) The clamp and unclamp operations are NOT executed between blocks.

(*3) The clamp and unclamp operations are executed during workpiece installation error compensation (G54.4) or during inclined surface machining command (G68.2).

(2) Clamp and unclamp operations between continuous blocks (Reference position return)
The operation during reference position return depends on the ignoring of intermediate points during return, and it depends on the MTB specifications (Parameter "#1091 Mpoint").

Command	Continuous block	Condition and result	
		#1091 = 1	#1091 = 0
1st reference position return (G28)	G00 -> G28	(*1)	(*2)
	G28 -> G00	(*3)	(*4)
2nd to 4th reference position return (G30)	G00 -> G30	(*1)	(*2)
	G30 -> G00	(*3)	(*4)

(*1) Performs the clamp operation at the end of G00 movement, and performs the unclamp operation before reference position return.

(*2) The clamp/unclamp operation will not be performed until the reference position return is completed.

(*3) Performs the clamp operation after the reference position return, and performs the unclamp operation before G00 movement.

(*4) The clamp/unclamp operation will not be performed when movement to the intermediate point is completed. The clamp operation will be performed after the reference position return, and the unclamp operation will be performed before G00 movement.

Macro interruption

Clamp/unclamp operations during macro interrupt are as follows.

Parameter		Settings
#1112 S_TRG	0	Edge trigger mode
#1113 INT_2	0	Immediately start the interrupt program without waiting for the completion of currently executing block.
#8101 MACRO SINGLE	1	

(1) When the macro interrupt program, executed during indexing axis movement, contains a movement command
The commands in the interrupted block are lost, and the interrupt program is executed. After completion of interrupt program, when executing from the block next to the interrupt block, the clamp/unclamp operation is executed even if the interrupt program and main program specify the continuous movement.
(Example)

[Main program]

```
O620(MINT MAIN)
N100 G90G94;
N110 G28B0.X0.;
N120 M96P621;
N130 G01X10.B10.F150.;
N132 G01B15.;
N140 G04X3.;
N150 M97;
M02;
```

[Interrupt program]

```
O621(MINT SUB)
N100 G01B1.F500.;
M99;
```

(a) Performs unclamp operation at the beginning of main program N130 block.

(b) Executes macro interrupt during main program N130 execution.

(c) Performs clamp operation after end of interrupt program O621 N100 block B1.

(d) Performs unclamp operation at the beginning of main program N132 block, and performs clamp operation after axis movement.

(2) When the macro interrupt program, executed during indexing axis movement, does not contain a movement command

When executing the remaining blocks after completion of interrupt program, perform the unclamp operation at the restart of main program.
Also, perform the clamp/unclamp operation even when the next block continues.
(Example)

[Main program]

```
O622(MINT MAIN)
N100 G90G94;
N110 G28B0.X0.;
N120 M96P623;
N130 G01X10.B10.F150.;
N132 G01B15.;
N140 G04X3.;
N150 M97;
M02;
```

[Interrupt program]

```
O623(MINT SUB)
N100 #100=#100+1;
M99;
```

(a) Performs unclamp operation at the beginning of main program N130 block.

(b) Executes macro interrupt during main program N130 execution.

(c) Interrupt program O623 execution is completed.

(d) Performs unclamp operation at the restart of main program N130 block, and performs clamp operation after completion of axis movement.

(e) Performs unclamp operation at the beginning of main program N132 block, and performs clamp operation after completion of axis movement.

Precautions

(1) Several axes can be set as index table indexing axes.

(2) The movement speed of index table indexing axes follows the feedrate of the modal (G00/G01) at that time.

(3) The unclamp command for the indexing axes is also issued when the index table indexing axes are commanded in the same block as other axes. Thus, the movement of other axes commanded in the same block is not carried out until the unclamp operation completes.
Note that the movement of other axes commanded in the same block is carried out during a non-interpolation commands.

(4) Index table indexing axes are used as normal rotation axes, but this function performs an unclamp operation even for linear axes.

(5) If some error that makes unclamp command OFF occurs during indexing axis movement in automatic operation, the unclamp state will remain, and the indexing axis will execute a deceleration stop.
Other axes commanded in the same block will also execute a deceleration stop, except during non-interpolation commands.

(6) If the axis movement is interrupted by an interlock, etc., during indexing axis movement, the unclamp state will remain.

(7) The clamp and unclamp operations are not executed when the movement commands of the index table indexing axis are successively issued.
Note that the clamp and unclamp operations are executed even when the movement commands are continued during single block operation. Refer to "Single block" of the "Relationship with other functions".

(8) Make sure that the command position is at a position where clamping is possible.

(9) Set the unidirectional positioning (G60) parameter "#8209 G60 SHIFT" in indexing increment. A program error (P20) will occur if it is not set by in indexing increment.
In a single block operation, the block stop is carried out at the "G60 SHIFT" position, and the clamp or unclamp operation is carried out.

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.

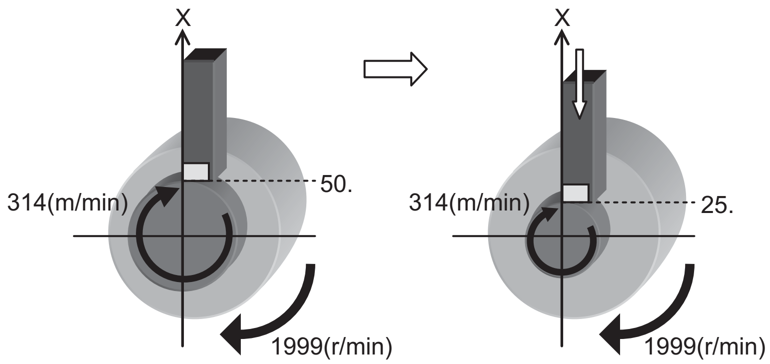
10.2 Constant Surface Speed Control ; G96,G97

Function and purpose

This function adjusts the spindle rotation speed (constant surface speed control) in accordance with the movement of the tool nose point so that the cutting point always remains at the constant speed (constant surface speed).
Using this function for processes such as a cutting-off process is effective in terms of machining time, tool life, etc.
Note that when the tool nose point is moving to the workpiece zero point, the rotation may be at the maximum rotation speed defined in the machine specifications; this is dangerous. Be sure to specify the maximum clamp rotation speed with the spindle clamp speed setting command (G92/G50).

Constant surface speed control at constant surface speed command G96 S314 m/min

Workpiece diameter: 50 mm (Radius value)   Workpiece diameter: 25mm (Radius value)



To keep the surface speed constant, this function obtains and automatically adjusts the spindle rotation speed in accordance with the movement of the tool nose point.
In the example above, to keep the surface speed (314 (m/min)) constant, the rotation speed is changed from 999 (r/min) to 1999 (r/min) with changes of the workpiece radius (50mm → 25mm).

Spindle rotation speed (r/min)	=	Surface speed (m/min)	/	Workpiece surface (m/r)
		G96 command value		Automatically calculated from the workpiece zero point and tool nose position

Command format

Constant surface speed ON

G96 S__ P__ ;	
S	Surface speed (-99999999 to 99999999 (m/min)), -99999999 to 99999999 (feet/min))
P	Constant surface speed control axis 0 to n (n: Number of axes that can be controlled in the part system with G96 commanded)

Note

(1) The S command is handled as the absolute value (the sign is ignored).

(2) If the value of the S command exceeds the allowable range, a program error will occur (P35).

(3) If the value of the P command exceeds the allowable range, a program error will occur (P133).

Constant surface speed cancel

G97 S__ ;	
S	Spindle rotation speed (-99999999 to 99999999 (r/min))

Note

(1) The S command is handled as the absolute value (the sign is ignored).

Detailed description

(1) When the P0 or P command is not specified, the operation depends on the MTB specifications (parameter "#1181 G96_ax").
However, if this parameter is set to "0", the first axis is used as the surface speed axis regardless of whether address P is specified or not.
0: Fixed at 1st axis (P command invalid)
1: 1st axis
2: 2nd axis
:

(2) To change the constant surface speed control axis in the constant surface speed control mode, specify the command in the G96 P_ format. (However, when the parameter above is set to "0", no change can be made.)
If the S command is issued simultaneously, the surface speed can also be changed.

<Example 1>

Machining program	Control axis sequence in constant surface speed command part system			Details of Operation
	1st axis	2nd axis	3rd axis	
: G96 S200 P1; : :	X1 ↓ ↓ ↓	Z1 ↓ ↓ ↓	C1 ↓ ↓ ↓	The X1 axis is used as the constant surface speed control axis. (Controls the spindle rotation so that the surface speed is set to 200 (m/min) for the X1 axis.)
G96 P2; :	↓ ↓	↓ ↓	↓ ↓	The Z1 axis is used as the constant surface speed control axis.

<Example 2>

Machining program	Control axis sequence in constant surface speed command part system			Details of Operation
	1st axis	2nd axis	3rd axis	
: G96 S200 P1; : : :	Z1 ↓ ↓ ↓ ↓	C1 ↓ ↓ ↓ ↓	- - - - -	The Z1 axis is used as the constant surface speed control axis. (Controls the spindle rotation so that the surface speed is set to 200 (m/min) for the Z1 axis.)

(3) The spindle to be controlled is determined in the MTB specifications (parameter "#1300 ext36/bit0").
For multiple-spindle control II (*1), the spindle is determined by the spindle selection signal from the PLC.

(*1) Multiple-spindle control with the PLC signal used. Whether the specification is provided and the details depend on models and MTB specifications.

(4) Specify the spindle surface speed with the S command when constant surface speed control ON is commanded.
In constant surface speed control mode, the surface speed can only be changed with the S command.

(5) The spindle clamp speed setting (G92 S__ Q__) is to be commanded 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.
Whether the spindle clamp speed setting is made valid only in the constant surface speed control mode or also made valid for normal spindle rotation commands depends on the MTB specifications (parameter "#1227 aux11/bit5").
Once the maximum clamp rotation speed and the minimum clamp rotation speed are set using the spindle clamp speed setting (G92 S__ Q __), the maximum speed clamp will not be canceled even if the command "G92 S0" is issued.
Whether the commanded spindle clamp speed setting is kept when NC is reset during constant surface speed control depends on the MTB specifications (parameter "#1210 RstGmd/bit19").

(6) Whether the surface speed is always calculated or at the end of a block when the rapid traverse command is issued depends on the MTB specifications (parameter "#1087 G96_G0").

(7) The constant surface speed cancel command (G97) cancels constant surface speed control in the part system that has executed the constant surface speed ON command (G96).
The constant surface speed control cannot be canceled from another part system.
The spindle rotation speed is maintained at the speed specified when the constant surface speed cancel command (G97) has been executed.

(8) If NC is reset during constant surface speed control, the spindle rotation speed is changed to "0" (r/min) after reset.

Relationship with Other Functions

Checking the maximum clamp rotation speed

When the constant surface speed control is commanded, check whether the spindle speed clamp is valid.
If the constant surface speed control axis is near the zero point, it causes the spindle to rotate at the maximum rotation speed.
Check the spindle speed clamp command to prevent the spindle from rotating at high speed.

(1) In multiple-spindle control II, if the speed clamp command is not valid for the selected spindle, it causes an operation error (M01 1043). When such an error occurs, reset to finish the program, and issue the spindle speed clamp command after selecting a spindle.
When the operation error above occurs, execute the commands in the same block.

(a)	G96 S100 M03 command:	When the spindle forward rotation signal is input from the user PLC by the M03 command, the spindle runs forward. (The spindle speed is set to the previously commanded rotation speed.)
(b)	G96 S100 X30. command:	If an error occurs, axis movement is performed until the program is reset.

(2) When spindle speed clamp command check is valid, the spindle speed clamp command value is set to "0" if the G92/G50 S0 command is issued.

(3) In multiple-spindle control II, spindle speed clamp check is conducted for the spindle selected in the G96S command. Specify the spindle speed clamp command for all the currently selected spindles.

(4) Whether to conduct spindle speed clamp command check depends on the MTB specifications (parameters "#1146 Sclamp" and "#1284 ext20/bit0".)
If parameter "#1146 Sclamp" is set to "0", the spindle speed clamp command cannot be executed when constant surface speed control is turned off; therefore, the spindle speed clamp command cannot be issued before constant surface speed control.

Parameter "#1284 ext20/bit0" has the following setting:

0: Checks the spindle speed clamp.

1: Does not check the spindle speed clamp.

(5) The spindle speed clamp may be performed only in the constant surface speed mode depending on the MTB specifications (parameter "#1227 aux11/bit5"). If the program is then reset, the clamp may be rendered ineffective.
For information on whether the setting is configured to keep the clamp status, refer to the MTB specifications. (Parameter "#1210 RstGmd/bit10, bit19")

BIT10: Group 17, constant surface speed control command modal

BIT19: Spindle rotation clamp speed

(6) When operating the system in the initial constant surface speed mode or with the constant surface speed modal by holding the constant surface speed control command modal, the constant surface speed control mode is set by the S command (surface speed). When spindle speed clamp command check is valid, issue the spindle speed clamp command before the S command.

Arbitrary axis exchange control

(1) If constant surface speed control axes are rearranged by the arbitrary axis exchange command, the spindle rotation speed is maintained at the value specified before rearrangement.

(2) If a new surface speed is specified by the S command while the spindle rotation speed is maintained, it becomes valid when the rearranged constant surface speed axes are returned to the original status.

(3) If the constant surface speed command is re-executed when constant surface speed axes are rearranged and the spindle rotation speed is maintained at the constant rotation speed, the kept spindle rotation speed is canceled, and the reissued constant surface speed control command is executed.

(4) If constant surface speed axes are returned to the original status by rearrangement while constant surface speed control is temporarily canceled, the spindle rotation speed will be maintained. After this, the surface speed becomes constant when it is specified with the S command.

(5) If the surface speed is specified by the S command with the rearrangement of the constant surface speed axes while the constant surface speed control is in the temporary cancel state, the spindle rotation speed kept at temporary cancellation is applied, and the surface speed becomes constant when the constant surface speed axes are returned to the original arrangement.

Other functions

Function name	Operation
Spindle Clamp Speed Setting (G92/G50)	The spindle clamp speed setting is valid in the constant surface speed control mode. Whether the commanded spindle clamp speed setting is kept when NC is reset during constant surface speed control depends on the MTB specifications. (parameter "#1210 RstGmd BIT19")
Cylindrical Interpolation (G07.1)	The constant surface speed control cannot be commanded during the cylindrical interpolation mode. Program error (P481) will occur. The cylindrical interpolation cannot be commanded during the constant surface speed control mode. Program error (P485) will occur.
Thread Cutting (Designation of lead or number of ridges) (G32)	When the constant surface speed command is issued in the same part system during execution of the thread cutting or thread cutting cycle command or when the thread cutting or thread cutting cycle command is issued in the same part system in the constant surface speed control mode, the spindle rotation speed for constant surface speed control remains unchanged. (The constant surface speed control is not performed.) This function keeps the spindle rotation speed specified at execution of the thread cutting or thread cutting cycle command. When the thread cutting or thread cutting cycle command is terminated, the spindle rotation speed is changed to the value obtained from the position of the constant surface speed control axis and the surface speed. The constant surface speed command cannot be issued from other part systems to the spindle for which the thread cutting command is currently executed. Also, the thread cutting command cannot be issued from other part systems to the spindle in the constant surface speed control mode. An operation error (M01 1113) will occur.

Function name	Operation
Tapping cycle (G84/G88) Synchronous tapping cycle (G84/G88)	If the constant surface speed command is issued in the same part system during execution of the tapping cycle or synchronous tapping cycle command or the tapping cycle command is issued in the same part system in the constant surface speed control mode, the spindle rotation speed for constant surface speed control remains unchanged. (The constant surface speed control is not performed.) This function keeps the spindle rotation speed specified at execution of the tapping cycle command. When the tapping cycle or synchronous tapping cycle command is terminated, the spindle rotation speed is changed to the value obtained from the position of the constant surface speed control axis and the surface speed. The constant surface speed command cannot be issued from other part systems to the spindle for which the tapping cycle or synchronous tapping cycle command is currently executed. Also, the tapping cycle or synchronous tapping cycle command cannot be issued from other part systems to the spindle in the constant surface speed control mode. An operation error (M01 1113) will occur. The synchronous tapping cycle command cannot be executed in the constant surface speed control mode. Program error (P182) will occur. Also, the constant surface speed command cannot be executed during execution of the synchronous taping cycle command. Program error (P186) will occur.
Spindle-Mode Servo Motor Control	The system also runs when a spindle-mode servo is specified for constant surface speed control
External Spindle Deceleration	If the external spindle deceleration signal is set to OFF for the spindle under constant surface speed control, the spindle is clamped at the external spindle deceleration speed. The S analog maximum/minimum over signal (SOVE) is set on.
High-Speed Simple Program Check	The surface speed is calculated, but the actual rotation speed of the spindle remains set to the value specified before the part system synchronization machine lock high-speed operation is selected.
NC reset (Reset 1/2, reset & rewind)	When NC is reset during constant surface speed control, the spindle rotation speed is set to "0" (r/min).

Precautions

(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 break tools or the machine or injure the operators. Therefore, 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

(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 200 m/min.
G92 S4000 Q200 ;	The spindle rotation speed is clamped up to 4000 r/min and down to 200 r/min.
M3 ;	The rotation command to the spindle

(Example 2) When the parameter "#1146 Sclamp" is set to "1"

G92 S4000 Q200 ;	The spindle rotation speed is clamped up to 4000 r/min and down to 200 r/min.
G96 S200 ;	The spindle rotation speed is controlled so that the surface speed is 200 m/min.
M3 ;	The rotation command to the spindle

<Note>
- For safety, issue the rotation command to the spindle after G92.

⚠ WARNING

⚠ **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 break tools or the machine or injure the operators.**

(2) When the G96 command is issued, do not omit the "S_" surface speed command. If omitted, the system will follow the previous "S_" command.
The S command ("S_" command) for the spindle in the constant surface speed control mode specifies the surface speed.

(3) If the spindle speed clamp is not commanded when the constant surface speed control axis is near the zero point, it causes the spindle to rotate at the maximum rotation speed. We recommend that you command the spindle speed clamp before the constant surface speed command.
In this case, the parameter "#1146 Sclamp" must be made valid, but this function depends on the MTB specifications.

(4) If an axis number not registered in the command part system is commanded when the constant surface speed command is specified, it causes a program error (P133).

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

Spindle clamp speed setting

G92 S__ Q__ ;

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) Whether to carry out rotation speed clamp only in the constant surface speed mode or even when the constant surface speed is canceled depends on the MTB specifications (parameters "#1146 Sclamp" and "#1227 aux11/bit5").

<Note>

- G92S command and rotation speed clamp operation

<table>
<tr><th colspan="2" rowspan="2"></th><th colspan="2">Sclamp=0</th><th colspan="2">Sclamp=1</th></tr>
<tr><th>aux11/bit5=0</th><th>aux11/bit5=1</th><th>aux11/bit5=0</th><th>aux11/bit5=1</th></tr>
<tr><td rowspan="2">Command</td><td>In G96</td><td colspan="2">Rotation speed clamp command</td><td colspan="2">Rotation speed clamp command</td></tr>
<tr><td>In G97</td><td colspan="2">Spindle rotation speed command</td><td colspan="2">Rotation speed clamp command</td></tr>
<tr><td rowspan="2">Operation</td><td>In G96</td><td colspan="2">Rotation speed clamp execution</td><td colspan="2">Rotation speed clamp execution</td></tr>
<tr><td>In G97</td><td colspan="2">No rotation speed clamp</td><td>Rotation speed clamp command</td><td>No rotation speed clamp</td></tr>
</table>

- 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 rotation 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.
It is set to "0" during power ON.

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. During this time, the Q__ value is still valid and S0 < Q__ is established. The Q__ value is treated as the maximum speed clamp, and S0 is treated as the minimum speed clamp.

(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

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 Position Control (Spindle/C Axis Control)

Function and purpose

This function controls a spindle as the rotary axis. After switching the spindle to the rotary axis, the positioning and the interpolation between the spindle and other servo axes can be operated in the same way as the servo axis by executing the position command (the movement command). Using this function, the servo axis for controlling the spindle stock as the rotary axis or the machinery for switching the spindle and servo axis (such as a gear switching machinery) had been necessary for controlling a spindle stock readily as the rotary axis, but they are not necessary with this function.
For information on how to validate or invalidate this function, each setting to use this function, and the mechanism of your machine, refer to the specifications or the instruction manual issued by the MTB.
There are two methods to switch the spindle and rotary axis: PLC signal method and program command method. The available method depends on the MTB specifications (parameter "#3129 cax_spec/bit0"). For details, refer to the specifications issued by the MTB.
This section describes the program command method.
In this manual, the state of controlling an axis as a spindle is referred to as "spindle mode", and the state of controlling an axis as a rotary axis is referred to as “C axis mode”.
The PLC signal processing and operation depends on the MTB specifications. Refer to the instruction manual issued by the MTB for details.

Coordinate zero point and zero point adjustment in C axis mode

For the encoder-based spindle position control (PLG and external encoder), set the Z phase position of the encoder as the first reference point of the C axis. This first reference point is used as the coordinate zero point; however, the spindle zero point position can be adjusted with the spindle/C axis reference position return shift amount parameter, which is determined in the MTB specifications. This parameter is determined in the MTB specifications (parameter "#3113 cax_sft").

Program command method

In the machining program, the program switches to the C axis mode with G00 command, and to the spindle mode with S command. The C axis servo OFF signal (*SVFn) must be always kept ON while the program command method is selected. This depends on the MTB specifications.
When the servo OFF signal is set to OFF, operations are performed as follows.

- The mode cannot be switched from the spindle mode to the C axis mode or from the C axis mode to the spindle mode.
- In the spindle mode, the axis does not run as a spindle even if the forward run command (SRN) or reverse run command (SRI) is executed.
- In the C axis mode, an operation error (M01 0005) occurs if the movement command is executed. In the servo OFF mode, operations follow the setting of the parameter "#1064 svof" (error correction) MTB specifications).
- Which mode (the spindle mode or the C axis mode) is set at power-ON or reset depends on the MTB specifications (parameter "#3129 cax_spec/bit2, bit3").
  If the power is turned on in the C axis mode setting, the mode shifts to the C axis mode after the Z phase detection and reference position return operations have been performed.
  For Z phase detection, the spindle rotates in the C axis zero point return direction at the C axis zero point return speed.

Command format

Switching from spindle mode to C axis mode (C axis)

G00 C__ ;

C	Target C axis in C axis mode

- Command "G00 C_ " in the NC program during the spindle mode. The axis is positioned directly to the specified position.
  The non-interpolation positioning for each axis is performed by specifying "G00 X__ Z__ C__ " regardless of the G00 interp OFF parameter ("#1086 G0Intp" in the MTB specifications) setting, and C axis is switched to the C axis mode.
- Only the G00 command is valid to switch the mode. If the mode is commanded with another G code, it causes a program error (P430).
- Designate the axis for spindle position control with the absolute value address or absolute value command (G90). If the axis is designated with the incremental value address or incremental value command (G91), it causes a program error (P32).
- The reference position return type (*1) is set at switching, and the direction to return from the rotation mode to the zero point follows the rotation direction (*2). The direction for returning from the stop mode to the reference position and the interpolation mode depend on the MTB specifications (parameters "#3106 zrn_typ/bit9,bitA", "#3106 zrn_typ/bitD,bitE" and "#1256 set28/bit1").

  (*1) Type to necessarily return to the reference position when switching from the spindle mode to the C axis mode. This depends on the MTB specifications (parameter "#3106 zrn_typ/bit8").

  (*2) Depends on the MTB specifications (parameter "#3106 zrn_typ/bitB").
- If the Z phase is not detected and if switching is commanded, the spindle is rotated in the zero point return direction (*4) at the spindle zero return speed (*3). Then, the zero point return operation is executed after the Z phase detection.

  (*3) Depends on the MTB specifications (parameter "#3112 cax_spd").

  (*4) Depends on the MTB specifications (parameter "#3106 zrn_typ/bitA-9").

[C axis mode switching conditions]

When switching is commanded, the following condition must be satisfied.

"The C axis servo OFF signal (*SVFn) is turned ON."

Switching C axis mode to spindle mode

- The switching is performed with the spindle forward run signal (SRN) ON or the spindle reverse run signal (SRI) ON and the S command.
- The switching is performed with the startup of the spindle forward run signal (SRN) or the spindle reverse run signal (SRI).

[Spindle mode switching condition]

When switching is commanded, the following condition must be satisfied.

"The C axis servo OFF signal (*SVFn) is turned ON and the C axis selection signal (CMD) is turned OFF."

Detailed description

Mode switching

(1) Example in which the mode is switched to the spindle mode with the forward run command and the rotation command (S command)
M03 command -> Forward run command (SRN) ON and reverse run command (SRI) OFF
M04 command -> Reverse run command (SRI) ON and forward run command (SRN) OFF

Program example	Mode	Description
M03 S1000 : :	Spindle mode	The spindle rotates at forward run speed 1000 (r/min).
G00 C90.	C axis mode	The axis is positioned at 90 degrees directly based on the rotation mode. After positioning, the mode is switched from the spindle mode to the C axis mode.
G01 X10. C20. F100 :		In the C axis mode, the spindle can be commanded as the rotary axis. In the C axis mode, interpolation with another servo axis is possible.
M03 S1500 : :	Spindle mode	The mode is switched from the C axis mode to the spindle mode with the forward run command and rotation command (S command). After being switched to the spindle mode, the spindle rotates at forward run speed 1500 (r/min).
G00 X20.C270.	C axis mode	The axis is positioned at 270 degrees directly based on the rotation mode, and stops at the position. Simultaneously, the X axis is positioned at 20mm with interpolation. After positioning, the mode is switched from the spindle mode to the C axis mode.

(2) Example in which the mode is switched to the spindle mode by a change from the forward run command to the reverse run command
M03 command -> Forward run command (SRN) ON and reverse run command (SRI) OFF
M04 command -> Reverse run command (SRI) ON and forward run command (SRN) OFF

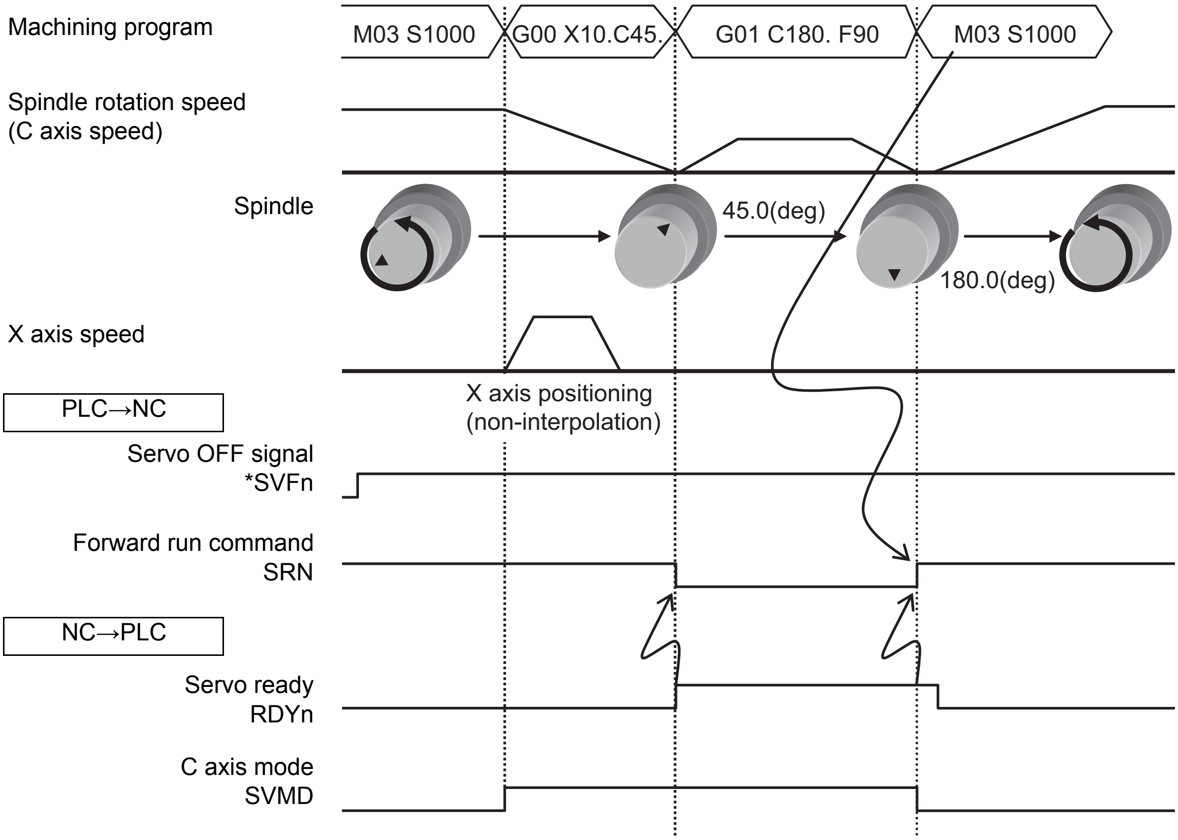
Program example	Mode	Description
M03 S1000 : :	Spindle mode	The spindle rotates at forward run speed 1000 (r/min).
G00 C90.	C axis mode	The axis is positioned at 90 degrees directly based on the rotation mode. After positioning, the mode is switched from the spindle mode to the C axis mode.
G01 X10. C20. F100 :		In the C axis mode, the spindle can be commanded as the rotary axis. In the C axis mode, interpolation with another servo axis is possible.
M4 : :	Spindle mode	The mode is switched from the C axis mode to the spindle mode with the reverse run command. After being switched to the spindle mode, the spindle rotates at reverse run speed 1000 (r/min).
G00 X20.C270.	C axis mode	The axis is positioned at 270 degrees directly based on the rotation mode, and stops at the position. Simultaneously, the X axis is positioned at 20 mm with interpolation. After positioning, the mode is switched from the spindle mode to the C axis mode.

(3) Example in which the mode is not switched from the C axis mode to the spindle mode
M03 command -> Forward run command (SRN) ON and reverse run command (SRI) OFF

Program example	Mode	Description
M03 S1000 : :	Spindle mode	The spindle rotates at forward run speed 1000 (r/min).
G00 C90.	C axis mode	The axis is positioned at 90 degrees directly based on the rotation mode. After positioning, the mode is switched from the spindle mode to the C axis mode.
G01 X10. C20. F100 :		In the C axis mode, the spindle can be commanded as the rotary axis. In the C axis mode, interpolation with another servo axis is possible.
M3 : :	C axis mode	The rotation command (S command) is omitted between the forward run commands, and the rising edge (change) of the forward run command is not detected; therefore, the mode is not switched to the spindle mode. The forward run command must be changed from OFF to ON with the rotation command (S command) or M3 command.

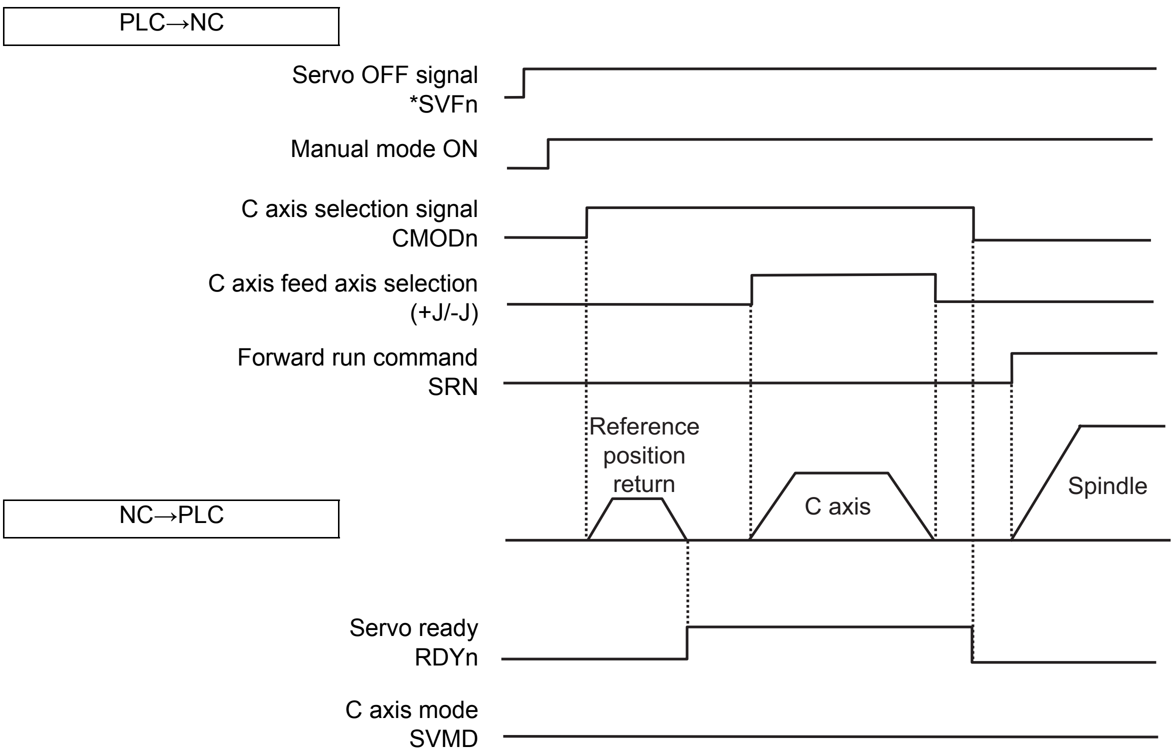
Switching operation

When the program command method is selected, switching operation is performed with the reference position return type.



Manual operation with the program command method selected

When moving the spindle/C axis as the C axis in the manual operation mode, change the C axis selection signal (CMOD) from OFF to ON to switch to the C axis mode. When switching to the spindle mode, change the C axis selection signal from ON to OFF. The switching operation is performed with the reference position return type. In the C axis mode, the axis can be moved by selecting the manual mode (jog mode, handle mode, incremental feed mode, manual arbitrary-feed mode, or reference position return mode).



If the C axis selection signal (CMOD) is changed while either the C axis mode or spindle mode is selected in the program command method, the mode is set as follows. The mode is not switched to the C axis mode or spindle mode in the program command method during automatic running when the C axis selection signal is turned ON. Switching follows the state of the C axis selection signal (CMOD).

C axis selection signal (CMOD)	**During automatic operation**		**During reset**	
	C axis mode by "G00 C_ command"	**Spindle mode by "S command"**	**C axis mode**	**Spindle mode**
OFF to ON	C axis mode	C axis mode	C axis mode	C axis mode
ON to OFF	C axis mode	Spindle mode	Spindle mode	Spindle mode
Remarks			Whether the mode is switched to the C axis mode during reset depends on the MTB specifications. (#3129 cax_spec/bit2) (#3129 cax_spec/bit3)	

Relationship with Other Functions

Spindle forward-run start (SRN) and spindle reverse-run start (SRI)

The mode is switched to the C axis mode regardless of the state of the spindle forward-run start (SRN) or spindle reverse-run start (SRI) signal. In the C axis mode, spindle forward-run start and spindle reverse-run start are invalid.
When [PLC signal method] is selected, the spindle rotates by carrying out the spindle forward-run start or spindle reverse-run start again (OFF to ON operation) after the C axis mode has been canceled.
When [Program command method] is selected, the spindle rotates by carrying out the spindle forward-run start or spindle reverse-run start again (OFF to ON operation) in the C axis mode or by issuing the S command with the spindle forward-run start or spindle reverse-run start set ON.

Spindle orientation signal (ORC)

In the C axis mode, the spindle orientation command (ORC) is invalid. Also, the spindle position control command (spindle/C axis control) is invalid if issued during spindle orientation.

Spindle gear switching

Gear switching cannot be performed in the C axis mode. After the mode has been changed from the C axis mode to the spindle mode, gear switching is performed. Also, the mode cannot be switched to the C axis mode during gear switching. After gear switching has been completed, the mode is switched to the C axis mode.

Coil switching

Coil switching is invalid in the C axis mode. Conduct coil switching before switching to the C axis mode. If switching to the C axis mode is commanded during coil switching, switching to the C axis mode is executed after coil switching has been completed.

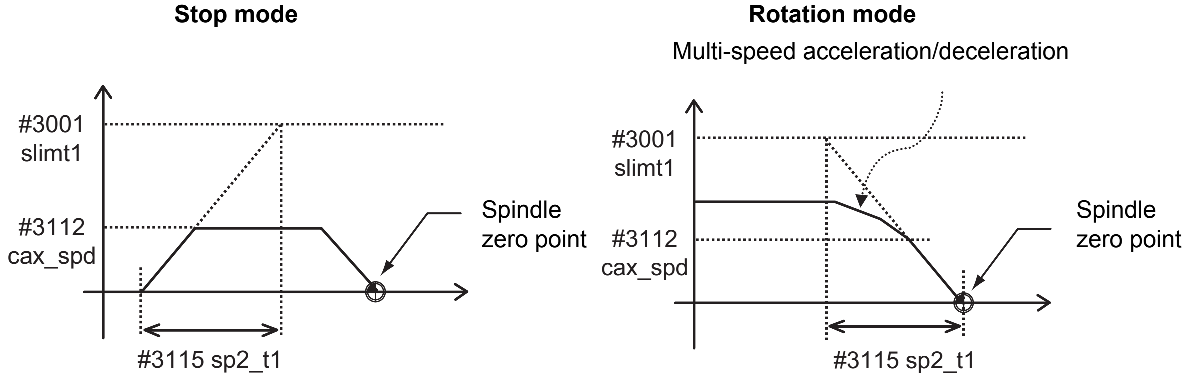
Spindle override

The spindle override is valid for the reference position return operation at switching to the C axis mode. In the C axis mode, the spindle override is invalid. The cutting feed override or rapid traverse override is valid in the C axis mode.

Spindle-Mode Servo Motor Control

The spindle position control is valid, excluding the following differences.

(1) Speed pattern of the reference position return operation
The reference position return from the stop mode is performed by the same operation as for the normal spindle. However, when the spindle returns from the rotation mode to the zero point, the rotation speed decelerates up to the C axis zero point return speed with the multi-step acceleration/deceleration pattern. After the C axis zero point return speed has been reached, the spindle inclines to the zero point, and stops while decelerating at the constant speed The multi-step acceleration/deceleration pattern and C axis reference position return speed depend on the MTB specifications (parameters "#3054 sptc1" to "#3061 spdiv1" and "#3112 cax_spd").



Analog spindle

The spindle position control is invalid.

Absolute position detection

The absolute position detection is invalid in the C axis mode.

Constant surface speed control

When [Program command method] is selected, the mode is switched from the C axis mode to the spindle mode if the surface speed command S (m/min) and the spindle forward-run start (SRN) or spindle reverse-run start (SRI) signal is set ON.

Manual arbitrary reverse run

When the program command method is selected, "block switched from spindle mode to C axis mode (example: G00 C_) and "block switched from C axis mode to spindle mode (example: M03 S1000)" is handled as a reverse run prohibited block. The reverse run cannot be carried out back through blocks with the mode switched.

Precautions and restrictions

(1) If the movement is commanded while the servo OFF signal (*SVFn) is set to OFF, it causes an operation error (M01 0005). Reset NC to cancel the error, and set the servo OFF signal on to restart machining. If the spindle command is issued, the spindle does not rotate.

(2) If the servo OFF signal (*SVFn) is set to OFF during C axis movement, it causes an operation error (M01 0005). Reset NC to cancel the error.

(3) To switch from the spindle mode to the C axis mode, issue the G00 command. If a command other than the G00 command is issued, it causes a program error (P430).

(4) The spindle position control axis must be commanded with the absolute value address or absolute value command (G90). If the incremental value address or incremental value command (G91) is used, it causes a program error (P32).

11

Tool Functions (T command)

11.1 Tool Functions (T8-digit BCD)

Function and purpose

The tool functions are also known as T functions and they assign the tool numbers. This control unit specifies a tool number with an 8-digit (0 to 99999999) number following the address T, and up to four sets can be commanded into one block. However, the number of sets that can be commanded within the same block depends on the MTB specifications (parameter "#12009 Tfig").
One of the following output signals is issued depending on the parameter setting (depends on the machine specifications).
- 8-digit BCD code and start signal
- Signed 32-bit binary data and start signal
- Unsigned 32-bit binary data and start signal

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

(1) The T function is executed after completion of the movement.

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

Processing and completion sequences are required for all T commands.

12

Tool Compensation Functions

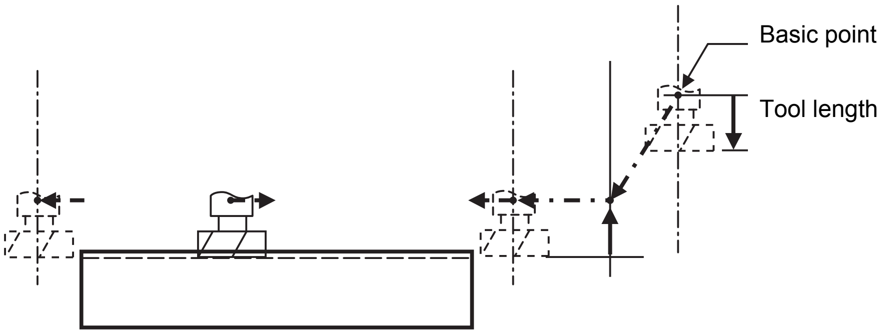
12.1 Tool Compensation

12.1.1 Tool Compensation

Function and purpose

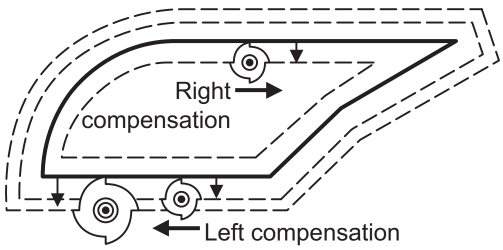
The basic tool compensation function includes the tool length compensation and tool radius compensation. Each compensation amount is designated with the tool compensation No. Each compensation amount is input from the setting and display unit or the program.

Tool length compensation



(Side view)

Tool radius compensation



(Plane view)

Tool compensation memory

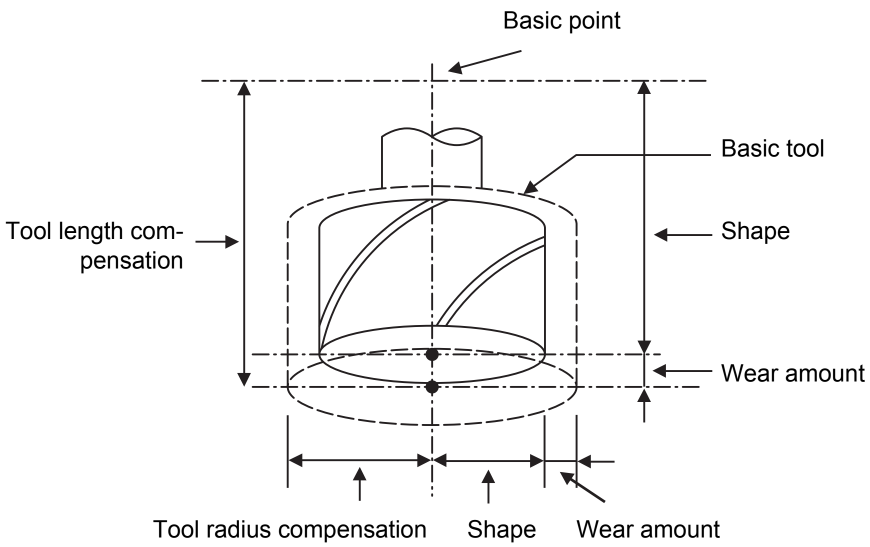
There are two types of tool compensation memories, types I and II, used to set and select the tool compensation amount. (The type used is determined by the MTB specifications.)
Each of types I and II can be changed to type III depending on the MTB settings (parameter "#1046 T-ofs disp type").
If the tool compensation memory is changed to type III, you can register the tool compensation amount of the base axes I, J, and K and the tool tip point, enabling tool compensation for a turning tool.
The details of type III are also displayed on the screen.
If the tool compensation memory is reset to the original type, the turning tool compensation items are not displayed. However, when it is changed to type III, the previously registered data is displayed.

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

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

Type of tool compensation memory	Classification of length compensation, radius compensation	Classification of shape compensation, wear compensation
Type I	No	No
Type II	Yes	Yes
Type III	(*1)	Yes

(*1) Distinguished between tool length compensation and tool nose radius compensation.



Type I

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

(D1) = a1 , (H1) = a1
(D2) = a2 , (H2) = a2
: :
(Dn) = an , (Hn) = an

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

If the tool compensation type is changed from type I to type III, the tool compensation amount of type I is handled as tool length Z of type III.

Type II

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

(H1) = b1 + c1 , (D1) = d1 + e1
(H2) = b2 + c2 , (D2) = d2 + e2
: :
(Hn) = bn + cn , (Dn) = dn + en

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

If the tool compensation type is changed from type II to type III, data registered for type II is handled as the following data.

Type II	Type III
Length dimension	Tool length Z
Length wear	Z wear
Radius dimension	Tool nose R
Radius wear	R wear

⚠ CAUTION

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

Tool compensation No. (H/D)

This address designates the tool compensation No.

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

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

(3) The compensation No. can be commanded once in each block. (If two or more Nos. are commanded, the latter one will be valid.)

(4) For 40 sets:
Designate with the H01 to H40 (D01 to D40) numbers.

(5) If a value larger than this is set, the program error (P170) will occur.

(6) The setting value ranges are as follows for each No.
The compensation amount for each compensation No. is preset with the setting and display unit.

Setting	Shape compensation amount		Wear compensation amount	
	Metric system	Inch system	Metric system	Inch system
#1003=B	± 999.999 (mm)	± 99.9999 (inch)	± 999.999 (mm)	± 99.9999 (inch)
#1003=C	± 999.9999 (mm)	± 99.99999 (inch)	± 999.9999 (mm)	± 99.99999 (inch)
#1003=D	± 999.99999 (mm)	± 99.999999 (inch)	± 999.99999 (mm)	± 99.999999 (inch)
#1003=E	± 999.999999 (mm)	± 99.9999999 (inch)	± 999.999999 (mm)	± 99.9999999 (inch)

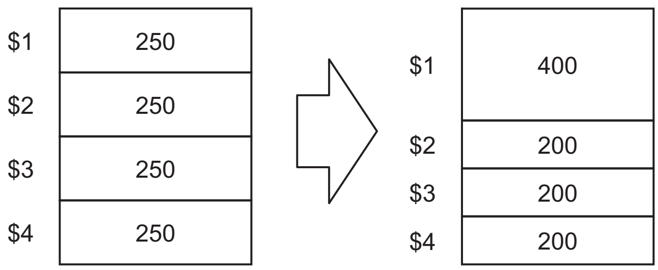
12.1.2 Number of Tool Offset Sets Allocation to Part Systems

Function and purpose

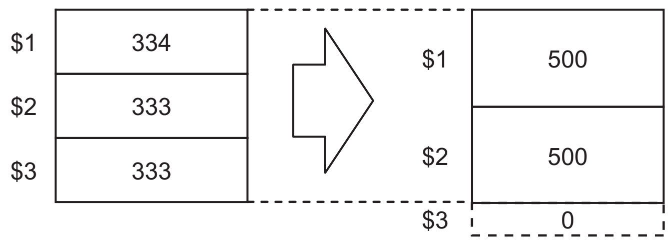
The number of tool offset sets can be set per part system.
This function is divided into the following methods and which one is used depends on the MTB specifications (parameters "#1438 Ofs-SysAssign", "#12054 Tol-Ofsnum").
Arbitrary allocation: Arbitrarily allocates to each part system.
Fixed allocation: Automatically and evenly allocates to each part system.
The arbitrary allocation enables the efficient allocation because when a certain part system needs only a small number of offset sets, the rest can be allocated to another part system. If an auxiliary-axis part system does not need the tool offset set at all, the number of tool offset sets can be set to "0" for the auxiliary-axis part system.
While this function is available if the specification allows allocation by tool compensation memory part system, this parameter depends on the MTB specification parameter "#1051 MemTol"). Subsequent description is an example in the case where the number of tool offset sets in the system is 999. Number of tool offset sets in system is the total number of tool offset sets of all part systems.

(1) Arbitrary allocation (with #1438=1)
The number of tool offset sets allocated to each part system depends on the MTB specifications (parameter "#12054 Tol-ofsnum").
The following example shows the number of tool offset sets allocated when the lathe system is a 4-part system.

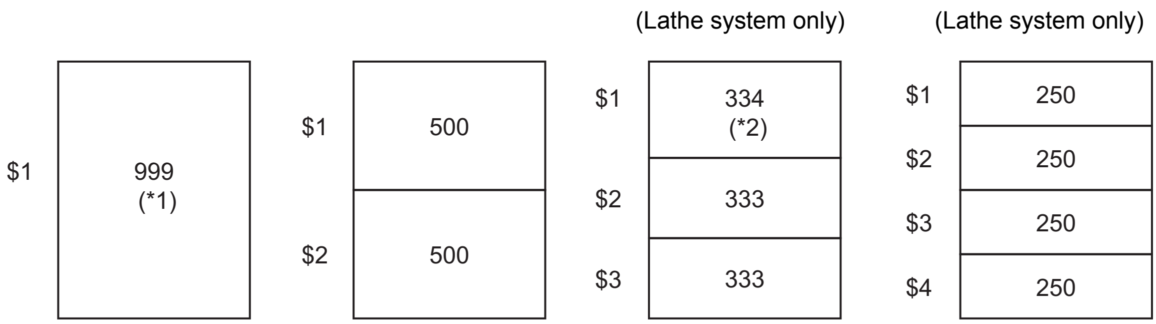
(a) When the number of tool offset sets is increased for the 1st part system of 4-part system



(b) When the number of offset sets is set to "0 sets" for the 3rd part system to use the 3rd part system as auxiliary-axis part system



(2) Fixed allocation (with #1438=0)



(*1) The maximum number of tool offset sets per part system is 999.

(*2) If there is any remainder, the remainder sets are allocated to the 1st part system.

Precautions

(1) The maximum number of tool offset sets for 1-part system is 999.

(2) For 1-part system, up to the number of tool offset sets in the system is available regardless of the parameter setting.

(3) When the value of the parameter "#12054 Tol-Ofsnum" (the number of tool offset sets by arbitrary allocation) is equal to or below the number of tool offset sets in the system, the remainder is not allocated to any part system even if the specification allows arbitrary allocation.

(4) When the tool compensation memory is provided commonly for the part systems ("#1051 MemTol"=1), the number of tool offset sets in the system are commonly used by all part systems regardless of the parameter setting. The setting of parameter #1051 depends on the MTB specifications, so check it in your machine specifications.

(5) Even if the specification allows arbitrary allocation, fixed allocation is applied if the parameter is "#12054 Tol-Ofsnum"= 0.

(6) When entering offset data, if the number of offset data exceeds that of current tool offset sets, the excess offset data cannot be entered.

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

Function and purpose

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

Command format

Tool length compensation start

G43 Zz Hh ; (+ direction)

G44 Zz Hh ; (- direction)

Tool length compensation cancel

G49 Zz ;

Detailed description

Tool length compensation movement amount

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

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

lh1; Compensation amount for compensation No. h1

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

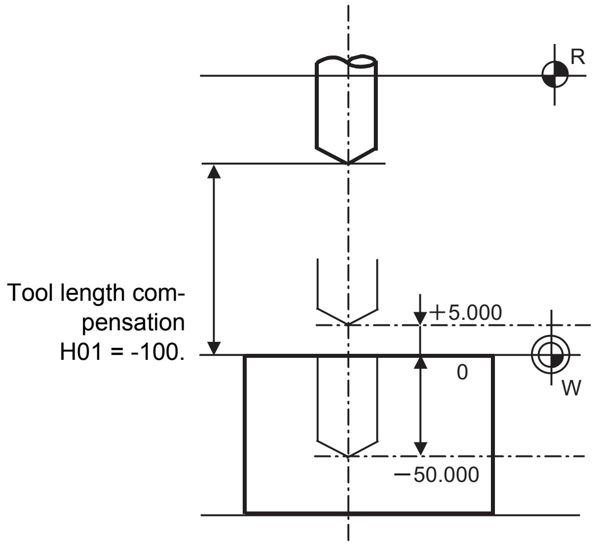
The G49 (tool length compensation cancel) mode is entered when the power is turned ON or when M02 has been executed.

(Example 1) For absolute value command H01=-100000

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

(Example 2) For incremental value command H01=-100000

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

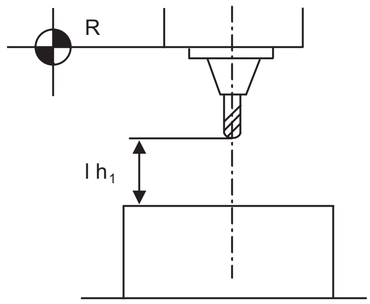


Compensation No.

(1) The compensation amount differs according to the compensation type. The following example shows a case in which "G43 Hh1;" is commanded.

Type I

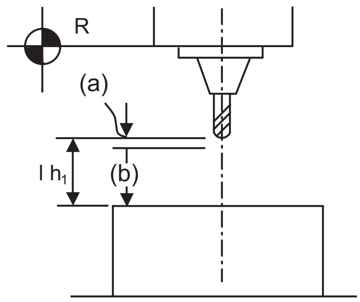
The compensation amount lh1 commanded with compensation No. h1 will be applied commonly regardless of the tool length compensation amount, tool radius compensation amount, shape compensation amount or wear compensation amount.



Type II

The compensation amount lh1 commanded with compensation No. h1 is as follows.

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



Type III

The compensation amount lh1 commanded with compensation No. h1 is as follows. (Refer to the figure of type II.)
lh1: Tool length compensation amount in Z axis direction (b) + Wear compensation amount in Z axis report (a)

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

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

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

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

```
G43 Zz1 Hh1 ;        Tool length compensation is executed with h1.
:
G45 Xx1 Yy1 Hh6 ;
:
G49 Zz2 ;            Tool length compensation is canceled.
:
G43 Zz2 ;            Tool length compensation is re-executed with h1.
:
```

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

```
G43 Zz1 Hh1 ;        The axis moves by "z1 + (lh1)".
  :
G43 Zz2 Hh2 ;        The axis moves by "z2 + (lh2-lh1)".
  :
```

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

Axis valid for tool length compensation

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

(2) When parameter "#1080 Dril_Z" is set to "0", the axis will depend on the axis address commanded in the same block as G43. The order of priority is shown below.
Zp > Yp > Xp
(Example 5)

G43 Xx1 Hh1;	+ compensation to X axis
:	
G49 Xx2 ;	
:	
G44 Yy1Hh2 ;	- compensation to Y axis
:	
G49 Yy2 ;	
:	
G43 αα1 Hh3;	+ compensation to additional axis
:	
G49 αα1 ;	
:	
G43 Xx3Yy3Zz3 ;	Compensation is applied on Z axis.
:	
G49;	

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

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

G43 Hh1 ;	Compensation and cancel to Z axis
:	
G49;	
:	

Movement during other commands in tool length compensation modal

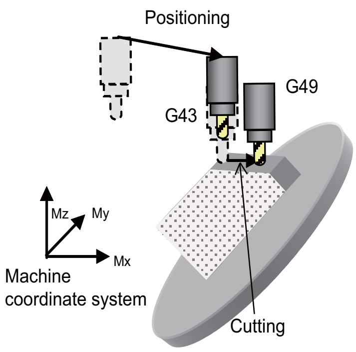
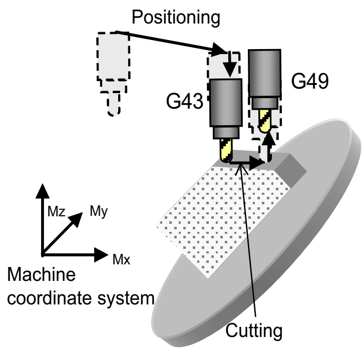
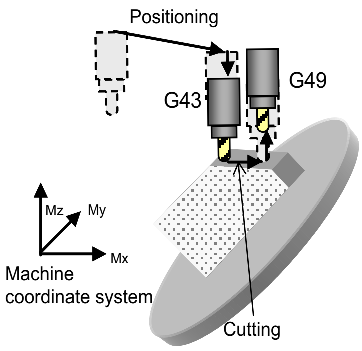
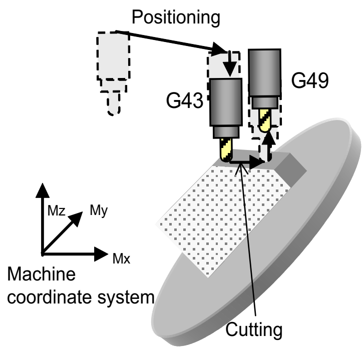
(1) If reference position return is executed with G28 and manual operation, the tool length compensation will be canceled when the reference position return is completed.
(Example 7)

G43 Zz1 Hh1 ;	
:	
G28 Zz2 ;	Canceled when reference position is reached. (Same as G49)
:	
G43 Zz2Hh2 ;	
:	
G49 G28Zz2 ;	The tool length compensation will be included when positioning the intermediate point. Canceled when reference position is reached.

(2) The movement is commanded to the G53 machine coordinate system, the axis will move to the machine position when the tool compensation amount is canceled.
When the G54 to G59 workpiece coordinate system is returned to, the position returned to will be the coordinates shifted by the tool compensation amount.

Movement by tool length compensation command

When the independent tool length compensation command (G43/G44) or independent tool length compensation cancel command (G49) is issued, the movement by the compensation amount may not be carried out depending on the MTB specifications (parameter "#1247/bit0" (movement by tool length compensation command)).

	Without movement (#1247/bit0 = 1)		With movement (#1247/bit0 = 0)	
Independent tool length compensation command	: G54 A0. C0. G68.2 G53.1 G00XxYyZz G43H1 : G49		: G54 A0. C0. G68.2 G53.1 G00XxYyZz G43H1 : G49	
	The axis does not move when Tool length compensation is commanded independently, but the tool compensation amount is applied to the program position counter.		The axis moves by the compensation amount when Tool length compensation is commanded independently.	
Tool length compensation command with movement	: G54 A0. C0. G68.2 G53.1 G00XxYyZz G43H1Z0 : G49Z10.		: G54 A0. C0. G68.2 G53.1 G00XxYyZz G43H1Z0 : G49Z10.	
	The axis moves by the compensation amount when the tool length compensation with travel is commanded.		The axis moves by the compensation amount when the tool length compensation with travel is commanded.	

12.3 Tool Radius Compensation ; G38,G39/G40/G41,G42

Function and purpose

This function compensates the radius of the tool. The compensation can be done in the random vector direction by the radius amount of the tool selected with the G command (G38 to G42) and the D command.
When using tool nose radius compensation, refer to "12.4 Tool Nose Radius Compensation (for Machining Center System)".

Command format

G40 X__Y__;	**Tool radius compensation cancel**

G41 X__Y__ D__;	**Tool radius compensation (Left)**

G42 X__Y__ D__;	**Tool radius compensation (Right)**

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

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

Detailed description

The number of sets for the compensation differ according to machine specification. (The No. of sets is the total of the tool length offset, tool position offset and tool radius compensation sets.)
The H command is ignored during the tool radius compensation, and only the D command is valid.
The compensation will be executed within the plane designated with the plane selection G code or axis address 2 axis, and axes other than those included in the designated plane and the axes parallel to the designated plane will not be affected. Refer to the section on plane selection for details on selecting the plane with the G code.

12.3.1 Tool Radius Compensation Operation

Detailed description

Tool radius compensation cancel mode

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

(1) After the power has been switched on
(2) After the reset button on the setting and display unit has been pressed
(3) After the M02 or M30 command with reset function has been executed
(4) After the compensation cancel command (G40) is issued

The compensation vectors are zero in the compensation cancel mode, and the tool nose point path coincides with the programmed path.
Programs including tool radius compensation must be terminated in the compensation cancel mode.

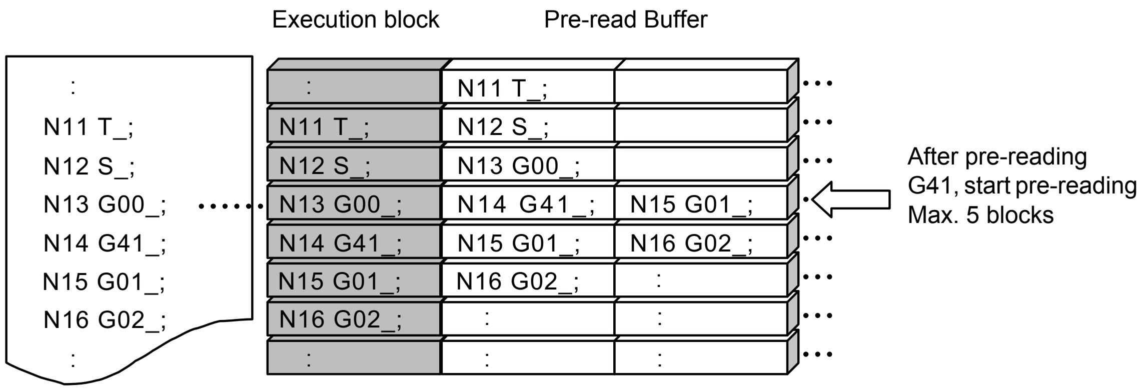
Tool radius compensation start (startup)

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

(1) The movement command is issued after G41 or G42.
(2) The tool radius compensation offset No. is 0 < D <= max. offset No.
(3) The movement command of positioning (G00) or linear interpolation (G01) is issued.

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

[Control state diagram]

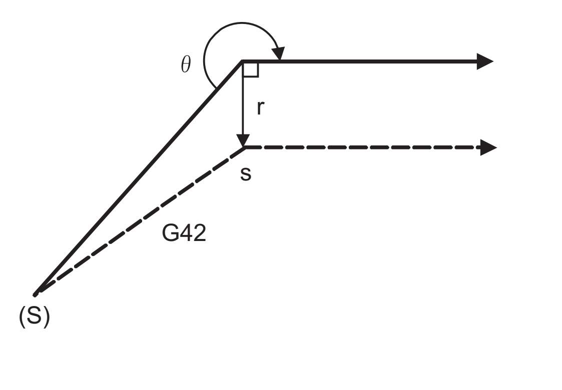


There are two ways of starting the compensation operation: type A and type B.
The type depends on the setting of the parameter "#8157 Radius comp type B".
This type is used in common with the compensation cancel type.

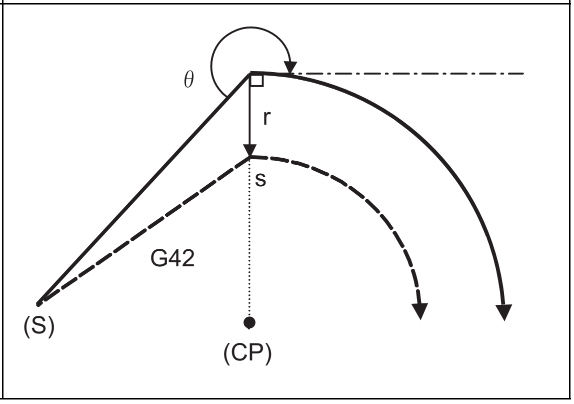
Start operation for tool radius compensation

(1) Machining an inside corner

Linear -> Linear



Linear -> Circular

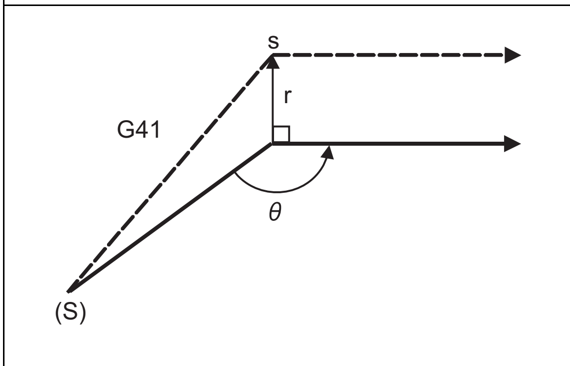


(S) Start point
r : Compensation amount
—— Program path

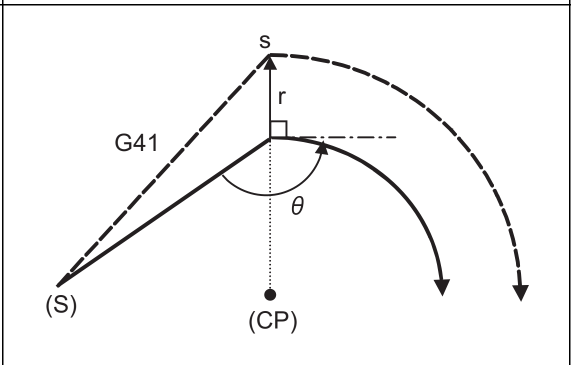
(CP) Center of circular
s: Stop point with single block
---- Tool center path

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

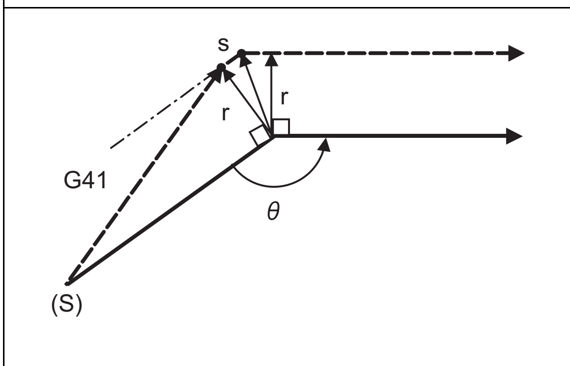
Linear -> Linear (Type A)



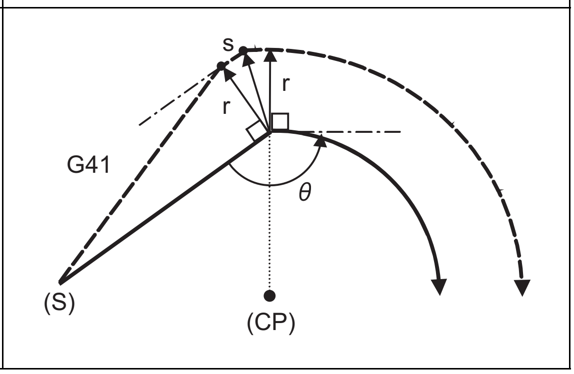
Linear -> Circular (Type A)



Linear -> Linear (Type B)



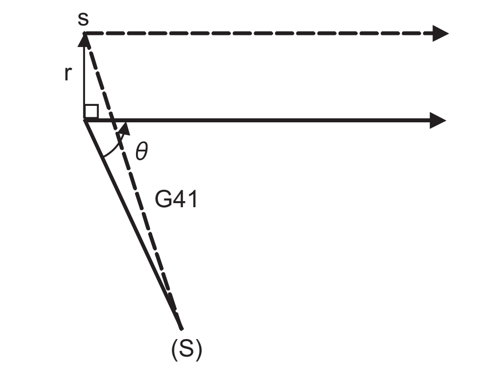
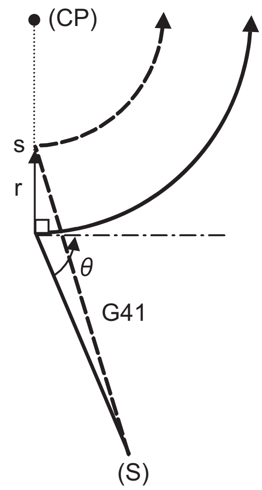
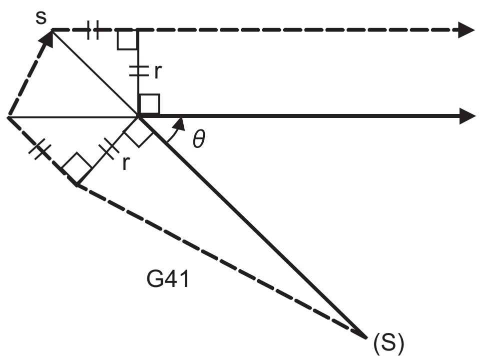
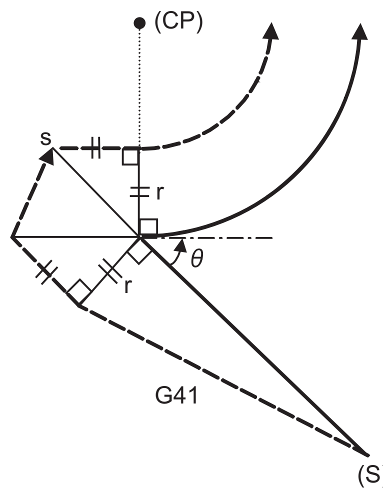
Linear -> Circular (Type B)



(S) Start point
r : Compensation amount
—— Program path

(CP) Center of circular
s: Stop point with single block
---- Tool center path

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

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

(S) Start point
r : Compensation amount
——— Program path

(CP) Center of circular
s: Stop point with single block
- - - - Tool center path

Note

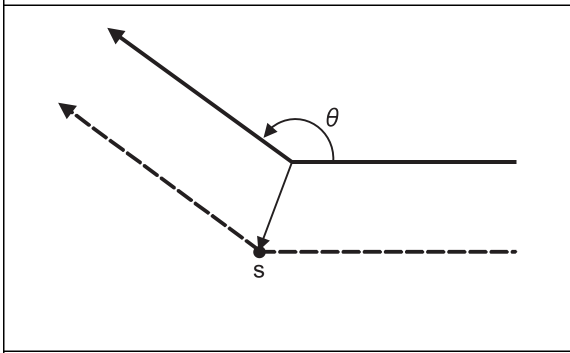
- If there is no axis movement command in the same block as G41 or G42, compensation is performed perpendicularly to the next block's direction.

Operation in compensation mode

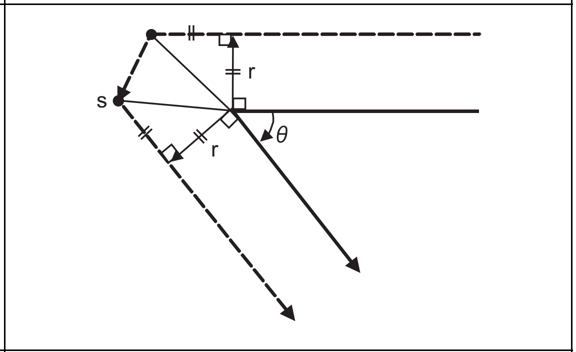
Calculate the tool center path from the linear line/circular arc to perform compensation to the program path (G00, G01, G02, G03).
Even if the same compensation command (G41, G42) is issued in the compensation mode, the command will be ignored.
When 4 or more blocks without movement command are continuously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

(1) Machining an outside corner

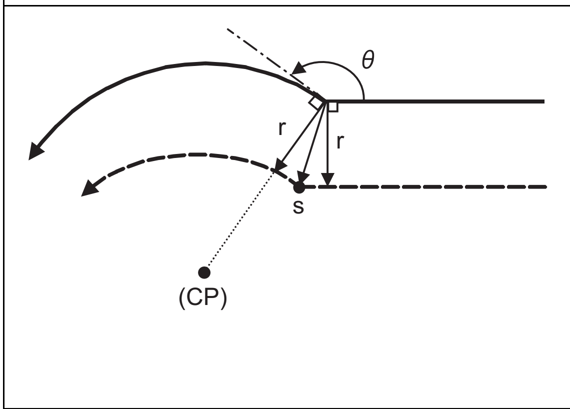
Linear -> Linear (90°<= θ < 180°)



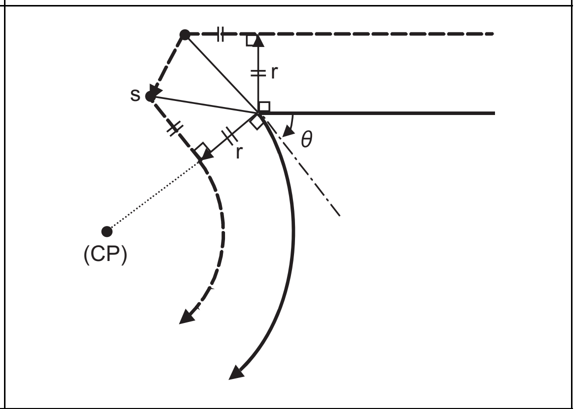
Linear -> Linear (0° < θ < 90°)



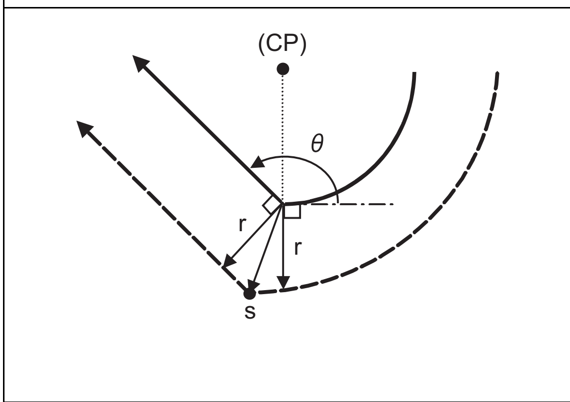
Linear -> Circular (90° <= θ < 180°)



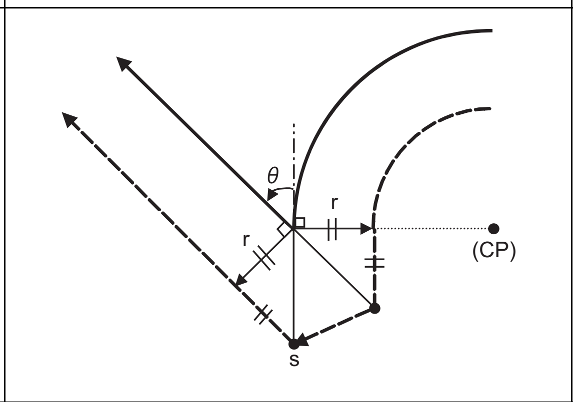
Linear -> Circular (0° < θ < 90°)



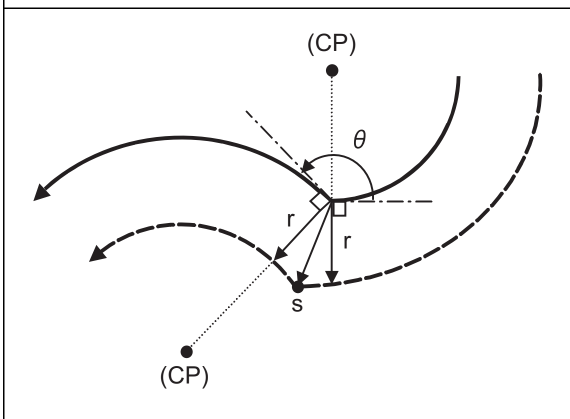
Circular -> Linear (90° <= θ < 180°)



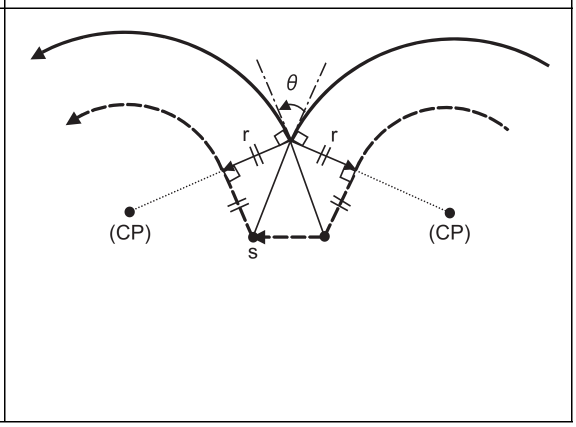
Circular -> Linear (0° < θ < 90°)



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



Circular -> Circular (0° < θ < 90°)



(CP) Center of circular

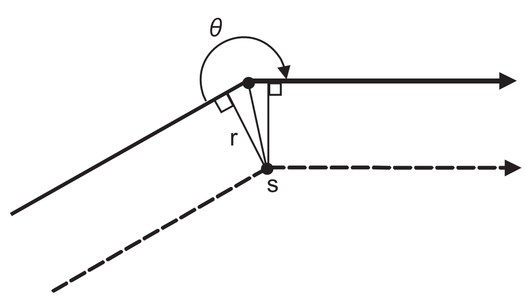
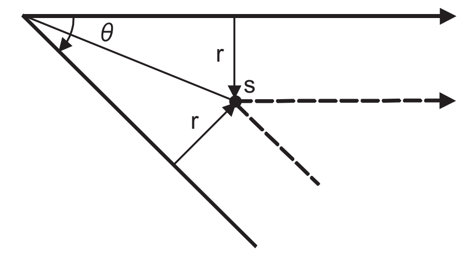
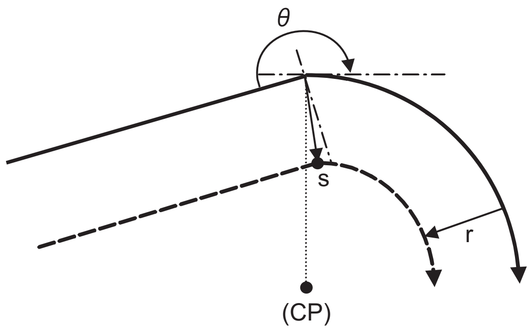
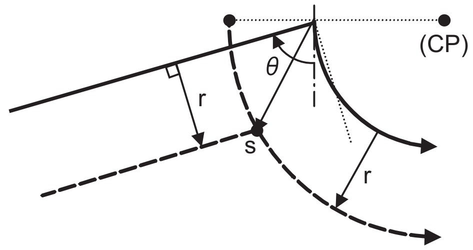
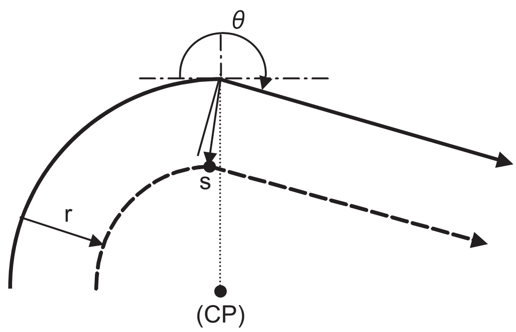
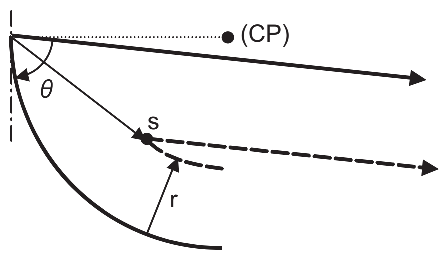
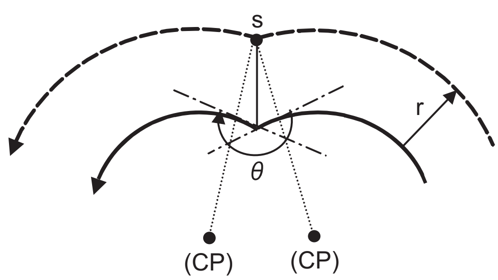
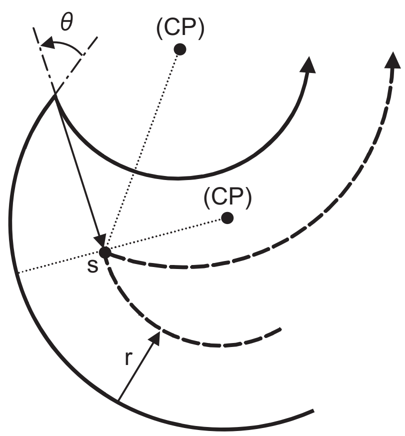
r : Compensation amount

s : Single block stop point

—— Program path

– – – – Tool center path

(2) Machining an inside corner

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

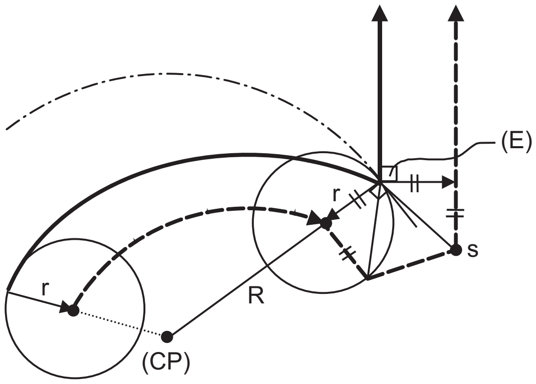
(CP) Center of circular

r : Compensation amount   s : Single block stop point

—— Program path   ---- Tool center path

(3) When the circular end point is not on the circular

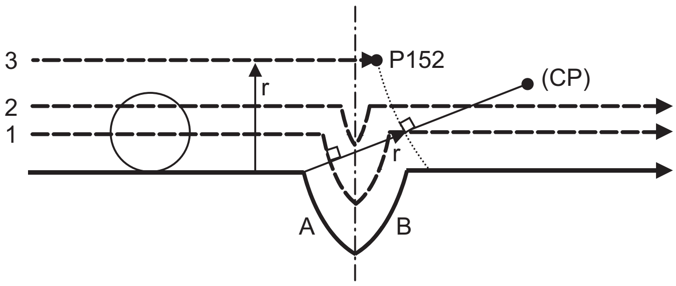
Spiral circular command ... The area from the arc start point to the end point is interpolated as a spiral arc.
Normal circular command If the error after compensation is within the parameter value ("#1084 RadErr"), it is interpolated as a spiral arc.



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

(4) When the inner intersection point does not exist

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



(CP) Center of circular A
r : Compensation amount

Program path — Tool center path

Tool radius compensation cancel

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

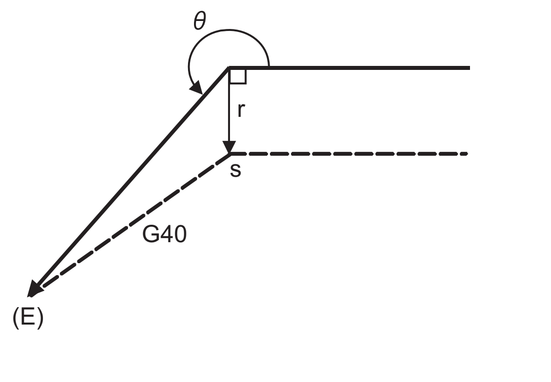
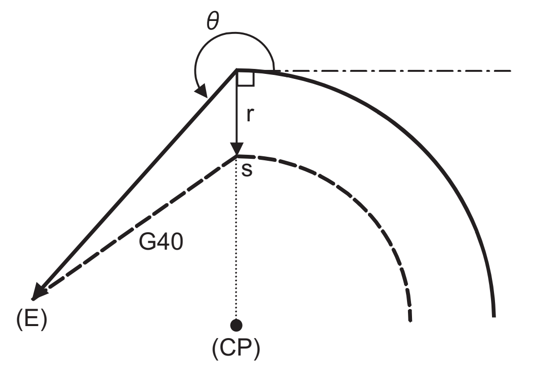
(1)The G40 command has been executed.

(2)Executed the compensation No.D00.

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

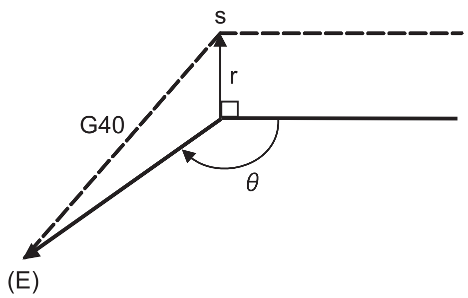
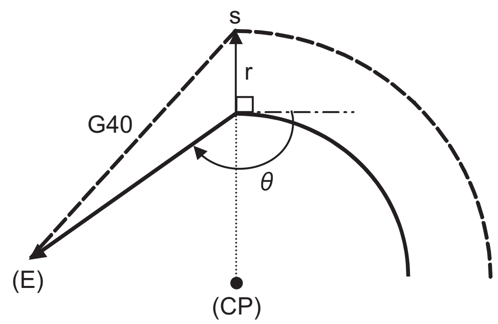
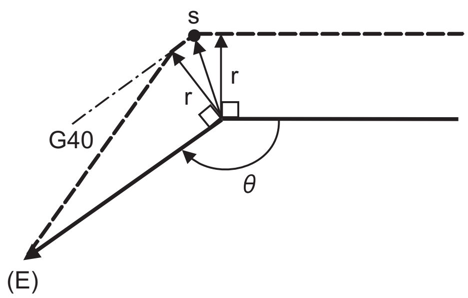
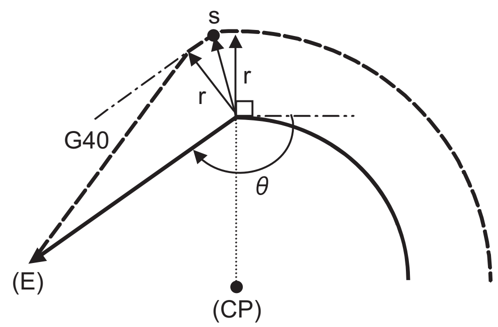
Tool radius compensation cancel operation

(1) Machining an inside corner

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

(E) End point
(CP) Center of circular
r : Compensation amount
s : Single block stop point
Program path
Tool center path

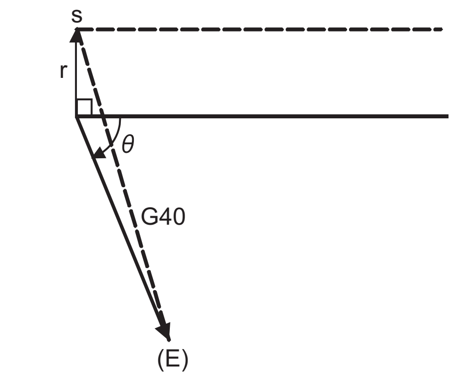
(2) Machining an outside corner (obtuse angle) [90° <= θ < 180°]

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

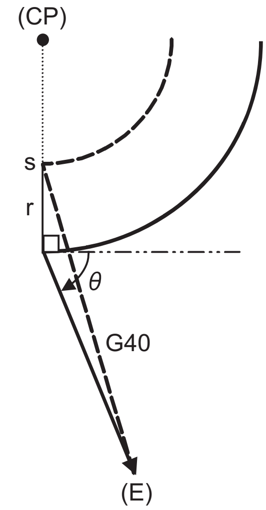
(E) End point
(CP) Center of circular
r : Compensation amount
s : Single block stop point
Program path
Tool center path

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

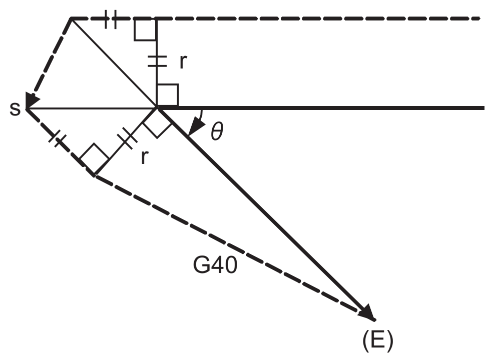
Linear -> Linear (Type A)



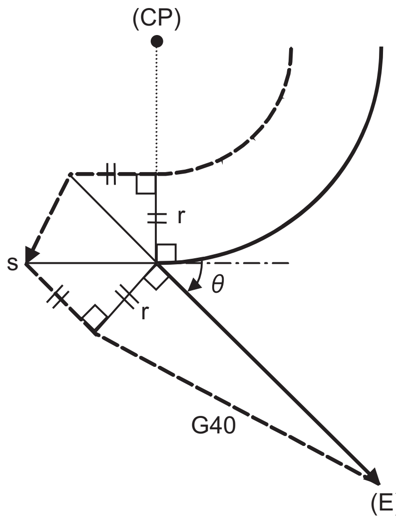
Circular -> Linear (Type A)



Linear -> Linear (Type B)



Circular -> Linear (Type B)



(E) End point
r : Compensation amount
—— Program path

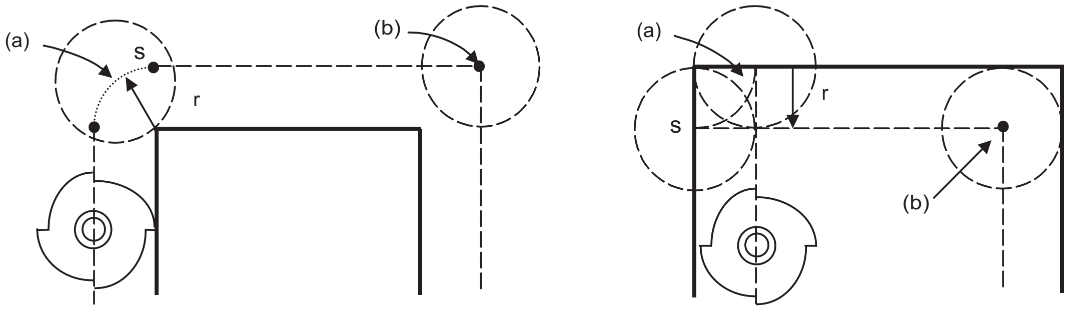
(CP) Center of circular
s : Single block stop point
- - - - Tool center path

12.3.2 Other Commands and Operations during Tool Radius Compensation

Detailed description

Insertion of corner arc

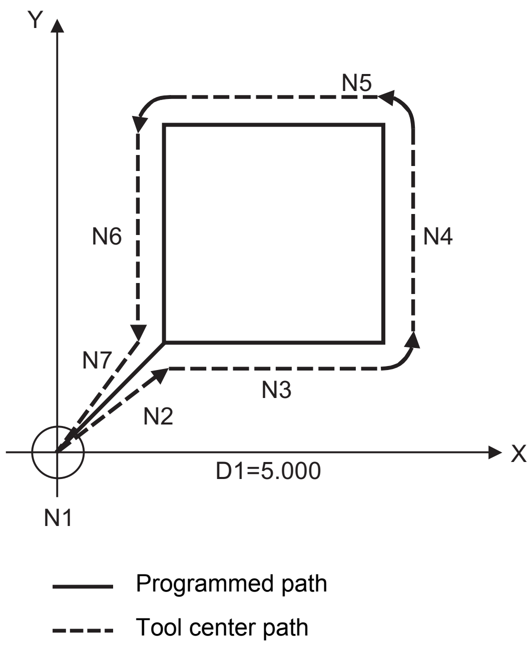
An arc that uses the compensation amount as the radius is inserted without calculating the point of intersection at the workpiece corner when G39 (corner arc) is commanded.



(With G39 command) (No G39 command)
[For outer side compensation]

(With G39 command) (No G39 command)
[For inner side compensation]

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




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

Changing and holding of compensation vector

The compensation vector can be changed or held during tool radius compensation by using the G38 command.

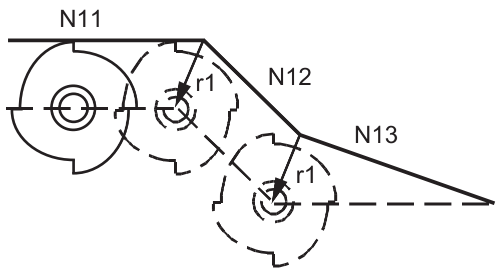
(1) **Holding of vector**

When G38 is commanded in a block having a movement command, the point of intersection will not be calculated at the program end point, and instead the vector of the previous block will be held.

G38 Xx Yy;

This can be used for pick feed, etc.

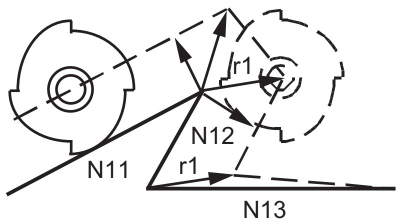
[Holding the inside compensation vector]



N11 G01 Xx11 ;
N12 G38 Xx12 Yy12 ;
N13 G40 Xx13 ;

r1: Vector at N11-N12 block intersection calculation

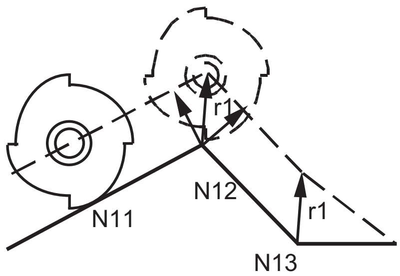
[Holding the outside compensation acute angle]



N11 G01 Xx11 Yy11 ;
N12 G38 Xx12 Yy12 ;
N13 G40 Xx13 ;

r1: Vector at N11-N12 block intersection calculation

[Holding the outside compensation obtuse angle]



N11 G01 Xx11 Yy11 ;
N12 G38 Xx12 Yy12 ;
N13 G40 Xx13 ;

r1: Vector at N11-N12 block intersection calculation

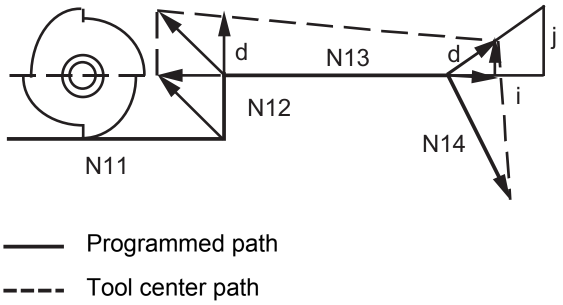
——— Programmed path
- - - - Tool center path

(2) Changing of vector

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

(These can be commanded in the same block as the movement command.)

G38 Ii Jj Dd ; (I, J and K will differ according to the selected plane.)



N11 G01 Xx11 ;
N12 Yy12 ;
N13 G38 Xx13 Ii Jj Dd ;
N14 G40 Xx14 Yy14 ;

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

Note

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

Changing the compensation direction during tool radius compensation

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

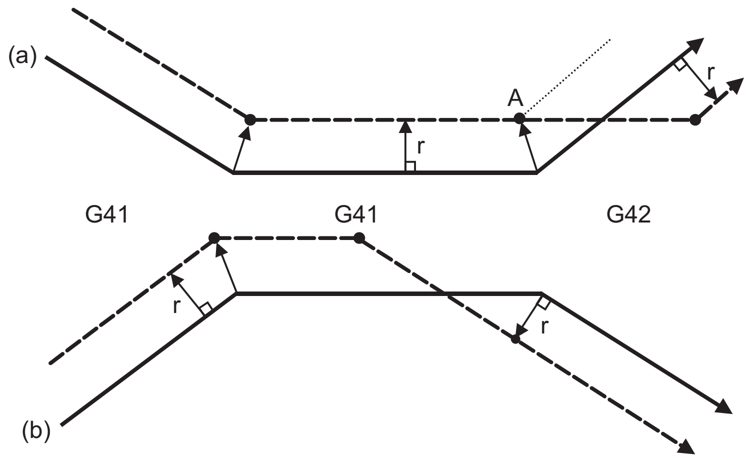
G Code	Compensation amount sign +	Compensation amount sign -
G41	Left-side compensation	Right-side compensation
G42	Right-side compensation	Left-side compensation

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

(1) Linear -> Linear

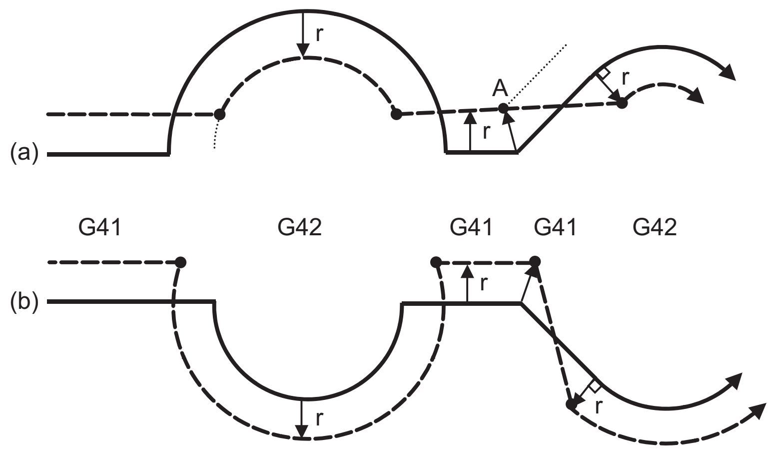
(a) When there is an intersection (A) at the change of compensation direction

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



(2) Linear <-> Circular

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



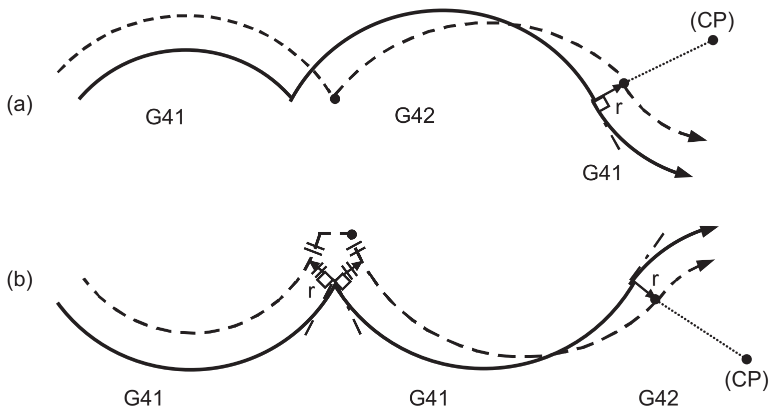
Program path　　Tool center path

(3) Circular -> Circular

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

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

(CP) Center of circular



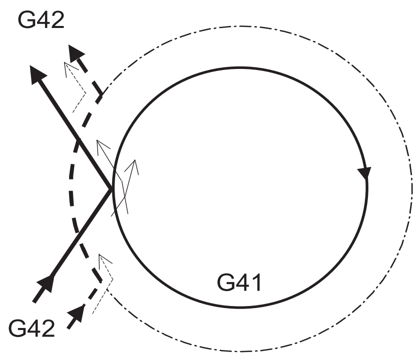
(4) Linear return

(5) Arc exceeding 360° due to compensation method.

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

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_ Y_;
G41 G02 X_ Y_ I_ J_;
G42 G01 X_ Y_;
:

Program path

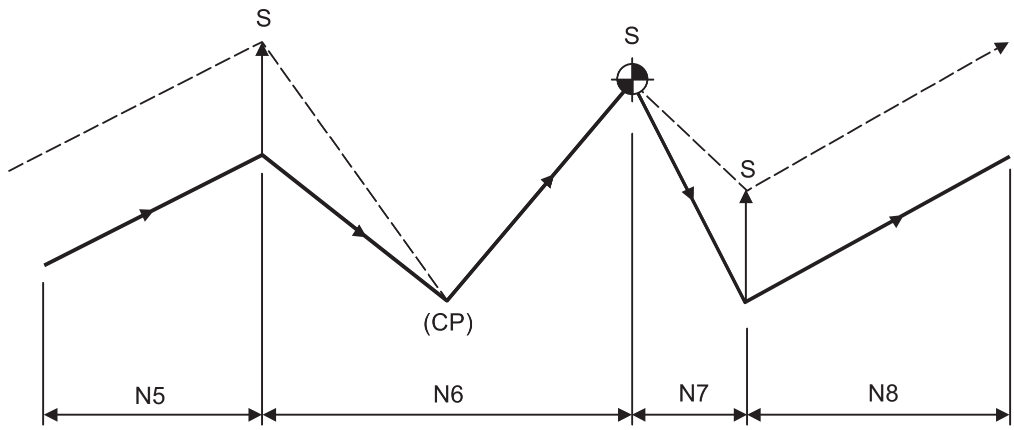
Tool center path

Section left uncut

Command for eliminating compensation vectors temporarily

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

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



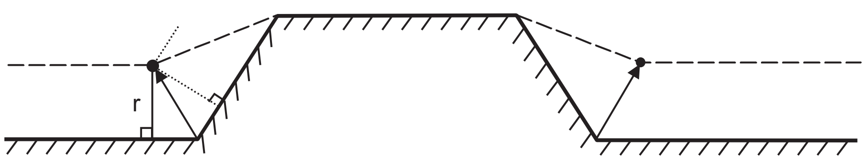

```
(G41)  :                                                    (CP) Intermediate point
N5 G91 G01 X60. Y30. ;
N6 G28 X50. Y-40. ;     ← Temporarily no compensation vectors at intermediate point.
N7      X30. Y-60. ;       (Reference position when there is no intermediate point)
N8      X70. Y40. ;
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

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

<Note>

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

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



Blocks without movement

The following blocks are known as blocks without movement.

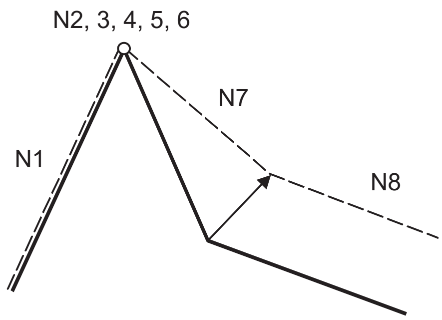
M03 ;	M command
S12 ;	S command
T45 ;	T command
G04X500 ;	Dwell
G22 X200. Y150. Z100 ;	Machining prohibited region setting
G10 L10; P01 R50;	Compensation amount setting
G92 X600. Y400. Z500. ;	Coordinate system setting
(G17) Z40. ;	Movement outside the compensation plane
G90;	G code only
G91 X0;	Movement amount 0

M00, M01, M02 and M30 are handled as pre-read inhibit M codes.

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

N1 X30.Y60.;
N2 G41 D10 ;
N3 G04 X1000 ;
N4 F100 ;
N5 S500 ;
N6 M3 ;
N7 X20.Y-50. ;
N8 X50.Y-20. ;

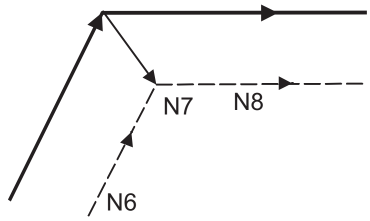
Block without movement (N2 to N6)



(2) When command is assigned in the compensation mode

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

N6 G91 X100. Y200.
;
N7 G04X P1000 ; ... Block without movement
N8 X200. ;

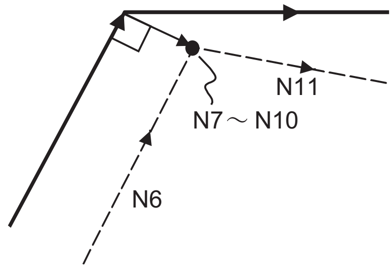


Block N7 is executed at N7 in the figure.

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

N6 X100. Y200. ;
N7 G04 X1000 ;
N8 F100 ;
N9 S500 ;
N10 M4 ;
N11 X100. ;

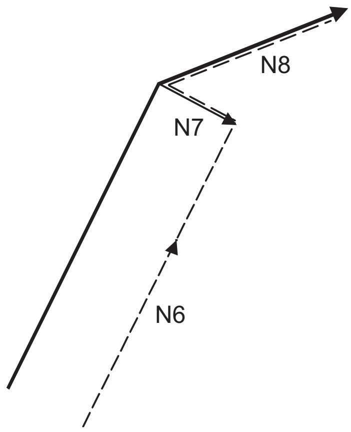
Block without movement



(3) When commanded together with compensation cancel

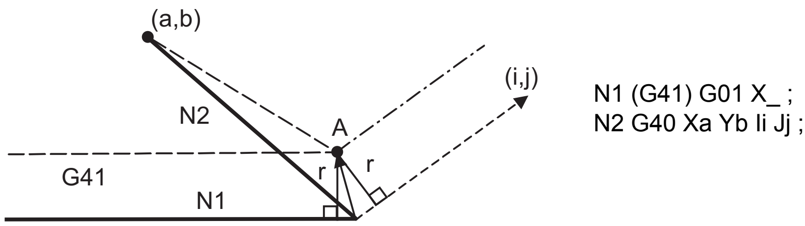
Only the compensation vectors are canceled when a block without movement is commanded together with the G40 command.

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



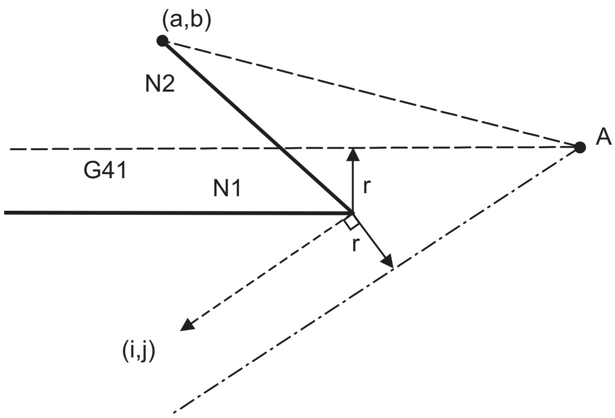
When I, J, K are commanded in G40

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

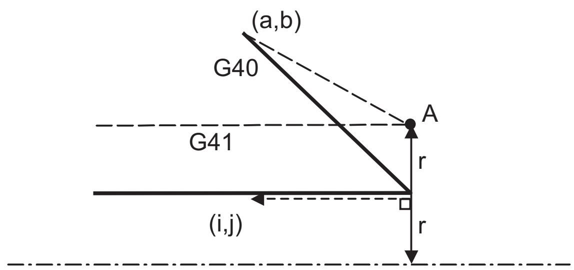


In this case, the point of intersection will always be obtained, regardless of the compensation direction, even when the commanded vector is incorrect as shown below.

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

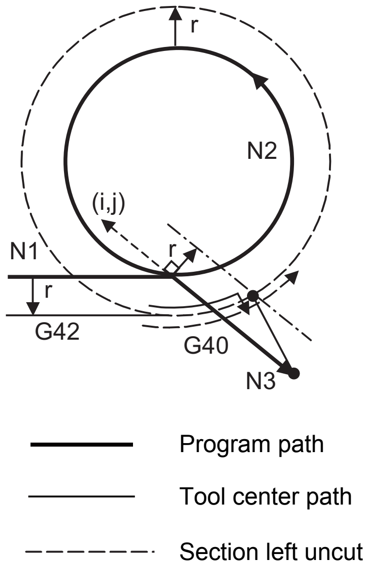


If the compensation vector obtained via a point of intersection calculation is extremely large, a perpendicular vector will be created in the block before G40.



———	Programmed path
- - - -	Tool center path
—·—·—	Hypothetical tool center path

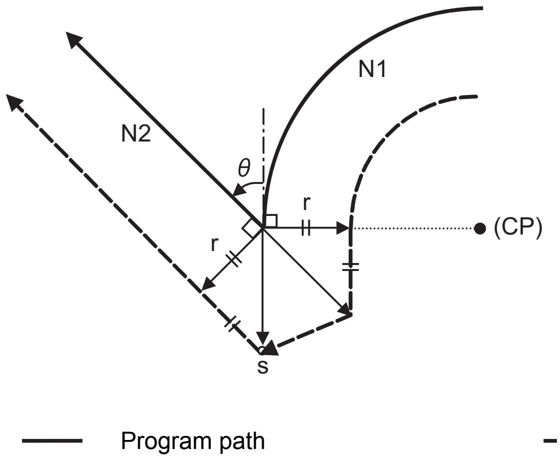
(2) If the arc is 360° or more due to the details of I, J and K at G40 after the arc command, an uncut section will occur.



N1 (G42,G91) G01 X200. ;
N2 G03 J150.;
N3 G40 G01 X150. Y-150. I-100. J100. ;

Corner movement

When a multiple number of compensation vectors are created at the joints between movement command blocks, the tool will move in a straight line between these vectors. This action is called corner movement.
When the vectors do not coincide, the tool moves in order to machine the corner although this movement is part and parcel of the joint block.
Consequently, operation in the single block mode will execute the previous block + corner movement as a single block and the remaining joining movement + following block will be executed as a single block in the following operation.



(CP) Center of circular
r : Compensation amount
s : Single block stop point

12.3.3 G41/G42 Commands and I, J, K Designation

Function and purpose

The compensation direction can be intentionally changed by issuing the G41/G42 command and I, J, K in the same block.

Command format

G17 (X-Y plane) G41/G42 X__ Y__ I__ J__ ;

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

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

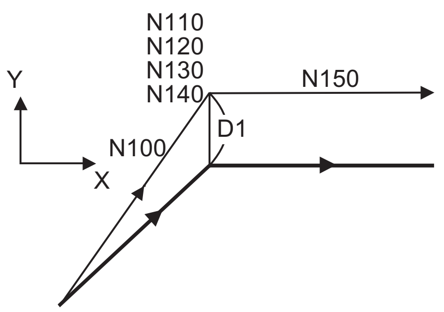
Assign a linear command (G00, G01) in a movement mode.

Detailed description

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

This section describes the new I,J type vectors (G17 plane) created by this command. (Similar descriptions apply to vector K, I for the G18 plane and to J, K for the G19 plane.)
As shown in the following figures, I, J type vectors create compensation vectors which are perpendicular to the direction designated by I, J and equivalent to the compensation amount, without the intersection point calculation of the programmed path. The I, J vectors can be commanded even in the mode (G41/G42 mode in the block before) and even at the compensation start (G40 mode in the block before).

(1) When I, J is commanded at compensation start

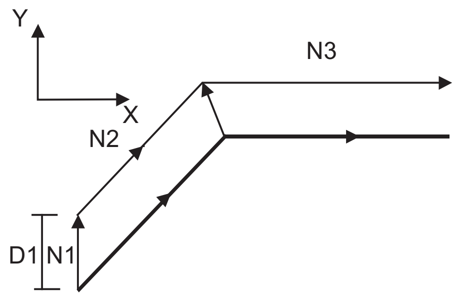



```
(G40)
 ~
N100  G91  G41  X100. Y100. I150. D1 ;
N110  G04  X1000 ;
N120  G01  F1000 ;
N130  S500 ;
N140  M03 ;
N150  X150. ;
 ~
```

Program path

Tool center path

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

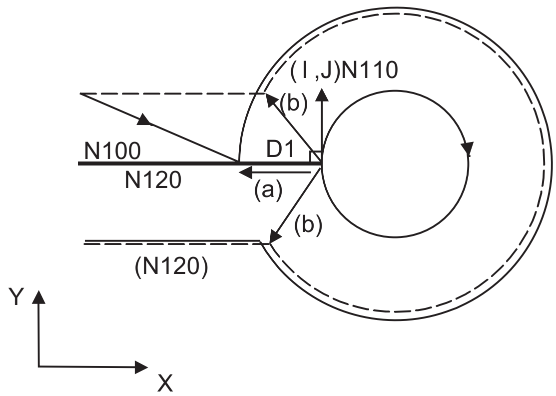



```
(G40)
~
N1  G41  I150. D1 ;
N2  G91  X100. Y100. ;
N3  X150. ;
~
```

Program path

Tool center path

(3) When I, J has been commanded in the G41/42 mode (G17 plane)




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

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

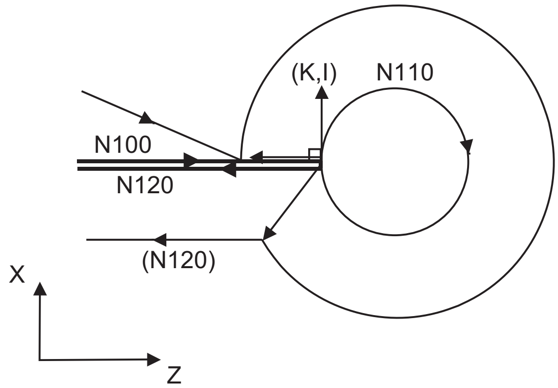
Program path

Tool center path

Path after intersection point calculation

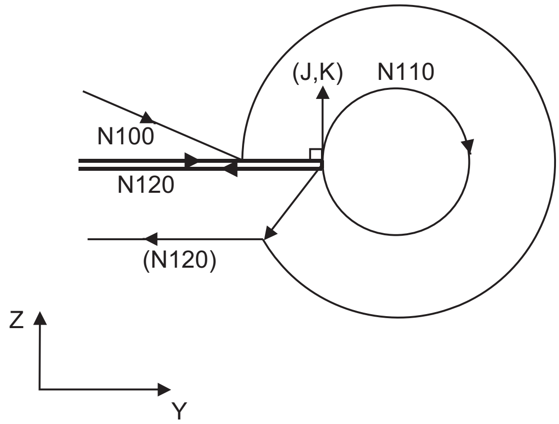
(Reference)

(a) G18 plane



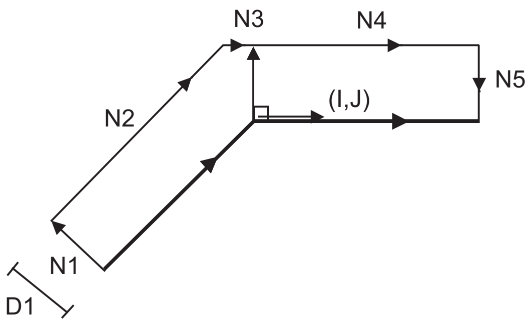

```
~
(G18  G41  G91)
N100  G41  G00  Z150. I50. ;
N110  G02  K50. ;
N120  G00  Z - 150. ;
~
```

(b) G19 plane




```
~
(G19  G41  G91)
N100  G41  G00 Y150. K50. ;
N110  G02  J50. ;
N120  G00  Y - 150. ;
~
```

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

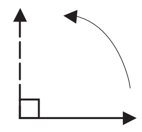
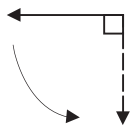



```
N1  G41  D1  G01  F1000 ;
N2  G91  X100. Y100. ;
N3  G41  I50. ;
N4  X150. ;
N5  G40 ;
```

Compensation vector direction

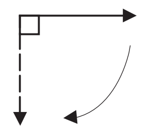
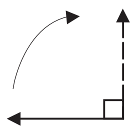
(1) In G41 mode

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

(Example 1) With I100.		(Example 2) With I-100.	
			
←———	(100, 0) IJ direction	←———	(-100, 0) IJ direction
←– – – –	Compensation vector direction	←– – – –	Compensation vector direction

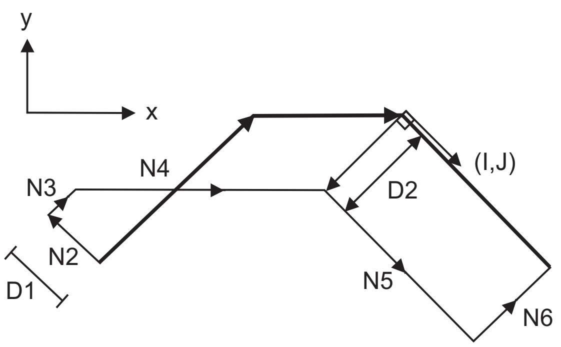
(2) In G42 mode

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

(Example 1) With I100.		(Example 2) With I-100.	
			
←———	(100, 0) IJ direction	←———	(-100, 0) IJ direction
←– – – –	Compensation vector direction	←– – – –	Compensation vector direction

Selection of compensation modal

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

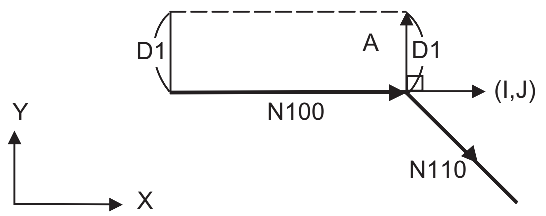



```
N1  G28  X0 Y0  ;
N2  G41  D1  F1000  ;
N3  G01  G91  X100.  Y100.  ;
N4  G42  X100.  I100.  J - 100.
     D2  ;
N5  X100.  Y- 100.  ;
N6  G40  ;
N7  M02  ;
%
```

Compensation amount for compensation vectors

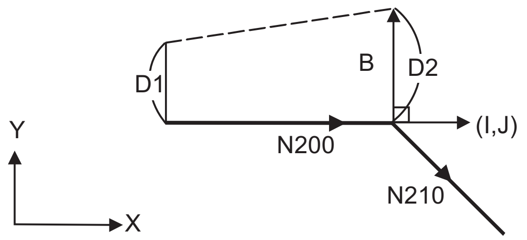
The compensation amount is determined by the offset No. (modal) in a block with the IJ designation.

<Example 1>
Vector A is the compensation amount registered in the compensation No. modal D1 of the N100 block.




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

<Example 2>
Vector B is the compensation amount registered in the compensation No. modal D2 of the N200 block.

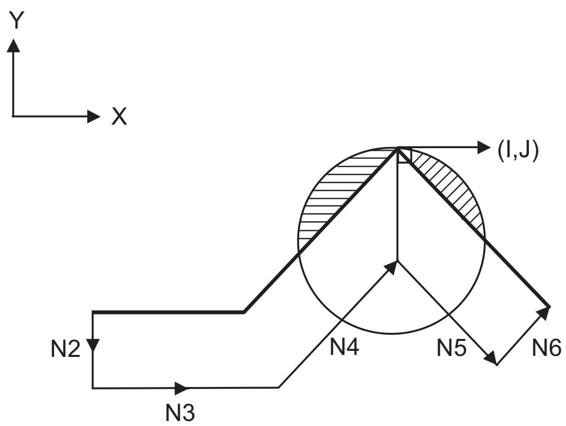



```
(G41  D1  G91)
        ~
N200  G41  X150.  I50.  D2 ;
N210  X100.  Y-100.  ;
        ~
```

Precautions

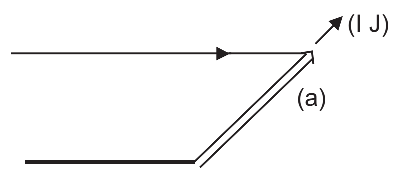
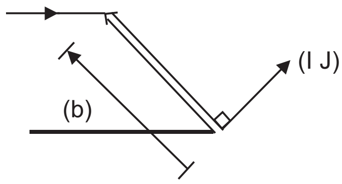
(1) Issue the I, J type vector in a linear mode (G00, G01). If it is in an arc mode at the start of compensation, program error (P151) will occur.
When it is in the compensation mode as well as in the arc mode, I, J will be designated at the center of the circular.

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



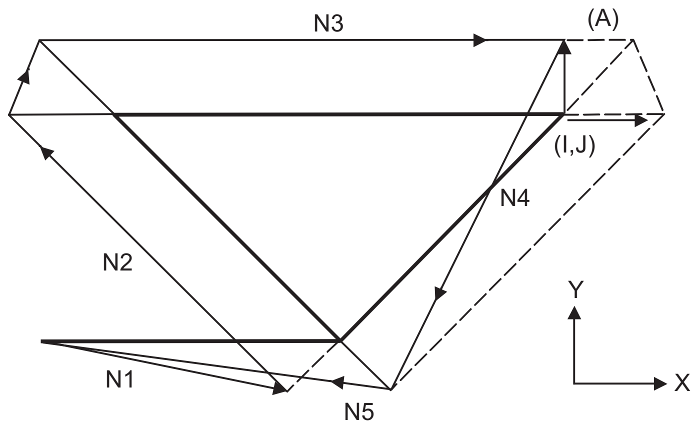

```
N1  G28  X0 Y0  ;
N2  G42  D1  F1000  ;
N3  G91  X100.  ;
N4  G42  X100.  Y100. I10.  ;
N5  X100.  Y-100.  ;
N6  G40  ;
N7  M02  ;
```

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

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

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

G41/42	I, J (K)	Compensation methods
No	No	Intersection point calculation type vector
No	Yes	Intersection point calculation type vector
Yes	No	Intersection point calculation type vector
Yes	Yes	I, J, type vector No insertion block



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

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

12.3.4 Interrupts during Tool Radius Compensation

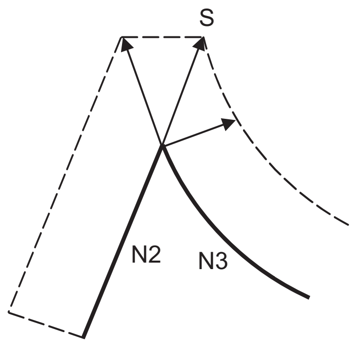
Detailed description

MDI interruption

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

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

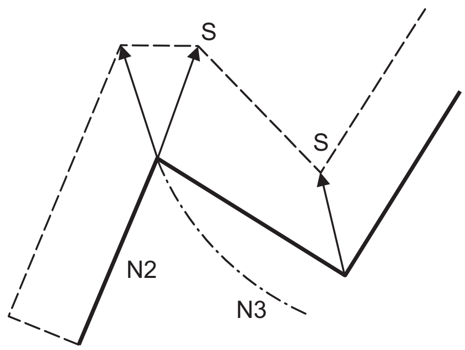
Automatic operation	MDI interruption
N1 G41 D1;	
N2 X20. Y50. ;	
	<--- S1000 M3;
N3 G03 X40. Y-40. R70. ;	



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

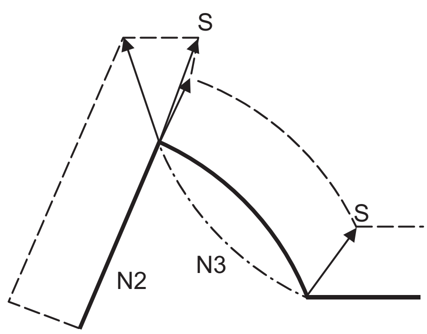
With linear interrupt

Automatic operation	MDI interruption
N1 G41 D1;	
N2 X20. Y50. ;	
	<--- X50. Y-30. ;
	X30. Y50. ;
N3 G03 X40.Y-40. R70. ;	



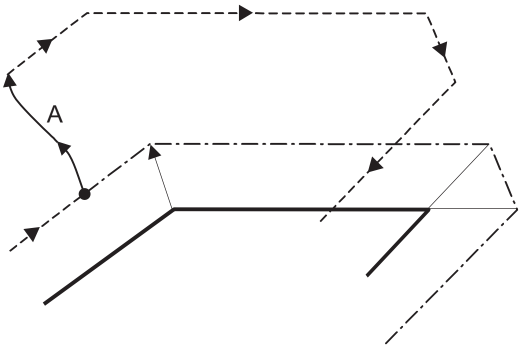
With circular interruption

Automatic operation	MDI interruption
N1 G41 D1;	
N2 X20. Y50. ;	
	<--- G02 X40. Y-40. R70. ;
	G01 X40. ;
N3 G03 X40. Y-40. R70. ;	



Manual interruption

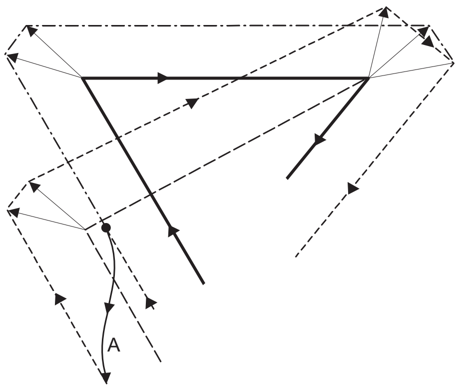
(1) Interrupt with manual absolute OFF.
The tool path will deviate from the compensated path by the interrupt amount.



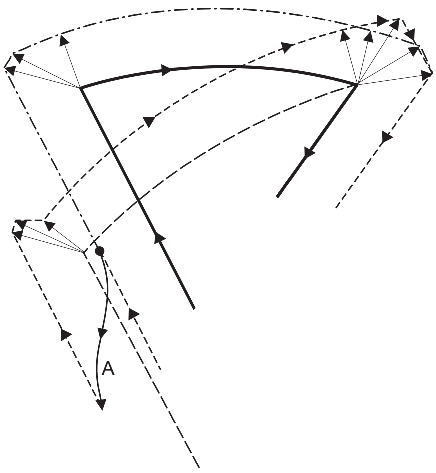
Program path
Tool path after compensation
Interrupt (A)
Tool path after interrupt

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

[Line-Line-Line]



[Line-arc-Line]



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

12.3.5 General Precautions for Tool Radius Compensation

Precautions

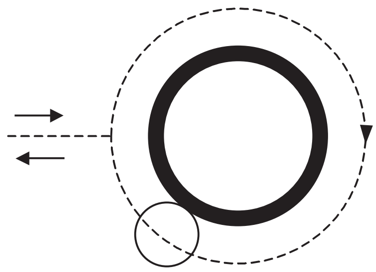
Assigning the compensation amounts

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

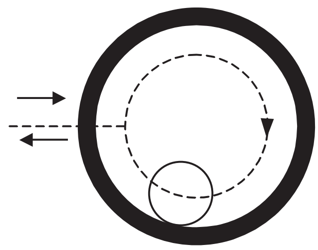
(2) Compensation amounts are normally changed when a different tool has been selected in the compensation cancel mode. However, when an amount is changed during the compensation mode, the vectors at the end point of the block are calculated using the compensation amount designated in that block.

Compensation amount symbols and tool center path

If the compensation amount is negative (-), the figure will be the same as if G41 and G42 are interchanged. Thus, the axis that was rotating around the outer side of the workpiece will rotate around the inner side, and vice versa. An example is shown below. Normally, the compensation amount is programmed as positive (+). However, if the tool path center is programmed as shown in (a) and the compensation amount is set to be negative (-), the movement will be as shown in (b). On the other hand, if the program is created as shown in (b) and the offset amount is set to be negative (-), the movement will be as shown in (a). Thus, only one program is required to execute machining of both male and female shapes. The tolerance for each shape can be randomly determined by adequately selecting the offset amount.
(Note that a circle will be divided with type A when compensation is started or canceled.)



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



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

---- Tool center path

12.3.6 Changing of Compensation No. during Compensation Mode

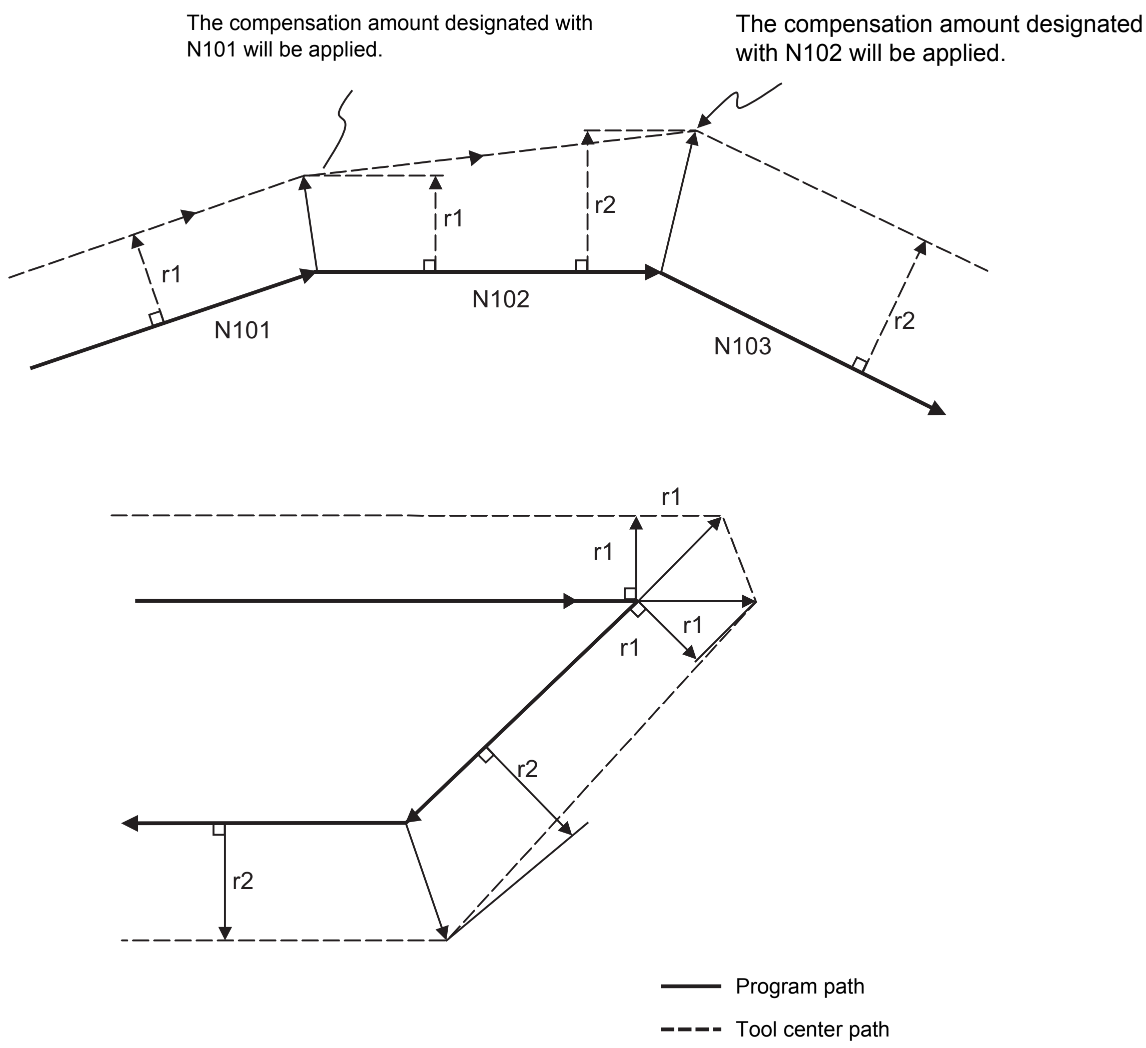
Function and purpose

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

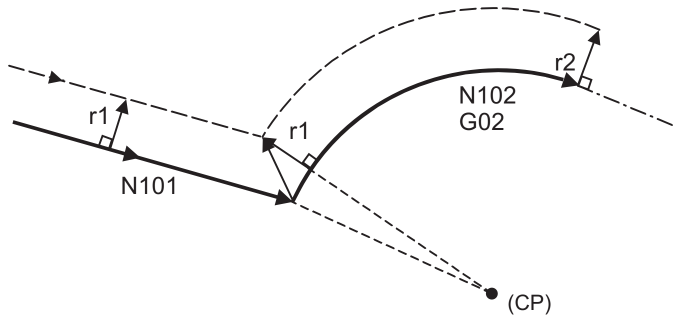
When compensation No. (compensation amount) is changed:

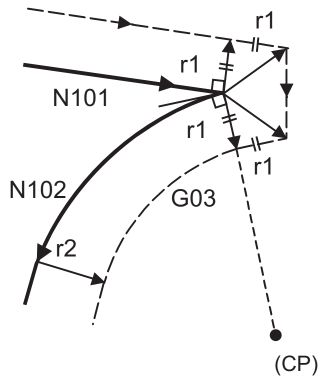
G41 G01 .... Dr1 ;
(α = 0,1,2,3)
N101 G0α Xx1 Yy1 ;
N102 G0α Xx2 Yy2 Dr2 ; .... Compensation No. changed
N103 Xx3 Yy3 ;

During linear -> linear



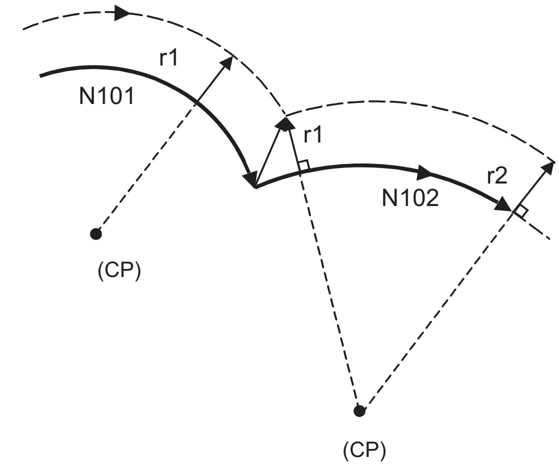
Linear ->circular

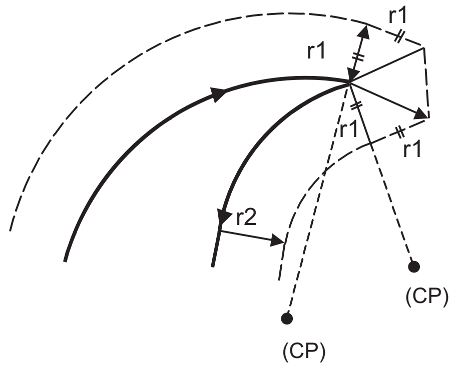




(CP) Arc center
Program path
Tool center path

Circular -> circular





(CP) Arc center
Program path
Tool center path

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

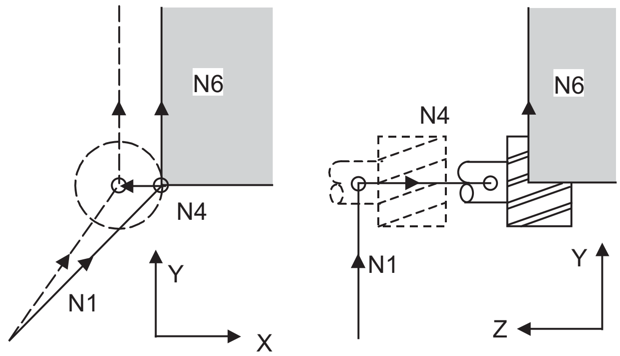
Function and purpose

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

Program example

When the following type of program is created:

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



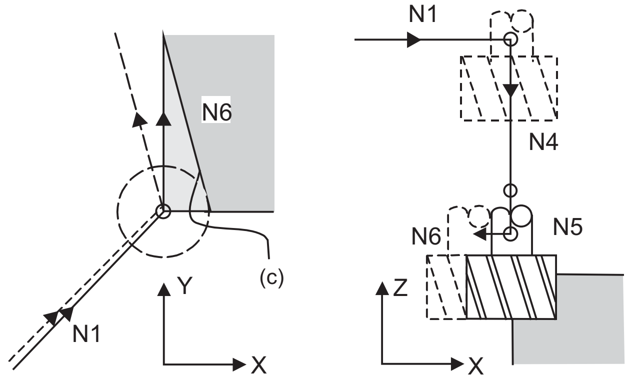
N4 Z axis lowers (1 block)

---- Tool center path

With this program, at the start of the N1 compensation the program will be read to the N6 block. The relation of N1 and N6 can be judged, and correct compensation can be executed as shown above.

If the above program's N4 block is divided into two

```
N1 G91 G00 G41 X500. Y500. D1 ;
N2 S1000 ;
N3 M3 ;
N4       Z-250. ;
N5 G01 Z-50. F1 ;
N6       Y100. F2 ;
```

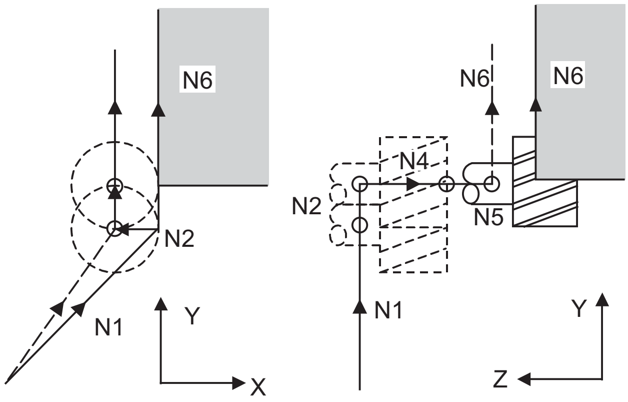


(c) Cut in

In this case, the four blocks N2 to N5 do not have a command in the XY plane, so when the N1 compensation is started, the program cannot be read to the N6 block.
As a result, the compensation is done based only on the information in the N1 block, and the compensation vector is not created at the start of compensation. Thus, an excessive cut in occurs as shown above.

In this case, consider the calculation of the inner side, and before the Z axis cutting, issue a command in the same direction as the direction that the Z axis advances in after lowering, to prevent excessive cutting.

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



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

12.3.8 Interference Check

Function and purpose

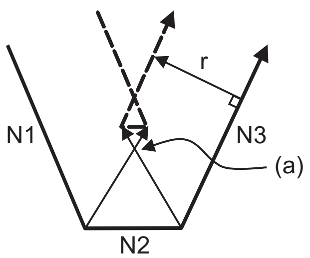
When tool radius is larger than the program path, a tool, compensated for by the tool radius compensation function, may sometimes cut into the workpiece. This is known as interference, and interference check is the function which prevents this from occurring.
The table below shows the three functions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

	Function	Parameters		Operation
		#8102 **COLL. ALM OFF**	**#8103** **COLL. CHK OFF**	
(1)	Interference check alarm function	0	0	Operation stops with a program error (P153) before executing a block which will cause cutting.
(2)	Interference check avoidance function	1	0	The tool path is changed to prevent cutting from occurring. If the path cannot be changed, a program error (P153) occurs and the program will be stopped.
(3)	Interference check invalid function	0/1	1	Cutting continues as is, even if the workpiece is cut into. Use in the fine segment program.

Detailed description

Conditions viewed as interference

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



r : Compensation amount

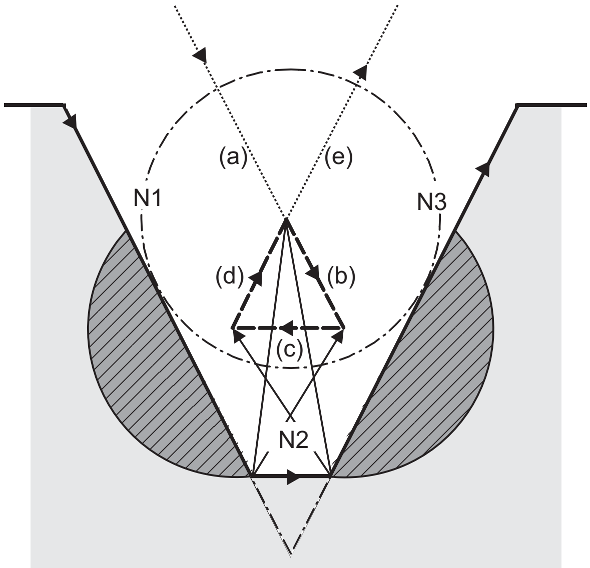
(a) Vectors intersect

 Program path

 Tool center path

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

(G41)
N1 G91 G01 X50. Y-100. ;
N2 X70. Y-100. ;
N3 X120. Y0 ;

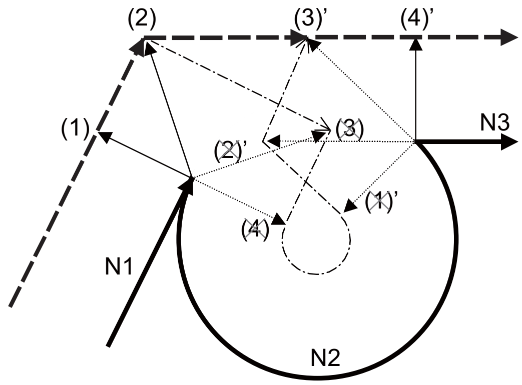


(1) With alarm function. An alarm is output before N1 is executed. The buffer correction function can thus be used to change N1 to the following, enabling machining to continue: N1 G01 X20. Y-40.;

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

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

(Example 2) When operating a program including a small circular with a tool with a large radius, cutting occurs near the start point/end point of the circular in the following figure.



Interference check processing

Vectors (1) (4)' check -> No interference

↓

Vectors (2) (3)' check -> No interference

↓

Vectors (3) (2)' check -> Interference -> Erase vectors (3) (2)'

↓

Erase vectors (4) (1)'

(1) With alarm function

The alarm occurs before N1 is executed.

(2) With avoidance function

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

(Thick broken line path)

(3) With interference check invalid function

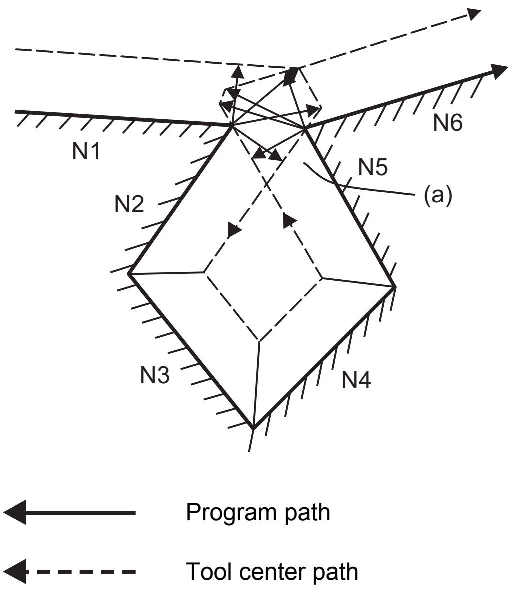
The tool center path will follow the path that connects (1), (2), (3), (4), (1)', (2)', (3)', (4)', as the interference avoidance path while cutting.

(Thin broken line path)

When interference check cannot be executed

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

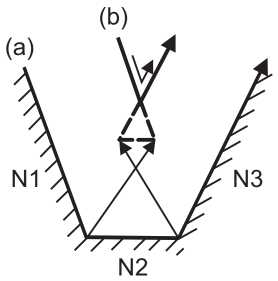
(2) When there is an interference following the fourth movement block



(a) Interference check is not possible

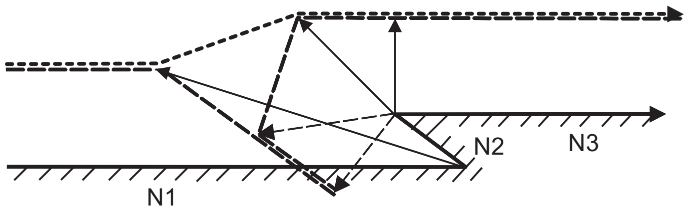
Operation when interference avoidance function is valid

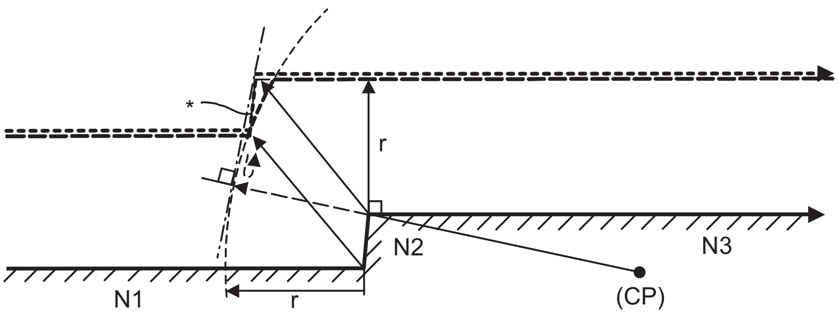
The movement will be as shown below when the interference avoidance check is valid.



(a) Program path

(b) Tool center path





Program path

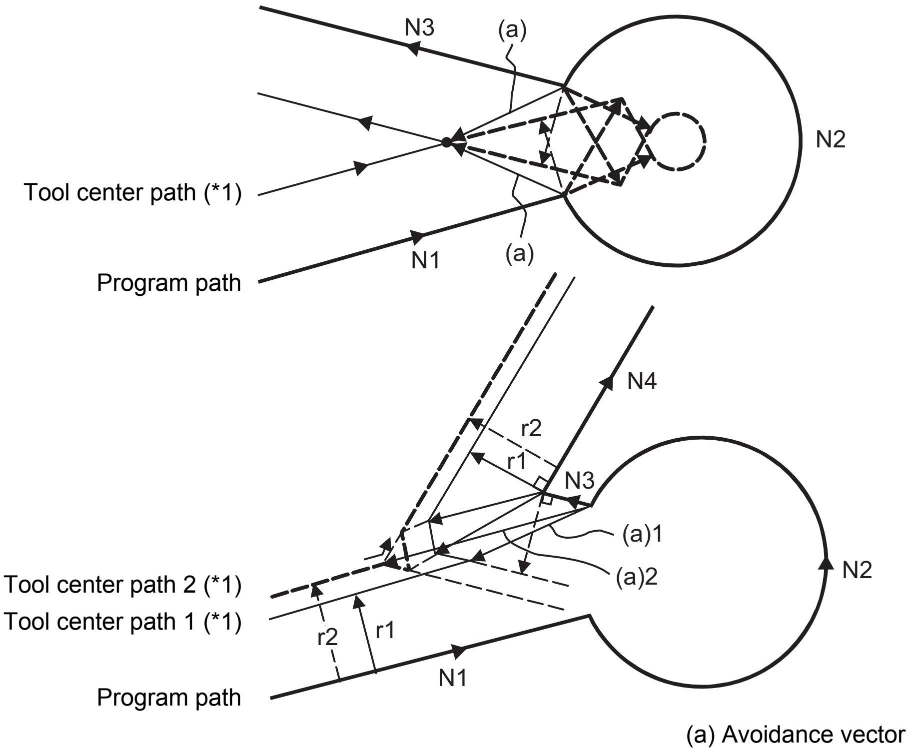
Tool center path without interference check

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

Valid vector

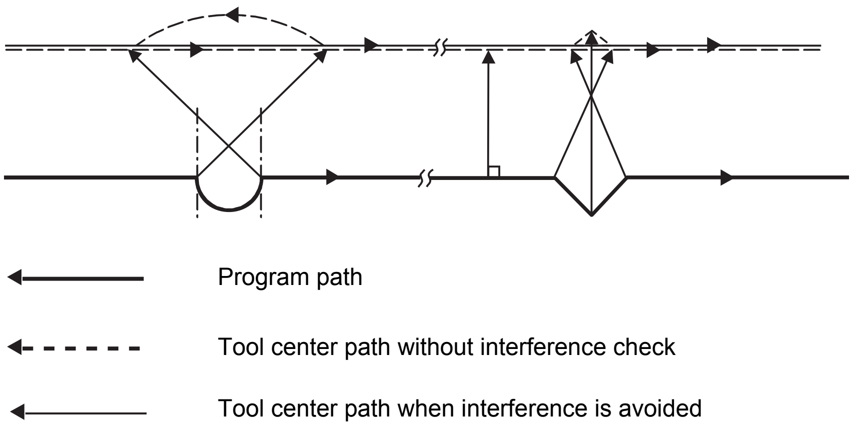
Invalid vector

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



(*1) Tool center path when interference is avoided

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

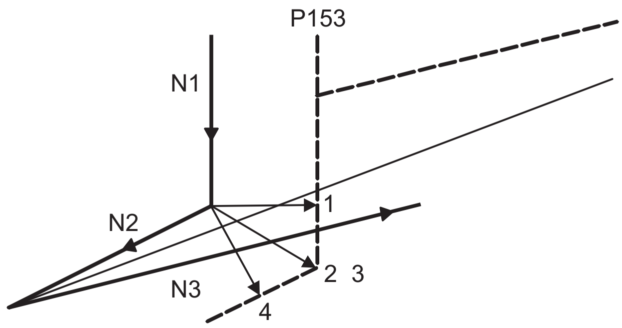


Interference check alarm operation

The interference check alarm occurs under the following conditions.

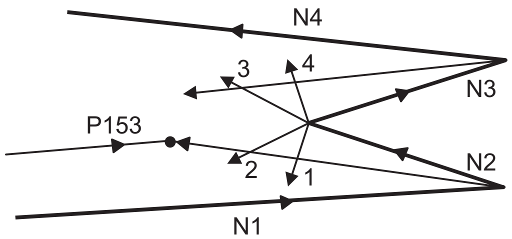
(1) When the interference check alarm function has been selected

When all vectors at the end of its own block have been deleted
As shown in the figure below, when vectors 1 through 4 at the end point of the N1 block have all been deleted, program error (P153) will occur prior to N1 execution.

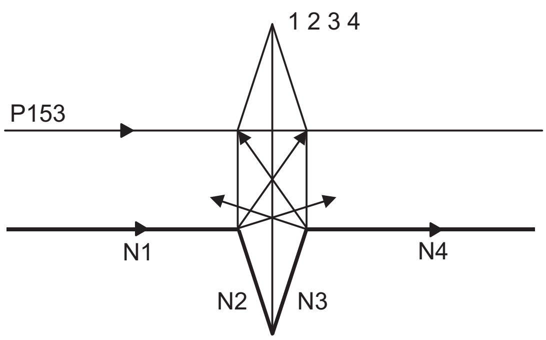


(2) When the interference check avoidance function has been selected

(Example 1) When there are valid vectors at the end point of the following blocks even when all the vectors at the end point of its own block have been deleted
When, in the figure below, the N2 interference check is conducted, the N2 end point vectors are all deleted but the N3 end point vectors are regarded as valid.
Program error (P153) now occurs at the N1 end point and the operation stops.

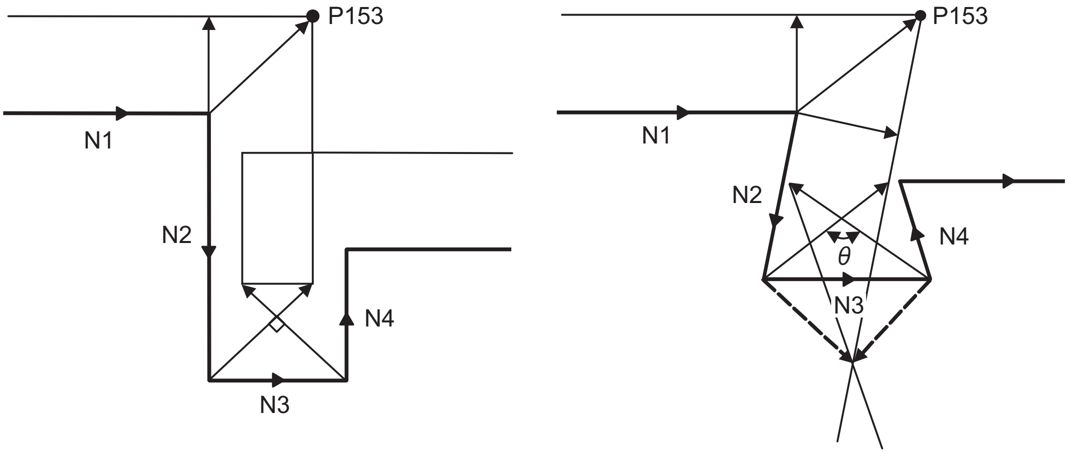


In the case shown in the figure below, the tool will move in the reverse direction at N2.
Program error (P153) now occurs before executing N1 and the operation stops.



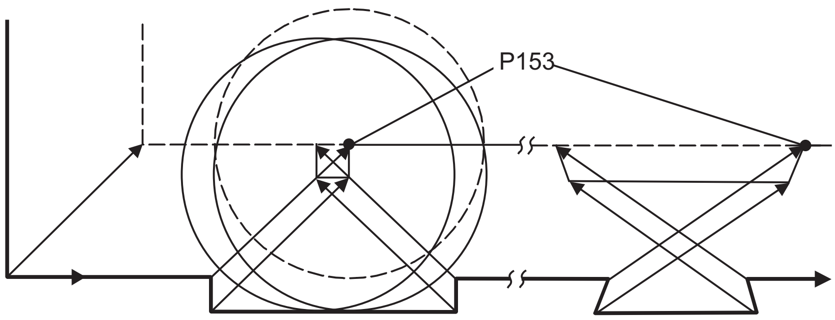
(Example 2) When avoidance vectors cannot be created

Even when, as in the figure below, the conditions for creating the avoidance vectors are satisfied, it may still be impossible to create avoidance vectors, or the interference vectors may interfere with N3. Program error (P153) will occur at the N1 end point when the vector intersecting angle is more than 90° and the operation will stop.



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

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

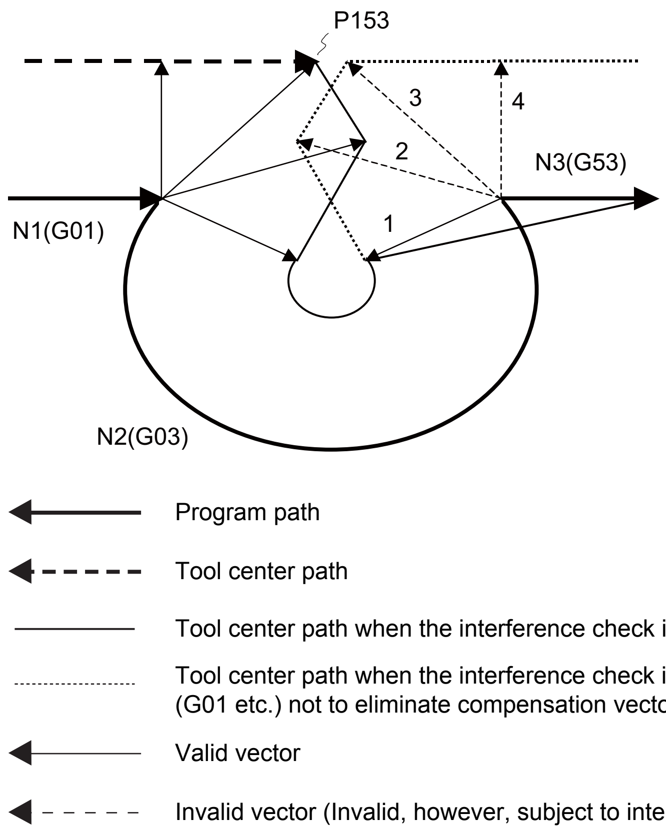


(Example 4) When vectors at the end point of the block immediately before the command to eliminate compensation vectors temporarily cause an interference

Interference check will be executed also at the end point of the block immediately before the command to eliminate compensation vectors temporarily, similarly with the case compensation vectors are not eliminated. It may be regarded as an interference even if there is actually no interference. If regarded as an interference, program error (P153) will occur.

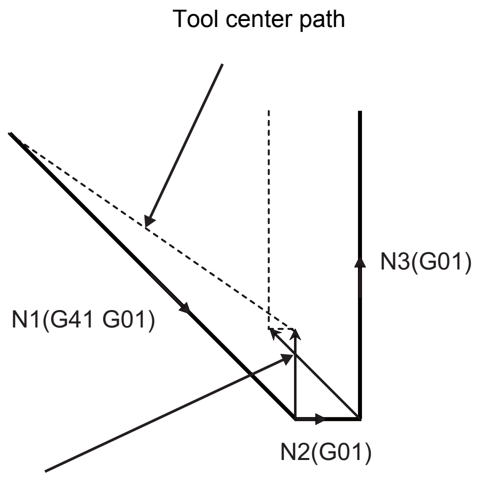
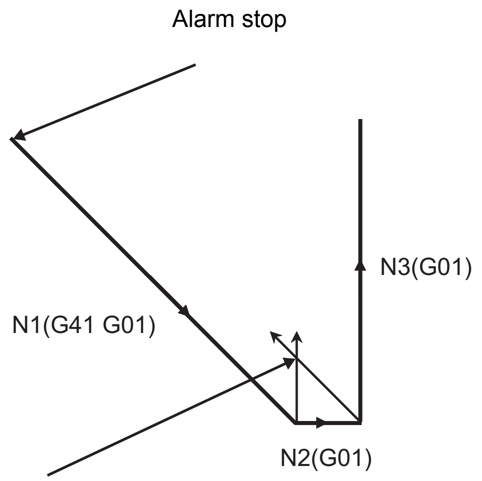
In the figure below, only vector 1 is left as an end point vector in N2 because of the N3 G53 command to temporarily eliminate compensation vectors. However, the interference check will still be conducted to vector 1 to 4 and an interference will be detected.

Program error (P153) now occurs at the end point of the previous block and the operation stops.



Interference check for start-up block

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

(a) Interference check for start-up block invalid	(b) Interference check for start-up block valid
"#1241 set13/bit1" = "0" or "#8103 COLL. CHK OFF" = "1"	"#1241 set13/bit1" = "1" and "#8103 COLL. CHK OFF" = "0"
 	
The interference with the compensation calculation vector of the contact between the N2 block and N3 block is not checked. Doing so will cause a cut in the N3 block.	The interference with the compensation calculation vector of the contact between the N2 block and N3 block is checked, and this is judged to be an alarm.

12.3.9 Diameter Designation of Compensation Amount

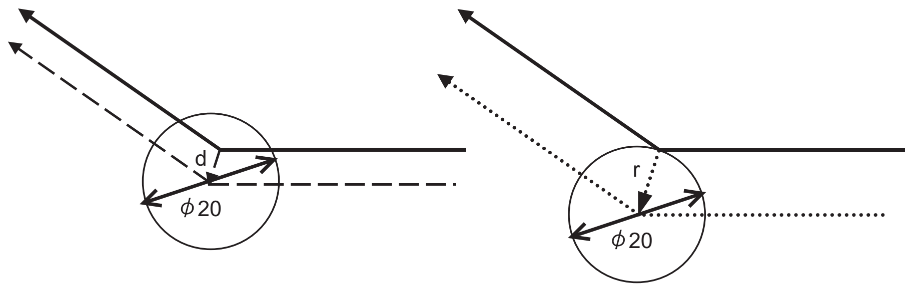
Function and purpose

With this function, the tool radius compensation amount can be designated by tool diameter. When the control parameter "#8117 OFS Diam DESIGN" is ON, the compensation amount specified to the commanded tool No. will be recognized as the diameter compensation amount, and the amount will be converted to the radius compensation amount when executing the compensation.

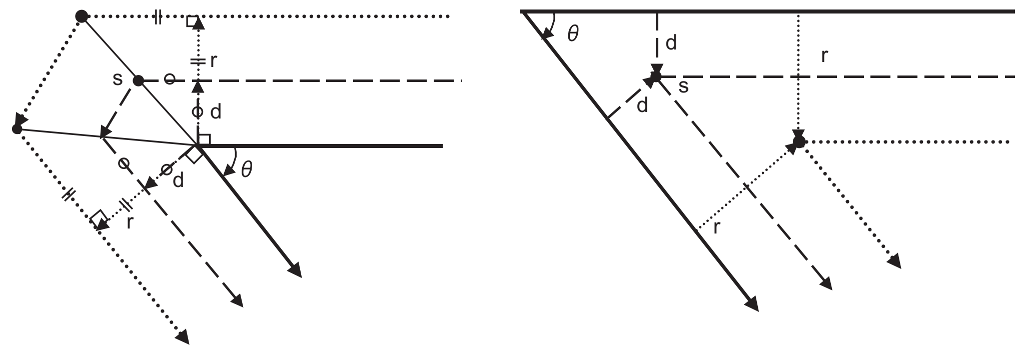
Operation example

Operations when designating the compensation amount with diameter

When the tool radius compensation amount D=10.0 is commanded, tool radius compensation amount "d" is 5.0 if the parameter "#8117" is ON (set to "1").
(Tool radius compensation amount "r" is 10.0 if the parameter "#8117" is OFF (set to "0").)



(1) Linear -> linear corner (acute angle)



Outside of the corner

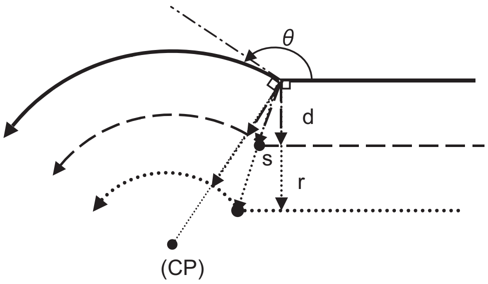
Inside of the corner

Program path

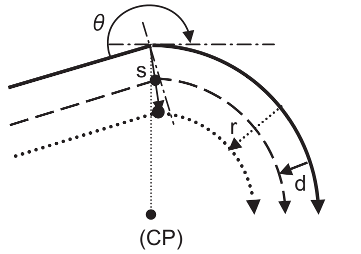
Tool center path (When #8117 is ON)

Tool center path (When #8117 is OFF)

(2) Linear -> arc (obtuse angle)

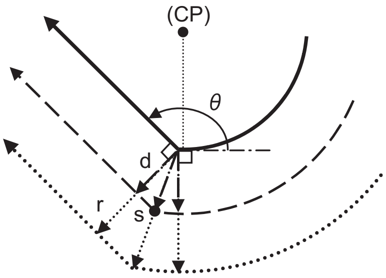


Outside of the corner

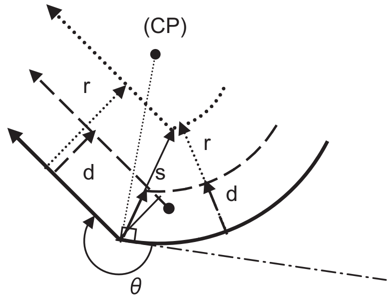


Inside of the corner

(3) Arc -> linear (obtuse angle)



Outside of the corner



Inside of the corner

———	Program path
- - - -	Tool center path (When #8117 is ON)
········	Tool center path (When #8117 is OFF)
(CP)	Arc center

Restrictions

(1) If tool radius compensation amount has already been set, the compensation amount is not be changed even if the parameter "8117" is changed.

(2) Make sure not to change the parameter #8117 during the compensation. When the parameter is changed using parameter input by program function, the program error (P421) will occur.

(3) If the parameter #8117 is set to ON with the parameter "#1037 cmdtyp" set to “2”, the tool radius wear data is also regarded as the diameter compensation amount, thus, it will be converted to the radius value and compensation will be performed.

(4) Diameter designation of tool radius compensation amount can be used for the tool life management data.

(5) There is no effect by #8117 on the tool radius measurement function.

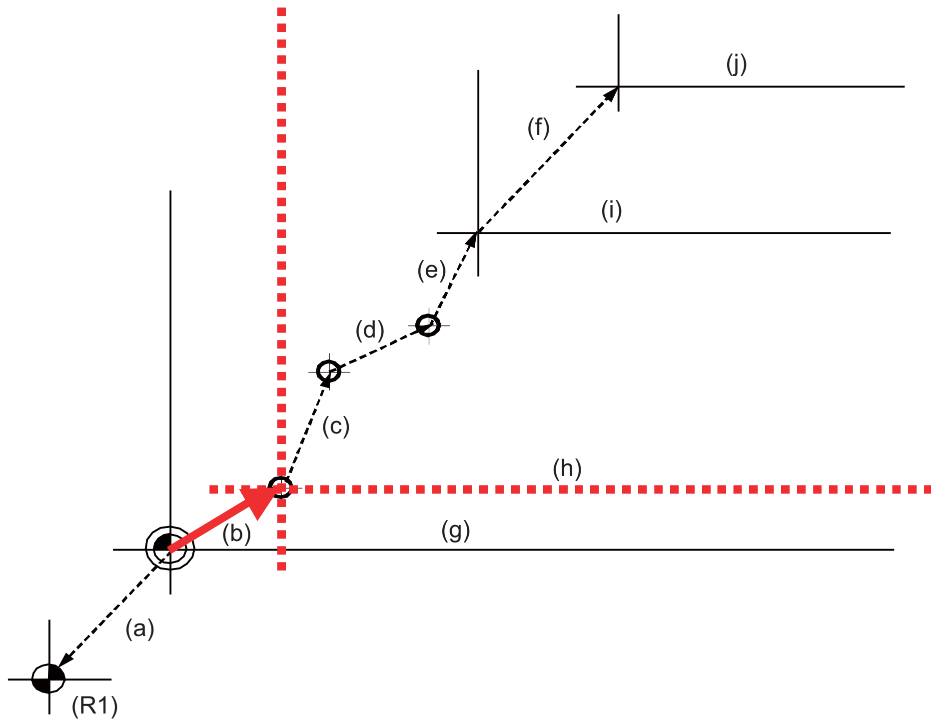
12.3.10 Workpiece Coordinate Changing during Radius Compensation

Function and purpose

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

Detailed description

When the parameter is "0", the tool radius compensation is calculated based on the position on the workpiece coordinate system.
When the parameter is "1", the tool radius compensation is calculated based on the position on the program coordinate system.
The program coordinate systems are defined as shown in the figure below.



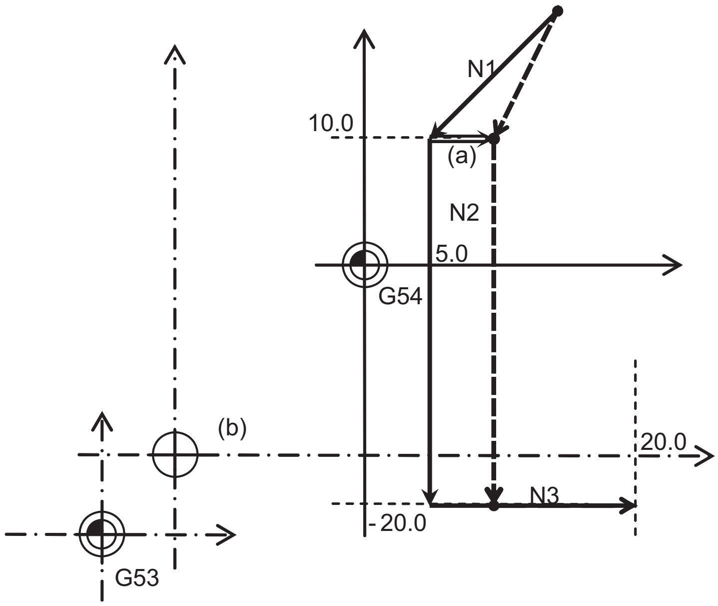
(R1) 1st reference position
(a) 1st reference position offset
(b) Interrupt amount offset
(c) Extended workpiece coordinate system offset
(d) G92 offset
(e) Workpiece coordinate system offset
(f) Local coordinate system offset
(g) G53 Basic machine coordinate system
(h) Program coordinate system
(i) G54 to G59/G54.1Pn Workpiece coordinate system/Extended workpiece coordinate system
(j) G52Local coordinate system

The coordinate system changed by parameter is as follows.

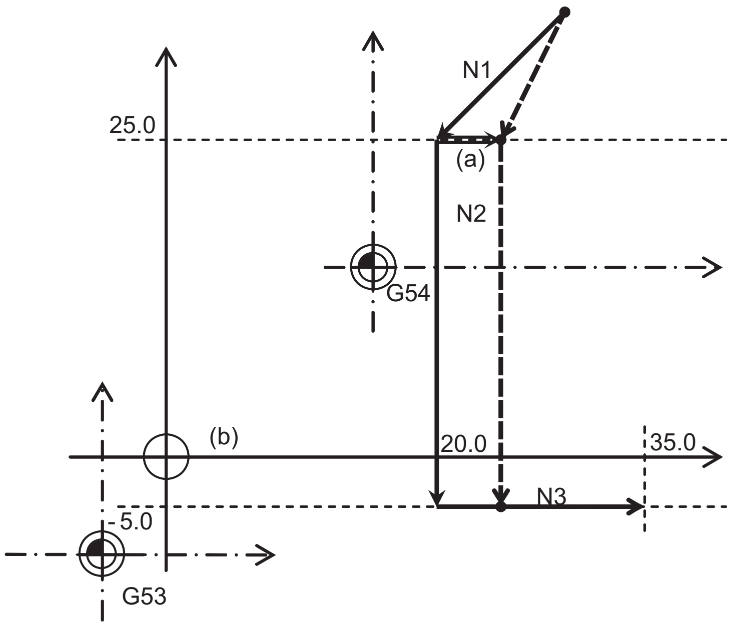
G90 G54 G00 X15. Y20.;
N1 G41 D3 X5. Y10. ;
N2 G01; Y-20 F1000;
N3 G40 X20. ;
M30 ;

D3 =5.000
G54 offset
X15.000
Y15.000

(1) Parameter = 0



(2) Parameter = 1

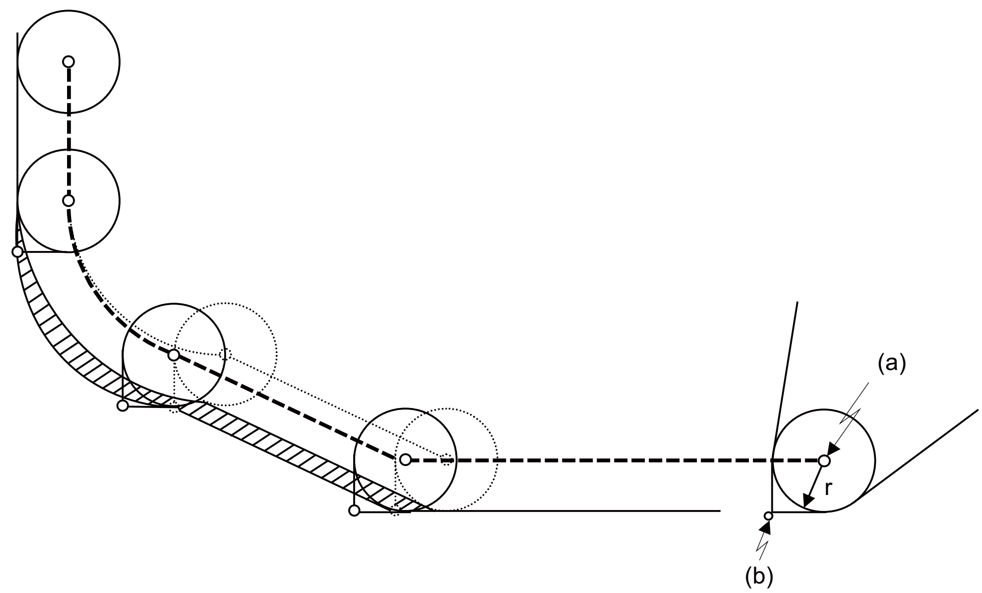


————	Program path	(a)	Compensation vector
- - - -	Tool center path	(b)	Program coordinate system

12.4 Tool Nose Radius Compensation (for Machining Center System)

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 (nose R compensation) compensation is a function for automatically calculating and offsetting this error by setting the tool nose radius (cutter radius) value.
The validity of this function depends on the MTB specifications. (The tool nose must be set to "1" to "8" in the parameter "#1037 cmdtyp".)
If the tool nose is set to "0" or "9", tool radius compensation is carried out.
When G46 is commanded, the tool position offset reduction function is enabled. (The automatic direction identification mode is not available.)
Refer to "Programming Manual Lathe System" (IB-1501275, IB-1501276) for details of the tool nose radius compensation.



(a) Tool nose center (b) Hypothetical tool nose point (r) Tool nose radius

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

Tool nose center path with nose R compensation

Command format

G40;	**Tool nose radius compensation cancel**

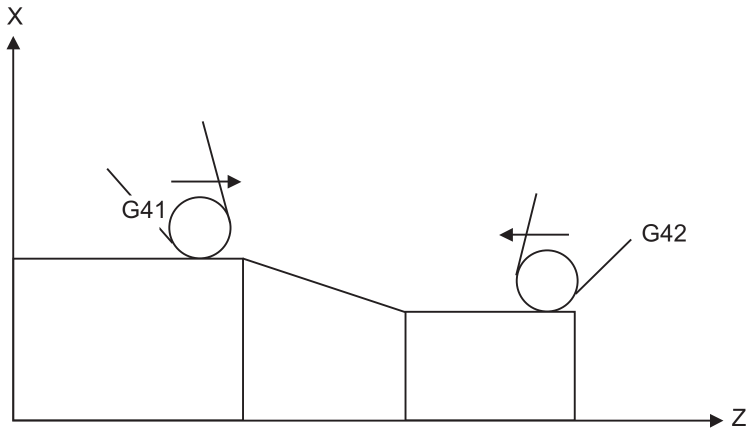
G41 (X__ Z__ D__);	**Tool nose radius compensation left**

G42 (X__ Z__ D__);	**Tool nose radius compensation right**

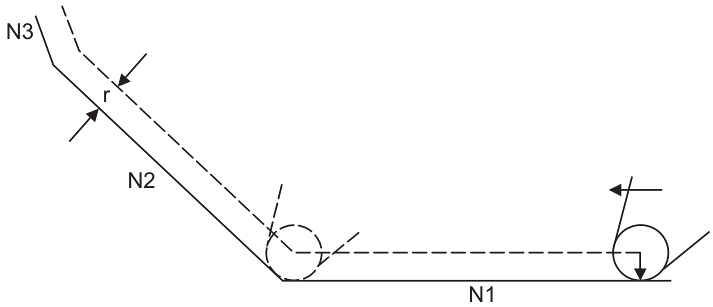
X	X axis end point coordinate (Absolute value of workpiece coordinate system)
Z	Z axis end point coordinate (Absolute value of workpiece coordinate system)
D	Compensation No. (The compensation No. setting range will differ according to the specifications (No. of compensation sets).)

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.
G40 cancels the tool nose radius compensation mode.



(2) Nose R 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 R.
In the figure below, "r" is the tool nose radius compensation amount (nose R).
The nose R 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 also valid for fixed cycle.

(5) Compensation mode will be temporarily canceled in 1 block before the thread cutting command block.

(6) The compensation plane, movement axes and next advance direction vector follow the plane selection command designated by G17, G18 or G19.

G17	XY plane X,Y,I,J
G18	ZX plane Z,X,K,I
G19	YZ plane Y,Z,J,K

Precautions

Assigning the compensation amounts

(1) For nose radius compensation for machining center system, the compensation amount should always be specified with radius value.

(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.

Corner judgment method

(1) The criterion to execute the outer rounding at the small corner in tool radius compensation depends on the MTB specifications (parameter "#1289 ext25/bit0").

Designating the tool nose point

(1) There two methods to set the tool nose point for machining center system. (This depends on the MTB specifications.)
- Set a value in the system variable #23000+n ("n" corresponds to the compensation number) using the machining program.
- Set the parameter "#1046 T-ofs disp type" to change the compensation type to III, then, set a value in the tool compensation amount screen.

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 (when a plane different from the one used during the compensation is commanded (P112))
G31 (P608)
G74,G75,G76 (P155)
G81 to G89(P155)

(2) A program error will occur when a circular command is issued in the first or last block of the tool nose radius compensation. (P151)

(3) A program error will occur during tool nose radius compensation when the intersection point of single block skip in the interference block processing cannot be calculated. (P152)

(4) A program error will occur when there is an error in one of the pre-read blocks during tool nose radius compensation.

(5) A program error will occur when an interference occurs under no interference avoidance conditions during tool nose radius compensation. (P153)

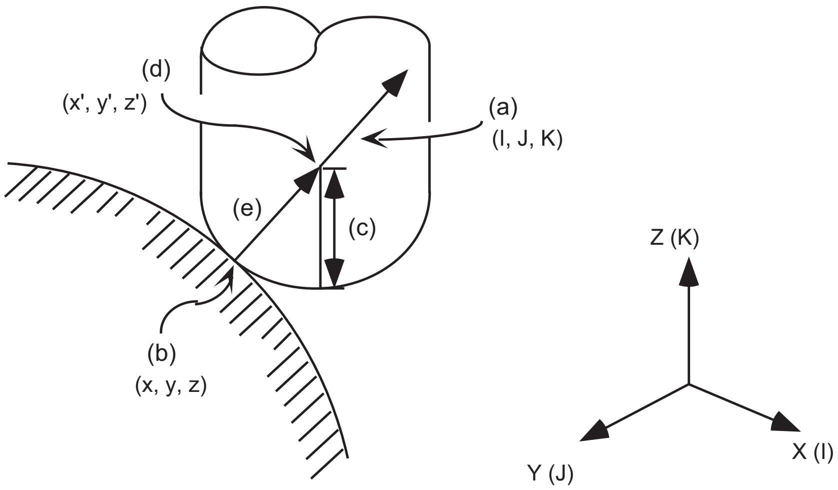
(6) A program error will occur when a tool nose radius compensation command is issued even though the tool nose radius compensation specification is not provided. (P150)

(7) If a tool nose radius compensation command is issued in mirror image, a program error will occur. (P803)

12.5 3-dimensional Tool Radius Compensation ; G40/G41,G42

Function and purpose

The 3-dimentional tool radius compensation compensates for the tool in a 3-dimensional space following the commanded three-dimensional vectors.



As shown above, the tool is moved in the tool center coordinate position (x', y', z') (d) which is compensated for by the tool radius "r" (c) in respect to the program coordinate position (x, y, z) (b) following the plane normal line vector (I, J, K) (a).

Though two-dimensional tool radius compensation creates the vectors at a right angle to the (I, J, K) direction, three-dimensional tool radius compensation creates the vector in the (I, J, K) direction. (The vector is created at the end point of the block.)

The three-dimensional compensation vector (compensation) (e) axis elements are as below.

$$H_x = \frac{I}{\sqrt{(I^2 + J^2 + K^2)}} \times r$$

$$H_Y = \frac{J}{\sqrt{(I^2 + J^2 + K^2)}} \times r$$

$$H_Z = \frac{K}{\sqrt{(I^2 + J^2 + K^2)}} \times r$$

Thus, the tool center coordinate position (x', y', z') (d) is each expressed as below. Note that (x, y, z) are the program coordinate position.

$x' = x + Hx$
$y' = y + Hy$
$z' = z + Hz$

Note

(1) Three-dimensional compensation vector (Hx, Hy, Hz) refers to the plane normal line vector whose direction is same as the plane normal line vector (I, J, K) and the size equals to the tool radius "r".

(2) When the machining parameter "#8071 3-D CMP" is set to a value other than "0", $\sqrt{(I^2 + J^2 + K^2)}$ the value of "#8071 3-D CMP" will be used as the value. (Refer to the s Setup Manual for details.)

Command format

3-dimensional tool radius compensation start

G41(G42) X__ Y__ Z__ I__ J__ K__ D__ ;

New plane normal line vector is commanded in the compensation mode.

X__ Y__ Z__ I__ J__ K__;

3-dimentional tool radius compensation cancel

G40; (or D00;)

G40 X__ Y__ Z__; (or X__ Y__ Z__ D00;)

G41	Three-dimensional tool radius compensation command (+ direction)
G42	Three-dimensional tool radius compensation command (- direction)
G40	Three-dimensional tool radius compensation cancel command
X, Y, Z	Movement axis command compensation space
I, J, K	Plane normal line vector
D	Compensation No. (Note that when "D00" is issued, three dimensional tool radius compensation will be canceled even if G40 is not commanded.)

Command, for all three axes, the compensation No. D and plane normal line vector (I, J, K) in the same block as the three-dimensional tool radius compensation command G41 (G42).

If only one or two axes are commanded, the normal tool radius compensation mode will be applied. (When the command value for I, J, K is set to "0", this command is valid.)

G Code	Compensation amount:		D00
	+	-	
G40	Cancel	Cancel	Cancel
G41	I, J, K direction	Reverse direction of I, J, K	Cancel
G42	Reverse direction of I, J, K	I, J, K direction	Cancel

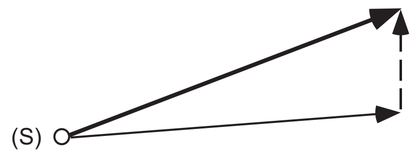
Detailed description

The compensation space is determined by the axis address commands (X, Y, Z, U, V, W) of the block where the three-dimensional tool radius compensation starts.	(Example) G17 ; G41 Xx Yy Zz Ii Jj Kk ;	X Y Z space
	G17 ; G41 Yy Ii Jj Kk ;	X Y Z space
Here, U, V and W are each the additional axes for the X, Y and Z axis. If the X axis and U axis (Y and V, Z and W) are commanded simultaneously in the three-dimensional tool radius compensation start block, the currently commanded plane selection axis will have the priority. If the axis address is not commanded, it will be interpreted that the X, Y and Z axes are commanded for the coordinate axes.	G17 V ; G41 Xx Vv Zz Ii Jj Kk ;	X V Z space
	G17 W ; G41 Ww Ii Jj Kk ;	X Y W space
	G17 ; G41 Xx Yy Zz Ww Ii Jj Kk ;	X Y Z space
	G17 W ; G41 Xx Yy Zz Ww Ii Jj Kk ;	X Y W space
	G17 ; G41 Ii Jj Kk ;	X Y Z space
	G17 U ; G41 Ii Jj Kk ;	U Y Z space

Operation example

Compensation start (1) When there is a movement command

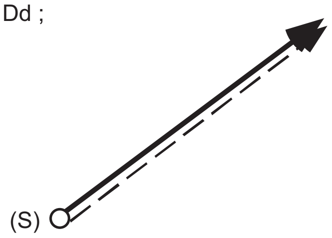
G41 Xx Yy Zz Ii Jj Kk Dd ;



(S) Start point
Tool center path
Programmed path
Three-dimensional compensation vector

Compensation start : When there is no movement command

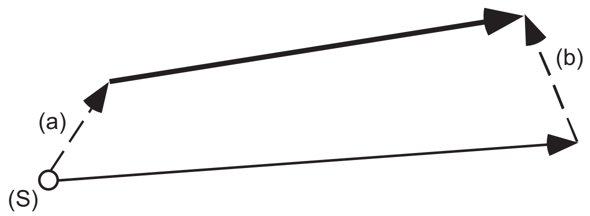
G41 Ii Jj Kk Dd ;



(S) Start point
Tool center path
Three-dimensional compensation vector

Movement during the compensation (3) When there is a movement command and a plane normal line vector command

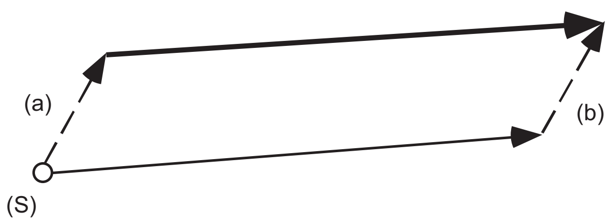
Xx Yy Zz Ii Jj Kk ;



(S) Start point
(a) Old vector
(b) New vector
Tool center path
Programmed path

Movement during the compensation (4) When there is no plane normal line vector command

Xx Yy Zz Ii Jj Kk ;



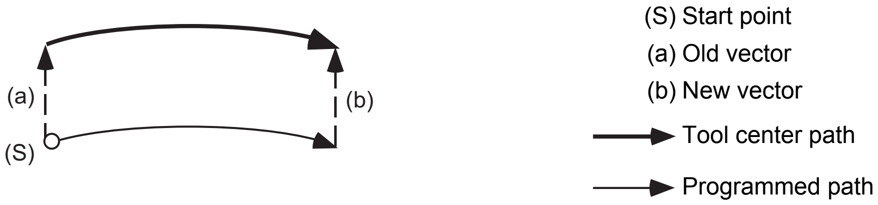
(S) Start point
(a) Old vector
(b) New vector
Tool center path
Programmed path

Movement during the compensation (5) For arc or helical cutting

The I, J, K commands for a circular or helical cutting are regarded as the circular center commands, thus, the new vector is equivalent to the old vector.

Even for the R-designation method, commanded I, J, K addresses will be ignored, then the new vector will be equivalent to the old vector.

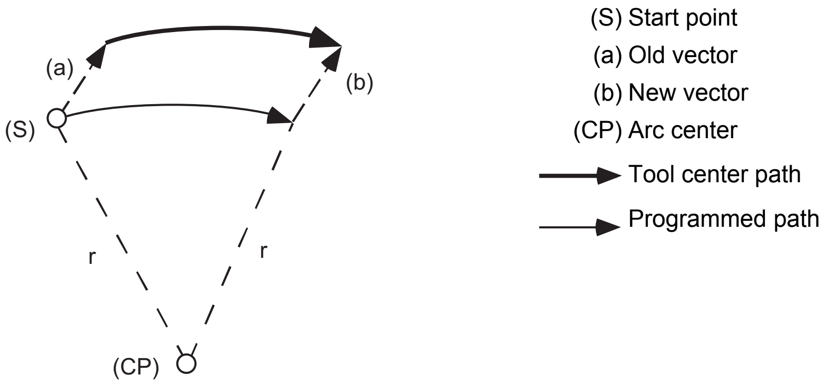
G02 Xx Yy (Zz) Ii Jj ; I, J(K) means the circular center
Or
G02 Xx Yy (Zz) Rr ; R-designated arc



Note

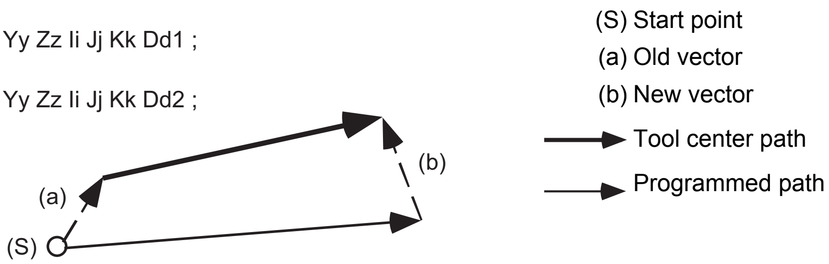
(1) The center coordinate will not shift during the circular or helical cutting. Thus, when I, J, K are commanded with the vector as below, the program error (P70) will occur.

G02 Xx Yy (Zz) Ii Jj ; I, J(K) means the circular center
Or
G02 Xx Yy (Zz) Rr ; R-designated arc



Movement during the tool radius compensation (6) When compensation amount is to be changed

G41 Xx Yy Zz Ii Jj Kk Dd1 ;
:
G41 Xx Yy Zz Ii Jj Kk Dd2 ;

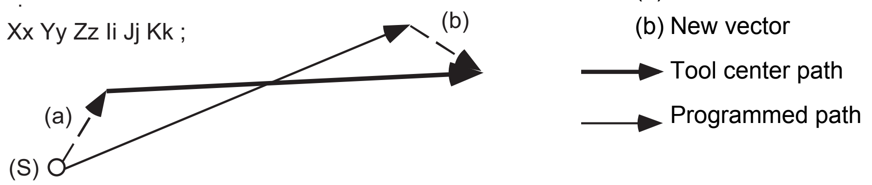


Note

(1) If I, J, K are not commanded in a block where the compensation amount is to be changed, the vector will be equivalent to the old vector. In this case, the modal will change, however, the compensation amount will change when I, J, K are commanded.

Movement during the tool radius compensation (7) When compensation direction is to be changed

G41 Xx Yy Zz Ii Jj Kk Dd1 ;
:
G42 Xx Yy Zz Ii Jj Kk ;



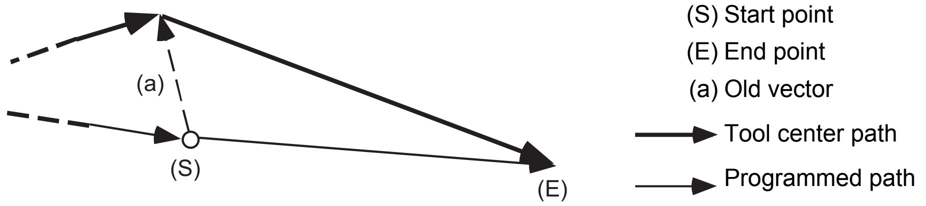
Note

(1) If I, J and K are not commanded in a block where the compensation direction is to be changed, the vector will be equivalent to the old vector and the compensation direction will not be changed. In this case, the modal will change, however, the compensation direction will change when I, J and K are commanded.

(2) If the compensation direction is changed in an arc (G02/G03) block, I, J will be the center of the arc, thus, the compensation direction will not change.
Even for the R-designation method, commanded I, J and K will be ignored, and the compensation direction cannot be changed.

Movement during the tool radius compensation : When there is a movement command

G40 Xx Yy Zz ; (or Xx Yy Zz D00 ;)



Tool radius compensation cancel: When there is no movement command

G40; (or D00;)

Relationship with other functions

Normal tool radius compensation

If the plane normal line vector (I, J, K) is not commanded for all three axes in the three-dimensional tool radius compensation start block, the normal tool radius compensation mode will take place.
If G41 (G42) is commanded without commanding the plane normal line vector during three-dimensional tool radius compensation, the modal will change, however, the old vector will be used.
If G41 (G42) with the plane normal line vector is commanded during tool radius compensation, this command will be ignored and the normal tool radius compensation will take place.

Tool length compensation

Tool length compensation is applied to the coordinate after three-dimensional tool radius compensation.

Tool position offset

Tool position offset is applied to the coordinate after three-dimensional tool radius compensation.

Fixed cycle

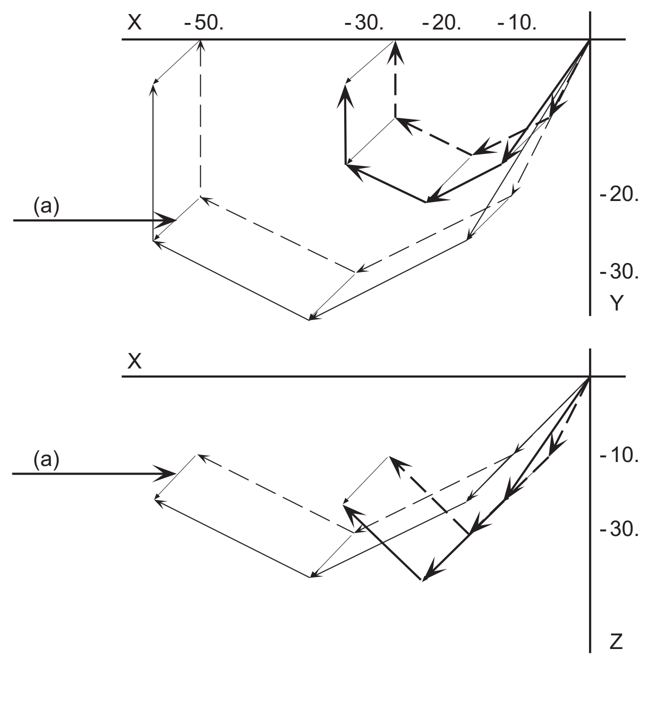
The program error (P155) will occur.

Scaling

Scaling is applied to the coordinate before three-dimensional tool radius compensation.
Scaling is not applied to the plane normal line vector (I, J, K).

```
D1=10.

G90;
G51 X0 Y0 P0.5 ;
N1 G41 D1 X-10. Y-20. Z-10. I-5. J-5. K-5. ;
N2 X-30. Y-30. Z-20. ;
N3 X-50. Y-20. Z-10. ;
N4 Y0. ;


N1(  -5.000, -10.000, -10.000 )
N1( -10.773, -15.773, -15.773 )

N2( -15.000, -15.000, -20.000 )
N2( -20.773, -20.773, -25.773 )

N3( -25.000, -10.000, -10.000 )
N3( -30.773, -15.773, -15.773 )

N4(  -25.000,  0.000,  -10.000 )
N4(  -30.773,  -5.773,  -15.773 )
```

* Upper: Program position after scaling
  Lower: Position after scaling and compensation



(a) Plane normal line vector

— — –Program path

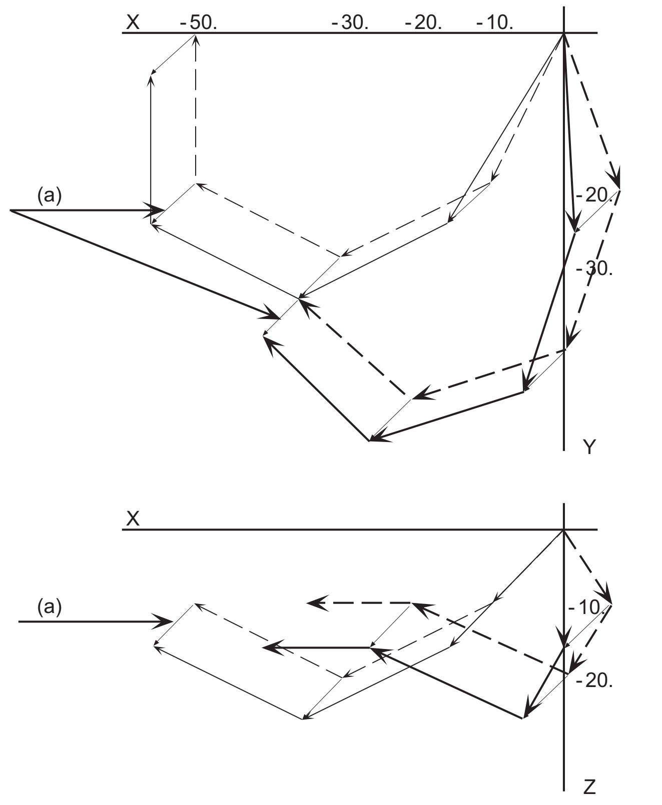
———Program path after compensation

– – – –Program path after scaling

———Program path after scaling and compensation

Coordinate rotation by program

Program coordinate rotation is applied to the coordinate before three-dimensional tool radius compensation.
The plane normal line vector (I, J, K) will not rotate.

D1=10. G90; G68 X0 Y0 R45. ; N1 G41 D1 X-10. Y-20. Z-10. I-5. J-5. K-5. ; N2 X-30. Y-30. Z-20. ; N3 X-50. Y-20. Z-10. ; N4 Y0. ; N1(7.071, -21.213, -10.000) N1(7.071, -29.378, -15.773) N2(0.000, -42.426, -20.000) N2(0.000, -50.591, -25.773) N3(-21.213, -49.497, -10.000) N3(-21.213, -57.662, -15.773) N4(-35.355, -35.355, -10.000) N4(-35.355, -43.520, -15.773) * Upper: Program position after coordinate rotation Lower: Position after coordinate rotation and compensation	

(a) Plane normal line vector

— — –Program path

———Program path after compensation

– – – –Program path after coordinate rotation

———Program path after coordinate rotation and compensation

Coordinate rotation by parameter

Parameter coordinate rotation is applied to the coordinates after three-dimensional tool radius compensation.
The plane normal line vector (I, J, K) rotates.

Mirror image

Mirror image is applied to the coordinates after three-dimensional tool radius compensation.
Mirror image is applied to the plane normal line vector (I, J, K).

Skip

The program error (P608) will occur.

Reference position check

The compensation amount will not be canceled. Thus, if this is commanded during three-dimensional tool radius compensation, the path will be deviated by the compensation amount, thus the program error (P434) will occur.

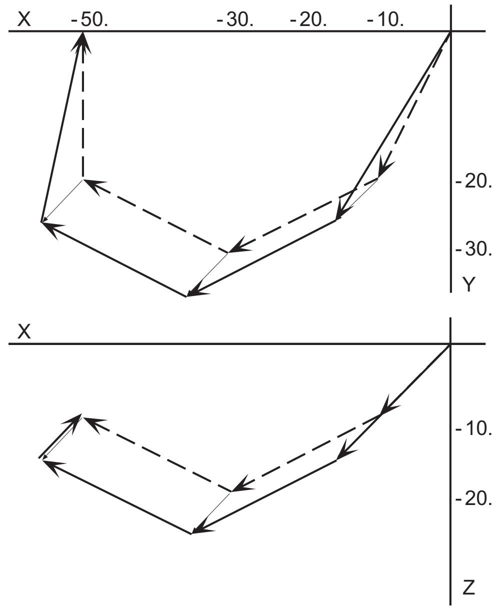
Automatic corner override

Automatic corner override is invalid during three-dimensional tool radius compensation.

Machine coordinate system selection

(1) For the absolute command, all axes will be temporarily canceled at the commanded coordinate position.

```
D1=10.

G90;
N1 G41 D1 X-10. Y-20. Z-10. I-5. J-5. K-5. ;
N2 X-30. Y-30. Z-20. ;
N3 X-50. Y-20. Z-10. ;
N4 G53 Y0 ;

N1( -10.000, -20.000, -10.000 )
N1( -15.773, -25.773, -15.773 )

N2( -30.000, -30.000, -20.000 )
N2( -35.773, -35.773, -25.773 )

N3( -50.000, -20.000, -10.000 )
N3( -55.773, -25.773, -15.773 )

N4(  -50.000,  0.000,  -10.000 )
N4(  -50.000,  0.000,  -10.000 )

* Upper: Program position
  Lower: Program position after compensation
```

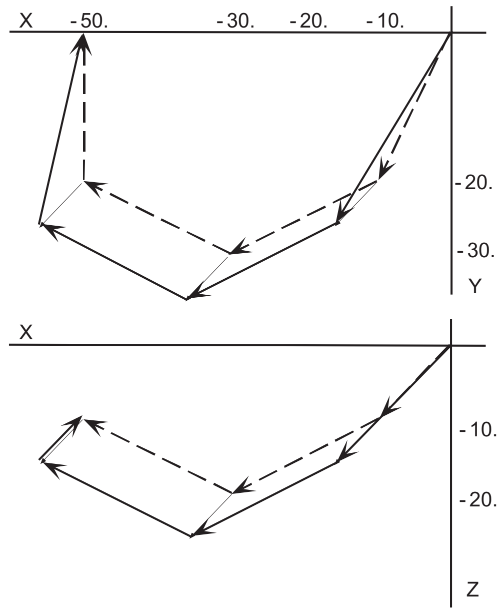


－－－－Program path

———Program path after compensation

(2) For the incremental command, the axis will move by the amount obtained by subtracting each axis vector from the incremental movement amount.
(The compensation amount is temporarily canceled.)

```
D1=10.

G91;
N1 G41 D1 X-10. Y-20. Z-10. I-5. J-5. K-5. ;
N2 X-20. Y-10. Z-10. ;
N3 X-20. Y10. Z10. ;
N4 G90 G53 Y0.;

N1( -10.000, -20.000, -10.000 )
N1( -15.773, -25.773, -15.773 )

N2( -30.000, -30.000, -20.000 )
N2( -35.773, -35.773, -25.773 )

N3( -50.000, -20.000, -10.000 )
N3( -55.773, -25.773, -15.773 )

N4(  -50.000,  0.000,  -10.000 )
N4(  -50.000,  0.000,  -10.000 )

* Upper: Program position
  Lower: Program position after compensation
```

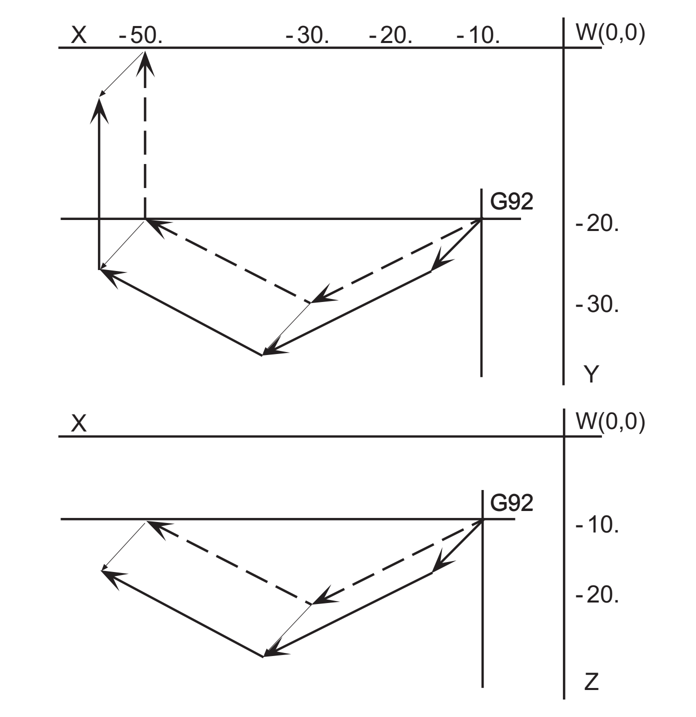


－－－－Program path

———Program path after compensation

Coordinate system setting

When commanded in the same block as the coordinate system setting, the coordinate system will be set, and operation will start up independently with the plane normal line vector (I, J, K).

D1=10. G91; N1 G92 G41 D1 X-10. Y-20. Z-10. I-5. J-5. K-5. ; N2 X-20. Y-10. Z-10. ; N3 X-30. Y-10. Z10. ; N4 Y20. ; N1(-10.000, -20.000, -10.000) N1(-10.000, -20.000, -10.000) N2(-30.000, -30.000, -20.000) N2(-35.773, -35.773, -25.773) N3(-50.000, -20.000, -10.000) N3(-55.773, -25.773, -15.773) N4(-50.000, 0.000, -10.000) N4(-55.773, -5.773, -15.773) * Upper: Program position Lower: Program position after compensation	

– – – – Program path

——— Program path after compensation

Reference position return completed

All the axes will be temporarily canceled at the intermediate point.

D1=10.

G91;
N1 G41 D1 X-10. Y-20. Z-10. I-5. J-5. K-5. ;
N2 X-20. Y-10. Z-10. ;
N3 X-20. Y10. Z10. ;
N4 G28 X0 Y0 Z0 ;

N1(-10.000, -20.000, -10.000)
N1(-15.773, -25.773, -15.773)

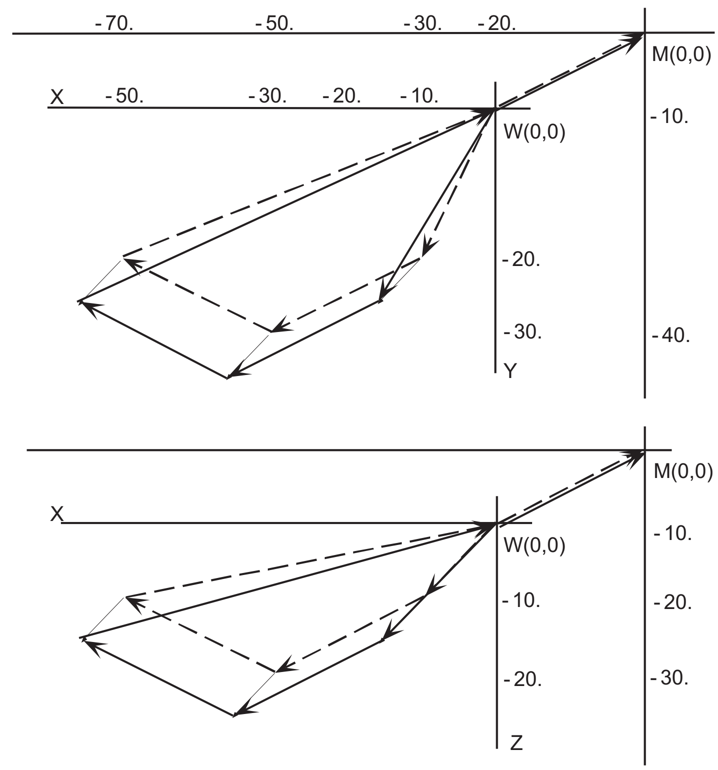
N2(-30.000, -30.000, -20.000)
N2(-35.773, -35.773, -25.773)

N3(-50.000, -20.000, -10.000)
N3(-55.773, -25.773, -15.773)

N4(0.000, 0.000, 0.000)
N4(0.000, 0.000, 0.000)

N4(20.000, 10.000, 10.000)
N4(20.000, 10.000, 10.000)

* Upper: Workpiece coordinate position
Lower: Program position after compensation



– – – – Program path

——— Program path after compensation

NC reset

Three-dimensional tool radius compensation will be canceled if NC reset is executed during three-dimensional tool radius compensation.

Emergency stop

Three-dimensional tool radius compensation will be canceled by the emergency stop or emergency stop cancel during three-dimensional tool radius compensation.

Restrictions

(1) The compensation No. is selected with the D address, however, the D address is valid only when G41 or G42 is commanded. If D is not commanded, the number of the previous D address will be valid.

(2) Switch the mode to the compensation mode in the G00 or G01 mode. When changed during the arc mode, the program error (P150) will occur.
The compensation direction and compensation amount after the mode change will become valid from the block where I, J and K are commanded in the G00 or G01 mode. If three-dimensional tool radius compensation is commanded in a block not containing the plane normal line vector (I, J, K) during the arc mode, only the modal information will be changed.
The plane normal line vector will be validated from the block where I, J and K are commanded next.

(3) During the 3-dimensional tool radius compensation mode in a certain space, it is not possible to switch the space to another one and to execute three-dimensional tool radius compensation. To switch the compensation space, always cancel the compensation mode with G40 or D00 first.
(Example)

G41 Xx Yy Zz Ii Jj Kk ;	Compensation starts in X, Y, Z space.
:	:
:	:
G41 Uu Yy Zz Ii Jj Kk ;	Compensation is carried out in X, Y, Z space, and U axis moves by commanded value.

(4) If the compensation No. D is other than the range of 1 to 40 with the standard specifications or 1 to 800 (max.) with the additional specifications, the program error (P170) will occur.

(5) Only the G40 and D00 commands can be used to cancel 3-dimensional tool radius compensation.

(6) If the size ($I^2+J^2+K^2$) of the vector commanded with I, J and K overflows, the program error (P35) will occur.

12.6 Tool Position Offset ; G45 to G48

Function and purpose

Using the G45 to G46 commands, the movement distance of the axes specified in the same block can be extended or reduced by a preset compensation length.

Furthermore, the compensation amount can be similarly doubled (x 2 expansion) or halved (x 2 reduction) with commands G47 and G48.

The number of sets for the compensation differ according to machine specification. Refer to Specifications Manual.

D01 to Dn

(The numbers given are the total number of sets for tool length compensation, tool position compensation and tool radius compensation.)

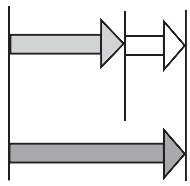
Start point End point

G45 command

Expansion by compensation amount only

Internal arithmetic processing

Movement amount

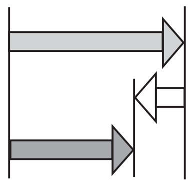


G46 command

Reduction by compensation amount only

Internal arithmetic processing

Movement amount

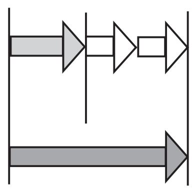


G47 command

2 expansion by compensation amount

Internal arithmetic processing

Movement amount

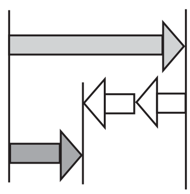


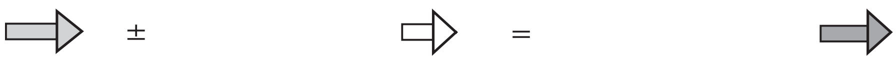
G48 command

2 reduction by compensation amount

Internal arithmetic processing

Movement amount





(Program command value) (compensation amount) (Movement amount after compensation)

Command format

G45 X__ Y__ Z__ D__ ; ... Expansion of movement amount by compensation amount set in compensation memory

G46 X__ Y__ Z__ D__ ; ... Reduction of movement amount by compensation amount set in compensation memory

G47 X__ Y__ Z__ D__ ; ... Expansion of movement amount by double the compensation amount set in compensation memory

G48 X__ Y__ Z__ D__ ; ... Reduction of movement amount by double the compensation amount set in compensation memory

X, Y, Z	Movement amount of each axis
D	Tool compensation No.

Detailed description

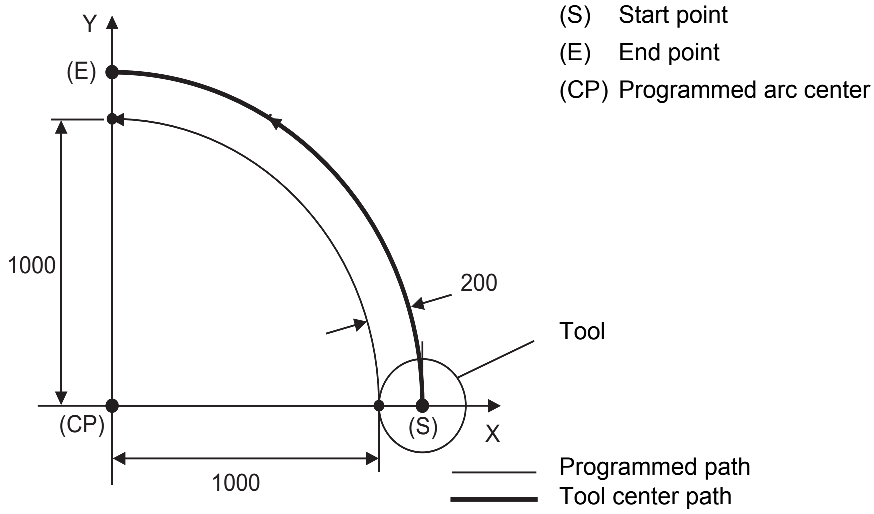
Details for incremental values are given below.

Command	Movement amount of equivalent command (assigned compensation amount = I)	Example (when X = 1000)
G45 Xx Dd	X(x+I)	I= 10 X= 1010 I= -10 X= 990
G45 X-x Dd	X-(x+I)	I= 10 X= -1010 I= -10 X= -990
G46 Xx Dd	X(x-I)	I= 10 X= 990 I= -10 X= 1010
G46 X-x Dd	X-(x-I)	I= 10 X= -990 I= -10 X= -1010
G47 Xx Dd	X(x+2*I)	I= 10 X= 1020 I= -10 X= 980
G47 X-x Dd	X-(x+2*I)	I= 10 X= -1020 I= -10 X= -980
G48 Xx Dd	X(x-2*I)	I= 10 X= 980 I= -10 X= 1020
G48 X-x Dd	X-(x-2*I)	I= 10 X= -980 I= -10 X= -1020

Program example

(Example 1)

Tool position compensation with 1/4 arc command



It is assumed that compensation has already been provided in the + X direction by D01 = 200.

```
G91 G45 G03 X -1000 Y1000 I -1000 F1000 D01 ;
```

Even if the compensation numbers are not assigned in the same block as the G45 to G48 commands, compensation is provided with the tool position compensation number previously stored in the memory.

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

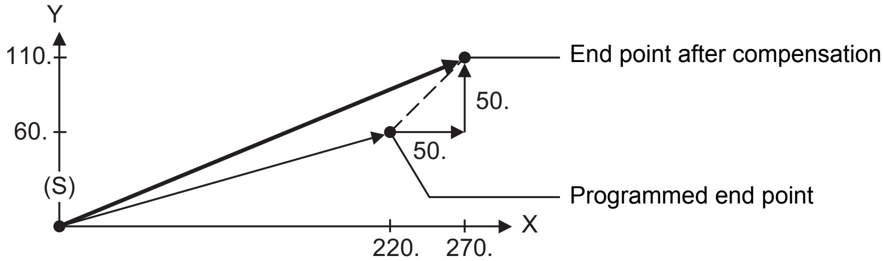
These G codes are unmodal and are effective only in the command block.

Even with an absolute value command, the amount of the movement is extended or reduced for each axis with respect to the direction of movement from the end point of the preceding block to the position assigned by the G45 to G48 block.

In other words, even for an absolute value command, compensation can be applied to movement amounts (incremental values) in the same block.

When a command for "n" number of simultaneous axes is given, the same compensation will be applied to all axes. It is valid even for the additional axes. (but it must be within the range of the number of axes that can be controlled simultaneously.)

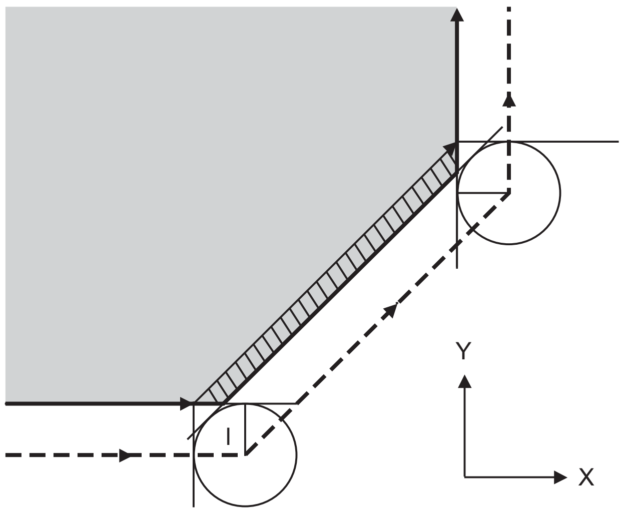
G01 G45 X220. Y60. D20 ;
(D20) = +50. 000



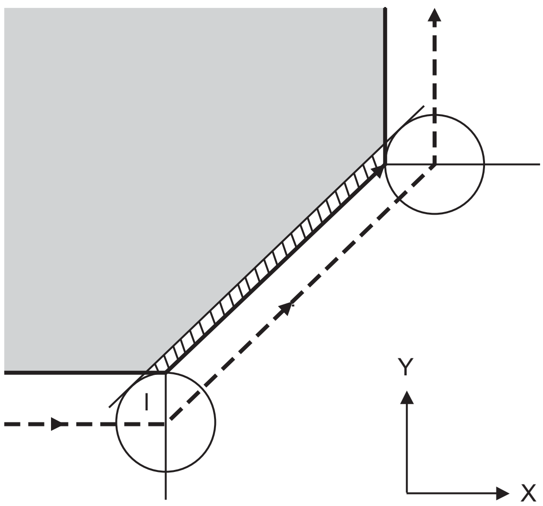
(S) Start point

Note

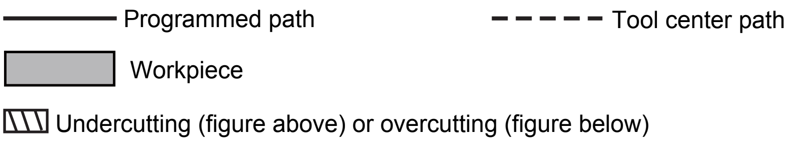
(1) If compensation is applied to two axes, over-cutting or under-cutting will result, as shown in the figures below. In cases like this, use the cutter compensation commands (G40 to G42).



G01 G45 Xx 1 Dd1 ;
Xx2 Yy 2 ;
G45 Yy3 ;

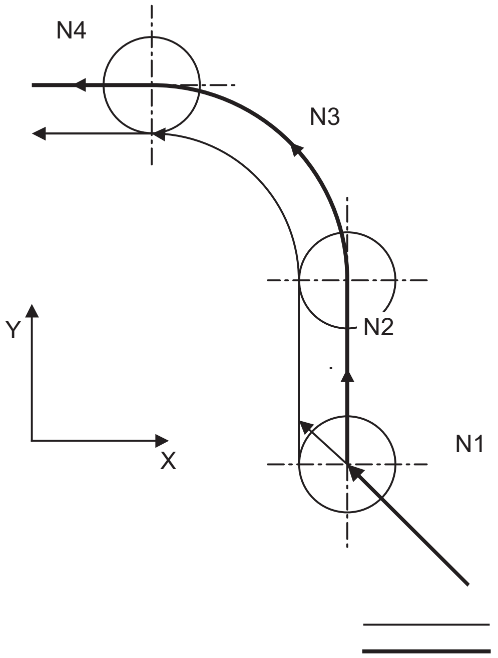


G01 Xx1 ;
G45 Xx2 Yy2 Dd 2 ;
Yy3 ;



I = Compensation amount setting

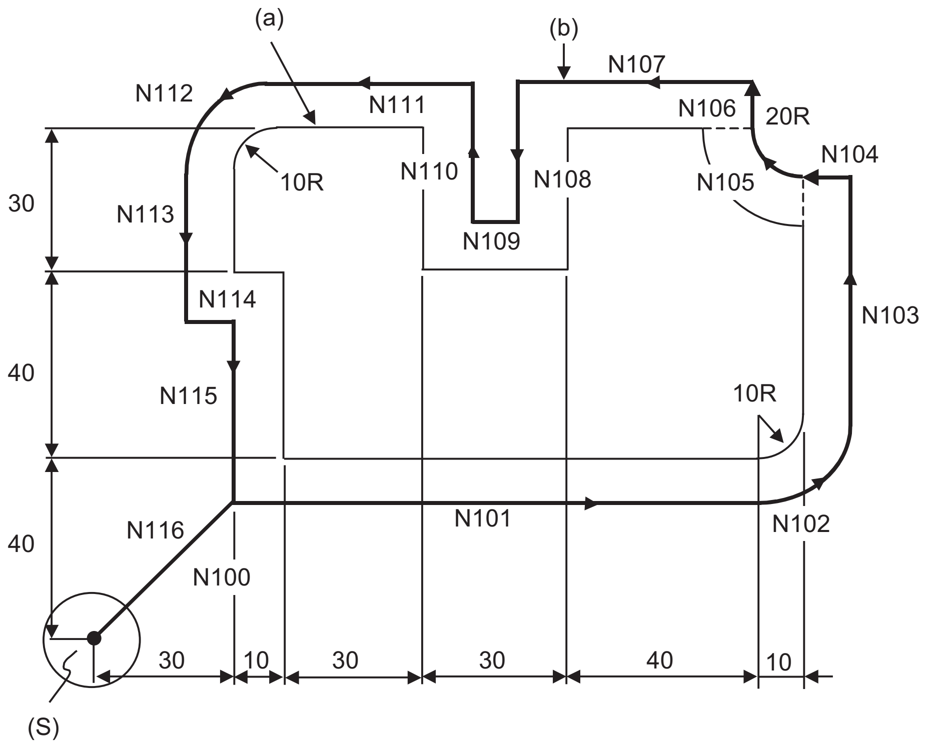
(Example 2)



N1 G46 G00 Xx1 Yy1 Dd1 ;
N2 G45 G01 Yy2 Ff2 ;
N3 G45 G03 Xx3 Yy3 Ii3 ;
N4 G01 Xx4 ;

(Example 3)

When the G45 to G48 command is assigned, the compensation amount for each pass is the amount assigned by the compensation number, and the tool does not move for the difference from the previous compensation as it would do with the tool length compensation command (G43).



(S) Start point
(a) Programmed path
(b) Tool center path

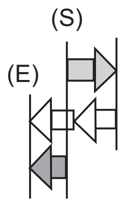
Compensation amount D01 = 10.000mm (Compensation amount of tool radius)

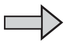
N100	G91	G46 G00 X40.0 Y40.0 D01 ;
N101	G45	G01 X100.0 F200 ;
N102	G45	G03 X10.0 Y10.0 J10.0 ;
N103	G45	G01 Y40.0 ;
N104	G46	X0 ;
N105	G46	G02 X-20.0 Y20.0 J20.0 ;
N106	G45	G01 Y0 ;
N107	G47	X-30.0 ;
N108		Y-30.0 ;
N109	G48	X-30.0 ;
N110		Y30.0 ;
N111	G45	X-30.0 ;
N112	G45	G03 X-10.0 Y-10.0 J-10.0 ;
N113	G45	G01 Y-20.0 ;
N114		X10.0 ;
N115		Y-40.0 ;
N116	G46	X-40.0 Y-40.0 ;
N117	M02	;
%		

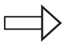
Precautions

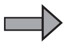
(1) These commands should be used when operation is not in a fixed cycle mode.
(They are ignored even if they are assigned during a fixed cycle.)

(2) As a result of the internal arithmetic processing based on the expansion or reduction, the tool will proceed to move in the opposite direction when the command direction is reversed.




Program command G48 X20.000 D01 ;


Compensation Compensation amount = +15.000


Tool movement Actual movement = X - 10.000

(S) Start point
(E) End point

(3) When a zero movement amount has been specified in the incremental value command (G91) mode, the result is as follows.

Compensation No. : D01
Compensation amount corresponding to D01 : 1234

NC command	G45 X0 D01 ;	G45 X-0 D01 ;	G46 X0 D01 ;	G46 X-0 D01;
Equivalent command	X1234;	X -1234 ;	X -1234 ;	X1234;

When a zero movement amount has been specified with an absolute value command, the operation is completed immediately and the tool does not move for the compensation amount.

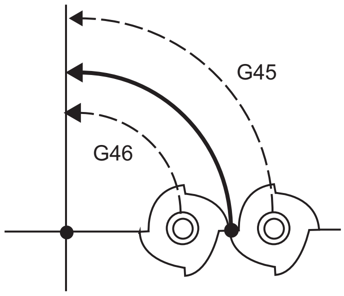
(4) In the case of circular interpolation, tool radius compensation is possible by the G45 to G48 commands only for one quadrant, two quadrants (semi sphere) or three quadrants when the start and end points are on the axis.

The commands are assigned as follows depending on whether the compensation is applied for outside or inside the arc programmed path.

However, in this case, compensation must already be provided in the desired direction at the arc start point. (If a compensation command is assigned for the arc independently, the arc start point and end point radius will shift by an amount equivalent to the compensation amount.)

The program path is indicated by the heavy line in the figure.

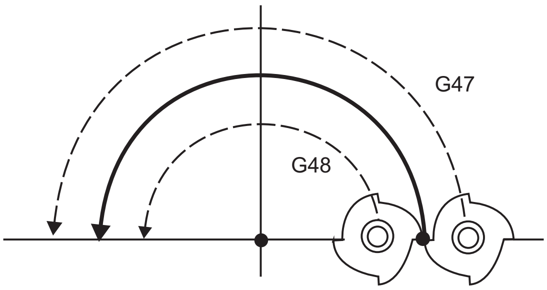
1/4 circle



G45 for compensation outside the circle
G46 for compensation inside the circle

Program path
Compensated path

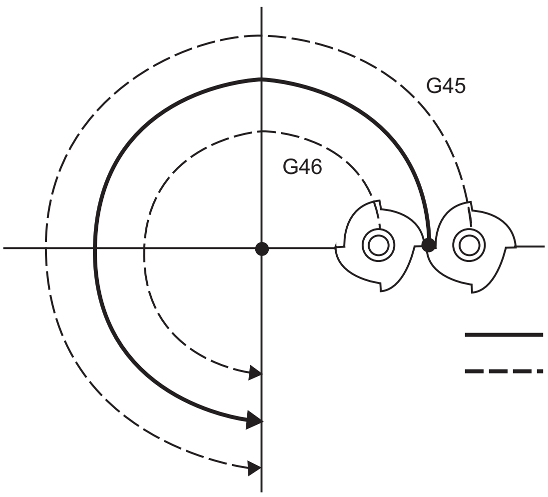
1/2 circle



G47 for compensation outside the circle
G48 for compensation inside the circle

Program path
Compensated path

3/4 circle



G45 for compensation outside the circle
G46 for compensation inside the circle

Program path
Compensated path

13

Fixed Cycle

13.1 Fixed Cycles

Function and purpose

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

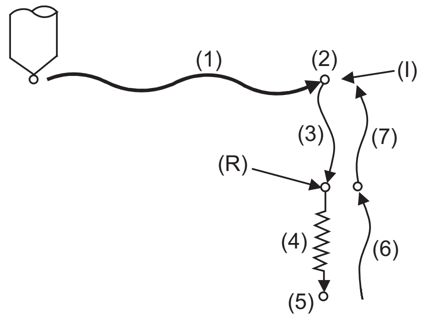
G Code	Hole drilling start (-Z direction)	Operation at hole bottom		Return operation (+Z direction)	Retract at high speed	Application
		Dwell	Spindle			
G80	-	-	-	-	-	Cancel
G81	Cutting feed	-	-	Rapid traverse	Possible	Drill, spot drilling cycle
G82	Cutting feed	Yes	-	Rapid traverse	-	Drill, counter boring cycle
G83	Intermittent feed	-	-	Rapid traverse	Possible	Deep hole drilling cycle
G84	Cutting feed	Yes	Reverse rotation	Cutting feed	-	Tapping cycle
G85	Cutting feed	-	-	Cutting feed	-	Boring cycle
G86	Cutting feed	Yes	Stop	Rapid traverse	-	Boring cycle
G87	Rapid traverse	-	Forward rotation	Cutting feed	-	Back boring cycle
G88	Cutting feed	Yes	Stop	Rapid traverse	-	Boring cycle
G89	Cutting feed	Yes	-	Cutting feed	-	Boring cycle
G73	Intermittent feed	Yes	-	Rapid traverse	Possible	Stepping cycle
G74	Cutting feed	Yes	Forward rotation	Cutting feed	-	Reverse tapping cycle
G75	Cutting feed	-	-	Rapid traverse	-	Circular cutting cycle
G76	Cutting feed	-	Oriented spindle stop	Rapid traverse	-	Fine boring cycle

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

Detailed description

Basic operations of fixed cycle for drilling

There are 7 actual operations which are each described below.



(I) Initial point
(R) R point return

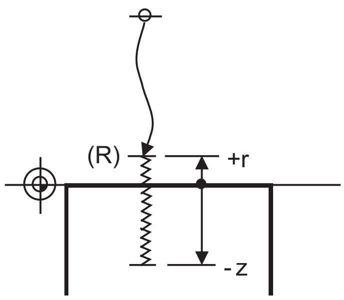
(1) This indicates the X and Y axes positioning, and executes positioning with G00.
(2) This is an operation done after positioning is completed (at the initial point), and when G87 is commanded, the M19 command is output from the control unit to the machine. When this M command is executed and the finish signal (FIN) is received by the control unit, the next operation will start. If the single block stop switch is ON, the block will stop after positioning.
(3) The tool is positioned to the R point by rapid traverse.
(4) Hole machining is conducted by cutting feed.
(5) This operation takes place at the hole bottom position, and depending on the fixed cycle mode, the operation can be the spindle stop (M05), the rotary tool reverse rotation (M04), rotary tool forward rotation (M03), dwell or tool shift.
(6) The tool is retracted to the R point at the cutting feed or the rapid traverse rate, depending on the fixed cycle mode.
(7) The tool is returned to the initial point at rapid traverse rate.

Note

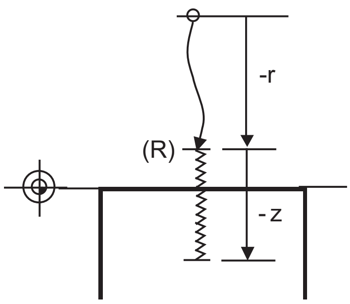
- Whether the fixed cycle is to be completed at operation 6 or 7 can be selected by G98/G99 commands. (Refer to "Initial point and R point level return; G98, G99")

Difference between absolute value command and incremental value command

For absolute value



For incremental value



(R) R point

Positioning plane and hole drilling axis

The fixed cycle has basic control elements for the positioning plane and hole drilling axis. The positioning plane is determined by the G17, G18 and G19 plane selection commands, and the hole drilling axis is the axis perpendicular (X, Y, Z or their parallel axis) to the above plane.

Plane selection	Positioning plane	Hole drilling axis
G17 (X-Y)	Xp-Yp	Zp
G18 (Z-X)	Zp-Xp	Yp
G19 (Y-Z)	Yp-Zp	Xp

Xp, Yp and Zp indicate the basic axes X, Y and Z or an axis parallel to the basic axis.

An arbitrary axis other than the hole drilling axis can be commanded for positioning.

The hole drilling axis is determined by the axis address of the hole drilling axis commanded in the same block as G81 to G89, G73, G74 or G76. The basic axis will be the hole drilling axis if there is no designation.

(Example 1) When G17 (X-Y plane) is selected, and the axis parallel to the Z axis is set as the W axis.

```
G81 .... Z_  ;        The Z axis is used as the hole drilling axis.
G81 .... W_;          The W axis is used as the hole drilling axis.
G81 ....     ;        (No Z or W) The Z axis is used as the hole drilling axis.
```

Note

(1) The hole drilling axis can be fixed to the Z axis with parameter #1080 Dril_Z.

(2) Changeover of the hole drilling axis must be done with the fixed cycle canceled.

In the following explanations on the movement in each fixed cycle mode, the XY plane is used for the positioning plane and the Z axis for the hole drilling axis. Note that all command values will be incremental values, the positioning plane will be the XY plane and the hole drilling axis will be the Z axis.

Programmable in-position width command in fixed cycle

This commands the in-position width for commanding the fixed cycle from the machining program. The commanded in-position width is valid only in the eight fixed cycles; G81 (drill, spot drill), G82 (drill, counter boring), G83 (deep drill cycle), G84 (tap cycle), G85 (boring), G89 (boring), G73 (step cycle) and G74 (reverse tap cycle). The ", I" address is commanded in respect to the positioning axis, and the ",J" address is commanded in respect to the drilling axis.

Address	Meaning	Command range (unit)	Remarks
,I	Positioning axis in-position width(position error amount)	0.001 to 999.999 (mm)	If a value exceeding the command range is commanded, a program error (P35) will occur.
,J	Drilling axis in-position width(position error amount)		

In-position check in fixed cycle

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

:

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

:

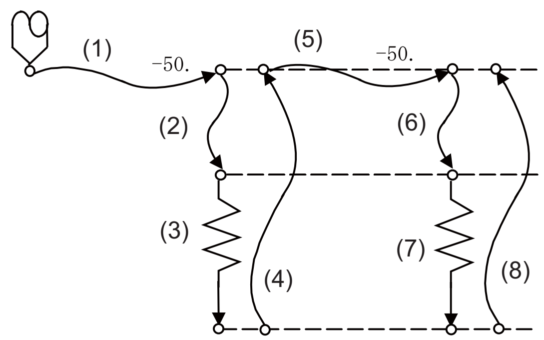


Fig. 1 Operation when number of repetitions L is designated

Operation pattern	,I	,J
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	Valid
(5)	Valid	-
(6)	-	Invalid
(7)	-	Invalid
(8)	-	Valid

In the following machining program, the commanded in-position width is valid for the Fig. 2 block. In the (B) block, the in-position width (, I) commanded regarding to positioning in the previous block (A) is invalid (5). However, when returning from the hole bottom, the in-position width (, J) commanded in the previous block (A) is valid (8).
To validate the in-position width for positioning, command again as shown in block (C) (9).

:

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

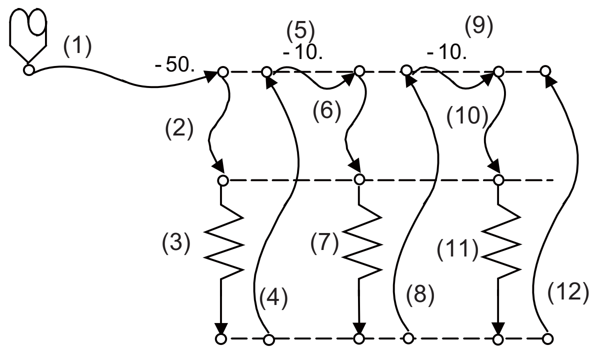


Fig. 2 Operation in fixed cycle modal

Operation pattern	,I	,J
(1)	Valid	-
(2)	-	Invalid
(3)	-	Invalid
(4)	-	Valid
(5)	Invalid	-
(6)	-	Invalid
(7)	-	Invalid
(8)	-	Valid
(9)	Valid	-
(10)	-	Invalid
(11)	-	Invalid
(12)	-	Valid

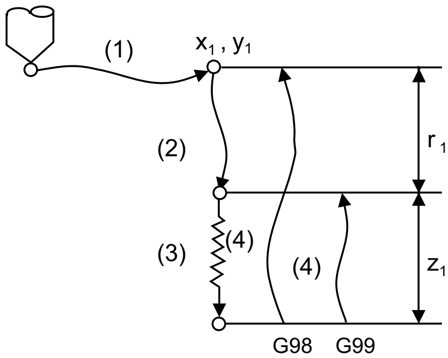
13.1.1 Drilling, Spot Drilling ; G81

Command format

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

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

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zz1 Ff1
(4)	-	Valid	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1

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

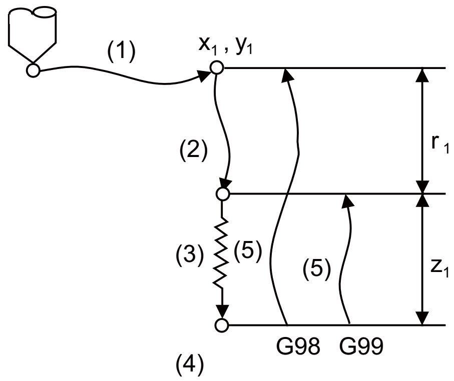
13.1.2 Drilling, Counter Boring ; G82

Command format

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

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

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zz1 Ff1
(4)	-	-	G04Pp1 (Dwell)
(5)	-	Valid	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1

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

13.1.3 Deep Hole Drilling Cycle ; G83

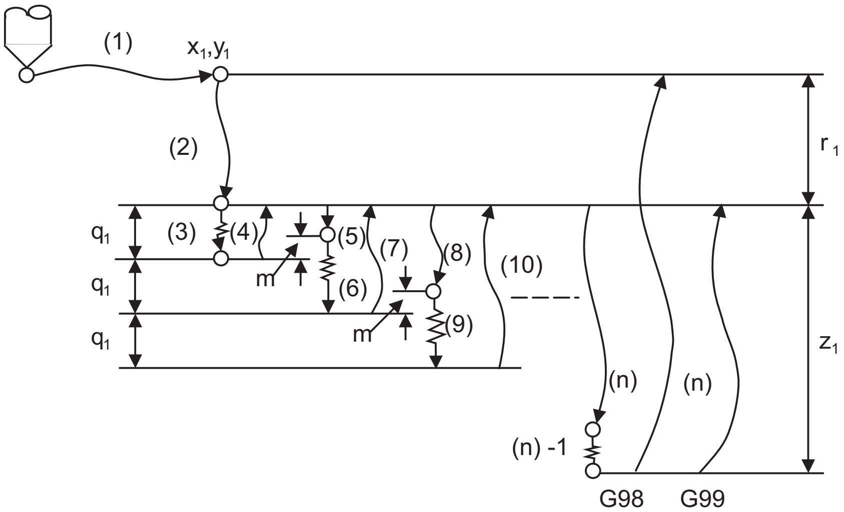
13.1.3.1 Deep Hole Drilling Cycle

Command format

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

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

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zq1 Ff1
(4)	-	Invalid	G00 Z-q1
(5)	-	Invalid	G00 Z(q1-m)
(6)	-	Invalid	G01 Z(q1+m) Ff1
(7)	-	Invalid	G00 Z-2*q1
(8)	-	Invalid	G00 Z(2*q1-m)
(9)	-	Invalid	G01 Z(q1+m) Ff1
(10)	-	Invalid	G00 Z-3*q1
:			
(n)-1	-	Invalid	
(n)	-	Valid	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1

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

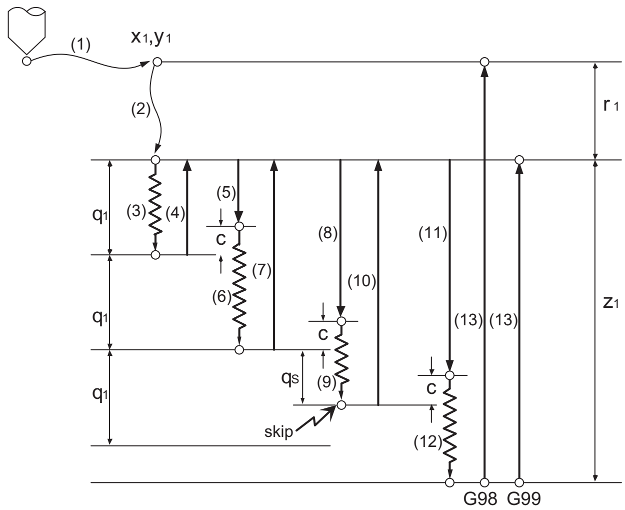
13.1.3.2 Small Diameter Deep Hole Drilling Cycle

Command format

G83 Xx1 Yy1 Zz1 Rr1 Qq1 Ff1 Ii1 Pp1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Qq1	Cut amount for each cutting pass (incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Ii1	The feedrate from R point to the cutting start position, the speed (mm/min) for returning from the hole bottom are stored only in the same block as G83, and it is valid until the small diameter deep hole drilling cycle is canceled. (It follows the setting of "#8085 G83S Forward F", "#8086 G83S Back F" when omitted.)
Pp1	Dwell time at hole bottom position

Detailed description



c: Parameter "#8084 G83S Clearance"

qs: Small diameter deep hole drilling cycle signal (YCCA)

skip: Small diameter deep hole drilling cycle signal (YCCA) input

Operation pattern	Program
(1)	G00 Xx1 Yy1,Ii1
(2)	G00 Zr1
(3)	G01 Zq1 Ff1
(4)	G01 Z-q1 Fi2
(5)	G01 Z(q1-c) Fi1
(6)	G01 Z(q1+c) Ff1
(7)	G01 Z-2•q1 Fi2
(8)	G01 Z(2•q1-c) Fi1
(9)	G01 Z(q1+c) Ff1
(10)	G01 Z-(2•q1+qs) Fi2
(11)	G01 Z(2•q1+qs-c) Fi1
(12)	G01 Z(z1-q1*n-qs) Ff1
(13)	G98 mode G01 Z-(z1+r1) Fi2 G99 mode G01 Z-z1 Fi2

"i1" follows the parameter "#8085 G83S Forward F" when there is no I command.

"i2" follows the parameter "#8086 G83S Back F" when there is no I command.

In deep hole drilling, cutting and retract are repeated and the workpiece is machined multiple times. In addition, when PLC signals are input during cutting, the cutting for the time concerned is skipped. In this way, this cycle reduces the load applied to the tool.

The small-diameter deep-hole drilling cycle mode is established by designating the M code command that was set in the parameter "#8083 G83S mode M".
If the G83 command is designated in this mode, the small-diameter deep-hole drilling cycle is executed.
The mode is canceled by the following conditions.
- Designation of a fixed cycle cancel command (G80, G commands in Group 1)
- Resetting

It is not immediately switched to the small diameter deep hole drilling cycle mode even the small diameter deep hole drilling cycle switch M command is issued during G83 deep hole drilling cycle modal. Then, when G83 is commanded, the small diameter deep hole drilling cycle mode is applied.

When the small diameter deep hole drilling cycle signal (YCCA) is input during the cutting operation (9), the remaining cutting command is skipped and the axis returns to the R point at the cutting speed i2.

"In small diameter deep hole cycle signal (XCC1)" is output between the positioning to the R point of drilling axis (2) and the R point/initial point return after finishing the drilling (13).

"c" depends on the parameter "#8084 G83S Clearance".

Program the small diameter deep hole drilling cycle to make it "q1 > c".

The operation stops at after the (1), (2) and (13) commands during single block operation.

If there is no "I" command, or either the parameter "#8085 G83S Forward F" or "#8086 G83S Back F" is set to "0", a program error (P62) will occur.

Confirm the following related parameters before using the small hole diameter drilling cycle.
- #8083 G83S Mode M
- #8084 G83S Clearance
- #8085 G83S Forward F
- #8086S Back F

13.1.4 Tapping Cycle ; G84

Command format

G84 Xx1 Yy1 Zz1 Rr1 Qq1 Ff1(Ee1) Pp1 ,Rr2 Ss1 ,Ss2 ,Ii1 ,Jj1 Ll1 (Kk1);

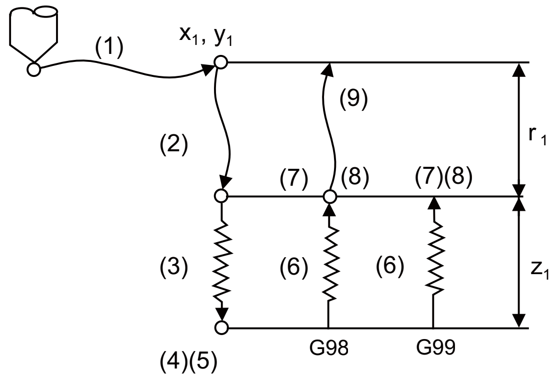
Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Qq1	Cut amount for each cutting pass (incremental value) (modal)
Ff1	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)
Ee	Cutting feedrate at synchronous tapping (Number of screw threads per inch) If this command is issued simultaneously with the F command, the F command is valid.
Pp1	Designation of dwell time at hole bottom position (after the decimal points will be ignored) (modal)
,Rr2	Synchronization method selection (r2=1 synchronous, r2=0 asynchronous) (When omitted, the mode will follow the setting of parameter "#8159 Synchronous tap")
Ss1	Spindle rotation speed command <Note> •At a synchronous tapping mode, "Sn = *****" type S command will be ignored. (n:spindle number, *****: rotation speed) •If an S command is issued during synchronous tapping modal, a program error (P186) will occur.
,Ss2	Spindle rotation speed during return
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width
Ll1	Designation of number of repetitions (0 to 9999) When ″0″ is set, processing is not executed.
Kk1	Number of repetitions (It can be commanded when the parameter "#1271 ext07/bit1" is "1")

Note

(1) ",S" command is held as a modal information.
When the value of the ",S" command is lower than the spindle rotation speed (S command), operations depend on the MTB specifications.
(Whether it operates on the spindle rotation speed at the return or on the spindle rotation speed of S command is determined according to the parameter "#1241 set13"/bit7 setting.)
When the spindle speed at the return is not "0", the value of tap return override (#1172 tapovr) is invalid.

Detailed description

Normal tapping cycle (When Q is not designated)



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zz1 Ff1
(4)	-	-	G04 Pp1
(5)	-	-	M4 (Spindle reverse rotation)
(6)	-	Invalid	G01 Z-z1 Ff1
(7)	-	-	G04 Pp1
(8)	-	-	M3 (Spindle forward rotation)
(9)	-	Valid	G98 mode G00 Z - r1 G99 mode No movement

When r2 = 1, the synchronous tapping mode will be applied, and when r2 = 0, the asynchronous tapping mode will be applied. If there is no r2 command, mode will follow the parameter setting.

When G84 is executed, the override will be canceled and automatically be set to 100%.

Dry run is valid for the positioning command when the control parameter "G00 DRY RUN" is on. If the feed hold button is pressed during G84 execution, and the sequence is at (3) to (6), the movement will not stop immediately, and instead will stop after (6). During the rapid traverse in sequence (1), (2) and (9), the movement will stop immediately.

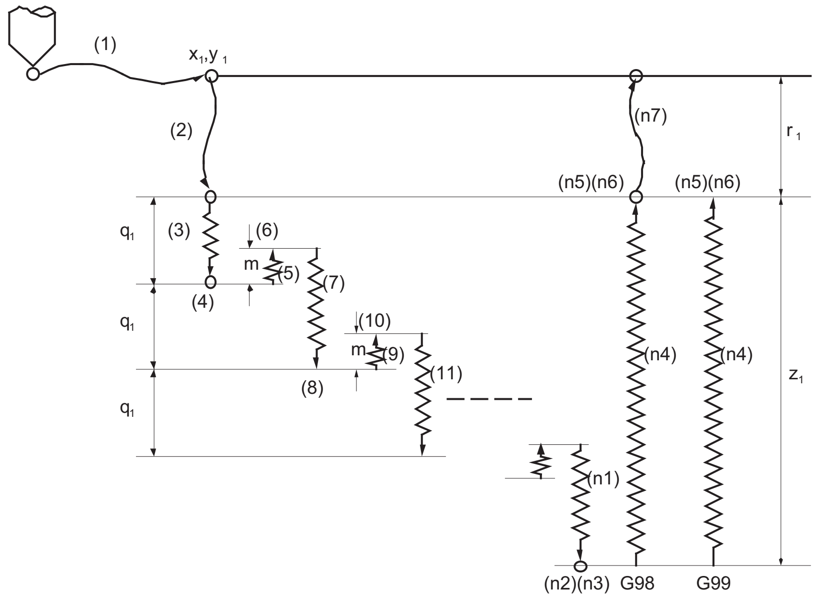
The operation stops at after the (1), (2) and (9) commands during single block operation.

(f) During the G84 mode, the NC signal "Tapping" will be output.

During the G84 synchronous tapping modal, the M3, M4, M5 and S code will not be output.

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

Pecking Tapping Cycle (When the Q command is designated #1272 ext08/bit4=0)



m : parameter (#8018 G84/G74 n)

Note

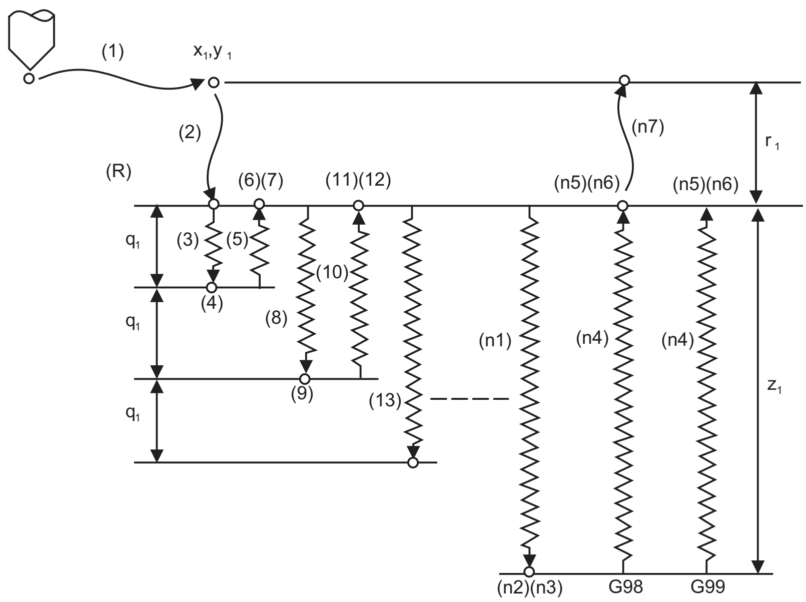
•This program is for the G84 command.

The spindle forward rotation (M3) and reverse rotation (M4) are reversed with the G74 command.

Operation pattern	Program
(1)	G00 Xx1 Yy1 ,Ii1
(2)	G00 Zr1
(3)	G01 Zq1 Ff1
(4)	M4 (Spindle reverse rotation)
(5)	G01 Z-m Ff1
(6)	M3 (Spindle forward rotation)
(7)	G01 Z(q1+m) Ff1
(8)	M4 (Spindle reverse rotation)
(9)	G01 Z-m Ff1
(10)	M3 (Spindle forward rotation)
(11)	G01 Z(q1+m) Ff1
:	:
(n1)	G01 Z(z1-q1*n) Ff1
(n2)	G04 Pp1
(n3)	M4 (Spindle reverse rotation)
(n4)	G01 Z-z1 Ff1 Ss2
(n5)	G04 Pp1
(n6)	M3 (Spindle forward rotation)
(n7)	G98 mode G00 Z-r1 ,Jj1 G99 mode No movement

The load applied to the tool can be reduced by designating the depth of cut per pass (Q) and cutting the workpiece to the hole bottom for a multiple number of passes. The amount retracted from the hole bottom is set to the parameter "#8018 G84/G74 n". Whether the pecking tapping cycle or the deep-hole tapping cycle is valid depends on the MTB specifications (parameter "#1272 ext08/bit4"). When "depth of cut per pass Q" is designated in the block containing the G84 or G74 command with the pecking tapping cycle selected, the pecking tapping cycle will be executed.
In the following cases, the normal tapping cycle will be carried out.
- When Q is not designated.
- When the command value of Q is zero.

Deep-hole Tapping Cycle (When the Q command is designated #1272 ext08/bit4=1)



(R) R point

Note

(1) This program is for the G84 command.

The spindle forward rotation (M3) and reverse rotation (M4) are reversed with the G74 command.

Operation pattern	Program
(1)	G00 Xx1 Yy1
(2)	G00 Zr1
(3)	G09 G01 Zq1 Ff1.
(4)	M4 (Spindle reverse rotation)
(5)	G09 G01 Z-q1 Ff1
(6)	G04 Pp1
(7)	M3 (Spindle forward rotation)
(8)	G09 G01 Z(2*q1) Ff1
(9)	M4 (Spindle reverse rotation)
(10)	G09 G01 Z-(2*q1) Ff1
(11)	G04 Pp1
(12)	M3 (Spindle forward rotation)
(13)	G09 G01 Z(3*q1) Ff1
:	:
(n1)	G09 G01 Zz1 Ff1

Operation pattern	Program
(n2)	G04 Pp1
(n3)	M4 (Spindle reverse rotation)
(n4)	G09 G01 Z-z1 Ff1
(n5)	G04 Pp1
(n6)	M3 (Spindle forward rotation)
(n7)	G98 mode G00 Z - r1 G99 mode No movement

(a)In the deep-hole tapping, the load applied to the tool can be reduced by designating the depth of cut per pass and cutting the workpiece to the hole bottom for a multiple number of passes.
Under the deep-hole tapping cycle, the tool is retracted to the R-point every time.

(b) Whether the pecking tapping cycle or the deep-hole tapping cycle is valid depends on the MTB specifications (parameter "#1272 ext08/bit4").
When "depth of cut per pass Q" is designated in the block containing the G84 or G74 command in the state where the deep-hole tapping cycle is selected by parameter, the deep-hole tapping cycle is executed.
In the following cases, the normal tapping cycle will be carried out.
- When Q is not designated.
- When the command value of Q is zero.

(c) When G84 is executed, the override will be canceled and the override will automatically be set to 100% in the cutting operation. And the override set in the parameter "#1172 tapovr" will also be disabled.
(When "#1272 ext08/bit5" = 1, the setting of "#1172 tapovr" will be enabled only during a pulling operation)

(d) Dry run is valid for a positioning command when the parameter "#1085 G00 DRY RUN" is "1" and is valid for the positioning command. If the feed hold button is pressed during G84 execution, the tool does not stop immediately during cutting or returning, and it stops after completing an R point return.

(e) During single block operation, the tool does not stop during cutting or returning, but stops after completing an R point/initial point return.

(f) During the G84 mode, the NC signal "Tapping" will be output.

(g) During the G84 synchronous tapping mode, the M3, M4, M5 or S code will not be output.

(h) If the command value of F becomes extremely small such as around "F < 0.01mm/rev" during synchronous tapping, the spindle does not rotate smoothly. So make sure to command a value larger than "0.01mm/rev". The unit of F can be selected between mm/rev and mm/min.

(i) If the external deceleration signal is turned ON during synchronous or asynchronous tapping, the feed rate does not change even when deceleration conditions are satisfied.

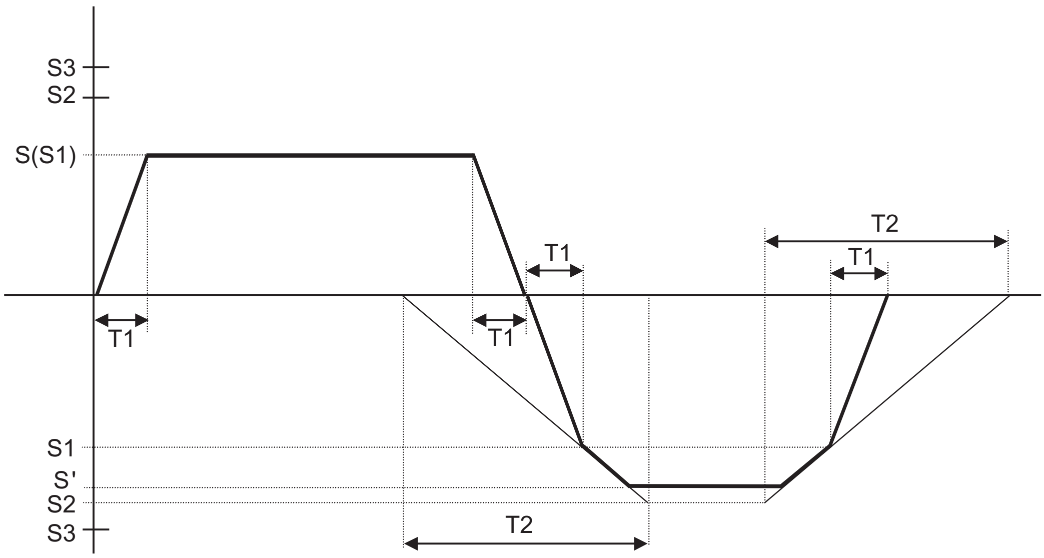
(j) If the operation is interrupted by a cause such as an emergency stop or reset during the deep-hole tapping cycle, a tap retract is executed when the tap retract signal is input.

(k) When the reference position return signal is input during the deep-hole tapping cycle, a tap retract is carried out, and a reference position return will be executed from the end point of the tap retract.

Spindle acceleration/deceleration pattern during synchronous tapping

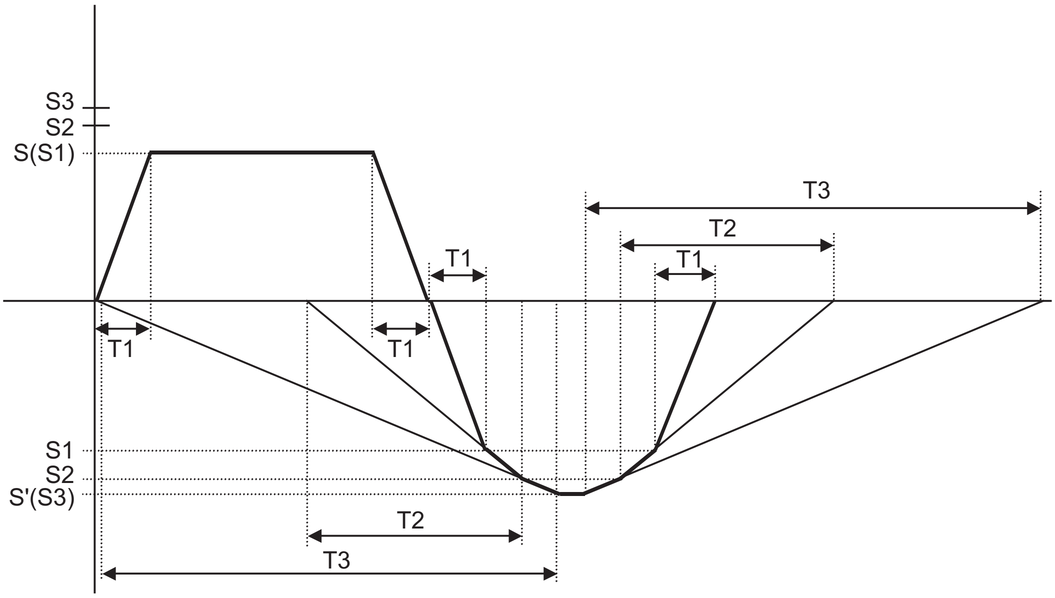
This function enables to make spindle acceleration/deceleration pattern closer to that of the speed loop by dividing the spindle and drilling axis acceleration/deceleration pattern into up to three stages during synchronous tapping.
The acceleration/deceleration pattern can be set up to three stages for each gear. (This depends on the MTB specifications.)
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 specification parameters #3013 to #3016)
S2	Synchronous tapping changeover spindle rotation speed 2 (spindle specification parameters #3037 to #3040)
S3	Maximum spindle rotation speed for synchronous tapping (spindle specification parameters #43046 to #43049) However, when those parameters are set to "0", processing is performed based on "#3005" to "#3008".
T1	Tapping time constant (spindle specification parameters #3017 to #3020)
T2	Synchronous tapping changeover time constant 2 (spindle 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 specification parameters #3013 to #3016)
S2	Synchronous tapping changeover spindle rotation speed 2 (spindle specification parameters #3037 to #3040)
S3	Maximum spindle rotation speed for synchronous tapping (spindle specification parameters #43046 to #43049) However, when those parameters are set to "0", processing is performed based on "#3005" to "#3008".
T1	Tapping time constant (spindle specification parameters #3017 to #3020)
T2	Synchronous tapping changeover time constant 2 (spindle specification parameters #3041 to #3044)
T3	Synchronous tapping changeover time constant 3 (spindle specification parameters #3045 to #3048)

Feedrate for tapping cycle and tapping return

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

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

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

- is irrelevant to the setting

(2) Selection of asynchronous tapping cycle feedrate

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

- is irrelevant to the setting

(3) Spindle rotation speed during return of synchronous tapping cycle

Address	Meaning	Command range (unit)	Remarks
,S	Spindle rotation speed during return	0 to 99999 (r/min)	The data is held as modal information. If the value is smaller than the spindle ro-tation speed, the spindle rotation speed value will be valid even during return. If the spindle rotation speed is not 0 during return, the taping retract override value will be invalid.

M code for forward/reverse rotation command in asynchronous tapping cycle

The M code set with the parameter "#3028 sprcmm" is output as the M code for spindle forward/reverse rotation that is output at "hole bottom" or at "R point" during asynchronous tapping cycle.
Note that the M code for forward rotation is output as "M3" and that for reverse rotation is as "M4" if the parameter "#3028 sprcmm" is set to "0".

Feed per minute command of the synchronous tapping

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

	During G94 modal (feed per minute)	During G95 modal (feed per revolution)
#1268/bit2 = 1	Feed per minute (*1)	Feed per revolution (*2)
#1268/bit2 = 0	Feed per revolution (*2)	Feed per revolution (*2)

(*1) The F command is set to feed per minute (mm/min, inch/min).
Pitch = F command value / S command value

(*2) The F command is set to feed per revolution (mm/rev, inch/rev).

Note

(1) The G94 command, which is a modal command, is valid until the G95 (feed per revolution) command is issued next.

(2) If the E address (number of screw threads per inch) is issued while feed per minute is valid, the program error (P32) will occur.

(3) The F address of the synchronous tapping command does not affect the F modal for cutting feed.

Range restriction of maximum cutting feedrate command for synchronous tapping

You can restrict the maximum value (minimum value of the E address for the number of screw threads) of the pitch F address for synchronous tapping (parameter "#19004 tap feedrate limit"). The program error (P184) will occur if the machining program is executed when the value of "F" address (pitch) exceeds the maximum value or when the value of "E" address (number of the screw threads per inch) is below the minimum value.
When the parameter "#19004" is set to "0", the pitch command by the F address is set as follows.

Command unit	Pitch F	E setting (number of screw threads per inch (*1))
B(0.001mm)	0.001 to 999.999 mm/rev	0.0255 to 999.99
C(0.0001mm)	0.0001 to 999.9999 mm/rev	0.026 to 999.999
D(0.00001mm)	0.00001 to 999.99999 mm/rev	0.0255 to 999.9999
E (0.000001 mm)	0.000001 to 999.999999 mm/rev	0.02541 to 999.99999
B(0.0001inch)	0.000001 to 39.370078 inch/rev	0.03 to 9999.9999
C(0.00001inch)	0.0000001 to 39.3700787 inch/rev	0.026 to 9999.99999
D(0.000001inch)	0.00000001 to 39.37007874 inch/rev	0.0255 to 9999.999999
E (0.0000001inch)	0.000000001 to 39.370078740 inch/rev	0.02541 to 9999.9999999

(*1) When feed per minute is commanded, the pitch calculation result for the spindle rotation speed is range-restricted in this parameter setting.

Synchronous tapping in-position check (Parameter setting values and tapping axis movement)

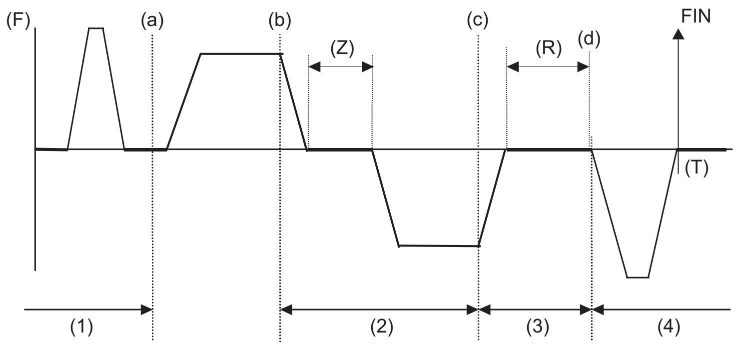
#1223 aux07				"P" designation of G84/G74 command	In-position check during synchronous tapping		
bit3	bit4	bit5	bit2				
In-position check during synchronous tapping	Hole bottom	R point	I point -> R point		Hole bottom	R point	I point -> R point
0	-	-	-	-	yes	yes	yes
1	-	-	-	No "P" designation Example : G84 F1. Z-5. S1000 R-5.	no	no	no
1	1	1	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	(*1)	yes	yes
1	1	0	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	(*1)	no	yes
1	0	1	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	yes	yes	yes
1	0	0	1	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	no	no	yes
1	1	1	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	(*1)	yes	no
1	1	0	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	(*1)	no	no
1	0	1	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	no	yes	no
1	0	0	0	"P" designation Example: G84 F1. Z-5. S1000 P0 R-5.	no	no	no

(*1) Carry out in-position check by tapping in-position width.

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

(a) In-position completion of the G00 feed from the R point

(b) G01 deceleration start at tapping cut-in

(c) G01 deceleration start at tapping return

(d) Start of G00 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 TapInp

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				
In-position check during synchronous tapping	Hole bottom	R point	I point -> R point				
0	-	-	-	Time designated by "P" Several 10 ms as processing time when no "P".	Operation determined by setting of inpos (#1193) and aux07 (#1223):bit 1 parameters.	Operation determined by setting of inpos (#1193) and aux07 (#1223):bit 1 parameters.	Operation determined by setting of inpos (#1193) and aux07 (#1223):bit 1 parameters.
1	0	0	1	The larger value of "P" and TapDwl (#1313) is valid. No dwell is 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	The larger value of "P" and TapDwl (#1313) is valid. No dwell is 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	The larger value of "P" and TapDwl (#1313) is valid. No dwell is 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	The larger value of "P" and TapDwl (#1313) is valid. Several 10 ms 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	The larger value of "P" and TapDwl (#1313) is valid. No dwell is executed if both values are 0.	Wait until time in the left column elapses.		
1	0	1	0	The larger value of "P" and TapDwl (#1313) is valid. No dwell is 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	The larger value of "P" and TapDwl (#1313) is valid. No dwell is 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	The larger value of "P" and TapDwl (#1313) is valid. Several 10 ms 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.

(2) Note that vibration or deterioration in accuracy may occur when invalidating the in-position check at R point. Confirm the accuracy when invalidating it.
Processing may take some time if no in-position check is performed at each point.

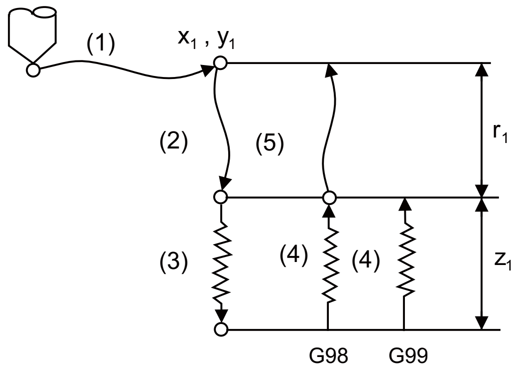
13.1.5 Boring ; G85

Command format

G85 Xx1 Yy1 Zz1 Rr1 Ff1 Ll1 ,Ii1 ,Jj1;

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

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zz1 Ff1
(4)	-	Invalid	G01 Z-z1 Ff1
(5)	-	Invalid	G98 mode G00 Z-r1 G99 mode No movement

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

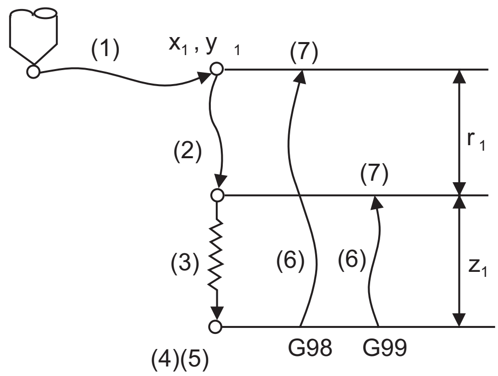
13.1.6 Boring ; G86

Command format

G86 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1 ;

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

Detailed description



Operation pattern	Program
(1)	G00 Xx1 Yy1
(2)	G00 Zr1
(3)	G01 Zz1 Ff1
(4)	G04 Pp1
(5)	M5 (Spindle stop)
(6)	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1
(7)	M3 (Spindle forward rotation)

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

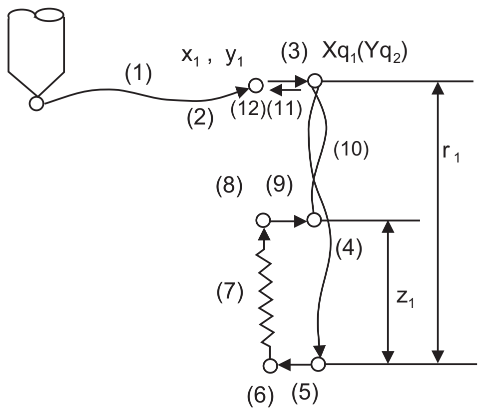
13.1.7 Back Boring ; G87

Command format

G87 Xx1 Yy1 Zz1 Rr1 Iq1 Jq2 Kq3 Ff1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Iq1 Jq2 Kq3	Designation of shift amount (incremental value) (modal) The command address for each plane selection is as follows. G17 plane: IJ G18 plane: KI G19 plane: JK Depending on the parameter setting, the shift amount can be designated by Q address. Refer to "Designation of shift amount (I,J,K)".
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions (0 to 9999) When ″0″ is set, processing is not executed.

Detailed description

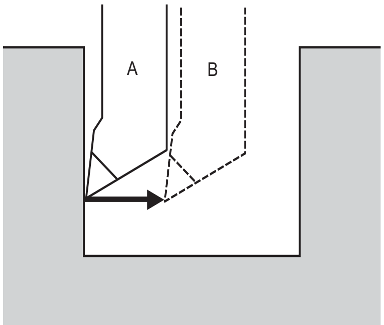


Operation pattern	Program
(1)	G00 Xx1 Yy1
(2)	M19 (Spindle orientation)
(3)	G00 Xq1 (Yq2) (shift)
(4)	G00 Zr1
(5)	G01 X-q1(Y-q2)Ff1 (shift)
(6)	M3 (Spindle forward rotation)
(7)	G01 Zz1 Ff1
(8)	M19 (Spindle orientation)
(9)	G00 Xq1 (Yq2) (shift)
(10)	G98 mode G00 Z - (z1+r1) G99 mode G00 Z - (r1+z1)
(11)	G00 X-q1 (Y-q2) (shift)
(12)	M3 (Spindle forward rotation)

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

Designation of shift amount (I,J,K)

When this command is issued, high precision drilling machining that does not scratch the machining surface can be done. Positioning to the hole bottom and escaping (return) after cutting are carried out in the state shifted to the direction opposite of the cutter.



A: Tool position during cutting

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

The command addresses to designate the shift amount for each plane selection are as follow;

G17 plane: IJ

G18 plane: KI

G19 plane: JK

The shift amount is executed with linear interpolation, and the feedrate follows the F command.
Command I, J, and K with incremental values in the same block as the hole position data.
I, J and K will be handled as modal during the fixed cycle.

Note

(1) If the parameter "#1080 Dril_Z" which fixes the hole drilling axis to the Z axis is set, the shift amount can be designated with address Q instead of I, J. In this case, whether to shift or not and the shift direction are set with parameter "#8207 G76/87 No shift" and "#8208 G76/87 Shift (-)". The sign for the Q value is ignored and the value is handled as a positive value. The Q value is a modal during the fixed cycle, and will also be used as the G83, G73 and G76 cutting amount.

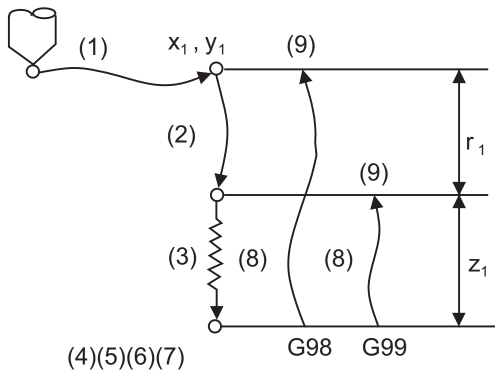
13.1.8 Boring ; G88

Command format

G88 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1 ;

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

Detailed description



Operation pattern	Program
(1)	G00 Xx1 Yy1
(2)	G00 Zr1
(3)	G01 Zz1 Ff1
(4)	G04 Pp1
(5)	M5 (Spindle stop)
(6)	Stop when single block stop switch is ON
(7)	Automatic start switch ON
(8)	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1
(9)	M3 (Spindle forward rotation)

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

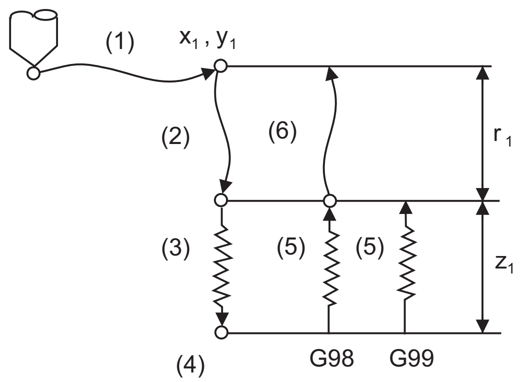
13.1.9 Boring ; G89

Command format

G89 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 Ll1 ,Ii1,Jj1;

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

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zz1 Ff1
(4)	-	-	G04 Pp1
(5)	-	Invalid	G01 Z-z1 Ff1
(6)	-	Valid	G98 mode G00 Z-r1 G99 mode No movement

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

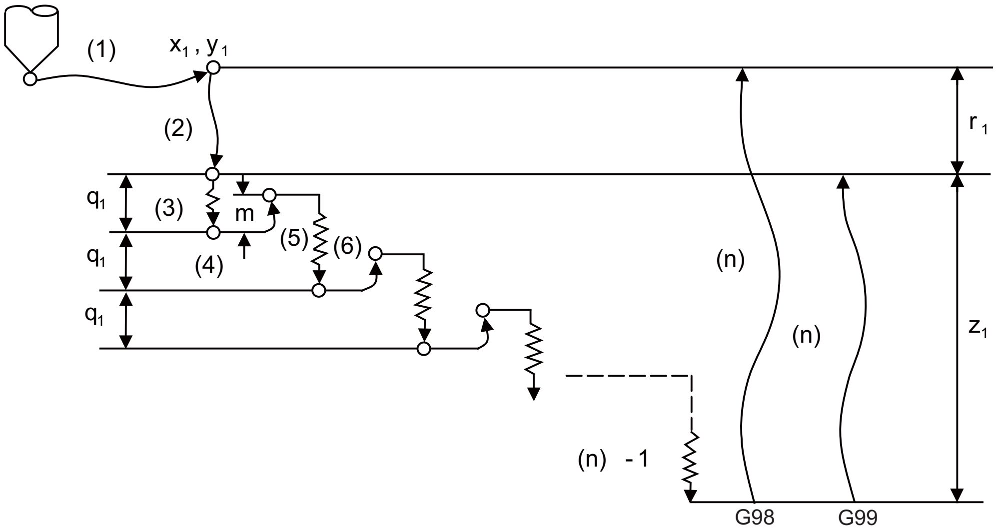
13.1.10 Stepping Cycle ; G73

Command format

G73 Xx1 Yy1 Zz1 Qq1 Rr1 Ff1 Pp1 Ll1 ,Ii1 ,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Qq1	Cut amount for each cutting pass (incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Pp1	Designation of dwell time at hole bottom position (decimal points will be ignored) (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zq1 Ff1
(4)	-	-	G04 Pp1
(5)	-	Invalid	G00 Z-m
(6)	-	Invalid	G01 Z(q1+m) Ff1
:			
(n)-1	-	Invalid	
(n)	-	Valid	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1

When executing a second and following cutting in the G73 as shown above, the movement will return several "m" mm with rapid traverse and then will change to cutting feed. The return amount "m" will differ according to the parameter "#8012 G73 return".
The operation stops at after the (1), (2) and (n) commands during single block operation.

13.1.11 Reverse Tapping Cycle ; G74

Command format

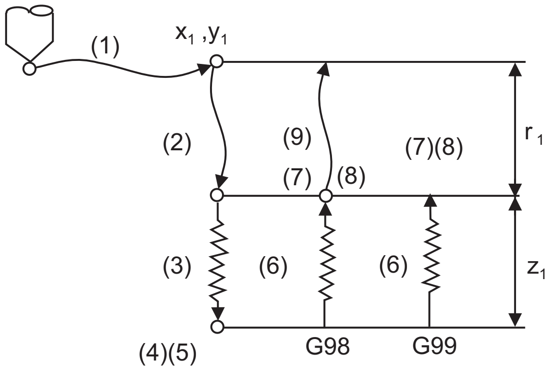
G74 Xx1 Yy1 Zz1 Rr1 Ff1 Pp1 ,Rr2 Ss1 ,Ss2 Ll1 ,Ii1,Jj1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Ff1	Z-axis feed amount (tapping pitch) per spindle rotation (modal)
Pp1	Designation of dwell time at hole bottom position (after the decimal points will be ignored) (modal)
,Rr2	Synchronization method selection (r2=1 synchronous, r2=0 asynchronous) (modal) (When omitted, the mode will follow the setting of parameter "#8159 Synchronous tap")
Ss1	Spindle rotation speed command <Note> ⬩At a synchronous tapping mode, "Sn = *****" type S command will be ignored. (n:spindle number, *****: rotation speed) ⬩If an S command is issued during synchronous tapping modal, a program error (P186) will occur.
,Ss2	Spindle rotation speed during return
Ll1	Designation of number of repetitions (0 to 9999) When ″0″ is set, processing is not executed.
,Ii1	Positioning axis in-position width
,Jj1	Drilling axis in-position width

Note

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

Detailed description



Operation pattern	i1	j1	Program
(1)	Valid	-	G00 Xx1 Yy1
(2)	-	Invalid	G00 Zr1
(3)	-	Invalid	G01 Zz1 Ff1
(4)	-	-	G04 Pp1
(5)	-	-	M3 (Spindle forward rotation)
(6)	-	Invalid	G01 Z-z1 Ff1
(7)	-	-	G04 Pp1
(8)	-	-	M4 (Spindle reverse rotation)
(9)	-	Valid	G98 mode G00 Z - r1 G99 mode No movement

When r2 = 1, the synchronous tapping mode will be applied, and when r2 = 0, the asynchronous tapping mode will be applied. If there is no r2 command, mode will follow the parameter setting.
When G74 is executed, the override will be canceled and the override will automatically be set to 100%.
Dry run is valid for the positioning command when the parameter "#1085 G00 Drn" is set to "1".
If the feed hold button is pressed during G74 execution, and the sequence is at (3) to (6), the movement will not stop immediately, and instead will stop after (6). During the rapid traverse in sequence (1), (2) and (9), the movement will stop immediately.
The operation stops at after the (1), (2) and (9) commands during single block operation.
During the G74 and G84 modal, the "Tapping" NC output signal will be output.
During the G74 synchronous tapping modal, the M3, M4, M5 and S code will not be output.

Spindle acceleration/deceleration pattern during synchronous tapping

Refer to "13.1.4 Tapping Cycle ; G84".

Feedrate for tapping cycle and tapping return

Refer to "13.1.4 Tapping Cycle ; G84".

M code for forward/reverse rotation command in asynchronous tapping cycle

Refer to "13.1.4 Tapping Cycle ; G84".

Parameter setting values and tapping axis

Refer to "13.1.4 Tapping Cycle ; G84".

13.1.12 Circular Cutting ; G75

Function and purpose

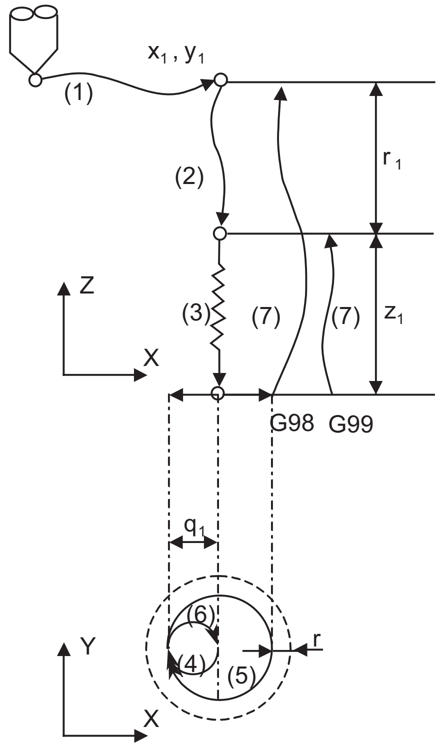
Circular cutting starts with the X and Y axes positioned at the center of the circle, and the Z axis cuts into the commanded position. Then, the tool cuts the inner circumference of the circle drawing a true circle and returns to the center position.

Command format

G75 Xx1 Yy1 Zz1 Rr1 Qq1 Pp1 Ff1 Ll1 ;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Qq1	Radius of outer circumference (modal)
Pp1	Tool radius compensation No. (modal)
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions. (0 to 9999) When ″0″ is set, no execution

Detailed description



Operation pattern	Program	
(1)	G00 Xx1 Yy1	
(2)	G00 Zr1	
(3)	G01 Zz1 Ff1	
(4)	Gn X-(q1-r) I-(q1/2) Inner circumference half circle	
(5)	Iq1	
(6)	X(q1-r) I(q1/2) Inner circumference half circle	n:q1 ≥0 → G02 q1 <0 → G03
(7)	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1	r: Tool radius compensation amount of the No. commanded with p1.

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

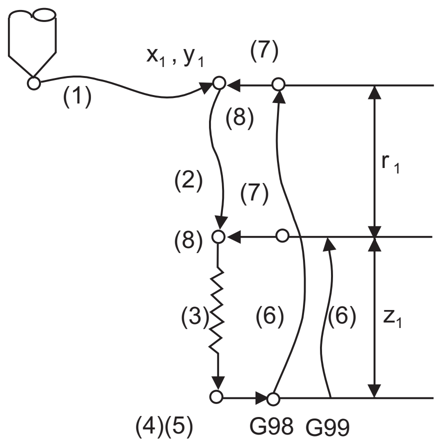
13.1.13 Fine Boring ; G76

Command format

G76 Xx1 Yy1 Zz1 Rr1 Iq1 Jq2 Kq3 Ff1 Ll1;

Xx1	Designation of hole drilling position (absolute value or incremental value)
Yy1	Designation of hole drilling position (absolute value or incremental value)
Zz1	Designation of hole bottom position (absolute value or incremental value) (modal)
Rr1	Designation of R point position (absolute value or incremental value) (modal)
Iq1 Jq2 Kq3	Designation of shift amount (incremental value) (modal) The command address for each plane selection is as follows. G17 plane: IJ G18 plane: KI G19 plane: JK Depending on the parameter setting, the shift amount can be designated by Q address. Refer to "Designation of shift amount (I,J,K)".
Ff1	Designation of feedrate for cutting feed (modal)
Ll1	Designation of number of repetitions (0 to 9999) When ″0″ is set, processing is not executed.

Detailed description

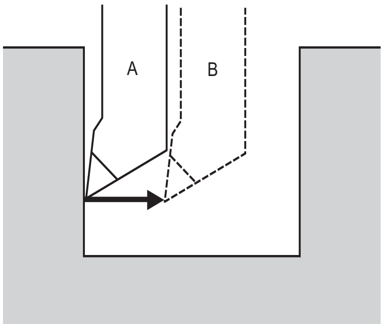


Operation pattern	Program
(1)	G00 Xx1 Yy1
(2)	G00 Zr1
(3)	G01 Zz1 Ff1
(4)	M19 (Spindle orientation)
(5)	G01 Xq1 (Yq2)Ff1 (shift)
(6)	G98 mode G00 Z-(z1+r1) G99 mode G00 Z-z1
(7)	G00 X-q1 (Y-q2) (shift)
(8)	M3 (Spindle forward rotation)

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

Designation of shift amount (I,J,K)

When this command is issued, high precision drilling machining that does not scratch the machining surface can be done.
Positioning to the hole bottom and the escape (return) after cutting is executed in the state shifted to the direction opposite of the cutter.



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

The command addresses to designate the shift amount for each plane selection are as follow;

G17 plane: IJ

G18 plane: KI

G19 plane: JK

The shift amount is executed with linear interpolation, and the feedrate follows the F command.

Command I, J, and K with incremental values in the same block as the hole position data.
I, J and K will be handled as modal during the fixed cycle.

Note

(1) If the parameter "#1080 Dril_Z" which fixes the hole drilling axis to the Z axis is set, the shift amount can be designated with address Q instead of I and J. In this case, whether to shift or not and the shift direction are set with parameter "#8207 G76/87 IGNR" and "#8208 G76/87 (-)". The sign for the Q value is ignored and the value is handled as a positive value.
The Q value is a modal during the fixed cycle, and will also be used as the G83, G87 and G73 cutting amount.

13.1.14 Precautions for Using a Fixed Cycle

Precautions

(1) Before the fixed cycle is commanded, the spindle must be rotating in a specific direction with a miscellaneous function command (M3; or M4;).
Note that for the G87 (back boring) command, the spindle rotation command is included in the fixed cycle so only the rotation speed command needs to be commanded beforehand.

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

(3) Command the hole machining data (Q, P, I, J, K) in a block where hole drilling is executed (Block containing a basic axis, additional axis or R data).

(4) The fixed cycle can be canceled by the G00 to G03 or G33 command besides the G80 command. If these are designated in the same block as the fixed cycle, the following will occur.

m = 00 to 03, 33 n = Fixed cycles

Gm Gn X_Y_Z_R_Q_P_L_F_;

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

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

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

(6) If an M function is commanded in the same block as the fixed cycle command, the M code and MF will be output during the initial positioning. The axis will move to the next operation with FIN (finish signal).
If there is a designation of No. of times, the above control will be executed only for the first drilling.

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

(8) If the No. of repetitions L is not designated, L1 will be set. If L0 is designated in the same block as the fixed cycle G code command, the hole machining data will be memorized, but the hole machining will not be executed.

(Example) G73 X_Y_Z_R_Q_P_F_L0_;

Memorize only the codes with an execution address

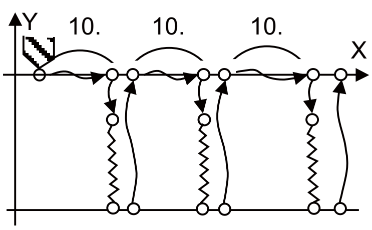
(9) When the fixed cycle is executed, only the modal command issued in the fixed cycle program will be valid in the fixed cycle subprogram. The modal of the program which called the fixed cycle will not be affected.

(10) Other subprograms cannot be called from the fixed cycle subprogram.

(11) Decimal points in the movement command of the fixed cycle subprogram will be ignored.

(12) If the No. of repetitions L is 2 or more during the incremental value mode, the positioning will also be incremented each time.

(Example) G91 G81 X10. Z-50. R-20. F100. L3;



(13) If the spindle rotation speed value during return is smaller than the spindle speed value, the spindle rotation speed value is valid even during return.

(14) If inclinations of the 2nd and 3rd acceleration/deceleration stages according to the spindle rotation speed and time constants set in the parameters are each steeper than the previous stage's inclinations, the previous stage's inclination will be valid.

(15) If the values set in the spindle specification parameter "tap rotation speed" and "the synchronous tap change-over spindle rotation speed 2" exceed the maximum rotation speed, the spindle rotation speed will be clamped at the maximum rotation speed.

(16) If the spindle rotation speed is not 0 during return, the taping retract override value will be invalid.

(17) 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.

```
G01 X100. ,I10.0;
    X-200.;
```

(18) If the in-position width commanded by the programmable in-position width command is increased, the positioning time and linear interpolation time can be reduced. However, the position error amount of the previous block will also increase before the next block starts, and the actual machining could be obstructed.

(19) The in-position width and the position error amount are constantly compared, so the position error amount at the point to be judged as in-position will be smaller than the commanded in-position width.

(20) If the in-position width commanded with the programmable in-position command is small, the commanded deceleration check or in-position check by the parameters may be carried out first.

(21) Synchronous or asynchronous tapping can be selected with the M function.

Base specification parameters

#	Item		Details	Setting range
1272 (PR)	ext08	bit1	M-function synchronous tapping cycle valid.	0: Invalid 1: Valid

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

Base 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.

To use this parameter, validate "#1272 ext08/bit1" (M function synchronous tapping cycle).

The selection of synchronous or asynchronous tappinf 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	B	B	B	B	A	B	B	B

× Not commanded.

○ Commanded.

A Asynchronous tapping

B Synchronous tapping

<Note>

- Do not use M00, 01, 02, 30, 98 and 99.

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

(23) If a tapping axis is under machine lock, normal synchronous tapping is applied even though high-speed synchronous tapping option is enabled.

13.1.15 Initial Point and R Point Level Return ; G98,G99

Function and purpose

Whether to use R point or initial level as the return level in the final sequence of the fixed cycle can be selected.

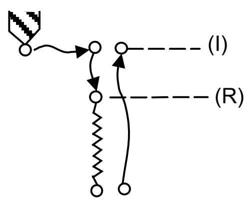
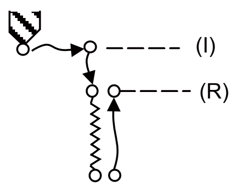
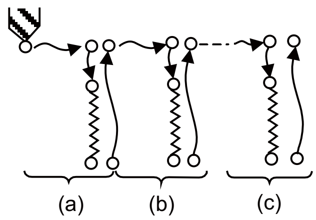
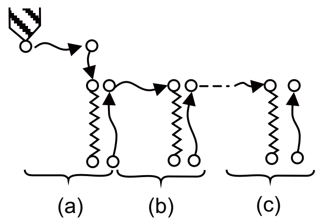
Command format

G98; ... Initial level return

G99; ... R point level return

Detailed description

The relationship of the G98/G99 mode and the number of repetition designation is as shown below.

No. of hole drilling times	Program example	G98 (At power ON, at cancel with M02, M30, and reset button)	G99
Only one execution	G81 X100. Y100. Z-50. R25. F1000 ;	 Initial level return is executed.	 R point level return is executed.
Two or more executions	G81 X100. Y100. Z-50. R25. L5 F1000 ;	 Initial level return is executed for all times.	

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

13.1.16 Setting of Workpiece Coordinates in Fixed Cycle Mode

Function and purpose

The designated axis moves in the workpiece coordinate system set for the axis.
The Z axis becomes valid from the R point positioning after positioning is completed or from Z axis movement.

Note

(1) When the workpiece coordinates change, re-program the addresses Z and R, even if the values are the same.

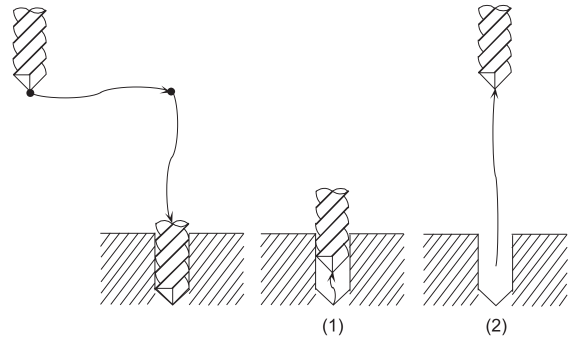
<Example> G54 Xx1 Yy1 Zz1;

```
G81 Xx1 Yy2 Zz2 Rr2;
G55 Xx3 Yy3 Zz2 Rr2        Re-command even if Z and R are the same as the previous value.
Xx4 Yy4 ;
Xx5 Yy5 ;
```

13.1.17 Drilling Cycle High-Speed Retract

Function and purpose

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



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

Command format

The command format is the same as fixed cycle.

Detailed description

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

(2) When "#8123 H-spd retract ON" is ON, the tool is retracted from the hole bottom at high speed using the lost motion compensation function.

(a) Set the lost motion compensation type 2 or 3 to the servo parameter. Then set the following parameters to adjust the retract amount.
- #2170 Lmc1QR (Lost motion compensation gain 1 for high-speed retract)
  (correspond to "#2216 SV016 LMC1 Lost motion compensation 1")
- #2171 Lmc2QR (Lost motion compensation gain 2 for high-speed retract)
  (correspond to "#2241 SV041 LMC2 Lost motion compensation 2")

(b) When the lost motion compensation timing, lost motion compensation 3 spring constant, or lost motion compensation 3 viscous coefficient is set in addition to the ordinary lost motion compensations, its setting value depends on the MTB specifications (parameter shown below).
- #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")

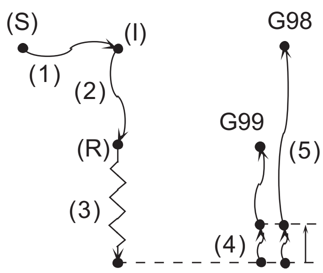
(c) If the drilling axis is synchronously controlled, set the same value in both parameters for master and slave axes.

(3) 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.

(4) This function is invalid during the following command modal .
In this case, the drill moves in the ordinary rapid traverse even if "#8123" is enabled.
- G43.1 (Tool length compensation in the tool axis direction)
- G43.4, G43.5 (Tool center point control)
- G68 (3-dimensional coordinate conversion)

Details of Operation

Operation at G81 command

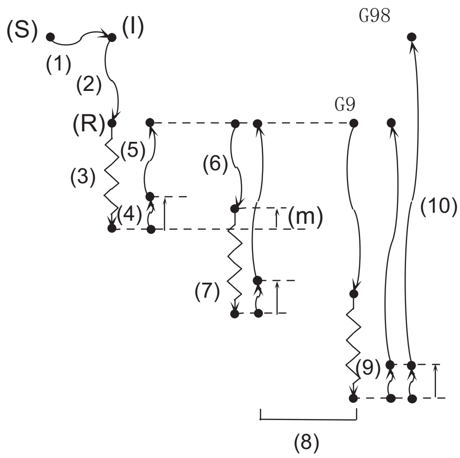


(1) Moves from start point to initial point
(2) Moves from initial point to R point
(3) Cutting feed
(4) Retracted at high-speed
(5) Returns to R point or initial point

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

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

Operation at G83 command

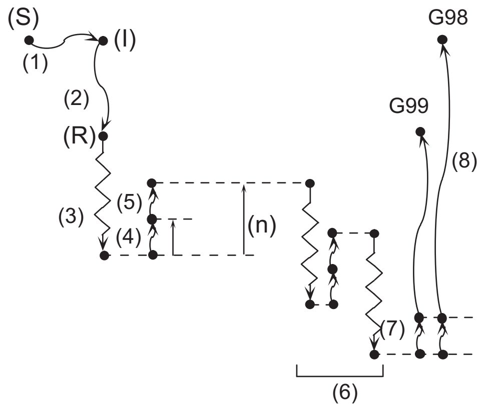


(1) Moves from start point to initial point
(2) Moves from initial point to R point
(3) Cutting feed
(4) Retracted at high-speed
(5) Returns to R point
(6) Moves to the “previous cutting feed position + G83 return amount” position
(7) Cutting feed
(8) Repeats (4) to (7)
(9) Retracted at high-speed
(10) Returns to R point or initial point

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

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

Operation at G73 command



(1) Moves from start point to initial point
(2) Moves from initial point to R point
(3) Cutting feed
(4) Retracted at high-speed
(5) Moves to the position set with “G73 return amount“
(6) Repeats (3) to (5)
(7) Retracted at high-speed
(8) Returns to R point or initial point

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

During single block operation, the axis stops after (1), (2) and (8) only.
If a dwell command is issued, the high-speed retract will be executed after the command.

13.1.18 Acceleration/Deceleration Mode Change in Hole Drilling Cycle

Function and purpose

This function switches the acceleration/deceleration mode for the 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.

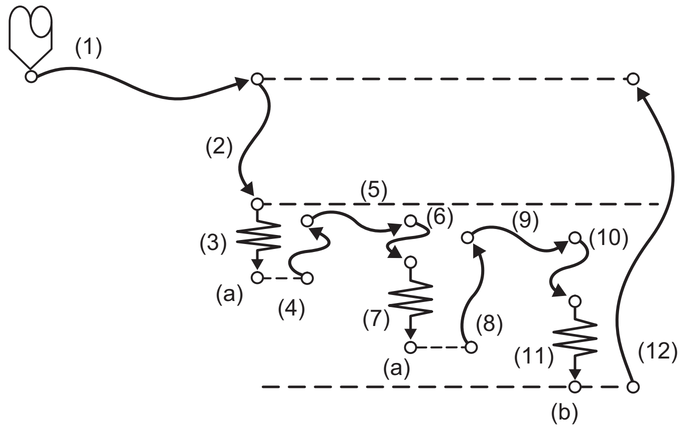
(1) Acceleration/deceleration mode will be either linear or soft method. (Unless soft acceleration/deceleration is applied, the linear method will always be applied.)

(2) The operation follows the parameter settings that determine whether the inclination-constant or postinterpolation 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		G81	G82	G83	G73
0	(a) Cut point	Deceleration Check		Perform no deceleration check.	
	(b) Hole bottom	Perform no deceleration check.			
1	(a) Cut point	Perform no command deceleration check.		Command deceleration check	
	(b) Hole bottom	Command deceleration check			
2	(a) Cut point	Command deceleration check			Perform in-position check (sv024).
	(b) Hole bottom	Perform in-position check (sv024).			

13.2 Special Fixed Cycle

Function and purpose

The special fixed cycle is used with the standard fixed cycle.
Before using the special fixed cycle, record the hole machining data except for the positioning data (except for X, Y plane) by the standard fixed cycle.
The tool is positioned to the hole drilling position when the special fixed cycle is executed. The drilling operation is executed with the fixed cycle for drilling.
Even after the special fixed cycle is executed, the recorded standard fixed cycle will be kept until canceled.
If the special fixed cycle is designated when not in the fixed cycle mode, only positioning will be executed, and the hole drilling operation will not be carried out.
If the special fixed cycle is commanded without commanding the fixed cycle for drilling, positioning will be executed following the current 01 group modal G code.

13.2.1 Bolt Hole Cycle ; G34

Function and purpose

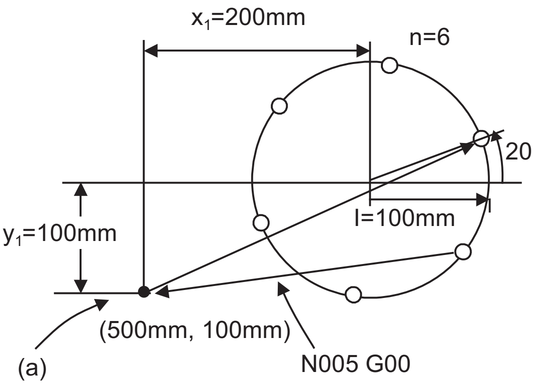
This function is used to drill "n" holes, dividing the circumference by "n", on the circumference with radius R centering the coordinates designated with X and Y. The drilling starts at the point which makes the angle θ with X axis. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G34 will not hold the data after the command is completed.

Command format

G34 Xx1 Yy1 Ir Jθ Kn ;

Xx1,Yy1	Positioning of bolt hole cycle center. This will be affected by G90/G91.
Ir	Radius r of the circle. The unit follows the input setting unit, and is given with a positive No.
Jθ	Angle θ of the point to be drilled first. The CCW direction is positive. (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)
Kn	No. of holes to be drilled: n 1 to 9999 can be designated, but 0 cannot be designated. When the value is positive, positioning will take place in the CCW direction, and when negative, will take place in the CW direction. If "0" is designated, a program error (P221) occurs.

Program example




```
N001 G91;
N002 G81 Z-10.000 R5.000 L0 F200 ;
N003 G90 G34 X200.000 Y100.000 I100.000 J20.000
     K6;
N004 G80 ; --------------- (G81 cancel)
N005 G90 G00 X500.000 Y100.000. ;
```

(a) Position before G34 execution

As shown in the example, the tool position after the G34 command is completed is above the final hole. When moving to the next position, the coordinate value must be calculated to issue the command with an incremental value. Thus, use of the absolute value mode is handy.

Note

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

13.2.2 Line at Angle ; G35

Function and purpose

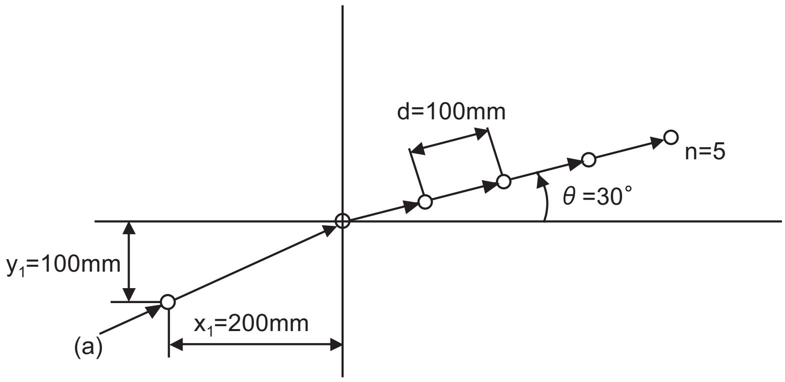
Using the position designated by X and Y as the start point, the n holes will be drilled with interval d in the direction which makes an angle θ with X axis. The hole drilling operation at each hole will follow the standard fixed cycle. The movement between hole positions will all be done in the G00 mode. G35 will not hold the data after the command is completed.

Command format

G35 Xx1 Yy1 Id Jθ Kn ;

Xx1,Yy1	Designation of start point coordinates. This will be affected by G90/G91.
Id	Interval d. The unit follows the input setting unit. If d is negative, the drilling will take place in the direction symmetrical to the center of the start point.
Jθ	Angle θ. The CCW direction is positive. (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)
Kn	Number of holes: n 1 to 9999 can be designated, and the start point is included.

Program example




```
G91 ;
G81    Z-10.000 R5.000 L0 F100 ;
G35    X200.000 Y100.000
       I100.000 J30.000 K5 ;
```

(a) Position before G35 execution

Note

(1) If the K command is K0 or if there is no K command, the program error (P221) will occur.

(2) If the K value is more than four digits, the last four digits will be valid.

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

(4) If G command of group 0 is issued in the same block as the G35 command, the command issued later has the priority.
(Example) G35 G28 Xx1 Yy1 Ii1 Jj1 Kk1 ;   G35 is ignored G 28 is executed as Xx1 Yy1

(5) If there is G72 to G89 commands in the same block as the G35 command, the fixed cycle will be ignored, and the G35 command will be executed.

13.2.3 Arc ; G36

Function and purpose

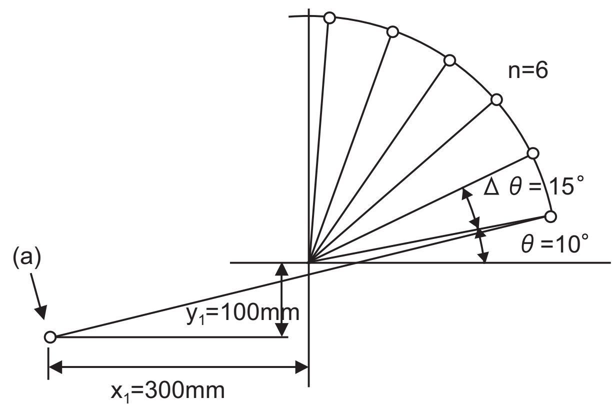
The "n" holes aligned with the angle interval Δθ will be drilled starting at the point which makes the angle θ with the X axis on the circumference with a radius R centering the coordinates designated with X and Y. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G36 will not hold the data after the command is completed.

Command format

G36 Xx1 Yy1 Ir Jθ PΔθ Kn ;

Xx1,Yy1	Center coordinates of arc. This will be affected by G90/G91.
Ir	Radius r of arc. The unit follows the input setting unit, and is given with a positive No.
Jθ	Angle θ of the point to be drilled first. The CCW direction is positive. (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)
P Δθ	Angle interval θ. When the value is positive, the drilling will take place in the CCW direction, and in the CW direction when negative. (The decimal point position will be the degree class. If there is no decimal point, the unit will be 0.001°.)
Kn	No. of holes n to be drilled. The setting range is 1 to 9999.

Program example




```
N001  G91 ;
N002  G81   Z-10.000 R5.000 F100 ;
N003  G36   X300.000 Y100.000 I300.000 J10.000
            P15000 K6 ;
```

(a) Position before G36 execution

Note

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

13.2.4 Grid ; G37.1

Function and purpose

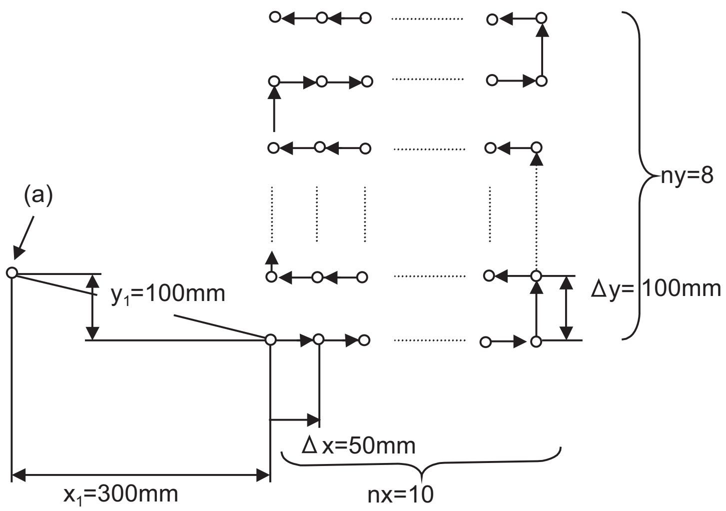
The nx points on a grid are drilled with an interval Δx parallel to the X axis, starting at the position designated with X, Y. The hole drilling operation at each hole will follow the standard fixed cycle.
The movement between hole positions will all be done in the G00 mode. G37.1 will not hold the data after the command is completed.

Command format

G37.1 Xx1 Yy1 IΔx Pnx JΔy Kny ;

Xx1,Yy1	Designate the coordinates at the start point. This will be affected by G90/G91.
I Δx	Interval Δx of the X axis. The unit will follow the input setting unit. If Δx is positive, the interval will be in the forward direction looking from the start point, and when negative, will be in the reverse direction looking from the start point.
Pnx	No. of holes nx in the X axis direction. The setting range is 1 to 9999.
J Δy	Interval Δy of the Y axis. The unit will follow the input setting unit. If Δy is positive, the interval will be in the forward direction looking from the start point, and when negative, will be in the reverse direction looking from the start point.
Kny	No. of holes ny in the Y axis direction. The setting range is 1 to 9999.

Program example




```
G91 ;
G81     Z-10.000 R5.000 F20 ;
G37.1   X300.000 Y-100.000 I50.000 P10
        J100.000 K8 ;
```

(a) Position before G37.1 is executed

Note

(1) If the P and K commands are P0 or K0, or if there is no P or K command, the program error (P221) will occur. If the P or K value is more than four digits, the last four digits will be valid.

(2) If an address other than the selected plane's vertical axis, horizontal axis, G, N, I, J, K, H, O, P, F, M, S or 2nd miscellaneous function is issued in the same block as the G37.1 command, a program error (P32) will occur.

(3) If G command of group 0 is issued in the same block as the G37.1 command, the command issued later has the priority.

(4) If there is G72 to G89 command in the same block as the G37.1 command, the fixed cycle will be ignored, and the G37.1 command will be executed.

(5) If the G22/G23 command is programmed in the same block as the G37.1 command, the G22/G23 command will be ignored, and the G37.1 command will be executed.

14

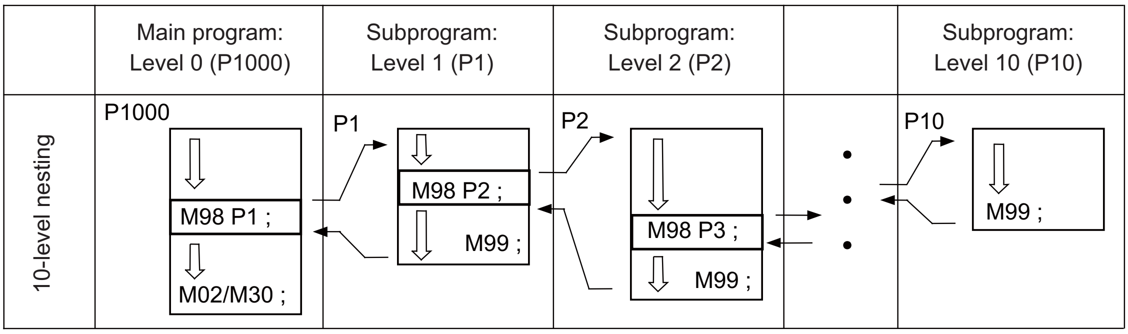
Macro Functions

14.1 Subprogram Control; M98, M99, M198

14.1.1 Subprogram Call ; M98,M99

Function and purpose

Fixed sequences or repeatedly used parameters can be stored in the memory as subprograms that 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 10 levels.



The table below shows the functions that 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 (*2)	×	○	○	×
5. Subprogram nesting level call (*3)	×	○	○	×
6. Fixed cycles	×	×	○	○
7. Editing subprogram for fixed cycle	×	×	○	○

(*1) Symbol “○” denotes available functions and symbol “×” denotes unavailable functions.

(*2) Variables cannot be transferred with the M98 command, but variable commands in subprograms are available if the variable command specifications are provided.

(*3) A maximum of 10 nesting levels form the nesting depth.

Command format

Subprogram call

M98 P__ H__ L__ ,D__ ;

M98 <file name> H__ L__ ,D__ ;

P	Program number in subprogram to be called (own program if omitted) Note that P can be omitted only for memory mode, MDI operation, high-speed program server operation, SD card operation, hard disk operation, or USB operation. (Numeric value of up to 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 inside brackets < >. (The file name can have up to 32 characters including the extension.) (Example) M98 <BUHIN-12. RAF> ;
H	Program sequence number in subprogram to be called (head block if omitted)
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, “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) If a program is called from a subprogram over the nesting depth determined in the specifications, the program error (P230) will occur.

(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, MDI, or BTR 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 that can be called even beyond the 10th nesting level.
- Fixed cycles
- Pattern cycles

(8) To repeatedly use the subprogram, it can be repeated l1 times by programming M98 Pp1 Ll1;.

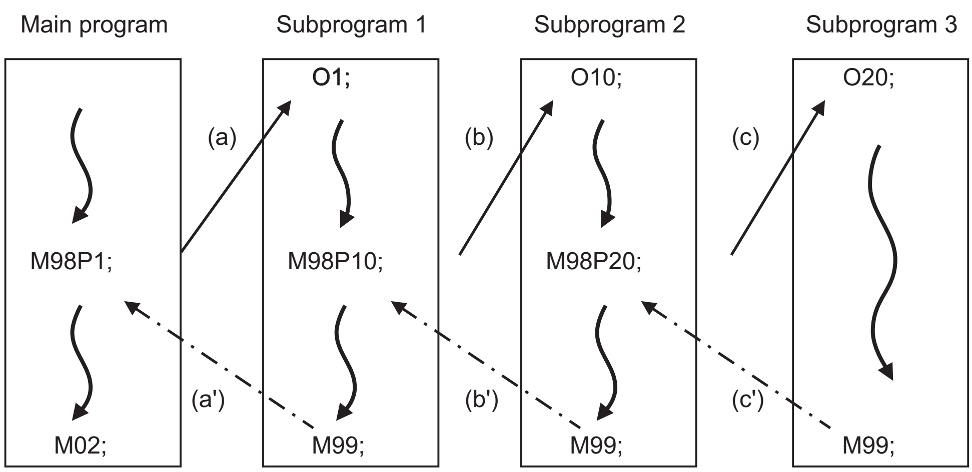
(9) When using the multi-part system, if the subprogram attributed to the part system with the call command is empty, the subprogram call operation will change according to the parameters. (These parameters depend on the MTB specifications.)

#1285 ext21/ bit1	Description
OFF	The subprogram registered in the memory for the selected part system is called out.
ON	The subprogram registered in the memory for the selected part system is called out. If the subprogram in the selected part system is empty, the subprogram with the same No. in the 1st part system is called out.

Program example

Program example 1

When there are 3 subprogram calls (known as 3 nesting levels)



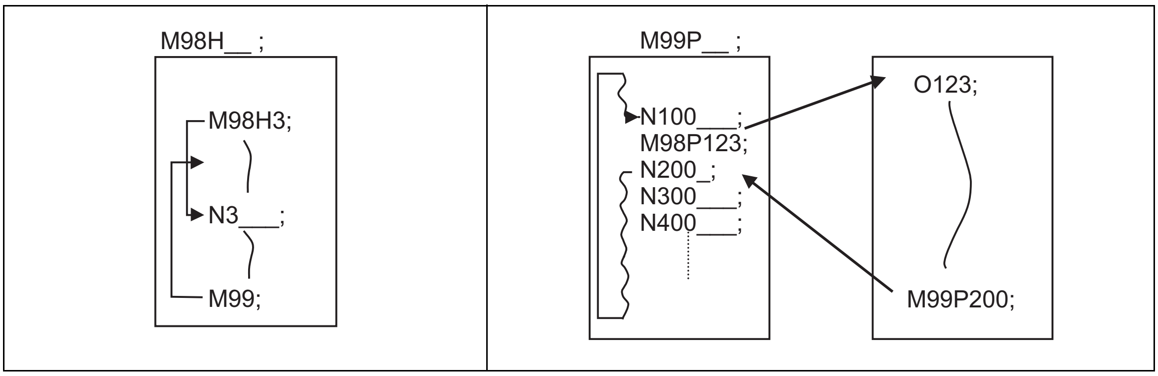
Sequence of execution : (a)-(b)-(c)-(c')-(b')-(a')

(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) The 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 the same as for MDI.)

(4) Branching from tape and BTR 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.

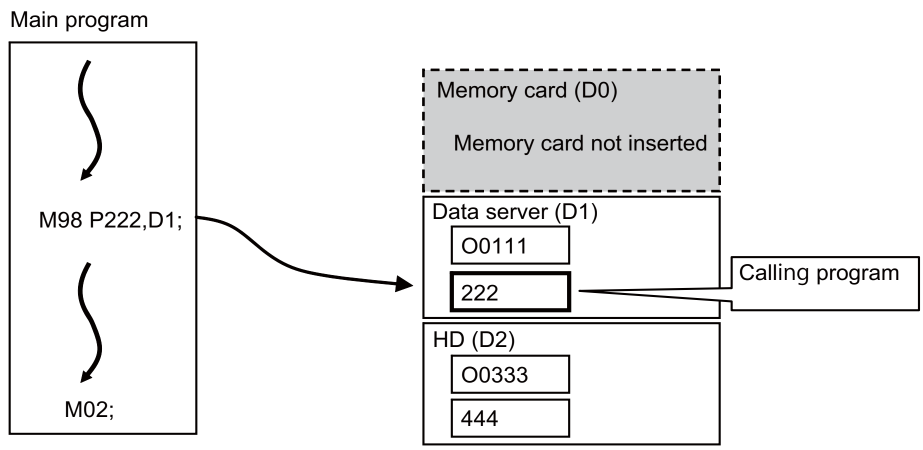
(8) A subprogram added O No. is searched with the parameter setting (#8129="1" or "2") which calls a subprogram with O No. as priority.
If a subprogram with O No. is not found, a subprogram with a name specified with the P command is searched.

<Note>

- To prevent any unintended program call, avoid using 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 disk)

(b) Without designation of device No.

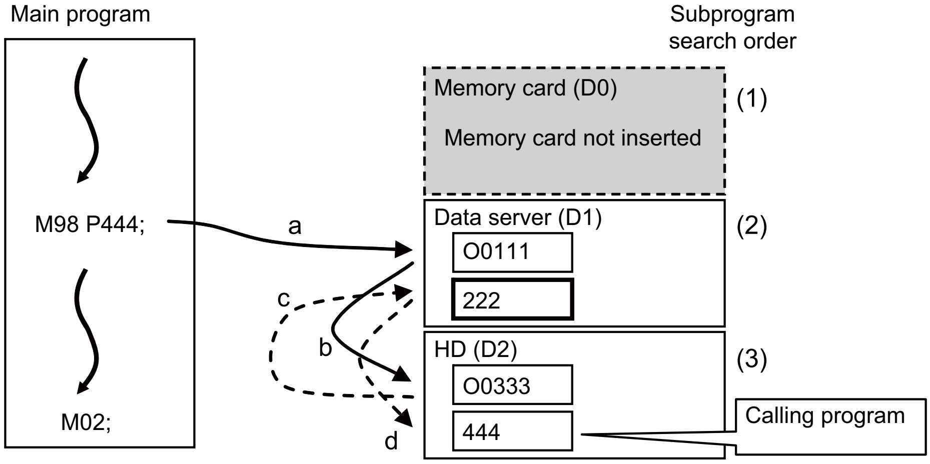
A subprogram with O No. is searched according to the settings of #8890 (D0 in order of subprogram search) to #8894 (D4 in order of subprogram search). (Refer to the solid line arrows "a" and "b" in the figure.)

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

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

Note

- 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 disk)
#8890 Subpro srch odr D0 = 1
#8891 Subpro srch odr D1 = 2
#8892 Subpro srch odr D2 = 3

(9) When a program in an external device such as a USB memory device is executed, a period of processing time is required in the subprogram call or in the instruction to change the flow of the program such as GOTO or DO-END; therefore, interpolation may be decelerated or stopped.

14.1.2 Subprogram Call ; M198

Function and purpose

Programs registered in the SD card can be called as a subprograms. To call a program in the SD card as a subprogram, command the following with the main program.

Command format

Subprogram call

```
M198  P__  L__  ;
```

```
M198 <File name> L__ ;
```

P	Program No. in SD card to be called as a 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 inside 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.

Note

(1) Sequence No. call (M198 H***) cannot be commanded.

Return to main program from subprogram

```
M99  ;
```

Detailed description

(1) The device that can be used for M198 subprogram call differs depending on the NC models.
The SD card in the front side is available for M800S/M80, and the SD card in the control unit is available for M800W. (The M198 command is not available for the C80 series.)

(2) 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.

(3) 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 O No., the program will be valid from the head.

(4) A program registered in an SD card can be executed from only one part system. A program error will occur if an attempt is made to execute the programs in the SD card simultaneously by two or more part systems. If all the part system is reset when the error occurred, programs will be displayed as only "%" except for the first part system.

(5) Refer to "14.1.1 Subprogram Call ; M98,M99" for <File name> and calling the subprogram with O No.

14.1.3 Figure Rotation ; M98 I_J_K_

Function and purpose

If the same pattern is used repeatedly on a concentric circle, one of the rotating machining patterns can be registered as a subprogram. When the subprogram is called from the main program, if the rotation center is designated, a path similar to the rotary phase can be easily created on the concentric circle. This simplifies creation of the program.

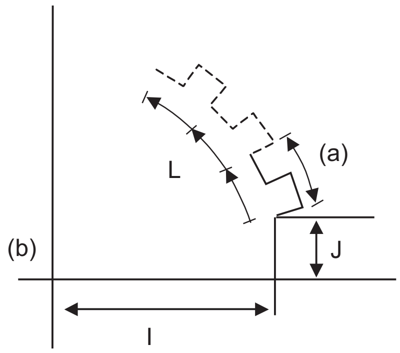
Command format

M98 I__ J__ K__ P__ H__ L__ ,D__ ; ... Subprogram call command

M98 I__ J__ K__ <file name> H__ L__ ,D__ ; ... Subprogram call command

I, J, K	Rotation center coordinates
P	Program number in subprogram to be called (own program if omitted) Note that P can be omitted only during memory mode and MDI mode. (Numeric value containing up to eight digits) 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.) (Example) M98 <BUHIN-12. RAF> ;
H	Program sequence number in subprogram to be called (head block if omitted)
L	Number of subprogram repetitions (If omitted, it is assumed to be "L1", and processing is not carried out when "L0" is set.) (1 to 9999 times depending on the 4-digit value)
,D	Subprogram device No. (0 to 4). The subprogram in the memory can be used when ,D is omitted. The device No. is set with the machining parameters.

Detailed description



(a) Basic figure
(b) Center of rotation

(1) The first subprogram called out with subprogram call is executed at 0° rotation angle. The path is created as commanded.

(2) If the number of repetitions is set to twice or more, the rotation angle is obtained from the called subprogram's start point, end point and rotation center coordinate. The path of the first subprogram is used as the basic figure and is rotated and arranged for the designated number of call repetitions, using the rotation center coordinates as a reference.

(3) All blocks in the subprogram are rotated.

(4) If the subprogram start point and end point are not on the same circle having the commanded figure rotation center coordinates as the center, the axis will interpolate using the subprogram's end point as the start point, and the end point in the first movement command block in the rotated subprogram as the end point.

(5) Both absolute values and incremental values can be used in the figure rotation subprogram. Even if commanded with an absolute value command, the rotation will be the same as when commanded with an incremental value.

(6) I, J and K are commanded with the incremental amount from the start point.

(7) A subprogram of which figure is rotating cannot be branched to the other subprogram.

(8) The figure is rotated on the workpiece coordinate system, and can be shifted with the G92, G52, G54 to G59 (workpiece coordinate system shift) command.

(9) Functions (reference position return, uni-direction positioning, etc.) on the machine coordinate system for the rotary plane axis cannot be used while the figure is rotated. However, the machine coordinate system functions can be used for axes other than the rotation plane.

(10) Refer to "Calling subprogram; M98,M99" for <file name> and calling the subprogram with O No.

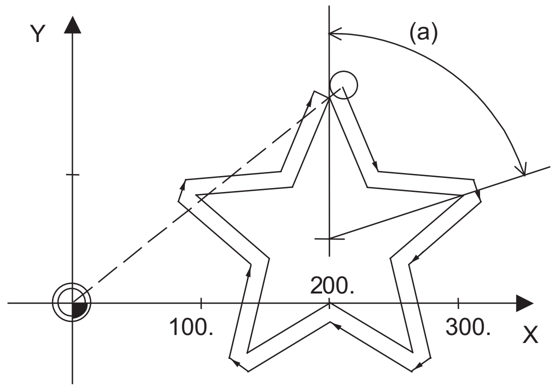
Program example

Main program (O1000)

```
N01 G90 G54 G00 X0 Y0 ;
N02 G01 G41 X200. Y150. D01 F500 ;
N03 G01 Z-50. F300 ;
N04 M98 P2200 L5 J-100. ;
N05 G90 G01 Z50. F500 ;
N06 G40;
N07 G00 X0 Y0 ;
```

Subprogram (O2200)

```
N01 G91 G01 X29.389 Y-59.549 ;
N02 X65.717 Y-9.549 ;
N03 M99 ;
```



(a) Basic figure

Precautions

(1) A program error will occur if figure rotation is commanded during figure rotation.

(2) Figure rotation and program coordinate rotation cannot be commanded simultaneously. The program error will occur.

14.2 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 conditions that exist when executing the program.
All common variables are retained even when the power is turned OFF.
When the power is turned OFF or reset, the common variables can be set to <null> by setting the parameter ("#1128 RstVCl", "#1129 PwrVCl").

Command format

#ΔΔΔ=○○○○○○○○ **;**

#ΔΔΔ = [formula] ;

Detailed description

Variable expressions

		Ex
#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 /.
(2) Functions cannot be used unless the user macro specifications are available.
(3) Error (P241) will occur when a variable No. is negative.
(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.

The common variables are divided into the following two types.

Common variables 1 : Used in common through all part systems

Common variables 2 : Used in common in the programs of the part system

<table>
<tr><th colspan="2">Type</th><th colspan="2">No.</th><th>Function</th></tr>
<tr><td colspan="2">Common variable</td><td>Common variables 1</td><td>Common variables 2</td><td rowspan="6">•Can be used in common throughout main, sub and macro programs.
•When using common variables in the multi-part system, the number of common variables shared between the part systems can be specified depending on the MTB specifications (parameter "#1052 MemVal").
(*2)</td></tr>
<tr><td rowspan="3">1 part systems</td><td>600 sets</td><td>500 to 999
100100 to 800199 (*4)</td><td>100 to 199</td></tr>
<tr><td>700 sets</td><td>400 to 999 (*1)
100100 to 800199 (*4)</td><td>100 to 199</td></tr>
<tr><td>8000 sets</td><td>400 to 999 (*1)
100100 to 800199 (*4)
900000 to 907399 (*3)</td><td>100 to 199</td></tr>
<tr><td rowspan="2">Multi-part systems
(n = number of part systems)</td><td>600 + 100 * n sets</td><td>400 to 999 (*1)
100100 to 800199 (*4)</td><td>100 to 199 *n</td></tr>
<tr><td>7900 + 100 * n sets</td><td>400 to 999 (*1)
100100 to 800199 (*4)
900000 to 907399 (*3)</td><td>100 to 199 *n</td></tr>
<tr><td colspan="2">Local variables</td><td colspan="2">1 to 33</td><td>Can be used as local variables in macro programs.</td></tr>
<tr><td colspan="2">System variable</td><td colspan="2">1000 to</td><td>Application is fixed by system.</td></tr>
<tr><td colspan="2">Fixed cycle variables</td><td colspan="2">1 to 32</td><td>Local variables in fixed cycle programs.</td></tr>
</table>

(*1) Common variable address #400s can only be used when there are 700 or more sets of common variables and the MTB specifications are valid (parameter "#1336 #400_Valtyp").
When common variable address #400s can be used, these can be displayed and set on the common variable screen.
It also becomes possible to input/output data of common variable address #400s.

(*2) When the parameter "#1052 MemVal" is set to "1" in multi-part system (MTB specifications), some or all of common variables "#100 to #199" and "#500 to #999" can be shared and used between part systems. The number of variables sharable in part systems depends on the MTB specifications (parameters "#1303 V1comN" and "#1304 V0comN").

(Example) When "#1304 V0comN" is set to "5":
#500 to #504 : Common for the part systems
#505 to #999 : Each part system

Depending on the MTB specifications, the common variables #100 to #199 are used for each part system, and variables #500 to #999 are common for the part systems (parameter "#1052 MemVal"). Address #400s, that can be used as common variable with 700 or more sets of variable, is common for the part systems regardless of the setting of parameter "#1052 MemVal".

(*3) When “#1052 MemVal” is set to “1”, #900000 to #907399 available for 8,000 sets of variable are not available.

(*4) When the parameter "#1316 CrossCom" is set to "1", the common variables #100100 to #800199 can be shared between the part systems. (This depends on the MTB specifications.) The part system common variable which can be used is shown in the table below.

Variable sets		Common variables 1 (When "#1316 CrossCom" = "1")
Variable sets specification	600 sets (500 + 100 sets)	#100100 to #100199 (Equivalent to # 100 to #199 in 1st part system) #200100 to #200199 (Equivalent to # 100 to #199 in 2nd part system) #300100 to #300199 (Equivalent to # 100 to #199 in 3rd part system) #400100 to #400199 (Equivalent to # 100 to #199 in 4th part system) #500100 to #500199 (Equivalent to # 100 to #199 in 5th part system) #600100 to #600199 (Equivalent to # 100 to #199 in 6th part system) #700100 to #700199 (Equivalent to # 100 to #199 in 7th part system) #800100 to #800199 (Equivalent to # 100 to #199 in 8th part system)
	700 sets (600 + 100 sets)	
	8000 sets (7900 + 100 sets)	

(Example)

<1-part system>

#100100=200 ;	Equivalent to #100 = 200 ;
#200105=#100 ;	"200" is set to #200105.
#300110=#100100 ;	"200" is set to #300110.
#800199=#500120 ;	The variable value of "#500120" is set to #800199.

<Multi-part system>

Common variables for each part system #100 to #199" in other part system can be used.

$1

#200100=-100 ;	"-100" is set to #100 of 2nd part system.
#101=#200102 ;	"#101" is set to #102 of 2nd part system.
#300105=#200103 ;	"#103" of 2nd part system is set to #105 of 3rd part system.
#110=#500107 ;	The variable value of "#500107" is set to #110.

- The PLC data reading function cannot be used, which uses system variables #100100 to #100110, and variables #100100 to #100110 are used as common variables.
- The setting of number of common variables shared between the part systems (The parameter #1052 MemVal" is set to "1") becomes invalid, thus the movement is the same as "0" is set.
- When the parameters "#1128 RstVCl", "#1129 PwrVCl" are set to "1", the operation is as follows.
  “#1128 RstVCl”
  The common variables shared between the part systems equivalent to #100 to #199 of the reset part system are cleared.
  (Example) If the 1st part system is reset, #100100 to #100199 are cleared.
  If the 2nd part system is reset, #200100 to #200199 are cleared.
  “#1129 PwrVCl”
  The common variables shared between the part systems equivalent to #100 to #199 in the valid part system are cleared.
  (Example) In 1st part system, #100100 to #100199 are cleared.
  In 2nd part system, #100100 to #100199 and #200100 to #200199 are cleared.
- Common variables shared between the part systems #100100 to #800199 can be displayed and set on the common variable screen.
- If common variables #100100 to #800199 are used when the number of sets of common variables is less than 600 sets or the parameter "#1316 CrossCom" is "0", a program error (P241) will occur.

Note

(1) 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 that is not common variables such as local variables (#1 to #33) or system variables (#1000 and later)
- Variable data of which the number of common variable sets does not match the pre-specified value

(Example)
If variables of # numbers undefined in the specifications exist in the input file when there are 700 sets of common variables (#100 to #199, #500 to #999, and #100100 to #800199), they are ignored, and only the variables defined in the specifications are input.

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. (“1000” is assumed to be “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 ;

(2) Up to five sets of square parentheses [ ] may be used.
#543 = -[[[[[#120]/2+15.]*3-#100]/#520+#125+#128]*#130+#132]

(3) There are no restrictions on the number of characters and number of variables for variable definition.

(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.

(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

(6) Variable quotations are always regarded as having a decimal point at the end.
When #100 is set to “10”
“X#100;” is set to “X10.”.

Protection of common variable

(1) If the common variable protection function is valid, the common variables in the range specified in the parameters (#12111 to #12114) cannot be changed from machining program or screen operation, or user operation such as file input. This function depends on the MTB specifications (parameter “#1391 User level protect”).

(2) If an attempt is made to change the value or name of the protected variable on the machining program, the program error (P243) will occur, causing the operation to be stop. Such a variable value or name can be changed using the machine tool builder macro program, but cannot be done by the user.
Multiple variable names can be changed in one block with the SETVNn command. However, if at least one of them is protected, the program error (P243) will occur.

(3) If "#1128 RstVCl" is set to "1", the variables (#100 to #199) are cleared after reset even if common variables (#100 to #199) are protected.

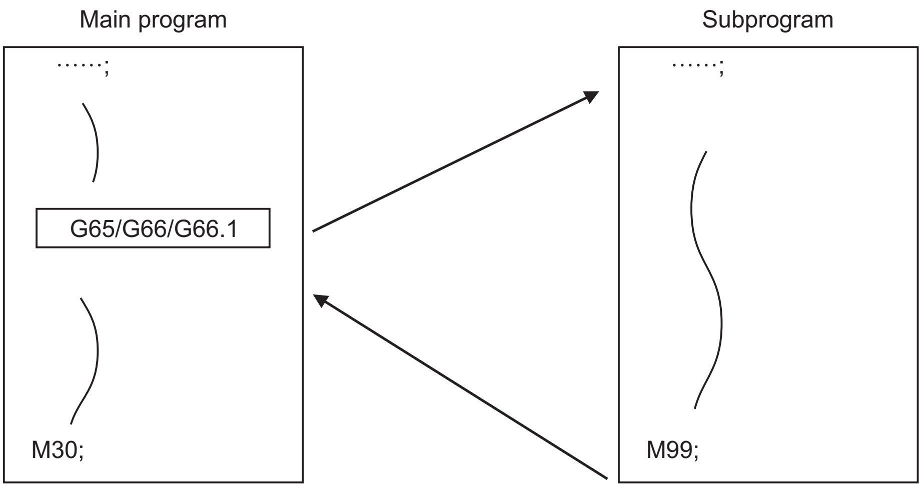
(4) If "#1129 PwrVCl" is set to "1", the variables (#100 to #199) are cleared at the power-ON even if common variables (#100 to #199) are protected.

(5) For common variables used common to the part systems, the variable values and variable names can be changed by the displayed part system.

14.3 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.



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.

14.4 Macro Call Instructions

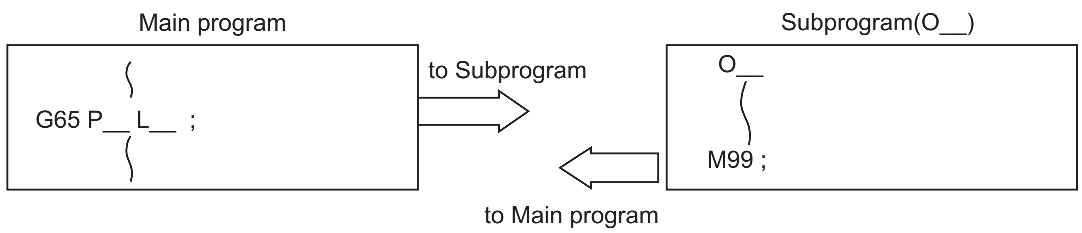
Function and purpose

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.

When the macro argument L/P valid function is enabled, the addresses L (number of subprogram repetitions) and P (calling program No.) used as commands in user macro can be used as arguments. The validity of this parameter depends on the MTB specifications. (Parameter "#1241 set13"/bit5 (Macro argument L/P valid))
When a program in an external device such as a USB memory device is executed, a machining program stored in USB memory cannot be called with a macro call such as G65, G66, or G66.1. Using such a macro calls a macro program in memory.

14.4.1 Simple Macro Calls ; G65

Function and purpose



M99 is used to terminate the user macro subprogram.

Command format

Simple macro calls

G65 P__ L__ argument ;

Simple macro calls

G65 <File name> L__ argument ;

P	Program No. (*1) 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 inside brackets < >. (The file name can have up to 32 characters including the extension.)
L	Number of repetitions (*1) If omitted, this value is set to "1". (0 to 9999)
Argument	Specify variable data

(*1) Can also be used as an argument at the same time as the macro argument L/P valid function is enabled.

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	○	○

○: Available
×: Unavailable
*: 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 designation II address	Variable in macro
A	#1
B	#2
C	#3
I1	#4
J1	#5
K1	#6
I2	#7
J2	#8
K2	#9
I3	#10
J3	#11
K3	#12
I4	#13
J4	#14
K4	#15
I5	#16

Argument designation II address	Variable in macro
J5	#17
K5	#18
I6	#19
J6	#20
K6	#21
I7	#22
J7	#23
K7	#24
I8	#25
J8	#26
K8	#27
I9	#28
J9	#29
K9	#30
I10	#31
J10	#32
K10	#33

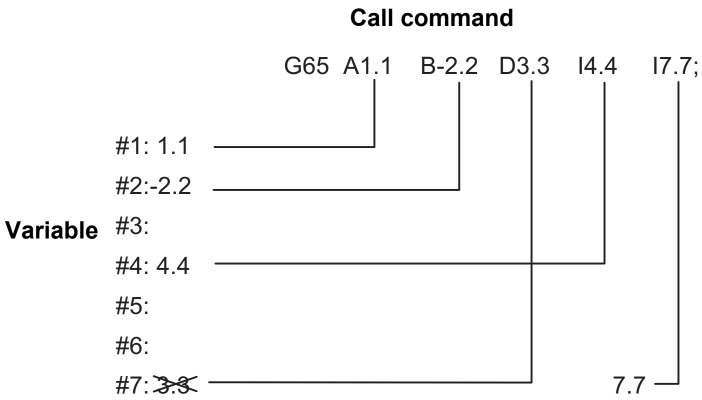
Note

(1) The numbers 1 to 10 accompanying I, J and K indicate the sequence of the commanded sets, and are not required in the actual command.

Using arguments designations I and II together

(1) 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.

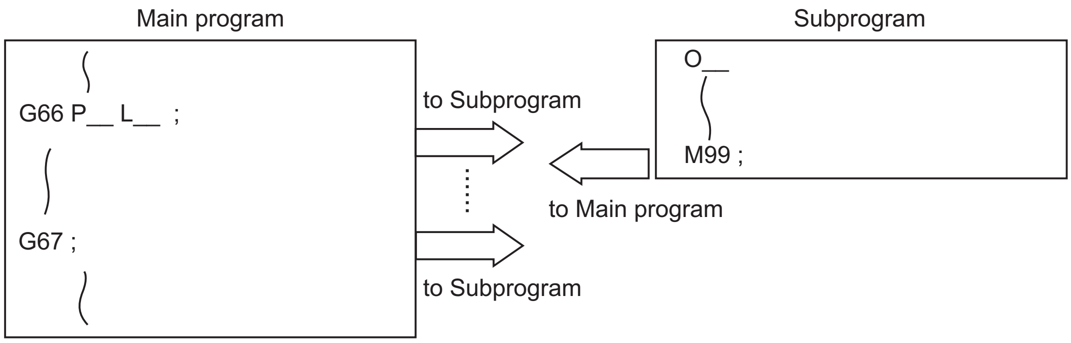
(2) If calling a subprogram numbered with O is enabled, a sub program number starting with O and specified by P command value is called with a priority.
However, when P command value is less than the digit number set with parameter "#8129 subprogram number selection", increase the digit number of command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

14.4.2 Modal Call A (Movement Command Call) ; G66

Function and purpose



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. (*1) 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 inside brackets < >. (The file name can have up to 32 characters including the extension.)
L	Number of repetitions (*1)
Argument	Specify variable data

(*1) Can also be used as an argument at the same time as the macro argument L/P valid function is enabled.

Modal call end

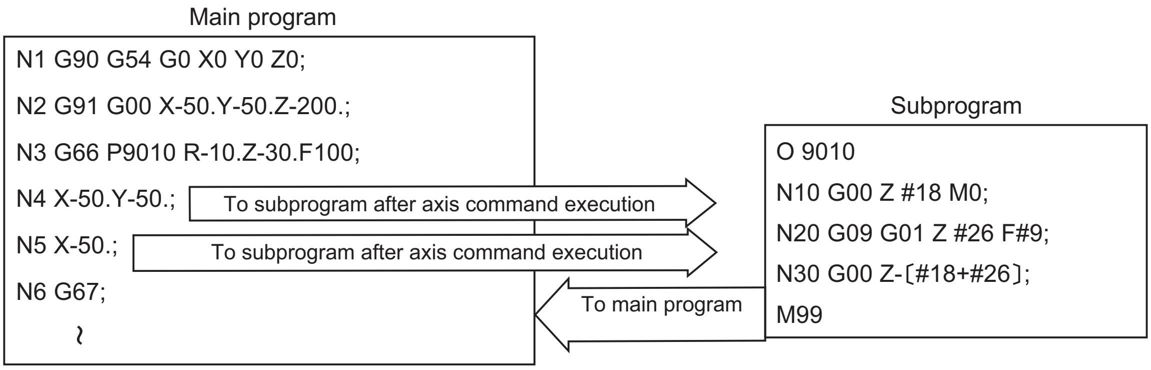
G67 ;

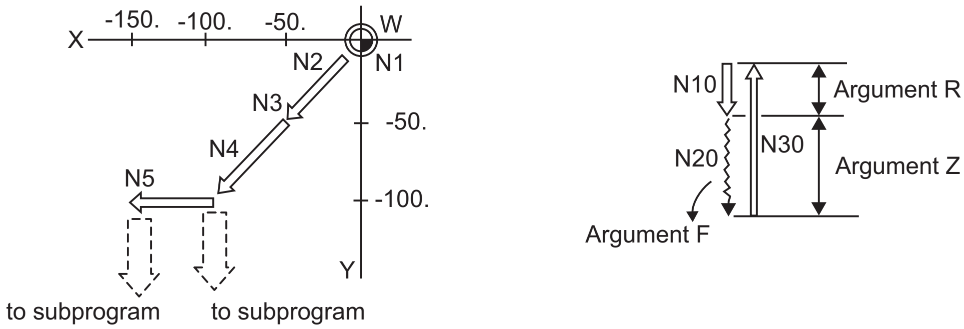
Detailed description

(1) When the G66 command is entered, the specified user macro program will be called after the movement command in a block with the movement commands has been executed, until the G67 (cancel) command is entered.

(2) The G66 and G67 commands must be paired in a same program.
A program error will occur when G67 is issued without G66.

(Example) Drill cycle





<Note>

- After the axis command is executed in the main program, the subprogram is executed.
- The subprogram is not executed in the blocks following G67.

(3) If calling a subprogram numbered with O is enabled, a sub 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.

14.4.3 Modal Call B (for Each Block) ; G66.1

Function and purpose

The specified user macro subprogram is called unconditionally for each command block that 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.
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. (*1) 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 inside brackets < >. (The file name can have up to 32 characters including the extension.)
L	Number of repetitions (*1)
Argument	Specify variable data

(*1) Can also be used as an argument at the same time as the macro argument L/P valid function is enabled.

Modal call end

G67 ;

Detailed description

(1) In the G66.1 mode, everything except the O, N and G codes in the various command blocks which are read are handled as the argument without being executed. Any G code designated last or any N code commanded after anything except O and N will function as the argument.

(2) All significant blocks in the G66.1 mode are handled as when G65P__ is assigned at the head of a block.
(Example 1)
In "G66.1 P1000 ; " mode, “N100 G01 G90 X100. Y200. F400 R1000 ;” is same as “N100 G65 P1000 G01 G90 X100. Y200. F400 R1000 ;”.

<Note>

- 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 that 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 sub 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.

14.4.4 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 code macro call

G P__ L__ argument ;**

G**	G code for macro call
P	(*1)
L	(*1)

(*1) Functions as an argument when the macro argument L/P valid function is enabled.
It cannot be used when the macro argument L/P valid function is disabled.

Detailed description

(1) The above instruction functions in the same way as the instructions below, however, the correspondence between G 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 parameters corresponding to items “c” and “d” 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 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 used in the system can also be used with parameter "#1081 Gmac_P". This parameter setting depends on the MTB specifications.)

<Note>
- G101 to G110 and G200 to G202 are user macro I codes, but if the parameters are set as the G code call codes, the G code call will be the priority, and these codes cannot be used for user macro I.

(4) These commands cannot be issued in a program which has been called by a G code macro. If issued in such a program, they will be handled as ordinary G commands.

(5) When ",D" or "<(Character string)>"is commanded in a block that is calling a G code macro, a miscellaneous command macro, or an ASCII macro while the macro argument L/P valid function is enabled, a program error (P33) will occur.
This parameter setting depends on the MTB specifications. (Parameter "#1241 set13"/bit5)

14.4.5 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

Miscellaneous command macro call

M P__ L__ ; (or S** ; , T** ; , B** ;)**

M**	M code for macro call (or S, T, B code)
P	(*1)
L	(*1)

(*1) Functions as an argument when the macro argument L/P valid function is enabled.
It cannot be used when the macro argument L/P valid function is disabled.

Detailed description

(1) The above instruction functions in the same way as the instructions below, however, the correspondence between M codes and instructions can be set by parameters. (Same for S, T and B codes)

a : M98 P**** ;	M98, M** are not output.
b : G65 P**** M** ;	
c:G66 P**** M** ;	
d:G66.1 P**** M** ;	

When the parameters corresponding to c and d above are set, issue the cancel command (G67) either in the user macro or after the call code has been commanded so as to cancel the modal call.

(2) The correspondence between the "M**" which conducts the macro call and the macro program number P**** to be called is set by parameters. Up to 10 M codes from M00 to M9999 can be registered.
Note that the codes to be registered should exclude those basically required for the machine and the following M codes.
M0, M1, M2, M30, M96, M97, M98, M99, M198, and M codes for G83 specified in the parameter "#8083"

(3) As with M98, it is displayed on the screen display of the setting and display unit but the M codes and MF are not output.

(4) Even if the registered miscellaneous commands above are issued in a user macro subprogram which are called by an M code, it will not be regarded as a macro call and will be handled as a normal miscellaneous command. (Same for S, T and B codes)

(5) All S, T and B codes call the subprograms in the prescribed program numbers of the corresponding S, T and B functions.

(6) Up to 10 M codes can be set.

<Note>

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

  (Example) To register M06 in M code macro

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

(7) When ",D" or "<(Character string)>"is commanded in a block that is calling a G code macro, a miscellaneous command macro, or an ASCII macro while the macro argument L/P valid function is enabled, a program error (P33) will occur.
This parameter setting depends on the MTB specifications. (Parameter "#1241 set13"/bit5)

14.4.6 Detailed Description for Macro Call Instruction

Detailed description

Differences between M98 and G65 commands

(1) The argument can be designated for G65 but not for M98.

(2) The sequence number can be designated for M98, but not for G65, G66 and G66.1.

(3) M98 executes subprograms after all the commands except M, P, H and L in the M98 block are executed, but G65 branches directly to the subprogram without any further operation.

(4) When any address except O, N, P, H or L is included in the M98 block, the single block stop will be conducted, but not for the G65.

(5) The level of the M98 local variables is fixed but it varies in accordance with the nesting depth for G65. ("#1" before and after M98, for instance, has the same significance, but they have different significance in G65.)

(6) The M98 nesting depth extends up to 10 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

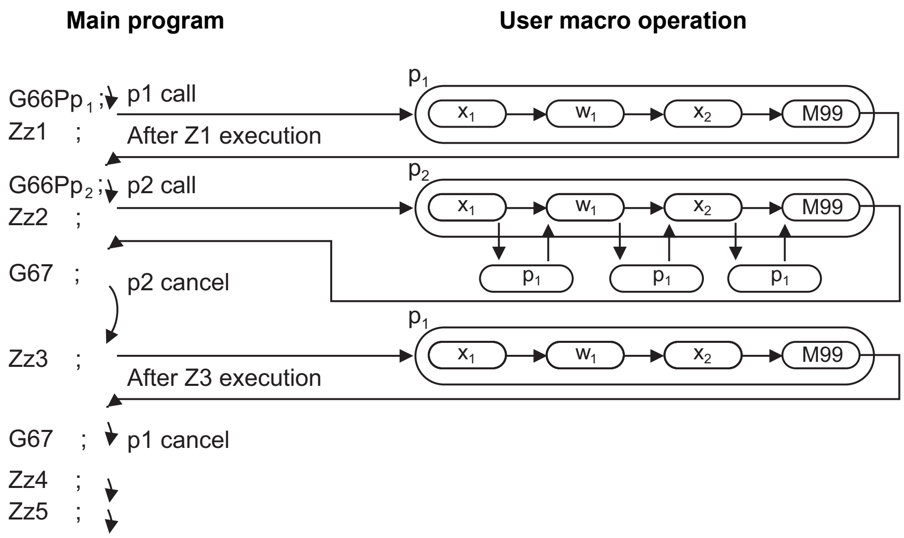
Up to 4 nesting levels are available for macro subprogram calls by simple call or modal call.
The argument for a macro call instruction is valid only within the called macro level. Since the nesting depth for macro calls extends up to 4 levels, the argument can be used as a local variable for the programs 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.

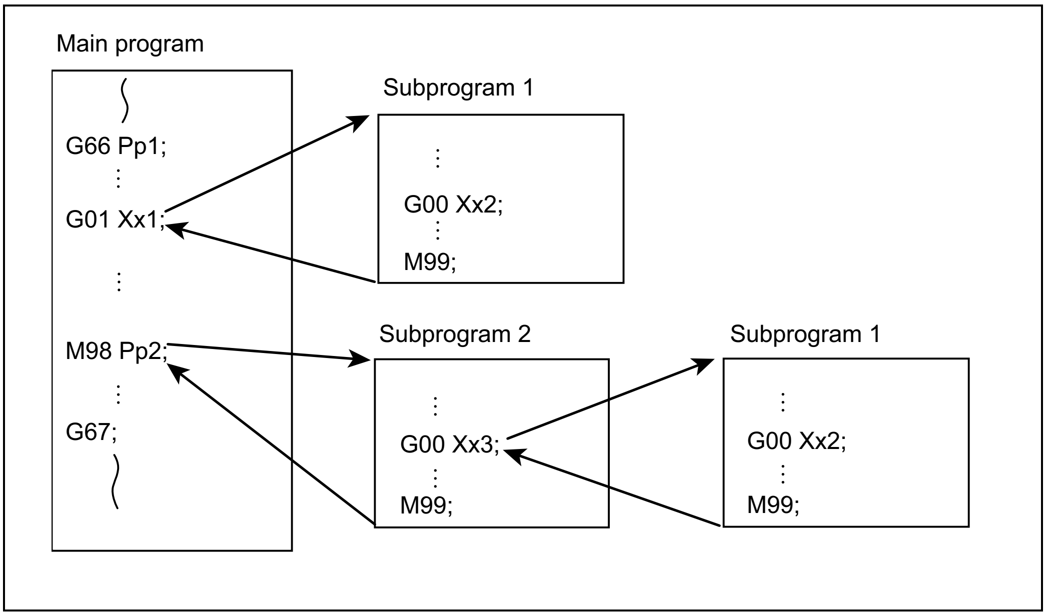
(2) With modal call A, the designated user macro subprogram is called every time a movement command is executed. However, when the G66 command is duplicated, the next user macro subprogram is called to movement commands in the macro every time an axis is moved.
User macro subprograms are called from the one commanded last.

(Example 1)



(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)



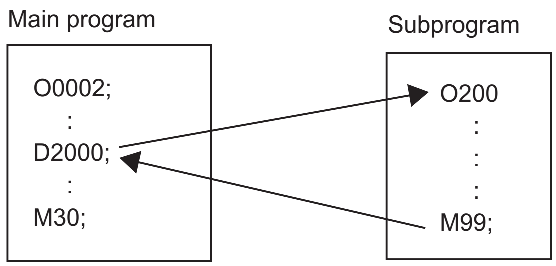
When the program numbers of p1 and p2 are same, the program numbers of subprograms 1 and 2 will be same.

14.4.7 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.
These parameters depend on the MTB specifications.

(Execution example 1) M98 type

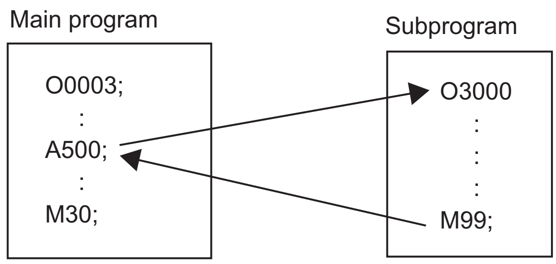


After outputting 2000 to common variable #146, the program No. 200 subprogram is called with the M98 subprogram call type.

Parameters

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

(Execution example 2) G65 type



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

Parameters

#7411 (ASCII call Valid/Invalid)	1 (Valid)
#7412 (ASCII code)	A
#7413 (Call type)	1 (G65 type)
#7414 (ASCII [01] Program No.)	3000
#7415 (ASCII [01] Variable)	100 (Not used)

Command format

□**** P__ L__ ; ... Designates the address and code

□	ASCII code for calling out a macro (one character)
****	Value or expression output to variable (Setting range: ±999999.9999)
P	(*1)
L	(*1)

(*1) Functions as an argument when the macro argument L/P valid function is enabled.
It cannot be used when the macro argument L/P valid function is disabled.

Detailed description

(1) The above command performs the same operations as the 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 items "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). (These parameters depend on the MTB specifications.)

(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

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

Precautions

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

A macro cannot be called with an ASCII code from a macro-called program with an ASCII code.
The other patterns are shown below.
If it is judged that a macro cannot be called, the command will be handled as a normal command.

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

Nest level of macro call commands

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 subprogram call command

Counting the main program as 0, up to ten 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 cycles
(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

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

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

If the other addresses do not correspond to the ASCII code macro command, they will be identified as normal commands. If the command to be used, overlaps with an ASCII code macro command, it must be commanded in the macro-called program with the ASCII code.
Note that there are cases where the command will be unconditionally handled as a normal command, as explained in below.

Conditions where the address set is handled as a normal command

(1) When there is a data setting command (G10) in the same block.

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

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

(3) When inputting parameters

(4) When there is a comma (,) before the address. Example) ,D ,R, etc.

(5) When commanded in fixed cycle

(6) When commanded in macro subprogram called with G code macro call
(Also applies when macro is called with M, S, T, B or ASCII)

(7) When ",D" or "<(Character string)>"is commanded in a block that is calling a G code macro, a miscellaneous command macro, or an ASCII macro while the macro argument L/P valid function is enabled, a program error (P33) will occur.
This parameter setting depends on the MTB specifications. (Parameter "#1241 set13"/bit5)

14.5 Variables Used in User Macros

Function and purpose

Both the variable specifications and user macro specifications are required for the variables that 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 PwrVCl".)

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>

G00 X#1 Z1000 ;	Equivalent to G00 Z1000 ;
G00 X#1+10 Z1000 ;	Equivalent to G00 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> >= <Blank> 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 compared only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.

14.5.1 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

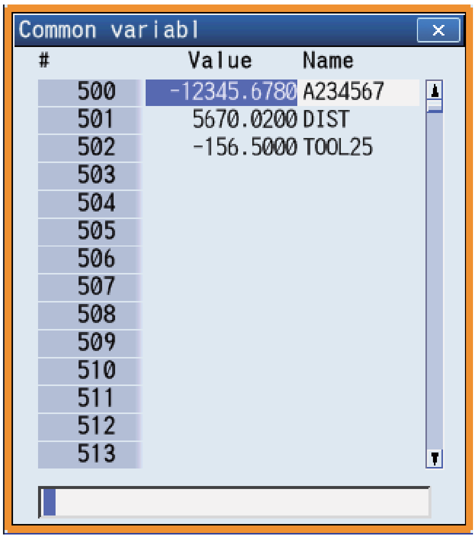
Any name (variable name) can be given to common variables #100 to #199 and #500 to #599. It must be composed of not more than 7 alphanumerics and it must begin with a letter. Do not use "#" in variable names. It causes an alarm when the program is executed.

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] ;



Note

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

14.5.2 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 correspondences points 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.
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
Δ	Δ *	L	#12
○	○	M	#13
×	× *	N	#14
×	×	O	#15
Δ	Δ *	P	#16

Call command		Argument address	Local variable No.
G65 G66	G66.1		
○	○	Q	#17
○	○	R	#18
○	○	S	#19
○	○	T	#20
○	○	U	#21
○	○	V	#22
○	○	W	#23
○	○	X	#24
○	○	Y	#25
○	○	Z	#26
		-	#27
		-	#28
		-	#29
		-	#30
		-	#31
		-	#32
		-	#33

"×" in the above table denotes 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.
A "Δ" mark denotes an argument address which may be used depending on the MTB specifications. (Parameter "#1241 set13"/bit5)

[Argument designation II]

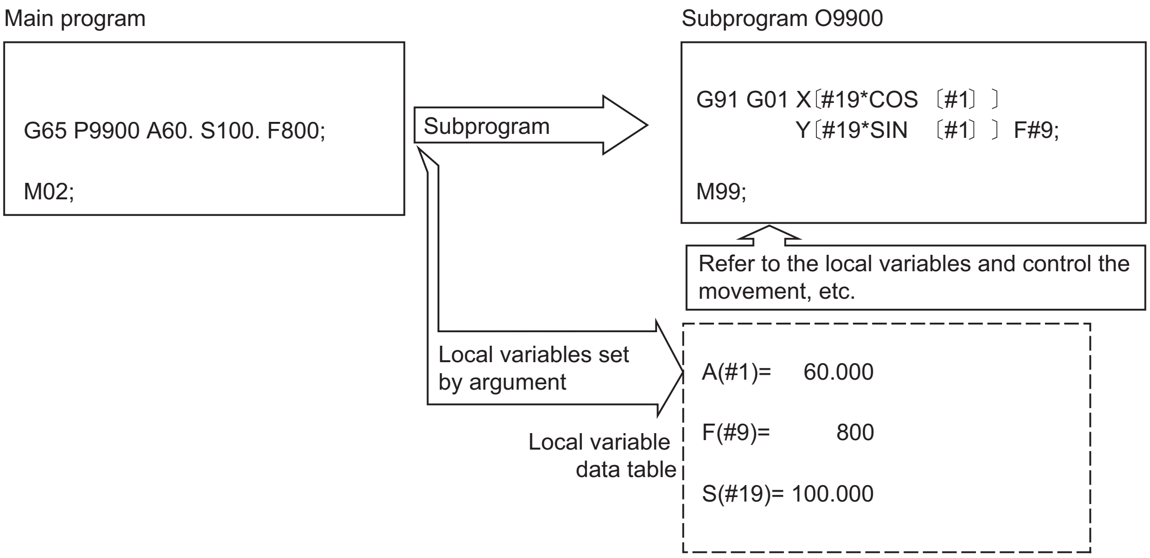
Argument designation II address	Variable in macro
A	#1
B	#2
C	#3
I1	#4
J1	#5
K1	#6
I2	#7
J2	#8
K2	#9
I3	#10
J3	#11
K3	#12
I4	#13
J4	#14
K4	#15
I5	#16

Argument designation II address	Variable in macro
J5	#17
K5	#18
I6	#19
J6	#20
K6	#21
I7	#22
J7	#23
K7	#24
I8	#25
J8	#26
K8	#27
I9	#28
J9	#29
K9	#30
I10	#31
J10	#32
K10	#33

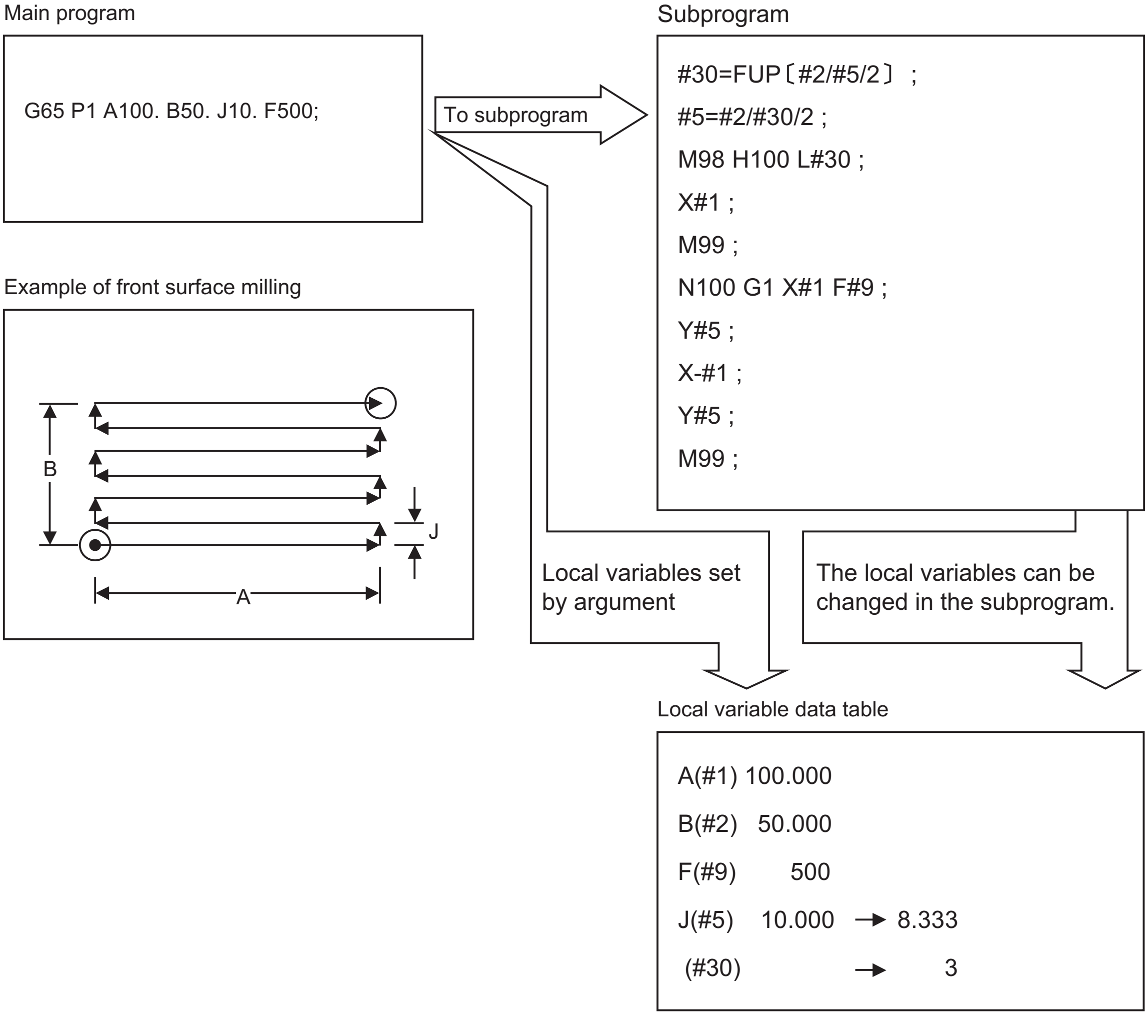
Note

- 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.)

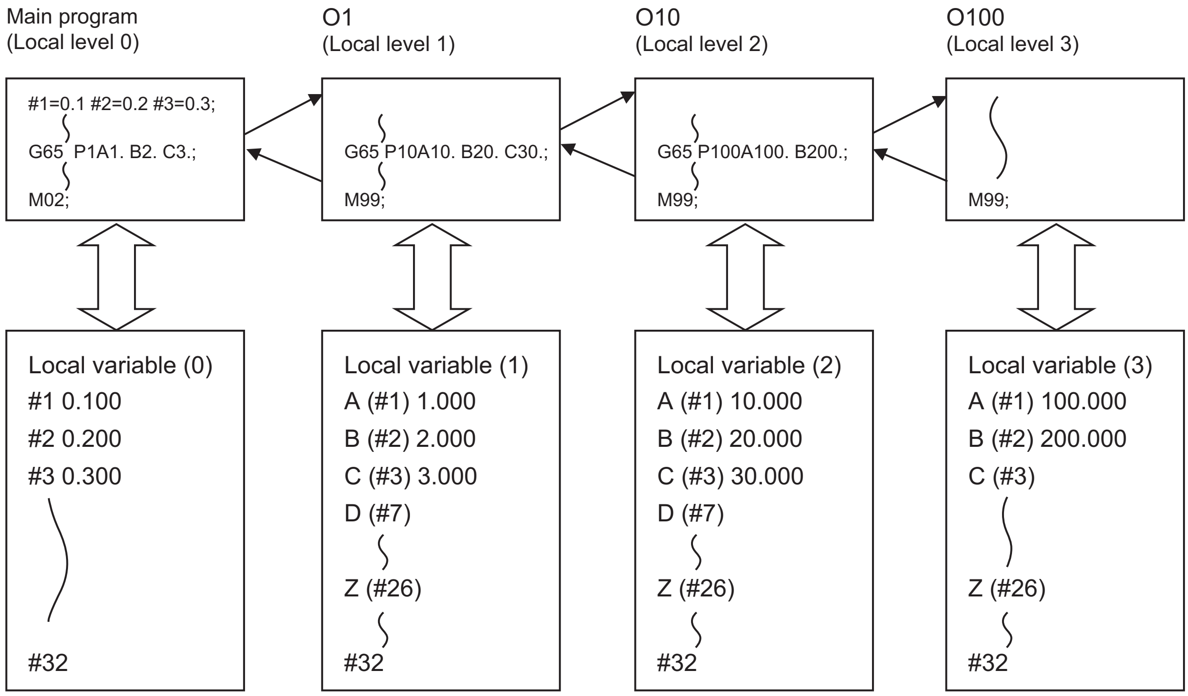


(2) The local variables can be used freely in that subprogram.



In the front surface milling example, argument J is programmed as the milling pitch 10. mm. However, this is changed to 8.333 mm to create an equal interval pitch.
The results of the No. of reciprocation data calculation is set in local variable #30.

(3) Local variables can be used independently on each of the macro call levels (4 levels).
Local variables are also provided independently for the main program (macro level 0).
Arguments cannot be used for the level 0 local variables.



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

14.5.3 System Variables

Data such as the workpiece offset amount can be read using system variables other than common variables or local variables.
Refer to "22 System Variables" for details.

14.6 User Macro Commands

14.6.1 Operation Commands

Function and purpose

A variety of operations can be performed between variables.

Command format

#i = <formula> ;

<Formula> is a combination of constants, variables, functions and operators.
Constants can be used instead of #j and #k below.

(1) Definition and substitution of variables	#i = #j	Definition, substitution
(2) Addition operation	#i = #j + #k	Addition
	#i = #j - #k	Subtraction
	#i = #j OR #k	Logical sum (at every bit of 32 bits)
	#i = #j XOR #k	Exclusive OR (at every bit of 32 bits)
(3) Multiplication operation	#i = #j * #k	Multiplication
	#i = #j / #k	Division
	#i = #j MOD #k	Remainder
	#i = #j AND #k	Logical product (at every bit of 32 bits)
(4) Functions	#i = SIN [#k]	Sine
	#i = COS [#k]	Cosine
	#i = TAN [#k]	Tangent tan θ uses sinθ/cosθ.
	#i = ASIN [#k]	Arcsine
	#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]	Discarding fractions after decimal point
	#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).

(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

(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〕

(a) Function
(b) Multiplication operation
(c) Addition operation

(2) The part to be given priority in the operation sequence should be enclosed in square parentheses [ ]. Up to 5 pairs of such parentheses, including those for the functions, may be used.

#101=SQRT〔〔〔#111-#112〕*SIN〔#113〕+#114〕*#115〕;

First pair of brackets
Second pair of brackets
Third pair of brackets

Examples of operation commands

(1) Main program and argument designation	G65 P100 A10 B20.; #101 = 100.000 #102 = 200.000;	#1 10.000 #2 20.000 #101 100.000 #102 200.000	
(2) Definition and 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 tool compensation
(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 Remainder 3 #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) Sine 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]

14.6.2 Control Commands

Function and purpose

The flow of programs can be controlled by IF-GOTO- and WHILE-DO-END.
When a program in an external device such as a USB memory device is executed, a period of processing time is required in the subprogram call or in the instruction change the flow of the program such as GOTO or DO-END; therefore, interpolation may be decelerated or stopped.

Detailed description

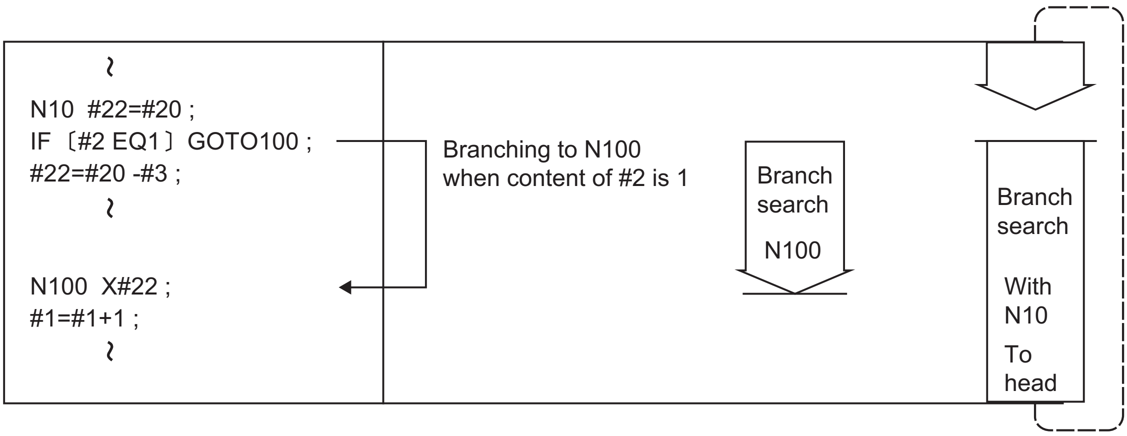
Branching

IF [conditional expression] GOTO n; (n = sequence number in the program)

When the condition is satisfied, control branches to "n" and when it is not satisfied, the next block is executed.
IF [conditional expression] can be omitted and, when it is, control branches to "n" unconditionally.
The following types of [conditional expressions] are available.

#i EQ #j	= When #i and #j are equal
#i NE #j	≠ When #i and #j are not equal
#i GT #j	> When #i is greater than #j
#i LT #j	< When #i is less than #j
#i GE #j	>= When #i is #j or more
#i LE #j	<= When #i is #j or less

"n" of "GOTO n" must always be in the same program. If not, program error (P231) will occur. A formula or variable can be used instead of i, #j and n.
In the block with sequence number "n" which will be executed after a "GOTO n" command, the sequence number "Nn" must always be at the head of the block. Otherwise, program error (P231) will occur.
If "/" is at the head of the block and "Nn" follows, control can be branched to the sequence number.



Note

(1) When searching the sequence number of the branch destination, the search is conducted up to the end of the program (% code) from the block following IF............; and if it is not found, it is then conducted from the top of the program to the block before IF............;. Therefore, branch searches in the opposite direction to the program flow will take longer time compared with branch searches in the forward direction.

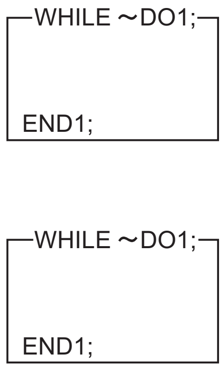
(2) EQ and NE should be compared only for integers. For comparison of numeric values with decimals, GE, GT, LE, and LT should be used.

Repetitions

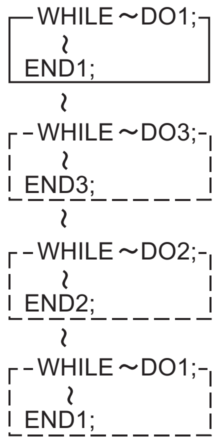
WHILE [conditional expression] DOm ; (m =1, 2, 3 ..... 127) :
:
END m ;

While the conditional expression is established, the blocks from the following block to ENDm are repeatedly executed; when it is not established, execution moves to the block following ENDm. DOm may come before WHILE. "WHILE [conditional expression] DOm" and "ENDm" must be used as a pair. If "WHILE [conditional expression]" is omitted, these blocks will be repeatedly ad infinitum. The repeating identification Nos. range from 1 to 127. (DO1, DO2, DO3, ...DO127) Up to 27 nesting levels can be used.

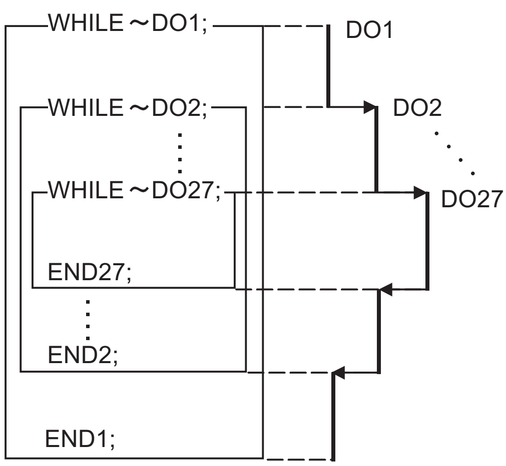
(1) Same identification No. can be used any number of times.



(2) Any number may be used as the WHILE-DOm identification No.



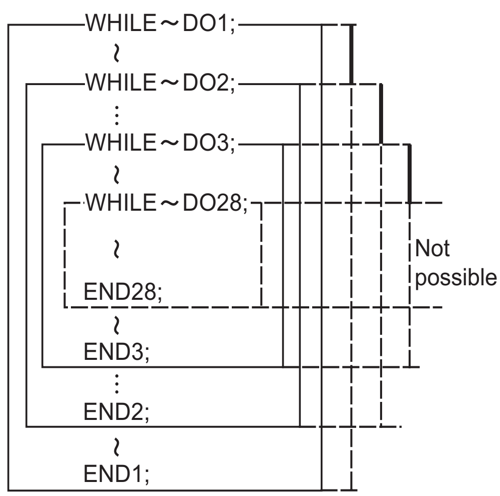
(3) Up to 27 nesting levels can be used for WHILE-DOm. "m" is any number from 1 to 127 for the nesting depth.



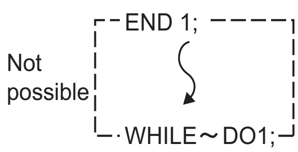
<Note>

- For nesting, "m" which has been used once cannot be used.

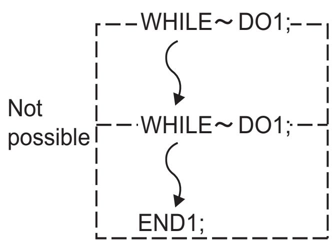
(4) The number of WHILE-DOm nesting levels cannot exceed 27.



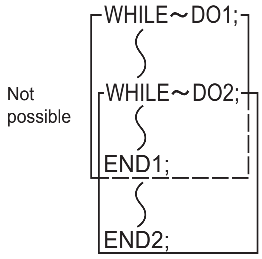
(5) WHILE - DOm must be designated first and ENDm last.



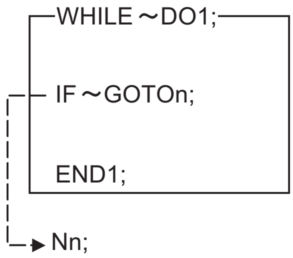
(6) WHILE - DOm and ENDm must correspond on a 1:1 (pairing) basis in a same program.



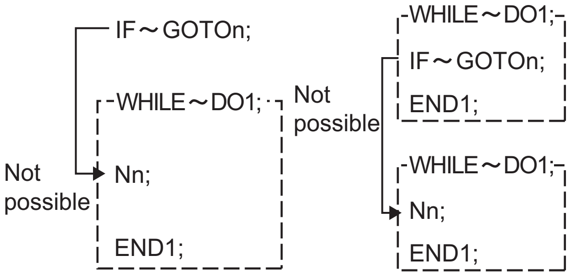
(7) Two WHILE - DOm's must not overlap.



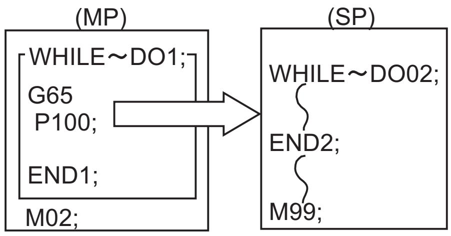
(8) Branching externally out of the WHILE - DOm range, is possible.



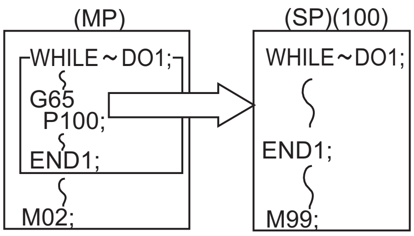
(9) No branching into WHILE - DOm, is possible.



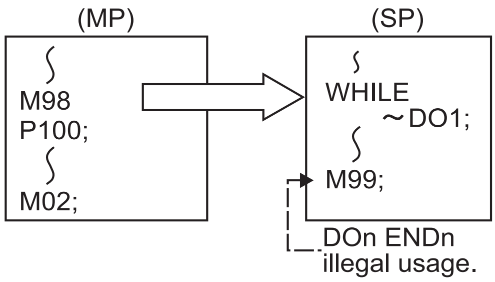
(10) Subprograms can be called by M98, G65 or G66 between WHILE - DOm's.



(11) Calls can be initiated by G65 or G66 between WHILE - DOm's and commands can be issued again from 1. Up to 27 nesting levels are possible for the main program and subprograms.



(12) A program error will occur in M99 if WHILE and END are not paired in the subprogram (including macro subprogram).



(MP) Main program (SP) Subprogram

Note

- Even if a fixed cycle containing WHILE is called, the nesting level will be counted up.

14.6.3 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 or memory card.

Command format

Open command

POPEN

Closed command

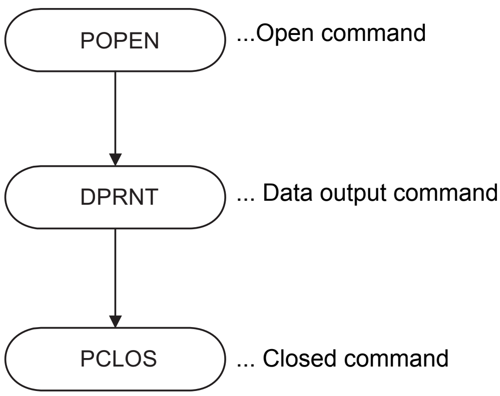
PCLOS

Data output command

DPRNT

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.

Note

- A data output command can be issued even in two-part system mode. In this case, however, note that the output channel is shared by both part systems. So, be careful not to execute data output by both part systems simultaneously.

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.

Use example:

<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[]; (*1)
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;
%
```

(*1) Designate a blank line when opening an output file on an edit screen. If not, it will be regarded that no information is provided in the head line of the file on the NC edit screen.

<Output file (file name = DPRNT_OUT)>

Blank

Linefeed code

(CR)	(LF)											
V	A	L	-	C	H	E	C	K	(CR)	(LF)		
1	2	3	4	5	6	7	8	9	0	(CR)	(LF)	
			1	2	.	3	4	0	0	(CR)	(LF)	
				0	.	0	0	0	0	(CR)	(LF)	
-	2	3	4	5	6	7	8	9	(CR)	(LF)		
							-	0	(CR)	(LF)		
-	.	0	0	0	0	0	0	0	0	(CR)	(LF)	
-	.	1	2	3	4	5	6	7	9	(CR)	(LF)	

Values above the number of significant figures are rounded down

Values below the number of significant figures are rounded off

Precautions

(1) An external output command during restart-searching is ignored.
After using the restart-search type 2 to restart-search between a POPEN command and a PCLOS command, execute a POPEN command by such as an MDI interruption before restarting the program.

(2) An external output command during graphic check is ignored.

(3) A program error (P460) will occur if an external output command is issued when the output device is unable to output due to a lack of connection, a low free space etc.

(4) The NC automatically conducts a closing processing when it is reset between a POPEN command and a PCLOS command. So, execute a POPEN command by such as an MDI interruption before executing the rest of the machining program.

(5) If a program error occurs between a POPEN command and a PCLOS command, NC will not automatically conduct a closing processing. So, there is no need to execute a POPEN command by such as an MDI interruption before executing the rest of the machining program.

(6) If a program error occurs to the output port due to the setting of the memory card, execute an NC reset and close the output file before demounting the card.

(7) When the output port is a memory card, the output file may be destroyed if the card is dismounted or the power is turned off without issuing a PCLOS command or NC reset after a POPEN command is issued.

(8) As for M800 series, output data of an external output command can be output to a memory card only when the drive name of the card is "E:" or "F:". Drive name "E" is given the priority. A program error (P460) will occur if the output port executes the external output command of the memory card when the drive name is neither "E:" nor "F:".

(9) When the data is output to a memory card, the maximum number of files that can be created is determined by the FAT16 format.

14.6.4 Precautions

Precautions

When the user macro commands are employed, it is possible to use the M, S, T and other NC control commands together with the arithmetic, decision, branching and other macro commands for preparing the machining programs. When the former commands are made into executable statements and the latter commands into macro statements, the macro statement processing should be accomplished as quickly as possible in order to minimize the machining time, because such processing is not directly related to machine control.
By setting the parameter "#8101 MACRO SINGLE", the macro statements can be processed concurrently with the execution of the executable statement.
(During normal machining, set the parameter OFF to process all the macro statements together, and during a program check, set it ON to execute the macro statements block by block. Setting can be chosen depending on the purpose.)
There is a signal to notify this macro single setting status to the sequence program. This signal is set ON when "1" (valid) is set to the control parameter "#8101 MACRO SINGLE". (Only for C80 series)
The operation of the PLC signal depends on the MTB specifications.

Program example

N1 G91 G28 X0 Y0 ;	......(1)	
N2 G92 X0 Y0 ;	......(2)	
N3 G00 X-100. Y-100. ;	......(3)	
N4 #101 = 100. * COS[210.] ;	......(4)	(4),(5) Macro statements
N5 #103 = 100. * SIN[210.] ;	......(5)	
N6 G01 X#101 Y#103 F800 ;	......(6)	

Macro statements are:

(a) Arithmetic commands (block including =)

(b) Control commands (block including GOTO, DO-END, etc.)

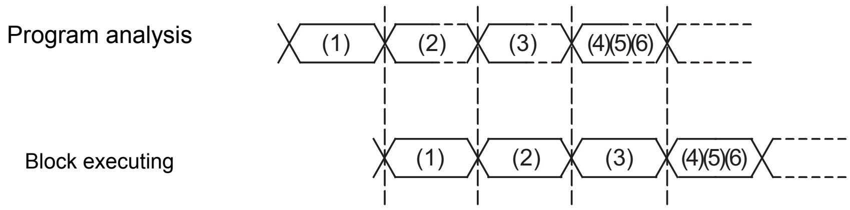
(c) Macro call commands (including macro calls based on G codes and cancel commands (G65, G66, G66.1, G67))

Execution statements refer to statements other than macro statements.

Flow of processing by the Program Example in the previous page

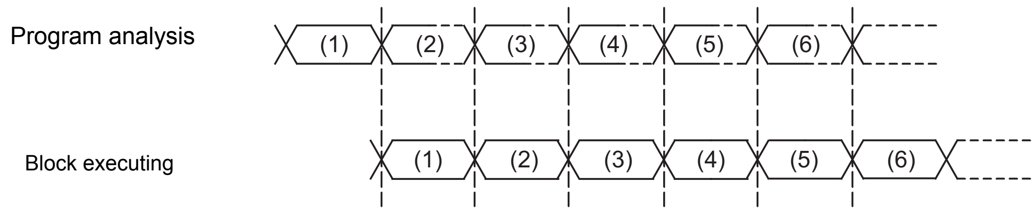
<Macro single OFF>

N4, N5 and N6 are processed in parallel with the control of the executable statement of N3. If the N4, N5 and N6 analysis is in time during N3 control, the machine movement will be continuously controlled.



<Macro single ON>

N4 is processed in parallel with the control of the executable statement of N3. After N3 is finished, N5 and N6 are analyzed, and then N6 is executed. So the machine control is held on standby during the N5 and N6 analysis time.



14.6.5 Actual Examples of Using User Macros

Program example

The following three examples will be described.

(Example 1) SIN curve

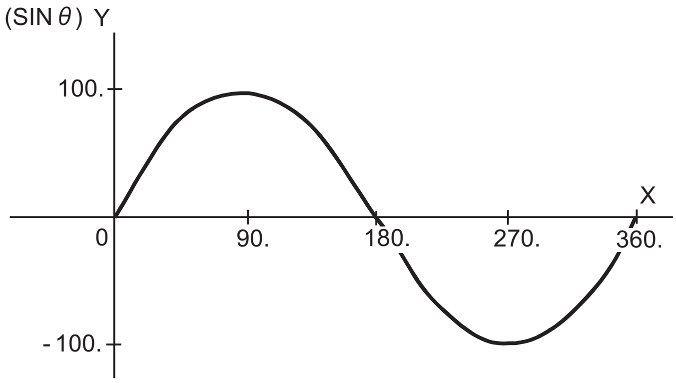
(Example 2) Bolt hole circle

(Example 3) Grid

(Example 1) SIN curve

G65 Pp1 Aa1 Bb1 Cc1 Ff1;

a1; Initial value 0°
b1; Final value 360°
c1; R of R*SINθ
f1; Feedrate



Main Program

```
∫
G65 P9910 A0 B360.C100.F100;
∫
```

To subprogram

O9910(Subprogram)

```
WHILE〔#1LE#2〕DO1;
#101=#3*SIN〔#1〕;
G90 G01 X#1 Y#10 F#9;
#1=#1+10.;
END1;
M99;
```

(Note 1)

Local variables set by argument

```
#1=0
#2=360.000
#3=100.000
#9=100.000
```

(Note 1)Commanding with one block is possbiel when 90G01X#1Y[#3*SIN[#1]] F#9; is issued.

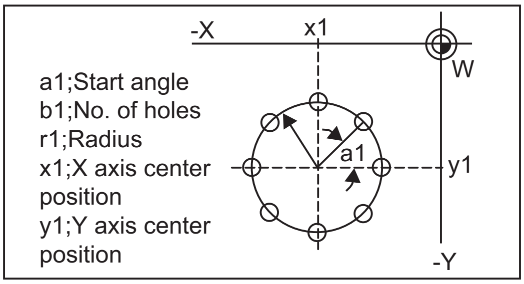
(Example 2) Bolt hole circle

After defining the hole data with fixed cycle (G72 to G89), the macro command is issued as the hole position command.

Main program

```
        ~
G81 Z-100.R50.F300L0
G65 P9920 Aa1 Bb1 Rr1 Xx1 Yy1;
        ~
```

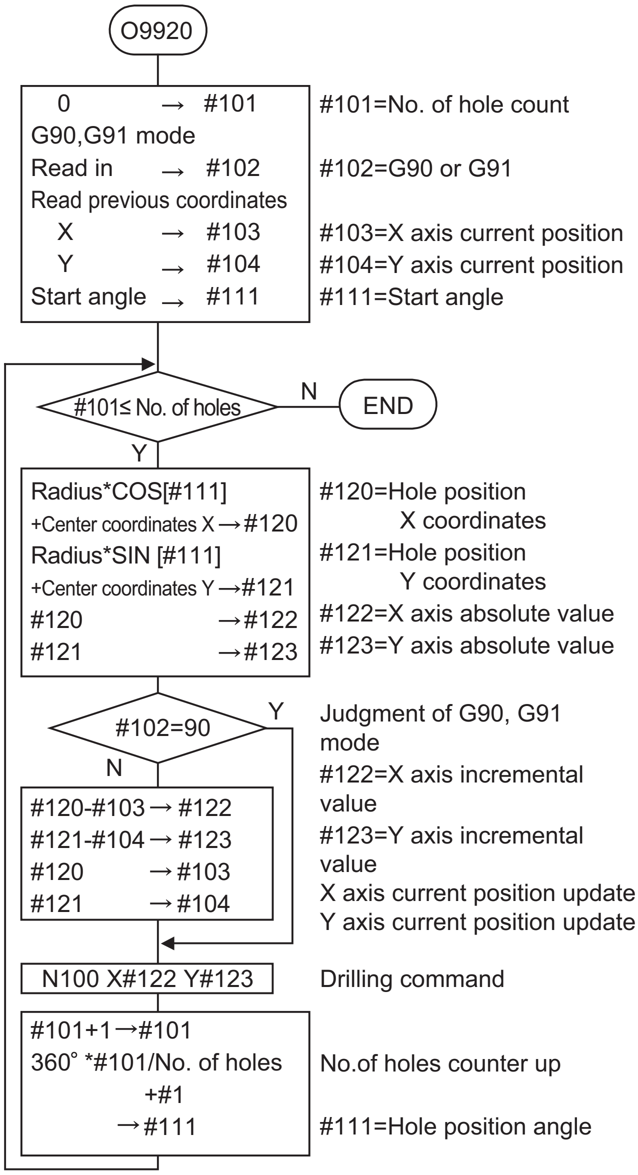
To subprogram



O9920(Subprogram)

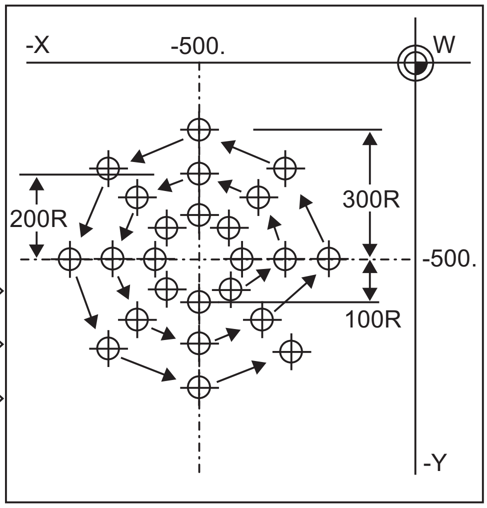
```
#101=0;          }
#102=#4003;      }
#103=#5001;      } (Note 1)
#104=#5002;      }
#111=#1;         }

WHILE〔#101LT#2〕DO1;

#120=#24+#18*COS〔#111〕;  } (Note 1)
#121=#25+#18*SIN 〔#111〕; }

#122=#120 #123=#121;
IF〔#102EQ90〕GOTO100;

#122=#120-#103;  } (Note 1)
#123=#121-#104;  }

#103=#120;  } (Note 1)
#104=#121;  }

N100 X#122 Y#123;
#101=#101+1;              } (Note 1)
#111=#1+360.*#101/#2;     }

END1;
M99;
```

(Note 1) The processing time can be shortened by programming in one block.



```
G28 X0 Y0 Z0;
T1 M06;
G90 G43 Z100.H01;
G54 G00 X0 Y0;
G81 Z-100.R3.F100 L0 M03;
G65 P9920 X-500.Y-500.A0 B8 R100.;
G65 P9920 X-500.Y-500.A0 B8 R200.;
G65 P9920 X-500.Y-500.A0 B8 R300.;
  ~
```

To subprogram
To subprogram
To subprogram

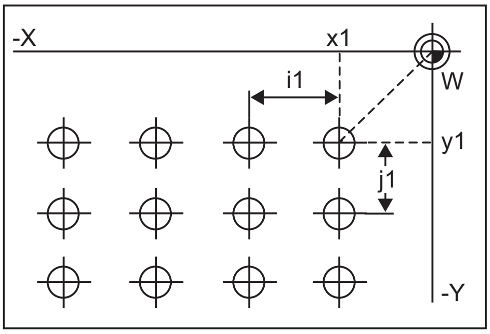


(Example 3) Grid

After defining the hole data with the fixed cycle (G72 to G89), macro call is commanded as a hole position command.

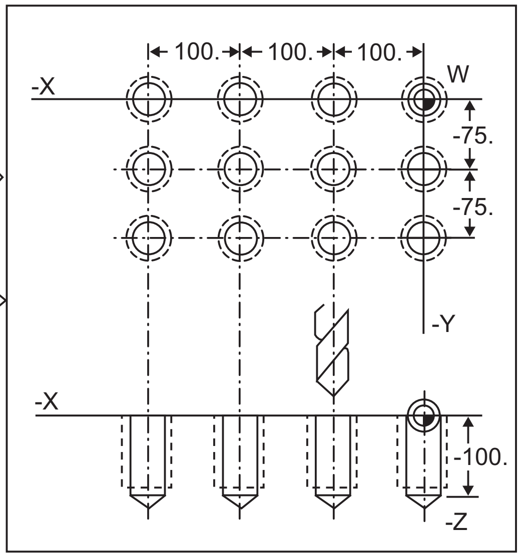
```
G81 Zz1 Rr1 Ff1;
G65 Pp1 Xx1 Yy1 Ii1 Jj1 Aa1 Bb1;
      x1; X axis hole position
      y1; Y axis hole position
      i1; X axis interval
      j1; Y axis interval
      a1; No. of holes in X direction
      b1; No. of holes in Y direction
```

Subprogram is on next page




```
G28 X0 Y0 Z0;
T1 M06;
G90 G43 Z100.H01;
G54 G00 X0 Y0;
G81 Z-100.R3.F100 L0 M03;
G65 P9930 X0 Y0 I-100. J-75. A5B3;
    ~
G84 Z-90. R3. F250 M03;
G65 P9930 X0 I-100. J-75. A5B3;
    ~
```

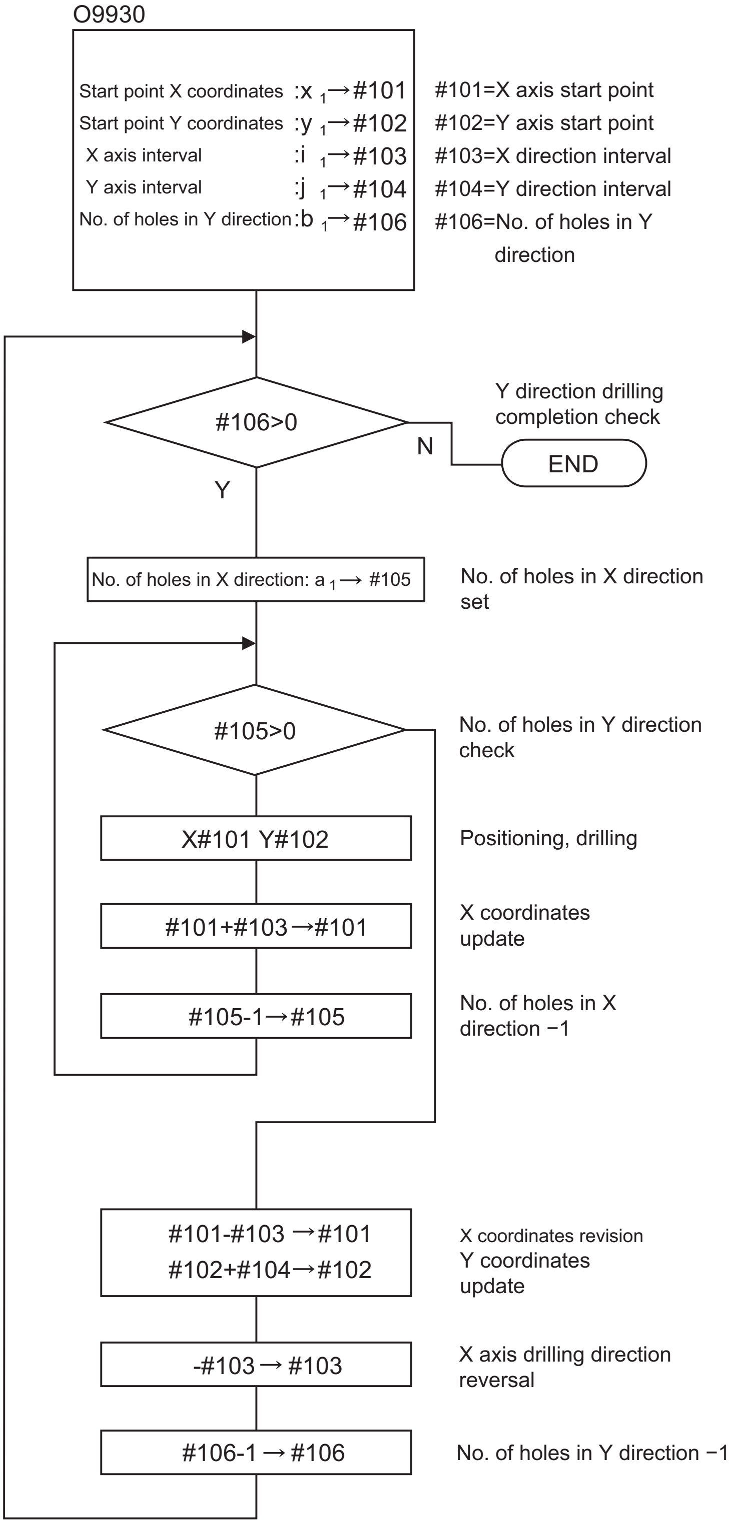
To subprogram
To subprogram



O9930(Subprogram)

```
      #101=#24;
      #102=#25;
                          (Note 1)
      #103=#4;
      #104=#5;

      #106=#2;

      WHILE〔#106GT0〕DO1;

      #105=#1;

      WHILE〔#105GT0〕DO2;
G90 X#101 Y#102;

      #101=#101+#103;
      #105=#105-1;        (Note 1)

      END2;

      #101=#101-#103;
      #102=#102+#104;
                          (Note 1)
      #103=-#103;
      #106=#106-1;

      END1;

      M99;
```

(Note 1) The processing time can be shortened by programming in one block.



14.7 Macro Interruption; M96, M97

Function and purpose

A user macro interrupt signal (UIT) is input from the machine to interrupt the program currently being executed, and instead calls and executes another program. 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 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 until M96 is commanded while one user macro interrupt is 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".
- An interrupt program starting with commanded O No. does not exist.

Enabling conditions

A user macro interruption is enabled only during execution of a program.
The enabling conditions are as follows:

(1) An automatic operation mode or MDI has been selected.

(2) The system is running in automatic mode.

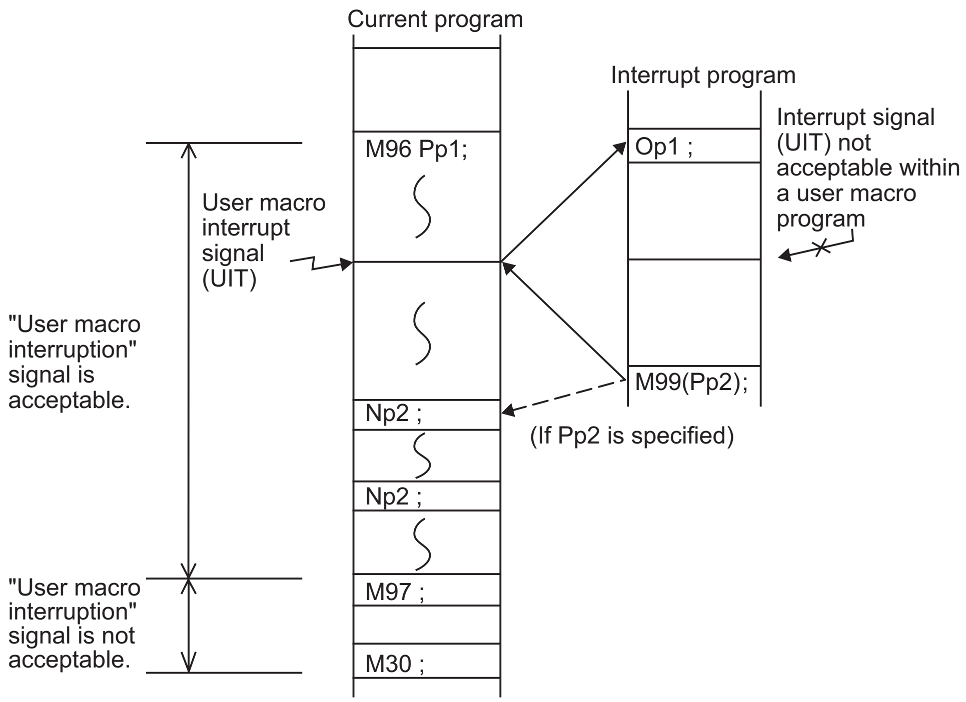
(3) No user macro interruption is being processed.

(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 M99 Pp2; is specified, the blocks from the one next to the interrupted block to the last one are searched. If none is found, blocks between the first block of the program and the one before the interrupted block are searched. Control then returns to the block with sequence number Np2 that is found first in the above search.



Interrupt type

Interrupt types 1 and 2 can be selected by the parameter "#1113 INT_2".

[Type 1]

(1) When an interrupt signal (UIT) is input, the system immediately stops moving the tool and interrupts dwell, then permits the interrupt program to run.

(2) If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are lost. After the interrupt program completes, the main program resumes operation from the block next to the interrupted one.

(3) If the interrupted program contains no move and miscellaneous (MSTB) commands, it resumes operation, after completion of the interrupt program, from the point in the block where the interrupt was caused.

If an interrupt signal (UIT) is input during execution of a miscellaneous function (MSTB) command, the NC system waits for a completion signal (FIN). The system thus executes a move or miscellaneous function command (MSTB) in the interrupt program only after input of FIN.

[Type 2]

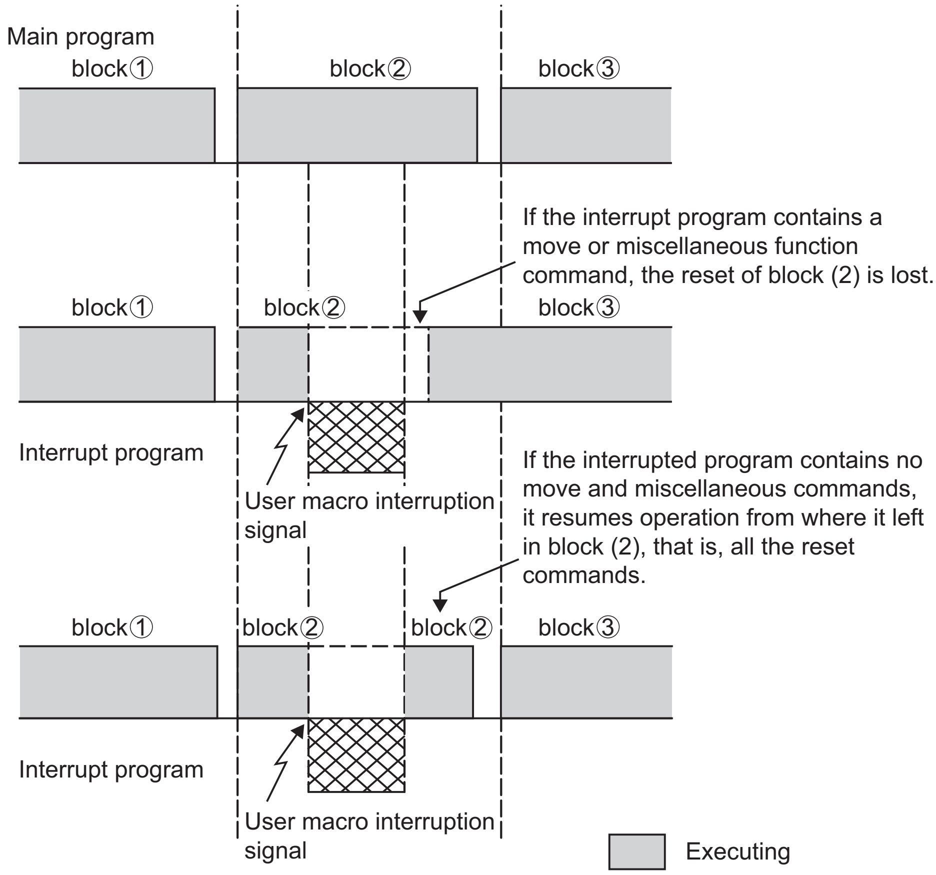
(1) When an interrupt signal (UIT) is input, the interrupt program will be executed without interrupting execution of the current block in parallel with the executing block.

(2) If the interrupt program contains a move or miscellaneous function (MSTB) command, the commands in the interrupted block are completed, then, these commands will be executed.

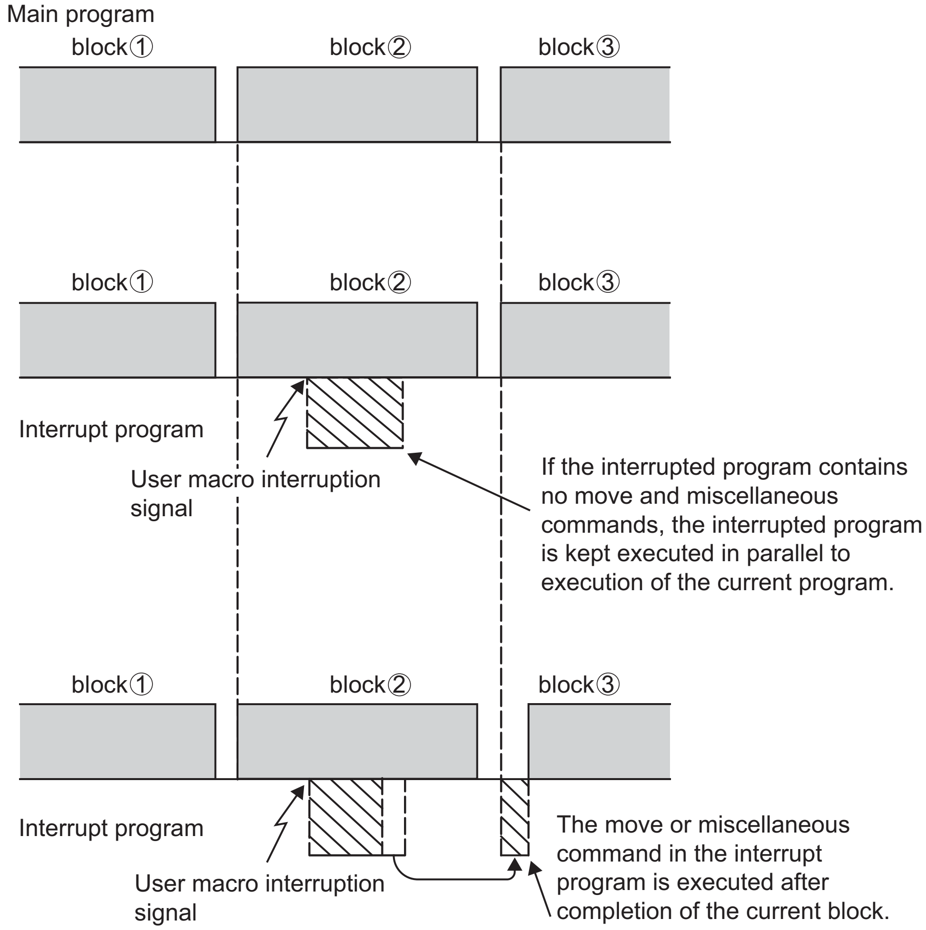
(3) If the interrupt program contains no move and miscellaneous function (MSTB) commands, the interrupt program is executed without interrupting execution of the current block.

However, if the interrupt program has not ended even after the execution of the original block is completed, the system may stop machining temporarily.

[Type 1]


Main program
block①
block②
block③
If the interrupt program contains a move or miscellaneous function command, the reset of block (2) is lost.
block①
block②
block③
Interrupt program
User macro interruption signal
If the interrupted program contains no move and miscellaneous commands, it resumes operation from where it left in block (2), that is, all the reset commands.
block①
block②
block②
block③
Interrupt program
User macro interruption signal
Executing

[Type 2]


Main program
block①
block②
block③
block①
block②
block③
Interrupt program
User macro interruption signal
If the interrupted program contains no move and miscellaneous commands, the interrupted program is kept executed in parallel to execution of the current program.
block①
block②
block③
Interrupt program
User macro interruption signal
The move or miscellaneous command in the interrupt program is executed after completion of the current block.

Calling method

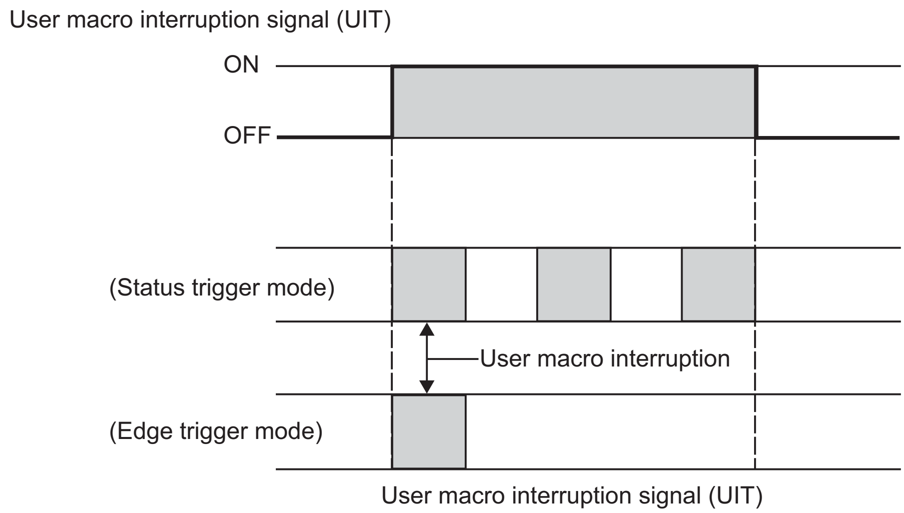
User macro interruption is classified into the following two types depending on the way an interrupt program is called. These two types of interrupt are selected by parameter "#8155 Sub-pro interrupt".
Both types of interrupt are added to the calculation of the nest level. The subprograms and user macros 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.



Returning from user macro interruption

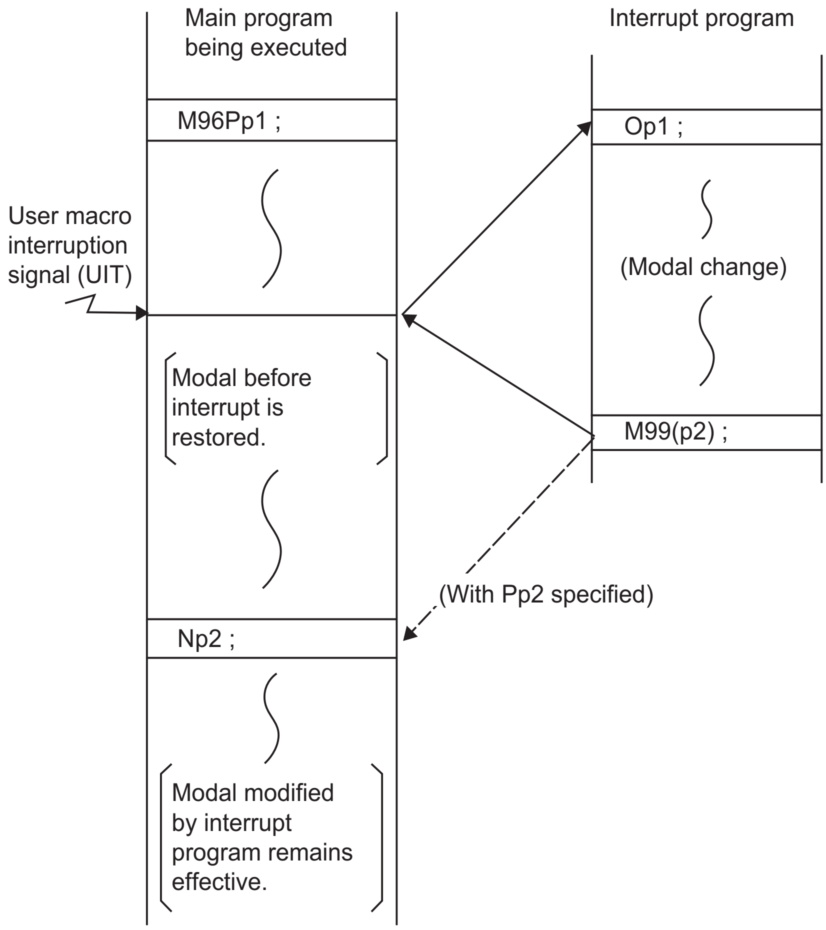
M99 (P__) ;

An M99 command is issued in the interrupt program to return to the main program.
Address P is used to specify the sequence number of the return destination in the main program.
The blocks from the one next to the interrupted block to the last one in the main program are first searched for the block with designated sequence No. If it is not found, all the blocks before the interrupted one are then searched. Control thus returns to the block with sequence No. that is found first in the above search.
(This is equivalent to M99P__ used after M98 calling.)

Modal information affected by user macro interruption

If modal information is changed by the interrupt program, it is handled as follows after control returns from the interrupt program to the main program.

Returning with M99;	The change of modal information by the interrupt program is invalidated and the original modal information is restored. With interrupt type 1, however, if the interrupt program contains a move or miscellaneous function (MSTB) command, the original modal information is not restored.
Returning with M99P__ ;	The original modal information is updated by the change in the interrupt program even after returning to the main program. This is the same as in returning with M99P__; from a program called by M98, etc.



Modal information affected by user macro interruption

Modal information variables (#4401 to #4520)

Modal information when control passes to the user macro interruption program can be known by reading system variables #4401 to #4520.
The unit specified with a command applies.

System variable	Modal information	
#4401 : #4421	G code (group 01) : G code (group 21)	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 operations. To be prepared for such cases, these command functions can be assigned to other M codes. (This invalidates program compatibility.)

User macro interrupt control with alternate M codes is possible by setting the alternate M code in parameters "#1110 M96_M" and "#1111 M97_M" and by validating the setting by selecting parameter "#1109 subs_M".
(M codes 03 to 97 except 30 are available for this purpose.)

If the parameter "#1109 subs_M" used to enable the alternate M codes is not selected, the M96 and M97 codes remain effective for user macro interrupt control.
In either case, the M codes for user macro interrupt control are processed internally and not output to the outside.

Parameters

(1)Subprogram call validity "#8155 Sub-pro interrupt"
1: Subprogram type user macro interruption
0: Macro type user macro interruption

(2) Status trigger mode validity "#1112 S_TRG"
1: Status trigger mode
0: Edge trigger mode

(3) Interrupt type 2 validity "#1113 INT_2"
1: The executable statements in the interrupt program are executed after completion of execution of the current block. (Type 2)
0: The executable statements in the interrupt program are executed before completion of execution of the current block. (Type 1)

(4) Validity of alternate M code for user macro interruption control "#1109 subs_M"
1: Valid
0: Invalid

(5) Alternate M codes for user macro interruption
Interrupt enable M code (equivalent to M96) "#1110 M96_M"
Interrupt disable M code (equivalent to M97) "#1111 M97_M"

(6) Subprogram number selection "#8129 subprogram number selection"
Select a subprogram number to be called preferentially under subprogram control.
0: Commanded program number
1: 4-digits program number beginning with O No.
2: 8-digits program number beginning with O No.

Precautions

(1) If the user macro interruption program uses system variables #5001 and after (position information) to read coordinates, the coordinates pre-read in the buffer are used.

(2) If an interrupt is caused during execution of the tool radius compensation, a sequence No. (M99P__;) must be specified with a command to return from the user macro interrupt program. If no sequence No. is specified, control cannot return to the main program normally.

(3) If the alternate M code for user macro interruption control is duplicated with the M code number for M code macro call, operations vary depending on the MTB specifications (parameter "#1109 subs_M").

(a) Specifications with M code for user macro interruption control valid:
The M code functions as the alternate M code for user macro interruption control.

(b) Specifications with M code for user macro interruption control invalid:
The M code functions as the M code macro call.

(4) With interrupt type 1, when the interrupt program contains a move or MSTB command, do not command the macro interruption to the waiting part system among multiple part systems. Doing so stops machining while the part system that does not perform an interruption remains set in the waiting standby state.
If an interruption is carried out, machining can be started by the "ignore the timing synchronization between part systems" signal; however, the operation of this signal depends on the MTB specifications.

Index

Refer to Programming Manual (Machining Center System) (1/2) for Chapter 14 and previous chapters (page 438 and before).
Refer to Programming Manual (Machining Center System) (2/2) for Chapter 15 and succeeding chapters (page 439 and later).

Revision History

Date of revision	Manual No.	Revision details
Apr. 2015	IB(NA)1501277-A IB(NA)1501278-A	First edition created.
Sep. 2015	IB(NA)1501277-B IB(NA)1501278-B	The descriptions of M800 Series/M80 Series were revised in response to S/W version A4. The following chapters were added. ⬩7.14.2 Inner Arc Override ⬩15.9.3 Tool Shape Input by Program; G10 L100, G11 ⬩15.9.4 R-Navi Data Input by Program; G10 L110, G11 ⬩17.2.3 Tolerance Control ⬩17.5 Spline Interpolation 2; G61.4 ⬩18.1.6 Define by Selecting the Registered Machining Surface The following chapters were revised. ⬩1.1 Coordinate Words and Control Axes ⬩3.4.2 G Code Lists ⬩5.3 Decimal Point Input ⬩7.3 F1-digit Feed ⬩7.12 Deceleration Check ⬩7.14 Automatic Corner Override ⬩10.2 Constant Surface Speed Control; G96, G97 ⬩11.1 Tool Functions (T8-digit BCD) ⬩12.3 Tool Length Compensation in the Tool Axis Direction; G43.1/G49 ⬩13.1.4 Tapping Cycle; G84 ⬩14.4 Macro Call Instructions ⬩14.5.2 Local Variables (#1 to #33) ⬩15.8 Manual Arbitrary Reverse Run Prohibition; G127 ⬩16.2 Sub Part System Control ⬩17.1 High-speed Machining Mode ⬩17.2 High-accuracy Control ⬩17.3 High-speed High-accuracy Control ⬩17.7 Machining Condition Selection I; G120.1, G121 ⬩18.1 Inclined Surface Machining; G68.2, G68.3 ⬩19.3 Basic Machine Coordinate System Selection; G53 ⬩19.6 Workpiece Coordinate System Setting and Offset; G54 to G59 (G54.1) ⬩19.8 3-dimensional Coordinate Conversion; G68/G69 The following chapters were moved. ⬩Parameter Input by Program; G10 L70/L100, G11 (15.6 -> 15.9.1) ⬩Compensation Data Input by Program; G10 L2/L10/L11, G11 (12.7 -> 15.9.2) ⬩Tool Life Management Data Input; G10,G11 (12.7 -> 15.10) Other contents were added/revised/deleted according to specification.
Apr. 2016	IB(NA)1501277-C IB(NA)1501278-C	The descriptions of M800 Series/M80 Series were revised in response to S/W version B2. The following chapters were added. ⬩12.5 Tool Nose Radius Compensation (for Machining Center System) ⬩12.8 Tool Position Compensation; G43.7 ⬩15.9.3 Compensation Data Input by Program (Turning Tool); G10 L12/L13, G11 ⬩16.2 Mixed Control ⬩16.2.1 Arbitrary Axis Exchange; G140, G141, G142 ⬩22 System Variables (Continue to the next page)

Date of revision	Manual No.	Revision details
		(Continued from the previous page) The following chapters were revised. •Introduction •3.4 G Codes •5.4 Decimal Point Input •6.3 Circular Interpolation; G02/G03 •6.4 R Specification Circular Interpolation; G02, G03 •6.7 Helical Interpolation; G17, G18, G19, and G02, G03 •7.1 Rapid Traverse Rate •7.3 F1-digit Feed •7.7 Rapid Traverse Constant Inclination Acceleration/Deceleration •7.13 Rapid Traverse Block Overlap; G0.5 P1 •9.3 Index Table Indexing •10.2 Constant Surface Speed Control; G96, G97 •10.4 Spindle Position Control (Spindle/C Axis Control) •12.1 Tool Compensation •13.1.4 Tapping Cycle; G84 •14.1 Subprogram Control; M98, M99, M198 •14.2 Variable Commands •14.4 Macro Call Instructions •14.6 User Macro Commands •15.7 Normal Line Control; G40.1/G41.1/G42.1 (G150/G151/G152) •15.8 Manual Arbitrary Reverse Run Prohibition; G127 •15.9 Data Input by Program •16.3.1 Sub Part System Control I; G122 •17.1 High-speed Machining Mode •17.2 High-accuracy Control •17.3 High-speed High-accuracy Control •18.1 Inclined Surface Machining; G68.2, G68.3 •19.6 Workpiece Coordinate System Setting and Offset; G54 to G59 (G54.1) •19.10 Coordinate Rotation Input by Parameter ; G10 I_ J_/K_ •21.2 Skip Function; G31 Other mistakes were corrected.
Sep. 2016	IB(NA)1501277-D IB(NA)1501278-D	The descriptions were revised in response to S/W version C1 of M800 Series/M80 Series. The descriptions were revised in response to S/W version A1 of C80 Series. The following chapters were added. •18.3 Tool Center Point Control; G43.4, G43.5/G49 •18.5 3-dimensional Tool Radius Compensation (Tool's vertical-direction compensation); G40/G41.2,G42.2 •20.2 Enable Interfering Object Selection Data; G186 •22.29 System Variables (Interfering Object Selection) •22.30 System Variables (ZR Device Access Variables) [C80] The following chapters were revised. •Introduction •Precautions for Safety •3.2 File Format •3.4 G Codes •6.9 Cylindrical Interpolation; G07.1 •7.4 Feed Per Minute/Feed Per Revolution (Asynchronous Feed/Synchronous Feed); G94, G95 •12.3 Tool Radius Compensation ; G38,G39/G40/G41,G42 •14.1 Subprogram Control; M98, M99, M198 •14.6 User Macro Commands •15.9.1 Parameter Input by Program; G10 L70/L100, G11 •16.3 Sub Part System Control •17.1 High-speed Machining Mode •17.3 High-speed High-accuracy Control •17.5 Spline Interpolation 2; G61.4 (Continue to the next page)

Date of revision	Manual No.	Revision details
		(Continued from the previous page) •18.1 Tool Position Compensation; G43.7 •18.4 Inclined Surface Machining; G68.2, G68.3 •18.5 3-dimensional Tool Radius Compensation (Tool's vertical-direction compensation); G40/G41.2, G42.2 •19.7 Workpiece Coordinate System Preset; G92.1 •19.10 Coordinate Rotation Input by Parameter ; G10 I_ J_/K_ •21.6 Torque Limitation Skip; G160 •22.1 System Variable List •22.5 System Variables (Tool Information) •22.15 System Variables (Time Read Variables) •22.21 System Variables (Rotary Axis Configuration Parameter) •22.27 System Variables (R Device Access Variables) •22.28 System Variables (PLC Data Reading) The following chapters were moved. •Tool Length Compensation in the Tool Axis Direction ; G43.1/G49 (12.3 -> 18.2) •Tool Position Compensation; G43.7/G49 (12.8 -> 18.1) Other mistakes were corrected.

M800/M80/C80 Series Manual List

These contents are described in the presupposition that all functions of M800/M80/C80 Series are available.
Some functions or screens may not be available depending on the machine or specifications set by MTB. (Confirm the specifications before use.)
The manuals issued by MTB take precedence over these manuals.

Manual	IB No.	Purpose and Contents
M800/M80 Series Instruction Manual	IB-1501274	- Operation guide for NC - Explanation for screen operation, etc.
C80 Series Instruction Manual	IB-1501453	- Operation guide for NC - Explanation for screen operation, etc.
M800/M80/C80 Series Programming Manual (Lathe System) (1/2)	IB-1501275	- G code programming for lathe system - Basic functions, etc.
M800/M80/C80 Series Programming Manual (Lathe System) (2/2)	IB-1501276	- G code programming for lathe system - Functions for multi-part system, high-accuracy function, etc.
M800/M80/C80 Series Programming Manual (Machining Center System) (1/2)	IB-1501277	- G code programming for machining center system - Basic functions, etc.
M800/M80/C80 Series Programming Manual (Machining Center System) (2/2)	IB-1501278	- G code programming for machining center system - Functions for multi-part system, high-accuracy function, etc.
M800/M80/C80 Series Alarm/Parameter Manual	IB-1501279	- Alarms - Parameters

Manuals for MTBs (NC)

Manual	IB No.	Purpose and Contents
M800/M80/C80 Series Specifications Manual	IB-1501267	- Model selection - Specifications of hardware unit - Outline of various functions
M800W/M80W Series Connection and Setup Manual	IB-1501268	- Detailed specifications of hardware unit - Installation, connection, wiring, setup (startup/adjustment)
M800S/M80 Series Connection and Setup Manual	IB-1501269	- Detailed specifications of hardware unit - Installation, connection, wiring, setup (startup/adjustment)
C80 Series Connection and Setup Manual	IB-1501452	- Detailed specifications of hardware unit - Installation, connection, wiring, setup (startup/adjustment)
M800/M80 Series PLC Development Manual	IB-1501270	- Electrical design - I/O relation (assignment, setting, connection), field network - Development environment (PLC on-board, peripheral development environment), etc.
M800/M80 Series PLC Programming Manual	IB-1501271	- Electrical design - Sequence programming - PLC support functions, etc.
M800/M80/C80 Series PLC Interface Manual	IB-1501272	- Electrical design - Interface signals between NC and PLC
M800/M80 Series Maintenance Manual	IB-1501273	- Cleaning and replacement for each unit - Other items related to maintenance
C80 Series Maintenance Manual	IB-1501454	- Cleaning and replacement for each unit - Other items related to maintenance

Manuals for MTBs (drive section)

Manual	IB No.	Contents
MDS-E/EH Series Specifications Manual	IB-1501226	- Specifications for power supply regeneration type
MDS-E/EH Series Instruction Manual	IB-1501229	- Instruction for power supply regeneration type
MDS-EJ/EJH Series Specifications Manual	IB-1501232	- Specifications for regenerative resistor type
MDS-EJ/EJH Series Instruction Manual	IB-1501235	- Instruction for regenerative resistor type
MDS-EM/EMH Series Specifications Manual	IB-1501238	- Specifications for multi-hybrid, power supply regeneration type
MDS-EM/EMH Series Instruction Manual	IB-1501241	- Instruction for multi-hybrid, power supply regeneration type
DATA BOOK	IB-1501252	- Specifications of servo drive unit, spindle drive unit, motor, etc.

Global Service Network

AMERICA

MITSUBISHI ELECTRIC AUTOMATION INC. (AMERICA FA CENTER)
Central Region Service Center (Chicago)
500 CORPORATE WOODS PARKWAY, VERNON HILLS, ILLINOIS 60061, U.S.A.
TEL: +1-847-478-2500 / FAX: +1-847-478-2650
Minneapolis, MN Service Satellite
Detroit, MI Service Satellite
Grand Rapids, MI Service Satellite
Lima, OH Service Satellite
Cleveland, OH Service Satellite
Indianapolis, IN Service Satellite
St. Louis, MO Service Satellite

South/East Region Service Center (Georgia)
1845 SATTELITE BOULEVARD STE. 450, DULUTH, GEORGIA 30097, U.S.A.
TEL +1-678-258-4529 / FAX +1-678-258-4519
Charleston, SC Service Satellite
Charlotte, NC Service Satellite
Raleigh, NC Service Satellite
Dallas, TX Service Satellite
Houston, TX Service Satellite
Hartford, CT Service Satellite
Knoxville, TN Service Satellite
Nashville, TN Service Satellite
Baltimore, MD Service Satellite
Pittsburg, PA Service Satellite
Allentown, PA Service Satellite
Syracuse, NY Service Satellite
Tampa, FL Service Satellite
Lafayette, LA Service Satellite

Western Region Service Center (California)
5900-B KATELLA AVE. - 5900-A KATELLA AVE. CYPRESS, CALIFORNIA 90630, U.S.A.
TEL: +1-714-699-2625 / FAX: +1-847-478-2650
San Francisco, CA Service Satellite
Seattle, WA Service Satellite

Canada Region Service Center (Tronto)
4299 14TH AVENUE MARKHAM, ONTARIO L3R OJ2, CANADA
TEL: +1-905-754-3805 / FAX: +1-905-475-7935
Edmonton, AB Service Satellite
Montreal, QC Service Satellite

Mexico Region Service Center (Queretaro)
Parque Tecnológico Innovación Querétaro, Lateral Carretera Estatal 431, Km 2+200, Lote 91 Modulos 1 y 2
Hacienda la Machorra, CP 76246, El Marqués, Querétaro, México
TEL: +52-442-153 4250
Monterrey, NL Service Satellite
Mexico City, DF Service Satellite

BRAZIL

MELCO CNC do Brasil Comércio e Serviços Ltda.
Brazil Region Service Center
AV. GISELE CONSTANTINO,1578, PARQUE BELA VISTA, VOTORANTIM-SP, BRAZIL CEP:18.110-650
TEL: +55-15-3023-9000
JOVIMAQ – Joinville, SC Service Satellite
MAQSERVICE – Canoas, RS Service Satellite

EUROPE

MITSUBISHI ELECTRIC EUROPE B.V.
European Service Headquarter (Dusseldorf, GERMANY)
Mitsubishi-Electric-Platz 1 40882 RATINGEN, GERMANY
TEL: +49-2102-486-1850 / FAX: +49-2102-486-5910

South Germany Service Center (Stuttgart)
KURZE STRASSE. 40, 70794 FILDERSTADT-BONLANDEN, GERMANY
TEL: + 49-711-770598-123 / FAX: +49-711-770598-141

France Service Center (Paris)
25, BOULEVARD DES BOUVETS, 92741 NANTERRE CEDEX FRANCE
TEL: +33-1-41-02-83-13 / FAX: +33-1-49-01-07-25

France Service Satellite (Lyon)
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 (Milan)
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 Service Satellite (Padova)
VIA G. SAVELLI, 24 - 35129 PADOVA, ITALY
TEL: +39-039-6053-342 / FAX: +39-039-6053-206

U.K. Service Center
TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.
TEL: +49-2102-486-1850 / 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: +48-12-347-6500 / FAX: +48-12-630-4701

Hungary Service Center
MADARASZ VIKTOR 47-49 , BUDAPEST XIII; HUNGARY
TEL: +48-12-347-6500 / FAX: +48-12-630-4701

Turkey Service Center
MITSUBISHI ELECTRIC TURKEY A.Ş
SERIFALI MAHALLESI NUTUK SOKAK. NO.5 34775
UMRANIYE, ISTANBUL, TURKEY
TEL: +90-216-526-3990 / FAX: +90-216-526-3995

Czech Republic Service Center
AutoCont Control Systems s.r.o (Service Partner)
KAFKOVA 1853/3, 702 00 OSTRAVA 2, CZECH REPUBLIC
TEL: +420-59-5691-185 / FAX: +420-59-5691-199

Russia Service Center
NC-TECH (Service Partner)
213, B.NOVODMITROVSKAYA STR., 14/2, 127015 MOSCOW, RUSSIA
TEL: +7-495-748-0191 / FAX: +7-495-748-0192

Sweden Service Center
HAMMARBACKEN 14, P.O.BOX 750 SE-19127, SOLLENTUNA, SWEDEN
TEL: +46-8-6251000 / FAX: +46-8-966877

Bulgaria Service Center
AKHNATON Ltd. (Service Partner)
4 ANDREJ LJAPCHEV BLVD. POB 21, BG-1756 SOFIA, BULGARIA
TEL: +359-2-8176009 / FAX: +359-2-9744061

Ukraine Service Center (Kharkov)
CSC Automation Ltd. (Service Partner)
APTEKARSKIY PEREULOK 9-A, OFFICE 3, 61001 KHARKOV, UKRAINE
TEL: +380-57-732-7774 / FAX: +380-57-731-8721

Belarus Service Center
TECHNIKON Ltd. (Service Partner)
NEZAVISIMOSTI PR.177, 220125 MINSK, BELARUS
TEL: +375-17-393-1177 / FAX: +375-17-393-0081

South Africa Service Center
MOTIONTRONIX (Service Partner)
P.O. BOX 9234, EDLEEN, KEMPTON PARK GAUTENG, 1625, SOUTH AFRICA
TEL: +27-11-394-8512 / FAX: +27-11-394-8513

ASEAN

MITSUBISHI ELECTRIC ASIA PTE. LTD. (ASEAN FA CENTER)
Singapore Service Center
307 ALEXANDRA ROAD #05-01/02 MITSUBISHI ELECTRIC BUILDING SINGAPORE 159943
TEL: +65-6473-2308 / FAX: +65-6476-7439

Philippines Service Center
Flexible (Service Partner)
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 Center
6TH FLOOR, DETECH TOWER, 8 TON THAT THUYET STREET, MY DINH 2 WARD,
NAM TU LIEM DISTRICT, HA NOI CITY, VIETNAM
TEL: +84-4-3937-8075 / FAX: +84-4-3937-8076

INDONESIA

PT. MITSUBISHI ELECTRIC INDONESIA
Indonesia Service Center (Cikarang)
JL. KENARI RAYA BLOK G2-07A, DELTA SILICON 5, LIPPO CIKARANG - BEKASI 17550, INDONESIA
TEL: +62-21-2961-7797 / FAX: +62-21-2961-7794

MALAYSIA

MITSUBISHI ELECTRIC SALES MALAYSIA SDN. BHD.
Malaysia Service Center (Kuala Lumpur Service Center)
LOT 11, JALAN 219, P.O BOX 1036, 46860 PETALING JAYA, SELANGOR DARUL EHSAN. MALAYSIA
TEL: +60-3-7960-2628 / FAX: +60-3-7960-2629
Johor Bahru Service satellite

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 / FAX: +66-2-682-6020

INDIA

MITSUBISHI ELECTRIC INDIA PVT., LTD.
CNC Technical Center (Bangalore)
PLOT NO. 56, 4TH MAIN ROAD, PEENYA PHASE 3,
PEENYA INDUSTRIAL AREA, BANGALORE 560058, KARNATAKA, INDIA
TEL : +91-80-4655-2121 FAX : +91-80-4655-2147
Chennai Service Satellite
Coimbatore Service Satellite
Hyderabad Service Satellite

North India Service Center (Gurgaon)
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
Panth Nagar Service Satellite
Delhi Service Satellite
Jamshedpur Service Satellite

West India Service Center (Pune)
EMERALD HOUSE, EL-3, J BLOCK, M.I.D.C., BHOSARI, PUNE - 411026, MAHARASHTRA, INDIA
TEL : +91-20-2710 2000 FAX : +91-20-2710 2100
Kolhapur Service Satellite
Aurangabad Service Satellite
Mumbai Service Satellite

West India Service Center (Ahmedabad)
UNIT NO: B/4, 3RD FLOOR, SAFAL PROFITAIRE, PRAHALADNAGAR CORPORATE ROAD,
PRAHALADNAGAR SATELLITE, AHMEDABAD – 380015, GUJRAT, INDIA
TEL : +91-265-2314699
Rajkot Service Satellite

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: +86-21-2322-3000*8422
China Ningbo Service Partner
China Wuxi Service Partner
China Jinan Service Partner
China Hangzhou Service Partner

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 Partner

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 Chengdu Service Center
1501-1503,15F,GUANG-HUA CENTRE BUILDING-C,NO.98 NORTH GUANG HUA 3th RD,
CHENGDU,610000,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: +86-755-8229-3686
China Xiamen Service Partner
China DongGuang Service Partner

China Dalian Service Center
DONGBEI 3-5, DALIAN ECONOMIC & TECHNICAL DEVELOPMENTZONE, LIAONING PROVINCE,
116600, CHINA
TEL: +86-411-8765-5951 / FAX: +86-411-8765-5952

KOREA

MITSUBISHI ELECTRIC AUTOMATION KOREA CO., LTD. (KOREA FA CENTER)
Korea Service Center
8F GANGSEO HANGANG XI-TOWER A, 401 YANGCHEON-RO, GANGSEO-GU,
SEOUL 07528 KOREA
TEL: +82-2-3660-9609 / FAX: +82-2-3664-8668
Korea Daegu Service Satellite

TAIWAN

MITSUBISHI ELECTRIC TAIWAN CO., LTD. (TAIWAN FA CENTER)
Taiwan Taichung Service Center
NO.8-1, INDUSTRIAL 16TH RD., TAICHUNG INDUSTRIAL PARK, SITUN DIST.,
TAICHUNG CITY 40768, TAIWAN
TEL: +886-4-2359-0688 / FAX: +886-4-2359-0689

Taiwan Taipei Service Center
10F, NO.88, SEC.6, CHUNG-SHAN N. RD., SHI LIN DIST., TAIPEI CITY 11155, TAIWAN
TEL: +886-2-2833-5430 / FAX: +886-2-2833-5433

Taiwan Tainan Service Center
11F-1., NO.30, ZHONGZHENG S. ROAD, YONGKANG DISTRICT, TAINAN CITY 71067, TAIWAN
TEL: +886-6-252-5030 / FAX: +886-6-252-5031

OCEANIA

MITSUBISHI ELECTRIC AUSTRALIA PTY. LTD.
Oceania Service Center
348 VICTORIA ROAD, RYDALMERE, N.S.W. 2116 AUSTRALIA
TEL: +61-2-9684-7269/ FAX: +61-2-9684-7245

Notice

Every effort has been made to keep up with software and hardware revisions in the contents described in this manual. However, please understand that in some unavoidable cases simultaneous revision is not possible.
Please contact your Mitsubishi Electric dealer with any questions or comments regarding the use of this product.

MITSUBISHI CNC

MITSUBISHI ELECTRIC CORPORATION
HEAD OFFICE : TOKYO BLDG.,2-7-3 MARUNOUCHI,CHIYODA-KU,TOKYO 100-8310,JAPAN

MODEL	M800/M80/C80 Series
MODEL CODE	100-442
Manual No.	IB-1501277

IB(NA)1501277(ENG)-D(1609)MEE
Made in JAPAN