GE Fanuc Automation

Computer Numerical Control Products

Series 15i-MB
Series 150i-MB

Operator's Manual (Programming)

GFZ-63784EN/01

February 2002

Warnings, Cautions, and Notes as Used in this Publication

Warning

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

Caution

Caution notices are used where equipment might be damaged if care is not taken.

Note

Notes merely call attention to information that is especially significant to understanding and operating the equipment.

This document is based on information available at the time of its publication. While efforts have been made to be accurate, the information contained herein does not purport to cover all details or variations in hardware or software, nor to provide for every possible contingency in connection with installation, operation, or maintenance. Features may be described herein which are not present in all hardware and software systems. GE Fanuc Automation assumes no obligation of notice to holders of this document with respect to changes subsequently made.

GE Fanuc Automation makes no representation or warranty, expressed, implied, or statutory with respect to, and assumes no responsibility for the accuracy, completeness, sufficiency, or usefulness of the information contained herein. No warranties of merchantability or fitness for purpose shall apply.

SAFETY PRECAUTIONS

This section describes the safety precautions related to the use of CNC units.
It is essential that these precautions be observed by users to ensure the safe operation of machines equipped with a CNC unit (all descriptions in this section assume this configuration). Note that some precautions are related only to specific functions, and thus may not be applicable to certain CNC units.
Users must also observe the safety precautions related to the machine, as described in the relevant manual supplied by the machine tool builder. Before attempting to operate the machine or create a program to control the operation of the machine, the operator must become fully familiar with the contents of this manual and relevant manual supplied by the machine tool builder.

CONTENTS

DEFINITION OF WARNING, CAUTION, AND NOTE

This manual includes safety precautions for protecting the user and preventing damage to the machine. Precautions are classified into Warning and Caution according to their bearing on safety. Also, supplementary information is described as a Note. Read the Warning, Caution, and Note thoroughly before attempting to use the machine

⚠ WARNING

Applied when there is a danger of the user being injured or when there is a damage of both the user being injured and the equipment being damaged if the approved procedure is not observed.

⚠ CAUTION

Applied when there is a danger of the equipment being damaged, if the approved procedure is not observed.

NOTE

The Note is used to indicate supplementary information other than Warning and Caution.

- Read this manual carefully, and store it in a safe place.

GENERAL WARNINGS AND CAUTIONS

⚠ **WARNING**

1. Never attempt to machine a workpiece without first checking the operation of the machine. Before starting a production run, ensure that the machine is operating correctly by performing a trial run using, for example, the single block, feedrate override, or machine lock function or by operating the machine with neither a tool nor workpiece mounted. Failure to confirm the correct operation of the machine may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.

2. Before operating the machine, thoroughly check the entered data.
   Operating the machine with incorrectly specified data may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.

3. Ensure that the specified feedrate is appropriate for the intended operation. Generally, for each machine, there is a maximum allowable feedrate.
   The appropriate feedrate varies with the intended operation. Refer to the manual provided with the machine to determine the maximum allowable feedrate.
   If a machine is run at other than the correct speed, it may behave unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.

4. When using a tool compensation function, thoroughly check the direction and amount of compensation.
   Operating the machine with incorrectly specified data may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.

5. The parameters for the CNC and PMC are factory-set. Usually, there is not need to change them. When, however, there is not alternative other than to change a parameter, ensure that you fully understand the function of the parameter before making any change.
   Failure to set a parameter correctly may result in the machine behaving unexpectedly, possibly causing damage to the workpiece and/or machine itself, or injury to the user.

6. Immediately after switching on the power, do not touch any of the keys on the MDI panel until the position display or alarm screen appears on the CNC unit.
   Some of the keys on the MDI panel are dedicated to maintenance or other special operations. Pressing any of these keys may place the CNC unit in other than its normal state. Starting the machine in this state may cause it to behave unexpectedly.

7. The operator's manual and programming manual supplied with a CNC unit provide an overall description of the machine's functions, including any optional functions. Note that the optional functions will vary from one machine model to another. Therefore, some functions described in the manuals may not actually be available for a particular model. Check the specification of the machine if in doubt.

⚠ WARNING

8. Some functions may have been implemented at the request of the machine-tool builder. When using such functions, refer to the manual supplied by the machine-tool builder for details of their use and any related cautions.

NOTE

Programs, parameters, and macro variables are stored in nonvolatile memory in the CNC unit. Usually, they are retained even if the power is turned off.
Such data may be deleted inadvertently, however, or it may prove necessary to delete all data from nonvolatile memory as part of error recovery.
To guard against the occurrence of the above, and assure quick restoration of deleted data, backup all vital data, and keep the backup copy in a safe place.

WARNINGS AND CAUTIONS RELATED TO PROGRAMMING

This section covers the major safety precautions related to programming. Before attempting to perform programming, read the supplied operator's manual and programming manual carefully such that you are fully familiar with their contents.

⚠ WARNING

1.Coordinate system setting

If a coordinate system is established incorrectly, the machine may behave unexpectedly as a result of the program issuing an otherwise valid move command.
Such an unexpected operation may damage the tool, the machine itself, the workpiece, or cause injury to the user.

2.Positioning by nonlinear interpolation

When performing positioning by nonlinear interpolation (positioning by nonlinear movement between the start and end points), the tool path must be carefully confirmed before performing programming. Positioning involves rapid traverse. If the tool collides with the workpiece, it may damage the tool, the machine itself, the workpiece, or cause injury to the user.

3.Function involving a rotation axis

When programming polar coordinate interpolation or normal-direction (perpendicular) control, pay careful attention to the speed of the rotation axis. Incorrect programming may result in the rotation axis speed becoming excessively high, such that centrifugal force causes the chuck to lose its grip on the workpiece if the latter is not mounted securely.
Such mishap is likely to damage the tool, the machine itself, the workpiece, or cause injury to the user.

4.Inch/metric conversion

Switching between inch and metric inputs does not convert the measurement units of data such as the workpiece origin offset, parameter, and current position.
Before starting the machine, therefore, determine which measurement units are being used. Attempting to perform an operation with invalid data specified may damage the tool, the machine itself, the workpiece, or cause injury to the user.

5.Constant surface speed control

When an axis subject to constant surface speed control approaches the origin of the workpiece coordinate system, the spindle speed may become excessively high. Therefore, it is necessary to specify a maximum allowable speed. Specifying the maximum allowable speed incorrectly may damage the tool, the machine itself, the workpiece, or cause injury to the user.

⚠ **WARNING**

6.Stroke check

After switching on the power, perform a manual reference position return as required. Stroke check is not possible before manual reference position return is performed. Note that when stroke check is disabled, an alarm is not issued even if a stroke limit is exceeded, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the user.

7. Tool post interference check

A tool post interference check is performed based on the tool data specified during automatic operation. If the tool specification does not match the tool actually being used, the interference check cannot be made correctly, possibly damaging the tool or the machine itself, or causing injury to the user. After switching on the power, or after selecting a tool post manually, always start automatic operation and specify the tool number of the tool to be used.

8. Absolute/incremental mode

If a program created with absolute values is run in incremental mode, or vice versa, the machine may behave unexpectedly.

9.Plane selection

If an incorrect plane is specified for circular interpolation, helical interpolation, or a canned cycle, the machine may behave unexpectedly. Refer to the descriptions of the respective functions for details.

10.Torque limit skip

Before attempting a torque limit skip, apply the torque limit. If a torque limit skip is specified without the torque limit actually being applied, a move command will be executed without performing a skip.

11.Programmable mirror image

Note that programmed operations vary considerably when a programmable mirror image is enabled.

12.Compensation function

If a command based on the machine coordinate system or a reference position return command is issued in compensation function mode, compensation is temporarily canceled, resulting in the unexpected behavior of the machine.
Before issuing any of the above commands, therefore, always cancel compensation function mode.

WARNINGS AND CAUTIONS RELATED TO HANDLING

This section presents safety precautions related to the handling of machine tools. Before attempting to operate your machine, read the supplied operator's manual and programming manual carefully, such that you are fully familiar with their contents.

⚠ WARNING

1.Manual operation

When operating the machine manually, determine the current position of the tool and workpiece, and ensure that the movement axis, direction, and feedrate have been specified correctly. Incorrect operation of the machine may damage the tool, the machine itself, the workpiece, or cause injury to the operator.

2.Manual reference position return

After switching on the power, perform manual reference position return as required. If the machine is operated without first performing manual reference position return, it may behave unexpectedly. Stroke check is not possible before manual reference position return is performed. An unexpected operation of the machine may damage the tool, the machine itself, the workpiece, or cause injury to the user.

3.Manual numeric command

When issuing a manual numeric command, determine the current position of the tool and workpiece, and ensure that the movement axis, direction, and command have been specified correctly, and that the entered values are valid.
Attempting to operate the machine with an invalid command specified may damage the tool, the machine itself, the workpiece, or cause injury to the operator.

4.Manual handle feed

In manual handle feed, rotating the handle with a large scale factor, such as 100, applied causes the tool and table to move rapidly. Careless handling may damage the tool and/or machine, or cause injury to the user.

5.Disabled override

If override is disabled (according to the specification in a macro variable) during threading, rigid tapping, or other tapping, the speed cannot be predicted, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the operator.

⚠ WARNING

6.Origin/preset operation

Basically, never attempt an origin/preset operation when the machine is operating under the control of a program. Otherwise, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the tool, or causing injury to the user.

7.Workpiece coordinate system shift

Manual intervention, machine lock, or mirror imaging may shift the workpiece coordinate system. Before attempting to operate the machine under the control of a program, confirm the coordinate system carefully.
If the machine is operated under the control of a program without making allowances for any shift in the workpiece coordinate system, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the operator.

8.Software operator's panel and menu switches

Using the software operator's panel and menu switches, in combination with the MDI panel, it is possible to specify operations not supported by the machine operator's panel, such as mode change, override value change, and jog feed commands.
Note, however, that if the MDI panel keys are operated inadvertently, the machine may behave unexpectedly, possibly damaging the tool, the machine itself, the workpiece, or causing injury to the user.

9.Manual intervention

If manual intervention is performed during programmed operation of the machine, the tool path may vary when the machine is restarted. Before restarting the machine after manual intervention, therefore, confirm the settings of the manual absolute switches, parameters, and absolute/incremental command mode.

10.Feed hold, override, and single block

The feed hold, feedrate override, and single block functions can be disabled using custom macro system variable #3004. Be careful when operating the machine in this case.

11.Dry run

Usually, a dry run is used to confirm the operation of the machine. During a dry run, the machine operates at dry run speed, which differs from the corresponding programmed feedrate. Note that the dry run speed may sometimes be higher than the programmed feed rate.

⚠ WARNING

12.Cutter and tool nose radius compensation in MDI mode

Pay careful attention to a tool path specified by a command in MDI mode, because cutter or tool nose radius compensation is not applied. When a command is entered from the MDI to interrupt in automatic operation in cutter or tool nose radius compensation mode, pay particular attention to the tool path when automatic operation is subsequently resumed. Refer to the descriptions of the corresponding functions for details.

13.Program editing

If the machine is stopped, after which the machining program is edited (modification, insertion, or deletion), the machine may behave unexpectedly if machining is resumed under the control of that program. Basically, do not modify, insert, or delete commands from a machining program while it is in use.

WARNINGS RELATED TO DAILY MAINTENANCE

⚠ WARNING

1.Memory backup battery replacement

When replacing the memory backup batteries, keep the power to the machine (CNC) turned on, and apply an emergency stop to the machine. Because this work is performed with the power on and the cabinet open, only those personnel who have received approved safety and maintenance training may perform this work.

When replacing the batteries, be careful not to touch the high-voltage circuits (marked ⚠ and fitted with an insulating cover).

Touching the uncovered high-voltage circuits presents an extremely dangerous electric shock hazard.

NOTE

The CNC uses batteries to preserve the contents of its memory, because it must retain data such as programs, offsets, and parameters even while external power is not applied.

If the battery voltage drops, a low battery voltage alarm is displayed on the machine operator's panel or screen.

When a low battery voltage alarm is displayed, replace the batteries within a week. Otherwise, the contents of the CNC's memory will be lost.

Refer to the Maintenance manual for details of the battery replacement procedure.

⚠ WARNING

2.Absolute pulse coder battery replacement

When replacing the memory backup batteries, keep the power to the machine (CNC) turned on, and apply an emergency stop to the machine. Because this work is performed with the power on and the cabinet open, only those personnel who have received approved safety and maintenance training may perform this work.

When replacing the batteries, be careful not to touch the high-voltage circuits (marked ⚠ and fitted with an insulating cover).

Touching the uncovered high-voltage circuits presents an extremely dangerous electric shock hazard.

NOTE

The absolute pulse coder uses batteries to preserve its absolute position.

If the battery voltage drops, a low battery voltage alarm is displayed on the machine operator's panel or screen.

When a low battery voltage alarm is displayed, replace the batteries within a week. Otherwise, the absolute position data held by the pulse coder will be lost.

Refer to the Maintenance manual for details of the battery replacement procedure.

⚠ WARNING

3.Fuse replacement

For some units, the chapter covering daily maintenance in the operator's manual or programming manual describes the fuse replacement procedure.

Before replacing a blown fuse, however, it is necessary to locate and remove the cause of the blown fuse.

For this reason, only those personnel who have received approved safety and maintenance training may perform this work.

When replacing a fuse with the cabinet open, be careful not to touch the high-voltage circuits (marked ⚠ and fitted with an insulating cover).

Touching an uncovered high-voltage circuit presents an extremely dangerous electric shock hazard.

TABLE OF CONTENTS

APPENDIX

I. GENERAL

1 GENERAL

Operator's Manuals consist of the PROGRAMMING Manual and OPERATION Manual.

About this Operator's Manual

OPERATOR'S MANUAL (PROGRAMMING) (B-63784EN)

I. GENERAL

Describes chapter organization, applicable models, related manuals, and notes for reading this manual.

II. PROGRAMMING

Describes each function: Format used to program functions in the NC language, characteristics, and restrictions.

APPENDIX

Lists tape codes, valid data ranges, and alarms.

OPERATOR'S MANUAL (OPERATION) (B-63784EN-1)

I. GENERAL

Describes chapter organization, applicable models, related manuals, and notes for reading this manual.

II. OPERATION

Describes the manual operation and automatic operation of a machine, procedures for inputting and outputting data, and procedures for editing a program.

III. MAINTENANCE

Describes investigation of trouble generation situation.

APPENDIX

Status when turning power on, when reset

Some functions described in this manual may not be applied to some products. For detail, refer to the DESCRIPTIONS manual(B-63782EN).

This manual does not describe parameters in detail. For details on parameters mentioned in this manual, refer to the manual for parameters (B-63790EN).

This manual describes all optional functions. Look up the options incorporated into your system in the manual written by the machine tool builder.

Applicable product name

The models covered by this manual, and their abbreviations are:

Product name	Abbreviations	
FANUC Series 15*i*-MB	15*i*-MB	Series 15*i*
FANUC Series 150*i*-MB	150*i*-MB	Series 150*i*

Special symbols

This manual uses the following symbols:

P_ : Indicates a combination of axes such as X__ Y__ Z (used in PROGRAMMING.).

; : Indicates the end of a block. It actually corresponds to the ISO code LF or EIA code CR.

Related manuals

The table below lists manuals related to MODEL B of Series 15*i*, and Series 150*i*. In the table, this manual is marked with an asterisk (*).

Table 1 (a) Related Manuals

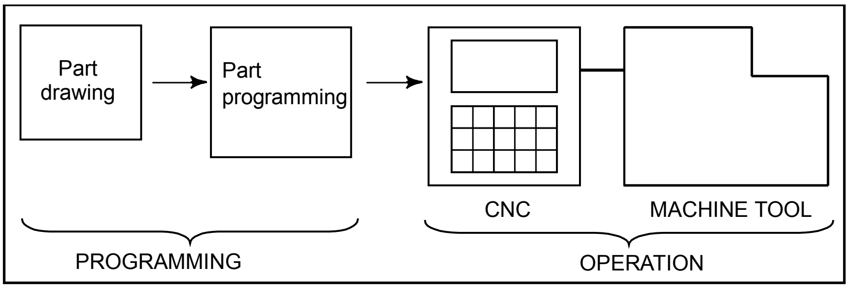
Manual name	Specification number	
DESCRIPTIONS	B-63782EN	
CONNECTION MANUAL (HARDWARE)	B-63783EN	
CONNECTION MANUAL (FUNCTION)	B-63783EN-1	
OPERATOR'S MANUAL (PROGRAMMING)	B-63784EN	*
OPERATOR'S MANUAL (OPERATION)	B-63784EN-1	
MAINTENANCE MANUAL	B-63785EN	
PARAMETER MANUAL	B-63790EN	

1.1 GENERAL FLOW OF OPERATION OF CNC MACHINE TOOL

When machining the part using the CNC machine tool, first prepare the program, then operate the CNC machine by using the program.

(1) First, prepare the program from a part drawing to operate the CNC machine tool. How to prepare the program is described in the OPERATOR'S MANUAL (PROGRAMMING).

(2) The program is to be read into the CNC system. Then, mount the workpieces and tools on the machine, and operate the tools according to the programming. Finally, execute the machining actually. How to operate the CNC system is described in the OPERATOR'S MANUAL (OPERATION).



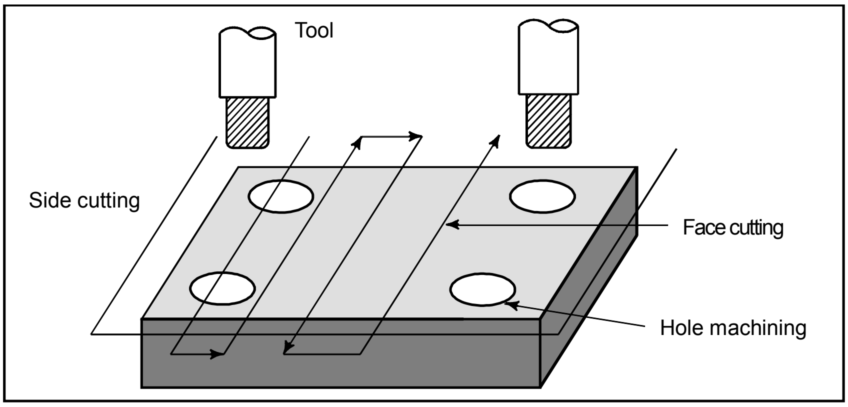
Before the actual programming, make the machining plan for how to machine the part.

Machining plan-

1. Determination of workpieces machining range
2. Method of mounting workpieces on the machine tool
3. Machining sequence in every machining process
4. Machining tools and machining

Decide the machining method in every machining process.

Machining process / Machining procedure	1	2	3
	Feed cutting	Side cutting	Hole machining
1. Machining method : Rough Semi Finish			
2. Machining tools			
3. Machining conditions : Feedrate Cutting depth			
4. Tool path			



Prepare the program of the tool path and machining condition according to the workpiece figure, for each machining.

1.2 NOTES ON READING THIS MANUAL

NOTE

1 The function of an CNC machine tool system depends not only on the CNC, but on the combination of the machine tool, its magnetic cabinet, the servo system, the CNC, the operator's panels, etc. It is too difficult to describe the function, programming, and operation relating to all combinations.
This manual generally describes these from the stand-point of the CNC. So, for details on a particular CNC machine tool, refer to the manual issued by the machine tool builder, which should take precedence over this manual.

2 Headings are placed in the left margin so that the reader can easily access necessary information. When locating the necessary information, the reader can save time by searching though these headings.

3 Machining programs, parameters, variables, etc. are stored in the CNC unit internal non-volatile memory. In general, these contents are not lost by the switching ON/OFF of the power. However, it is possible that a state can occur where precious data stored in the non-volatile memory has to be deleted, because of deletions from a maloperation, or by a failure restoration.
In order to restore rapidly when this kind of mishap occurs, it is recommended that you create a copy of the various kinds of data beforehand.

4 This manual describes as many reasonable variations in equipment usage as possible. It cannot address every combination of features, options and commands that should not be attempted.
If a particular combination of operations is not described, it should not be attempted.

II PROGRAMING

1 GENERAL

1.1 TOOL MOVEMENT ALONG WORKPIECE PARTS FIGURE-INTERPOLATION

The tool moves along straight lines and arcs constituting the workpiece parts figure (See II-4).

Explanation

The function of moving the tool along straight lines and arcs is called the interpolation.

-Tool movement along a straight line

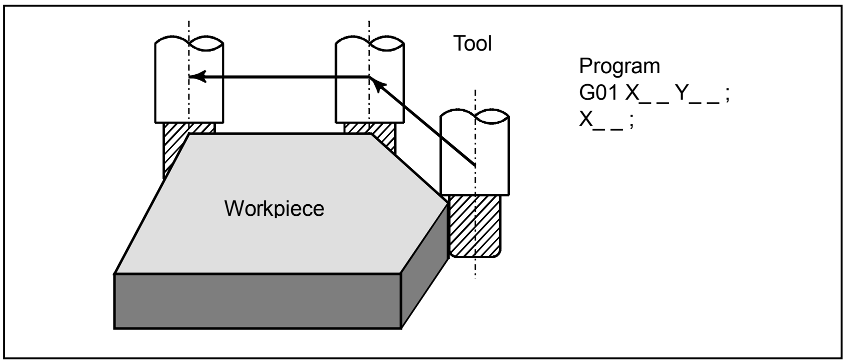


Fig.1.1 (a) Tool movement along a straight line

-Tool movement along an arc

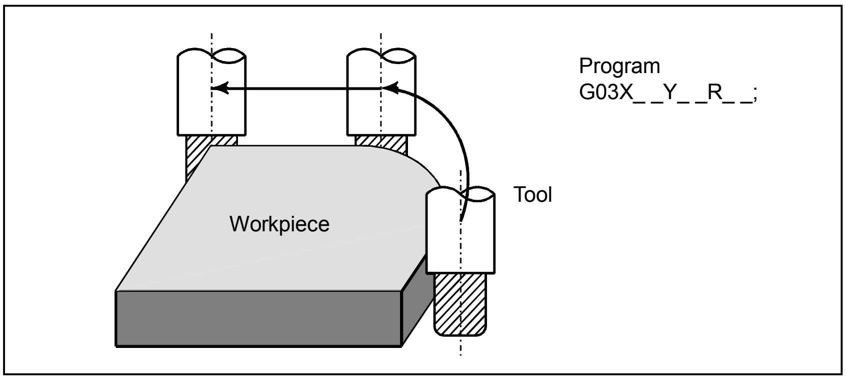


Fig.1.1 (b) Tool movement along an arc

Symbols of the programmed commands G01, G02, ... are called the preparatory function and specify the type of interpolation conducted in the control unit.

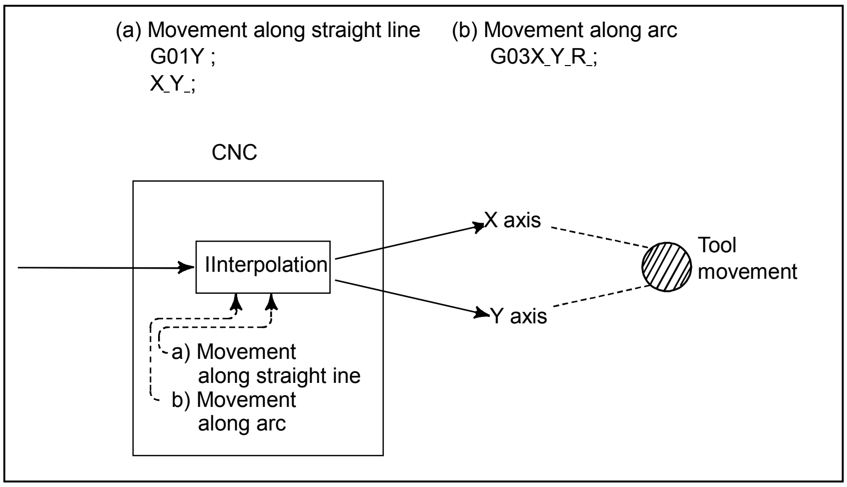


Fig.1.1 (c) Interpolation function

NOTE

Some machines move tables instead of tools but this manual assumes that tools are moved against workpieces.

1.2 FEED-FEED FUNCTION

Movement of the tool at a specified speed for cutting a workpiece is called the feed.

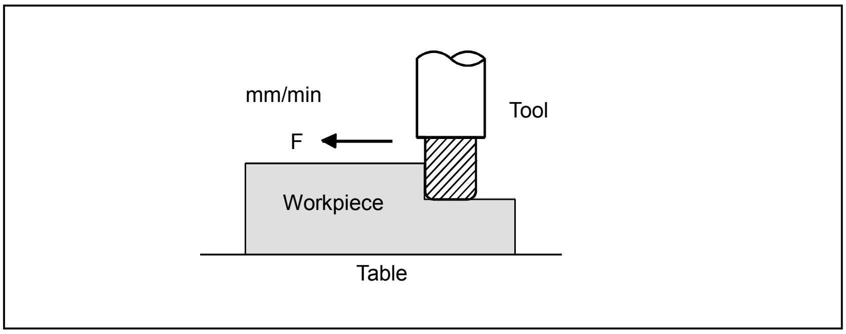


Fig.1.2 (a) Feed function

Feedrates can be specified by using actual numerics.
For example, to feed the tool at a rate of 150 mm/min, specify the following in the program:

F150.0

The function of deciding the feed rate is called the feed function (See II-5).

1.3 PART DRAWING AND TOOL MOVEMENT

1.3.1 Reference Position (Machine-Specific Position)

A CNC machine tool is provided with a fixed position. Normally, tool change and programming of absolute zero point as described later are performed at this position. This position is called the reference position.

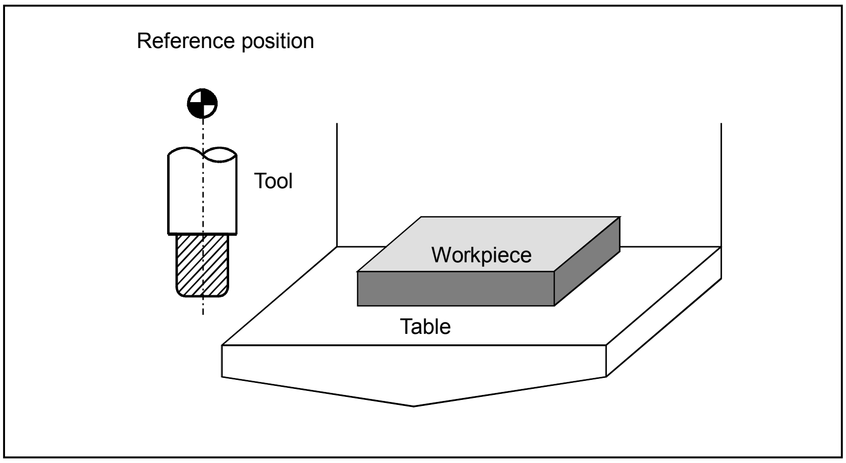


Fig.1.3.1 (a) Reference position

Explanations

The tool can be moved to the reference position in two ways:

1. Manual reference position return (See Operation II-3.1)
   Reference position return is performed by manual button operation.
2. Automatic reference position return (See II-6)
   In general, manual reference position return is performed first after the power is turned on. In order to move the tool to the reference position for tool change thereafter, the function of automatic reference position return is used.

1.3.2 Coordinate System on Part Drawing and Coordinate System Specified by CNC - Coordinate System

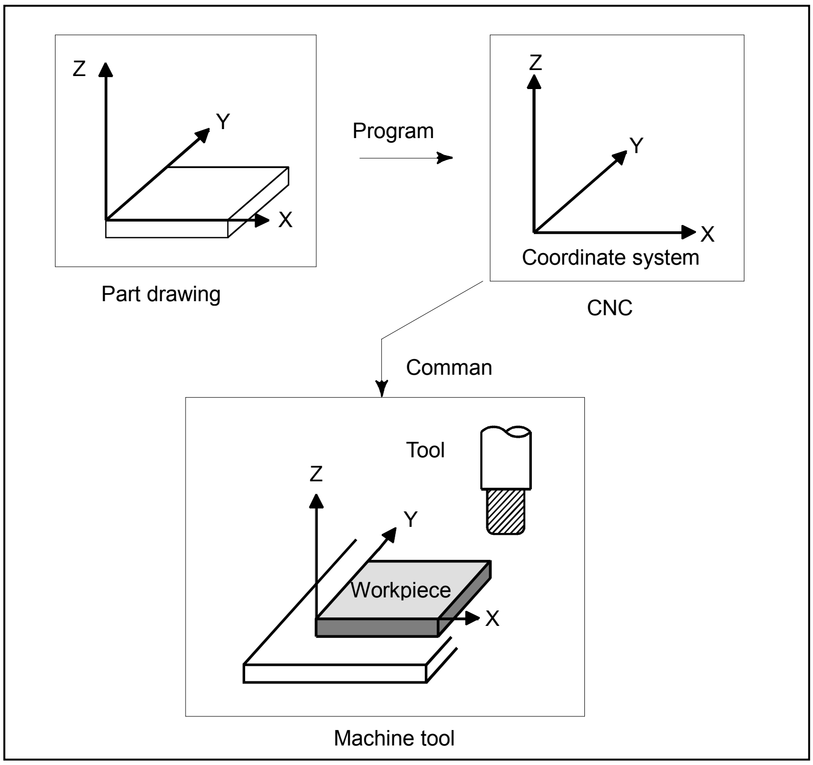


Fig.1.3.2 (a) Coordinate system

Explanation

-Coordinate system

The following two coordinate systems are specified at different locations: (See II-7)

1. Coordinate system on part drawing

   The coordinate system is written on the part drawing. As the program data, the coordinate values on this coordinate system are used.

2. Coordinate system specified by the CNC

   The coordinate system is prepared on the actual machine tool table.

   This can be achieved by programming the distance from the current position of the tool to the zero point of the coordinate system to be set.

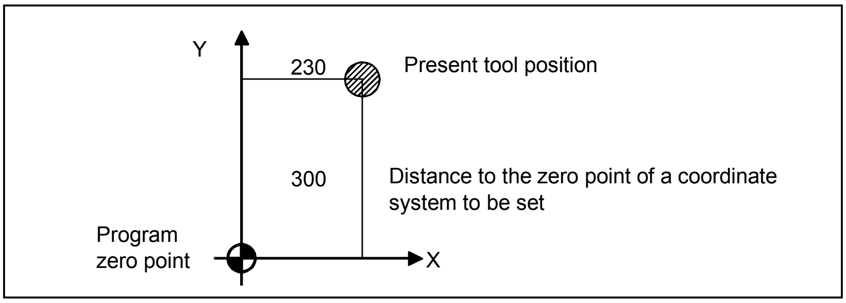


Fig.1.3.2 (b) Coordinate system specified by the CNC

The positional relation between these two coordinate systems is determined when a workpiece is set on the table.

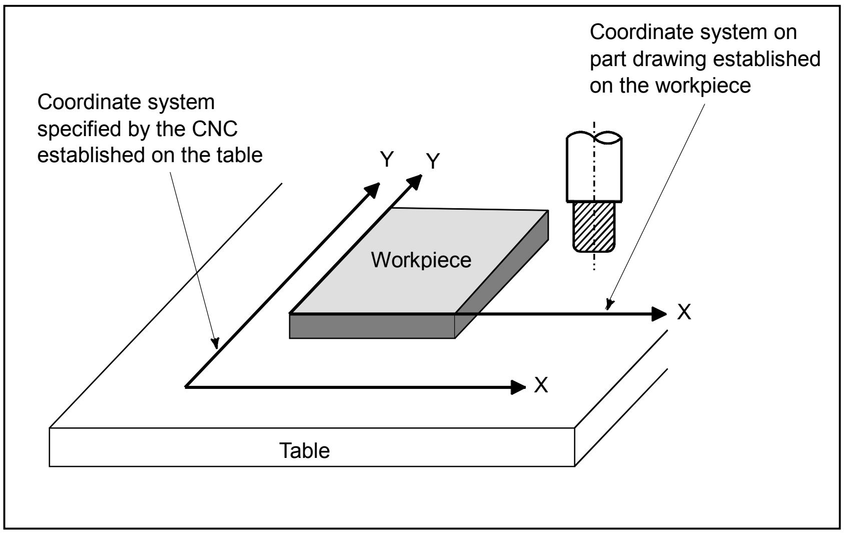

Coordinate system on part drawing established on the workpiece
Coordinate system specified by the CNC established on the table
Y
Y
Workpiece
X
X
Table

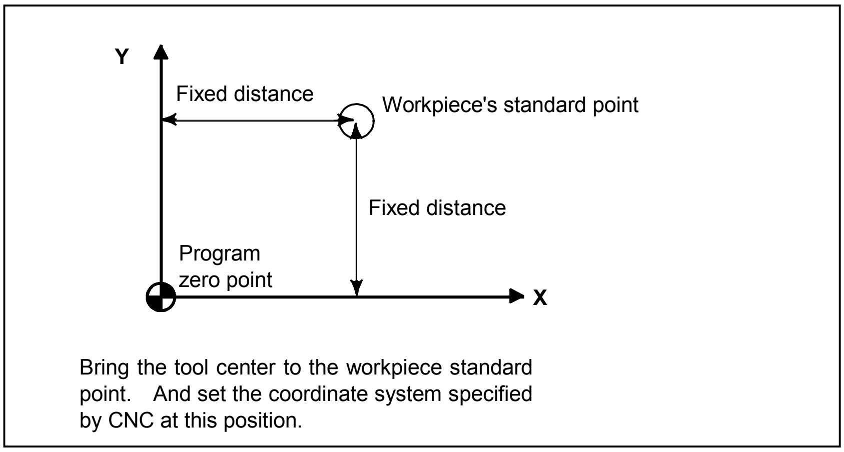
Fig. 1.3.2 (c) Coordinate system specified by CNC and coordinate system on part drawing

The tool moves on the coordinate system specified by the CNC in accordance with the command program generated with respect to the coordinate system on the part drawing, and cuts a workpiece into a shape on the drawing.
Therefore, in order to correctly cut the workpiece as specified on the drawing, the two coordinate systems must be set at the same position.

-Methods of setting the two coordinate systems in the same position

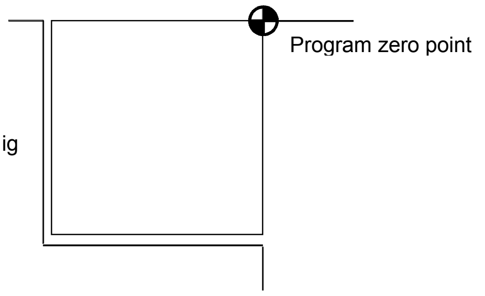
To set the two coordinate systems at the same position, simple methods shall be used according to workpiece shape, the number of machinings.

1 Using a standard plane and point of the workpiece.



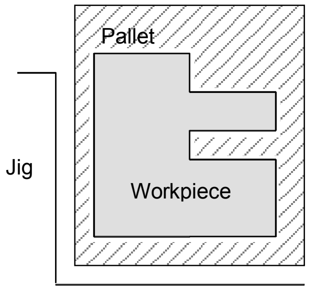
Bring the tool center to the workpiece standard point. And set the coordinate system specified by CNC at this position.

2 Mounting a workpiece directly against the jig



Meet the tool center to the reference position.
And set the coordinate system specified by CNC at this position.
(Jig shall be mounted on the predetermined point from the reference position.)

3 Mounting a workpiece on a pallet, then mounting the workpiece and pallet on the jig



(Jig and coordinate system shall be specified by the same as 2).

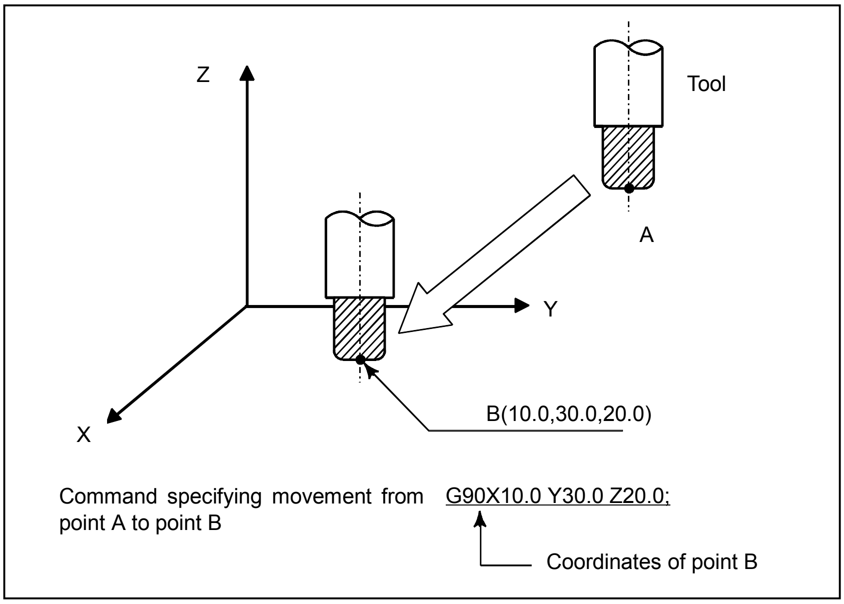
1.3.3 How to Indicate Command Dimensions for Moving the Tool - Absolute, Incremental Commands

Explanation

Command for moving the tool can be indicated by absolute command or incremental command (See II-8.1).

-Absolute command

The tool moves to a point at "the distance from zero point of the coordinate system" that is to the position of the coordinate values.

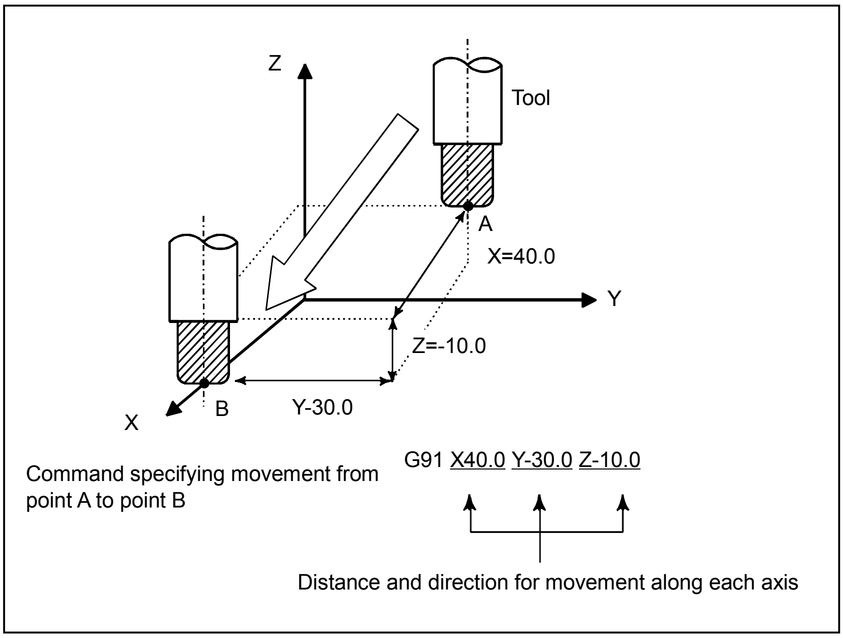


Command specifying movement from point A to point B

```
G90X10.0 Y30.0 Z20.0;
```

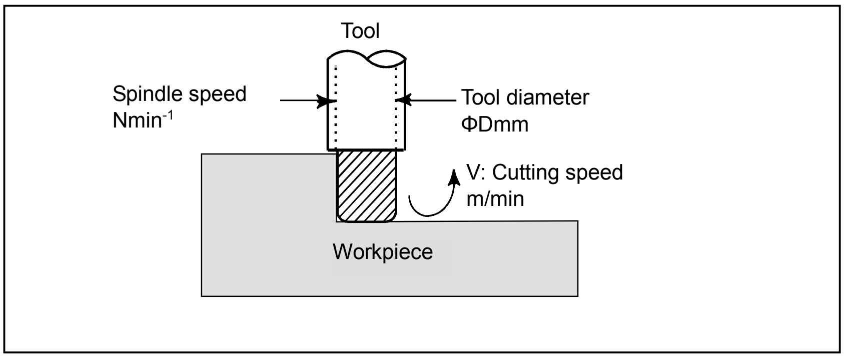
-Incremental command

Specify the distance from the previous tool position to the next tool position.


Z
Tool
A
X=40.0
Y
Z=-10.0
X
B
Y-30.0
Command specifying movement from point A to point B
G91 X40.0 Y-30.0 Z-10.0
Distance and direction for movement along each axis

1.4 CUTTING SPEED - SPINDLE SPEED FUNCTION

The speed of the tool with respect to the workpiece when the workpiece is cut is called the cutting speed.
As for the CNC, the cutting speed can be specified by the spindle speed in min^{-1} unit.



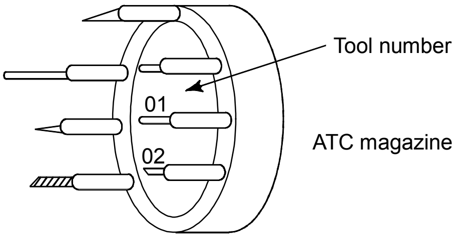
<When a workpiece should be machined with a tool 100 mm in diameter at a cutting speed of 80 m/min. >
The spindle speed is approximately 250 min^{-1}, which is obtained from N=1000v/πD. Hence the following command is required:

```
S250;
```

Commands related to the spindle speed are called the spindle speed function (See II-9) .

1.5 SELECTION OF TOOL USED FOR VARIOUS MACHINING - TOOL FUNCTION

When drilling, tapping, boring, milling or the like, is performed, it is necessary to select a suitable tool.
When a number is assigned to each tool and the number is specified in the program, the corresponding tool is selected.



Example

<When No.01 is assigned to a drilling tool>
When the tool is stored at location 01 in the ATC magazine,

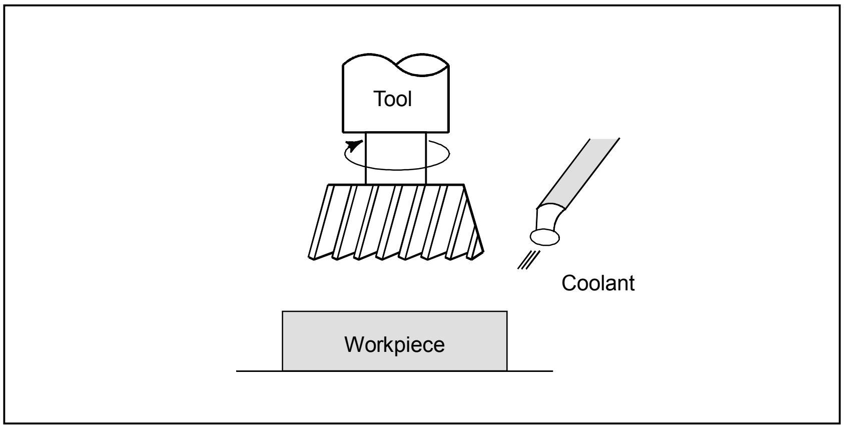
T01

the tool can be selected by specifying T01. This is called the tool function (See II-10).

1.6 COMMAND FOR MACHINE OPERATIONS - MISCELLANEOUS FUNCTION

When machining is actually started, it is necessary to rotate the spindle, and feed coolant. For this purpose, on-off operations of spindle motor and coolant valve should be controlled.

The function of specifying the on-off operations of the components of the machine is called the miscellaneous function. In general, the function is specified by an M code (See II-11).
For example, when M03 is specified, the spindle is rotated clockwise at the specified spindle speed.



1.7 PROGRAM CONFIGURATION

A group of commands given to the CNC for operating the machine is called the program. By specifying the commands, the tool is moved along a straight line or an arc, or the spindle motor is turned on and off. In the program, specify the commands in the sequence of actual tool movements.

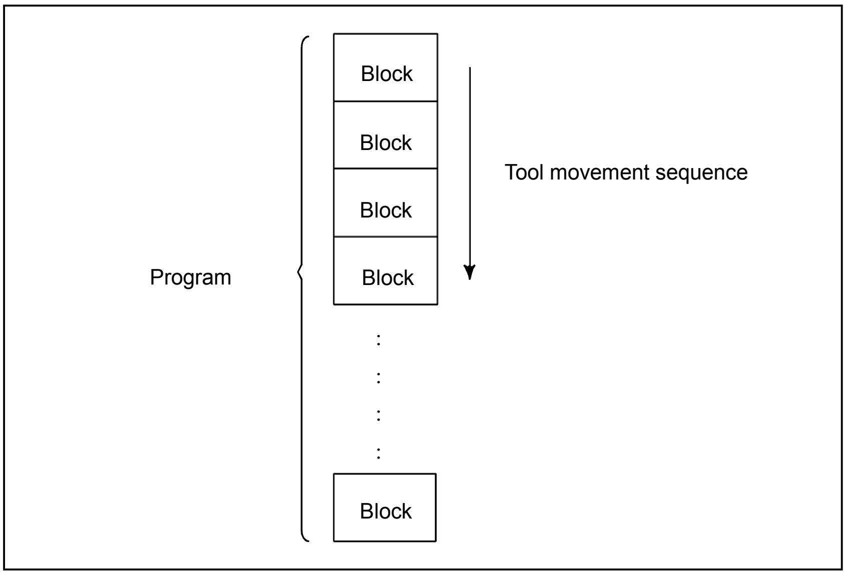


Fig.1.7 (a) Program configuration

A group of commands at each step of the sequence is called the block. The program consists of a group of blocks for a series of machining. The number for discriminating each block is called the sequence number, and the number for discriminating each program is called the program number (See II-12).

Explanation

- Block

The block and the program have the following configurations.

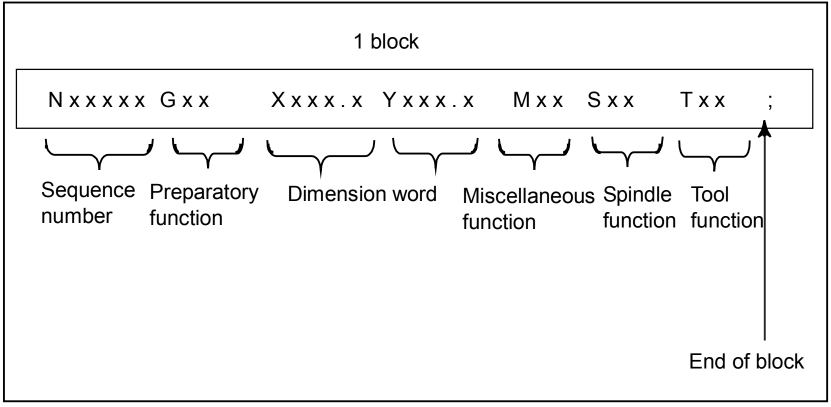


Fig.1.7 (b) Block configuration

A block starts with a sequence number to identify the block and ends with an end-of-block code.

This manual indicates the end-of-block code by ; (LF in the ISO code and CR in the EIA code).

- Program

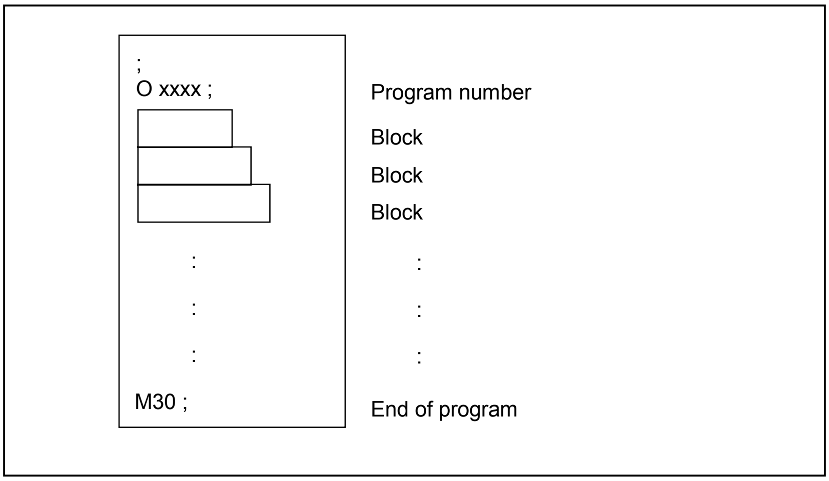


Fig.1.7 (c) Program configuration

Normally, a program number is specified after the end-of-block (;) code at the beginning of the program, and a program end code (M02 or M30) is specified at the end of the program.

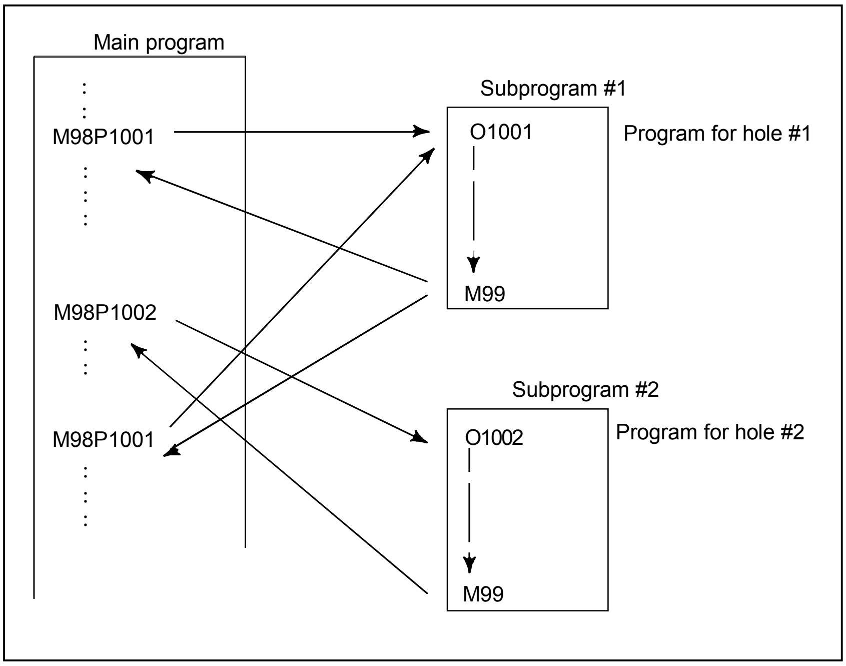
- Main program and subprogram

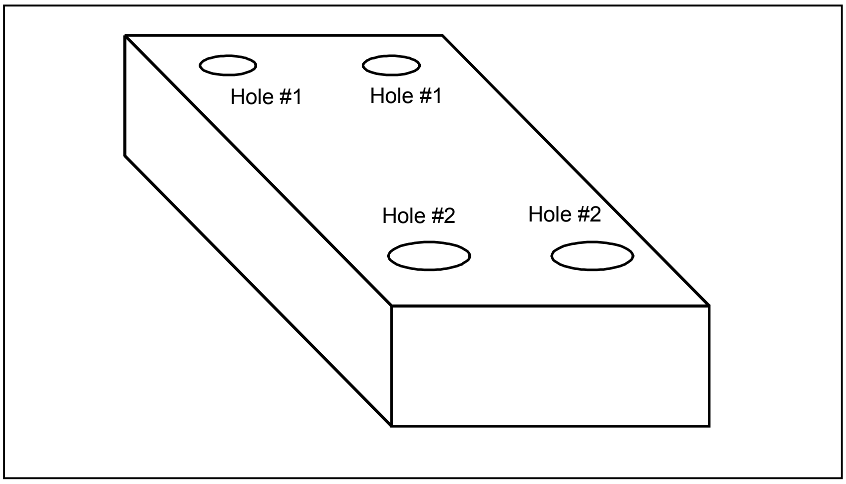
When machining of the same pattern appears at many portions of a program, a program for the pattern is created. This is called the subprogram.

On the other hand, the original program is called the main program.

When a subprogram execution command appears during execution of the main program, commands of the subprogram are executed.

When execution of the subprogram is finished, the sequence returns to the main program.



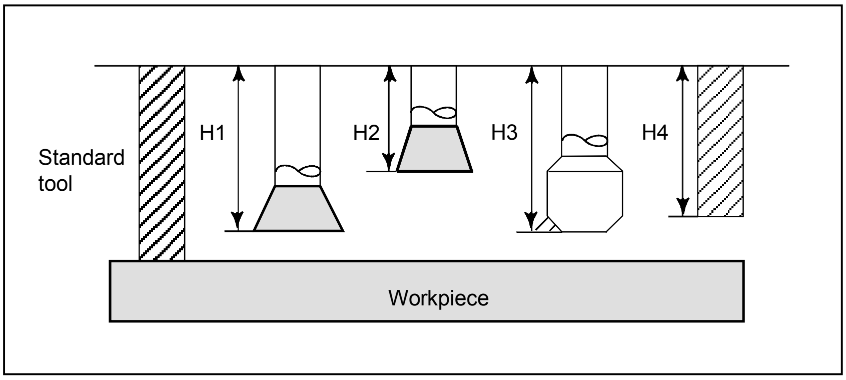


1.8 TOOL FIGURE AND TOOL MOTION BY PROGRAM

Explanation

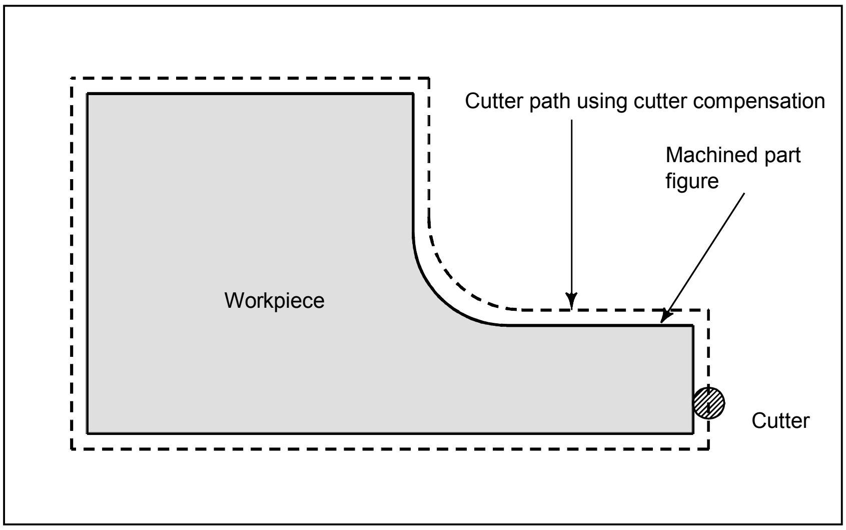
-Machining using the end of cutter - Tool length compensation function

Usually, several tools are used for machining one workpiece.
The tools have different tool length. It is very troublesome to change the program in accordance with the tools.
Therefore, the length of each tool used should be measured in advance. By setting the difference between the length of the standard tool and the length of each tool in the CNC (Operation : see II-9), machining can be performed without altering the program even when the tool is changed. This function is called tool length compensation (See II-14.1).



-Machining using the side of cutter - Cutter compensation function

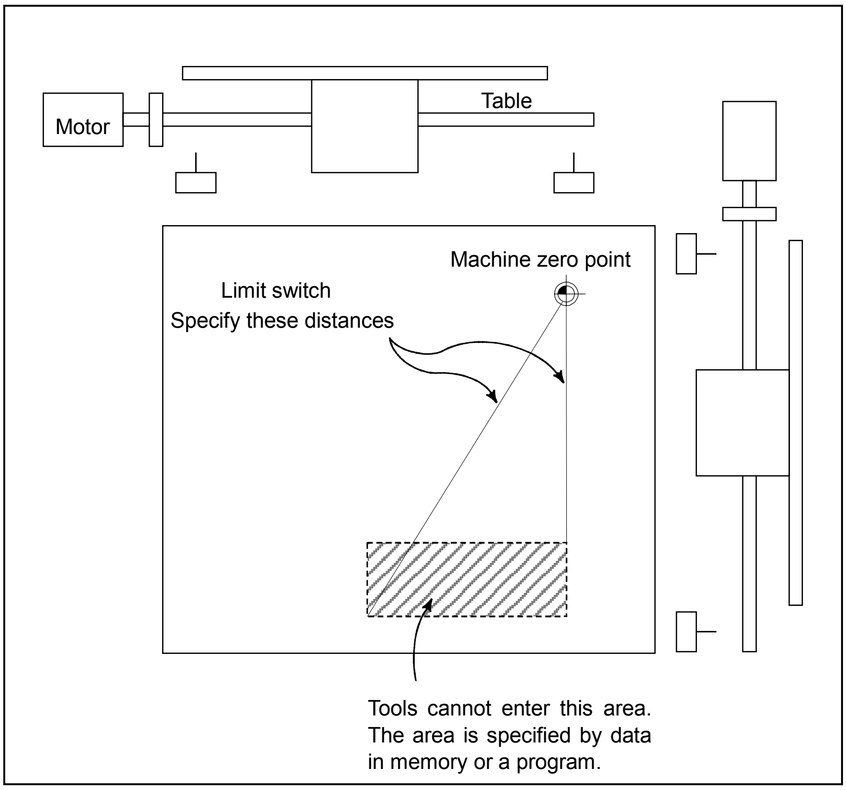
Because a cutter has a radius, the center of the cutter path goes around the workpiece with the cutter radius deviated.



If radius of cutters are stored in the CNC (See Operation II-9), the tool can be moved by cutter radius apart from the machining part figure. This function is called cutter compensation (See II-14.3, 14.4).

1.9 TOOL MOVEMENT RANGE - STROKE

Limit switches are installed at the ends of each axis on the machine to prevent tools from moving beyond the ends.
The range in which tools can move is called the stroke.



Besides strokes defined with limit switches, the operator can define an area which the tool cannot enter using a program or data in memory. This function is called stroke check (see Operation II-6.3).

2 CONROLLED AXES

2.1 CONTROLLED AXES

Series 15*i*/150*i*

Item	Standard type	Multiple axes type
No. of basic controlled axes	3 axes (2 axes)	
Controlled axes expansion (total)	Max. 10 axes (Cs axis is 2 axes)	Max. 24 axes
Basic simultaneously controlled axes	2 axes	
Simultaneously controlled axes expansion (total)	Up to Max. control axes (Cs axis is disabled.)	

2.2 AXIS NAME

Names of axes can be optionally selected from X, Y, Z, A, B, C, U, V, and W. They can be set by parameter No. 1020.

Explanation

- Axis name expansion function

With the optional axis name expansion function, I, J, K, and E can also be used as axis names.

When I, J, K, and E are used as the names of axes, these addresses have the following functions and restrictions:

(1) These addresses are addresses for coordinate words.
  Example) G17I-K- ; The I-K plane is selected.
(2) The numeric values to be specified must consist of up to 8 digits.
(3) A decimal point can be input.
  If a decimal point is omitted, its position is determined according to the increment system of the axis for that address.
  Example) G00 E0.5 I 100 K 100.0;
(4) A signed value can be input.
  Example) G01 E-10.5 F100;

Limitation

- Axis name expansion function

- When I, J, K, and E are used as axis names, they cannot be used for the ordinary purposes listed below.

Address	G code or variable	Normal use	Used for controlled axes	Remarks
I,J,K	G02 G03	Center position of an arc	Coordinate words for I, J, and K	Use an R command to specify the center.
	G41 G42	Three- dimensional offset vector	Coordinate words for I, J, and K	Three- dimensional tool compensation is disabled.
	G76 G87	Canned cycle shift amount	Coordinate words for I, J, and K	An amount of shift cannot be specified.
	G22	Stroke limit coordinates	Stroke limit coordinates	A limit position cannot be specified.
	G65 G66 G66.1	Argument	Argument	The position of the decimal point is determined by the increment system.
K	G06.2	Not value of NURBS interpolation	K-axis coordinate word	NURBS interpolation is disabled.
E	G33	Screw pitch (number of thread for inch screws)	E-axis coordinate word	The number of threads for inch screws cannot be specified in G33 threading.
	#4108	Macro variable, address E continuous-state information	No meaning	Custom macro variable #4108 is unavailable.

⚠ **CAUTION**
When this function is used, the second auxiliary function cannot be used.

2.3 INCREMENT SYSTEM

The increment system uses least input increment (for input) and least command increment (for output). The least input increment is the least increment for programming the travel distance. The least command increment is the least increment for moving the tool on the machine.
Both increments are represented in mm, inches, or deg.
There are five types of increment systems, as shown in Table2.3 (a). One of the five types can be set for each axis by using bits 0 (ISA), 1 (ISC), 2 (ISD), and 3(ISE) of Parameter No. 1012.
The least input increment is in either metric or inch units. One can be selected using a G code (G20, G21) or setting parameter.
The least command increment is in either metric or inch units depending on the machine tool. Set metric or inch in bit 1 (INM) of parameter No. 1002 in advance.
The metric and inch systems cannot be used together. There are functions that cannot be used for axes with different unit systems (circular interpolation, cutter compensation, and so forth).
IS-D and IS-E are optional.
For the increment system, see the manual provided by the machine tool builder manual.

Table2.3 (a) Increment system

Name of increment system	Least input increment	Least command increment	Maximum stroke
IS-A	0.01 mm	0.01 mm	999999.99 mm
	0.001 inch	0.001 inch	99999.999 inch
	0.01 deg	0.01 deg	999999.99 deg
IS-B	0.001 mm	0.001 mm	99999.999 mm
	0.0001 inch	0.0001 inch	9999.9999 inch
	0.001 deg	0.001 deg	99999.999 deg
IS-C	0.0001 mm	0.0001 mm	9999.9999 mm
	0.00001 inch	0.00001 inch	999.99999 inch
	0.0001 deg	0.0001 deg	9999.9999 deg
IS-D	0.00001 mm	0.00001 mm	9999.99999 mm
	0.000001 inch	0.000001 inch	999.999999 inch
	0.00001 deg	0.00001 deg	9999.99999 deg
IS-E	0.000001 mm	0.000001 mm	999.999999 mm
	0.0000001 inch	0.0000001 inch	99.9999999 inch
	0.000001 deg	0.000001 deg	999.999999 deg

By setting bit 0 (IM0) of parameter No. 1013 for ten-fold input unit, each increment system is set as shown in Table2.3 (b).

Table2.3 (b)

Name of increment system	Least input increment		Least command increment		Maximum stroke
IS-B	0.01	mm	0.001	mm	99999.999 mm
	0.001	inch	0.0001	inch	9999.9999 inch
	0.01	deg	0.001	deg	99999.999 deg
IS-C	0.001	mm	0.0001	mm	9999.9999 mm
	0.0001	inch	0.00001	inch	999.99999 inch
	0.001	deg	0.0001	deg	9999.9999 deg
IS-D	0.0001	mm	0.00001	mm	9999.99999 mm
	0.00001	inch	0.000001	inch	999.999999 inch
	0.0001	deg	0.00001	deg	9999.99999 deg
IS-E	0.00001	mm	0.000001	mm	999.999999 mm
	0.000001	inch	0.0000001	inch	99.9999999 inch
	0.00001	deg	0.000001	deg	999.999999 deg

2.4 MAXIMUM STROKE

Maximum stroke = Least command increment × 99999999
(For IS-D and IS-E, 999999999)
See 2.3 Increment System.

Table2.4 (a) Maximum stroke

Increment system		Maximum stroke
IS-A	Metric machine system	±999999.99 mm ±999999.99 deg
	Inch machine system	±99999.999 inch ±999999.99 deg
IS-B	Metric machine system	±99999.999 mm ±99999.999 deg
	Inch machine system	±9999.9999 inch ±99999.999 deg
IS-C	Metric machine system	±9999.9999 mm ±9999.9999 deg
	Inch machine system	±999.99999 inch ±9999.9999 deg
IS-D	Metric machine system	±9999.99999 mm ±9999.99999 deg
	Inch machine system	±999.999999 inch ±9999.99999 deg
IS-E	Metric machine system	±999.999999 mm ±999.999999 deg
	Inch machine system	±99.9999999 inch ±999.999999 deg

NOTE

1 A command exceeding the maximum stroke cannot be specified.

2 The actual stroke depends on the machine tool.

3 PREPARATORY FUNCTION (G FUNCTION)

A preparatory function is specified using a numeric value following address G. This determines the meanings of the commands specified in the block. G codes are divided into the following two types:

Type	Meaning
One-shot G code	The G code is effective only in the block in which it is specified.
Continuous-state G code	The G code is effective until another G code of the same group is specified.

(Example) G01 and G00 are continuous-state G codes.

```
G01 X_ :  ┐
    Z_ :  │  G01 is effective in this range.
    X_ :  ┘
G00 Z_ :
```

Explanation

1. G codes marked ◤ are the initial G codes set when the power is turned on or when the system is reset.
   For G00 and G01, G17, G18, and G19, G43, G44, and G49, G94 and G95, and G90 and G91, the G code status can be selected by setting bit 0 (G01), bit 1 (G90), bit 2 (G43), bit 3 (G44), bit 4 (G95), and bit 5 (G18) of parameter No. 2401.
   For G20 and G21, the system enters the state immediately before the power is turned off or the reset button is pressed.
2. G codes other than G10 and G11 are one-shot G codes.
3. When a G code not listed in the G code list is specified, or a G code that has no corresponding option is specified, P/S alarm No. 010 is output.
4. Multiple G codes can be specified in the same block if each G code belongs to a different group. If multiple G codes that belong to the same group are specified in the same block, only the last G code specified is valid.
5. If a G code belonging to group 01 is specified in a canned cycle, the canned cycle is cancelled. This means that the same state set by specifying G80 is set. Note that the G codes in group 01 are not affected by a G code specifying a canned cycle.
6. G codes are indicated by group.
7. The group of G60 is switched according to the setting of the MDL bit (bit 0 of parameter 7616). (When the MDL bit is set to 0, the 00 group is selected. When the MDL bit is set to 1, the 01 group is selected.)

Table3 G code list

Code	Group	Function	
◤ G00	01	Positioning	
◤ G01		Linear interpolation	
G02		Circular interpolation/Helical interpolation CW	
G03		Circular interpolation/Helical interpolation CCW	
G02.2		Involute interpolation CW	
G03.2		Involute interpolation CCW	
G02.3		Exponential interpolation CW	
G03.3		Exponential interpolation CCW	
G02.4		3-dimensional circular interpolation	
G03.4		3-dimensional circular interpolation	
G06.1		Spline interpolation	
G06.2		NURBS interpolation	
G04	00	Dwell	
G05.1		Multi-buffer	
G07		Hypothetical axis interpolation	
G07.1		Cylindrical interpolation	
G09		Exact stop	
G10		Programmable data input	
G10.1		PMC Data setting	
G10.6		Tool retract & recover	
G10.9		Programmable diameter/radius specification switching function	
G11		Programmable data input mode cancel	
G12.1	26	Polar coordinate interpolation mode	
◤ G13.1		Polar coordinate interpolation cancel mode	
◤ G15	17	Polar coordinates command cancel	
G16		Polar coordinates command	
◤ G17	02	XpYp plane	where, Xp:X axis or a parallel axis Yp:Y axis or a parallel axis Zp:Z axis or a parallel axis
◤ G18		ZpXp plane	
G19		YpZp plane	
G20	06	Inch input	
G21	06	Metric input	
◤ G22	04	Stored stroke check function on	
G23		Stored stroke check function off	
◤ G25	25	Spindle speed fluctuation detection off	
G26		Spindle speed fluctuation detection on	
G27	00	Reference position return check	
G28		Return to reference position	
G29		Return from reference position	
G30		Return to 2nd, 3rd, or 4th reference position	
G30.1		Return to floating reference position	
G31		Skip function	
G31.1		Multistage skip function 1	
G31.2		Multistage skip function 2	
G31.3		Multistage skip function 3	
G31.4		Multistage skip function 4	
G31.8		EGB skip function	
G31.9		High succession skip function	

Table3 G code list

Code	Group	Function
G33	01	Threading
G37	00	Automatic tool length measurement
G38		Cutter compensation C vector retention
G39		Cutter compensation C corner rounding
◤ G40	07	Cutter compensation cancel / Three dimensional compensation cancel
G41		Cutter compensation left / Three dimensional compensation
G42		Cutter compensation right
G41.2		3 dimensional cutter compensation left
G42.2		3 dimensional cutter compensation right
G41.3		Leading edge offset
◤ G40.1	19	Normal direction control cancel mode
G41.1		Normal direction control left side on
G42.1		Normal direction control right side on
G43	08	Tool length compensation (+ve)
G43.1		Tool length compensation in tool axis direction
G44		Tool length compensation (-ve)
G45	00	Tool offset increase
G46		Tool offset decrease
G47		Tool offset double increase
G48		Tool offset double decrease
◤ G49	08	Tool length compensation cancel
◤ G50	11	Scaling cancel
G51		Scaling
◤ G50.1	18	Programmable mirror image cancel
G51.1		Programmable mirror image
G52	00	Local coordinate system setting
G53		Machine coordinate system selection
◤ G54	14	Workpiece coordinate system 1 selection
G54.1		Additional workpiece coordinate system selection
G54.2		Fixture offset selection
G55		Workpiece coordinate system 2 selection
G56		Workpiece coordinate system 3 selection
G57		Workpiece coordinate system 4 selection
G58		Workpiece coordinate system 5 selection
G59		Workpiece coordinate system 6 selection
G60	00/01	Unidirectional positioning
G61	15	Exact stop mode
G62		Automatic corner override
G63		Tapping mode
◤ G64		Cutting mode
G65	00	Macro call
G66	12	Macro modal call A
G66.1		Macro modal call B
◤ G67		Macro modal call cancel
G68	16	Coordinate system rotation
◤ G69		Coordinate system rotation cancel
G72.1	00	Rotation copy
G72.2		Linear copy

Table3 G code list

Code	Group	Function
G73	09	Peck drilling cycle
G74		Counter tapping cycle
G76		Fine boring cycle
◤ G80		Canned cycle cancel / external operation function cancel / Electronic gear box synchronous cancel (Command for hobbing machine or 1 axis)
G81		Drill cycle, stop boring /external operation function / Electronic gear box synchronous start (Command for hobbing machine or 1 axis)
G80.5		Electronic gear box synchronous cancel (Command for 2 axes)
G81.5		Electronic gear box synchronous start (Command for 2 axes)
G81.1	00	Chopping mode on
G82	09	Drill cycle, counter boring
G83		Peck drilling cycle
G84		Tapping cycle
G84.2		Rigid tapping cycle
G84.3		Reverse rigid tapping cycle
G85		Boring cycle
G86		Boring cycle
G87		Back boring cycle
G88		Boring cycle
G89		Boring cycle
◤ G90	03	Absolute command
◤ G91		Incremental command
◤ G92	00	Setting for workpiece coordinate system or clamp at maximum spindle speed
◤ G92.1		Workpiece coordinate system preset
G93	05	Inverse time feed
◤ G94		Feed per minute
G95		Feed per rotation
G96	13	Constant surface speed control
◤ G97		Constant surface speed control cancel
◤ G98	10	Return to initial level in canned cycle
G99		Return to R-point level in canned cycle

4 INTERPOLATION FUNCTIONS

4.1 POSITIONING (G00)

The G00 command moves a tool to the position in the workpiece system specified with an absolute or an incremental command at a rapid traverse rate.
In the absolute command, coordinate value of the end point is programmed.
In the incremental command the distance the tool moves is programmed.

Format

G00 IP_ ;

IP_ : For an absolute command, the coordinates of an end position, and for an incremental commnad, the distance the tool moves.

Explanation

Either of the following tool paths can be selected according to bit 1 of parameter LRP No. 1401.

- Nonlinear interpolation positioning
  The tool is positioned with the rapid traverse rate for each axis separately. The tool path is normally straight.
- Linear interpolation positioning
  The tool path is the same as in linear interpolation (G01). The tool is positioned within the shortest possible time at a speed that is not more than the rapid traverse rate for each axis.

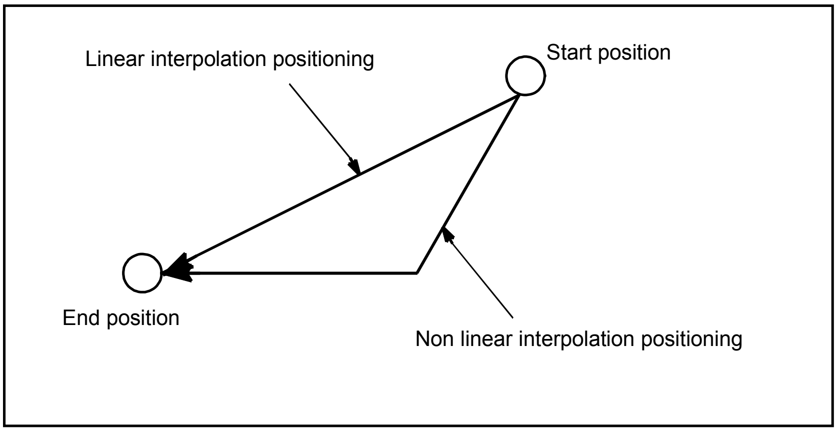

Linear interpolation positioning
Start position
End position
Non linear interpolation positioning

Fig.4.1 (a) Tool path

The rapid traverse rate in G00 command is set to the parameter No. 1420 for each axis independently by the machine tool builder. In the posiitoning mode actuated by G00, the tool is accelerated to a predetermined speed at the start of a block and is decelerated at the end of a block. Execution proceeds to the next block after confirming the in-position.
"In-position " means that the feed motor is within the specified range.

This range is determined by the machine tool builder by setting to parameter (No. 1827).
In-position check for each block can be disabled by setting bit 0 (CIP) of parameter No.1000 accordingly.

Limitation

(1) The rapid traverse rate cannot be specified in the address F.
(2) Even if linear interpolation positioning is specified, nonlinear interpolation positioning is used in the following cases. Therefore, be careful to ensure that the tool does not foul the workpiece.
  - G28 specifying positioning between the reference and intermediate positions.
  - G53

4.2 SINGLE DIRECTION POSITIONING (G60)

For accurate positioning without play of the machine (backlash), final positioning from one direction is available.

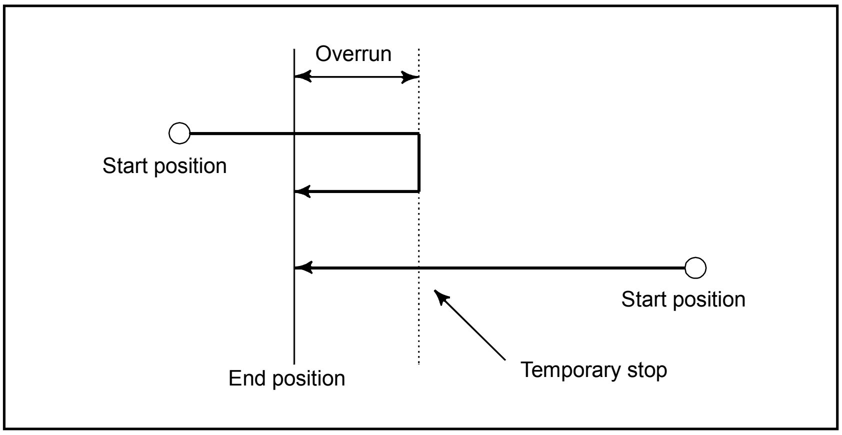


Fig.4.2 (a) Direction positioning process

Format

G60 IP_ ;

IP_ : For an absolute command, the coordinates of an end position, and for an incremental commnad, the distance the tool moves.

Explanation

An overrun and a positioning direction are set by the parameter (No. 6820).
Even when a commanded positioning direction coincides with that set by the parameter, the tool stops once before the end point.
G60, which is an one-shot G-code, can be used as a modal G-code in group 01 by setting 1 to the parameter (No. 7616 bit 0 G60).
This setting can eliminate specifying a G60 command for every block.
Other specifications are the same as those for an one-shot G60 command.
When an one-shot G code is specified in the single direction positioning mode, the one-shot G command is effective like G codes in group 01.

(Example)

```
When one-shot G60 commands        When modal G60 command
are used.                         is used.
   :                                 :
   :                                 :
   G90;                           G90 G60;      Single direction
   G60 X0 Y0;                         X0 Y0;    positioning mode start
   G60 X100;   Single direction       X100;     Single direction
   G60 Y100;   positioning            Y100;     positioning
   G04 X10;                       G04 X10;
   G00 X0 Y0;                     G00 X0 Y0;    Single direction
                                                positioning mode
                                                cancel
```

Limitation

- During canned cycle for drilling(G81to G89, G73, G74, G76), no single direction positioning is effected in drilling axis.
- No single direction positioning is effected in an axis for which no overrun has been set by the parameter.
- When the move distance 0 is commanded, the single direction positioning is not performed.
- The direction set to the parameter is not effected by mirror image.
- The single direction positioning does not apply to the shift motion in the canned cycles of G76 and G87.
- The single Direction Positioning is always non linear interpolation type positioning.
- The parameter FPI (No. 1000#3) can be used to determine whether the in-position check is to be done or not at the temporary stop point

4.3 LINEAR INTERPOLATION (G01)

Tools can move along a line

Format

G01 IP_ F_ ;

IP_ : For an absolute command, the coordinates of an end point , and for an incremental commnad, the distance the tool moves.

F_ : Speed of tool feed (Feedrate)

Explanation

A tools move along a line to the specified position at the feedrate specified in F. The feedrate specified in F is effective until a new value is specified. It need not be specified for each block.

- The feedrate commanded by the F code is measured along the tool path. If the F code is not commanded, the feedrate is regarded as zero.
- The feedrate of each axis direction is as follows.

G01 $\alpha_{\alpha}\,\beta_{\beta}\,\gamma_{\gamma}\,\zeta_{\zeta}$ Ff ;

Feed rate of α axis direction : $F\alpha = \frac{\alpha}{L} \times f$

Feed rate of β axis direction : $F\beta = \frac{\beta}{L} \times f$

Feed rate of γ axis direction : $F\gamma = \frac{\gamma}{L} \times f$

Feed rate of ζ axis direction : $F\zeta = \frac{\zeta}{L} \times f$

$L = \sqrt{\alpha^2 + \beta^2 + \gamma^2 + \zeta^2}$

- The feed rate of the rotary axis is commanded in the unit of deg/min (the unit is decimal point position).
- When the straight line axis α (such as X, Y, or Z) and the rotating axis β (such as A, B, or C) are linearly interpolated, the feed rate is that in which the tangential feed rate in the α and β Cartesian coordinate system is commanded by F(mm/min).

  β-axis feedrate is obtained ; at first, the time required for distribution is calculated by using the above formula, then the β-axis feedrate unit is changed to deg 1min.

(Example)

G91 G01 X20.0 C40.0 F300.0 ;

This changes the unit of the C axis from 40.0 deg to 40mm with metric input.

The time required for distribution is calculated as follows:

$$\frac{\sqrt{20^2 + 40^2}}{300} \cong 0.14907 \ \text{min}$$

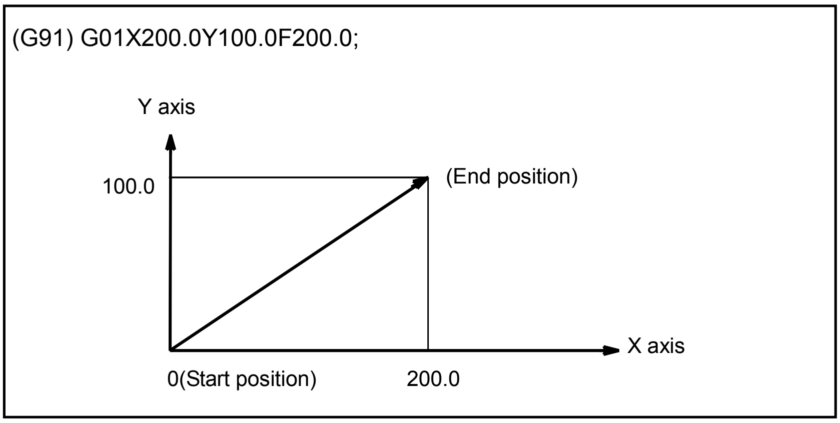
The feed rate for the C axis is

$$\frac{40 \ \text{deg}}{0.14907 \ \text{min}} \cong 268.3 \text{deg} / \text{min}$$

- In simultaneous 3 axes control, the feed rate is calculated the same way as in 2 axes control.

Example

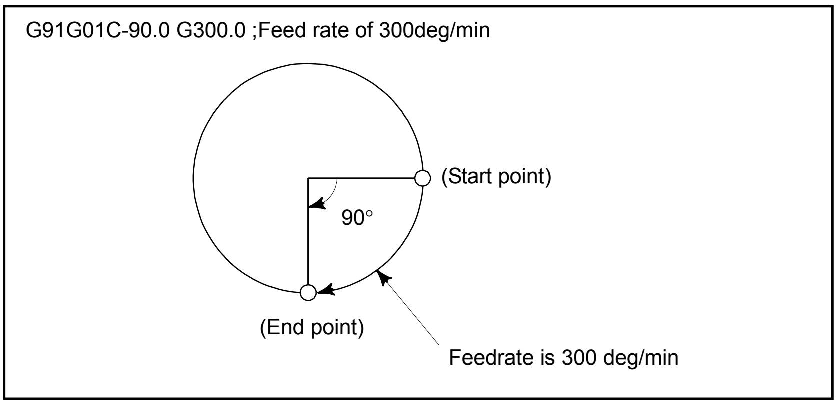
- Linear interpolation



```
(G91) G01X200.0Y100.0F200.0;
```

Fig.4.3 (a) Linear interpolation

- Feedrate for the rotation axis



```
G91G01C-90.0 G300.0 ;Feed rate of 300deg/min
```

Fig.4.3 (b) Feedrate for the rotation axis

4.4 CIRCULAR INTERPOLATION (G02,G03)

The command below will move a tool along a circular arc.

Format

Arc in the XpYp plane

$$\text{G17}\begin{Bmatrix}\text{G02}\\ \text{G03}\end{Bmatrix}\text{Xp\_ Yp\_}\begin{Bmatrix}\text{I\_ J\_}\\ \text{R\_}\end{Bmatrix}\text{F\_ ;}$$

Arc in the ZpXp plane

$$\text{G18}\begin{Bmatrix}\text{G02}\\ \text{G03}\end{Bmatrix}\text{Zp\_ Xp\_}\begin{Bmatrix}\text{K\_ I\_}\\ \text{R\_}\end{Bmatrix}\text{F\_ ;}$$

Arc in the YpZp plane

$$\text{G19}\begin{Bmatrix}\text{G02}\\ \text{G03}\end{Bmatrix}\text{Yp\_ Zp\_}\begin{Bmatrix}\text{J\_ K\_}\\ \text{R\_}\end{Bmatrix}\text{F\_ ;}$$

Description of the Command Format

Command	Description
G17	Specification of arc on XpYp plane
G18	Specification of arc on ZpXp plane
G19	Specification of arc on YpZp plane
G02	Circular Interpolation Clockwise direction (CW)
G03	Circular Interpolation Counterclockwise direction (CCW)
Xp_	Command values of X axis or its parallel axis (set by parameter No. 1022)
Yp_	Command values of Y axis or its parallel axis (set by parameter No. 1022)
Zp_	Command values of Z axis or its parallel axis (set by parameter No. 1022)
I_	Xp axis distance from the start point to the center of an arc with sign (with sign)
J_	Yp axis distance from the start point to the center of an arc with sign (with sign)
K_	Zp axis distance from the start point to the center of an arc with sign (with sign)
R_	Arc radius (with sign)
F_	Feedrate along the arc

Explanation

- Direction of the circular interpolation

"Clockwise"(G02) and "counterclockwise"(G03) on the XpYp plane (ZpXp plane or YpZp plane) are defined when the XpYp plane is viewed in the positive-to-negative direction of the Zp axis (Yp axis or Xp axis, respectively) in the Cartesian coordinate system. See the figure below.

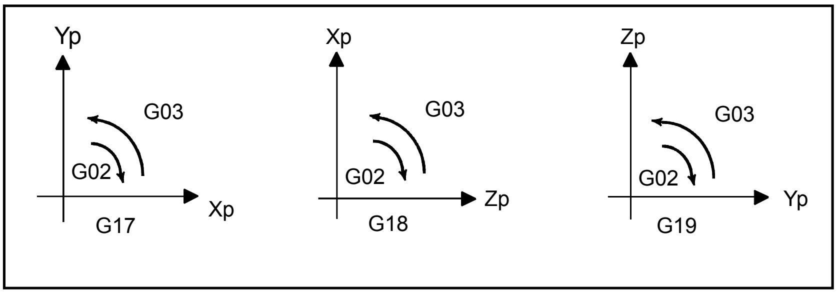


Fig.4.4 (a) Direction of the circular interpolation

- Distance moved on an arc

The end point of an arc is specified by address Xp, Yp or Zp, and is expressed as an absolute or incremental value according to G90 or G91. For the incremental value, the distance of the end point which is viewed from the start point of the arc is specified.

- Distance from the start point to the center of arc

The arc center is specified by addresses I, J, and K for the Xp, Yp, and Zp axes, respectively. The numerical value following I, J, or K, however, is a vector component in which the arc center is seen from the start point, and is always specified as an incremental value irrespective of G90 and G91, as shown below.

I, J, and K must be signed according to the direction.

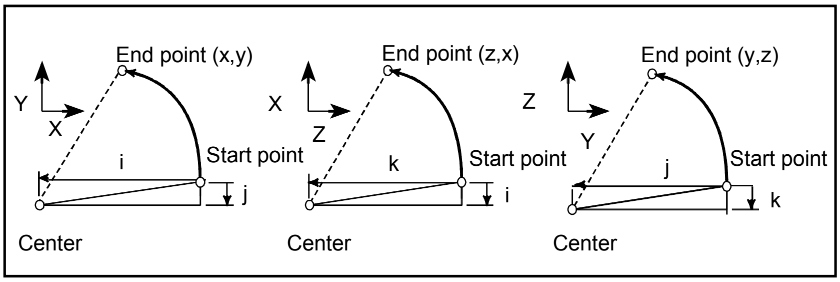


Fig.4.4 (b) Distance from the start point to the center of arc

I0,J0, and K0 can be omitted.

When Xp, Yp , and Zp are omitted (the end point is the same as the start point) and the center is specified with I, J, and K, a 360deg. arc (circle) is specified.

G02 I_ ; Command for a circle

If the difference between the radius at the start point and that at the end point exceeds the permitted value in a parameter (No.2410), an P/S alarm (No.191) occurs.

- Arc radius

The distance between an arc and the center of a circle that contains the arc can be specified using the radius, R, of the circle instead of I, J, and K.
In this case, one arc is less than 180deg., and the other is more than 180deg. are considered. When an arc exceeding 180deg. is commanded, the radius must be specified with a negative value. If Xp, Yp, and Zp are all omitted, if the end point is located at the same position as the start point and when R is used, an arc of 0deg. is programmed G02R ; (The cutter does not move.)

(Example)

For arc (1) (less than 180°)

```
G91 G02 X60.0 Y20.0 R50.0 F300.0 :
```

For arc (2) (greater than 180°)

```
G91 G02 X60.0 Y20.0 R-50.0 F300.0 :
```

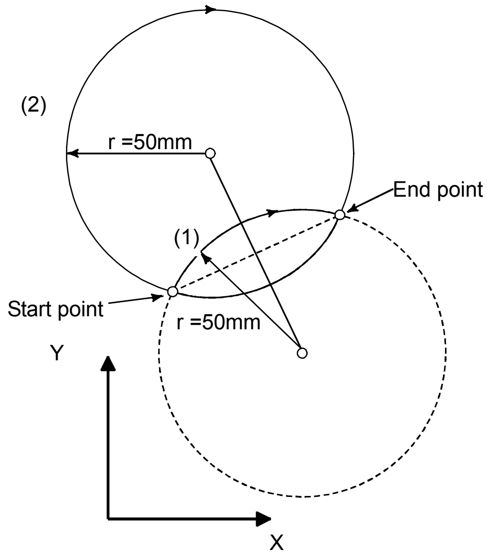


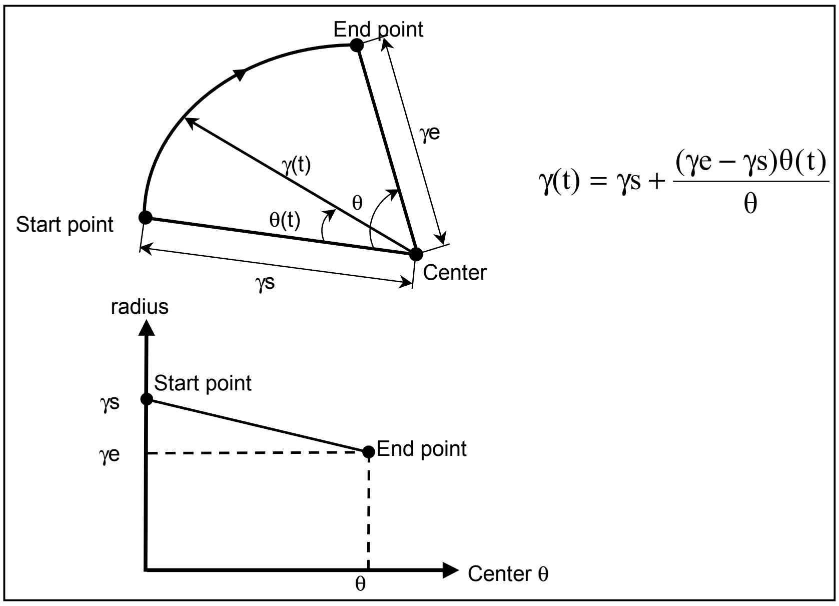
Fig.4.4 (c) Arc radius

- Feedrate

The feedrate in circular interpolation is equal to the feed rate specified by the F code, and the feedrate along the arc (the tangential feedrate of the arc) is controlled to be the specified feedrate.

- Cases where a spiral results

When an end point does not lie on the arc, a spiral results, as shown below.



$$\gamma(t) = \gamma s + \frac{(\gamma e - \gamma s)\theta(t)}{\theta}$$

Fig.4.4 (d) Case Where a Spiral Is Produced

The arc radius changes linearly with the center angle θ(t). Spiral interpolation is performed using a circular command that specifies one arc radius for the start point and another arc radius for the end point. To use spiral interpolation, set a large value in parameter No. 2410, used to specify the limit on the arc radius error.

Limitation

- Simultaneous specification of I, J, K, and R

If I, J, K, and R addresses are specified simultaneously, the arc specified by address R takes precedence and the other are ignored.

- Commandment of an axis not comprising the specified plane

If an axis not comprising the specified plane is commanded, an alarm is displayed.

For example, if axis U is specified as a parallel axis to X axis when plane XY is specified, an P/S alarm (No.0186)is displayed.

- Specifying a semicircle by R

If an arc has a central angle of nearly 180°, an error may occur when the position of the center is calculated. To avoid this error, specify the semicircle by R using two arcs having a central angle of nearly 90° or specify its center position directly by I, J, and K.

Example

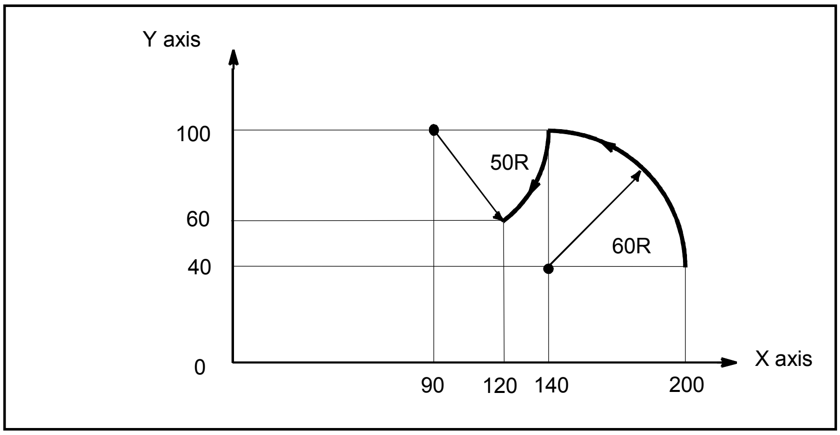


Fig.4.4 (e) Sample program

The above tool path can be programmed as follows ;

(1) In absolute programming

```
G92X200.0 Y40.0 Z0 ;
G90 G03 X140.0 Y100.0R60.0 F300.;
G02 X120.0 Y60.0R50.0 ;
or
G92X200.0 Y40.0Z0 ;
G90 G03 X140.0 Y100.0I-60.0 F300.;
G02 X120.0 Y60.0I-50.0 ;
```

(2) In incremental programming

```
G91 G03 X-60.0 Y60.0  R60.0 F300.;
G02 X-20.0 Y-40.0 R50.0 ;
or
G91 G03 X-60.0 Y60.0  I-60.0 F300. ;
G02 X-20.0 Y-40.0 I-50.0 ;
```

4.5 HELICAL INTERPOLATION (G02,G03)

Helical interpolation which moved helically is enabled by specifying up to two other axes which move synchronously with the circular interpolation by circular commands.

Format

Synchronously with arc of XpYp plane

$$\mathrm{G17} \begin{Bmatrix} \mathrm{G02} \\ \mathrm{G03} \end{Bmatrix} \ \mathrm{Xp\_\ Yp\_} \begin{Bmatrix} \mathrm{I\_\ J\_} \\ \mathrm{R\_} \end{Bmatrix} \alpha\_\ (\beta\_)\ \mathrm{F\_}\ ;$$

Synchronously with arc of ZpXp plane

$$\mathrm{G18} \begin{Bmatrix} \mathrm{G02} \\ \mathrm{G03} \end{Bmatrix} \ \mathrm{Zp\_\ Xp\_} \begin{Bmatrix} \mathrm{K\_\ I\_} \\ \mathrm{R\_} \end{Bmatrix} \alpha\_\ (\beta\_)\ \mathrm{F\_}\ ;$$

Synchronously with arc of YpZp plane

$$\mathrm{G19} \begin{Bmatrix} \mathrm{G02} \\ \mathrm{G03} \end{Bmatrix} \ \mathrm{Yp\_\ Zp\_} \begin{Bmatrix} \mathrm{J\_\ K\_} \\ \mathrm{R\_} \end{Bmatrix} \alpha\_\ (\beta\_)\ \mathrm{F\_}\ ;$$

α, β : Any one axis where circular interpolation is not applied.
Up to two other axes can be specified.

Explanation

The basic command method involves simply adding a move command for one or two axes, other than circular interpolation axes, to a circular interpolation command (see II-4.4).
As the feedrate, either a feedrate tangent to an arc or a tangential feedrate determined by also considering movement along the linear axes can be specified. The feedrate to be specified can be selected by setting bit 2 (HTG) of parameter No. 1401. If HTG is set to 0, a feedrate along an arc is specified by an F command. Therefore, the feedrate on a linear axis is as follows:

$$F \times \frac{\mathit{Length\ of\ linear\ axis}}{\mathit{Length\ of\ circular\ arc}}$$

Determine the feedrate so that the linear axis feedrate does not exceed any of the limit values.

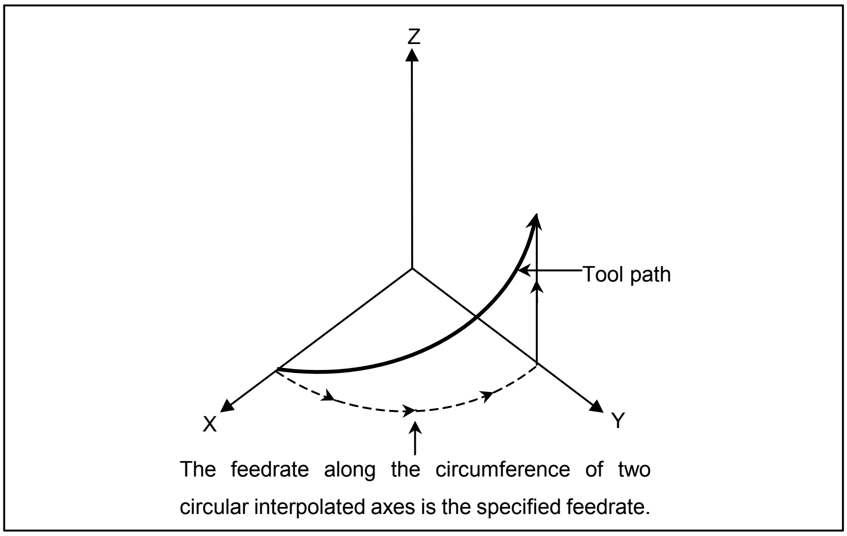

Z
Tool path
X
Y
The feedrate along the circumference of two circular interpolated axes is the specified feedrate.

Fig.4.5 (a) Feedrate When Parameter HTG = 0

When bit 2 (HTG) of parameter No. 1401 is set to 1, the speed command specifies the feedrate along the actual tool path, including movement along the linear axis.

In this case, the feedrate along the arc on the plane is:

$$F \times \frac{Length\ of\ circular\ arc}{\sqrt{(Length\ of\ circular\ arc)^2 + (Length\ of\ linear\ axis)^2}}$$

The feedrate along the linear axis is:

$$F \times \frac{Length\ of\ linear\ axis}{\sqrt{(Length\ of\ circular\ arc)^2 + (Length\ of\ linear\ axis)^2}}$$

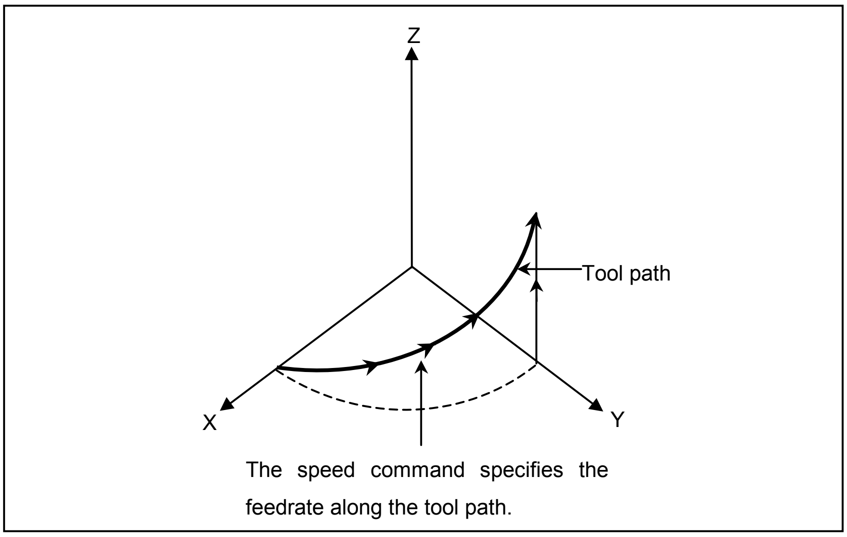

Z
Tool path
X
Y
The speed command specifies the feedrate along the tool path.

Fig.4.5 (b) Feedrate When Parameter HTG = 1

Limitation

- Cutter compensation is applied only for a circular arc.
- Tool offset and tool length compensation cannot be used in a block in which a helical cutting is commanded.

4.6 HELICAL INTERPOLATION B (G02,G03)

Helical interpolation B allows the tool to move in helically. This can be done by specifying the circular interpolation command together with up to four axes.

Format

Synchronously with arc of XpYp plane

$$\mathrm{G17} \left\{ \begin{array}{c} \mathrm{G02} \\ \mathrm{G03} \end{array} \right\} \mathrm{Xp\_ \ Yp\_} \left\{ \begin{array}{c} \mathrm{I\_ \ J\_} \\ \mathrm{R\_} \end{array} \right\} \alpha\_ \ \beta\_ \ \gamma\_ \ \delta\_ \ \mathrm{F\_} \ ;$$

Synchronously with arc of ZpXp plane

$$\mathrm{G18} \left\{ \begin{array}{c} \mathrm{G02} \\ \mathrm{G03} \end{array} \right\} \mathrm{Zp\_ \ Xp\_} \left\{ \begin{array}{c} \mathrm{K\_ \ I\_} \\ \mathrm{R\_} \end{array} \right\} \alpha\_ \ \beta\_ \ \gamma\_ \ \delta\_ \ \mathrm{F\_} \ ;$$

Synchronously with arc of YpZp plane

$$\mathrm{G19} \left\{ \begin{array}{c} \mathrm{G02} \\ \mathrm{G03} \end{array} \right\} \mathrm{Yp\_ \ Zp\_} \left\{ \begin{array}{c} \mathrm{J\_ \ K\_} \\ \mathrm{R\_} \end{array} \right\} \alpha\_ \ \beta\_ \ \gamma\_ \ \delta\_ \ \mathrm{F\_} \ ;$$

Explanation

The command format for helical interpolation B consists of the command format for normal helical interpolation and move commands for two axes. As with normal helical interpolation, the feedrate of helical interpolation B is controlled so that the feedrate of circular interpolation can achieve the specified feedrate.(See II-4.5)
Bit 2 (HTG) of parameter No. 1401 can be used to specify whether the speed command specifies the feedrate along the tangential line of the arc on the plane, or the feedrate along the tangential line of the actual tool path, including movement along the linear axis.

Limitation

- Cutter compensation is applied only for a circular arc.
- Tool offset and tool length compensation cannot be used in a block in which a helical cutting is commanded.

4.7 HYPOTHETICAL AXIS INTERPOLATION (G07)

In helical interpolation, when pulses are distributed with one of the circular interpolation axes set to a hypothetical axis, sine interpolation is enabled.
When one of the circular interpolation axes is set to a hypothetical axis, pulse distribution causes the speed of movement along the remaining axis to change sinusoidally. If the major axis for threading (the axis along which the machine travels the longest distance) is set to a hypothetical axis, threading with a fractional lead is enabled. The axis to be set as the hypothetical axis is specified with G07.

Format

G07 α0; Hypothetical axis setting
G07 α1; Hypothetical axis cancel
Where, α is any one of the addresses of the controlled axes.

Explanation

- Sine interpolation

The α axis is regarded as a hypothetical axis for the period of time from the G07 α0 command until the G07 α1 command appears.
Suppose sine interpolation is performed for one cycle in the YZ plane. The hypothetical axis is then the X axis.

$X^2+Y^2 = r^2$ (r is the radius of an arc.)

$Y = r\mathrm{SIN}\ (\frac{2\pi}{l}Z)$ (l is the distance traveled along the Z-axis in one cycle.)

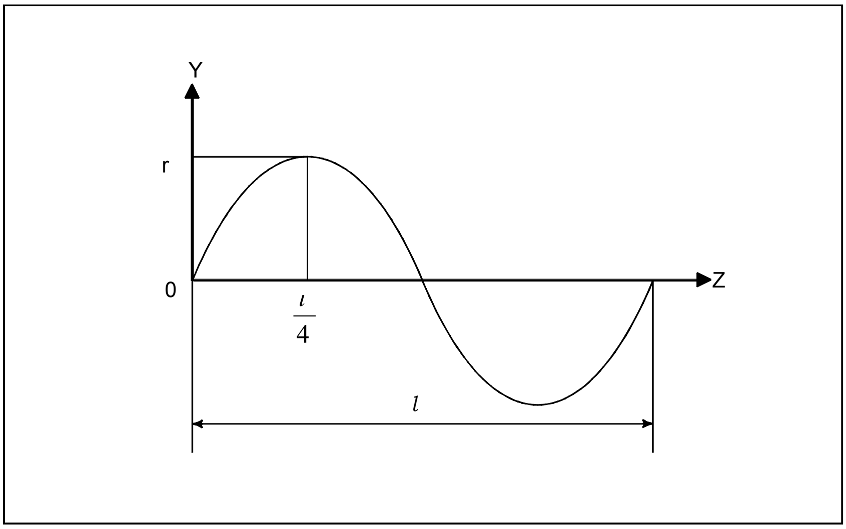


Fig.4.7(a) Sine interpolation

- Interlock, stroke limit, and external deceleration

Interlock, stroke limit, and external deceleration can also apply to the hypothetical axis.

- Handle interrupt

Specify hypothetical axis interpolation only in the incremental mode.

Limitation

- Manual operation

The hypothetical axis can be used only in automatic operation. In manual operation, it is not used, and movement takes place.

- Move command

Specify hypothetical axis interpolation only in the incremental mode.

- Coordinate rotation

Hypothetical axis interpolation does not support coordinate rotation.

Example

- Sine interpolation

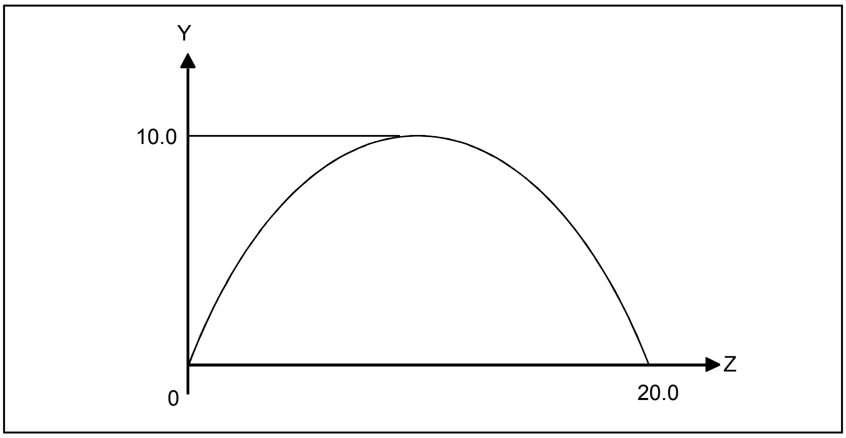


Fig.4.7(b) Sine interpolation (sample)

```
N001  G07  X0  ;
N002  G91  G17  G03  X-20.0  Y0.0  I-10.0  Z20.0  F100 ;
N003  G01  X10.0 ;
N004  G07  X1 ;
```

From the N002 to N003 blocks, the X-axis is set to a hypothetical axis.
The N002 block specifies helical cutting in which the Z-axis is the linear axis. Since no movement takes place along the X axis, movement along the Y-axis is performed while performing sine interpolation along the Z-axis.
In the N003 block, there is no movement along the X-axis, and so the machine dwells until interpolation terminates.

- Changing the feedrate to form a sine curve

(Sample program)

G07Z0 ;	The Z-axis is set to a hypothetical axis.
G02X0Z0I10.0F4. ;	The feedrate on the X-axis changes sinusoidally.
G07Z1 ;	The use of the Z-axis as a hypothetical axis is canceled.

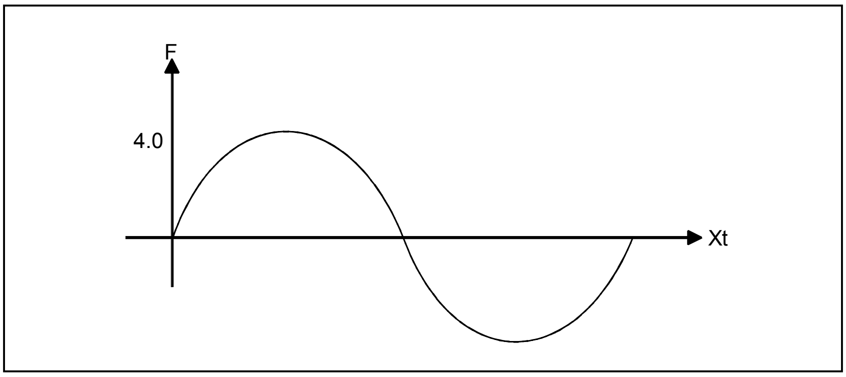


Fig.4.7(c) Changing the feedrate to from a sine curve (sample)

4.8 POLAR COORDINATE INTERPOLATION (G12.1,G13.1)

Polar coordinate interpolation is a function that exercises contour control in converting a command programmed in a Cartesian coordinate system to the movement of a linear axis (movement of a tool) and the movement of a rotary axis (rotation of a workpiece). This function is useful for grinding a cam shaft.

Format

G12.1; Starts polar coordinate interpolation mode (enables polar coordinate interpolation)

⋮ Specify linear or circular interpolation using coordinates in a Cartesian coordinate system consisting of a linear axis and rotary axis (virtual axis).

G13.1 Polar coordinate interpolation mode is cancelled (for not performing polar coordinate interpolation)

Specify G12.1 and G13.1 in Separate Blocks.

Explanation

- Polar coordinate interpolation plane

G12.1 starts the polar coordinate interpolation mode and selects a polar coordinate interpolation plane (Fig.4.8(a)). Polar coordinate interpolation is performed on this plane.

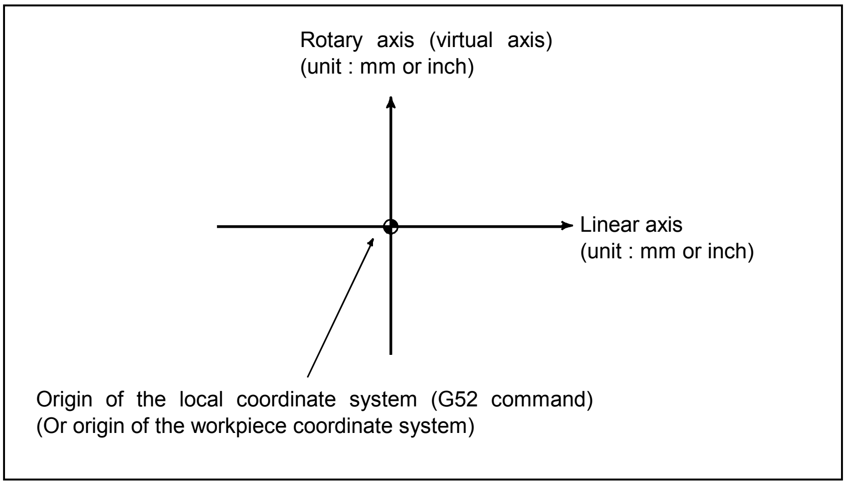

Rotary axis (virtual axis)
(unit : mm or inch)
Linear axis
(unit : mm or inch)
Origin of the local coordinate system (G52 command)
(Or origin of the workpiece coordinate system)

Fig.4.8(a) Polar coordinate interpolation plane.

When the power is turned on or the system is reset, polar coordinate interpolation is canceled (G13.1).

The linear and rotation axes for polar coordinate interpolation must be set in parameters (No. 1032 and 1033) beforehand.

> ⚠ **CAUTION**
>
> The plane used before G12.1 is specified (plane selected by G17, G18, or G19) is canceled. It is restored when G13.1 (canceling polar coordinate interpolation) is specified.
>
> When the system is reset, polar coordinate interpolation is canceled and the plane specified by G17, G18, or G19 is used.

- Distance moved and feedrate for polar coordinate interpolation

The unit for coordinates on the hypothetical axis is the same as the unit for the linear axis (mm/inch)

In the polar coordinate interpolation mode, program commands are specified with Cartesian coordinates on the polar coordinate interpolation plane. The axis address for the rotation axis is used as the axis address for the second axis (virtual axis) in the plane. Whether a diameter or radius is specified for the first axis in the plane is the same as for the rotation axis regardless of the specification for the first axis in the plane.

The virtual axis is at coordinate 0 immediately after G12.1 is specified. Polar interpolation is started assuming the angle of 0 for the position of the tool when G12.1 is specified.

The unit for the feedrate is mm/min or inch/min

Specify the feedrate as a speed (relative speed between the workpiece and tool) tangential to the polar coordinate interpolation plane (Cartesian coordinate system) using F.

- G codes which can be specified in the polar coordinate interpolation mode

G01	Linear interpolation
G02, G03	Circular interpolation
G04,G09	Dwell, Exact stop
G40, G41, G42	Cutter compensation (Polar coordinate interpolation is applied to the path after cutter compensation.)
G65, G66, G67	Custom macro command
G90, G91	Absolute command, incremental command
G94, G95	Feed per minute, feed per revolution

- Circular interpolation in the polar coordinate plane

The addresses for specifying the radius of an arc for circular interpolation (G02 or G03) in the polar coordinate interpolation plane depend on the first axis in the plane (linear axis).

- I and J in the Xp-Yp plane when the linear axis is the X-axis or an axis parallel to the X-axis.
- J and K in the Yp-Zp plane when the linear axis is the Y-axis or an axis parallel to the Y-axis.
- K and I in the Zp-Xp plane when the linear axis is the Z-axis or an axis parallel to the Z-axis.

The radius of an arc can be specified also with an R command.

- Movement along axes not in the polar coordinate interpolation plane in the polar coordinate interpolation mode

The tool moves along such axes normally, independent of polar coordinate interpolation.

- Current position display in the polar coordinate interpolation mode

Actual coordinates are displayed. However, the remaining distance to move in a block is displayed based on the coordinates in the polar coordinate interpolation plane (Cartesian coordinates).

Limitation

- Coordinate system for the polar coordinate interpolation

Before G12.1 is specified, a local coordinate system (or workpiece coordinate system) where the center of the rotary axis is the origin of the coordinate system must be set. In the G12.1 mode, the coordinate system must not be changed (G92, G52, G53, relative coordinate reset, G54 through G59, etc.).

- Tool offset command

The polar coordinate interpolation mode cannot be started or terminated (G12.1 or G13.1) in the tool offset mode (G41 or G42). G12.1 or G13.1 must be specified in the tool offset canceled mode (G40).

- Tool length offset command

Tool length offset must be specified in the polar coordinate interpolation cancel mode before G12.1 is specified. It cannot be specified in the polar coordinate interpolation mode. Furthermore, no offset values can be changed in the polar coordinate interpolation mode.

- Tool offset command

A tool offset must be specified before the G12.1 mode is set. No offset can be changed in the G12.1 mode.

- Program restart

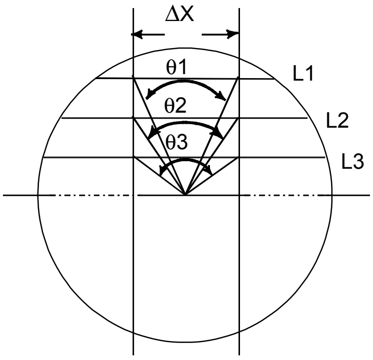
For a block in the G12.1 mode, the program cannot be restarted.

- Cutting feedrate for the rotation axis

Polar coordinate interpolation converts the tool movement for a figure programmed in a Cartesian coordinate system to the tool movement in the rotation axis (C-axis) and the linear axis (X-axis). When the tool moves closer to the center of the workpiece, the C-axis component of the feedrate becomes larger and may exceed the maximum cutting feedrate for the C-axis (set in parameter (No. 1422)), causing an alarm (see the warning below). To prevent the C-axis component from exceeding the maximum cutting feedrate for the C-axis, reduce the feedrate specified with address F or create a program so that the tool (center of the tool when cutter compensation is applied) does not move close to the center of the workpiece.

⚠ **WARNING**

1 Consider lines L1, L2, and L3. ΔX is the distance the tool moves per time unit at the feedrate specified with address F in the Cartesian coordinate system. As the tool moves from L1 to L2 to L3, the angle at which the tool moves per time unit corresponding to ΔX in the Cartesian coordinate system increases from θ1 to θ2 to θ3.
In other words, the C-axis component of the feedrate becomes larger as the tool moves closer to the center of the workpiece. The C component of the feedrate may exceed the maximum cutting feedrate for the C-axis because the tool movement in the Cartesian coordinate system has been converted to the tool movement for the C-axis and the X-axis.



L : Distance (in mm) between the tool center and workpiece center when the tool center is the nearest to the workpiece center
R : Maximum cutting feedrate (deg/min) of the C axis Then, a speed specifiable with address F in polar coordinate interpolation can be given by the formula below.

Specify a speed allowed by the formula. The formula provides a theoretical value; in practice, a value slightly smaller than a theoretical value may need to be used due to a calculation error.

$$F < L \times R \times \frac{\pi}{180} \text{(mm/min)}$$

The speed can be controlled so as to prevent the issue of alarm OT512 (excessive speed). See II-5.5.2 for details.

2 The following functions cannot be used for the rotation axis for polar coordinate interpolation. Using any of these functions results in abnormal operation.
- Roll-over function
- Multiple-rotary axis control
- Index table indexing function

Example

Example of Polar Coordinate Interpolation Program Based on X Axis(Linear Axis) and C Axis (Rotary Axis)

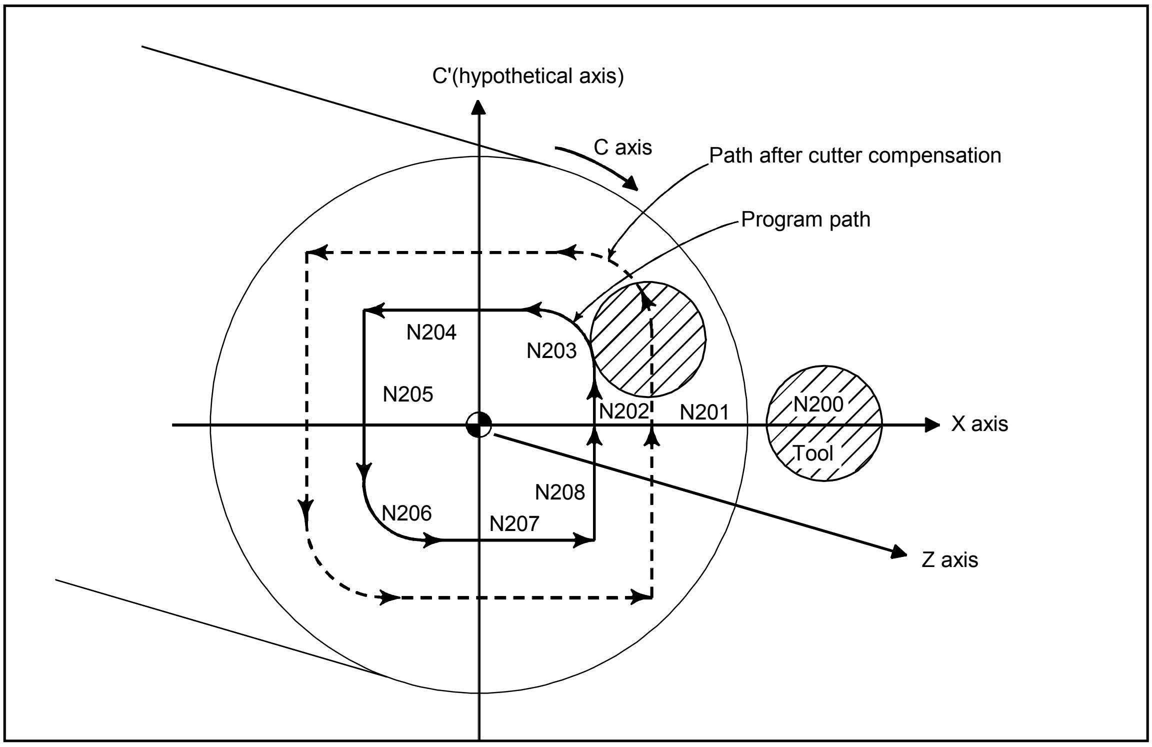


Fig.4.8 (b) Polar Coordinate Interpolation Program Based on X Axis(Linear Axis) and C Axis (Rotary Axis)

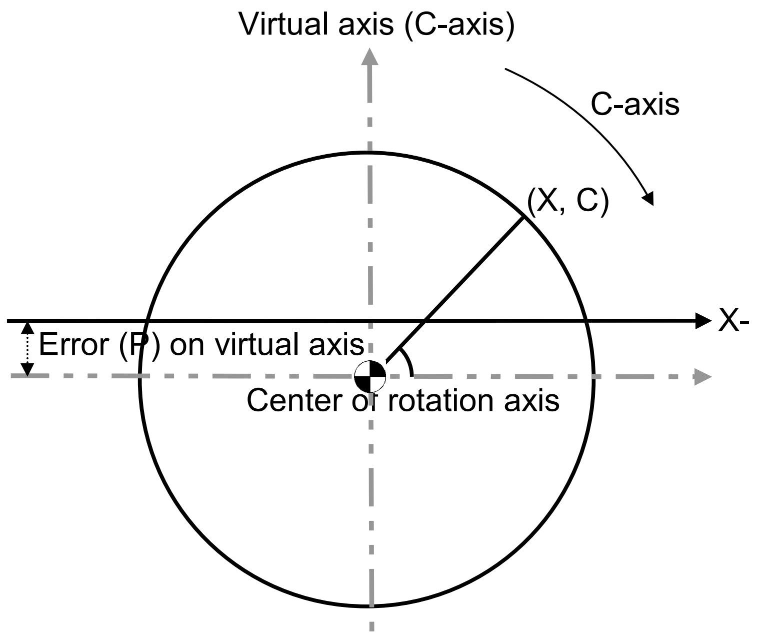
```
O001;
  :
N010 T0101
  :
N0100 G90 G00 X60.0 C0 Z_ ;  Positioning to start position
N0200 G12.1;                  Start of polar coordinate
N0201 G42 G01 X20.0F_ ;
N0202 C10.0;
N0203 G03 X10.0 C20.0 R10.0;
N0204 G01 X-20.0;
N0205 C-10.0;                  Geometry program
N0206 G03 X-10.0-20.0 I10.0 J0;  (program based on Cartesian
N0207 G01 X20.0;              coordinates on X-C' plane)
N0208 C0;
N0209 G40 X60.0;
N0210 G13.1:       Cancellation of polar coordinate interpolation
N0300 Z_ :
N0400 X_ C_ ;
  :
N0900M30;
```

4.8.1 Virtual Axis Direction Compensation for Polar Coordinate Interpolation

In polar coordinate interpolation, this function compensates a machine if it has an error on the virtual axis, that is, the center of the rotation axis is not on the X-axis.

Explanation

- Virtual axis direction error



(X, C) Point on the X-C plane (the origin of the X-C plane being assumed the center of the rotation axis)
X X coordinate in the X-C plane
C Virtual C-axis coordinate on the X-C plane
P Error on the virtual axis (set with parameter No. 6085.)

If, on a machine on which polar coordinate interpolation is performed on the X-axis (linear axis) and the C-axis (rotation axis) as shown in the figure above, there is an error on the virtual axis, this function compensates for the error before interpolation.

- Polar coordinate travel distance and calculation expression

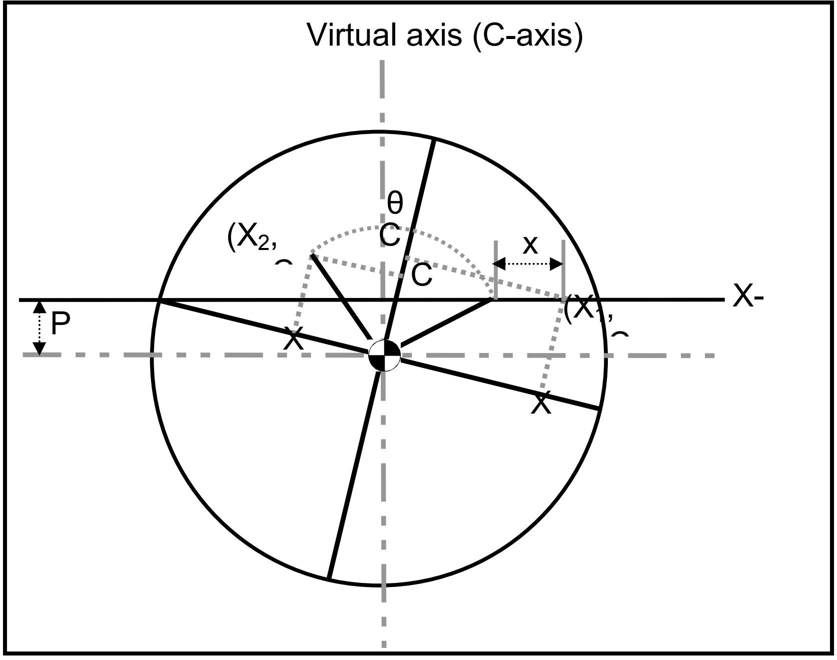
In the following figure, if the point (X_2, C_2) is specified when the tool is at the point (X_1, C_1) where the X-axis and the virtual C-axis interest with each other, the travel distance (X, θ) of the polar coordinates (actual X- and C-axes) is represented by:

$$x = \sqrt{X_2^{\,2} + C_2^{\,2} - P^2} - \sqrt{X_1^{\,2} + C_1^{\,2} - P^2}$$

$$\theta = \left(\tan^{-1}\frac{C_2}{X_2} - \sin^{-1}\frac{P}{\sqrt{X_2^{\,2} + C_2^{\,2}}} \right) - \left(\tan^{-1}\frac{C_1}{X_1} - \sin^{-1}\frac{P}{\sqrt{X_1^{\,2} + C_1^{\,2}}} \right)$$

X : Linear axis (X-axis) travel distance

θ : Rotation axis (C-axis) travel distance



Caution

⚠ CAUTION

1 This function is intended to apply compensation in the virtual axis direction during polar coordinate interpolation. It is unusable for any interpolation other than polar coordinate interpolation.

2 The rotation center is the origin of the "polar coordinate interpolation plane."

3 When C12.1 (polar coordinate mode start) command is issued, the coordinate of the virtual axis is "P" (parameter-specified virtual axis direction error) rather than "0."

4 The inside of a circle having the rotation axis center as its center and the error P as its radius is defined as a movement forbidden area. An attempt to move into that area will result in the OT514 alarm being issued.

4.9 CYLINDRICAL INTERPOLATION (G07.1)

The amount of travel of a rotary axis specified by an angle is once internally converted to a distance of a linear axis along the outer surface so that linear interpolation or circular interpolation can be performed with another axis. After interpolation, such a distance is converted back to the amount of travel of the rotary axis.
The cylindrical interpolation function allows the side of a cylinder to be developed for programming. So programs such as a program for cylindrical cam grooving can be created very easily.

Format

```
G07.1 IPr ;    Starts the cylindrical interpolation
 :             mode
 :             (enables cylindrical interpolation).
G07.1 IP0 ;    The cylindrical interpolation mode is
               cancelled.
  IP : An address for the rotation axis
  r :  The radius of the cylinder
       Specify G07.1 IPr ; and G07.1 IP0; in separate blocks.
       G107 can be used instead of G07.1.
```

Explanation

- Plane selection (G17,G18,G19)

Use parameter (No. 1022) to specify whether the rotation axis is the X-, Y-, or Z-axis, or an axis parallel to one of these axes. Specify the G code to select a plane for which the rotation axis is the specified linear axis.
For example, when the rotation axis is an axis parallel to the X-axis, G17 must specify an Xp-Yp plane, which is a plane defined by the rotation axis and the Y-axis or an axis parallel to the Y-axis.
Only one rotation axis can be set for cylindrical interpolation.

- Feedrate

A feedrate specified in the cylindrical interpolation mode is a speed on the developed cylindrical surface.

- Circular interpolation (G02,G03)

In the cylindrical interpolation mode, circular interpolation is possible with the rotation axis and another linear axis. Radius R is used in commands in the same way as described in II-4.4.

The unit for a radius is not degrees but millimeters (for metric input) or inches (for inch input).

Example) Circular interpolation between the Z axis and C axis

For the C axis of parameter (No.1022), 5 (axis parallel with the X axis) is to be set. In this case, the command for circular interpolation is

```
G18 Z_ C_ ;
G02(G03) Z_ C_ R_ ;
```

For the C axis of parameter (No.1022), 6 (axis parallel with the Y axis) may be specified instead. In this case, however, the command for circular interpolation is

```
G19 C_ Z_ ;
G02(G03) Z_ C_ R_ ;
```

- Cylindrical interpolation accuracy

In the cylindrical interpolation mode, the amount of travel of a rotary axis specified by an angle is once internally converted to a distance of a linear axis on the outer surface so that linear interpolation or circular interpolation can be performed with another axis. After interpolation, such a distance is converted back to an angle. For this conversion, the amount of travel is rounded to a least input increment.

So when the radius of a cylinder is small, the actual amount of travel can differ from a specified amount of travel. Note, however, that such an error is not accumulative.

If manual operation is performed in the cylindrical interpolation mode with manual absolute on, an error can occur for the reason described above.

$$\text{The actual Amount of travel} = \left[\frac{\text{MOTION REV}}{2\times 2\pi\times R} \times \left[\text{Specified value} \times \frac{2\times 2\pi\times R}{\text{MOTION REV}} \right] \right]$$

MOTION REV :The amount of travel per rotation of the rotation axis (Setting value of parameter No. 1260)

R :Workpiece radius

$\left[\ \ \right]$:Rounded to the least input increment

Limitation

- Arc radius specification in the cylindrical interpolation mode

In the cylindrical interpolation mode, an arc radius cannot be specified with word address I, J, or K.

- Cutter compensation

To perform cutter compensation, specify G41, G42, and G40 in cylindrical interpolation mode. Note that when cylindrical interpolation mode is set during cutter compensation, correct compensation cannot be applied.

- Positioning

In the cylindrical interpolation mode, positioning operations (including those that produce rapid traverse cycles such as G28, G53, G73, G74, G76, G80 through G89) cannot be specified. Before positioning can be specified, the cylindrical interpolation mode must be cancelled. Cylindrical interpolation (G07.1) cannot be performed in the positioning mode (G00).

- Coordinate system setting

In the cylindrical interpolation mode, a workpiece coordinate system (G92, G54 through G59) or local coordinate system (G52) cannot be specified.

- Cylindrical interpolation mode setting

In the cylindrical interpolation mode, the cylindrical interpolation mode cannot be reset. The cylindrical interpolation mode must be cancelled before the cylindrical interpolation mode can be reset.

- Rotation axis

Only one rotation axis can be set in cylindrical interpolation. With the G97.1 command, no more than one rotation axis can be specified.

- Parallel axis

The rotation axis for cylindrical interpolation is not a parallel axis.

- Tool offset

A tool offset must be specified before the cylindrical interpolation mode is set. No offset can be changed in the cylindrical interpolation mode.

- Index table indexing function

Cylindrical interpolation cannot be specified when the index table index function is being used.

- Rotation axis rollover function

If the rotation axis for which the rollover function is used is specified as the rotation axis used with cylindrical interpolation, the rollover function is disabled in cylindrical interpolation mode.

When cylindrical interpolation mode is cancelled, the rollover function is automatically enabled.

- Multiple-rotary axis control function

If the rotation axis for which the multiple-rotary-axis control function is used is specified as the rotation axis used with cylindrical interpolation, the multiple-rotary axis control function is disabled in cylindrical interpolation mode.

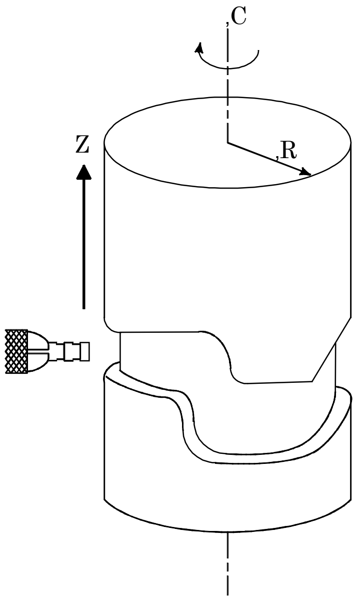
When cylindrical interpolation mode is cancelled, the multiple-rotary axis control function is automatically enabled.

Example

Example of a Cylindrical Interpolation Program

```
O0001 (CYLINDRICAL INTERPOLATION);
N01 G00 G90 Z100.0 C0 ;
N02 G01 G91 G18 Z0 C0 ;
N03 G07.1 C57299 ;
N04 G90 G01 G42 Z120.0 D01 F250 ;
N05 C30.0 ;
N06 G02 Z90.0 C60.0 R30.0 ;
N07 G01 Z70.0 ;
N08 G03 Z60.0 C70.0 R10.0 ;
N09 G01 C150.0 ;
N10 G03 Z70.0 C190.0 R75.0 ;
N11 G01 Z110.0 C230.0 ;
N12 G02 Z120.0 C270.0 R75.0 ;
N13 G01 C360.0 ;
N14 G40 Z100.0 ;
N15 G07.1 C0 ;
N16 M30 ;
```

Note) In the sample program given here, the C-axis in parameter No. 1022 is set to 5 (an axis parallel to the X-axis).



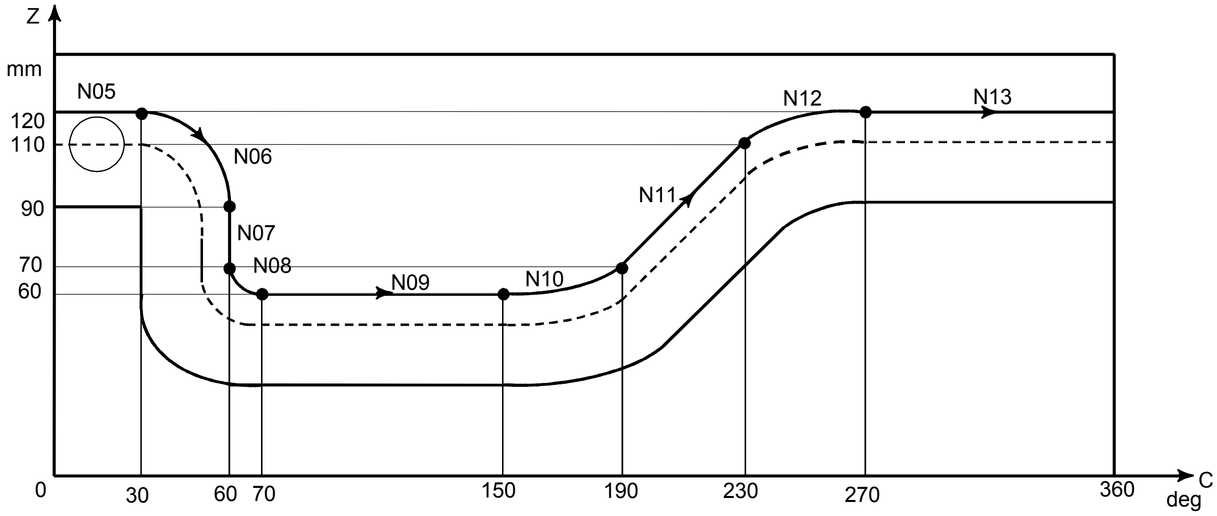


Fig.4.9 (a) Cylindrical Interpolation

4.10 CYLINDRICAL INTERPOLATION CUTTING POINT CONTROL (G07.1)

The conventional cylindrical interpolation function controls the tool center so that the tool axis always moves along a specified path on the cylindrical surface, towards the rotation axis (cylindrical axis) of the workpiece. On the other hand, this function controls the tool so that the tangents to the tool and a contour figure cutting surface always pass through the rotation center of a workpiece

Format

As shown below, the same command as that for the conventional cylindrical interpolation function is used.

```
  G07.1 IPr ;  Sets cylindrical interpolation mode
               (enables cylindrical interpolation).
      :
      :
  G07.1 IP0 ;  Clears cylindrical interpolation mode.
     IP  : One rotation axis address
     r   : Cylinder radius of rotation axis
     Specify each of G07.1 IPr; and G07.1 IP0; singly in a block.
```

Explanation

- Comparison with conventional cylindrical interpolation

As shown in Fig.4.10(a) , control is exercised along the offset axis (Y-axis) direction that is perpendicular to the tool, tool center axis, and workpiece rotation center axis.

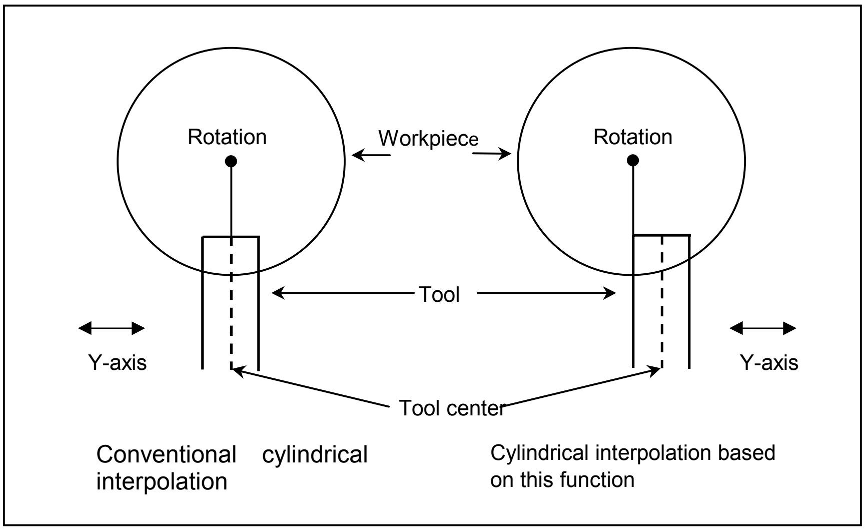

Rotation
Workpiece
Rotation
Tool
Y-axis
Y-axis
Tool center
Conventional cylindrical interpolation
Cylindrical interpolation based on this function

Fig.4.10(a) Comparison with Conventional Interpolation

- Cutting point compensation

(1) Cutting point compensation between blocks

As shown in Fig.4.10(b), cutting point compensation is achieved by moving between blocks N1 and N2.

1) Let C1 and C2 be the heads of the vectors normal to N1 and N2 from S1, which is the intersection of the tool center paths of blocks N1 and N2

2) After the tool moves to S1 according to the command of N1, the tool moves through V on the C-axis as a result of cutting point compensation, then through $-V\times\frac{\pi}{180}\times r$ along the Y-axis.

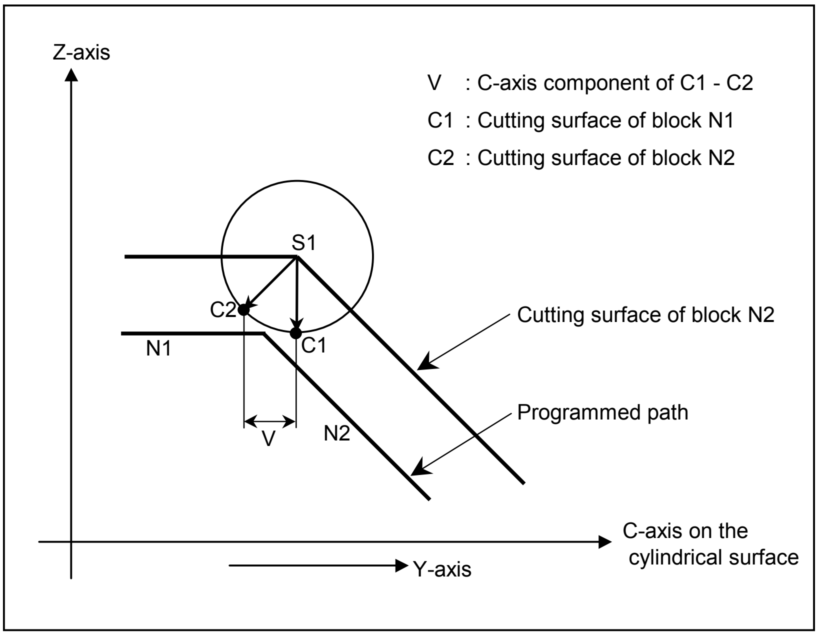

Z-axis
V : C-axis component of C1 - C2
C1 : Cutting surface of block N1
C2 : Cutting surface of block N2
S1
C2
C1
N1
N2
V
Cutting surface of block N2
Programmed path
C-axis on the cylindrical surface
Y-axis

Fig.4.10(b) Cutting Point Compensation between Blocks

(2) Cutting point compensation in a circular command block

As shown in Fig.4.10(c), the movement required for cutting point compensation is made simultaneously with circular interpolation in block N1.

1) Let C0 be the head of the vector normal to N1 from S0, which is the tool center position at the start point of circular block N1. Let C1 be the head of the similar vector at the end point.

2) As the tool moves from S0 to S1, a superimposed movement is made by the C-axis component of (C1 - C2) (V in the figure) on the C-axis, and a superimposed movement is made by $-V\times\frac{\pi}{180}\times r$ along the Y-axis. along the Y-axis.

That is, the following expressions are valid. As movement is made through L as shown in Fig.4.10(c), the superimposed movements are made on the C-axis and Y-axis as follows: $\Delta C=\Delta V$

$$\Delta Y = -\frac{\pi}{180}(\Delta V)r$$

ΔV :Cutting point compensation value (ΔV2 - ΔV1) for movement of ΔL

ΔV1 :C-axis component of the vector normal to N1 from the tool
center of the start point of ΔL

ΔV2 :C-axis component of the vector normal to N1 from the tool
center of the end point of ΔL

R :Arc radius

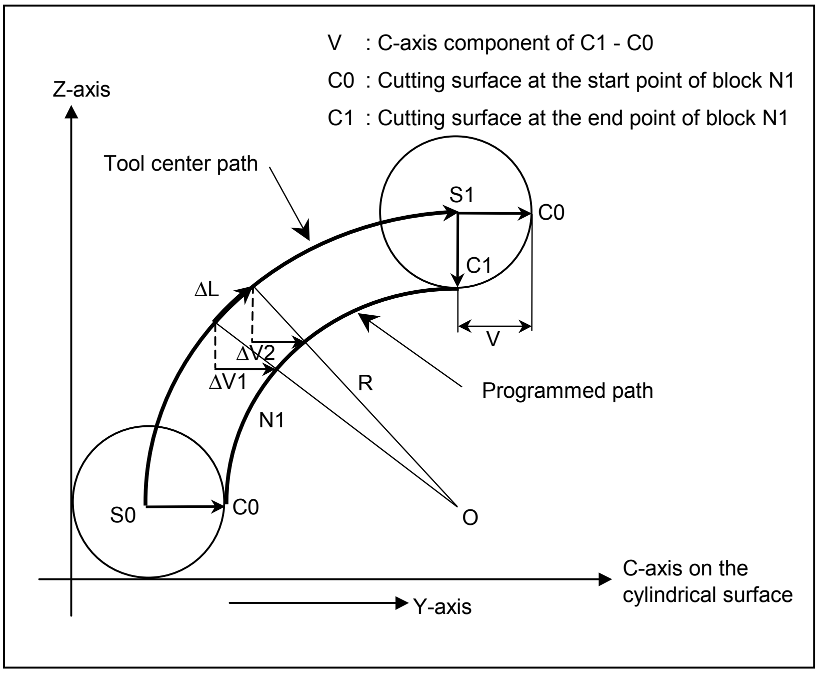

V : C-axis component of C1 - C0
C0 : Cutting surface at the start point of block N1
C1 : Cutting surface at the end point of block N1
Z-axis
Tool center path
S1
C0
C1
ΔL
V
ΔV2
ΔV1
R
Programmed path
N1
S0
C0
O
C-axis on the cylindrical surface
Y-axis

Fig.4.10(c) Cutting Point Compensation in a Circular Command Block

(3) When cutting point compensation is not applied between blocks

When, as shown in Fig.4.10(d) and Fig.4.10(e), the cutting point compensation value (V in the figures) is less than the value set in parameter No. 6112, one of the operations below is performed. (The operation that is performed depends on the setting of bit 6 (CYS) of parameter No. 6004.

1) When bit 6 (CYS) of parameter No. 6004 is set to 1
Cutting point compensation is not applied between blocks N1 and N2, but is applied when block N2 is executed.

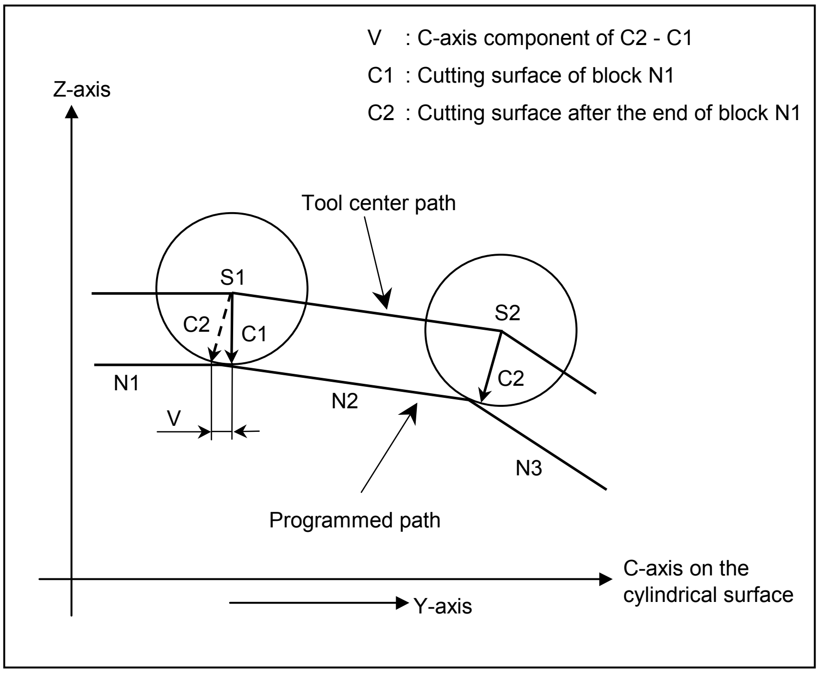

V : C-axis component of C2 - C1
C1 : Cutting surface of block N1
C2 : Cutting surface after the end of block N1
Z-axis
Tool center path
S1
C2
C1
S2
C2
N1
N2
V
N3
Programmed path
C-axis on the cylindrical surface
Y-axis

Fig.4.10(d) When Bit 6 (CYS) of Parameter No. 6004 Is Set to 1

2) When bit 6 (CYS) of parameter No. 6004 is set to 0 Cutting point compensation is not performed between blocks N1 and N2. Whether to apply cutting point compensation between block N2 and N3 is determined by taking the cutting point compensation value between blocks N2 and N3 (V1 in the figure) into consideration.

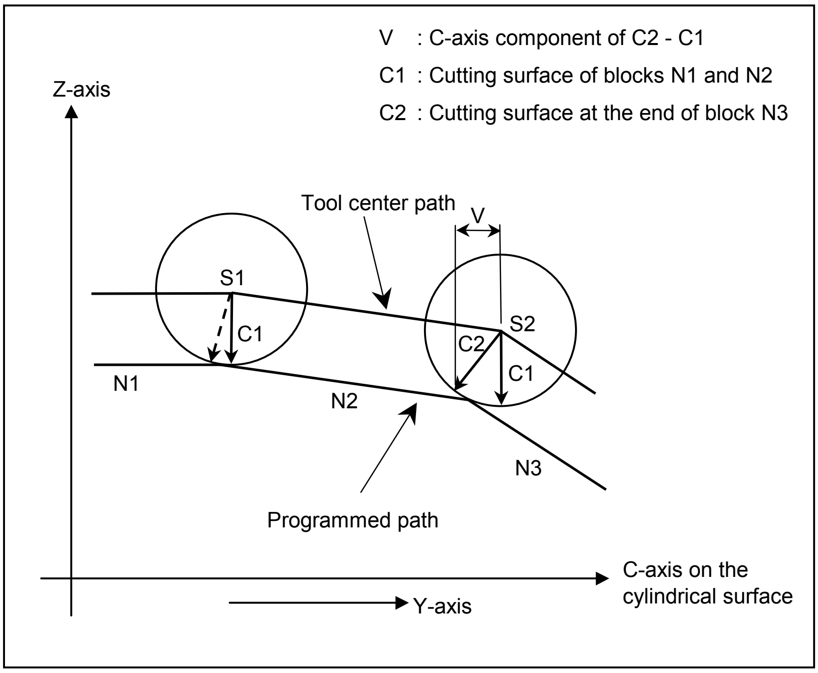

V : C-axis component of C2 - C1
C1 : Cutting surface of blocks N1 and N2
C2 : Cutting surface at the end of block N3
Z-axis
Tool center path
V
S1
C1
S2
C2
C1
N1
N2
N3
Programmed path
C-axis on the cylindrical surface
Y-axis

Fig.4.10(e) When Bit 6 (CYS) of Parameter No. 6004 Is 0

3) When the amount of travel (L1) of block N2 is less than the value set in parameter No. 6113, as shown in Fig.4.10(f), cutting point compensation is not applied between blocks N1 and N2. Instead, block N2 is executed with the cutting point compensation of the previous block. When the amount of travel (L2) of block N3 is greater than the value set in parameter No. 6113, cutting point compensation is applied between blocks N2 and N3.

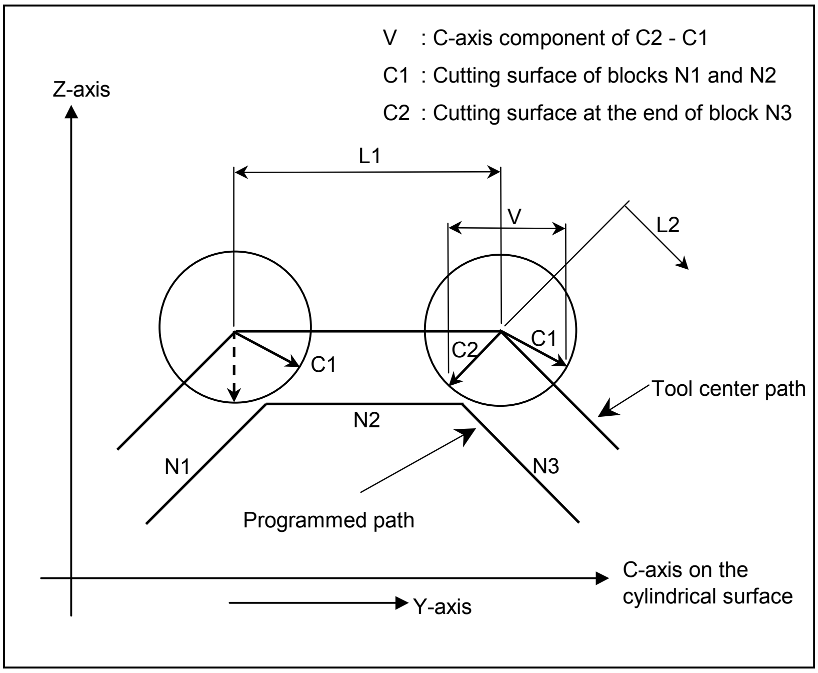

V : C-axis component of C2 - C1
C1 : Cutting surface of blocks N1 and N2
C2 : Cutting surface at the end of block N3
Z-axis
L1
V
L2
C1
C2
C1
Tool center path
N2
N1
N3
Programmed path
C-axis on the cylindrical surface
Y-axis

Fig.4.10(f) When the Amount of Travel (L1) of Block N2 Is Less Than the Parameter Value

4) When, as shown in Fig.4.10(g), the diameter of an arc (R in the figure) is less than the value set in parameter No. 6113, cutting point compensation is not applied simultaneously with circular interpolation

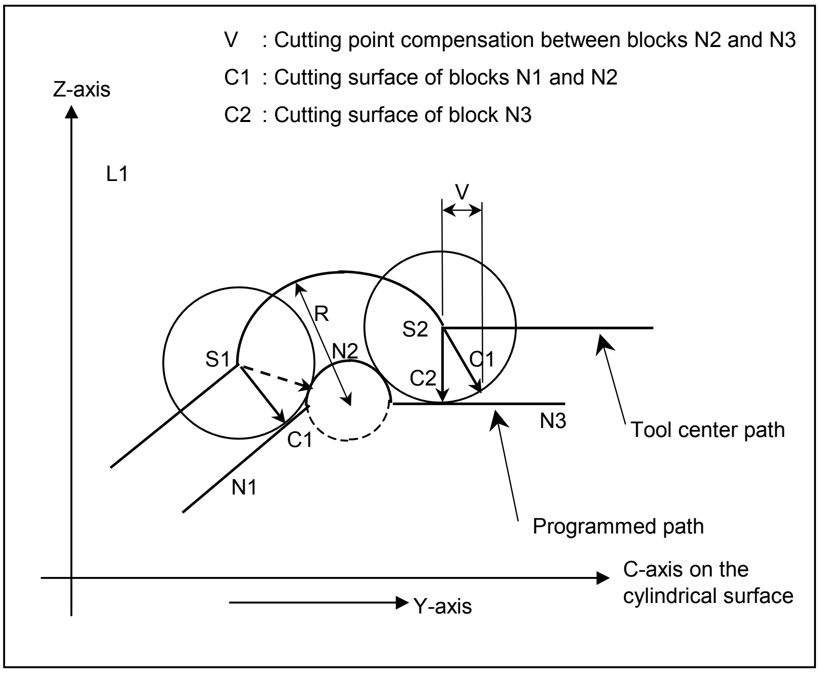

V : Cutting point compensation between blocks N2 and N3
C1 : Cutting surface of blocks N1 and N2
C2 : Cutting surface of block N3
Z-axis
L1
V
R
S1
N2
S2
C2
C1
C1
N1
N3
Tool center path
Programmed path
C-axis on the cylindrical surface
Y-axis

Fig.4.10(g) When the Diameter of an Arc Is Less Than the Parameter Value

- Applying cutting point compensation together with normal direction control

When applying cutting point compensation together with normal direction control, cutting point compensation between specified blocks is applied in combination with movement on the normal direction control axis (C-axis) as shown below, regardless of the cutting point compensation method described above.

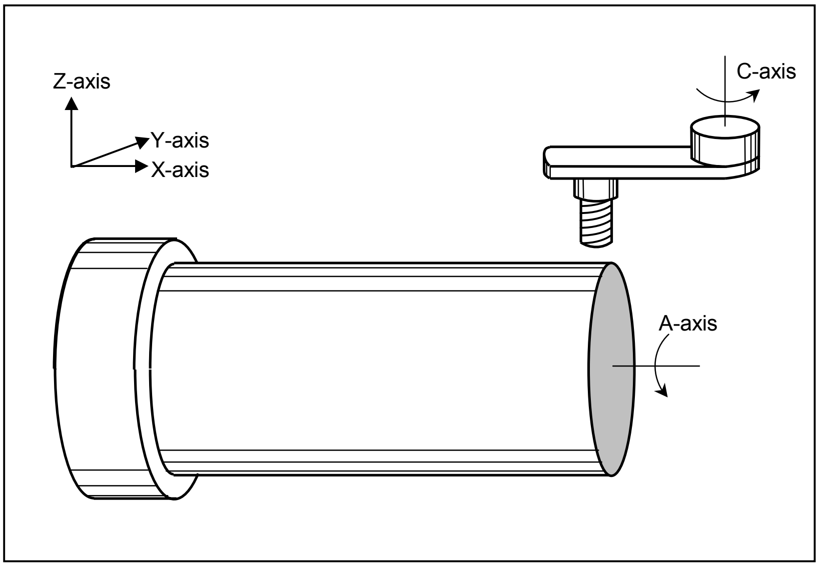


Fig.4.10(h) Using Cutting Point Compensation with Normal Direction Control

(1) When the normal direction changes between blocks N1 and N2, cutting point compensation is applied between blocks N1 and N2. As shown in Fig.4.10(i), cutting point compensation is applied according to (1) of cutting point compensation, described above, together with the movement resulting from normal direction control between blocks N1 and N2.

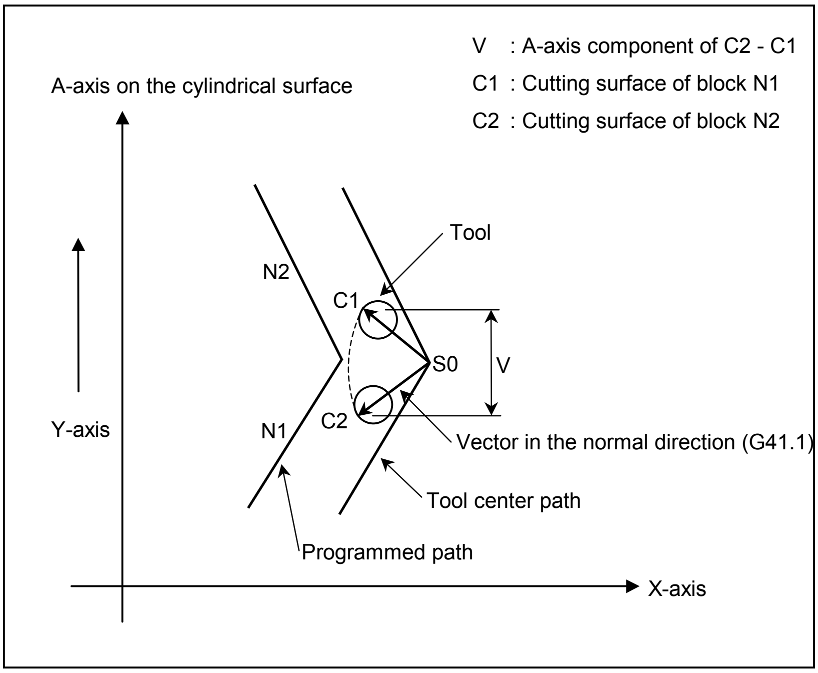

V : A-axis component of C2 - C1
C1 : Cutting surface of block N1
C2 : Cutting surface of block N2
A-axis on the cylindrical surface
Tool
N2
C1
S0
V
C2
N1
Y-axis
Vector in the normal direction (G41.1)
Tool center path
Programmed path
X-axis

Fig.4.10(i) When the Normal Direction Changes between Blocks N1 and N2

(2) When the normal direction is changed by gradual curve normal direction control together with the movement of a specified block, cutting point compensation is performed together with the movement of the specified block. When a rotation of θ1 is made on the normal direction control axis together with the movements of blocks N1 and N2 as shown in Fig.4.10(j), cutting point compensation based on vector V1 movement is also performed together with the movement of block N2.

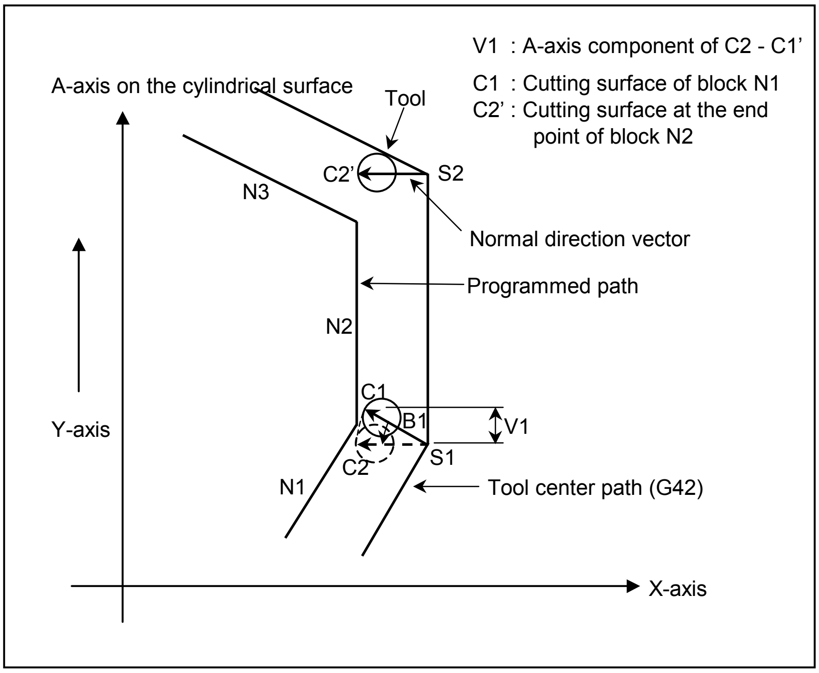

V1 : A-axis component of C2 - C1'
A-axis on the cylindrical surface
Tool
C1 : Cutting surface of block N1
C2' : Cutting surface at the end point of block N2
C2'
S2
N3
Normal direction vector
Programmed path
N2
C1
B1
V1
Y-axis
C2
S1
N1
Tool center path (G42)
X-axis

Fig.4.10(j) Gradual Curve Normal Direction Control

(3) If normal direction control is applied in a specified block with the normal direction at the end point of the previous block left as is, cutting point compensation is not applied; the cutting point compensation of the previous block is used as is. In Fig.4.9(k), the movement of block N2 (L1 in the figure) is smaller than the value of parameter No. 7794, so that no rotation is made at S1 on the normal direction control axis, while rotation is made at S2 on the normal direction axis because the movement of block N3 (L2 in the figure) is larger than the value of parameter No. 7794. In this case, cutting point compensation is not applied at S1, but movement by vector V2 is made at S2.

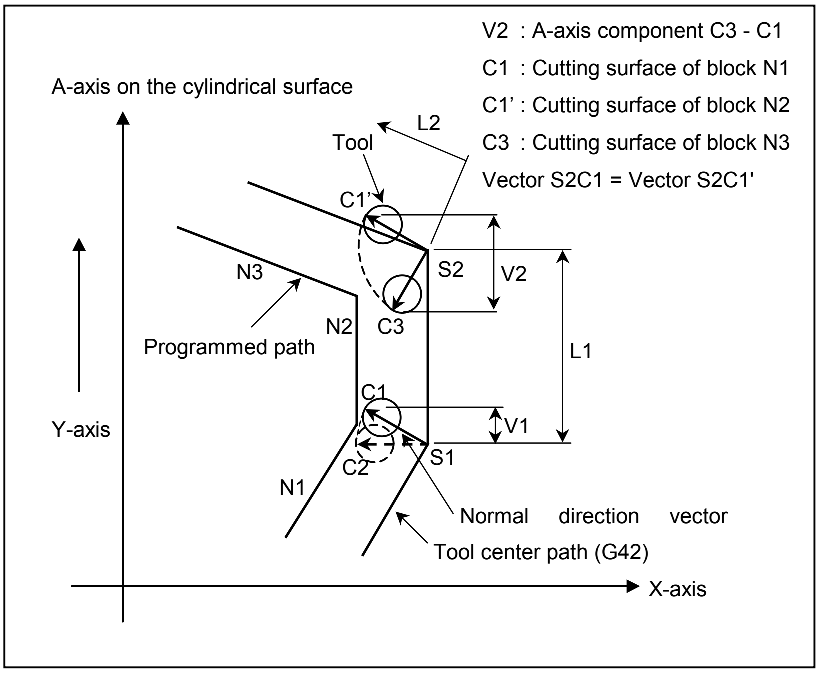

V2 : A-axis component C3 - C1
C1 : Cutting surface of block N1
C1' : Cutting surface of block N2
C3 : Cutting surface of block N3
Vector S2C1 = Vector S2C1'
A-axis on the cylindrical surface
L2
Tool
C1'
N3
S2
V2
N2
C3
Programmed path
L1
C1
V1
Y-axis
S1
C2
N1
Normal direction vector
Tool center path (G42)
X-axis

Fig.4.10 (k)

- Feedrate during cutting point compensation

(1) The tool moves at a specified feedrate while cutting point compensation is being applied between blocks.

(2) The actual speed indication and feedrate during circular interpolation are as described below.

Actual speed indication

The speed component of each axis after cutting point compensation at a point in time during circular interpolation is as follows:

$Fz' = Fz$ .... Speed component of linear axis

$Fc' = Fc + (Vce - Vcs)$ ...... Speed component of rotation axis

$Fy' = -(Vce - Vcs)\dfrac{\pi r}{180}$ ...... Speed component of offset axis

Fz : Speed component of a cylindrical interpolation linear axis before cutting point compensation

Fc : Speed component of cylindrical interpolation rotation axis before cutting point compensation

Vcs: Rotation axis component of a tool contact point vector (Vs in the figure) at the start point at a point in time

Vce: Rotation axis component of tool contact point vector (Ve in the figure) at the end point at a point in time

r : Radius of the cylinder of a rotation axis

Accordingly, the actual speed indication during circular interpolation is greater than the specified value when |Fc'| > |Fc|(inner offset of the arc). Conversely, the actual speed indication during circular interpolation is less than the specified value when |Fc'| < |Fc|(outer offset of the arc).

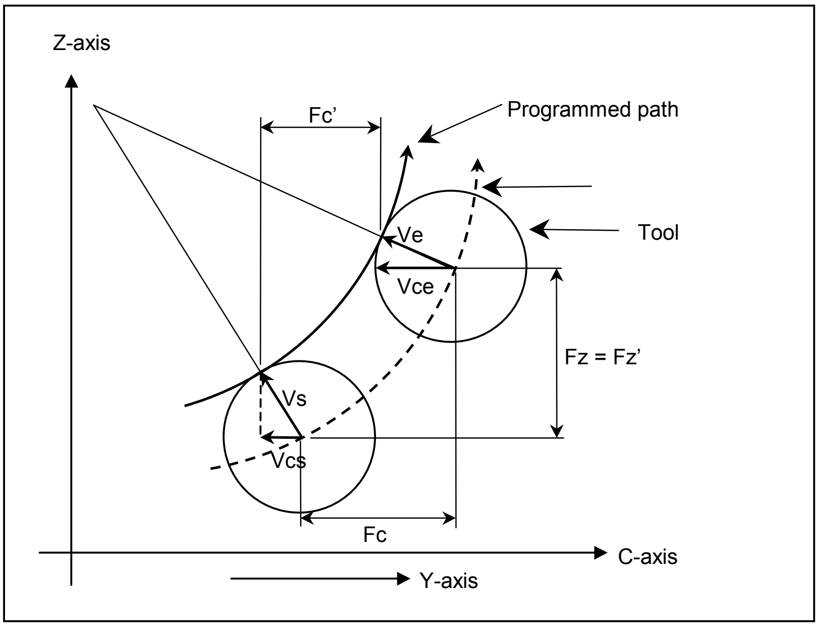


Fig.4.10 (I) Actual Speed Indication during Circular Interpolation

- Usable G codes

(1) In any of the following G code modes, cylindrical interpolation cutting point compensation can be specified:

G17,G18,G19	:	Plane selection
G22	:	Stored stroke check function on
G64	:	Cutting mode
G90,G91	:	Absolute command programming, incremental command programming
G94	:	Feed per minute

(2) Any of the following G codes can be specified in cylindrical interpolation cutting point compensation mode:

G01,G02 ,G03	:	Linear interpolation, circular interpolation
G04	:	Dwell
G40,G41,G42	:	Cutter compensation
G40.1-G42.1	:	Normal direction control
G64	:	Cutting mode
G65-G67	:	Macro call
G90,G91	:	Absolute command programming, incremental command programming

Limitation

- Overcutting during inner corner cutting

Theoretically, when the inner area of a corner is cut using linear interpolation as shown in Fig. 4.10(m), this function slightly overcuts the inner walls of the corner. This overcutting can be avoided by specifying a value of R that is slightly greater than the radius of the tool at the corner.

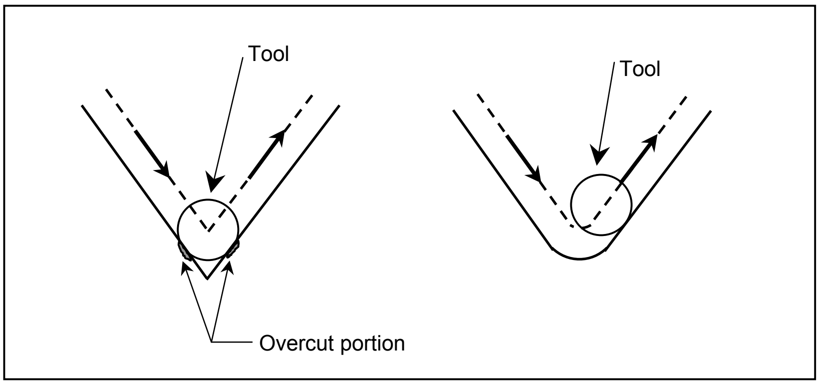


Fig.4.10 (m) Overcutting

- Setting the minimum input increment for an offset axis (Y-axis)

Set the same minimum input increment for an offset axis and linear axis when cylindrical interpolation is performed.

- Workpiece radius specification

When specifying the radius of a workpiece, use the minimum input increment (with no decimal point) for the linear axis used in cylindrical interpolation.

- Reference axis setting (parameter No. 1031)

If the increment system of a linear axis differs from that of a rotation axis in cylindrical interpolation, set, as the reference axis, the axis number of the linear axis for cylindrical interpolation.

Example

- Example of cylindrical interpolation cutting point compensation

The sample program below indicates the positional relationships between a workpiece and tool.

```
O0001(CYLINDRICAL INTERPOLATION1) ;
N01 G00 G90 Z100.0 C0 ;
N02 G01 G91 G19 Z0 C0 ;
N03 G07.1 C57299 ;
N04 G01 G42 G90 Z120.0 D01 F250. ; - - - (1)
N05 C20.0 ;                          - - - (2)
N06 G02 Z80.0 C60.0 R40.0 ;          - - - (3)
N07 G01 Z70.0 ;                      - - - (4)
N08 G03 Z60.0 C70.0 R10.0 ;          - - - (5)
  :
M30 ;
```

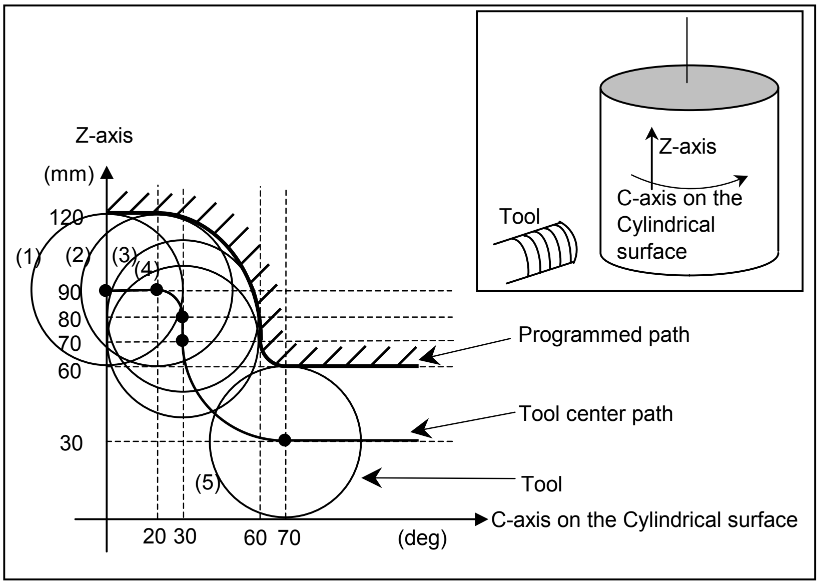


Fig.4.10 (n) Path of Sample Program for Cylindrical Interpolation Cutting Point Compensation

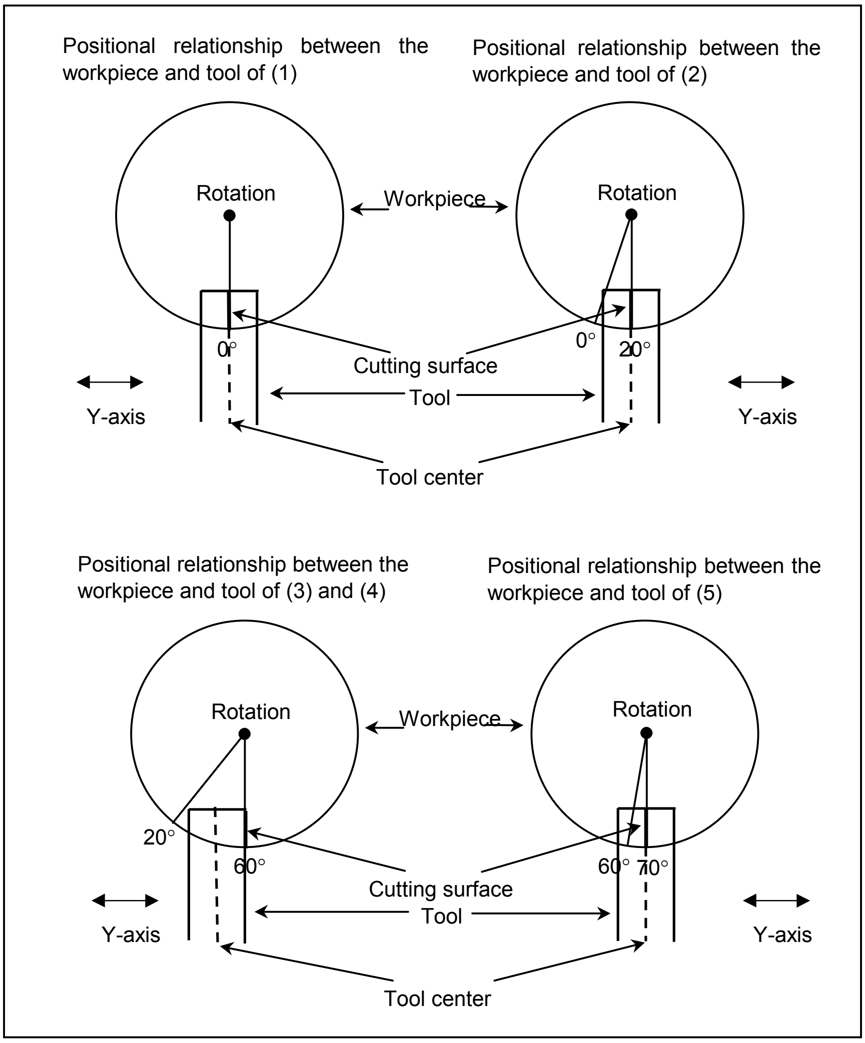

Positional relationship between the workpiece and tool of (1)
Positional relationship between the workpiece and tool of (2)
Rotation
Workpiece
Rotation
0°
0°
20°
Cutting surface
Tool
Y-axis
Y-axis
Tool center
Positional relationship between the workpiece and tool of (3) and (4)
Positional relationship between the workpiece and tool of (5)
Rotation
Workpiece
Rotation
20°
60°
60°
70°
Cutting surface
Tool
Y-axis
Y-axis
Tool center

Fig.4.10 (o) Positional Relationships between Workpiece and Tool of Sample Program

- Example of specifying cylindrical interpolation cutting point compensation and normal direction control at the same time

Cutter compensation value No. 01 = 30 mm

```
O0002(CYLINDRICAL INTERPOLATION2) ;
N01 G00 G90 X100.0 A0 ;
N02 G01 G91 G17 X0 A0 ;
N03 G07.1 C57299 ;
N04 G01 G41 G42.1 G90 X120.0 D01 F250. ;
N05 A20.0 ;
N06 G03 X80.0 A60.0 R40.0 ;
N07 G01 X70.0 ;
N08 G02 X70.0 A70.0 R10.0 ;
N09 G01 A150.0 ;
N10 G02 X70.0 A190.0 R85.0 ;
N11 G01 X110.0 A265.0 ;
N12 G03 X120.0 A305.0 R85.0 ;
N13 G01 A360.0 ;
N14 G40 G40.1 X100.0 ;
N15 G07.1 A0 ;
N16 M30 ;
```

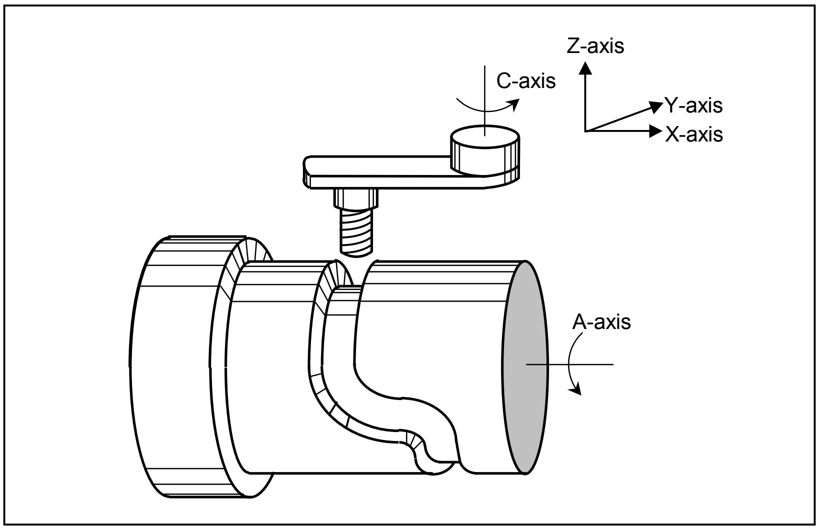


Fig.4.10 (p) Sample Program Specifying Cylindrical Interpolation Cutting Point Compensation and Normal Direction Control at the Same Time

4.11 EXPONENTIAL INTERPOLATION (G02.3,G03.3)

Exponential interpolation exponentially changes the rotation of a workpiece with respect to movement on the rotary axis. Furthermore, exponential interpolation performs linear interpolation with respect to another axis. This enables tapered groove machining with a constant helix angle (constant helix taper machining). This function is best suited for grooving and grinding tools such as end mills.

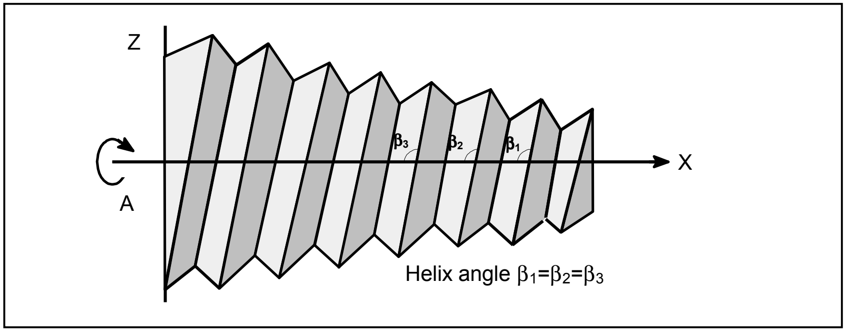
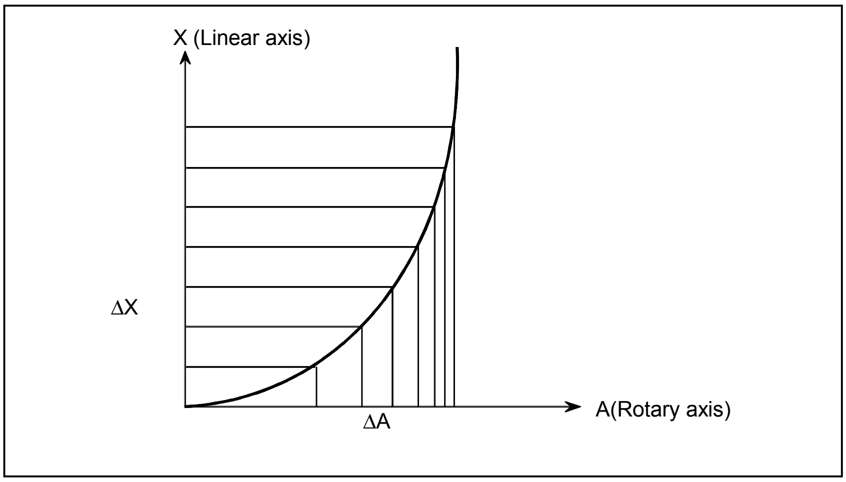


Fig.4.11 (a) Exponential interpolation



Format

positive rotation (ω=0)
G02.3 X_ Y_ Z_ I_ J_ K_ R_ F_ Q_ ;
Negative rotation (ω=1)
G03.3 X_ Y_ Z_ I_ J_ K_ R_ F_ Q_ ;

X_ : Specifies an end point with an absolute or incremental value.

Y_ : Specifies an end point with an absolute or incremental value.

Z_ : Specifies an end point with an absolute or incremental value.

I_ : Specifies angl I. The specification units conform to the setting made for the reference axis (parameter No. 1031).

J_ : Specifies angle J. The specification units conform to the setting made for the reference axis.

K_ : Specifies the amount to divide the linear axis for exponentia interpolation (span value). Specify a positive value.
When no value is specified, the setting made in bit 7 (CBK) of parameter No. 7610 is assumed.
If CBK is 0, the value is set in parameter No. 7685.
If CBK is 1, the value specified in K is used.

R_: Specifies constant R for exponential interpolation.

F_: Specifies the initial feedrate.
Specified in the same way as an ordinary F code.
Specify a composite feedrate including a feedrate on the rotary axis.

Q_: Specifies the feedrate at the end point.
The same unit used for F is used. The CNC internally performs interpolation between the initial feedrate (F) and final feedrate (Q), depending on the travel distance on the linear axis.

Explanation

- Exponential relational expressions

Exponential relational expressions for a linear axis and rotary axis are defined as follows:

$$X(\theta) = R \times (e^{\frac{\theta}{k}} - 1) \times \frac{1}{\tan(l)} \quad \text{Movement on the linear axis (1)}$$

$$A(\theta) = (-1)^{\omega} \times 360 \times \frac{\theta}{2\pi} \quad \text{Movement on the linear axis (2)}$$

where

$$K = \frac{\tan(J)}{\tan(l)}$$

$$\omega = 0/1 \quad \text{Rotation direction}$$

R, I, and J are constants, and θ represents an angle (radian)
The following is obtained from Expression (1)

$$\theta(X) = K \times \ln(\frac{X \times \tan(I)}{R} + 1)$$

When there is movement from X_1 to X_2 on the linear axis, the amount of movement on the rotary axis is determined by :

$$\Delta\theta = K \times \left\{ \ln(\frac{X_2 \times \tan(I)}{R} + 1) - \ln(\frac{X_1 \times \tan(I)}{R} + 1) \right\}$$

Specify Expressions (1) and (2) in the format described earlier.

- Rotation angle θ

In exponential interpolation, the X coordinate and angular displacement θ of the A axis to X are expressed by equation (1).

$$\theta(X) = K \times \ln(\frac{x \times \tan(I)}{R} + 1) \text{ ---------- (1)}$$

where, I is the gradient.

In equation (1), the portion of the natural logarithm l n in parentheses must satisfy equation (2). Because it is impossible to find the logarithm of a negative number, which means that the value in the parentheses must be positive.

$$\frac{x \times \tan(I)}{R} > -1 \text{ ------------------ (2)}$$

As illustrated below, if the value of xtan(I)/R becomes smaller than -1, the position moves beyond point (A), which is not realistic. In this case, alarm PS897 occurs.

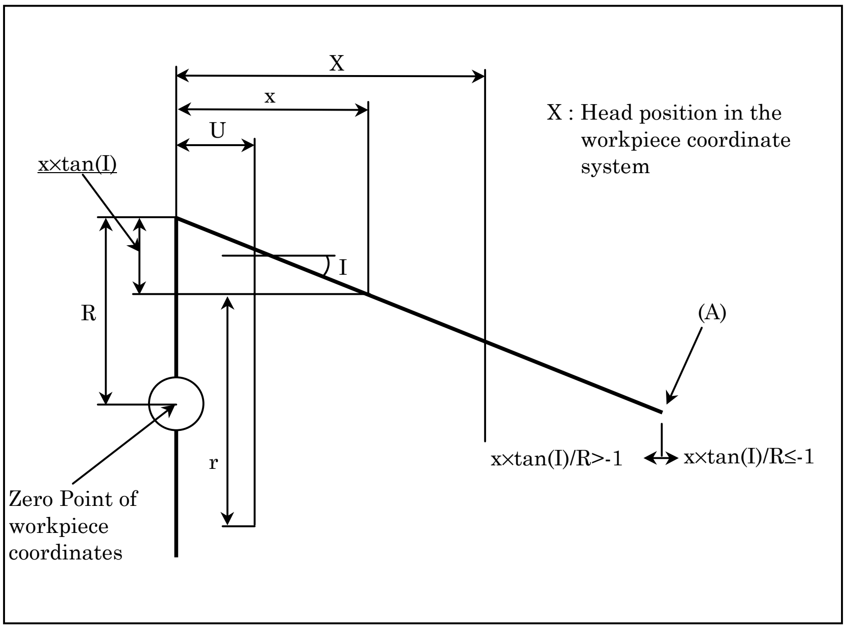

X
x
U
X : Head position in the workpiece coordinate system
x×tan(I)
I
R
(A)
r
Zero Point of workpiece coordinates
x×tan(I)/R>-1
x×tan(I)/R≤-1

Fig.4.11 (b) Rotation angle θ

The values of X, Y, Z, and U in the equations for exponential interpolation (refer to the operator's manual) are treated using coordinates in the workpiece coordinate system. Therefore, even when a positive value is specified in an incremental command, it is regarded as negative in the equations if it is positioned at a negative coordinate in the workpiece coordinate system.

- Gradient I

The relationship between the machining profile and the sign of the gradient I is as follows:
- For a slope going upward from left to right, I is a positive value.
- For a slope going downward from left to right, I is a negative value.

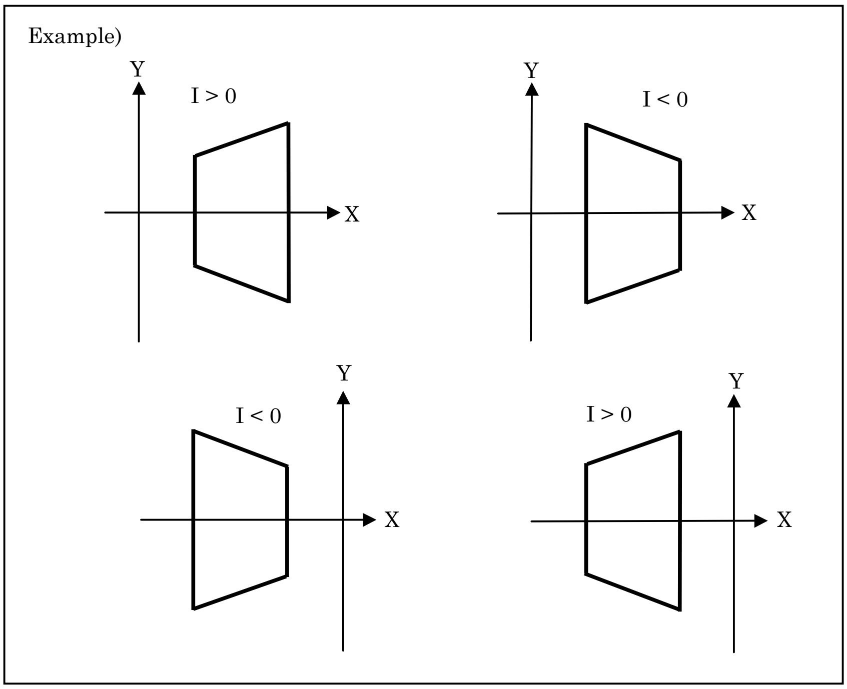


Fig.4.11 (c) Gradient I

- Helix angle J

The sign of helix angle J is determined in the same manner as I as shown below.

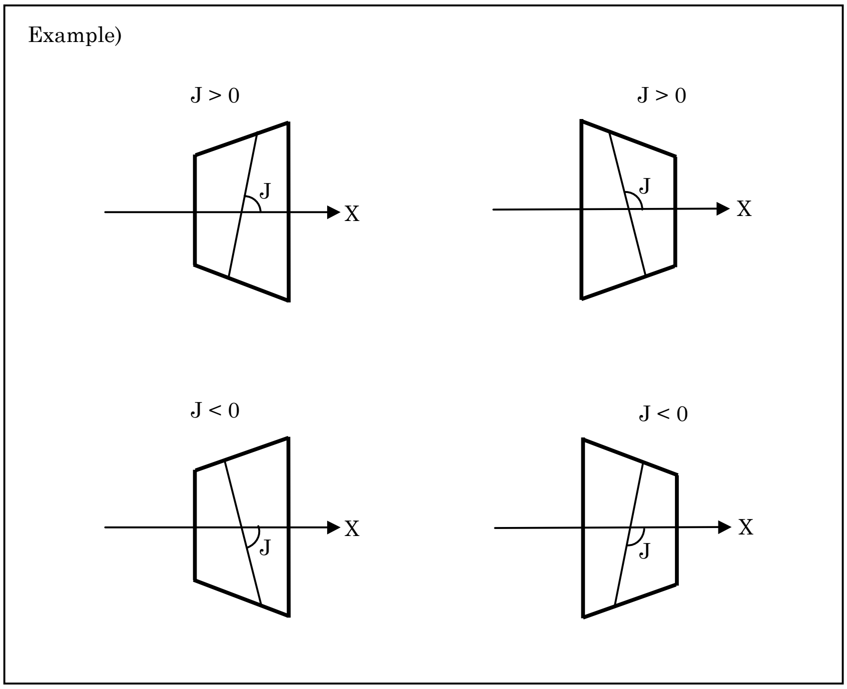


Fig.4.11 (d) Helix angle J

Limitation

- Cases where linear interpolation is performed

Even when the G02.3 or G03.3 mode is set, linear interpolation is performed in the following cases:

- When the linear axis specified in parameter(No.7636) is not specified, or the amount of movement on the linear axis is 0
- When the rotary axis specified in parameter (No.7637) is specified
- When the amount for dividing the linear axis (span value) is 0

- Tool length compensation / cutter compensation

Neither tool length compensation nor cutter compensation can be used in the G02.3 and G03.3 modes.

> ⚠ **CAUTION**
> The amount for dividing the linear axis for exponential interpolation (span value) affects figure precision. However, if an excessively small value is set, the machine may stop during interpolation. Try to specify an optimal span value depending on the machine being used.

Example

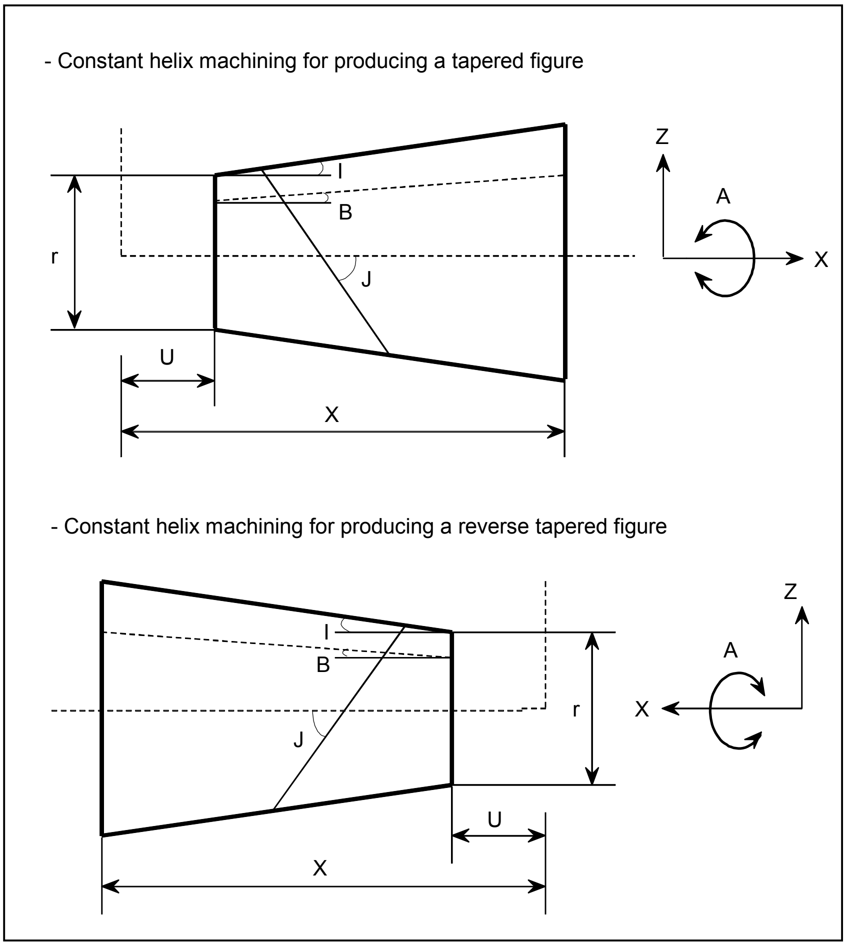

- Constant helix machining for producing a tapered figure
I
B
r
J
U
X
Z
A
X
- Constant helix machining for producing a reverse tapered figure
I
B
J
r
U
X
Z
A
X

Fig.4.11 (e) Constant Helix Machining for Producing a Tapered Figure

Relational expressions

$$Z(\theta) = \left\{ \frac{r}{2} - U \times \tan(I) \right\} \times (e^{\frac{\theta}{k}} - 1) \times \frac{\tan(B)}{\tan(l)} + Z(0) \qquad (3)$$

$$X(\theta) = \left\{ \frac{r}{2} - U \times \tan(I) \right\} \times (e^{\frac{\theta}{k}} - 1) \times \frac{1}{\tan(l)} + U \qquad (4)$$

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

where

$$K = \frac{\tan(J)}{\tan(I)}$$

X (θ), Z (θ), A (θ) : Absolute value on the X-axis, Z-axis, and A-axis from the origin

r : Left end diameter

U : Excess length

I : Taper angle

B : Groove bottom taper angle

J : Helix angle

X : Amount of movement on the linear axis

ω : Helix direction (0: Positive, 1: Negative)
θ : Workpiece rotation angle

From expressions (3) and (4), the following is obtained ;

$$Z(\theta) = \tan(B) \times (X(\theta) - U) + Z(0) \qquad (5)$$

The groove bottom taper angle (B) is determined from the end point position on the X-axis and Z-axis according to Expression (5). The amount of movement on the Z-axis is determined from a groove bottom taper angle (B) and X-axis position.
From Expressions (1) and (4), the following is determined:

$$R = r/2 - U \times \tan(I) \qquad (6)$$

Constant R is determined from the left end diameter (r) and excess length (U) according to Expression (6). Specify a taper angle (I) in address I, and specify a helix angle (J) in address J. Note, however, that a negative value must be specified as the taper angle (I) for constant helix machining in order to produce a reverse tapered figure. Select a helix direction with G02.3 or G03.3.
The user can perform constant helix machining to produce a tapered figure or a reverse tapered figure.

4.12 INVOLUTE INTERPOLATION (G02.2,G03.2)

Involute curve machining can be performed by using involute interpolation. Involute interpolation ensures continuous pulse distribution even in high-speed operation in small blocks, thus enabling smooth and high-speed machining. Furthermore, machining tapes can be created easily and efficiently, reducing the required length of tape.

Format

Involute interpolation on the Xp-Yp plane

G17 G02.2 Xp_ Yp_ I_ J_ R_ F_ ;
G17 G03.2 Xp_ Yp_ I_ J_ R_ F_ ;

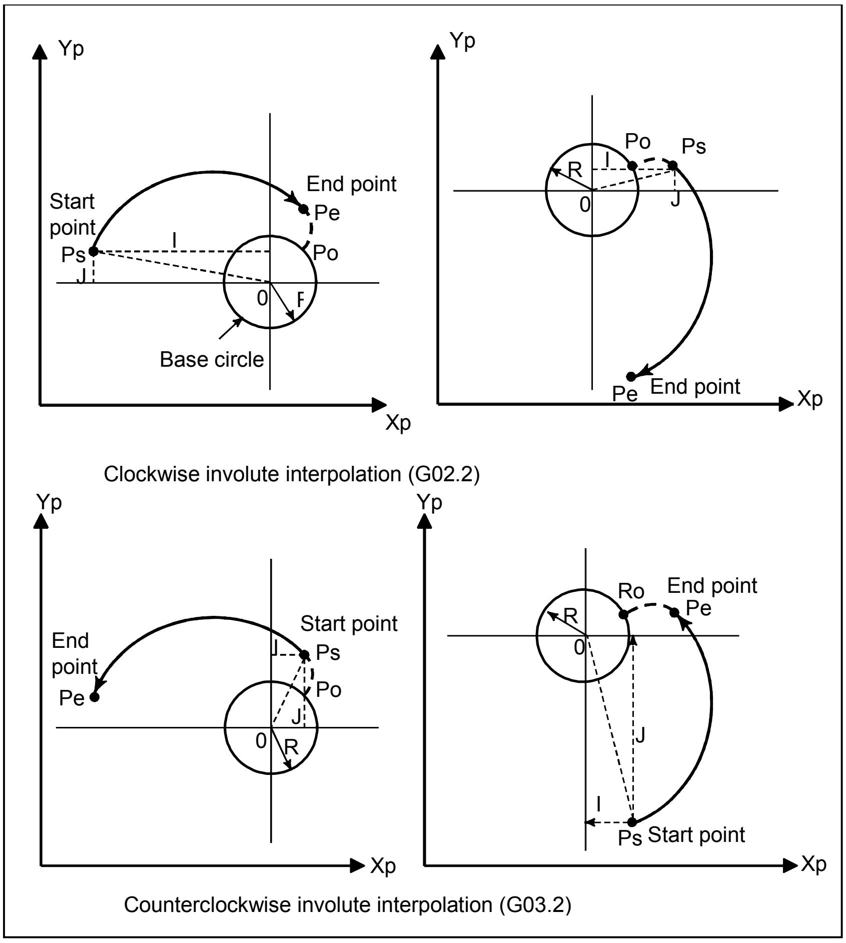
Involute interpolation on the Zp-Xp plane

G18 G02.2 Zp_ Xp_ K_ I_ R_ F_ ;
G18 G03.2 Zp_ Xp_ K_ I_ R_ F_ ;

Involute interpolation on the Yp-Zp plane

G19 G02.2 Yp_ Zp_ J_ K_ R_ F_ ;
G19 G03.2 Yp_ Zp_ J_ K_ R_ F_ ;

Where,
G02.2 : Involute interpolation (clockwise)
G03.2 : Involute interpolation (counterclockwise)
G17/G18/G19: Xp-Yp / Zp-Xp / Yp-Zp plane selection
Xp_: X-axis or a parallel axis (set in parameter No. 1022)
Yp_: Y-axis or a parallel axis (set in parameter No. 1022)
Zp_: Z-axis or a parallel axis (set in parameter No. 1022)
I_,J_,K_ : Center of the base circle for an involute curve viewed from the start point
R_ : Base circle radius
F_ : Cutting feedrate



Clockwise involute interpolation (G02.2)

Counterclockwise involute interpolation (G03.2)

Fig.4.12 (a) Actual Movement

Explanation

- Involute curve

An involute curve on the X-Y plane is defined as follows ;

$X(\theta)=R[\cos\theta + (\theta - \theta o) \sin\theta] + Xo$

$Y(\theta)=R[\sin\theta + (\theta - \theta o) \cos\theta] + Yo$

where,

Xo, Yo : Coordinates of the center of a base circle

R : Base circle radius

θo : Angle of the start point of an involute curve

θ : Angle of the point where a tangent from the current position to the base circle contacts the base circle

X(θ), Y(θ) :Current position on the X-axis and Y-axis

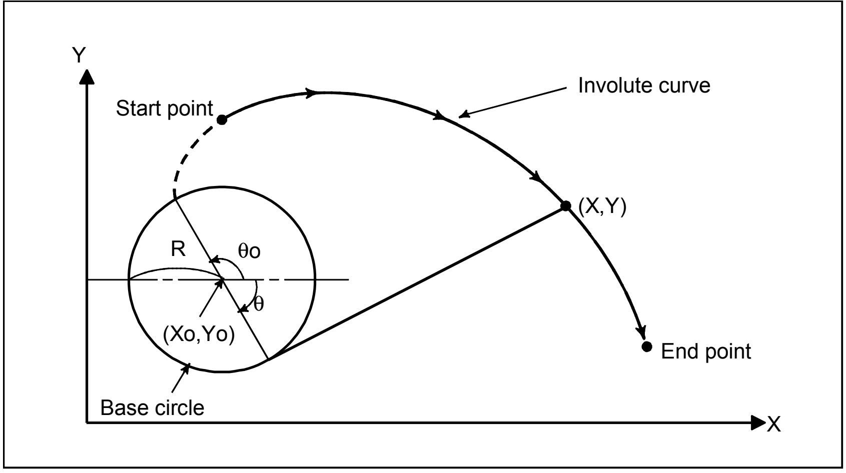


Fig.4.12 (b) Involute Curve

Involute curves on the Z-X plane and Y-Z plane are defined in the same way as an involute curve on the X-Y plane.

- Start point and end point

The end point of an involute curve is specified using address Xp, Yp, or Zp. An absolute value or incremental value is used to specify an Xp, Yp, or Zp value. When using an incremental value, specify the coordinates of the end point viewed from the start point of the involute curve.

When no end point is specified, P/S alarm No. 935 is issued.

If the specified start point or end point lies within the base circle, P/S alarm No. 936 is issued. The same alarm is issued if cutter compensation C causes the offset vector to enter the base circle. Be particularly careful when applying an offset to the inside of an involute curve.

- Base circle specification

The center of a base circle is specified with I, J, and K, corresponding to X, Y, and Z. The value following I, J, or K is a vector component defined when the center of the base circle is viewed from the start point of the involute curve; this value must always be specified as an incremental value, regardless of the G90/G91 setting. Assign a sign to I, J, and K according to the direction.

If I, J, and K are all left unspecified, or I0J0K0 is specified, P/S alarm No. 935 or No. 936 is issued.

If R is not specified, or R≤0, P/S alarm No. 935 or No. 936 is issued.

- Choosing from two types of involute curves

When only a start point and I, J, and K data are given, two types of involute curves can be created. One type of involute curve extends towards the base circle, and the other extends away from the base circle. When the specified end point is closer to the center of the base circle than the start point, the involute curve extends toward the base circle. In the opposite case, the involute curve extends away from the base circle.

- Feedrate

The cutting feedrate specified in an F code is used as the feedrate for involute interpolation. The feedrate along the involute curve (feedrate along the tangent to the involute curve) is controlled to satisfy the specified feedrate.

- Plane selection

As with circular interpolation, the plane to which to apply involute interpolation can be selected using G17, G18, and G19.

- Cutter compensation

Cutter compensation can be applied to involute curve machining. As with linear and circular interpolation, G40, G41, and G42 are used to specify cutter compensation.

G40:Cutter compensation cancel
G41:Cutter compensation left
G42:Cutter compensation right

At each of the start and end points of an involute curve, an intersection point with a straight line or arc is approximated. The involute curve passing through the obtained intersection points at the start and end points is used as the tool center path.

In involute interpolation mode, the G codes for cutter compensation, which are G41, G42, and G40, cannot be specified.

- Automatic speed control

To improve the machining precision, an override can be automatically applied to the specified feedrate during involute interpolation. See II-5.5.1 for details.

- Specifiable G codes

The following G codes can be specified in involute interpolation mode:

G04:Dwell
G10:Data setting
G17:X-Y plane selection
G18:Z-X plane selection
G19:Y-Z plane selection
G65:Macro call
G66:Macro modal call
G67:Macro modal call cancel
G90:Absolute command
G91:Incremental command

- Modes that allow involute interpolation specification

Involute interpolation can be specified in the following G code modes:

G41	: Cutter compensation left
G42	: Cutter compensation right
G51	: Scaling
G51.1	: Programmable mirror image
G68	: Coordinate rotation

- End point error

As shown below the end point may not be located on an involute curve that passes through the start point.

When an involute curve that passes through the start point deviates from the involute curve that passes through the end point by more than the value set in parameter No. 2510, P/S alarm No. 0937 is issued.

When there is an end point error, the feedrate is not guaranteed.

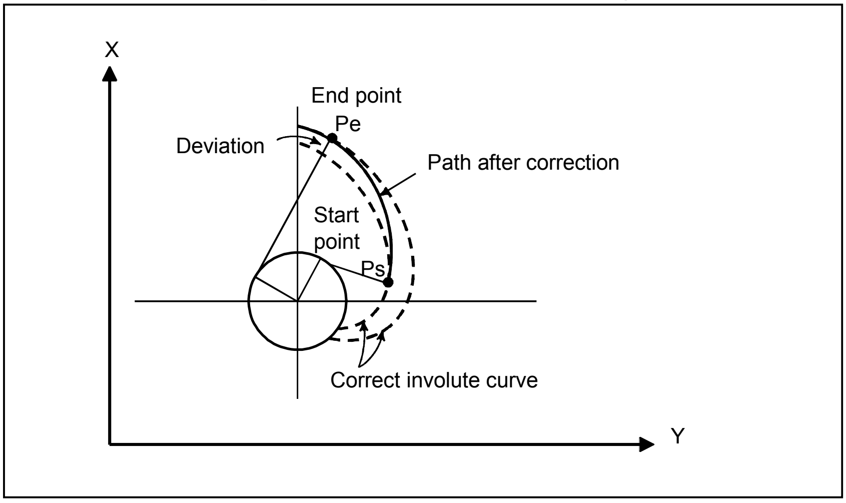


Fig.4.12 (c) End Point Error in Counterclockwise Involute Interpolation (G03.2)

Limitation

- Number of involute curve turns

Both the start point and end point must be within 100 turns from the point where the involute curve starts. An involute curve can be specified to make one or more turns in a single block.

- Unspecifiable functions

In involute interpolation mode, chamfer corner R (with an arbitrary angle).

- Modes that do not allow involute interpolation specification

Involute interpolation cannot be used in the following modes:

G07.1 (G107): Cylindrical interpolation

4.12.1 Involute Interpolation with a Linear Axis and Rotation Axis (G02.2,G03.3)

In the polar coordinate interpolation mode, an involute curve can be machined using involute interpolation. The involute curve to be machined is drawn in the plane of the linear axis and rotation axis.

Format

When the linear axis is the X-axis or an axis parallel to the X-axis

$\begin{Bmatrix}\textbf{G02.2}\\ \\ \textbf{G03.2}\end{Bmatrix}$ **X_C_ I_J_R_F_ ;**

When the linear axis is the Y-axis or an axis parallel to the Y-axis

$\begin{Bmatrix}\textbf{G02.2}\\ \\ \textbf{G03.2}\end{Bmatrix}$ **Y_C_ J_K_R_F_ ;**

When the linear axis is the Z-axis or an axis parallel to the Z-axis

$\begin{Bmatrix}\textbf{G02.2}\\ \\ \textbf{G03.2}\end{Bmatrix}$ **Z_C_ K_I_R_F_ ;**

G02.2 : Clockwise involute interpolation

G03.2 : Counterclockwise involute interpolation

Example) When the linear axis is the X-axis

X,C : End point linear axis coordinate of the involute curve, rotation axis

I,J : Center position of the base circle of the involute curve viewed from the start point

R : Radius of the base circle

F : Cutting feedrate

Explanation

- Center of the base circle

The center of the base circle of an involute curve, as viewed from the start point, is determined depending on the axis in the basic coordinate system that is used as the first axis of the plane.

- When the linear axis is the X-axis or an axis parallel to the X-axis, the Xp-Yp plane is assumed, and the center is specified using I and J.
- When the linear axis is the Y-axis or an axis parallel to the Y-axis, the Yp-Zp plane is assumed, and the center is specified using J and K.
- When the linear axis is the Z-axis or an axis parallel to the Z-axis, the Zp-Xp plane is assumed, and the center is specified using K and I.

- Specifying end coordinates

In polar coordinate interpolation mode, each position is represented by a distance from the center and an angle. The end coordinates are specified using Cartesian coordinates on the polar coordinate interpolation plane.

- Setting axes

The first axis (linear axis) and second axis (rotation axis) of the plane are set in parameter Nos. 1032 and 1033.

Limitation

- G codes that cannot be specified

In involute interpolation for a polar coordinate interpolation plane, only those G codes that can be used in both involute interpolation and polar coordinate interpolation can be used. The following G codes cannot be specified:

G17	: Xp-Yp plane selection
G18	: Zp-Xp plane selection
G19	: Yp-Zp plane selection
G10	: Programmable data input

- Modes that must be canceled

Before this function can be specified, the following modes must be canceled:

G51.1	: Programmable mirror image
G68	: Coordinate system rotation

Example

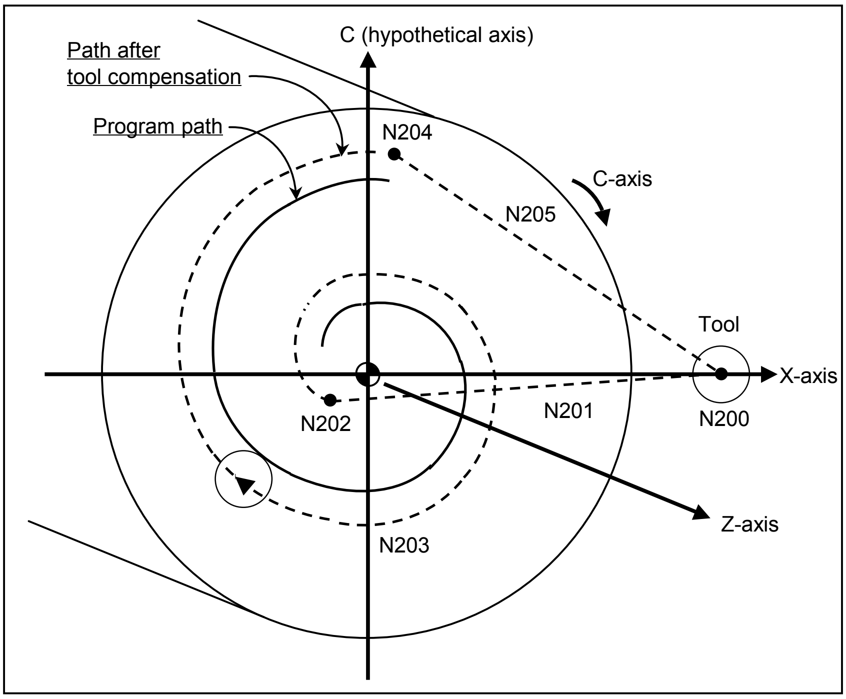


Fig.4.12.1 (a) Involute interpolation during polar coordinate interpolation

```
O0001 ;
  :
N010 T0101 ;
  :
N100 G90 G00 X15.0 C0 Z0 ;          Positioning to the start position
N200 G12.1 ;                        Start of polar coordinate
                                    interpolation
N201 G41 G00 X-1.0 ;
N202 G01 Z-2.0 F_ ;
N203 G02.2 X1.0 C9.425 I1.0 J0 R1.0 ;  Involute interpolation
                                       during polar
N204 G01 Z0 ;                          coordinate interpolation
N205 G40 G00 X15.0 C0 ;
N206 G13.1 ;                        Polar coordinate interpolation
                                    cancel
N300 Z_ ;
N400 X_ C_ ;
  :
M30 ;
```

4.13 HELICAL INVOLUTE INTERPOLATION (G02.2,G03.3)

This interpolation function applies involute Interpolation to two axes and directs movement for up to four other axes at the same time. This function is similar to the helical function used in circular interpolation.

Format

Involute interpolation in the Xp-Yp plane,

$$\text{G17} \left\{ \begin{matrix} \text{G02.2} \\ \\ \text{G03.2} \end{matrix} \right\} \text{Xp\_Yp\_} \quad \text{I\_J\_R\_} \quad \alpha\_\ \beta\_\ \gamma\_\ \delta\_\ \text{F\_;}$$

Involute interpolation in the Zp-Xp plane,

$$\text{G18} \left\{ \begin{matrix} \text{G02.2} \\ \\ \text{G03.2} \end{matrix} \right\} \text{Zp\_Xp\_} \quad \text{K\_I\_R\_} \quad \alpha\_\ \beta\_\ \gamma\_\ \delta\_\ \text{F\_;}$$

Involute interpolation in the Yp-Zp plane,

$$\text{G19} \left\{ \begin{matrix} \text{G02.2} \\ \\ \text{G03.2} \end{matrix} \right\} \text{Yp\_Zp\_} \quad \text{J\_K\_R\_} \quad \alpha\_\ \beta\_\ \gamma\_\ \delta\_\ \text{F\_}$$

α, β, γ, δ : Any one axis where circular interpolation is not applied.

Explanation

- Commanded axes

Addresses α, β, γ, and δ must indicate axes other than Yp and Zp. Other addresses must be the same as those used in the conventional involute interpolation.

- Tool offset

Cutter compensation is only applied to involute curves.

- Linear axis and rotation axis

This function can be used for involute interpolation for a linear axis and rotation axis.

4.14 SPLINE INTERPOLATION (G06.1)

Spline interpolation produces a spline curve connecting specified points. When this function is used, the tool moves along the smooth curve connecting the points. The spline interpolation command eliminates the need to approximate the smooth curve with minute straight lines or arcs. A machining program coded with this command requires less tape than that including the approximation.

Format

The following command sets spline interpolation mode:
G06.1 ;
In the G06.1 block, a tangent vector at the start point can be specified.
G06.1 X_ Y_ Z_ ;
X_ : X-axis component of the tangent vector
Y_ : Y-axis component of the tangent vector
Z_ : Z-axis component of the tangent vector

Explanation

- Spline curve

A spline curve connecting n + 1 points consists of n parametric cubic curves. The general formula f(t) of the cubic curve is expressed as:

$$P = At^3 + Bt^2 + Ct + D$$

where P is the position vector at a point on the curve, t is a parameter, and A, B, C, and D are vector coefficients.

The CNC unit performs spline interpolation by calculating the coefficients according to the specified points and changing t.

- Conditions of specified points

In the spline interpolation mode, two curves joining at a point satisfy the following conditions at that point. This results in a smooth spline curve.

- The two curves have an identical joining point (specified point).
- The two curves have an identical tangential vector at the joining point. (The two curves have an identical first-order differential vector obtained for t.)
- The two curves have an identical curvature at the joining point. (The two curves have an identical second-order differential vector obtained for t.)

- Feedrate

As the feedrate in spline interpolation mode, specify a tangential feedrate using the F code.

This tangential feedrate is found in the Cartesian coordinate system generated by the specified axes. During spline interpolation, the feedrate component on each axis changes with time. Let Fx, Fy, and Fz be the feedrate components on the axes at a given point of time. Then, tangential feedrate F is obtained as follows:

$$F = (Fx^2 + Fy^2 + Fz^2)^{\frac{1}{2}}$$

- Specifying a G06.1 block or next block

The axes to be specified in spline interpolation mode must all be specified in a block containing G06.1 or the next block.

- When a tangent vector is specified in the G06.1 block, it is specified together with the G06.1 block.

```
N100 G06.1 X10. Y10. Z-10. ;   ←  Specified in this block
N110 X15. ;
N120 Y20. Z10. ;
N130 X20. Y25. Z-10.
```

- When a tangent vector is not specified in the G06.1 block, commands for all axes are specified in the block next to the G06.1 block.

```
N100 G06.1 ;
N110 X15. Y0 Z0 ;   ←  Specified in this block
N120 Y20. Z10. ;
N130 X20. Y25. Z-10.
```

- Smooth spline curve

To obtain a smooth spline curve, observe the following items when creating a program:

- Points specified for spline interpolation should be equally spaced wherever possible.
- The magnitude of the tangent vector specified in the G06.1 block should match the distance between the start point of spline interpolation and the second point wherever possible.
- Blocks in spline interpolation mode must always contain a move command.
- Neither non-buffering G codes nor M codes must be specified.

- G code group

The spline interpolation command (G06.1) belongs to G code group 01.

- Auxiliary functions

When an auxiliary function is specified, it must be specified in a block containing a move command.

- Necessary option

This function requires the use of the multi-buffer option.

- G codes that can be specified in spline interpolation mode

The following G codes can be specified in the spline interpolation mode:

G61	: Exact stop check
G64	: Cutting mode
G65	: Macro calling
G90	: Absolute command
G91	: Incremental command

The following G codes must be specified in a block containing the corresponding move command: G61, G64, G90, and G91.

- Modes in which spline interpolation can be specified

The spline interpolation mode can be specified in the following G-code modes:

G17	: Selection of the XY plane
G18	: Selection of the ZX plane
G19	: Selection of the YZ plane
G20	: Input in inches
G21	: Input in millimeters
G22	: Stored stroke check function on
G23	: Stored stroke check function off
G40	: Cancellation of cutter compensation or three-dimensional cutter compensation
G41	: Three-dimensional tool compensation
G43	: Tool length compensation mode
G49	: Cancellation of tool length compensation
G50	: Cancellation of scaling
G51	: Scaling
G50.1	: Cancellation of programmable mirror image
G51.1	: Programmable mirror image
G54	: Selection of workpiece coordinate system 1
G55	: Selection of workpiece coordinate system 2
G56	: Selection of workpiece coordinate system 3
G57	: Selection of workpiece coordinate system 4
G58	: Selection of workpiece coordinate system 5
G59	: Selection of workpiece coordinate system 6
G61	: Exact stop check
G64	: Cutting mode
G66	: Continuous-state macro calling A
G66.1	: Continuous-state macro calling B
G67	: Cancellation of continuous-state macro calling A or B
G68	: Coordinate system rotation
G69	: Cancellation of coordinate system rotation
G80	: Cancellation of canned cycle
G90	: Absolute command
G91	: Incremental command
G94	: Feed per minute

- Sample program

The system is in the spline interpolation mode from N120 to N500 of the program below:

N110 G00 X_Y_Z_ ; P_{1}
N120 G06.1 ;
N130 X_Y_Z_ ; P_{2}
N140 X_Y_Z_ ; P_{3}
N150 X_Y_Z_ ; P_{4}
:
N500 X_Y_Z_ ; P_{n}
N510 G00 X_Y_Z_ ; P_{n+1}

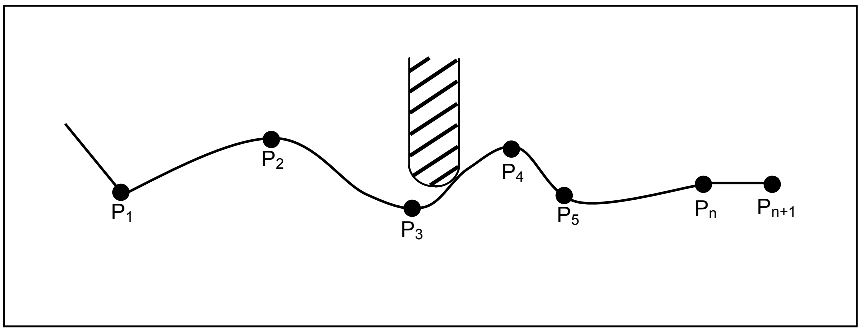


Fig.4.14 (a) Spline interpolation

- Three-dimensional offset

Spline interpolation can be executed in the three-dimensional tool compensation mode. The spline interpolation function automatically produces vectors for three-dimensional tool compensation in the spline interpolation mode. In the three-dimensional tool compensation mode, a spline curve connects the specified points which are offset by the vectors for three-dimensional tool compensation.

The spline interpolation function determines the vectors for three-dimensional tool compensation as shown below:

1) Three-dimensional tool compensation vector at the start point

   The three-dimensional tool compensation vector specified in the block of three-dimensional tool compensation command is used.

   Three-dimensional tool compensation vector K specified by G41 Ii Jj Kk Dd; has components Kx, Ky, and Kz, each of which is calculated as follows:

   $Kx \ = (i \times r) \div p$

   $Ky \ = (j \times r) \div p$

   $Kz \ = (k \times r) \div p$

   where

   r is the offset corresponding to the specified offset number d.

   $p = \sqrt{i^2 + j^2 + k^2}$

2) Three-dimensional tool compensation vector at the second or subsequent point

   Position : The vector is on the plane containing the point, previous point, and next point. It is perpendicular to the straight line connecting the previous and next points.

   Direction : The direction of the vector is close to that of the three-dimensional tool compensation vector at the previous point. (When the direction of three-dimensional tool compensation vector V at the point is close to that of three-dimensional tool compensation vector V0 at the previous point, the angle θ between V0 and V satisfies the following condition : $|\theta| < 90°$)

   Magnitude : The magnitude of the vector is the offset corresponding to the offset number specified by G41.

3) Three-dimensional tool compensation vector at the last point

   Position : The vector is on the plane containing the point, previous point, and next point. It is perpendicular to the straight line connecting the previous and next points.

   Direction : The direction of the vector is close to that of the three-dimensional tool compensation vector at the previous point. (When the direction of three-dimensional tool compensation vector V at the point is close to that of three-dimensional tool compensation vector V0 at the previous point, the angle θ between V0 and V satisfies the following condition : $|\theta| < 90°$)

   Magnitude : The magnitude of the vector is the offset corresponding to the offset number specified by G41.

   After the system exits from the spline interpolation mode, the three-dimensional tool compensation vector produced at the last point is used.

4) Other specification

   - If a point after the first point and the previous and next points are on a straight line, the vector plane cannot be determined for the three-dimensional tool compensation vector at the second or subsequent point. If this happens, the vector plane is determined by the straight line and the three-dimensional tool compensation vector at the previous point.

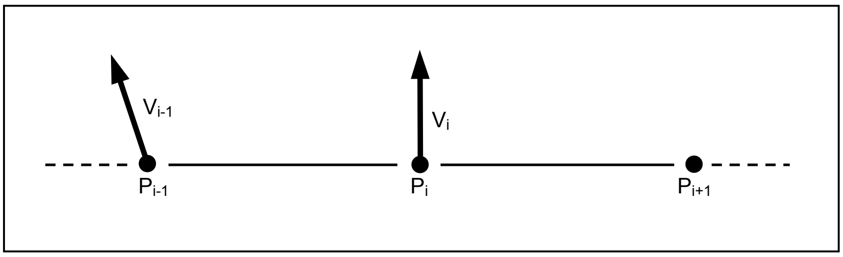


Fig.4.14 (b) Vector 1

- If the points before and after a point other than the first point are identical, the three-dimensional tool compensation vector at the second or later point is produced in the direction of the straight line connecting the point and the previous (or next) point.

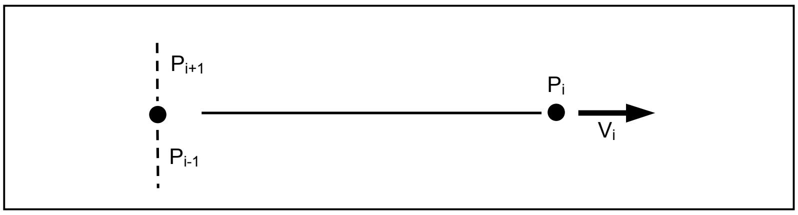


Fig.4.14 (c) Vector 2

- If a point after the first point and the previous and next points are all identical, the three-dimensional tool compensation vector at the previous point is used as that at the point.
- Specify three points subjected to spline interpolation for which the three-dimensional tool compensation vectors are produced in such a manner that they make an obtuse angle. If they make an

angle of 90° or less, the vector may not be produced in the correct direction.

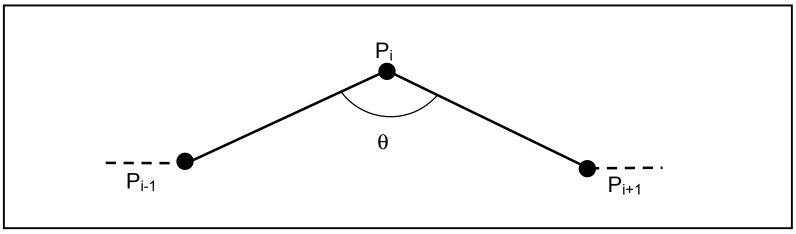


Fig.4.14 (d) Vector 3

- Sample program of three-dimensional tool offset

The system is in the spline interpolation mode included in the three-dimensional tool compensation mode from N120 to N600 of the program below:

O1000 ;		Specified point	Offset point
N110 G01 G41 X_Y_ Z_ I_ J_ K_ D_ F ;	:	P_1	Q_1
N120 G06.1 ;	:	P_2	Q_2
N130 X_ Y_ Z_ ;	:	P_3	Q_3
N140 X_ Y_ Z_ ;	:	P_4	Q_4
N150 X_ Y_ Z_ ;	:	P_5	Q_5
:			
N600 X_ Y_ Z_ ;	:	P_n	Q_n
N610 G01 X_ Y_ Z_ ;	:	P_{n+1}	

K : Three-dimensional tool compensation vector specified by G41

V_1 to Vn :Three-dimensional tool compensation vector used in the spline interpolation mode (V1 = K)

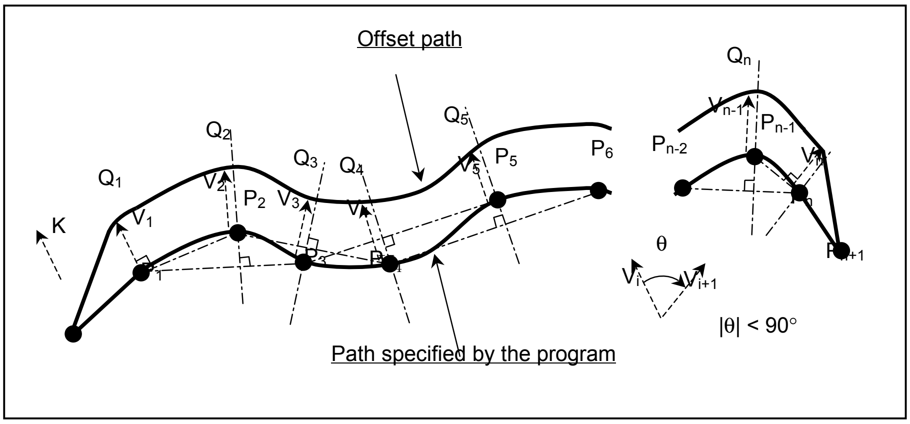


Fig.4.14 (e) Three-dimensional tool offset

Limitation

- Modes not allowed

Before specifying G06.1, cancel canned cycle mode, tool offset mode, and cutter compensation mode if these modes are set.

- First block of the subprogram

Specify a move command in the first block of the subprogram to be called in the spline interpolation mode.

<Example> O1000; ⟶ O1000 X_ Y_ Z_ ;

4.15 SPIRAL INTERPOLATION, CONICAL INTERPOLATION (G02,G03)

Spiral interpolation is enabled by specifying the circular interpolation command together with a desired number of revolutions or a desired increment (decrement) for the radius per revolution.
Conical interpolation is enabled by specifying the spiral interpolation command together with one or two additional axes of movement, as well as a desired increment (decrement) for the position along the additional axes per spiral revolution.

Format

- Spiral interpolation

Xp-Yp plane

$$\text{G17}\left\{\begin{array}{c}\text{G02}\\ \\ \text{G03}\end{array}\right\}\text{X\_ Y\_ I\_ J\_ Q\_ L\_ F\_ ;}$$

Zp-Yp plane

$$\text{G18}\left\{\begin{array}{c}\text{G02}\\ \\ \text{G03}\end{array}\right\}\text{Z\_ X\_ K\_ I\_ Q\_ L\_ F\_ ;}$$

Yp-Zp plane

$$\text{G19}\left\{\begin{array}{c}\text{G02}\\ \\ \text{G03}\end{array}\right\}\text{Y\_ Z\_ J\_ K\_ Q\_ L\_ F\_ ;}$$

X,Y,Z:Coordinates of the end point
L :Number of revolutions (positive value without a decimal point) (*1)
Q :Radius increment or decrement per spiral revolution (*1)
I,J,K :Signed distance from the start point to the center (same as the distance specified for circular interpolation)
F :Feedrate

(*1) Either the number of revolutions (L) or the radius increment or decrement (Q) can be omitted. When L is omitted, the number of revolutions is automatically calculated from the distance between the current position and the center, the position of the end point, and the radius increment or decrement. When Q is omitted, the radius increment or decrement is automatically calculated from the distance between the current position and the center, the position of the end point, and the number of revolutions. If both L and Q are specified but their values contradict, Q takes precedence. Generally, either L or Q should be specified. The L value must be a positive value without a decimal point. To specify four revolutions plus 905, for example, round the number of revolutions up to five and specify L5.

- Conical interpolation

Xp-Yp plane

$\text{G17}\left\{\begin{array}{l}\text{G02}\\ \\ \text{G03}\end{array}\right\}$ **X_ Y_ I_ J_ Z_ Q_ L_ F_ ;**

Zp-Yp plane

$\text{G18}\left\{\begin{array}{l}\text{G02}\\ \\ \text{G03}\end{array}\right\}$ **Z_ X_ K_ I_ Y_ Q_ L_ F_ ;**

Yp-Zp plane

$\text{G19}\left\{\begin{array}{l}\text{G02}\\ \\ \text{G03}\end{array}\right\}$ **Y_ Z_ J_ K_ X_ Q_ L_ F_ ;**

X,Y,Z:Coordinates of the end point

L : Number of revolutions (positive value without a decimal point) (*1)

Q : Radius increment or decrement per spiral revolution (*1)

I,J,K: Two of the three values represent a signed vector from the start point to the center. The remaining value is a height increment or decrement per spiral revolution in conical interpolation(*1))

When the XpYp plane is selected:

The I and J values represent a signed vector from the start point to the center.
The K value represents a height increment or decrement per spiral revolution.

When the ZpXp plane is selected:

The K and I values represent a signed vector from the start point to the center.
The J value represents a height increment or decrement per spiral revolution.

When the YpZp plane is selected:

The J and K values represent a signed vector from the start point to the center.
The I value represents a height increment or decrement per spiral revolution.

F :Feedrate (*2)

(*1) One of the height increment/decrement (I, J, K), radius increment/decrement (Q), and the number of revolutions (L) must be specified. The other two items can be omitted.

- Sample command for the Xp-Yp plane

$$\mathbf{G17} \begin{Bmatrix} \mathbf{G02} \\ \mathbf{G03} \end{Bmatrix} \mathbf{X\_\ Y\_\ I\_\ J\_\ Z\_} \begin{Bmatrix} \mathbf{K\_} \\ \mathbf{Q\_} \\ \mathbf{L\_} \end{Bmatrix} \mathbf{F\_\ ;}$$

If both L and Q are specified, but their values contradict, Q takes precedence. If both L and a height increment or decrement are specified, but their values contradict, the height increment or decrement takes precedence. If both Q and a height increment or decrement are specified, but their values contradict, Q takes precedence. The L value must be a positive value without a decimal point. To specify four revolutions plus 905, for example, round the number of revolutions up to five and specify L5.

(*2)As the feedrate, whether to specify a feedrate tangent to an arc or a tangential feedrate, determined also by considering movement along the linear axes, can be set in bit 2 (HTG) of parameter No. 1401.

Explanation

- Function of spiral interpolation

Spiral interpolation in the XY plane is defined as follows:

$(X-X_0)^2+(Y-Y_0)^2=(R+Q')^2$

X_0 : X coordinate of the center

Y_0 : Y coordinate of the center

R : Radius at the beginning of spiral interpolation

Q' : Variation in radius

When the programmed command is assigned to this function, the following expression is obtained:

$$(X-X_S-I)^2+(Y-Y_S-J)^2= ((R + (L'+ \frac{\theta}{360})Q)^2$$

where

X_S : X coordinate of the start point

Y_S : Y coordinate of the start point

I : X coordinate of the vector from the start point to the center

J : Y coordinate of the vector from the start point to the center

R : Radius at the beginning of spiral interpolation

Q : Radius increment or decrement per spiral revolution

L' : (Current number of revolutions)-1

θ : Angle between the start point and the current position(degrees)

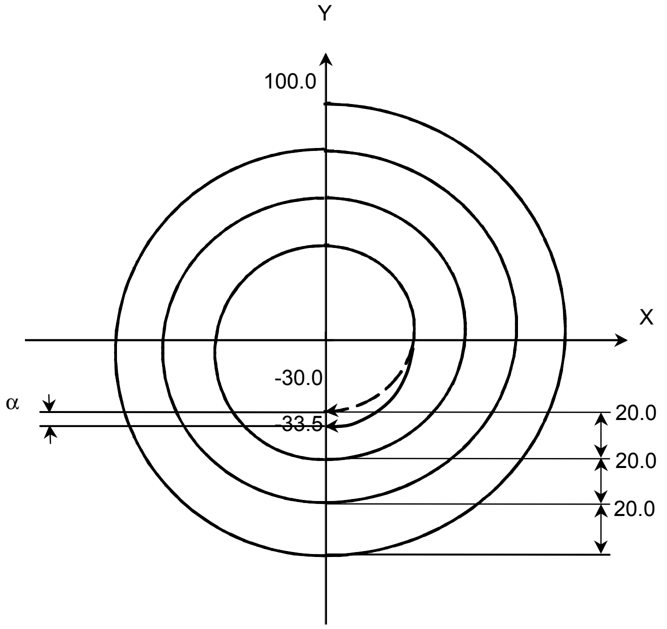
- Controlled axes

For conical interpolation, two axes of a plane and two additional axes, that is, four axes in total, can be specified. A rotation axis can be specified as the additional axis.

- Difference in end position

If the difference between the positions of specified and calculated end points of a helix along an axis in the specified plane exceeds the value set in parameter 2511, alarm PS281 is issued.

If the difference in height between the positions of specified and calculated end points of a helix exceeds the value set in parameter 2511, alarm PS281 is issued.



G90 G02 X0 Y-33.5 I0 J-100. $\left\{\begin{array}{c}\text{Q-20.}\\ \text{L4}\end{array}\right\}$ F300 ;

When the specified end point is (0, -33.5), the calculated end point is (0, -30.0). Specify a value greater than the difference (a : permissible error) in parameter 2511. Also set the parameter in this way, for the difference in height in conical interpolation.

Fig.4.15 (a) Difference in end position

- Tool offset

The spiral interpolation function and conical interpolation function can be used in cutter compensation C mode. The same compensation is applied as that described in (d) Exceptional case, (3) Offset mode, II-14.4.3 Detailed explanation of cutter compensation C. Cutter compensation is applied along a virtual circle whose center is located at the center of spiral interpolation and which passes through the end point of a block. Spiral interpolation is applied to the path obtained after cutter compensation is applied.

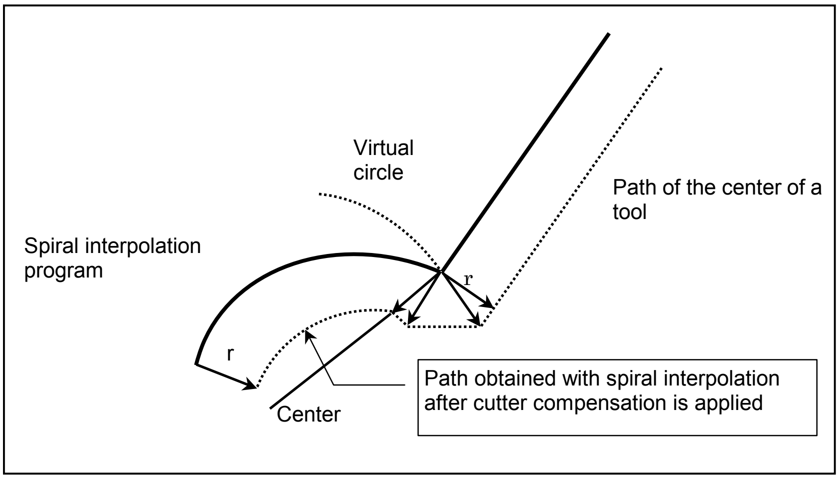
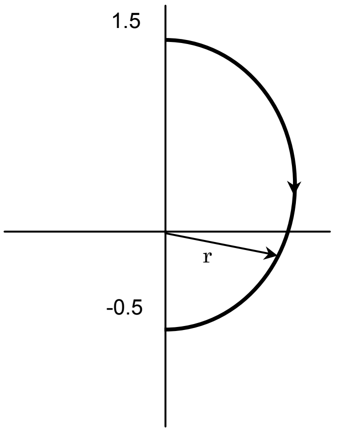

Virtual circle
Path of the center of a tool
Spiral interpolation program
r
r
Center
Path obtained with spiral interpolation after cutter compensation is applied

Fig.4.15 (b) Tool offset

- Actual speed

In spiral interpolation and conical interpolation, the feedrate is usually kept constant. When the radius of a spiral becomes small near the center of the spiral, however, the corresponding angular velocity may become very high. To prevent this from occurring, the system keeps the angular velocity constant when the radius of a spiral reaches the value specified in parameter No.2440, resulting in a lower velocity than before.

An example is shown below.



Specified feedrate : F100
Parameter No.2440(r) : 1000
Increment system : IS-B

When spiral interpolation is used under the above conditions, feedrate is F100 for a radius larger then the value specified in parameter No.2440. The feedrate is reduced as the radius becomes smaller and is about F65 at the end point of the spiral.

Fig.4.15 (c) Actual speed

- Feedrate clamping by acceleration

During spiral interpolation, the function for clamping the feedrate by acceleration (parameter No. 1663) is enabled. The feedrate may decrease as the tool approaches the center of the spiral.

- Dry run

When the dry run signal is inverted from 0 to 1 or from 1 to 0 during movement along an axis, the movement is accelerated or decelerated to the desired speed without first reducing the speed to zero.

Limitation

- Radius

In spiral or conical interpolation, R for specifying an arc radius cannot be specified.

- Feed functions

The functions of feed per rotation, inverse time feed, F command with one digit, and automatic corner override cannot be used.

- Axis name expansion function

If the axis name extension function specifies I, J, K, or Q as an axis name, the spiral or conical interpolation function cannot be used.

- Additional function

To use the conical interpolation function, the optional helical interpolation is required in addition to this option.

Example

- Spiral interpolation

The path indicated above is programmed with absolute and incremental values, as shown below:

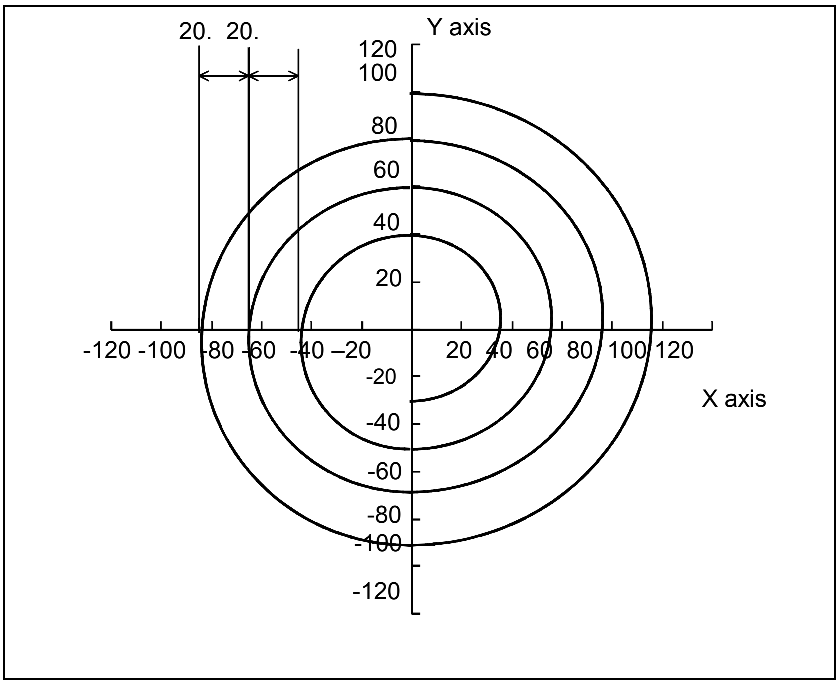


Fig.4.15 (d) Spiral interpolation

This sample path has the following values:

- Start point : (0,100.0)
- End point (X,Y) : (0,-30.0)
- Distance to the center (I,J) : (0,-100.0)
- Radius increment or decrement (Q) : -20.0
- Number of revolutions (L) : 4

(1) With absolute values, the path is programmed as follows:
G90 G02 X0 Y-30.0 I0 J-100.0 {Q-20.0 or L4} F300 ;

(2) With incremental values, the path is programmed as follows:
G91 G02 X0 Y-30.0 I0 J-100.0{Q-20.0 or L4} F300 ;
(Either the Q or L setting can be omitted.)

- Conical interpolation

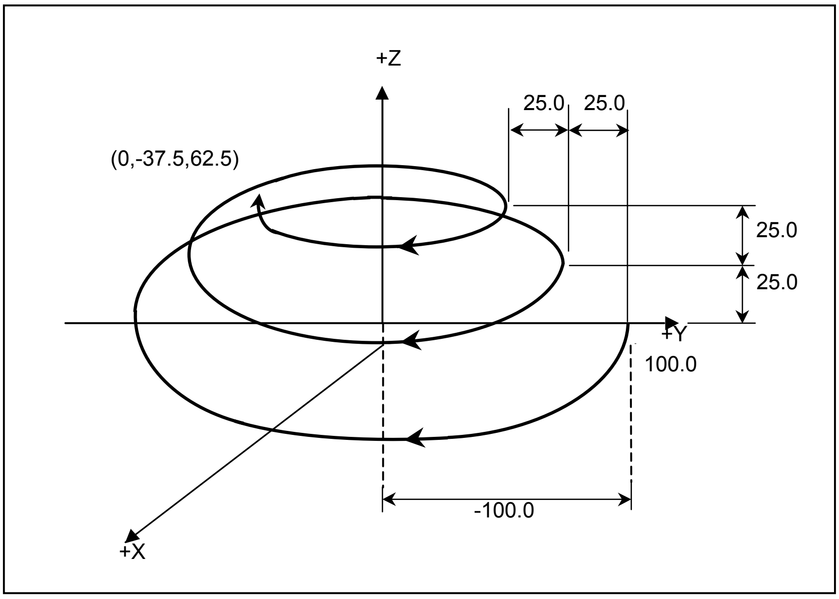


Fig.4.15 (e) Conical interpolation

The sample path shown above is programmed with absolute and incremental values as follows:

This sample path has the following values:

- Start point : (0,100.0,0)
- End point (X,Y) : (0,-37.5,62.5)
- Distance to the center (I,J) : (0,-100.0)
- Radius increment or decrement (Q) : -25.0
- Height increment or decrement (K) : 25.0
- Number of revolutions (L) : 3

(1) With absolute values, the path is programmed as follows:
G90 G02 X0 Y-37.5 Z62.5 I0 J-100.0 { K25.0, Q-25.0, or L3} F300 ;

(2) With incremental values, the path is programmed as follows:
G90 G02 X0 Y-137.5 Z62.5 I0 J-100.0 {K25.0, Q-25.0, or L3} F300 ;

4.16 SMOOTH INTERPOLATION (G05.1)

Either of two types of machining can be selected, depending on the program command.

1) For those portions where the accuracy of the figure is critical, such as at corners, machining is performed exactly as specified by the program command.
2) For those portions having a large radius of curvature where a smooth figure must be created, points along the machining path are interpolated with a smooth curve, calculated from the polygonal lines specified with the program command (smooth interpolation).

Format

```
G5.1Q2X0Y0Z0 ;     : Starting of smooth interpolation mode

G5.1Q0 ;           : Cancellation of smooth interpolation mode
```

Explanation

- Characteristics of smooth interpolation

To machine a part having sculptured surfaces, such as metal moldings used in automobiles and airplanes, a part program usually approximates the sculptured surfaces with minute line segments. As shown in the following figure, a sculptured curve is normally approximated using line segments with a tolerance of about 10 µm.

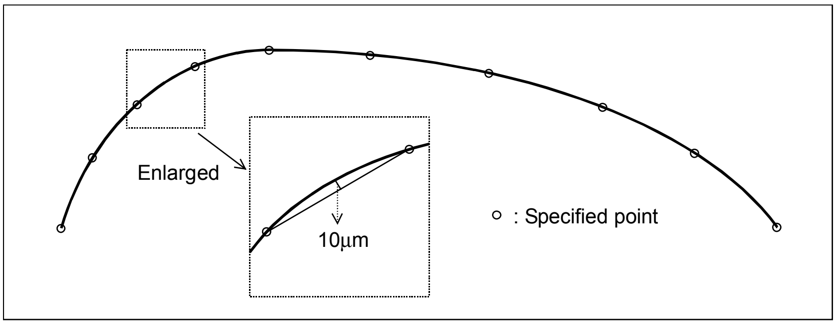


Fig.4.16 (a) Approximation with Line Segments

When a program approximates a sculptured curve with line segments, the length of each segment differs between those portions that have a small radius of curvature and those that have a large radius of curvature. The line segments are short in those portions having a small radius of curvature, while they are long in those portions having a large radius of curvature.

The CNC moves the tool along a programmed path thus enabling highly precise machining. This means that the tool movement precisely follows the line segments used to approximate a sculptured curve. This may result in a non-smooth machined curve when a curve in which the radius of curvature is large and changes only gradually is machined. Although this effect is a result of high-precision machining, which precisely follows a programmed path, the uneven surfaces will be judged as being unsatisfactory when smooth surfaces are required.

Table 4.16 (a) Profiles and Radius of Curvature

Profile	Small radius of curvature	Large radius of curvature
Example of machined parts	Parts used within an automobile	Automobile body parts
Length of line segment	Short	Long
Resulting surface produced using linear interpolation	Smooth surface	Uneven surface may result

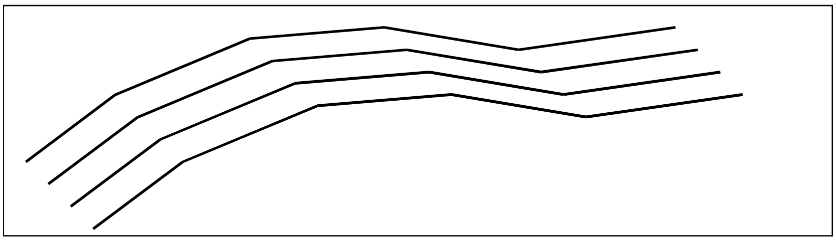


Fig.4.16 (b) Example of Uneven Machined Surface

The CNC automatically determines, according to the program commands, whether an accurate figure is required, such as at corners, or whether a smooth figure is required where the radius of curvature is large. If a block specifies a travel distance or direction which differs greatly from that in the preceding block, smooth interpolation is not performed for that block. Linear interpolation is performed exactly as specified by the program command. Programming is thus very simple.

NOTE

When the CNC judgement of smooth interpolation ON/OFF is not suitable, please change the followings;
- The command point of the part program
- The parameters No.7672-7677.
- The part program so that smooth interpolation is canceled temporary.

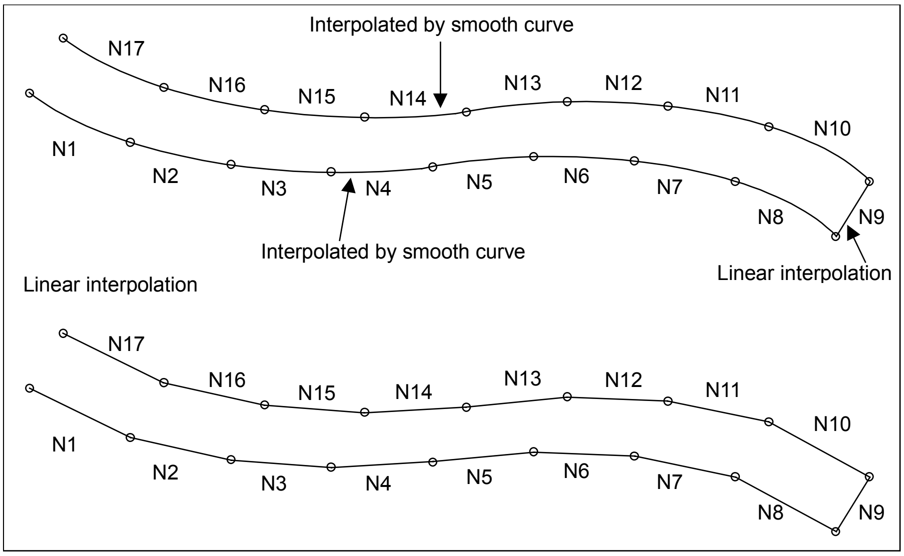


Fig.4.16 (c) Smooth Interpolation and Linear Interpolation

- Conditions for performing smooth interpolation

Smooth interpolation is performed when all the following conditions are satisfied. If any of the following conditions is not satisfied for a block, that block is executed without smooth interpolation then the conditions are checked for the next block.

(1) The machining length specified in the block is shorter than the length specified with parameter No. 7672 and longer than that specified with parameter No. 7675.

(2) The difference between the angles specified in blocks is smaller than the value specified with parameter No. 7673.

(3) The tolerance specified in the block is smaller than the value specified with parameter No. 7676 and larger than that specified with parameter No. 7677.

(4) The modes are:
G01 : Linear interpolation
G13.1 : Polar coordinate interpolation cancel
G15 : Polar coordinate command cancel
G40 : Cutter compensation cancel
(except for 3-dimensional tool compensation)
G64 : Cutting mode
G80 : Canned cycle cancel
G94 : Feed per minute

(5) Machining is specified only along the axes specified with G05.1Q2.

(6) The internal algorithm of the CNC judges the block to be suitable for smooth interpolation.

- Commands which cancel smooth interpolation

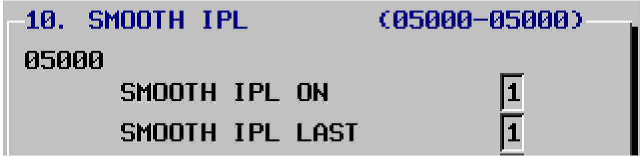
When one of the following commands is specified, smooth interpolation is canceled:

(1) G04 : Dwell
G09 : Exact stop check
G31,G31.1,G31.2,G31.3: Skip function
G37 : Tool length measurement

(2) M code that does not buffer.

- Checking whether smooth interpolation is specified

By referring to data (No. 5000) on the diagnosis screen, the user can check whether smooth interpolation has been specified in the block being executed.



```
10. SMOOTH IPL          (05000-05000)
05000
      SMOOTH IPL ON                1
      SMOOTH IPL LAST              1
```

If Smooth Interpolation On is set to 1, the block being executed is performing smooth interpolation.

If Smooth Interpolation Last is set to 1, smooth interpolation will be canceled at the end point of the block being executed.

Limitation

- Controlled axes

Only the X, Y, and Z-axes can be specified for smooth interpolation.

- Back-and-forth path machining

When back-and-forth path machining is performed for a path that includes inflection points, the back and forth level difference may increase.

In particular, if the positions of specified points differ greatly between the back and forth paths, a large back and forth level difference is produced.

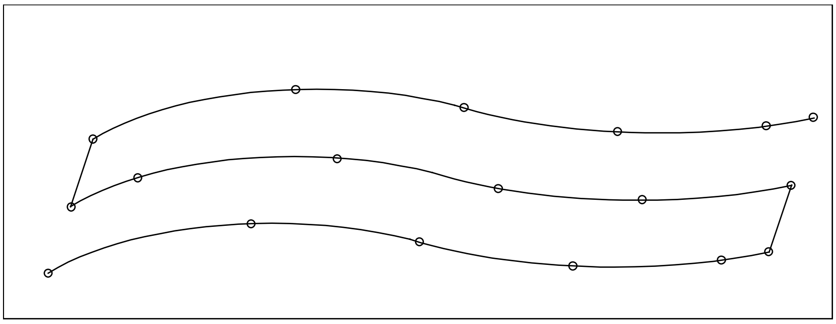


Fig.4.16 (d) Example When a Large Back and Forth Level Difference is Produced

In such a case, modify the program as follows:

1) Perform machining with a unidirectional path.
2) In portions near inflection points, specify closely spaced points.

Example

<Sample program of smooth interpolation>

```
G91
G5. 1 Q2 X0 Y0 Z0
N01 G01 X1000 Z-300 F500
N02 X1000 Z-200
N03 X1000 Z-50
N04 X1000 Z50
N05 X1000 Z50
N06 X1000 Z-25
N07 X1000 Z-175
N08 X1000 Z-350
N09 Y1000
N10 X-1000 Z350
N11 X-1000 Z175
N12 X-1000 Z25
N13 X-1000 Z-50
N14 X-1000 Z-50
N15 X-1000 Z50
N16 X-1000 Z200
N17 X-1000 Z300
G5.1 Q0
```

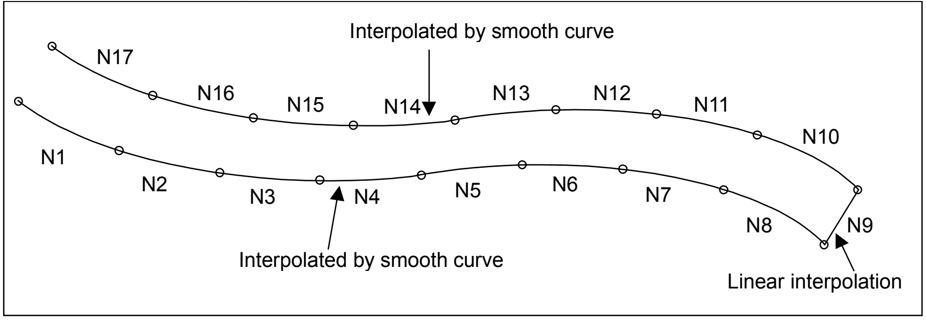


Fig.4.16 (e) Sample program of smooth interpolation

- Machining of figures containing inflection points

If a figure containing inflection points is machined, the path may rise greatly.

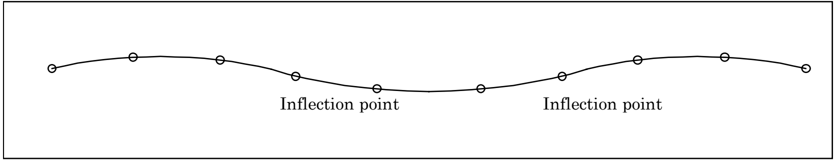


Fig.4.16 (f) Example of a figure containing inflection points

If this occurs, take the following actions:

1) Specify the points near the inflection points in detail.
2) Set a smaller value for the maximum tolerance parameter (No. 7676).
   (This is possible for systems having parameter No.7676.)
3) If the above actions do not solve the problem, turn smooth interpolation off in the affected portions.

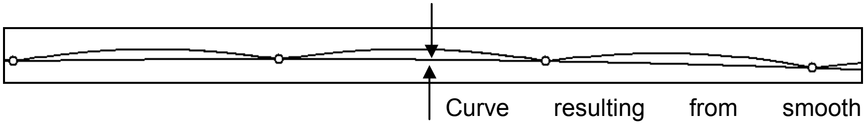
- Intervals of specified points

The intervals of specified points must be equal wherever possible. Otherwise, the path may rise greatly.

The figure below shows an enlarged view of the rises of a curve.

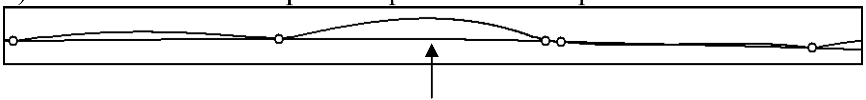
1) If the intervals of specified points are equal in general

Enlarged view of the rises of a curve (vertically magnified 20 times)



Curve resulting from smooth interpolation (equal scale for height and width)

2) If the intervals of specified points are not equal



Because the interval differs greatly from the subsequent one, the curve rises greatly in this section.

If this occurs, take the following actions:

1) Make the intervals of specified points equal wherever possible.
2) Set a smaller value for the maximum tolerance parameter (No. 7676).
   (This is possible for systems having parameter No.7676.)
3) If the above actions do not solve the problem, turn smooth interpolation off in the affected portions.

4.17 NURBS INTERPOLATION (G06.2)

Many computer-aided design (CAD) systems used to design metal dies for automobiles utilize non-uniform rational B-spline (NURBS) to express a sculptured surface or curve for the metal dies.

This function enables NURBS curve expression to be directly specified to the CNC. This eliminates the need for approximating the NURBS curve with minute line segments. This offers the following advantages:

1. No error due to approximation of a NURBS curve by small line segments
2. Short part program
3. No break between blocks when small blocks are executed at high speed
4. No need for high-speed transfer from the host computer to the CNC

When this function is used, a computer-aided machining (CAM) system creates a NURBS curve according to the NURBS expression output from the CAD system, after compensating for the length of the tool holder, tool diameter, and other tool elements. The NURBS curve is programmed in the NC format by using these three defining parameters: control point, weight and knot.

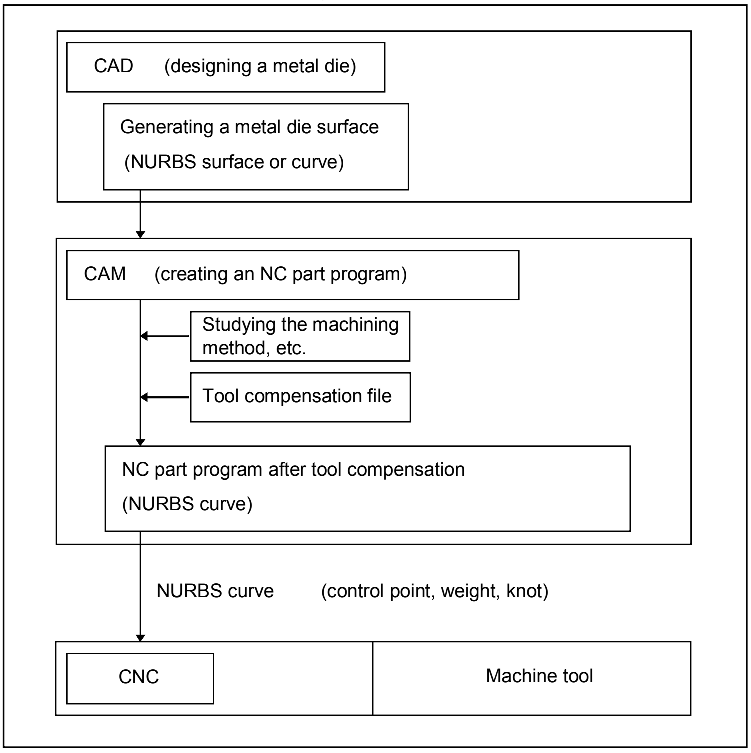

CAD (designing a metal die)
Generating a metal die surface
(NURBS surface or curve)
CAM (creating an NC part program)
Studying the machining method, etc.
Tool compensation file
NC part program after tool compensation
(NURBS curve)
NURBS curve (control point, weight, knot)
CNC
Machine tool

Fig.4.17 (a) NC part program for machining a metal die according to a NURBS curve

The CNC executes NURBS interpolation while smoothly accelerating or decelerating the movement so that the acceleration on each axis will not exceed the allowable maximum acceleration of the machine. In this way, the CNC automatically controls the speed in order to prevent excessive strain being imposed on the machine.

Format

```
G06.2 [P_ ] K_ IP_ [R_ ] [F_ ] ;
      K_ IP_ [R_ ] ;
      K_ IP_ [R_ ] ;
      K_ IP_ [R_ ] ;
      :
      K_ IP_ [R_ ] ;
      K_ ;
      …
      K_ ;
G01…
 :
 G06.2   : Start NURBS interpolation mode
 P_      : Rank of NURBS curve
 IP_     : Control point (Up to the maximum number of
           controlled axes can be specified.)
 R_      : Weight
 K_      : Knot
 F_      : Feedrate
```

Explanation

- NURBS interpolation mode

The NURBS interpolation mode is selected when G06.2 is programmed. G06.2 is a modal G code of group 01. So, the NURBS interpolation mode ends when a G code of group 01 other than G06.2 (e.g. G00, G01, G02, G03) is specified.

- NURBS rank

The NURBS rank can be specified with address P. The rank setting, if any, must be specified in the same block as the first control point. If the rank setting is omitted, a rank of four (degree of three) is assumed for NURBS. The valid data range for P is 2 to 4. The P values have the following meanings:
P2 : NURBS having a rank of two (degree of one)
P3 : NURBS having a rank of three (degree of two)
P4 : NURBS having a rank of four (degree of three)
This rank is expressed by k in the defining expression indicated in the description of the NURBS curve below. For example, a NURBS curve having a rank of four has a degree of three. The NURBS curve can be expressed by the constants t^3, t^2 and t^1.

- Weight

The weight of a control point programmed in a single block can be defined. When the weight setting is omitted, a weight of 1.0 is assumed. Weight is always specified as a 9-digit absolute value of the minimum data unit of the reference axis. For example, if the unit of the reference axis is set as millimeter input at IS-B, then weight can be specified within the range -999999.999 to +999999.999.

- Knot

The number of specified knots must equal the number of control points plus the rank value. In the blocks specifying the first to last control points, each control point and a knot are specified in the same block. After these blocks, as many blocks (including only a knot) as the rank value are specified. P/S alarm No. 1002 is issued for blocks in which knot is not specified.

The NURBS curve programmed for NURBS interpolation must start from the first control point and end at the last control point. The first k knots (where k is the rank) must have the same values as the last k knots (multiple knots). If multiple knots are not specified, P/S alarm No. 1007 is issued.

Knots of the same value that continue in excess of the rank value cannot be specified in other blocks. If knots of the same value are specified, P/S alarm No. 1008 is issued.

Knots are always specified as a 9-digit absolute value of the minimum data unit of the reference axis. For example, if the unit of the reference axis is set as millimeter input at IS-B, then weight can be specified within the range -999999.999 to +999999.999.

- NURBS curve

Using these variables:

k : Rank

$\boldsymbol{P}_i$: Control point

w_i : Weight

x_i : Knot $(x_i \le x_{i+1})$

Knot vector $[x_0, x_1, \ldots, x_m]$ $(m = n + k)$

t : Spline parameter,

the spline basis function N can be expressed with the de Boor-Cox recursive formula, as indicated below:

$$N_{i.1}(t) = \begin{cases} 1 & (x_i \le t \le x_{i+1}) \\ 0 & (t < x_i, x_{i+1} < t) \end{cases}$$

$$N_{i,k}(t) = \frac{(t - x_i) N_{i,k-1}(t)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - t) N_{i+1,k-1}(t)}{x_{i+k} - x_{i+1}}$$

The NURBS curve P(t) of interpolation can be expressed as follows:

$$\boldsymbol{P}(t) = \frac{\sum_{i=0}^{n} N_{i,k}(t) w_i \boldsymbol{P}_i}{\sum_{i=0}^{n} N_{i,k}(t) w_i}$$

$$(x_0 \le t \le x_m)$$

- NUBRS curved line segments

From the definition of NUBRS curved lines given above, it can be seen that the points on an NURBS curved line of rank n (degree (n-1)) consist of n successive control points. A part of an NUBRS curved line, generated by n successive control points, is referred to as a segment. When single block operation is performed, a single block stop occurs at each joint between the segments. Moreover, the valid speed command range is one segment. (For information about the valid speed command range, see the next item.)

Example 1) NUBRS curved line of rank 3 (degree 2) with 4 control points

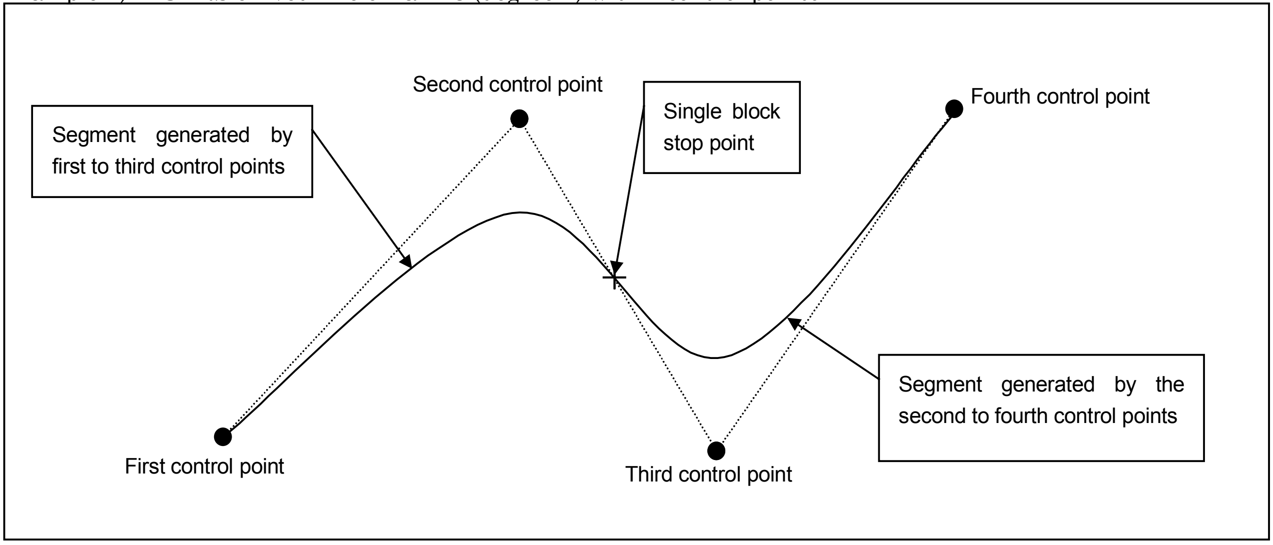

Second control point
Single block stop point
Fourth control point
Segment generated by first to third control points
Segment generated by the second to fourth control points
First control point
Third control point

Fig.4.17 (b) Segments of an NURBS Curved Line of Rank 3 (Degree 2)

Example 2) NUBRS curved line of rank 4 (degree 3) with 5 control points

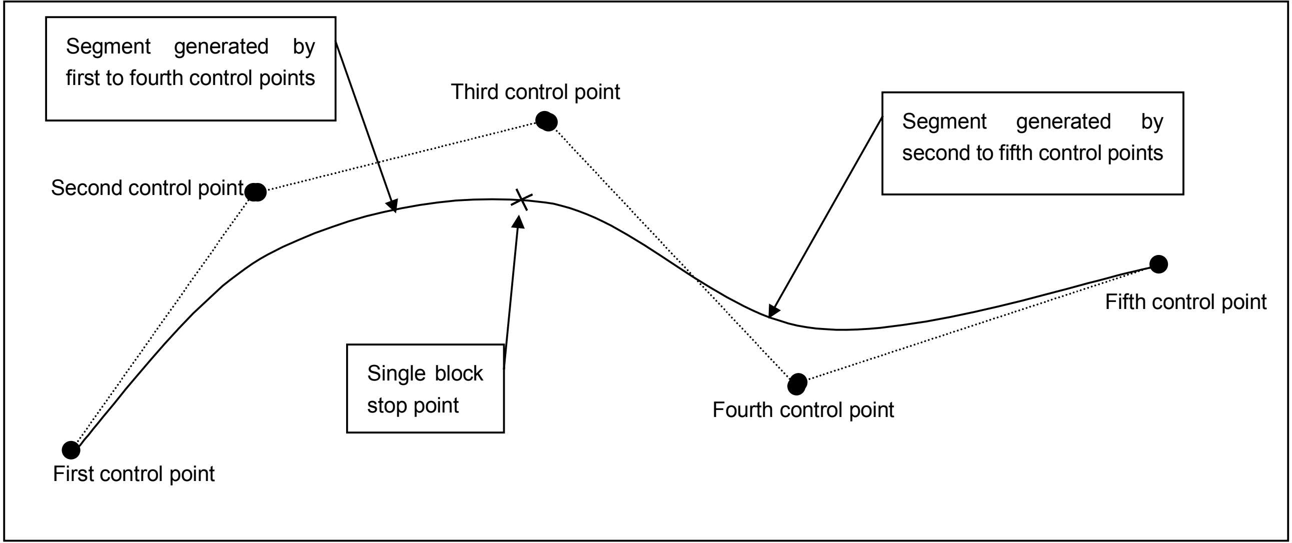


Fig.4.17 (c) Segments of an NURBS Curved Line of Rank 4 (Degree 3)

- Valid speed command range

An NUBRS curved line of rank n (degree (n-1)) that has m control points includes (m - n + 1) segments. The speed command (address F) for a block that ranges from the first control point to the (m-n+1)-th control point applies to each segment.

If a speed is specified in a block containing the j-th control point, for example, the specified speed applies to the NURBS segments starting from that NUBRS segment generated by the j-th control point to (j+n-1)-th control point.

Example) For an NUBRS curved line of rank 4 (degree 3)

```
G06.2  X_ Y_ K_ F1000 ;      (First control point)
       X_ Y_ K_ ;            (Second control point)
        :
       X_ Y_ K_ F2000 ;      (j-th control point)
       X_ Y_ K_ ;            ((j+1)-th control point)
       X_ Y_ K_ ;            ((j+2)-th control point)
       X_ Y_ K_ ;            ((j+3)-th control point)
        :
```

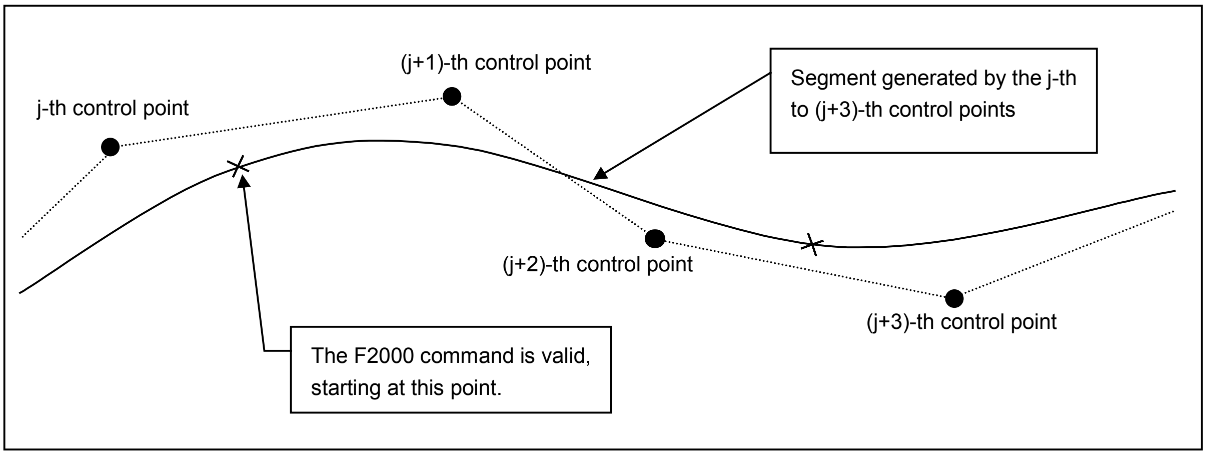

(j+1)-th control point
j-th control point
Segment generated by the j-th to (j+3)-th control points
(j+2)-th control point
(j+3)-th control point
The F2000 command is valid, starting at this point.

Fig.4.17 (d) Valid Speed Command Range

- Automatic speed control

During NUBRS interpolation, automatic acceleration/deceleration control is exercised according to the varying curvature of the NURBS curved line so that acceleration on each axis does not exceed the allowable acceleration limit specified by parameter No. 1663. (See Fig.4.17 (e))

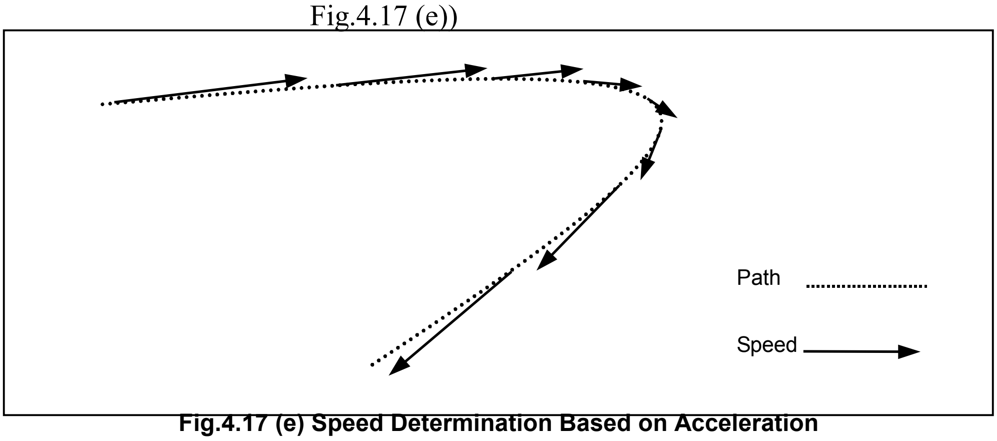


Fig.4.17 (e) Speed Determination Based on Acceleration

At a corner, automatic speed control is exercised so that speed changes on each axis do not exceed the allowable speed difference limit specified with parameter No. 1478. (See Fig.4.17 (f))

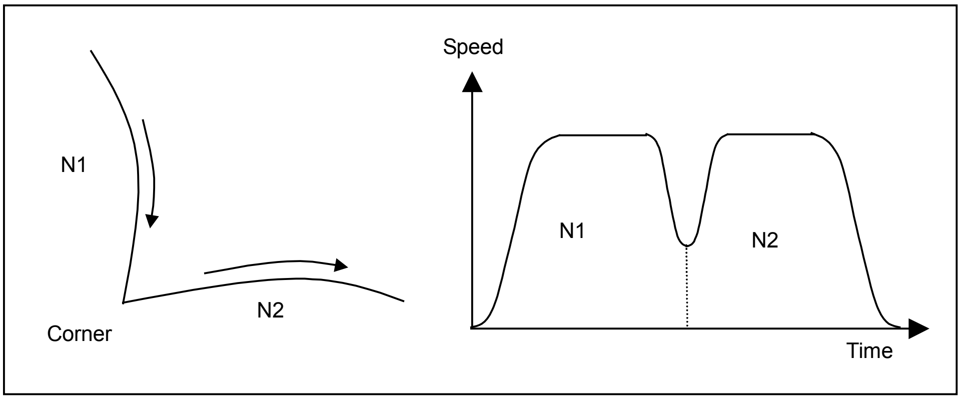


Fig.4.17 (f) Speed Determination Based on Corner Speed Differences

Limitation

- Control point

Each control point is specified with a combination of an axis name and coordinate word. No limit is imposed on the number of axes and axis name combinations subject to NURBS interpolation. For an axis subject to NURBS interpolation, however, a control point must be specified in the first block. If no control point is specified for an axis in the first block, the axis cannot be specified until the next NURBS curved line starts or NURBS interpolation mode ends. A control point can be specified either by absolute programming or by incremental programming. If the absolute coordinates differ from the first control point when NURBS interpolation starts, a PS alarm (No. 1009) is output. (If control points are specified using incremental programming, the first control point must be specified so that the amount of travel is 0 as in G06.2 X0 Y0 Z0 K_.)

- Command in NURBS interpolation mode

In NURBS interpolation mode, commands (such as auxiliary functions) other than NURBS interpolation cannot be specified.

- Manual intervention

If manual intervention is performed with manual absolute on, a PS alarm (No. 1010) is issued.

- Cutter compensation

NURBS interpolation cannot be used together with cutter compensation. Therefore, you must cancel cutter compensation before specifying NURBS interpolation. If NURBS interpolation is specified in cutter compensation mode, a PS alarm (No. 1006) is issued.

- Functions that must be specified before setting NURBS interpolation mode

When using a function such as tool length compensation, coordinate rotation, and scaling together with NURBS interpolation, specify the function before setting NURBS interpolation mode. Cancel these functions after NURBS interpolation mode ends.

Alarm

No.	Display message	Description
PS1001	NURBS interpolation error	A rank specification is incorrect.
PS1002	NURBS interpolation error	No knot is specified. (In NURBS interpolation mode, a block that is not related to NURBS interpolation is specified.)
PS1003	NURBS interpolation error	An axis for which no control point is specified in the first block is specified.
PS1004	NURBS interpolation error	The number of blocks containing only a knot is insufficient.
PS1005	NURBS interpolation error	NURBS interpolation mode was cancelled before the end of NURBS interpolation.
PS1006	NURBS interpolation error	In NURBS interpolation mode, a mode that can not be used together with NUBRS interpolation mode is specified.
PS1007	NURBS interpolation error	As many multi-knots as the rank value are not specified at the start and end points.
PS1008	NURBS interpolation error	The number of knots does not increase linearly.
PS1009	NURBS interpolation error	The specification of the first control point is incorrect. (There is no continuity with the previous block.)
PS1010	NURBS interpolation error	An attempt was made to restart NURBS interpolation after manual intervention was performed with manual absolute on.

Alarms of NURBS interpolation

In NURBS interpolation, the alarms listed in the above table may be issued. Of these alarms, those that may be caused by a programming error are described below, using examples.

- PS1001 WRONG NUMBER OF ORDERS

In NURBS interpolation, the number of orders, 2, 3, or 4, is specified after address P in the block of the first control point. If address P is not specified, 4-order NURBS interpolation is assumed. If the value of address P is other than 2, 3, or 4 in the block of the first control point, alarm PS1001 is issued.

```
O0001
G06.2 P5 X0 Y0 Z0 K0 (Alarm PS1001 is issued.)
```

- PS1002 NO KNOT SPECIFIED

A knot must be specified in each block of NURBS interpolation. If there is a block without address K, alarm PS1002 is issued.

```
O0002
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
X40. Y40. Z40. K1.
X50. Y50. Z50. K2.
X60. Y60. Z60.     (Alarm PS1002 is issued.)
X70. Y70. Z70. K4.
K5.
K5.
K5.
K5.
G01
```

In NURBS interpolation mode, a block irrelevant to NURBS interpolation cannot be specified. An irrelevant block is considered as being illegal, like a block without a knot, and alarm PS1002 is issued as described above. Note that the alarm is issued if such a block is specified between the end point of a NURBS curve and the start point of another NURBS curve. If an irrelevant block must be specified, NURBS interpolation mode must be ended beforehand by specifying G01.

```
O0003
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
F1000          (Alarm PS1002 is issued.)
X40. Y40. Z40. K1.
X50. Y50. Z50. K2.
K3.
K3.
K3.
K3.
G01
```

```
O0004
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
K1.
K1.
K1.
K1.
F1000          (Alarm PS1002 is issued.)
```

```
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
K1.
K1.
K1.
K1.
G01

O0005
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
K1.
K1.
K1.
K1.
G01 F1000           (No alarm is issued.)
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
K1.
K1.
K1.
K1.
G01
```

- PS1003 AXIS NOT SPECIFIED IN THE FIRST BLOCK

In NURBS interpolation, all the axes to be used must be specified in the first block. If an axis which is not specified in the first block is specified in the second or subsequent block, an alarm is issued.

```
O0006
G06.2 P4 X0. Z0. K0.(An X-Z NURBS curve is specified.)
X10. Z10. K0.
X20. Z20. K0.
X30. Y30. Z30. K0.  (Alarm PS1003 is issued because Y is
K1.                  specified.)
K1.
K1.
K1.
```

- PS1004 INSUFFICIENT SIMPLE KNOT BLOCKS

In NURBS interpolation, the end of a NURBS curve command is determined by detecting as many knot commands as the number of orders. If the system encounters a command specifying another mode before detecting as many knot commands as the number of orders, an alarm is raised.

```
O0007
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
K1.
K1.
K1.
G01   (Alarm PS1004 is issued because only three K
      commands are specified.)
```

- PS1005 NURBS INTERPOLATION MODE TURNED OFF BEFORE NURBS INTERPOLATION ENDS

In NURBS interpolation, the end of a NURBS curve command is judged by detecting as many knot commands as the number of orders. If the system encounters a command specifying another mode before detecting simple knot commands, an alarm is raised.

```
O0008
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
G01   (Alarm PS1005 is issued because G01 is specified
      before a simple K block is specified.)
```

- PS1006 MODE INCOMPATIBLE WITH NURBS INTERPOLATION

If G06.2 is specified in G41/G42, G93/G95, or G12.1 mode, an alarm is raised.

- PS1007 WRONG MULTIPLE KNOTS AT THE START POINT OR END POINT

NURBS interpolation requires as many knot commands as the number of orders at the start point and end point.

In the examples given below, the knot commands of N1 to N4 and of N7 to N10 must have identical values.

```
O0009
N1 G06.2 P4 X0. Y0. Z0. K0.
N2 X10. Y10. Z10. K0.
```

```
N3 X20. Y20. Z20. K0.
N4 X30. Y30. Z30. K0.           (No alarm is issued.)
N5 X40. Y40. Z40. K1.
N6 X50. Y50. Z50. K2.
N7 K3.
N8 K3.
N9 K3.
N10 K3.      (Alarm PS1007 is issued.)

O0010
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
X40. Y40. Z40. K1.
X50. Y50. Z50. K2.
K3.
K3.
K4.            (Alarm PS1007 is issued.)
K4.
```

- PS1008 WRONG KNOT INCREMENT

In NURBS interpolation, the knot value must increase monotonously. The number of multiple knots accepted at the start and end points is the same as the number of orders. At another point, up to the number of orders minus 1 is accepted. If the knot value decreases or if the number of consecutive multiple knots exceeds the limit, an alarm is issued.

```
O0011
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
X40. Y40. Z40. K1.
X50. Y50. Z50. K0.8  (Alarm PS1008 is issued because
K3.                  the knot value does not increase
K3.                  monotonously.)
K3.
K3.

O0012
G06.2 P4 X0. Y0. Z0. K0.
X10. Y10. Z10. K0.
X20. Y20. Z20. K0.
X30. Y30. Z30. K0.
X40. Y40. Z40. K1.
X50. Y50. Z50. K1.
X60. Y60. Z60. K1.
X70. Y70. Z70. K1.   (Alarm PS1008 is issued because
K2.                  too many knots are specified.)
K2.
```

```
K2.
K2.
```

- PS1009 WRONG FIRST CONTROL POINT

The first control point of NURBS interpolation must be the start point of a NURBS curve, which is the current position when the previous block ends.

```
O0013
G90 G01 X100. Y100. Z100. F1000
G06.2 P4 X100. Y100. Z100. K0.   (No alarm is issued.)
X110. Y110. Z110. K0.

O0014
G90 G01 X100. Y100. Z100. F1000
G91 G06.2 P4 X0. Y0. Z0. K0. (An incremental command
G90 X110. Y110. Z110. K0.     with zero travel does not
                              cause an alarm.)
O0015
G90 G01 X100. Y100. Z100. F1000
G06.2 P4 X100. Y100. Z120. K0.  (Alarm PS1009 is
X110. Y110. Z110. K0.            issued.)
```

Example

Sample <NURBS interpolation program>

```
G90;
…
G06.2  K0. X0. Z0. B-30.;
       K0. X300. Z100. B-10.;
       K0. X700. Z100. B10.;
       K0. X1300. Z-100. B10.;
       K0.5 X1700. Z-100. B-10.;
       K0.5 X2000. Z0. B-30.;
       K1.0;
       K1.0;
       K1.0;
       K1.0;
G01    Y0.5;
G06.2  K0. X2000. Z0. B-30.;
       K0. X1700. Z-100. B-10.;
       K0. X1300. Z-100. B10.;
       K0. X700. Z100. B10.;
       K0.5 X300. Z100. B-10.;
       K0.5 X0. Z0. B-30.;
       K1.0;
       K1.0;
       K1.0;
       K1.0;
G01    Y1.0;
G06.2 ...
```

```
...
G01 ...
```

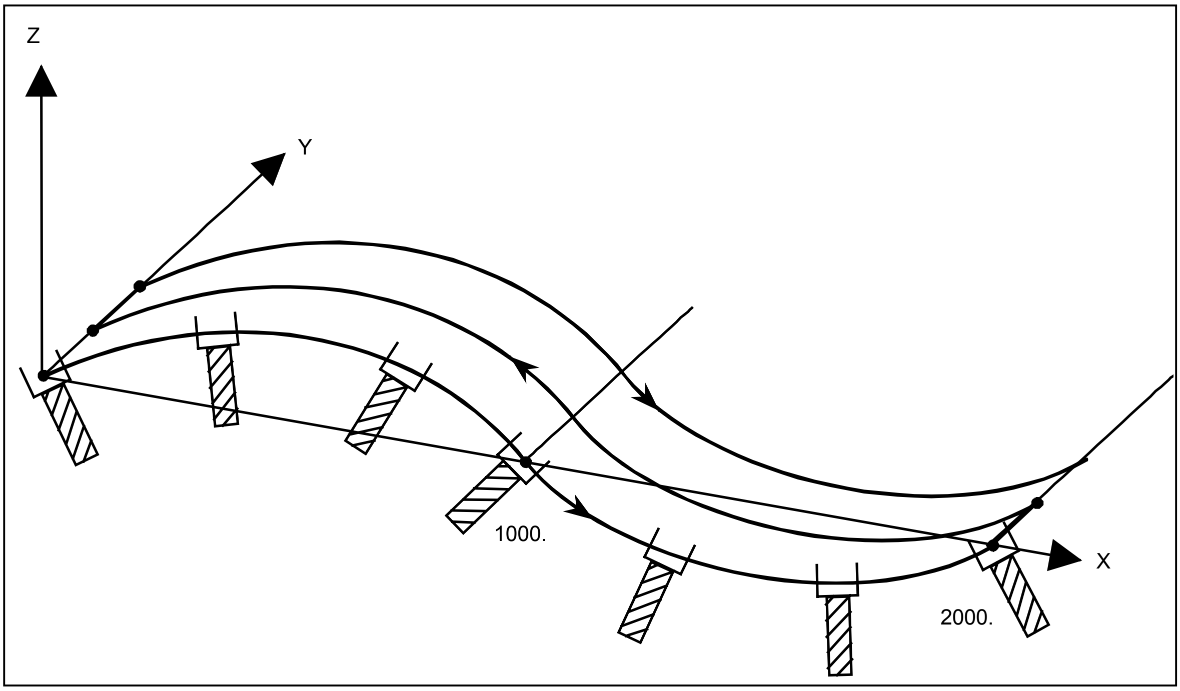


Fig.4.17 (g) Sample Program

4.17.1 NURBS Interpolation Additional Functions

The functions below are added to NURBS interpolation of the FANUC Series 15*i*.

- Parametric feedrate control

The maximum feedrate of each segment is determined by a specified feedrate and acceleration value. For successive segments, a feedrate at a segment start point and a feedrate at a segment end point are determined as described below. Then, the feedrate changes successively during movement from the start point to the end point. This function is applicable only to NURBS interpolation when bit 5 (FDI) of parameter No. 8412 is set to 1.

1. Start point feedrate
   (1) When NURBS interpolation is started
   When the travel distance of the previous block is 0, the maximum feedrate of the first segment is the start point feedrate. When the travel distance of the previous block is not 0, the smaller of the specified feedrate of the previous block and the maximum feedrate of the first segment is the start point feedrate.
   Previous block: Block immediately before a NURBS interpolation specification block
   (2) Second segment and up
   The smaller of the maximum feedrate of the previous segment and the maximum feedrate of this segment is the start point feedrate.
2. End point feedrate
   (1) When the segment is not the last segment
   The smaller of the maximum feedrate of this segment and the maximum feedrate of the next segment is the end point feedrate.
   (2) When NURBS interpolation is terminated
   When the travel distance of the next block is 0, the maximum feedrate of this segment is the end point feedrate. When the travel distance of the next block is not 0, the smaller of the maximum feedrate of this segment and the specified feedrate of the next block is the end point feedrate.

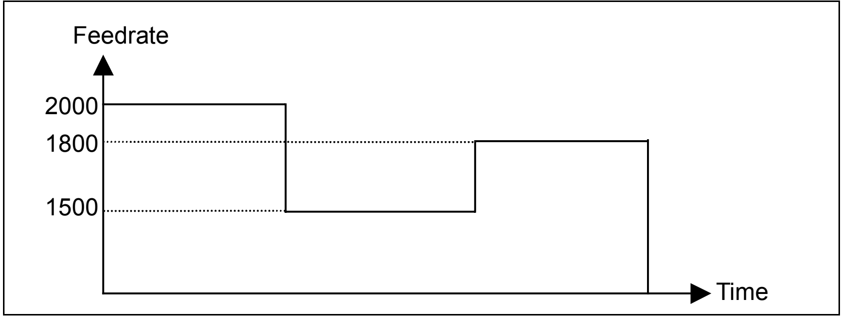
As the start point feedrate for restarting after a single block stop, the maximum feedrate of this segment is used.
As the feedrate for restarting after a feed hold stop, the feedrate present at the time of the feed hold stop is used.

Example:
1. Specified program

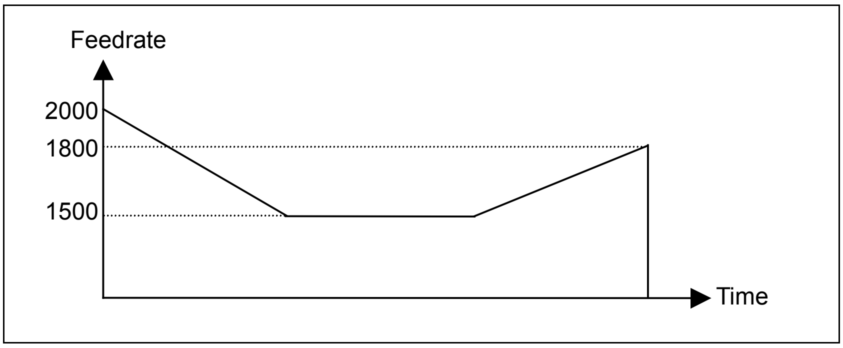
```
G90 G06.2 X0. Y0. K0. F2000 ;
X10. Y10. K0. F1500 ;
X20. Y20. K0. F1800 ;
X30. Y30. K0. ;
X40. Y40. K1.
```

```
X50. Y50. K2.
K3.
K3.
K3.
K3.
```

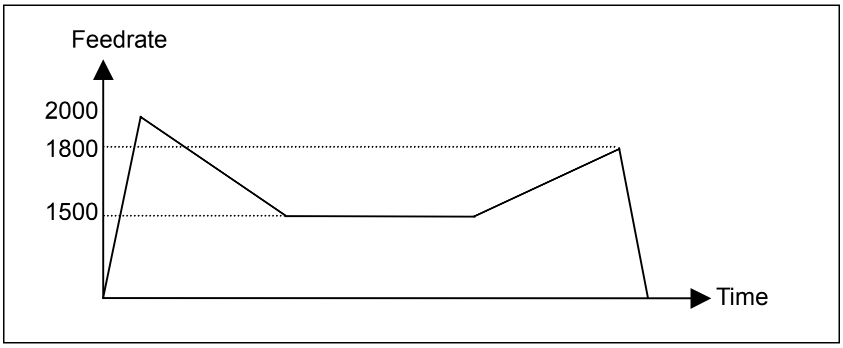
2. Specified feedrate



3. Parametric feedrate control



4. After acceleration/deceleration before interpolation, the actual feedrate is as follows:



NOTE

The actual acceleration/deceleration method is bell-shaped acceleration/deceleration.

- High-precision knot command

When bit 1 (HIK) of parameter No. 8412 is set to 1, a knot command consisting of up to 12 integer digits and up to 12 fraction digits can be specified. This function is applicable only to a knot command (address K) including a decimal point during NURBS interpolation.

When a high-precision knot command is used, the knot command format is as follows:

K (A-digit number).(B-digit number)

A command with the sum of A + B not exceeding 12 can be specified.
Specify a command not longer than 14 characters including address K and a decimal point.

```
(Example)
  Valid
    K.999999999999
    K1234.56789012
    K999999999999.
  Invalid (Alarm PS0012 is issued.)
    K0.999999999999
    K1234.5678901230
    K1234.5678901234
    K999999999999.0
```

- Simple start command

When bit 0 (EST) of parameter No. 8412 is set to 1, a control point command can be omitted at the first control point. The knot values of the first block and the second block are the same, so that the knot command can be omitted for the first block only.

The NURBS interpolation command format using this function is as follows:

```
G06.2 [P_ ] [K_ ] [IP_ ] [R_ ] [F_ ] ;
            K_  IP_  [R_ ] ;
            K_  IP_  [R_ ] ;
            K_  IP_  [R_ ] ;
             :
            K_  IP_  [R_ ] ;
            K_ ;
             :
            K_ ;
G01...
...
```

G06.2	: NURBS interpolation mode ON
P_	: NURBS curve rank
IP_	: Control point
R_	: Weight
K_	: Knot
F_	: Feedrate

The word enclosed in [ ] is omissible.

If a control point command is omitted in the first block, the current position where NURBS interpolation is started is the first control point. Only those axes that are specified at the control point of the second block are handled as NURBS interpolation axes. So, specify all symmetric axes for NURBS interpolation.

If a knot command is omitted in the first block, the same value as specified in the second block is assumed.

- Maximum cutting feedrate along each axis

With the conventional specification, the specified feedrate F during NURBS interpolation is clamped to the minimum value of the maximum cutting feedrate (parameter No. 1422) of each axis as indicated by the expression below.

$$F \le \mathrm{Min}(F_{\max}(X), F_{\max}(Y), F_{\max}(Z), F_{\max}(A), F_{\max}(B))$$

So, when the maximum cutting feedrate of a rotation axis $F_{\max}$ is small, the specified feedrate F during NURBS interpolation may be clamped to $F_{\max}$ of the rotation axis, resulting in an increase in machining time.

This function changes the method of clamping the specified feedrate F as described below.

The specified feedrate F is clamped so that the component of F along each axis does not exceed the maximum cutting feedrate (parameter No. 1422) of each corresponding axis (Fig. 4.17.1(a)).

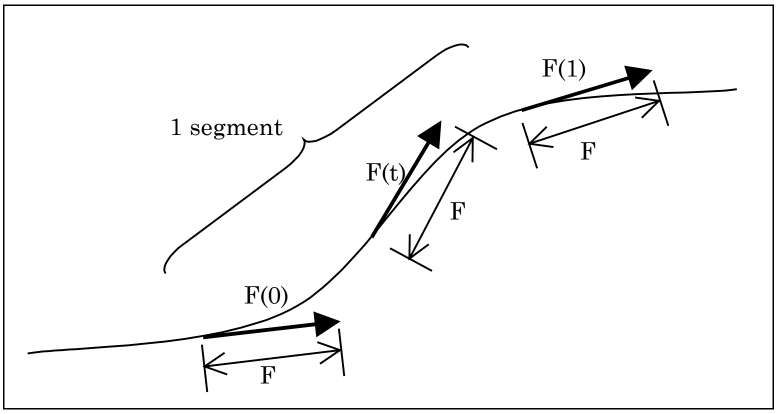


Fig. 4.17.1(a)

An example of NURBS interpolation using the five axes X, Y, Z, A, and B is explained below.

Here, the feedrate F is represented by vector F(t) (where t changes from 0 to 1 in one segment). Let $F_x(t)$, $F_y(t)$, $F_z(t)$, $F_a(t)$, and $F_b(t)$ be the components of the individual axes. Then, the following expression is given:

$$F(t) = (F_x(t), F_y(t), F_z(t), F_a(t), F_b(t)) \quad \text{(t = 0 to 1 in one segment)}$$

Then,

$$F = |F(t)|$$

At this time, F is clamped so that the following states are satisfied at all times in the segment:

$$F_x(t) \le F_{\max}(X)$$

$$F_y(t) \le F_{\max}(Y)$$

$$F_z(t) \le F_{\max}(Z)$$

$$F_a(t) \le F_{\max}(A)$$

$$F_b(t) \le F_{\max}(B)$$

In other words, the conditions below must be satisfied when the clamped feedrate F is multiplied by the proportion of each axis component (such as $\frac{F_x(t)}{|F(t)|}$):

$$\mathrm{Max}(\frac{F_x(t)}{|F(t)|}F) \le F_{\max}(X)$$

$$\mathrm{Max}(\frac{F_y(t)}{|F(t)|}F) \le F_{\max}(Y)$$

$$\mathrm{Max}(\frac{F_z(t)}{|F(t)|}F) \le F_{\max}(Z)$$

$$\mathrm{Max}(\frac{F_a(t)}{|F(t)|}F) \le F_{\max}(A)$$

$$\mathrm{Max}(\frac{F_b(t)}{|F(t)|}F) \le F_{\max}(B)$$

(t=0 to 1)

So, F is clamped as follows:

$$F \le \mathrm{Min}(\frac{F_{\max}(X)}{\mathrm{Max}(\frac{F_x(t)}{|F(t)|})}, \frac{F_{\max}(Y)}{\mathrm{Max}(\frac{F_y(t)}{|F(t)|})}, \frac{F_{\max}(Z)}{\mathrm{Max}(\frac{F_z(t)}{|F(t)|})}, \frac{F_{\max}(A)}{\mathrm{Max}(\frac{F_a(t)}{|F(t)|})}, \frac{F_{\max}(B)}{\mathrm{Max}(\frac{F_b(t)}{|F(t)|})})$$

(t=0 to 1)

Note) In the expression, the values of $F_x(t)$, $F_y(t)$, $F_z(t)$, $F_a(t)$, and $F_b(t)$ are assumed to be absolute positive values.

- Rollover

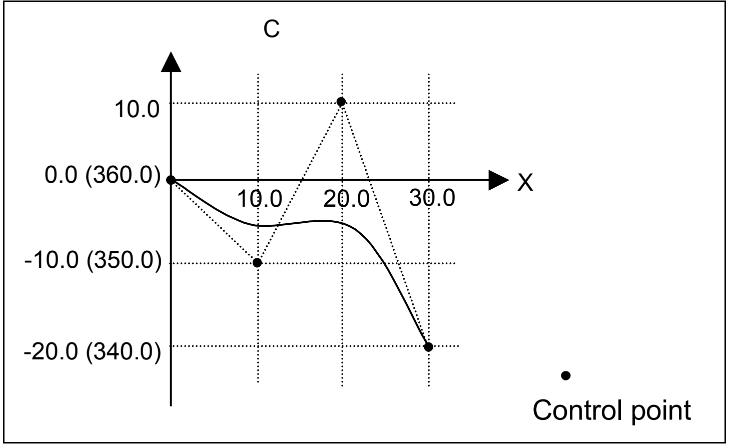
If a control point is specified in the absolute mode (G90) for a rotation axis subject to rollover, the relative position shift of the control point based on a shortcut is calculated after rollover processing for the control point.

Example: When the C-axis is a rotation axis subject to rollover

Program

```
G90 G06.2 P4 K0.0 X0.0 C0.0 F1000
K0.0 X10.0 C350.0
K0.0 X20.0 C10.0
K0.0 X30.0 C-20.0
K1.0
K1.0
K1.0
K1.0
G01
```

Actual operation



- Inverse time feed

If G93 is specified during NURBS interpolation, the inverse time command (G93) mode is set. Specify an inverse time (FRN) with F code. FRN for NURBS interpolation is represented by the following expression:

$$\mathrm{FRN} = \frac{\mathrm{Feedrate}}{\mathrm{Distance}}$$

Feedrate: mm/min (metric input) or inch/min (inch input)
Distance: mm (metric input) or inch (inch input)
(Travel distance along a NURBS curve. This distance does not always represent a travel distance if a rotation axis is involved.)

The format of F code is F4.3 (0.001 to 9999.999) as with the ordinary inverse time command. The valid data range of a feedrate command is the same as for a feedrate command in the feed per minute mode.

NOTE

The actual operation time may slightly vary from a specified time due to CNC internal processing.

4.18 3-DIMENSIONAL CIRCULAR INTERPOLATION (G02.4 AND G03.4)

General

Specifying an intermediate and end point on an arc enables circular interpolation in a 3-dimensional space.

Format

The command format is as follows:
G02.4 X_{X1} Y_{Y1} Z_{Z1} $\alpha_{\alpha1}$ $\beta_{\beta1}$; First block (mid-point of the arc)
X_{X1} Y_{Y1} Z_{Z1} $\alpha_{\alpha1}$ $\beta_{\beta1}$; Second block (end point of the arc)
Or,
G03.4 X_{X1} Y_{Y1} Z_{Z1} $\alpha_{\alpha1}$ $\beta_{\beta1}$; First block (mid-point of the arc)
X_{X1} Y_{Y1} Z_{Z1} $\alpha_{\alpha1}$ $\beta_{\beta1}$; Second block (end point of the arc)

α,β : Arbitrary axes other than the 3-dimensional circular interpolation axis (up to two axes)

Explanation

- G code group

G02.4 and G03.4 are modal G codes of group 01. They therefore remain effective until another G code in group 01 is specified.

- Start point, mid-point, and end point

An arc in a 3-dimensional space is uniquely defined with its start point (current position) and a specified intermediate point and end point, as shown below. Two command blocks are used to define this arc. The first command block specifies the tool path between the start point and intermediate point. The second command block specifies the tool path between the intermediate point and end point.

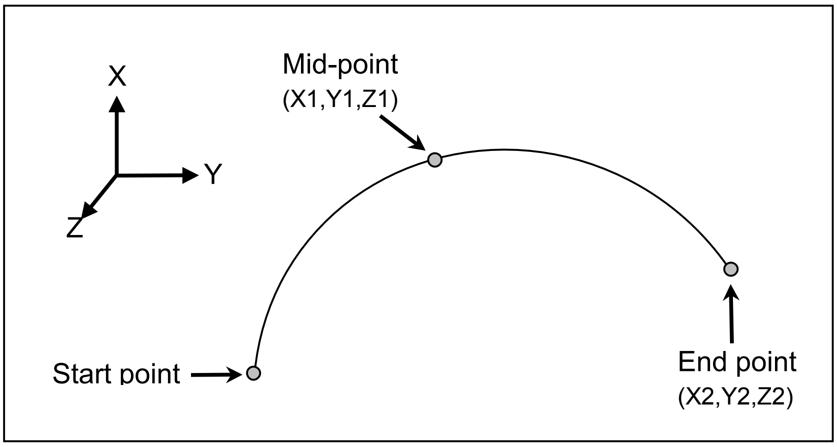


Fig. 4.18 Start, Mid, and End Points

If the modal code is changed by specifying a code such as G01 with the end point not specified, the arc cannot be obtained, and alarm PS0712 is issued. During MDI operation, alarm PS0712 is also issued if a cycle start is applied with only the mid-point specified.

- Movement along axes other than the 3-dimensional circular interpolation axis

In addition to the 3-dimensional circular interpolation axis (X/Y/Z), up to two arbitrary axes (/) can be specified at a time. If / are omitted from the first block (mid-point specification) and are specified only in the second block (end point specification), the tool moves to the specified point along the / axes during movement from the mid-point of the arc to the end point. If / are omitted from the second block (end point specification) and are specified only in the first block (mid-point specification), the tool moves to the specified point along the / axes during movement from the start point of the arc to the mid-point.

- Incremental commands

With an incremental command, the position of the mid-point relative to the start point must be specified in the first block, and the position of the end point relative to the mid-point must be specified in the second block.

- Direction of rotation

The direction of rotation cannot be specified. The movement is the same regardless of whether G02.4 or G03.4 is specified.

- Single block

When operation is performed using a single block, one cycle start causes movement from the start point to the end point. A single-block stop is not performed between the first block (mid-point specification) and the second block (end point specification).

- Start point assumed if 3-dimensional circular interpolations are specified consecutively

If 3-dimensional circular interpolations are specified consecutively, the end point in one interpolation is assumed to be the start point in the next interpolation.

- Velocity commands

Velocity commands follow HTG (bit 2 of parameter No. 1401), in the same as for helical interpolation. If HTG is equal to 0, specify the tangent velocity along an arc in the 3-dimensional space. If HTG is equal to 1, specify the tangent velocity including the velocities along the specified arbitrary axes other than the 3-dimensional circular interpolation axis.

Limitation

- Functions that cannot be specified

In the mode for this function, the following functions cannot be used:

- Functions related to cutter raiuds compensation (group 07)
- 3-dimensional coordinate conversion
- Coordinate system rotation
- Scaling
- Programmable mirror image
- Polar coordinate interpolation
- Polar coordinate command
- Normal direction control
- Hypothetical axis interpolation

- Cylindrical interpolation
- Drilling canned cycle/electronic gearbox
- Modal call
- Exact stop mode
- Automatic corner override
- Tapping mode
- Chopping
- Interrupt-type custom macro
- Unidirectional positioning
- Dwell
- Miscellaneous functions (except M00/M01/M02/M30)
- Data setting (G10)
- Reference position return function
- Skip function, multistage skip
- Tool length automatic measurement
- Macro call
- Linear copy
- Rotational copy
- Exact stop
- Machine coordinate system selection
- Workpiece coordinate system setting
- Workpiece coordinate system selection/workpiece
- coordinate system setting
- Workpiece coordinate system preset
- Local coordinate system preset
- Optional-angle chamfering/corner rounding

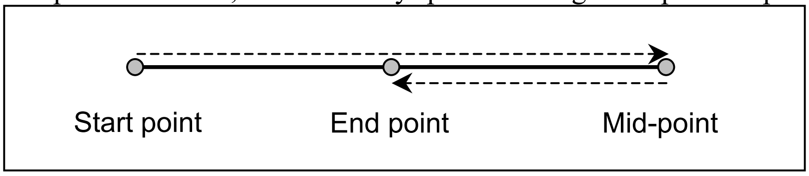
- Modes in which this function cannot be specified

This function cannot used in the modes of the following functions:

- Functions related to cutter radius compensation (group 07)
- 3-dimensional coordinate conversion
- Coordinate system rotation
- Scaling
- Programmable mirror image
- Polar coordinate interpolation
- Polar coordinate command
- Normal direction control
- Hypothetical axis interpolation
- Cylindrical interpolation
- Drilling canned cycle/electronic gearbox
- Modal call
- Exact stop mode
- Automatic corner override
- Tapping mode
- Chopping
- Interrupt-type custom macro
- Data setting (G10)

- Cases in which linear interpolation is performed

- If the start point, mid-point, and end-point are on the same line, linear interpolation is performed.
- If the start point coincides with the mid-point, the mid-point coincides with the end point, or the end point coincides with the start point, linear interpolation is performed up to the end point.
- If the start point, mid-point, and end-point are on the same line and the end point lies between the start point and the mid-point, the tool first moves with linear interpolation from the start point to the mid-point, then returns from the mid-point to the end point with linear interpolation. Thus, the tool always passes through the specified point.



- Whole circles

A whole circle (360 arc) cannot be specified. (This corresponds to the case in which linear interpolation is performed, as described earlier.)

- Compensation functions

Before using this function, cancel the compensation functions of group 07, such as cutter radius compensation.

- Manual absolute

While this function is in use, manual intervention is not possible with the manual absolute switch set to the ON position. If intervention is performed, alarm PS0713 is issued when operation restarts.

- Background graphic

This function cannot be used with background graphic drawing.

- Optional-angle chamfering/corner rounding functions

This function cannot be used together with optional-angle chamfering/corner rounding.

4.19 THREADING (G33)

The G33 command produces a straight or tapered thread having a constant lead.

Format

G33 IP_ F_ Q_ ;

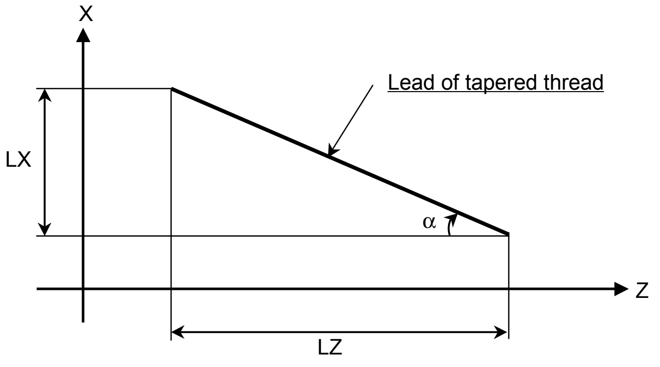
F_ : Larger component of lead

Q_ : Angle by which the threading start angle is shifted (0 to 360deg.)

Explanation

In general, thread cutting is repeated along the same tool path in rough cutting through finish cutting for a screw. Since thread cutting starts when the position coder mounted on the spindle outputs a 1-turn signal, threading is started at a fixed point and the tool path on the workpiece is unchanged for repeated thread cutting. Note that the spindle speed must remain constant from rough cutting through finish cutting. If not, incorrect thread lead will occur.

When a tapered thread is produced, the lead must be specified with the magnitude of a larger component. A lathe which holds and rotates a workpiece can produce a tapered thread on the workpiece.



When angle α is less than or equal to 45°, specify LZ.
When angle α is greater than or equal to 45°, specify LX.

Fig.4.19 (a) Lead Position

In general, the lag of the servo system, etc. will produce somewhat incorrect leads at the starting and ending points of a thread cut. To compensate for this, a thread cutting length somewhat longer than required should be specified.
Table 4.19 (a) lists the ranges for specifying the thread lead.

Table4.19 (a) Ranges of lead sizes that can be specified

	Least command increment	Command value range of the lead
mm input	0.01 mm	0.001 to 5000.0000 mm/rev
	0.001 mm	0.00001 to 500.00000 mm/rev
	0.0001 mm	0.000001 to 50.000000 mm/rev
	0.00001 mm	0.0000001 to 5.0000000 mm/rev
	0.000001 mm	0.00000001 to 0.50000000 mm/rev
Inch input	0.001 inch	0.00001 to 500.00000 inch/rev
	0.0001 inch	0.000001 to 50.00000 inch/rev
	0,00001 inch	0.0000001 to 5.0000000 inch/rev
	0.000001 inch	0.00000001 to 0.50000000 inch/rev

NOTE

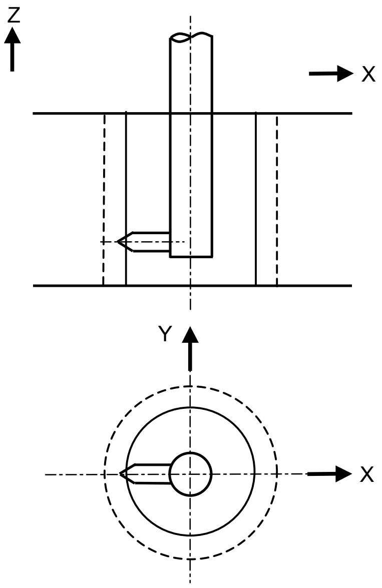
1 The spindle speed is limited as follows :
1 ≤ spindle speed ≤ (Maximum feedrate) /(Thread lead)
Spindle speed : min^{-1}
Thread lead : mm or inch
Maximum feedrate : mm/min or inch/min ; maximum command-specified feedrate for feed-per-minute mode or maximum feedrate that is determined based on mechanical restrictions including those related to motors, whichever is smaller

2 Cutting feedrate override is not applied to the converted feedrate in all machining process from rough cutting to finish cutting. The feedrate is fixed at 100%

3 The converted feedrate is limited by the upper feedrate specified.

4 Feed hold is disabled during threading. Pressing the feed hold key during thread cutting causes the machine to stop at the end point of the next block after threading (that is, after the G33 mode is terminated)

Example



```
Program
 N20 G90 G00 X100.0Y…S45 M3;
 N21 Z200.0;
 N22 G33 Z120.0 F5.0;
 N23 M19;
 N24 G00 X105.0;
 N25 Z200.0 M0;
 N26 X100.0 M3;
 N27 G04 X2.0;
 N28 G33 Z120.0 F5.0;
  :
```

N20,N21 The center of the tool is aligned with the center of a prepared hole. The spindle rotates in the forward direction.

N22 The first step of threading is started. The lead is specified by address F.

N23 The M19 command stops the spindle at a specific position on the circumference as shown in the figure above. (M19: Spindle stop at a specific position)

N24 The tool is moved away along the X-axis.

N25 The tool is lifted to a position above the hole. The M00 command stops the program. Meanwhile, the operator can adjust a tool for the second threading operation.

N26 The center of the tool is aligned with the center of the prepared hole. The spindle rotates in the forward direction.

N27 As the move command in the N26 block is short, this block specifies the dwell command. The system waits until the spindle speed becomes constant.

N28 The second threading operation is started.

Fig.4.19 (b) Threading

4.20 INCH THREADING (G33)

When a number of thread ridges per inch is specified with address E, an inch thread can be produced with high precision.

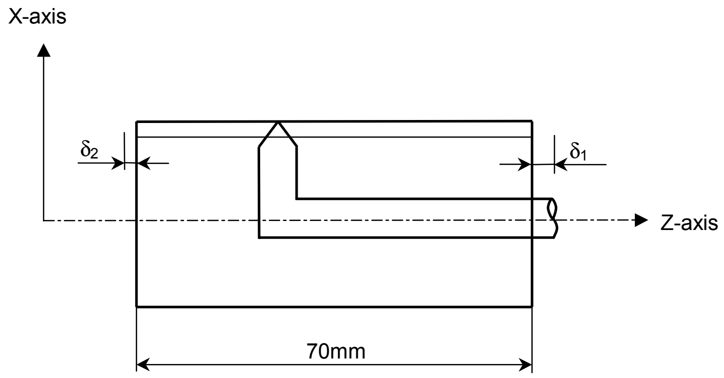
Format

G33 IP_ E_ Q_;

E_ : Number of thread ridges per inch

Q_ : Number of thread ridges per inch at threading start angle

Example



Lead : Three ridges per inch (= 8.4666……)

δ_1 = 3mm

δ_2 = 1.5mm

Depth of cut : 1 mm (Two cuts are made.)

The program is specified with the conditions above.

(Data is input in millimeters and the diameter is specified.)

```
G91  G33  Z-74.5 E3.0 ;  (Three ridges per inch)
     G00  X-10.0 ;
          Z74.5 ;
          X-11.0 ;       (A cut of another 1mm is made during
                         the second pass.)
     G33  Z-74.5 E3.0;
     G00  X-10.0 ;
          Z74.5 ;
```

Fig.4.20 (a) Inch threading

4.21 CONTINUOUS THREADING (G33)

Continuous threading can be executed when multiple blocks containing the threading command are specified in succession.

Explanation

At the interface between blocks, the system keeps synchronous control of the spindle as much as possible. The lead or profile of a thread can be changed in the middle of threading.

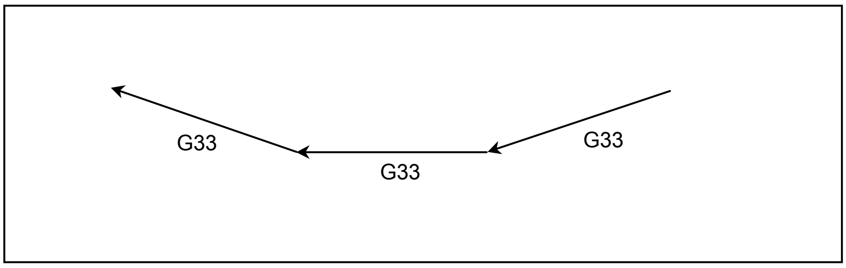


Fig.4.21 (a) Continuous threading

Repeating the threading operations along an identical path with a different depth of cut enables the thread to be produced correctly.

- Threading start angle

The threading start angle can be shifted only in the block in which the first threading operation is started.

5 FEED FUNCTIONS

5.1 GENERAL

The feed functions control the feedrate of the tool. The following two feed functions are available:

- Feed functions

1. Rapid traverse
   When the positioning command (G00) is specified, the tool moves at a rapid traverse feedrate set in the CNC (parameter No. 1420).
2. Cutting feed
   The tool moves at a programmed cutting feedrate.

- Override

Override can be applied to a rapid traverse rate or cutting feedrate using the switch on the machine operator's panel.

- Automatic acceleration/deceleration

To prevent a mechanical shock, acceleration/deceleration is automatically applied when the tool starts and ends its movement (Fig.5.1(a)).

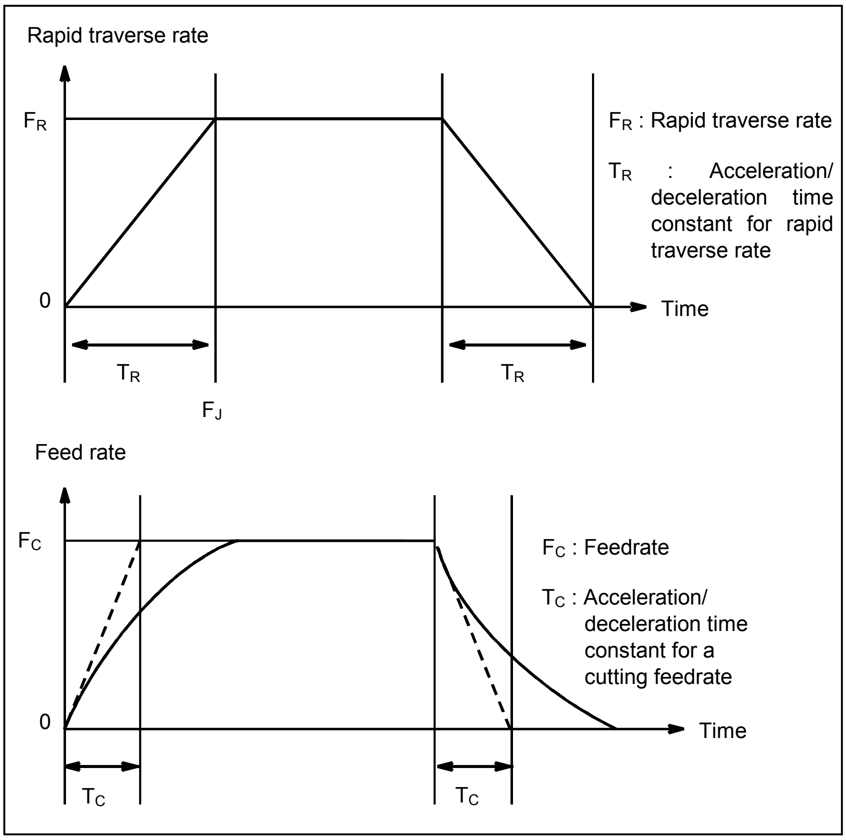


Fig.5.1(a) Automatic acceleration/deceleration (example)

- Tool path in a cutting feed

If the direction of movement changes between specified blocks during cutting feed, a rounded-corner path may result (Fig.5.1(b)).

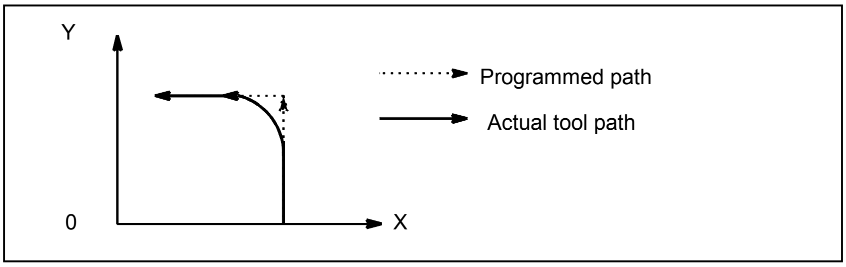


Fig.5.1(b) Example of Tool Path between Two Blocks

In circular interpolation, a radial error occurs (Fig.5.1(c)).

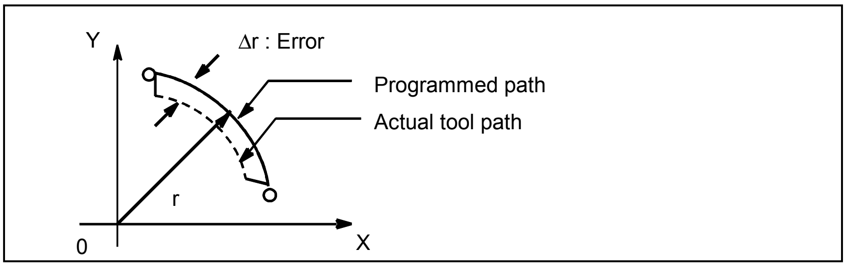


Fig.5.1(c) Example of Radial Error in Circular Interpolation

The rounded-corner path shown in Fig.5.1(b) and the error shown in Fig.5.1(c) depend on the feedrate. So, the feedrate needs to be controlled for the tool to move as programmed.

5.2 RAPID TRAVERSE

Format

G00 IP_;

G00 : G code (group 01) for positioning (rapid traverse)
IP ; Dimension word for the end point

Explanation

The positioning command (G00) positions the tool by rapid traverse. In rapid traverse, the next block is executed after the specified feedrate becomes 0 and the servo motor reaches a certain range set by the machine tool builder (in-position check).
A rapid traverse rate is set for each axis by parameter No. 1420, so no rapid traverse feedrate need be programmed.
The following overrides can be applied to a rapid traverse rate with the switch on the machine operator's panel (See II-5.4.2)
For detailed information, refer to the appropriate manual of the machine tool builder.

NOTE
When the IS-B increment system is used, a feedrate set in the parameter has a maximum error 8mm/min.

5.3 CUTTING FEED

Feedrate of linear interpolation (G01), circular interpolation (G02, G03), etc. are commanded with numbers after the F code. In cutting feed, the next block is executed so that the feedrate change from the previous block is minimized.

Four modes of specification are available:

1. Feed per minute (G94)
   After F, specify the amount of feed of the tool per minute.
2. Feed per revolution (G95)
   After F, specify the amount of feed of the tool per spindle revolution.
3. Inverse time feed (G93)
   Specify the inverse time (FRN) after F.
4. F1-digit feed
   Specify a desired one-digit number after F. Then, the feedrate set with the CNC for that number is set.

Format

```
Feed per minute
        G94 ; G code (group 05) for feed per minute
        F_;     Feedrate command (mm/min or inch/min)
Feed per revolution
        G95 ;  G code (group 05) for feed per revolution
        F_ ;    Feedrate command (mm/rev or inch/rev)
Inverse time feed
        G93 ;  Inverse time feed command
                G code (05 group)
        F_ ;    Feedrate command (1/min)
F1-digit feed
        Fn ;
        n : Number from 1 to 9
```

Explanation

- Tangential speed constant control

Cutting feed is controlled so that the tangential feedrate is always set at a specified feedrate.

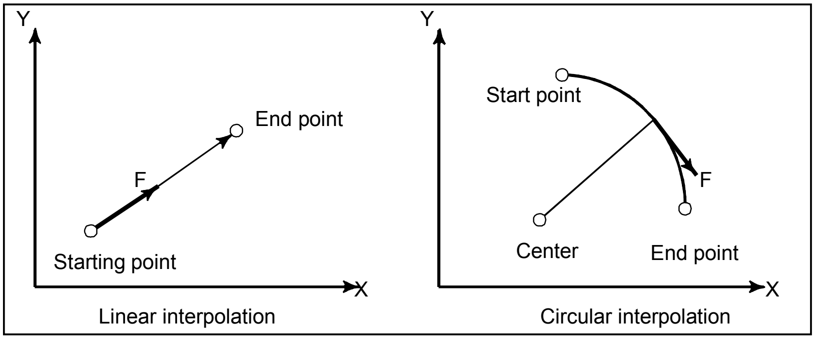


Fig.5.3 (a) Tangential feedrate (F)

- Feed per minute (G94)

After specifying G94 (in the feed per minute mode), the amount of feed of the tool per minute is to be directly specified by setting a number after F. G94 is a modal code. Once a G94 is specified, it is valid until G95 (feed per revolution) is specified. At power-on, the feed per minute mode is set.

An override can be applied to feed per minute with the switch on the machine operator's panel. (See II-5.3.1)

For detailed information, see the appropriate manual of the machine tool builder.

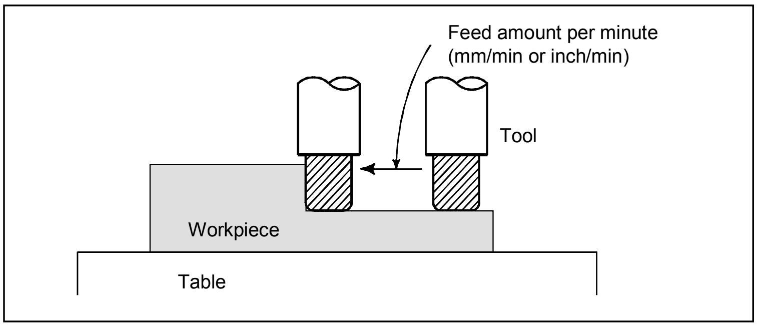


Fig.5.3 (b) Feed per minute

> ⚠ **CAUTION**
> No override can be used for some commands such as for threading.

Unit of F value

(1) With no decimal point(*1)

Metric input	**Linear axis**	1 mm/min(0.1 mm/min) *2
	Rotation axis	1 deg/min(0.1 deg /min) *2
Inch input	**Linear axis**	0.01 inch/min
	Rotation axis	0.01 deg/min

> **NOTE**
> *1 If bit 0 (DPI) of parameter No. 2400 is set to 1, calculator-type decimal point input is enabled.
> *2 If bit 1 (F41) of parameter No. 2400 is set to 1, the units in parentheses are used. (For increment system IS-A, this setting is invalid.)

(2) With a decimal point

The position of the decimal point represents the position of "mm," "deg," or "inch."

- Feed per revolution (G95)

After specifying G95 (in the feed per revolution mode), the amount of feed of the tool per spindle revolution is to be directly specified by setting a number after F. G95 is a modal code. Once a G95 is specified, it is valid until G94 (feed per minute) is specified. An override can be applied to feed per revolution with the switch on the machine operator's panel. (See II-5.4.1)

For detailed information, see the appropriate manual of the machine tool builder.

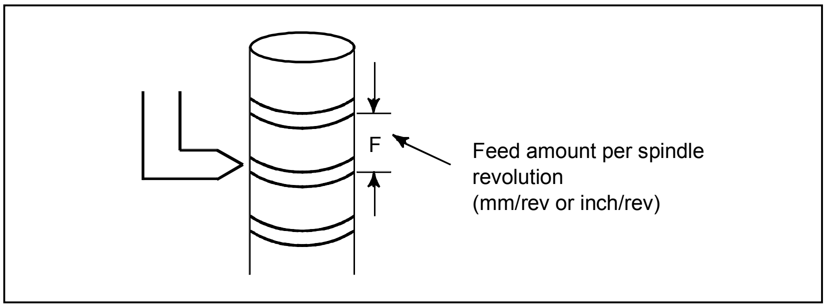


Fig.5.3 (c) Feed per revolution

> ⚠ **CAUTION**
> When the speed of the spindle is low, feedrate fluctuation may occur. The slower the spindle rotates, the more frequently feedrate fluctuation occurs.

Unit of F value

(1) With no decimal point(*1)

Metric input	**Linear axis**	0.01 mm/rev
	Rotation axis	0.01 deg/rev
Inch input	**Linear axis**	0.0001 inch/rev
	Rotation axis	0.0001 deg/rev

> **NOTE**
> *1 If bit 0 (DPI) of parameter No. 2400 is set to 1, calculator-type decimal point input is enabled.

(2) With a decimal point

The position of the decimal point represents the position of "mm," "deg," or "inch."

- Inverse time feed (G93)

When G93 is specified, the inverse time specification mode (G93 mode) is set. Specify the inverse time (FRN) with an F code.

A value from 0.001 to 9999.999 can be specified as FRN, regardless of whether the input mode is inches or metric, or the increment system is IS-A, IS-B, IS-C IS-D, or IS-E.

F code specification value	FRN
F1	0.001
F1 *1	1.000
F1.0	1.000
F9999999	9999.999
F9999 *1	9999.000
F9999.999	9999.999

> **NOTE**
> *1 Value specified in fixed-point format with bit 0 (DPI) of parameter No. 2400 set to 1

Explanation

- For linear interpolation (G01)

$$\mathrm{FRN} = \frac{1}{\mathrm{time\,(min)}} = \frac{\mathrm{feedrate}}{\mathrm{dis\,tan\,ce}}$$

Feedrate: mm/min (for metric input)
inch/min (for inch input)
Distance: mm (for metric input)
inch (for inch input)

To end a block in 1 (min)

$$\mathrm{FRN} = \frac{1}{\mathrm{time\,(min)}} = \frac{1}{1(\mathrm{min})} = 1 \text{ Specify F1.0.}$$

To end a block in 10 (sec)

$$\mathrm{FRN} = \frac{1}{\mathrm{time\,(sec)/60}} = \frac{1}{10/60(\mathrm{sec})} = 6 \text{ Specify F6.0.}$$

To find the movement time required when F0.5 is specified

$$\mathrm{Time(min)} = \frac{1}{\mathrm{FRN}} = \frac{1}{0.5} = 2 \text{ 2 (min) is required.}$$

To find the movement time required when F10.0 is specified

$$\mathrm{Time(min)} = \frac{1\times 60}{\mathrm{FRN}} = \frac{60}{10} = 6 \text{ 6(sec) is required.}$$

- For circular interpolation (G01)

$$\mathrm{FRN} = \frac{1}{\mathrm{time\,(min)}} = \frac{\mathrm{feedrate}}{\mathrm{arcradius}}$$

Feedrate: mm/min (for metric input)
inch/min (for inch input)
Arc radius: mm (for metric input)
inch (for inch input)

NOTE
In the case of circular interpolation, the feedrate is calculated not from the actual amount of movement in the block but from the arc radius.

G93 is a modal G code and belongs to group 05 (includes G95 (feed per revolution) and G94 (feed per minute)).
When an F value is specified in G93 mode and the feedrate exceeds the maximum cutting feedrate, the feedrate is clamped to the maximum cutting feedrate.
In the case of circular interpolation, the feedrate is calculated not from the actual amount of movement in the block but from the arc radius. This means that actual machining time is longer when the arc radius is longer than the arc distance and shorter when the arc radius is shorter

than the arc distance. Inverse time feed can also be used for cutting feed in a canned cycle.

⚠ **CAUTION**

When the cutter compensation function is used, actual movement is made after compensation is applied for a programmed command. As a result, the actual feedrate may differ from a specified feedrate.

NOTE

1 In the G93 mode, an F code is not handled as a modal code and therefore needs to be specified in each block. If an F code is not specified, P/S alarm (No.202) is issued.

2 When F0 is specified in G93 mode, the maximum cutting speed is used.

3 Inverse time feed cannot be used when PMC axis control is in effect.

- One-digit F code feed

When a one-digit number from 1 to 9 is specified after F, the feedrate set for that number in a parameter (Nos. 1451 to 1459) is used. When F0 is specified, the rapid traverse rate is applied.

The feedrate corresponding to the number currently selected can be increased or decreased by turning on the switch for changing F1-digit feedrate on the machine operator's panel, then by rotating the manual pulse generator.

The increment/decrement, DF, in feedrate per scale of the manual pulse generator is as follows:

$$\Delta = \frac{F\,\mathrm{max}}{100X}$$

Fmax : Feedrate upper limit for F1-F4 set by parameter (No.1460), or feedrate upper limit for F5-F9 set by parameter (No.1461)

X : any value of 1-127 set by parameter (No.1450)

The feedrate set or altered is kept even while the power is off. The current feed rate is displayed on the CRT screen.

When more than one handle is provided, the first handle is always used.

- Cutting feedrate clamp

A common upper limit can be set on the cutting feedrate along each axis with parameter No. 1422. If an actual cutting feedrate (with an override applied) exceeds a specified upper limit, it is clamped to the upper limit.

When the cutting feedrate along an axis exceeds the maximum feedrate for the axis as a result of interpolation, the cutting feedrate is clamped to the maximum feedrate.

⚠ **CAUTION**

An upper limit is set in mm/min or inch/min.
CNC calculation may involve a feedrate error of +2% with respect to a specified value. However, this is not true for acceleration/deceleration.

- Setting input of cutting feedrate

With some machines, the cutting feedrate need not be changed frequently during machining. For such machines, a cutting feedrate (a non-zero value) can be set in parameter No. 1493. With this function, the cutting feedrate (F code) need not be specified in the NC command data.

Even when a cutting feedrate is specified by setting input, a cutting feedrate command specified in the NC command data is effective. The specified cutting feedrate is maintained until another setting feedrate is specified or the power is turned off.

When a cutting feedrate is specified by setting input, an F code is not cleared by a reset regardless of the setting of bit 7 (NCM) of parameter No. 2401.

Reference

See Appendix C for range of feedrate command value.

5.4 OVERRIDE

The rapid traverse rate and cutting feedrate can be overridden from the machine operator's panel.

5.4.1 Feedrate Override

A programmed feedrate can be reduced or increased by a percentage (%) selected by the override dial. This feature is used to check a program.
For example, when a feedrate of 100 mm/min is specified in the program, setting the override dial to 50% moves the tool at 50 mm/min.

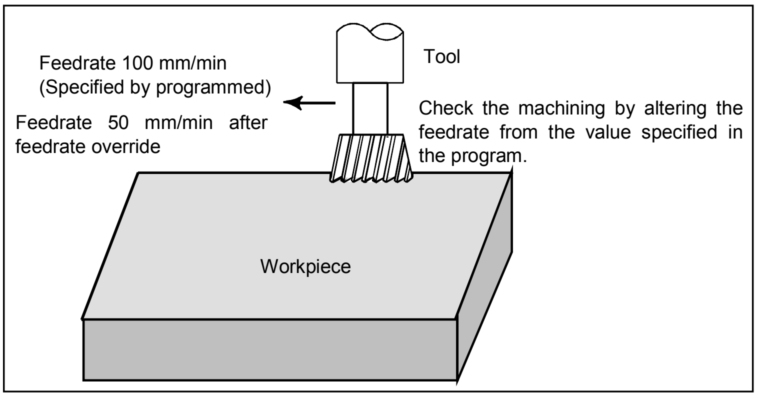
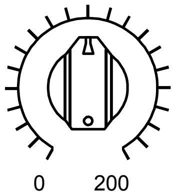

Feedrate 100 mm/min
(Specified by programmed)
Feedrate 50 mm/min after feedrate override
Tool
Check the machining by altering the feedrate from the value specified in the program.
Workpiece

Fig. 5.4.1 (a) Feedrate override



JOG FEED RATE OVERRIDE

Set the feedrate override dial to the desired percentage (%) on the machine operator's panel, before or during automatic operation.
On some machines, the same dial is used for the feedrate override dial and jog feedrate dial. Refer to the appropriate manual provided by the machine tool builder for feedrate override.

Restrictions

- Override Range

The override that can be specified ranges from 0 to 254%. (When the second feedrate override is used, an additional override of 0% to 254% (0% to 655.34% when the second feedrate override B is used) is applied.
For individual machines, the range depends on the specifications of the machine tool builder.

- Override during thread cutting or tapping

During threading or tapping, the override is ignored and the feedrate remains as specified to 100%.

- Override Cancel

When an override cancel switch is provided on the machine operator's panel, the feedrate override (together with the second feedrate override) can be clamped to 100%.

5.4.2 Rapid Traverse Override

The rapid traverse rate can be overridden as follows:

F0, F1%, 50%, 100%

F0 : Feedrate to be set for each axis (parameter No. 1421)

F1 : Percentage (parameter No. 1412)

or,0% to 100% (in steps of 1%) by setting bit 0 (ROV) of parameter No. 1402

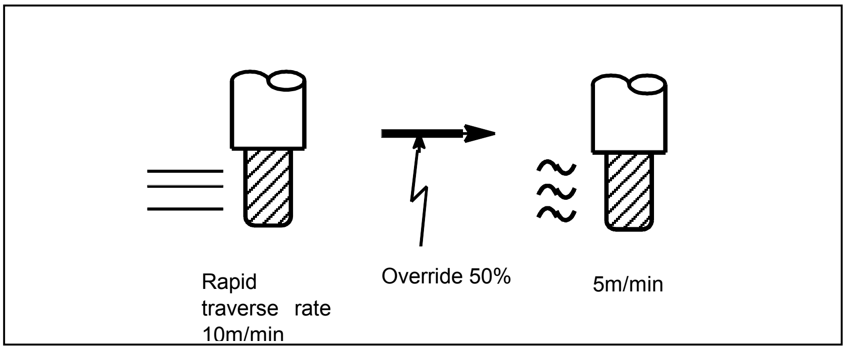
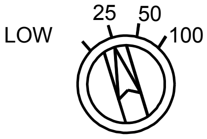


Fig. 5.4.2 (a) Rapid traverse override



Rapid traverse override

During rapid traverse, a rapid traverse override of 0% to 100% can be applied. Select a desired feedrate with the override switch.
Refer to the appropriate manual provided by the machine tool builder for rapid traverse override.

Explanation

The following types of rapid traverse are available. Rapid traverse override can be applied for each of them.
1) Rapid traverse by G00
2) Rapid traverse during a canned cycle
3) Rapid traverse in G27, G28, G29, G30, G53
4) Manual rapid traverse
5) Rapid traverse of manual reference position

5.5 CUTTING FEEDRATE CONTROL

Cutting feedrate can be controlled, as indicated in Table 5.5 (a).

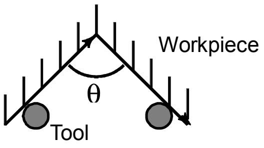
Table 5.5 (a) Cutting Feedrate Control

Function name		G code	Validity of G code	Description
Exact stop		G09	This function is valid for specified blocks only.	The tool is decelerated at the end point of a block, then an in-position check is made. Then the next block is executed.
Exact stop mode		G61	Once specified, this function is valid until G62, G63, or G64 is specified.	The tool is decelerated at the end point of a block, then an in-position check is made. Then the next block is executed.
Cutting mode		G64	Once specified, this function is valid until G61, G62, or G63 is specified.	The tool is not decelerated at the end point of a block, but the next block is executed.
Tapping mode		G63	Once specified, this function is valid until G61, G62, or G64 is specified.	The tool is not decelerated at the end point of a block, but the next block is executed. When G63 is specified, feedrate override and feed hold are invalid.
Automatic corner override	Automatic override for inner corners	G62	Once specified, this function is valid until G61, G63, or G64 is specified.	When the tool moves along an inner corner during cutter compensation, override is applied to the cutting feedrate to suppress the amount of cutting per unit of time so that a good surface finish can be produced.
	Circular cutting feedrate change	—	This function is valid in the cutter compensation mode, regardless of the G code.	The circular cutting feedrate is changed.

NOTE

1 The purpose of in-position check is to check that the servo motor has reached within a specified range (specified with a parameter by the machine tool builder). In-position check is not performed when bit 1 (CSZ) of parameter No. 1000 is set to 1.

2 Inner corner angle θ : $2° < \theta \leq \alpha \leq 178°$
(α is a set value)



Format

Exact stop	G09 IP_;
Exact stop mode	G61;
Cutting mode	G64;
Tapping mode	G63;
Automatic corner override	G62;

5.5.1 Exact Stop (G09, G61)Cutting Mode (G64)Tapping Mode (G63)

Explanation

The inter-block paths followed by the tool in the exact stop mode, cutting mode, and tapping mode are different (Fig.5.5.1(a)).

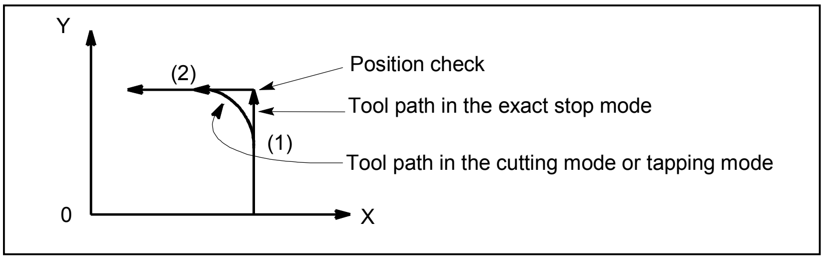

Y
(2)
Position check
Tool path in the exact stop mode
(1)
Tool path in the cutting mode or tapping mode
0
X

Fig.5.5.1 (a) Example of Tool Paths from Block (1) to Block (2)

⚠ **CAUTION**
The cutting mode (G64 mode) is set at power-on or system clear.

5.5.2 Automatic Corner Override

When cutter compensation is performed, the movement of the tool is automatically decelerated at an inner corner and internal circular area. This reduces the load on the cutter and produces a smoothly machined surface.

5.5.2.1 Automatic override for inner corners (G62)

Explanation

- Override condition

When G62 is specified, and the tool path with cutter compensation applied forms an inner corner, the feedrate is automatically overridden at both ends of the corner.
There are four types of inner corners (Fig.5.5.2 (a)).
2° ≤ θ ≤ θ p ≤ 178° in Fig. 5.4.2.1 (a)
θp is a value set with parameter No. 6611. When θ is approximately equal to θp, the inner corner is determined with an error of 0.001°,or less.

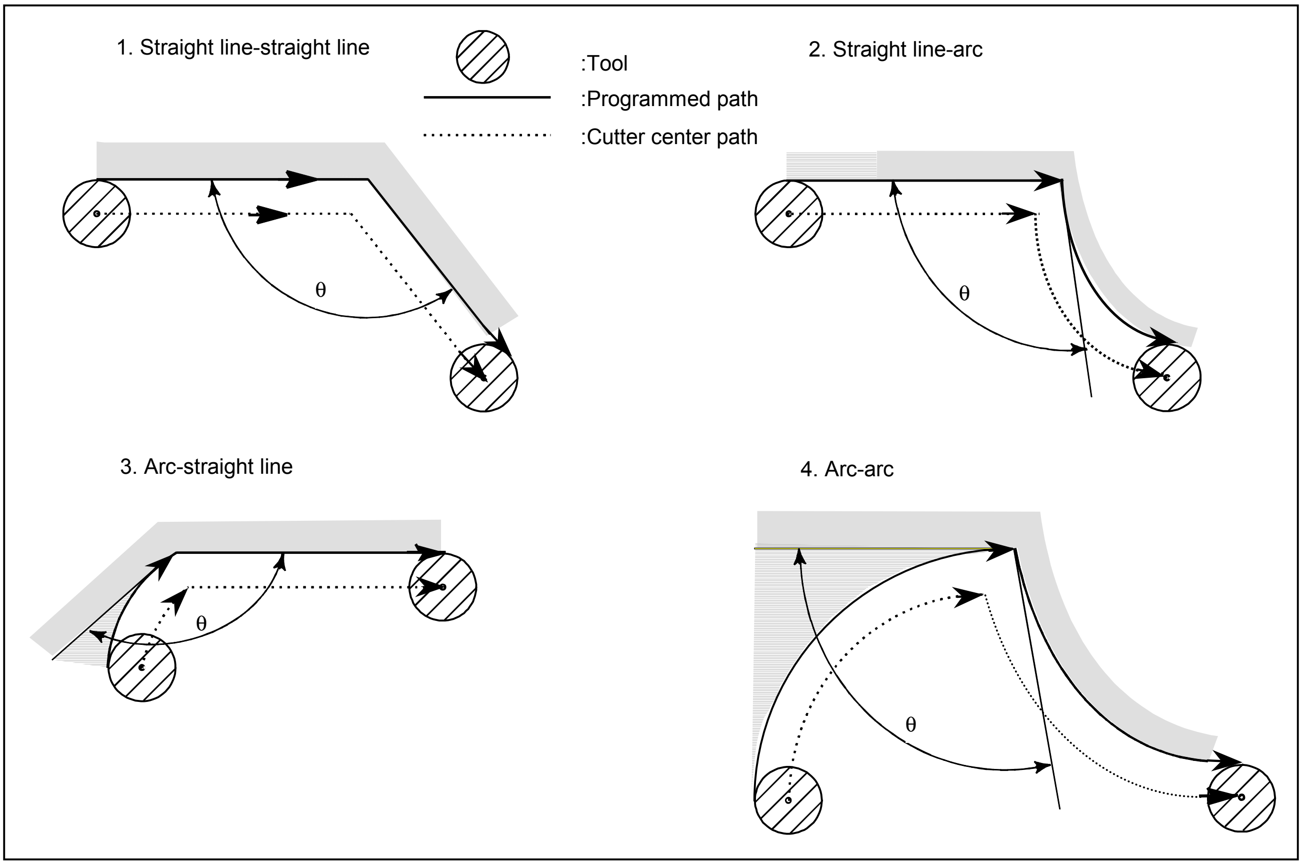
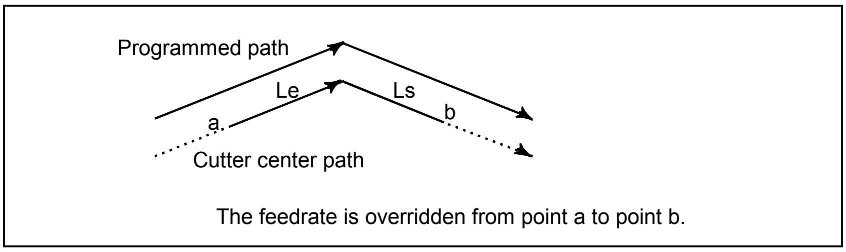


Fig.5.5.2 (a) Inner corner

- Override range

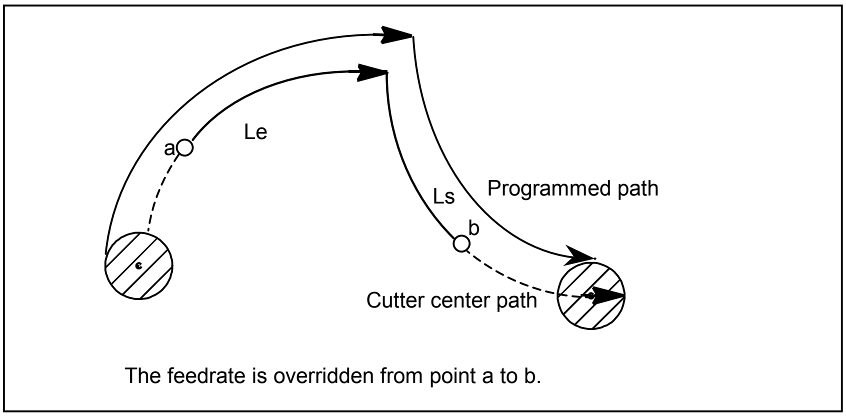
When a corner is determined to be an inner corner, the feedrate is overridden before and after the inner corner. The distances Ls and Le, where the feedrate is overridden, are distances from points on the cutter center path to the corner (Fig.5.5.2 (b), Fig.5.5.2 (c)). Ls and Le are set with parameter Nos. 6613 and 6614.



The feedrate is overridden from point a to point b.

Fig.5.5.2 (b) Override Range (Straight Line to Straight Line)

When a programmed path consists of arcs, override is enabled for the end point of a block if the distance is within Le. Similarly, override is enable for the start point of a block if the distance is within Ls. (Fig.5.5.2 (c)).



The feedrate is overridden from point a to b.

Fig.5.5.2 (c) Override Range (Arc to Arc)

Regarding program (2) of an arc, the feedrate is overridden from point a to point b and from point c to point d (Fig.5.5.2 (d)).

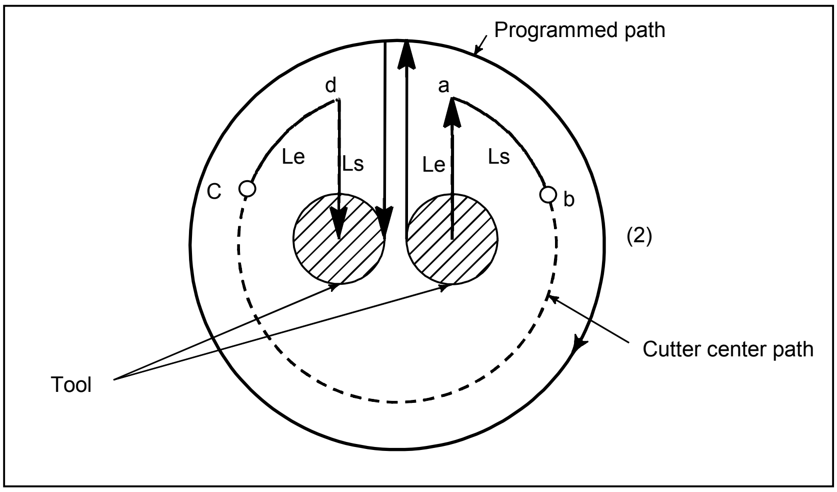


Fig.5.5.2 (d) Override Range (Straight Line to Arc, Arc to Straight Line)

- Override value

An override value is set with parameter No. 6612. An override value is valid even for dry run and F1-digit specification.
In the feed per minute mode, the actual feedrate is as follows:
F×(automatic override for inner corners) × (feedrate override)

Limitation

- Start-up/G41, G42

Override for inner corners is disabled if the corner is preceded by a start-up block or followed by a block including G41 or G42.

- Offset

Override for inner corners is not performed if the offset is zero.

5.5.2.2 Circular cutting feedrate change

The feedrate along a programmed path is set to a specified feedrate (F) by setting a circular cutting feedrate with respect to F, as follows: (Fig. 5.5.2(e))

$$F \times \frac{Rc}{Rp}$$

Rc : Cutter center path radius
Rp : Programmed radius
This is also applicable to dry run and a single-digit F command.

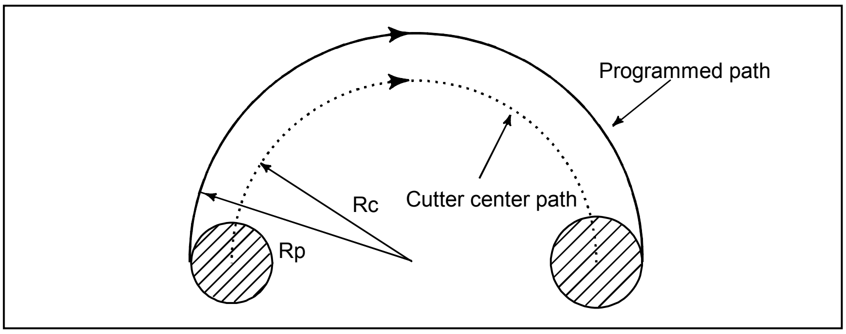


Fig.5.5.2 (e) Internal circular cutting feedrate change

If Rc is so much smaller than Rp that Rc/Rp = 0, the tool stops. Set a minimum deceleration ratio (MDR) in parameter No. 6610, and a tool feedrate of F x MDR when Rc/Rp MDR.

⚠ **CAUTION**

When internal circular cutting must be performed together with override for inner corners, the feedrate of the tool is as follows:

$$F \times \frac{Rc}{Rp} \times (\text{override for the inner corners}) \times (\text{feedrate override})$$

NOTE

1 By setting bit 5 (CAFC) of parameter No. 1402, a cutting feedrate change can be made even when an external offset is applied.

2 This function is enabled in cutter compensation mode, regardless of the setting of G62. However, this function can be enabled only in G62 mode by setting bit 0 (COV) of parameter No. 6600.

5.6 AUTOMATIC VELOCITY CONTROL

5.6.1 Automatic Velocity Control during Involute Interpolation

To enhance the machining precision, the function for automatic velocity control during involute interpolation automatically overrides the specified feedrate as follows:

- Override applied when an internal offset is applied for cutter compensation
- Override applied near the basic circle

Explanation

- Override applied when an internal offset is applied for cutter compensation(OVRa)

When cutter compensation is applied to involute interpolation, the tangential velocity along the path of the tool center is continuously limited to the specified velocity during usual involute interpolation.

In this case, the actual cutting speed at the tool periphery (the cutting point) on the programmed path, changes because the curvature of the involute curve sometimes changes.

In particular when the tool is offset inside the involute curve, the closer to the basic circle the tool is, the greater than the specified feedrate the actual cutting speed is.

To ensure smooth machining, it is desirable that the actual cutting speed is limited to the specified feedrate. This function calculates the override value according to the curvature of the involute curve, which is changeable. The function continually limits the actual cutting speed, which is the tangential velocity at the cutting point, to the specified feedrate. This function is particularly effective during involute interpolation with internal offset.

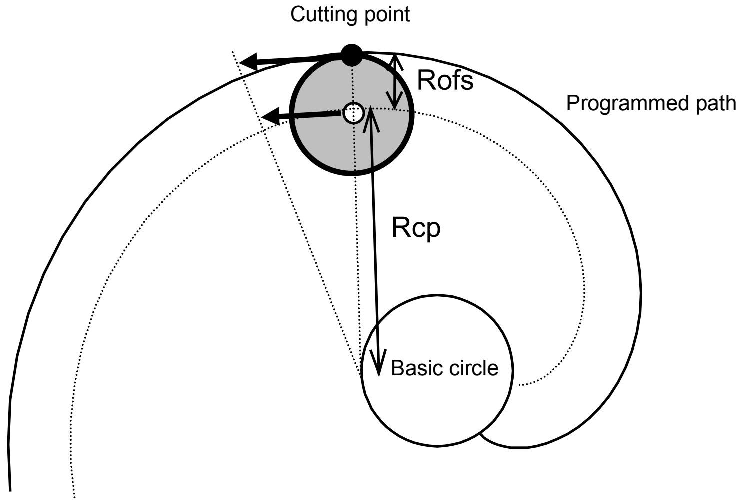


Fig.5.6.1 (a) Override applied when an internal offset is applied for cutter compensation

$$\mathrm{OVR} = \frac{\mathrm{Rcp}}{\mathrm{Rcp} + \mathrm{Rofs}} \times 100 \text{ (for external offset)}$$

$$\mathrm{OVR} = \frac{\mathrm{Rcp}}{\mathrm{Rcp} - \mathrm{Rofs}} \times 100 \text{ (for internal offset)}$$

Rcp : Radius of the involute curve at the center of the tool (The involute curve passes through the center of the tool.)
Rofs :Radius of the tool

- Override clamping

When an override is applied to the specified feedrate either when an internal offset is applied for cutter compensation or when near the basic circle, the velocity at the center of the tool may be 0 in the proximity of the basic circle. To prevent this, the lower override limit (OVR1o) must be specified using its parameter (No.6630).
For an internal offset, the velocity at the tool center may become too low in the proximity of the basic circle.
To prevent this, the lower override limit (OVR1o) must be specified using its parameter (No.6630).
When the override lower than the setting of parameter (No. 6630) is applied, the velocity is clamped at the velocity to which override (OVR1o) is applied.
For an external offset, an override may become too large. In this case, the feedrate is clamped by the maximum cutting feedrate.

- Acceleration/deceleration clamping near a base circle

If an acceleration calculated from the curvature radius of an involute curve exceeds the value specified in the parameter, the feedrate in the tangent direction is controlled so that the acceleration specified in the parameter is not exceeded. An acceleration not exceeding a limit can be maintained, so that efficient feedrate control tailored to the machine limits can be exercised. Moreover, smooth, continuous feedrate control can be exercised to reduce stress and strain in machining near a base circle. When calculating acceleration, the curvature radius of an involute curve and a feedrate in the tangent direction are used with a circular acceleration expression, as follows:

$$\text{Acceleration} = F \times F/R$$

F : Feedrate in the tangent direction
R : Curvature radius

Specify the maximum allowable acceleration in parameter No. 1663.

⚠ **CAUTION**
A maximum allowable acceleration/deceleration value is to be specified for each axis. So, use the smaller of the maximum allowable acceleration/deceleration values for the two involute interpolation axes. If one of the values for the two involute interpolation axes is 0, the acceleration/deceleration clamp function is not disabled, but the non-zero value is used as a maximum allowable acceleration/deceleration value. If both values are 0, the acceleration/deceleration clamp function is disabled.

If a calculated acceleration/deceleration value is greater than a maximum allowable value, the feedrate is clamped according to the formula below:

$$\text{Clamp feedrate} = \sqrt{\text{curvature radius} \times \text{maximum allowable acceleration/deceleration}}$$

If a resultant clamp feedrate is less than the minimum allowable feedrate specified in parameter No. 1491, the minimum allowable feedrate is used as a clamp feedrate.

5.6.2 Automatic Velocity Control during Polar Coordinate Interpolation

If the feedrate component of a rotation axis exceeds a maximum allowable cutting feedrate in polar coordinate interpolation mode, an OT512 alarm (feedrate excess alarm) is issued. However, this function can automatically control the feedrate so that a feedrate excess alarm is not issued.

Explanation

- Automatic Override

If the feedrate component of a rotation axis exceeds a maximum allowable feedrate (obtained by multiplying a maximum allowable cutting feedrate by an allowance rate specified in parameter No. 1056), the following override is automatically applied:
Override = (maximum allowable feedrate)/(feedrate component of rotation axis) x 100 (%)

- Automatic velocity clamp

When the velocity for the rotation axis exceeds the maximum cutting feedrate even after the method in automatic override is used, the velocity is clamped so that it does not exceed the maximum cutting feedrate.
This function is used only when the center of the tool is very close to that of the rotation axis.

Example

```
G90 G00 X10.0 C0. ;
G12.1 ;
G01 C0.1 F1000 ;
X-10.0 :
G13.1 ;
```

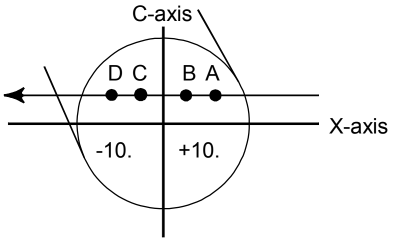


Fig.5.6.2 (a) Automatic velocity control during polar coordinate interpolation

For example, the above program is executed after the maximum cutting feedrate for the rotation axis is specified as 360 (3600 deg/min), and the permissible ratio of automatic override during polar coordinate interpolation (parameter 1056) is specified as 0 (90%). In this case, the automatic override starts at point A (X = 2.273), and automatically clamping the velocity starts at point B (X = 0.524). The minimum value of the automatic override is 3%. The velocity remains clamped up to point C (X = -0.524), and the automatic override is applied up to point D (X = -2.273).
(The above coordinates are Cartesian coordinate.)

NOTE

1 The machine lock or interlock function sometimes does not work as soon as the corresponding switch is turned on while the automatic clamp function is being executed.

2 If the feed hold switch is turned on while the automatic clamp function is being executed, the feed hold is accepted, and the SPL signal (automatic operation hold) is output. In this case, however, the tool sometimes does not stop immediately.

5.7 DWELL

Format

DwellG04 X_ ; or G04 P_ ;
X_; Specify a time (decimal point permitted)
P_; Specify a time (decimal point not permitted)

Explanation

By specifying a dwell, the execution of the next block is delayed by the specified time.
Bit 5 (DWL) of parameter No. 2400 can specify dwell for each rotation in feed per rotation mode (G95).

The command unit is as follows:
- Command by X

Command value range	Dwell time unit
The command value range of the X-axis is used.	s(or rev)

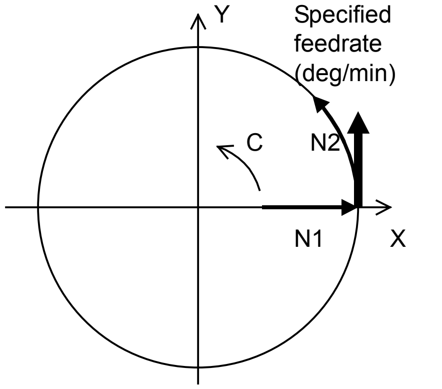
- Command by P

Command value range	Dwell time unit
1 to 99999999	0.001 s(or rev)

5.8 FEEDRATE SPECIFICATION ON A VIRTUAL CIRCLE FOR A ROTARY AXIS

The method of feedrate specification on a machine with a rotation axis is improved.

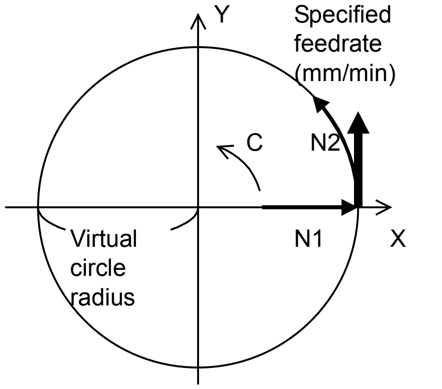
[Conventional method]



Sample program:
N1G91G01X10.F100.
N2C10.F50

The feedrate of the rotation axis is specified using the speed of the rotation axis.

[Method of feedrate specification on a virtual circle for a rotation axis]



Sample program:
N1G91G01X10.F100.
N2C10.

The travel feedrate on the virtual circle of a radius specified by a parameter is the specified feedrate.
By setting a virtual radius of 0, the rotation axis can be excluded from feedrate calculation.

NOTE

By using this function, the travel feedrate on a virtual circle becomes the specified feedrate. In general, however, the feedrate at a cutting point does not become a specified feedrate.

Explanation

An example is explained using a machine that has three linear axes (X, Y, Z) and two rotation axes (B, C).

- Rotation axis feedrate calculation

[Conventional method]

Conventionally, feedrate control is exercised so that the time T calculated by the expression below is used to travel a specified distance.

$$L = \sqrt{\Delta X^2 + \Delta Y^2 + \Delta Z^2 + \Delta B^2 + \Delta C^2}$$

$$T = \frac{L}{F}$$

[Method of feedrate specification on a virtual circle for a rotation axis]

With the method of feedrate specification on a virtual circle for a rotation axis, feedrate control is exercised so that the time T' calculated by the expression below is used to travel a specified distance.

$$L' = \sqrt{\Delta X^2 + \Delta Y^2 + \Delta Z^2 + (l_B * \Delta B * \pi / 180)^2 + (l_C * \Delta C * \pi / 180)^2}$$

$$T' = \frac{L'}{F}$$

l_B, l_C	: Virtual circle radius
$\Delta X, \Delta Y, \Delta Z$	: Travel distance on each axis ($mm, inch$)
$\Delta B, \Delta C$	: Travel distance on each axis (deg)
L, L'	: Distance
F	: Specified feedrate (mm/min, $inch$/min)
T, T'	: Time

- Usable functions

This function can be used with the following functions:

- Linear interpolation
- Dry run
- Tool length compensation in a specified direction (when bit 2 (FWR) of parameter No. 7711) is set to 1)

- Acceleration/deceleration before look-ahead interpolation

Acceleration/deceleration before look-ahead interpolation is applied to the feedrate on a virtual circle. However, parameters such as for an allowable feedrate difference (parameter No. 1478) and a maximum allowable acceleration (parameter No. 1660) are to be set for each rotation axis.

- Parameter setting

For a rotation axis for which this function is to be used, set the following parameters:

- Bit 0 (ROT) of parameter No. 1006 = 1
- Bit 1 (RFD) of parameter No. 1010 = 1
- Parameter No. 2524 = Virtual circle radius

- Setting for excluding rotation axes from feedrate calculation

With the following parameter settings (virtual circle radius = 0.0), rotation axes are excluded from feedrate calculation:

- Bit 0 (ROT) of parameter No. 1006 = 1
- Bit 1 (RFD) of parameter No. 1010 = 1
- Parameter No. 2524 = 0.0

If only a rotation axis is specified when the parameter settings above are made, the rotation axis is moved at a maximum cutting feedrate.

- Rewriting of parameters with G10

The following parameters can be rewritten with G10 in the part program:

- Bit 1 (RFD) of parameter No. 1010
- Parameter No. 2524

Limitations

- Unusable functions

This function cannot be used with the following functions:

- G functions of group 01 listed below
  - Positioning
  - Circular interpolation, helical interpolation, spiral interpolation, conical interpolation
  - Circular threading B
  - Involute interpolation
  - Exponential function
  - Three-dimensional circular interpolation
  - Spline interpolation
  - NURBS interpolation
- Cylindrical interpolation
- Polar coordinate interpolation
- Normal direction control
- Feed per revolution
- Inverse time feed

- Parallel axis control, twin table control

When using this function for parallel axis control or twin table control, set the same value in parameter No. 2524 for the master and slave axes.

Examples

- When a rotation axis is specified singly

Travel time when G91 G01 B10. F10. is specified

(1) When parameter No. 2524 = 10.0 (mm), the travel time is:

$$10.0(mm)*10.0(\deg)*\pi/180/10(mm/\min)*60 = 10.47(\text{seconds})$$

(2) When parameter No. 2524 = 36.0 (mm), the travel time is:

$$36.0(mm)*10.0(\deg)*\pi/180/10(mm/\min)*60 = 37.70(\text{seconds})$$

(3) When parameter No. 2524 = 0.0 (mm)

A movement is made at a maximum cutting feedrate.

- When a rotation axis and linear axis are specified at the same time

Travel time when G91 G01 X5. B10. F10. is specified

(1) When parameter No. 2524 = 10.0 (mm), the travel time is:

$$\sqrt{5.0(mm)^2 + (10.0(mm)*10.0(\deg)*\pi/180)^2}/10(mm/\min)*60 = 31.78(\text{seconds})$$

(2) When parameter No. 2524 = 36.0 (mm), the travel time is:

$$\sqrt{5.0(mm)^2 + (36.0(mm)*10.0(\deg)*\pi/180)^2}/10(mm/\min)*60 = 48.18(\text{seconds})$$

(3) When parameter No. 2524 = 0.0 (mm), the travel time is:

$$5.0(mm)/10(mm/\min)*60 = 30.00(\text{seconds})$$

Reference items

FANUC Series 15*i*/150*i*-MB	Operator's Manual (Programming) (B-63784EN)	II. 5.3	Cutting Feed

5.9 AUTOMATIC FEEDRATE CONTROL BY AREA

Overview

When an area on the XY plane(*1) is specified in cutting mode in automatic operation, the area override can be applied to a specified feedrate(*2) if the tool is in the specified area.
To do this, first set an area on the XY plane by setting parameter Nos. 1280 to 1287.
Up to four areas can be set on the XY plane.
(For details on how to set areas, see the explanation of operation.)
When the tool is in an area set as mentioned above, the tool can be moved at the specified feedrate overriden by the area override set in parameter No. 2060 to 2063.
For each of the four areas, a separate area override can be set.

NOTE
1 X and Y denote the two axes set to 1 and 2 in parameter No.1022, respectively.
2 (Specified feedrate) =
(Cutting feedrate specified with F in program) *
(Feedrate override)

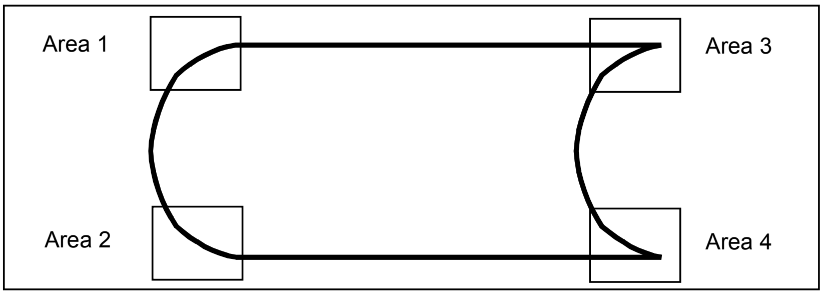


Fig.5.9 (b) Example to Setting Four Areas on the XY Plane

Defining areas

- Up to four areas can be set. We'll call these area 1 to area 4.
- Each area is set as a quadrangle having sides parallel to the X- and Y-axes.
  Each area includes the border of the quadrangle.
- An area is defined by setting one of the two pairs of diagonal vertexes of a quadrangle. (See Fig. 5.9(b).)
  Either of the two vertex pairs may be used to determine a unique quadrangle.
  If two diagonal vertexes have the same coordinates, the area is a point.
- When two or more areas overlap, the area override for the area with the smallest area number is used for the overlapping portion. (See Fig. 5.9(c).)

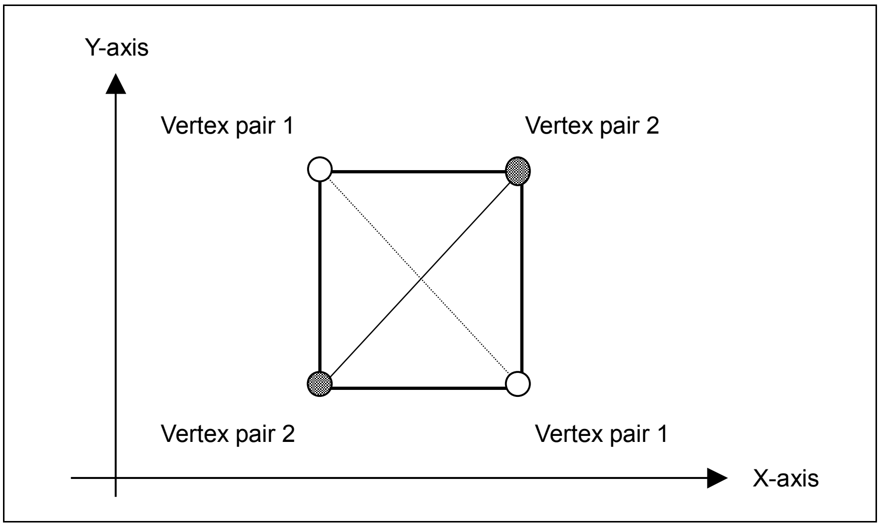


Fig.5.9 (c) Two Pairs of Diagonal Vertexes of a Quadrangle

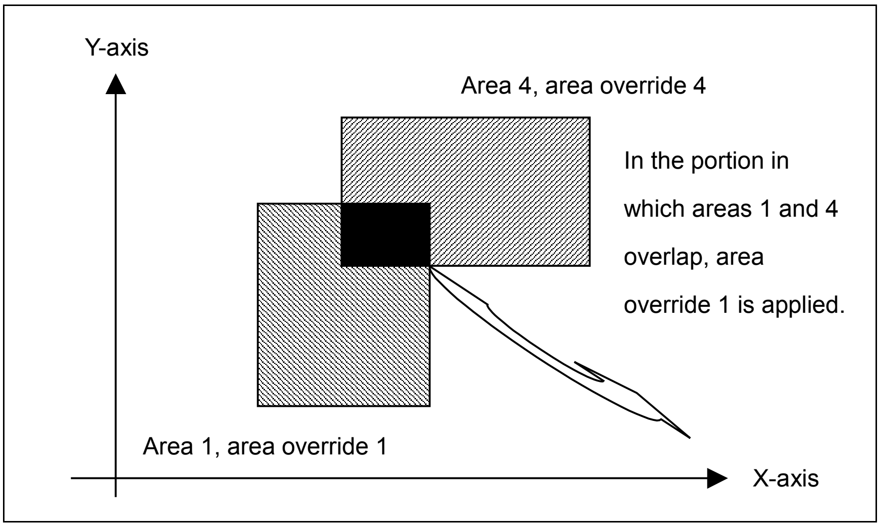

Y-axis
Area 4, area override 4
In the portion in which areas 1 and 4 overlap, area override 1 is applied.
Area 1, area override 1
X-axis

Fig.5.9 (d) When Areas Overlap One Another

Determining whether the tool is in an area

Whether the tool has entered an area is checked based on the machine coordinates.

Setting an area

There are three methods of setting an area, that is, for setting two diagonal vertexes, as follows:

(1) Specify and write a parameter number in G10.
Press the [Setting] key several times to display the area setting screen for automatic feedrate control.

(2) Enter the necessary data directly in MDI mode.

(3) Move the tool manually to a desired position, then press CRT/MDI soft key [AREA SETTING] to record the position.
Then, the coordinates of the current cursor position change.

Setting an area override

An area override is set within the range of 0% to 127%.
For each of the four areas, a separate area override can be set.
There are two methods of setting an area override, as follows:
(1) Specify and write a parameter number in G10.

Press the [Setting] key several times to display the area setting screen for automatic feedrate control.
(2) Enter the necessary data directly in MDI mode.

Cautions

⚠ **CAUTION**

1 Only in cutting mode in automatic operation, a check is made to see whether the tool has entered an area, and the tool is moved at a feedrate multiplied by the percentage set in parameter No. 2060 to 2063.
2 This function is disabled in the following modes: Threading, Feed per revolution, Tapping.
3 This function is also disabled when the X-axis and Y-axis are used for the axes subjected to the following:
PMC axis control, Chopping.
4 The percentage of the cutting feedrate is obtained by multiplying the normal feedrate override by the area override of this function. If the multiplication result exceeds 254%, it is clamped to 254%.
5 This function is disabled when the tool moves at a dry run speed.
6 In three-dimensional coordinate conversion mode, this function is valid for areas on the XY plane in the machine coordinate system.
7 In synchronization control, areas are set for the master axis.
The area override of this function is applied to all the synchronized axes.

6 REFERENCE POSITION

A CNC machine tool has a special position where, generally, the tool is exchanged or the coordinate system is set, as described later. This position is referred to as a reference position.

6.1 REFERENCE POSITION RETURN

The reference position is a fixed position on a machine tool to which the tool can easily be moved by the reference position return function. For example, the reference position is used as a position at which tools are automatically changed. Up to four reference positions can be specified by setting coordinates in the machine coordinate system in parameters (No. 1240 to 1243).

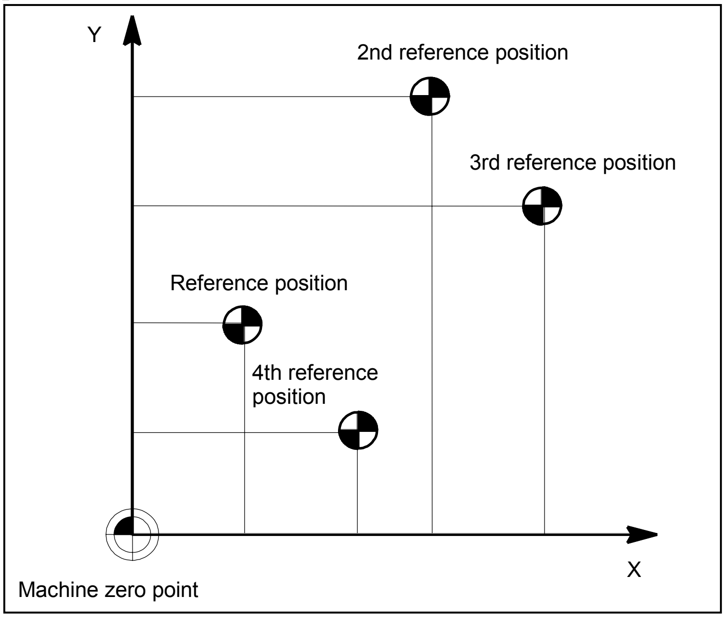


Fig.6.1 (a) Machine coordinate system and reference positions

- Reference position return and movement from the reference position

Tools are automatically moved to the reference position via an intermediate position along a specified axis. Or, tools are automatically moved from the reference position to a specified position via an intermediate position along a specified axis.
When reference position return is completed, the lamp for indicating the completion of return goes on.

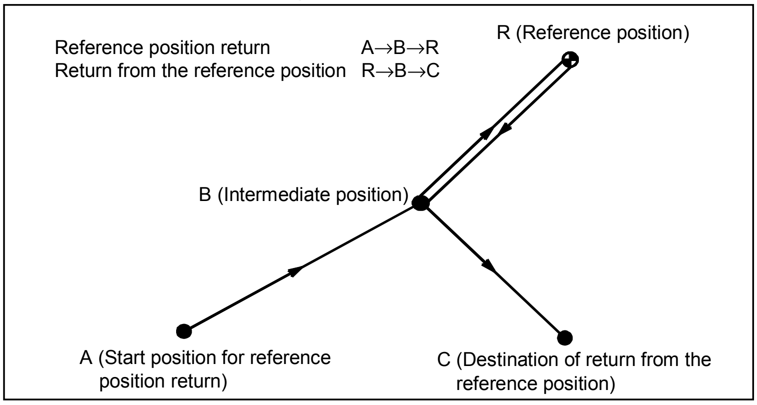

Reference position return A→B→R
Return from the reference position R→B→C
R (Reference position)
B (Intermediate position)
A (Start position for reference position return)
C (Destination of return from the reference position)

Fig.6.1 (b) Reference position return and return form the reference position

- Reference position return check

The reference position return check (G27) is the function which checks whether the tool has correctly returned to the reference position as specified in the program. If the tool has correctly returned to the reference position along a specified axis, the lamp for the axis goes on.

Format

- Reference position return

```
G28 IP_;  Reference position return
G30 P2 IP   2nd reference position return
            (P2 can be omitted.)
G30 P3 IP_; 3rd reference position return
G30 P4 IP_; 4th reference position return
   IP: Command specifying the intermediate position
        (Absolute/incremental command)
```

- Return from reference position

```
G29 IP_;
 IP: Command specifying the destination of return from
     reference position
     (Absolute/incremental command)
```

- Reference position return check

```
G27 IP_;
 IP: Command specifying the reference position
       (Absolute/incremental command)
```

Explanation

- Reference position return (G28)

Positioning to the intermediate or reference positions are performed at the rapid traverse rate of each axis.

Therefore, for safety, the cutter compensation, and tool length compensation should be cancelled before executing this command.

The coordinates for the intermediate position are stored in the CNC only for the axes for which a value is specified in a G28 block. For the other axes, the previously specified coordinates are used.

```
(Example)     N1 G28 X40.0 ; Intermediate position (X40.0)
              N2 G28 Y60.0 ; Intermediate position (X40.0,  Y60.0)
```

- 2nd, 3rd, and 4th reference position return (G30)

In a system without an absolute-position detector, the first, third, and fourth reference position return functions can be used only after the reference position return (G28) or manual reference position return (See Operation II-3.1) is made. The G30 command is generally used when the automatic tool changer (ATC) position differs from the reference position.

- Return from the reference position (G29)

In general, it is commanded immediately following the G28 command or G30.

For incremental programming, the command value specifies the incremental value from the intermediate point. Positioning to the intermediate or reference points are performed at the rapid traverse rate of each axis.

When the workpiece coordinate system is changed after the tool reaches the reference position through the intermediate point by the G28 command, the intermediate point also shifts to a new coordinate system. If G29 is then commanded, the tool moves to the commanded position through the intermediate point which has been shifted to the new coordinate system.

The same operations are performed also for G30 commands.

- Reference position return check (G27)

G27 command positions the tool at rapid traverse rate. If the tool reaches the reference position, the reference position return lamp lights up. However, if the position reached by the tool is not the reference position, an alarm (No. 185) is displayed.

Limitation

- Status the machine lock being turned on

The lamp for indicating the completion of return does not go on when the machine lock is turned on, even when the tool has automatically returned to the reference position. In this case, it is not checked whether the tool has returned to the reference position even when a G27 command is specified.

- First return to the reference position after the power has been turned on (without an absolute position detector)

When the G28 command is specified when manual return to the reference position has not been performed after the power has been turned on, the movement from the intermediate point is the same as in manual return to the reference position.

In this case, the tool moves in the direction for reference position return specified in parameter ZMIx (bit 5 of No. 1006). Therefore the specified intermediate position must be a position to which reference position return is possible.

- Reference position return check in an offset mode

In an offset mode, the position to be reached by the tool with the G27 command is the position obtained by adding the offset value.

Therefore, if the position with the offset value added is not the reference position, the lamp does not light up, but an alarm is displayed instead. Usually, cancel offsets before G27 is commanded.

- Lighting the lamp when the programmed position does not coincide with the reference position

When the machine tool system is an inch system with metric input, the reference position return lamp may also light up even if the programmed position is shifted from the reference position by the least setting increment. This is because the least setting increment of the machine tool system is smaller than its least command increment.

Reference

- Manual reference position return

See Operation II-3.1 Manual reference position return.

Example

```
G28G90X1000.0Y500.0 ;    (Programs movement  from A to B)
T1111 ; (Changing the tool at the reference position)
G29X1300.0Y200.0 ;         (Programs movement from B to C)
```

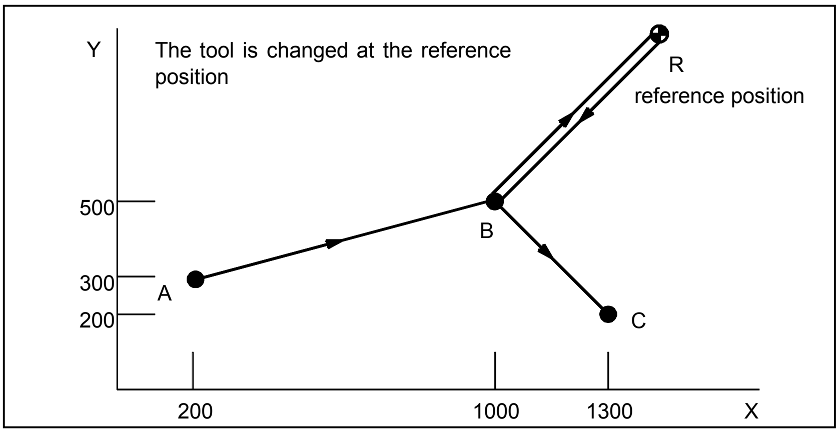


Fig.6.1 (c) Reference position return and return from the reference position

6.2 FLOATING REFERENCE POSITION RETURN (G30.1)

Tools ca be returned to the floating reference position. A floating reference point is a position on a machine tool, and serves as a reference point for machine tool operation. A floating reference point need not always be fixed, but can be moved as required.

Format

G30.1 IP_;

IP_: Command of the intermediate position of the floating reference position
(Absolute command/incremental command)

Explanation

Generally speaking, on a machining center or milling machine, cutting tools can be replaced only at specific positions. A position where tools can be replaced is defined as the second or third reference point. Using G30 can easily move the cutting tools back to these points. On some machine tools, the cutting tools can be replaced at any position unless they interfere with the workpiece.

With these machines, the cutting tools should be replaced at a position as close to the workpiece as possible so as to minimize the machine cycle time. For this purpose, the tool change position is to be changed, depending on the figure of the workpiece. This operation can easily be performed using this function. That is, a tool change position suitable for the workpiece is memorized as a floating reference point. Then command G30. 1 can easily cause return to the tool change position

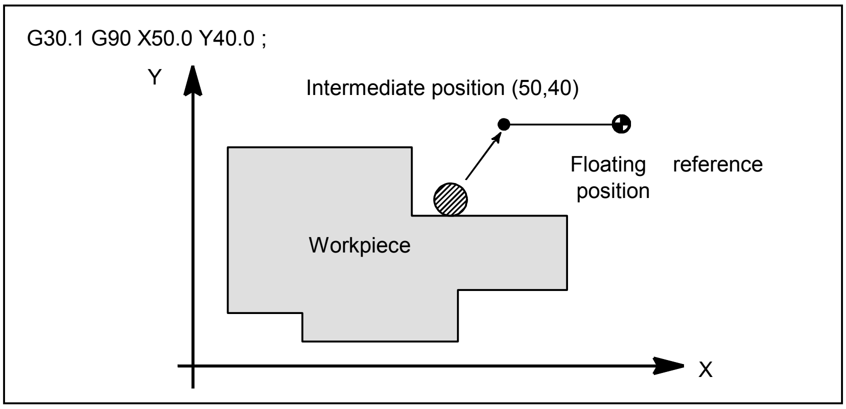
A floating reference point becomes a machine coordinate position memorized by pressing the soft key [SET FRP] on the current positions display screen (See Operation II-9.15).

The G30.1 block first positions the tool at the intermediate point along the specified axes at rapid traverse rate, then further moves the tool from the intermediate point to the floating reference point at rapid traverse rate.

Before using G30.1, cancel cutter compensation and tool length compensation. A floating reference point is not lost even if power is turned off.

The function for returning from the reference position (G29) can be used for moving the tool from the floating reference position (see II-6.1).

Example



```
G30.1 G90 X50.0 Y40.0 ;
```

7 COORDINATE SYSTEM

By teaching the CNC a desired tool position, the tool can be moved to the position. Such a tool position is represented by coordinates in a coordinate system. Coordinates are specified using program axes.
When three program axes, the X-axis, Y-axis, and Z-axis, are used, coordinates are specified as follows:

X_Y_Z_

This command is referred to as a dimension word.

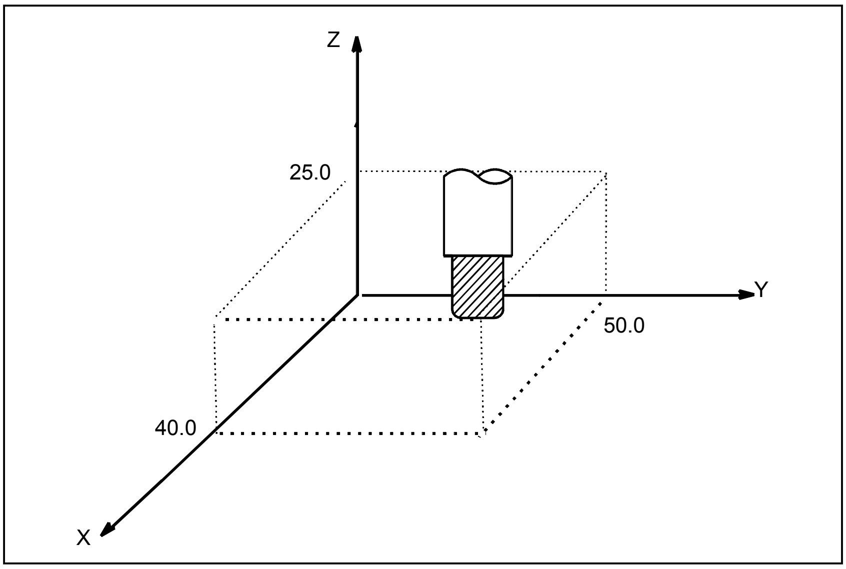


Fig.7 Tool Position Specified by X40.0Y50.0Z25.0

Coordinates are specified in one of following three coordinate systems:

(1) Machine coordinate system
(2) Workpiece coordinate system
(3) Local coordinate system

The number of the axes of a coordinate system varies from one machine to another. So, in this manual, a dimension word is represented as IP_.

7.1 MACHINE COORDINATE SYSTEM

The point that is specific to a machine and serves as the reference of the machine is referred to as the machine zero point. A machine tool builder sets a machine zero point for each machine.
A coordinate system with a machine zero point set as its origin is referred to as a machine coordinate system.
A machine coordinate system is set by performing manual reference position return after power-on (see Operation II-3.1). A machine coordinate system, once set, remains unchanged until the power is turned off.

Format

(G90)G53 IP_;
IP_; Absolute dimension word

Explanation

- Selecting a machine coordinate system (G53)

When a command is specified the position on a machine coordinate system, the tool moves to the position by rapid traverse. G53, which is used to select a machine coordinate system, is a one-shot G code; that is, it is valid only in the block in which it is specified on a machine coordinate system. Specify an absolute command (G90) for G53. When an incremental command (G91) is specified, the G53 command is ignored. When the tool is to be moved to a machine-specific position such as a tool change position, program the movement in a machine coordinate system based on G53.

Limitation

- Cancel of the compensation function

When the G53 command is specified, cancel the cutter compensation, tool length offset, and tool offset.

- G53 specification immediately after power-on

Since the machine coordinate system must be set before the G53 command is specified, at least one manual reference position return or automatic reference position return by the G28 command must be performed after the power is turned on. This is not necessary when an absolute-position detector is attached.

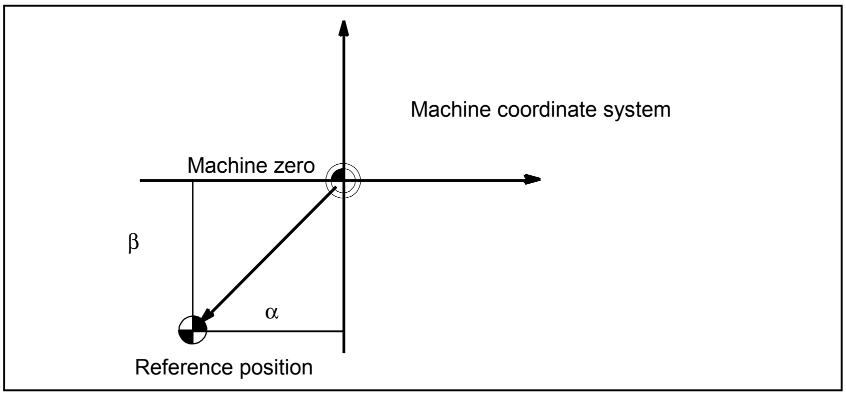
- Specification in the same block

Note that G50/G51, G50.1/G51.1, and G68/G69 codes are ignored in a block where a G53 code is specified.

Reference

- Machine coordinate system

When manual reference position return is performed after power-on, a machine coordinate system is set so that the reference position is at the coordinate values of (α,β) set using parameter No.1240.



7.2 WORKPIECE COORDINATE SYSTEM

A coordinate system used for machining a workpiece is referred to as a workpiece coordinate system.
A workpiece coordinate system can be set using one of the two methods described below.

(1) Method using G92
A workpiece coordinate system is set by specifying a value after G92 in the program.

(2) Method using G54 to G59
Six workpiece coordinate systems are set in advance, based on settings made from the MDI panel, and the desired workpiece coordinate system is selected according to programmed commands G54 to G59.

7.2.1 Setting a Workpiece Coordinate System (G92)

A programmed command establishes a workpiece coordinate system according to the value after G92.

Format

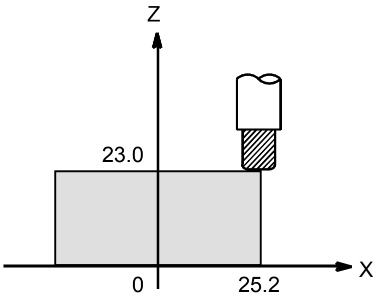
(G90) G92 IP

Explanation

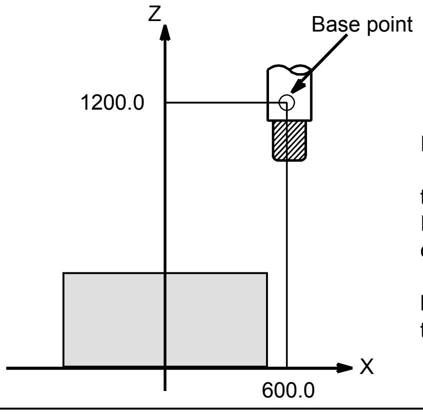
A workpiece coordinate system is set so that a point on the tool, such as the tool tip, is at specified coordinates. If a coordinate system is set using G92 during tool length offset, a coordinate system in which the position before offset matches the position specified in G92 is set.
Cutter compensation is cancelled temporarily with G92.
When a manual reference position return operation is performed in the reset state (with the OP signal set to 0), a workpiece origin offset created by G92 is cancelled, and the original workpiece coordinate system is set. When a manual reference position return operation is performed in the automatic operation state (with the OP signal set to 1), whether the original workpiece coordinate system is set can be chosen by setting bit 3 (PLZ) of parameter No. 1005 and bit 2 (ZNP) of parameter No. 2402.

Example

(Example 1)
Setting the coordinate system by the G92X600.0Z1200.0; command (The base point on the tool holder is the start point for the program.)



(Example 2)
Setting the coordinate system by the G92X2600.2Z1200.0; command (The tool tip is the start point for the program.)



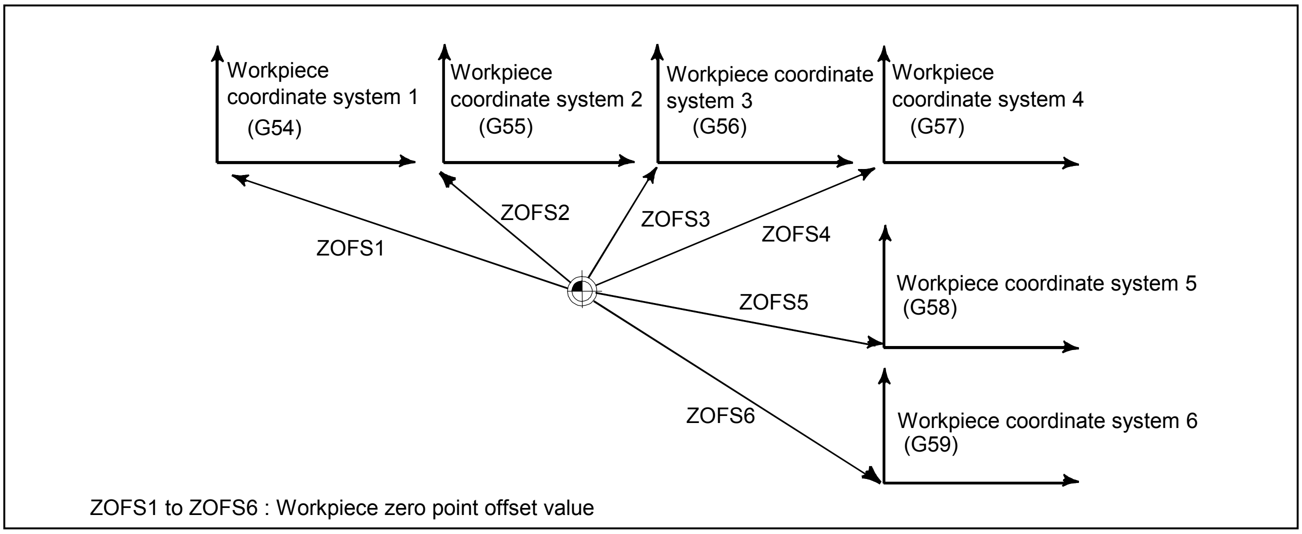
If an absolute command is issued, the base point moves to the commanded position.
In order to move the tool tip to the commanded position, the difference from the tool tip to the base point is compensated by tool length offset. (See II-14.1)

7.2.2 Setting Workpiece Coordinate System (G54 to G59)

Explanation

- Setting workpiece coordinate system

Six workpiece coordinate systems can be set. These six systems are decided by setting the distances of each axis from the machine zero point to the zero points of the coordinate systems, (i.e. the workpiece zero point offset value, by using the CRT/MDI panel).


Workpiece coordinate system 1 (G54)
Workpiece coordinate system 2 (G55)
Workpiece coordinate system 3 (G56)
Workpiece coordinate system 4 (G57)
Workpiece coordinate system 5 (G58)
Workpiece coordinate system 6 (G59)
ZOFS1
ZOFS2
ZOFS3
ZOFS4
ZOFS5
ZOFS6

ZOFS1 to ZOFS6 : Workpiece zero point offset value

ZOFS1:Workpiece zero point offset value of workpiece coordinate system 1 (parameter (No.1221))

ZOFS2:Workpiece zero point offset value of workpiece coordinate system 2 (parameter (No.1222))

ZOFS3:Workpiece zero point offset value of workpiece coordinate system 3 (parameter (No.1223))

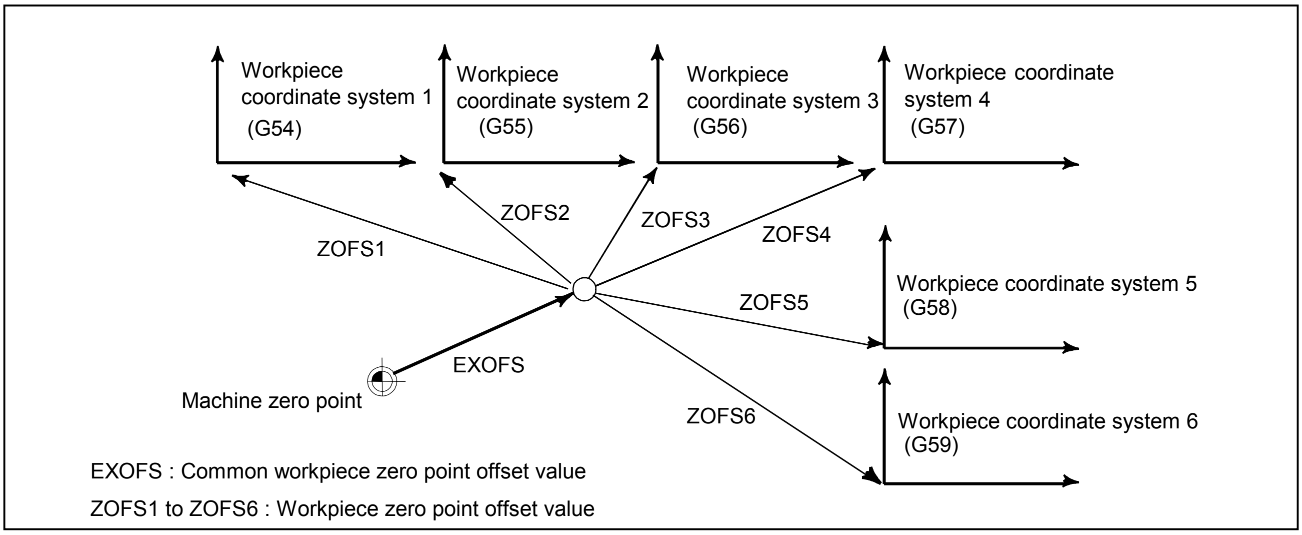
ZOFS4:Workpiece zero point offset value of workpiece coordinate system 4 (parameter (No.1224))

ZOFS5:Workpiece zero point offset value of workpiece coordinate system 5 (parameter (No.1225))

ZOFS6:Workpiece zero point offset value of workpiece coordinate system 6 (parameter (No.1226))

- Shifting workpiece coordinate system

Six workpiece coordinate systems can be shifted by a specified value (common workpiece zero point offset value)


Workpiece coordinate system 1 (G54)
Workpiece coordinate system 2 (G55)
Workpiece coordinate system 3 (G56)
Workpiece coordinate system 4 (G57)
Workpiece coordinate system 5 (G58)
Workpiece coordinate system 6 (G59)
ZOFS1
ZOFS2
ZOFS3
ZOFS4
ZOFS5
ZOFS6
EXOFS
Machine zero point
EXOFS : Common workpiece zero point offset value
ZOFS1 to ZOFS6 : Workpiece zero point offset value

Workpiece zero point offset value can be changed in three ways.

(1) Using MDI panel
See Operator's Manual (Operation).

(2) Using program (G10)
See II-7.2.4, "Changing workpiece coordinate system by program command."

(3) Using external workpiece coordinate system shift function
An input signal to the CNC from the PMC enables you to change the external workpiece zero point offset amount. Refer to machine tool builder's manual for details.

⚠ **CAUTION**

1 The external workpiece zero point offset amount can be set within +7.999 mm or +0.79999 inch for every axis.

2 When G92 is specified after a shift value has been specified, the shift value is ignored.
(Example)
When G92X100.0Z80.0; is commanded, a workpiece coordinate system in which present tool position is X = 100.0, Z = 80.0 is established, irrespective of set shift value.

3 The diameter or radius for movement along the X-axis conforms to the diameter or radius specified in the part program.

7.2.3 Selecting Workpiece Coordinate System(G54 to G59)

A set workpiece coordinate system is selected with a programmed command.

Format

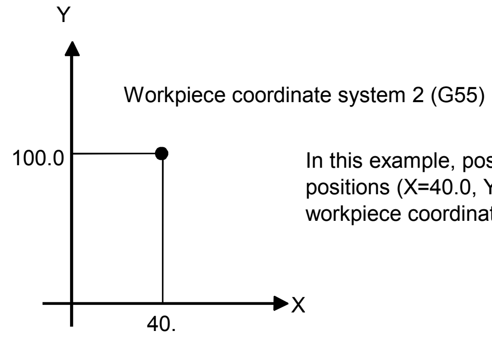
G54	**Workpiece coordinate system 1**
G55	**Workpiece coordinate system 2**
G56	**Workpiece coordinate system 3**
G57	**Workpiece coordinate system 4**
G58	**Workpiece coordinate system 5**
G59	**Workpiece coordinate system 6**

Explanation

Workpiece coordinate system 1 to 6 are established after reference position return after the power is turned on. When the power is turned on, G54 coordinate system is selected.

Examples

```
G90 G55 G00 X40.0 Y100.0 ;
```



In this example, positioning is made to positions (X=40.0, Y=100.0) in workpiece coordinate system 2.

7.2.4 Changing Workpiece Coordinate System

The six workpiece coordinate systems specified with G54 to G59 can be changed by changing an common workpiece zero point offset value or workpiece zero point offset value.

Three methods are available to change an common workpiece zero point offset value or workpiece zero point offset value.

(1) Inputting from the MDI panel (See Operation II-9.2.)

(2) Programming by G10 or G92

(3) Using the common data input function
An external workpiece zero point offset value can be changed by input signal to CNC. Refer to machine tool builder's manual for details

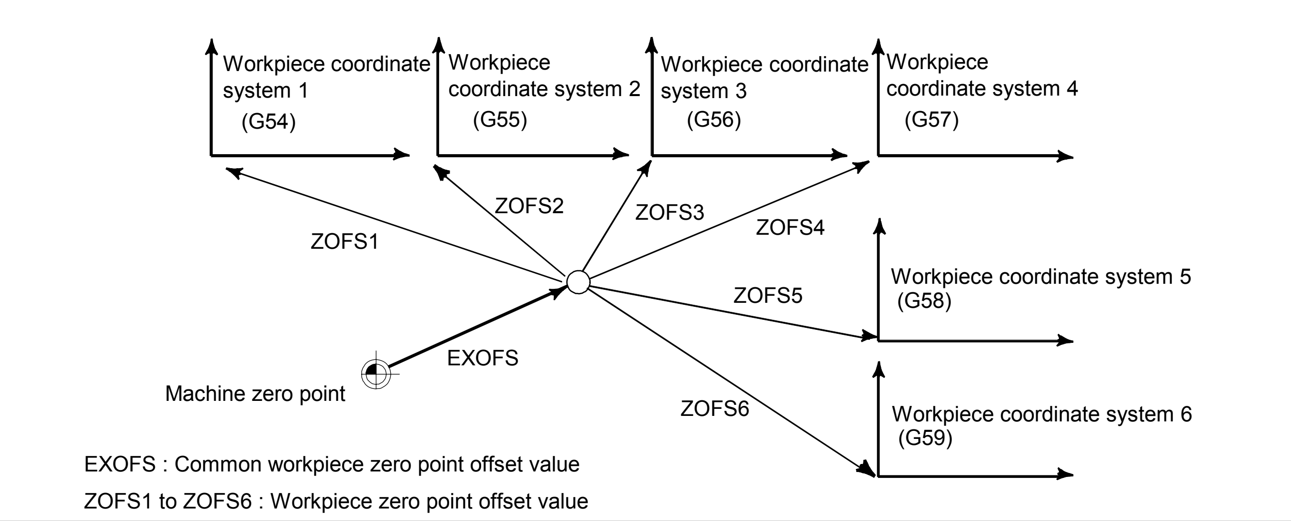

Workpiece coordinate system 1 (G54)
Workpiece coordinate system 2 (G55)
Workpiece coordinate system 3 (G56)
Workpiece coordinate system 4 (G57)
ZOFS1
ZOFS2
ZOFS3
ZOFS4
ZOFS5
Workpiece coordinate system 5 (G58)
EXOFS
Machine zero point
ZOFS6
Workpiece coordinate system 6 (G59)
EXOFS : Common workpiece zero point offset value
ZOFS1 to ZOFS6 : Workpiece zero point offset value

Fig.7.2.3 (a) Changing an external workpiece zero point offset value or workpiece zero point offset value

Format

- Changing by G10

G10 L2 Pp IP_;

p=0 : External workpiece zero point offset value

p=1 to 6: Workpiece zero point offset value correspond to workpiece coordinate system 1 to 6

IP: For an absolute command (G90), workpiece zero point offset for each axis.
For an incremental command (G91), value to be added to the set workpiece zero point offset for each axis (the result of addition becomes the new workpiece zero point offset).

- Changing by G92

G92 IP_;

Explanation

- Changing by G10

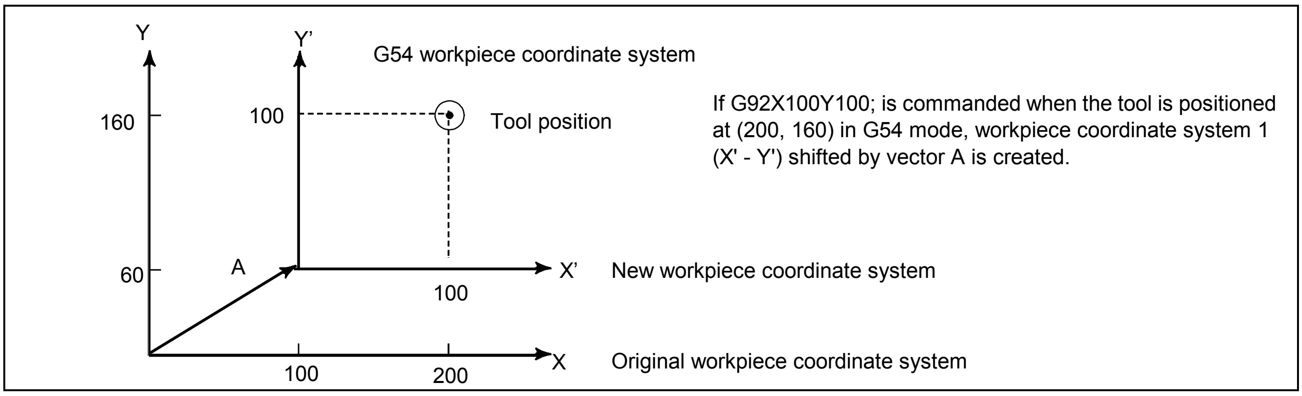
With the G10 command, each workpiece coordinate system can be changed separately.

- Changing by G92

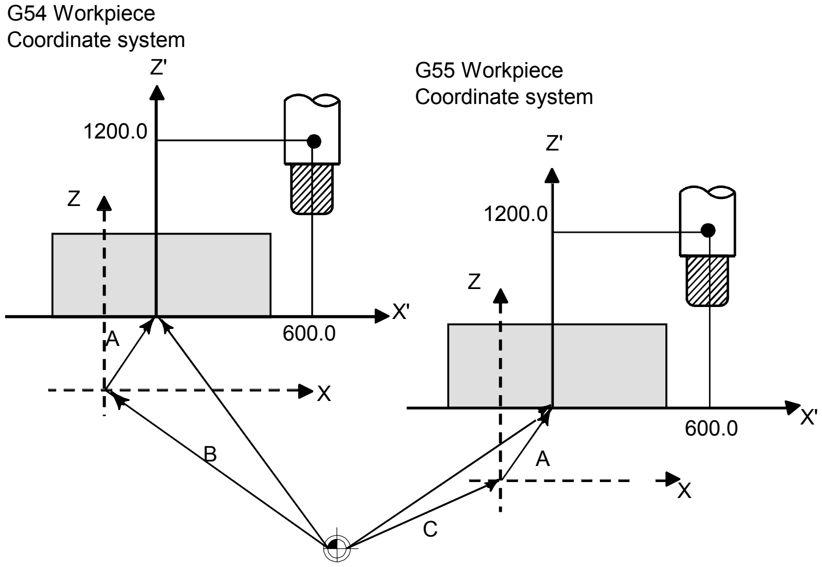
By specifying G92IP_;, a workpiece coordinate system (selected with a code from G54 to G59) is shifted to set a new workpiece coordinate system so that the current tool position matches the specified coordinates (IP_).
Then, the amount of coordinate system shift is added to all the workpiece zero point offset values. This means that all the workpiece coordinate systems are shifted by the same amount.

> ⚠ **WARNING**
> When a coordinate system is set with G92 after an common workpiece zero point offset value is set, the coordinate system is not affected by the common workpiece zero point offset value. When G92X100.0Z80.0; is specified, for example, the coordinate system having its current tool reference position at X = 100.0 and Z = 80.0 is set.

Example


Y
Y'
G54 workpiece coordinate system
160
100
Tool position
60
A
X'
New workpiece coordinate system
100
100
200
X
Original workpiece coordinate system

If G92X100Y100; is commanded when the tool is positioned at (200, 160) in G54 mode, workpiece coordinate system 1 (X' - Y') shifted by vector A is created.



X' - Z'.............New workpiece coordinate system
X - Z ....Original workpiece coordinate system
A ; Offset value created by G92
B : Workpiece zero point offset value in theG54
C : Workpiece zero point offset value in the G55

Suppose that a G54 workpiece coordinate system is specified. Then, a G55 workpiece coordinate system where the black circle on the tool (figure at the left) is at (600.0,12000.0) can be set with the following command if the relative relationship between the G54 workpiece coordinate system and G55 workpiece coordinate system is set correctly:G92X600.0Z1200.0;Also, suppose that pallets are loaded at two different positions.

G92X600.0 Z1200.0;

If the relative relationship of the coordinate systems of the pallets at the two positions is correctly set by handling the coordinate systems as the G54 workpiece coordinate system and G55 workpiece coordinate system, a coordinate system shift with G92 in one pallet causes the same coordinate system shift in the other pallet. This means that workpieces on two pallets can be machined with the same program just by specifying G54 or G55.

7.2.5 Adding Workpiece Coordinate Systems (G54.1)

Besides the six workpiece coordinate systems (standard workpiece coordinate systems) selectable with G54 to G59, 48 additional workpiece coordinate systems (additional workpiece coordinate systems) can be used. Alternatively, up to 300 additional workpiece coordinate systems can be used.

Format

```
G54.1 Pn ;
  n = 1 to 48
```

Explanation

- Selecting the additional workpiece coordinate systems

When a P code is specified together with G54.1 (G54), the corresponding coordinate system is selected from the additional workpiece coordinate systems (1 to 48).
A workpiece coordinate system, once selected, is valid until another workpiece coordinate system is selected. Standard workpiece coordinate system 1 (selectable with G54) is selected at power-on.

G54.1 P1	Additional workpiece coordinate system 1
G54.1 P2	Additional workpiece coordinate system 2
:	
G54.1 P48	Additional workpiece coordinate system 48

As with the standard workpiece coordinate systems, the following operations can be performed for a workpiece zero point offset in an additional workpiece coordinate system:

1) The OFFSET function key can be used to display and set a workpiece zero point offset value.
2) The G10 function enables a workpiece zero point offset value to be set by programming (refer to II-15).
3) A custom macro allows a workpiece zero point offset value to be handled as a system variable.
4) Workpiece zero point offset data can be entered or output as external data.
5) The PMC window function enables workpiece zero point offset data to be read as program command modal data.

- Setting of the workpiece origin offset in an additional workpiece coordinate system (G10)

The following command can be used to set the workpiece origin offset in an additional workpiece coordinate system.

G10 L20 Pn IP_ ; (n=1 to 48)

Pn specifies a desired workpiece coordinate system by using its coordinate system number.

IP specifies a setting together with the address of each axis.

When the specified workpiece origin offset is an absolute value, it is assumed to be the new offset. When it is an incremental value, the sum of the current offset and the specified offset is assumed to be the new offset.

Limitation

A P code must be specified after G54.1 (G54). If G54.1 is not followed by a P code in the same block, additional workpiece coordinate system 1 (G54.1P1) is assumed.

If a value not within the specifiable range is specified in a P code, an P/S alarm (PS0305) is issued.

P codes other than workpiece offset numbers cannot be specified in a G54.1 (G54) block.

Example) G54.1G04 P1000 ;

7.2.6 Workpiece Coordinate System Preset (G92.1)

The workpiece coordinate system preset function presets a workpiece coordinate system shifted by manual intervention to the pre-shift workpiece coordinate system. The latter system is displaced from the machine zero point by a workpiece zero point offset value.
There are two methods for using the workpiece coordinate system preset function. One method uses a programmed command (G92.1). The other uses MDI operations on the absolute position display screen, relative position display screen, and overall position display screen (Operation II-10.5.2).

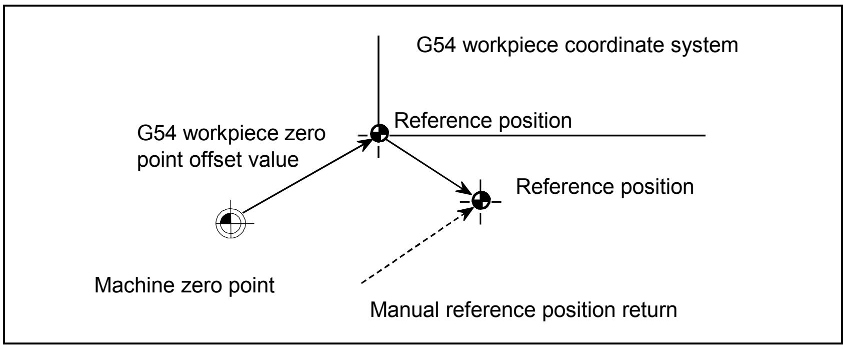
Format

G92.1 IP 0;

IP 0 ; Specifies axis addresses subject to the workpiece coordinate system preset operation.
Axes that are not specified are not subject to the preset operation.

Explanation

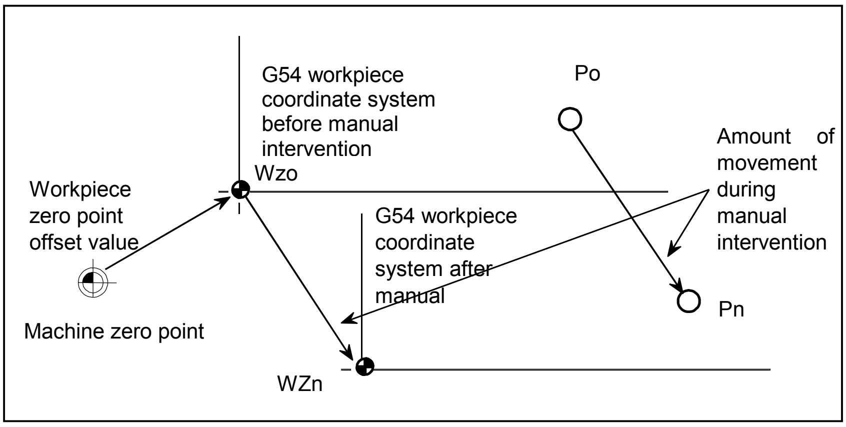
When manual reference position return operation is performed in the reset state, a workpiece coordinate system is shifted by the workpiece zero point offset value from the machine coordinate system zero point. Suppose that the manual reference position return operation is performed when a workpiece coordinate system is selected with G54. In this case, a workpiece coordinate system is automatically set which has its zero point displaced from the machine zero point by the G54 workpiece zero point offset value; the distance from the zero point of the workpiece coordinate system to the reference position represents the current position in the workpiece coordinate system.


G54 workpiece coordinate system
G54 workpiece zero point offset value
Reference position
Reference position
Machine zero point
Manual reference position return

If an absolute position detector is provided, the workpiece coordinate system automatically set at power-up has its zero point displaced from the machine zero point by the G54 workpiece zero point offset value. The machine position at the time of power-up is read from the absolute position detector and the current position in the workpiece coordinate system is set by subtracting the G54 workpiece zero point offset value from this machine position. The workpiece coordinate system set by these operations is shifted from the machine coordinate system using the commands and operations listed following case.

(a) Manual intervention performed when the manual absolute signal is off
(b) Move command executed in the machine lock state
(c) Movement by handle interrupt
(d) Operation using the mirror image function
(e) Setting the local coordinate system using G52, or shifting the workpiece coordinate system using G92

In the case of (a) above, the workpiece coordinate system is shifted by the amount of movement during manual intervention.


G54 workpiece coordinate system before manual intervention
Po
Amount of movement during manual intervention
Wzo
Workpiece zero point offset value
G54 workpiece coordinate system after manual
Pn
Machine zero point
WZn

In the operation above, a workpiece coordinate system once shifted can be preset using G code specification or MDI operation to a workpiece coordinate system displaced by a workpiece zero point offset value from the machine zero point. This is the same as when manual reference position return operation is performed on a workpiece coordinate system that has been shifted. In this example, such G code specification or MDI operation has the effect of returning workpiece coordinate system zero point WZn to the original zero point WZo, and the distance from WZo to Pn is used to represent the current position in the workpiece coordinate system.

Bit 1 (DSE) of parameter No. 2202 specifies whether to preset relative coordinates (RELATIVE) as well as absolute coordinates.

Limitation

- Cutter compensation, tool length compensation, tool offset

By specifying G92.1, cutter compensation, tool length compensation, and tool offset are cancelled.

- Program restart

The workpiece coordinate system preset function is not executed during program restart.

- Prohibited modes

Do not use the workpiece coordinate system preset function when the scaling, coordinate system rotation, programmable image, or drawing copy mode is set.

7.2.7 Automatically Presetting the Workpiece Coordinate System

This function automatically presets the workpiece coordinate system to the position where machine lock is applied, after the machine is operated with machine lock set on and machine lock is released. This means that when this function is used, the current workpiece coordinate system can be restored to the original system, which has the specified offset from the machine zero point. This function is useful in machining after performing a test run with machine lock set ”1”.

This function is automatically activated when the following conditions are satisfied:

1. ACP, bit 7 of parameter No.1200, is set ”1”. (This setting enables this function.)
2. Automatic operation (MDI, memory, or tape) was performed with the all-axis machine lock signal turned ”1” (the each-axis machine lock signals are ignored).
3. When the system is reset (the STP, SPL, and OP signals are all ”0”) after operation, the machine lock signal is ”0” or turned ”0”.

7.3 LOCAL COORDINATE SYSTEM

When a program is created in a workpiece coordinate system, a child workpiece coordinate system can be set for easier programming. Such a child coordinate system is referred to as a local coordinate system.

Format

G52 IP_; Setting the local coordinate system

:

G52 IP 0; Canceling of the local coordinate system

IP_ ; Origin of the local coordinate system

Explanation

By specifying G52IP_;, a local coordinate system can be set in all the workpiece coordinate systems (G54 to G59). The origin of each local coordinate system is set at the position specified by IP_ in the workpiece coordinate system.
When a local coordinate system is set, the move commands in absolute mode (G90), which is subsequently commanded, are the coordinate values in the local coordinate system. The local coordinate system can be changed by specifying the G52 command with the zero point of a new local coordinate system in the workpiece coordinate system.
To cancel the local coordinate system and specify the coordinate value in the workpiece coordinate system, match the zero point of the local coordinate system with that of the workpiece coordinate system.

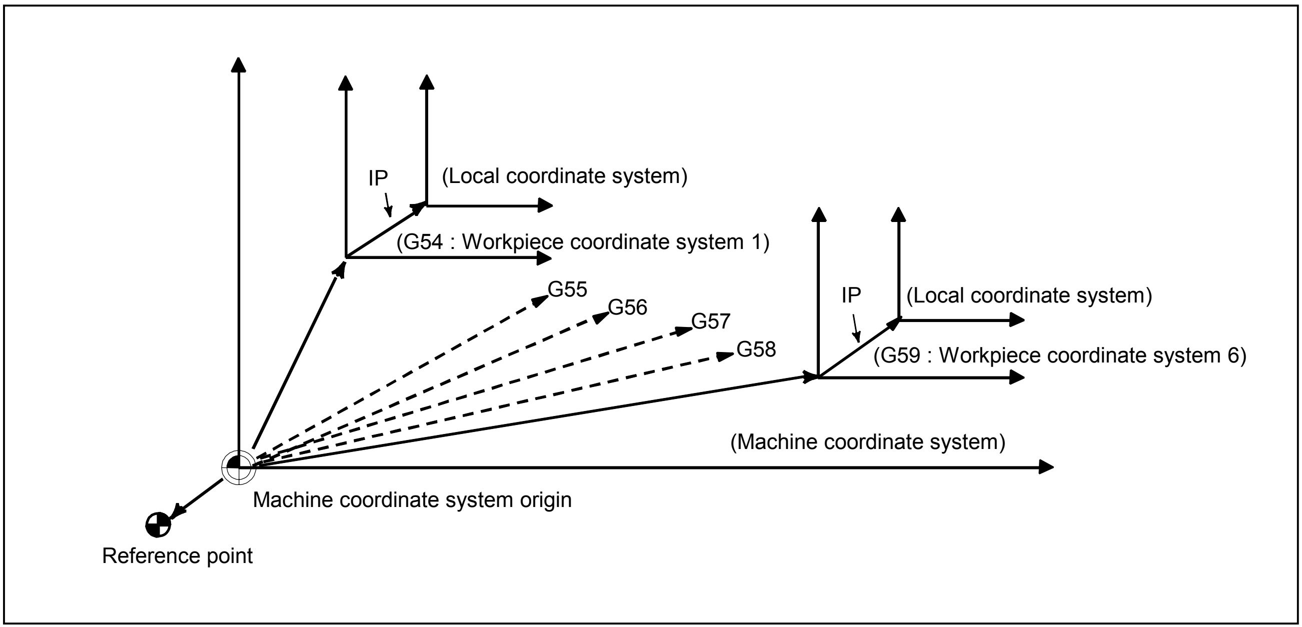

IP
(Local coordinate system)
(G54 : Workpiece coordinate system 1)
G55
G56
G57
G58
IP
(Local coordinate system)
(G59 : Workpiece coordinate system 6)
(Machine coordinate system)
Machine coordinate system origin
Reference point

Fig.7.3 (a) Setting the local coordinate system

⚠ **CAUTION**

1 When an axis returns to the reference point by the manual reference point return function,the zero point of the local coordinate system of the axis matches that of the work coordinate system. The same is true when the following command is issued:
G52 α 0; (α:Axis which returns to the reference point)

2 The local coordinate system setting does not change the workpiece and machine coordinate systems.

3 Upon a reset, the local coordinate system is cancelled.

4 If coordinate values are not specified for all axes when setting a workpiece coordinate system with the G92 command, the local coordinate systems of axes for which coordinate values were not specified are not cancelled, but remain unchanged.

5 In cutter compensation, G52 temporarily cancels offset. However, when an offset vector for cutter compensation remains (when no movement is specified with G40), the local coordinate system is cancelled if even one axis is specified with G52.

6 The G52 command suppresses buffering. By setting bit 2 (O52) of parameter No. 2409 to 1, G52 changes into a G code that does not suppress buffering. In this case, the first move command since the G52 command must be specified in absolute mode.

7.4 PLANE SELECTION

Select the planes for circular interpolation, cutter compensation, and drilling by G-code. The following table lists G-codes and the planes selected by them.

Explanation

Table7.4 Plane selected by G code

G code	Selected plane	Xp	Yp	Zp
G17	Xp Yp plane	X-axis or an axis parallel to it	Y-axis or an axis parallel to it	Z-axis or an axis parallel to it
G18	Zp Xp plane			
G19	Yp Zp plane			

Xp, Yp, Zp are determined by the axis address appeared in the block in which G17, G18 or G19 is commanded. When an axis address is omitted in G17, G18 or G19 block, it is assumed that the addresses of basic three axes are omitted. Parameter No. 1022 is used to specify that an optional axis be parallel to the each axis of the X, Y-, and Z-axes as the basic three axes.

The plane is unchanged in the block in which G17, G18 or G19 is not commanded.
When the power is turned on or the CNC is reset, G17 (XY plane)or G18 (ZX plane) is selected by bits 5 (G18) of parameter 2401.
The movement instruction is irrelevant to the plane selection.

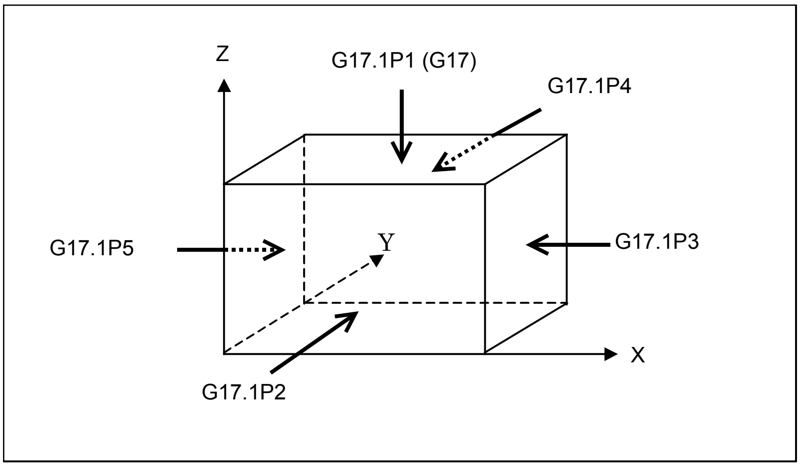
Example

Plane selection when the X-axis is parallel with the U-axis.

G17X_Y_	;	XY plane,
G17U_Y_	;	UY plane
G18X_Z_	;	ZX plane
X_Y_	;	Plane is unchanged (ZX plane)
G17	;	XY plane
G18	;	ZX plane
G17U_	;	UY plane
G18Y_	;	ZX plane, Y axis moves regardless without any relation to the plane.

7.5 PLANE CONVERSION FUNCTION

This function converts a machining program created on the G17 plane in the right-hand Cartesian coordinate system to programs for other planes specified by G17.1Px commands, so that the same figure appears on each plane when viewed from the directions indicated by arrows.



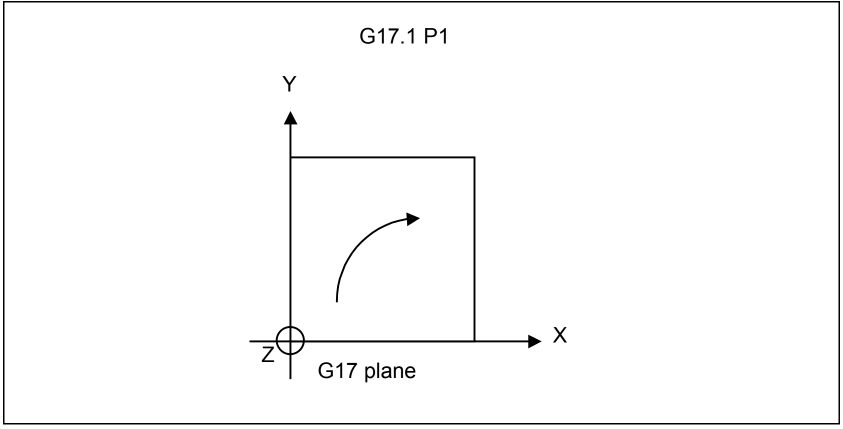
Format

G17.1 P_ ;
P_ : P1 to P5 Plane conversion specification

Explanation

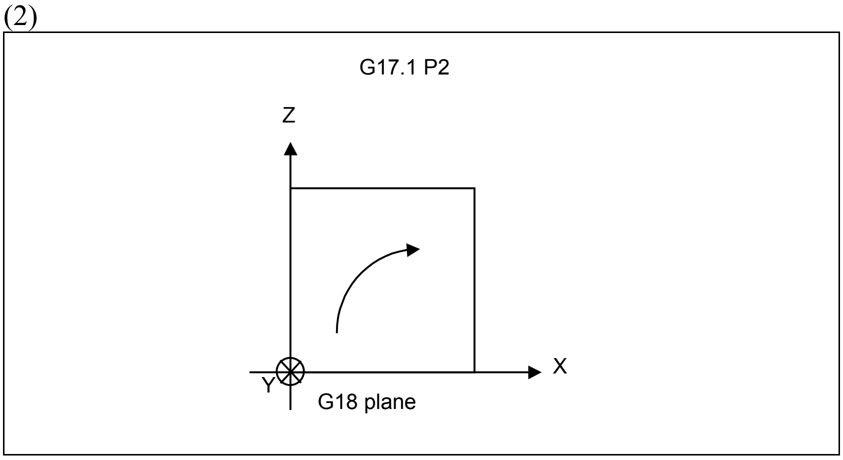
The plane conversion for a machining figure on the G17 plane is performed as follows:

(1)



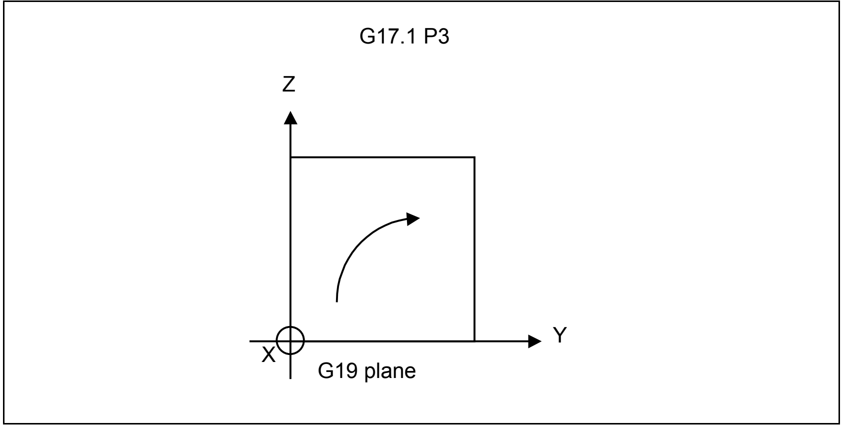
○ indicates that the positive direction of the axis perpendicular to the page is the direction coming out of the page (in this case, the Z-axis perpendicular to the XY plane).

(2)

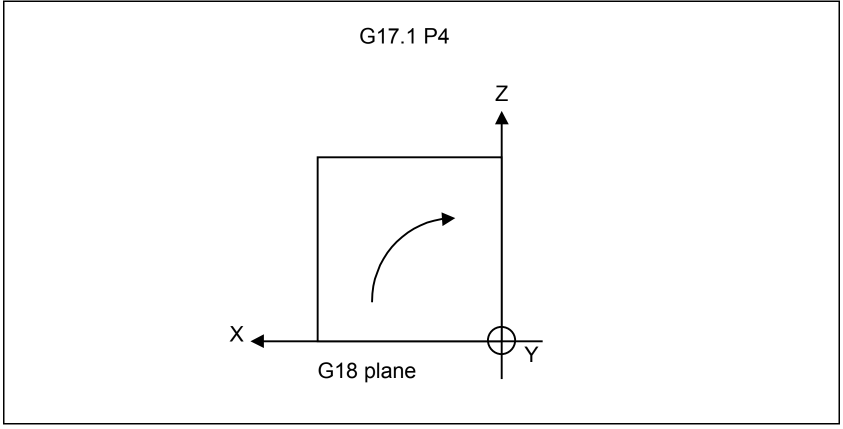


○ indicates that the negative direction of the axis perpendicular to the page is the direction coming out the page (in this case, the Y-axis perpendicular to the XZ plane).

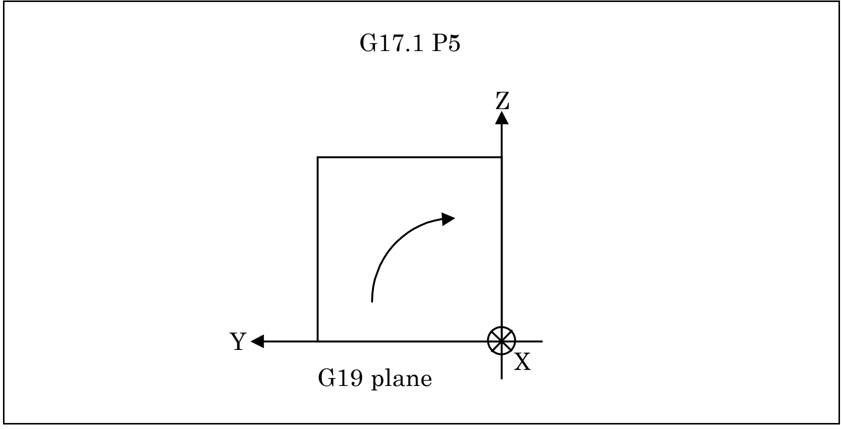
(3)



(4)



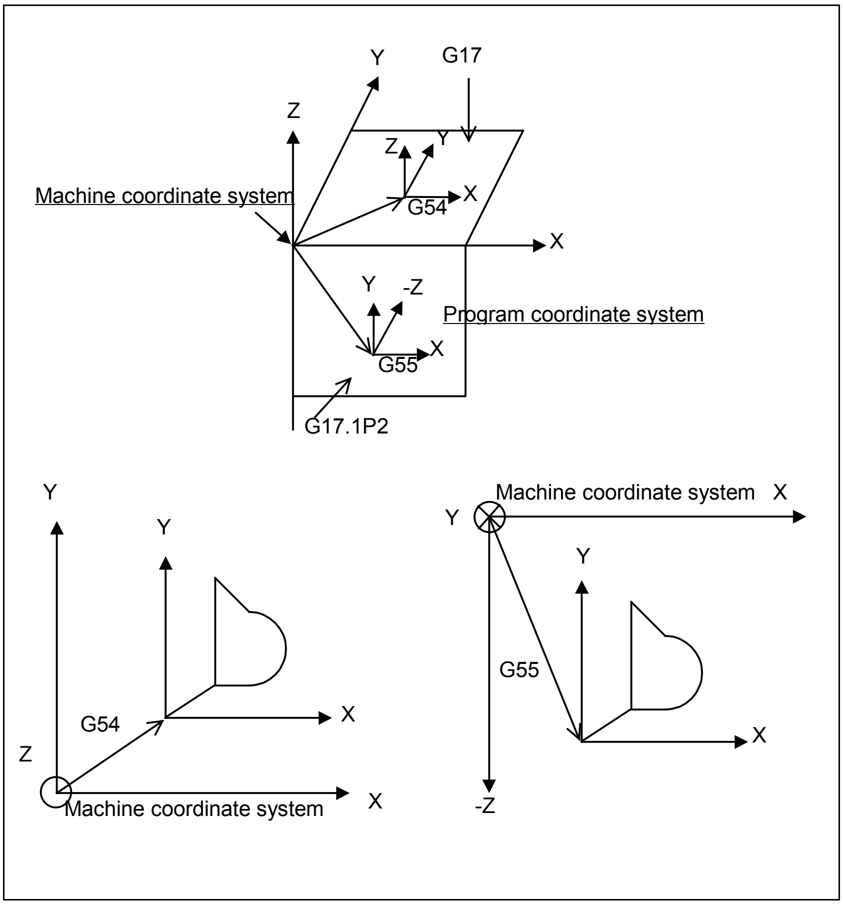
(5)



Program commands on the G17 plane are converted to the following commands by plane conversion:

Command	G17.1P1	G17.1P2	G17.1P3	G17.1P4	G17.1P5
X	X	X	Y	-X	-Y
Y	Y	Z	Z	Z	Z
Z	Z	-Y	-X	Y	-X
G02	G02	G03	G02	G02	G03
G03	G03	G02	G03	G03	G02
I	I	I	J	-I	-J
J	J	K	K	K	K
K	K	-J	I	J	-I
G41	G41	G42	G41	G41	G42
G42	G42	G41	G42	G42	G41
Tool length compensation	+	-	+	+	-
Direction of coordinate rotation	+	-	+	+	-
Direction of the drilling axis	+	-	+	+	-
Plane	G17	G18	G19	G18	G19

Example


Y
G17
Z
Z
Y
Machine coordinate system
G54
X
X
Y
-Z
Program coordinate system
G55
X
G17.1P2
Y
Y
G54
X
Z
Machine coordinate system
X
Machine coordinate system
X
Y
Y
G55
X
-Z

```
O1000 (MAIN PROGRAM)
N10 G91 G28 X0 Y0 Z0
N20 G54
N30 G17
N40 M98 P2000
N50 G55
N60 G17.1 P2
N70 M98 P2000
N80 G91 G28 X0 Y0 Z0
N90 M02

O2000 (SUB PROGRAM)
N2010 G90 G0 Z0
N2020 G0 X0 Y0
N2030 G0 X30. Y20.
N2040 G01 Z-50. F200
N2050 Y90. F500
N2060 X60. Y70.
N2070 G02 Y20. J-25.
N2080 G01 X30.
N2090 G0 Z0
N2100 M99
```

⚠ **CAUTION**

1 Plane conversion can be performed only for commands for the X-, Y-, or Z-axis.

2 Plane conversion cannot be performed for manual operation.

3 Plane conversion cannot be performed for the following commands for moving the tool to a specified position, commands related to the machine coordinate system, and commands for setting a coordinate system:

(1) Automatic reference position return (G28 and G30)
(2) Floating reference position return (G30.1)
(3) Return from the reference position (G29)
(4) Selecting the machine coordinate system (G53)
(5) Stored stroke limit (G22)
(6) Setting the coordinate system (G54 to G59 and G92)
(7) Presetting the workpiece coordinate system (G92.1)
(8) Setting the offset (G10)

Setting bit 0 of parameter No. 2407 performs plane conversion for a G92 or G92.1 command.(Fig.7.5(a) and Table 7.5(a))

4 The current position display on the CRT shows the coordinates after plane conversion. (Table 7.5(b))

5 Plane conversion cannot be performed together with the axis switching function.

6 Specify plane conversion commands after canceling the following modes.

(1) Cutter compensation
(2) Tool length compensation
(3) Canned cycle
(4) Three-dimensional coordinate conversion
(5) Coordinate rotation
(6) Scaling
(7) Programmable mirror image

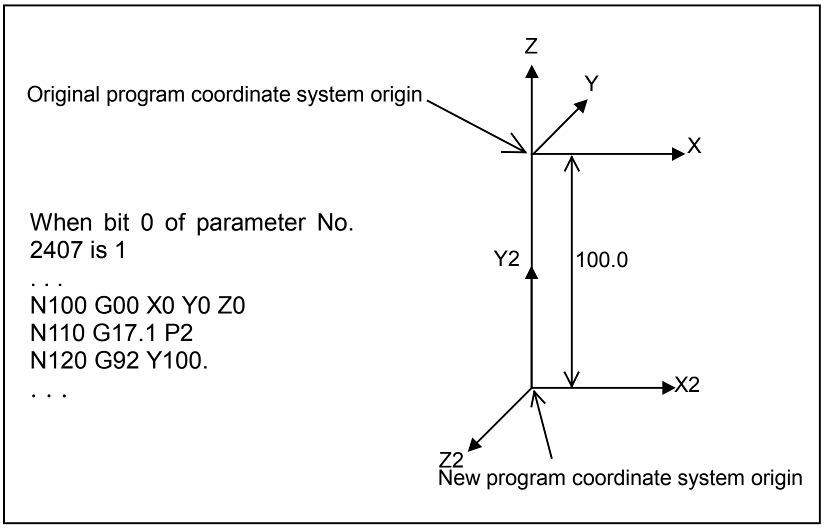
7 Plane conversion cannot be performed for the following commands which control a rotation axis together with the X-, Y-, or Z-axis:

(1) Polar coordinate interpolation
(2) Cylindrical interpolation
(3) Control in normal directions
(4) Exponential interpolation
(5) Circular threading B

8 If a G17, G18, or G19 command is executed during plane conversion, the conversion is disabled and the plane specified by the command is selected.

⚠ **CAUTION**

9 When 1 is set in NCM (bit 7 of parameter No. 2401), resetting the system in the plane conversion mode does not change the mode.



When bit 0 of parameter No. 2407 is 1

```
. . .
N100 G00 X0 Y0 Z0
N110 G17.1 P2
N120 G92 Y100.
. . .
```

Z
Y
X
Original program coordinate system origin
Y2
100.0
X2
Z2
New program coordinate system origin

Fig.7.5 Plane Conversion and G92

Table 7.5 (a) Plane Conversion and G92 (Coordinates)

Command block	Absolute coordinates		
	X	Y	Z
N100	0.	0.	0.
N110	0.	0.	0.
N120	0.	0.	100.

Table 7.5 (b) Current Position Indication

Program command	Absolute coordinates		
	X	Y	Z
G90 G00 X0 Y0 Z0	0.	0.	0.
G17.1 P3	0.	0.	0.
G00 X10. Y20.	0.	10.0	20.0
G01 Z-50. F200	-50.0	10.0	20.0
G02 X50. Y60. I40.	-50.0	50.0	60.0

8 COORDINATE VALUE AND DIMENSION

This chapter contains the following topics.

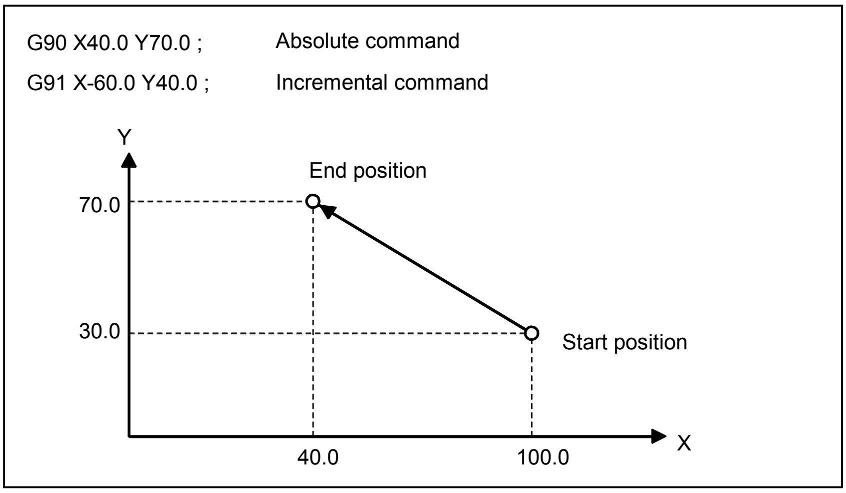
8.1 ABSOLUTE AND INCREMENTAL PROGRAMMING

There are two ways to command travels of the tool; the absolute command, and the incremental command. In the absolute command, coordinate value of the end position is programmed; in the incremental command, move distance of the position itself is programmed.
G90 and G91 are used to command absolute or incremental command, respectively.

Format

Absolute command	**G90 IP_;**
Incremental command	**G91 IP_;**

Examples



```
G90 X40.0 Y70.0 ;        Absolute command
G91 X-60.0 Y40.0 ;       Incremental command
```

8.2 POLAR COORDINATE COMMAND (G15,G16)

The end point coordinate value can be input in polar coordinates (radius and angle).

The plus direction of the angle is counterclockwise of the selected plane first axis + direction, and the minus direction is clockwise.

Both radius and angle can be commanded in either absolute or incremental command (G90, G91). By specifying a radius by either absolute or incremental programming, the user can choose from the following two specification methods:

(1) The origin of the workpiece coordinate system is used as the origin of the polar coordinate system.

(2) The current position is used as the origin of the polar coordinate system.

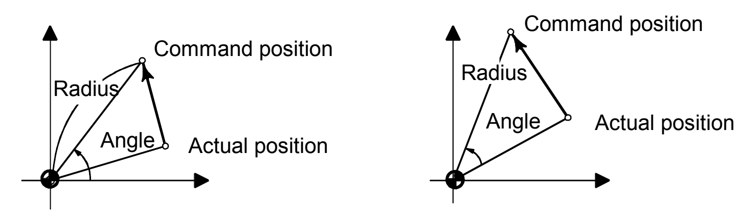
Format

```
GxxG16;      Starting the polar coordinate command
             (polar coordinate mode)
  G00 IP ; ┐
   :       │ Polar coordinate command
   :       ┘
G15;         Canceling the polar coordinate command
             (polar coordinate mode)

 G16 : Polar coordinate command
 G15 : Polar coordinate command cancel
 Gxx : Plane selection of the polar coordinate command
       (G17, G18 or G19)
 IP_ : Specifying the addresses of axes constituting the
       plane selected for the polar coordinate system, and
       their values
       First axis: radius of polar coordinate
       Second axis: radius of polar coordinate
```

Explanation

- When the radius is specified with absolute command

The local coordinate system becomes the center of the polar

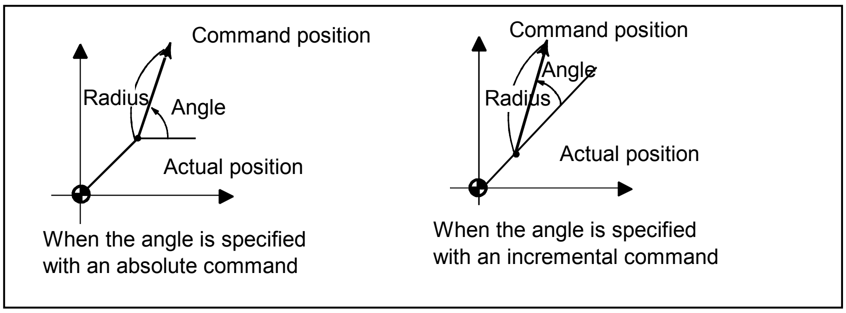


When the angle is specified with an absolute command

When the angle is specified with an incremental command

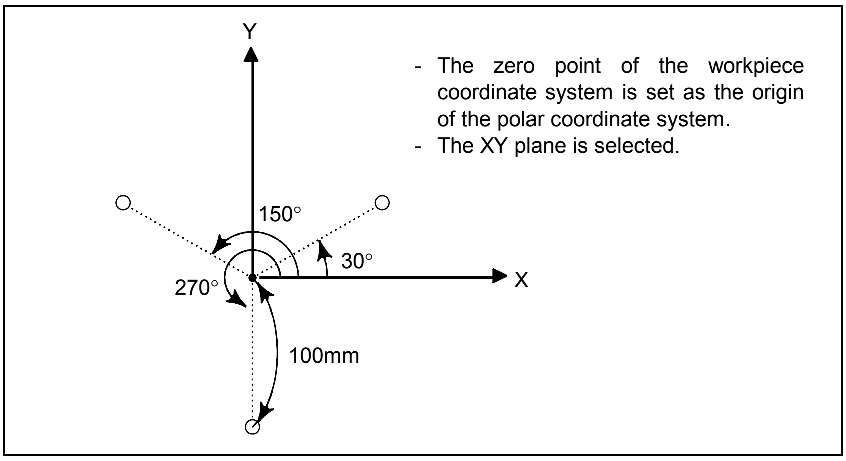
- When the radius is specified with incremental command

The current position is used as the origin of the polar coordinate system.


Command position
Radius
Angle
Actual position
When the angle is specified with an absolute command
Command position
Angle
Radius
Actual position
When the angle is specified with an incremental command

Examples

Bolt hole circle


Y
X
150°
30°
270°
100mm
- The zero point of the workpiece coordinate system is set as the origin of the polar coordinate system.
- The XY plane is selected.

- Specifying angles and a radius with absolute commands

N1 G17 G90 G16 ;

Specifying the polar coordinate command and selecting the XY plane

Setting the zero point of the workpiece coordinate system as the origin of the polar coordinate system

N2 G81 X100.0 Y30.0 Z-20.0 R-5.0 F200.0 ;

Specifying a distance of 100 mm and an angle of 30 degrees

N3 Y150.0 ;

Specifying a distance of 100 mm and an angle of 150 degrees

N4 Y270.0 ;

Specifying a distance of 100 mm and an angle of 270 degrees

N5 G15 G80 ;

Canceling the polar coordinate command

- Specifying angles with incremental commands and a radius with absolute commands

N1 G17 G90 G16;

Specifying the polar coordinate command and selecting the XY plane

Setting the zero point of the workpiece coordinate system as the origin of the polar coordinate system

N2 G81 X100.0 Y30.0 Z-20.0 R-5.0 F200.0 ;

Specifying a distance of 100 mm and an angle of 30 degrees

N3 G91 Y120.0 ;

Specifying a distance of 100 mm and an angle of +120 degrees

N4 Y120.0 ;

Specifying a distance of 100 mm and an angle of +120 degrees

N5 G15 G80 ;

Canceling the polar coordinate command

Limitations

- Specifying a radius in the polar coordinate mode

In the polar coordinate mode, specify a radius for circular interpolation or helical cutting (G02, G03) with R.

- Axes that are not considered part of a polar coordinate command in the polar coordinate mode

Axes specified for the following commands are not considered part of the polar coordinate command:

- Dwell (G04)
- Programmable data input (G10)
- Setting the local coordinate system (G52)
- Converting the workpiece coordinate system (G92)
- Selecting the machine coordinate system (G53)
- Stored stroke check (G22)
- Coordinate system rotation (G68)
- Scaling (G51)
- Programmable mirror image(G50.1)

8.3 INCH/METRIC CONVERSION (G20,G21)

Either inch or metric input can be selected by G code.

Format

G20 ; Inch input
G21 ; mm input

This G code must be specified in an independent block before setting the coordinate system at the beginning of the program.
After the G code for inch/metric conversion is specified, the unit of input data is switched to the least inch or metric input increment of increment system IS-B or IS-C (See II-2.3) The unit of data input for degrees remains unchanged.
The unit systems for the following values are changed after inch/metric conversion:

- Feedrate commanded by F code
- Positional command
- Workpiece zero point offset value
- Tool compensation value
- Unit of scale for manual pulse generator
- Movement distance in incremental feed
- Some parameters

When the power is turned on, the G code is the same as that held before the power was turned off.

⚠ **WARNING**
G20 and G21 must not be switched during a program.

⚠ **CAUTION**
When the input unit is switched between inch input (G20) and metric input (G21), a workpiece origin offset value and tool offset value are automatically converted according to the selected input unit; those values need not be set again.

NOTE
1 When the least input increment and the least command increment systems are different, the maximum error is half of the least command increment. This error is not accumulated.
2 The inch and metric input can also be switched using settings. (See Operation II-12.2.1)

8.4 DECIMAL POINT INPUT/POCKET CALCULATOR TYPE DECIMAL POINT INPUT

Numerals can be input with decimal points. Decimal points can be used basically in numerals with units of distance, speed, and angle.
Following addresses can be commanded.
X, Y, Z, U, V, W, A, B, C, I, J, K, Q, R, F, S

Explanations

There are two types of decimal point notation: calculator-type notation and standard notation.
When calculator-type decimal notation is used, a value without decimal point is considered to be specified in millimeters, inch or deg. When standard decimal notation is used, such a value is considered to be specified in least input increments.
When a decimal point is not given to a velocity command (address F), the minimum unit varies depending on whether the feed mode is feed per minute (G94) or feed per revolution (G95). (For details, see Section 5.3, "Cutting Feed" in II.)
Use parameter DPI (No.2400#0) to select input method; whether to input by pocket calculator type input, or by the usual decimal point input. Values can be specified both with and without decimal point in a single program.

Examples

Program command	Pocket calculator type decimal point programming	Usual decimal point programming
X1000 Command value without decimal point	1000mm Unit : mm	1mm Unit : Least input increment (0.001 mm)
X1000.0 Command value with decimal point	1000mm Unit : mm	1000mm Unit : mm

⚠ **CAUTION**

The appropriate G code should be specified before the numerical values are specified in block. Decimal point may be moved in a certain command.

Example

G20;	Inch input
X1.0 G04;	Because the value X1.0 is not regarded as the number of seconds, but the distance of motion (in inches), X10000G04 is assumed resulting in dwelling for 10 seconds.
G04X1.0;	This is regarded as G04X1000 and dwell is performed for a second.

NOTE

1 A value is rounded off to the number of decimal places of the least input increment.

Example

X1.23456: When the least input increment is 0.001 mm, the value is set to X1.235.
When the least input increment is 0.0001 inch, the value is set to X1.2346.

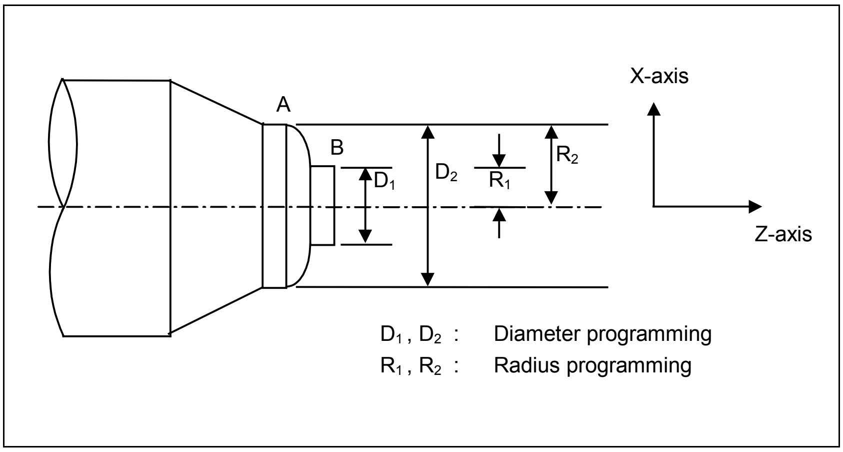
2 The number of digits must not exceed nine. Otherwise, an alarm is issued.
When a decimal input value is converted to an integer according to the least input increment, the number of digits after conversion is also checked.

Example

X0.123456789: The number of digits exceeds nine, resulting in an alarm (PS0003).
X1234567.8: When the least input increment is 0.001 mm, the integer is 1234567800. It goes beyond the 9-digit limit, resulting in an alarm.

8.5 DIAMETER AND RADIUS PROGRAMMING

Since the section of a workpiece to be machined in a lathe is usually circular, the sectional dimensions can be programmed with diameters or radiuses in an NC unit.



Specifying the dimensions of a workpiece using diameters is called diameter programming. Specifying the dimensions using radiuses is called radius programming. DIA(bit3) of parameter 1006 is used to specify whether the dimensions for each axis are specified using diameters or radiuses.
Conform to the conditions in the table below when specifying the dimensions of a workpiece using diameters for the X-axis.

Item	Notes
Commands for the Z-axis	Irrespective of diameters and radiuses.
Commands for the X-axis	Use diameters.
Incremental commands for the X-axis	Use diameters. B → A in the above figure: D2→D1
Coordinate system specification (G92)	Specify the X coordinates using diameters.
Radius commands (R, I, J, K) for circular interpolation	Use radiuses.
Feedrate along the X-axis	Radius change/rev Radius change/min
Displaying X coordinates	Use diameters

⚠ **CAUTION**
The following does not describe the details of diameter and radius programming. In short, the gradations for the X-axis are defined in diameters during diameter programming, and defined in radiuses during radius programming.

8.6 PROGRAMMABLE SWITCHING OF DIAMETER/RADIUS SPECIFICATION

Assume that diameter or radius specification has been selected for each controlled axis by using bit 3 (DIA) of parameter No. 1006.
This function allows the use of a G code to switch between diameter specification and radius specification for programmed axis commands only.
When bit 5 (PDC) of parameter No. 1001 is set, the G code can be used to specify whether programmed axis commands are to be specified in diameters or radii.

Format

G10.9 IP_ ;

IP: Axis address for which switching between diameter specification and radius specification is performed. IP is followed by one of the following values:

0 : Radius specification

1 : Diameter specification

Specifying G10.9 rewrites bit 3 (PDA) of parameter No. 1009.

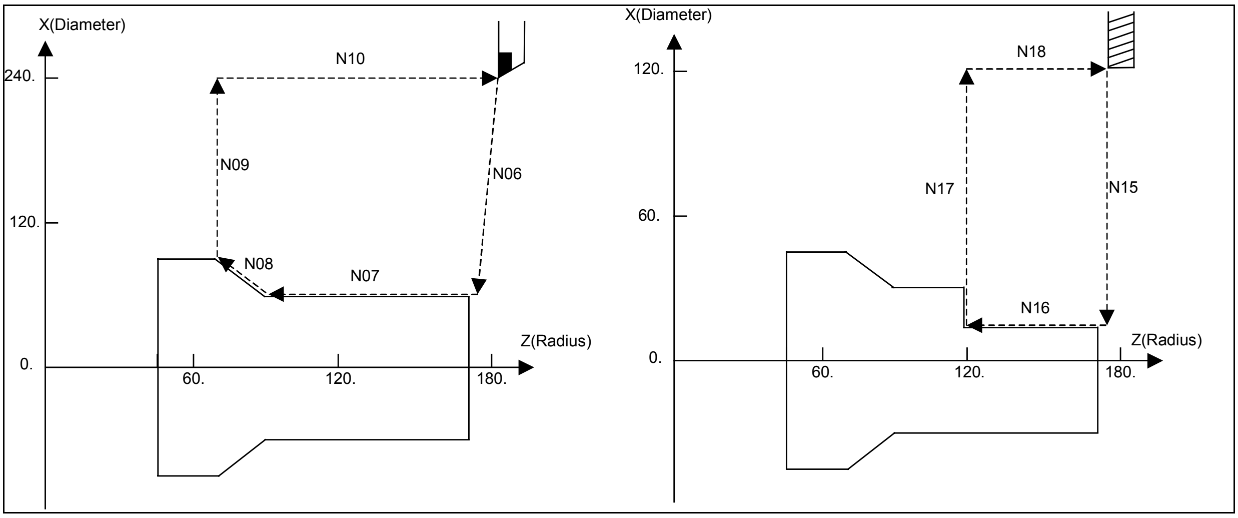
⚠ **CAUTION**

1 After IP, specify a value without a decimal point.
2 The block in which G10.9 is specified must not contain any other command.
3 Parameters and offsets must be set according to the setting of bit 3 (DIA) of parameter No. 1006.
4 Positions are indicated according to the setting of bit 3 (PDA) of parameter No. 1009.
5 When G10.9 is specified during parallel axis control, the setting of bit 3 (PDA) of parameter No. 1009 for the master and slave axes is changed regardless of the parallel axis parking on/off state.
6 G10.9 is ignored during background graphic display.
7 The scale indication on the graphic screen is displayed according to the setting of bit 3 (DIA) of parameter No. 1006.
8 Axis movements by manual numeric commands G00 and G01 are made according to the setting of bit 3 (PDA) of parameter No. 1009.
9 The current positions and workpiece coordinate preset values are set according to the setting of bit 3 (PDA) of parameter No. 1009.

Example

Program

```
O0010
N01 T1001
N02 G10.9 X1 Z0
N03 S200
N04 M03
N05 G00 G90 X240. Z180.
N06 X60. Z170.
N07 G01 Z90.
N08 X90. Z70.
N09 G00 X240.
N10 Z180.
N11 M05
N12 T2002
N13 G10.9 X0 Z0
N14 G00 G90 X120. Z175.
N15 X15.
N16 G01 Z120.
N17 G00 X120.
N18 Z175.
N19 M30
```



9 SPINDLE SPEED FUNCTION (S FUNCTION)

The spindle speed can be controlled by specifying a value following address S.

9.1 SPECIFYING THE SPINDLE SPEED WITH A CODE

When a value is specified after address S, the code signal and strobe signal are sent to the machine to control the spindle rotation speed.
A block can contain only one S code.
Refer to the appropriate manual provided by the machine tool builder for details such as the number of digits in an S code or the execution order when a move command and an S code command are in the same block.

9.2 CONSTANT SURFACE SPEED CONTROL (G96, G97)

Specify the surface speed (relative speed between the tool and workpiece) following S. The spindle is rotated so that the surface speed is constant regardless of the position of the tool.

Format

- Constant surface speed control command

G96Sxxxxx ;
↑ Surface speed (m/min or feet/min)

Note :This surface speed unit may change according to machine tool builder's specification.

- Constant surface speed control cancel command

G97Sxxxxx ;
↑ Spindle speed (min^{-1})

Note :This surface speed unit may change according to machine tool builder's specification.

- Constant surface speed controlled axis command

G96Pα ; P0 : Axis set in the parameter (No. 5884)
P1 : X axis, P2 : Y axis, P3 : Z axis, P4 : U axis
P5 : V axis, P6 : W axis, P7 : A axis, P8 : B axis
P9 : C axis

- Clamp of maximum spindle speed

G92S_ ;
The maximum spindle speed (min^{-1}) follows S.

Explanations

- Constant surface speed control command (G96)

G96 (constant surface speed control command) is a modal G code. After a G96 command is specified, the program enters the constant surface speed control mode (G96 mode) and specified S values are assumed as a surface speed. A G97 command cancels the G96 mode. S (surface speed) commands in the G96 mode are assumed as S = 0 (the surface speed is 0) until M03 (rotating the spindle in the positive direction) or M04 (rotating the spindle in the negative direction) appears in the program.

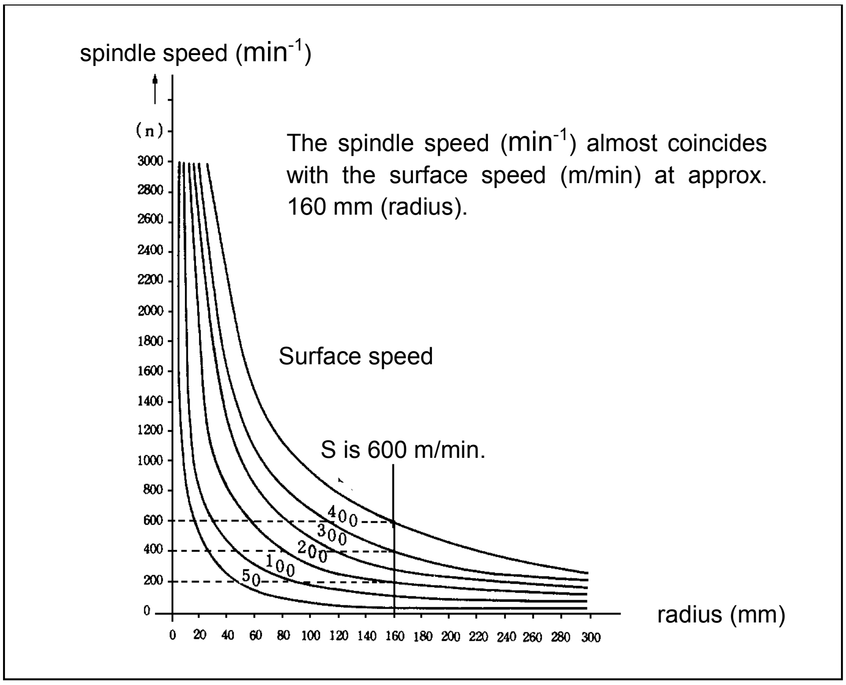


Fig.9.2 (a) Relation between workpiece radius, spindle speed and surface speed

- Clamp of maximum spindle speed(G92)

With the specification of G92, the maximum spindle speed signal is output.

In general, the output signal is used for spindle speed clamping so that the maximum spindle speed is not exceeded during constant surface speed control. The method of clamping may vary from one machine tool builder to another. So, for details, refer to the manual provided by the machine tool builder.

- Constant surface speed control axis

The constant surface speed controlled axis is selectable by a parameter (No. 5844) a programmed command. When it is specified by the program, specify the axis for the constant surface speed control by address P of the block of G96. When address P is omitted, or if P0 is specified, an axis set by parameter (No. 5844) becomes effective.

Table 9.2 (a) Constant Surface Speed Control Axis

Value of P	Axis subject to constant surface speed control
1	X
2	Y
3	Z
4	U
5	V
6	W
7	A
8	B
9	C

- Setting the workpiece coordinate system for constant surface speed control

To execute the constant surface speed control, it is necessary to set the work coordinate system , and so the coordinate value at the center of the rotary axis, for example, Z axis, (axis to which the constant surface speed control applies) becomes zero.

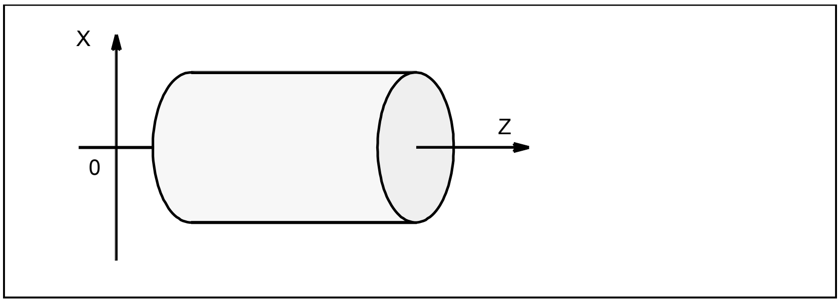


Fig.9.2 (b) Example of the workpiece coordinate system for constant surface speed control

- Value of S specified in G96 mode

The S value specified in G96 mode is stored even after G97 mode has turned

G96 S50	: 50 m/min or 50 feet/min
G97 S1000	: 1000 min^{-1}
G96 X3000	: 50 m/min or 50 feet/min

- Speed when G97 mode is resumed

If S (min^{-1}) is not specified in a G97 block when the mode switches from G96 mode to G97 mode, the last speed to be used at the end of G96 mode is used as S in G97 mode.

N10 G97 S800	: 800 min^{-1}
N20 G96 S100	: 100 m/min or 100 feet/min
N30 G97	: X min^{-1}

X represents the speed used in the previous mode. This means that the spindle speed is maintained upon a transition from G96 mode to G97 mode.

If S (min^{-1}) is not specified in a G96 block when the mode initially switches from G97 mode to G96 mode, S = 0 m/min is used.

- Constant surface speed control in rapid traverse (G00)

In rapid traverse, the surface speed is constantly being calculated as the tool position changes. When bit 6 (RSC) of parameter No. 5602 is set to 1, however, the surface speed is calculated from the position of the end point of the block

- Constant surface speed control when machine lock is used

With machine lock, the constant surface speed is calculated according to a change of the coordinate values of the axis to which the constant surface speed control is applied, even if the machine tool does not move.

- Position coder-less feed per revolution and constant surface speed control

These functions are enabled when bit 6 (FPR) of parameter No. 2405 is set to 1.

On a machine on which no position coder is installed, feed per revolution is enabled assuming the specified spindle speed signals RO0s to RO15s to be the spindle speed, even if constant surface speed control is applied

Example) A diameter is specified for the X-axis

```
N1 G90 X50. Z10.
N2 G96 G95 S12 :      Starts constant surface speed control and
                      feed per revolution
N3 G01 X20. Z30. F10.
N4 Z50.
N5 G97 S200            : Cancels constant surface speed control.
N6 G90 G00 Z200.
N7 M05
N8 M30
```

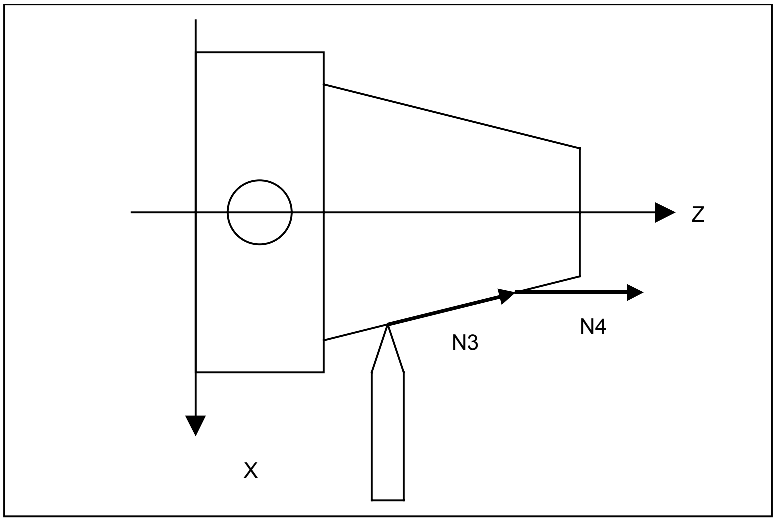


Fig.9.2 (c) Position Coder-less Feed per Revolution

N2 of the above program specifies the constant surface speed control command (G96), a surface speed of 12 m/min, and a feed per revolution (G96). During positioning up to X = 20.0 in N3, the CNC changes the command spindle speed from 76.4 min^{-1} to 191.0 min^{-1}.

On the other hand, the feedrate for feed per revolution conforms to the varying command spindle speed signals to produce movement along a feed axis. In this example, the actual speed in N3 is 764 to 1910 mm/min.

However, the command spindle speed is clamped with maximum spindle speed signals MR0s to MR15s.

Limitation

- Constant surface speed control for threading

The constant surface speed control is also effective during threading. If face threading or taper threading is performed in G96 mode, however, the spindle speed changes, and tool feedrate also changes accordingly, resulting in a change in the servo delay. This can produce an incorrect thread pitch.

9.3 SPINDLE POSITIONING FUNCTION

Turning is described as follows: The spindle connected to the spindle motor is rotated at a certain speed. As a result, the workpiece fixed to the spindle is rotated, and turning is performed. The spindle positioning function is described as follows: The spindle connected to the spindle motor is rotated up to a certain angle to position the workpiece fixed to the spindle at a certain angle. The use of this function enables a workpiece to be drilled at any position.

The operations performed by the spindle positioning function are classified as follows:

1. Operation from canceling spindle rotation mode to changing to spindle positioning mode
2. Positioning the spindle in spindle positioning mode
3. Operation from canceling spindle positioning mode to changing to spindle rotation mode

- spindle control system

The following figure shows the spindle control system.
When the spindle is rotated for turning (referred to hereinafter as spindle rotation mode), the value corresponding to the spindle speed is input from the spindle control unit to the D/A converter. When the spindle is positioned (referred to hereinafter as spindle positioning mode), the specified travel distance is input to an error counter and converted into the velocity command for the spindle motor via the D/A converter. The position of the spindle is then detected by the position coder installed in the spindle.

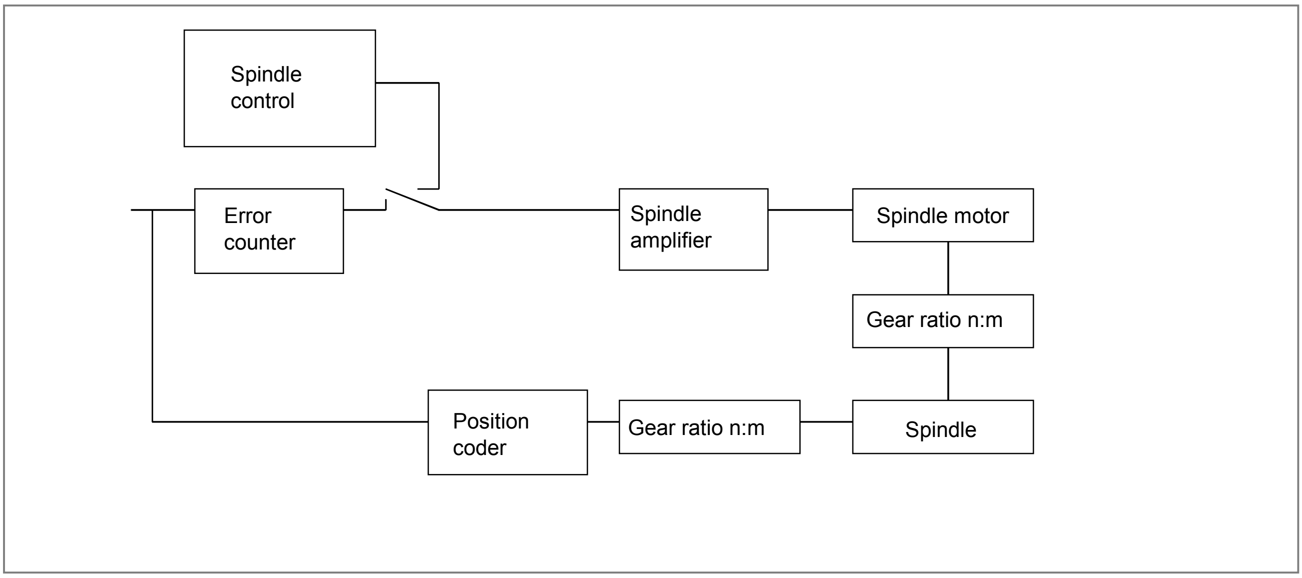


Fig.9.3 spindle control system

- Least command increment(detection unit)

The table below indicates the least command increment for a spindle positioning axis.

Table9.3 Least command increment

Gear ratio (1:p) of spindle to position coder	Least command increment (detection unit: deg)
1:1	0.088 (1×360/4096)
1:2	0.176 (2×360/4096)
1:4	0.352 (4×360/4096)
1:8	0.703 (8×360/4096)
:	:
1:N	(N×360/4096)

9.3.1 Spindle Positioning

Explanation

There are two programming methods: indexing at an arbitrary angle, and indexing at a semi-fixed angle.

- Indexing at a semi-fixed angle with an M code

This is specified with a two-digit numeric value following address M. Up to six codes of Mα to Mβ can be specified. α is preset with parameter (No.5896,5897). The following table shows the correspondence between Mα to Mβ and indexing angles. γ is preset with parameter 5898. Note that a rotation direction can be specified using the sign of γ.

Table9.3.1 M Codes for Positioning at Semi-Fixed Angles

M code	Indexing angle (deg)	Example (When γ= 30°)
Mα	γ	30°
M(α+1)	2γ	60°
M(α+2)	3γ	90°
M(α+3)	4γ	120°
M(α+4)	5γ	150°
M(α+5)=Mβ	6γ	180°

- Positioning to an arbitrary angle by an axis address

An index position is specified using an axis address followed by a numeric value. (A signed value can be specified.) Either absolute or incremental programming can be used. Specify an axis address in G00 mode. The position of the decimal point represents the position of "degrees."
Example) C35.0 = C35 degrees

- Program zero-point

The position after orientation is assumed to be a program zero-point. The program zero-point can, however, be changed by setting a coordinate system (G92).

- Absolute and incremental programming

For indexing at a semi-fixed angle based on an M code, incremental programming must always be used. For indexing at an arbitrary angle based on an axis address, a command must be specified.

- Feedrate

The rapid traverse rate specified in parameter No. 1420 is used as a feedrate for positioning, and linear acceleration/deceleration is applied.

9.3.2 Orientation

Orientation must be performed before:

- the spindle is positioned (indexed) for the first time after the spindle is used in normal machining.
- the positioning of the spindle is suspended.

Explanations

Orientation can be executed by specifying an M code set in parameter No. 5680.

Upon the completion of orientation, the point at which orientation is completed is regarded as the program origin of the spindle. The coordinates along the coordinate axes are regarded as being 0.

When reference position return is not required at orientation (for example, when only incremental indexing from the current position is to be performed when no start position is specified), reference position return at orientation can be omitted by setting bit 0 (ZRN) of parameter No. 1005.

This means that, when the M code for orientation is specified, only spindle control mode switching is performed, after which the operation terminates; reference position return is not performed. In this case, the pitch error compensation function is disabled. In addition,the reference position return completion signal is not set to 1.

- Rotation direction

The rotation direction of the spindle at orientation is set using the following parameters:

1. For an analog spindle
   Bit 0 (OMI) of parameter No. 5808 is used for setting.
2. For a serial spindle
   Bit 4 (SVO) of parameter No. 3000 is used for setting.

- Operation and feedrate

1. For an analog spindle
   When orientation is specified, movement starts at the rapid traverse rate set in parameter No. 5977. Then, upon the detection of the one-rotation signal from the position coder, the feedrate is automatically reduced to the FL feedrate set in parameter No. 5979 to perform reference position return.
2. For a serial spindle
   The orientation feedrate depends on the spindle. For an analog spindle, the rapid traverse rate is reduced to an FL feedrate to stop at the orientation position. With a serial spindle, however, the command immediately stops the spindle at the orientation position.

- Grid shift

The grid shift function is used to shift the orientation stop position.

1. For an analog spindle
   Parameter No. 5980 is used for setting.
2. For a serial spindle
   Parameter No. 3073 is used for setting.

9.3.3 Canceling the Spindle Positioning Mode

Explanation

The mode can be switched from spindle positioning mode to spindle rotation mode (with positioning cancelled) by specifying the M code set in parameter No. 5681.

Positioning mode is also cancelled in any of the following cases:

1. A servo alarm is issued
2. A spindle alarm is issued.
3. The spindle is stopped during orientation due to a reset or alarm.
4. The spindle is stopped upon the cancellation of positioning due to a reset or alarm.
5. When a reset is caused: Depends on the setting of bit 4 (IOR) of parameter No. 5605.
6. When an emergency stop is caused: Depends on the setting of bit 1 (IOE) of parameter No. 5809.

NOTE

1 An M code related to spindle positioning must be specified singly in a block. In the same block, no other commands may be specified.

2 An axis address for spindle positioning must be specified singly in a block. In the same block, no commands other than those below may be specified:
G00, G90, G91, G92, G10

3 The spindle cannot be positioned manually.

4 The program or block cannot be restarted for spindle indexing. To restart it, use MDI commands

5 A feed hold function is ineffective during spindle positioning.

6 The dry run function, machine lock function, and auxiliary function lock function are ineffective during spindle positioning.

7 For a spindle positioning axis, the stored stroke limit check function is disabled.

8 A spindle positioning axis does not allow axis removal.

9 The pitch error compensation function is disabled with a serial spindle. With an analog spindle, only the stored pitch error compensation function is enabled.

10 In orientation, a check for all-axis interlock/axis-by-axis interlock is made only at the start of a block. Any signal entered during block execution is ignored. Axis-by-axis interlock is enabled or disabled by setting bit 0 (AIT) of parameter No. 5809.

NOTE

11 The spindle positioning function is enabled only when the number of position coder pulses is 4096, and the gear ratio between the spindle and position coder is as follows:

1 :2^n n : Integer greater than 0

12 For a spindle positioning axis, CMR = 1 and DMR = 4 are used at all times.

13 For a spindle positioning axis, set the following parameters as axis attributes:

1) The increment system is IS-B (0.001 deg)
Bit 0 (ISA) of parameter No. 1012 = 0
Bit 1 (ISC) of parameter No. 1012 = 0
Bit 2 (ISD) of parameter No. 1012 = 0
Bit 3 (ISE) of parameter No. 1012 = 0

2) The grid method is used as the reference position return method.
Bit 1 (ZMG) of parameter No. 1005 = 0

3) In automatic reference position return (G28), the tool returns to the reference position by rapid traverse.
Bit 2 (ALZ) of parameter No. 1005 = 0

4) In manual reference position return, the workpiece coordinate system is always preset.
Bit 3 (PLZ) of parameter No. 1005 = 1

5) Axis (rotation axis) for which inch/metric switching is not required
Bit 0 (ROT) of parameter No. 1006 = 1

6) The machine coordinate system for stroke checking and automatic reference position return is of the rotation axis type.
Bit 1 (ROS) of parameter No. 1006 = 1

7) The machine coordinate system for stored pitch error compensation is of the rotation axis type.
Bit 2 (ROP) of parameter No. 1006 = 1

8) Radius specification is used for specifying the amount of travel on each axis.
Bit 3 (DIA) of parameter No. 1006 = 0

9) Command multiplication ratio = 1
Parameter No. 1820 = 2

10) The machine position is indicated in degrees.
Bit 1 (MCN) of parameter No. 2203 = 1

11) The amount of movement per rotation axis revolution is 360 degrees.
Parameter No. 1260 = 360000

9.4 SPINDLE SPEED FLUCTUATION DETECTION (G26, G25)

General

If the actual spindle speed becomes lower or higher than that specified because of the condition of the machine, an overheat alarm (SP0242) is issued, and spindle speed fluctuation detection alarm signal SPAL is output. This signal can be used to prevent the guide bushing from burning out.

Format

- Enabling of spindle speed fluctuation detection

G26 P_ Q_ R_ I_ ;

P : Time from the point a new spindle rotation command (S command) is supplied or the G26 command is issued until spindle speed fluctuation detection is actually started.
Units: msec
If the specified speed is reached within time P, spindle speed fluctuation detection starts at that point.

Q: Permissible error assumed to determine whether the specified speed is reached
Units: 1.0% if FLR (bit 1 of parameter No. 5808) is equal to 0; and
0.1% if FLR (bit 1 of parameter No. 5808) is equal to 1.
If the speed falls within the range of the specified spindle speed +/- the permissible error, the specified spindle speed is assumed to have been reached, and spindle speed fluctuation detection starts.

R: Spindle speed fluctuation that does not cause a spindle speed fluctuation detection alarm to be issued
Units: 1.0% if FLR (bit 1 of parameter No. 5808) is equal to 0; and
0.1% if FLR (bit 1 of parameter No. 5808) is equal to 1.

I: Permissible spindle speed fluctuation width that does not cause a spindle speed fluctuation detection alarm to be issued
Units: min^{-1}

G26 places the system in spindle speed fluctuation detection enabled mode and sets the P, Q, R, and I command addresses in parameters Nos. 5071, 5702, 5721, and 5722. The parameter numbers corresponding to the command addresses are as follows:

Command address	Parameter number
Q	5701
R	5702
I	5721
P	5722

If any of the P, Q, R, and I command addresses is omitted, spindle speed fluctuation detection is performed with the value set in the corresponding parameter (No. 5071, 5702, 5721, or 5722).

- Disabling spindle speed fluctuation detection

```
G25;
```

G25 places the system in spindle speed fluctuation detection disabled mode.

Specifying G25 does not change the settings of parameters Nos. 5071, 5702, 5721, and 5722.

If G26 (bit 4 of parameter No. 2409) is "0," the system enters spindle speed fluctuation detection disabled mode (G25) when the power is turned on or a reset is performed.

Explanation

- Method for detecting spindle speed fluctuation

If a difference between the actual spindle speed and the specified spindle speed becomes larger than the allowable fluctuation width specified in the address R and I commands in a G26 command block or parameter Nos. 5702 and 5721 (if the following two conditions are satisfied), an alarm is issued to indicate that the fluctuation has become higher than a permissible level.

(1) $|Sc - Sa|>=Sr$

(2) $|Sc - Sa|>=Si$

Sc: Specified spindle speed

Sa: Actual spindle speed

Sr: Spindle speed fluctuation range calculated from spindle speed fluctuation ratio (address R command in a G26 block or parameter No. 5702)

If parameter FLR (bit 1 of parameter No. 5808) is 0

$Sr = Sc * r / 100$

If parameter FLR (bit 1 of parameter No. 5808) is 1

$Sr = Sc * r / 1000$

Si: Permissible range of fluctuation that does not cause a spindle speed fluctuation detection alarm to be output (address I command in a G26 block or parameter (No. 5721) setting)

NOTE

Even when the conditions for issuing an alarm related to spindle speed fluctuation detection have not been satisfied in spindle speed detection enabled mode (G26), a spindle speed fluctuation detection overheat alarm is issued if:

- A spindle speed has been specified, and the actual spindle speed remains at 0 min^{-1} for at least one second.

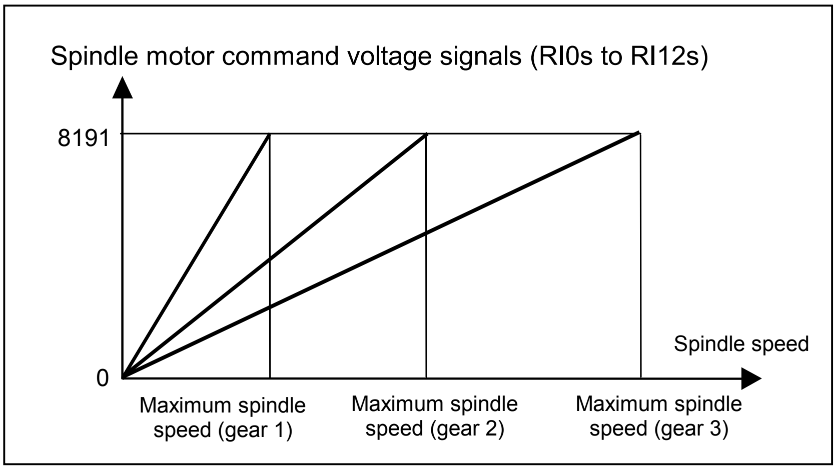
- Specified spindle speed

A spindle speed to be specified for spindle speed fluctuation detection is determined as described below.

For the spindle motor and each spindle gear, the spindle speed corresponding to the maximum output is set in parameter Nos. 5621 to 5628.

The spindle speed to be specified for an analog spindle is determined from the maximum spindle speed parameters (parameter Nos. 5621 to 5628) selected using the spindle gear selection signals (GS1s to GS4s) and the spindle motor command voltage signals (RI0s to RI12s).

The spindle speed to be specified for a serial spindle is determined from the maximum spindle speed parameters (parameter Nos. 5621 to 5628) selected using the serial spindle clutch/gear signals (CTH1s and CTH2s) and the spindle motor command voltage signals (RI0s to RI12s).



The following tables list the relationships among the spindle gear select signals (GS1s to GS4s), serial spindle clutch/gear signals (CTH1s and CTH2s), and maximum spindle speed parameters (Nos. 5621 to 5628).

1) Serial spindle

CTH1s	CTH2s	Maximum spindle speed parameter
0	0	No.5621
0	1	No.5622
1	0	No.5623
1	1	No.5624

2) Analog spindle

GS4s	GS2s	GS1s	Maximum spindle speed parameter
0	0	0	No.5621
0	0	1	No.5622
0	1	0	No.5623
0	1	1	No.5624
1	0	0	No.5625
1	0	1	No.5626
1	1	0	No.5627
1	1	1	No.5628

- Actual spindle speed

The actual spindle speed is calculated from the feedback pulse received from the position coder mounted on the spindle.

- Conditions for starting spindle speed fluctuation detection

If the specified spindle speed changes after G26 is issued or in spindle speed fluctuation detection enabled mode, spindle speed fluctuation detection begins when one of the following conditions is satisfied.

(1) The time specified in parameter No. 5722 elapses since a change in the specified spindle speed after G26 is issued or in spindle speed fluctuation detection enabled mode

(2) The actual spindle speed and specified spindle speed permission ratio becomes lower than or equal to the value specified in parameter No. 5701, so it is assumed that the specified speed has been reached.

Actual spindle speed and specified spindle speed permission ratio
= (1 - Actual spindle speed / Specified spindle speed) * 100

NOTE

Even if a condition for starting spindle speed fluctuation detection is satisfied, spindle speed fluctuation detection is not started under any of the following conditions.

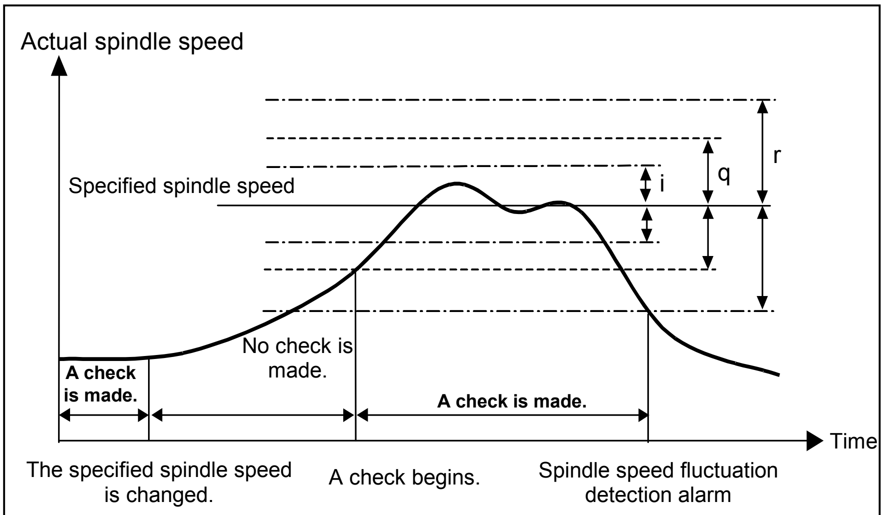
1 The machine is in spindle speed fluctuation detection disabled mode (G25).

2 The specified spindle speed is 0 min^{-1}.

3 A search due to program restart is under way.

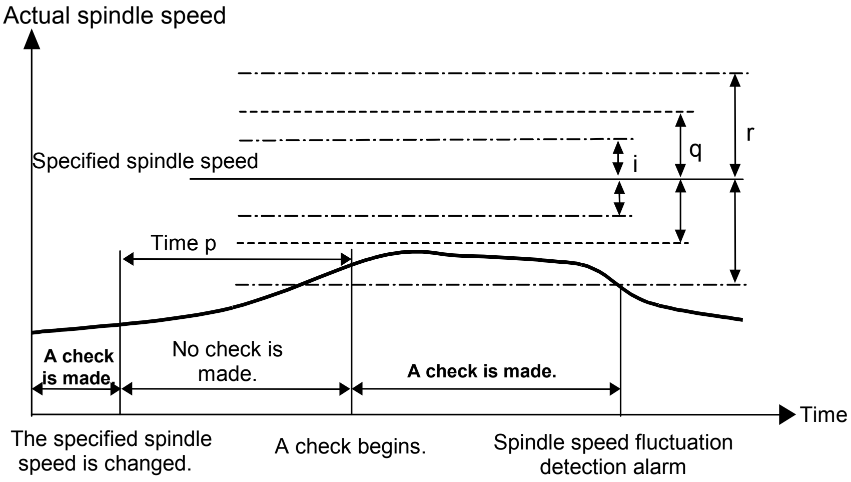
- Examples of alarms issued for spindle speed fluctuation detection

1) Example where an alarm is issued after the specified spindle speed is reached



i : Fluctuation range not triggering an alarm (parameter No. 5721)
q : (Permission ratio at which the attainment of the specified spindle speed is assumed (parameter No. 5701)) x (specified spindle speed)
r : Fluctuation ratio not triggering an alarm
(parameter No. 5702)) x (specified spindle speed)

2) Example in which an alarm is issued before the specified spindle speed is attained



i : Fluctuation range not triggering an alarm (parameter No. 5721)
q : (Permission ratio at which the attainment of the specified spindle speed is assumed (parameter No. 5701)) x (specified spindle speed)
r : Fluctuation ratio not triggering an alarm
(parameter No. 5702)) x (specified spindle speed)
p : Time from when the specified spindle speed is changed until a check begins (parameter No. 5722)

- System with more than one spindle

In a system with more than one spindle, spindle speed fluctuation detection is performed for the spindle described below.

1) If the system has no spindle control switching function
   Spindle speed fluctuation detection is performed for the first spindle.
2) If the system has a spindle control switching function
   Spindle speed fluctuation detection is performed for the spindle selected in spindle control switching.
   Spindle speed fluctuation detection is not performed for a spindle that is not selected in spindle control switching.

10 TOOL FUNCTION (T FUNCTION)

Two tool functions are available. One is the tool selection function, and the other is the tool life management function.

10.1 TOOL SELECTION FUNCTION

By specifying an up to 10-digit numerical value following address T, tools can be selected on the machine.
One T code can be commanded in a block.
Refer to the machine tool builder's manual for the number of digits commandable with address T and the correspondence between the T codes and machine operations.
When a move command and a T code are specified in the same block, the commands are executed in one of the following two ways:

(i) Simultaneous execution of the move command and T function commands.

(ii) Executing T function commands upon completion of move command execution.

The selection of either (i) or (ii) depends on the machine tool builder's specifications.

10.2 TOOL LIFE MANAGEMENT FUNCTION

Tools are classified into various groups, with the tool life (time or frequency of use) for each group being specified.
The function of accumulating the tool life of each group in use and selecting and using the next tool previously sequenced in the same group, is called the tool life management function.

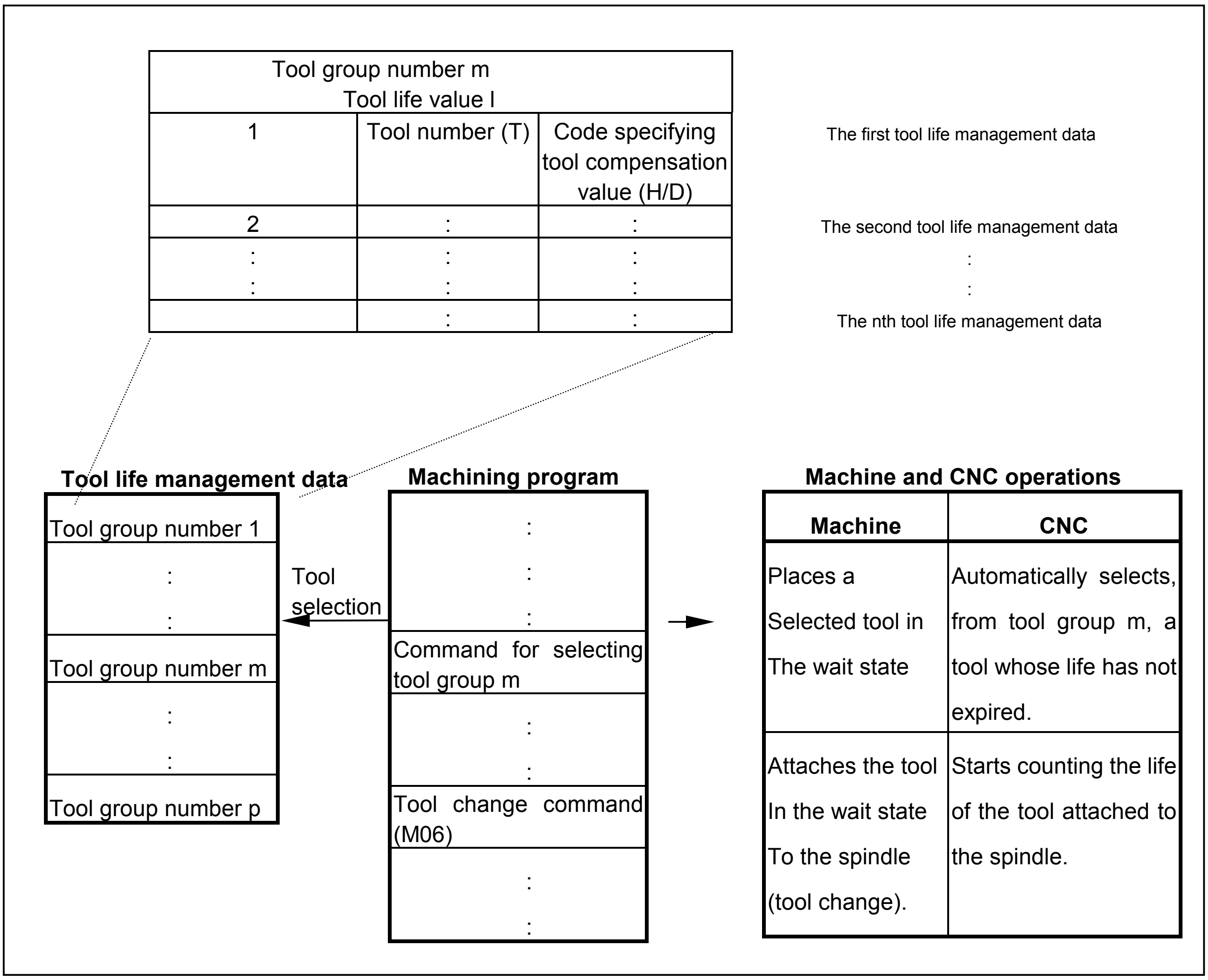

Tool group number m
Tool life value l
1
Tool number (T)
Code specifying tool compensation value (H/D)
The first tool life management data
2
The second tool life management data
The nth tool life management data
Tool life management data
Tool group number 1
Tool group number m
Tool group number p
Tool selection
Machining program
Command for selecting tool group m
Tool change command (M06)
Machine and CNC operations
Machine
CNC
Places a Selected tool in The wait state
Automatically selects, from tool group m, a tool whose life has not expired.
Attaches the tool In the wait state To the spindle (tool change).
Starts counting the life of the tool attached to the spindle.

Fig.10.2 Tool Selection by Machining Program

10.2.1 Tool Life Management Data

Tool life management data consists of tool group numbers, tool numbers, codes specifying tool compensation values, and tool life value.

Explanations

- Tool group number

The Max. number of groups and the number of tools per group that can be registered are set by parameter (GS1,GS2 No. 7400#0, #1).

Table10.2.1 (a) The Max. number of groups and tools that can be registered

GS2 (No. 7400#1)	GS1 (No. 7400#0)	The Max. number of groups and tools without optional function of tool pairs		The Max. number of groups and tools without optional function of 512 tool pairs		The Max. number of groups and tools without optional function of 1024 tool pairs	
		Number of group	Number of tool	Number of group	Number of tool	Number of group	Number of tool
0	0	16	16	64	32	128	32
0	1	32	8	128	16	256	16
1	0	64	4	256	8	512	8
1	1	128	2	512	4	1024	4

⚠ **CAUTION**

When bits 0 or 1 of parameter GS1,GS2 No.7400 is changed, re-register tool life management data with the G10L3 command (for registering and deleting data for all groups).
Otherwise, new data pairs cannot be set.

- Tool number

A tool number is specified using a T code. A number of up to eight digits (99999999) can be specified.

- Tool offset specification code

Two types of tool offset specification codes are used: H code (for tool length compensation) and D code (for cutter compensation). A number not exceeding three digits (999) can be specified as a tool offset specification code, but a specified number must not exceed the number of tool offsets usable with the CNC.

NOTE

When codes specifying tool offset values are not used, registration can be omitted.

- Tool life value

Refer to II-10.2.2 and II-10.2.4.

10.2.2 Register, Change and Delete of Tool Life Management Data

In a program, tool life management data can be registered in the CNC unit, and registered tool life management data can be changed or deleted.

Explanations

A different program format is used for each of the four types of operations described below.

- Register with deleting all groups

After all registered tool life management data is deleted, programmed tool life management data is registered.

- Addition and change of tool life management data

Programmed tool life management data for a group can be added or changed.

- Deletion of tool life management data

Programmed tool life management data for a group can be deleted.

- Register of tool life count type

Count types (time or frequency) can be registered for individual groups.

Format

- Register with deleting all groups

Format	Meaning of command
G10L3;	G10L3:Register with deleting all groups
P-L-;	P-: Group number
T-H-D-;	L-: Life value
T-H-D-;	T-: Tool number
:	H-: Code specifying tool offset value (H code)
P-L-;	D-: Code specifying tool offset value (D code)
T-H-D-;	G11: End of registration
T-H-D-;	
:	
G11;	
M02(M30);	

- Addition and change of tool life management data

Format	Meaning of command
G10L3P1;	G10L3P1: Addition and change of group
P-L-;	P-: Group number
T-H-D-;	L-: Life value
T-H-D-;	T-: Tool number
:	H-: Code specifying tool offset value (H code)
P-L-;	D-: Code specifying tool offset value (D code)
T-H-D-;	G11: End of addition and change of group
T-H-D-;	
:	
G11;	
M02(M30);	

- Deletion of tool life management data

Format	Meaning of command
G10L3P2;	G10L3P2: Deletion of group
P-;	P-: Group number
P-;	G11: End of deletion of group
P-;	
P-;	
:	
G11;	
M02(M30);	

- Setting a tool life count type for groups

Format	Meaning of command
G10L3	Q: Life count type
(or G10L3P1);	(1:Frequency, 2:Time)
P-L-Q-;	
T-H-D-;	
T-H-D-;	
:	
P-L-Q-;	
T-H-D-;	
T-H-D-;	
:	
G11;	
M02(M30);	

⚠ **CAUTION**

When the Q command is omitted, the value set in bit 3 (LTM) of parameter No.7400 is used as the life count type.

Life values

A life value can be registered as either a time or frequency, by using bit 3 (LTM) of parameter No. 7400 or setting the corresponding count type (with the Q command). The maximum values are as follows:

Table 10.2.2 (a) Life Count Types and Maximum Life Values

LTM (No.7400#3)	Life count type	Maximum life value
0	Frequency	99999999 times
1	Time	999 time units

When the count type is time, the units for the life value specified for address L can be toggled between minutes and tenths of a second, using bit 1 (FGL) of parameter No. 7403).

Table 10.2.2 (b) Life Value Units and Maximum Allowable Values for L

FGL (No.7403#1)	Life value units	Maximum allowable value for L	Example
0	Minutes	59940	L1000: The life value is 1000 minutes.
1	Tenths of a second	35964000	L1000: The life value is 100 seconds.

10.2.3 Tool Life Management Command in a Machining Program

Explanations

- Command

The following command is used for tool life management:

Txxxxxxxx ; Specifies a tool group number.
The tool life management function selects, from a specified group, a tool whose life has not expired, and outputs its T code.
In Txxxxxxxx specify a number calculated by adding the tool life management cancel number specified in parameter7440 to a group number. For example, to set tool group 1 when the tool life management cancel number is 100, specify T101;

> **NOTE**
> When xxxxxxxx is less than a tool life management cancel number, the T code is treated as an ordinary T code.

M06 ; Terminates life management for the previously used tools, and begins counting the life of the new tools selected with the T code.

> **NOTE**
> 1 M06 is treated as an M code without buffering.
> 2 When an option for speciofying multiple M codes is selected, specify this code by itself or as the first M code. (See II-11.2)

H99 ; H code registered in the tool life management data of the tool currently being used, which enables tool length compensation. By using parameter No. 7443, tool length compensation can be enabled by an H code other than 99.

H00 ; Cancels tool length offset

D99 ; D code registered in the tool life management data of the tool currently being used, which enables cutter compensation. By using parameter No. 7444, cutter compensation can be enabled by a D code other than 99.

D00 ; Cancels cutter compensation

> **NOTE**
> H99 or D99 must be specified after the M06 command.
> If a code other than H99, D99, and the H/D codes set for parameters Nos. 7443 and 7444 is specified after the M06 command, H or D codes in the tool life management data cannot be selected.

- Types

For tool life management, the four tool change types (types A to D) indicated below are available. The type used varies from one machine to another. For details, refer to the appropriate manual of each machine tool builder.

Table 10.2.3 (a) Tool Change Type

Tool change type	A		B		C		D	
Parameter CT2,CT1 (No.7401#1, #0)	CT2	CT1	CT2	CT1	CT2	CT1	CT2	CT1
	0	0	0	1	1	0	1	1
Tool group number specified in the same block as the tool change command (M06)	Previously used tools		Tools to be used next					
Tool life count timing	Life counting is performed for a tool in the specified tool group when M06 is specified next.						Life counting is performed when a tool in the tool group specified in the same block as M06 is specified.	
Remarks	If a T command (return tool group) after M06 does not match the tool group currently being used, an alarm (PS0442) is issued (when bit 6 (ABT) of parameter No. 7400 = 0).		Normally, when a tool group number is specified by itself, type B is used. However, no alarm is raised even if the tool group number is specified by itself as type C				When only M06 is specified, P/S alarm No. 0440 is issued.	

NOTE

When a tool group number is specified and a new tool is selected, the new tool selection signal is output.

- Tool length compensation along tool axis Tool center point control

Also in these functions, the compensation amount selected by tool life management is used for compensation.

Example

- Tool change method A

A tool group command (T code) specified in a block containing the tool change command (M06) functions as a command for returning the tool to the magazine. By specifying a tool group number with a T code, the number of the tool being used is output using a T code signal. If a specified tool number differs from the tool group of the tool being used, an alarm (PS0442) is issued. By setting bit 6 (ABT) of parameter No. 7400 to 1, however, the alarm can be suppressed.

```
Example:  Assume that the tool life management ignore number is 100.
      T101 ;        Selects a group 1 tool whose service life has not expired.
        :                (Assume that tool number 010 is selected.)
      M06 ;              Counts the service life of a tool of group 1.
        :                (The service life counting for tool number 010 starts.)
       T102 ;            Selects a group 2 tool whose service life has not expired.
        :                 (Assume that tool number 100 is selected.)
       M06 T101 ;        Counts the service life of a tool of group 2.
        :                 (The service life counting for tool number 100 starts.)
        :                The number of the tool being used (tool of group 1) is output using a
        :                T code signal. (Tool number 010 is output.)
      T103 ;             Selects a group 3 tool whose life has not expired.
        :                (Assume that tool number 200 is selected.)
        M06 T102 ;       Counts the service life of a tool of group 3.
        :                (The service life counting for tool number 200 starts.)
      G43 H99 ;          Uses tool length compensation for the tool selected from group 3.
        :
      G41 D99 ;         Uses cutter compensation for the tool selected from group 3.
        :
       D00 ;              Cancels cutter compensation.
        :
      H00 ;              Cancels tool length compensation
```

- Tool change methods B and C

A tool group command (T code) specified in a block containing the tool change command (M06) functions as a tool group number command that performs life counting with the next tool change command.

Example: Assume that the tool life management ignore number is 100.

```
T101 ;         Selects a group 1 tool whose service life has not expired.
 :             (Assume that tool number 100 is selected.)
M06 T102 ;     Counts the service life of a tool of group 1.
 :              (The service life counting of tool number 010 starts.)
 :             Selects a group 2 tool whose service life has not expired.
 :             (Assume that tool number 100 is selected.)
M06 T103 ;     Counts the service life of a tool of group 2.
 :              (The service life counting of tool number 100 starts.)
 :              Selects a group 3 tool whose service life has not expired.
 :             (Assume that tool number 200 is selected.)
G43 H99 ;      Uses tool length compensation for the tool selected from group 2.
 :
 G41 D99 ;     Uses cutter compensation for the tool selected from group 2.
 :
 D00 ;         Cancels cutter compensation.
 :
 H00 ;         Cancels tool length compensation.
 :
 M06 T104 ;    Counts the service life of a tool of group 3.
 :             (The service life counting of tool number 200 starts.)
:              Selects a group 4 tool whose service life has not expired.
```

NOTE

When a tool group number command is specified singly in a block, method B must generally be used. When a tool group number command is specified singly in a block according to method C, however, no alarm is issued.

- Tool change method D

The life of the tool selected with a tool group command (T code) is counted with the tool change command (M06) specified in the same block. If the T command is not specified in the same block as M06, the T command is treated as a tool selection code, and M06 is treated as a tool change command (for starting count operation). In this case, however, an alarm (PS0440) may be issued by setting bit 7 (TAD) of parameter No. 7400.

```
Example:  Assume that the tool life management ignore number is 100.
     T101 M06 ;      Selects a group 1 tool whose service life has not expired.
       :             (Assume that tool number 010 is selected.)
       :             Counts the service life of the tool of group 1.
       :             (The service life counting of tool number 010 starts.)
      T102 M06 ;    Selects a group 2 tool whose life has not expired.
       :             (Assume that tool number 100 is selected.)
       :             Counts the service life of a tool of group 2.
       :             (The service life counting of tool number 100 starts.)
     G43 H99 ;       Uses tool length compensation for the tool selected from group 2.
       :
     G41 D99 ;       Uses cutter compensation for the tool selected from group 2.
       :
     D00 ;           Cancels cutter compensation.
       :
     H00 ;           Cancels tool length compensation.
       :
     T103 M06 ;      Selects a group 3 tool whose service life has not expired.
       :             Counts the service life of a tool of group 3.
```

10.2.4 Tool Service Life Count and Tool Selection

A count-based or time-based tool service life count system is selected using bit 3 (LTM) of parameter No. 7400. Service life counting is performed group by group. Service life count data is not lost when the power is turned off.

Table 10.2.4 Tool Life Count System and Life Count Interval

Tool life count system	Count specification	Time specification
Bit 3 (LTM) of parameter No. 7400	0	1
Life count interval	Incremented by 1 for a tool used in a program. Recounting is made possible with the M code for tool life count restart M code(parameter No. 7442).	Bit 0 (FCO) of parameter No. 7403 0:At intervals of 1 second 1:At intervals of 0.1 second These intervals can be overridden.

Explanations

- Count specification (LTM = 0)

When a tool group command (T code) is specified, a tool whose service life has not expired is selected from the group. Then, the service life count for the selected tool is incremented by 1 with the tool change command (M06). Unless the tool service life count restart M code is used, however, the selection and count-up of a new tool are allowed using only the first tool group number command and tool change command after the control unit state is changed from the reset state to the automatic operation start state.

⚠ **WARNING**

Even if the same tool group number is specified repeatedly in a program, the use count is not incremented, and a new tool is not selected.

- Time specification (LTM = 1)

Once all the registered tool life management data has been deleted, programmed tool life management data is registered.

When a tool group command (T code) is specified, a tool whose service life has not expired is selected from the group. The service life management for the selected tool is started with the tool change command (M06). Service management (counting) is performed at certain intervals (1 second or 0.1 second) while the tool is actually being used in cutting mode. This service life count interval can be specified using bit 0 (FCO) of parameter No. 7403. The time required for single block stop, feed hold, dwell, machine lock, and interlock is not counted.

By setting bit 3 (LFV) of parameter No. 7401, service life count can be overridden using the service life count override signal. An override of 0 to 99.8 times can be used. When 0 times is specified, no count is performed.

NOTE

1 When a tool whose life has not expired is selected, a search is made forward, starting with the tool that is currently being used. When the last tool is reached, search restarts from the first tool. If there are no tools whose service life has not expired, the last tool is selected. When the tool currently being used is replaced using the tool skip signal, a new tool is selected as described above.

2 When the service life of the final tool in a group expires as judged from tool life counting, the tool change signal is output. If the time-based service life count system is selected, the tool change signal is output when the service life of the final tool in a group expires. If the count-based service life count system is selected, the tool change signal is output when the tool service life count restart M code is specified, or when the CNC is reset by M02 or M03 after the service life of the final tool in a group expires.

3 The selection of a group based on a T code and tool selection within the group are performed during buffering. So if, during machining with a group selected, a buffered block contains a T code specifying the same group, the T code is buffered even if the service life expires during machining. In this case, the next tool is not selected. When the time-based service life count system is used, and a T code is specified to select the same group in succession, insert an M code for suppressing buffering immediately before the T code.

10.2.5 Tool Life Count Restart M Code

Explanations

When the life count type is frequency, a tool-change signal is output if at least one tool group has expired when the tool life count restart M code is specified. After the tool life count restart M code is specified, the tool group command (T code) selects, from a specified group, a tool whose life has not expired, and the next tool-change command (M06) increments the tool life counter by 1. This makes it possible to count the tool life using commands other than the first tool-change command (M06) since the CNC entered the automatic operation start state from the reset state. The tool life count restart M code is specified using parameter No. 7442.

Example : Suppose that M16 is the tool life count restart M code and 100 is the tool life management invalidation number. Also suppose that the life count type is frequency.

```
T101 ;      Selects, from group 1, a tool whose life has not expired.
  :
 M06 ;      Performs tool life management on group 1.
  :          (The tool life counter is incremented by 1.)
 T102 ;     Selects, from group 2, a tool whose life has not expired.
  :
 M06 ;      Performs tool life management on group 2.
  :         (The tool life counter is incremented by 1.)
M16 ;       Restarts the tool life counting.
T101 ;      Selects, from group 1, a tool whose life has not expired.
  :
 M06 ;      Performs tool life management on group 1.
  :         (The tool life counter is incremented by 1.)
```

NOTE

1 The tool life count restart M code is treated as an M code without buffering.

2 When the life count type is frequency, a tool-change signal is output if at least one tool group has expired when the tool life count restart M code is specified. When the life count type is time, no operation is performed when the tool life count restart M code is specified.

11 AUXILIARY FUNCTION

General

There are two types of auxiliary functions ; miscellaneous function (M code) for specifying spindle start, spindle stop program end, and so on, and secondary auxiliary function (B code) for specifying index table positioning.

When a move command and miscellaneous function are specified in the same block, the commands are executed in one of the following two ways:

(1) Simultaneous execution of the move command and miscellaneous function commands.

(2) Executing miscellaneous function commands upon completion of move command execution.

The selection of either sequence depends on the machine tool builder's specification.

Refer to the manual issued by the machine tool builder for details.

Example: N1G91G01X-100.0Z50.0M05; (Spindle stop)

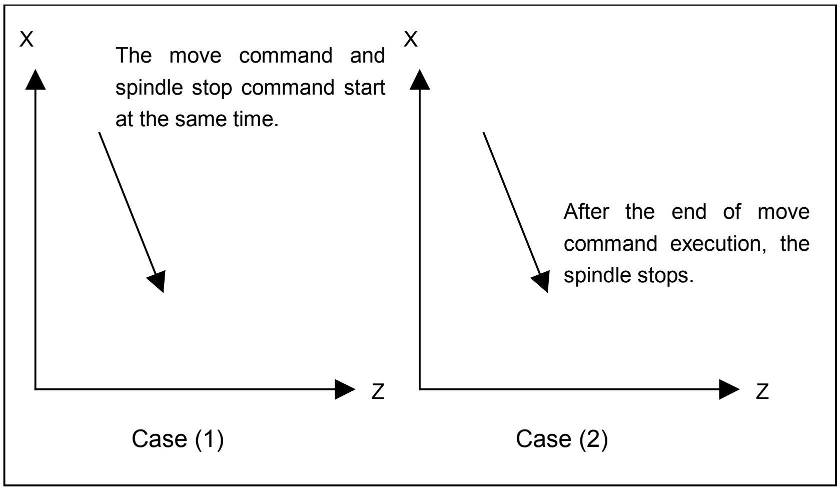

X
The move command and spindle stop command start at the same time.
Z
Case (1)
X
After the end of move command execution, the spindle stops.
Z
Case (2)

Fig.11 When a Move Command and Auxiliary Function Are Specified in the Same Block

11.1 AUXILIARY FUNCTION (M FUNCTION)

When a numeral is specified following address M, code signal and a strobe signal are sent to the machine. The machine uses these signals to turn on or off its functions.
Usually, only one M code can be specified in one block. In some cases, however, up to three M codes can be specified for some types of machine tools.
Which M code corresponds to which machine function is determined by the machine tool builder.
The machine processes all operations specified by M codes except those specified by M98, M99,M198 or called subprogram(Parameter No.7071 to 7079), called custom macro (Parameter No.7080 to 7089), or controlled interrupt custom macro on/off (M96, M97, parameter No.7033 or 7034). Refer to the machine tool builder's instruction manual for details.

Explanations

The following M codes have special meanings.

- M02,M03 (End of program)

This indicates the end of the main program
Automatic operation is stopped and the CNC unit is reset.
This differs with the machine tool builder.

- M00 (Program stop)

Automatic operation is stopped after a block containing M00 is executed.
When the program is stopped, all existing modal information remains unchanged. The automatic operation can be restarted by actuating the cycle operation. This differs with the machine tool builder.

- M01 (Optional stop)

Similarly to M00, automatic operation is stopped after a block containing M01 is executed. This code is only effective when the Optional Stop switch on the machine operator's panel has been pressed.

- M98 (Calling of sub-program)

This code is used to call a subprogram. The code and strobe signals are not sent. See the subprogram II-12.2 for details.

- M99 (End of subprogram)

This code indicates the end of a subprogram.
M99 execution returns control to the main program. The code and strobe signals are not sent. See the subprogram II-12.2 for details.

- M198 (Calling a subprogram)

This code is used to call a subprogram of a file in the external input/output function. See the description of the subprogram call function (II-12.2) for details.

NOTE

The block following M00, M01, M02, or M30 is not pre-read (buffered). Similarly, eight M codes which do not buffer can be set by parameters (Nos. 2411 to 2418). Refer to the machine tool builder's instruction manual for these M codes.

11.2 MULTIPLE M COMMANDS IN A SINGLE BLOCK

In general, only one M code can be specified in a block. However, up to five M codes can be specified at once in a block. Up to five M codes specified in a block are simultaneously output to the machine. This means that compared with the conventional method of a single M command in a single block, a shorter cycle time can be realized in machining.

Explanations

CNC allows up to three M codes to be specified in one block. However, some M codes cannot be specified at the same time due to mechanical operation restrictions. For detailed information about the mechanical operation restrictions on simultaneous specification of multiple M codes in one block, refer to the manual of each machine tool builder.
M00, M01, M02, M30, M98, M99, or M198 must not be specified together with another M code.
Some M codes other than M00, M01, M02, M30, M98, M99, and M198 cannot be specified together with other M codes; each of those M codes must be specified in a single block.
Such M codes include these which direct the CNC to perform internal operations in addition to sending the M codes themselves to the machine. To be specified, such M codes are M codes for calling program numbers 9001 to 9009 and M codes for disabling advance reading (buffering) of subsequent blocks.

Examples

One M command in a single block	Multiple M commands in a single block
M40 ; M50 ; M60 ; G28G91X0Y0Z0 ; :	M40M50M60 ; G28G91X0Y0Z0 ; : : :

11.3 SECOND AUXILIARY FUNCTIONS

When a numeric value is specified after address B, the code signal and strobe signal are output. This code is held until the next B code is output. A B code is used, for example, for rotation axis indexing on the machine. One block can contain only one B code. By setting parameter No. 1030, A, C, U, V, or W can be used instead of address B. However, duplication with a controlled axis address is not allowed. For details, refer to the manual provided by the machine tool builder.

Explanations

- Command range

The maximum number of digits that can be used for a second auxiliary function is specified using parameter No. 2033; a number from 1 to 10 can be specified. If you specify 8, for example, you can subsequently specify a value from 0 to 99999999.

- Decimal point input

The number of decimal places for a second auxiliary function is specified using parameter No. 2428. When you specify 0, you can not enter any data in the fraction part.

Limitations

An address (B or an address specified by parameter No. 1030) used for a second auxiliary function cannot be used as a controlled axis name (parameter No. 1020).

12 PROGRAM CONFIGURATION

General

- Main program and subprogram

There are two program types, main program and subprogram. Normally, the CNC operates according to the main program. However, when a command calling a subprogram is encountered in the main program, control is passed to the subprogram. When a command specifying a return to the main program is encountered in a subprogram, control is returned to the main program.

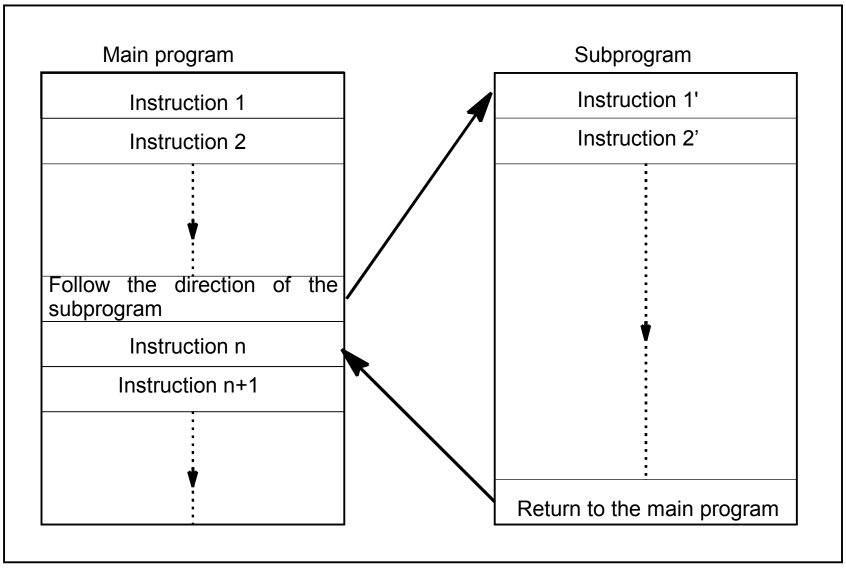


Fig.12 (a) Main program and Subprogram

The CNC memory can hold up to 1000 main programs and subprograms (100 as standard). A main program can be selected from the stored main programs to operate the machine.
See II-8 in OPERATION for the methods of registering and selecting programs.

- Program components

A program consists of the following components:

Table12 Program components

Components	Descriptions
File start	Symbol indicating the start of a program file
Leader section	Used for the title of a program file, etc.
Program start	Symbol indicating the start of a program
Program section	Commands for machining
Comment section	Comments or directions for the operator
Tape end	Symbol indicating the end of a program file

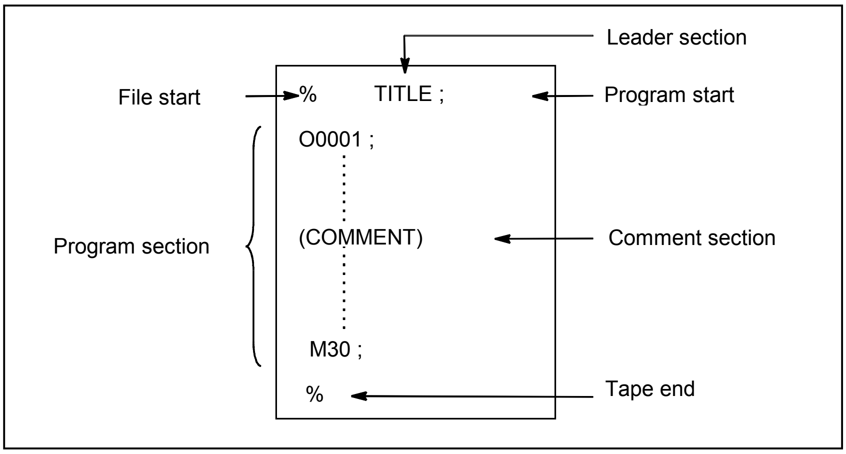


Fig.12 (b) Program configuration

- Program section configuration

A program section consists of several blocks. A program section starts with a program number and ends with a program end code.

Program section configuration	Program section
Program number	O0001 ;
Block 1	N1 G91 G00 X120.0 Y80.0 ;
Block 2	N2 G43 Z-32.0 H01 ;
:	:
Block n	Nn Z0 ;
Program end	M30 ;

A block contains information necessary for machining, such as a move command or coolant on/off command.
Specifying a slash (/) at the start of a block disables the execution of some blocks (see "optional block skip" in II-12.1).

12.1 PROGRAM SECTION CONFIGURATION

This section describes elements of a program section.
See II-12.4 for program components other than program sections.

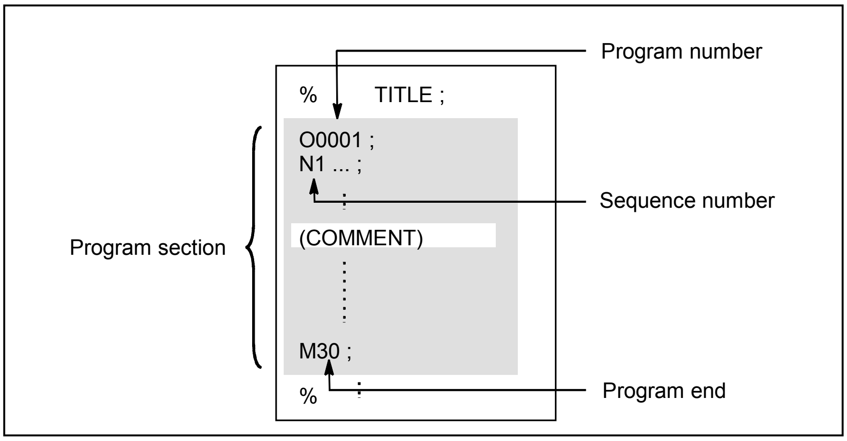


Fig.12.1 (a) Program configuration

- Program number

To distinguish between multiple programs stored in memory, address O followed by an 8-digit number (1 to 99999999) is prefixed to each program.
In ISO code, the colon (:) can be used instead of O.
When no program number is specified at the start of a program, the sequence number (N....) at the start of the program is regarded as its program number. If a five-digit sequence number is used, the lower four digits are registered as a program number. Note, however, that N0 cannot be used for a program number.
If there is no program number or sequence number at the start of a program, a program number must be specified using the MDI panel when the program is stored in memory (See Operation II-8.1)

> **NOTE**
> Program numbers 9000 to 9899 may be used by machine tool builders, and the user may not be able to use these numbers.

- Sequence number and block

A program consists of several commands. One command unit is called a block. One block is separated from another with an EOB of end of block code.

Table12.1 (a) EOB code

Name	ISO code	EIA code	Notation in this manual
End of block (EOB)	LF	CR	;

At the head of a block, a sequence number consisting of address N followed by a number not longer than eight digits (1 to 99999999) can be placed. Sequence numbers can be specified in a random order, and any numbers can be skipped. Sequence numbers may be specified for all blocks or only for desired blocks of the program. In general, however, it is convenient to assign sequence numbers in ascending order in phase with the machining steps (for example, when a new tool is used by tool replacement, and machining proceeds to a new surface with table indexing.)

N300×200.0 Z300.0 ; A sequence number is underlined.

Fig.12.1 (b) Sequence number and block (example)

NOTE

N0 must not be used for the reason of file compatibility with other CNC systems. Program number 0 cannot be used. So 0 must not be used for a sequence number regarded as a program number.

- TV check (Vertical parity check along tape)

A parity check is made for a block on input tape vertically. If the number of characters in one block (starting with the code immediately after an EOB and ending with the next EOB) is odd, an P/S alarm (No.0591) is output. No TV check is made only for those parts that are skipped by the label skip function. Bit 1 (CTV) of parameter No. 0000 is used to specify whether comments enclosed in parentheses are counted as characters during TV check. The TV check function can be enabled or disabled by setting on the MDI unit (See Operation II-12.2.1.).

- Block configuration(word and address)

A block consists of one or more words.
A word consists of an address followed by a number some digits long. (The plus sign (+) or minus sign (-) may be prefixed to a number.)
For an address, one of the letters (A to Z) is used ; an address defines the meaning of a number that follows the address.

Word = Address + number (Example : X-1000)

Table 12.2 (b) indicates the usable addresses and their meanings.
The same address may have different meanings, depending on the preparatory function specification.

Table12.1 (b) Major functions and addresses

Function	Address	Meaning
Program number	O*	Program number
Sequence number	N	Sequence number
Preparatory function	G	Specifies a motion mode (linear, arc, etc.)

Function	Address	Meaning
Dimension word	X,Y,Z,U,V,W,A,B,C	Coordinate axis move command
	I,J,K	Coordinate of the arc center
	R	Arc radius
Feed function	F	Rate of feed per minute, Rate of feed per revolution
Spindle speed function	S	Spindle speed
Tool function	T	Tool number
Auxiliary function	M	On/off control on the machine tool
	B	Table indexing, etc.
Offset number	D,H	Offset number
Dwell	P,X	Dwell time
Program number designation	P	Subprogram number
Number of repetitions	L	Number of subprogram repetitions
Parameter	P,Q	Canned cycle parameter

* In ISO code, the colon (:) can also be used as the address of a program number.

N_	G_	X_ Y_	F_	S_	T_	M_	;
Sequence number	Preparatory function	Dimensio n word	Feed- function	Spindle speed function	Tool function	Miscellaneous function	

Fig.12.1 (c) 1 block (example)

- Major addresses and ranges of command values

Major addresses and the ranges of values specified for the addresses are shown below.

Note that these figures represent limits on the CNC side, which are totally different from limits on the machine tool side.

For example, the CNC allows a tool to traverse up to about 100 m (in millimeter input) along the X axis. However, an actual stroke along the X axis may be limited to 2 m for a specific machine tool. Similarly, the CNC may be able to control a cutting federate of up to 240 m/min, but the machine tool may not allow more than 3 m/min.

When developing a program, the user should carefully read the manuals of the machine tool as well as this manual to be familiar with the restrictions on programming.

Table12.1 (c) Major addresses and ranges of command values

Function		Address	Input in mm	Input in inch
Program number		O[*1]	1 to 99999999	1 to 99999999
Sequence number		N	1 to 99999999	1 to 99999999
Preparatory function		G	0 to 99.9	0 to 99.9
Dimension word	Increment system IS-A	X,Y,Z,U,V,W,A,B, C,I,J,K,R[*2]	±999999.99mm ±999999.99deg.	±99999.999inch[*3] ±999999.99deg.
	Increment system IS-B		±999999.999mm ±999999.999deg.	±99999.9999inch[*3] ±999999.999deg.
	Increment system IS-C		±99999.9999mm ±99999.9999deg.	±9999.99999inch[*3] ±99999.9999deg.
	Increment system IS-D		±9999.99999mm ±9999.99999deg.	±999.999999inch[*3] ±9999.99999deg.
	Increment system IS-E		±999.999999mm ±999.999999deg.	±99.9999999inch[*3] ±999.999999deg.
Feed per minute	Increment system IS-A	F	1 to 2400000mm/min	0.01 to 96000.00inch/min
	Increment system IS-B		1 to 240000mm/min	0.01 to 9600.00inch/min
	Increment system IS-C		1 to 99999mm/min	0.01 to 4000.00inch/min
	Increment system IS-D		1 to 9999mm/min	0.01 to 400.00inch/min
	Increment system IS-E		1 to 999mm/min	0.01 to 40.00inch/min
Feed per revolution		F	0.001 to 50000mm/rev	0.0001 to 50.0000inch/rev
Spindle speed function		S[*4]	0 to 4294967295	0 to 4294967295
Tool function		T[*4]	0 to 4294967295	0 to 4294967295
Auxiliary function		M[*4]	0 to 4294967295	0 to 4294967295
		B[*4]	0 to 4294967295	0 to 4294967295
Offset number		H, D	0 to 999	0 to 999
Dwell	Increment system IS-A	X, P	0 to 999999.99 S	0 to 999999.99 S
	Increment system IS-B		0 to 999999.999 S	0 to 999999.999 S
	Increment system IS-C		0 to 99999.9999 S	0 to 99999.9999 S
	Increment system IS-D		0 to 9999.99999 S	0 to 9999.99999 S
	Increment system IS-E		0 to 999.999999 S	0 to 999.999999 S
Designation of a program number		P	1 to 99999999	1 to 99999999
Number of subprogram repetitions		L	1 to 99999999	1 to 99999999

*1 In ISO code, the colon (:) can also be used as the address of a program number.

*2 When address I, J, K, or R is used to specify a radius for circular interpolation, the following limitations are imposed:

Increment system	Input in mm	Input in inch
IS-A	±999999999.99mm	±99999999.999inch
IS-B	±999999999.999mm	±99999999.9999inch
IS-C	±99999999.9999mm	±9999999.99999inch
IS-D	±9999999.99999mm	±999999.999999inch
IS-E	±999999.999999mm	±99999.9999999inch

*3 When a millimeter machine is used with inch input, the maximum specifiable range of a dimension word is as follows:

Increment system	The maximum specifiable range
IS-A	±39370.078inch
IS-B	±39370.0787inch
IS-C	±3937.00787inch
IS-D	±393.700787inch
IS-E	±39.3700787inch

*4 With address M, S, T, or B, an unsigned integer of no more than 32 bits can be specified. However, the value specifiable in a parameter-set M, S, T, or B code ranges from 0 to 99999999.
However, the number of digits greater than the value of parameters (Nos. 2030 to 2033) that specify an allowable number of digits cannot be specified. Parameter No. 2003 can be used to enter a negative value.
Numbers to be set and usage purposes are restricted on some codes, such as M codes that are not buffered, depending on parameter settings. For details, see the parameter manual.

- Optional block skip

When a slash followed by a number (/n (n=1 to 9)) is specified at the head of a block, and optional block skip switch n on the machine operator panel is set to on, the information contained in the block for which /n corresponding to switch number n is specified is ignored in DNC operation or memory operation.
When optional block skip switch n is set to off, the information contained in the block for which /n is specified is valid. This means that the operator can determine whether to skip the block containing /n. Number 1 for /1 can be omitted. However, when two or more optional block skip switches are used for one block, number 1 for /1 cannot be omitted.

Example)

```
    (Incorrect)          (Correct)
    //3  G00X10.0;      /1 /3 G00X10.0;
```

This function is ignored when programs are loaded into memory.
Blocks containing /n are also stored in memory, regardless of how the optional block skip switch is set. Programs held in memory can be output, regardless of how the optional block skip switches are set.
Optional block skip is effective even during sequence number search operation.
Depending on the machine tool, all optional block skip switches (1 to 9) may not be usable. Refer to manuals of the machine tool builder to find which switches are usable.

⚠ **CAUTION**

1 Position of a slash
A slash (/) must be specified at the head of a block. If a slash is placed elsewhere, the information from the slash to immediately before the EOB code is ignored.

2 Disabling an optional block skip switch
Optional block skip operation is processed when blocks are read from memory or tape into a buffer. Even if a switch is set to on after blocks are read into a buffer, the blocks already read are not ignored.

NOTE

TV and TH check
When an optional block skip switch is on. TH and TV checks are made for the skipped portions in the same way as when the optional block skip switch is off.

- Program end

The end of a program is indicated by programming one of the following codes at the end of the program:

Table12.1 (a) Code of a program end

Code	Meaning usage
M02	For main program
M30	
M99	For subprogram

If one of the program end codes is executed in program execution, the CNC terminates the execution of the program, and the reset state is set. When the subprogram end code is executed, control returns to the program that called the subprogram.

⚠ **CAUTION**

A block containing an optional block skip code such as /M02 ; , /M30 ; , or /M99 ; is not regarded as the end of a program, if the optional block skip switch on the machine operator's panel is set to on.(See "Optional block skip".)

12.2 SUBPROGRAM (M98, M99)

If a program contains a fixed sequence or frequently repeated pattern, such a sequence or pattern can be stored as a subprogram in memory to simplify the program. A subprogram can be called from the main program. A called subprogram can also call another subprogram.

Format

- Subprogram configuration

One subprogram

```
O1234;      Subprogram number (or the colon (:)
 :          optionally in the case of ISO)
 :
M99;        Program end
```

M99 need not constitute a separate block as indicated below.

Example)X100.0Y100.0M99;

- Subprogram call

M98 P<u>xxxxxxxx</u> L<u>xxxxxxxx</u> ;

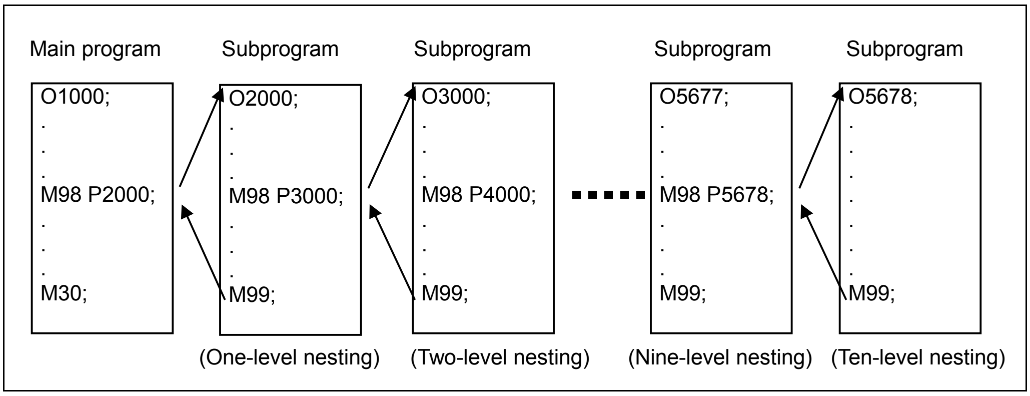
↑ Subprogram number

↑ Number of times the subprogram is called repeatedly

When no repetition data is specified, the subprogram is called just once.

Explanation

When the main program calls a subprogram, it is regarded as a one-level subprogram call. Thus, subprogram calls can be nested up to ten levels as shown below.



A single call command can repeatedly call a subprogram up to 9999 times.
For compatibility with automatic programming systems, in the first block, Nxxxx can be used instead of a subprogram number that follows O (or :). A sequence number after N is registered as a subprogram number.

Reference

See Operation II-8.1.1for the method of registering a subprogram.

NOTE
1 The M98 and M99 code signal and strobe signal are not output to the machine tool.
2 If the subprogram number specified by address P cannot be found, an alarm (No. 0078) is output.

Example

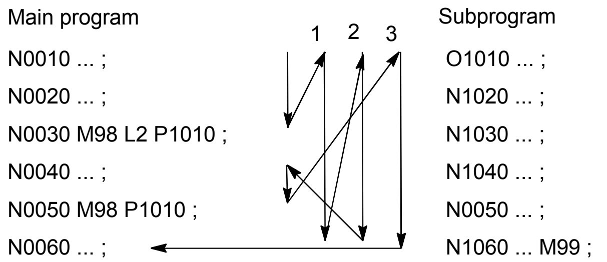
M98 P1002 L5;

This command specifies "Call the subprogram (number 1002) five times in succession."
A subprogram call command (M98P_) can be specified in the same block as a move command.

X1000.0 M98 P1200;

This example calls the subprogram (number 1200) after an X movement.

Execution sequence of subprograms called from a main program



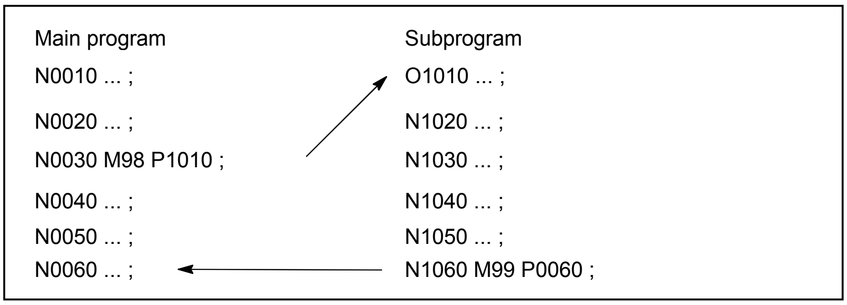
```
Main program                 Subprogram
N0010 ... ;                  O1010 ... ;
N0020 ... ;                  N1020 ... ;
N0030 M98 L2 P1010 ;         N1030 ... ;
N0040 ... ;                  N1040 ... ;
N0050 M98 P1010 ;            N0050 ... ;
N0060 ... ;                  N1060 ... M99 ;
```

A subprogram can call another subprogram in the same way as a main program calls a subprogram.

Special Usage

- Specifying the sequence number for the return destination in the main program

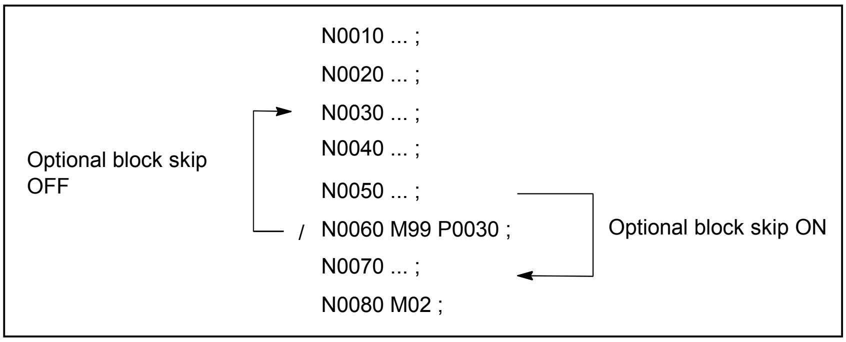
If P is used to specify a sequence number when a subprogram is terminated, control does not return to the block after the calling block, but returns to the block with the sequence number specified by P.
Note, however, that P is ignored if the main program is operating in a mode other than memory operation mode.
This method consumes a much longer time than the normal return method to return to the main program.



```
Main program                 Subprogram
N0010 ... ;                  O1010 ... ;
N0020 ... ;                  N1020 ... ;
N0030 M98 P1010 ;            N1030 ... ;
N0040 ... ;                  N1040 ... ;
N0050 ... ;                  N1050 ... ;
N0060 ... ;                  N1060 M99 P0060 ;
```

- Using M99 in the main program

If M99 is executed in a main program, control returns to the start of the main program.
For example, M99 can be executed by placing /M99 ; at an appropriate location of the main program and setting the optional block skip function to off when executing the main program. When M99 is executed, control returns to the start of the main program, then execution is repeated starting at the head of the main program.
Execution is repeated while the optional block skip function is set to off.
If the optional block skip function is set to on, the /M99 ; block is skipped ; control is passed to the next block for continued execution.
If/M99Pn ; is specified, control returns not to the start of the main program, but to sequence number n. In this case, a longer time is required to return to sequence number n.

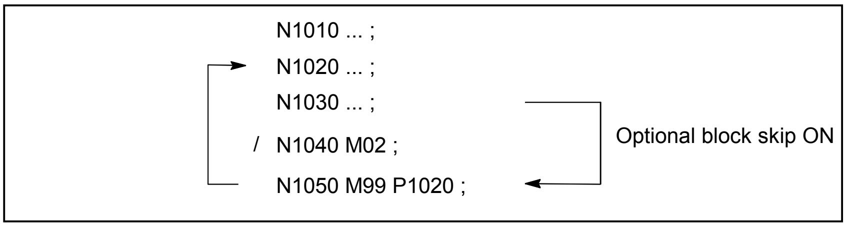


```
  N0010 ... ;
  N0020 ... ;
  N0030 ... ;
  N0040 ... ;
  N0050 ... ;
/ N0060 M99 P0030 ;
  N0070 ... ;
  N0080 M02 ;
```

- Using a subprogram only

A subprogram can be executed just like a main program by searching for the start of the subprogram with the MDI. (See Operation II-10.3- for information about search operation.)

In this case, if a block containing M99 is executed, control returns to the start of the subprogram for repeated execution. If a block containing M99Pn is executed, control returns to the block with sequence number n in the subprogram for repeated execution. To terminate this program, a block containing /M02 ; or /M30 ; must be placed at an appropriate location, and the optional block switch must be set to off ; this switch is to be set to on first.



```
  N1010 ... ;
  N1020 ... ;
  N1030 ... ;
/ N1040 M02 ;
  N1050 M99 P1020 ;
```

12.3 PROGRAM NUMBER

The 8-digit program number function enables specification of program numbers with eight digits following address O (1 to 99999999).

Explanation

- Disabling editing of programs

Editing of subprograms O00008000 to O00008999 and O00009000 to O00009999 can be disabled.

Parameter	Program numbers for which editing is disabled
NE8(No.0011#0)	O00008000 to O00008999
NE9(No.2201#0)	O00009000 to O00009999

- Indication-prohibited program

The indication of subprograms of the program numbers O00008000 to O00008999 and O00009000 to O00009999 during execution can be disabled.

Parameter	Program numbers for which editing is disabled
ND8(No.0011#1)	O00008000 to O00008999
ND9(No.2201#1)	O00009000 to O00009999

- Single block stop program

For program punch by specifying a range, files are named as follows: When punching by specifying O00000001 and O000123456: "O00000001-G"

Parameter	Program numbers for which editing is disabled
SB7(No.0010#3)	O00007000 to O00007999
SB8(No.0010#4)	O00008000 to O00008999
SB9(No.2201#2)	O00009000 to O00009999
SBM(No.0010#5)	All program

- External program number search

External input signals can be used to search for a program number.

A program stored in CNC memory can be selected by externally inputting a program number, between 1 and 99999999, to the CNC.

For details, refer to the appropriate manual supplied from the machine tool builder.

Limitation

- Limit on the number of digits

With the FANUC Floppy Cassette, no program number may be longer than four digits.

12.4 PROGRAM COMPONENTS OTHER THAN PROGRAM SECTIONS

This section describes program components other than program sections. See Operation II-12.1 for a program section.

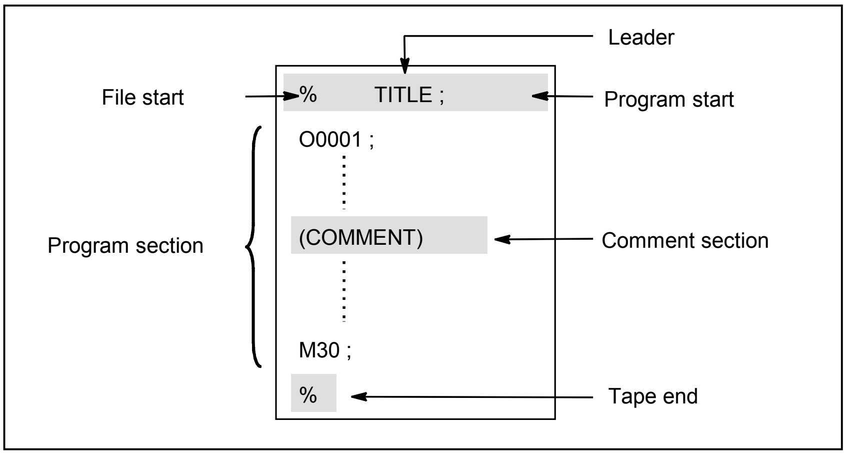


Fig.12.4 Program configuration

Explanation

- Tape start

The file start indicates the start of a file that contains NC programs. The mark is not required when programs are entered using SYSTEM P or ordinary personal computers.
The mark is not displayed on the screen. However, if the file is output, the mark is automatically output at the start of the file.

Table12.4(a) Code of file start

Name	ISO code	EIA code	Notation in this manual
File start	%	ER	%

- Leader section

Data entered before the programs in a file constitutes a leader section. When machining is started, the label skip state is usually set by turning on the power or resetting the system. In the label skip state, all information is ignored until the first end-of-block code is read. When a file is read into the CNC unit from an I/O device, leader sections are skipped by the label skip function.
A leader section generally contains information such as a file header. When a leader section is skipped, even a TV parity check is not made. So a leader section can contain any codes except the EOB code.

- Program start

The program start code is to be entered immediately after a leader section, that is, immediately before a program section.
This code indicates the start of a program, and is always required to disable the label skip function.
With SYSTEM P or ordinary personal computers, this code can be entered by pressing the return key.

Table12.4 (b) Code of a program start

Name	ISO code	EIA code	Notation in this manual
Program start	LF	CR	;

NOTE

If one file contains multiple programs, the EOB code for label skip operation must not appear before a second or subsequent program number.

- Comment section

Any information enclosed by the control-out and control-in codes is regarded as a comment.
The user can enter a header, comments, directions to the operator, etc.

Table12.4 (c) Codes of a control-in and a control-out

Name	ISO code	EIA code	Notation in this manual	Meaning
Control-out	(	2-4-5	(	Start of comment section
Control-in	)	2-4-7	)	End of comment section

When a program is read into memory for memory operation, comment sections, if any, are not ignored but are also read into memory. Note, however, that codes other than those listed in the code table in Appendix A are ignored, and thus are not read into memory.
When data in memory is output on external I/O device (See Operation II-12.3), the comment sections are also output. When a program is displayed on the screen, its comment sections are also displayed.
However, those codes that were ignored when read into memory are not output or displayed.
During memory operation or DNC operation, all comment sections are ignored.
The TV check function can be used for a comment section by setting parameter CTV (bit 1 of No. 0000).

⚠ **CAUTION**

If a long comment section appears in the middle of a program section, a move along an axis may be suspended for a long time because of such a comment section. So a comment section should be placed where movement suspension may occur or no movement is involved.

NOTE

1 If only a control-in code is read with no matching control-out code, the read control-in code is ignored.
2 The EOB code cannot be used in a comment.

- File end

A tape end is to be placed at the end of a file containing NC programs. If programs are entered using the automatic programming system, the mark need not be entered.

The mark is not displayed on the screen. However, when a file is output, the mark is automatically output at the end of the file. If an attempt is made to execute % when M02 or M03 is not placed at the end of the program, the alarm (SR 0592) is occurred.

Table12.4 (d) Code of a file end

Name	ISO code	EIA code	Notation in this manual
File end	%	ER	%

12.5 EXTERNAL DEVICE SUBPROGRAM CALL (M198)

During memory operation, subprograms registered in an external device (such as Handy File, data server, and so forth) connected to the CNC can be called and executed.

Format

M198 P[program-number (or file-number)] L[number of repeats]

Explanations

M198 is an M code used to call an external device subprogram. An external device subprogram can also be called using an M code set in parameter No. 2431. (When an M code other than M198 is used to call an external device subprogram, M198 is handled as an ordinary M code.) Address P is used to specify the program number of a subprogram registered in an external device. If a program having a specified program number (or file number) does not exist on the connected external device, an alarm (PS00076 or PS0079) is output.
Address L is used to specify the number of times the subprogram is repeated. When address L is omitted, the subprogram is executed once only.

- Call by file number

With some external devices, file numbers such as file 1 and file 2 may be assigned to the registered programs in the order of program registration. Bit 5 (SFL) of parameter No. 2404 allows a subprogram to be called by specifying its file number instead of its program number.

NOTE

1 This function is a basic function. To enable this function, however, bit 3 (EXT) of parameter No. 7616 must be set.

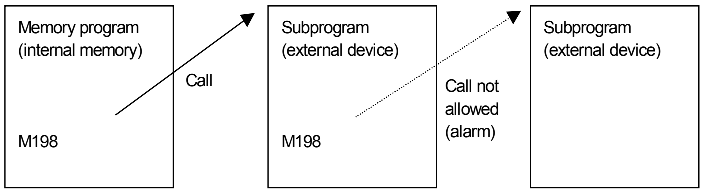
2 Subprograms can be called from the external devices listed below.

Device name	Call by program number	Call by file number
Handy File	Possible	Possible
Floppy Cassette	Possible	Possible
FA Card	Possible	Possible
Program File Mate	Possible	Possible
Data server	Possible	Impossible
Intelligent terminal	Possible	Impossible
Personal computer connected via HSSB	Possible	Impossible

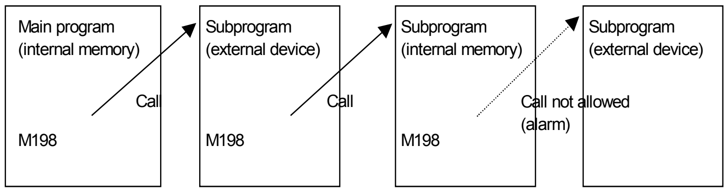
NOTE

3 External device subprograms can be called only during memory operation. If an attempt is made to call an external device subprogram in other than memory mode, an alarm (PS0081) is output.

4 An additional external device cannot be called from a called external device subprogram. (If such an attempt is made, an alarm (PS0080) is output.)



5 A subprogram stored in internal memory can be called from a called external device subprogram. However, an additional external device subprogram cannot be called from such a called subprogram. (If such an attempt is made, an alarm (PS0080) is output.)



6 An external device subprogram call is counted as one subprogram call.

13 FUNCTIONS TO SIMPLIFY PROGRAMMING

This chapter explains the following items:

13.1 CANNED CYCLE

Canned cycles make it easier for the programmer to create programs. With a canned cycle, a frequently-used machining operation can be specified in a single block with a G function; without canned cycles, normally more than one block is required. In addition, the use of canned cycles can shorten the program to save memory.
Table 13.1 (a) lists canned cycles.

Table 13.1 (a) Canned cycles

G code	Drilling(-Z direction)	Operation at the bottom of a hole	Retraction(+ Z direction)	Application
G73	Intermittent feed	-	Rapid traverse	High-speed peck drilling cycle
G74	Feed	Spindle CW	Feed	Left-hand tapping cycle (Canned cycle II only)
G76	Feed	Spindle orientation	Rapid traverse	Fine boring cycle
G80	-	-	-	Cancel
G81	Feed	-	Rapid traverse	Drilling cycle, counter drilling cycle
G82	Feed	Dwell	Rapid traverse	Drilling cycle, counter boring cycle
G83	Intermittent feed	-	Rapid traverse	Peck drilling cycle
G84	Feed	Spindle CCW	Feed	Tapping cycle
G85	Feed	-	Feed	Boring cycle
G86	Feed	Spindle stop	Rapid traverse	Boring cycle
G87	Feed	Spindle stop/ Spindle CW	Manual/Rapid traverse	Boring cycle/ Back boring cycle
G88	Feed	Dwell→spindle stop	Manual/Rapid traverse	Boring cycle
G89	Feed	Dwell	Feed	Boring cycle

NOTE

The operation of G87 differs between canned cycle I and canned cycle II.

Explanation

A canned cycle consists of a sequence of six operations (Fig. 13.1 (a))
Operation 1 ..... Positioning of axes X and Y
(including also another axis)
Operation 2 ..... Rapid traverse up to point R level
Operation 3 ..... Hole machining
Operation 4 ..... Operation at the bottom of a hole
Operation 5...... Retraction to point R level
Operation 6...... Rapid traverse up to the initial point

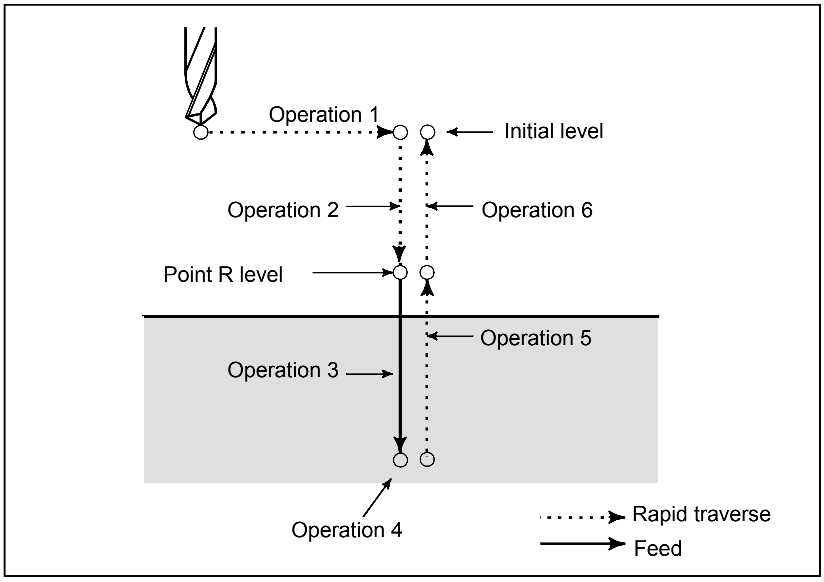


Fig.13.1 (a) Canned cycle operation sequence

- Canned cycle I

An independent output signal is used for each of reverse spindle rotation and spindle stop.
When bit 0 (FXB) of parameter No. 6201 is set to 0, canned cycle I is set.

- Canned cycle II

M codes are used as output signals for reverse spindle rotation, spindle stop, and spindle orientation.
When bit 0 (FXB) of parameter No. 6201 is set to 1, canned cycle II is set.

- Positioning plane

The positioning plane is determined by plane selection code G17, G18, or G19.
The positioning axis is an axis other than the drilling axis.

- Drilling axis

Although canned cycles include tapping and boring cycles as well as drilling cycles, in this chapter, only the term drilling will be used to refer to operations implemented with canned cycles.
The drilling axis is a basic axis (X, Y, or Z) not used to define the positioning plane, or any axis parallel to that basic axis.
The axis (basic axis or parallel axis) used as the drilling axis is determined according to the axis address for the drilling axis specified in the same block as G codes G73 to G89.
If no axis address is specified for the drilling axis, the basic axis is assumed to be the drilling axis.

Table 13.1 (b) Positioning plane and drilling axis

G code	Positioning plane	Drilling axis
G17	Xp-Yp plane	Zp
G18	Zp-Xp plane	Yp
G19	Yp-Zp plane	Xp

Xp : X axis or an axis parallel to the X axis
Yp : Y axis or an axis parallel to the Y axis
Zp : Z axis or an axis parallel to the Z axis

Assume that the U, V and W axes be parallel to the X, Y, and Z axes respectively. This condition is specified by parameter No. 1022.

G17 G81- Z _ : The Z axis is used for drilling.
G17 G81- W_ : The W axis is used for drilling.
G18 G81- Y_ : The Y axis is used for drilling.
G18 G81- V_ : The V axis is used for drilling.
G19 G81- X_ : The X axis is used for drilling.
G19 G81- U_ : The U axis is used for drilling.

G17 to G19 may be specified in a block in which any of G73 to G89 is not specified.

⚠ **CAUTION**

Switch the drilling axis after canceling a canned cycle.

NOTE

A parameter FXY (No. 6200 #0) can be set to the Z axis always used as the drilling axis. When FXY=0, the Z axis is always the drilling axis.

- Travel distance along the drilling axis G90/G91

The travel distance along the drilling axis varies for G90 and G91 as follows:

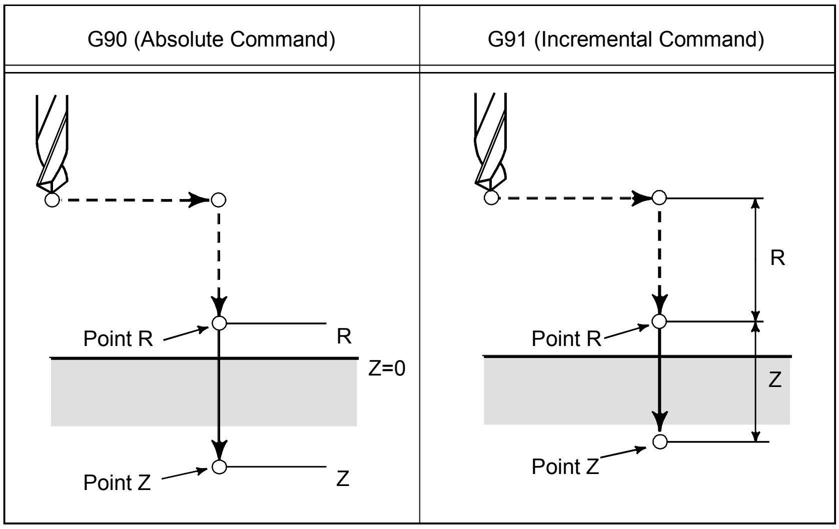


Fig. 13.1 (b) Absolute command and incremental command

- Drilling mode

G73, G74, G76, and G81 to G89 are modal G codes and remain in effect until canceled. When in effect, the current state is the drilling mode.

Once drilling data is specified in the drilling mode, the data is retained until modified or canceled.

Specify all necessary drilling data at the beginning of canned cycles; when canned cycles are being performed, specify data modifications only.

- Return point level G98/G99

When the tool reaches the bottom of a hole, the tool may be returned to point R or to the initial level. These operations are specified with G98 and G99.

The following illustrates how the tool moves when G98 or G99 is specified.

Generally, G99 is used for the first drilling operation and G98 is used for the last drilling operation.

The initial level does not change even when drilling is performed in the G99 mode.

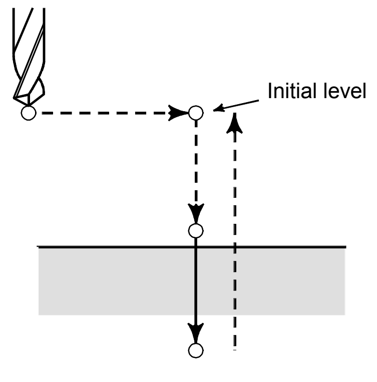
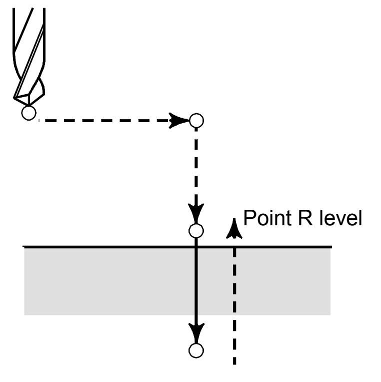
G98(Return to initial level)	G99(Return to point R level)
	

Fig.13.1 (c) Initial level and point R level

- Repeat

To repeat drilling for equally-spaced holes, specify the number of repeats in L_.

K is effective only within the block where it is specified.

Specify the first hole position in incremental mode (G91).

If it is specified in absolute mode (G90), drilling is repeated at the same position.

Number of repeats L
The maximum command value = 9999

If L0 is specified, drilling data is stored, but drilling is not performed.

- Cancel

To cancel a canned cycle, use G80 or a group 01 G code.

Group 01 G codes

G00 :	Positioning (rapid traverse)
G01 :	Linear interpolation
G02 :	Circular interpolation or helical interpolation (CW)
G03 :	Circular interpolation or helical interpolation (CCW)

- Symbols in figures

Subsequent sections explain the individual canned cycles. Figures in these explanations use the following symbols:

- - -➤	Positioning (rapid traverse G00)
⟶	Cutting feed (linear interpolation G01)
∿➤	Manual feed
OSS	Oriented spindle stop (The spindle stops at a fixed rotation position)
⇨	Shift (rapid traverse G00)
P	Dwell

13.1.1 High-speed Peck Drilling Cycle (G73)

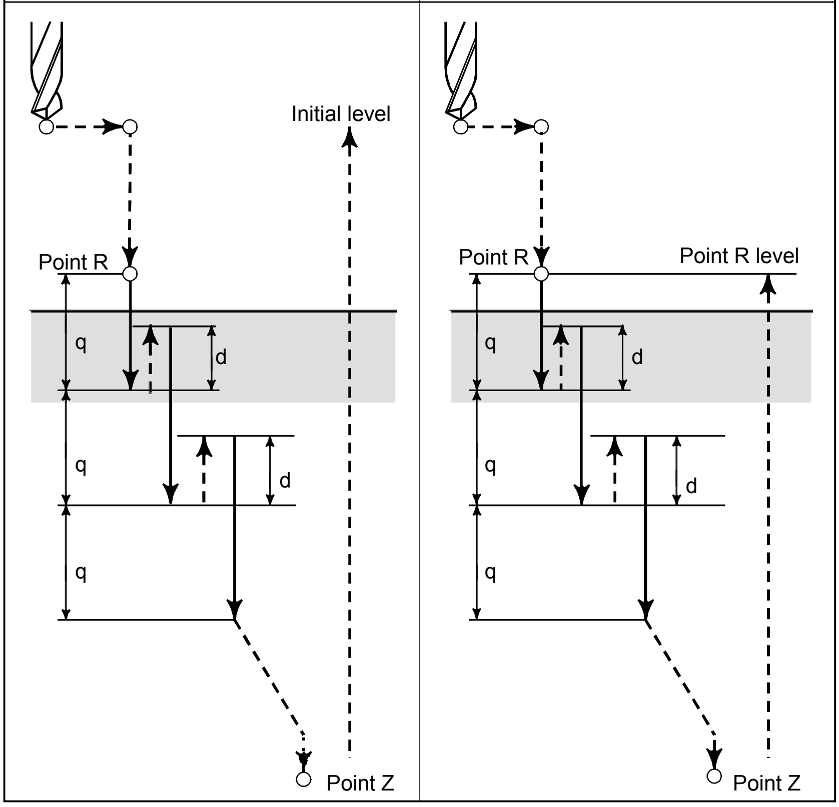
This cycle performs high-speed peck drilling.
It performs intermittent cutting feed to the bottom of a hole while removing chips from the hole.

Format

```
G73 X_ Y_ Z_ R_ Q_ F_ L_ ;
```

X_ Y_	: Hole position data
Z_	: The distance from point R to the bottom of the hole
R_	: The distance from the initial level to point R level
Q_	: Depth of cut for each cutting feed
F_	: Cutting feedrate
L_	: Number of repeats (if required)

G73(G98) G73(G99)



Explanation

- Operation

The high-speed peck drilling cycle performs intermittent feeding along the Z-axis. When this cycle is used, chips can be removed from the hole easily, and a smaller value can be set for retraction. This allows, drilling to be performed efficiently.
Set the clearance, d, in parameter 6210.
The tool is retracted in rapid traverse.

- Spindle rotation

Before specifying G73, rotate the spindle using a miscellaneous function (M code).

- Miscellaneous function

When the G73 code and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- Q

Specify Q and R in blocks that perform drilling. If they are specified in a block that does not perform drilling, they cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G73 in a single block. Otherwise, G73 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

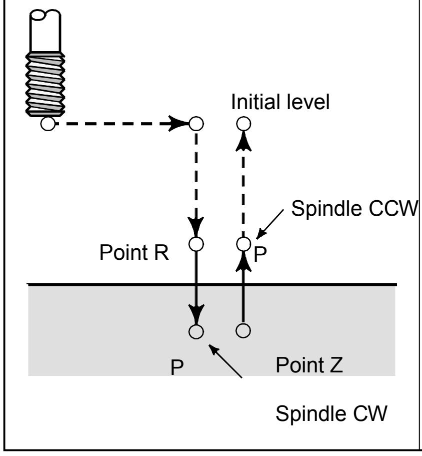
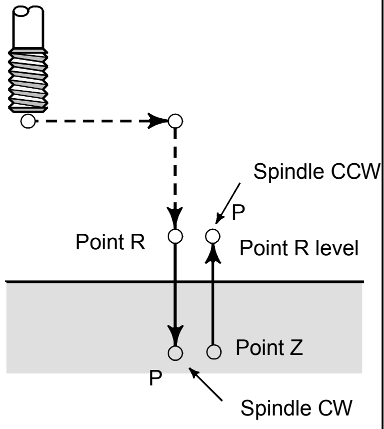
```
M3 S2000 ;      Cause the spindle to start rotating.
G90 G99 G73 X300. Y-250. Z- 150. R-100. Q15. F120. ;
                Position, drill hole 1, then return to point R.
Y-550. ;        Position, drill hole 2, then return to point R.
Y-750. ;        Position, drill hole 3, then return to point R.
X1000. ;        Position, drill hole 4, then return to point R.
Y-550. ;        Position, drill hole 5, then return to point R.
G98 Y-750. ;    Position, drill hole 6, then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return
M5 ;            Cause the spindle to stop rotating.
```

13.1.2 Left-handed Tapping Cycle (G74)

This cycle performs left-handed tapping.
In the left-handed tapping cycle, when the bottom of the hole has been reached, the spindle rotates clockwise.

Format

```
G74 X_ Y_ Z_ R_ P_ F_ L_ ;

  X_Y_        : Hole position data
  Z_          : The distance from point R to the bottom of the hole
  R_          : The distance from the initial level to point R level
  P_          : Dwell time
  F_          : Cutting feedrate
  L_          : Number of repeats (if required)
```

G74(G98)	G74(G99)
 Initial level Spindle CCW Point R P P Point Z Spindle CW	 Spindle CCW P Point R Point R level Point Z P Spindle CW

Explanation

- Operation

Tapping is performed by turning the spindle counterclockwise. When the bottom of the hole has been reached, the spindle is rotated clockwise for retraction. This creates a reverse thread.

> ⚠ **CAUTION**
> Feedrate overrides are ignored during left-handed tapping. A feed hold does not stop the machine until the return operation is completed.

- Dwell

By setting bit 1 (DWL) of parameter No. 6200 to 1, dwell is performed for the duration specified with P before forward or reverse spindle rotation. This function is useful when a special tapper is used. (While dwell operation is being performed with no movement made along the Z-axis, a tapper performs forward/backward movement for threading only with a rotational force.)

- Spindle rotation

Before G74 is specified, turn the spindle in the reverse direction with a miscellaneous function (M code).
When successive hole machining operations which involve a short distance from a hole position and the initial level to the R point level, the spindle may not reach the normal speed before starting a hole cutting operation. In such a case, L for the number of repeats must not be specified. Instead, a dwell time based on G04 must be inserted before each hole machining operation. However, some machines do not require this specification. For details, refer to the manual provided by the machine tool builder.

- Miscellaneous function

When the G74 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation.
When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- P

Specify P in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G74 in a single block. Otherwise, G74 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

```
M4 S100 ;        Cause the spindle to start rotating.
G90 G99 G74 X300. Y-250. Z-150. R -120. F120. ;
                 Position, tapping hole 1, then return to point R.
Y-550. ;         Position, tapping hole 2, then return to point R.
Y-750. ;         Position, tapping hole 3, then return to point R.
X1000. ;         Position, tapping hole 4, then return to point R.
Y-550. ;         Position, tapping hole 5, then return to point R.
G98 Y-750. ;     Position, tapping hole 6,
                 then return to the initial level.
G80 G28 G91 X0 Y0 Z0  ;   Return to the reference position return
M5 ;             Cause the spindle to stop rotating.
```

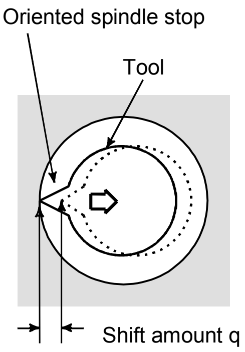
13.1.3 Fine Boring Cycle (G76)

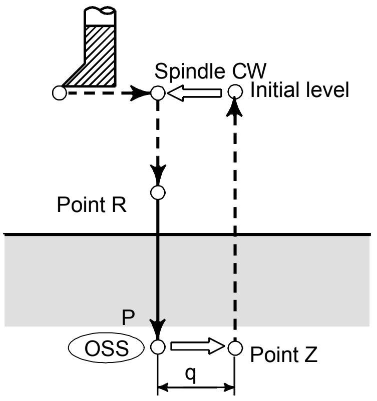
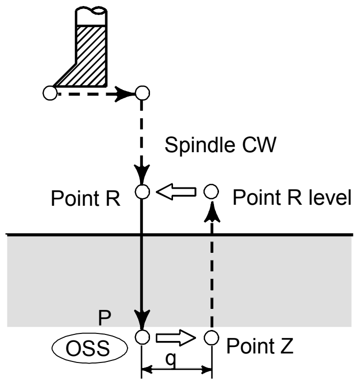
The fine boring cycle bores a hole precisely.
When the bottom of the hole has been reached, the spindle stops, and the tool is moved away from the machined surface of the workpiece and retracted.

Format

```
G76 X_ Y_ Z_ R_ I_ J_ P_ F_ L_ ;(When the parameter SIJ (No. 6200#2) is 1)
 or
G76 X_ Y_ Z_ R_ Q_ P_ F_ L_ ;(When the parameter SIJ (No. 6200#2) is 0)
```

X_ Y_	: Hole position data
Z_	: The distance from point R to the bottom of the hole
R_	: The distance from the initial level to point R level
I_ J_	: Shift amount at the bottom of a hole
Q_	: Shift amount at the bottom of a hole
P_	: Dwell time at the bottom of a hole
F_	: Cutting feedrate
L_	: Number of repeats (if required)



G76(G98)	G76(G99)
	

Explanation

- Operation

When the bottom of the hole has been reached, the spindle is stopped at the fixed rotation position, and the tool is moved in the direction opposite to the tool tip and retracted. This ensures that the machined surface is not damaged and enables precise and efficient boring to be performed.

- Spindle rotation

Before specifying G76, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G76 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

- Shift specification by I_J_

When bit 2 (SIJ) of parameter No. 6200 is set to 1, specify a shift with I_J_. Specify a shift for each positioning plane as described below.
Assume the following:
Xp : X-axis or axis parallel with the X-axis
Yp : Y-axis or axis parallel with the Y-axis
Zp : Z-axis or axis parallel with the Z-axis
Then, specify a shift as follows:
G17 (XpYp plane): Use I and J for specification.
G18 (ZpXp plane): Use K and I for specification.
G19 (YpZp plane): Use J and K for specification.
For example, when the XY plane is selected, a shift is made with linear interpolation along the X- and Y-axes by the incremental value specified by I and J. This means that a shift is made in an arbitrary direction on the positioning plane. The feedrate specified in F is used. I, J, and K are modal in a canned cycle. Specify I, J, and K in a block containing hole position data.

- Shift specification by Q_

When bit 2 (SIJ) of parameter No. 6200 is set to 0, specify a shift with Q_. Specify a positive value for Q. If a negative value is specified, the sign is ignored. Set the shift direction, +X, -X, +Y, or -Y, in parameter No. 6240 beforehand.

⚠ **CAUTION**

Q (shift at the bottom of a hole) is modal information in a canned cycle, and is also used to specify a depth of cut with G73 and G83.

NOTE

G76 can be used only when canned cycle II is set (bit 0 (FXB) of parameter No. 6201 is set to 1).

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any additional axes, drilling is not performed.

- I,J,KQ,P

Specify I, J, K, Q, and P in a block that performs hole machining. If any of these codes are specified in a block that does not perform hole machining, the codes are not stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G76 in a single block. Otherwise, G76 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

```
M3 S500 ;           Cause the spindle to start rotating.
G90 G99 G76 X300. Y-250.
  Z- 150. R-120. Q5.Position, bore hole 1, then return to point R
  P1000 F120. ;     Orient at the bottom of the hole, then shift by 5mm.
                    Stop at the bottom of the hole for 1 s
Y-550. ;            Position, drill hole 2, then return to point R.
Y-750. ;            Position, drill hole 3, then return to point R.
X1000. ;            Position, drill hole 4, then return to point R.
Y-550. ;            Position, drill hole 5, then return to point R.
G98 Y-750. ;        Position, drill hole 6,
                    then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;  Return to the reference position return
M5 ;                Cause the spindle to stop rotating.
```

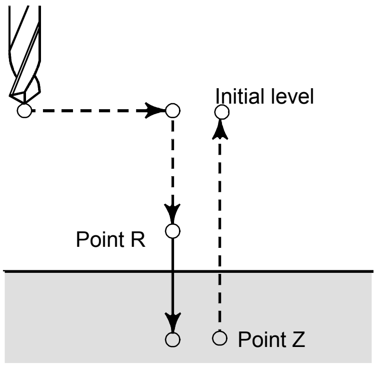
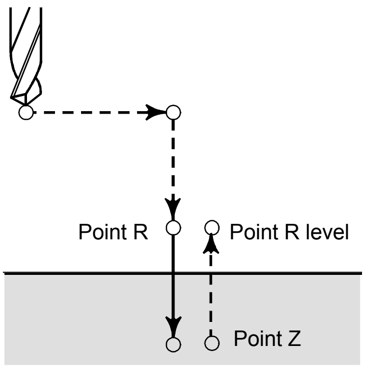
13.1.4 Drilling Cycle, Spot Drilling (G81)

This cycle is used for normal drilling.
Cutting feed is performed to the bottom of the hole. The tool is then retracted from the bottom of the hole in rapid traverse.

Format

G81 X_ Y_ Z_ R_ F_ L_ ;

X_ Y_	: Hole position data
Z_	: The distance from point R to the bottom of the hole
R_	: The distance from the initial level to point R level
F_	: Cutting feedrate
L_	: Number of repeats (if required)

G81(G98)	G81(G99)
	

Explanation

- Operation

After positioning along the X- and Y-axes, rapid traverse is performed to point R.
Drilling is performed from point R to point Z.
The tool is then retracted in rapid traverse.

- Spindle rotation

Before specifying G81, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G81 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is performed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Restriction

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G81 in a single block. Otherwise, G81 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

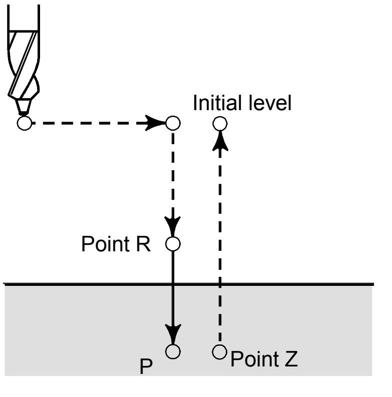
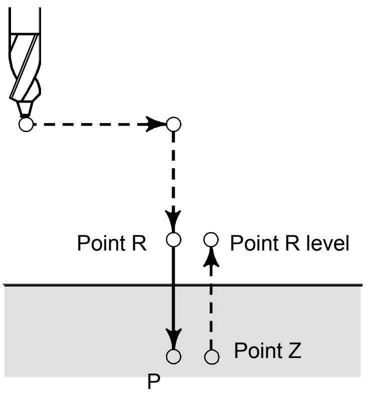
```
M3 S2000 ;      Cause the spindle to start rotating.
G90 G99 G81 X300. Y-250. Z-150. R -100. F120. ;
                Position, drill hole 1, then return to point R.
Y-550. ;        Position, drill hole 2, then return to point R.
Y-750. ;        Position, drill hole 3, then return to point R.
X1000. ;        Position, drill hole 4, then return to point R.
Y-550. ;        Position, drill hole 5, then return to point R.
G98 Y-750. ;    Position, drill hole 6, then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return
M5 ;            Cause the spindle to stop rotating.
```

13.1.5 Drilling Cycle Counter Boring Cycle (G82)

This cycle is used for normal drilling.
Cutting feed is performed to the bottom of the hole. At the bottom, a dwell is performed, then the tool is retracted in rapid traverse.
This cycle is used to drill holes more accurately with respect to depth.

Format

```
G82 X_ Y_ Z_ R_ P_ F_ L_ ;

  X_ Y_     : Hole position data
  Z_        : The distance from point R to the bottom of the hole
  R_        : The distance from the initial level to point R level
  P_        : Dwell time at the bottom of a hole
  F_        : Cutting feed rate
  L_        : Number of repeats (if required)
```

G82(G98)	G82(G99)
	

Explanation

- Operation

After positioning along the X- and Y-axes, rapid traverse is performed to point R.
Drilling is then performed from point R to point Z.
When the bottom of the hole has been reached, a dwell is performed.
The tool is then retracted in rapid traverse.

- Spindle rotation

Before specifying G82, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G82 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Restriction

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- P

Specify P in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G81 in a single block. Otherwise, G81 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

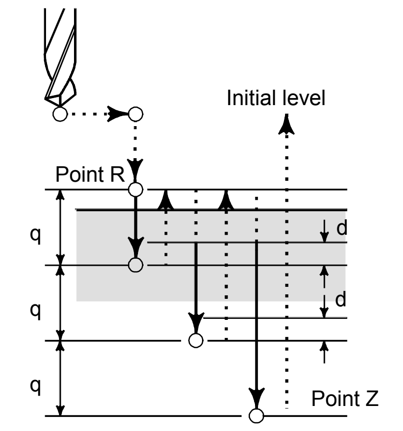
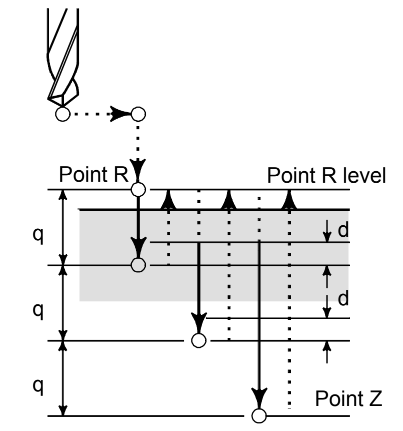
```
M3 S2000 ;          Cause the spindle to start rotating.
G90 G99 G82 X300. Y-250. Z-150. R -100. P1000 F120. ;
                    Position, drill hole 2, and dwell for 1 s at the
                    bottom of the hole, then return to point R.
Y-550. ;            Position, drill hole 2, then return to point R.
Y-750. ;            Position, drill hole 3, then return to point R.
X1000. ;            Position, drill hole 4, then return to point R.
Y-550. ;            Position, drill hole 5, then return to point R.
G98 Y-750. ;        Position, drill hole 6,
                    then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return
M5 ;                Cause the spindle to stop rotating.
```

13.1.6 Peck Drilling Cycle (G83)

This cycle performs peck drilling.
It performs intermittent cutting feed to the bottom of a hole while removing shavings from the hole.

Format

```
G83 X_ Y_ Z_ R_ Q_ F_ L_ ;

  X_ Y_     : Hole position data
  Z_        : The distance from point R to the bottom of the hole
  R_        : The distance from the initial level to point R level
  Q_        : Depth of cut for each cutting feed
  F_        : Cutting feedrate
  L_        : Number of repeats (if required)
```

G83(G98)	G83(G99)
	

Explanation

- Operation

Q represents the depth of cut for each cutting feed. It must always be specified as an incremental value.
In the second and subsequent cutting feeds, rapid traverse is performed up to a d point just before where the last drilling ended, and cutting feed is performed again. d is set in parameter (No.6211).
Be sure to specify a positive value in Q. Negative values are ignored.

- Spindle rotation

Before specifying G83, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G83 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation.
When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- Q

Specify Q in blocks that perform drilling. If they are specified in a block that does not perform drilling, they cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G82 in a single block. Otherwise, G82 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

```
M3 S2000 ;             Cause the spindle to start rotating.
G90 G99 G83 X300. Y-250. Z-150. R-100. Q15. F120. ;
                       Position, drill hole 1, then return to point R.
Y-550. ;               Position, drill hole 2, then return to point R.
Y-750. ;               Position, drill hole 3, then return to point R.
X1000. ;               Position, drill hole 4, then return to point R.
Y-550. ;               Position, drill hole 5, then return to point R.
G98 Y-750. ;           Position, drill hole 6,
                       then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;    Return to the reference position return
M5 ;                   Cause the spindle to stop rotating.
```

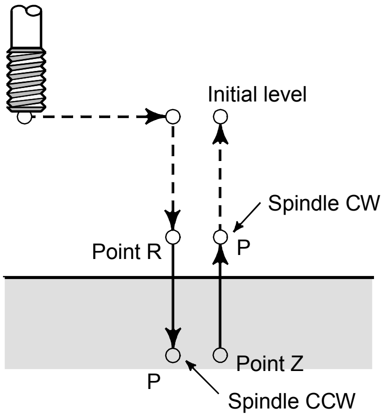
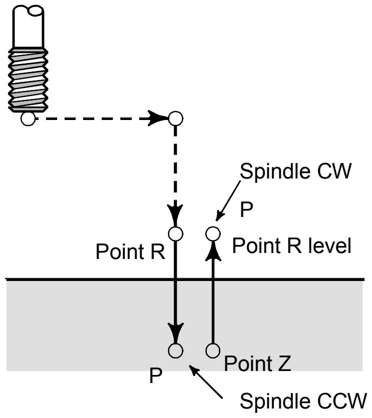
13.1.7 Tapping Cycle (G84)

This cycle performs tapping.
In this tapping cycle, when the bottom of the hole has been reached, the spindle is rotated in the reverse direction.

Format

G84 X_ Y_ Z_ R_ P_ F_ L_ ;

X_ Y_	: Hole position data
Z_	: The distance from point R to the bottom of the hole
R_	: The distance from the initial level to point R level
P_	: Dwell time
F_	: Cutting feedrate
L_	: Number of repents (if required)

G84(G98)	G84(G99)
	

Explanation

- Operation

Tapping is performed by rotating the spindle clockwise. When the bottom of the hole has been reached, the spindle is rotated in the reverse direction for retraction. This operation creates threads.

> ⚠ **CAUTION**
> Feedrate overrides are ignored during tapping. A feed hold does not stop the machine until the return operation is completed.

- Dwell

By setting bit 1 (DWL) of parameter No. 6200 to 1, dwell is performed for the duration specified with P before forward or reverse spindle rotation. This function is useful when a special tapper is used. (While dwell operation is being performed with no movement made along the Z-axis, a tapper performs forward/backward movement for threading only with a rotational force.)

- Spindle rotation

Before G84 is specified, turn the spindle in the reverse direction with a miscellaneous function (M code).
When successive hole machining operations which involve a short distance from a hole position and the initial level to the R point level, the spindle may not reach the normal speed before starting a hole cutting operation. In such a case, L for the number of repeats must not be specified. Instead, a dwell time based on G04 must be inserted before each hole machining operation. However, some machines do not require this specification. For details, refer to the manual provided by the machine tool builder.

- Miscellaneous function

When the G84 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation.
When the L is used to specify number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- P

Specify P in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G84 in a single block. Otherwise, G84 will be canceled.

Example

```
M3 S100 ;            Cause the spindle to start rotating.
G90 G99 G84 X300. Y-250. Z-150. R-120. P300 F120. ;
                     Position, drill hole 1, then return to point R.
Y-550. ;             Position, drill hole 2, then return to point R.
Y-750. ;             Position, drill hole 3, then return to point R.
X1000. ;             Position, drill hole 4, then return to point R.
Y-550. ;             Position, drill hole 5, then return to point R.
G98 Y-750. ;         Position, drill hole 6,
                     then return to the initial level.
G80 G28 G91 X0 Y0 Z0  ;   Return to the reference position return
M5 ;                 Cause the spindle to stop rotating.
```

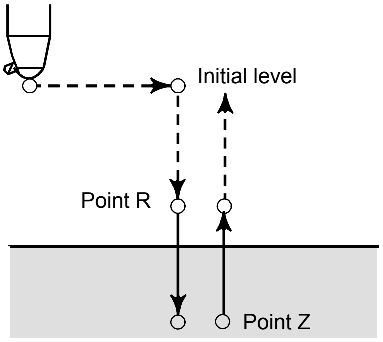
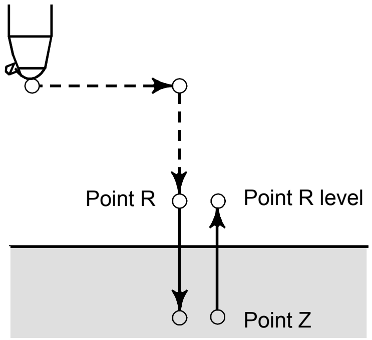
13.1.8 Boring Cycle (G85)

This cycle is used to bore a hole.

Format

G85 X_ Y_ Z_ R_ F_ L_ ;

X_ Y_	: Hole position data
Z_	: The distance from point R to the bottom of the hole
R_	: The distance from the initial level to point R level
F_	: Cutting feed rate
L_	: Number of repeats (if required)

G85(G98)	G85(G99)
	

Explanation

- Operation

After positioning along the X- and Y- axes, rapid traverse is performed to point R.
Drilling is performed from point R to point Z.
When point Z has been reached, cutting feed is performed to return to point R.

- Spindle rotation

Before specifying G85, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G85 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation.
When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not
executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G85 in a single block. Otherwise, G85 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

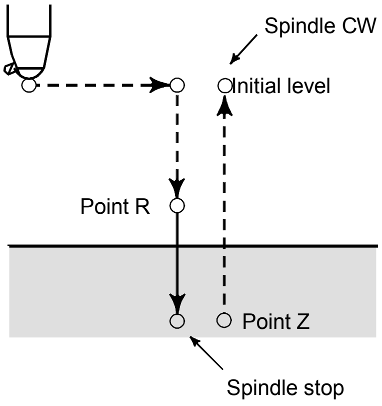
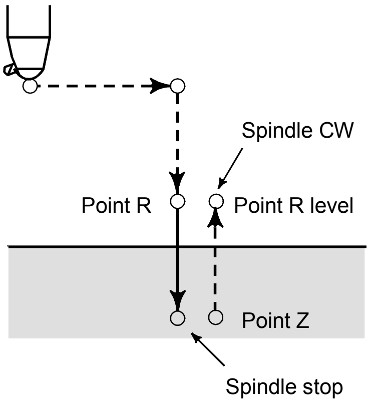
```
M3 S100 ;              Cause the spindle to start rotating.
G90 G99 G85 X300. Y-250. Z-150. R-120. F120. ;
                       Position, drill hole 1, then return to point R.
Y-550. ;               Position, drill hole 2, then return to point R.
Y-750. ;               Position, drill hole 3, then return to point R.
X1000. ;               Position, drill hole 4, then return to point R.
Y-550. ;               Position, drill hole 5, then return to point R.
G98 Y-750. ;           Position, drill hole 6,
                       then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return
M5 ;                   Cause the spindle to stop rotating.
```

13.1.9 Boring Cycle (G86)

This cycle is used to bore a hole.

Format

```
G86 X_ Y_ Z_ R_ F_ L_ ;

  X_ Y_    : Hole position data
  Z_       : The distance from point R to the bottom of the hole
  R_       : The distance from the initial level to point R level
  F_       : Cutting feed rate
  L_       : Number of repeats (if required)
```

G86(G98)	G86(G99)
	

Explanation

- Operation

After positioning along the X- and Y-axes, rapid traverse is performed to point R.
Drilling is performed from point R to point Z.
When the spindle is stopped at the bottom of the hole, the tool is retracted in rapid traverse.

- Spindle rotation

Before G86 is specified, turn the spindle in the reverse direction with a miscellaneous function (M code).
When successive hole machining operations which involve a short distance from a hole position and the initial level to the R point level, the spindle may not reach the normal speed before starting a hole cutting operation. In such a case, L for the number of repeats must not be specified. Instead, a dwell time based on G04 must be inserted before each hole machining operation. However, some machines do not require this specification. For details, refer to the manual provided by the machine tool builder.

- Miscellaneous function

When the G86 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G86 in a single block. Otherwise, G86 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

```
M3 S2000 ;          Cause the spindle to start rotating.
G90 G99 G86 X300. Y-250. Z-150. R-100. F120. ;
                    Position, drill hole 1, then return to point R.
Y-550. ;            Position, drill hole 2, then return to point R.
Y-750. ;            Position, drill hole 3, then return to point R.
X1000. ;            Position, drill hole 4, then return to point R.
Y-550. ;            Position, drill hole 5, then return to point R.
G98 Y-750. ;        Position, drill hole 6,
                    then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return
M5 ;                Cause the spindle to stop rotating.
```

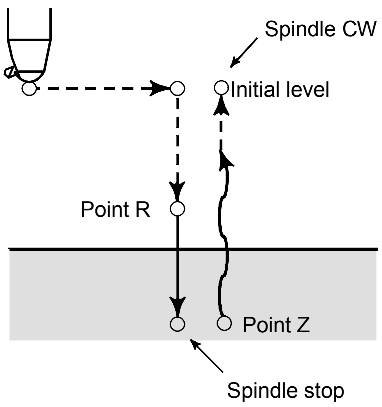
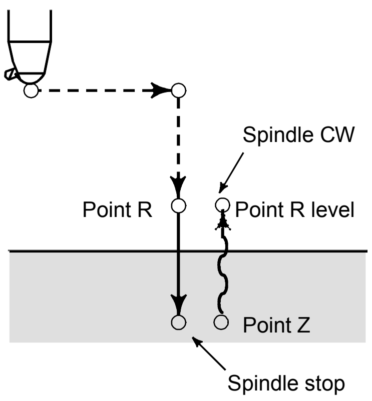
13.1.10 Boring Cycle/Back Boring Cycle (G87)

This cycle performs accurate boring.

Format

- Canned cycle I (boring cycle)

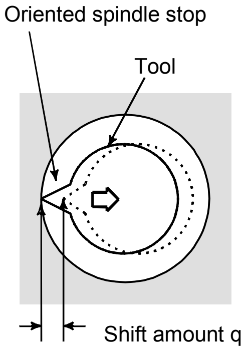
```
G87 X_ Y_ Z_ R_ F_ L_ ;

  X_ Y_     : Hole position data
  Z_        : The distance from point R to the bottom of the hole
  R_        : The distance from the initial level to point R level
  F_        : Cutting feed rate
  L_        : Number of repeats (if required)
```

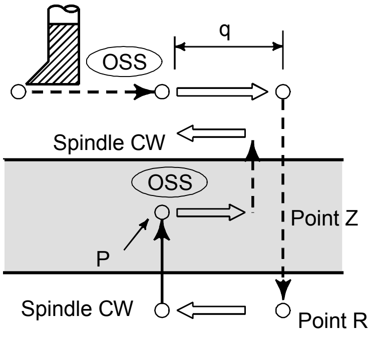
G87(G98)	G87(G99)
	

- Canned cycle II (back boring cycle)

G87 X_ Y_ Z_ R_ I_ J_ P_ F_ L_ ; (when the parameter SIJ(No.6200#2) is 1)
or
G87 X_ Y_ Z_ R_ Q_ P_ F_ L_ ; (when the parameter SIJ(No.6200#2) is 0)

X_ Y_	: Hole position data
Z_	: The distance from the bottom of the hole to point Z
R_	: The distance from the initial level to point R (the bottom of the hole) level
Q_	: Shift amount in the bottom of hole
I_, J_	: Shift amount in the bottom of hole
P_	: Dwell time
F_	: Cutting feed rate
L_	: Number of repeats (if required)



G87(G98)	G87(G99)
 q OSS Spindle CW OSS Point Z P Spindle CW Point R	Not used

Explanation

- In the case of canned cycle I (boring cycle)

After positioning along the X-axis and Y-axis, movement is made by rapid traverse to the R point level. Next, boring is performed from the R point level to point Z. Then, the spindle stops at the hole bottom. At this time, the mode can be switched to manual mode to move the tool. Any manual operation is allowed. For safety, however, the tool should ultimately be pulled out of the hole.

To restart machining, set either tape mode or memory mode. The spindle returns to the initial level or R point level according to G98 or G99, then starts rotating in the forward direction. Then, operation is restarted according to the command programmed in the next block.

- In the case of canned cycle II (back boring cycle)

After positioning along the X- and Y-axes, the spindle is stopped at the fixed rotation position.

The tool is moved in the direction opposite to the tool tip, positioning (rapid traverse) is performed to the bottom of the hole (point R).

The tool is then shifted in the direction of the tool tip and the spindle is rotated clockwise. Boring is performed in the positive direction along the Z-axis until point Z is reached.

At point Z, the spindle is stopped at the fixed rotation position again, the tool is shifted in the direction opposite to the tool tip, then the tool is returned to the initial level. The tool is then shifted in the direction of the tool tip and the spindle is rotated clockwise to proceed to the next block
operation.

NOTE

When bit 0 (FXB) of parameter No. 6201 is set to 0, canned cycle I is set. (Special signals are used as output signals for rotating the spindle in the forward direction and stopping the spindle.)
When bit 0 (FXB) of parameter No. 6201 is set to 1, canned cycle II is set. (M codes are used as output signals for rotating the spindle in the forward direction, stopping the spindle, and spindle orientation.)

- Spindle rotation

Before G87 is specified, turn the spindle in the reverse direction with a miscellaneous function (M code).
When successive hole machining operations which involve a short distance from a hole position and the initial level to the R point level, the spindle may not reach the normal speed before starting a hole cutting operation. In such a case, L for the number of repeats must not be specified. Instead, a dwell time based on G04 must be inserted before each hole machining operation. However, some machines do not require this specification. For details, refer to the manual provided by the machine tool builder.

- Miscellaneous function

When the G87 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

- Shift amount specification using I_J_

When bit 2 (SIJ) of parameter No. 6200 is set to 1, I_J_ is used to specify a shift amount. Depending on the positioning plane, specify a shift amount as described below. Let Xp, Yp, and Zp be as follows:

Xp : X-axis or an axis parallel with the X-axis
Yp : Y-axis or an axis parallel with the Y-axis
Zp : Z-axis or an axis parallel with the Z-axis

Then,

G17 (XpYp plane): To be specified by I and J

G18 (ZpXp plane): To be specified by K and I
G19 (YpZp plane): To be specified by J and K

When the XY plane is selected, for example, a shift is made along the X-axis and Y-axis by linear interpolation for an incremental amount specified by I and J. This means that a shift can be made in an arbitrary direction on a positioning plane. The feedrate specified by F is used. I, J, and K data is modal in a canned cycle. Specify I, J, and K in a block that specifies hole position data.

- Shift specification using Q_

When bit 2 (SIJ) of parameter No. 6200 is set to 0, a shift is specified using Q_. A Q value must always be positive. If a negative value is specified, the sign is ignored. With parameter No. 6240, a shift direction must be selected from +X, -X, +Y, and -Y.

> ⚠ **CAUTION**
> Q (shift amount at a hole bottom) is modal information in a canned cycle, and is also used as a depth of cut for G73 and G83.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Boring

In a block that does not contain X, Y, Z, R, or any additional axes, boring is not performed.

- I,J,K,Q,P

Specify I, J, K, Q, and P in a block that performs hole machining. If any of these codes are specified in a block that does not perform hole machining, the codes are not stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G87 in a single block. Otherwise, G87 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

```
M3 S500 ;              Cause the spindle to start rotating.
G90 G87 X300. Y-250.         Position, bore hole 1.
  Z-150. R-120. Q5. Orient at the initial level, then shift by 5 mm.
  P1000 F120. ;        Stop at point Z for 1 s.
Y-550. ;               Position, drill hole 2.
Y-750. ;               Position, drill hole 3.
X1000. ;               Position, drill hole 4.
Y-550. ;               Position, drill hole 5.
Y-750. ;               Position, drill hole 6
G80 G28 G91 X0 Y0 Z0 ;     Return to the reference position return
M5 ;                   Cause the spindle to stop rotating.
```

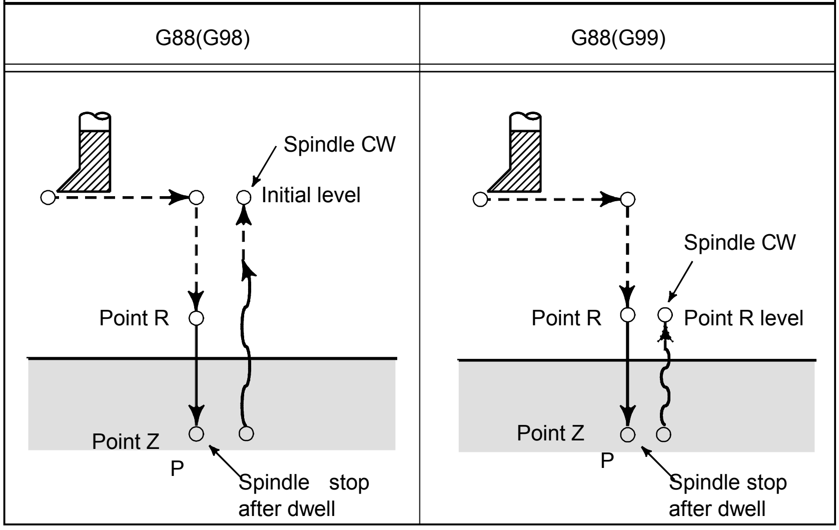
13.1.11 Boring Cycle (G88)

This cycle is used to bore a hole.

Format

```
G88 X_ Y_ Z_ R_ P_ F_ L_ ;
```

X_ Y_	: Hole position data
Z_	: The distance from point R to the bottom of the hole
R_	: The distance from the initial level to point R level
P_	: Dwell time at the bottom of a hole
F_	: Cutting feed rate
L_	: Number of repeats (if required)



Explanation

- Operation

After the tool is positioned along the X-axis and Y-axis, the tool is moved by rapid traverse to the R point level.
Then, boring is performed from the R point level to point Z.
Next, dwell is performed at the hole bottom, then the spindle stops and is placed in the hold state. At this time, the mode can be switched to manual mode to move the tool manually. Any manual operation can be performed. For safety, however, the tool should ultimately be pulled out of the hole.
When restarting machining, use either tape mode or memory mode. Then, the tool returns to the initial level or R point level according to G98 or G99, then the spindle rotates in the forward direction. Next, operation is restarted according to the program command in the next block.

- Spindle rotation

Before specifying G88, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G88 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- P

Specify P in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G88 in a single block. Otherwise, G88 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Example

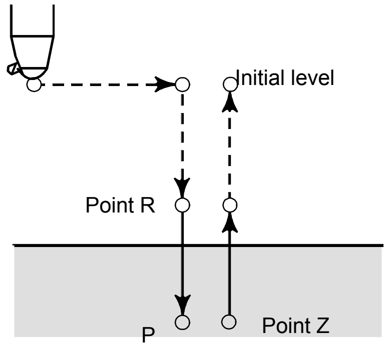
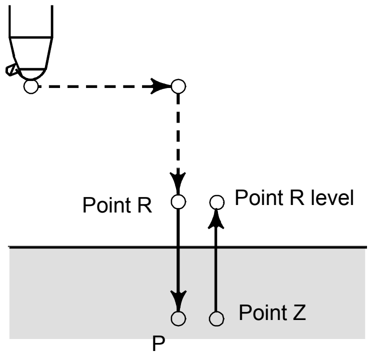
```
M3 S2000 ;         Cause the spindle to start rotating.
G90 G99 G88 X300. Y-250. Z-150. R-100. P1000 F120. ;
                   Position, drill hole 1, return to point R
                   then stop at the bottom of the hole for 1 s.
Y-550. ;           Position, drill hole 2, then return to point R.
Y-750. ;           Position, drill hole 3, then return to point R.
X1000. ;           Position, drill hole 4, then return to point R.
Y-550. ;           Position, drill hole 5, then return to point R.
G98 Y-750. ;       Position, drill hole 6,
                   then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;  Return to the reference position return
M5 ;               Cause the spindle to stop rotating.
```

13.1.12 Boring Cycle (G89)

This cycle is used to bore a hole.

Format

```
G89 X_ Y_ Z_ R_ P_ F_ L_ ;

  X_ Y_:   Hole position data
  Z_   :   The distance from point R to the bottom of the hole
  R_   :   The distance from the initial level to point R level
  P_   :   Dwell time at the bottom of a hole
  F_   :   Cutting feed rate
  L_   :   Number of repeats (if required)
```

G89(G98)	G89(G99)
 	

Explanation

- Operation

This cycle is almost the same as G85. The difference is that this cycle performs a dwell at the bottom of the hole.

- Spindle rotation

Before specifying G89, use a miscellaneous function (M code) to rotate the spindle.

- Miscellaneous function

When the G89 command and an M code are specified in the same block, the M code is executed at the time of the first positioning operation. When L is used to specify the number of repeats, the M code is executed for the first hole only; for the second and subsequent holes, the M code is not executed.

- Tool length compensation

When a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Drilling

In a block that does not contain X, Y, Z, R, or any other axes, drilling is not performed.

- P

Specify P in blocks that perform drilling. If it is specified in a block that does not perform drilling, it cannot be stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G89 in a single block. Otherwise, G89 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

Examples

```
M3 S100 ;          Cause the spindle to start rotating.
G90 G99 G89 X300. Y-250. Z-150. R-120. P1000 F120. ;
                   Position, drill hole 1, return to point R
                   then stop at the bottom of the hole for 1 s.
Y-550. ;           Position, drill hole 2, then return to point R.
Y-750. ;           Position, drill hole 3, then return to point R.
X1000. ;           Position, drill hole 4, then return to point R.
Y-550. ;           Position, drill hole 5, then return to point R.
G98 Y-750. ;       Position, drill hole 6,
                   then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return
M5 ;               Cause the spindle to stop rotating.
```

13.1.13 Canned Cycle Cancel (G80)

G80 cancels canned cycles.

Format

```
G80 ;
```

Explanation

All canned cycles are canceled to perform normal operation.
This means that R = 0 and Z = 0 in incremental mode. Other drilling data is also canceled (cleared).

Example

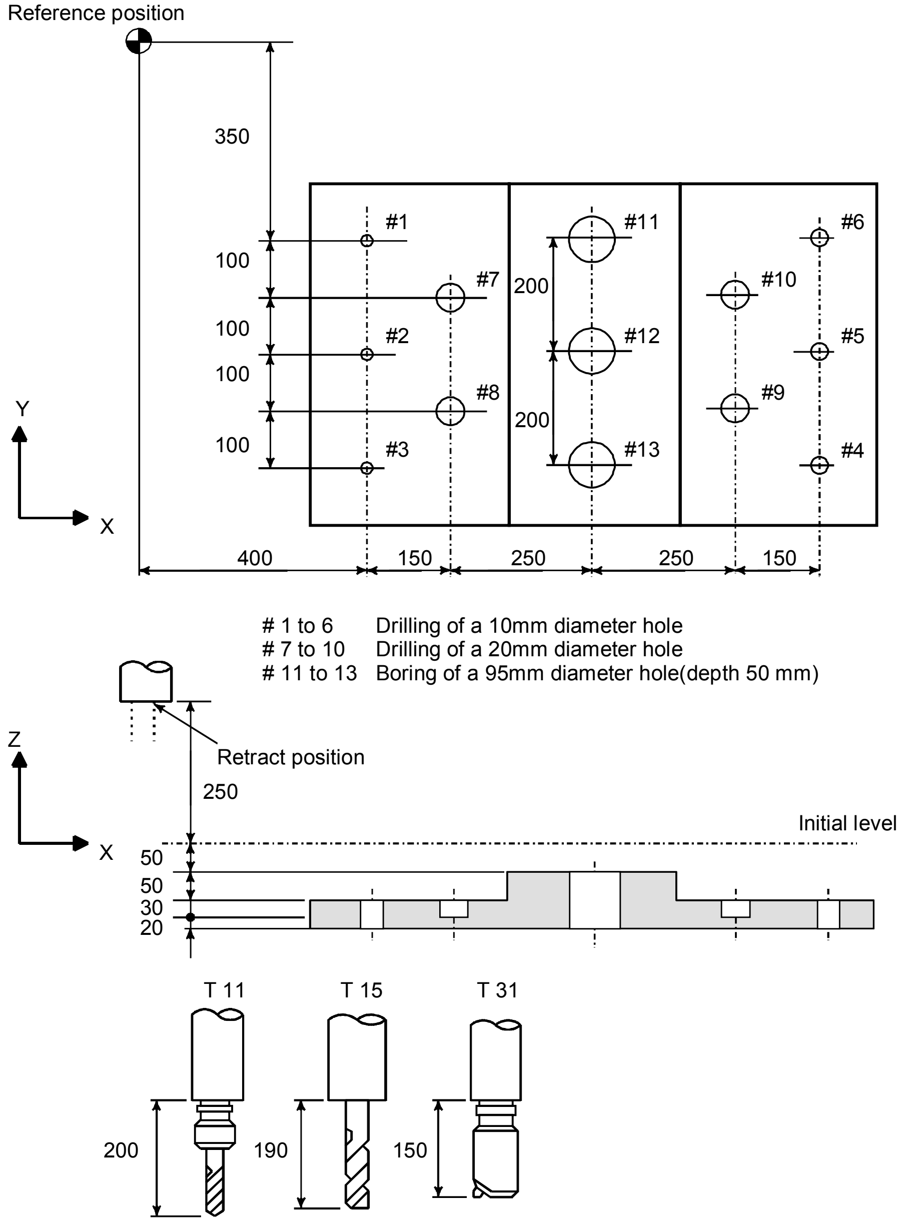
```
M3 S100 ;            Cause the spindle to start rotating.
G90 G99 G88 X300. Y-250. Z-150. R-120. F120. ;
                     Position, drill hole 1, then return to point R.
Y-550. ;             Position, drill hole 2, then return to point R.
Y-750. ;             Position, drill hole 3, then return to point R.
X1000. ;             Position, drill hole 4, then return to point R.
Y-550. ;             Position, drill hole 5, then return to point R.
G98 Y-750. ;         Position, drill hole 6,
                     then return to the initial level.
G80 G28 G91 X0 Y0 Z0 ;   Return to the reference position return,
                         canned cycle cancel
M5 ;                 Cause the spindle to stop rotating.
```

13.1.14 Example of Canned Cycle

Offset value +200.0 is set in offset No.11, +190.0 is set in offset No.15, and +150.0 is set in offset No.31

```
N001G92X0Y0Z0;                                  Coordinate setting at reference position
N002G90G00Z250.0T11M6;                          Tool change
N003G43Z0H11;                                   Initial level, tool length offset
N004S30M3;                                      Spindle start
N005G99G81X400.0Y-350.0Z-153.0R-97.0F120;       Positioning, then #1 drilling
N006Y-550.0;                                    Positioning, then #2 drilling and point R level return
N007G98Y-750.0;                                 Positioning, then #3 drilling and initial level return
N008G99X1200.0;                                 Positioning, then #4 drilling and point R level return
N009Y-550.0;                                    Positioning, then #5 drilling and point R level return
N010G98Y-350.0;                                 Positioning, then #6 drilling and initial level return
N011G00X0Y0M5;                                  Reference position return, spindle stop
N012G49Z250.0T15M6;                             Tool length offset cancel, tool change
N013G43Z0H15;                                   Initial level, tool length offset
N014S20M3;                                      Spindle start
N015G99G82X550.0Y-450.0 Z-130.0R-97.0P300F70;   Positioning, then #7 drilling, point R level return
N016G98Y-650.0;                                 Positioning, then #8 drilling, initial level return
N017G99X1050.0;                                 Positioning, then #9 drilling, point R level return
N018G98Y-450.0;                                 Positioning, then #10 drilling, initial level return
N019G00X0Y0M5;                                  Reference position return, spindle stop
N020G49Z250.0T31M6;                             Tool length offset cancel, tool change
N021G43Z0H31;                                   Initial level, tool length offset
N022S10M3;                                      Spindle start
N023G85G99X800.0Y-350.0 Z-153.0R47.0F50;        Positioning, then #11 drilling, point R level return
N024G91Y-200.0L2;                               Positioning, then #12, 13 drilling. point R level return
N025G28X0Y0M5;                                  Reference position return, spindle stop
N026G49Z0;                                      Tool length offset cancel
N027M0;                                         Program stop
```

Program example using tool length offset and canned cycles



# 1 to 6	Drilling of a 10mm diameter hole
# 7 to 10	Drilling of a 20mm diameter hole
# 11 to 13	Boring of a 95mm diameter hole(depth 50 mm)

Fig. 13.1.14 Example of canned cycle

13.2 RIGID TAPPING

In tapping, an amount of travel per spindle revolution along the Z-axis must match the screw pitch of the tapper. This means that the optimum tapping must satisfy the following condition:

P = F/S, where
P: Tapper screw pitch (mm, inch)
F: Feedrate along the Z-axis (mm/min, inch/min)
S: Spindle speed (min^{-1})

In the tapping cycle (G84) and reverse tapping cycle (G74), spindle speed control is exercised independently of Z-axis feed control. This means that the above condition is not always satisfied. At a hole bottom, particularly, both spindle rotation and Z-axis feed are stopped after deceleration, then are restarted by acceleration in the opposite direction. Acceleration/deceleration operations for these motions are performed independently of each other, so that the above condition is not usually satisfied. So, in general, feed compensation is performed by installing a spring inside the tapper holder to improve the tapping precision.

On the other hand, in the rigid tapping cycle (G84.2) and reverse rigid tapping cycle (G84.3), spindle rotation and Z-axis feed are controlled so that these two motions are synchronized with each other at all times. That is, only speed-related control is applied to the spindle for normal rotation. For rigid tapping, however, position control is applied to the spindle rotation, so that spindle rotation and Z-axis feed are controlled by two-axis linear interpolation. Thus, the condition P = F/S is satisfied even upon decelerating/accelerating at a hole bottom to achieve high-precision tapping.

13.2.1 Rigid Tapping (G84.2)

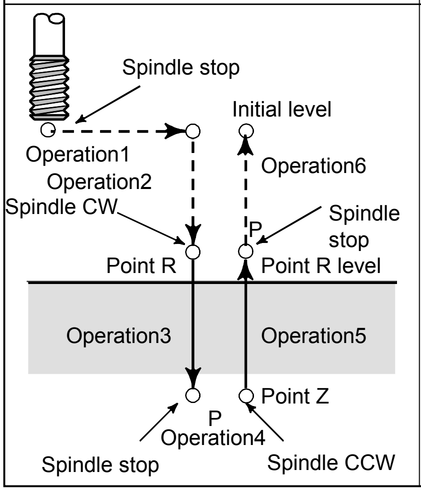
When the spindle motor is controlled as if it were a servo motor, a tapping cycle can be sped up.

The only difference from the reverse rigid tapping cycle (G84.3) is the spindle rotation direction in tapping.

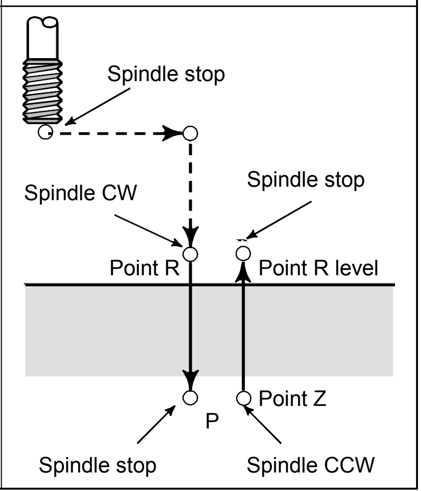
Format

```
G84.2 X_ Y_ Z_ R_ P_ F_ L_ S_ ;
  X_ Y_:  Hole position data
  Z_  : The distance from point R to the bottom of the hole and the position of
        the bottom of the hole
  R_  :  The distance from the initial level to point R level
  P_  : Dwell time at the bottom of the hole and at point R when a return is
        made
  F_  : Cutting feedrate
  L_  : Number of repeats (Only for necessity of repeat)
  S_  : Spindle speed
```

G84.2(G98)



G84.2(G99)



Explanation

After positioning along the X- and Y-axes, rapid traverse is performed to point R.

Tapping is performed from point R to point Z. When tapping is completed, the spindle is stopped and a dwell is performed . The spindle is then rotated in the reverse direction, the tool is retracted to point R, then the spindle is stopped. Rapid traverse to initial level is then performed for G98.

While tapping is being performed, the feedrate override and spindle override are assumed to be 100%.

However, the speed for extraction (operation 5) can be overridden by up to 200% depending on the setting at parameter No.5883.

- Thread lead

In feed-per-minute mode, the thread lead is obtained from the expression, feedrate y spindle speed. In feed-per-revolution mode, the thread lead equals the feedrate speed.

- Tool length compensation

If a tool length compensation (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled. If the drilling axis is changed in rigid mode, P/S alarm (No. 206) is issued.

- Hole machining

Hole machining is not performed by a block that does not contain X, Y, Z, R, and additional axes.

- P

Specify P in a block that performs drilling. If R is specified in a non-drilling block, it is not stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) is set to 1)) and G84.2 in a single block. Otherwise, G84.2 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

- Amount of movement and feedrate

The table below indicates an amount of travel along the Z-axis and the amount of spindle movement based on linear interpolation, and feedrate/speed.

Table 13.2.1 (a) Amounts of Movement and Feedrates

	Amount of movement	Feedrate/speed
Z-axis	z = Distance from point R to point Z (mm, inch)	Fz = F command value (mm/min, inch/min)
Spindle	s = z×(S command value/F command value)×360 (deg)	Fs = S command value (min^{-1})

- Feedrate command

As indicated in the table below, the function of an F command with a decimal point depends on the setting of bit 3 (RFA) of parameter No. 6201 and bit 7 (RFE) of parameter No. 6201.

Table 13.2.1 (b) Feedrate Command

Example of F command	**F10.0 or F10.0**	**F10.5**
RFA = 1, RFE = 0 or 1	F10.0	The PS530 alarm (incorrect use of a decimal point) is issued.
RFA=0, RFE=0	F10.0	F10.0
RFA=0, RFE=1	F10.0	F10.5

> ⚠ **CAUTION**
> If an inch input is used, an F command with no decimal point is assumed to have a decimal point between the second and third digits in it as counted from the least significant digit. If RFA=0/RFE=0 is specified, therefore, note the following point:

Example

Z-axis feedrate 1000 mm/min
Spindle speed 1000 min^{-1}
Thread lead 1.0 mm

```
<Programming of feed per minute>
G94 ;                                Specify a feed-per-minute
                                     command.
G00 X120.0 Y100.0 ;                  Positioning
G84.2 Z-100.0 R-20.0 F1000 S1000 ;   Rigid tapping
<Programming of feed per revolution>
G95 ;                                Specify a feed-per-revolution
                                     command.
G00 X120.0 Y100.0 ;                  Positioning
G84.2 Z-100.0 R-20.0 F1.0 S1000 ;    Rigid tapping
```

> ⚠ **CAUTION**
> Ensure that the tapper thread pitch matches a programmed pitch (F, S). Otherwise, the tool or a part being machined may be destroyed.

13.2.2 Left-handed Rigid Tapping Cycle (G84.3)

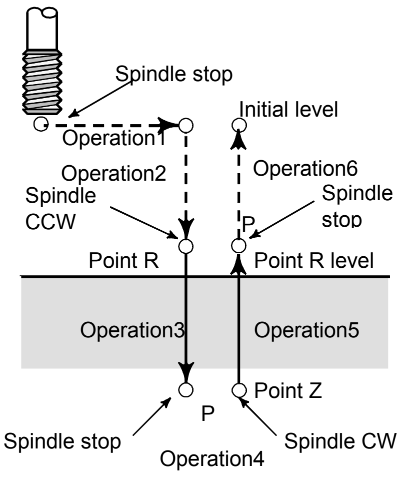
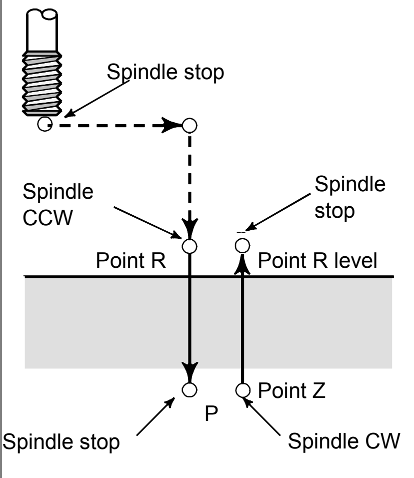
When the spindle motor is controlled as if it were a servo motor, tapping cycles can be sped up.

The only difference from the rigid tapping cycle (G84.2) is the spindle rotation direction during tapping.

Format

G84.3 X_ Y_ Z_ R_ P_ F_ L_ S_ ;

X_ Y_: Hole position data
Z_ : The distance from point R to the bottom of the hole and the position of the bottom of the hole
R_ : The distance from the initial level to point R level
P_ : Dwell time at the bottom of the hole and at point R when return is made.
F_ : Cutting feedrate
L_ : Number of repeats (Only for necessity of repeat)
S_ : Spindle speed

G84.3(G98)	G84.3(G99)
	

Explanation

After positioning along the X- and Y-axes, rapid traverse is performed to point R.

Tapping is performed from point R to point Z. When tapping is completed, the spindle is stopped and a dwell is performed. The spindle is then rotated in the normal direction, the tool is retracted to point R, then the spindle is stopped. Rapid traverse to initial level is then performed for G98.

While tapping is being performed, the feedrate override and spindle override are assumed to be 100%. However, the speed for extraction (operation 5) can be overridden by up to 200% depending on the setting at parameter 5883.

- Thread lead

In feed-per-minute mode, the thread lead is obtained from the expression, feedrate × spindle speed.

In feed-per-revolution mode, the thread lead equals the feedrate.

- Tool length compensation

If a tool length offset (G43, G44, or G49) is specified in the canned cycle, the offset is applied at the time of positioning to point R.

Limitation

- Axis switching

Before the drilling axis can be changed, the canned cycle must be canceled.

- Hole machining

Hole machining is not performed by a block that does not contain X, Y, Z, R, and additional axes.

- P

Specify P in a block that performs drilling. If R is specified in a non-drilling block, it is not stored as modal data.

- Cancel

Do not specify a G code of the 01 group (G00 to G03 or etc.) and G84.3 in a single block. Otherwise, G84.3 will be canceled.

- Tool offset

In the canned cycle mode, tool offsets are ignored.

- Amount of movement and feedrate

The table below indicates an amount of travel along the Z-axis and the amount of spindle movement based on linear interpolation, and feedrate/speed.

Table 13.2.2 (a) Amounts of Movement and Feedrates

	Amount of movement	Feedrate/speed
Z-axis	z = Distance from point R to point Z (mm, inch)	Fz = F command value (mm/min, inch/min)
Spindle	s = z × (S command value/F command value) × 360 (degrees)	Fs = S command value (min^{-1})

- Feedrate command

As indicated in the table below, the function of an F command with a decimal point depends on the setting of bit 3 (RFA) of parameter No. 6201 and bit 7 (RFE) of parameter No. 6201.

Table 13.2.2 (b) Feedrate Command

Example of F command	F10.0 or F10.0	F10.5
RFA = 1, RFE = 0 or 1	F10.0	The PS530 alarm (incorrect use of a decimal point) is issued.
RFA=0, RFE=0	F10.0	F10.0
RFA=0, RFE=1	F10.0	F10.5

> ⚠ **CAUTION**
> For inch inputs, an F command with no decimal point is assumed to have a decimal point in between its second and third places as counted from the lowest place. Note that the settings of RFA = 0 and RFE = 0 can produce the following example.
> Example) When F1234 is issued, F12.34 (inch/min) is assumed, resulting in F12.00 (inch/min).

Example

Z-axis feedrate 1000 mm/min
Spindle speed 1000 min^{-1}
Thread lead 1.0 mm

```
<Programming for feed per minute>
G94 ;                            Specify a feed-per-minute command.
G00 X120.0 Y100.0 ;                       Positioning
G84.3 Z-100.0 R-20.0 F1000 ;     Rigid tapping
<Programming for feed per revolution>
G95 ;                            Specify a feed-per-revolution
command.
G00 X120.0 Y100.0 ;              Positioning
G84.3 Z-100.0 R-20.0 F1.0 ;      Rigid tapping
```

> ⚠ **CAUTION**
> Ensure that the tapper thread pitch matches a programmed pitch (F, S). Otherwise, the tool or a part being machined may be destroyed.

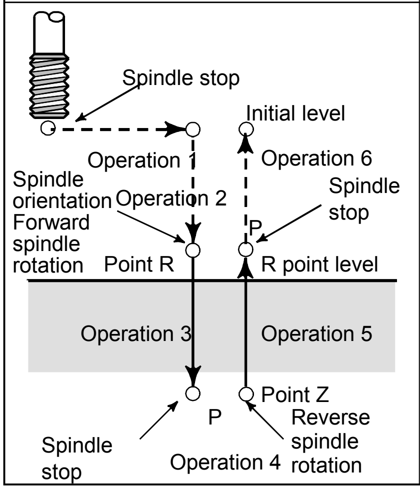
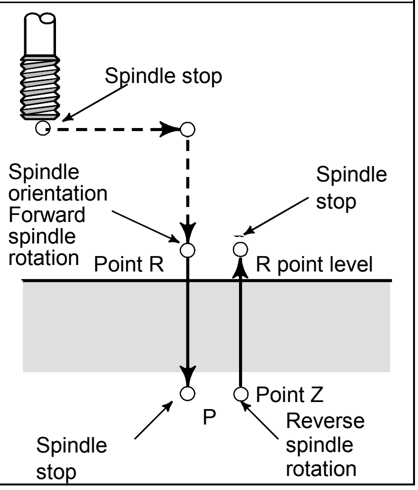
13.2.3 Rigid tapping Orientation Function

Before performing rigid tapping, the spindle can be oriented.

Format

G84.2 (or G84.3) X_Y_Z_R_P_F_L_I_ ;

X_ Y_: Hole position data
Z_ : Distance from point R to a hole bottom, and hole position
R_ : Distance from initial level to point R
P_ : Dwell duration at a hole bottom and upon return to point R
F_ : Cutting feedrate
L_ : Number of repeats (only when repeats are required)
I_ : Orientation command
0 : Positive-direction orientation
- 0 : Negative-direction orientation

G84.2 / G84.3 (G98)	G84.2 / G84.3 (G99)
	

Explanation

After positioning along the X-axis and Y-axis, movement is made by rapid traverse to the R point level.
When the R point level is reached, the spindle is oriented.
Then, tapping is performed from the R point level to point Z. Upon the completion of tapping, the spindle stops and dwells. Next, the spindle starts reverse rotation, is pulled up to the R point level, then stops. Then, if G98 is specified, movement is made by rapid traverse to the initial level.
During tapping, the feedrate override and spindle override are assumed to be 100%. However, a fixed override not exceeding 200% can be set for an extraction operation (operation 5) with parameter No. 5883.
(G84.2 and G84.3 differ only in the spindle rotation directions in operations 3 and 5.)

- Orientation operation and speed

1. For an analog spindle
   When orientation is specified, movement starts at the rapid traverse rate set in parameter No. 5977. Then, upon the detection of the one-rotation signal from the position coder, the feedrate is automatically reduced to the FL feedrate set in parameter No. 5979 to perform reference position return.
2. For a serial spindle
   The orientation feedrate depends on the spindle. With an analog spindle, the rapid traverse rate is reduced to an FL feedrate to stop at the orientation position. With a serial spindle, however, executing this command immediately stops the spindle at the orientation position.

- Grid shift

The grid shift function is used to shift the orientation stop position.

1. For an analog spindle
   Parameter No. 5980 is used for setting.
2. For a serial spindle
   Parameter No. 3073 is used for setting.

Limitation

The limitations imposed on the rigid tapping cycle and reverse rigid tapping cycle apply.

- Number of commands

The orientation command is valid only if it is specified in the first block in rigid tapping mode. All orientation commands specified in the second and subsequent blocks in rigid tapping mode are ignored.

Example

Feedrate along the Z-axis: 1000 mm/min
Spindle speed: 1000 min^{-1}
Screw lead: 1.0 mm
Positive-direction orientation

```
G94 ;                                     Command for feed per minute
G00 X120.0 Y100.0 ;                       Positioning
G84.2 Z-100.0 R-20.0 F1000 S1000 I0 ; Rigid tapping
```

⚠ **CAUTION**

1 The orientation function can be used only when the number of position coder pulses is 4096, and the ratio of the number of spindle gears to the number of position coder gears is:
1 :2^n (n : integer (0 or more))

2 When a serial interface spindle is used, the orientation direction is determined by the setting of bit 4 (SVO) of parameter No. 3000, regardless of the sign of I.

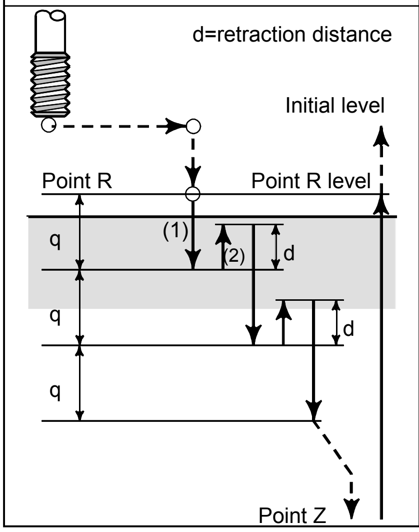
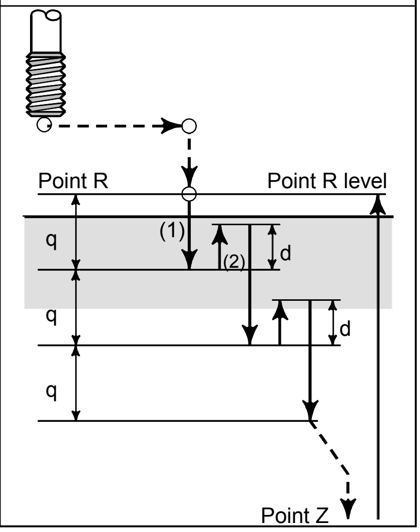
13.2.4 Peck Rigid Tapping Cycle (G84 or G74)

Tapping a deep hole in rigid tapping mode may be difficult due to chips sticking to the tool or increased cutting resistance. In such cases, the peck rigid tapping cycle is useful. So, the peck rigid tapping cycle function, which performs cutting several times from point R to point Z, can be specified.

Format

```
G84.2 (or G84.3) X_ Y_ Z_ R_ P_ Q_ F_ L_ S_ ;
  X_ Y_ : Hole position data
  Z_  : The distance from point R to the bottom of the hole and the position of the bottom of the hole
  R_  : The distance from the initial level to point R level
  P_  : Dwell time at the bottom of the hole and at point R when a return is made
  Q_  : Depth of cut for each cutting feed
  F_  : The cutting feedrate
  L_  : Number of repeats (if required)
  S_  : Spindle speed
```

- Peck rigid tapping cycle
  (1), the cutting feedrate specified by F is used.
  (2), the feedrate can be overridden by parameter setting.

G84.2 / G84.3 (G98)	G84.2 / G84.3 (G99)
	

Explanation

After positioning along the X- and Y-axes, rapid traverse is performed to point R. From point R, cutting is performed with depth Q (depth of cut for each cutting feed), then the tool is retracted by distance d. The parameter 5883 specifies whether retraction can be overridden or not. When point Z has been reached, the spindle is stopped, then rotated in the reverse direction for retraction.

- Retraction distance

The retraction distance to be moved each time is set by parameter No. 6221.

- Retraction feedrate

To the feedrate used for each retraction operation, an override value from 1% to 200% can be applied by using parameter No. 5883.
During rigid tapping, the retraction feedrate override function is enabled even in retraction from point Z to point R.

Limitation

The limitations set for the rigid tapping cycle and reverse rigid tapping cycle apply.

Example

Feedrate along the Z-axis:	1000 mm/min
Spindle speed:	1000 min^{-1}
Thread lead:	1.0 mm

```
G94 ;          Feed per minute command
G00 X120.0 Y100.0 ;                        Positioning
G84.2 Z-100.0 R-20.0 F1000 S1000 Q10.0 ; Rigid tapping
```

⚠ **CAUTION**
The peck rigid tapping cycle command for rigid tapping is enabled by the Q command in the rigid tapping cycle or reverse rigid tapping cycle mode or in a block specifying G84.2 (G84.3).

13.2.5 Three-dimensional rigid tapping

When the machine is provided with axes for swiveling the tool, this function allows rigid tapping in the direction in which the tool is pointing after the tool is swiveled about the specified axes. This three-dimensional rigid tapping function must be used together with the three- dimensional coordinate conversion function (G68) .

Format

The command for three- dimensional rigid tapping is specified in the same format as for usual rigid tapping.

Limitation

The limitations set for the rigid tapping cycle and reverse rigid tapping cycle apply.

Example

```
G68 X0 Y0 Z0 I1 J0 K0 R60. ; ........................ (1)
G68 X0 Y0 Z0 I0 J1 K0 R30. ; ........................ (1)’
G90 G00 A60. ; ...................................... (2)
G90 G00 B30. ; ...................................... (2)’
X100. Y100. Z100. ;
G91 G84.2 X10. Y10. Z-50. R-20. F12.345 S1000 ; ..... (3)
X20. ; .............................................. (4)
Y20. ; .............................................. (5)
X-20. ; ............................................. (6)
G80 ; ............................................... (7)
G90 G00 X0 Y0 Z0 ;
G69 ; ............................................... (8)
```

(1) Rotate the coordinate system about the X-axis 60_ by using three-dimensional coordinate conversion.
(1)’ Rotate the coordinate system about the Y-axis 30_ by using three-dimensional coordinate conversion.
(2) Rotate the A axis 60_.
(2)’ Rotate the B-axis 30_.
(3) Enter the rigid tapping mode. Perform positioning, then tap hole #1.
(4) Perform positioning, then tap hole #2.
(5) Perform positioning, then tap hole #3.
(6) Perform positioning, then tap hole #4.
(7) Cancel the rigid tapping mode.
(8) Cancel the three- dimensional coordinate conversion mode.

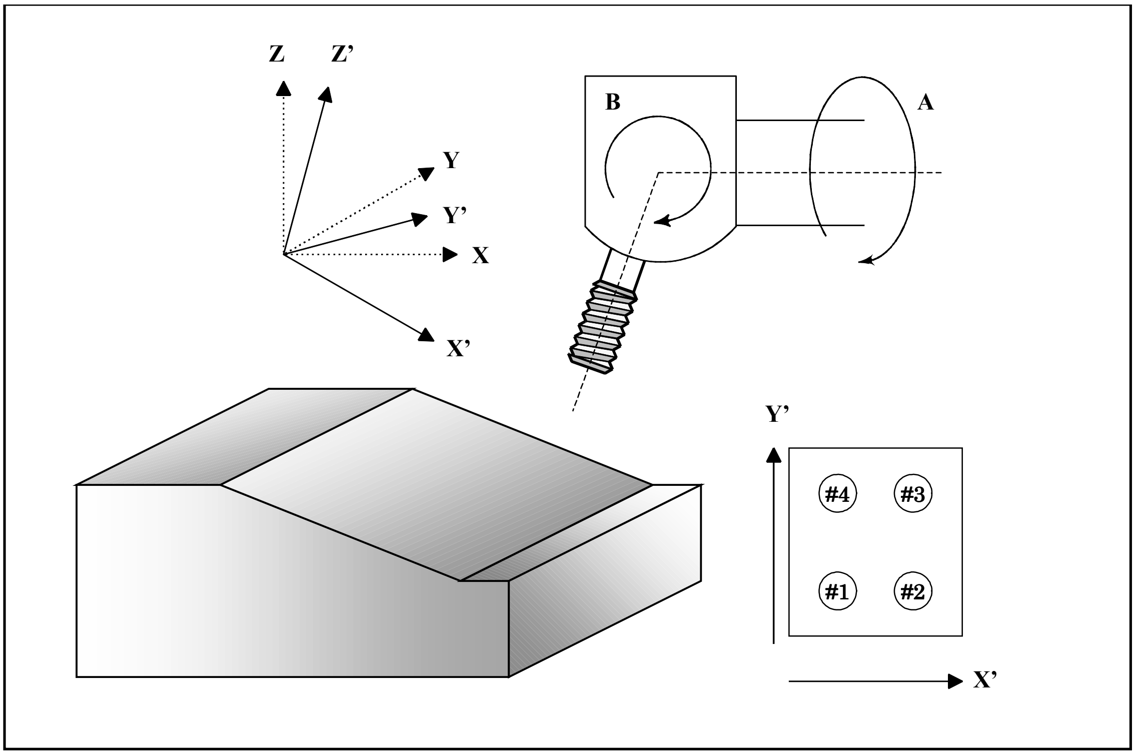


Fig. 13.2.5 Three-dimensional rigid tapping

13.3 EXTERNAL MOTION FUNCTION (G81)

Upon completion of positioning in each block in the program, an external operation function signal can be output to allow the machine to perform specific operation.
Concerning this operation, refer to the manual supplied by the machine tool builder.

Format

```
G81IP_L_;
  IP_ : Axis move command
  L_  : Number of repeats (if required)
```

Explanation

Every time positioning for the IP_ move command is completed, the CNC sends a external operation function signal to the machine. An external operation signal is output for each positioning operation until canceled by G80 or a group 01 G code.
Upon a reset or at power-up, the cancel state based on G80 arises.

Limitation

- Movement command

In a block which does not contain a move command, the external operation signal is not sent.

- Relationship with canned cycle G81

G81 can also be used for a drilling canned cycle. Whether G81 is to be used for an external motion function or for a drilling canned cycle is specified with EXC, bit 5 of parameter No.6200.

13.4 OPTIONAL ANGLE CHAMFERING AND CORNER ROUNDING

Chamfering and corner rounding blocks can be inserted automatically between the following:

- Between linear interpolation and linear interpolation blocks
- Between linear interpolation and circular interpolation blocks
- Between circular interpolation and linear interpolation blocks
- Between circular interpolation and circular interpolation blocks

Format

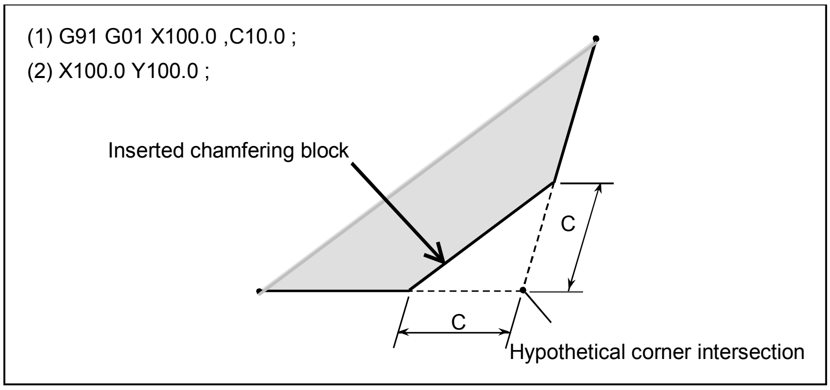
,C_	**Chamfering**
,R_	**Corner R**

Explanation

When the above specification is added to the end of a block that specifies linear interpolation (G01) or circular interpolation (G02 or G03), a chamfering or corner rounding block is inserted.
Blocks specifying chamfering and corner rounding can be specified consecutively.

- Chamfering

After C, specify the distance from the virtual corner point to the start and end points. The virtual corner point is the corner point that would exist if chamfering were not performed.

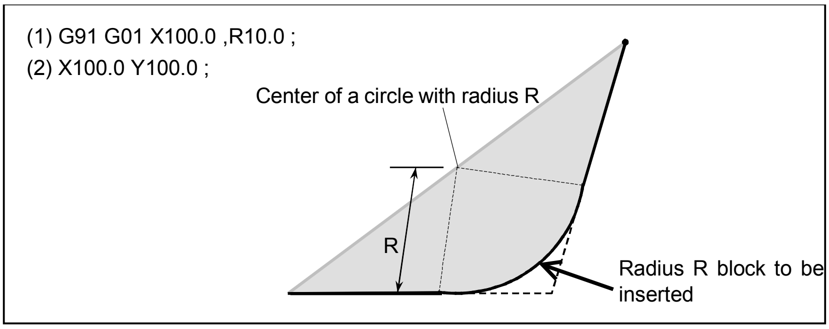


```
(1) G91 G01 X100.0 ,C10.0 ;
(2) X100.0 Y100.0 ;
```

Fig.13.4 (a) chamfering

- Corner R

After R, specify the radius for corner rounding.



```
(1) G91 G01 X100.0 ,R10.0 ;
(2) X100.0 Y100.0 ;
```

Center of a circle with radius R
R
Radius R block to be inserted

Fig.13.4 (b) Corner R

Limitation

- Next block

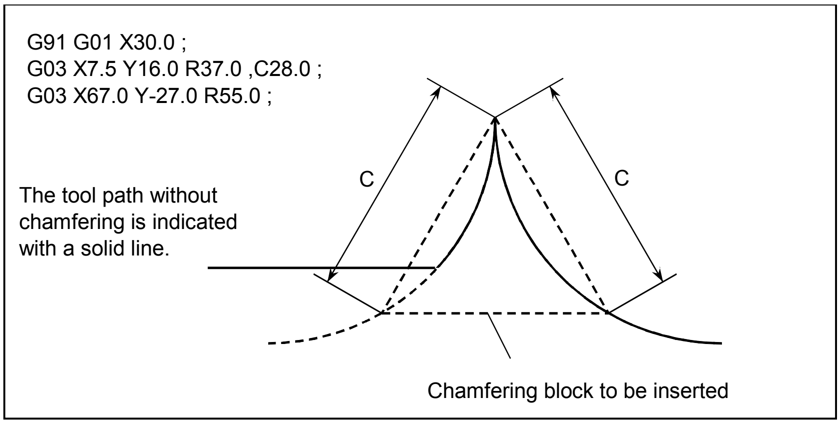
A block specifying chamfering or corner rounding must be followed by a block that specifies a move command using linear interpolation (G01) or circular interpolation (G02 or G03). If the next block does not contain these specifications, alarm PS0431 is issued.

- Plane switching

A chamfering or corner rounding block can be inserted only for move commands which are performed in the same plane. In a block that comes immediately after plane switching (G17, G18, or G19 is specified), neither chamfering nor corner rounding can be specified.

- Exceeding the move range

If the inserted chamfering or corner rounding block causes the tool to go beyond the original interpolation move range, alarm PS0429 is issued.



```
G91 G01 X30.0 ;
G03 X7.5 Y16.0 R37.0 ,C28.0 ;
G03 X67.0 Y-27.0 R55.0 ;
```

C
C
The tool path without chamfering is indicated with a solid line.
Chamfering block to be inserted

Fig.13.4 (c) Exceeding the move range

- Coordinate system

In a block that comes immediately after the coordinate system is changed (G92, or G52 to G59) or a return to the reference position (G28 to G30) is specified, neither chamfering nor corner rounding can be specified.

- Travel distance 0

When two linear interpolation operations are performed, the chamfering or corner rounding block is regarded as having a travel distance of zero if the angle between the two straight lines is within +/-1deg. When linear interpolation and circular interpolation operations are performed, the corner rounding block is regarded as having a travel distance of zero if the angle between the straight line and the tangent to the arc at the intersection is within +/-1deg. When two circular interpolation operations are performed, the corner rounding block is regarded as having a travel distance of zero if the angle between the tangents to the arcs at the intersection is within +/-1deg.

- Single block operation

If a block in which chamfering or corner R is specified is executed in single block mode, the block is executed up to the block end point, and a feed hold stop occurs at the end point.

- Invalid command

Chamfering (, C) or corner R (, R) is ignored if it is specified in a block that does not specify either linear interpolation (G01) or circular interpolation (G02 or G03).

Example

```
N001 G92 G90 X0 Y0 ;
N002 G00 X10.0 Y10.0 ;
N003 G01 X50.0 F10.0 ,C5.0 ;
N004 Y25.0 ,R8.0 ;
N005 G03 X80.0 Y50.0 R30.0 ,R8.0 ;
N006 G01 X50.0 ,R8.0 ;
N007 Y70.0 ,C5.0 ;
N008 X10.0 ,C5.0 ;
N009 Y10.0 ;
N010 G00 X0 Y0 ;
N011 M0 ;
```

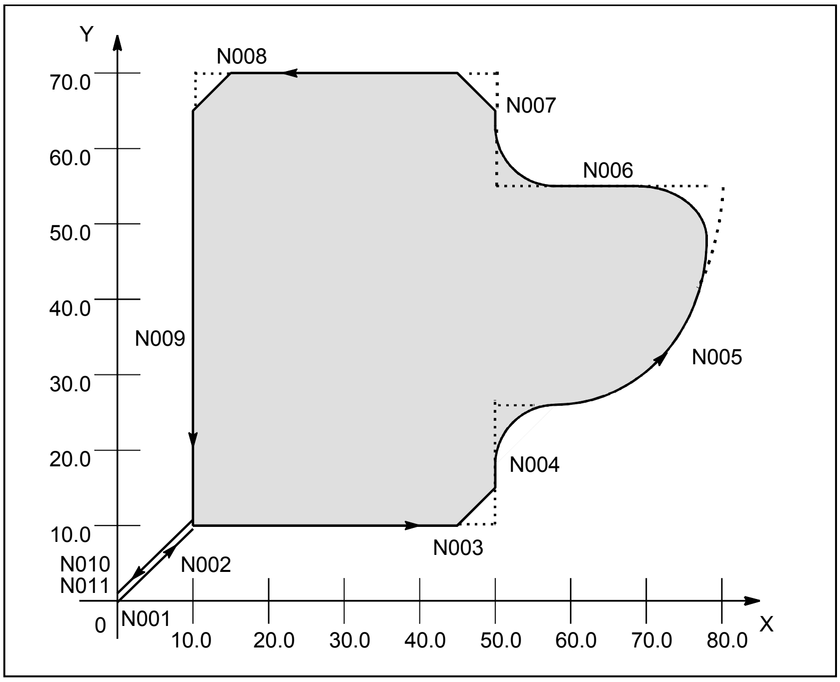
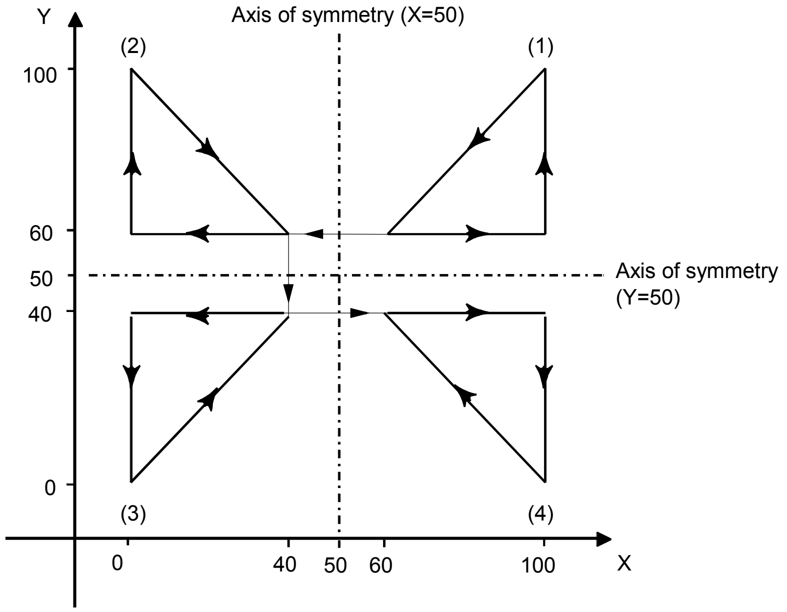


Fig.13.4 (d) Example of chamfering and corner R

13.5 PROGRAMMABLE MIRROR IMAGE (G50.1, G51.1)

By a programmed command, the mirror image function can be used for each axis.



(1) Original image of a programmed command
(2) Image symmetrical about a line parallel to the Y-axis and crossing the X-axis at 50
(3) Image symmetrical about point (50, 50)
(4) Image symmetrical about a line parallel to the X-axis and crossing the Y-axis at 50

Fig14.12 Programmable Mirror image

Format

G51.1 IP_ ;	Setting a programmable image
: **:**	A mirror image of a command specified in theseblocks is produced with respect to the axis of symmetry specified by G51.1 _;.
G50.1 IP_ ;	Canceling a programmable mirror image
IP_ :	Point (position) and axis of symmetry for producing a mirror image when specified with G51.1. Axis of symmetry for producing a mirror image when specified with G50.1. Point of symmetry is not specified.

Explanation

- Mirror image by setting

If the programmable mirror image function is specified when the command for producing a mirror image is also selected by a CNC external switch or CNC setting, the programmable mirror image function is executed first.

Limitation

- Scaling/coordinate system rotation

During coordinate system rotation (G68) or scaling (G51), the programmable mirror image commands (G50.1, G51.1) must not be specified.

⚠ **CAUTION**

1 When a mirror image is produced on the other side of one axis on a specified plane, other commands are processed as follows:
(1) Circular command:
The direction of rotation is reversed.
(2) Cutter compensation:
The direction of offset is reversed.
(3) Coordinate system rotation:
The direction of rotation is reversed.

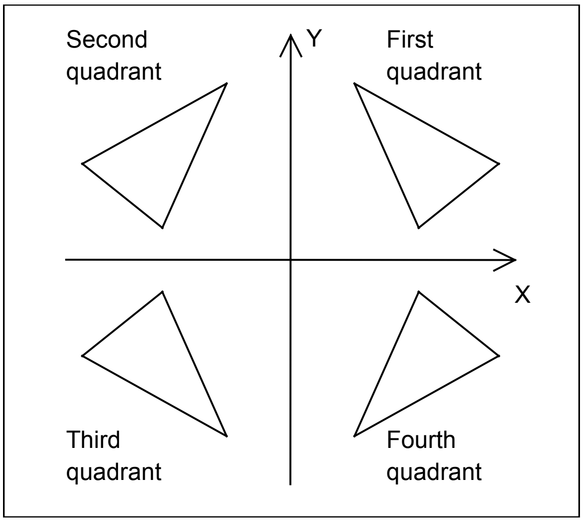
2 Relation between move command and machine travel/coordinate value is as follows in the programmable mirror image.

Movement command direction	Machine movement	Machine coordinate system	Relative coordinate system	Absolute coordinate system
+ direction	- direction	- direction	- direction	- direction
- direction	+ direction	+ direction	+ direction	+ direction

3 The first move command coming after G50.1 or G51.1 must be specified with absolute values.

- Three-dimensional cutter compensation / tool center point control

In mirror operation, there must be no conflict between the linear axes and rotation axes.



Example 1: XY Plane in a BC-Type Machine

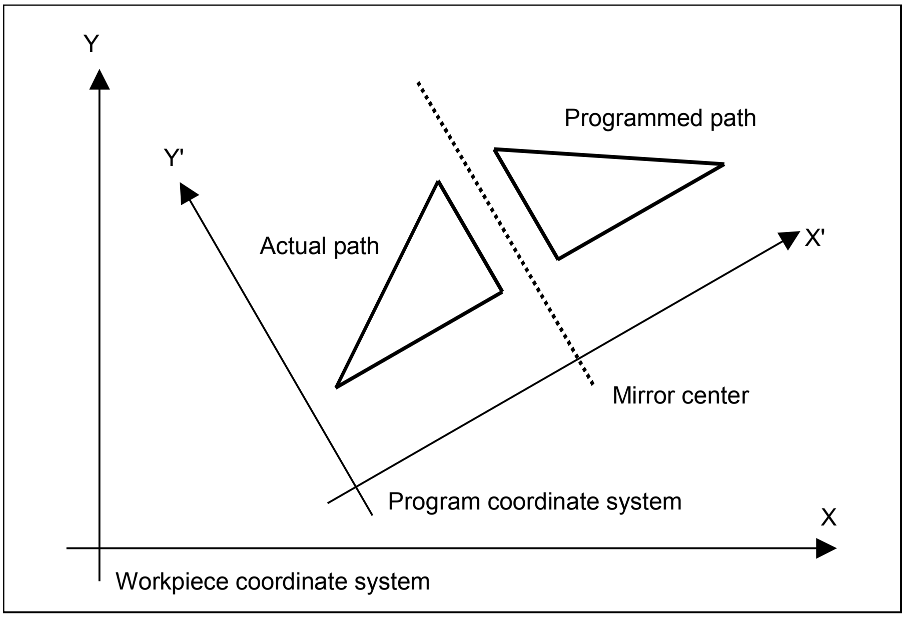
	X	Y	Z	B	C
First quadrant	Normal	Normal	Normal	Normal	Normal
Second quadrant	Mirror	Normal	Normal	Normal	Mirror 90° about center
Third quadrant	Mirror	Mirror	Normal	Mirror 0° about center	Normal
Fourth quadrant	Normal	Mirror	Normal	Normal	Mirror 0° about center

Example 2: XY Plane in an AB-Type Machine (with A being the Master)

	X	Y	Z	A	B
First quadrant	Normal	Normal	Normal	Normal	Normal
Second quadrant	Mirror	Normal	Normal	Normal	Mirror 0° about center
Third quadrant	Mirror	Mirror	Normal	Mirror 0° about center	Mirror 0° about center
Fourth quadrant	Normal	Mirror	Normal	Mirror 0° about center	Normal

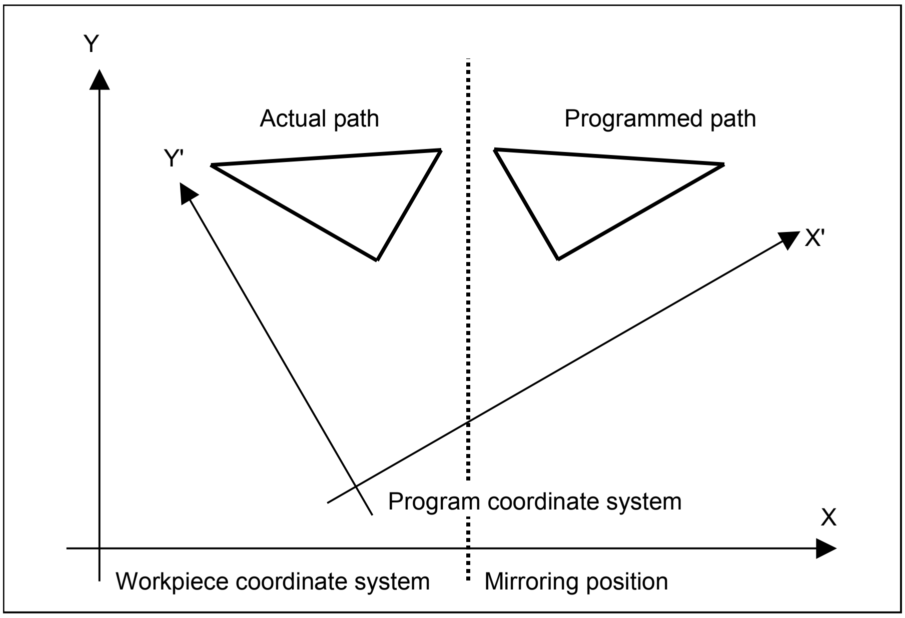
- Three-dimensional coordinate conversion

When three-dimensional coordinate conversion and programmable mirror image are used at the same time, programmable mirror image is applied to the coordinates in the program coordinate system, then three-dimensional coordinate conversion is performed.



- Three-dimensional coordinate conversion and external mirror image

When three-dimensional coordinate conversion and external mirror image are used at the same time, three-dimensional coordinate conversion is performed first, then external mirror image is applied.



Additional programmable mirror image functions

- A programmable mirror image will not be cleared with a reset (if bit 0 of parameter No. 6401 is 1).
- An alarm will be issued if a G code for three-dimensional coordinate conversion is specified in a programmable mirror image.
- When a programmable mirror image is effective, G41.1 and G42.1 for normal direction control will be automatically replaced with each other (if bit 2 of parameter No. 6401 is 0).
- If a programmable mirror image is applied to the rotation-direction-controlled axis (parameter No.1007#2(RSR)=1) controlled with the sign of multiple-rotary axis control, the rotation direction will be reversed with the sign (if bit 1 of parameter No. 6401 is 0).

13.6 INDEX TABLE INDEXING FUNCTION

By specifying indexing positions (angles) for the indexing axis (one arbitrary axis), the index table of the machining center can be indexed. Before and after indexing, the index table is automatically unclamped or clamped.

Explanation

- Indexing axis name

The name of an indexing axis is A, B, C, X, Y, Z, U, V, or W. For information about which is used, refer to the manual provided by the machine tool builder.

- Indexing axis setting

Set a rotation axis as an index table indexing axis. That is, set the following:

Bit 0 (ROT) of parameter No. 1006 = 1
Bit 1 (ROS) of parameter No. 1006 = 1
Bit 2 (ROP) of parameter No. 1006 = 1

Disable the servo-off signal for an index table indexing axis. That is, set the following:

Bit 1 (SVF) of parameter No. 1802 = 0

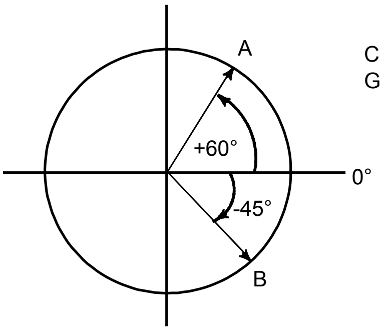
Usually, the servo-off state is set for an index table indexing axis.

- Absolute/incremental programming

One of the following indexing positions is used, depending on bit 3 (G90) of parameter No. 7602:

0: Value according to an absolute/incremental G code (G90/G91)
1: Absolute value at all times

An indexing position assumes a positive value for the counterclockwise direction, and a negative value for the clockwise direction.



Command value from A to B (when parameter G90 (bit 3 of parameter No. 7602) = 0)
G90 B-45.0; or
G91 B-105.0;

Fig. 14.6 Absolute/Incremental Programming

For information about whether an absolute or incremental value is used, refer to the manual provided by the machine tool builder.

- Indexing direction

If a value other than 0 is set in the M code for specifying negative direction rotation (parameter No.7632), movement in the negative direction is made only when a move command is specified together with the M code. In this case, movement is performed in the negative direction, regardless of whether absolute/incremental programming is used.

If 0 is set in the M code for specifying negative direction rotation (parameter No.7632), the rotation direction in G90 mode is determined by bit 4 (INC) of parameter No. 7602, as described below.

The rotation direction in G90 mode is
0: Not shortcut direction
1: Shortcut direction (The amount of movement is always less than 180 degrees.)

For information about which is used, refer to the manual provided by the machine tool builder.

- Minimum indexing angle

The minimum index table indexing angle set in parameter No. 7682 is used. An integral multiple of a set value can be specified as an indexing angle. If a value other than an integral multiple is specified, an alarm (PS0561) is issued.

- Feedrate

The table is always rotated around the indexing axis in rapid traverse mode.

Dry run cannot be executed for the indexing axis.

- Index table clamping/unclamping

Before or after index table indexing axis movement, the index table is automatically clamped or unclamped.

- Reset

If a reset is performed in the clamp or unclamp completion wait state, the clamp or unclamp signal is cleared, and the CNC exits from the completion wait state.

- Index table indexing function and other functions

Item	Explanation
Relative position display	This value can be rounded by setting bit 2 (REL) of parameter No. 7602.
Absolute position display	This value is rounded at all times.
Automatic return from the reference position (G29) Second reference position return (G30)	Return is impossible.
Machine coordinate system selection (G53)	No movement is allowed.
Single direction positioning (G60)	Not specifiable
Second auxiliary function	Ensure that a second auxiliary function axis name does not duplicate an indexing axis name.

Item	Explanation
Operation during index table indexing axis movement	Unless otherwise processed by the machine, feed hold, interlock, and emergency stop can be executed during index table indexing axis movement. Machine lock can be executed after indexing has been completed.
Servo-off signal	Disable the servo-off signal for the index table indexing axis. That is, set bit 1 (SVF) of parameter No. 1802 to 0. Usually, the index table indexing axis is in the servo-off state.
Incremental command for the index table indexing axis	When incremental programming is used for index table indexing (when bit 3 (G90) of parameter No. 7602 is set to 0), the workpiece origin offset of the index table axis must always be 0. That is, there must always be a match between the workpiece coordinate system and machine coordinate system of an index table indexing axis.
Operation for an index table indexing axis	Operation in jog/step/handle mode for an index table indexing axis is disabled. However, manual reference position return is possible. If the axis selection signal is set to 0 during manual reference position return, the movement stops immediately, and the clamp command is not executed.

Limitation

- Simultaneous specification together with other controlled axes

Specify an indexing command singly in a block. If an indexing command is specified in a block that specifies other controlled axes as well, an alarm (PS0564) is issued.

- Command specifying zero move amount

When the amount of movement is 0, a clamp/unclamp operation is not performed. In automatic reference position return based on G28, clamp/unclamp is performed even if the amount of movement is 0.

⚠ **CAUTION**

If a reset is made during index table indexing axis movement, a reference position return operation must always be performed before index table indexing.

NOTE

Dry run cannot be used to position the index table indexing axis.

13.7 FIGURE COPY (G72.1,G72.2)

Machining can be repeated after moving or rotating the figure using a subprogram.

Format

- Rotational copy

Xp-Yp plane (specified by G17) : G72.1 P_ L_ Xp_ Yp_ R_ ;
Zp-Xp plane (specified by G18) : G72.1 P_ L_ Zp_ Xp_ R_ ;
Yp-Zp plane (specified by G19) : G72.1 P_ L_ Yp_ Zp_ R ;_

P : Subprogram number
L : Number of times the operation is repeated
Xp : Center of rotation on the Xp axis (Xp : X-axis or an axis parallel to the X-axis)
Yp : Center of rotation on the Yp axis (Yp: Y-axis or an axis parallel to the Y-axis)
Zp : Center of rotation on the Zp axis (Zp: Z-axis or an axis parallel to the Z-axis)
R : Angular displacement (A positive value indicates a counterclockwise angular displacement. Specify an incremental value.)

Specify a plane selection command (G17, G18, or G19) to select the plane on which the rotational copy is made.

NOTE

1 In the G72.1 block, addresses other than P, L, Xp, Yp, Zp, and R are ignored.
2 P, Xp, Yp, Zp, and R must always be specified.
2 If L is not specified, the figure is copied once.
3 The coordinate of the center of rotation is handled as an absolute value even if it is specified in the incremental mode.
4 Specify an increment in the angular displacement at address R. The angular displacement (degree) for the Nth figure is calculated as follows: Rx(N-1).

- Linear copy

Xp-Yp plane (specified by G17) : G72.2 P_ L_ I_ J_ ;
Zp-Xp plane (specified by G18) : G72.2 P_ L_ K_ I_ ;
Yp-Zp plane (specified by G19) : G72.2 P_ L_ J_ K_;
P : Subprogram number
L : Number of times the operation is repeated
I : Shift along the Xp axis
J : Shift along the Yp axis
K : Shift along the Zp axis
Specify a plane selection command (G17, G18, or G19) to select the plane on which the linear copy is made.

NOTE

1 In the G72.2 block, addresses other than P, L, I, J, and K are ignored.
2 P, I, J, and K must always be specified.
2 If L is not specified, the figure is copied once.
3 For shifts (I, J, and K), specify increments. The n-th geometric shift is equal to the specified shift times (n - 1).

Explanation

- First block of the subprogram

Always specify a move command in the first block of a subprogram that performs a rotational or linear copy. If the first block contains only the program number such as O00001234; and does not have a move command, movement may stop at the start point of the figure made by the n-th (n = 1, 2, 3, ...) copying.

Example of an incorrect program)

```
    O00001234 ;
    G00 G90 X100.0 Y200.0 ;
    - - - - ;
    - - - - ;
    M99 ;
```

Example of a correct program)

```
    O00001000 G00 G90 X100.0 Y200.0 ;
    - - - - ;
    - - - - ;
    M99 ;
```

- Subprogram calling

In a subprogram for rotational or linear copying, M98 for calling another subprogram or G65 for calling a macro can be specified.

- Nesting level of a subprogram

If a subprogram is called by G72.1 or G72.2, the nesting level is increased by one in the same manner as when M98 is specified.

- Block end position

The coordinates of a figure moved rotationally or linearly (block end position) can be read from #5001 and subsequent system variables of the custom macro of rotational or linear copy.

- Disagreement between end point and start point

If the end point of the figure made by the n-th copy does not agree with the start point of the figure to be made by the next (n + 1) copy, the figure is moved from the end point to the start point, then copying is started. (Generally, this disagreement occurs if an incorrect angular displacement or shift is specified.)

Main program

```
    O1000 ;
    N10 G92 X-20.0 Y0 ;
    N20 G00 G90 X0 Y0 ;
    N30 G01 G17 G41 X20.0 Y0 D01 F100 ;  (P0)
    N40 Y20.0 ;                          (P1)
    N50 X30.0 ;                          (P2)
    N60 G72.2 P2000 L3 I90.0 J0 ;
     :
```

Although a shift of 70 mm was required, I90.0 was specified instead of I70.0. Since an incorrect shift was specified, the end point of the figure made by the n-th copy disagrees with the start point of the figure to be made by the next (n + 1) copy.

Subprogram

```
    O2000 G90 G01 X40.0 ;                (P3)
    N100 Y40.0 ;                         (P4)
    N200 G01 X80.0 ;                     (P5)
    N300 G01 Y20.0 ;                     (P6)
    N400 X100.0 ;                        (P7)
    N500 M99 ;
```

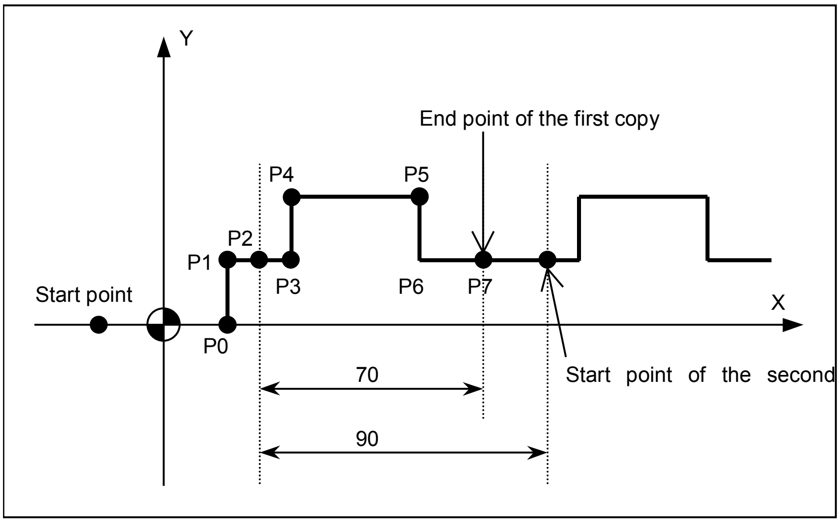


Fig.13.7 (a) Disagreement between end point and start point

Limitation

- Specifying two or more commands to copy a figure

G72.1 cannot be specified more than once in a subprogram for making a rotational copy (If this is attempted, alarm PS0900 will occur). G72.2 cannot be specified more than once in a subprogram for making a linear copy (If this is attempted, alarm PS0901 will occur).
In a subprogram that specifies rotational copy, however, linear copy can be specified. Similarly, in a subprogram that specifies linear copy, rotational copy can be specified.

- Commands that must not be specified

Within a program that performs a rotational or linear copy, the following must not be specified:

- Command for changing the selected plane (G17 to G19)
- Command for specifying polar coordinates (G16)
- Reference position return command(G28)
- Axis switching
- Coordinate system rotation (G68)
- scaling (G51)
- programmable mirror image (G51.1)

The command for rotational or linear copying can be specified after a command for coordinate system rotation, scaling, or programmable mirror image is executed.

- Single block

Single-block stops are not performed in a block with G721.1 or G72.2.

Example

- Rotational copy

Main program

```
     O1000 ;
     N10 G90 G00 X80. Y100. ;
     N20 Y50. ;                                    (P0)
     N30 G01 G17 G42 X43.301 Y25. D01 F100 ;       (P1)
     N40 G72.1 P1100 L3 X0 Y0 R120. ;
     N50 G90 G40 G01 X80. Y50. ;                   (P0)
     N60 G00 X80. Y100. ;
     N70 M30 ;
```

Sub program

```
     O1100 G91 G03 X-18.301 Y18.301 R50. ;         (P2)
     N100 G01 X-5. Y50. ;                          (P3)
     N200 G03 X-40. I-20. ;                        (P4)
     N300 G01 X-5. Y-50. ;                         (P5)
     N400 G03 X-18.301 Y-18.301 R50. ;             (P6)
     N500 M99 ;
```

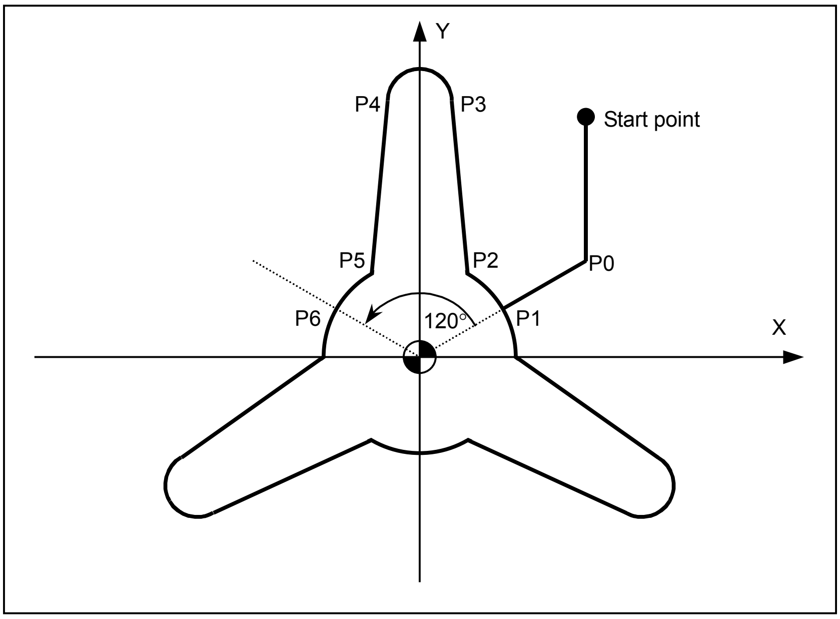


Fig.13.7 (b) Rotational copy

- Rotational copy (Spot boring)

Main program

```
    O2000 ;
    N10 G90 G00 G17 X250. Y100. Z100. ;                (P0)
    N20 G72.1 P2100 L6 X100. Y50. R60. ;
    N30 G80 G00 X250. Y100. ;                          (P0)
    N40 M30 ;
```

Sub program

```
    O2100 N100 G90 G81 X100. Y150. R60. Z10. F200. ; (P1)
    N200 M99 ;
```

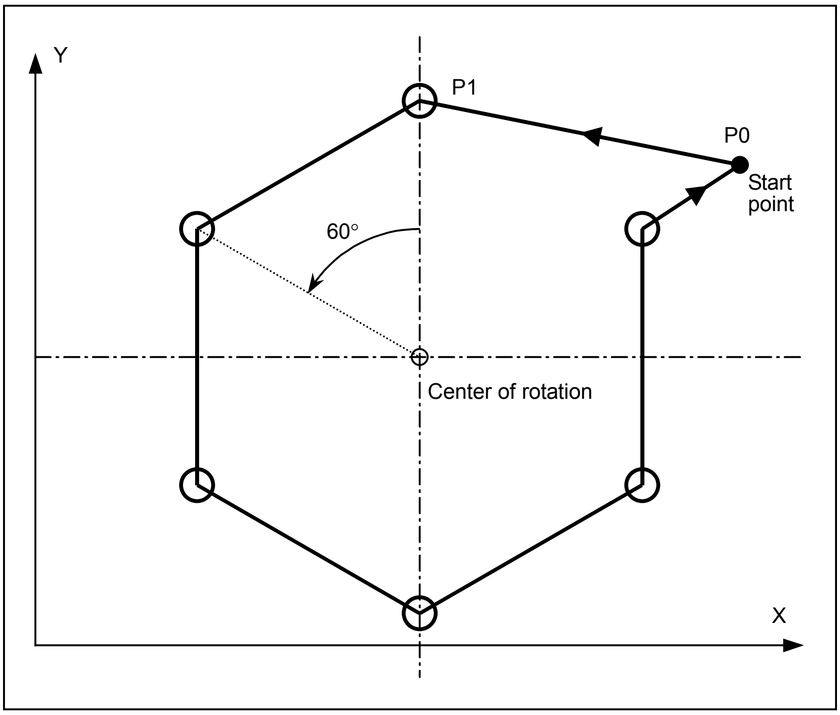


Fig.13.7 (c) Rotational copy

- Linear copy

Main program

```
    O3000 ;
    N10 G90 G00 X-30. Y0 ;
    N20 X0 ;
    N30 G01 G17 G41 X30. D01 F100 ;                (P0)
    N40 Y20. ;                                     (P1)
    N50 X40. ;                                     (P2)
    N60 G72.2 P3100 L3 I90.0 J0 ;
    N70 G90 X310. Y0 ;                             (P8)
    N80 X0 ;
    N90 G40 G00 X-30.0 ;
    N100 M30 ;
```

Sub program

```
    O3100 G91 G01 X20. ;                           (P3)
    N100 Y30. ;                                    (P4)
    N200 G02 X40. I20. ;                           (P5)
    N300 G01 Y-30. ;                               (P6)
    N400 X30. ;                                    (P7)
    N500 M99 ;
```

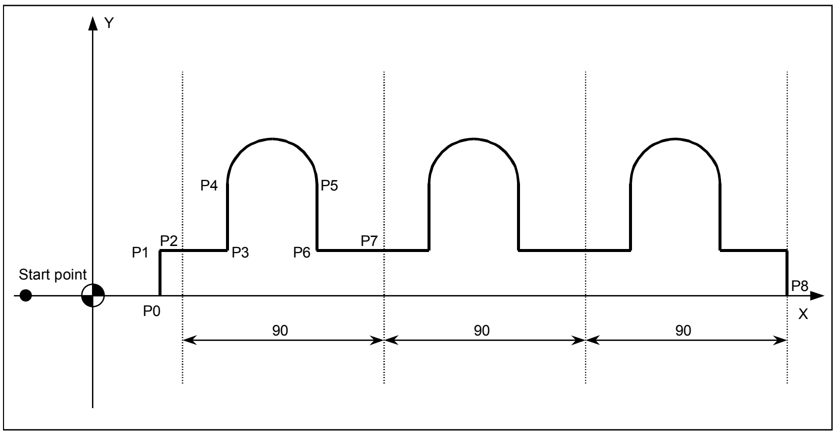


Fig.13.7 (d) Linear copy

- Combination of rotational copying and linear copying (Bolt hole circle)

Main program

```
O4000 ;
N10 G90 G00 G17 X240. Y230. Z100. ;          (P0)
N20 G72.1 P4100 X120. Y120. L8 R45. ;
N30 G80 G00 X240. Y230. ;                    (P0)
N40 M30 ;
```

Sub program (rotation copy)

```
O4100 N100 G72.2 P4200 I0 J20. L3 ;
N200 M99 ;
```

Sub program (linear copy)

```
O4200 N110 G90 G81 X120. Y180. R60. Z10. F20 ; (P1)
N210 M99 ;
```

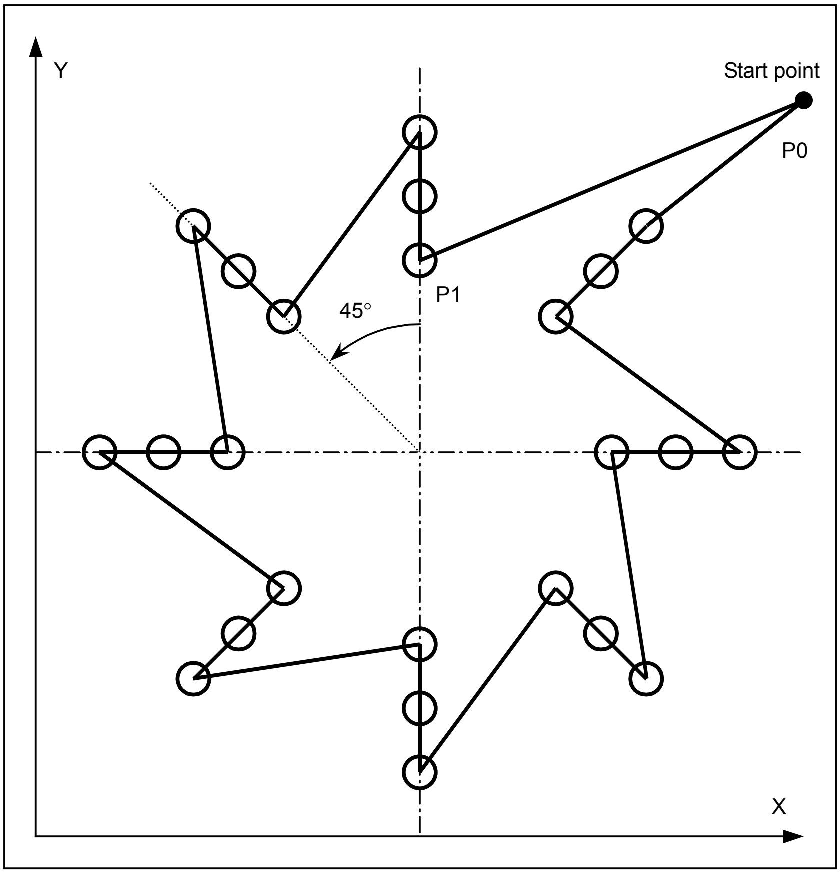


Fig.13.7 (e) Combination of rotational copying and linear copying

13.8 NORMAL DIRECTION CONTROL (G40.1, G41.1, G42.1)

When a tool with a rotation axis (C-axis) is moved in the XY plane during cutting, the normal direction control function can control the tool so that the C-axis is always perpendicular to the tool path (Fig. 14.11 (a)).

Between blocks in normal direction control mode, movement on the C-axis is inserted so that, as the move direction changes, the tool is positioned perpendicular to the move direction at the start point of each block.

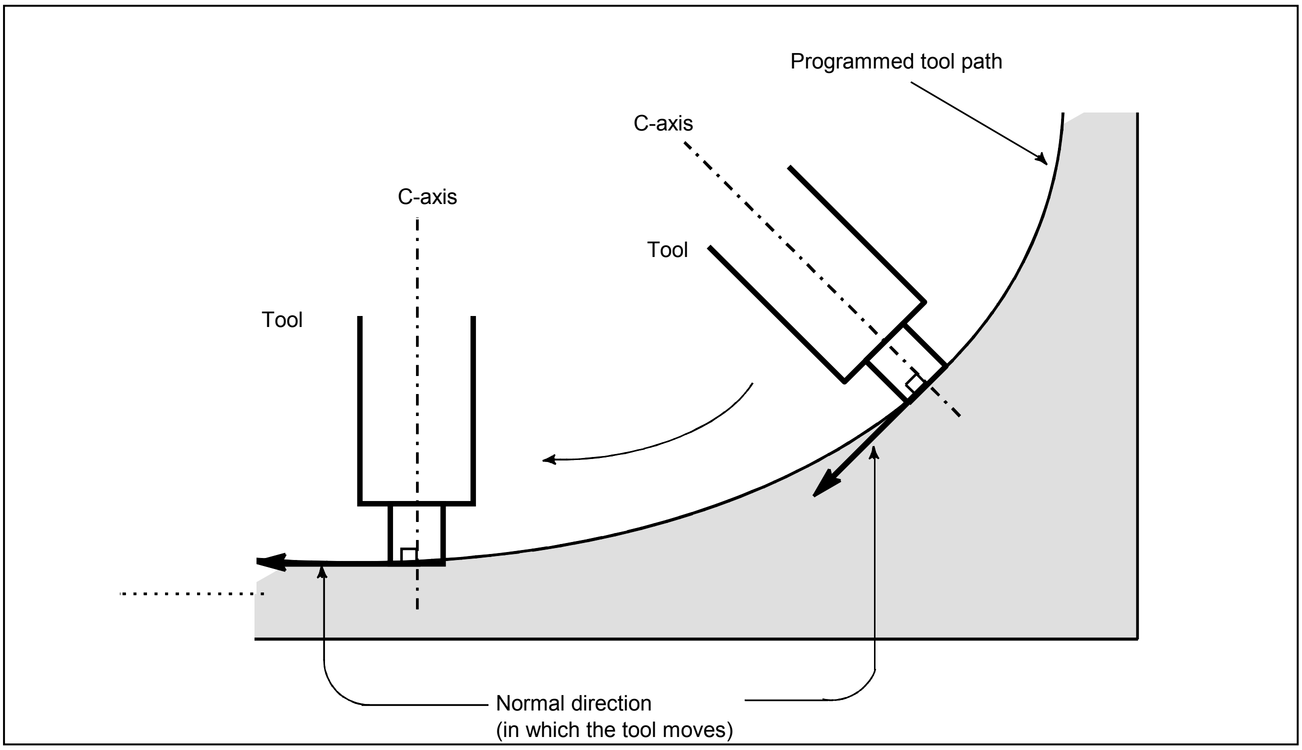


Fig.13.8 (a) Sample Movement of the tool

Format

G code	Function	Explanation
G41.1	Normal direction control left	If the workpiece is to the right of the tool path looking toward the direction in which the tool advances, the normal direction control left (G41.1) function is specified. After G41.1 or G42.1 is specified, the normal direction control function is enabled (normal direction control mode). When G40.1 is specified, the normal direction control mode is canceled.
G42.1	Normal direction control right	
G40.1	Normal direction control cancel	

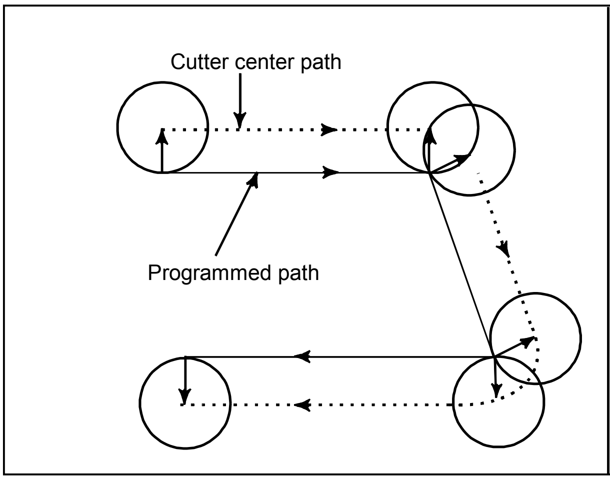


Fig.13.8 (b) Normal direction control left (G41.1)

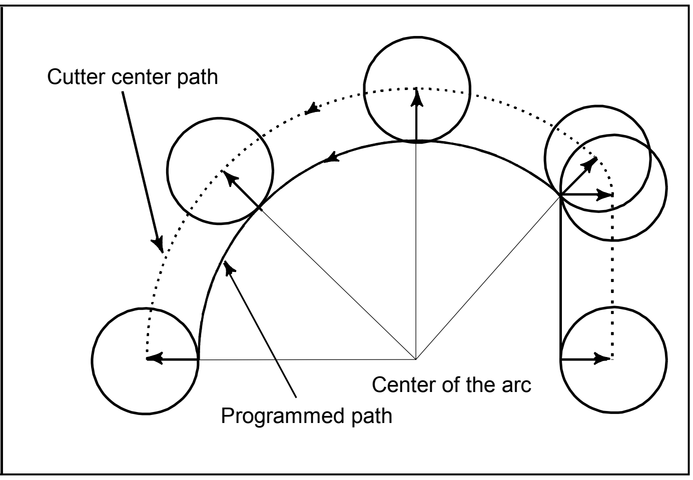


Fig.13.8 (c) Normal direction control right (G42.1)

Explanation

- Angle of the C axis

When viewed from the center of rotation around the C-axis, the angular displacement about the C-axis is determined as shown in Fig.13.8 (d). The positive side of the X-axis is assumed to be 0 , the positive side of the Y-axis is 90deg, the negative side of the X-axis is 180deg, and the negative side of the Y-axis is 270deg.

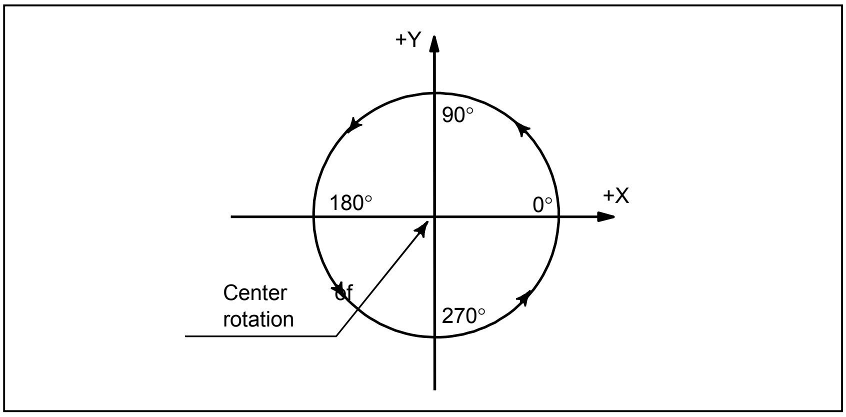


Fig.13.8 (d) Angle of the C axis

- Normal direction control of the C axis

When the cancel mode is switched to the normal direction control mode, the C-axis becomes perpendicular to the tool path at the beginning of the block containing G41.1 or G42.1.

In the interface between blocks in the normal direction control mode, a command to move the tool is automatically inserted so that the C-axis becomes perpendicular to the tool path at the beginning of each block. The tool is first oriented so that the C-axis becomes perpendicular to the tool path specified by the move command, then it is moved along the X- and Y axes.

In the cutter compensation mode, the tool is oriented so that the C-axis becomes perpendicular to the tool path created after compensation. In single-block operation, the tool is not stopped between a command for rotation of the tool and a command for movement along the X- and Y-axes. A single-block stop always occurs after the tool is moved along the X- and Y-axes.

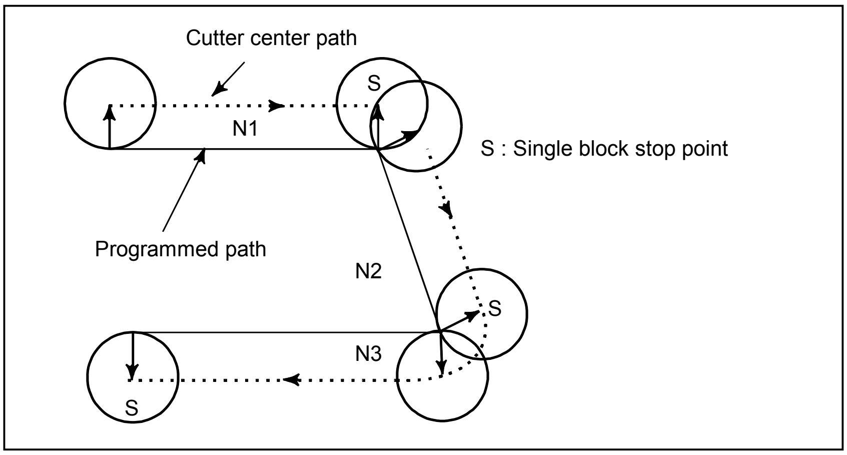


Fig.13.8 (e) Point at which a Single-Block Stop Occurs in the Normal Direction Control Mode

Before circular interpolation is started, the C-axis is rotated so that the C-axis becomes normal to the arc at the start point. During circular interpolation, the tool is controlled so that the C-axis is always perpendicular to the tool path determined by circular interpolation.

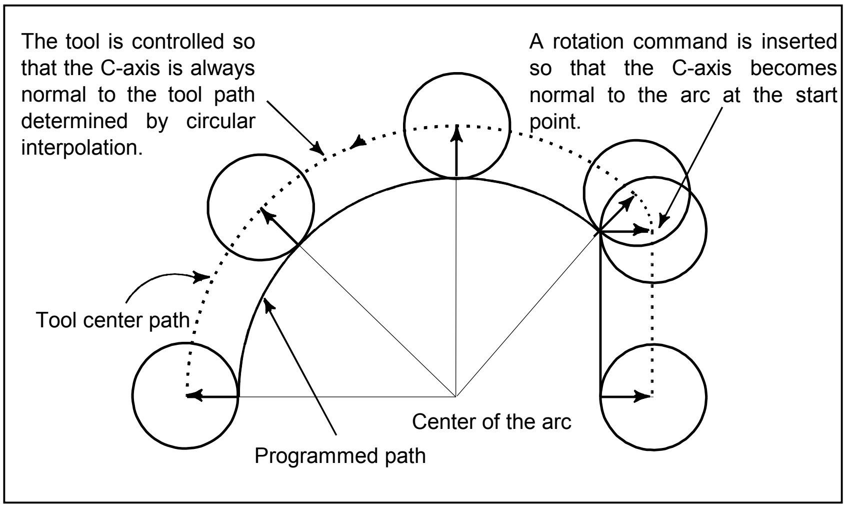

The tool is controlled so that the C-axis is always normal to the tool path determined by circular interpolation.
A rotation command is inserted so that the C-axis becomes normal to the arc at the start point.
Tool center path
Center of the arc
Programmed path

Fig.13.8 (f) Normal direction control of the circular interpolation

NOTE

During normal direction control, the C axis always rotates through an angle less than 180 deg. I.e., it rotates in whichever direction provides the shorter route.

- C axis feedrate

Movement of the tool inserted at the beginning of each block is executed at the feedrate set in parameter 1472. If dry run mode is on at that time, the dry run feedrate is applied. If the tool is to be moved along the X-and Y-axes in rapid traverse (G00) mode, the rapid traverse feedrate is applied.

The federate of the C axis during circular interpolation is defined by the following formula.

$$F\times \frac{\text{Amount of movement of the C axis (deg)}}{\text{Length of arc (mm or inch)}} \text{[deg/min]}$$

F : Federate (mm/min or inch/min) specified by the corresponding block of the arc

Amount of movement of the C axis :
The difference in angles at the beginning and the end of the block.

NOTE

1 If the federate of the C axis exceeds the maximum cutting speed of the C axis specified to parameter No. 1422, the federate of each of the other axes is clamped to keep the federate of the C axis below the maximum cutting speed of the C axis.
2 Do not specify any command to the C axis during normal direction control. Any command specified at this time is ignored.
3 Before machining is started, a match between the workpiece coordinate on the C-axis and the actual position on the C-axis of the machine must be ensured using the origin key or coordinate system setting (G92).
4 In normal direction control mode, helical interpolation involving two linear axes cannot be specified; helical interpolation involving only one linear axis can be specified.
5 Normal direction control cannot be performed by the G53 move command.
6 The C-axis must be a rotation axis.
7 If a parameter related to normal direction control contains an error, a PS alarm (No. 0470) is issued.
8 The above expression, used for calculating a C-axis speed when a circular command is specified, assumes that bit 0 (HTG) of parameter No. 1401 is set to 0 (for specifying a circular tangent speed for helical interpolation speed specification).

13.9 THREE-DIMENSIONAL COORDINATE CONVERSION (G68,G69)

Coordinate conversion about an axis can be carried out if the center of rotation, direction of the axis of rotation, and angular displacement are specified. This function is very useful in three-dimensional machining by a die-sinking machine or similar machine. For example, if a program specifying machining on the XY plane is converted by the three-dimensional coordinate conversion function, the identical machining can be executed on a desired plane in three-dimensional space.

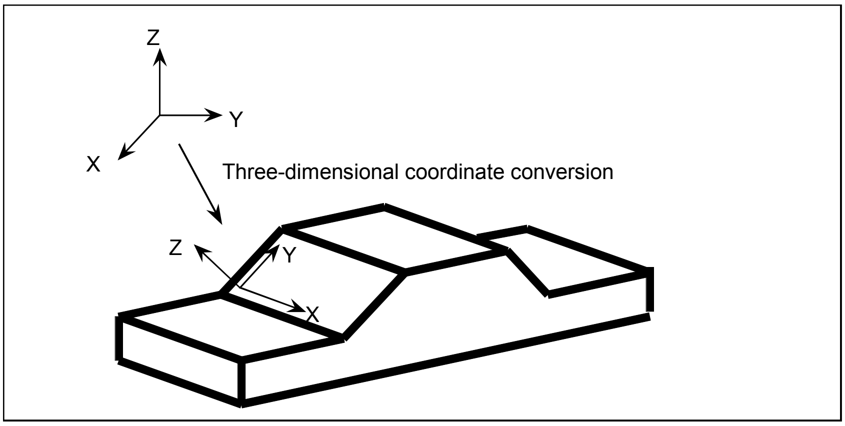


Fig.13.9 (a) Three-dimensional coordinate conversion

Format

Format	
G68 Xp $\underline{x_1}$ Yp $\underline{y_1}$ Zp $\underline{z_1}$ I $\underline{i_1}$ J $\underline{j_1}$ K $\underline{k_1}$ R$\underline{\alpha}$;	Starting first three-dimensional coordinate conversion
G68 Xp $\underline{x_2}$ Yp $\underline{y_2}$ Zp $\underline{z_2}$ I $\underline{i_2}$ J $\underline{j_2}$ K $\underline{k_2}$ R$\underline{\beta}$;	Starting second three-dimensional coordinate conversion
: **:**	} Three-dimensional coordinate conversion mode
G69 ;	Canceling three-dimensional coordinate conversion

Symbol	
Xp, Yp, Zp	: Center of rotation (absolute coordinates) on the X, Y, and Z axis or parallel axes
I, J, K	: Direction of the axis of rotation
R	: Angular displacement

NOTE

1 Use absolute programming for Xp, Yp, and Zp specified in G68.

2 When only one rotation is sufficient, the second G68 is not required.

3 If the second G68 does not specify Xp, Yp, or Zp, the center of the second rotation is the same as that of the first rotation.

4 If any of the following is specified, an alarm (PS0626) is issued:

(1) In a three-dimensional coordinate conversion command block, none of I, J, and K is specified. (The coordinate system rotation option is not selected.)

(2) In a three-dimensional coordinate conversion command block, 0 is specified for all of I, J, and K.

(3) In a three-dimensional coordinate conversion command block, angular displacement R is not specified.

Explanation

- Program coordinate system and workpiece coordinate system

Sample program:

N1 G68 Xp $\underline{x_1}$ Yp $\underline{y_1}$ Zp $\underline{z_1}$ I $\underline{i_1}$ J $\underline{j_1}$ K $\underline{k_1}$ R$\underline{\alpha}$;
N2 G68 Xp $\underline{x_2}$ Yp $\underline{y_2}$ Zp $\underline{z_2}$ I $\underline{i_2}$ J $\underline{j_2}$ K $\underline{k_2}$ R$\underline{\beta}$;
N3 G90 G01 Xp $\underline{x}$ Yp $\underline{y}$ Zp $\underline{z}$;

Coordinate system definition

Coordinate system X, Y, Z :

Original workpiece coordinate system

Coordinate system X', Y', Z' :

Coordinate system after the first coordinate conversion

N1 specifies the rotation center, rotation center axis direction, and angular displacement of the first rotation. As a result of the first rotation, coordinate system X', Y', and Z' is produced which has its center shifted by (x1,y1,z1) with respect to the original workpiece coordinate system, and is rotated about (i1,j1,k1) by α.

Coordinate system X'', Y'', Z'' :

Program coordinate system

N2 specifies the rotation center, rotation center axis direction, and angular displacement of the second rotation. As a result of the second rotation, coordinate system X'', Y'', and Z'' is produced which has its center shifted by (x2,y2,z2) with respect to coordinate system X', Y', Z', and is rotated about (i2,j2,k2) by β.

Note that Xp, Yp, Zp, I, J, K, and R in N2 are the coordinates and angle in the coordinate system produced by conversion by N1.

Programmed values Xp, Yp, and Zp in N3 are regarded as being the coordinates in program coordinate system X", Y", and Z".

X,Y,Z	: Workpiece coordinate system
X',Y',Z'	: Coordinate system produced after the first conversion
X'',Y'',Z''	: Coordinate system produced after the second conversion
α	: Angular displacement in the first conversion
β	: Angular displacement in the second conversion
O (x_0, y_0, z_0)	: Center of rotation
P (x, y, z)	: Coordinates in coordinate system X", Y", Z" (program coordinate system)

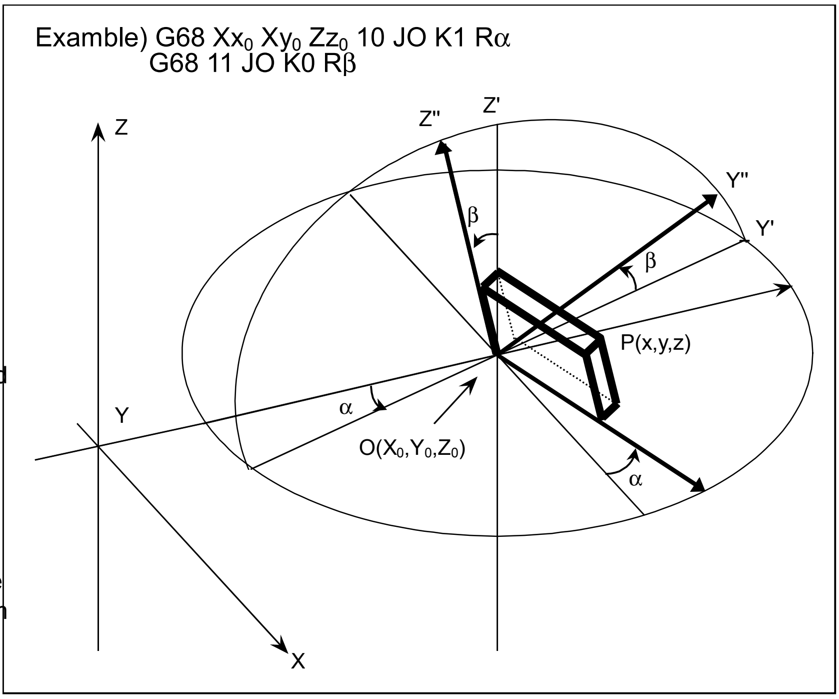


Fig.13.9 (b) Program coordinate system and workpiece coordinate system

- Angular displacement R

A positive angular displacement R indicates a clockwise rotation along the axis of rotation.

- Least command increment of angular displacement

The least command increment of angular displacement is 0.001 degree or 0.00001 degree.
When bit 2 (RTR) of parameter No. 6400 is set to 0, 0.00001 degree is used. When the bit is set to 1, 0.001 degree is used.

> **NOTE**
> When a decimal point is used in the angular displacement command (R_), the position of the decimal point represents degrees.

- Reset

If a reset occurs during three-dimensional coordinate conversion mode, the mode is canceled and the continuous-state G code is changed to G69.
When bit 4 (D3R) of parameter No. 6400 is set to 1, however, three-dimensional coordinate conversion mode (G68) can be cancelled only by the G69 command. That is, when the bit is so set, three-dimensional coordinate conversion mode (G68) is not cancelled even by a reset, or by resetting the CNC by the input of the ERS, ESP, or RRW signal from the PMC.

NOTE

Even if bit 4 (D3R) of parameter No. 6400 is set to 1, G69 mode is assumed when program execution is restarted.

- Custom macro system variable

If the workpiece coordinates of the tool currently being used are read using custom macro system variables #5041 to #5050 (ABSOT), the coordinates in the program coordinate system before coordinate conversion can be read even in three-dimensional coordinate conversion mode (G68). When bit 5 (D3M) of parameter No. 6400 is set to 1, the coordinates in the workpiece coordinate system after coordinate conversion can be read.

- Rapid traverse rate for hole machining in a canned cycle

In three-dimensional coordinate conversion mode, rapid traverse for hole machining in a canned cycle is performed using the cutting feedrate specified in parameter No. 2052. When this parameter is set to 0, the maximum cutting feedrate is used.

- Compensation functions

If tool length compensation, cutter compensation, or tool offset is specified with three-dimensional coordinate conversion, compensation is performed first, followed by three-dimensional coordinate conversion.

- Relationship between three-dimensional and two-dimensional coordinate conversion (G68, G69)

Three-dimensional and two-dimensional coordinate conversion use identical G codes (G68 and G69). A G code specified with I, J, and K is processed as the command for three-dimensional coordinate conversion. A G code not specified with I, J, and K is processed as the command for two-dimensional coordinate conversion.

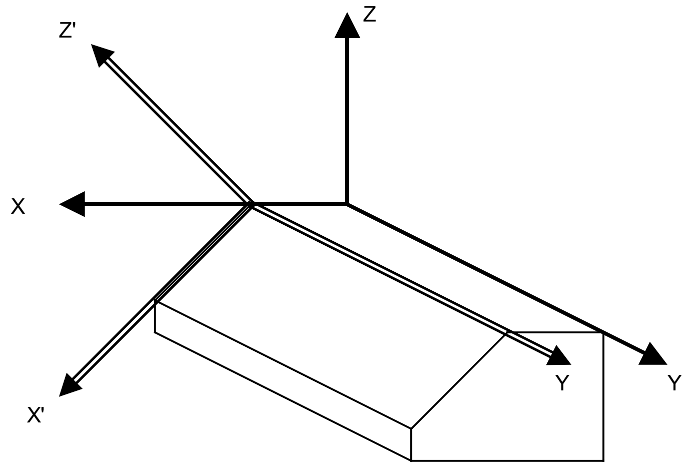
- Manual operation

In three-dimensional coordinate conversion mode (G68), the following two types of manual operation (jog feed, manual incremental feed, and manual handle feed) are supported:

(1) Operation in a workpiece coordinate system (with three-dimensional coordinate conversion not applied)

(2) Operation in a program coordinate system (with three-dimensional coordinate conversion applied)

For information about which type is used, refer to the manual provided by the machine tool builder.



X, Y ,Z : Coordinate system before conversion (workpiece coordinate system)

X', Y' ,Z' : Coordinate system after conversion (program coordinate system)

When manual movement is made along the Z-axis:
(1) A movement is made in the Z direction.
(2) A movement is made in the Z' direction.

Fig.13.9 (c) Manual Operation during Three-dimensional Coordinate Conversion

In manual feed, the same time constant, FL feedrate, and acceleration/deceleration method as that for the cutting mode are used for the three axes selected in three-dimensional coordinate conversion. In manual rapid traverse, the same time constant, feedrate, and acceleration/deceleration method as that for programmed rapid traverse (positioning by G00) is used. When manual movement is made during three-dimensional coordinate conversion, the tangent feedrate in the coordinate system after conversion (program coordinate system) is the lowest feedrate among those along the selected axes.

- Absolute position display

The absolute coordinates based on the program or workpiece coordinate system can be displayed in the three-dimensional coordinate conversion mode. Specify a desired coordinate system in the DAK bit (bit 2 of parameter 2204).

NOTE

To enable absolute position display in three-dimensional coordinate conversion, set bits 2 and 3 (DTL and DCR) of parameter No. 2202 to 0.

- Indication of remaining amounts of travel

By setting bit 5 (D3D) of parameter No. 2208, the user can choose whether a remaining amount of travel in three-dimensional coordinate conversion mode is indicated in the program coordinate system or workpiece coordinate system.

- G codes that can be specified

The following G codes can be specified in the three-dimensional coordinate conversion mode:

G00	Positioning
G01	Linear interpolation
G02	Circular interpolation (clockwise)
G03	Circular interpolation (counterclockwise)
G04	Dwell
G06.2	NURBS interpolation
G10	Data setting (However, commands such as G10L2 for rewriting coordinate system data cannot be specified.)
G17	Plane selection (XY)
G18	Plane selection (ZX)
G19	Plane selection (YZ)
G28	Reference position return
G29	Return from the reference position
G30	Return to the second, third, or fourth reference position
G40	Canceling cutter compensation
G41	Cutter compensation to the left
G42	Cutter compensation to the right
G43	Increasing tool length compensation
G43.1	Tool length compensation in tool axis direction
G44	Decreasing tool length compensation
G45	Increasing the tool offset
G46	Decreasing the tool offset
G47	Doubling the tool offset
G48	Halving the tool offset
G49	Canceling tool length compensation
G50.1	Canceling programmable mirror image
G51.1	Programmable mirror image
G53	Selecting the machine coordinate system
G65	Custom macro calling
G66	Continuous-state custom macro calling
G67	Canceling continuous-state custom macro calling
G73	Canned cycle (peck drilling cycle)
G74	Canned cycle (reverse tapping cycle)
G76	Canned cycle (fine boring cycle)
G80	Canceling a canned cycle
G81 to G89	Canned cycle
G90	Absolute mode
G91	Incremental mode
G94	Feed per minute
G95	Feed per rotation
G98	Canned cycle (return to the initial level)
G99	Canned cycle (return to the level of point R)

Limitation

- Increment system

NOTE
The same increment system must be used for all of the three basic axes used for three-dimensional coordinate conversion.

- Diameter and radius specification

NOTE
The same diameter and radius specification must be used for all of the three basic axes used for three-dimensional coordinate conversion.

- Rapid traverse command

NOTE
Set bit 4 (LRP) of parameter No. 1400 to 1 to specify linear rapid traverse.

- Positioning in the machine coordinate system

Three-dimensional coordinate conversion does not affect positioning in the machine coordinate system (e.g. specified with G28, G30, or G53).

- Block with G68 or G69

In a block with G68 or G69, other G codes must not be specified. G68 must be specified with I, J, and K.

- Mirror image

External mirror image (performed by DI or setting) is applied to movement after three-dimensional coordinate conversion on each controlled axis.

- Three-dimensional coordinate conversion and other continuous-state commands

G41, G42, and G40 (cutter compensation), G43 and G49 (tool length offset), G51.1 and G50.1 (programmable mirror image), and canned cycle commands must be nested between G68 and G69. Specify G68 with these modes turned off, turn these modes on and back off, then specify G69.

To specify G41, G42, and G40 (cutter compensation), G51.1 and G50.1 (programmable mirror image), and canned cycle commands during three-dimensional coordinate conversion, perform programming using either of the following methods:

1) Before starting three-dimensional coordinate conversion, specify absolute commands for axes subjected to three-dimensional coordinate conversion. Then, start three-dimensional coordinate conversion.

```
Example) G90 X100.0 Y200.0 Z300.0 ;
         G68 X50.0 Y100.0 Z150.0 I0.0 J0.0 K1.0 R60.0 ;
```

2) During three-dimensional coordinate conversion, specify absolute commands for axes subjected to three-dimensional coordinate conversion after these modes are turned off. Then, specify G69.

(Example)

```
G68 X100. Y100. Z100. I0. J0. K1. R45. ;
 :
G41 D01 ;
 :
G40 ;
G90 G00 X100. Y100. Z0 ;  Always use absolute programming.
G69 ;
```

- Parallel axis control

Even if a parking signal is enabled for an axis in parallel axis control, three-dimensional coordinate conversion is applied to another axis for which a move command is specified, if any. So, a movement may be made on an axis for which the parking signal is enabled.

- Single-direction positioning

In three-dimensional coordinate conversion mode, the single-direction positioning command (G60) cannot be specified.

Equation for three-dimensional coordinate conversion

The following equation shows the general relationship between (x, y, z) in the program coordinate system and (X, Y, Z) in the original coordinate system (workpiece coordinate system).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} & & \\ & M_1 & \\ & & \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}$$

When conversion is carried out twice, the relationship is expressed as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} & & \\ & M_1 & \\ & & \end{bmatrix} \begin{bmatrix} & & \\ & M_2 & \\ & & \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} & & \\ & M_1 & \\ & & \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}$$

X,Y,Z : Coordinates in the original coordinate system (workpiece coordinate system)
x,y,z : Programmed value (coordinates in the program coordinate system)
x_1,y_1,z_1 : Center of rotation of the first conversion
x_2,y_2,z_2 : Center of rotation of the second conversion (coordinates in the coordinate system formed after the first conversion)
M_1 : First conversion matrix
M_2 : Second conversion matrix

M$_1$ and M$_2$ are conversion matrices determined by an angular displacement and rotation axis. Generally, the matrices are expressed as shown below:

$$\begin{bmatrix} n_1{}^2+(1-n_1{}^2)\cos\theta & n_1n_2(1-\cos\theta)-n_3\sin\theta & n_1n_3(1-\cos\theta)+n_2\sin\theta \\ n_1n_2(1-\cos\theta)+n_3\sin\theta & n_2{}^2+(1-n_2{}^2)\cos\theta & n_2n_3(1-\cos\theta)-n_1\sin\theta \\ n_1n_3(1-\cos\theta)-n_2\sin\theta & n_2n_3(1-\cos\theta)+n_1\sin\theta & n_3{}^2+(1-n_3{}^2)\cos\theta \end{bmatrix}$$

n_1 : Cosine of the angle made by the rotation axis and X-axis

$$\frac{i}{\sqrt{i \ + j^2 + k^2}}$$

n_2 : Cosine of the angle made by the rotation axis and Y-axis

$$\frac{j}{\sqrt{i \ + j^2 + k^2}}$$

n_3 : Cosine of the angle made by the rotation axis and Z-axis

$$\frac{k}{\sqrt{i \ + j^2 + k^2}}$$

θ : Angular displacement

Conversion matrices for rotation on two-dimensional planes are shown below:

(1) Coordinate conversion on the XY plane

$$M=\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

(2) Coordinate conversion on the YZ plane

$$M=\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

(3) Coordinate conversion on the ZX plane

$$M=\begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\vartheta \end{bmatrix}$$

Example

```
N1 G90 X0 Y0 Z0 ;                              (1)
N2 G68 X10. Y0 Z0 I0 J1 K0 R30. ;              (2)
N3 G68 X0 Y-10. Z0 I0 J0 K1 R-90. ;            (3)
N4 G90 X0 Y0 Z0 ;                              (4)
N5 X10. Y10. Z0 ;                              (5)
```

(1) Carries out positioning to zero point H.

(2) Forms new coordinate system X'Y'Z'.

(3) Forms other coordinate system X"Y"Z". The origin agrees with (0, -10, 0) in coordinate system X'Y'Z.

(4) Carries out positioning to zero point H on coordinate system X"Y"Z".

(5) Carries out positioning to (10, 10, 0) on coordinate system X"Y"Z".

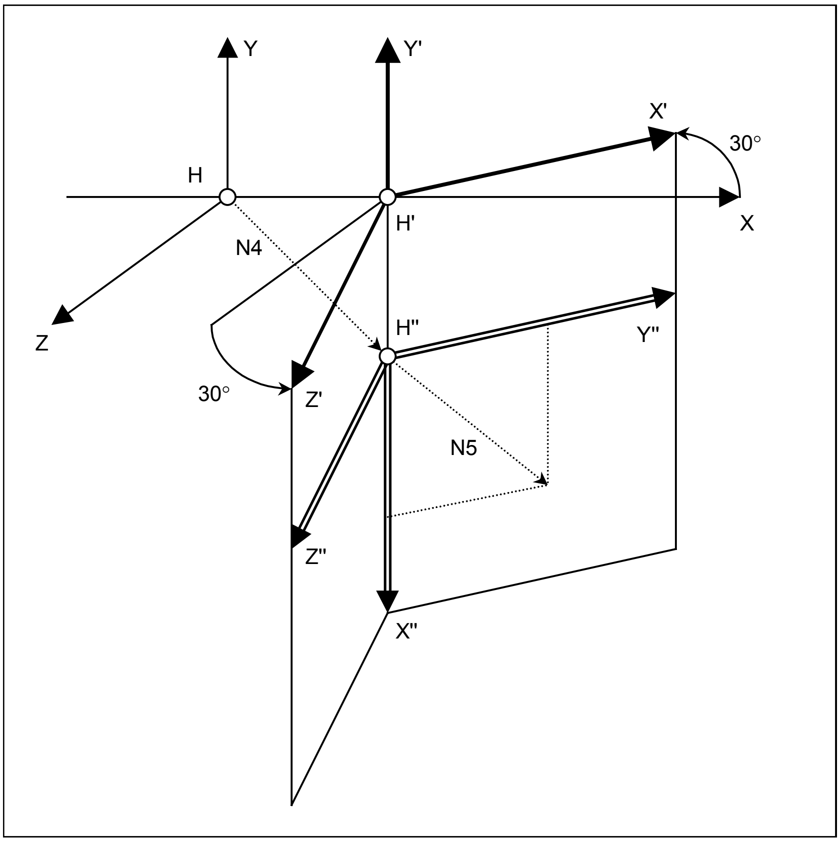
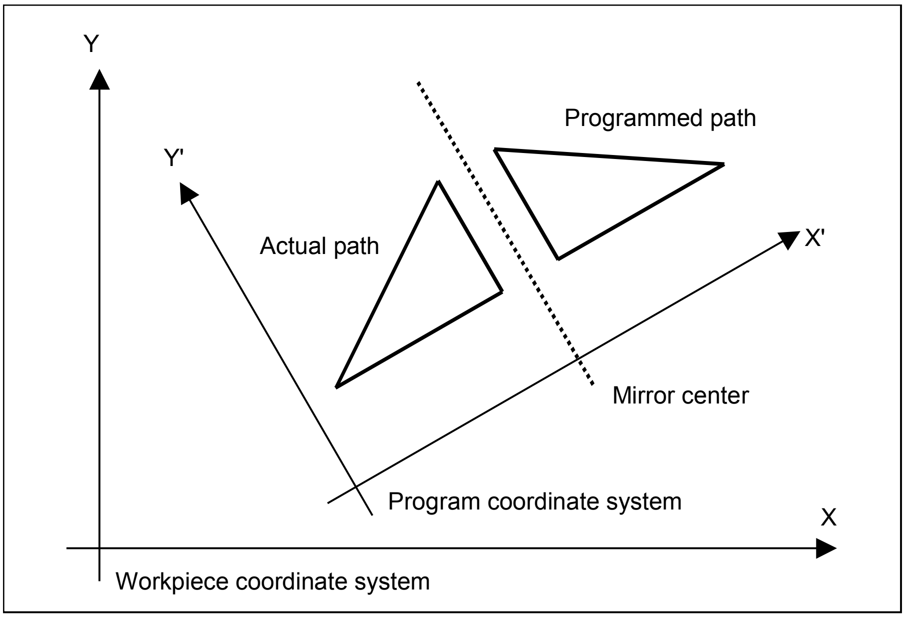


Fig.13.9 (d) Example of three-dimensional conversion

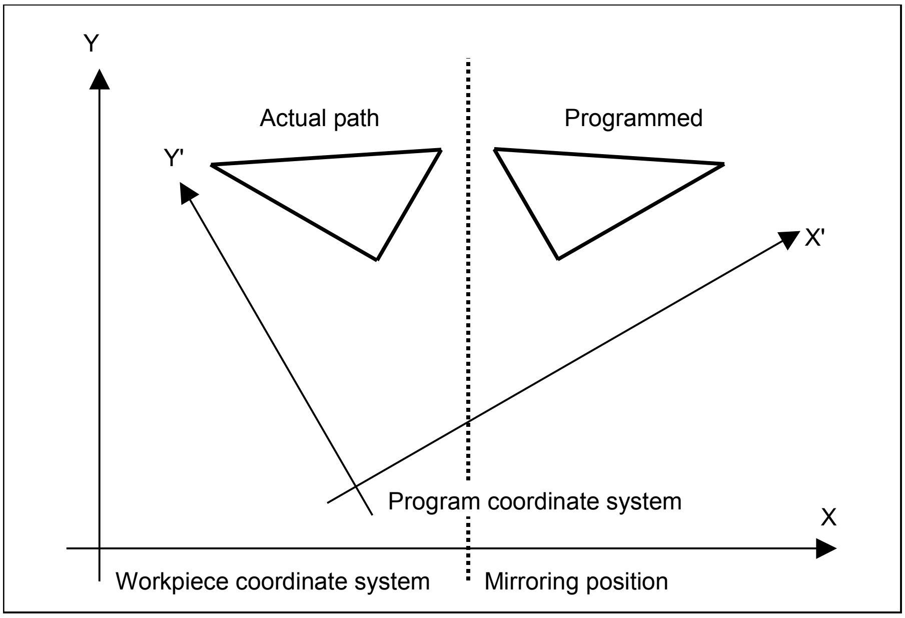
- Programmable mirror image

When three-dimensional coordinate conversion and programmable mirror image are used at the same time, programmable mirror image is applied to coordinates in the program coordinate system, then three-dimensional coordinate conversion is performed.



- External mirror image

When three-dimensional coordinate conversion and external mirror image (mirror image performed by signals or setting) are used at the same time, three-dimensional coordinate conversion is performed first, then external mirror image is applied.



13.9.1 Three-dimensional Coordinate Conversion and Parallel Axis Control

Overview

If three-dimensional coordinate conversion is to be performed on a machine operating with parallel axis control, this function allows combinations of parallel axes subject to three-dimensional coordinate conversion to be specified with parameter No. 1016, so that three-dimensional coordinate conversion can be used on machines with a variety of parallel axis configurations.
Parallel axis combinations may be specified with parameter No. 1016. Up to eight combinations may be specified.

- Parking signal

If, in three-dimensional coordinate conversion during an automatic operation command, the combinations specified with parameter No. 1016 include a combination in which the parking signal is set to 1 for at least one axis, three-dimensional coordinate conversion will be stopped for that combination. Thus, no move command will not be executed on that combination.
For example, if three-dimensional coordinate conversion is performed on a 7-axis system with an axis configuration of X, Y1, Y2, Z1, Z2, Z3, and Z4, and four combinations (X,Y1,Z1), (X,Y1,Z2), (X,Y2,Z3), and (X,Y2,Z4) (first to fourth combinations from left to right) are specified with parameter No. 1016, three-dimensional coordinate conversion will be stopped for the first combination if the parking signal for Z1 is 1. Similarly, conversion will be stopped for the fourth combination if the parking signal for Z4 is 1.

- Three-dimensional coordinate conversion manual feed function

In three-dimensional coordinate conversion mode, the three-dimensional coordinate conversion manual feed function can be used. In the above example, if the jog feed switch for Y2 is turned on, for example, the tool will operate on the X-, Y2-, Z3-, and Z4-axes. That is, the tool will operate on all axes in the combinations containing that axis.
If, in three-dimensional coordinate conversion mode, manual feed is performed on a combination containing an axis for which the parking signal is 1, thee-dimensional coordinate conversion is not performed on that combination, nor is axial movement. Similarly, if manual feed is performed on an axis for which the parking signal is set to 1, no axial movement is performed.

- Three-dimensional coordinate conversion manual interrupt

The three-dimensional coordinate conversion manual interrupt function does not support this function.

13.10 TILTED WORKING PLANE COMMAND

Overview

Programming for creating holes and pockets in a surface tilted from the datum plane of a workpiece would be easy if commands can be issued in a coordinate system fixed to this surface (called a feature coordinate system). This function enables commands to be issued in the feature coordinate system.

The feature coordinate system is defined in the workpiece coordinate system.

See Fig.13.10(a) for explanations about the relationships between the feature coordinate system and workpiece coordinate system.

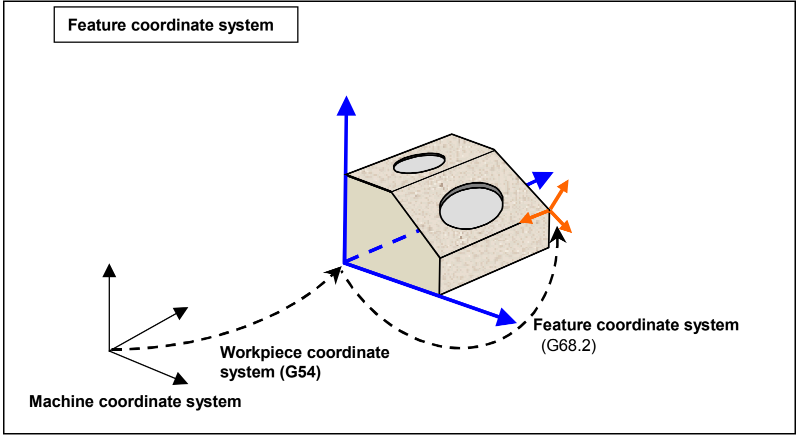

Feature coordinate system
Feature coordinate system
(G68.2)
Workpiece coordinate system (G54)
Machine coordinate system

Fig. 13.10 (a)

G68.2 selects the feature coordinate system as a coordinate system for programming. The commands in all the subsequent blocks are assumed to be issued in the feature coordinate system until G69 appears.

If G68.2 specifies the relationships of the feature coordinate system to the workpiece coordinate system, G53.1 automatically specifies the +Z direction of the feature coordinate system as the tool axis direction even if no angle is specified for the rotary axis. (See Fig.13.10(c).)

See Fig.13.10(b) for explanations about the tool axis direction.

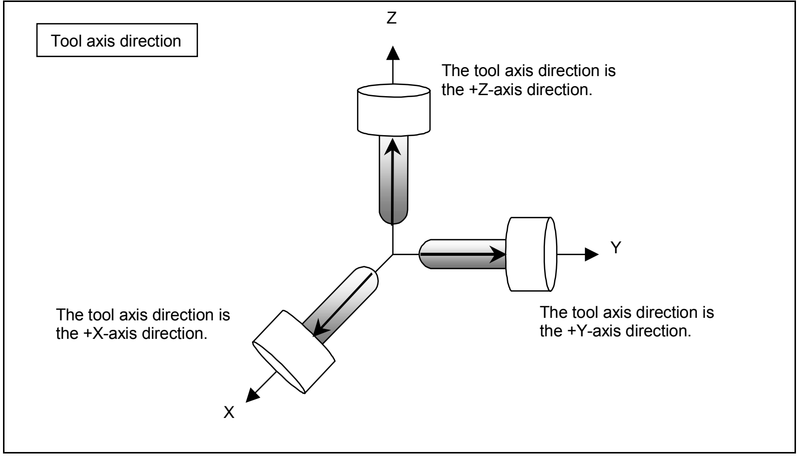

Tool axis direction
Z
The tool axis direction is the +Z-axis direction.
Y
The tool axis direction is the +X-axis direction.
The tool axis direction is the +Y-axis direction.
X

Fig. 13.10 (b)

This function sets the direction normal to the cut surface as the +Z-axis direction of the feature coordinate system. Once G53.1 is issued, the tool is kept perpendicular to the cut surface.

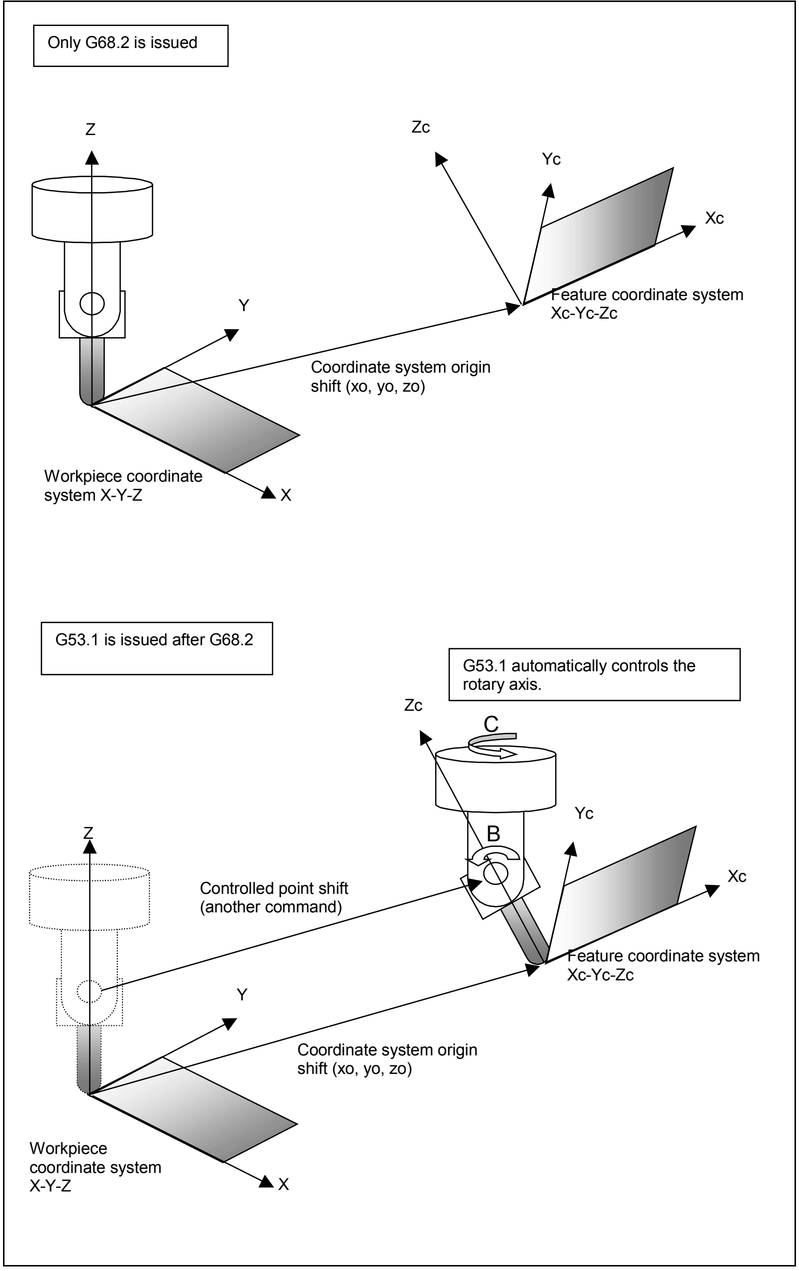


Fig. 13.10 (c)

This function is applicable to the following machine configurations. (See Fig.13.10(d).)

<1> Tool rotation type machine controlled with two tool rotary axes

<2> Table rotation type machine controlled with two table rotary axes

<3> Mixed-type machine controlled with one tool rotary axis and one rotary axis

This function is usable also in a configuration where the rotary axis for controlling the tool and the rotary axis for controlling the table do not cross each other.

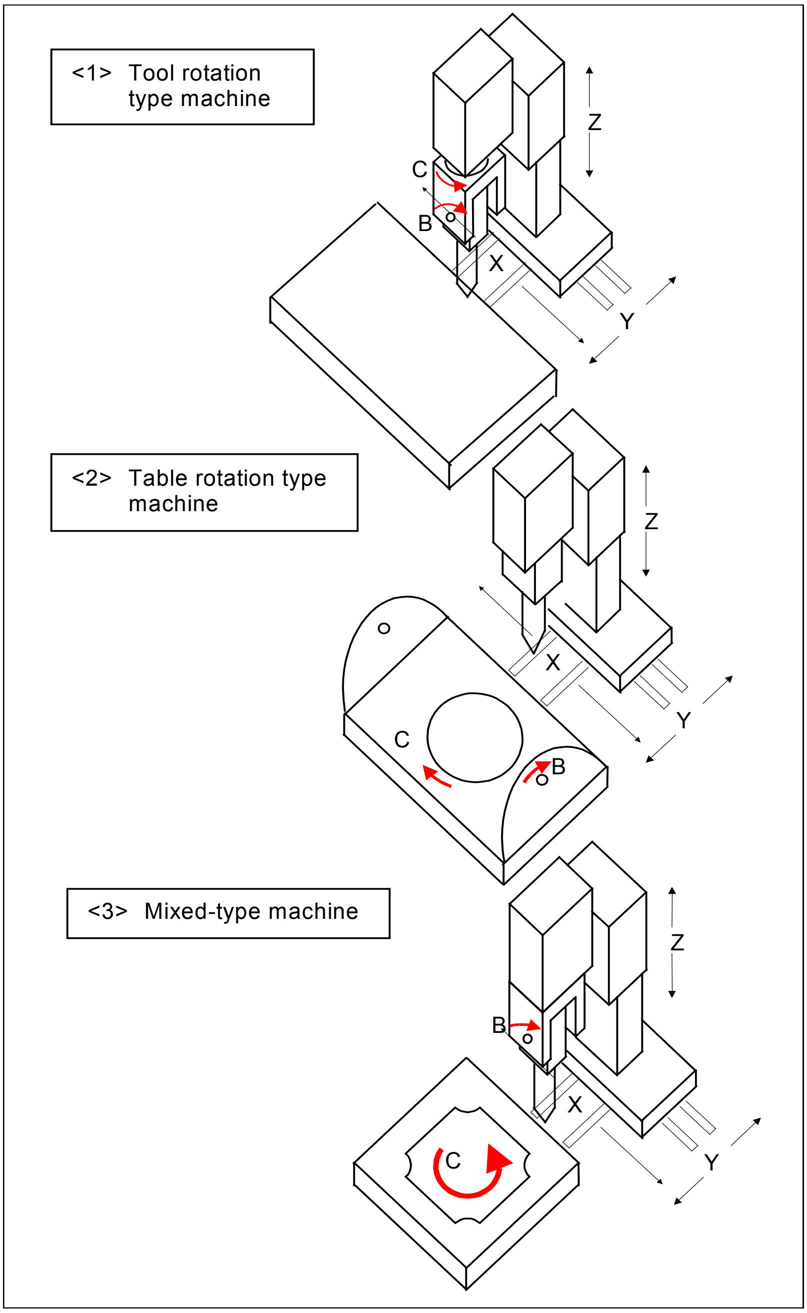

<1> Tool rotation type machine
C
B
Z
X
Y
<2> Table rotation type machine
Z
X
Y
C
B
<3> Mixed-type machine
Z
B
X
Y
C

Fig.13.10 (d)

Format

- Feature coordinate system setting

Format	
G68.2 X $\underline{x_0}$ Y $\underline{y_0}$ Z $\underline{z_0}$ I$\underline{\alpha}$ J$\underline{\beta}$ K$\underline{\gamma}$; Feature coordinate system setting **G69 ;** Cancels the feature coordinate system setting.	
Symbol description	
X, Y, Z	: Feature coordinate system origin
I, J, K	: Euler's angle for determining the orientation of the feature coordinate system

- Tool axis direction control

Format
G53.1 ; Controls the tool axis direction.

⚠ **CAUTION**

1 G53.1 must be issued in a block after the block that contains G68.2.

2 G53.1 must be issued in a block in which there is no other command.

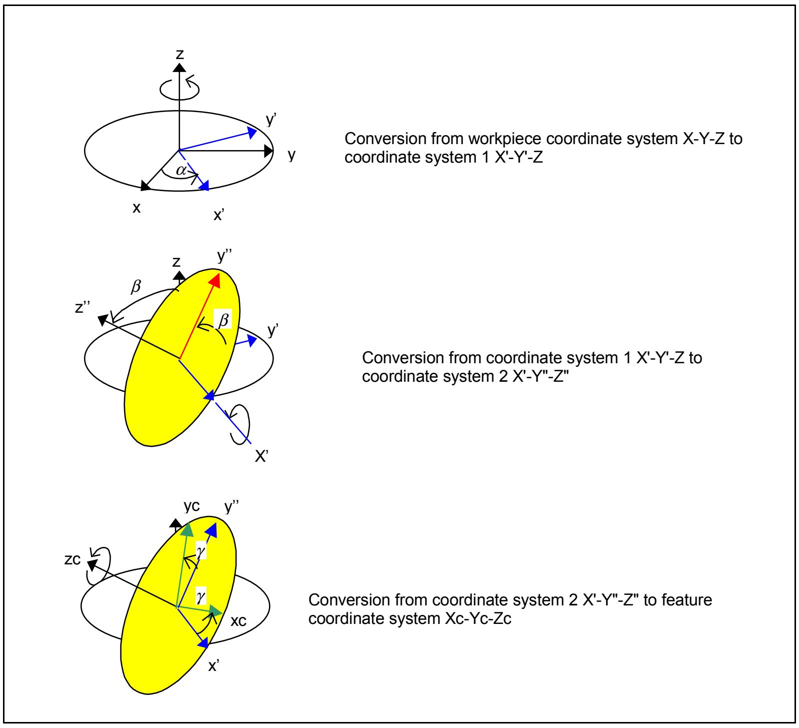
3 The movement speed of the rotary axis depends on the current modal information.

Coordinate conversion in which an Euler's angle is used

Coordinate conversion by rotation is assumed to be performed around the workpiece coordinate system origin.

Let the coordinate system obtained by rotating the workpiece coordinate system around the Z-axis through an angle of (degrees be coordinate system 1. Also let the coordinate system obtained by rotating coordinate system 1 around the X-axis through an angle of (degrees be coordinate system 2. The feature coordinate system is the coordinate system obtained by shifting the coordinate system that is obtained by rotating coordinate system 2 around the Z-axis through an angle of (degrees from the workpiece coordinate system origin by (Xo, Yo, Zo). Fig.13.10(e) shows the relationships between the workpiece coordinate system and the feature coordinate system.

The figure also shows how the X-Y plane is displaced, as an example.



Conversion from workpiece coordinate system X-Y-Z to coordinate system 1 X'-Y'-Z

Conversion from coordinate system 1 X'-Y'-Z to coordinate system 2 X'-Y"-Z"

Conversion from coordinate system 2 X'-Y"-Z" to feature coordinate system Xc-Yc-Zc

Fig.13.10 (e)

Tool rotation type machine

The following paragraphs describe several operations of a tool rotation type machine.

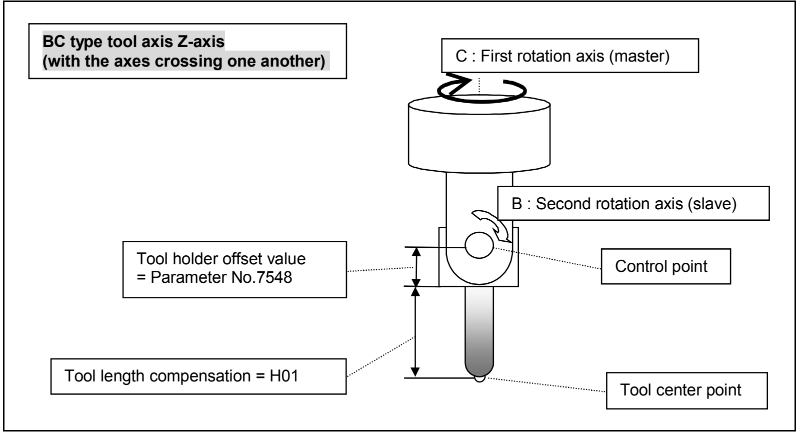
- Operation description 1: If G43 (tool length compensation) is issued on a machine with axes crossing each other

G53.1 issued after G68.2 automatically controls the rotary axis in such a way that the tool axis will be oriented in the +Z direction of the "feature coordinate system."

O100 is sample program 1.

```
O100 (Sample Program1) ;
N1 G55 ;
N2 G90 G01 X0 Y0 Z30.0 F1000 ;
N3 G68.2 X100.0 Y100.0 Z50.0 I30.0 J15.0 K20.0 ;
N4 G01 X0 Y0 Z30.0 F1000 ;
N5 G53.1 ;
N6 G43 H01 X0 Y0 Z0 ;
N7 . . .
```

In this example of a machine configuration, the "BC type tool axis Z-axis" is used. In addition, the tool axis, tool rotary axis B, and tool rotary axis C cross one another.


BC type tool axis Z-axis
(with the axes crossing one another)
C : First rotation axis (master)
B : Second rotation axis (slave)
Tool holder offset value
= Parameter No.7548
Control point
Tool length compensation = H01
Tool center point

Block N3:

Defines a feature coordinate system in the workpiece coordinate system.

Block N4:

Shifts the controlled point to point Z30.0 in the feature coordinate system.

Block N5:

Places the rotary axes under automatic control.

Block N6:

Performs tool length compensation in the feature coordinate system. The tool center point is shifted to the origin of the feature coordinate system.

Fig.13.10(f) shows the behavior of the machine when it is under control of sample program 1.

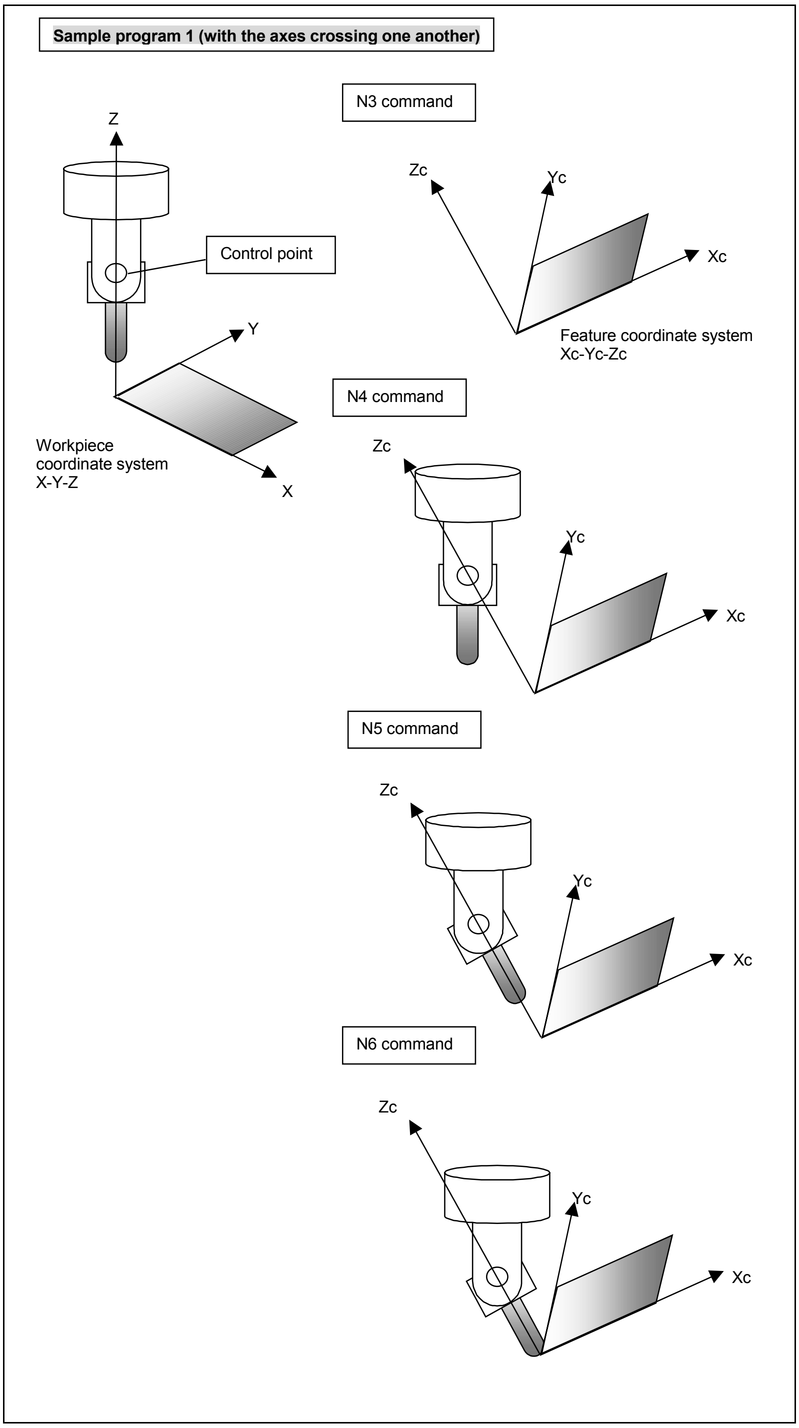
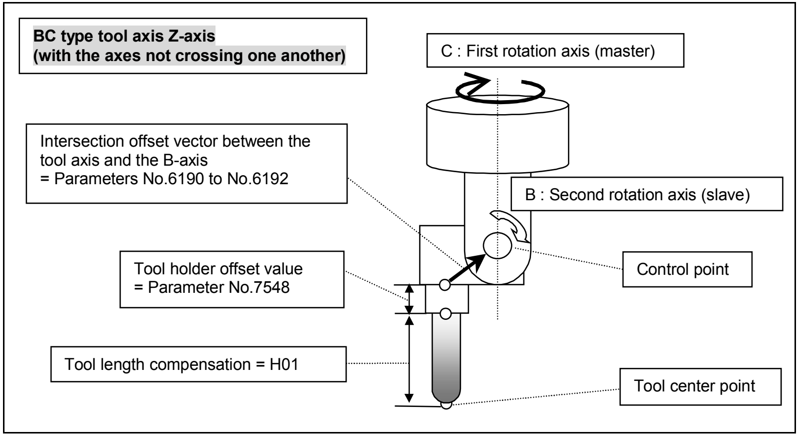


Fig.13.10 (f)

- Operation description 2: If G43 (tool length compensation) is issued on a machine with no axis crossing another

In this example, no axis crosses another.
Let us see how sample program 1 works.

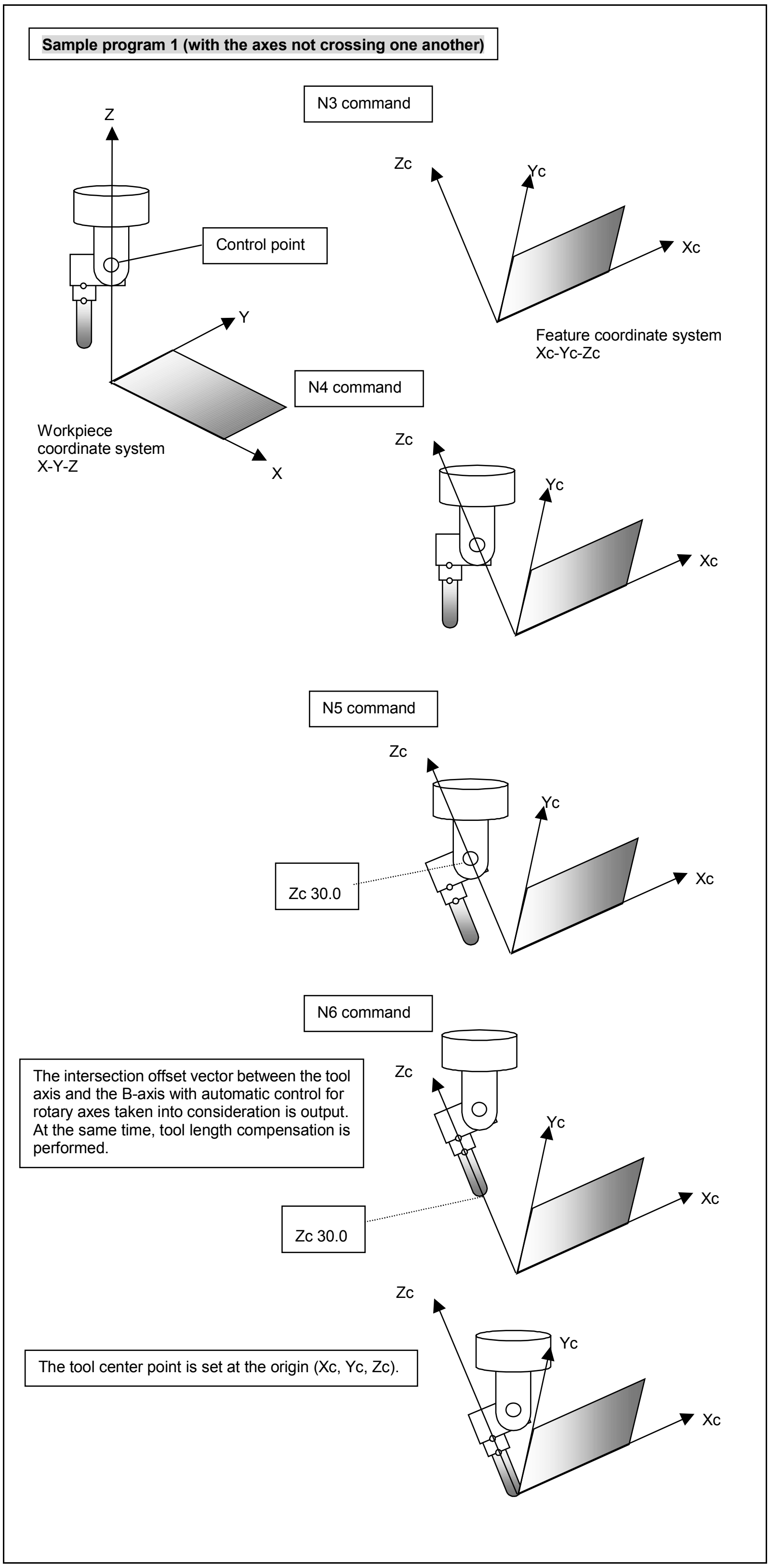
In this example of a machine configuration, the "BC type tool axis Z-axis" is used.
However, the descriptions focus on the case in which the tool axis does not cross the B-axis, but the B-axis and C-axis cross each other.


BC type tool axis Z-axis
(with the axes not crossing one another)
C : First rotation axis (master)
Intersection offset vector between the tool axis and the B-axis = Parameters No.6190 to No.6192
B : Second rotation axis (slave)
Tool holder offset value = Parameter No.7548
Control point
Tool length compensation = H01
Tool center point

Block N3:
Defines a feature coordinate system in the workpiece coordinate system.
Block N4:
Shifts the controlled point to point Z30.0 in the feature coordinate system.
Block N5:
Places the rotary axes under automatic control.
Block N6:
An intersection offset vector between the tool axis and the B-axis with automatic control for rotary axes taken into consideration is output in the feature coordinate system.
Performs tool length compensation in the feature coordinate system.
The tool center point is shifted to the origin of the feature coordinate system.

This is also true when the B-axis does not cross the C-axis.
See descriptions about parameter No. 6193, No. 6194, and No. 6195 for explanations about what the offset is like when the B-axis does not cross the C-axis.

Fig.13.10(g) shows the behavior of the machine when it is under control of sample program 1.



Sample program 1 (with the axes not crossing one another)

The intersection offset vector between the tool axis and the B-axis with automatic control for rotary axes taken into consideration is output. At the same time, tool length compensation is performed.

The tool center point is set at the origin (Xc, Yc, Zc).

Fig.13.10 (g)

- Operation description 3: If no G43 (tool length compensation) is issued or if no G53.1 (tool axis direction control) is issued

Sample program 2 (O200) is equivalent to sample program 1 except that sample program 2 has no tool length compensation command (G43).

```
O200 (Sample Program2) ;
N1 G55 ;
N2 G90 G01 X0 Y0 Z30.0 F1000 ;
N3 G68.2 X100.0 Y100.0 Z50.0 I30.0 J15.0 K20.0 ;
N4 G01 X0 Y0 Z0 F1000 ;
N5 G53.1 ;
N6 . . . ;
```

In this example of a machine configuration, the "BC type tool axis Z-axis" is used.
The descriptions focus on the case in which the axes cross one another and the case in which no axis crosses another.
Fig.13.10(h) shows the behavior of the machine when it is under control of sample program 2.

Sample program 3 (O300) is equivalent to sample program 1 except that sample program 3 has no tool axis direction control command (G53.1).

```
O300 (Sample Program3) ;
N1 G55 ;
N2 G90 G01 X0 Y0 Z30.0 F1000 ;
N3 G68.2 X100.0 Y100.0 Z50.0 I30.0 J15.0 K20.0 ;
N4 G01 X0 Y0 Z0 F1000 ;
N5 G43 H01 ;
N6 . . . ;
```

In this example of a machine configuration, the "BC type tool axis Z-axis" is used.
The descriptions focus on the case in which the axes cross one another and the case in which no axis crosses another.
Tool length compensation is applied in the +Z axis direction of the feature coordinate system.
Fig.13.10(i) shows the behavior of the machine when it is under control of sample program 3.

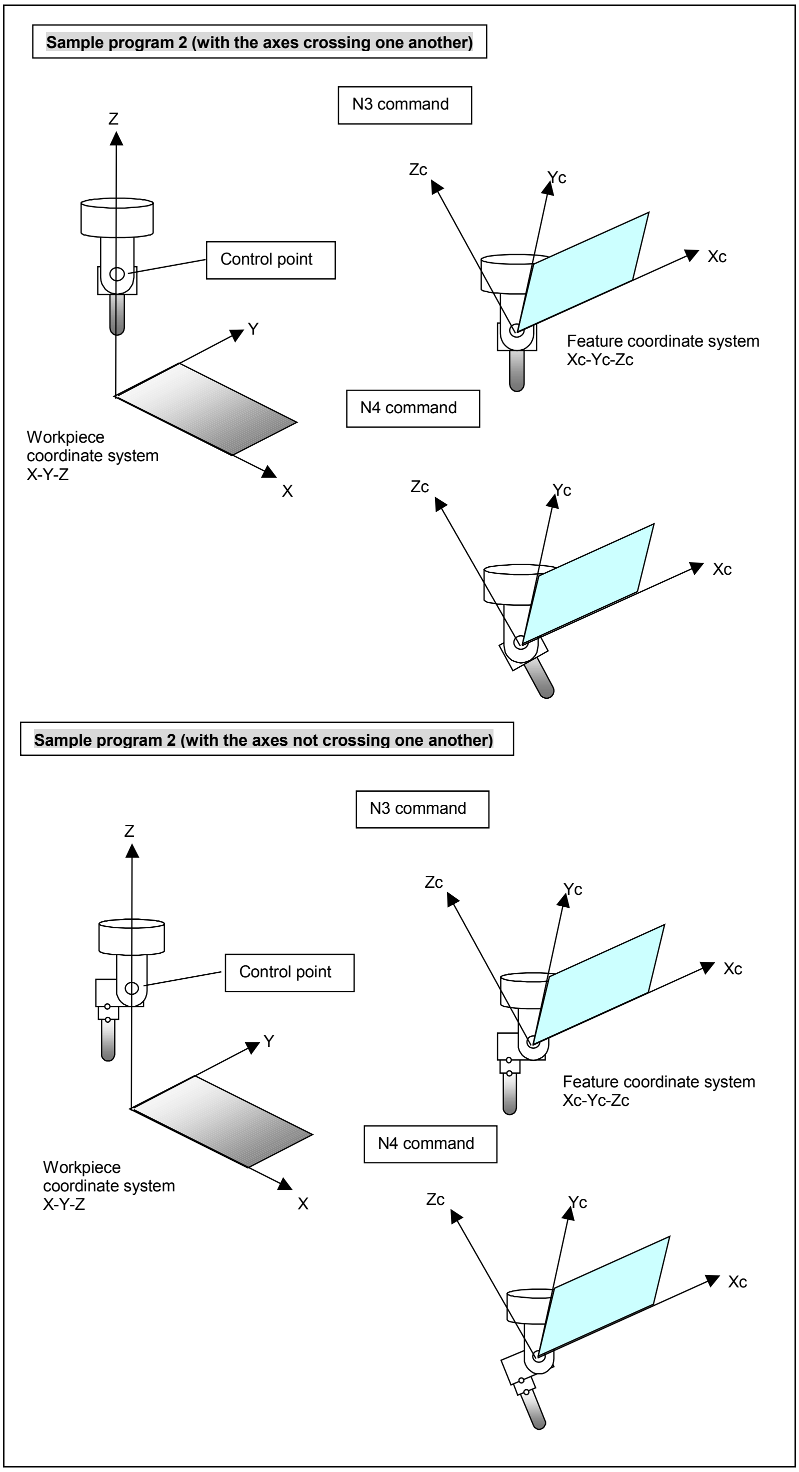
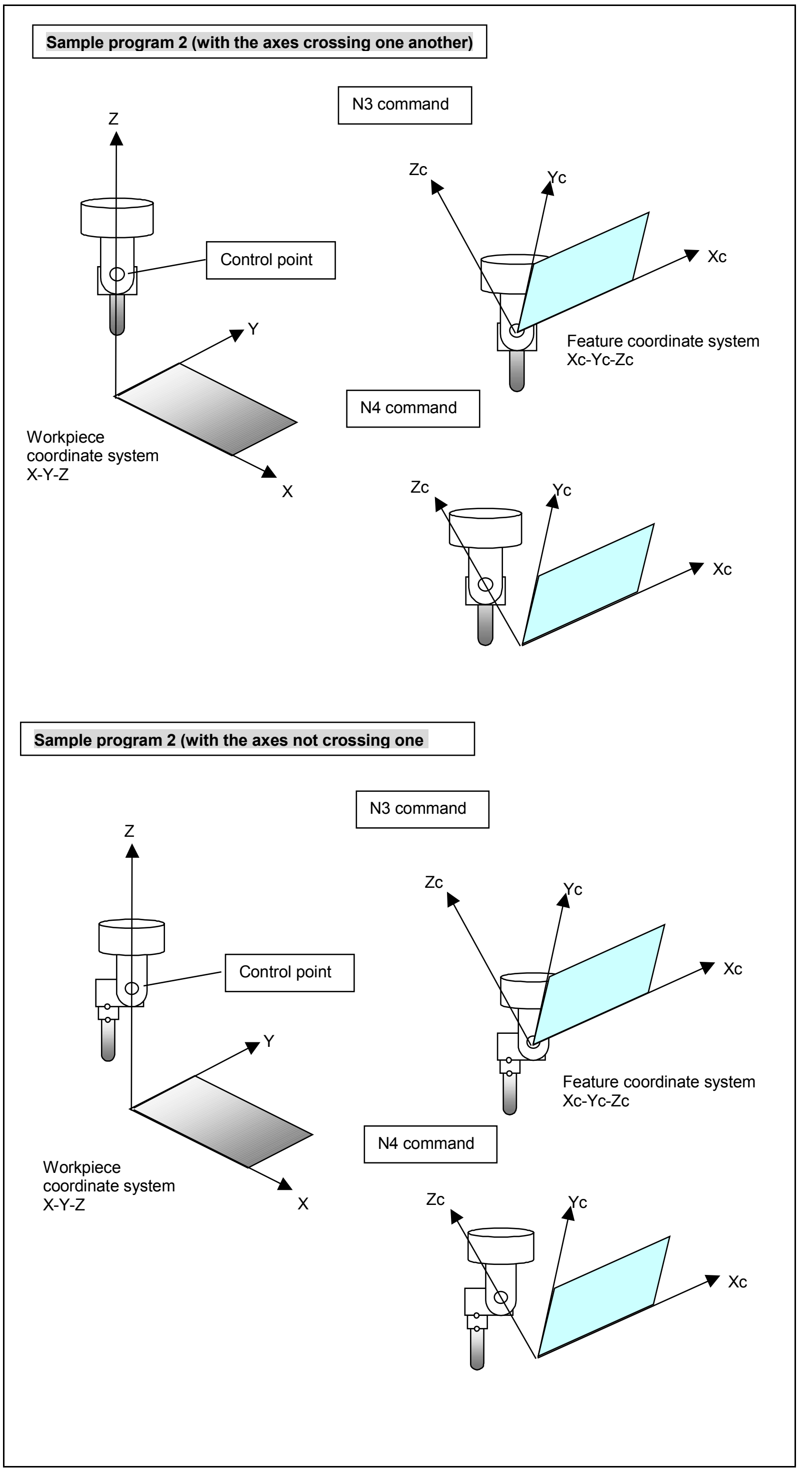

Sample program 2 (with the axes crossing one another)
N3 command
Z
Zc
Yc
Xc
Control point
Y
Feature coordinate system
Xc-Yc-Zc
N4 command
Workpiece
coordinate system
X-Y-Z
X
Zc
Yc
Xc
Sample program 2 (with the axes not crossing one another)
N3 command
Z
Zc
Yc
Xc
Control point
Y
Feature coordinate system
Xc-Yc-Zc
N4 command
Workpiece
coordinate system
X-Y-Z
X
Zc
Yc
Xc

Fig.13.10 (h)



Sample program 2 (with the axes crossing one another)

Sample program 2 (with the axes not crossing one

Fig.13.10 (i)

Mixed-type machine

- Basic operation

This function is usable also for a mixed-type machine in which the tool head rotates on the tool rotary axis and the table rotates on the table rotary axis.

The feature coordinate system Xc-Yc-Zc is set with a coordinate system origin shift (xo, yo, zo) and an Euler's angle in the workpiece coordinate system.

Given the A-axis and B-axis shown in Fig.13.10(j), rotary axis control is performed in such a way that the A-axis rotates until Zc comes in the X-Z plane and the B-axis is controlled so that the tool axis direction becomes the +Z axis of the feature coordinate system.

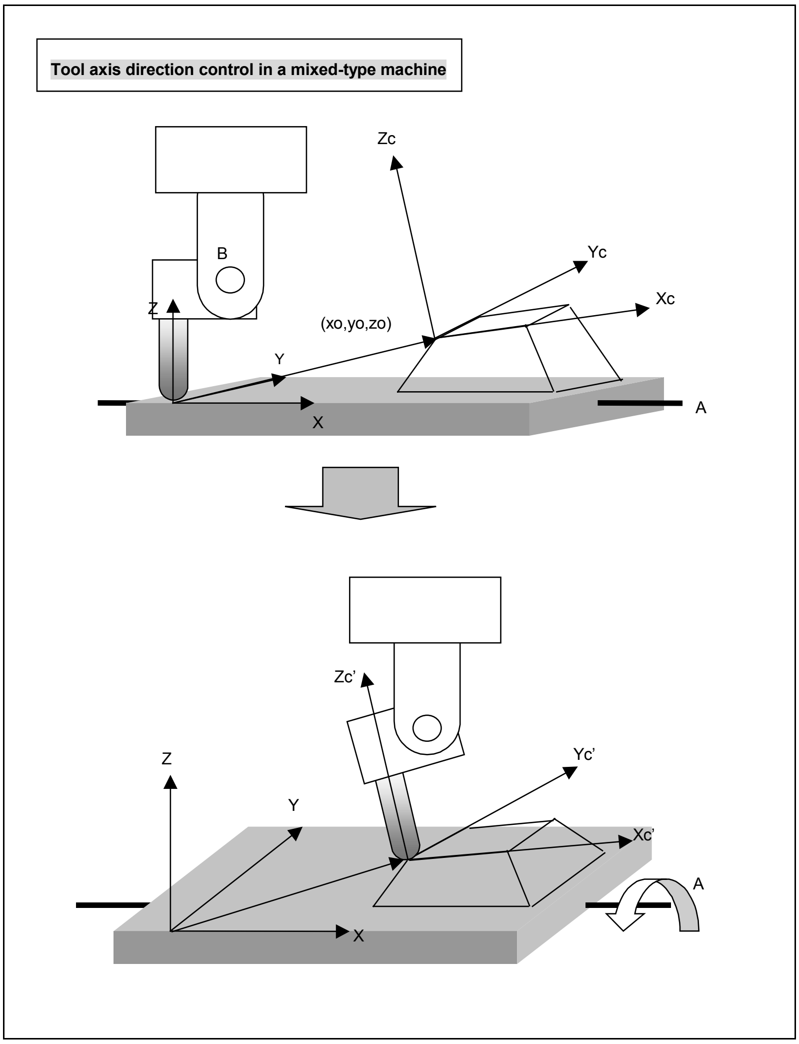


Fig.13.10 (j)

- Feature coordinate system with the table rotated by G53.1 (tool axis direction control)

Let's take a mixed-type machine shown in Fig.13.10(j) as an example. If the table rotates under tool axis direction control (G53.1), the feature coordinate system (called the first feature coordinate system) that is set in the workpiece coordinate system as directed with the feature coordinate system set command (G68.2) rotates as much as the table rotates.

Let the feature coordinate system that has rotated be the second feature coordinate system.

Once G53.1 is issued, the subsequent cutting commands are assumed to be issued in the second feature coordinate system. (See Fig.13.10(k).)

In the mixed-type and table rotation type machines, the specified feature coordinate system (the first feature coordinate system) may vary from the feature coordinate system to be used in cutting (the second feature coordinate system).

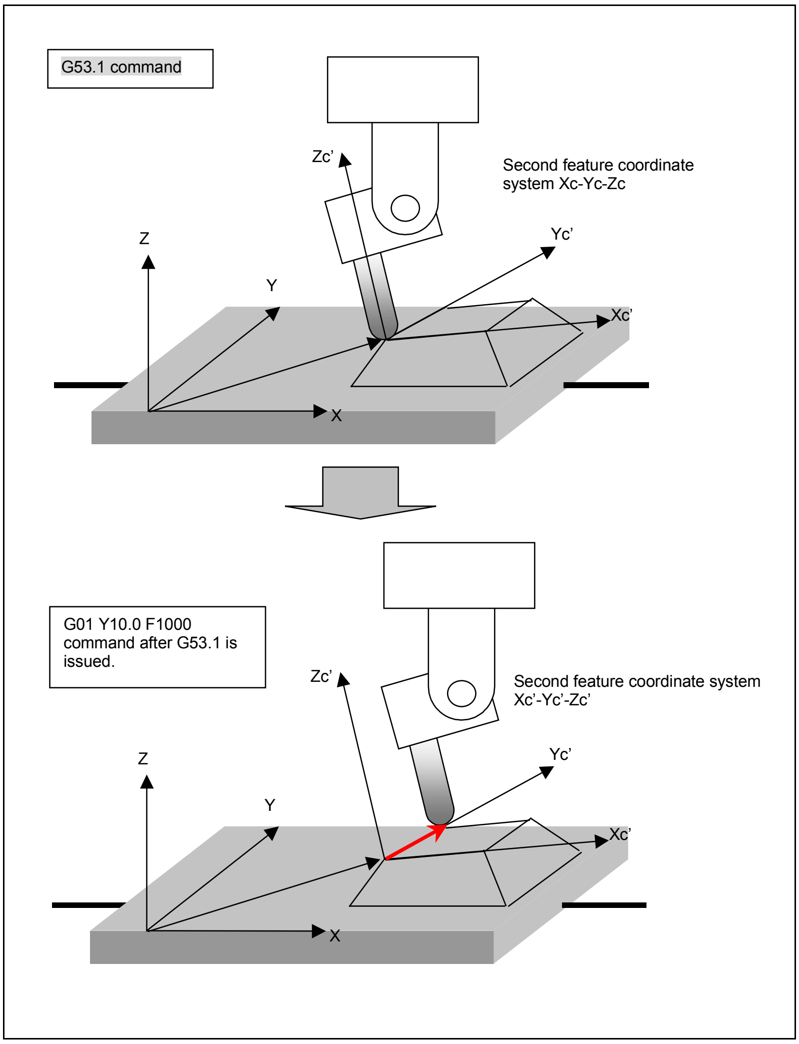

G53.1 command
Zc'
Second feature coordinate system Xc-Yc-Zc
Z
Yc'
Y
Xc'
X
G01 Y10.0 F1000 command after G53.1 is issued.
Zc'
Second feature coordinate system Xc'-Yc'-Zc'
Z
Yc'
Y
Xc'
X

Fig.13.10 (k)

- Rotation direction of the table rotary axis

Let's take a mixed type machine shown Fig.13.10(j) as an example.

Setting parameter No. 6170 to 1 specifies that the rotation direction of the rotary table corresponding to the positive-direction move command be clockwise as viewed from the positive direction of the rotation center axis on which the table rotary axis rotates. Resetting parameter No. 6170 to 0 specifies that that rotation direction be counterclockwise.

Let's take the movement of the table when G53.1 is issued in O400 (sample program 4) as an example.

If parameter No. 6170 = 1, control is performed in such a way that A-45.0 is attained.

If parameter No. 6170 = 0, control is performed in such a way that A45.0 is attained.

```
O400 (Sample Program4) ;
N1 G68.2 X100.0 Y100.0 Z0 I180.0 J45.0 K0 ;
N2 G53.1 ;
N3 . . . ;
```

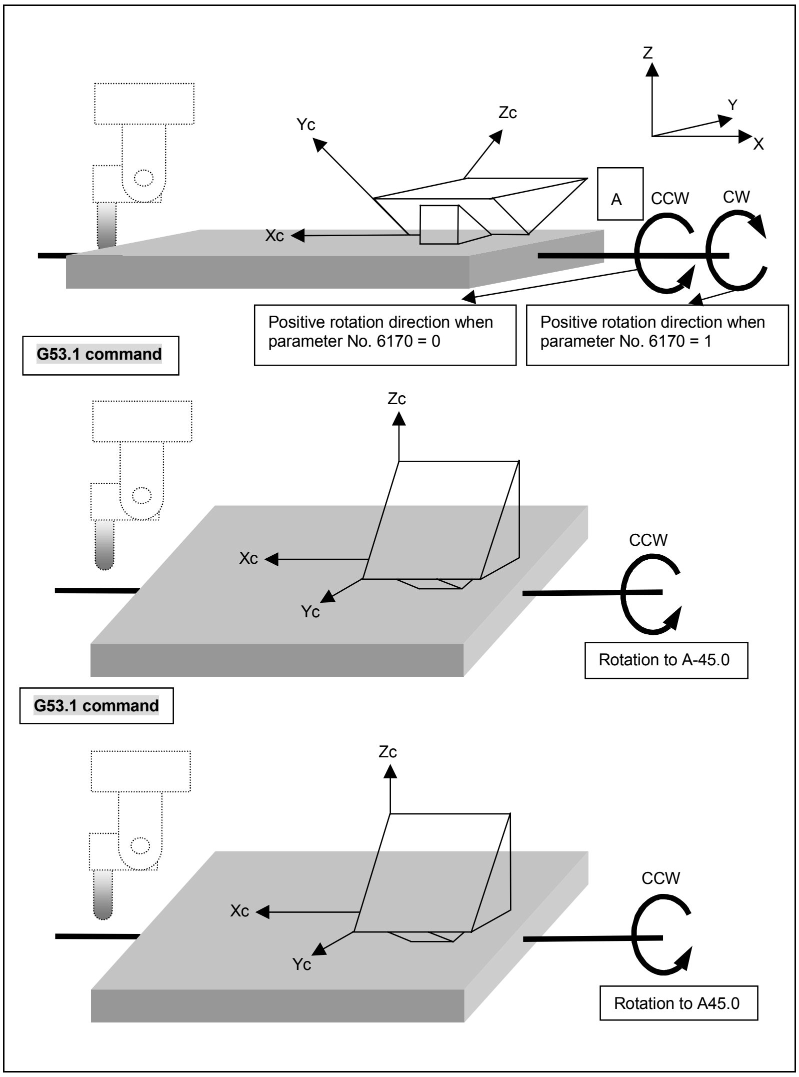

Z
Y
X
Zc
Yc
Xc
A
CCW
CW
Positive rotation direction when parameter No. 6170 = 0
Positive rotation direction when parameter No. 6170 = 1
G53.1 command
Zc
Xc
Yc
CCW
Rotation to A-45.0
G53.1 command
Zc
Xc
Yc
CCW
Rotation to A45.0

Fig.13.10 (l)

Table rotation type machine

- Basic operation

This function is usable also for a table rotation type machine with two table rotary axes.

The feature coordinate system Xc-Yc-Zc is set with a coordinate system origin shift (xo, yo, zo) and an Euler's angle in the workpiece coordinate system.

Given the A-axis and C-axis shown in Fig.13.10(m), the A-axis and C-axis rotate until Zc comes in the X-Z plane and the tool axis direction becomes the +Z axis of the feature coordinate system.

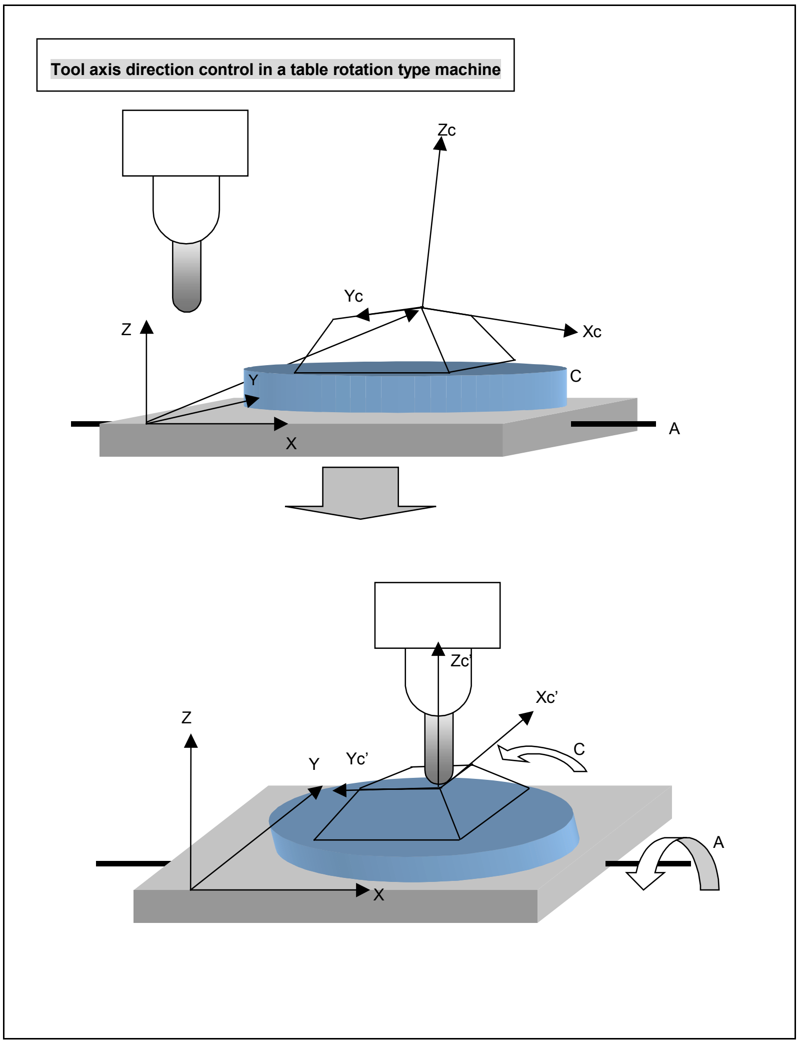

Tool axis direction control in a table rotation type machine
Zc
Yc
Xc
Z
Y
C
A
X
Zc'
Xc'
Z
Y
Yc'
C
A
X

Fig.13.10 (m)

- Feature coordinate system with the table rotated by G53.1 (tool axis direction control)

Let's take a table rotation type machine shown in Fig.13.10(m) as an example.
If the table rotates under tool axis direction control (G53.1), the feature coordinate system (called the first feature coordinate system) that is set in the workpiece coordinate system as directed with the feature coordinate system set command (G68.2) rotates as much as the table rotates.
Let the feature coordinate system that has rotated be the second feature coordinate system.
Once G53.1 is issued, the subsequent cutting commands are assumed to be issued in the second feature coordinate system. (See Fig.13.10(n).)
In the table rotation type machines, the specified feature coordinate system (the first feature coordinate system may vary from the feature coordinate system to be used in cutting the second feature coordinate system.

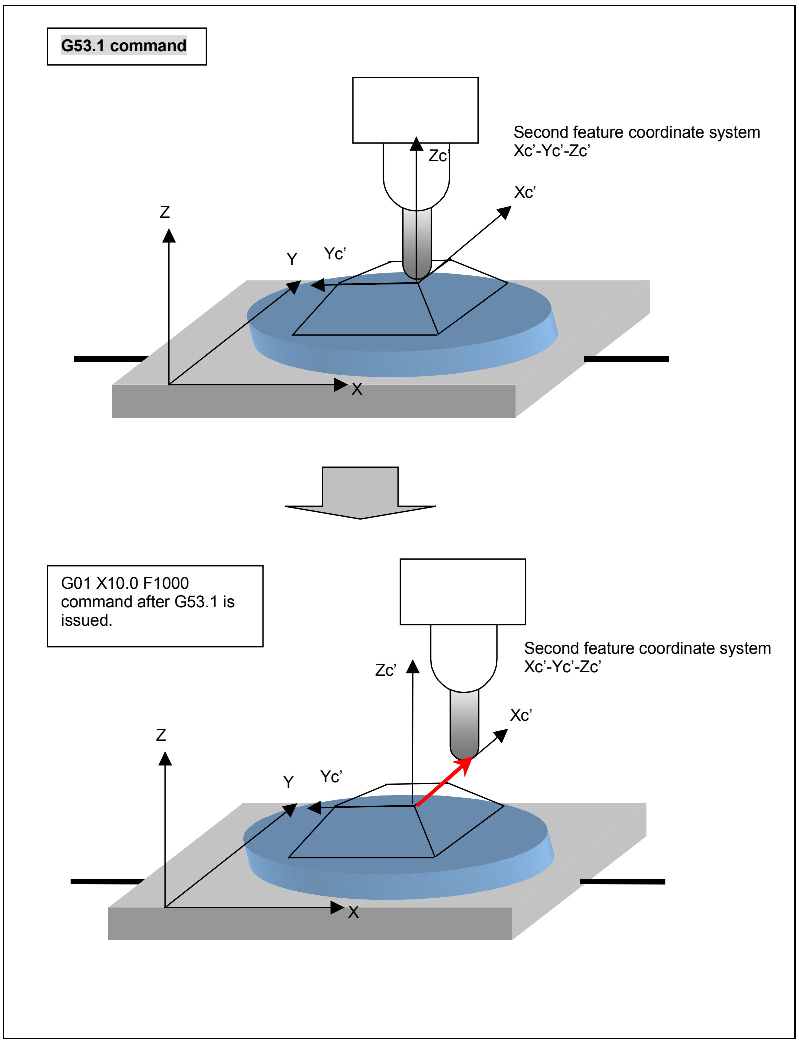


Fig.13.10 (n)

Restrictions

- Basic restrictions

The restrictions for incline cutting commands are similar to those for three-dimensional coordinate conversion. Following are the restrictions that require special attention.

- Increment system

The same increment system must be used for the basic three axes used by this function.

- Rapid traverse command

The rapid traverse command must specify linear rapid traverse (parameter LRP (bit 4 of parameter No. 1400) = 1).

- Feature coordinate system and three-dimensional coordinate conversion

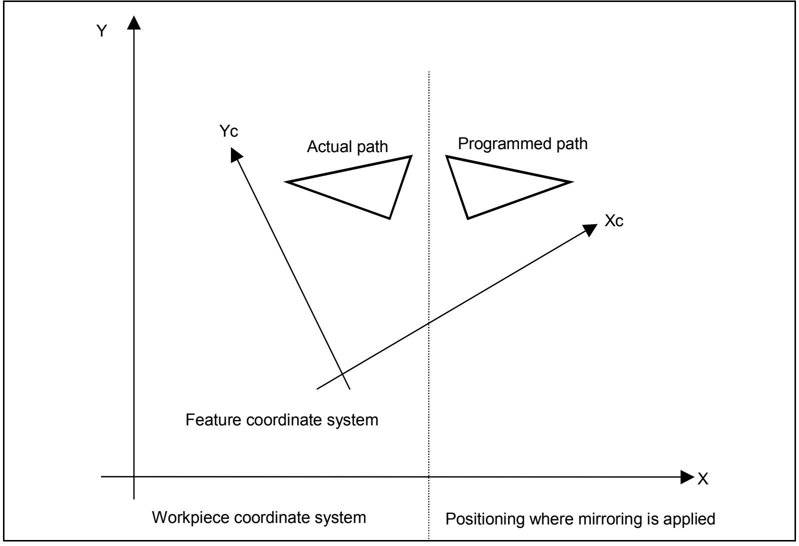
An alarm is raised if an attempt is made to set a feature coordinate system in another feature coordinate system.
An alarm is raised also if an attempt is made to set a new coordinate system by performing three-dimensional coordinate conversion in a coordinate system.

- Positioning in the machine coordinate system

Positioning commands in the machine coordinate system, such as G28, G30, and G53, operate in the machine coordinate system rather than the feature coordinate system.

- External mirror image

If an attempt is made to use this function and an external mirror image function simultaneously, this function takes effect before the external mirror image function.


Y
Yc
Actual path
Programmed path
Xc
Feature coordinate system
X
Workpiece coordinate system
Positioning where mirroring is applied

- Relationships with other modal commands

G41, G42, and G40 (cutter compensation), G43 and G49 (tool length compensation), and G51.1 and G50.1 (programmable mirror image), and canned cycle commands must have nesting relationships with G68.2.

To put another way, first issue G68.2 when the modes mentioned above are off, turn the modes on and off again, and then issue G69.

- Parallel axis control

When a parking signal is issued to one of axes under parallel control, if a move command is issued to another axis, feature coordinate conversion is applied to the axis to which the move command is issued. For this reason, movement may occur also for the axis to which the parking signal was issued.

Alarm and message

Number	Message	Contents
PS645	TOO MANY G68.2 NESTING	A feature coordinate system set command was issued more than once. To newly set a feature coordinate system, cancel the previous commands, then newly issue a feature coordinate system command.
PS646	G68.2 FORMAT ERROR	In a feature coordinate system set command block, the I, J, and K commands are all 0s.
PS647	ILLEGAL USE OF G53.1	(1) A tool axis direction control command was issued before a feature coordinate system set command. Issue the tool axis direction control command in a block after the feature coordinate system set command. (2) There is no angle solution for setting the tool axis direction to the +Z-axis direction of the feature coordinate system. Specify a feature coordinate system that can produce an angle solution.
PS1100	ILLEGAL PARAMETER OF MACHINE COMPONENT	A parameter (No. 6161 to No. 6195 or No. 7540 to No. 7548) for configuring the machine is incorrect.

Reference item

FANUC Series 15*i*/150*i*-MB	Operator’s Manual (Programming) (B-63784EN)	II-13.9	Three-dimensional coordinate conversion (G68,G69)

14 COMPENSATION FUNCTION

General

This chapter describes the following compensation functions:

14.1 TOOL LENGTH OFFSET (G43,G44,G49)

This function can be used by setting the difference between the tool length assumed during programming and the actual tool length of the tool used into the offset memory. It is possible to compensate the difference without changing the program.

Specify the direction of offset with G43 or G44. Select a tool length offset value from the offset memory by entering the corresponding address and number (H code).

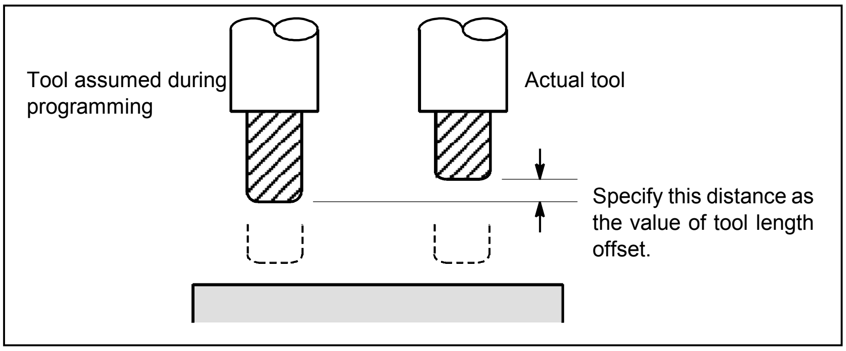

Tool assumed during programming
Actual tool
Specify this distance as the value of tool length offset.

Fig. 14.1 Tool length offset

14.1.1 General

Format

Tool length offset	G43 α_H_ ; G44 α_H_ ;	Explanation of each address G43 : Positive offset G44 : Negative offset α : Address of a specified axis H : Address for specifying the
Tool length offset cancel	G49; or H0;(when the parameter LXY (No.6000#4) :s1)	

Explanation

- Direction of the offset

When G43 is specified, the tool length offset value (stored in offset memory) specified with the H code is added to the coordinates of the end position specified by a command in the program. When G44 is specified, the same value is subtracted from the coordinates of the end position. The resulting coordinates indicate the end position after compensation, regardless of whether the absolute or incremental mode is selected.

If movement along an axis is not specified, the system assumes that a move command that causes no movement is specified. When a positive value is specified for tool length offset with G43, the tool is moved accordingly in the positive direction. When a positive value is specified with G44, the tool is moved accordingly in the negative direction. When a negative value is specified, the tool is moved in the opposite direction.

G43 and G44 are modal G codes. They are valid until another G code belonging to the same group is used. By setting bit 2 (G43) of parameter No. 2401 or bit 3 (G44) of parameter No. 2401, G43 or G44 can be set upon power-up.

- Specification of the tool length offset value

The tool length offset value assigned to the number (offset number) specified in the H code is selected from offset memory and added to or subtracted from the moving command in the program.

The tool length offset value may be set in the offset memory through the MDI panel.

The range of values that can be set as the tool length offset value is as follows.

⚠ **WARNING**

When the tool length offset value is changed due to a change of the offset number, the offset value changes to the new tool length offset value, the new tool length offset value is not added to the old tool length offset value.

H1 : tool length offset value 20.0,H2 : tool length offset value 30.0

G90 G43 Z100.0 H1; Z will move to 120.0

G90 G43 Z100.0 H2; Z will move to 130.0

NOTE
The tool length offset value corresponding to offset No. 0, that is, H0 always means 0. It is impossible to set any other tool length offset value to H0.

- Performing tool length offset along two or more axes

When bit 4 (LXY) of parameter No. 6000 is set to 1, offset can be performed along any axis specified in the program. When two or more axes are specified, offset can be performed on all the specified axes.

```
Offset in X and Y axes
 G43 X_ Y_ H_ ;
```

- Canceling tool length compensation

To cancel the offset, command a G49 or assign offset H0. When G49 is commanded, the canceling action is taken immediately.

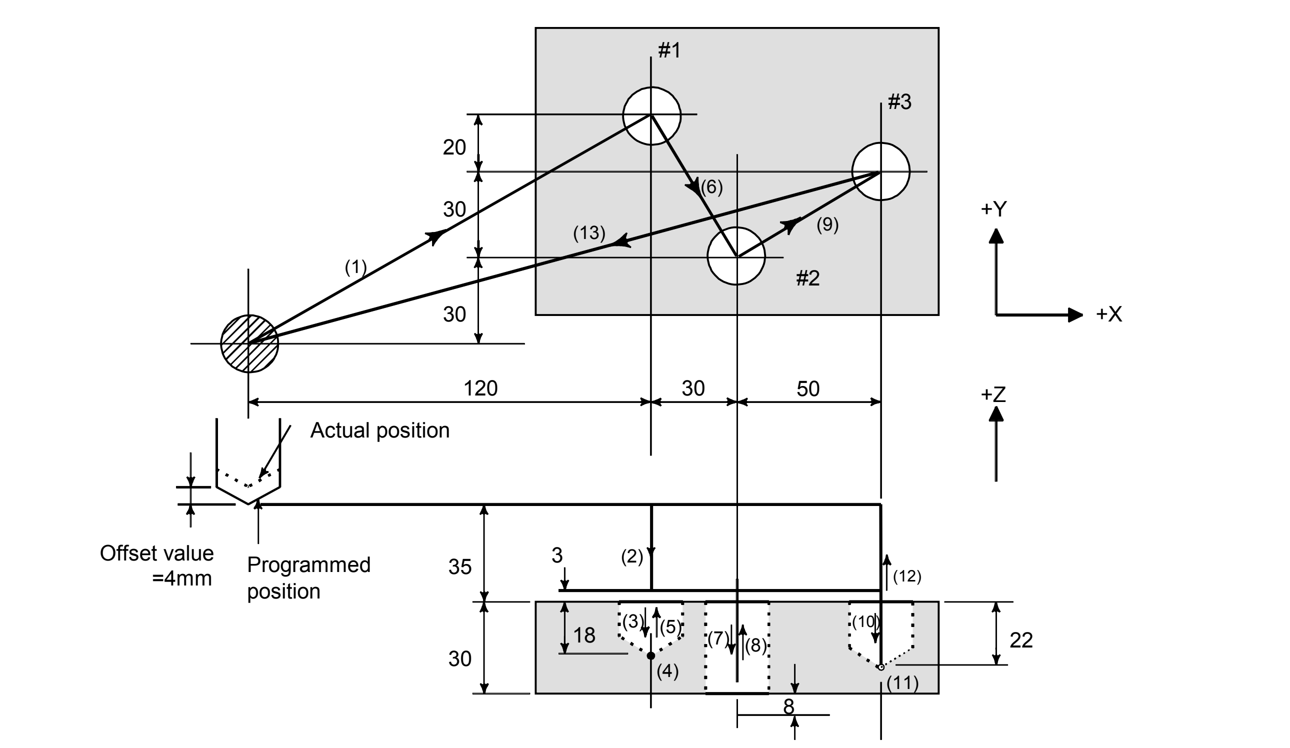
When LXY, bit 4 of parameter No.6000, is set to 0 (tool length compensation is applied along the Z-axis), the offset can be canceled simply by specifying H00.

When LXY, bit 4 of parameter No.6000, is set to 1, tool length compensation is applied along the axis specified in the program.

```
G43 X_ H_ ; (specified tool length compensation along the X-axis)
      :
G43 X_ H0 ; (Cancels tool length compensation along the X-axis)
```

Example

Tool length offset (in boring holes #1, #2, #3)



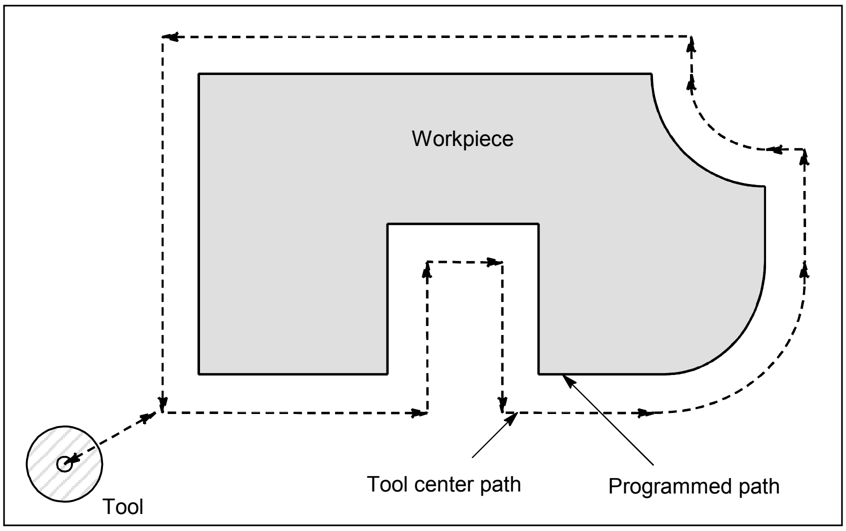
- Program

H1=-4.0 (Tool length offset value)

```
N1  G91 G00 X120.0 Y80.0 ; ................. (1)
N2  G43 Z-32.0 H1 ;  ....................... (2)
N3  G01 Z-21.0 F1000 ;  .................... (3)
N4  G04 P2000 ;............................. (4)
N5  G00 Z21.0 ;............................. (5)
N6  X30.0 Y-50.0 ;.......................... (6)
N7  G01 Z-41.0 ;............................ (7)
N8  G00 Z41.0 ;............................. (8)
N9  X50.0 Y30.0 ;........................... (9)
N10 G01 Z-25.0 ;........................... (10)
N11 G04 P2000 ;  .......................... (11)
N12 G00 Z57.0 H0 ;  ....................... (12)
N13 X-200.0 Y-60.0 ;....................... (13)
N14 M2 ;
```

14.2 TOOL OFFSET(G45-G48)

The programmed travel distance of the tool can be increased or decreased by a specified tool offset value or by twice the offset value. The tool offset function can also be applied to an additional axis.



Format

G45 IP_ D_ ;Increase the travel distance by the tool offset value
G46 IP_ D_ ;Decrease the travel distance by the tool offset value
G47 IP_ D_ ;Increase the travel distance by twice the tool offset value
G48 IP_ D_ ;Decrease the travel distance by twice the tool offset value

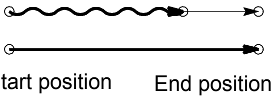
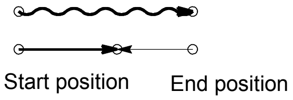
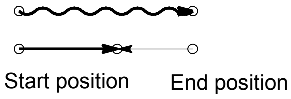
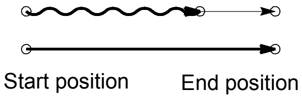
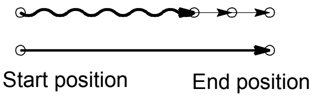
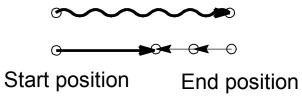
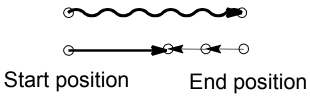
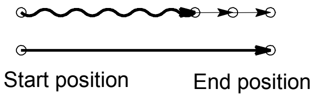
G45 to 48 : One-shot G code for increasing or decreasing the travel distance
IP : Command for moving the tool
D : Code for specifying the tool offset value

Explanation

- Increase and decrease

As shown in Table 14.2(a), the travel distance of the tool is increased or decreased by the specified tool offset value.
In the absolute mode, the travel distance is increased or decreased as the tool is moved from the end position of the previous block to the position specified by the block containing G45 to G48.

Table 14.2 (a) Increase and decrease of the tool travel distance

G code	When a positive tool offset value is specified	When a negative tool offset value is specified
G45		
G46		
G47		
G48		

〰► Programmed movement
→ Tool offset value
⟶ Actual movement position

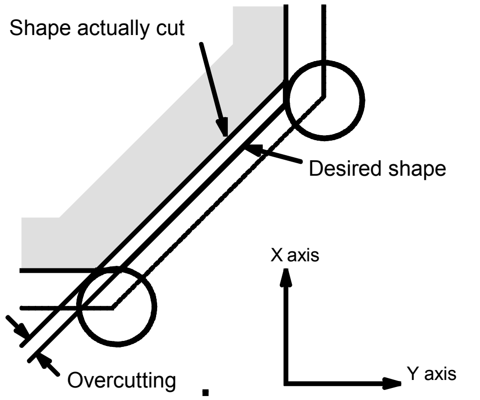
If a move command with a travel distance of zero is specified in the incremental command (G91) mode, the tool is moved by the distance corresponding to the specified tool offset value.
If a move command with a travel distance of zero is specified in the absolute command (G90) mode, the tool is not moved.

- Tool offset value

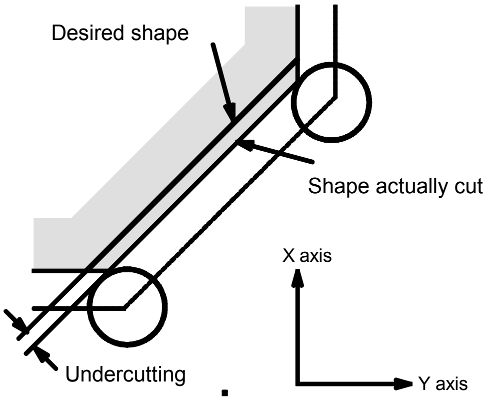
Once selected by D code, the tool offset value remains unchanged until another tool offset value is selected.

⚠ **WARNING**

1 When G45 to G48 is specified to n axes (n=1-6) simultaneously in a motion block, offset is applied to all n axes.
When the cutter is offset only for cutter radius or diameter in taper cutting, overcutting or undercutting occurs.
Therefore, use cutter compensation (G40 or G42) shown in II-14.4 or 14.5.



```
    ⋮
G01 X_ F_ ;
G47 X_ Y_ D_ ;
   Y_ ;
    ⋮
```

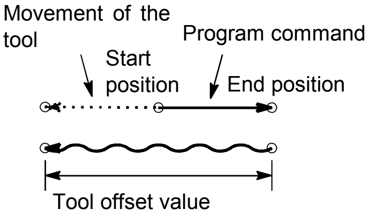


```
           ⋮
G01 G45 X_ F_ D_;
       X_ Y_ ;
G45 Y_ ;
           ⋮
```

2 G45 to G48 (tool offset) must not be used in the G41 or G42 (cutter compensation) mode.

NOTE

1 When the specified direction is reversed by decrease as shown in the figure below, the tool moves in the opposite direction.



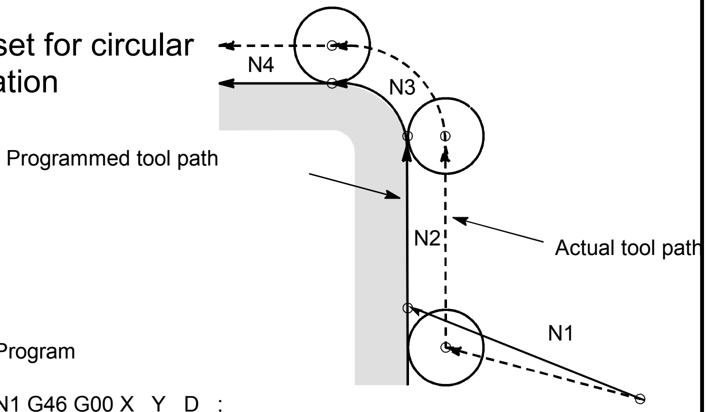
Example

G46 X2.50 ;
Tool offset value
+3.70

Equivalent command
X-1.20 ;

2 Tool offset can be applied to circular interpolation (G02, G03) with the G45 to G48 commands only for 1/4 and 3/4 circles using addresses I, J and K by the parameter setting, providing that the coordinate rotation be not specified at the same time. This function is provided for compatibility with the conventional CNC tape without any cutter compensation. The function should not be used when a new CNC program is prepared.

Tool offset for circular interpolation



Program

```
N1 G46 G00 X_ Y_ D_ ;
N2 G45 G01 Y_ F_ ;
N3 G45 G03 X_ Y_ I_ ;
N4 G01 X_ ;
```

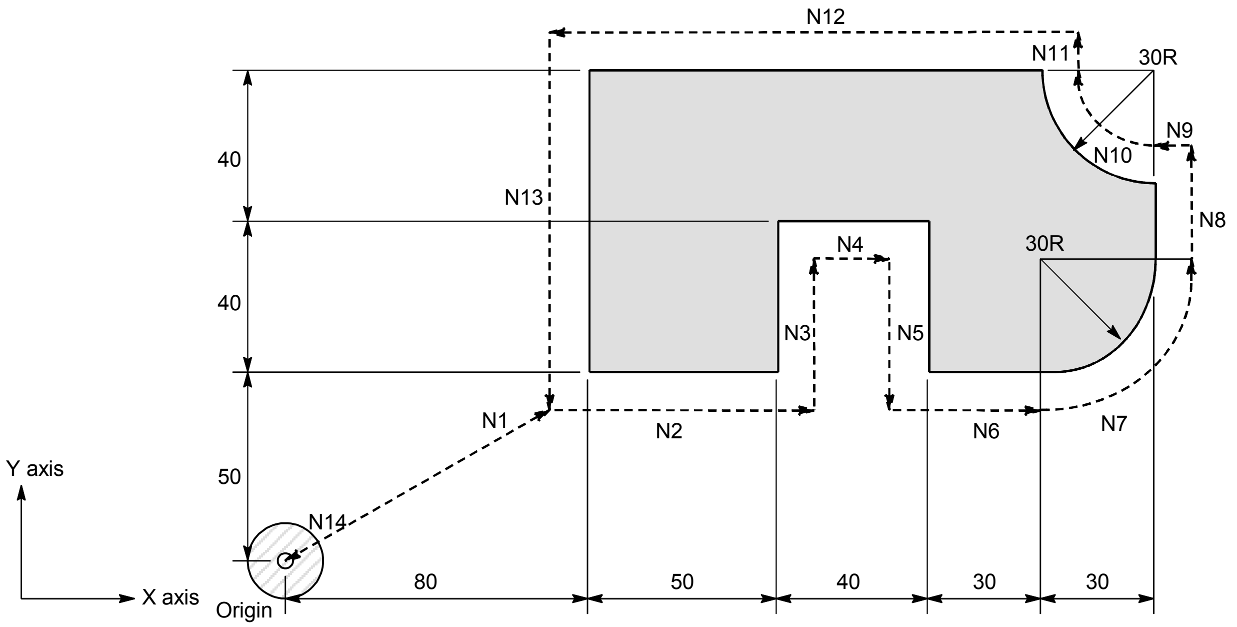
3 D code should be used in tool offset mode (G45 to G48). However, H code can be used by setting the parameter PNH (No. 6001#5) because of compatibility with conventional CNC tape format. The H code must be used under tool length offset cancel (G49).

4 G45 to G48 are ignored in canned cycle mode. Perform tool offset by specifying G45 to G48 before entering canned cycle mode and cancel the offset after releasing the canned cycle mode.

5 Never use G45 through G48 during coordinate system rotation.

Example

Program using tool offset



Tool diameter	: 20ϕ
Offset No.	: 01
Tool offset value	: +10.0

Program

```
N1 G91 G46 G00 X80.0 Y50.0 D01 ;
N2 G47 G01 X50.0 F120.0 ;
N3 Y40.0 ;
N4 G48 X40.0 ;
N5 Y-40.0 ;
N6 G45 X30.0 ;
N7 G45 G03 X30.0 Y30.0 J30.0 ;
N8 G45 G01 Y20.0 ;
N9 G46 X0 ;        (Decreases toward the positive direction for
                   movement amount "0".  The tool moves in the -X
                   direction by the offset value.)
N10 G46 G02 X-30.0 Y30.0 J30.0 ;
N11 G45 G01 Y0 ; (Increase toward the positive direction for
                   movement amount "0".  The tool moves in the +Y
                   direction by the offset value.)
N12 G47 X-120.0 ;
N13 G47 Y-80.0 ;
N14 G46 G00 X-80.0 Y-50.0 ;
```

14.3 OVERVIEW OF CUTTER COMPENSATION (G40 - G42)

When the tool is moved, the tool path can be shifted by the radius of the tool (Fig.14.3 (a)).
To make an offset as large as the radius of the tool, CNC first creates an offset vector with a length equal to the radius of the tool (start-up).
The offset vector is perpendicular to the tool path. The tail of the vector is on the workpiece side and the head positions to the center of the tool.
If a linear interpolation or circular interpolation command is specified after start-up, the tool path can be shifted by the length of the offset vector during machining.
To return the tool to the start position at the end of machining, cancel the cutter compensation mode.

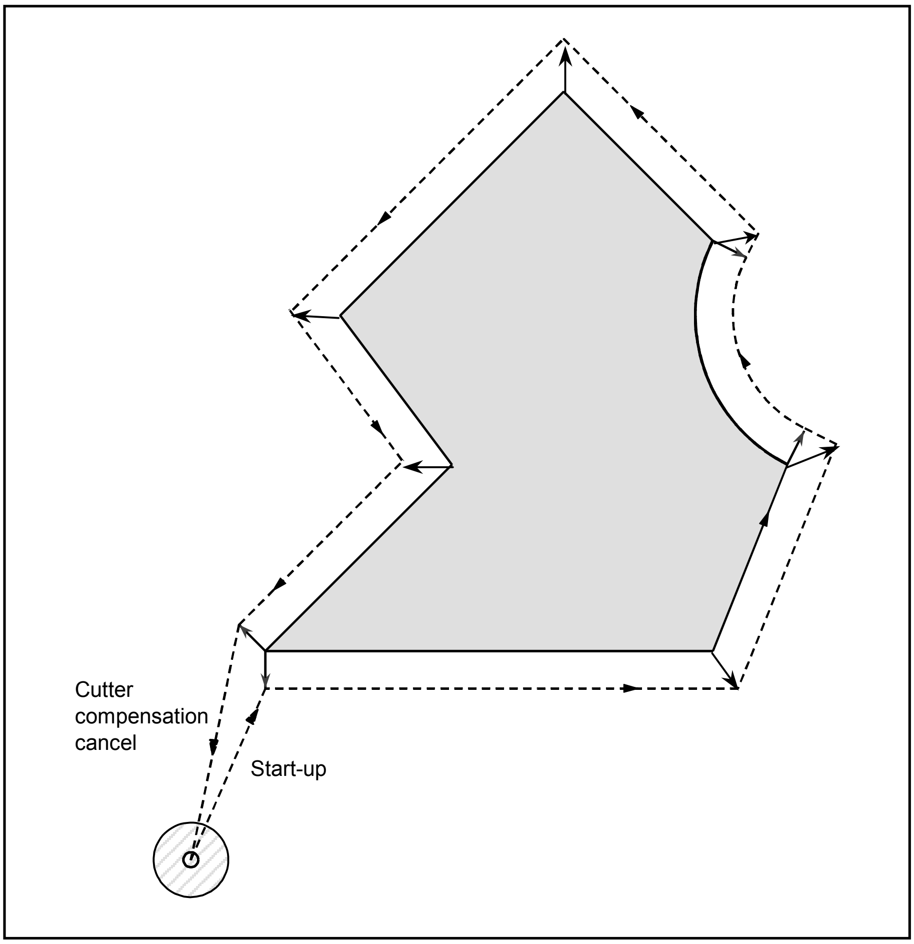


Fig.14.3 (a) Outline of Cutter Compensation

Format

- Start up(Tool compensation start)

G00 (or G01) G41 (or G42) IP_ D_ ;

G41 : Cutter compensation left (Group07)
G42 : Cutter compensation right (Group07)
IP_ : Command for axis movement
D_ : Code for specifying as the cutter compensation value(1-3digits) (D code)

- Cutter compensation cancel (offset mode cancel)

G40 IP_ ;

G40 : Cutter compensation cancel(Group 07)
(Offset mode cancel)
IP_ : Command for axis movement

- Selection of the offset plane

Offset plane	Command for plane selection	IP_
XpYp	G17 ;	Xp_Yp_
ZpXp	G18 ;	Xp_Zp_
YpZp	G19 ;	Yp_Zp_

Explanation

- Offset cancel mode

At the beginning when power is applied the control is in the cancel mode. In the cancel mode, the vector is always 0, and the tool center path coincides with the programmed path.

- Start Up

When a cutter compensation command (G41 or G42, nonzero dimension words in the offset plane, and D code other than D0) is specified in the offset cancel mode, the CNC enters the offset mode.
Moving the tool with this command is called start-up.
Specify positioning (G00) or linear interpolation (G01) for start-up. If circular interpolation (G02, G03) is specified, alarm PS0270 occurs.
When reading a start-up block and subsequent blocks, the CNC first reads the number of blocks specified in parameter No. 6009.

- Offset mode

In the offset mode, compensation is accomplished by positioning (G00), linear interpolation (G01), or circular interpolation (G02, G03). If two or more blocks that do not move the tool (miscellaneous function, dwell, etc.) are processed in the offset mode, the tool will make either an excessive or insufficient cut. If the offset plane is switched in the offset mode, alarm PS0271 occurs and the tool is stopped.

- Offset mode cancel

In the offset mode, when a block which satisfies any one of the following conditions is executed, the CNC enters the offset cancel mode, and the action of this block is called the offset cancel.

1. G40 has been commanded.
2. 0 has been commanded as the offset number for cutter compensation.

When performing offset cancel, circular arc commands (G02 and G03) are not available. If a circular arc is commanded, an P/S alarm (No. 034) is generated and the tool stops. In the offset cancel, the control executes the instructions in that block and the block in the cutter compensation buffer. In the meantime, in the case of a single block mode, after reading one block, the control executes it and stops. By pushing the cycle start button once more, one block is executed without reading the next block.

Then the control is in the cancel mode, and normally, the block to be executed next will be stored in the buffer register and the next block is not read into the buffer for cutter compensation.

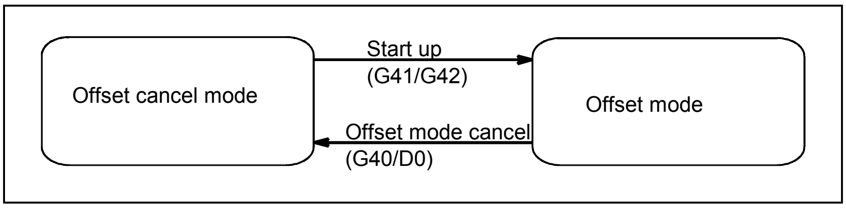


Fig.14.3 (b) Changing the offset mode

- Change of the cutter compensation value

In general, the cutter compensation value shall be changed in the cancel mode, when changing tools. If the cutter compensation value is changed in offset mode, the vector at the end point of the block is calculated for the new cutter compensation value.

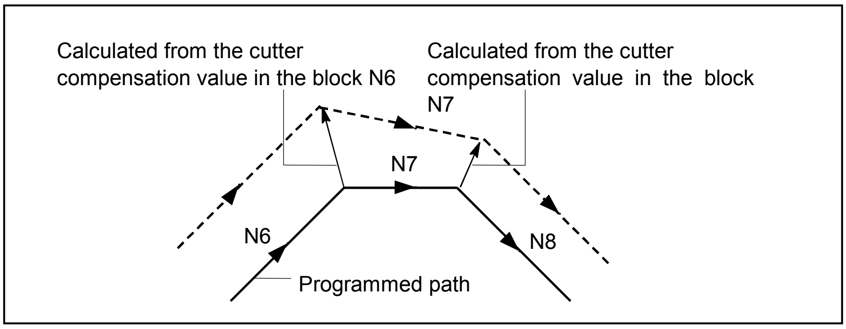

Calculated from the cutter compensation value in the block N6
Calculated from the cutter compensation value in the block N7
N7
N6
N8
Programmed path

Fig.14.3 (c) Changing the cutter compensation value

- Positive/negative cutter compensation value and tool center path

If the offset amount is negative (-), distribution is made for a figure in which G41's and G42's are all replaced with each other on the program. Consequently, if the tool center is passing around the outside of the The figure below shows one example. Generally, the offset amount is programmed to be positive (+).
See Fig.14.3 (d).
When a tool path is programmed as in (1), if the offset amount is made negative (-), the tool center moves as in (2), and vice versa. Consequently, the same tape permits cutting both male and female shapes, and any gap between them can be adjusted by the selection of the offset amount.
Applicable if start-up and cancel is A type.
(See “tart of cutter compensation (start-up)” in the Subsec. II-14.4.1.)

- Setting the cutter compensation value

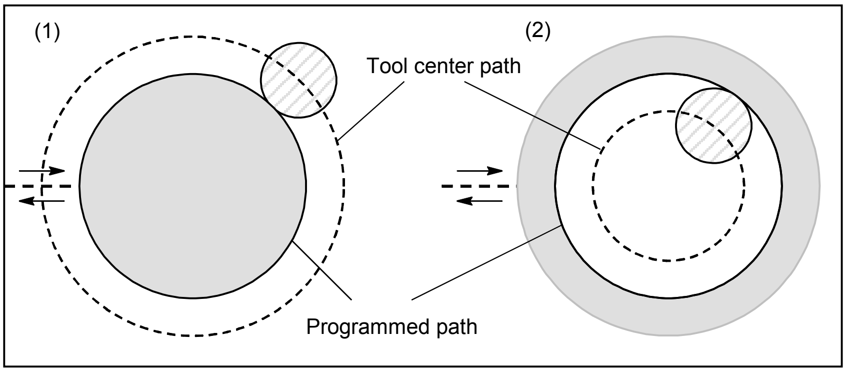


Fig.14.3 (d) Tool Center Paths when Positive and Negative Cutter Compensation Values are Specified

Assign a cutter compensation values to the D codes on the MDI panel.

NOTE
The cutter compensation value corresponding to offset No. 0, that is, D0 always means 0. It is impossible to set D0 to any other offset amount.

- Offset vector

The offset vector is the two dimensional vector that is equal to the cutter compensation value assigned by D code. It is calculated inside the control unit, and its direction is up-dated in accordance with the progress of the tool in each block.
The offset vector is deleted by reset.

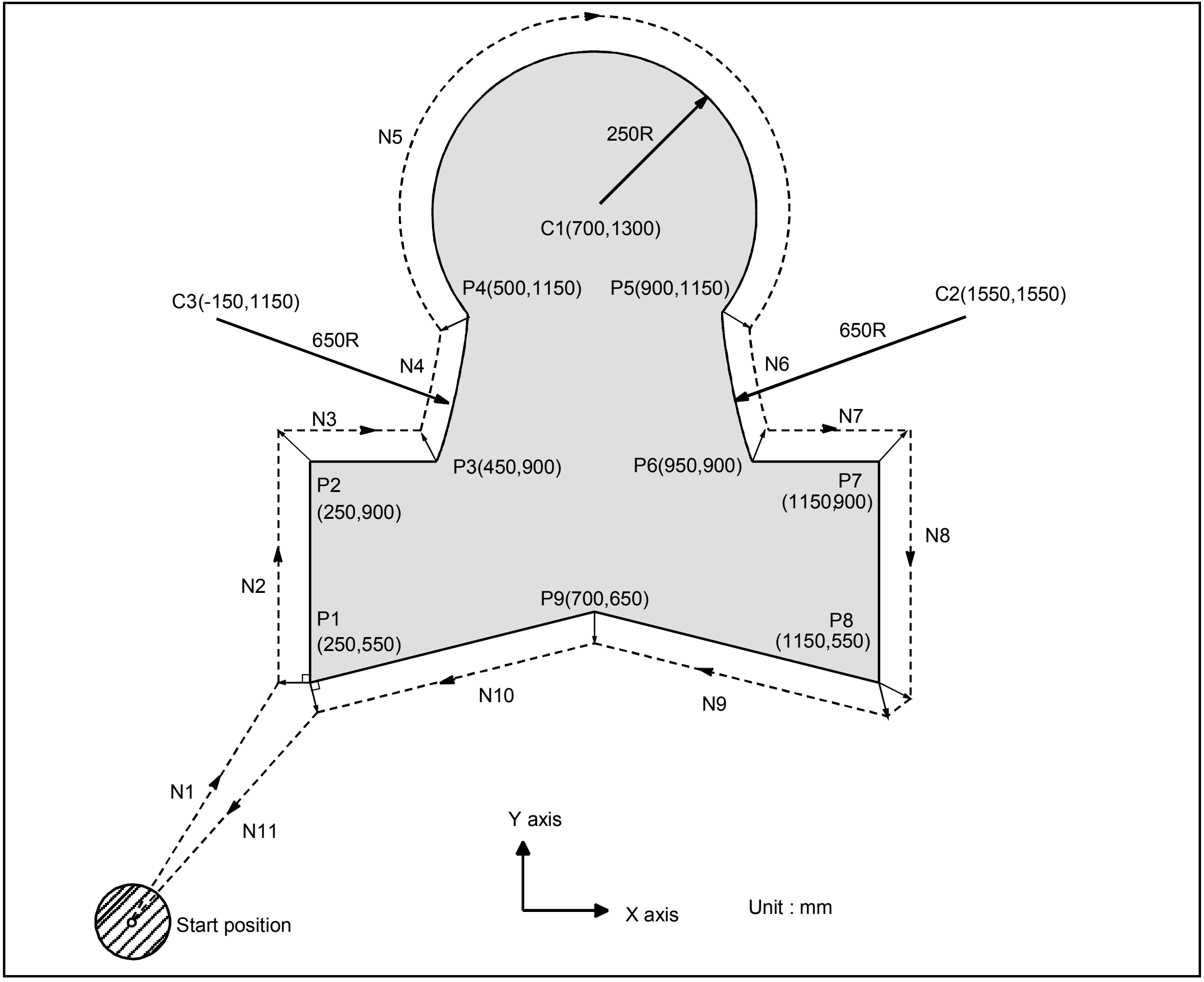
- Specifying a cutter compensation value

Specify a cutter compensation value with a number assigned to it. The number consists of 1 to 3 digits after address D (D code). The D code is valid until another D code is specified. The D code is used to specify the tool offset value as well as the cutter compensation value.

- Plane selection and vector

Offset calculation is carried out in the plane determined by G17, G18 and G19, (G codes for plane selection). This plane is called the offset plane.
Compensation is not executed for the coordinate of a position which is not in the specified plane. The programmed values are used as they are In simultaneous 3 axes control, the tool path projected on the offset plane is compensated.
The offset plane is changed during the offset cancel mode. If it is performed during the offset mode, a alarm (PS0271) is displayed and the machine is stopped.

Example



```
   G92 X0 Y0 Z0 ; ......................... Specifies absolute coordinates.
                                             The tool is positioned at the start position(X0,Y0,Z0).
N1 G90 G17 G00 G41 D07 X250.0 Y550.0 ;
                                             Starts cutter compensation (start-up).
                                             The tool is shifted to the left of the programmed
                                             path by the distance specified in D07.
                                             In other words the tool path is shifted by the radius
                                             of the tool (offset mode) because D07 is set to 15
                                             beforehand (the radius of the tool is 15 mm).
N2 G01 Y900.0 F150 ;................. Specifies machining from P1 to P2.
N3 X450.0 ;................................. Specifies machining from P2 to P3.
N4 G03 X500.0 Y1150.0 R650.0 ;Specifies machining from P3 to P4.
N5 G02 X900.0 R-250.0 ; ............ Specifies machining from P4 to P5.
N6 G03 X950.0 Y900.0 R650.0 ; . Specifies machining from P5 to P6.
N7 G01 X1150.0 ;........................ Specifies machining from P6 to P7.
N8 Y550.0 ;................................. Specifies machining from P7 to P8.
N9 X700.0 Y650.0 ;...................... Specifies machining from P8 to P9.
N10  X250.0 Y550.0 ;................... Specifies machining from P9 to P1.
N11  G00 G40 X0 Y0 ; ................. Cancels the offset mode.
                                             The tool is returned to the start position (X0, Y0, Z0)
```

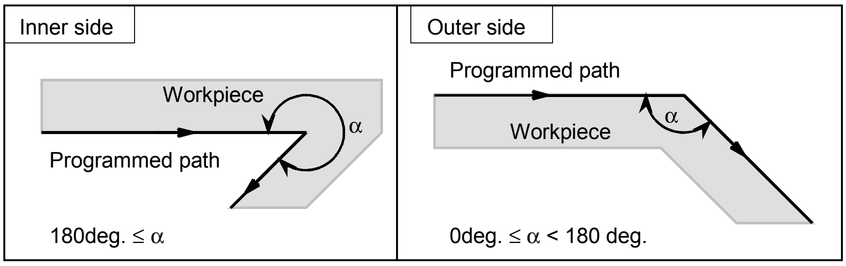
14.4 DETAILS OF CUTTER COMPENSATION

This section provides a detailed explanation of the movement of the tool for cutter compensation outlined in Section 14.6.
This section consists of the following subsections:

14.4.1 General

- Inner side and outer side

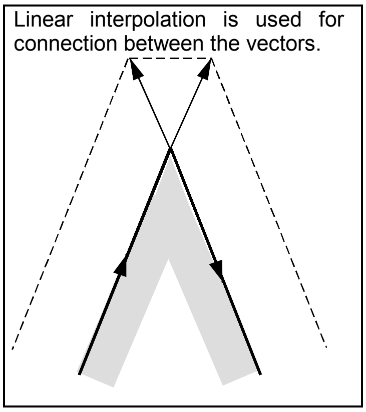
When an angle of intersection created by tool paths specified with move commands for two blocks is over 180deg., it is referred to as "inner side." When the angle is between 0deg. and 180deg., it is referred to as "outer side."



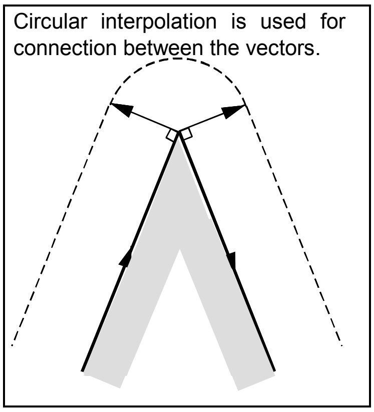
- Outer corner connection method

When the tool moves around an outer corner in cutter compensation mode, the user can choose whether to connect the compensation vectors with each other by linear interpolation or circular interpolation, by setting bit 2 (CCC) of parameter No. 6008.

(1) Linear connection type (bit 2 (CCC) of parameter No. 6008 = 0)


Linear interpolation is used for connection between the vectors.

(2) Circular connection type (bit 2 (CCC) of parameter No. 6008 = 1)


Circular interpolation is used for connection between the vectors.

- Cutter compensation cancel mode

In any of the cases below, cutter compensation cancel mode is set. (The cancel mode may not be set, depending on the specification of the machine tool builder.)

(1)Immediately after power-up
(2)After the reset button on the MDI panel is pressed
(3)After the program is terminated by executing M02 or M30
(4)After execution of the cutter compensation cancel command (G40)

In cancel mode, the magnitude of a compensation vector is always 0, and the tool center path matches a programmed path. A program must end in cancel mode. If a program ends in cutter compensation mode, the tool cannot be positioned to the end point, but is instead moved away from the end point by the compensation vector.

- Start of cutter compensation (start-up)

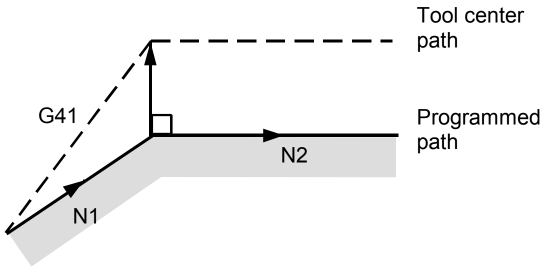
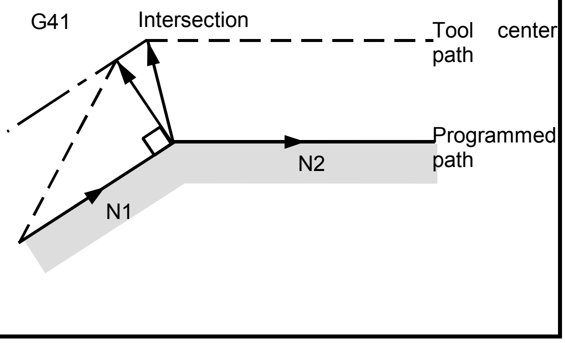
If a block satisfying all the conditions listed below is executed in cancel mode, the machine is placed in cutter compensation mode. This operation is referred to as start-up.

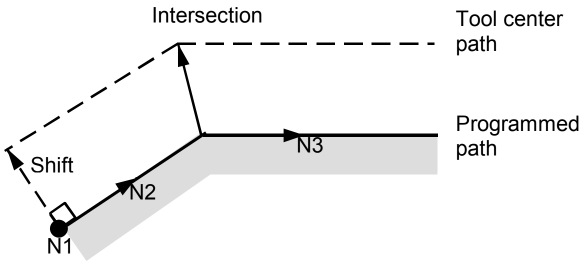
(1) G41 or G42 is specified. Alternatively, G41 or G42 has already been specified, and G41 or G42 mode has already been set.
(2) 0 < cutter compensation number maximum compensation number
(3) Positioning (G00) or linear interpolation (G01) mode is set.
(4) A command specifying a non-zero amount of travel is specified for an axis on the compensation plane. (This applies to types other than start-up type C.)

If start-up is specified in a mode such as the circular interpolation (G02 or G03) mode, an alarm (PS0270) is issued.

As indicated below, three types of start-up operations are supported: type A, type B, and type C. The user can choose from these three types by setting bit 0 (CSU) of parameter No. 6001 and bit 7 (CSC) of parameter No. 6003. However, these three types produce no difference when the tool moves around an inner corner.

Fig.14.4.1 (a) Start-up/Cancel Operation

CSC	CSU	Type	Operation
0	0	Type A	A compensation vector perpendicular to the block after start-up or perpendicular to the block before cancellation is output. 
0	1	Type B	A compensation vector perpendicular to the start-up block or cancellation block, and an intersection vector are output.  G41 Intersection Tool center path Programmed path N2 N1

CSC	CSU	Type	Operation
1	0 1	Type C	When the start-up block or cancellation block specifies no movement, the tool is shifted by the cutter compensation value in the direction perpendicular to the block after start-up or perpendicular to the block before cancellation.  Intersection Tool center path Programmed path Shift N1 N2 N3 When the start-up block or cancellation block specifies movement, type A is selected if CSU is set to 0; type B is selected if CSU is set to 1.

- Reading of input commands after the start of cutter compensation

After the start of cutter compensation, three blocks are read by default until a cancellation command is specified, regardless of whether the blocks specify movement; when parameter No. 6009 is set, up to eight input command blocks are read instead of the three blocks to be read by default. Thus, an intersection calculation and interference check described later are made.
To perform an intersection calculation, at least two blocks specifying movement need to be read. To make an interference check, at least three blocks specifying movement need to be read.

As more blocks are read by setting parameter No. 6009, an overcutting (interference) forecast can be made for more forward commands. In this case, however, more time is required for reading and analysis because more blocks need to be read and analyzed.

- Cancellation of cutter compensation

Cutter compensation mode is cancelled when a block that satisfies any of the following conditions is executed:
(1) G40 is specified.
(2) D00 is specified as a cutter compensation number.
The circular commands (G02, G03) must not be used when cutter compensation is to be cancelled. Otherwise, an alarm (PS270) is issued.
As with start-up, three types of cancellation operations are supported: type A, type B, and type C. The user can select any of these three types by setting bit 0 (CSU) of parameter No. 6001 and bit 7 (CSC) of parameter No. 6003. However, these three types produce no difference when the tool moves around an inner corner.

- Symbols used in the figures

The symbols used in the figures of Section II-14.4.2 and later have the following meanings:

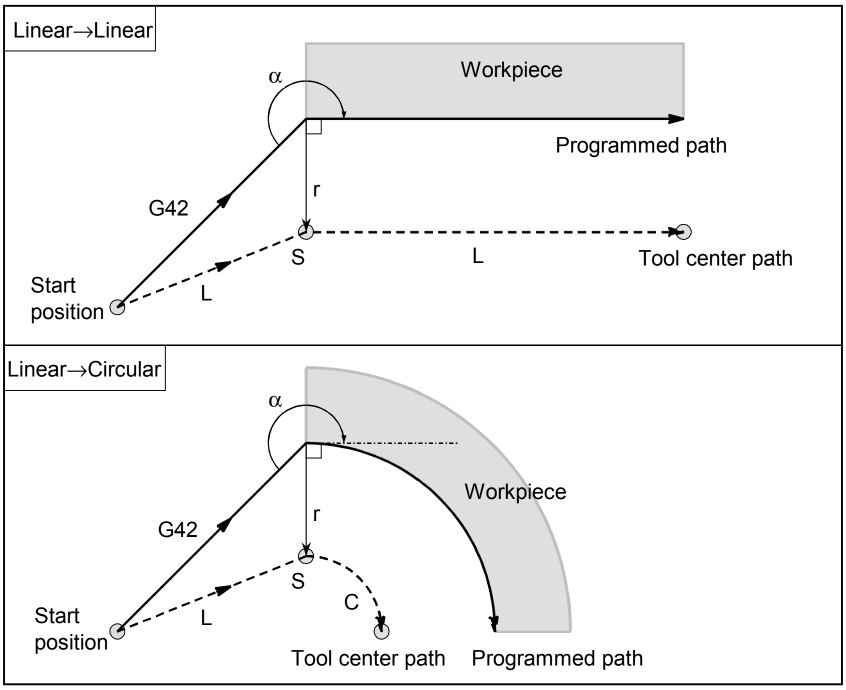
- S represents a point where single block operation is performed once.
- SS represents a point where single block operation is performed twice.
- SSS represents a point where single block operation is performed three times.
- L means that the tool moves linearly.
- C means that the tool moves circularly.
- r represents a cutter compensation value.
- Intersection is the intersection of the programmed paths of two blocks shifted by r.
- ○ represents the center of the tool.

14.4.2 Tool Movement in Start-up

When the offset cancel mode is changed to offset mode, the tool moves as illustrated below (start-up):

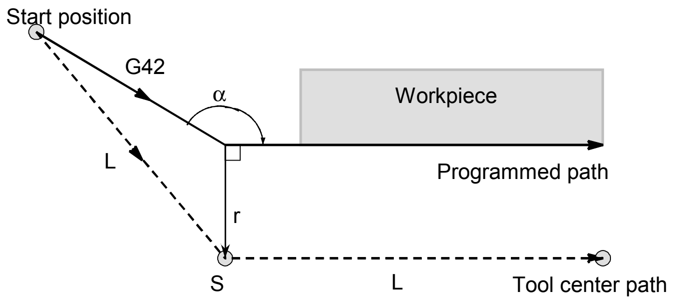
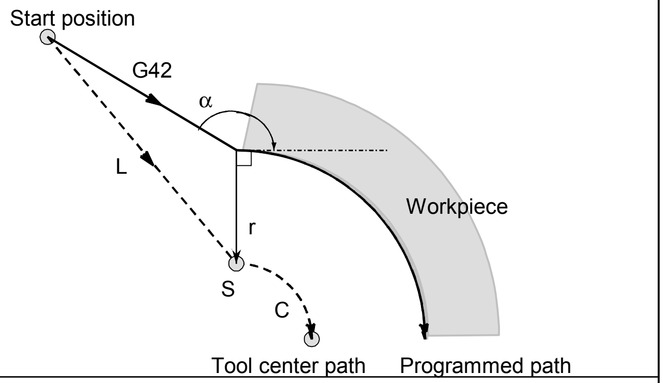
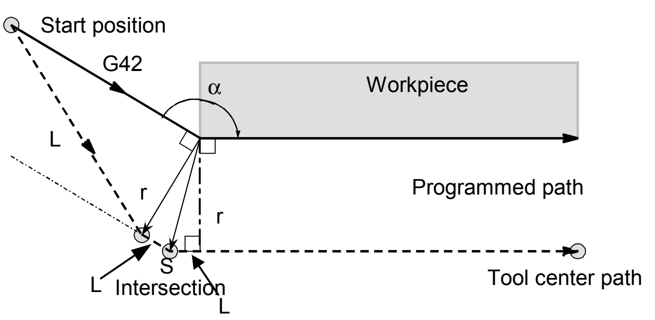
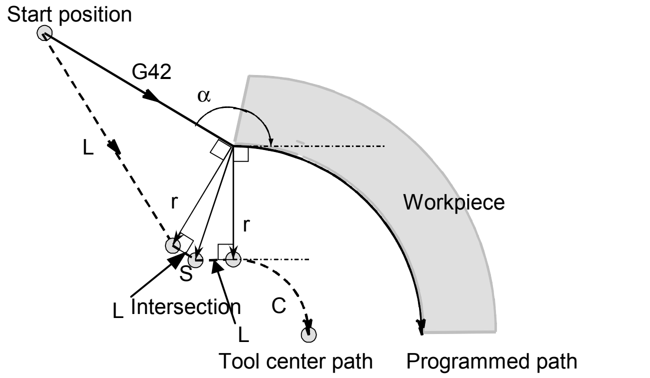
Explanation

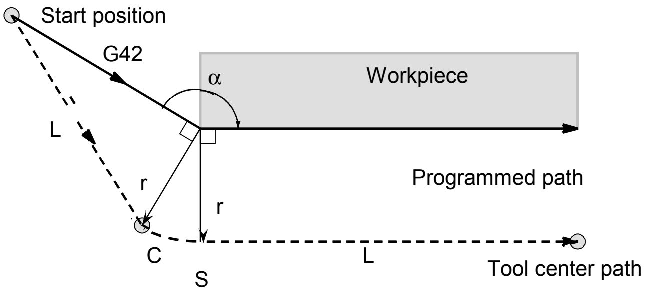
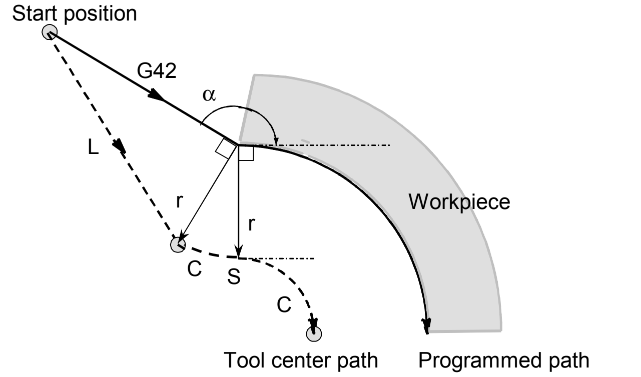
- Tool movement around an inner side of a corner(180deg.≤α)



- When a start-up block involves an outer and obtuse movement(90deg. ≤α<180deg.)

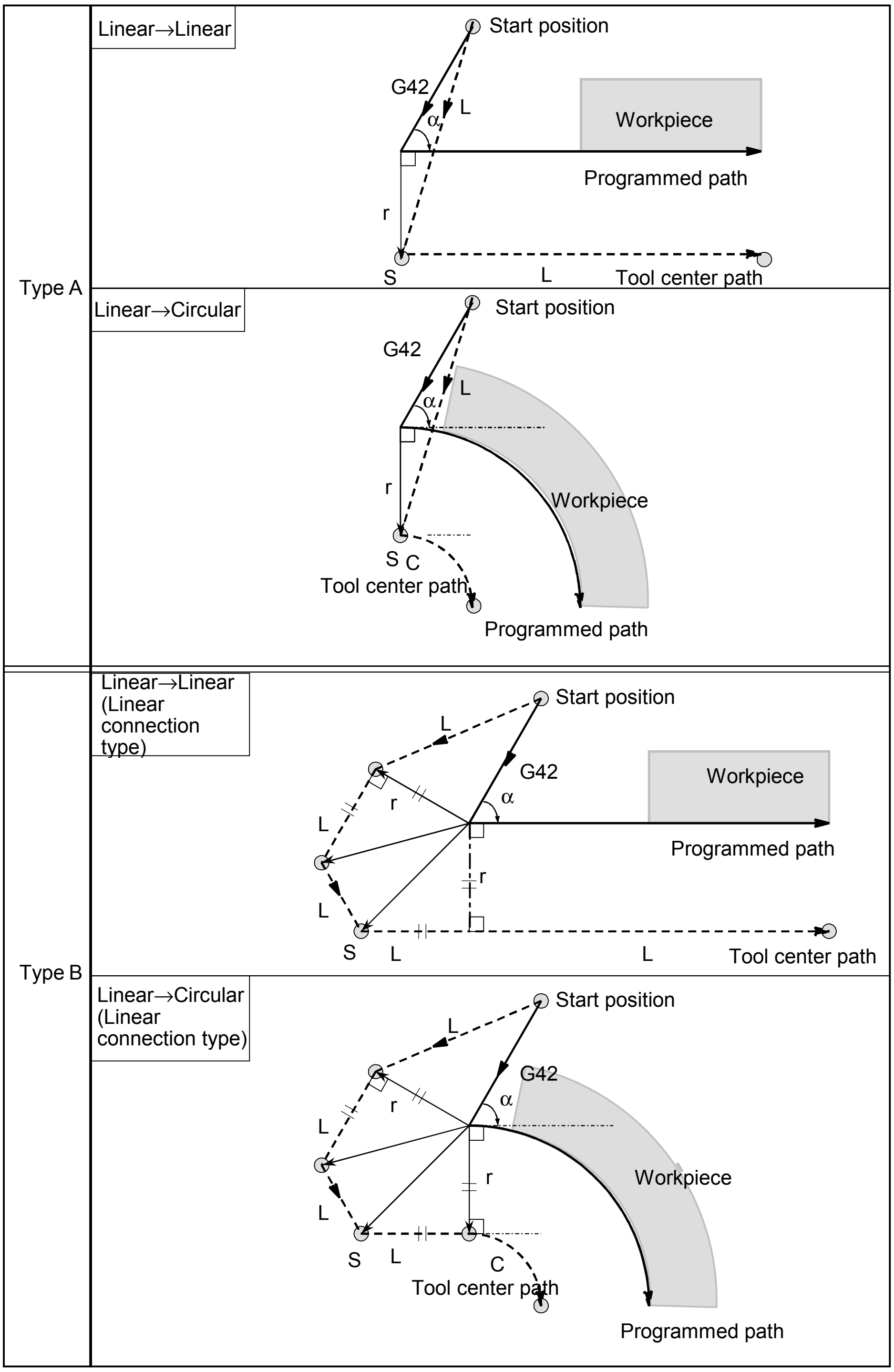
Tool path in start-up has two types A and B, and they are selected by parameter CSU (No. 6001#0).

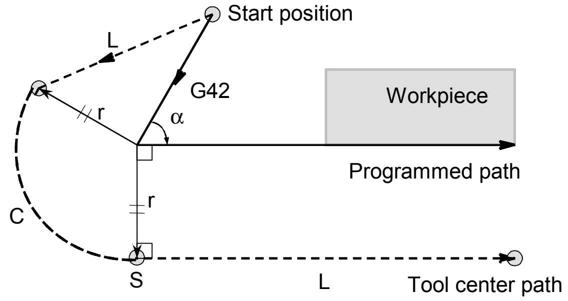
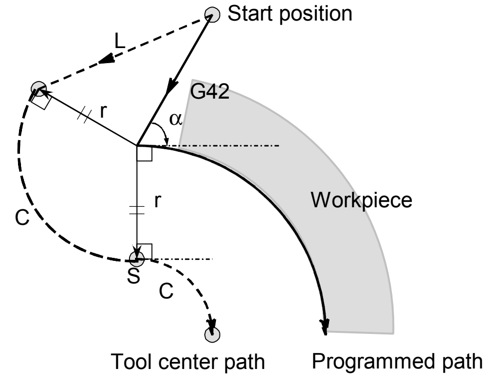
Type A	Linear→Linear	 Start position G42 α Workpiece L Programmed path r S L Tool center path
	Linear→ Circular	 Start position G42 α L Workpiece r S C Tool center path Programmed path
Type B	Linear→Linear (Linear connection type)	 Start position G42 α Workpiece L Programmed path r r S Tool center path L Intersection L
	Linear→Circular (Linear connection type)	 Start position G42 α L Workpiece r r S L Intersection C L Tool center path Programmed path

Type B	Linear→Linear (Circular connection type)	 Start position G42 α Workpiece L r r Programmed path C S L Tool center path
	Linear→Circular (Circular connection type)	 Start position G42 α L r Workpiece r C S C Tool center path Programmed path

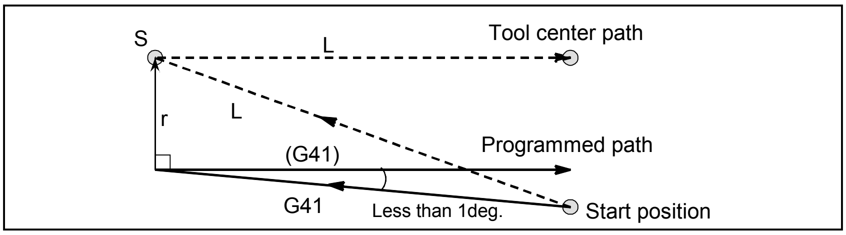
- When a start-up block involves an outer and acute movement(α<90deg.)

Tool path in start-up has two types A and B, and they are selected by parameter CSU (No.6001#0).



Type B	Linear→Linear (Circular connection type)	 Start position L G42 α r Workpiece Programmed path r C S L Tool center path
	Linear→Circular (Circular connection type)	 Start position L G42 α r r Workpiece C S C Tool center path Programmed path

- Tool movement around the outside linear → linear at an acute angle less than 1 degree (α<1deg.)

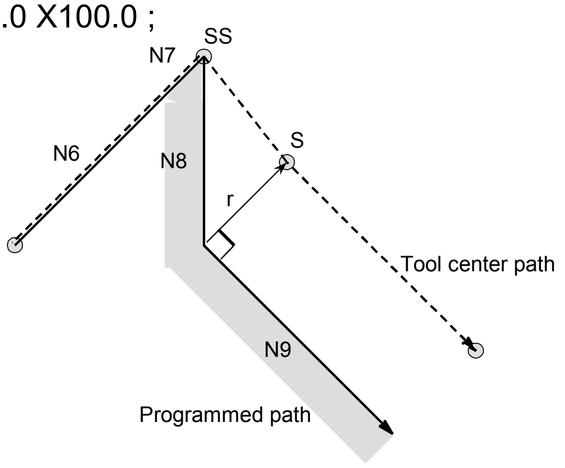


- A block without tool movement specified at start-up

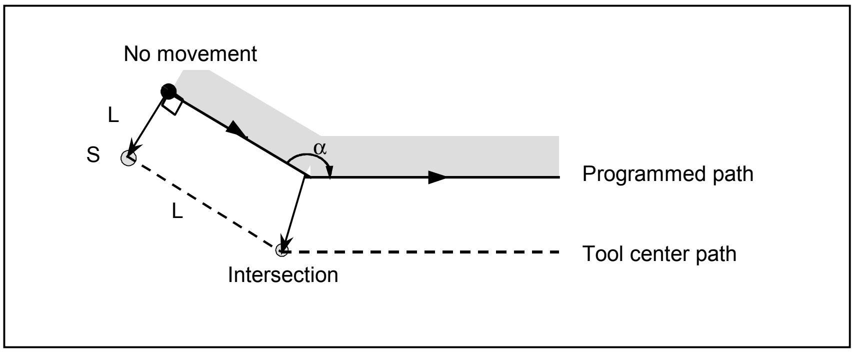
When type A or type B is selected

No offset vector is generated. This means that a start-up block performs no operation. In the next block involving movement, start-up is performed according to the selected type.

```
(Type A)
     G91 G40… ;

     N6 X100.0 Y100.0 ;
     N7 G41 X0 ;
     N8 Y-100.0 ;
     N9 Y-100.0 X100.0 ;
```



When type C is selected
The programmed path is shifted by an offset, perpendicularly from the block specifying movement after start-up.



14.4.3 Tool Movement in the Offset Mode

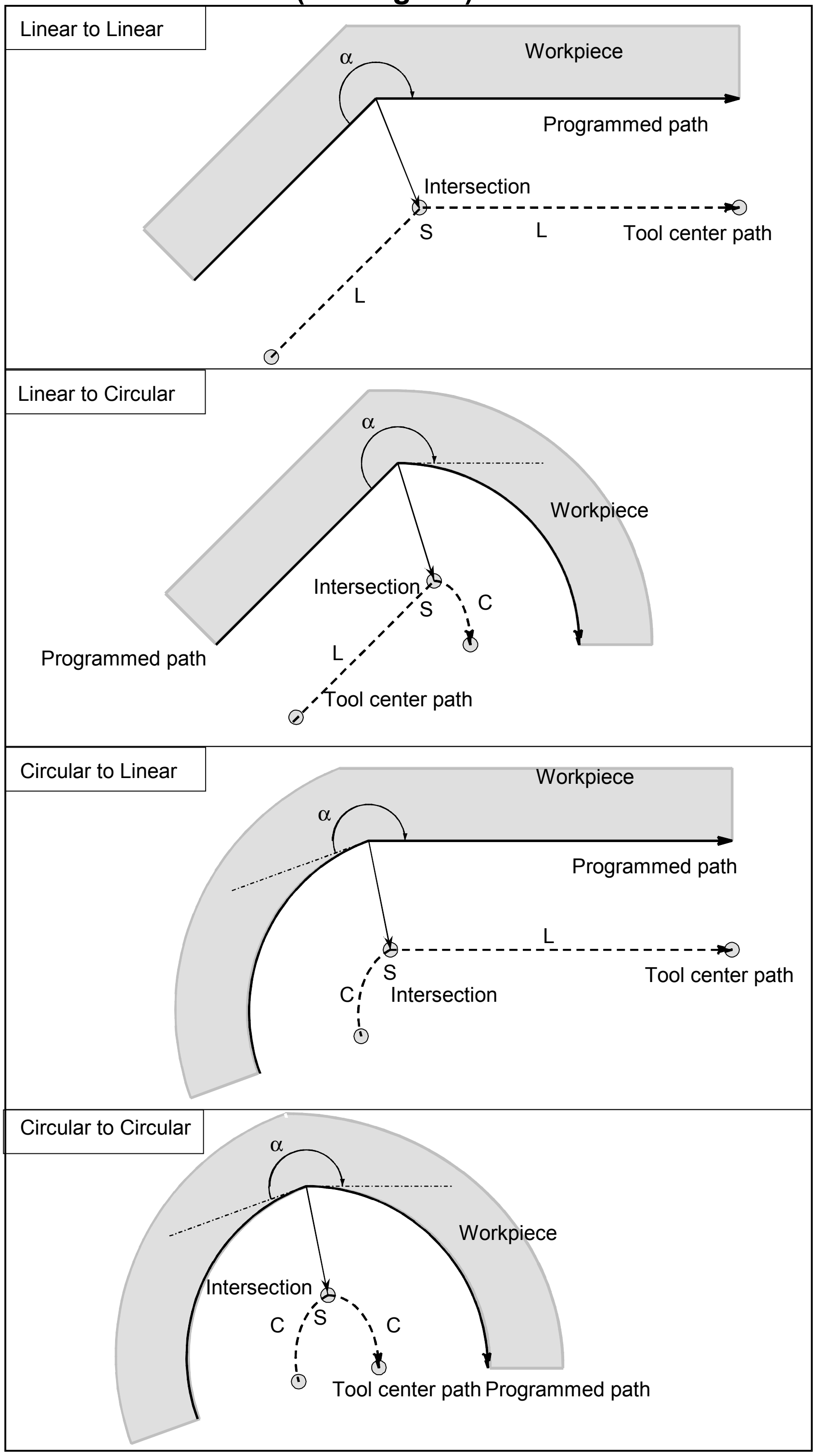
In offset mode, compensation is carried out for positioning commands as well as for linear and circular interpolation commands. To perform intersection calculation, it is necessary to read at least two blocks that involve tool movement. If more than one block that specifies tool movement cannot be read because of blocks not involving tool movement (such as independently specified auxiliary functions and dwell) being specified continuously, intersection calculation is disabled, resulting in either overcutting or insufficient cutting. Assuming that N and M are the number of blocks that are read in offset mode (determined by parameter No. 6009) and the number of blocks (among the N blocks) not specifying tool movement, respectively, the condition for enabling intersection calculation is (N-2) ≥ M. If the maximum number of blocks that are read in offset mode is five, for example, intersection calculation is possible even when up to three blocks do not specify tool movement.

NOTE

The requirement for interference check (described later) is not the same as the above condition. See Section 14.4.6, "Interference check" for details.

If a G or M code, in which buffering is suppressed, is specified in a block, it is impossible to pre-read any command before the block with the G or M code is executed, regardless of the setting of parameter 6009. In this case, therefore, note that overcutting or insufficient cutting may occur because of intersection calculation being impossible.

- Tool movement around the inside of a corner (180deg. ≤α)



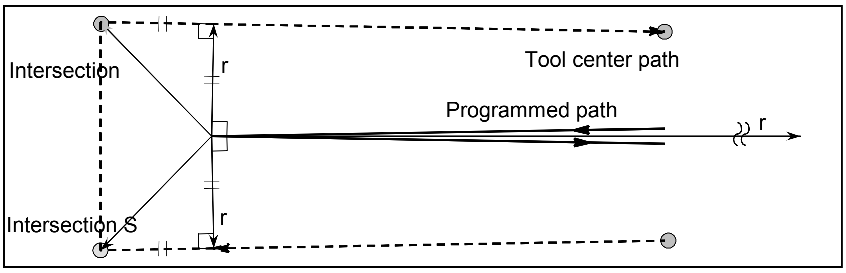
Linear to Linear

Linear to Circular

Circular to Linear

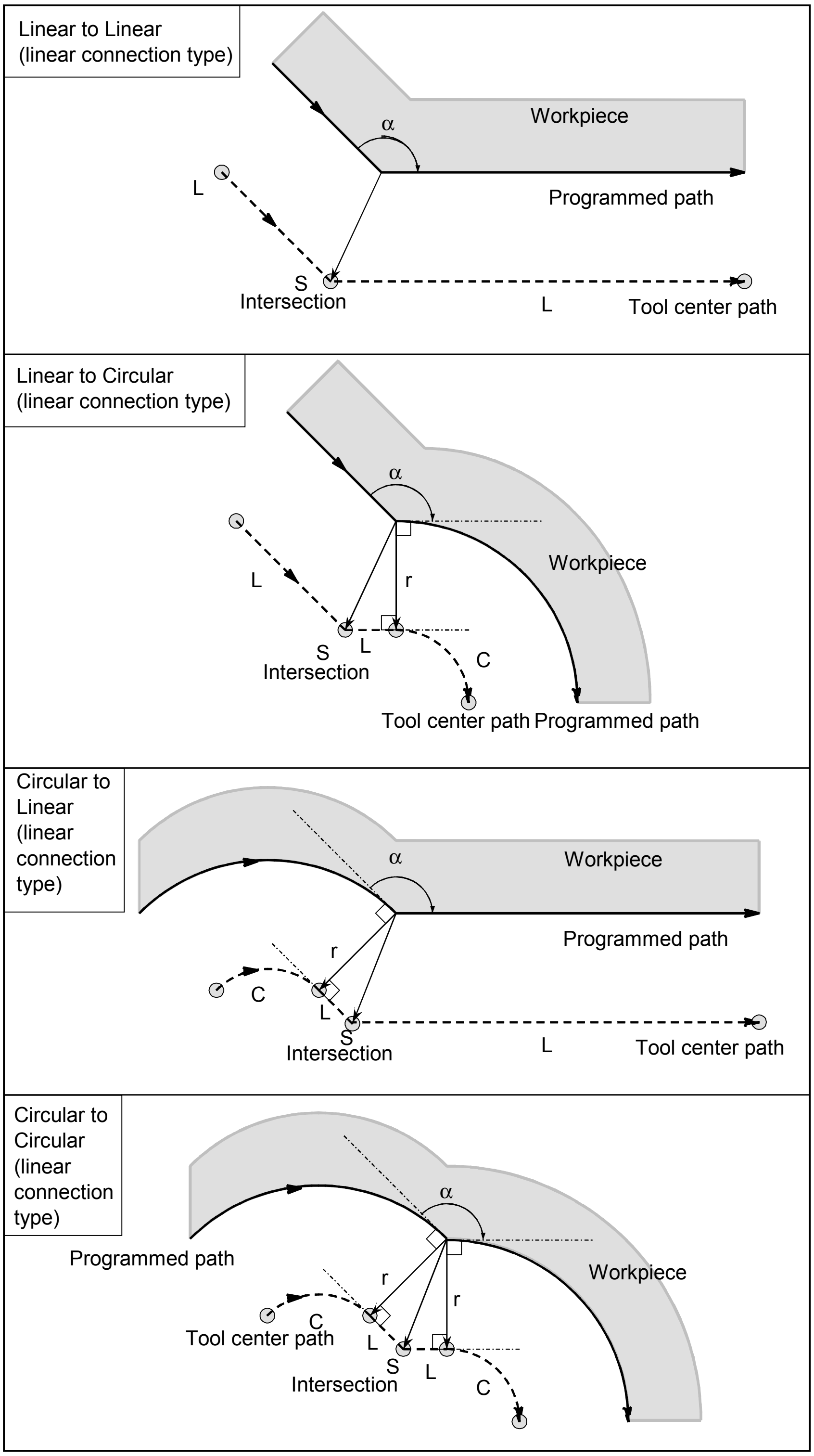
Circular to Circular

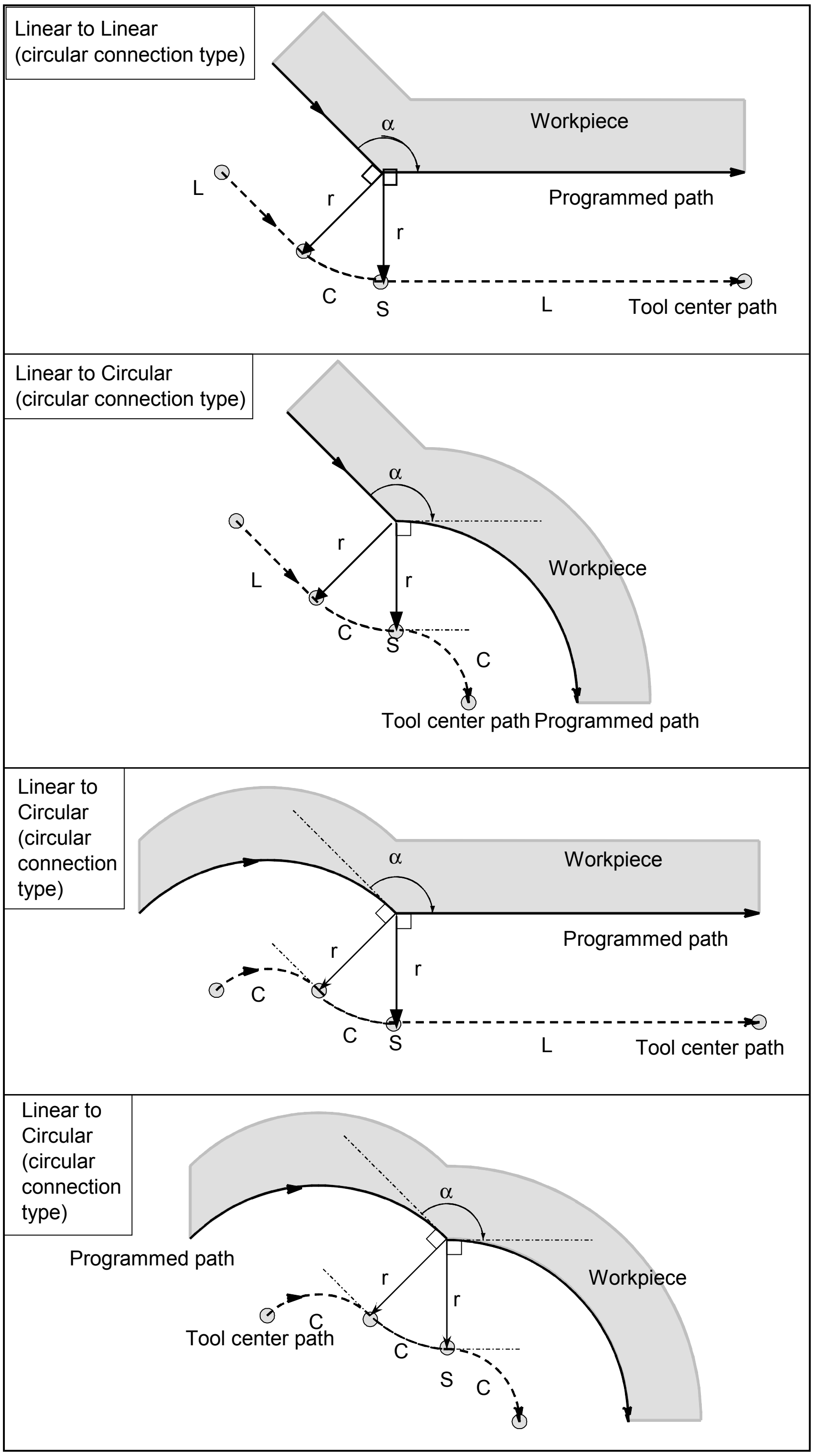
- Tool movement around the inside (α<1deg.) with an abnormally long vector, linear to linear



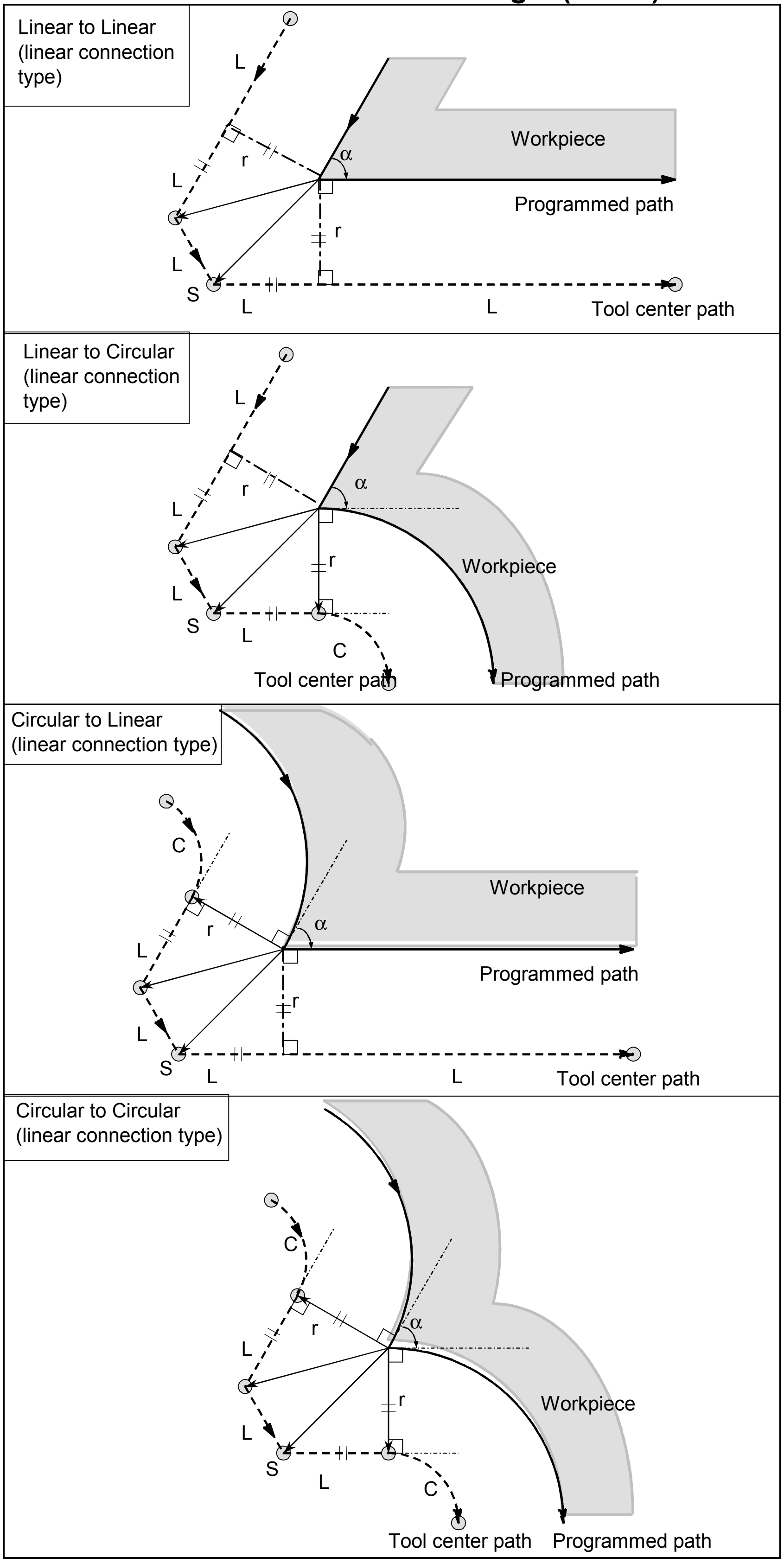
Also in case of arc to straight line, straight line to arc and arc to arc, the reader should infer in the same procedure.

- Tool movement around the outside corner at an obtuse angle (90° ≤ α < 180°)




Linear to Linear
(circular connection type)
α
Workpiece
L
r
Programmed path
r
C
S
L
Tool center path
Linear to Circular
(circular connection type)
α
r
Workpiece
L
r
C
S
C
Tool center path
Programmed path
Linear to
Circular
(circular
connection
type)
α
Workpiece
Programmed path
r
r
C
C
S
L
Tool center path
Linear to
Circular
(circular
connection
type)
α
Programmed path
r
Workpiece
r
C
Tool center path
C
S
C

- Tool movement around the outside corner at an acute angle (α<90°)

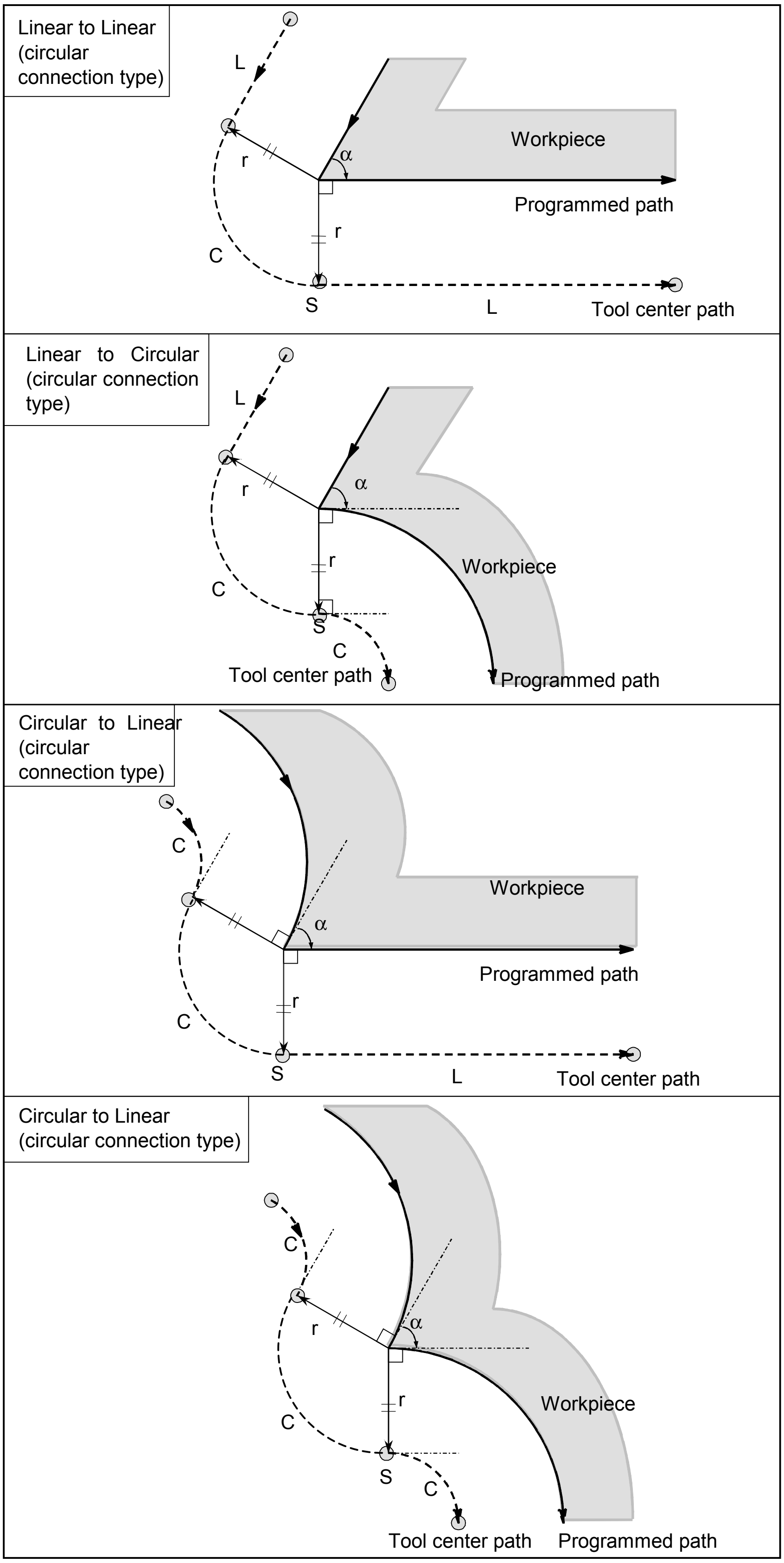


Linear to Linear (linear connection type)

Linear to Circular (linear connection type)

Circular to Linear (linear connection type)

Circular to Circular (linear connection type)

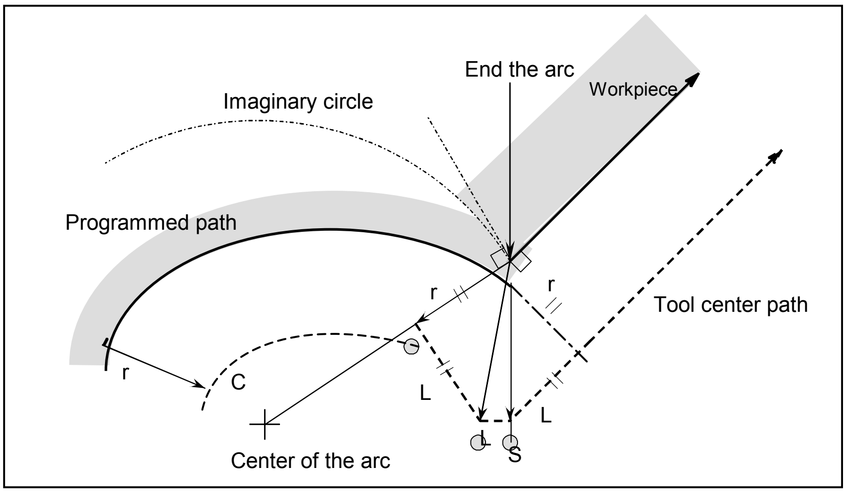

Linear to Linear (circular connection type)
L
r
α
Workpiece
Programmed path
r
C
S
L
Tool center path
Linear to Circular (circular connection type)
L
r
α
r
Workpiece
C
S
C
Tool center path
Programmed path
Circular to Linear (circular connection type)
C
Workpiece
α
Programmed path
r
C
S
L
Tool center path
Circular to Linear (circular connection type)
C
α
r
r
Workpiece
C
S
C
Tool center path
Programmed path

- When it is exceptional

End position for the arc is not on the arc

If the end of a line leading to an arc is programmed as the end of the arc by mistake as illustrated below, the system assumes that cutter compensation has been executed with respect to an imaginary circle that has the same center as the arc and passes the specified end position. Based on this assumption, the system creates a vector and carries out compensation. The resulting tool center path is different from that created by applying cutter compensation to the programmed path in which the line leading to the arc is considered straight.

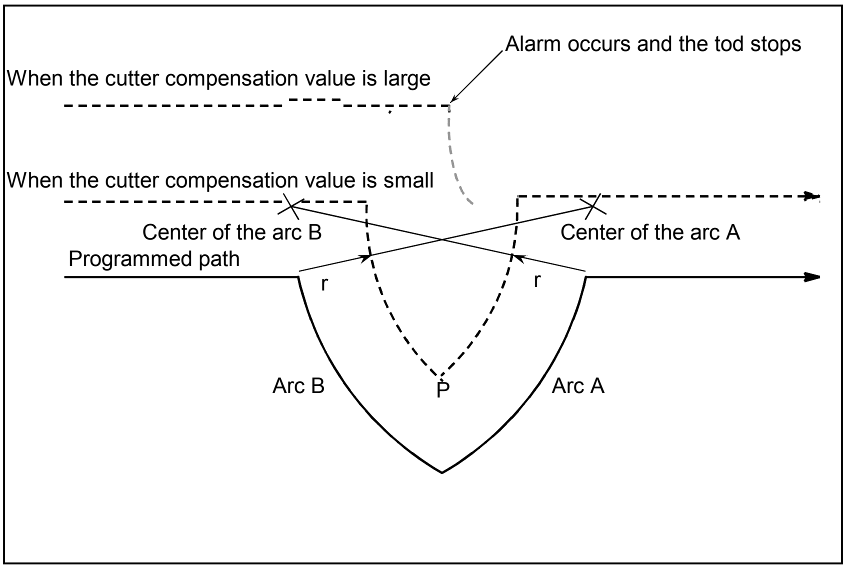
The same description applies to tool movement between two circular paths.



There is no inner intersection

If the cutter compensation value is sufficiently small, the two circular tool center paths made after compensation intersect at a position (P). Intersection P may not occur if an excessively large value is specified for cutter compensation.

When this is predicted, P/S alarm No0276 occurs at the end of the previous block and the tool is stopped. In the example shown below, tool center paths along arcs A and B intersect at P when a sufficiently small value is specified for cutter compensation. If an excessively large value is specified, this intersection does not occur.


Alarm occurs and the tod stops
When the cutter compensation value is large
When the cutter compensation value is small
Center of the arc B
Center of the arc A
Programmed path
r
r
Arc B
P
Arc A

- Change in the offset direction in the offset mode

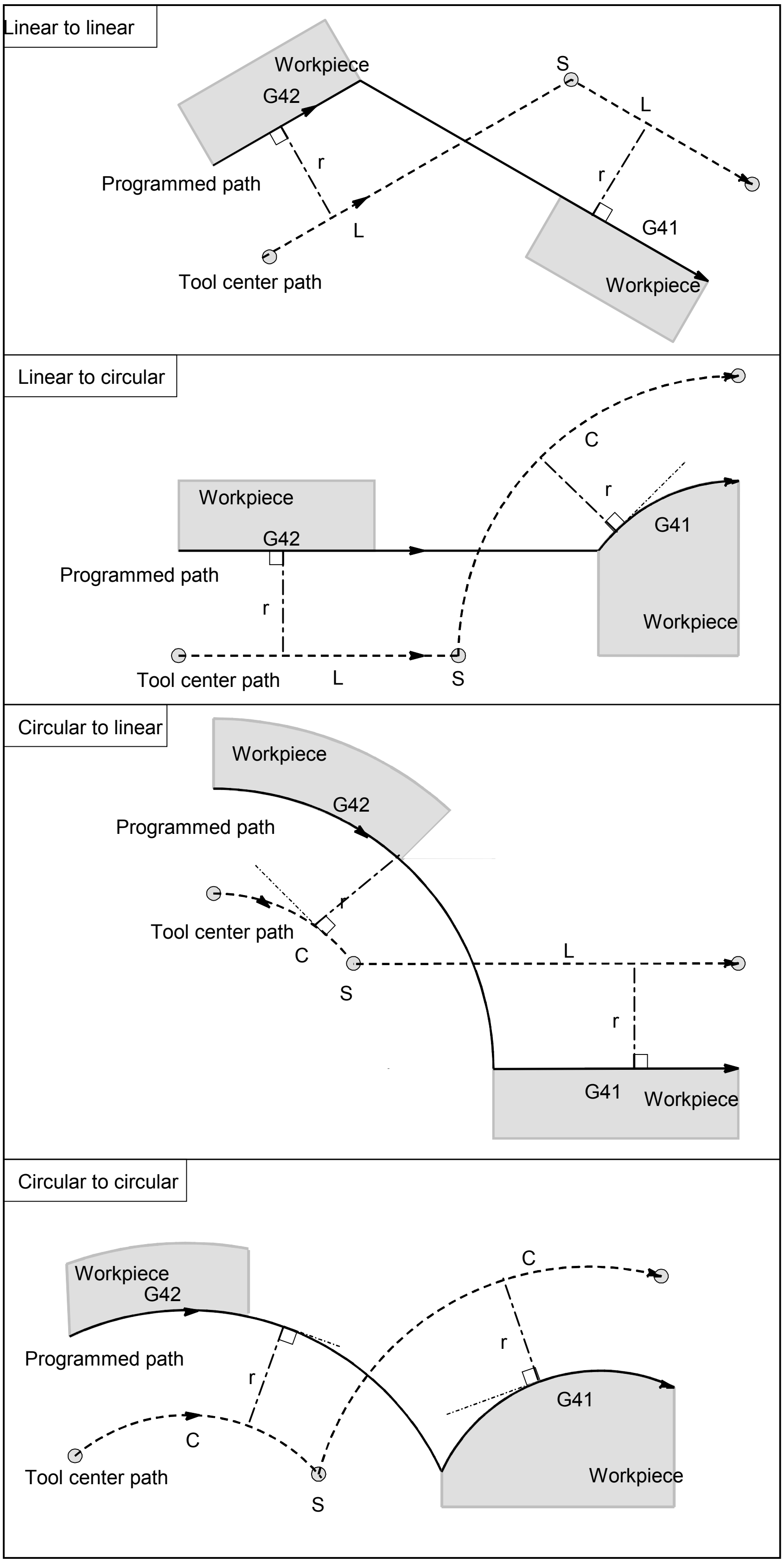
The offset direction is decided by G codes (G41 and G42) for cutter radius

and the sign of cutter compensation value as follows.

Sign of offset amount / G code	+	-
G41	Left side offset	Right side offset
G42	Right side offset	Left side offset

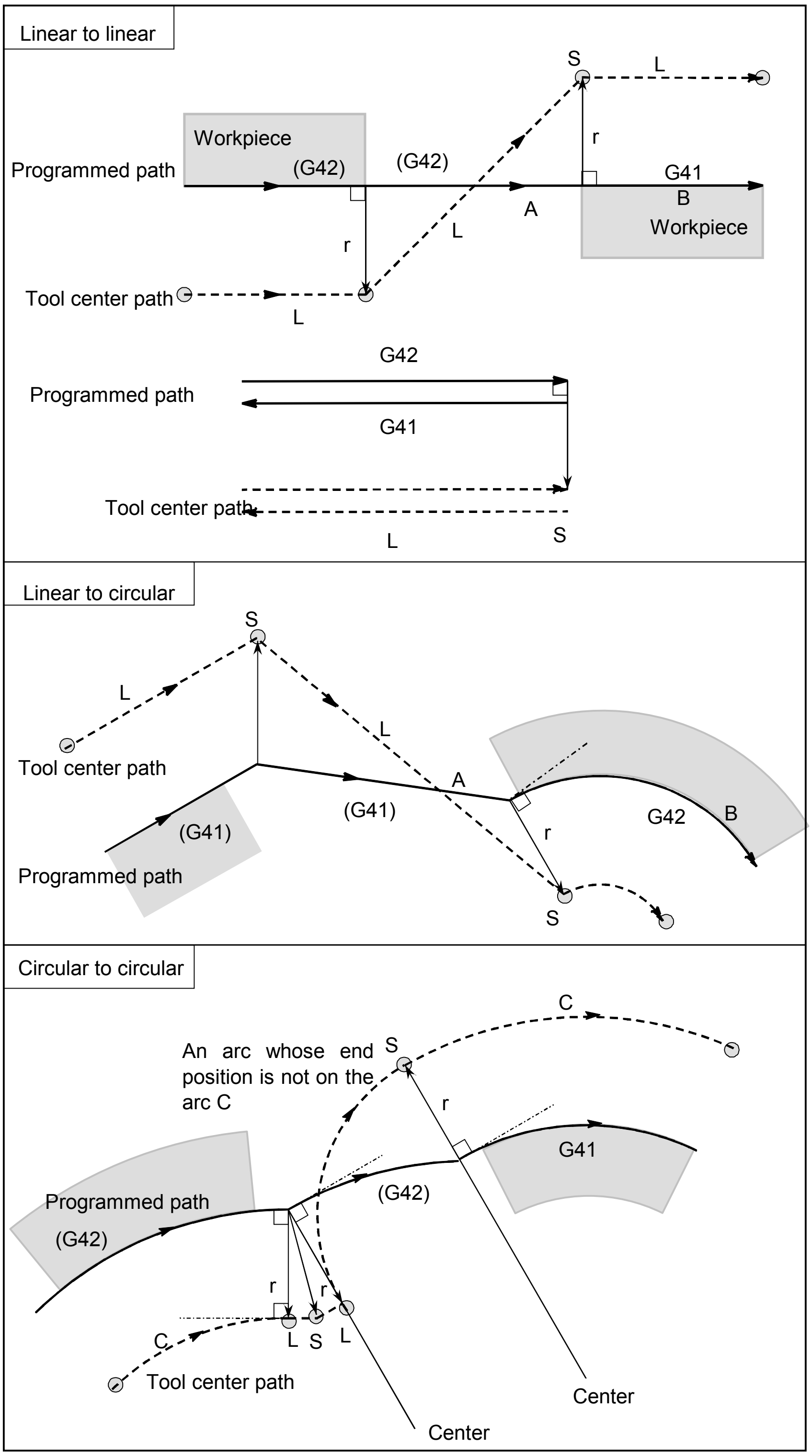
The offset direction can be changed in the offset mode. If the offset direction is changed in a block, a vector is generated at the intersection of the tool center path of that block and the tool center path of a preceding block. However, the change is not available in the start-up block and the block following it.

Tool center path with an intersection



Tool center path without an intersection

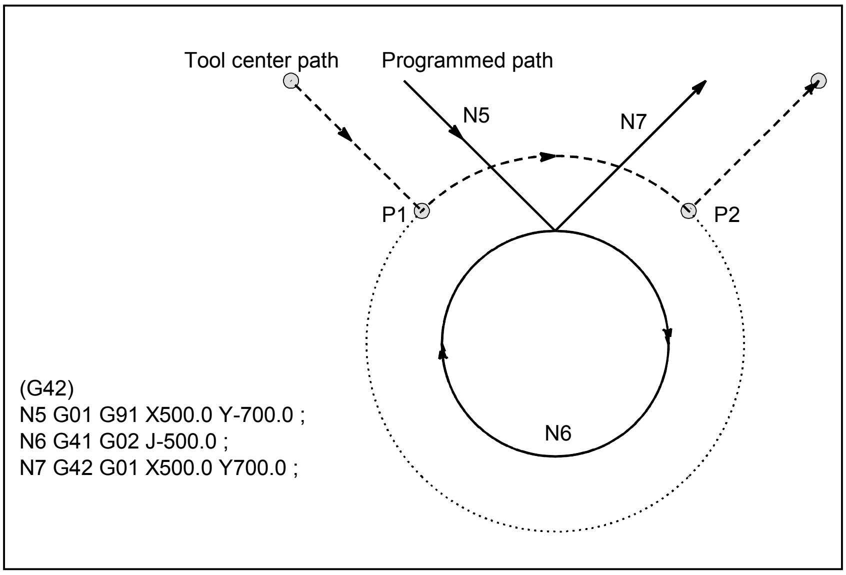
When changing the offset direction in block A to block B using G41 and G42, if intersection with the offset path is not required, the vector normal to block B is created at the start point of block B.



The length of tool center path larger than the circumference of a circle

Normally there is almost no possibility of generating this situation. However, when G41 and G42 are changed, or when a G40 was commanded with address I, J, and K this situation can occur.

In this case of the figure, the cutter compensation is not performed with more than one circle circumference: an arc is formed from P_1 to P_2 as shown. Depending on the circumstances, an alarm may be displayed due to the "Interference Check" described later. To execute a circle with more than one circumference, the circle must be specified in segments.



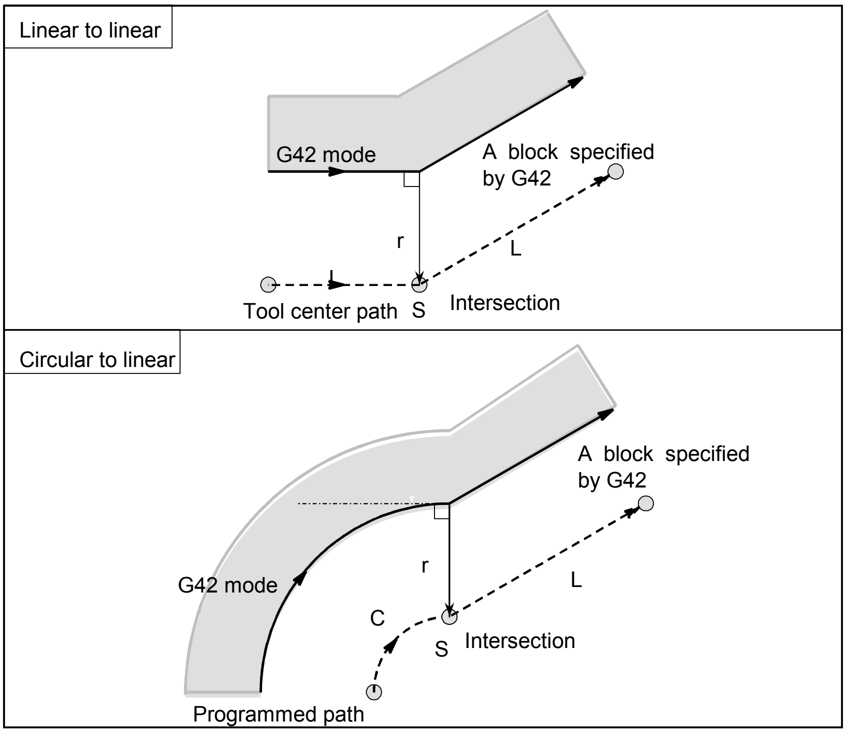
```
(G42)
N5 G01 G91 X500.0 Y-700.0 ;
N6 G41 G02 J-500.0 ;
N7 G42 G01 X500.0 Y700.0 ;
```

- Cutter compensation G code in the offset mode

The offset vector can be set to form a right angle to the moving direction in the previous block, irrespective of machining inner or outer side, by commanding the cutter compensation G code (G41, G42) in the offset mode, independently.

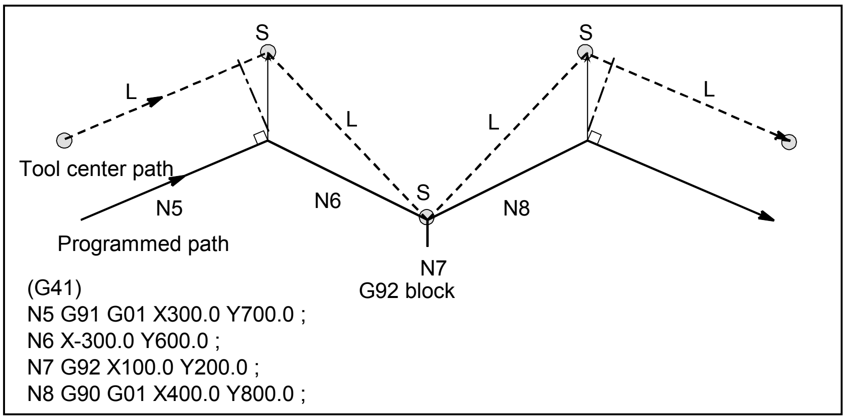
If this code is specified in a circular command, correct circular motion will not be obtained.

When the direction of offset is expected to be changed by the command of cutter compensation G code (G41, G42), refer to the Item "Change in the offset direction in the offset mode" in this Subsection.



- Command canceling the offset vector temporarily

During offset mode, if G92 (absolute zero point programming) is commanded, the offset vector is temporarily cancelled and thereafter offset mode is automatically restored.
In this case, without movement of offset cancel, the tool moves directly from the intersecting point to the commanded point where offset vector is canceled.
Also when restored to offset mode, the tool moves directly to the intersecting point.



```
(G41)
N5 G91 G01 X300.0 Y700.0 ;
N6 X-300.0 Y600.0 ;
N7 G92 X100.0 Y200.0 ;
N8 G90 G01 X400.0 Y800.0 ;
```

Before executing G28 (reference position return), G29 (return from reference position), G30 (second, third, or fourth reference position return), G30.1 (floating reference position return), or G53 (machine coordinate system selection), cancel offset mode by issuing G40. If any of these commands is issued in offset mode, the offset vector will disappear temporarily.

- If I, J, and K are specified in G00/G01 mode block

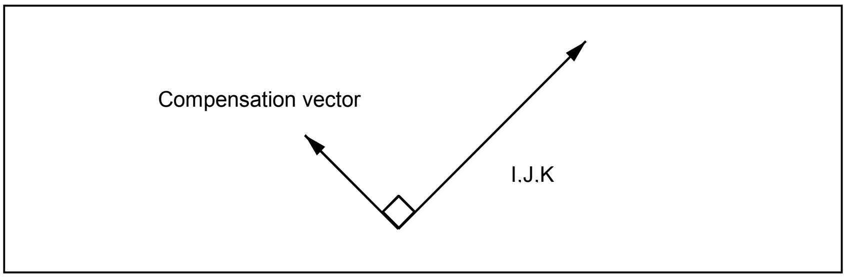
When cutter compensation begins or is already being applied, specifying I, J, and K in a block specifying positioning mode (G00) or linear interpolation mode (G01) can make the compensation vector at the end point of the block vertical to the direction specified by I, J, and K. In this way, it is possible to change the direction in which compensation is applied.

IJ type vector (XY plane)

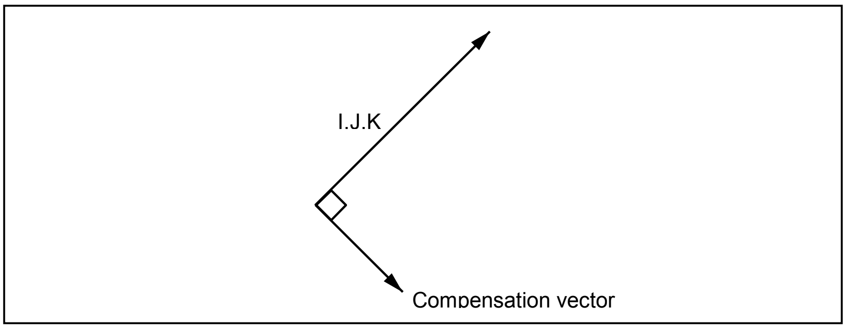
This section explains the compensation vector (IJ-type vector) for which the compensation plane is generated in the XY plane (G17 mode). (This explanation also applies to the KI-type vector in the G18 plane and the JK-type vector in the G19 plane.) The compensation vector for the IJ-type vector is determined as being vertical to the direction specified by I and J and as large as the amount of offset without performing intersection calculation for a programmed path, as shown below. I and J specification is possible when cutter compensation begins, as well as when it is already being applied. If I and J are specified when compensation begins, any parameter-specified startup type becomes invalid, resulting in the IJ-type vector being produced.

Offset vector direction

In G41 mode, a virtual tool movement direction is assumed to be one specified by I, J, and K, and an offset vector perpendicular to the virtual direction is produced toward the left of the virtual direction.



In G42 mode, a virtual tool movement direction is assumed to be that specified by I, J, and K, and an offset vector perpendicular to the virtual direction is produced toward the right of the virtual direction.

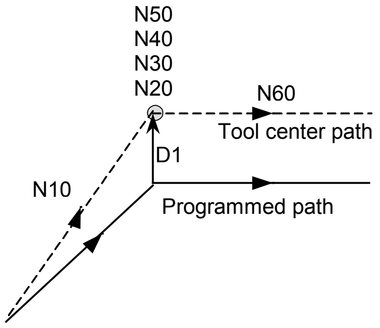


Example

If I and J are specified in a block involving tool movement when compensation begins

```
  (G40)
N10 G91 G41 X100.0 Y100.0
   I1 D1 ;
N20 G04 X1000 ;
N30 G01 F1000 ;
N40 S300 ;
N50 M50 ;
N60 X150. ;
```

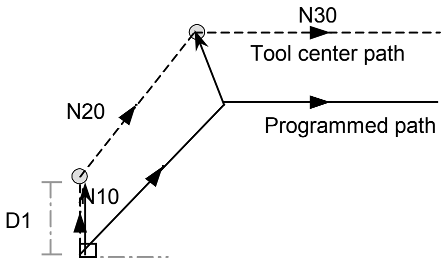
Note) In N10, I1 specifies a vector that is perpendicular to the X-axis and as large as D1.



If I and J are specified in a block not involving tool movement when compensation begins

```
  (G40)
N10 G41 I1 D1 ;
N20 G91 X100. Y100. ;
N30 X150. ;
```

Note) In N10, I1 specifies a vector that is perpendicular to the X-axis and as large as D1.

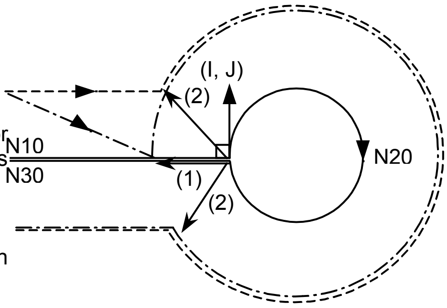


If I and J are specified in a block involving tool movement in compensation mode

```
  (G17 G41 G91 D1)
N10 G00 X150. J50. ;
N20 G02 I150. ;
N30 G00 X-150. ;
```

Note) In N10, J50 specifies a vector that is perpendicular to the Y-axis and as large as D1.

(1) IJ type vector
(2) Vector produced by intersection calculation



—·—·—·— Tool center path

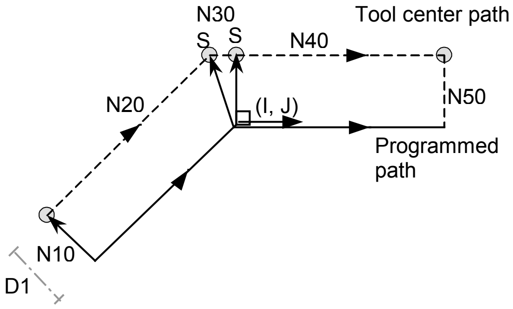
———— Programmed path

- - - - - - Path produced by intersection calculation (if I and J are not specified)

If I and J are specified in a block not involving tool movement during the compensation mode

Startup/cancel type C

```
N10 G41 D1 G01 F1000 ;
N20 G91 X100. Y100. ;
N30 I10. ;
N40 X150. ;
N50 G40 ;
```

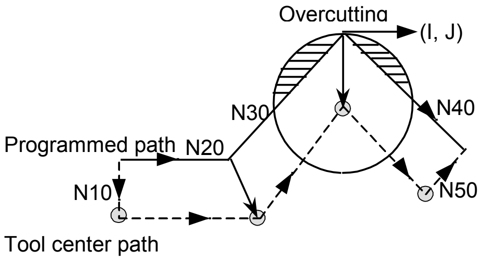


Restrictions

If the IJ-type vector is specified, it may cause tool interference depending on its direction vector even if no additional vector is specified. In this case, a tool interference alarm is not issued, nor is an attempt made to avoid tool interference, possibly resulting in overcutting.

Startup/cancel type C

```
N10 G42 D1 F1000 ;
N20 G91 X100. ;
N30 X100. Y100. I10. ;
N40 X100. Y-100. ;
N50 G40 ;
```



- A block without tool movement

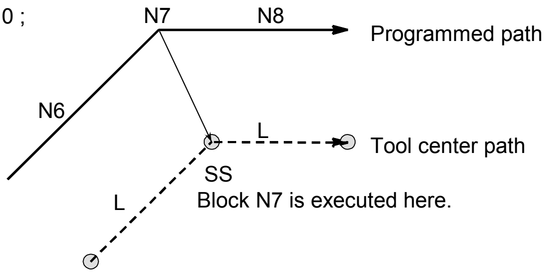
The following blocks have no tool movement. In these blocks, the tool will not move even if cutter compensation is effected.

M05 ;	: M code output
S21 ;	: S code output
G04 X10.0 ;	: Dwell
G22 X100000 ;	: Machining area setting
G10 L11 P01 R10.0 ;	: Cutter compensation value setting/change
(G17) Z200.0 ;	: Move command not included in the offset plane.
G90 ;, O10 ;, N20 ;	: G, O, or N code only
G91 X0 ;	: Move distance is zero

A block without tool movement specified in offset mode

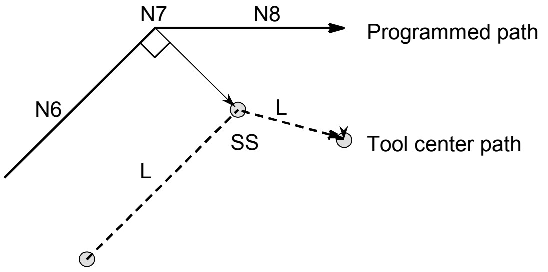
Unless more than N-2 blocks not involving tool movement (where N is the number of blocks read in offset mode and specified by parameter No. 6009) are specified continuously, the vector and tool center path remain unchanged, so that these blocks are executed at the single-block stop point.

```
N6 G91 X100.0 Y100.0 ;
N7 G04 X10.0 ;
N8 X100.0 ;
```



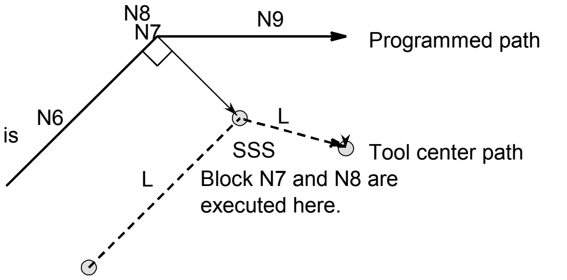
If an axis command with a tool movement amount of 0 is specified even in a single block, a vector that is vertical to the movement direction and which has the same offset specified in the previous block is produced. Be careful when using this command, because it may cause overcutting.

```
N6 G91 X100.0 Y100.0 ;
N7 X0 ;
N8 X100.0 ;
```



Do not specify more than N-2 blocks not involving tool movement (where N is the number of blocks read in offset mode and which is specified by parameter No. 6009) continuously in offset mode. If commanded, a vector whose length is equal to the offset value is produced in a normal direction to tool motion in earlier block, so overcutting may result.

```
N6 G91 X100.0 Y100.0 ;
N7 S21 ;
N8 G04 X10.0 ;
N9 X100.0 ;
(Number of reading block is
three in the offset mode.)
```

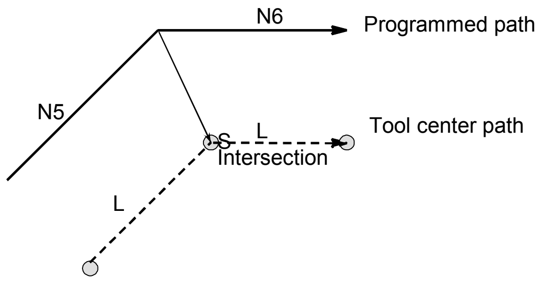


- If an M or G code, for which buffering is suppressed, is specified

If a M or G code, for which buffering is suppressed, is specified in offset mode, it becomes impossible to read and analyze subsequent commands no matter how many blocks parameter (No. 6009) specifies to read in offset mode. Therefore, neither intersection calculation nor interference check (described later) can be performed. In this case, overcutting may occur because a vector perpendicular to the immediately preceding block is output.

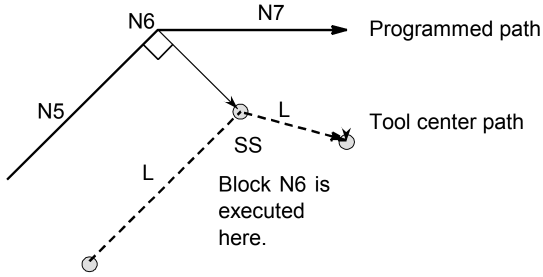
If an M code (M50), in which buffering is suppressed, is not specified

```
  (G42)
 N5 G91 G01 X20.0 Y40.0 ;
 N6 X40.0 ;
      :
      :
```



If an M code (M50), in which buffering is suppressed, is specified

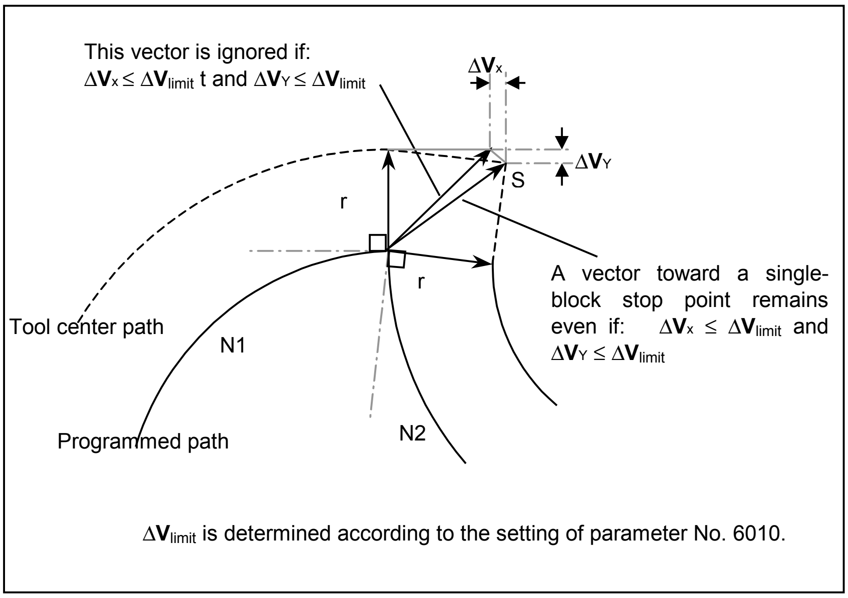
```
  (G42)
 N5 G91 G01 X20.0 Y40.0 ;
 N6 M50 ;
 N7 X40.0 ;
      :
      :
```



- Corner movement

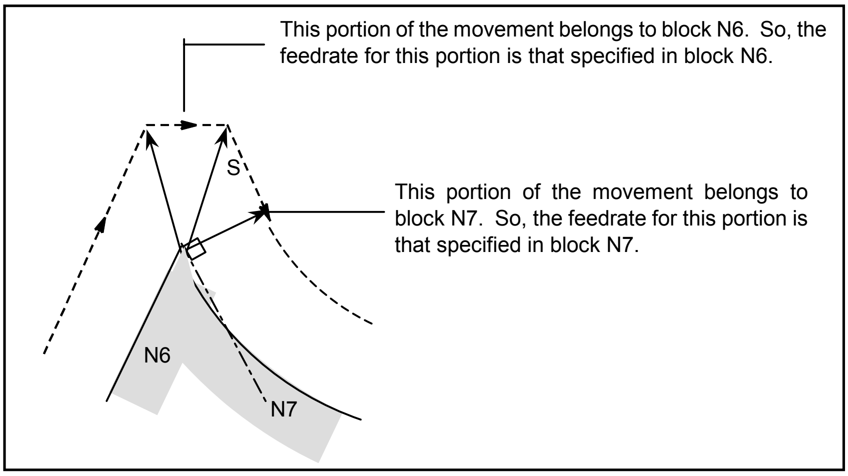
If more than one offset vector is produced at the end point of a block, these vectors are connected using either a straight line or arc, depending on the specification made in parameter CCC (bit 2 of parameter No. 6008). This is called corner movement.

If these vectors are almost equal and the distance of the corner movement between the vectors is judged as being small based on the setting of parameter No. 6010, no corner movement is made. In this case, a vector toward the single-block stop point takes precedence and remains valid, causing the other vectors to be ignored. In this way, it is possible to ignore the minute movements that occur due to cutter compensation and which prevent changes in the cutting rate that may otherwise occur due to interrupted buffering.


This vector is ignored if:
$\Delta \mathbf{V}_X \leq \Delta \mathbf{V}_{limit}$ t and $\Delta \mathbf{V}_Y \leq \Delta \mathbf{V}_{limit}$
$\Delta \mathbf{V}_X$
$\Delta \mathbf{V}_Y$
S
r
r
A vector toward a single-block stop point remains even if: $\Delta \mathbf{V}_X \leq \Delta \mathbf{V}_{limit}$ and $\Delta \mathbf{V}_Y \leq \Delta \mathbf{V}_{limit}$
Tool center path
N1
Programmed path
N2

$\Delta \mathbf{V}_{limit}$ is determined according to the setting of parameter No. 6010.

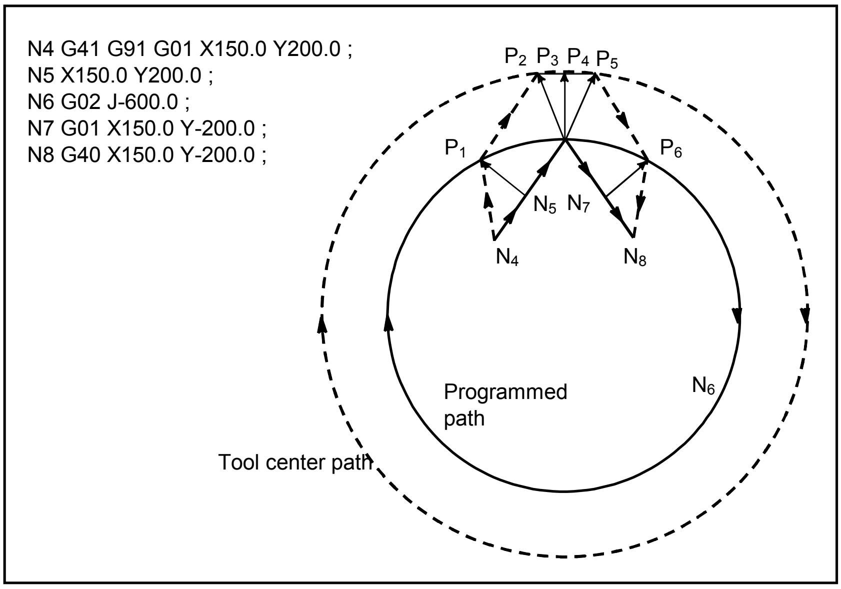
If the vectors are not judged as being almost equal (or cannot be removed), commands for movement around the corner are executed. Tool movement around the corner before the single-block stop point belongs to those blocks before the block for the corner, while tool movement around the corner after the single-block stop point belongs to those blocks after the block for the corner.



This portion of the movement belongs to block N6. So, the feedrate for this portion is that specified in block N6.

This portion of the movement belongs to block N7. So, the feedrate for this portion is that specified in block N7.

However, if the path of the next block is semicircular or more, the above function is not performed.
The reason for this is as follows:



```
N4 G41 G91 G01 X150.0 Y200.0 ;
N5 X150.0 Y200.0 ;
N6 G02 J-600.0 ;
N7 G01 X150.0 Y-200.0 ;
N8 G40 X150.0 Y-200.0 ;
```

If the vector is not ignored, the tool path is as follows:
P_1 -> P_2 -> P_3 -> (Circle) -> P_4 -> P_5 -> P_6 But if the distance between P_2 and P_3 is negligible, the point P_3 is ignored.
Therefore, the tool path is as follows: P_2 -> P_4 Namely, circle cutting by the block N6 is ignored.

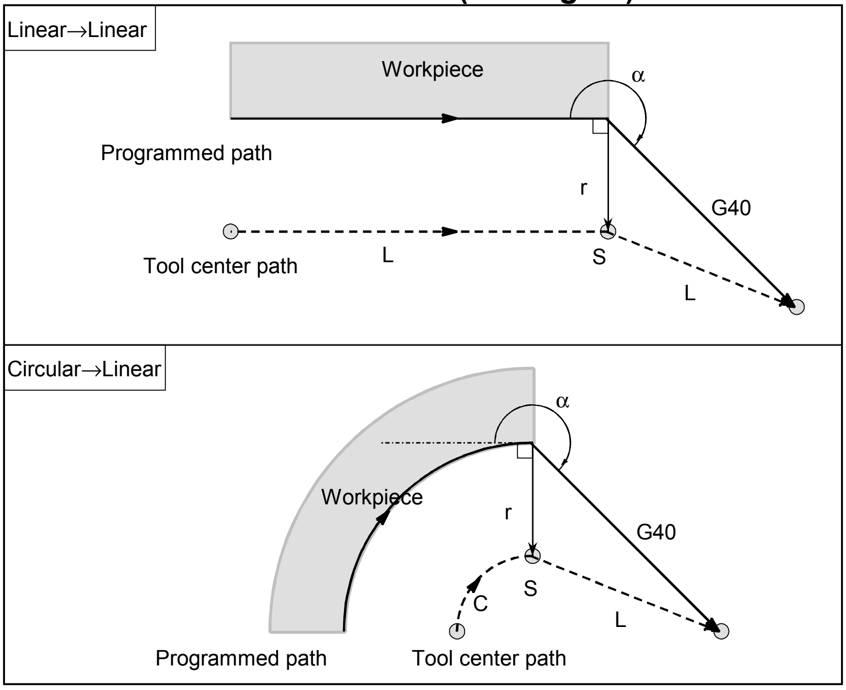
- Interruption of manual operation

See "Manual absolute on/off" for an explanation of the processing performed if manual intervention is applied in offset mode.

14.4.4 Tool Movement in Offset Mode Cancel

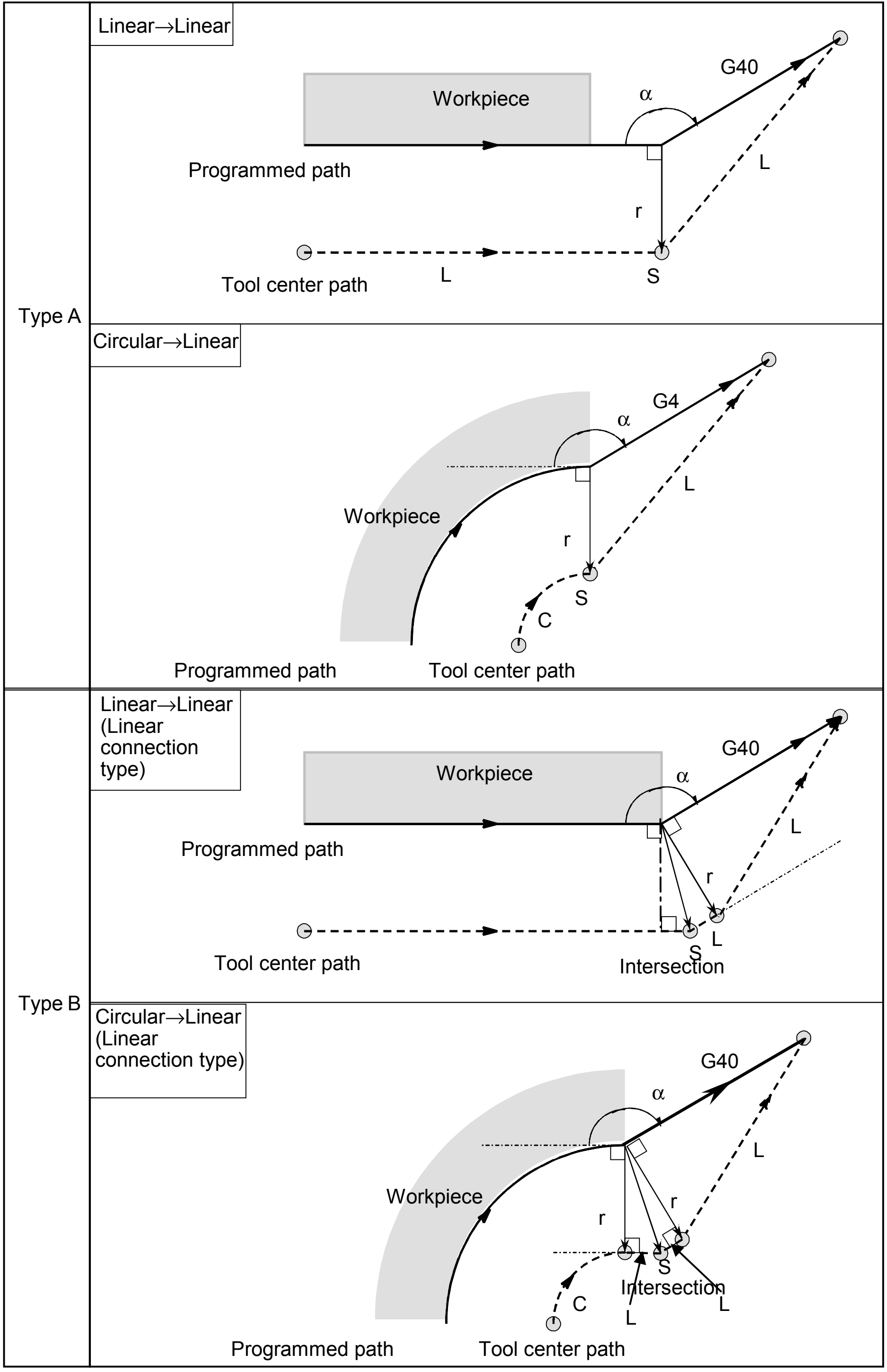
Explanation

- When a cancellation block involves an inner movement (180deg.≤α)



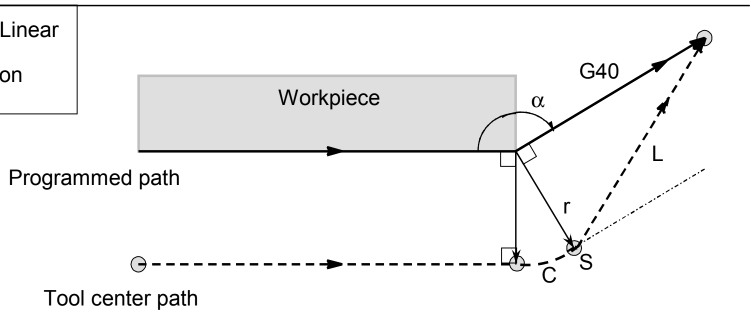
- When a cancellation block involves an outer and obtuse movement (90≤α<180deg.)

Two types are supported: type A and type B. The user can select from the two types by setting bit 0 (CSU) of parameter No. 6001.



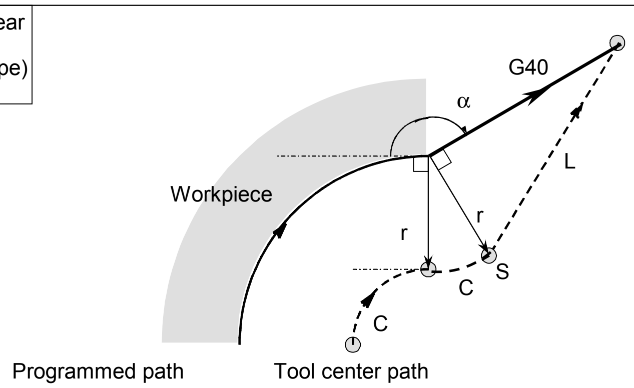
<table>
<tr>
<td rowspan="2">Type B</td>
<td>

Linear→Linear (Circular connection type)



</td>
</tr>
<tr>
<td>

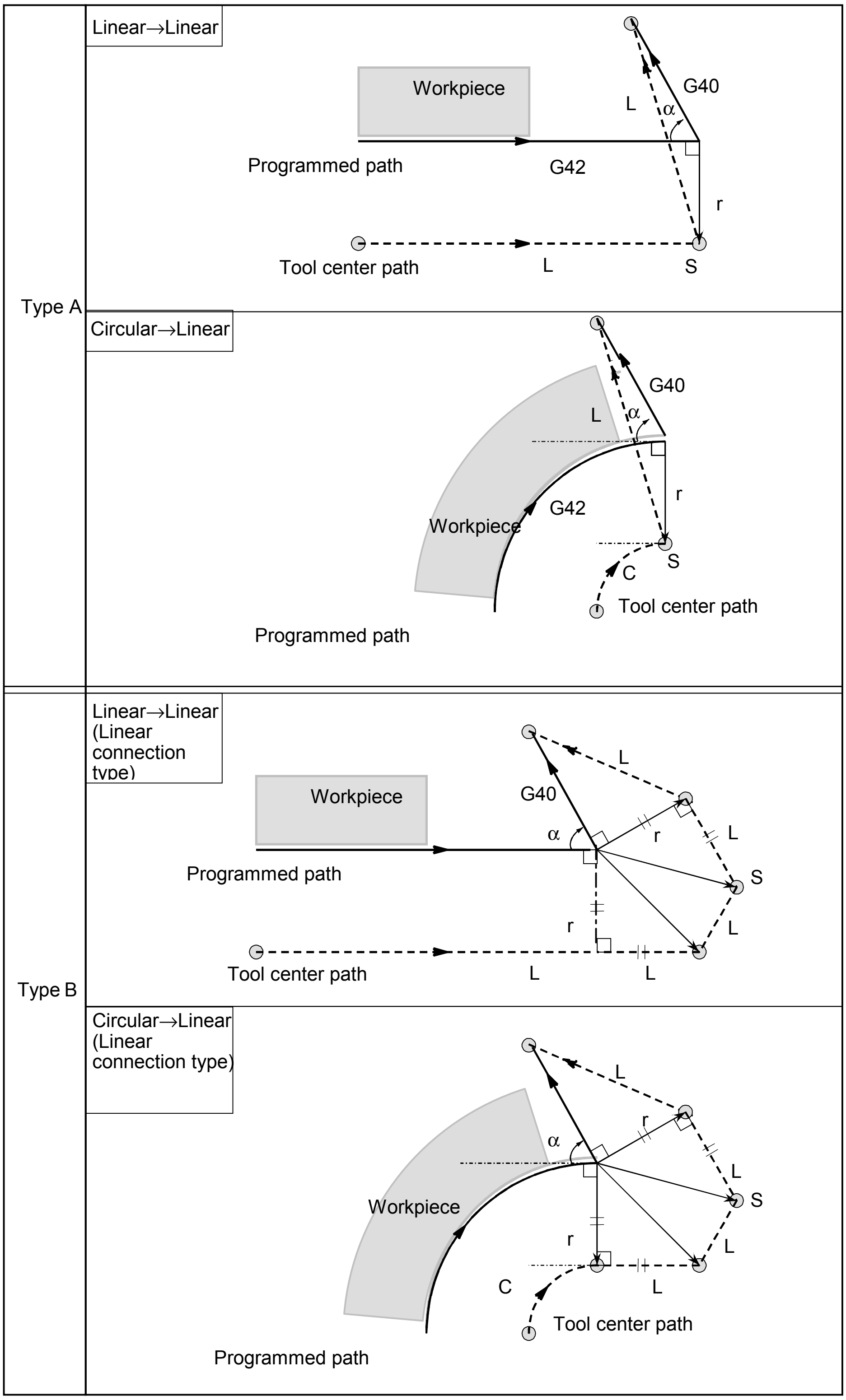
Circular→Linear (Circular connection type)

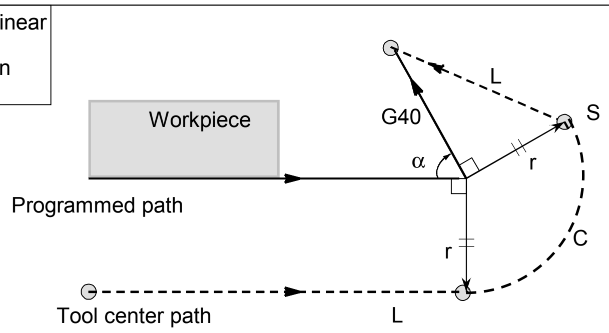
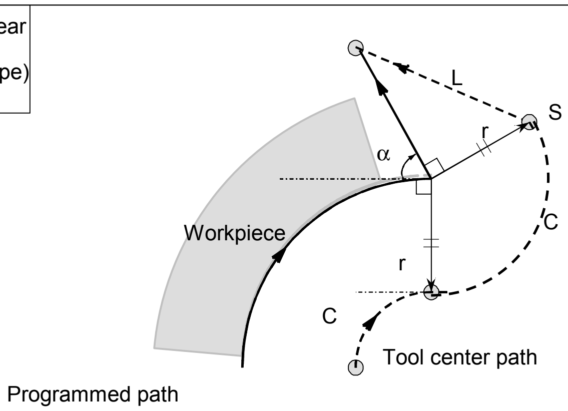


</td>
</tr>
</table>

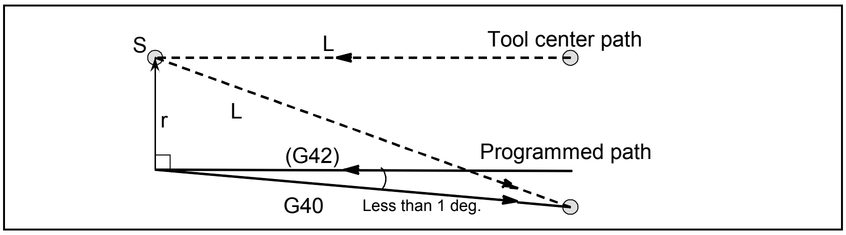
- When a cancellation block involves an outer and acute movement (α<90deg.)

Tool path has two types, A and B : and they are selected by parameter CSU (No. 6001#0)



Type B	Linear→Linear (Circular connection type)  L S Workpiece G40 α r Programmed path C r Tool center path L
	Circular→Linear (Circular connection type)  L S r α Workpiece C r C Tool center path Programmed path

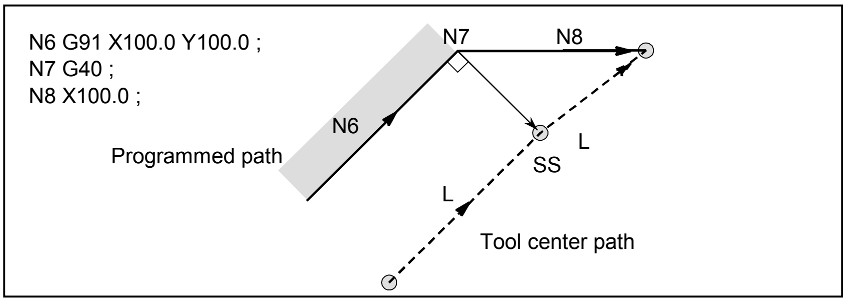
When a cancellation block involves linear-to-linear movement around the outside of an acute angle not greater than 1° (α≤1°)



- A block without tool movement specified together with offset cancel

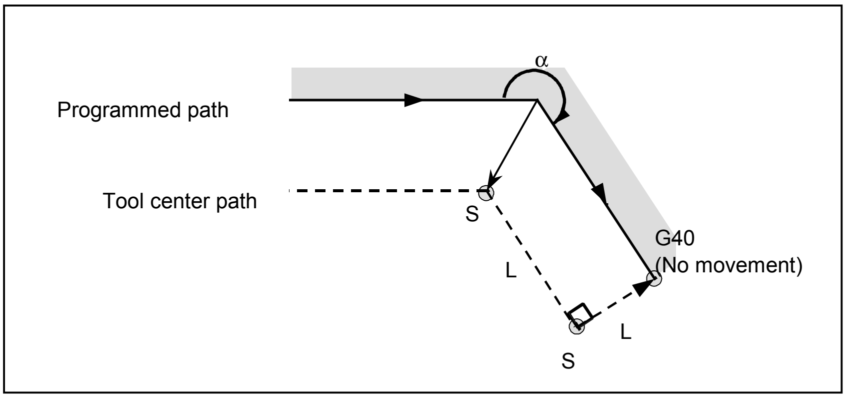
When type A or type B is selected

In the block before cancellation, a perpendicular vector that has the same magnitude as the cutter compensation value is generated. No operation is performed in the cancellation block. Any remaining vector is cancelled when the next move command is specified.



When type C is selected

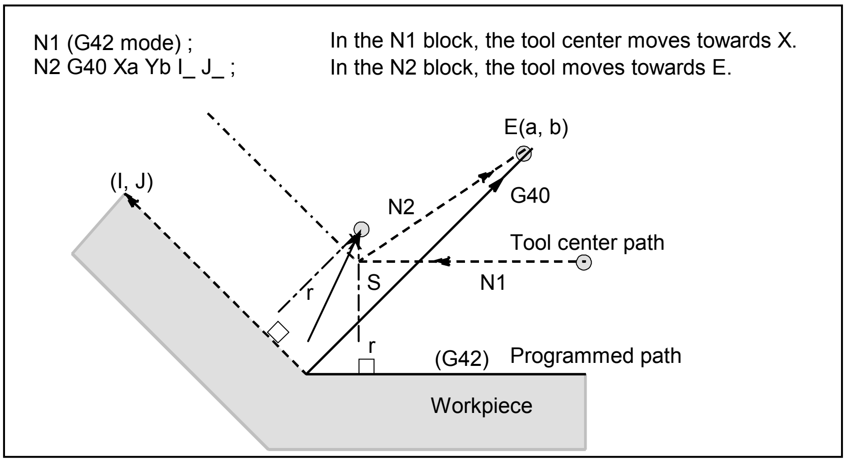
The programmed path is shifted by an offset, perpendicularly from the block before cancellation.



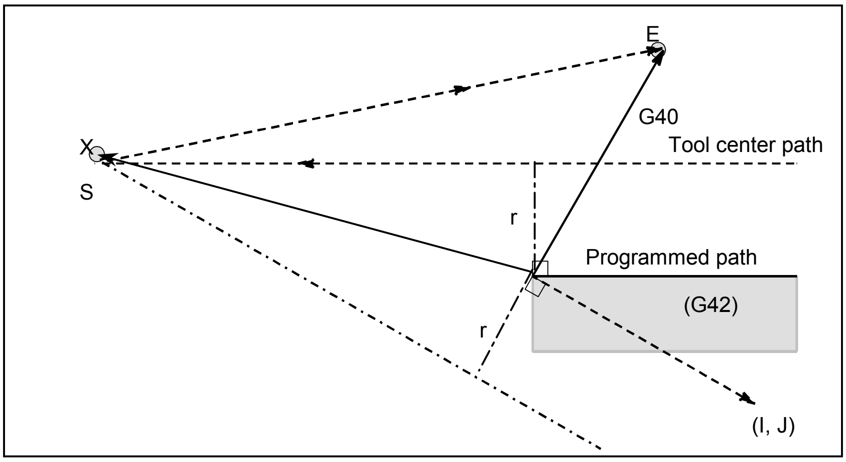
- Block containing G40 and I_J_K_

The previous block contains G41 or G42

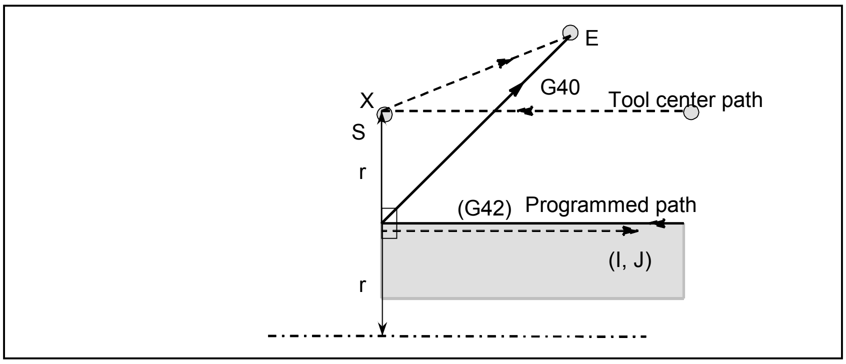
If a G41 or G42 block precedes a block in which G40 and I_, J_, K_ are specified, the system assumes that the path is programmed as a path from the end position determined by the former block to a vector determined by (I,J), (I,K), or (J,K). The direction of compensation in the former block is inherited.



In this case, note that the CNC obtains an intersection of the tool path irrespective of whether inner or outer side machining is specified.



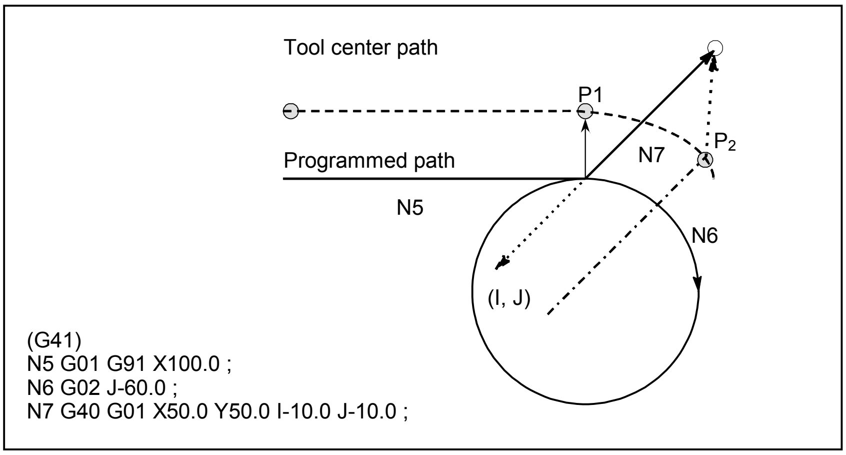
When an intersection is not obtainable, the tool comes to the normal position to the previous block at the end of the previous block.



The length of the tool center path larger than the circumference of a circle

In the example shown below, the tool does not trace the circle more than once. It moves along the arc from P1 to P2. The interference check function described in II-14.4.6 may raise an alarm.

To make the tool trace a circle more than once, Program two or more arcs.

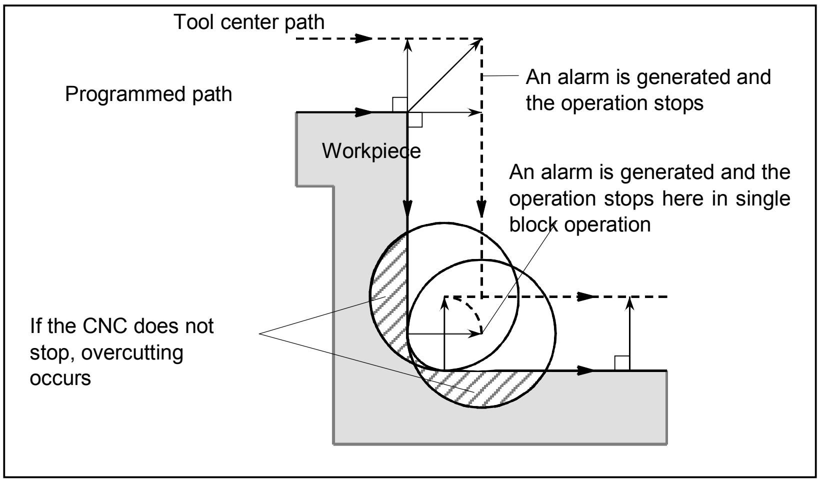


14.4.5 Overcutting by Cutter Compensation

Explanations

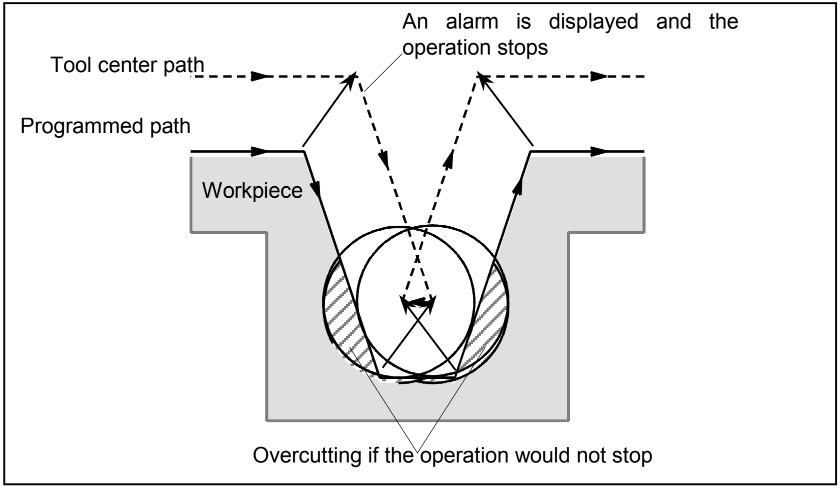
- Machining an inside corner at a radius smaller than the cutter radius

When the radius of a corner is smaller than the cutter radius, because the inner offsetting of the cutter will result in overcuttings. When the interference check alarm function described later is enabled (parameter CNI (No.6001#6)=0, parameter CAV (No.6008#5)=0), an alarm is generated immediately after the previous block starts, and the operation stops. When the previous block stops in single block operation, however, overcutting may occur because the cutter moves up to the end of the block.


Tool center path
Programmed path
An alarm is generated and the operation stops
Workpiece
An alarm is generated and the operation stops here in single block operation
If the CNC does not stop, overcutting occurs

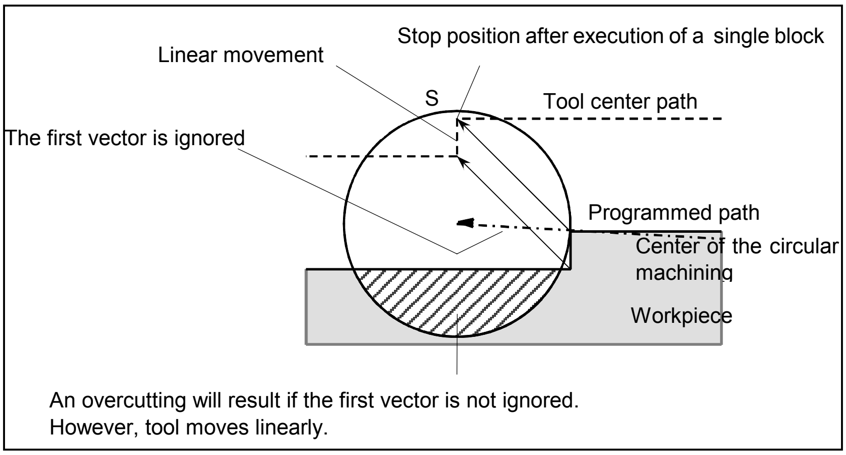
- Machining a groove smaller than the tool radius

Since the cutter compensation forces the path of the center of the tool to move in the reverse of the programmed direction, overcutting will result. When the interference check alarm function described later is enabled (parameter CNI (No.6001#6)=0, parameter CAV (No.6008#5) =0), an alarm is generated immediately after the previous block starts, and the operation stops.


An alarm is displayed and the operation stops
Tool center path
Programmed path
Workpiece
Overcutting if the operation would not stop

- Machining a step smaller than the tool radius

When machining of the step is commanded by circular machining in the case of a program containing a step smaller than the tool radius, the path of the center of tool with the ordinary offset becomes reverse to the programmed direction.
In this case, the first vector is ignored, and the tool moves linearly to the second vector position. The single block operation is stopped at this point. If the machining is not in the single block mode, the cycle operation is continued. If the step is of linear, no alarm will be generated and cut correctly. However uncut part will remain.


Stop position after execution of a single block
Linear movement
S
Tool center path
The first vector is ignored
Programmed path
Center of the circular machining
Workpiece
An overcutting will result if the first vector is not ignored.
However, tool moves linearly.

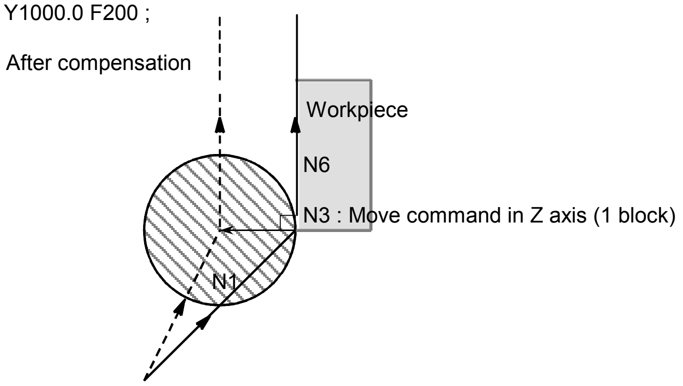
- Starting compensation and cutting along the Z-axis

It is usually used such a method that the tool is moved along the Z axis after the cutter compensation is effected at some distance from the workpiece at the start of the machining.

In the case above, if it is desired to divide the motion along the Z axis into rapid traverse and cutting feed, follow the procedure below.

The number of reading locks in the cutter compensation mode is supposed 3, and it thinks about the following program.

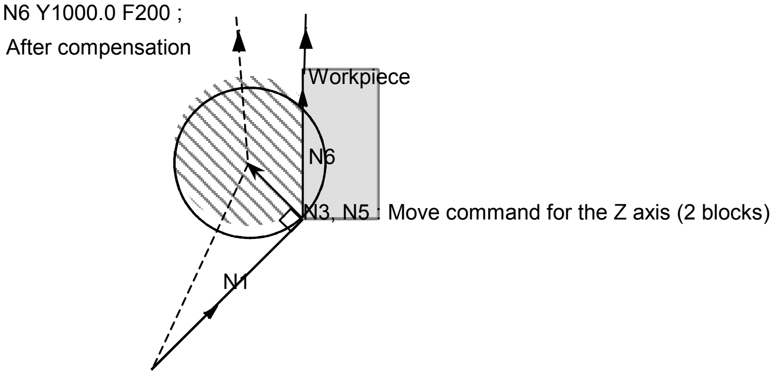
```
N1 G91 G00 G41 X500.0 Y500.0 D1 ;
N3 G01 Z-300.0 F100 ;
N6 Y1000.0 F200 ;
```



In the program example above, when executing block N1, blocks N3 and N6 are also entered into the buffer storage, and by the relationship among them the correct compensation is performed as in the figure above.

Then, if the block N3 is divided as follows:

```
N1 G91 G00 G41 X500.0 Y500.0 D1 ;
N3 G01 Z-250.0 ;
N5 G01 Z-50.0 F100 ;
N6 Y1000.0 F200 ;
```

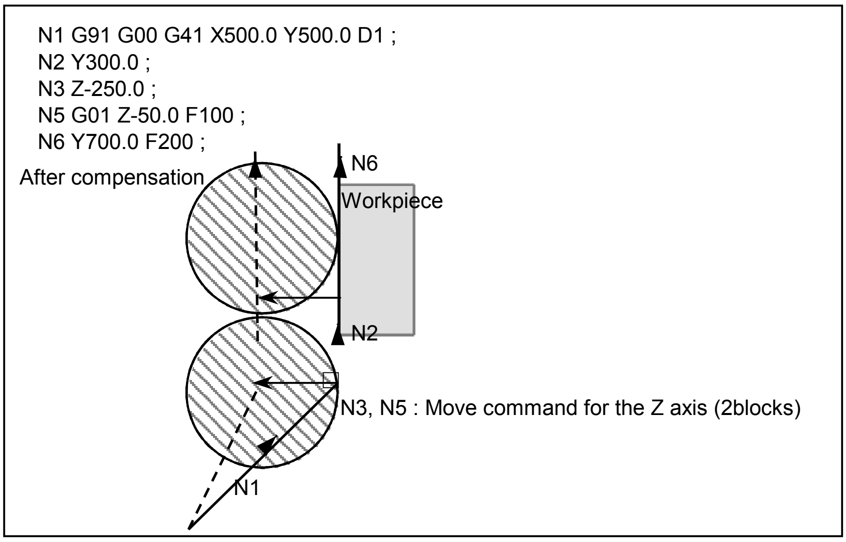


At this time, it can't be read to N5 to the block of N6 though it can be read when the compensation of N1 starts because the number of reading blocks is 3.

Only information on the block of N1 is moved to the point, and compensation is decided to be done, and the vector which is vertical to the end of the compensation start block is made.

The above example should be modified as follows:

To prevent overcutting in this case, instruct the tool to move in the direction in which it is fed after moving along the Z axis according to the above rule immediately before the tool is moved along the Z axis.



```
N1 G91 G00 G41 X500.0 Y500.0 D1 ;
N2 Y300.0 ;
N3 Z-250.0 ;
N5 G01 Z-50.0 F100 ;
N6 Y700.0 F200 ;
```

As the block with sequence No. N2 has the move command in the same direction as that of the block with sequence No. N6, the correct compensation is performed.

Overcutting can also be prevented by specifying a command which specifies the startup block so that the IJ type vector points to the direction in which the tool is fed after it moves along the Z axis (for example, N1 G91 G00 G41 X500, Y500, I0 J1 D1;).

14.4.6 Interference Check

Tool overcutting is called interference. The interference check function checks for tool overcutting in advance. However, not all instances of interference can be checked by this function. The interference check is performed even if overcutting does not occur.

Explanations

- Conditions enabling interference checks

To perform interference check, it is necessary to read at least three blocks that involve tool movement. If at least three blocks that involve tool movement cannot be read because of blocks not involving tool movement (such as independently specified auxiliary functions and dwelling) being specified continuously, interference check becomes impossible, resulting in either overcutting or insufficient cutting. Assuming N and M to be the number of blocks that are read in offset mode (determined by parameter No. 6009) and the number of blocks (among the N blocks) that do not involve tool movement, respectively, the condition enabling intersection check is (N-3) ≥ M. If the maximum number of blocks that are read in offset mode is eight, for example, intersection check is possible even when up to five blocks do not involve tool movement. In this case, interference check is possible for three adjacent blocks, but it is impossible to detect the interference that may occur later.

- Interference check method

Two interference check methods are supported: Direction check and arc angle check methods (described below). Parameter CNC (bit 1 of parameter No. 6001) and parameter CNI (bit 6 of parameter No. 6001) are used to specify whether to enable these methods.

CNI	CNC	Operation
0	0	Interference check is enabled. Both the direction and arc angle check methods can be used.
0	1	Interference check is enabled. Only the arc angle check method can be used.
1	__	Interference check is disabled.

NOTE

It is impossible to specify the use of only the direction check method.

- Criterion 1 for detecting interference (direction check)

Let N be the number of blocks that are read during tool compensation. The check method first checks the compensation vector group calculated between blocks 1 and 2 that are to be output at this time with the compensation vector group calculated between blocks N-1 and N in order to determine whether any vector intersects another. If a vector intersection is detected, it is determined that interference will occur. If no vector intersection is detected, checks are sequentially made on the following pairs of vector groups in such a way that the compensation vector group output this time will be approached:

Blocks 1 and 2 versus blocks N-2 and N-1
Blocks 1 and 2 versus blocks N-3 and N-2

:

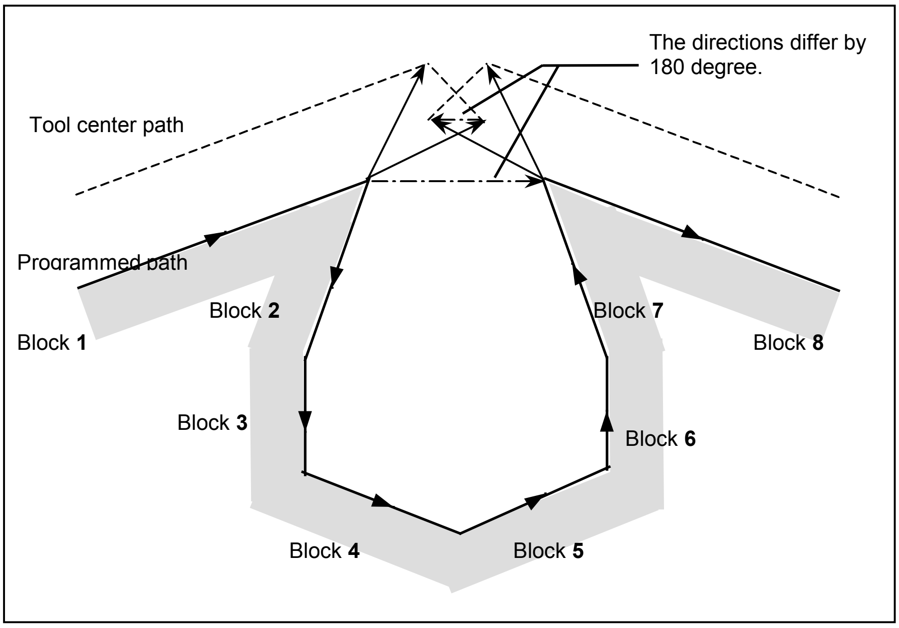
:

Blocks 1 and 2 versus blocks 2 and 3

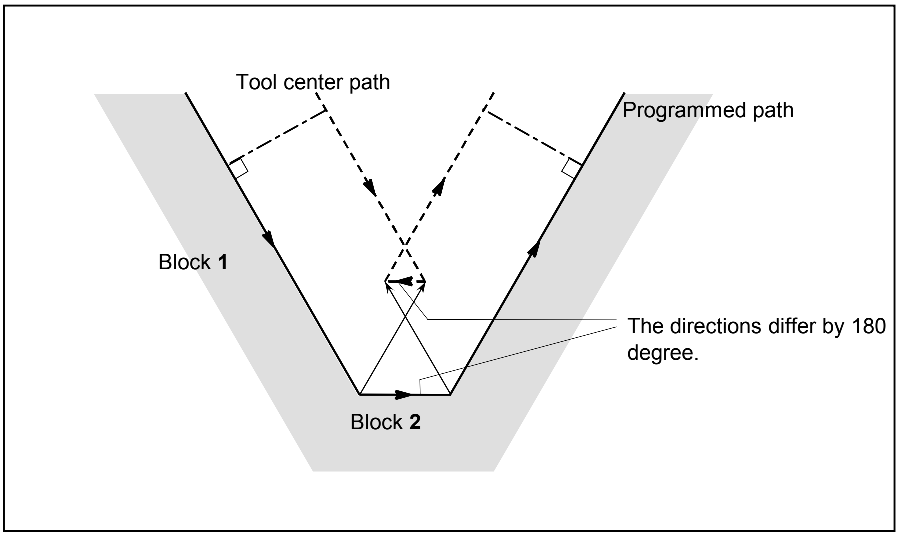
If more than one compensation vector group arises, checks are made on all the pairs.

Decisions are made as follows: In checking the compensation vector group for blocks 1 and 2 with the compensation vector group for blocks N-1 and N, the direction vector from the end point of block 1 to the end point of block N-1 is compared with the direction vector from the point produced by applying the compensation vector to be checked to the end point of block 1 to the point produced by applying the compensation vector to be checked to the end point of block N-1. If the resulting direction is between 90 degree and 270 degree, there is a vector intersection, which indicates that interference will occur. This method is referred to as the direction check method.

Example for criterion 1 for detecting interference (when the vector at the end point of block 1 intersects with the vector at the end point of block 7)



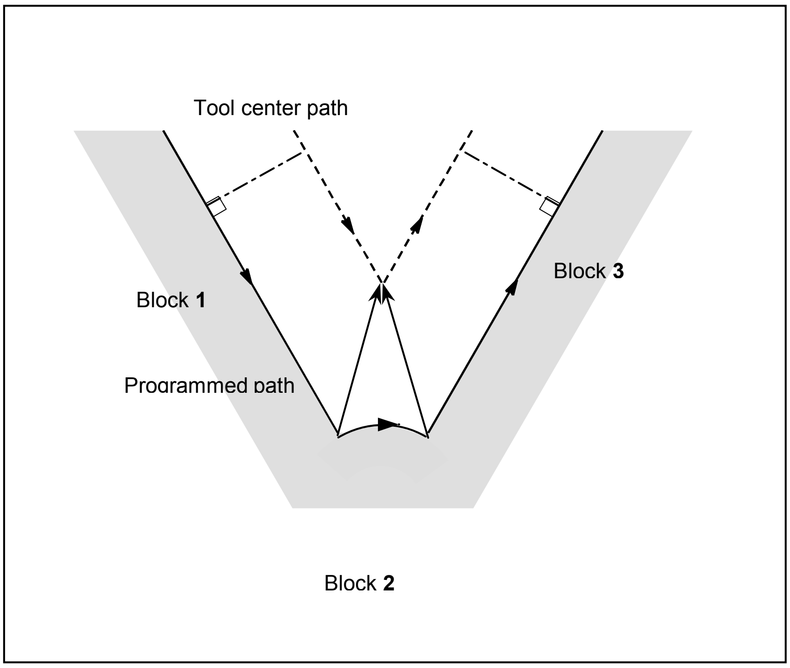
Example for criterion 1 for detecting interference (when the vector at the end point of block 1 intersects with the vector at the end point of block 2)



- Criterion 2 for detecting interference (arc angle check)

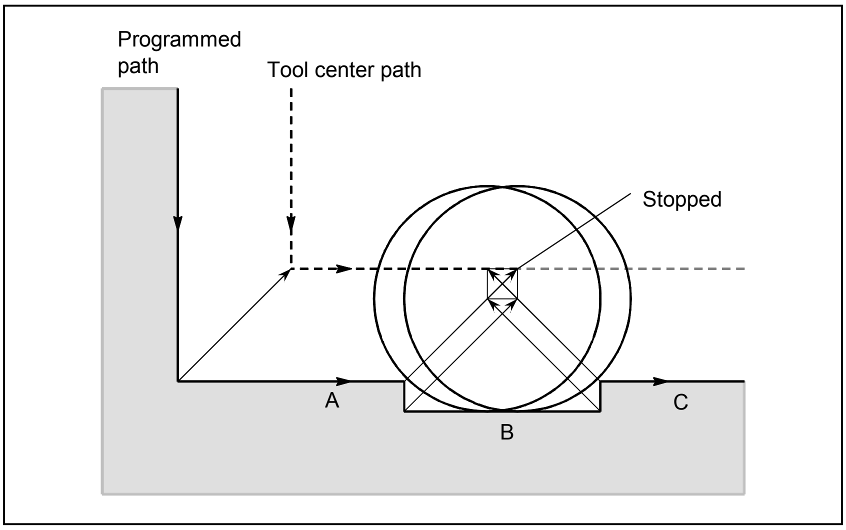
In a check for interference between three adjacent blocks, that is, a check between the compensation vector group calculated between blocks 1 and 2 and the compensation vector group calculated between blocks 2 and 3, an arc angle between the start and end points of the programmed path is checked with an arc angle between the start and end points of the post-compensation path, in addition to the direction check (criterion 1), if block 2 specifies an arc. If the difference is not smaller than 180degree , it indicates that interference will occur. This method is referred to as the arc angle check method.

Example for criterion 2 for detecting interference (when block 2 specifies an arc, and the start and end points of a post-compensation arc coincide with those of the original arc)



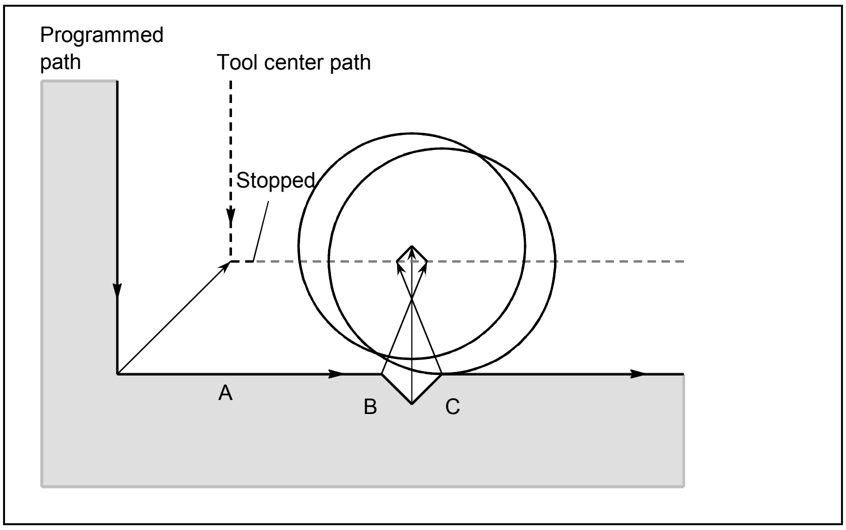
- When interference is assumed although actual interference does not occur

1 Depression which is smaller than a cutter compensation value



There is no actual interference, but since the direction programmed in block B is opposite to that of the path after cutter compensation, the tool stops and an alarm is displayed.

2 Groove which is smaller than a cutter compensation value



Like 1, an alarm is displayed because of the interference as the direction is opposite in block B.

Correction of interference in advance

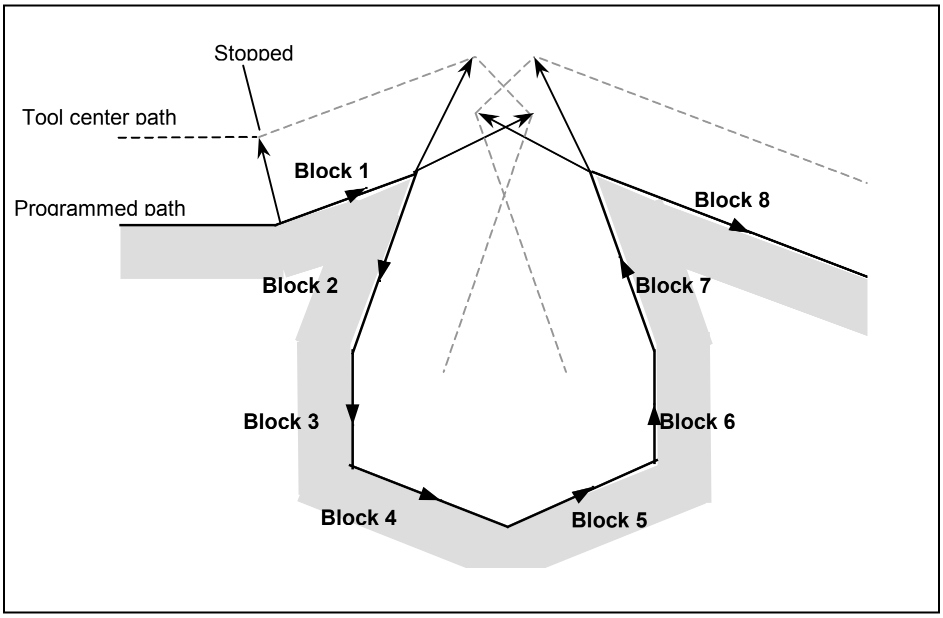
If interference check detects interference (overcutting), the operation to be performed is selected from the following two types, according to the setting of parameter CAV (bit 5 of parameter No. 6008):

CAV	Function	Operation
0	Interference check alarm function	An alarm is displayed and tool movement is stopped just before overcutting (interference) occurs.
1	Interference check avoidance function	The tool path is altered to continue machining so that overcutting (interference) will not occur.

Interference check alarm function

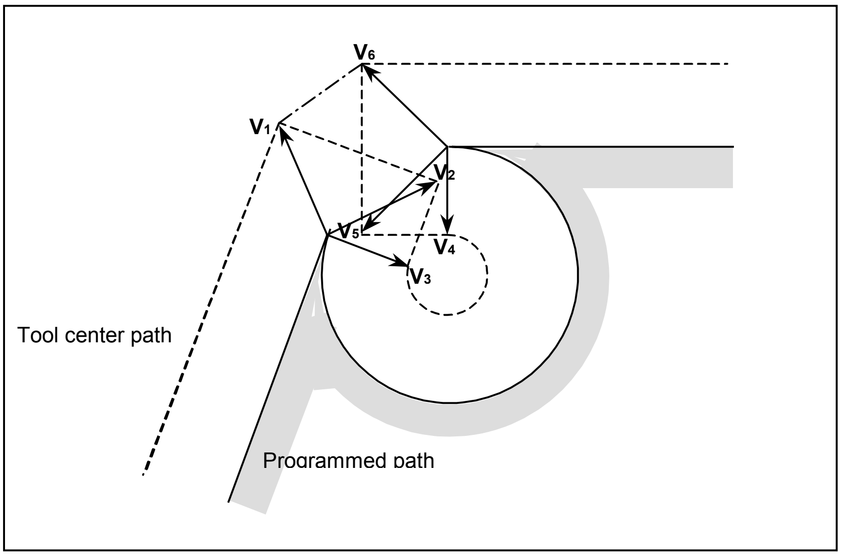
- Interference other than one among three adjacent blocks

If interference is detected between the vector at the end point of block 1 and the vector at the end point of block 7 as shown in the following figure, an alarm is issued and tool movement is stopped before the operation specified in block 1 is performed. In this case, the vectors are not removed.

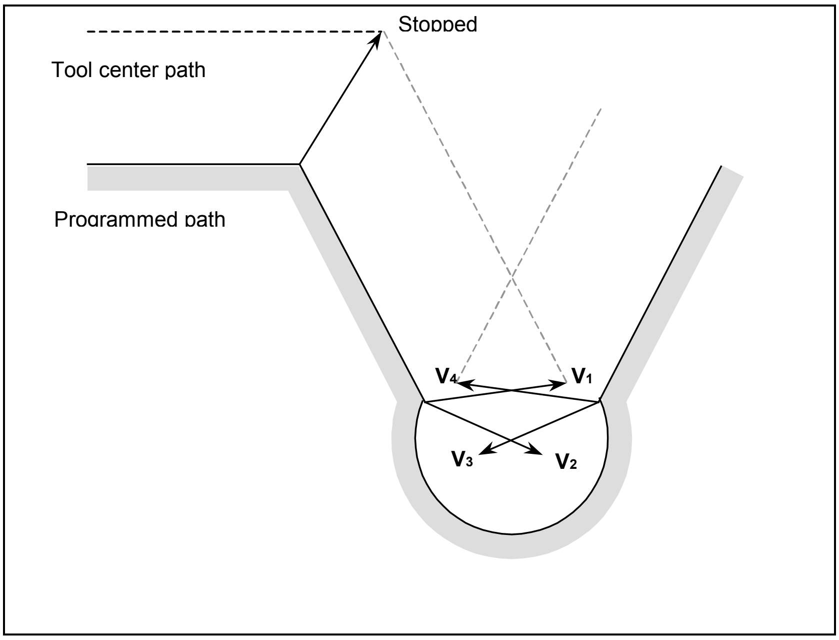


- Interference between three adjacent blocks

If interference is detected between three adjacent blocks, the interfering vectors and those within them are removed, and a path is produced to connect the remaining vectors. In the following figure, V2 and V5 interfere with each other, they are removed, and V3 and V4, which are within them, are also removed, then V1 is connected to V6. This operation is implemented using linear interpolation.



If interference still occurs when there is only one vector after all other vectors have been removed or when there has been only one vector since the beginning, an alarm (PS0272) is issued and tool movement is stopped immediately after the beginning of the previous block (or at the end point of a single block). In the following figure, V2 interferes with V3. Even if this interference is removed, an alarm is displayed because the last pair of vectors, V1 and V4, interferes.



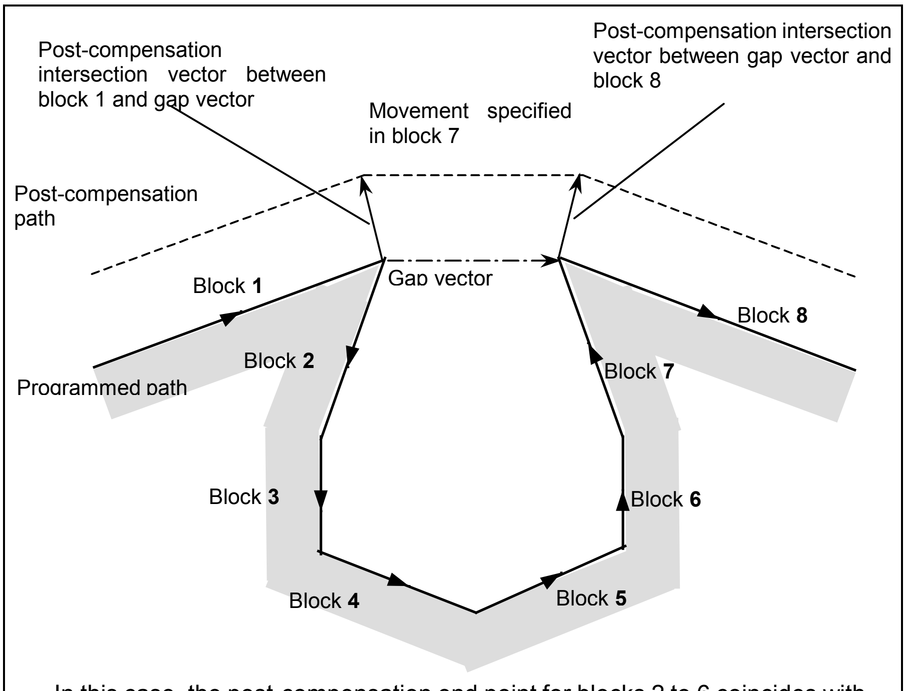
Interference avoidance function

- Overview

Upon the issue of a command that satisfies a condition under which an interference alarm (PS272) is displayed by the interference check alarm function, selecting the interference avoidance function suppresses the interference alarm. Instead, it calculates a new compensation vector to enable machining to continue. On a tool path where interference is avoided, however, the cutting will become insufficient. In addition, a tool path for interference avoidance cannot be produced for some specified figures or may be judged as being dangerous. In these cases, an alarm is displayed, and tool movement is stopped. Because of this, interference avoidance is not necessarily available for all commands.

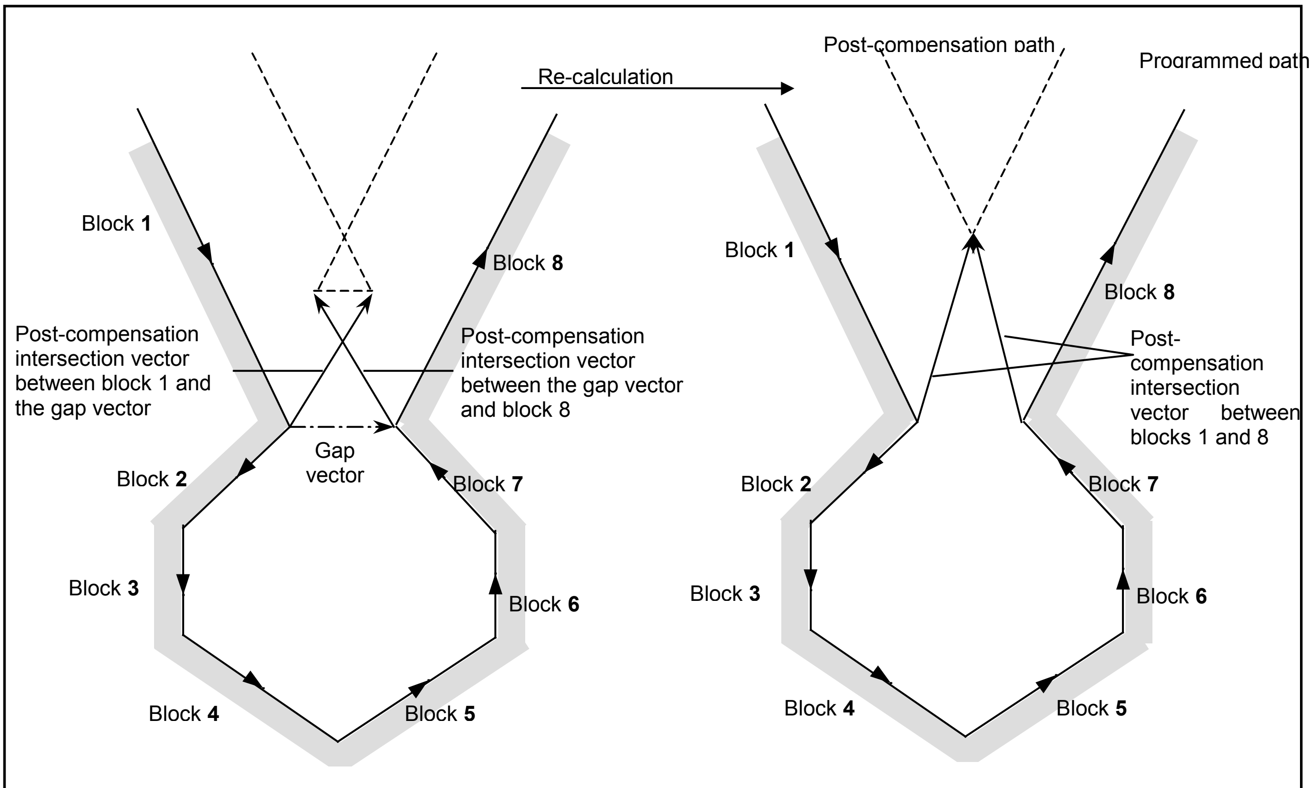
- Interference avoidance method

Let us examine the interference that occurs between the compensation vector between blocks 1 and 2 and the compensation vector between blocks N-1 and N. A direction vector from the end point of block 1 to that of block N-1 is called a gap vector. For interference avoidance, a post-compensation intersection vector between blocks 1 and gap vector is produced. A post-compensation intersection vector between the gap vector and block N is also produced. Then, a path connecting these vectors is generated.


Post-compensation intersection vector between block 1 and gap vector
Post-compensation intersection vector between gap vector and block 8
Movement specified in block 7
Post-compensation path
Block 1
Gap vector
Block 8
Block 2
Programmed path
Block 7
Block 3
Block 6
Block 4
Block 5

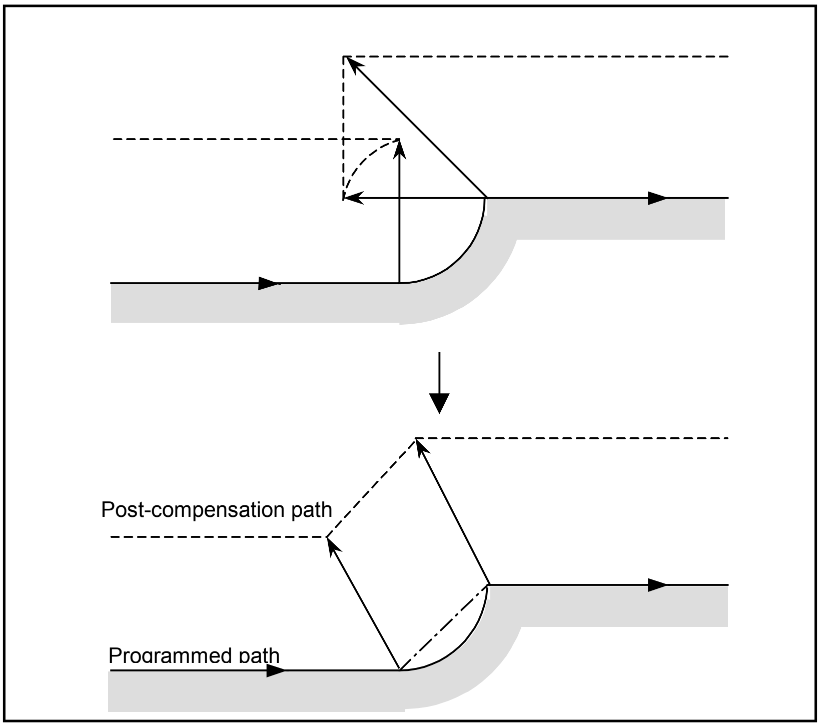
In this case, the post-compensation end point for blocks 2 to 6 coincides with that of block 1. So, blocks 2 to 6 do not involve movement after compensation.

If the post-compensation intersection vector between block 1 and gap vector intersects again with the post-compensation vector between the gap vector and block N, vector removal is carried out first using the same method as that for "Interference between three adjacent blocks." If there still remain vectors that intersect with one another, the post-compensation intersection vector between blocks 1 and N is re-calculated.


Re-calculation
Block 1
Block 8
Post-compensation intersection vector between block 1 and the gap vector
Post-compensation intersection vector between the gap vector and block 8
Gap vector
Block 2
Block 7
Block 3
Block 6
Block 4
Block 5
Post-compensation path
Programmed path
Block 1
Block 8
Post-compensation intersection vector between blocks 1 and 8
Block 2
Block 7
Block 3
Block 6
Block 4
Block 5

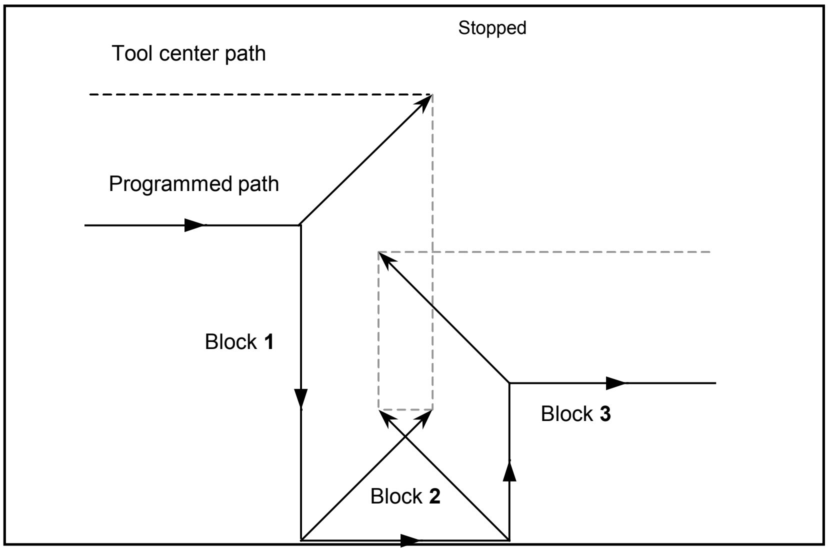
In this case, the post-compensation end point for blocks 2 to 7 coincides with that of block 1. So, blocks 2 to 7 will not involve movement after compensation.

If a cutter compensation value is larger than the radius of a specified arc and a compensation command is issued for the inside of the arc as shown below, interference is avoided by performing intersection calculation where the arc command is assumed to be a liner command. In this case, vectors for interference avoidance are connected using linear interpolation.

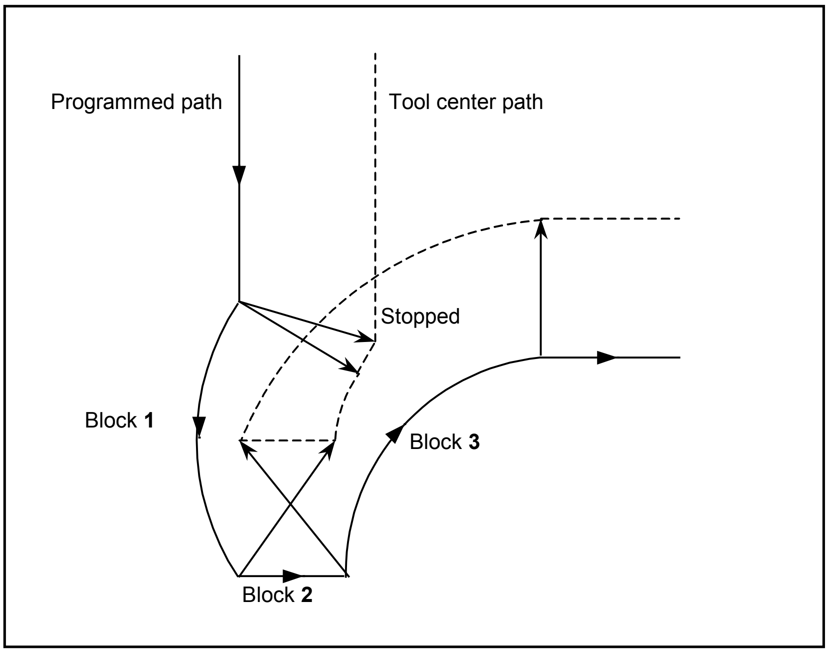


- If there is no interference avoidance vector

In parallel pocketing shown below, interference is detected between the vector at the end point of block 1 and that at the end point of block 2, and an attempt is made to calculate an intersection vector between the post-compensation path for block 1 and the post-compensation path for block 3 for interference avoidance. In this case, there is no such intersection because the paths programmed in blocks 1 and 3 are parallel to each other. So, an alarm (PS0277) is displayed and tool movement is stopped just before block 1.

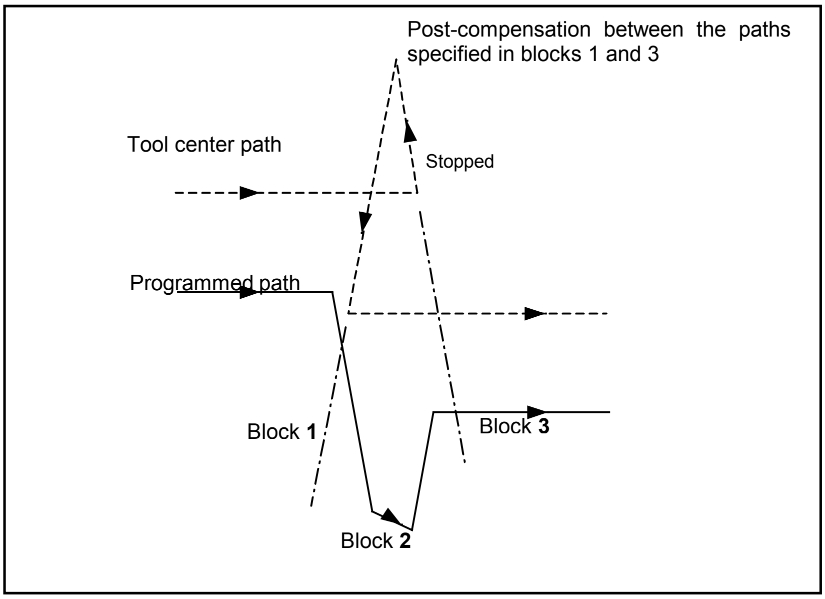


In the circular pocketing shown below, interference is detected between the vector at the end point of block 1 and that of the end point of block 2, and an attempt is made to calculate an intersection vector between the post-compensation path for block 1 and the post-compensation path for block 3 for interference avoidance. In this case, there is no such post-compensation intersection because the paths programmed in blocks 1 and 3 are circular. So, an alarm (PS0277) is displayed and tool movement is stopped immediately before block 1, as in the previous example.

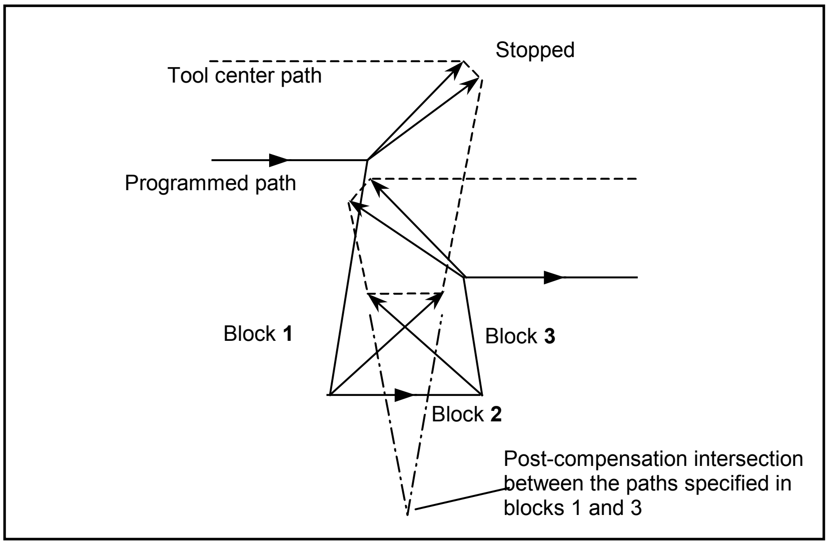


- If interference avoidance is judged as being dangerous

In the acute-angle pocketing shown below, interference is detected between the vector at the end point of block 1 and that at the end point of block 2, and an attempt is made to calculate an intersection vector between the post-compensation path for block 1 and the post-compensation path for block 3 for interference avoidance. In this case, interference avoidance is assumed as being dangerous if the direction of the post-avoidance path will differ considerably from the original path (between 90 degree and 270 degree). So, an alarm (PS0278) is displayed and tool movement is stopped immediately before block 1.

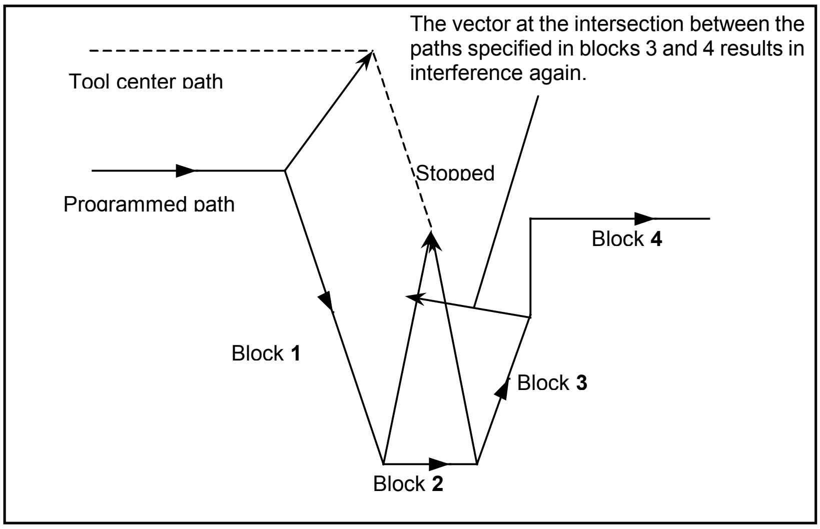

Post-compensation between the paths specified in blocks 1 and 3
Tool center path
Stopped
Programmed path
Block 1
Block 3
Block 2

In machining a pocket that is wider toward the bottom, as shown below, interference is detected between the vector at the end point of block 1 and that at the end point of block 2. An attempt is made to calculate an intersection vector between the post-compensation path for block 1 and the post-compensation path for block 3 for interference avoidance. In this case, the post-avoidance path will result in overcutting, compared with the original path, because the relationship between blocks 1 and 3 is judged as being "outside." This interference avoidance is also judged as being dangerous. So, an alarm (PS0278) is displayed and tool movement is stopped immediately before block 1.


Stopped
Tool center path
Programmed path
Block 1
Block 3
Block 2
Post-compensation intersection between the paths specified in blocks 1 and 3

- If an interference avoidance vector may result in interference again

In the pocketing shown below, interference is detected between the vector at the end point of block 1 and that at the end point of block 2 if three blocks are read, and a vector at the intersection between post-compensation paths for blocks 1 and 3 is calculated for interference avoidance. In this case, however, the vector at the end point of block 3 to be calculated next will interfere with the previous interference avoidance vector. If an interference avoidance vector may result in interference like this again, an alarm (PS0279) is displayed, and tool movement is stopped immediately before execution of the block.


The vector at the intersection between the paths specified in blocks 3 and 4 results in interference again.
Tool center path
Stopped
Programmed path
Block 4
Block 1
Block 3
Block 2

NOTE

1 "If interference avoidance is judged as being dangerous" and "if an interference avoidance vector may result in interference again," setting parameter NAA appropriately (bit 6 of parameter No. 6008) enables machining to be continued without displaying an alarm. "If there is no interference avoidance vector," however, it is impossible to avoid an alarm condition regardless of the parameter setting.

2 If a single-block stop is used to perform a manipulation that produces a difference from the original movement, like manual intervention, MDI intervention, or cutter compensation value change, a new path is used in intersection calculation. Note that this manipulation may result in interference again although interference avoidance has been attempted.

14.4.7 Cutter Compensation by Input from MDI

Explanation

- MDI operation

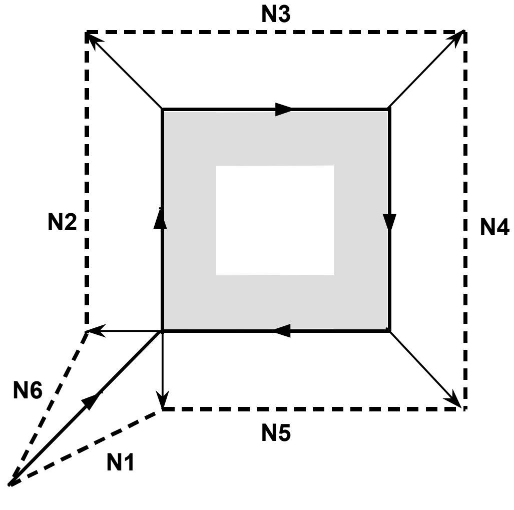
If MDI operation is performed, that is, if a cycle is started from the reset state by a programmed command in MDI mode, an intersection calculation is performed to apply compensation in the same way as for memory operation or DNC operation. Compensation is also applied when a subprogram is called from program memory in MDI operation.

MDI command

```
G90 G00 X0 Y0 ;
M98 P9000 ;
M02 ;
```

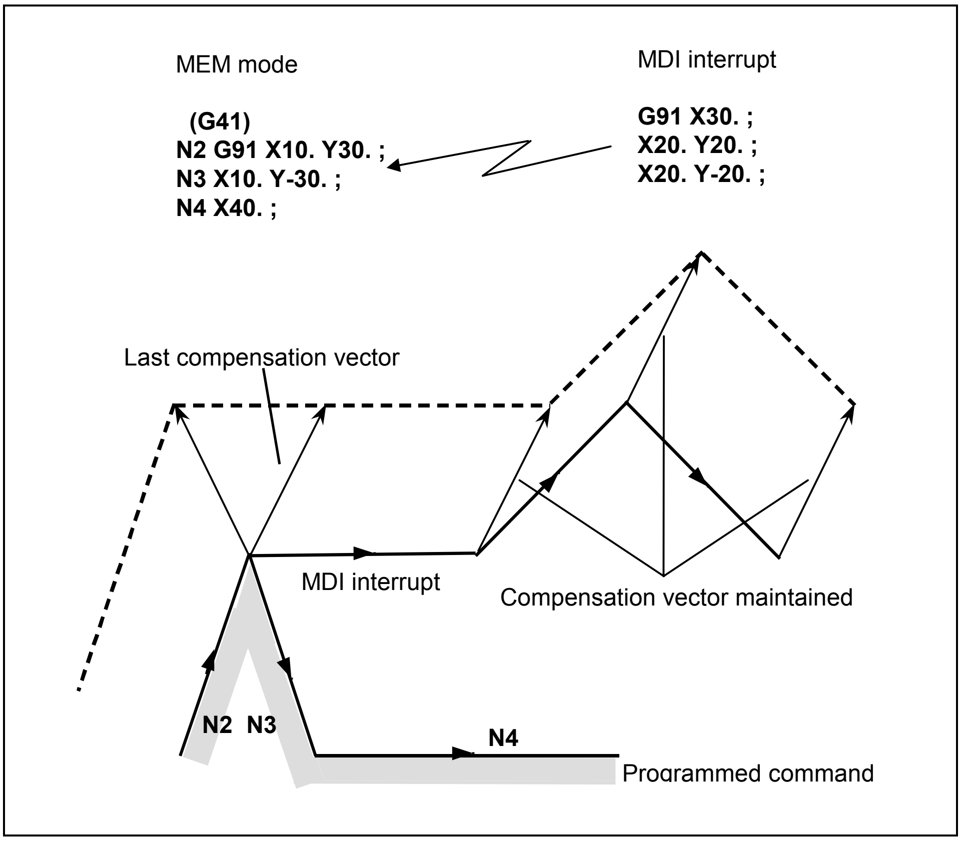
Subprogram in program memory

```
O9000 ;
N1 G41 G17 G91 G01 X10. Y10.
   D1 ;
N2 Y15. ;
N3 X15. ;
N4 Y-15. ;
N5 X-15. ;
N6 G40 X-10. Y-10. ;
M99 ;
```



- MDI interrupt

If an MDI interrupt is generated, that is, if a single block stop is caused during memory operation or DNC operation to enter the automatic operation stop state, then a cycle is started by a programmed command in MDI mode. No intersection calculation is performed, but the last compensation vector before the interrupt is generated is maintained.



14.4.8 Vector Holding (G38)

Issuing G38 in the offset mode when the cutter compensation C function is effective enables the offset vector at the end point for the previous block to be held without calculating the intersection.

Format

(In the offset mode)
G38 IP_ ;

IP : Specified value for axis movement

Explanation

- Vector holding

By specifying the command above, a vector perpendicular to the block immediately before G38 is output at the end point of that block. In the G38 block, the perpendicular vector output in the previous block is held. G38 is a one-shot G code. This means that a compensation vector is re-created by the next move command that is not accompanied by G38.

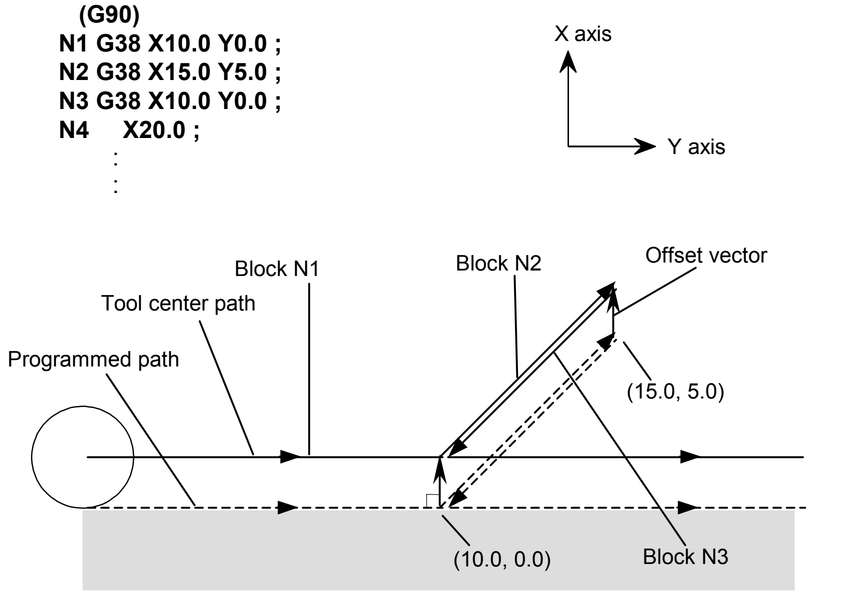
Limitation

- Mode

Specify G38 in G00 or G01 mode. If G38 is specified in G02 or G03 mode (circular interpolation mode), a radius error may result at the start and end points.

Example

```
       :
       :  (In offset mode)
  (G90)
N1 G38 X10.0 Y0.0 ;
N2 G38 X15.0 Y5.0 ;
N3 G38 X10.0 Y0.0 ;
N4     X20.0 ;
       :
       :
```



14.4.9 Corner Circular Interpolation (G39)

By specifying G39 in offset mode during cutter compensation C, corner circular interpolation can be performed. The radius of the corner circular interpolation equals the compensation value.

Format

(In offset mode)

G39 ;

Or,

$$\mathbf{G39}\left\{\begin{array}{l}\mathbf{I\_\ J\_}\\ \mathbf{I\_\ K\_}\\ \mathbf{J\_\ K\_}\end{array}\right\};$$

Explanation

- Corner circular interpolation

When the command indicated above is specified, corner circular interpolation in which the radius equals compensation value can be performed. G41 or G42 preceding the command determines whether the arc is clockwise or counterclockwise. G39 is a one-shot G code.

- G39 without I, J, or K

When G39; is programmed, the arc at the corner is formed so that the vector at the end point of the arc is perpendicular to the start point of the next block.

- G39 with I, J, and K

When G39 is specified with I, J, and K, the arc at the corner is formed so that the vector at the end point of the arc is perpendicular to the vector defined by the I, J, and K values.

Limitation

- Move command

In a block containing G39, no move command can be specified. If a move command is specified in a G39 block, a PS0273 alarm is issued.

- Inner corner

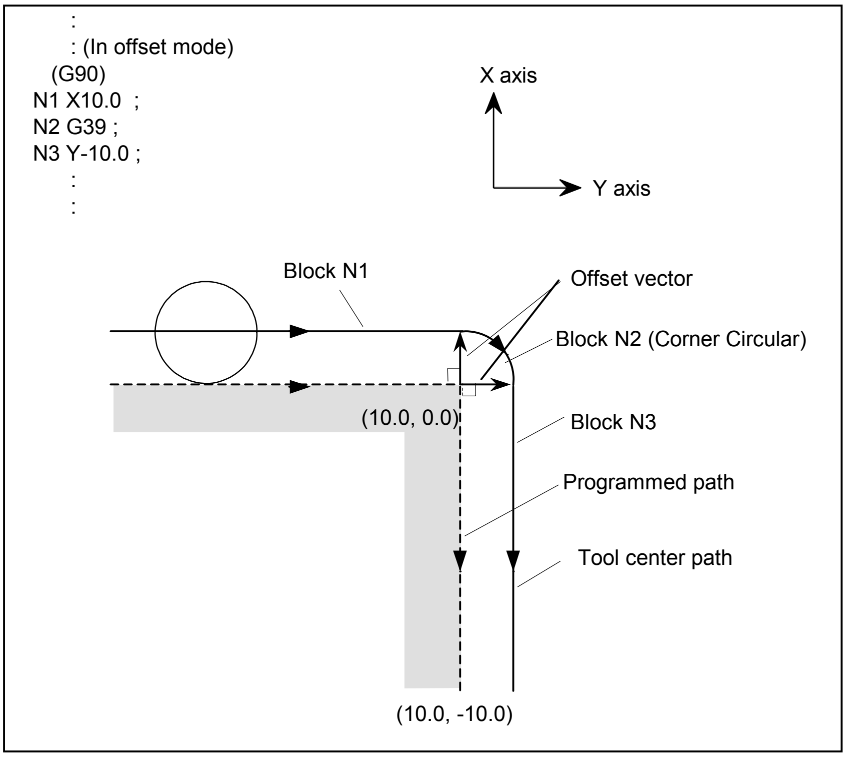
G39 must not be used for blocks that form an inner corner. Otherwise, overcutting results.

- Feedrate for corner circular interpolation

Even if corner circular interpolation is specified using G39 in G00 mode, the feedrate specified by the previously specified F command is used in the block that specifies corner circular interpolation. When G39 is specified in the case where the F command is not specified even once in the program, the feedrate specified in parameter No. 1493 is used as the feedrate for the block specifying corner circular interpolation.

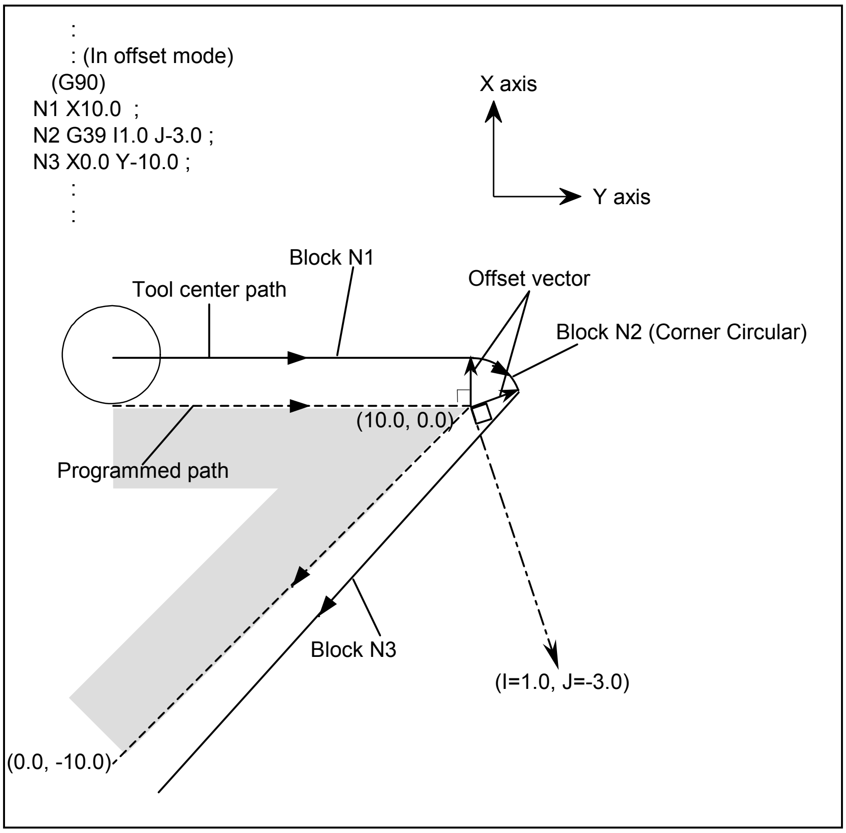
Example

- G39 without I, J, or K



```
      :
      : (In offset mode)
   (G90)
N1 X10.0  ;
N2 G39 ;
N3 Y-10.0 ;
      :
      :
```

- G39 with I, J, and K



```
    :
    : (In offset mode)
  (G90)
N1 X10.0  ;
N2 G39 I1.0 J-3.0 ;
N3 X0.0 Y-10.0 ;
    :
    :
```

14.5 THREE-DIMENSIONAL TOOL COMPENSATION (G40, G41)

In cutter compensation C, two-dimensional offsetting is performed for a selected plane. In three-dimensional tool compensation, the tool can be shifted three-dimensionally when a three-dimensional offset direction is programmed.

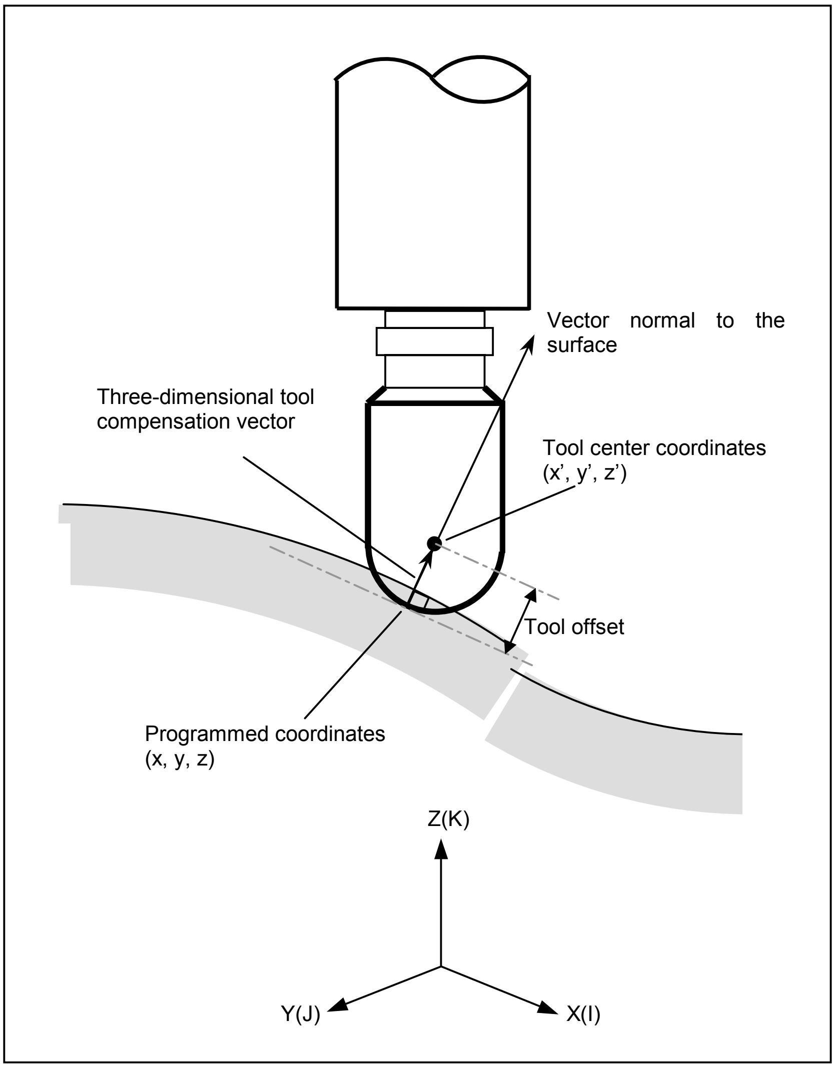

Vector normal to the surface
Three-dimensional tool compensation vector
Tool center coordinates (x', y', z')
Tool offset
Programmed coordinates (x, y, z)
Z(K)
Y(J)
X(I)

Fig.14.5 (a) Three-dimensional Tool Compensation

As described in Fig.14.5(a) the programmed coordinates (x, y, z) are shifted by the tool offset value according to the programmed vector (I, J, K) which is normal to the surface to allow the tool to move using the tool center coordinates (x', y', z').
In this case, the magnitude of the three-dimensional tool compensation vector equals the tool offset value, and the direction of this vector is the same as that of the vector normal to the surface (I, J, K).

Format

- Start up (Starting three-dimensional tool compensation)

When the following command is executed in the cutter compensation cancel mode, the three-dimensional tool compensation mode is set:

G41 Xp_Yp_Zp_ I_ J_ K_D_ ;

Xp : X-axis or a parallel axis
Yp : Y-axis or a parallel axis
Zp : Z-axis or a parallel axis
I :
J : See "Explanation".
K :
D : Code for specifying as the cutter compensation value(1-3 digits) (D code)

- Canceling three-dimensional tool compensation

When the following command is executed in the three-dimensional tool compensation mode, the cutter compensation cancel mode is set:

- When canceling the three-dimensional tool compensation mode and tool movement at the same time

  G40 Xp_Yp_Zp_ ;

  or

  Xp_Yp_Zp_ D00 ;

- When only canceling the vector

  G40;

  or

  D00;

- Selecting offset space

The three-dimensional space where three-dimensional tool compensation is to be executed is determined by the axis addresses specified in the startup block containing the G41 command. If Xp, Yp, or Zp is omitted, the corresponding axis, X-, Y-, or Z-axis (the basic three axis), is assumed.

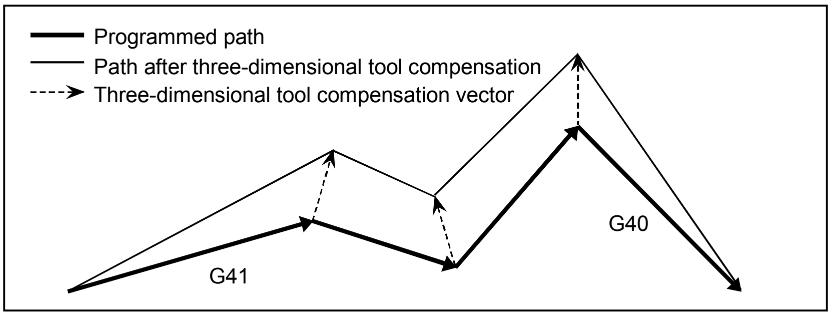
(Example) When the U-axis is parallel to the X-axis, the V-axis is parallel to the Y-axis, and the W-axis is parallel to the Z-axis

```
G41 X_I_J_K_D_;      X, Y, Z space
G41 U_V_Z_I_J_K_D_; U, V, Z space
G41 W_I_J_K_D_;     X, Y, W space
```

Explanation

- Three-dimensional tool compensation vector

In three-dimensional tool compensation mode, the following three -dimensional compensation vector is generated at the end of each block:


Programmed path
Path after three-dimensional tool compensation
Three-dimensional tool compensation vector
G41
G40

The three-dimensional tool compensation vector is obtained from the following expressions:

$$Vx = \frac{i \times r}{p} \text{ (Vector component along the Xp-axis)}$$

$$Vy = \frac{j \times r}{p} \text{ (Vector component along the Yp-axis)}$$

$$Vz = \frac{k \times r}{p} \text{ (Vector component along the Zp-axis)}$$

In the above expressions, i, j, and k are the values specified in addresses I, J, and K in the block. r is the offset value corresponding to the specified offset number. p is the value obtained from the following expression:

$$P = \sqrt{i^2 + j^2 + k^2}$$

When the user wants to program the magnitude of a three-dimensional tool compensation vector as well as its direction, the value of p in the expressions of Vx, Vy, and Vz can be set as a constant in parameter (No. 6011.) If the parameter is set to 0, however, p is determined as follows:

$$P = \sqrt{i^2 + j^2 + k^2}$$

- Relationships between three-dimensional tool compensation and other compensation functions

Tool length compensation	The path is shifted by three-dimensional tool compensation then further shifted by tool length compensation.
Tool offset	Tool offset cannot be specified in three-dimensional tool compensation mode.
Cutter compensation	When addresses I, J, and K are all specified at startup, three-dimensional tool compensation mode is set. When not all of the three addresses are specified, cutter compensation mode is set. Therefore, cutter compensation cannot be specified in three-dimensional tool compensation mode, and three-dimensional tool compensation cannot be specified in cutter compensation mode.

- Specifying I, J, and K

Addresses I, J, and K must all be specified to start three-dimensional tool compensation. When even one of the three addresses is omitted, two-dimensional cutter compensation C is activated. When a block specified in three-dimensional tool compensation mode contains none of addresses I, J, and K, the same vector as the vector generated in the previous block is generated at the end of the block.

- Modal I, J, and K commands

When bit 0 (OKI) of parameter No. 6030 is set to 1, three-dimensional tool compensation can be performed even if one or two of the I, J, and K commands are omitted; the omitted addresses are assumed to be the same as the previous values.

Example

```
N1    G41 X_ Y_ Z_ I_ J_ K_ D_
N2    X_ Y_ Z_ I_ K_    Vector calculation is
                        performed with the omitted J
                        value assumed to be the
                        value in N1.
N3    X_ Y_ Z_ I_       Vector calculation is
                        performed with the omitted J
                        and K value assumed to be
                        the values in N1 and N2,
                        respectively.
```

- Accurate vector command

When bit 0 (ONI) of parameter No. 6029 is set to 1, up to nine significant digits (nine digits in the integer part to nine digits in the decimal part) can be specified in the I, J, and K commands of three-dimensional tool compensation, regardless of the increment system of the reference axis.

Example

Permitted

```
I.999999999
I1.23456789
I999999999.
```

Not permitted (PS0012 is issued.)

```
I0.999999999
I1.234567890
I999999999.0
```

NOTE

1 When bit 0 (ONI) of parameter No. 6029 is set to 1, the functions using the I, J, and K commands listed below must not be used in three-dimensional tool compensation mode. Otherwise, a PS0282 alarm is issued.
Exponential interpolation (I, J, and K commands of G02.3/G03.2)
Canned cycle shift amount (I and J commands of G76 and G87)
Type 2 of tool center point control (I, J, and K commands of G43.5)
Direction of rigid tapping orientation (I command of G84.2/G84.3)

2 When bit 0 (ONI) of parameter No. 6029 is 1, the I, J, and K commands must be specified using a decimal point. If these commands are specified without the decimal point, the PS0283 alarm is issued.

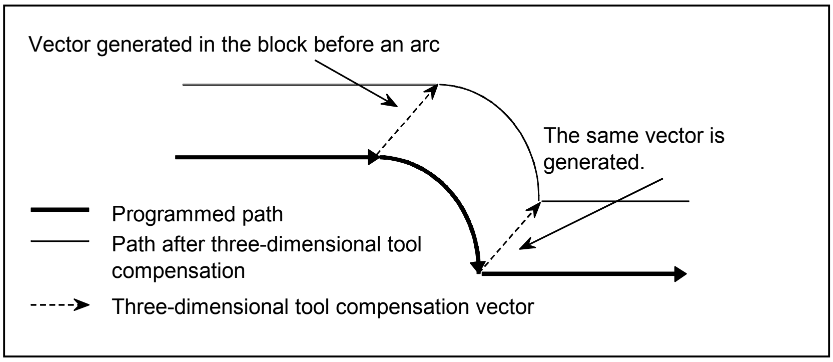
3 When bit 0 (ONI) of parameter No. 6029 is set to 1, macro variables cannot be used for the I, J, and K commands.

- G42

Generally, G41 is specified to start three-dimensional tool compensation. Instead of G41, G42 can be specified for startup. With G42, three-dimensional tool compensation is performed in the opposite direction.

- Offset vector in interpolation

When circular interpolation, helical interpolation (both specified with G02, G03), or involute interpolation (G02.2, G03.2) is specified, the vector generated in the previous block is maintained.


Vector generated in the block before an arc
The same vector is generated.
Programmed path
Path after three-dimensional tool compensation
Three-dimensional tool compensation vector

- Return to a reference position (G28, G30, G30.1)

When return to the reference position (G28), to the second, third, or fourth reference position (G30), or to the floating reference position (G30.1) is specified, the vector is cleared at a middle point.

- Reference position return check (G27)

Before specifying reference position return check (G27), cancel three-dimensional tool compensation.

- Alarm during three-dimensional tool compensation

If one of the following G codes is specified in the three-dimensional tool compensation mode, an alarm is issued:

G31 Skip function (PS alarm (No.0282))
G37 Automatic tool length measurement
(PS alarm (No.0282))

14.6 TOOL COMPENSATION VALUES

Tool compensation values include tool geometry compensation values and tool wear compensation (Fig. 14.6 (a)).

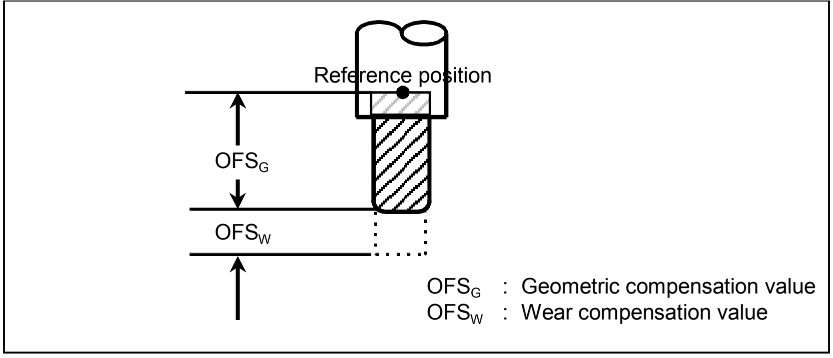

Reference position
OFS_G
OFS_W
OFS_G : Geometric compensation value
OFS_W : Wear compensation value

Fig.14.6 Geometric compensation and wear compensation

Tool compensation memory A, B, or C can be used.

Table14.6 (a) Setting contents tool compensation memory and tool compensation value

Tool compensation value	Tool compensation memory A	Tool compensation memory B	Tool compensation memory C
Tool geometry compensation value for address D	Set tool geometry + tool wear compensation values for addresses D and H (values can be specified with either address).	Set tool geometry compensation values for addresses D and H (values can be specified with either address).	set
Tool geometry compensation value for address H			set
Tool wear compensation for value address D		Set tool wear compensation values for addresses D and H (values can be specified with either address).	set
Tool wear compensation value for address H			set

Explanation

- Increment system and valid range of tool offset values

The increment system and valid range of tool offset values depend on the following parameters:

Parameter OFA(No.6002#0)
Parameter OFC(No.6002#1)
Parameter OFD(No.6004#0)
Parameter OFE(No.6007#0)

Table 14.6 (b) Unit and valid setting range of the tool offset value (metric input)

OFE	OFD	OFC	OFA	Unit	Geometry offset value	Wear offset value
0	0	0	1	0.01 mm	±9999.99 mm	±9999.99 mm
0	0	0	0	0.001 mm	±9999.999 mm	±9999.999 mm
0	0	1	0	0.0001 mm	±9999.9999 mm	±9999.9999 mm
0	1	0	0	0.00001 mm	±9999.99999 mm	±9999.99999 mm
1	0	0	0	0.000001 mm	±999.999999 mm	±999.999999 mm

Table 14.6 (c) Unit and valid setting range of the tool offset value (inch input)

OFE	OFD	OFC	OFA	Unit	Geometry offset value	Wear offset value
0	0	0	1	0.001 inch	±999.999 inch	±999.999 inch
0	0	0	0	0.0001 inch	±999.9999 inch	±999.9999 inch
0	0	1	0	0.00001 inch	±999.99999 inch	±999.99999 inch
0	1	0	0	0.000001 inch	±999.999999 inch	±999.999999 inch
1	0	0	0	0.0000001 inch	±99.9999999 inch	±99.9999999 inch

14.6.1 Tool Compensation Memory A

The memory for geometric compensation and that for wear compensation are not separated in tool compensation memory A. Therefore, the sum of the geometric compensation amount and wear compensation amount is set in the memory.

In addition, the memory for cutter compensation (for D code) and that for tool length compensation (for H code) are not separated.

Example

Offset No.	Compensation amount(geometric + wear)	D code/H code common
001	10.1	For D code
002	20.2	For D code
003	100.1	For H code

14.6.2 Tool Compensation Memory B

The memory for geometric compensation and that for wear compensation are separated in tool compensation memory B. The geometric compensation amount and wear compensation amount can thus be set separately.

However, the memory for cutter compensation (for D code) and that for tool length compensation (for H code) are not separated.

Example

Offset No.	Compensation amount(geometric + wear)	For wear compensation	D code/H code common
001	10.0	0.1	For D code
002	20.0	0.2	For D code
003	100.0	0.1	For H code

14.6.3 Tool Compensation Memory C

The memory for geometric compensation and that for wear compensation are separated in the tool compensation memory C. The geometric compensation amount and wear compensation amount can thus be set separately.

In addition, separate memories are provided for cutter compensation (for D code) and for tool length compensation (for H code).

Example

Offset No.	For D code		For H code	
	For geometric compensation	For wear compensation	For geometric compensation	For wear compenation
001	10.0	0.1	100.0	0.1
002	20.0	0.2	200.0	0.2

14.7 NUMBER OF TOOL COMPENSATION SETTINGS

(1) 32 tool compensation settings
Applicable offset Nos. (D code/H code) are 0 to 32.
D00 to D32 or H00 to H32

(2) 99 tool compensation settings
Applicable offset Nos. (D code/H code) are 0 to 99.
D00 to D99 or H00 to H99

(3) 200 tool compensation settings
Applicable offset Nos. (D code/H code) are 0 to 200
D00 to D200 or H00 to H200

(4) 499 tool compensation settings
Applicable offset Nos. (D code/H code) are 0 to 499.
D00 to D499 or H00 to H499

(5) 999 tool compensation settings
Applicable offset Nos. (D code/H code) are 0 to 999.
D00 to D999 or H00 to H999

14.8 CHANGING THE TOOL COMPENSATION AMOUNT

The tool compensation amount can be set or changed with the G10 command.
When G10 is used in absolute input (G90), the compensation amount specified in the command becomes the new tool compensation amount.
When G10 is used in incremental input (G91), the compensation amount specified in the command is added to the amount currently set.

Format

- For tool compensation memory A

```
G10 L11 P_ R_ ;
    P_: Offset No.
    R_: Tool compensation amount
```

- For tool compensation memory B

```
G10 L10 P_ R_ ; Geometric compensation amount
G10 L11 P_ R_ ; Wear compensation amount
    P_: Offset No.
    R_: Tool compensation amount
```

- For tool compensation memory C

```
G10 L10 P_ R_ ; Geometric compensation amount
                for H code
G10 L11 P_ R_ ; Wear compensation amount for H
                code
G10 L12 P_ R_ ; Geometric compensation amount
                for D code
G10 L13 P_ R_ ; Wear compensation amount for D
                code
    P_: Offset No.
    R_: Tool compensation amount
```

NOTE

The L1 command may be used instead of L11 for format compatibility of the conventional CNC.

14.9 SCALING (G50,G51)

A programmed figure can be magnified or reduced (scaling).
Two types of scaling are supported. One type applies the same rate of magnification to all axes (X, Y, and Z). The other type applies a different rate of magnification to each axis.
Unless specified in the program, the magnification rate specified in the parameter is applied.

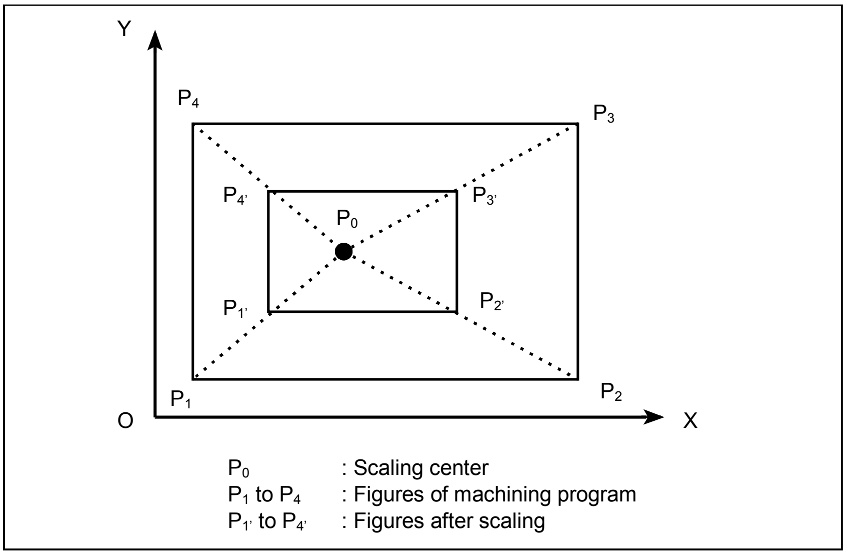

Y
P4
P3
P4'
P3'
P0
P1'
P2'
P1
P2
O
X
P0 : Scaling center
P1 to P4 : Figures of machining program
P1' to P4' : Figures after scaling

Fig.14.9 (a) Scaling

Format

Scaling up or down along all axes at the same rate of magnification (When the parameter XSC (No.7611#4)is 0)	
Format	**Meaning of command**
G51 X_Y_Z_P_ ; Scaling start : Scaling is effective. : (Scaling mode) **G50 ;** Scaling cancel	X_Y_Z_ : Absolute command for center coordinate value of scaling P_ : Scaling magnification

Scaling up or down along each axes at a different rate of magnification (When the parameter XSC(No.7611#4)is 1)	
Format	**Meaning of command**
G51 X_Y_Z_I_J_K_ ;Scaling start : Scaling is effective. : (Scaling mode) **G50 ;** Scaling cancel	X_Y_Z_ : Absolute command for center coordinate value of scaling I_J_K : Scaling magnification for X axis Y axis and Z axis respectively

NOTE

1 Specify G51 in a separate block.
2 After the figure is enlarged or reduced, specify G50 to cancel the scaling mode.
3 No decimal point must be used to specify rates of scaling magnification (P, I, J, and K). Otherwise, an alarm (PS0006) is issued.
4 Even in the case of decimal point input in fixed-point format (when bit 0 (DPI) of parameter No. 2400 is set to 0), the specification of no decimal point is assumed for I, J, and K.

Explanation

- Axes for which scaling is enabled

For each axis for which scaling is to be enabled, set bit 1 (SCL) of parameter No. 0012 to 1.

- Least input increment for specifying rates of magnification

The least input increment for specifying rates of magnification is 0.001 or 0.00001. When bit 1 (SCR) of parameter No. 6400 is set to 0, 0.0001 is used; when the bit is set to 1, 0.001 is used.

- Center of scaling

If X,Y,Z are omitted, the tool position where the G51 command was specified serves as the scaling center.

⚠ **CAUTION**

If a setting value is employed as a scale factor without specifying P, the setting value at G51 command time is employed as the scale factor, and a change of this value by another command, if any, is not effective.

- Scaling along all axes at the same rate of magnification

Set bit 4 (XSC) of parameter No. 7611 to 0. If scaling magnification rate P is not specified, the magnification specified in parameter No. 6410 is assumed

- Scaling along each axis at a different rate of magnification

Set bit 4 (XSC) of parameter No. 7611 to 1. If scaling magnification rates I, J, and K are not specified, the magnification set in parameter No. 6421 is used. Set a non-zero value in parameter No. 6421.

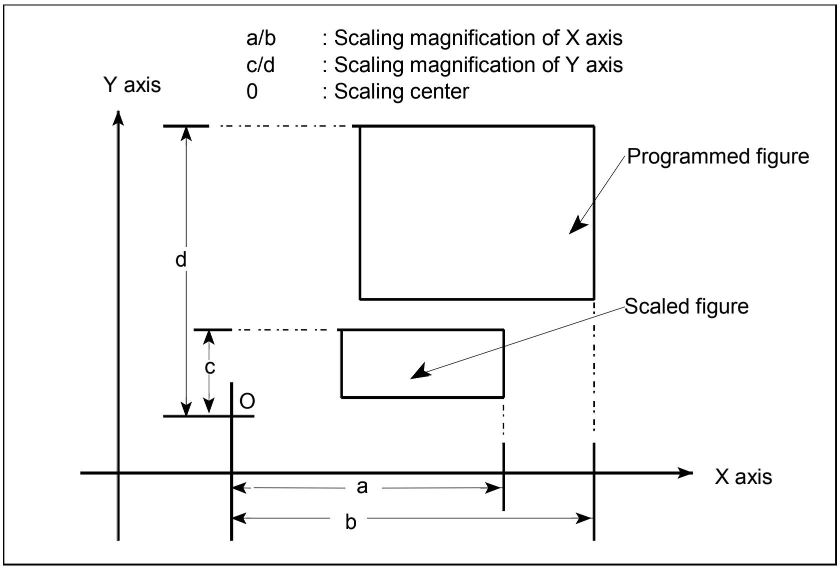

a/b : Scaling magnification of X axis
c/d : Scaling magnification of Y axis
0 : Scaling center
Y axis
Programmed figure
d
Scaled figure
c
O
X axis
a
b

Fig.14.9 (b) Scaling of each axis

- Scaling of circular interpolation

Even if different magnifications are applied to each axis in circular interpolation, the tool will not trace an ellipse.

```
G90 G00 X0.0 Y100.0 Z0.0 ;
G51 X0.0 Y0.0 Z0.0 I2000 J1000;
G02 X100.0 Y0.0 I0 J-100.0 F500 ;
```

Above commands are equivalent to the following command:

```
G90 G00 X0.0 Y100.0 Z0.0 ;
G02 X200.0 Y0.0 I0 J-100.0 F500 ;
```

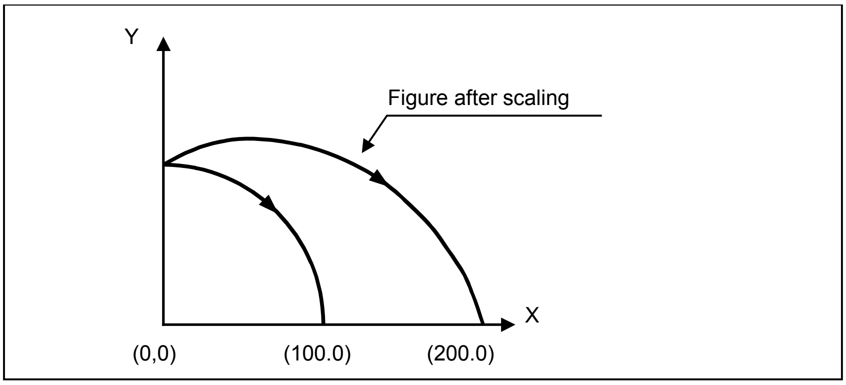


Fig.14.9 (c) Scaling of circular interpolation

In the case of an arc with R specified, the radius (R) is converted to a vector (I, J, K) directed toward the center of each axis. Then, scaling is applied to each of I, J, and K. This means that if the G02 block above specifies an arc with R specified, as indicated below, the same operation as that performed when I and J are specified is performed.

```
G02 X100.0 Y0.0 R100.0 F500 ;
```

- Scaling and coordinate system rotation

When both scaling and coordinate system rotation are specified, the coordinate system is rotated after scaling is applied. In this case, scaling is effective for the center of rotation.

Main program

```
O1
G90 G00 X20.0 Y10.0 ;
M98 P1000 ;
G51 X20.0 Y10.0 I3000 J2000 ;
M98 P1000 ;
G17 G68 X35.0 Y20.0 R30. ;
M98 P1000 ;
G69 ;
G50 ;
M30 ;
```

Sub program

```
O1000 ;
G01 X20.0 Y10.0 F500 ;
G01 X50.0 ;
G01 Y30.0 ;
G01 X20.0 ;
G01 Y10.0 ;
M99 ;
```

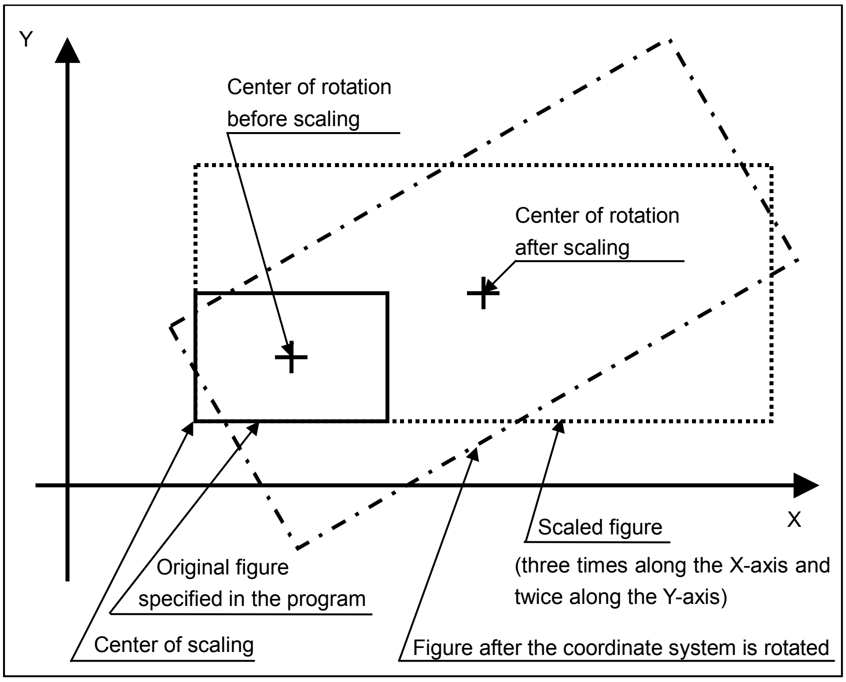
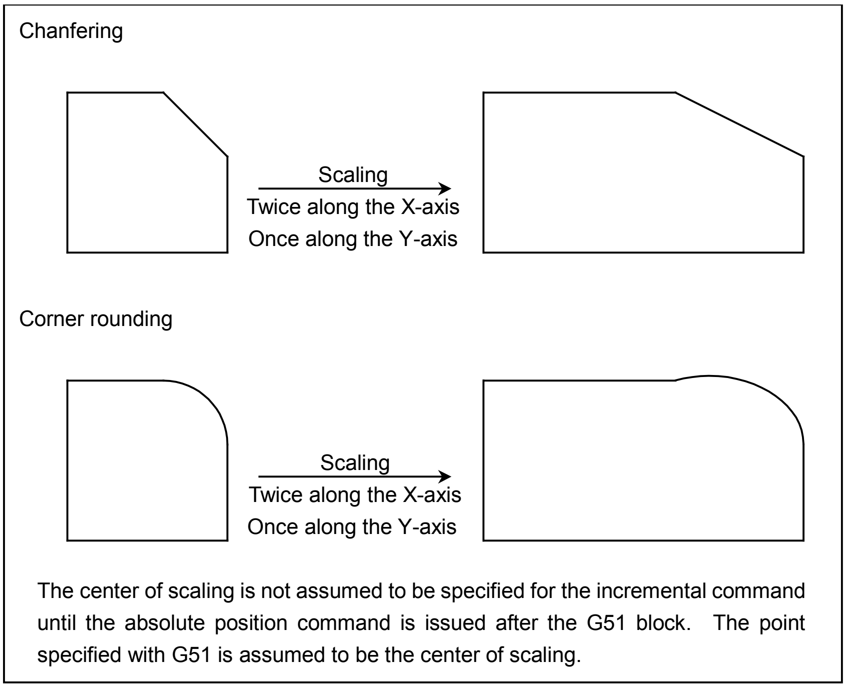

Y
Center of rotation before scaling
Center of rotation after scaling
X
Scaled figure
(three times along the X-axis and twice along the Y-axis)
Original figure specified in the program
Center of scaling
Figure after the coordinate system is rotated

Fig.14.9 (d) Scaling and coordinate system rotation

- Scaling and optional angle chamfering and corner rounding



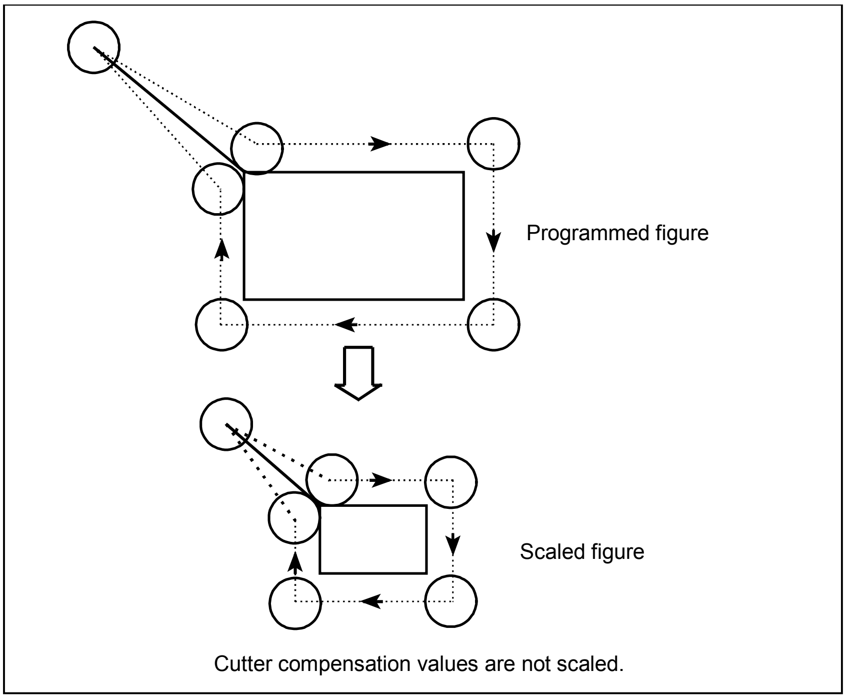
The center of scaling is not assumed to be specified for the incremental command until the absolute position command is issued after the G51 block. The point specified with G51 is assumed to be the center of scaling.

Fig.14.9 (e) Scaling and optional angle chamfering and corner rounding

Limitation

- Tool offset

This scaling is not applicable to cutter compensation values, tool length offset values, and tool offset values (Fig. 14.9 (f)).



Cutter compensation values are not scaled.

Fig.14.9 (f) Scaling during cutter compensation

- Invalid scaling

Scaling is not applicable to the Z-axis movement in case of the following canned cycle.
-Cut-in value Q and retraction value d of peck drilling cycle(G83,G73).
-Fine boring cycle (G76)
-Shift value Q of X and Y axes in back boring cycle (G87).
In manual operation, the travel distance cannot be increased or decreased using the scaling function.

⚠ **CAUTION**

1 If a parameter setting value is employed as a scaling magnification without specifying P, the setting value at G51 command time is employed as the scaling magnification, and change of this value, if any, is not effective.

2 Before specifying the G code for reference position return (G27, G28, G29, G30) or coordinate system setting (G92), cancel the scaling mode.

3 If scaling results are rounded by counting fractions of 5 and over as a unit and disregarding the rest, the move amount may become zero. In this case, the block is regarded as a no movement block, and therefore, it may affect the tool movement by cutter compensation C. See the Item "A block without tool movement" at II-14.4.3.

NOTE

The position display represents the coordinate value after scaling.

14.10 COORDINATE SYSTEM ROTATION (G68,G69)

A programmed shape can be rotated. By using this function it becomes possible, for example, to modify a program using a rotation command when a workpiece has been placed with some angle rotated from the programmed position on the machine. Further, when there is a pattern comprising some identical shapes in the positions rotated from a shape, the time required for programming and the length of the program can be reduced by preparing a subprogram of the shape and calling it after rotation.

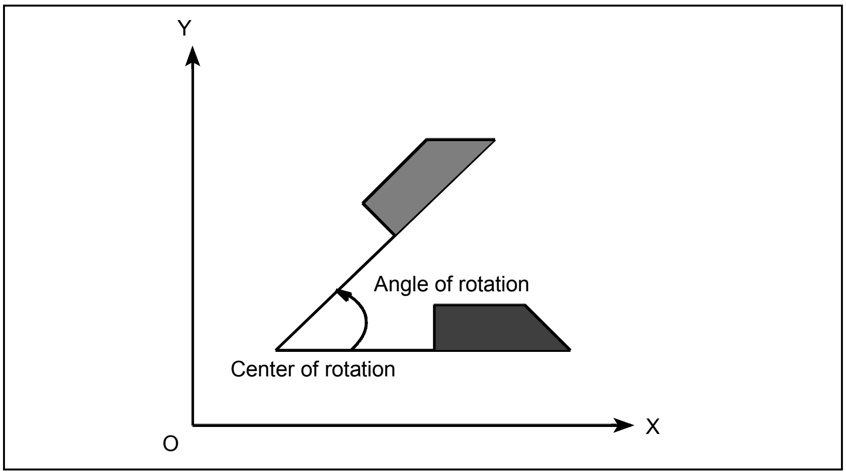


Fig.14.10 (a) Coordinate system rotation

Format

Format
$\left\{\begin{matrix}\mathbf{G17}\\ \mathbf{G18}\\ \mathbf{G19}\end{matrix}\right\}$ **G68**α_ β_**R_ ;**Start rotation of a coordinate system. **:** $\}$ Coordinate system rotation mode **:** **G69 ;** Coordinate system rotation cancel command
Meaning of command
G17(G18 or G19) : Select the plane in which contains the figure to be rotated.
α_ β_ : Absolute command for two of the X_, Y_, and Z_ axes that correspond to the current plane selected by a command (G17, G18, or G19). The command specifies the coordinates of the center of rotation for the values specified subsequent to G68.
R_ : Angular displacement with a positive value indicates counter clockwise rotation.

Explanation

- G code for selecting a plane: G17,G18 or G19

The G code for selecting a plane (G17,G18,or G19) can be specified before the block containing the G code for coordinate system rotation (G68).

> ⚠ **CAUTION**
> G17, G18 or G19 must not be designated in the mode (G68) of coordinate system rotation.

- Least input increment for specifying coordinate system rotation angles

The least input increment for specifying coordinate system rotation angles is 0.001 degree or 0.00001 degree. When bit 2 (RTR) of parameter No. 6400 is set to 0, 0.00001 degree is used; when the bit is set to 1, 0.001 degree is used.

> **NOTE**
> When a decimal fraction is used to specify angular displacement (R_), the 1's digit corresponds to degree units.

- Center of rotation

When α_ β_ is not programmed, the tool position when G68 was programmed is assumed as the center of rotation.

> ⚠ **CAUTION**
> The center of rotation for an incremental command programmed after G68 but before an absolute command is the tool position when G68 was programmed.

- specify angular displacement (R_)

Bit 0 of parameter 6400 selects whether the specified angular displacement is always considered an absolute value or is considered an absolute or incremental value depending on the specified G code (G90 or G91).

When R_ is not specified, the value specified in parameter 6411 is assumed as the angular displacement.

- Tool compensation

Cutter compensation, tool length compensation, tool offset, and other compensation operations are executed after the coordinate system is rotated.

- Relationship with three-dimensional coordinate conversion (G68, G69)

Both coordinate system rotation and three-dimensional coordinate conversion use the same G codes: G68 and G69. The G code with I, J, and K is processed as a command for three-dimensional coordinate conversion. The G code without I, J, and K is processed as a command for two-dimensional coordinate system rotation.

Limitation

- Coordinate system rotation command

Specify the coordinate system rotation command (G68) in G00 or G01 mode.

- Commands related to reference position return and the coordinate system

In coordinate system rotation mode, G codes related to reference position return (G27, G28, G29, G30, etc.) and those for changing the coordinate system (G52 to G59, G92, etc.) must not be specified. If any of these G codes is necessary, specify it only after canceling coordinate system rotation mode.

Example

- Absolute/Incremental position commands

<Sample program>

```
N1 G92 X-50.0 Y-50.0 G17 ;
N2 G68 X70.0 Y30.0 R60.0 ;
N3 G90 G01 X0 Y0 F200;
  (G91 X50.0 Y50.0;)
N4 G91 X100.0 ;
N5 G02 Y100.0 R100.0 ;
N6 G03 X-100.0 I-50.0 J-50.0 ;
N7 G01 Y-100.0 ;
N8 G69 ;
N9 G90 X-50.0 Y-50.0 ;
N10 M02 ;
```

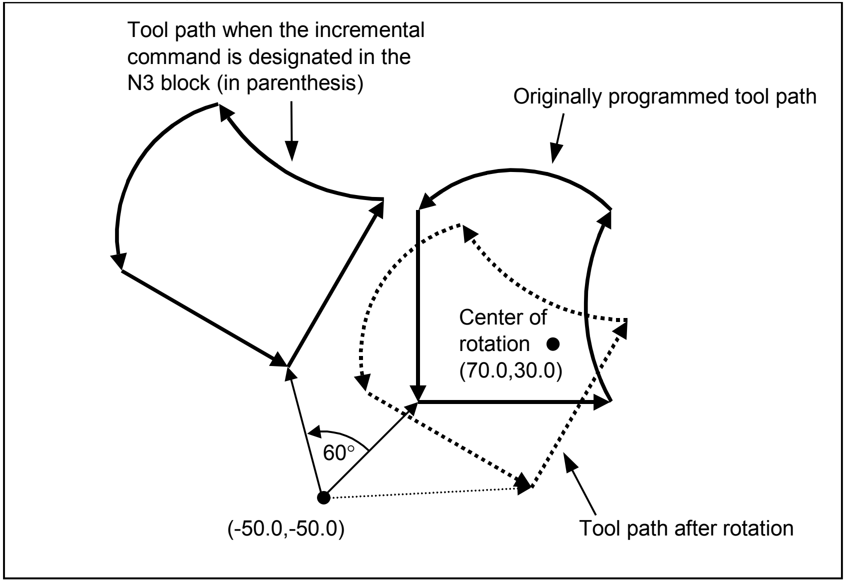

Tool path when the incremental command is designated in the N3 block (in parenthesis)
Originally programmed tool path
Center of rotation ●
(70.0,30.0)
60°
(-50.0,-50.0)
Tool path after rotation

Fig.14.10 (b) Absolute/incremental command during coordinate system rotation

- Cutter compensation and coordinate system rotation

It is possible to specify G68 and G69 in cutter compensation mode. The rotation plane must coincide with the plane of cutter compensation.

<Sample program>

```
    N1 G92 X0 Y0 ;
    N2 G42 G90 G01 X10.0 Y10.0 F1000 D01 ;
    N3 G68 R-30.0 ;
    N4 G91 X20.0 ;
    N5 G03 Y10.0 R10.0 ;
    N6 G01 X-20.0 ;
    N7 Y-10.0 ;
    N8 G69 ;
    N9 G40 G90 X0 Y0 ;
    N10 M30 ;
```

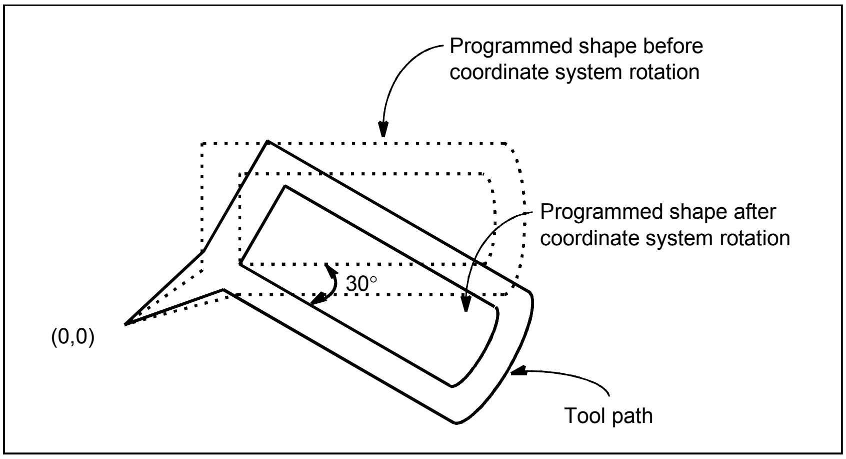

Programmed shape before coordinate system rotation
Programmed shape after coordinate system rotation
30°
(0,0)
Tool path

Fig.14.10 (c) Cutter compensation and coordinate system rotation

- Scaling and coordinate system rotation

If a coordinate system rotation command is executed in the scaling mode (G51 mode), the coordinate value (α, β) of the rotation center will also be scaled, but not the rotation angle (R). When a move command is issued, the scaling is applied first and then the coordinates are rotated.

A coordinate system rotation command (G68) should not be issued in cutter compensation mode (G41, G42) on scaling mode (G51). The coordinate system rotation command should always be specified prior to setting the cutter compensation mode.

Specify the commands in the following order

```
G51;scaling mode start
G68;coordinate system rotation mode start
G41;Cutter compensation mode start
  :
G40;Cutter compensation mode cancel
G69;coordinate system rotation mode cancel
G50;scaling mode cancel
```

<Sample program>

```
G92 X0 Y0 ;
G51 X300.0 Y150.0 P500 ;
G68 X200.0 Y100.0 R45.0 ;
G01 X400.0 Y100.0 ;
Y100.0 ;
X-200.0 ;
Y-100.0 ;
X200.0 ;
```

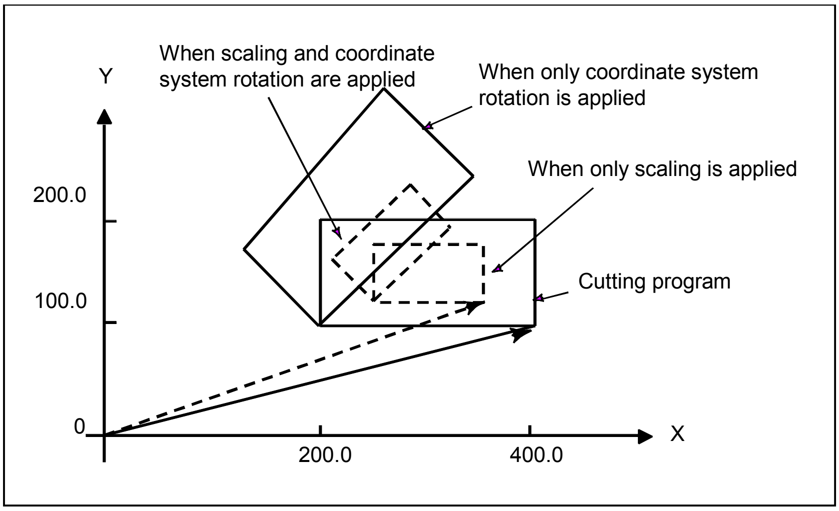

When scaling and coordinate system rotation are applied
When only coordinate system rotation is applied
When only scaling is applied
Cutting program
Y
X
200.0
100.0
0
200.0
400.0

Fig.14.10 (d) Scaling and coordinate system rotation in cutter compensation C mode

- Repetitive commands for coordinate system rotation

It is possible to store one program as a subprogram and recall subprogram by changing the angle.

Sample program for when the RIN bit (bit 0 of parameter 6400) is set to 1.

The specified angular displacement is treated as an absolute or incremental value depending on the specified G code (G90 or G91).

<Sample program>

```
G92 X0 Y0 G69 G17 ;
G01 F200 H01 ;
M98 P2100 ;
M98 P2200 L7;
G00 G90 X0 Y0
M30 ;

O2200 G68 X0 Y0 G91 R45.0 ;
G90 M98 P2100 ;
M99 ;

O2100 G90 G01 G42 X0 Y-10.0 ;
X4.142 ;
X7.071 Y-7.071 ;
G40 ;
M99 ;
```

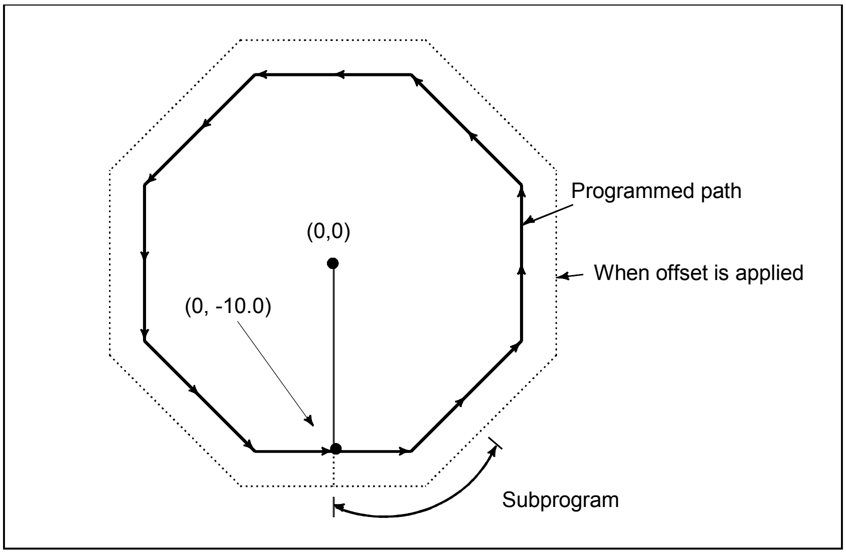


Fig.14.10 (e) Coordinate system rotation command

14.11 TOOL OFFSETS BASED ON TOOL NUMBERS

Cutter compensation data, tool length compensation data, and the tool pot number can be set for a specific tool number (T code). Up to 300 sets of data can be set. If a certain tool number is specified, the pot number corresponding to that tool number is output as a T code to the PMC. If cutter compensation or tool length compensation is specified, compensation is performed using the cutter compensation or tool length compensation data set for the tool number.

14.11.1 Tool Data Registration, Modification, and Deletion

Explanation

- Setting tool data

After all the registered tool data has been deleted, programmed tool data can be registered.

- Adding or modifying tool data

The tool data programmed for a group can be added to or modified. If a previously registered tool number is specified, the tool data for that tool number is updated to specified data.

- Deleting tool data

Registered tool data can be deleted. If an attempt is made to delete the tool data for the currently selected tool, a PS0425 alarm is issued. Any attempt to delete tool data by specifying a tool number and pot number fails if no tool matches the two numbers.

Format

Registration involving tool data deletion

Format	Meaning of command
G10L70;	G10L70 : Starts the deletion of registered tool data and the
T-P-K-R-;	registration of new tool data.
T-P-K-R-;	T- : Tool number(0 to 99999999)
:	P- : Pot number(0 to 9999)
T-P-K-R-;	K- : Tool length offset(Note)
G11;	R- : Cutter compensation value(Note)
M02(M30);	G11 : Ends the registration of setting data.

Adding and modifying tool data

Format	Meaning of command
G10L71;	G10L71 : Starts the addition to or modification of registered tool
T-P-K-R-;	data.
T-P-K-R-;	T- : Tool number(0 to 99999999)
:	P- : Pot number(0 to 9999)
T-P-K-R-;	K- : Tool length offset(Note)
G11;	R- : Cutter compensation value(Note)
M02(M30);	G11 : Ends the addition to or modification of setting data.

⚠ **CAUTION**

The unit of data and valid data range of tool length offsets and cutter compensation values depend on the settings of bit 0 (OFA) of parameter No. 6002, bit 1 (OFC) of parameter No. 6002, bit 0 (OFD) of parameter No. 6004, and bit 0 (OFE) of parameter No. 6007.

Deleting tool data

Format	Meaning of command
G10L72; T-; : P-; : T- P-; : G11; MO2(M30);	G10L72 : Starts the deletion of registered tool data. T- : Delete tool data for the specified tool number. P- : Delete all tool data for the specified pot number. T- P- : Delete tool data for the specified tool number and pot number. G11 : Ends the deletion of setting data.

Example :

Tool data setting

```
N01 G10 L70 ;
N02 T01 P10 K11.0 R12.0 ;
N03 T02 P20 K21.0 R22.0 ;
N04 T03 P20 K31.0 R32.0 ;
N05 T04 P30 K32.0 R42.0 ;
N06 G11 ;
```

Tool data deleting

```
N11 G10 L72 ;
N12 T01 ;          : Deletes the tool data (set in N02) corresponding to T01.
N13 P20 ;          : Deletes the tool data (set in N03 and N04) corresponding to P02.
N14 T04 P30 ;      : Deletes the tool data (set in N05) corresponding to T04 and P30.
N15 T04 P20 ;      : Issues an alarm.
N16 G11 ;
```

14.11.2 Tool Offset Based on Tool Numbers

Explanation

- Tool pot number output

When a tool number (T code) is specified, the corresponding tool pot number is read from the tool data file, then is output to the machine as a tool function code signal (T0 to T31) together with the tool function strobe signal (TF).

- Compensation based on a cutter compensation value and tool length compensation value

When an M code for tool change is specified, the cutter compensation value and tool length compensation value of the previously specified tool number (T code) become valid. Then, compensation is applied using the valid compensation values when a compensation command (such as G43 and G41) is executed.

Example)

```
  Tool data setting
  N01 G10 L70 ;
  N02 T01 P10 K11.0 R12.0 ;
  N03 T02 P20 K21.0 R22.0 ;
  N04 G11 ;


  Compensation based on tool data (when the M code for tool change is M06)
  N11 T01 ;        :  The pot number, 10, corresponding to T01 is output as a code signal.
  N12 M06 ;        :  The cutter compensation value and tool length compensation value
                      corresponding to T01 become valid.
  N13 G43 _ _ _ ;  :  Tool length compensation is performed using the valid compensation
                      value (K11.0 of T01).
  N14 T02 ;        :  The pot number, 20, corresponding to T02 is output as a code signal.
  N15 G41 _ _ _ ;  :  Cutter compensation is performed using the valid compensation value
                      (R12.0 of T02).
  N16 G40 G49 ;
  N17 M06 ;        :  The cutter compensation value and tool length compensation value
                      corresponding to T02 become valid.
  N18 G43 _ _ _ ;  :  Tool length compensation is performed using the valid compensation
                      value (K21.0 of T02).
```

NOTE

Use parameter No. 2429 to specify an M code for tool change.

- Tool change methods

The execution of an M code for tool change and tool number (T code) that are specified in the same block depends on the settings of bit 1 (CT2) and bit 0 (CT1) of parameter No. 7401, as indicated in the table below. The method that is used depends on the machine. For details, refer to the relevant manual provided by the machine tool builder.

Table14.11.2 Differences between Tool Change Methods

CT2 (#1)	CT1 (#0)	Tool change method	Description (tool offset based on tool numbers)
0	0	A	The compensation values corresponding to the previously specified tool number (T code) become valid, and the tool pot number corresponding to a tool number specified in a block where an M code for tool change is specified is output as a code signal. The compensation values corresponding to this tool number do not become valid for the next M code for tool change.
0	1	B	The compensation values corresponding to the previously specified tool number (T code) become valid, and the tool pot number corresponding to a tool number specified in a block where an M code for tool change is specified is output as a code signal. The compensation values corresponding to this tool number become valid for the next M code for tool change.
1	0	C	Same as method B
1	1	D	The tool pot number corresponding to a tool number specified in a block where an M code for tool change is specified is output as a code signal; the compensation values become valid immediately.

Example

- Tool change method A

Example:

N01 T10 ; : The tool pot number corresponding to T10 is output as a code signal.

N02 M06 T11 ; : The cutter compensation value and tool length compensation value corresponding to T10 become valid. The T11 tool is returned to the magazine.

The tool number specified in a block where M06 is specified specifies the number of the tool to be returned to the magazine. Accordingly, the compensation values corresponding to the tool number (T11) specified in the block where M06 is specified do not become valid for the next M code (M06) for tool change.

- Tool change methods B and C

Example:

N01 T10 ; : The tool pot number corresponding to T10 is output as a code signal.

N02 M06 T11 ; : The cutter compensation value and tool length compensation value corresponding to T10 become valid, and the tool pot number corresponding to T11 is output as a code signal.

N03 M06 T10 ; : The compensation values corresponding to T11 become valid, and the tool pot number corresponding to T10 is output.

- Tool change method D

Example:

N01 T10 M06 ; : The tool pot number corresponding to T10 is output as a code signal, and the cutter compensation value and tool length compensation value corresponding to T10 become valid.

N02 T11 M06 ; : The tool pot number corresponding to T11 is output as a code signal, and the cutter compensation value and tool length compensation value corresponding to T11 become valid.

- Notification output to the machine when tools having the same pot number are specified

If there are two or more programmed tool numbers having the same pot number, the pot number duplication signal (TDUP) is output to the machine.

Example:

```
  Tool data setting
  N01 G10 L70 ;
  N02 T01 P10 K11.0 R12.0 ;
  N03 T02 P20 K21.0 R22.0 ;
  N04 T03 P20 K31.0 R32.0 ;
  N05 T04 P30 K32.0 R42.0 ;
  N06 G11 ;

  Compensation based on tool data
  N11 T01 ;          :  The tool pot number, 10, corresponding to T01 is output as a code signal.
  N12 T02 ;          :  The tool pot number, 20, corresponding to T02 is output as a code signal,
                        and the pot number duplication signal is output to the machine at the same
                        time.
```

- T code handling with the program check screen and system variables

Using bit 4 (TLN) of parameter No. 2203, the user can select whether to use a tool pot number or tool number as a T code to be displayed on the program check screen and read by a system variable.

⚠ CAUTION

Relationship with compensation based on an H code or D code

Compensation can be performed by specifying an offset number for tool offset with an H code/D code. However, tool length compensation or cutter compensation cannot be performed by specifying an offset number with an H code/D code. If such an attempt is made, a compensation value of 0 is assumed. By setting bit 5 (NOT) of parameter No. 0011 to 1, however, tool length compensation or cutter compensation can be performed by specifying an offset number with an H code/D code. In such a case, tool offset based on tool numbers (tool pot number output, and compensation based on the compensation value for a tool number) becomes impossible. Set the parameter to the reset state.

14.11.3 Relationships with Other Functions

Tool life management

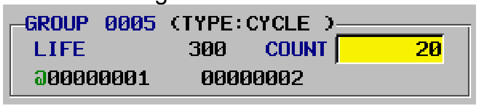
When tool offset based on tool numbers is enabled (when bit 5 (NOT) of parameter No. 0011 is set to 0), a D code and H code cannot be registered as tool life management data.

Compensation enable commands (such as H99/D99), specified during tool life management, cannot be used. If an attempt is made to use a compensation enable command during tool life management, the command is ignored.

If tool offset based on tool numbers and tool life management are used at the same time, rather than a programmed T code (group number), the actually selected tool number (T code) belonging to that group acts as a tool number (T code) command for tool offset based on tool numbers.

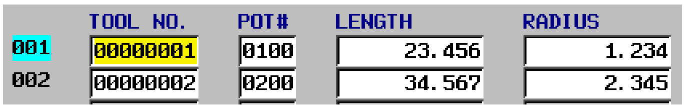
Example :

Tool life management data



```
GROUP 0005 (TYPE:CYCLE )
 LIFE          300    COUNT        20
 @00000001      00000002
```

Tool offset data based on tool numbers



	TOOL NO.	POT#	LENGTH	RADIUS
001	00000001	0100	23.456	1.234
002	00000002	0200	34.567	2.345

Program (the tool life management ignore number is 1000, and the M code for tool change is 6.)

N1 T1005 ;

N2 M06 ;

(1) As T1005 is specified in N1, tool number 1 in tool life management group 05 is selected. Then, in tool offsetting based on tool numbers, the pot number (100) corresponding to T1 is output as a code signal to the machine.

(2) As M06 is specified in N2, life count starts for the tool corresponding to tool number 1. In tool offset based on tool numbers, the compensation values (23.456, 1.234) of T1 become valid.

Automatic tool length measurement

With the automatic tool length measurement command (G37), the tool length compensation value for the currently valid tool number is updated.

Never specify the automatic tool length measurement command in a block in which an M code for tool change is specified.

Custom macro system variable

With the system variables used for tool compensation values, tool offset values cannot be read or updated based on tool numbers.

Tool length compensation along tool axis
Tool center point control

Also in these functions, the compensation amount selected by the tool offset by tool number is used for compensation.

By setting bit 5 (NOT) of parameter No. 0011 to 1, however, you can specify a desired value with an H code to perform compensation. In this case, tool offset by tool number (compensation with the compensation amount for tool pot number output and tool number) is disabled. Set the parameter to the reset state.

14.12 TOOL AXIS DIRECTION TOOL LENGTH COMPENSATION

When a five-axis machine that has two axes for rotating the tool is used, tool length compensation can be performed in a specified tool axis direction on a rotation axis. When a rotation axis is specified in tool axis direction tool length compensation mode, tool length compensation is applied in a specified tool axis direction on the rotation axis by the compensation value specified in the H code. That is, movement is made along the three linear axes (Xp, Yp, Zp).

Unless otherwise noted in the explanation of this function, the two rotation axes are assumed to be the B-axis and C-axis.

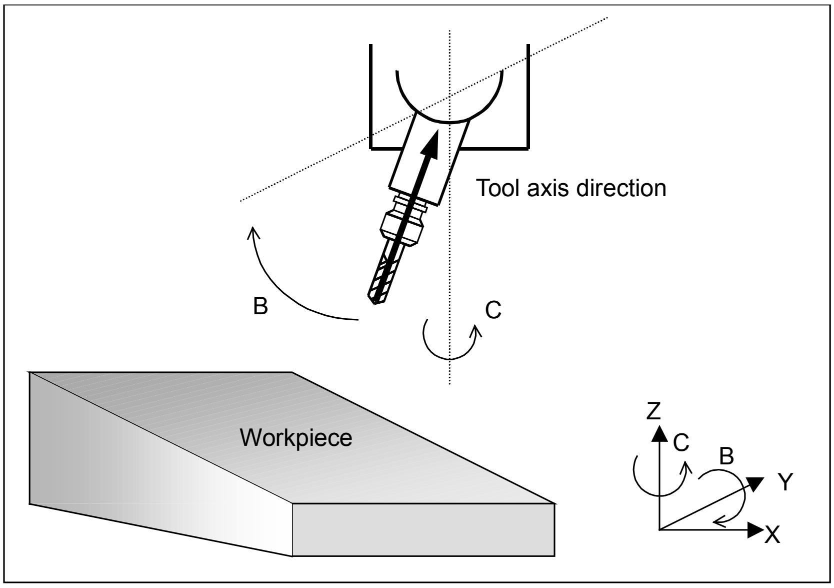


Fig.14.12 (a) Tool Axis Direction Tool Length Compensation

Format

- Command for tool axis direction tool length compensation

```
G43.1 Hn ;
  n : Offset number
```

- Command for cancellation of tool axis direction tool length compensation

```
G49 ;
```

Explanation

- Command for tool axis direction tool length compensation

The tool compensation vector changes as the offset value changes or movement is made on a rotation axis. When the tool compensation vector changes, movement is made according to the change value along the X-axis, Y-axis, and Z-axis.

When the command specifies movement on a rotation axis only, the position of the tool tip is the same both before and after execution of the command. (During rotation axis movement, however, the tool tip moves.)

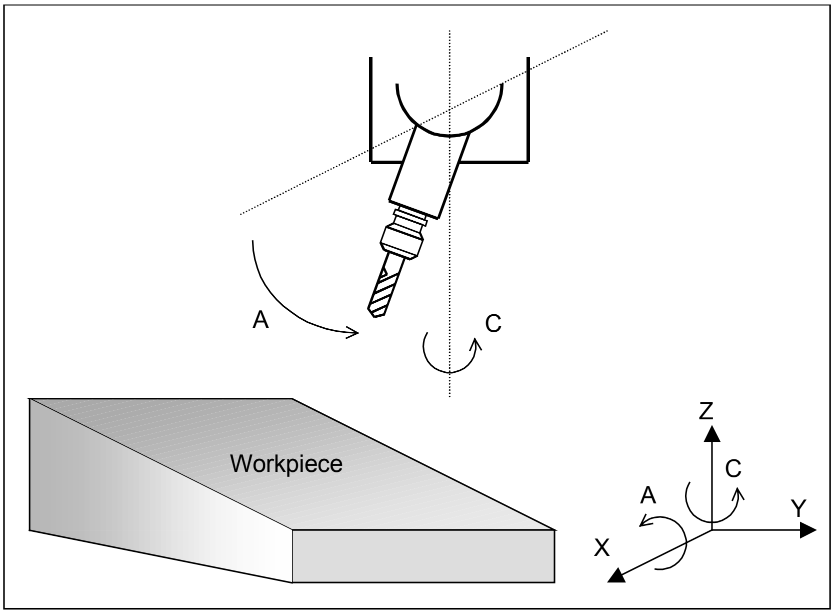
- Examples of machine configuration and rotation axis calculation formats

Let Vx, Vy, Vz, Lc, a, b, and c be as follows :

Vx,Vy,Vz	:	Tool compensation vectors along the X-axis, Y-axis, and Z-axis
Lc	:	Offset value
a,b,c	:	Absolute coordinates on the A-axis, B-axis and C-axis

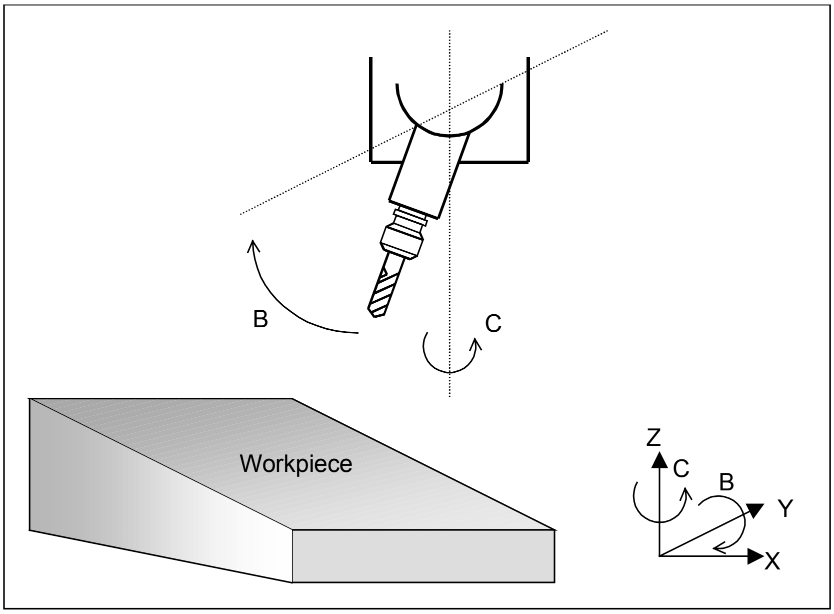
Then, the tool compensation vector on each axis in each machine configuration is indicated below.

(1) A-axis and C-axis, with the tool axis on the Z-axis



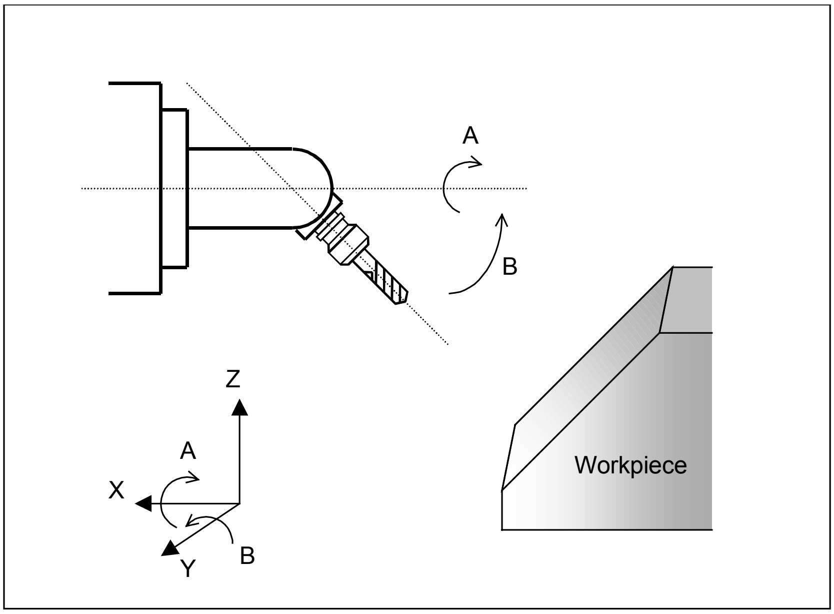
Vx = Lc * sin(a) * sin(c)
Vy = -Lc * sin(a) * cos(c)
Vz = Lc * cos(a)

(2) B-axis and C-axis, with the tool axis on the Z-axis



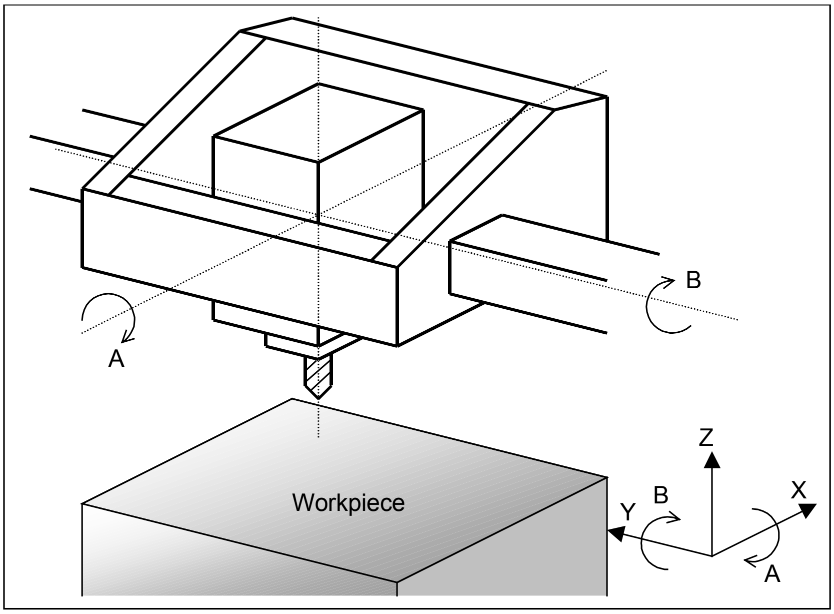
Vx = Lc * sin(b) * cos(c)
Vy = Lc * sin(b) * sin(c)
Vz = Lc * cos(b)

(3) A-axis and B-axis, with the tool axis on the X-axis



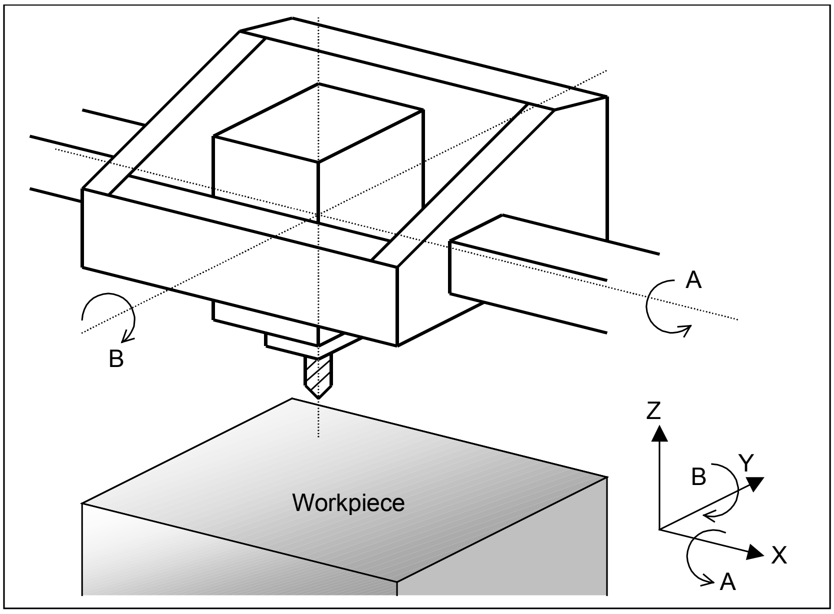
Vx = Lc * cos(b)
Vy = Lc * sin(b) * sin(a)
Vz = -Lc * sin(b) * cos(a)

(4) A-axis and B-axis, with the tool axis on the Z-axis, and the B-axis used as the master



Vx = Lc * cos(a) * sin(b)
Vy = -Lc * sin(a)
Vz = Lc * cos(a) * cos(b)

(5) A-axis and B-axis, with the tool axis on the Z-axis, and the A-axis used as the master



Vx = Lc * sin(b)
Vy = -Lc * sin(a) * cos(b)
Vz = Lc * cos(a) * cos(b)

- Tool holder offset

The machine-specific length from the rotation center of the tool rotation axes (A- and B-axes, A- and C-axes, and B- and C-axes) to the tool mounting position is referred to as the tool holder offset. Unlike a tool length offset value, a tool holder offset value is set in parameter No. 7648. When tool axis direction tool length compensation is applied, the sum of the tool holder offset and tool length offset is handled as a tool length for compensation calculation.

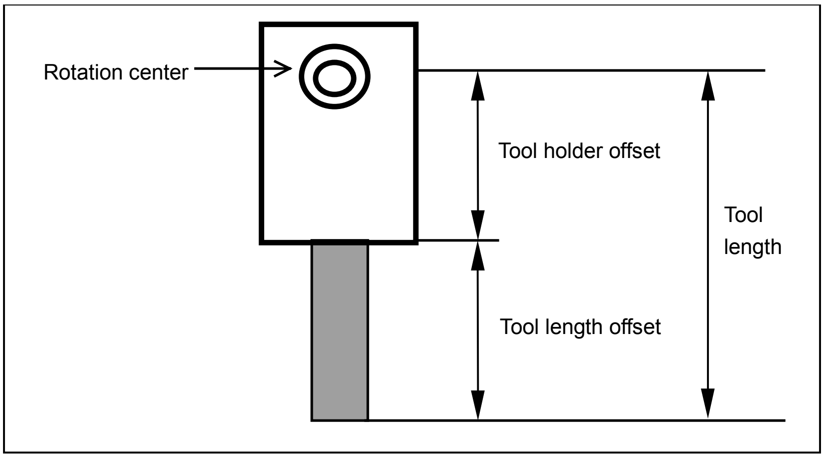


Fig.14.12 (b) Tool Holder Offset

- Parameter-based rotation angle specification

A tool compensation vector is found from the coordinates on the rotation axes for controlling the tool axis direction. However, the configuration of some machines is such that the tool axis is inclined using a fixed attachment. In such a case, the rotation angles of the rotation axes can be set using parameters.
Set bit 1 (RAP) of parameter No. 1014 to 1, and set the coordinates in parameter No. 7516.

- Rotation axis origin compensation

This function compensates for a slight shift of the rotation axis origin caused, for example, by thermal displacement. Specify a compensation value in parameter No. 7518.
When the tool axis is on the Z-axis, and the rotation axes are the B-axis and C-axis, a compensation vector is calculated as follows :

$Xp = Lc * \sin(B-Bz) * \cos(C-Cz)$
$Yp = Lc * \sin(B-Bz) * \sin(C-Cz)$
$Zp = Lc * \cos(B-Bz)$

Xp,Yp,Zp : Compensation pulse on each axis after origin shift compensation
Lc : Offset value
B,C : Machine position on B-axis and C-axis
Bz,Cz : Origin compensation value on B-axis and C-axis

- Rotation axis offset

Set offsets relative to the rotation angles of the rotation axes in parameter No. 7517. The compensation vector calculation formula is the same as that used for rotation axis origin compensation, except that Bp and Cp are changed to rotation axis offsets.

When rotation axis origin compensation and rotation offsetting are set at the same time, both compensations are performed.
When the tool axis is on the Z-axis, and the rotation axes are the B-axis and C-axis, compensation vector calculation is performed as follows :

$$Xp = Lc * \sin(B-(Bz+Bo)) * \cos(C-(Cz+Co))$$
$$Yp = Lc * \sin(B-(Bz+Bo)) * \sin(C-(Cz+Co))$$
$$Zp = Lc * \cos(B-(Bz+Bo))$$

Bz,Cz	: B-axis and C-axis origin compensation values
Bo,Co	: B-axis and C-axis rotation axis offset values

> **NOTE**
> Even in three-dimensional handle feed/interrupt, rotation axis origin compensation and rotation axis offset can be used. Note, however, that a compensation vector for tool axis direction tool length compensation is calculated using absolute coordinates, while machine coordinates are used for three-dimensional handle feed/interrupt. This means that, if there is a mismatch between the absolute coordinates and machine coordinates, a different compensation calculation result is produced by each function. So, when using three-dimensional handle feed/interrupt together with tool axis direction tool length compensation, ensure that the machine coordinates and absolute coordinates match.

Limitation

- Automatic reference position return command (G28, G29, G30)

Never specify an automatic reference position return command (G28, G29, or G30) in tool axis direction tool length compensation mode.
If an automatic reference position return command is specified in tool axis direction tool length compensation mode, the compensation vector is cancelled at the time of reference position return. So, correct tool axis direction tool length compensation is not performed in subsequent movement along linear axes.

- Machine coordinate system positioning (G53)

When machine coordinate system positioning (G53) is performed, the compensation vector is temporarily cancelled in the block, but is applied when movement is next performed.

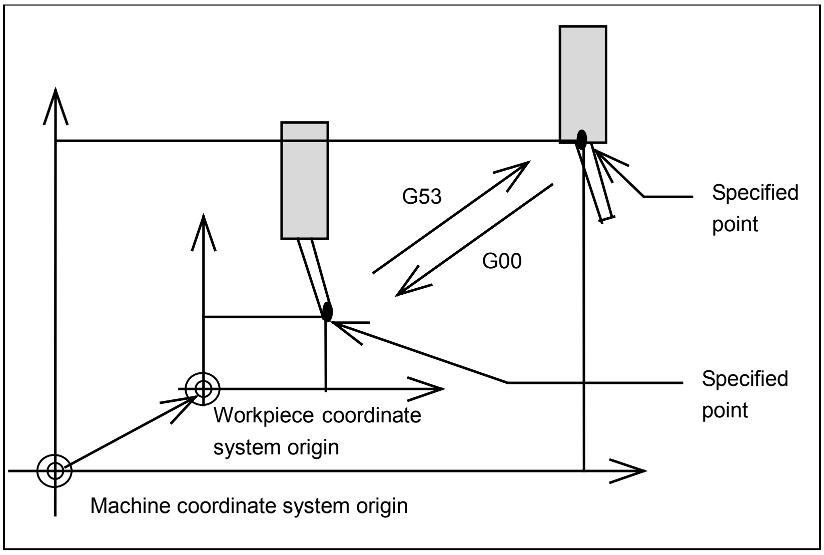

G53
G00
Specified point
Specified point
Workpiece coordinate system origin
Machine coordinate system origin

Fig.14.12 (c) Machine Coordinate System Positioning during Tool Axis Direction Tool Length Compensation

- Tool offset by tool number

When the tool offset by tool number is used, the tool length compensation amount corresponding to a tool number (T code) is used for tool length compensation along the tool axis.

- Tool life management

When tool life management is used, the tool length compensation amount for the tool being used is used for tool length compensation along the tool axis.

14.13 ROTARY TABLE DYNAMIC FIXTURE OFFSET

The rotary table dynamic fixture offset function saves the operator the trouble of re-setting the workpiece coordinate system when the rotary table rotates before cutting is started. With this function the operator simply sets the position of a workpiece placed at a certain position on the rotary table as a reference fixture offset. If the rotary table rotates, the system automatically obtains a current fixture offset from the angular displacement of the rotary table and creates a suitable workpiece coordinate system. After the reference fixture offset is set, the workpiece coordinate system is prepared dynamically, wherever the rotary table is located.

The zero point of the workpiece coordinate system is obtained by adding the fixture offset to the offset from the workpiece reference point.

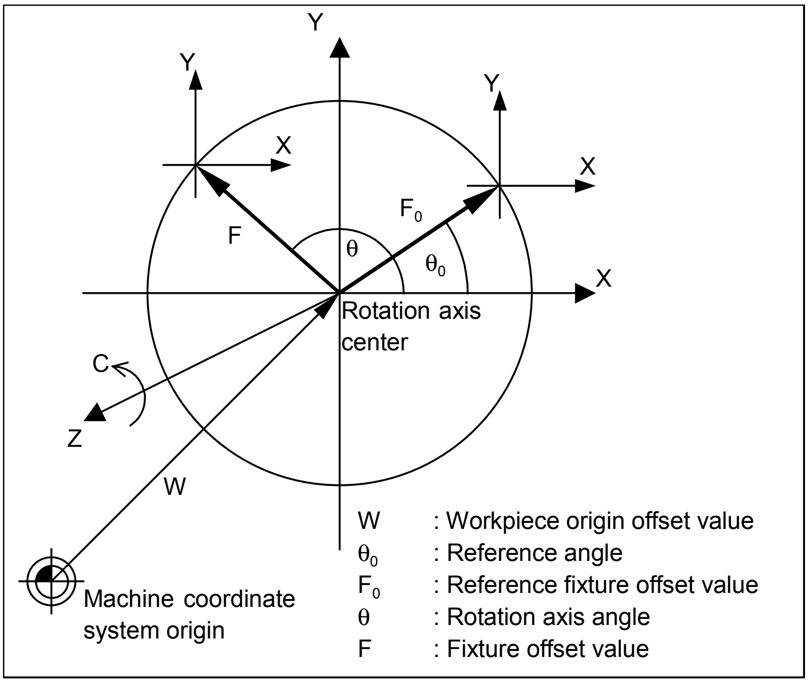


Fig.14.13 (a) Fixture offset

Format

- Fixture offset command

G54.2 Pn ;
n : Reference fixture offset value number (1 to 8)

- Fixture offset cancel command

G54.2 P0 ;

Explanation

- Fixture offset command

When command G54.2 Pn is specified, a fixture offset is calculated from the rotary axis angular displacement and the data of n. The fixture offset becomes valid.
If n is set to 0, the fixture offset becomes invalid.

- When a move command is specified for a rotation axis in G54.2 mode

When a move command is specified for a rotation axis related to fixture offset in G54.2 mode, vector calculation is performed using the coordinates of the rotation axis at the end point of the block, and movement is made to the specified position pointed to by the vector on the workpiece coordinate system.

- Operation at reset

The KWZ bit (bit 6 of parameter 2409) determines whether the fixture offset is canceled at reset.
When the bit is set to 1, the vector before the reset is retained.
When the bit is set to 0, the vector is cleared.
When the vector is cleared, the machine does not move according to the vector cleared upon a reset.

- Data setting

(1) Setting a group of three parameters which specify one rotation axis and two linear axes constituting the plane of rotation
(Parameter No.6068 to 6076)
In each group, specify the number of the rotation axis as the first parameter and the numbers of the linear axes as the second and third parameters. The rotation in the normal direction about the rotation axis must agree with the rotation from the positive side of the linear axis set as the second parameter to the positive side of the linear axis set as the third parameter.

Example) Suppose that a machine has four axes, X, Y, Z, and C. The X-, Y-, and Z-axes form a right-handed coordinate system. The C-axis is a rotation axis. When viewed from the positive side of the Z-axis, a rotation in the normal direction about the C-axis is treated as the counterclockwise rotation around the Z-axis. For this machine, specify the parameters as follows :

First parameter	:	4 (C-axis)
Second parameter	:	1 (X-axis)
Third parameter	:	2 (Y-axis)

Up to three groups of parameters can be set. In calculation of the fixture offset, the data of the rotation axis specified in the first group is calculated first. Then, the data of the second and third groups are calculated.
If a machine has two or more rotation axes and the plane of rotation depends on the rotation about another rotation axis, the plane of rotation is set when the angular displacement about the rotation axis is 0 .

(2) Setting the reference angle of the rotation axis and the corresponding reference fixture offset
Set the reference angle of the rotation axis and the fixture offset that corresponds to the reference angle.
Set the data on the fixture offset screen. Eight groups of data items can be specified.

(3) Setting a parameter for enabling or disabling the fixture offset of each axis (bit 0 (FAX) of parameter 1007)
For the axis for which the fixture offset is enabled, set the parameter to 1. This need not be specified for a rotation axis.

(4) Setting the type of fixture offset (bit 1 (FTP) of parameter 6004)
Specify whether to cause a movement according to the increment or decrement of the fixture offset vector when the vector changes (when G54.2 is specified or when a rotation axis movement occurs in the G54.2 mode).
When 0 is set, the movement is made. (The current position on the workpiece coordinate system does not change. The position on the machine coordinate system changes.)
When 1 is set, the movement is not made. (The current position on the workpiece coordinate system changes. The position on the machine coordinate system does not change.)

- Inputting and outputting the fixture offset

The setting of a program and external data can be input and output as described below :

(1) Setting the reference fixture offset by G10
G10 L21 Pn **P** ;
n : Number of fixture offset
P : Reference fixture offset or reference angle of each axis
This command sets the reference fixture offset or reference angle in a program.
When the command is executed in the G90 mode, the specified value is set directly.
When the command is executed in the G91 mode, the specified value plus a value set before the execution is set.

> **NOTE**
> The programmable data input function (G10) is required.

(2) Reading and writing the data by a system variable of a custom macro
The reference fixture offset or reference angle can be read and written by the following system variables :
15001+20*(n-1)+(m-1)
n : Number of fixture offset (1 to 8)
m : Axis number (1 to the number of controlled axes)

> **NOTE**
> The custom macro function is required.

(3) Reading and writing the data by the PMC window
The data can be read and written as a system variable of a custom macro by the PMC window.

> **NOTE**
> The NC window function and custom macro function are required.

(4) Outputting the data to an external device
By selecting <PUNCH> on the fixture offset screen, the data can be output to a Floppy Cassette or other external device via RS-232C.
The data is output in the G10 form without a program number.

> **NOTE**
> The reader/punch interface function and programmable data input function (G10) are required.

(5) Input from an external device
Data can be entered from an external device such as the Floppy Cassette via the RS-232C interface by selecting the read soft key on the fixture offset screen. Data can be entered in the G10 format that assigns no program number.

> **NOTE**
> The reader/punch interface function and programmable data input function (G10) are required.

- Calculating the Fixture Offset

(1) Relationship between the rotation axis and linear axes

First group	: 4(B-axis) , 3(Z-axis) , 1(X-axis)
Second group	: 5(C-axis) , 1(X-axis) , 2(Y-axis)
Third group	: 0, 0, 0

(2) Reference angle and reference fixture offset
X : F_{0X}
Y : F_{0Y}
Z : F_{0Z}
B : θ_0
C : ϕ_0

Then, let O, W, F0, FA, and F be as follows :
O : Rotary table center
W : Workpiece origin offset value
F_0 : Fixture offset value when $B = \theta_0$ and $C = \phi_0$
F_A : Fixture offset value (F_{AX},F_{AY},F_{AZ}) when B = 0 and C = 0
F : Fixture offset value (F_X,F_Y,F_Z) when $B = \theta$ and $C = \phi$

Then, the following expression is used for fixture offset calculation.

$$\begin{bmatrix} F_{AX} \\ F_{AY} \\ F_{AZ} \end{bmatrix} = \begin{bmatrix} \cos(-\theta_0) & 0 & \sin(-\theta_0) \\ 0 & 1 & 0 \\ -\sin(-\theta_0) & 0 & \cos(-\theta_0) \end{bmatrix} \begin{bmatrix} \cos(-\phi_0) & -\sin(-\phi_0) & 0 \\ \sin(-\phi_0) & \cos(-\phi_0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} F_{0X} \\ F_{0Y} \\ F_{0Z} \end{bmatrix}$$

$$\begin{bmatrix} F_X \\ F_Y \\ F_Z \end{bmatrix} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} F_{AX} \\ F_{AY} \\ F_{AZ} \end{bmatrix}$$

Limitation

- When data is modified in G54.2 mode

If a modification is made to a parameter or reference fixture offset in G54.2 mode, the modification becomes effective starting with the next buffered block.

- Movement due to a fixture offset vector change

It depends on the current continuous-state code of the 01 group whether a change in the fixture offset vector causes a movement. If the system is in a mode other than the G00 or G01 mode (G02, G03, etc.), the movement is made temporarily in the G01 mode.

- When a rotation axis is manually adjusted

When the automatic operation is stopped using the SBK stop function during the G54.2 mode and a manual movement is made about the rotation axis, the fixture offset vector does not change. The vector is calculated when a rotation axis command or G54.2 is specified for automatic operation or MDI operation.

If a manual intervention is made when bit 3 (ABS) of parameter 2409 is set to 1 and the manual absolute switch is on, a rotation axis command in the incremental mode (G91 mode) calculates the vector using the coordinates in which the manual intervention is not reflected.

Example)

```
    N1 G90 G00 C10.0 ;
    N2 G54.2 P1 ;
```

After these commands are executed, a manual intervention is made while the manual absolute switch is on. A movement of +20.0 is made about the C-axis.

```
    N3 G91 C30.0 ;
```

If this command is specified after the operation is resumed, the coordinate of the C-axis on the workpiece coordinate system becomes 60.0. When the fixture offset is calculated, the coordinate of the C-axis is assumed to be 40.0.

If the ABS bit (bit 3 of parameter 2409) is set to 0 or the G90 mode is selected when N3 is specified, the coordinate of the C-axis is assumed to be 40.0(30.0+10.0), which is the specified value, in calculation.

- When compensation is applied to a rotation axis

In calculation of the fixture offset, the coordinate of the rotation axis on the workpiece coordinate system is used. If a tool offset or another offset is applied, the coordinate before the offset is used. If the mirror image function or scaling function is executed, the coordinate before the function was executed is used.

- Command for suppressing fixture offset calculation

If the following commands are specified for the rotation axis in the G54.2 mode, the fixture offset vector is not calculated :

Command related to the machine coordinate system : G53
Command specifying a change of the workpiece coordinate system : G54 to G59, G54.1, G92, G52
Command specifying a return to the reference position : G27, G28, G29, G30, G30.1

- Rotation axis used for fixture offset

The rotation axis used for polar coordinate interpolation (G12.1) cannot be set as the rotation axis for the fixture offset.

Example

Parameter
Parameter 6068=4 (C-axis)
Parameter 6069=1 (X-axis)
Parameter 6070=2 (Y-axis)
Parameter 6071 to 6076=0
Parameter 1007#0(X)=1 (The offset is valid for the X-axis.)
1007#0(Y)=1 (The offset is valid for the Y-axis.)
6004#1=0 (When bit 1 of parameter 6004 is set to 1, the values in square brackets ([ ]) are calculated.)

Data of fixture offset 1 (n = 1)
C= 180.0 (reference angle)
X= -10.0
Y= 0.0

When these parameters and data are set, the machine operates as shown below :

Table14.13 Example of fixture offset

Coordinates / Program	Position on the workpiece coordinate system (ABSOLUTE)			Position on the machine coordinate system (MACHINE)			Fixture offset		
	X	Y	C	X	Y	C	X	Y	C
N1 G90 G00 X0 Y0 C90. ;	0.0	0.0	90.0	0.0	0.0	90.0	0.0	0.0	0.0
N2 G54.2 P1 ;	0.0 [0.0	0.0 -10.0	90.0 90.0]	0.0 [0.0	10.0 0.0	90.0 90.0]	0.0 [0.0	10.0 10.0	0.0 0.0]
N3 G01 X10. Y2. F100. ;	10.0	2.0	90.0	10.0	12.0	90.0	0.0	10.0	0.0
N4 G02 X2. Y10. R10. ;	2.0	10.0	90.0	2.0	20.0	90.0	0.0	10.0	0.0
N5 G01 X0 Y0 ;	0.0	0.0	90.0	0.0	10.0	90.0	0.0	10.0	0.0
:									

The values enclosed in brackets ([ ]) apply when bit 1 (FTP) of parameter No. 6004 is set to 1.

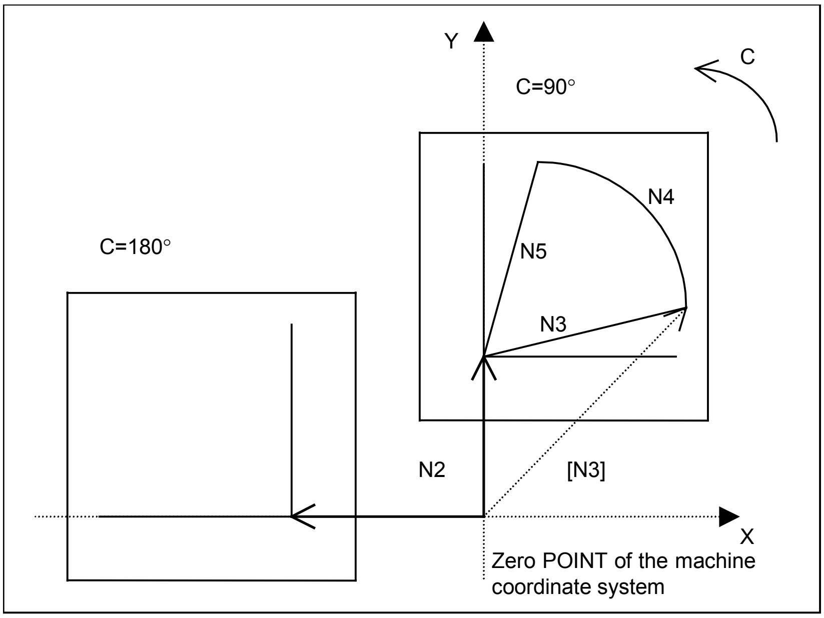


Fig.14.13 (b) Example of fixture offset

When G54.2 P1 is specified in the N2 block, the fixture offset vector (0, 10.0) is calculated. The vector is handled in the same way as the offset from the workpiece reference point. The current position on the workpiece coordinate system is (0, -10.0).

If bit 1 (FTP) of parameter 6004 is set to 0, the tool is moved according to the vector. The resultant position on the workpiece coordinate system is (0, 0), the position before the command is specified.

14.14 THREE-DIMENSIONAL CUTTER COMPENSATION

The three-dimensional cutter compensation function is used with machines that can control the direction of tool axis movement by using rotation axes (such as the B- and C-axes). This function performs cutter compensation by calculating a tool vector from the positions of the rotation axes, then calculating a compensation vector in a plane (compensation plane) that is perpendicular to the tool vector.
There are two types of cutter compensation :
Tool side compensation and leading edge compensation. Which is used depends on the type of machining.

14.14.1 Tool Side Compensation

Tool side compensation is a type of cutter compensation that performs three-dimensional compensation on a plane (compensation plane) perpendicular to a tool direction vector.

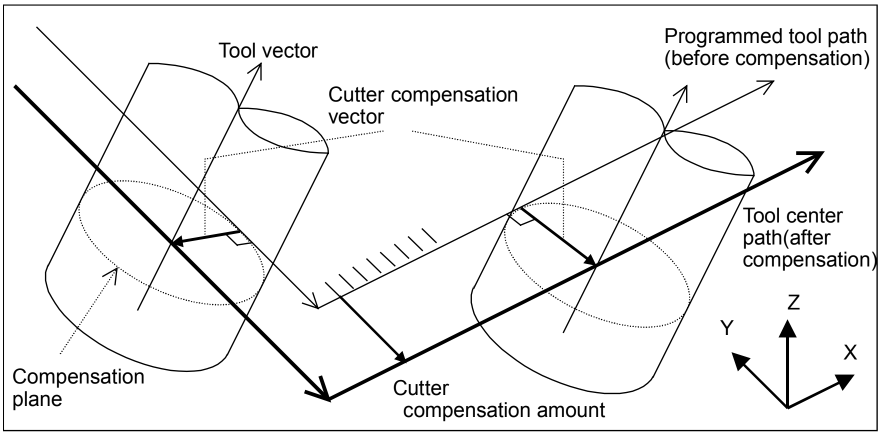

Tool vector
Programmed tool path (before compensation)
Cutter compensation vector
Tool center path(after compensation)
Z
Y
X
Compensation plane
Cutter compensation amount

Fig.14.14.1 (a) Tool side compensation

Format

- Tool side compensation(left)

G41.2 X_ Y_ Z_ D_ ;

If type C of cancellation operation is selected at start-up, never specify a move command such as X_Y_Z_ in the G41.2 block.

- Tool side compensation(right)

G42.2 X_ Y_ Z_ D_ ;

If type C of cancellation operation is selected at start-up, never specify a move command such as X_Y_Z_ in the G42.2 block.

- Tool side offset cancel

G40 X_ Y_ Z_ ;

Explanation

- Operation at compensation start-up and cancellation

(1) Type A
Type A operation is similar to cutter compensation as shown below.

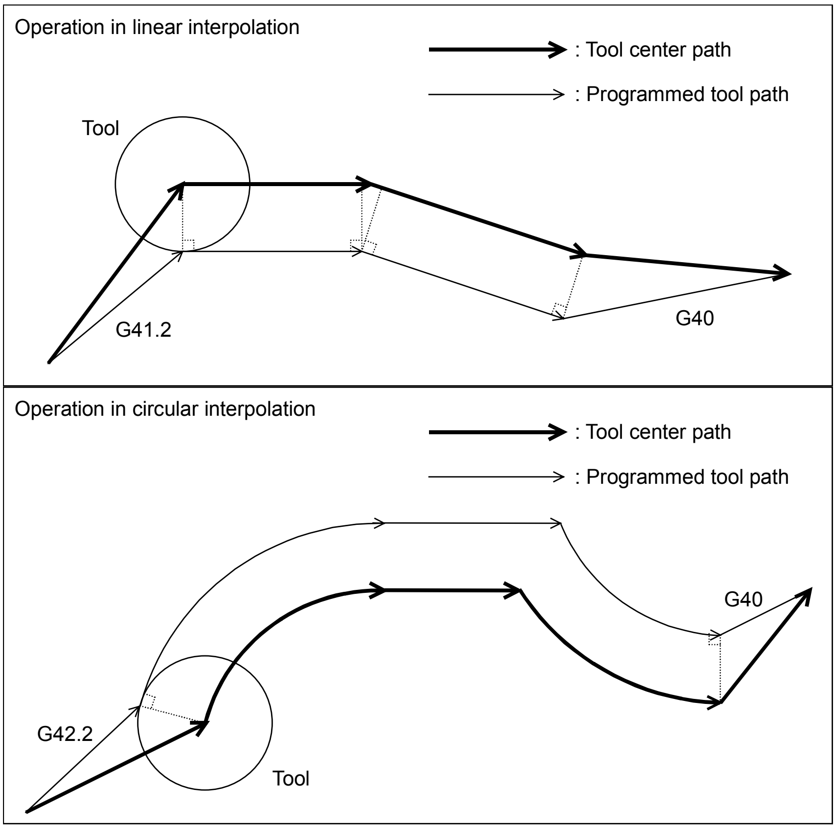
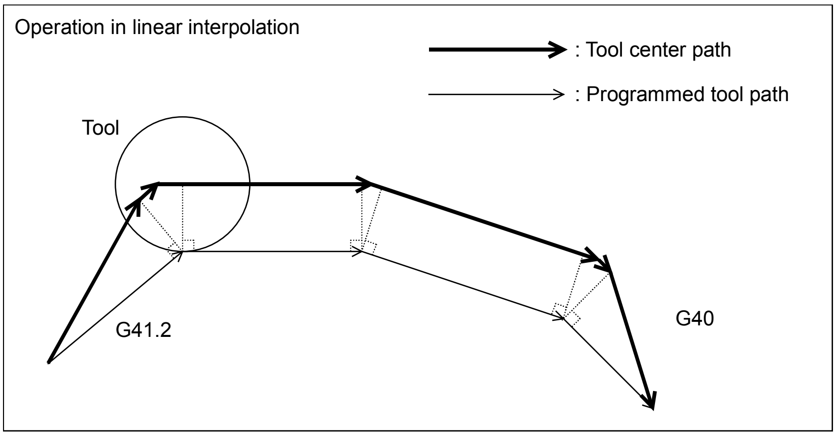


Fig.14.14.1 (b) Operation at compensation start-up and cancellation (Type A)

(2) Type B
Type B operation is similar to cutter compensation as shown below.



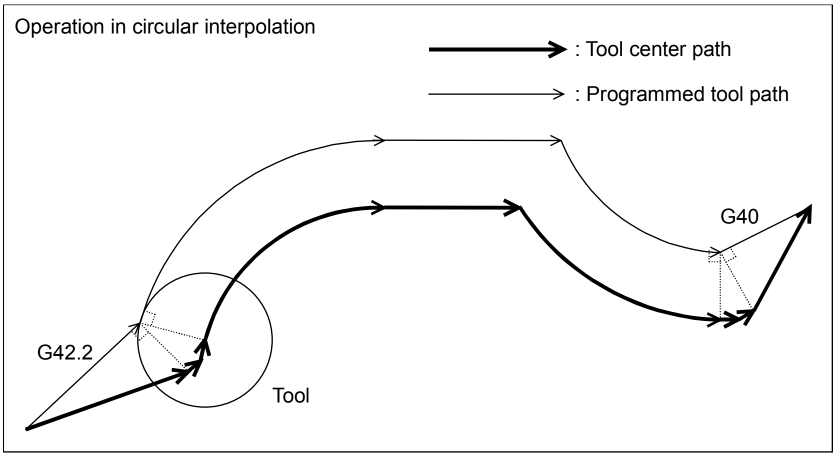


Fig.14.14.1 (c) Operation at compensation start-up and cancellation (Type B)

(3) Type C
As shown in the following figures, when G41.2, G42.2, or G40 is specified, a block is inserted which moves the tool perpendicularly to the movement direction specified in the next block by the distance of the tool radius.

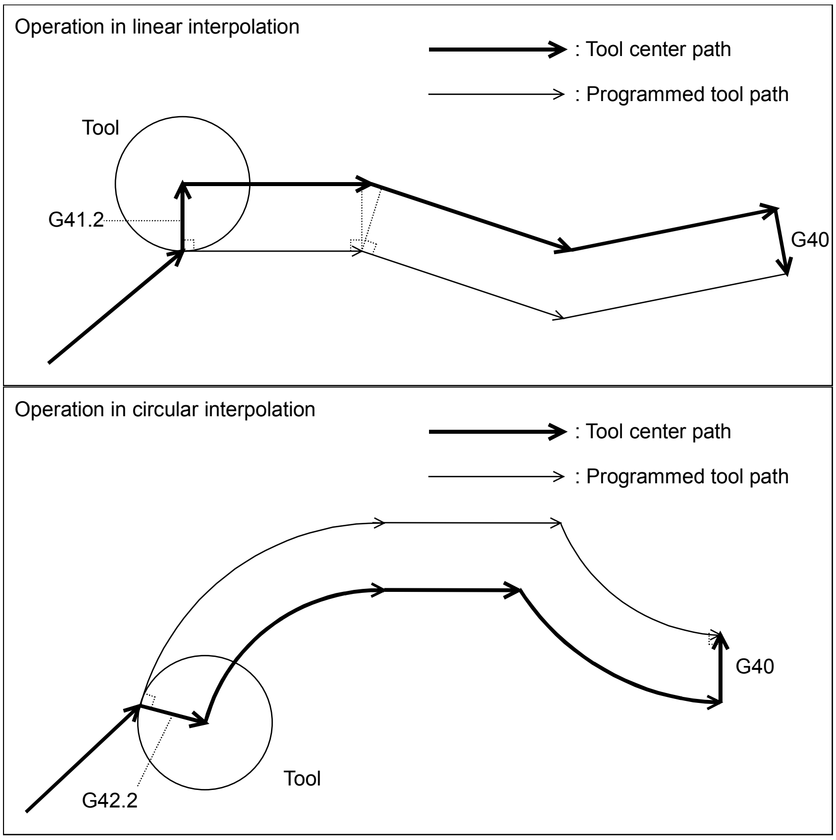


Fig.14.14.1 (d) Operation at compensation start-up and cancellation (Type C)

NOTE

For type C operation, the following conditions must be satisfied when tool side compensation is started up or canceled :

1. The block containing G40, G41.2, or G42.2 must be executed in the G00 or G01 mode.
2. The block containing G40, G41.2, or G42.2 must have no move command.
3. The block after the block containing G41.2 or G42.2 must contain a G00, G01, G02, or G03 move command.

- Operation in the compensation mode

Changing of offset directions and offset values, holding of vectors, interference checks, and so on are performed in the same way as for cutter compensation. G39 (corner rounding) cannot be specified. So, note the following :

(1) When the tool center path goes outside the programmed tool path at a corner, a linear movement block, instead of an arc movement block, is inserted to move the tool around the corner. When the tool center path goes inside the programmed tool path, no block is inserted.

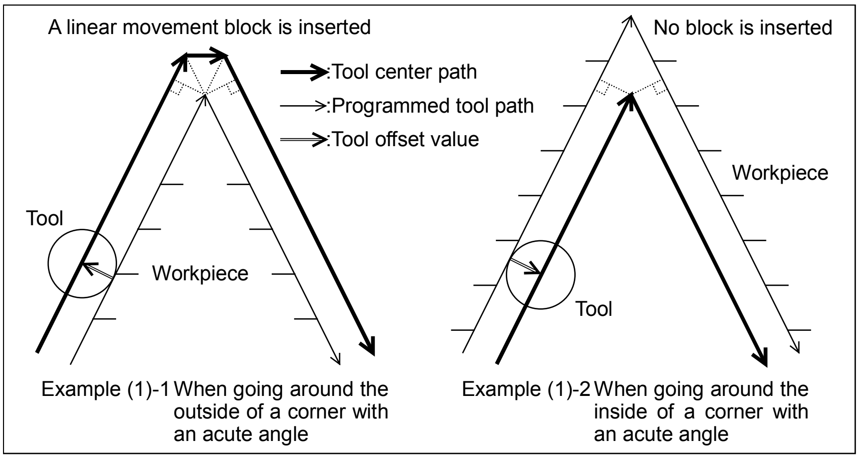

A linear movement block is inserted
:Tool center path
:Programmed tool path
:Tool offset value
Tool
Workpiece
Example (1)-1 When going around the outside of a corner with an acute angle
No block is inserted
Workpiece
Tool
Example (1)-2 When going around the inside of a corner with an acute angle

Fig.14.14.1 (e) Operation in the compensation mode (1)-1, 2

In the above examples, the term "inside" means that the tool center path is positioned inside the programmed tool path at a corner, and "outside" means that the tool center path is positioned outside the programmed tool path. In Example (1)-3, the relationship between the tool center path and the programmed tool path is the same as in Example (1)-1; the tool center path is positioned outside the programmed tool path. Example (1)-4 has the same relationship as Example (1)-2, where the tool center path is positioned inside the programmed tool path.

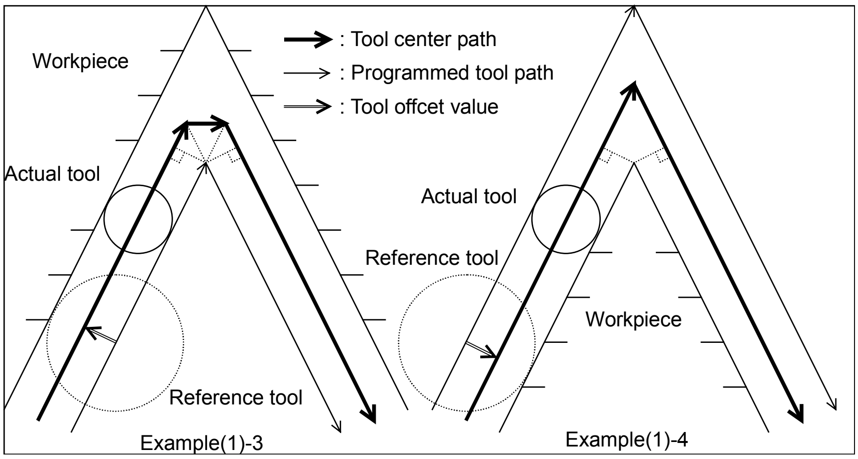
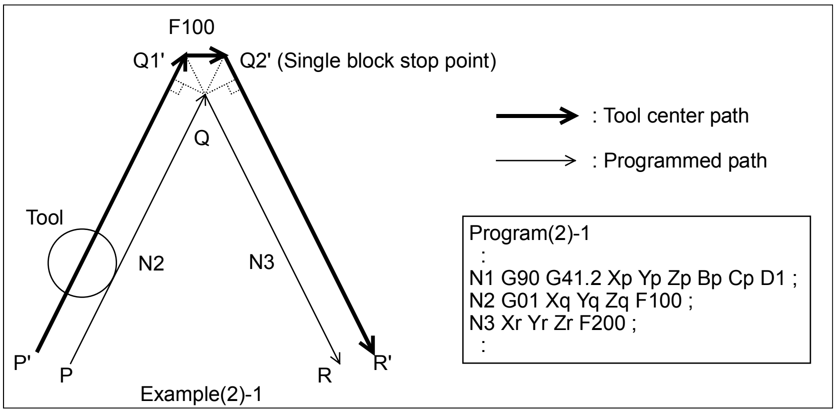


Fig.14.14.1 (f) Operation in the compensation mode (1)-3, 4

(2) When the tool moves at a corner, the feedrate of the previous block is used if the corner is positioned before a single-block stop point; if the corner is after a single-block stop point, the feedrate of the next block is used.

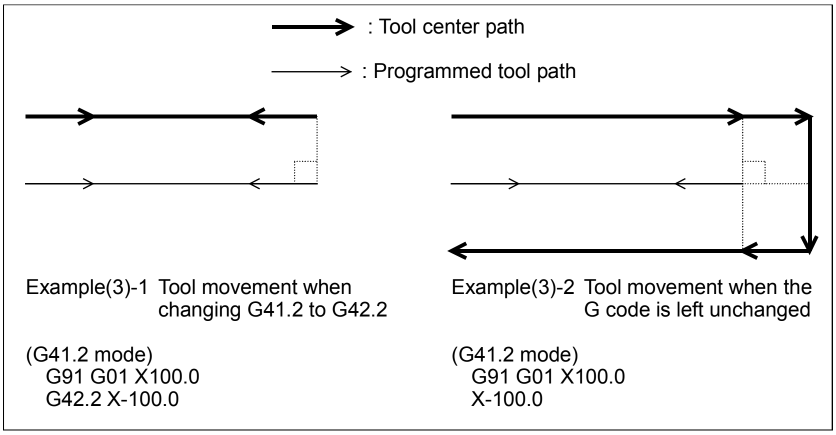


```
Program(2)-1
  :
N1 G90 G41.2 Xp Yp Zp Bp Cp D1 ;
N2 G01 Xq Yq Zq F100 ;
N3 Xr Yr Zr F200 ;
  :
```

Fig.14.14.1 (g) Operation in the compensation mode (2)

In the above example, the single block stop point of N2 is Q2', so that the feedrates along paths P'-Q1' and Q1'-Q2' are the same, namely, F100.

(3) When a command is specified to make the tool retrace the path specified in the previous block, the tool path can match the locus of the previous block by changing the G code to change the offset direction. If the G code is left unchanged, the operation shown in Example (3)-2 results :



Example(3)-1 Tool movement when changing G41.2 to G42.2

```
(G41.2 mode)
   G91 G01 X100.0
   G42.2 X-100.0
```

Example(3)-2 Tool movement when the G code is left unchanged

```
(G41.2 mode)
   G91 G01 X100.0
   X-100.0
```

Fig.14.14.1 (h) Operation in the compensation mode (3)

(4) Even when the tool movement changes linear to circular (helical), circular (helical) to linear, or circular (helical) to circular (helical), the start, end, and center points of a circular (helical) movement are projected on the compensation plane that is perpendicular to the tool axis, a compensation vector is calculated for the plane, then the vector is added to the originally specified position to obtain the command position. Then the tool is moved linearly or circularly (helically) to the obtained command position. In this case, the tool moves circularly (helically) in the currently selected plane. The tool does not move circularly (helically) in the compensation plane. Therefore, when the tool compensation is for circular movement, the compensation plane must be the XY, YZ, or ZX plane.

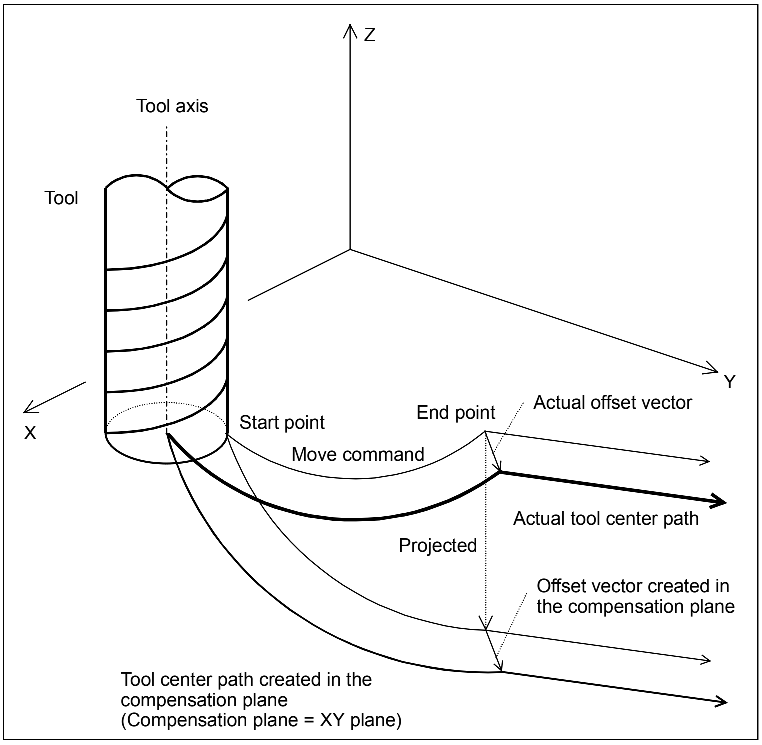

Z
Tool axis
Tool
X
Y
Start point
End point
Actual offset vector
Move command
Actual tool center path
Projected
Offset vector created in the compensation plane
Tool center path created in the compensation plane
(Compensation plane = XY plane)

Fig.14.14.1 (i) Operation in the compensation mode (4)

- Compensation vector calculation

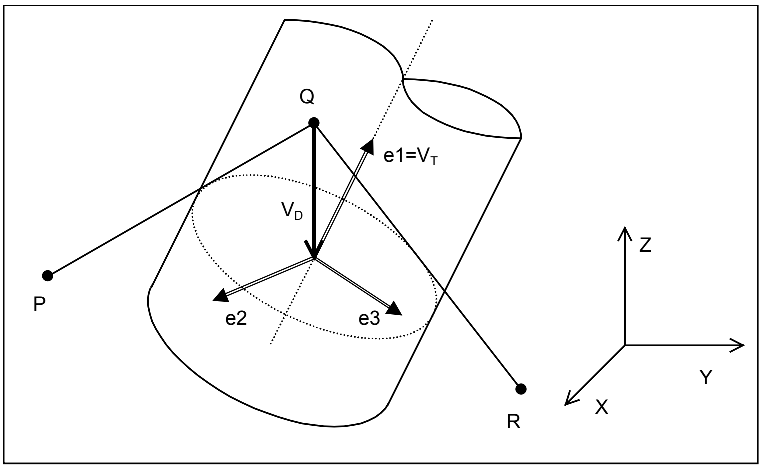


Fig.14.14.1 (j) Compensation vector calculation

In above figure, cutter compensation vector VD at point Q is calculated as follows :

(1) Calculating the tool vector (VT)

(2) Calculating the coordinate conversion matrix (M)
Coordinate systems are defined as follows :
- Coordinate system C1 : {O; X, Y, Z}
Cartesian coordinate system whose fundamental vectors are the following unit vectors along the X-, Y-, and Z-axes :
(1, 0, 0)
(0, 1, 0)
(0, 0, 1)

- Coordinate system C2 : {O; e2, e3, e1}

Cartesian coordinate system whose fundamental vectors are the following unit vectors :

e2
e3
e1

where, e2, e3, and e1 are defined as follows :

$e1 = V_T$

e2 = b2 / |b2| , b2 = a2 - (a2,e1)- e1

e3 = b3 / |b3| , b3 = a3 - (a3,e1)- e1 - (a3,e2)- e2

a2 is an arbitrary vector linearly independent of e1, and a3 is an arbitrary vector linearly independent of e2 and e1.

The coordinate conversion matrix M from coordinate system C1 to C2, and the coordinate conversion matrix M^{-1} from coordinate system C2 to C1 are expressed as :

$$M = \begin{bmatrix} e2 \\ e3 \\ e1 \end{bmatrix}, \quad M^{-1} = \left({}^{t}e2 \quad {}^{t}e3 \quad {}^{t}e1 \right)$$

(3) Converting coordinates from coordinate system C1 to coordinate system C2

The coordinates of the start and end points P and Q of a block and coordinates of the end point R of the next block in coordinate system C1 are converted to coordinates P', Q', and R' in coordinate system C2, respectively, by using the following expressions :

$$P' = MP$$
$$Q' = MQ$$
$$R' = MR$$

(4) Calculating the intersection vector (V_D') in the compensation plane

{O; e2, e3}

In the coordinates in coordinate system C2 obtained in (3), two components (the e1 component, the component of the tool direction, is excluded) are used to calculate intersection vector V_D' in the compensation plane.

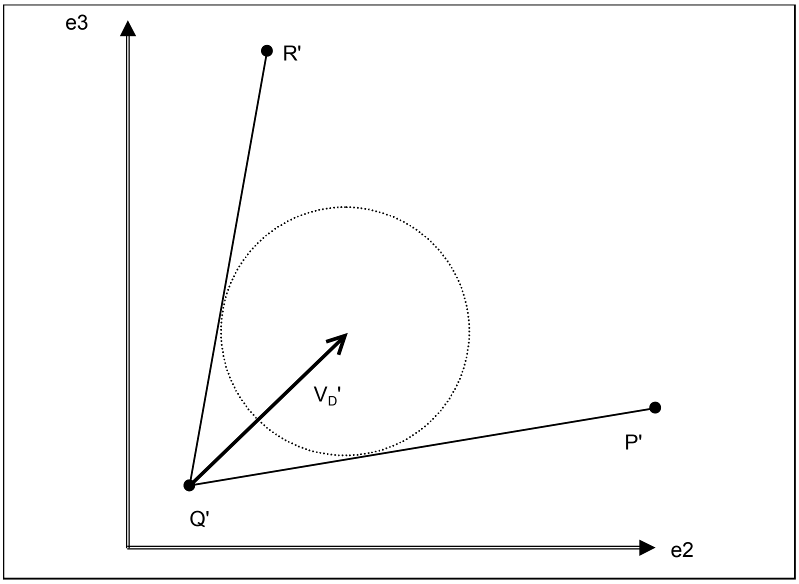


Fig.14.14.1 (k) Compensation vector calculation

The e1 component of V_D' is assumed to be always 0. The calculation is similar to the calculation of cutter compensation C. Although one vector is obtained in this example, up to four vectors may be calculated.

If the difference between the e2 and e3 components (in the compensation plane) between two points is smaller than the value set in parameter No. 6114 in intersection vector calculation, the block is assumed to specify no movement. In this case, intersection calculation is performed using the coordinates of one block ahead.

(5) Converting the intersection vector from coordinate system C2 to coordinate system C1

From the following expression, vector V_D' in coordinate system C2 is converted to vector V_D in coordinate system C1 :

$$V_D = M^{-1} V_{D'}$$

Vector V_D is the compensation vector in the original XYZ coordinate system.

- Calculation used when the compensation plane is changed

(1) When a rotation axis and linear axis are specified at the same time

When a rotation axis and linear axis are specified in the same block in the G41.2 or G42.2 mode (the compensation plane changes frequently), the cutter compensation vector is calculated using the coordinates of the rotation axis at each point at which the vector is obtained.

<Example>

```
G90 G00 X0 Y0 Z0 B0 C0 ;
G01 F1000 ;
N1 G42.2 Xp Yp Zp Bp Cp D1 ;
N2 Xq Yq Zq Bq Cq ;
N3 Xr Yr Zr Br Cr ;
N4 Xs Ys Zs Bs Cs ;
:
```

Vector calculation at the end point (Q) of block N2

- The tool vector (V_T) and coordinate conversion matrix (M) are calculated using the coordinates (Bq, Cq) of the rotation axis at point Q.
- The cutter compensation vector is calculated using the resultant coordinates into which three points, P, Q, and R, are converted by matrix M.

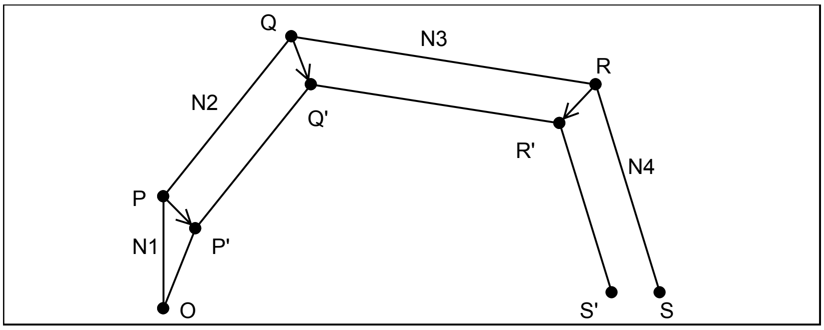


Fig.14.14.1 (l) When a Rotation Axis and Linear Axis Are Specified at the Same Time

(2) When a rotation axis is specified alone

When a rotation axis is specified alone in the G41.2 or G42.2 mode (the compensation plane changes), the cutter compensation vector is calculated as follows :

<Example>

```
G90 G00 X0 Y0 Z0 B0 C0 ;
G01 F1000 ;
N1 G42.2 Xp Yp Zp D1 ;
N2 Xq Yq Zq ;
N3 Br Cr ;
N4 Xs Ys Zs ;
:
```

Vector calculation at the end point (Q) of block N2

- The tool vector (V_T) and coordinate conversion matrix (M_{N2}) are calculated using the coordinates (B=0, C=0) of the rotation axis at point Q.
- The cutter compensation vector (V_{N2}) is calculated using the resultant coordinates into which three points, P, Q, and S, are converted by matrix M_{N2}.

Vector calculation at the end point of block N3

- A vector in the end of the block N2 is used as a vector of the block N3.

  $V_{N3} = V_{N2}$

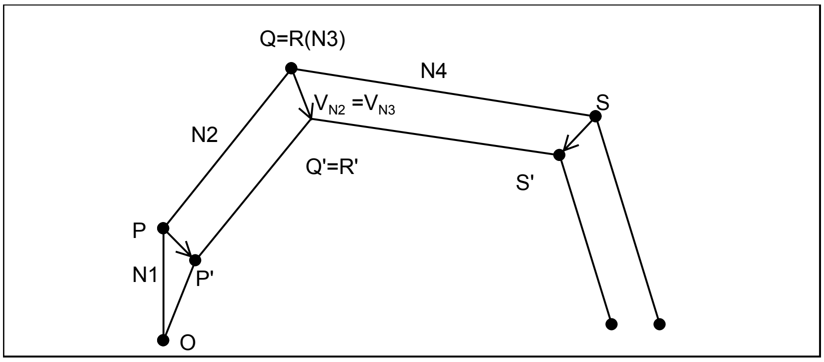


Fig.14.14.1 (m) When a rotation axis is specified alone

- Interference check made when the compensation plane is changed

An interference check is made when the compensation plane (plane perpendicular to a tool direction vector) is changed.

<Example>

If the program below is executed, a PS0272 alarm (overcutting due to offsetting) is issued from N4.

```
O100 F3000
N1 G90 G00 X0 Y0 Z0 A-46 C180
N2 G41.2 D1
N3 G01 X100
N4 Y-200 Z-200
N5 A45
N6 Y-400 Z0
N7 X0
N8 Y-200 Z-200
N9 A-46
N10 Y0 Z0
N11 G40
M30
```

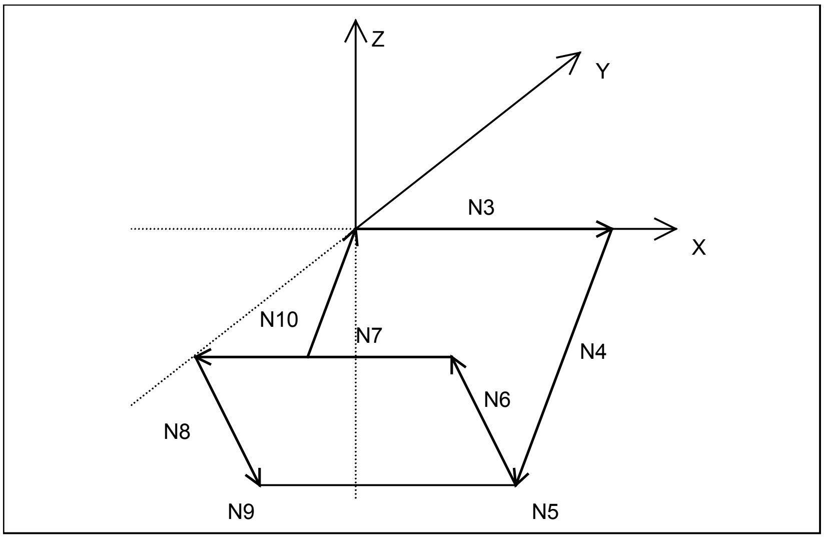


Fig.14.14.1 (n) Conceptual Diagram

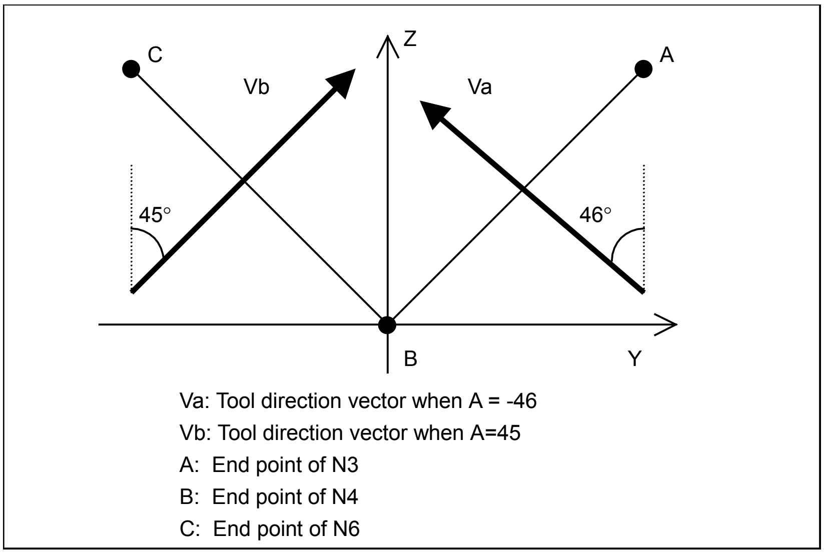

Z
C
A
Vb
Va
45°
46°
B
Y
Va: Tool direction vector when A = -46
Vb: Tool direction vector when A=45
A: End point of N3
B: End point of N4
C: End point of N6

Fig.14.14.1 (o) Tool Direction Vector

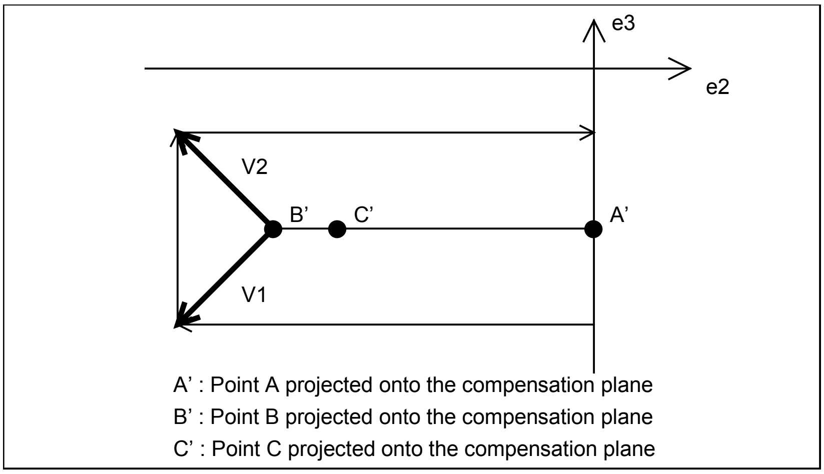

e3
e2
V2
B'
C'
A'
V1
A' : Point A projected onto the compensation plane
B' : Point B projected onto the compensation plane
C' : Point C projected onto the compensation plane

Fig.14.14.1 (p) Compensation Vector at the End Point (Point B) of N4 (on Compensation Plane)

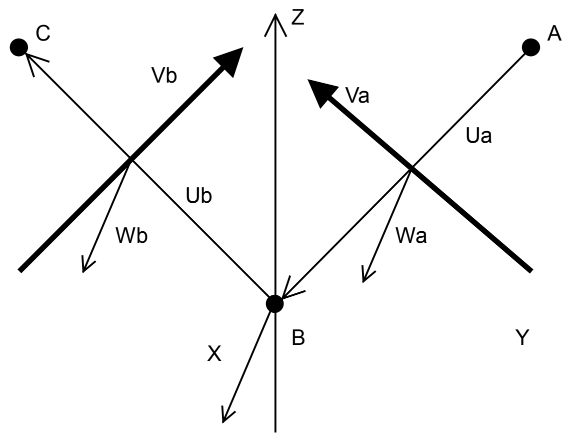
The move direction of A'B' is opposite to that of B'C', so that two compensation vectors, V1 and V2, are produced at point B' (end point of N4). There is a possibility of overcutting in this case, so an alarm (PS0272) is issued from N4.

(1) Conditions for issuing the interference alarm

Suppose that the tool direction vector changes considerably from one block to another due to a move command for a rotation axis. In this case, an interference alarm is assumed because compensation vectors are regarded as being generated in the wrong directions when the path angle difference on the compensation plane is large, even though the angle difference of the directions of compensation vectors to be generated by those blocks is small.

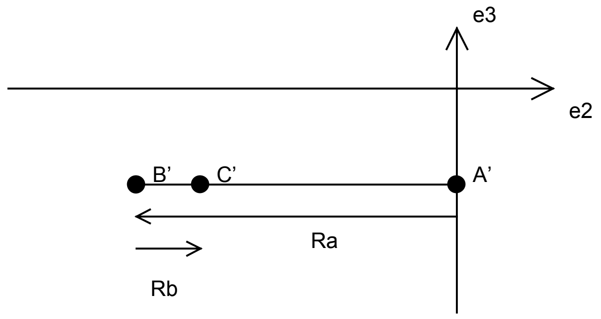
Here, the compensation plane is perpendicular to the tool direction (Va in Fig. 14.14.1(q)) of the first of the two blocks.

Specifically, the conditions below are used for issuing the alarm.



Ua: Vector AB
Ub: Vector BC
Va: Tool direction vector between A and B
Vb: Tool direction vector between B and C
Wa: Va × Ua
Wb: Vb × Ub
(Here, × represents an outer product operator.)

Fig.14.14.1 (q) Conceptual Diagram



A' : Point A projected onto the compensation plane
B' : Point B projected onto the compensation plane
C' : Point C projected onto the compensation plane
Ra: Vector A'B'
Rb: Vector B'C'

Fig.14.14.1 (r) Programmed Path (on the Compensation Plane) before and after the End Point (Point B) of N4

When all the following conditions are satisfied, an alarm (PS0272) Issued.

(1) The tool direction vector changes remarkably.

α : Angle for determination set in parameter No. 6261 (The default is 45°.)

$(Va,Vb) \le \cos(\alpha)$(Here, (Va,Vb) means an inner product.)

(2) The difference between the directions of the compensation vectors to be generated is small.

Wa : Direction of a compensation vector to be generated by the AB block.

Wb : Direction of a compensation vector to be generated by the BC block.

Wa = Va × Ua

Wb = Vb × Ub

(Wa,Wb) ≥ 0

(3) The path angle difference on the compensation plane is large.

(Ra,Rb) < 0

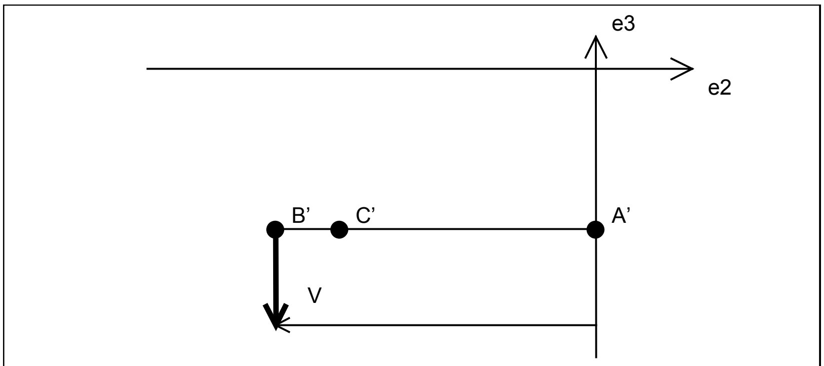
(2) Suppressing the issue of the alarm with a Q command

By inserting a Q command into a block that issued the alarm, the issue of the alarm can be suppressed.

(1) Q1 command

By inserting a Q1 command, a vertical vector is generated.

Example) N4 Y-200 Z-200 Q1



At point B', a vector (V) perpendicular to A'B' is generated.

Fig.14.14.1 (s) Q1 Command

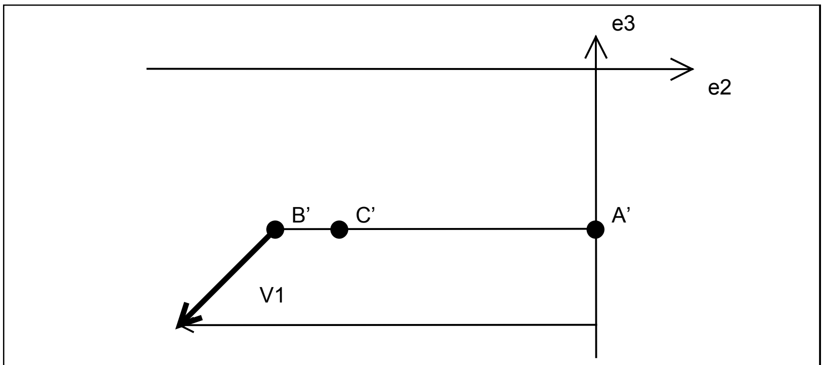
A vertical vector can also be generated by specifying G41.2 or G42.1 in the next block as indicated in the example below.

Example) N6 G41.2 Y-400 Z0

(2) Q2 command

With a program specifying a linear-to-linear connection, up to two compensation vectors are generated. In this case, the second vector is deleted by inserting a Q2 command. The Q2 command has no effect on circular interpolation.

Example) N4 Y-200 Z-200 Q2



The second vector (V2) is deleted, and only V1 is used as a compensation vector.

Fig.14.14.1 (t) Q2 Command

(3) Q3 command
By inserting a Q3 command, the issue of the alarm can be suppressed.
Example) N4 Y-200 Z-200 Q3

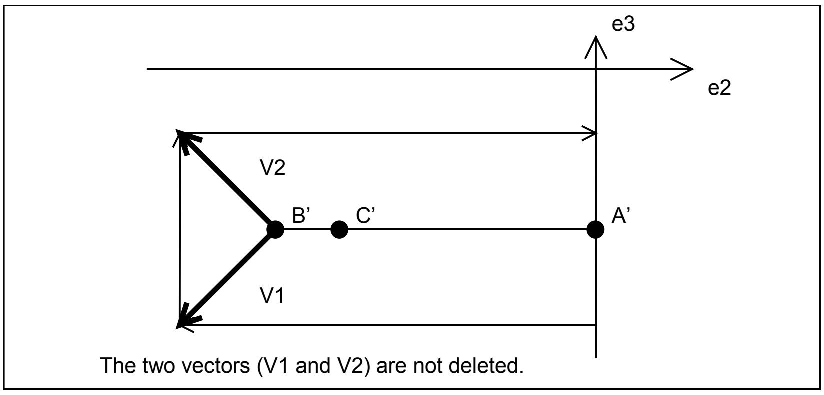


Fig.14.14.1 (u) Q3 Command

Limitation

- G41.2, G42.2, and G41.3 modes

G41.2, G42.2, G41.3, and G40 are continuous-state G codes that belong to the same group. Therefore, the G41.2, G42.2, and G41.3 modes cannot exist at the same time.

- Canned cycle command and reference position return command

Specify a canned cycle command and reference position return command in the compensation cancel mode (G40).

- Command that must not be specified in two or more successive blocks

In the compensation mode, never successively specify two or more blocks that involve no movement. Blocks involving no movement include :

- M05;	M code output
- S21;	S code output
- G04X1000;	Dwell
- G22X100000;	Machining area setting
- G10P01R100;	Offset value setting
- (G17)Z2000;	Movement not in the offset plane (movement in the tool axis direction)
- G90;,O10;,N20;	Block containing no move command

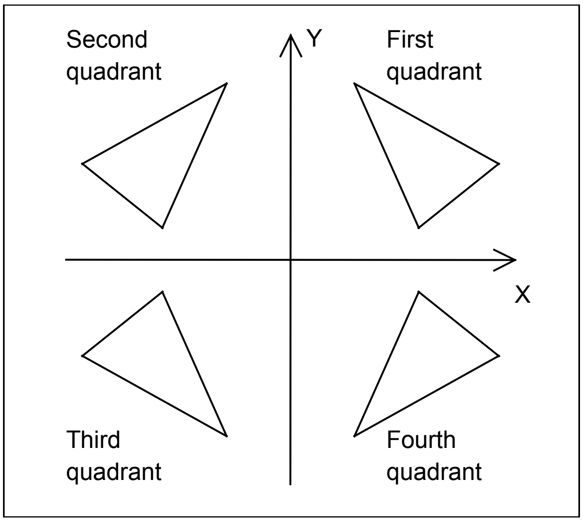
- Blocks regarded as involving no movement according to parameter 6114 (for tool side compensation only)

- Reset

Resetting the system in a compensation mode (G41.2, G42.2, or G41.3) always results in the cancellation mode (G40).

- Programmable mirror image

In mirror operation, there must be no conflict between the linear axes and rotation axes.



Example 1: XY Plane on a BC-Type Machine

	X	Y	Z	B	C
First quadrant	Normal	Normal	Normal	Normal	Normal
Second quadrant	Mirror	Normal	Normal	Normal	Mirror 90° about center
Third quadrant	Mirror	Mirror	Normal	Mirror 0° about center	Normal
Fourth quadrant	Normal	Mirror	Normal	Normal	Mirror 0° about center

Example 2: XY Plane on an AB-Type Machine (with A being the Master)

	X	Y	Z	A	B
First quadrant	Normal	Normal	Normal	Normal	Normal
Second quadrant	Mirror	Normal	Normal	Normal	Mirror 0° about center
Third quadrant	Mirror	Mirror	Normal	Mirror 0° about center	Mirror 0° about center
Fourth quadrant	Normal	Mirror	Normal	Mirror 0° about center	Normal

14.14.2 Leading Edge Offset

Leading edge offset is a type of cutter compensation that is used when a workpiece is machined with the edge of a tool. A tool is automatically shifted by a specified cutter compensation value on the line where a plane formed by a tool direction vector and tool movement direction intersects a plane perpendicular to the tool axis direction.

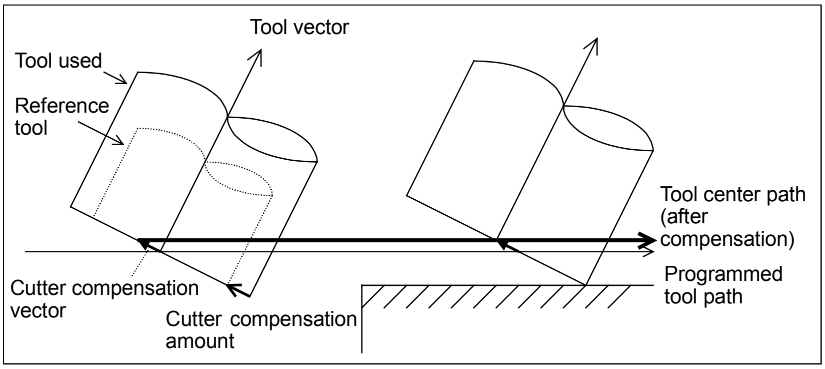

Tool vector
Tool used
Reference tool
Tool center path (after compensation)
Programmed tool path
Cutter compensation vector
Cutter compensation amount

Fig.14.14.2 (a) Reading edge offset

Format

- Leading edge offset

```
G41.3 D_ ;
```

- Leading edge offset cancel

```
G40 ;
```

> **NOTE**
> 1 G41.3 can be specified only in the G00 or G01 mode. In a block containing G41.3 or G40, only the addresses D, O, and N can be specified.
> 2 The block after a block containing a G41.3 command must contain a move command. In that block, however, tool movement in the same direction as the tool axis direction or the opposite direction cannot be specified.
> 3 No continuous-state G code that belongs to the same group as G00 and G01 can be specified in the G41.3 mode.

Explanation

- Operation at compensation start-up and cancellation

Unlike tool side compensation the operation performed at leading edge compensation start-up and cancellation does not vary. When G41.3 is specified, the tool is moved by the amount of compensation (V_C) in the plane formed by the movement vector (V_M) of the block after a G41.3 block and the tool vector (V_T) obtained at the time of G41.3 specification. The tool movement is perpendicular to the tool vector. When G40 is specified, the tool is moved to cancel VC. The following illustrates how the compensation is performed.

(1) When the tool vector is inclined in the direction the tool moves

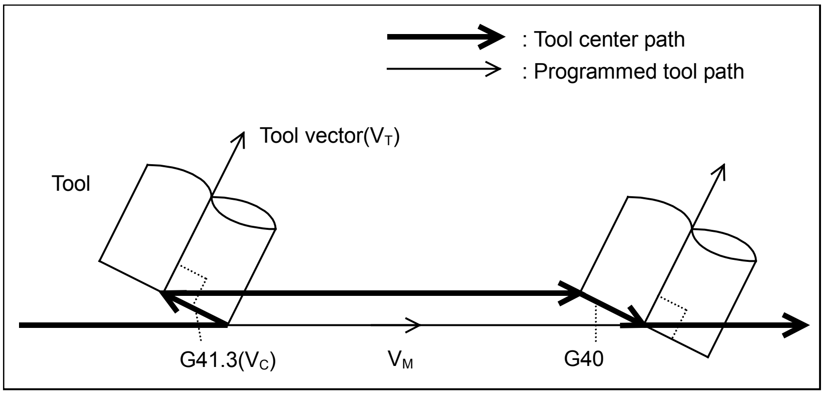


Fig.14.14.2 (b) When the tool vector is inclined in the direction the tool moves

(2) When the tool vector is inclined in the direction opposite to the direction the tool moves

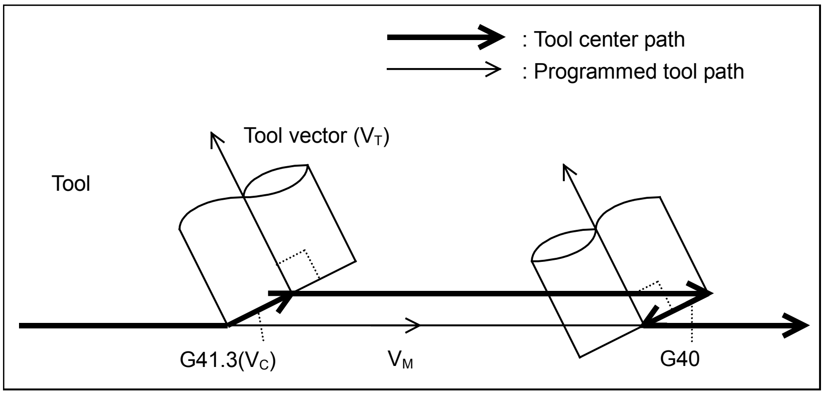
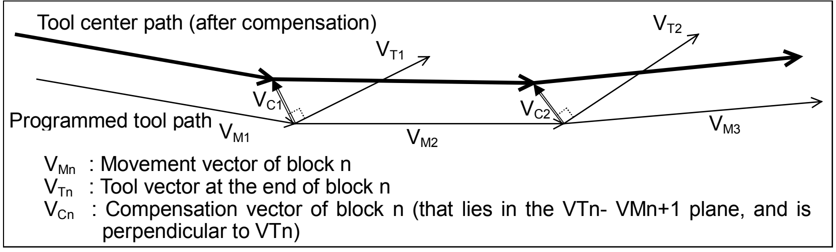


Fig.14.14.2 (c) When the tool vector is inclined in the direction opposite to the direction the tool moves

- Operation in the compensation mode

The tool center moves so that a compensation vector (V_C) perpendicular to the tool vector (V_T) is created in the plane formed by the tool vector (V_T) at the end point of each block and the movement vector (V_M) of the next block.



V_{Mn} : Movement vector of block n
V_{Tn} : Tool vector at the end of block n
V_{Cn} : Compensation vector of block n (that lies in the VTn- VMn+1 plane, and is perpendicular to VTn)

Fig.14.14.2 (d) Operation in the compensation mode

If a G code or M code that suppresses buffering is specified in the compensation mode, however, the compensation vector created immediately before the specification is maintained.

When a block involving no movement (including a block containing a move command for a rotation axis only) is specified, the movement vector of the block after the block involving no movement is used to create a compensation vector as shown below.

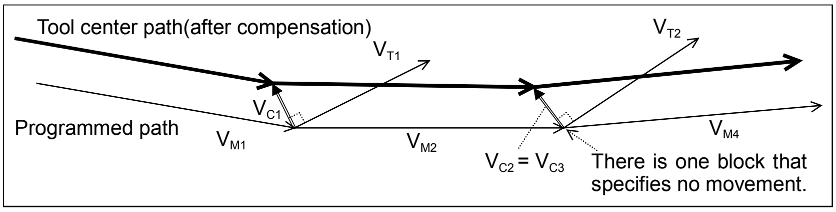

Tool center path(after compensation)
V_{T2}
V_{T1}
V_{C1}
Programmed path
V_{M1}
V_{M2}
V_{M4}
$V_{C2} = V_{C3}$
There is one block that specifies no movement.

Fig.14.14.2 (e) There is one block that specifies no movement.

If block 3 involves no movement, the compensation vector of block 2 (V_{C2}) is created so that it is perpendicular to V_{T2} and lies in the plane formed by the movement vector (V_{M4}) of block 4 and the tool vector (V_{T2}) at the end point of block 2.

NOTE

If two or more successive blocks involve no movement, the previously created compensation vector is maintained. However, such specification should be avoided.

- Block immediately before the offset cancel command (G40)

In the block immediately before the compensation cancel command (G40), a compensation vector is created from the movement vector of that block and the tool vector at the end point of the block as shown below :

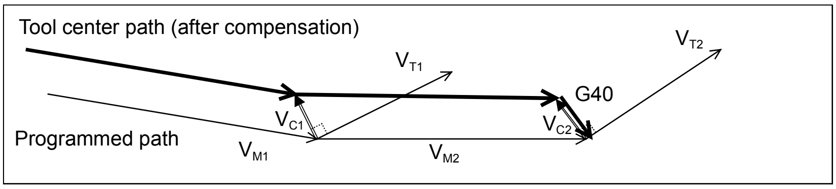


Fig.14.14.2 (f) Block Immediately before G40

The compensation vector (V_{C2}) of block 2 is created so that it is perpendicular to V_{T2} and lies in the plane formed by the tool vector (V_{T2}) at the end point of block 2 and the movement vector (V_{M2}) of block 2.

- Method of compensation vector calculation

In leading edge compensation, the compensation vector is calculated as follows :

(1) Tool vector(V_T)

(2) Movement vector

The movement vector (V_{Mn+1}) of block n+1 is obtained from the following expression :

$$V_{M_{n+1}} = \begin{bmatrix} X_{n+1} - X_n \\ Y_{n+1} - Y_n \\ Z_{n+1} - Z_n \end{bmatrix}$$

X_n : Absolute coordinate value of X axis at end point of block n

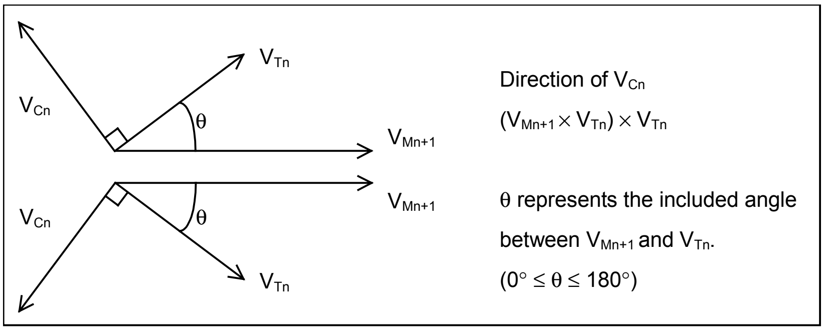
Y_n : Absolute coordinate value of Y axis at end point of block n

Z_n : Absolute coordinate value of Z axis at end point of block n

(3) Compensation vector

The direction of the compensation vector (V_{Cn}) of block n is defined as follows :

(1) $(V_{Mn+1,} V_{Tn}) > 0$ (0deg < θ < 90deg.)



Direction of V_{Cn}

$(V_{Mn+1} \times V_{Tn}) \times V_{Tn}$

θ represents the included angle between V_{Mn+1} and V_{Tn}.

($0° \leq \theta \leq 180°$)

Fig.14.14.2 (g) Direction of the compensation vector (1)

(2) $(V_{Mn+1},V_{Tn}) < 0$ (90deg < θ < 180deg.)

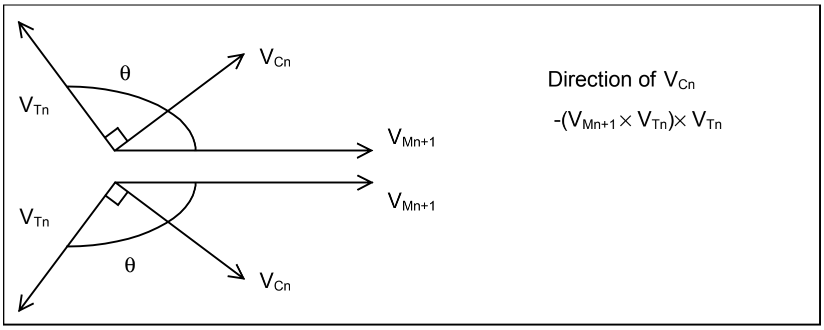


Fig.14.14.2 (h) Direction of the compensation vector (2)

The compensation vector (V_{Cn}) of block n is calculated from V_{Tn} and V_{Mn+1} as described below.

R= offset value

$$V_{T_n} = \begin{bmatrix} V_{TX} \\ V_{TY} \\ V_{TZ} \end{bmatrix}$$

$$V_{M_{n+1}} = \begin{bmatrix} V_{MX} \\ V_{MY} \\ V_{MZ} \end{bmatrix}$$

$$V = \begin{bmatrix} V_X \\ V_Y \\ V_Z \end{bmatrix} = \left(V_{M_{n+1}} \times V_{T_n}\right) \times V_{T_n}$$

$$= \begin{bmatrix} V_{TZ}\left(V_{MZ}V_{TX} - V_{MX}V_{TZ}\right) - V_{TY}\left(V_{MX}V_{TY} - V_{MY}V_{TX}\right) \\ V_{TX}\left(V_{MX}V_{TY} - V_{MY}V_{TX}\right) - V_{TZ}\left(V_{MY}V_{TZ} - V_{MZ}V_{TY}\right) \\ V_{TY}\left(V_{MY}V_{TZ} - V_{MZ}V_{TY}\right) - V_{TX}\left(V_{MZ}V_{TX} - V_{MX}V_{TZ}\right) \end{bmatrix}$$

Then,

(1) When(V_{Mn+1}, V_{Tn}) > 0 (0deg. < θ < 90deg.)

$$V_{C_n} = \frac{R}{\sqrt{V_{X^2} + V_{Y^2} + V_{Z^2}}} \begin{bmatrix} V_X \\ V_Y \\ V_Z \end{bmatrix}$$

(2) When(V_{Mn+1}, V_{Tn}) < 0 (90deg. < θ < 180deg.)

$$V_{C_n} = \frac{-R}{\sqrt{V_{X^2} + V_{Y^2} + V_{Z^2}}} \begin{bmatrix} V_X \\ V_Y \\ V_Z \end{bmatrix}$$

- Compensation performed when θ is approximately 0deg., 90deg., or 180deg.

When the included angle θ between V_{Mn+1} and V_{Tn} is regarded as 0deg., 180deg., or 90deg., the compensation vector is created in a different way. So, when creating an NC program, note the following points :

(1) Setting a variation range for regarding θ as 0deg., 180deg., or 90deg.

When the included angle (θ) between the tool vector (V_T) and the movement vector (V_M) becomes approximately 0deg., 180deg., or 90deg., the system regards θ as 0deg., 180deg., or 90deg., then creates a compensation vector which is different from the normal compensation vector. The variation range used for regarding θ as 0deg., 180deg., and 90deg. is or in parameter 6115.

For example, suppose that the angle set in parameter 6115 is Δθ.

(1) If $0 \leq \theta \leq \Delta\theta$, θ is regarded as 0deg.

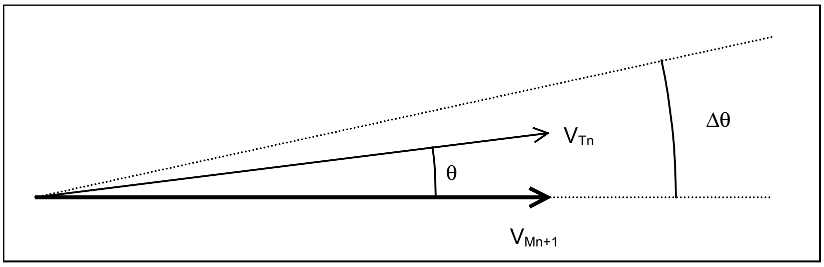


Fig.14.14.2 (i) Determination of θ=0deg.

(2) If $(180-\Delta\theta) \leq \theta \leq 180$, θ is regarded as 180deg.

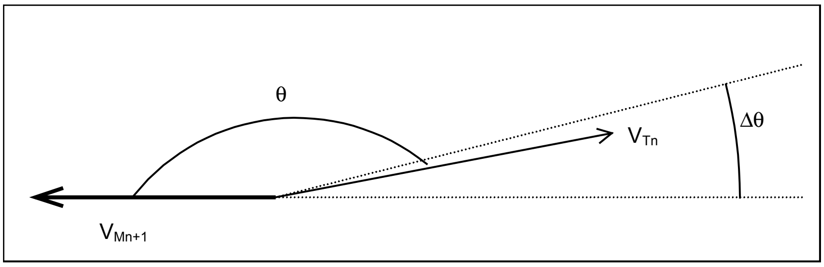


Fig.14.14.2 (j) Determination of θ=180deg.

(3) If $(90-\Delta\theta) \leq \theta \leq (90+\Delta\theta)$, θ is regarded as 90deg.

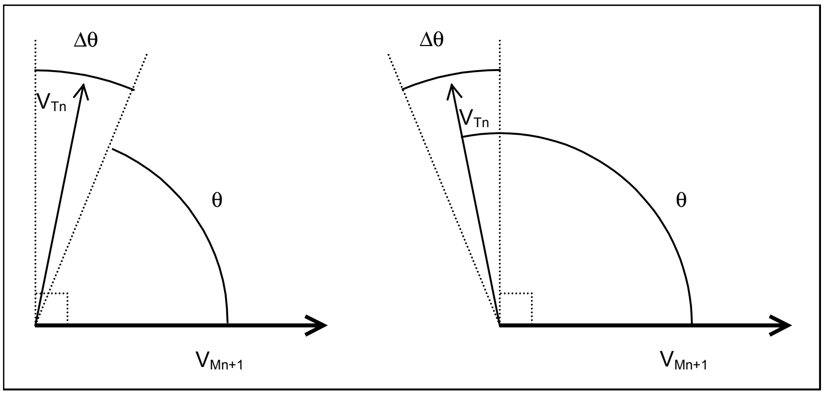


Fig.14.14.2 (k) Determination of θ=90deg.

(2) Compensation vector when θ is regarded as 0deg.or 180deg.

If θ is regarded as 0deg.or 180deg.when G41.3 is specified to start leading edge compensation, alarm PS998 is issued. This means that the tool vector of the current block and the movement vector of the next block must not point in the same direction or in opposite directions at start-up.

The previously created compensation vector is maintained other than at start-up at all times.

If the included angles between V_{T2} and V_{M3}, V_{T3} andV_{M4}, and V_{T4} and V_{M4}, and V_{T4} and V_{M5} are regarded as 05, the compensation vector V_{C1} of block 1 is maintained as the compensation vectors V_{C2}, V_{C3}, and V_{C4}, of blocks 2, 3, and 4, respectively.

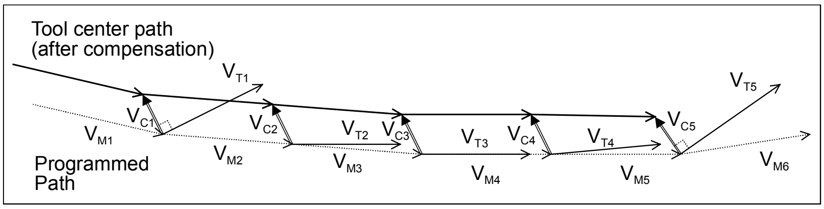


Fig.14.14.2 (l) When θ=0deg.Is Determined

If the included angles between V_{T2} and V_{M3}, V_{T3} andV_{M4}, and V_{T4} and V_{M5} are regarded as 180deg., the compensation vector V_{C1} of block 1 is maintained as the compensation vectors V_{C2}, V_{C3}, and V_{C4} of blocks 2, 3, and 4, respectively.

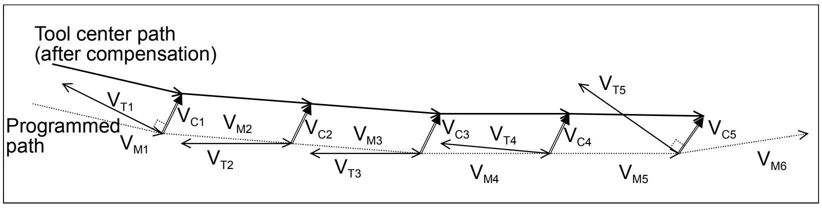


Fig.14.14.2 (m) When θ=180deg. Is Determined

(3) Compensation vector when θ is regarded as 90deg.

If the previous compensation vector (V_{Cn-1}) points in the opposite direction (($V_{Mn} \times V_{Tn-1}$) $\times V_{Tn-1}$ direction) to V_{Mn} with respect to VTn-1, the current compensation vector (V_{Cn}) is created so it points in the ($V_{Mn+1} \times V_{Tn}$) $\times V_{Tn}$ direction.

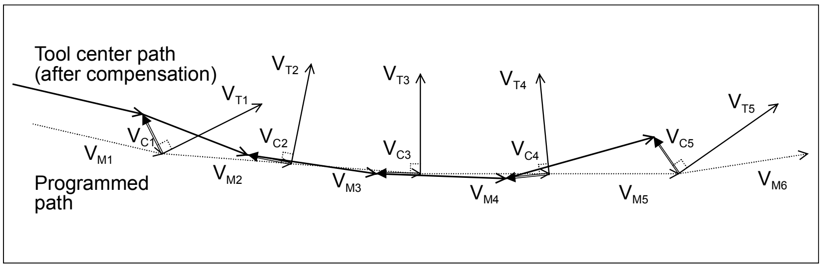


Fig.14.14.2 (n) When θ=90deg. Is Determined (1)

If the previous compensation vector (V_{Cn-1}) points in the same direction ($-(V_{Mn} \times V_{Tn-1}) \times V_{Tn-1}$ direction) as V_{Mn} with respect to V_{Tn-1} , the current compensation vector (V_{Cn}) is created so it points in the $-(V_{Mn+1} \times V_{Tn}) \times V_{Tn}$ direction.

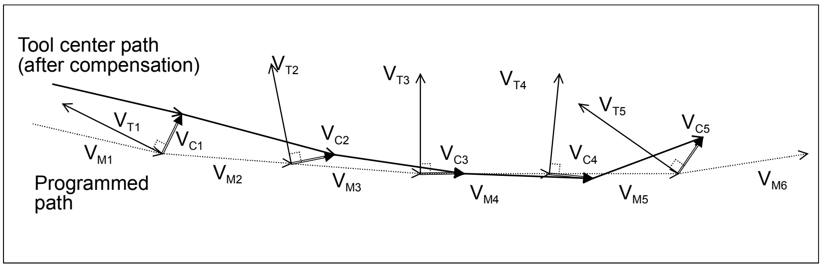


Fig.14.14.2 (o) When θ=90deg. Is Determined (2)

Limitation

- G41.2, G42.2, and G41.3 modes

G41.2, G42.2, G41.3, and G40 are continuous-state G codes that belong to the same group. Therefore, the G41.2, G42.2, and G41.3 modes cannot exist at the same time.

- Canned cycle command and reference position return command

Specify a canned cycle command and reference position return command in the compensation cancel mode (G40).

- Command that must not be specified in two or more successive blocks

In the compensation mode, never successively specify two or more blocks that involve no movement. Blocks involving no movement include :

- M05;	M code output
- S21;	S code output
- G04X1000;	Dwell
- G22X100000;	Machining area setting
- G10P01R100;	Offset value setting
- G90;,O10;,N20;	Block containing no move command

- Reset

Resetting the system in a compensation mode (G41.2, G42.2, or G41.3) always results in the cancellation mode (G40).

14.14.3 Three-dimensional Cutter Compensation at Tool Center Point

For machines with a rotation axis for rotating a tool, this function performs three-dimensional cutter compensation at the tool tip position if the program-specified point is specified with a pivot point.

When this function is used, the program-specified point (pivot point) is converted into a tool tip position (cutting point) and a three-dimensional cutter compensation vector is calculated for the latter position. Then, the program-specified point (pivot point) is compensated for with the three-dimensional cutter compensation vector.

If the tool side offset (G41.2/G42.2) of three-dimensional cutter compensation is performed, the operation of this function will be as follows:

(1) If parameter No. 6130 is 0 (conventional specification)
The three-dimensional cutter compensation vector is calculated at the program-specified point (pivot point).

(2) If parameter No. 6130 is not 0 (this function)
The three-dimensional cutter compensation vector is calculated at the tool tip position (cutting point).

Description

This function calculates a vector at the tool tip position for the three-dimensional cutter compensation function as described below.

(1) Convert the programmed coordinates from a program-specified point (pivot point) to a tool tip position (cutting point). Parameter No. 6130 is used to store the distance from the program-specified point (pivot point) to the tool tip position (cutting point).

(2) Calculate a three-dimensional cutter compensation vector at the tool tip position (cutting point).

(3) Add the cutter compensation vector to the program-specified point (pivot point).

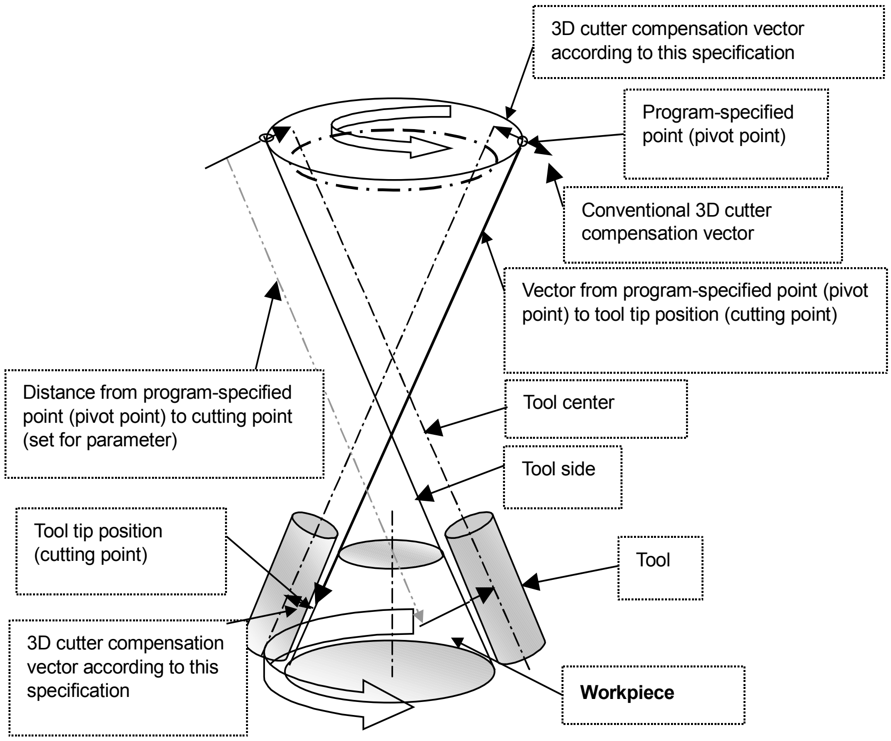

3D cutter compensation vector according to this specification
Program-specified point (pivot point)
Conventional 3D cutter compensation vector
Vector from program-specified point (pivot point) to tool tip position (cutting point)
Distance from program-specified point (pivot point) to cutting point (set for parameter)
Tool center
Tool side
Tool tip position (cutting point)
Tool
3D cutter compensation vector according to this specification
Workpiece

Fig.14.14.3 (a) Basic operation (for G42.2)

- Operation example

For a machine configuration in which the tool direction is along the Z-axis and the rotation axes are the B&C axes (Fig.14.14.3 (b))
Assume that
LC: Parameter (No. 6130) for the distance from the program-specified point (pivot point) to the tool tip position (cutting point)
b: B-axis specified value, c: C-axis specified value
Q=(Qx,Qy,Qz): Program-specified point (pivot point)
P, R: Program-specified points (pivot points) in the preceding and succeeding blocks
QT=(QTx,QTy,QTz): Tool position (tool tip position (cutting point)) resulting from conversion
PT,RT: Tool positions (tool tip positions (cutting points) in the preceding and succeeding blocks that result from conversion, then

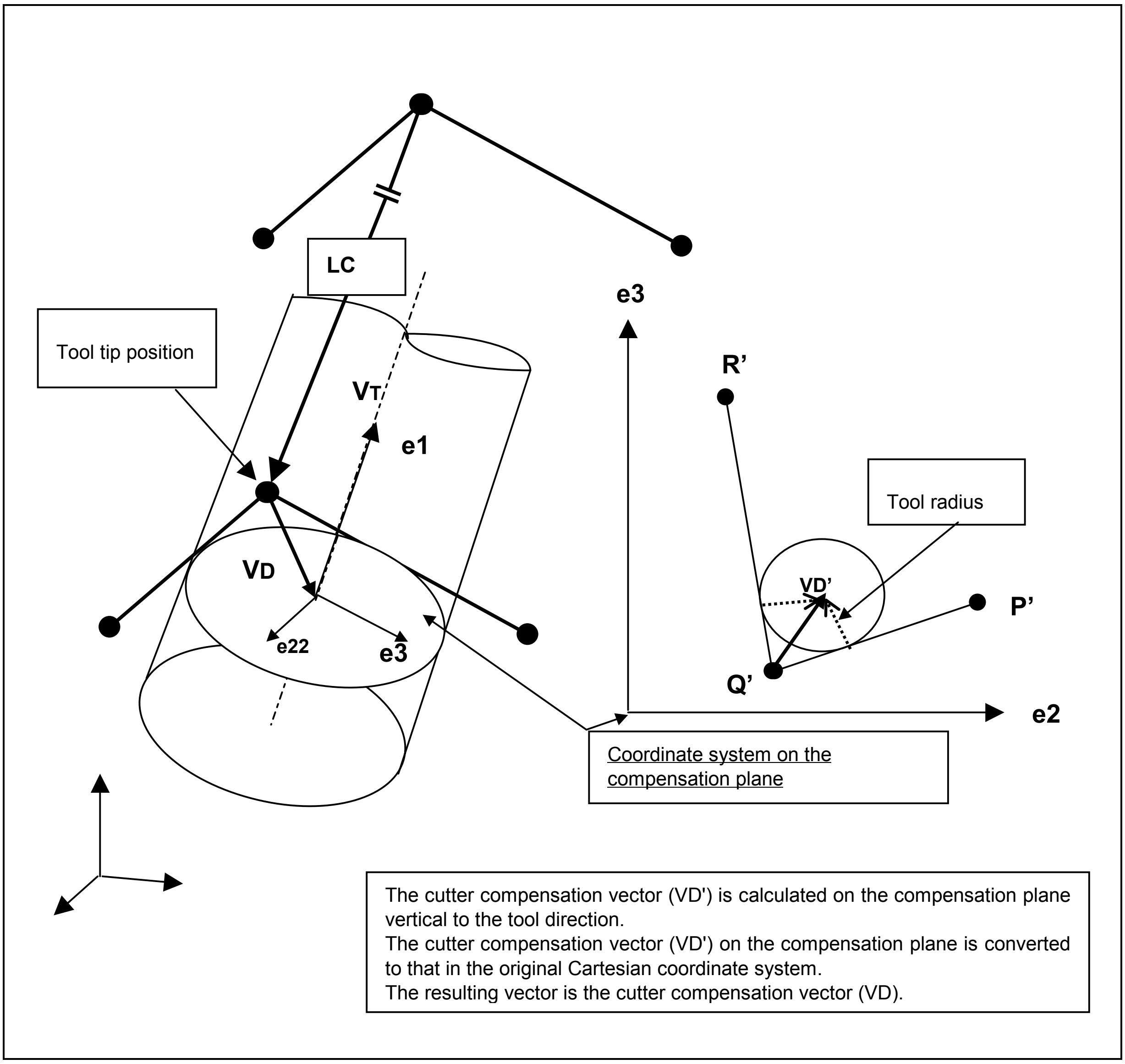
(1) Convert the program-specified points (pivot points) P, Q, and R to the tool tip positions (cutting points) PT, QT, and RT.

$$QTx = LC \times \sin(b) \times \cos(c) \qquad + Qx$$
$$QTy = LC \times \sin(b) \times \sin(c) \qquad + Qy$$
$$QTz = LC \times \cos(b) \qquad + Qz$$

(The same applies to PT and RT.)

(2) Calculate the cutter compensation vector VD with the tool tip positions (cutting points) PT, QT, and RT and the tool gradient VT.

(3) Add the cutter compensation vector VD to the program-specified point (pivot point) and set the result as the end point position.



The cutter compensation vector (VD') is calculated on the compensation plane vertical to the tool direction.
The cutter compensation vector (VD') on the compensation plane is converted to that in the original Cartesian coordinate system.
The resulting vector is the cutter compensation vector (VD).

Fig.14.14.3 (b) Calculation method

- Cautions

⚠ CAUTION

1 This function is not effective for leading edge offset.
2 With a command for a rotation axis only, this function does not calculate a cutter compensation vector.
3 This function cannot be used in three-dimensional coordinate conversion mode.
4 The cautions for the three-dimensional cutter compensation function also apply to this function

- Specification of parameters

The parameters used with this function are described below. Parameter numbers are enclosed in brackets [ ].

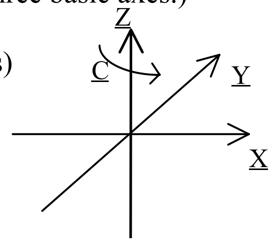
Relationships between rotation axes and rotation planes [6080 to 6089]

These parameters set the relationships between rotation axes and rotation planes. In general, the direction vector of a rotation axis has components in three directions. This function, however, can perform calculation on a rotation axis having a vector of components in any two directions.

(1) For a component in a single direction ... type 1
(The rotation axis rotates about any of the three basic axes.)

Example:

Axis number of the rotation axis = 5 (C-axis)
Axis number of linear axis 1 = 1 (X-axis)
Axis number of linear axis 2 = 2 (Y-axis)

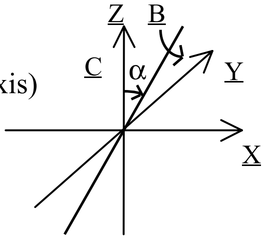


1) The rotation axis:
   - Rotates about the axis orthogonal to the plane formed by linear axes 1 and 2.
   - Assumes that the rotation from the forward direction of linear axis 1 to the forward direction of linear axis 2 is the forward rotation of the rotation axis.

(2) For components in two directions ... type 2
(The rotation axis rotates about an axis on the plane formed by any two of the three basic axes.)

Example:

Axis number of the rotation axis = 4 (B-axis)
Axis number of linear axis 1 = 1 (X-axis)
Axis number of linear axis 2 = 2 (Y-axis)
Axis number of linear axis 3 = 3 (Z-axis)
Angle = α



1) The linear axes 1, 2, and 3 are assumed to form a right-handed system in that order.
2) The angle:
   - Rotates on the plane formed by linear axes 3 and 1.
   - Assumes that the rotation from the forward direction of linear axis 3 to the forward direction of linear axis 1 is the forward rotation of the angle.
   - Assumed to be 0° when it matches the direction of linear axis 3.
3) When the angle is equal to 0°, the rotation axis:
   - Rotates about the axis orthogonal to the plane formed by linear axes 1 and 2.
   - Assumes that the rotation from the forward direction of linear axis 1 to the forward direction of linear axis 2 is the forward rotation of the rotation axis.
4) If 0 is set for linear axis 3, type 1 is assumed.

Up to two sets of such parameter settings can be specified. Thus, it is possible to compensate a slant rotary head controlled with two rotation axes. For the calculation of the compensation amount, calculation is performed on the first rotation axis, and using the results, calculation is performed on the second rotation axis. For two rotation axes, if the rotation plane varies due to the rotation of another rotation axis, set the rotation plane assumed when the position of the rotation axis is 0°. For a single rotation axis, set 0 for the second rotation axis.

Reference angles of rotation axes and directions of compensation vectors [6104 to 6107]

These parameters set the reference angles of rotation axes and the directions of compensation vectors.

(1) Reference angles (θ0, ϕ0) of rotation axes [6104 and 6105]
Set the position of a rotation axis when setting a compensation vector as described in (2).
Usually, set 0.

(2) Directions of compensation vectors [6106 and 6107]
As the direction of a compensation vector, set the rotation angle (RA, RB) in relation to the direction of linear axis 3 (usually, Z direction).
RA: The rotation from the forward direction of linear axis 2 to the forward direction of linear axis 3 in the plane formed by linear axes 2 and 3 is assumed to be forward rotation.
RB: The rotation from the forward direction of linear axis 3 to the forward direction of linear axis 1 in the plane formed by linear axes 3 and 1 is assumed to be forward rotation.
Linear axes 1, 2, and 3 are those specified for the parameters described in the previous section.
If linear axis 3 is not set (type 1), linear axis 3 is assumed to be the Z-axis.

14.15 DESIGNATION DIRECTION TOOL LENGHT COMPENSATION

In a five-axis machine tool having three basic axes and two rotation axes for turning the tool, tool length compensation can be applied in the direction of the tool axis.

The tool axis direction is specified with I, J, and K; a move command for the rotation axes is not specified directly. When I, J, and K are specified in designation direction tool length compensation mode, the following operation is performed automatically :

- Tool length compensation is applied in the direction specified with I, J, and K, by the amount specified with the D code; thus, three linear axes (Xp, Yp, and Zp) are moved. This operation is the same as three-dimensional tool compensation.
- The two rotation axes operate so that the tool axis is oriented in the direction specified by I, J, and K. (This specifications manual explains this operation.)

The two rotation axes in the explanation of this function refer to the B- and C-axes unless otherwise specified.

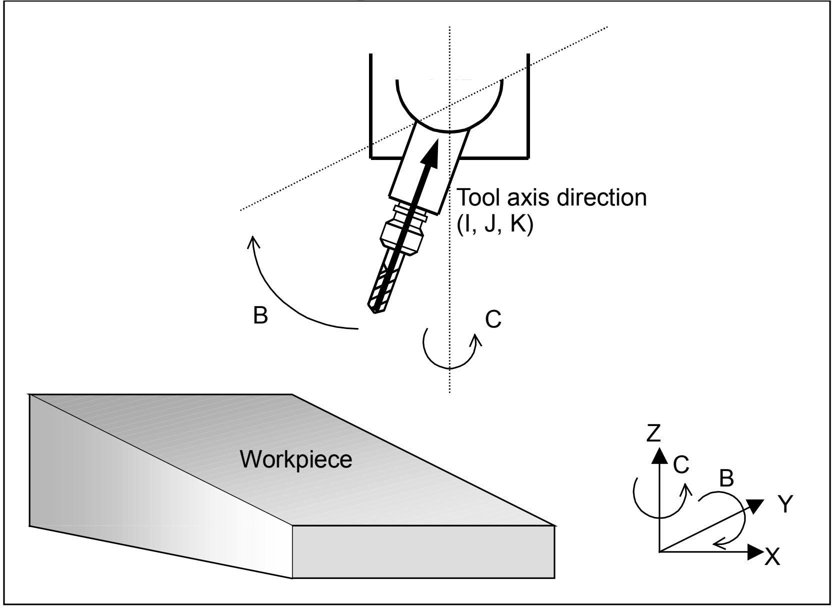


Fig.14.15 (a) Designation direction tool length compensation

Format

- Designation direction tool length compensation

G41 Xp_ Yp_ Zp_ I_ J_ K_ D_ ;

Xp	:	X-axis or an axis parallel to the X-axis
Yp	:	Y-axis or an axis parallel to the Y-axis
Zp	:	Z-axis or an axis parallel to the Z-axis
I,J,K	:	Direction of tool axis
D	:	Offset number

NOTE

1 The format of specified-direction tool length compensation is the same as that for three-dimensional tool compensation. When using specified-direction tool length compensation, set bit 0 (DDT) of parameter No. 7711 to 1.

2 A three-dimensional space in which specified-direction tool length compensation is to be performed is determined by the axis addresses specified in G41. If Xp, Yp, or Zp is not specified, specified-direction tool length compensation is performed along the X-axis, Y-axis, or Z-axis.

3 Specify I, J, and K in a G41 block at all times. If any of I, J, or K is missing, ordinary tool length compensation is performed.

4 Usually, specified-direction tool length compensation is started by the G41 command. When the G42 command is specified, compensation is performed in the direction opposite to that of G41.

- Canceling designation direction tool length compensation

```
G40 Xp_ Yp_ Zp_ B_ C_ ;
```

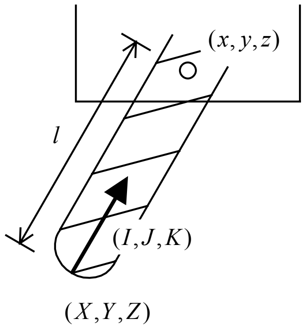
NOTE

A compensation vector can be cancelled by specifying G40; only. In this case, no movement is made on a rotation axis.

Explanation

- Operation in the specified-direction tool length compensation mode

(1) Movement is made on all five axes simultaneously to the position (x, y, z, b, c) calculated as shown below.



x,y,z	: Tool center position
b,c	: Rotation axis position
X,Y,Z	: tip position (programmed position)
I,J,K	: Tool axis direction
l	: Tool offset value

All positions are represented by absolute coordinates.

$$x = X + l\frac{I}{\sqrt{I^2+J^2+K^2}}$$

$$y = Y + l\frac{J}{\sqrt{I^2+J^2+K^2}}$$

$$z = Z + l\frac{K}{\sqrt{I^2+J^2+K^2}}$$

$$b = \tan^{-1}\frac{\sqrt{I^2+J^2}}{K}$$

$$c = \tan^{-1}\frac{J}{I}$$

Fig.14.15 (b) Expression for Compensation Calculation

(2) When circular interpolation or helical interpolation (G02, G03) is specified
I, J, and K have no effect on specified-direction tool length compensation. A compensation vector generated in the previous block is used as is.

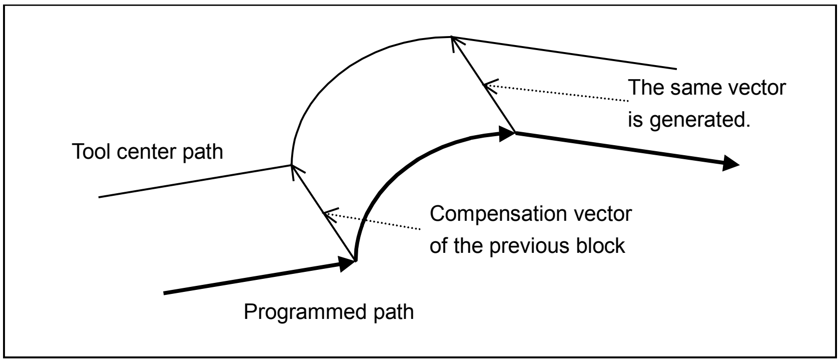


Fig.14.15 (c) Compensation Vector in Circular Interpolation

NOTE
1 In a block in which all of I, J, and K are omitted, the compensation vector for the previous block is used.
2 If any one of I, J, or K is omitted, 0 is assumed in place of the omitted value.
3 Shortcut control is exercised on the amount of movement on a rotation axis so that 180° is not exceeded.

- Specification of the magnitude of a compensation vector

By setting parameter No. 6011, the magnitude of a compensation vector can be specified.

$$x = X + l\frac{I}{S}$$

$$y = Y + l\frac{J}{S}$$

$$z = Z + l\frac{K}{S}$$

where,

x, y, z : Tool center position (absolute coordinates)

X, Y, Z : Tool tip position (absolute coordinates) (Programmed position)

I, J, K : Tool axis direction

l : Tool offset value

S : Parameter No. 6011

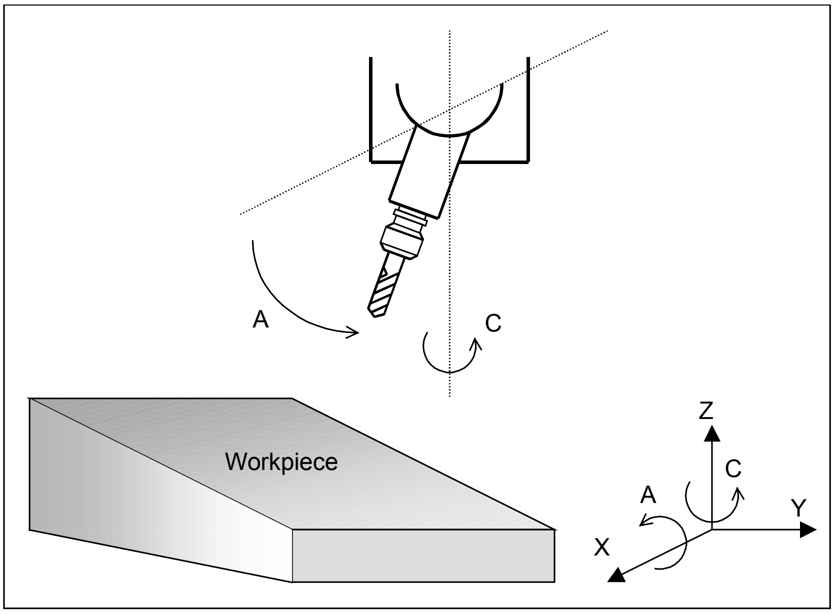
When parameter No. 6011 = 0, however, $S = \sqrt{I^2 + J^2 + K^2}$.

- Feedrate in the specified-direction tool length compensation mode

As the feedrate used in specified-direction tool length compensation, the feedrate on the axes excluding the rotation axes is used. When bit 2 (FWR) of parameter No. 7711 is set to 1, however, the feedrate on all axes including the rotation axes is used.

- Example of machine configuration and expression for rotation axis calculation

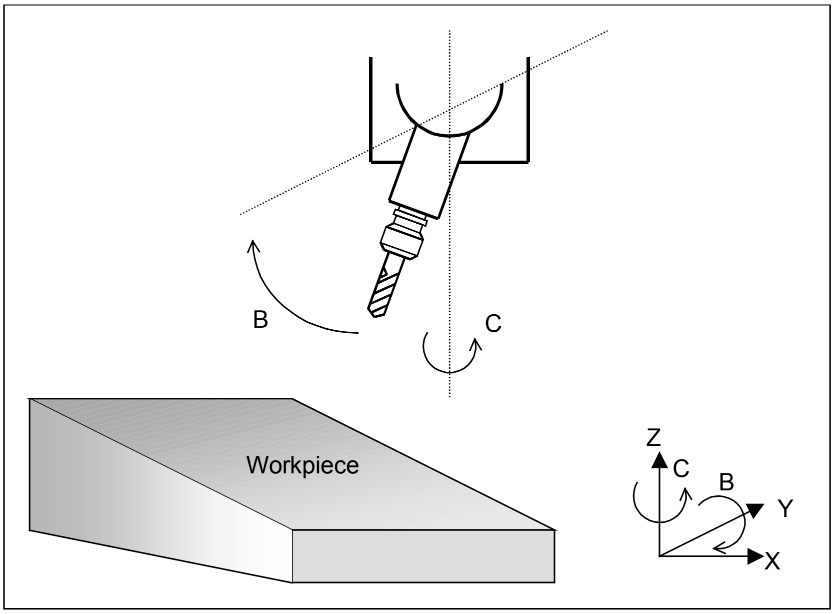
(1) When the rotation axes are the A- and C-axes, and the tool axis is the Z-axis



$$a = \tan^{-1}\frac{\sqrt{I^2 + J^2}}{K}$$

$$c = \tan^{-1}\frac{I}{-J}$$

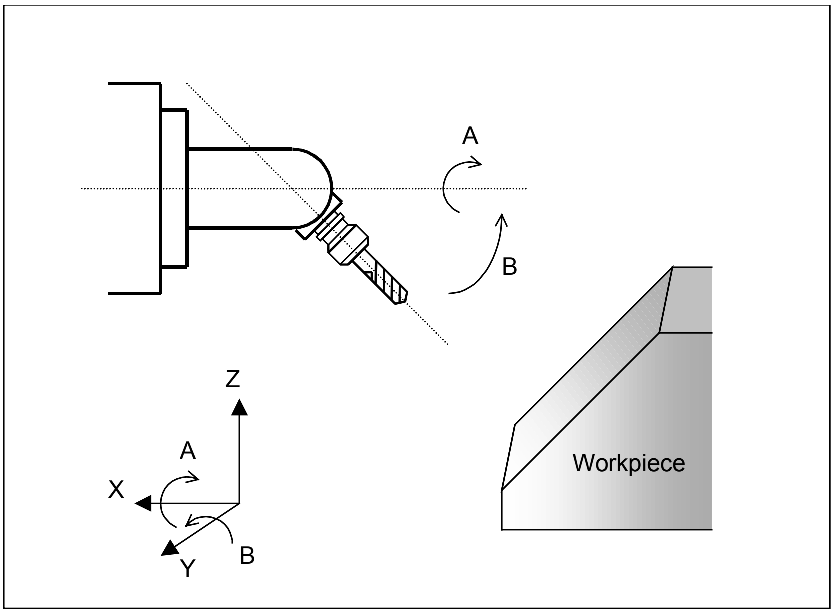
(2) When the rotation axes are the B- and C-axes, and the tool axis is the Z-axis



$$b = \tan^{-1}\frac{\sqrt{I^2+J^2}}{K}$$

$$c = \tan^{-1}\frac{J}{I}$$

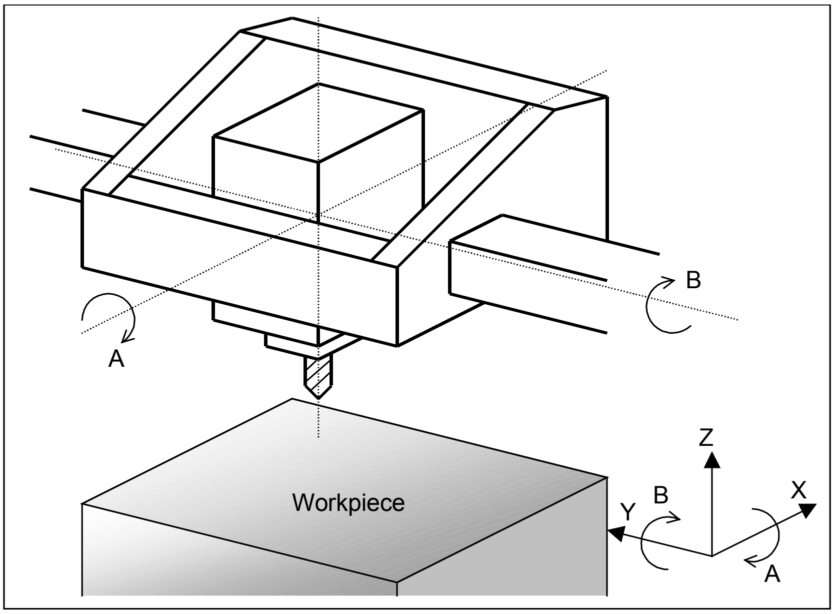
(3) When the rotation axes are the A- and B-axes, and the tool axis is the X-axis



$$a = \tan^{-1}\frac{J}{-K}$$

$$b = \tan^{-1}\frac{\sqrt{J^2+K^2}}{I}$$

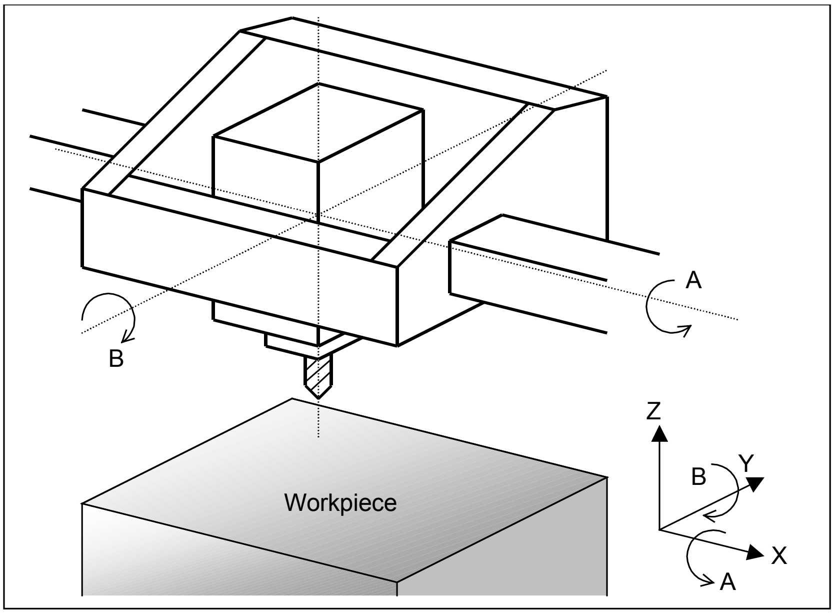
(4) When the rotation axes are the A- and B-axes, and the tool axis is the Z-axis (master axis : B-axis)



$$a = \tan^{-1}\frac{-J}{\sqrt{I^2+K^2}}$$

$$b = \tan^{-1}\frac{I}{K}$$

(5) When the rotation axes are the A- and B-axes, and the tool axis is the Z-axis (master axis : A-axis)



$$a = \tan^{-1}\frac{-J}{K}$$

$$b = \tan^{-1}\frac{I}{\sqrt{J^2+K^2}}$$

Limitation

- Rotation axis specification

A rotation axis must not be specified in specified-direction tool length compensation mode. Otherwise, an alarm (PS0809) is issued.

- Commands related to reference position return

The specified-direction tool length compensation mode must be cancelled before any of the following can be specified :
- Reference position return check (G27)
- Return to reference position (G28)
- Return to 2nd reference position (G30)
- Return from reference position (G29)

- Coordinate system rotation, scaling

In specified-direction tool length compensation mode, coordinate system rotation and scaling are applied to the tool tip position.

- Three-dimensional coordinate conversion

When specified-direction tool length compensation is used during three-dimensional coordinate conversion, three-dimensional coordinate conversion is applied to the tool axis directions (I, J, K) as well.

- Relationships with other compensation functions

(1) Tool length compensation
Tool length compensation is applied to a path resulting from specified-direction tool length compensation.

(2) Tool offset
The tool offset function cannot be used in specified-direction tool length compensation mode.

(3) Cutter compensation
If all of I, J, and K are specified in a G41 block, the specified-direction tool length compensation mode is set. If any one of I, J, or K is omitted, cutter compensation mode is set. This means that both modes cannot be set at the same time.

(4) Tool axis direction tool length compensation
In specified-direction tool length compensation mode, tool axis direction tool length compensation cannot be used.

14.16 TOOL CENTER POINT CONTROL

On a five-axis machine having two rotation axes that turn a tool, tool length compensation can be performed momentarily even in the middle of a block.

This tool length compensation is classified into one of two types based on the programming method. In the explanation of this function, the two rotation axes are assumed to be the B- and C-axes.

(1) Type 1
The rotation axis position (B, C) is specified.
The CNC applies tool length compensation according to the compensation amount along the tool axis whose orientation is calculated from the specified rotation axis position. This means that compensation is performed by moving the three linear axes.

(2) Type 2
The tool axis orientation (I, J, K) is specified.
The CNC controls the two rotation axes so that the tool is oriented as specified, and performs tool length compensation along the tool axis according to the compensation amount. This means that compensation is performed by moving the two rotation axes and three linear axes.

Tool center point control (type 1) differs from tool length compensation along the tool axis as shown below:

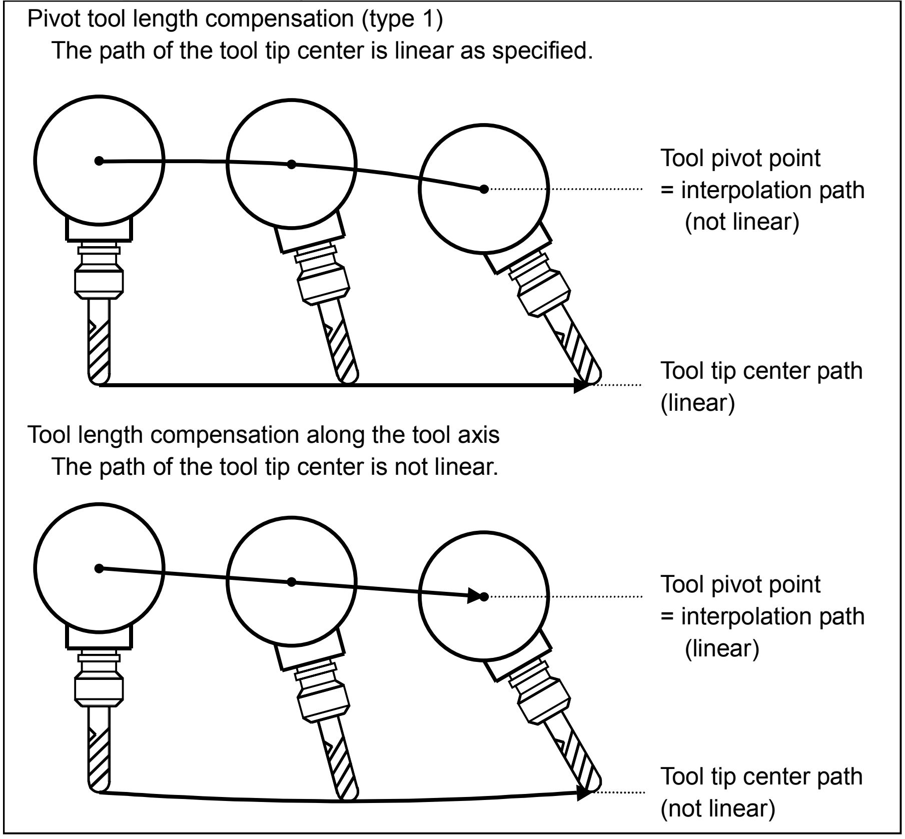

Pivot tool length compensation (type 1)
The path of the tool tip center is linear as specified.
Tool pivot point
= interpolation path
(not linear)
Tool tip center path
(linear)
Tool length compensation along the tool axis
The path of the tool tip center is not linear.
Tool pivot point
= interpolation path
(linear)
Tool tip center path
(not linear)

Fig. 14.16 (a) Difference between Tool Center Point Control and Tool Length Compensation along the Tool Axis

> **NOTE**
> The length from the tool tip to tool pivot point must equal the sum of the tool length compensation amount and tool holder offset value.

The difference between tool center point control (type 2) and designation direction tool length compensation is also the same as that shown in Fig. 14.16 (a).

Format

- Specifying tool center point control (type 1)

```
G43.4 H_ ;
   H  : Offset number
```

- Specifying tool center point control (type 2)

```
G43.5 I_ J_ K_ H_ Q_ ;
   I,J,K   : Tool axis orientation
   H  : Offset number
   Q  : Tool inclination angle (degrees)
```

> **NOTE**
> 1 The command format of tool center point control (type 2) differs considerably from the command format of designation direction tool length compensation (G41I_J_K_D_).
> 2 When I, J, and K are all omitted from a block, the compensation vector in the previous block is used.
> 3 When any of I, J, and K is omitted, the omitted I, J, or K is assumed to be 0.
> 4 Movement of the rotation axes is controlled by shortcut control so that the amount of movement does not exceed 180°.
> 5 To change the specification for Q, G43.5 must be specified again.

- Canceling tool center point control

```
G49 ;
```

> **NOTE**
> The command for canceling tool center point control (type 2) differs from the command for canceling designation direction tool length compensation (G40).

Explanations

- Specification of tool center point control

The tool compensation vector changes in the following cases:

Type 1 : The offset value is changed, or the rotation axis position (B, C) is specified.

Type 2 : The offset value is changed, or the tool axis orientation (I, J, K) is specified.

As the tool compensation vector changes, movement is performed along the X-, Y-, and Z-axes by an amount equal to the change. The time at which the tool compensation vector is calculated is as follows:

Tool center point control :
Calculated momentarily even in the middle of a block.

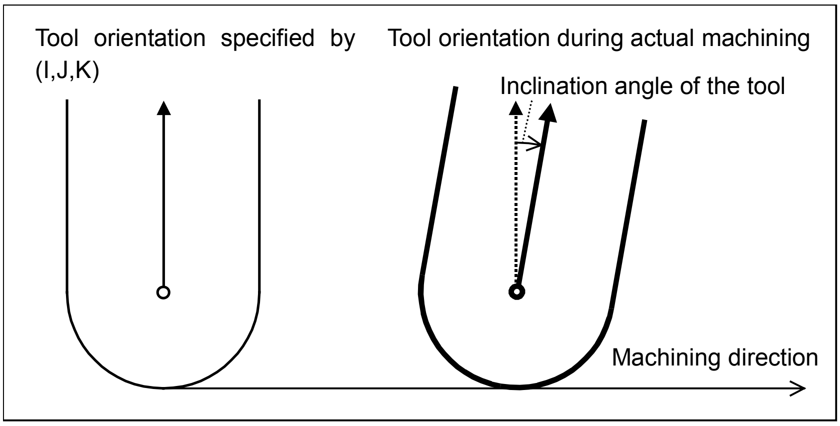
Tool length compensation along the tool axis :
Calculated only at the end point of a block.

Tool length compensation in a specified direction :
Calculated only at the end of a block

When only the rotation axis position is specified in tool center point control (type 1) mode, and when only I, J, and K are specified in tool center point control (type 2) mode, the tool tip center position remains unchanged before and after the specification. (Also, while the rotation axes are being moved, the tool tip center does not move.)

- Inclination of the tool

For tool center point control (type 2), the inclination angle of the tool can be specified with address Q of G43.5. The inclination angle of the tool is the difference in the angle between the tool orientation specified by (I, J, K) and the tool orientation set for actual machining.
If the tool orientation specified by (I, J, K) matches the tool orientation set for actual machining, Q need not be specified.

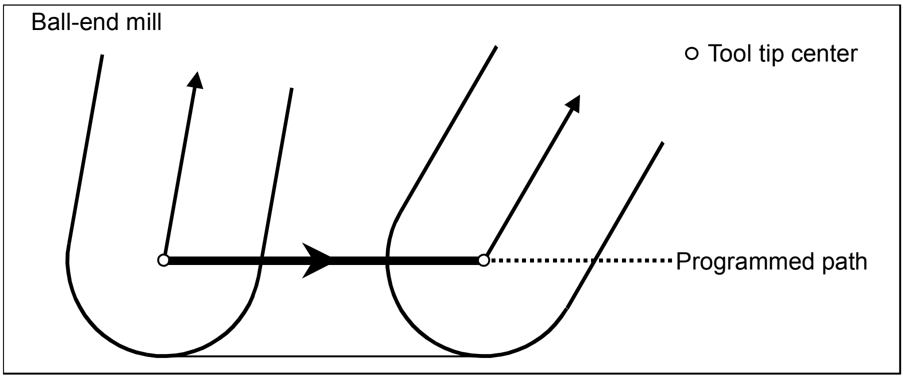

Tool orientation specified by (I,J,K)
Tool orientation during actual machining
Inclination angle of the tool
Machining direction

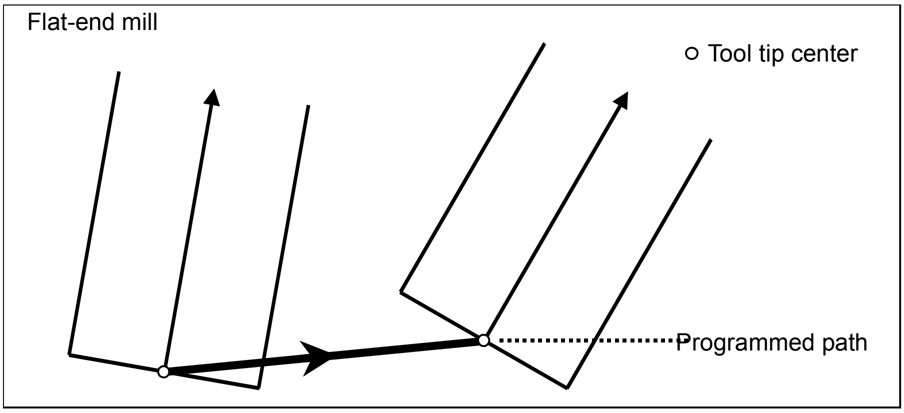
Example: For machining with the tool tilted toward the machining direction by 2 degrees, specify the following:

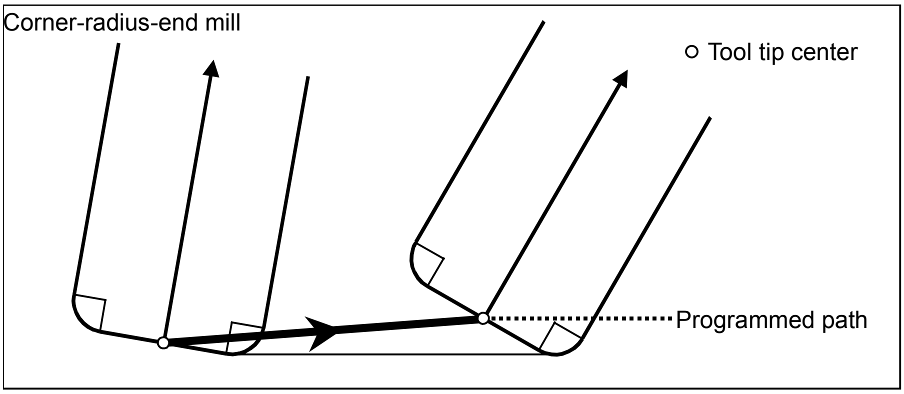
G43.5 I_ J_ K_ H_ Q2.0

- Programmed point

In programming, the position of the tool tip center is specified.







- Linear interpolation (G01)

When linear interpolation (G01) is specified in tool center point control mode, the feedrate is controlled so that the tool tip center moves at a specified feedrate.
Also, in tool center point control (type 2), the feedrate is controlled so that the tool tip center moves at a specified feedrate regardless of the setting of bit 2 (FWR) of parameter No. 7711.

- Specification of rotation axes

(1) Type 1
When only the rotation axes are specified in tool center point control (type 1) mode, the feedrate of the rotation axes is set to the maximum cutting feedrate (parameter No. 1422).

(2) Type 2
In tool center point control (type 2) mode, the rotation axes cannot be specified. If the rotation axes are specified, alarm (PS1061) occurs.

- Positioning (G00)

NOTE

1 Set the following parameters:
   (1) Bit 4 (LRP) of parameter No.1400 = 1:
       Linear-type rapid traverse
   (2) Bit 5 (FRP) of parameter No.1603 = 1:
       Acceleration/deceleration before interpolation is used in rapid traverse.
   (3) Parameter No.1671:
       Acceleration of acceleration/deceleration before interpolation for rapid traverse
   (4) Parameter No.1672:
       Acceleration change period of bell-shaped acceleration/deceleration before interpolation for rapid traverse

2 If the above settings are not made, or if look-ahead acceleration/deceleration before interpolation is not valid, axis movement may be performed at a higher feedrate than the rapid traverse rate.

- Operation at start and cancellation

(1) Type 1
When tool center point control (type 1) starts (G43.4H_) and when it is canceled (G49), the CNC calculates the compensation vector only at the end of the block.

(2) Type 2
When tool center point control (type 2) starts (G43.5H_) and when it is canceled (G49), the CNC calculates the compensation vector only at the end of the block.

- Operation of tool center point control (type 1)

The following items are the same as for tool length compensation along the tool axis:

- Machine configuration example and equation for rotation axis calculation
- Tool holder offset
- Specification of angular displacement in a parameter
- Zero-point compensation for the rotation axes
- Rotation axis offset

- Operation of tool center point control (type 2)

The following item is the same as for tool length compensation along the tool axis:

- Tool holder offset

  The following items are the same as for tool length compensation in a specified direction:
- Operation in tool length compensation in a specified direction
- Machine configuration example and equation for rotation axis calculation

- Tool offset by tool number

When the tool offset by tool number is used, the tool length compensation amount corresponding to a tool number (T code) is used for tool center point control.

- Tool life management

When tool life management is used, the tool length compensation amount of the tool used is used for tool center point control.

- Three-dimensional cutter compensation

Tool center point control and three-dimensional cutter compensation can be used at the same time.

Three-dimensional cutter compensation is applied to a specified tool tip point. Three-dimensional cutter compensation, however, is not performed momentarily in the middle of a block; it is performed only at the end of a block.

- Three-dimensional coordinate conversion

When tool center point control (type 2) is used during three-dimensional coordinate conversion, the tool axis orientation (I, J, K) is also subjected to three-dimensional coordinate conversion.

- Rotation axis rollover

NOTE

Whenever using tool center point control (type 2), set bit 2 (ROL) of parameter No. 1009 to 1 to perform rotation axis rollover.

Restrictions

- Axis name extension

NOTE

When I, J, and K are used as axis names by axis name extension, tool center point control (type 2) cannot be used.

- Manual intervention

When manual intervention is applied to a rotation axis with the manual absolute switch off, the compensation vector is calculated using the rotation axis position that was found immediately before manual intervention.

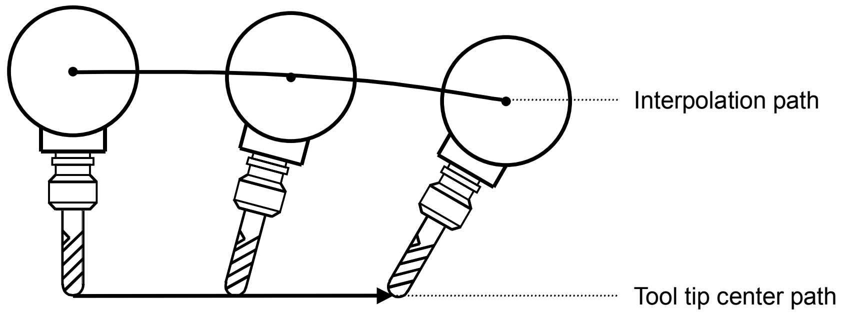
Example:

When the manual absolute switch is turned off with B positioned to 30.0, then a manual intervention by 1.0 degree is applied to the B-axis, the machine coordinate position of the B-axis is 31.0. Then, starting automatic operation sets the absolute coordinate position of the B-axis to 30.0. In tool center point control, therefore, the compensation vector is calculated with B set to 30.0.

- Look-ahead acceleration/deceleration before interpolation

When using tool center point control, also use look-ahead acceleration/ deceleration before interpolation. If look-ahead acceleration/ deceleration before interpolation is not used, the feedrate may exceed the maximum cutting feedrate as a result of tool center point control, which will cause alarm OT0553 to be issued.

Example:



In the above example, the feedrate is controlled so that the tool tip center moves at a specified feedrate. As a result, a higher feedrate than the specified feedrate is detected on the interpolation path. In such a case, if look-ahead acceleration/deceleration before interpolation is used, the feedrate is clamped so that the feedrate on the interpolation path does not exceed the maximum cutting feedrate. If look-ahead acceleration/deceleration before interpolation is not used, the feedrate on the interpolation path may become higher than the maximum cutting feedrate, resulting in alarm OT0553 being issued.

- Mirror image

When a mirror image by the external switch or CNC setting is used in tool center point control mode, the movement of the tool control point is reversed. Therefore, the movement at the tool center point does not match the path reversed by a program command. When a programmable mirror image was specified, the path at the tool center point can be reversed by a program command. In this case, however, mirroring must be performed so that there is no contradiction between the linear axis and the rotary axis. For details, see the programmable mirror image section (II-13.5).

- Functions resulting in the same operation as tool length compensation along the tool axis Functions resulting in the same operation as tool length compensation in a specified direction

When the following functions are used in tool center point control mode, the same operation as tool length compensation along the tool axis (type 1) or tool length compensation in a specified direction (type 2) results:

-Specification of an axis not related to tool center point control
-Skip function (G31 to G31.9)
-Unidirectional positioning (G60)
-The following G functions of group 01:
    Circular interpolation, helical interpolation, spiral interpolation, conical interpolation (G02, G03)
    Circular threading B (G2.1, G3.1)
    Involute interpolation (G2.2, G3.2)
    Three-dimensional circular interpolation (G2.4, G3.4)
    Threading (G33)
-Selection of a workpiece coordinate system (G54 to G59)
-Setting of a workpiece coordinate system (G92)
-Feed per revolution (G95)
-Inverse time feed (G93)

- Unavailable functions 1

In tool center point control mode, the functions listed below cannot be used.

If these functions are used, the compensation vector of the previous block is used as is.

-The following G functions of group 01:
    Exponential interpolation (G2.3, G3.3)
    Spline interpolation (G6.1)
    Smooth interpolation (G5.1Q2)
    NURBS interpolation (G6.2)
-Cylindrical interpolation (G7.1)
-Polar coordinate interpolation (G12.1)
-Normal-direction control (G41.1, G42.1)

- Unavailable functions 2

In tool center point control mode, the functions listed below cannot be used.

To use these functions, specify G49 to cancel tool center point control, then specify these functions.

-Reference position return check (G27)
-Automatic reference position return command (G28, G29, G30)
-Positioning of the machine coordinate system (G53)

Reference

This Manual		II.14.12	Tool Length Compensation along Tool Axis
		II.14.17	Tool Length Compensation in a Specified Direction
FANUC Series 15*i*/150*i*-MB	Parameter Manual (B-63790EN)	4.4.29	Parameters related to the 5-axis control function

14.16.1 Tool Center Point Control for 5-Axis Machining

Overview

There are three different types of five-axis machines. They are <1> a tool rotation type, <2> a table rotation type, and <3> a tool and table rotation type. (See Fig.14.16.1(d).)

The conventional tool center point control method can be applied only to <1> tool rotation type machines.

(See Fig.14.16.1(a).)

This function is intended to make tool center point control applicable to <2> table rotation type and <3> mixed-type machines.

To put another way, tool center point control for 5-axis machining (called tool center point control later)is intended to perform machining on a five-axis machine having table rotary axes as well as three orthogonal axes (X-, Y-, and Z-axes) while changing the attitude of the tool. With this function, the tool center point can move along the specified path even if the tool is tilted with respect to the workpiece. (See Fig.14.16.1(b).)

Programming is possible in a coordinate system (programming coordinate system) fixed to the table, making CAM-based programming easy. Also, the cutting speed becomes easy to control, because the tool center point moves at a specified speed.

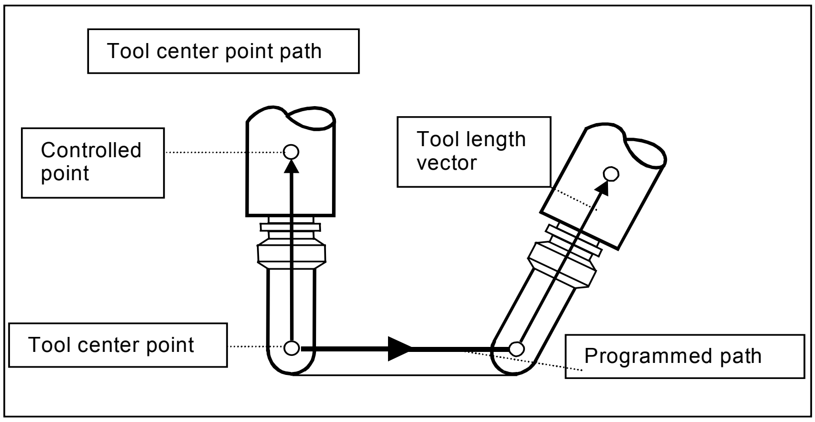


Fig.14.16.1 (a)

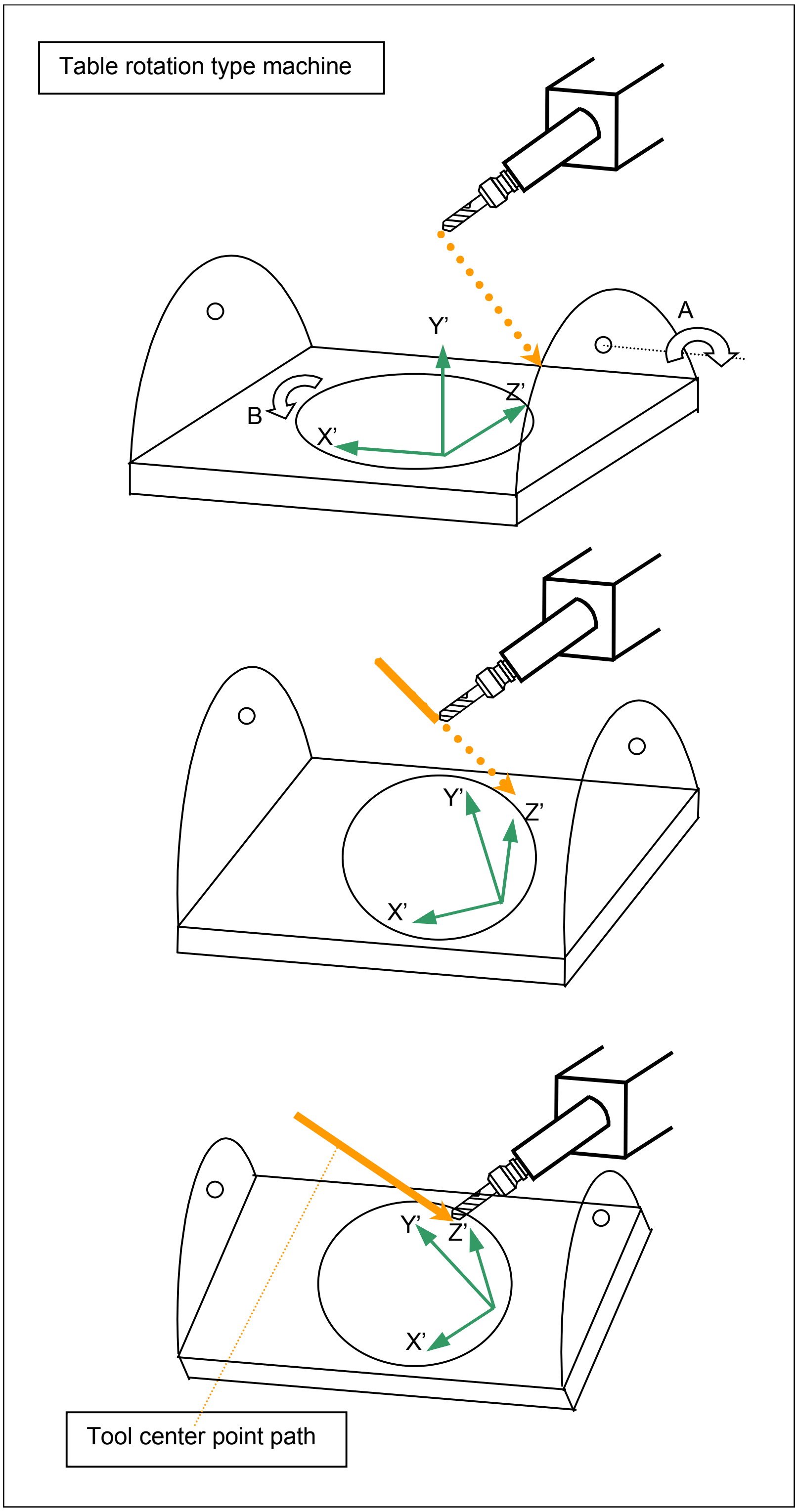


Table rotation type machine
A
Y'
Z'
B
X'
Y'
Z'
X'
Y'
Z'
X'
Tool center point path

Fig.14.16.1 (b)

As the table rotates, the position and orientation of a workpiece fixed on the table change. However, programmed positions are specified in the coordinate system fixed on the table (programming coordinate system). Because the programming coordinate system does not move with respect to the table, specifying a line with a program generates a straight path as viewed in the programming coordinate system.

Fig.14.16.1(c) shows how linear interpolation in the programming coordinate system is performed on a mixed-type machine.

If linear interpolation is specified in this function mode, speed control is carried out in such a way that tool center point moves at a specified speed in the programming coordinate system.

Example

Machine configuration: The A-axis is a rotary axis for controlling the tool.
The B-axis is a rotary axis for controlling the table.
Program: Created in the "programming coordinate system"

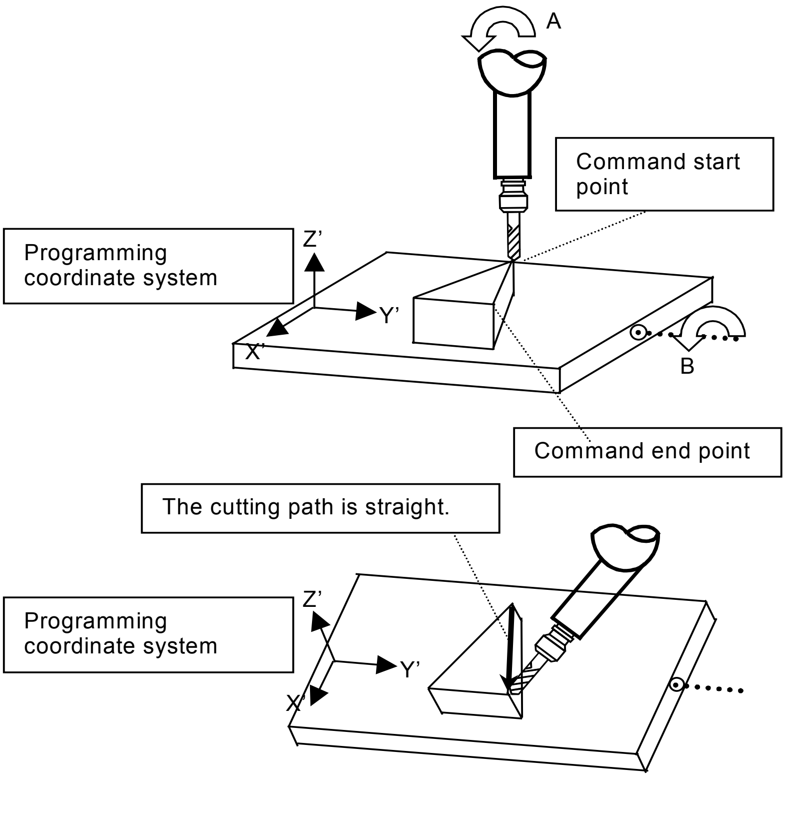

A
Command start point
Programming coordinate system
Z’
Y’
X’
B
Command end point
The cutting path is straight.
Programming coordinate system
Z’
Y’
X’

Fig.14.16.1 (c)

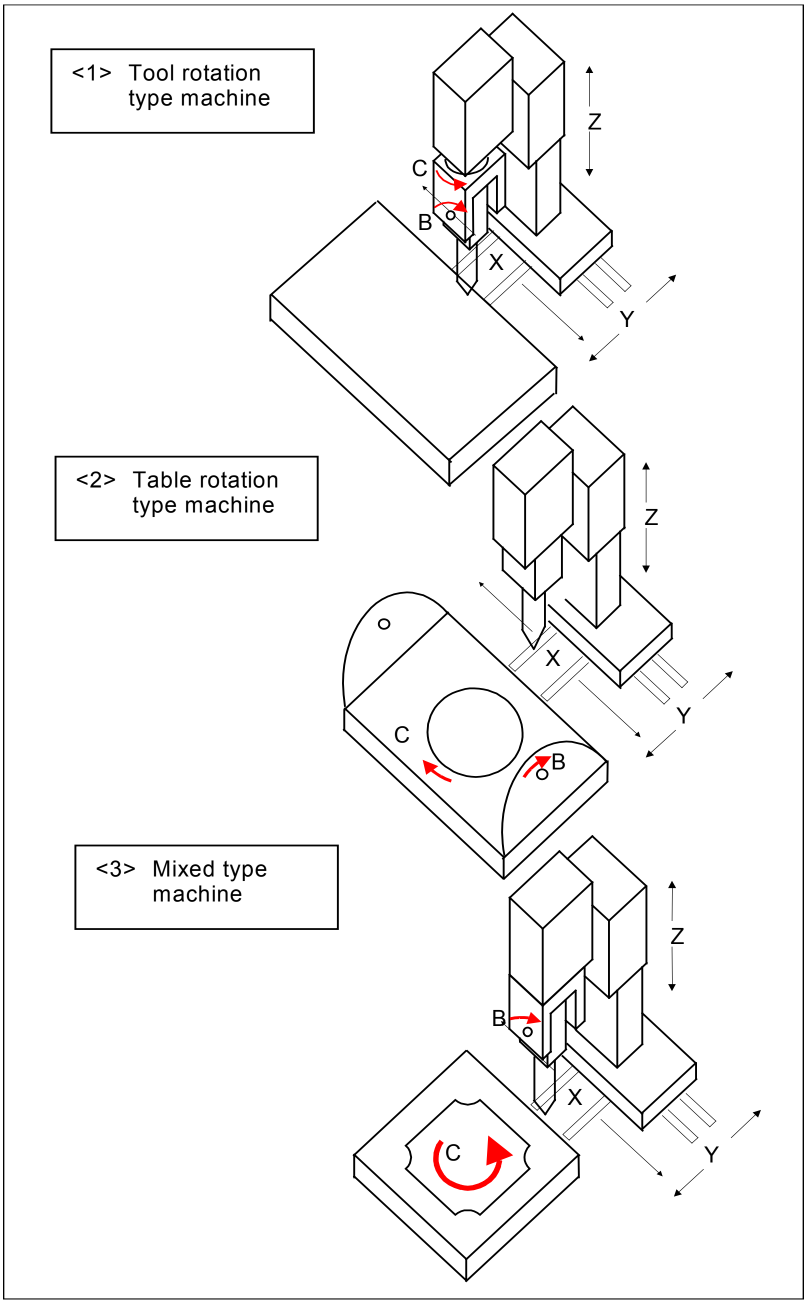

<1> Tool rotation type machine
Z
C
B
X
Y
<2> Table rotation type machine
Z
X
Y
C
B
<3> Mixed type machine
Z
B
X
Y
C

Fig.14.16.1 (d)

This function can be used also when the rotary axis for controlling the tool and the rotary axis for controlling the table do not cross each other.

Format

- Tool center control command

Format
G43.4 H_ ; Starts tool center point control (TYPE1)
G49 ; Cancels tool center point control.
Symbol description
H : Tool offset number

Once this command is issued, linear interpolation for the X-, Y-, and X-axes is issued in the programming coordinate system.
Then, a rotation command for a table or tool is issued.

The CNC controls the controlled point so that the tool center point moves along the specified straight line in the programming coordinate system at the same time with interpolation for the rotary axis.

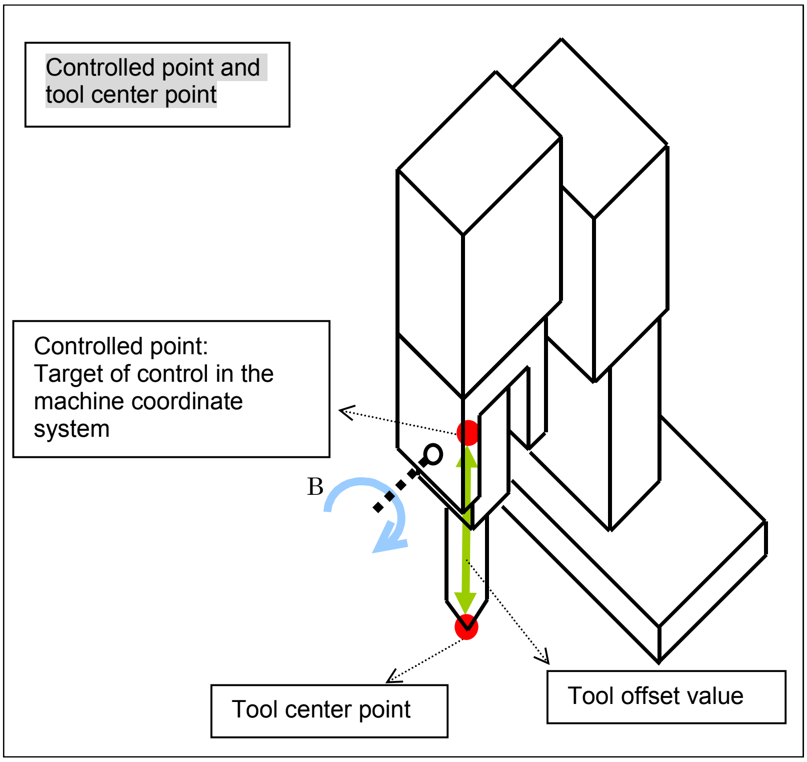
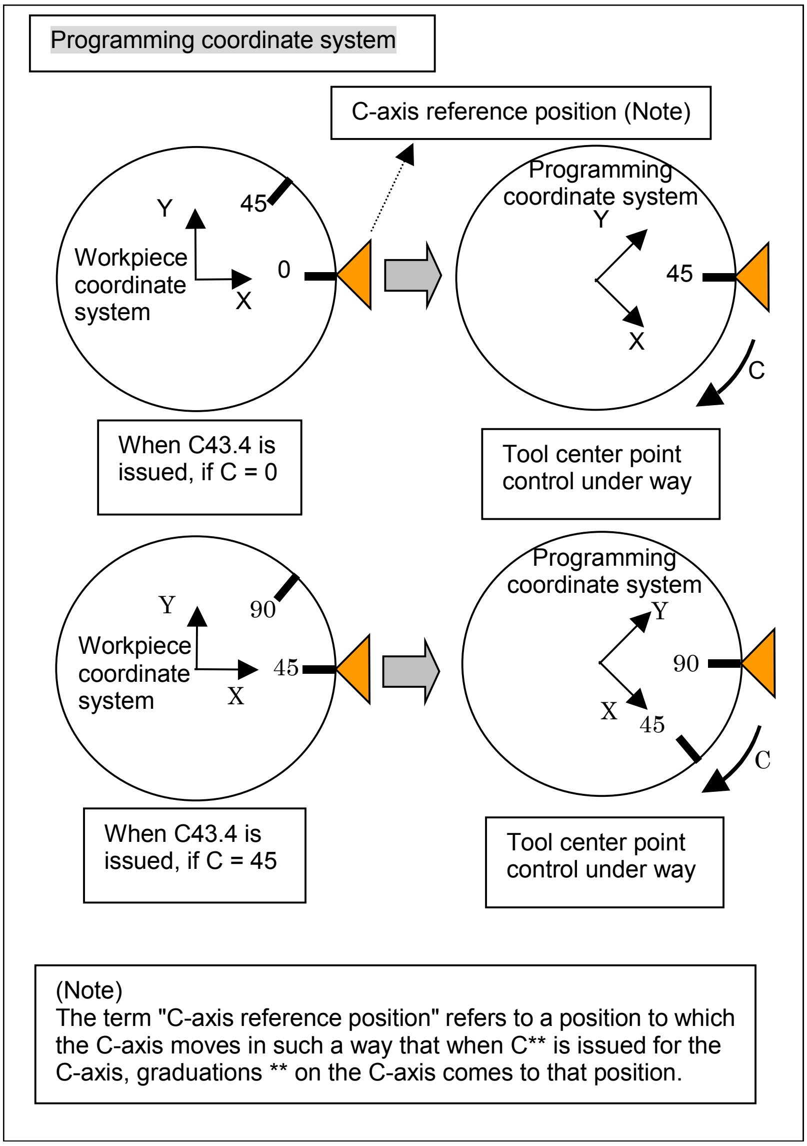

Controlled point and tool center point
Controlled point: Target of control in the machine coordinate system
B
Tool center point
Tool offset value

Fig.14.16.1 (e)

Programming coordinate system

Issuing G43.4 makes the CNC use the current workpiece coordinate system as its programming coordinate system (fixed on the table).
The programming coordinate system is used for tool center point control.
It rotates as the table rotates.
It does not rotate when the tool head rotates.
Once G43.4 is issued, it is assumed that the subsequent X, Y, and Z commands are issued in the programming coordinate system.

If the table rotary axis moves in the same block as for G43.4 or in a block before G43.4, the angle of the table rotary axis is used as the initial status of the programming coordinate system.


Programming coordinate system
C-axis reference position (Note)
Y
45
Workpiece coordinate system
0
X
Programming coordinate system
Y
45
X
C
When C43.4 is issued, if C = 0
Tool center point control under way
Programming coordinate system
Y
90
Workpiece coordinate system
45
X
Y
90
X
45
C
When C43.4 is issued, if C = 45
Tool center point control under way

(Note)
The term "C-axis reference position" refers to a position to which the C-axis moves in such a way that when C** is issued for the C-axis, graduations ** on the C-axis comes to that position.

Fig.14.16.1 (f)

Operation descriptions

- Tool center point control command

When tool center point control is in use, a move command is issued in the programming coordinate system.
The program specifies the tool center point.
For rotary axes, the positions of each block end are specified.
The feed rate is specified with F, using a tangential speed (relative speed between the workpiece and tool) in the programming coordinate system.

- Commands that can be issued when tool center point control is in use

The commands that can be issued when tool center point control is in use are linear interpolation (G01) and positioning (G00).
If a linear interpolation command (G01) is issued when tool center point control is in use, speed control is carried out in such a way that the tool center point can move at the specified speed.
The speed of the controlled point is displayed as the actual speed.
For positioning (G00), observe the following cautions.

⚠ **CAUTION**

1 Be sure to set the following parameters.
   1) Parameter LRP(No.1400#4=1)=1 :
      Linear rapid traverse
   2) Parameter FRP(No.1603#5=1)=1 :
      Acceleration/deceleration before interpolation is used for rapid traverse.
   3) Parameter No.1671 :
      Acceleration before interpolation for rapid traverse
   4) Parameter No.1672 :
      Change time for bell-shaped acceleration before interpolation for rapid traverse

2 If the above settings are not made, or look-ahead acceleration/deceleration before interpolation is not in effect, an axis may move faster than the rapid traverse rate.

- Rotary axis command

If only a rotary axis is specified when tool center point control is in use, the maximum cutting feed rate (parameter No. 1422) becomes effective as the feed speed for the rotary axis.

- Tool behavior at start and cancellation

When tool center point control is started (G43.4) or canceled (G49), the tool moves by a tool offset value.
Compensation vector calculation is performed only at the end of a block.

- Current position display when tool center point control is in use

For a machine coordinate system for which tool center control is in use, the position of the controlled point (rotation center of the tool rotary axis) is displayed.

Which to use, absolute or relative coordinates, is selected using parameter DET (bit 6 of parameter No. 2219). If parameter DET = 0 (bit 6 of parameter No. 2219 = 0), the position of the tool center point in the programming coordinate system is displayed.If parameter DET = 1 (bit 6 of parameter No. 2219 = 1), the position of the tool center point in the machine coordinate system is displayed.

- Tool offset

If a tool number-based tool offset is used, tool center point control is carried out using the tool length compensation value that corresponds to a tool number (T code).

If tool life management is in use, tool center point control is carried out using the tool length compensation value for a tool in use.

Concrete examples of operations

One of the examples explained below uses two table rotary axes. The other example uses one table rotary axis and one tool rotary axis.

- Table rotation type

Explained below is a machine configuration (trunnion) in which a rotary table that rotates on the Y-axis is placed on the X-axis as the table rotary axis. (See Fig.14.16.1(g).)

If a command for a rotary axis for moving a rotary table and a command for linear interpolation for the X-, Y-, and Z-axes of the table coordinate system fixed at the rotary table are issued, control is performed in such a way that as the rotary table rotates, the tool center point moves on a specified straight line in the programming coordinate system.
Also speed control is performed in such a way that the tool center point moves at the specified speed in the programming coordinate system.

O100 is sample program 1.

```
O100 (Sample Program1) ;
N1 G00 G90 A0 B0 ;
N2 G55 ;
    Gets the programming coordinate system ready.
N3 G43.4 H01 ;
    Starts tool center point control.
    H01 is a tool compensation number.
N4 G00 X20.0 Y100.0 Z0 ;
    Moves the tool to the start point.
N5 G01 X100.0 Y20.0 Z30.0 A60.0 B45.0 F500 ;
    Linear interpolation.
N6 G49;
    Cancels tool center point control.
N7 M30;
```

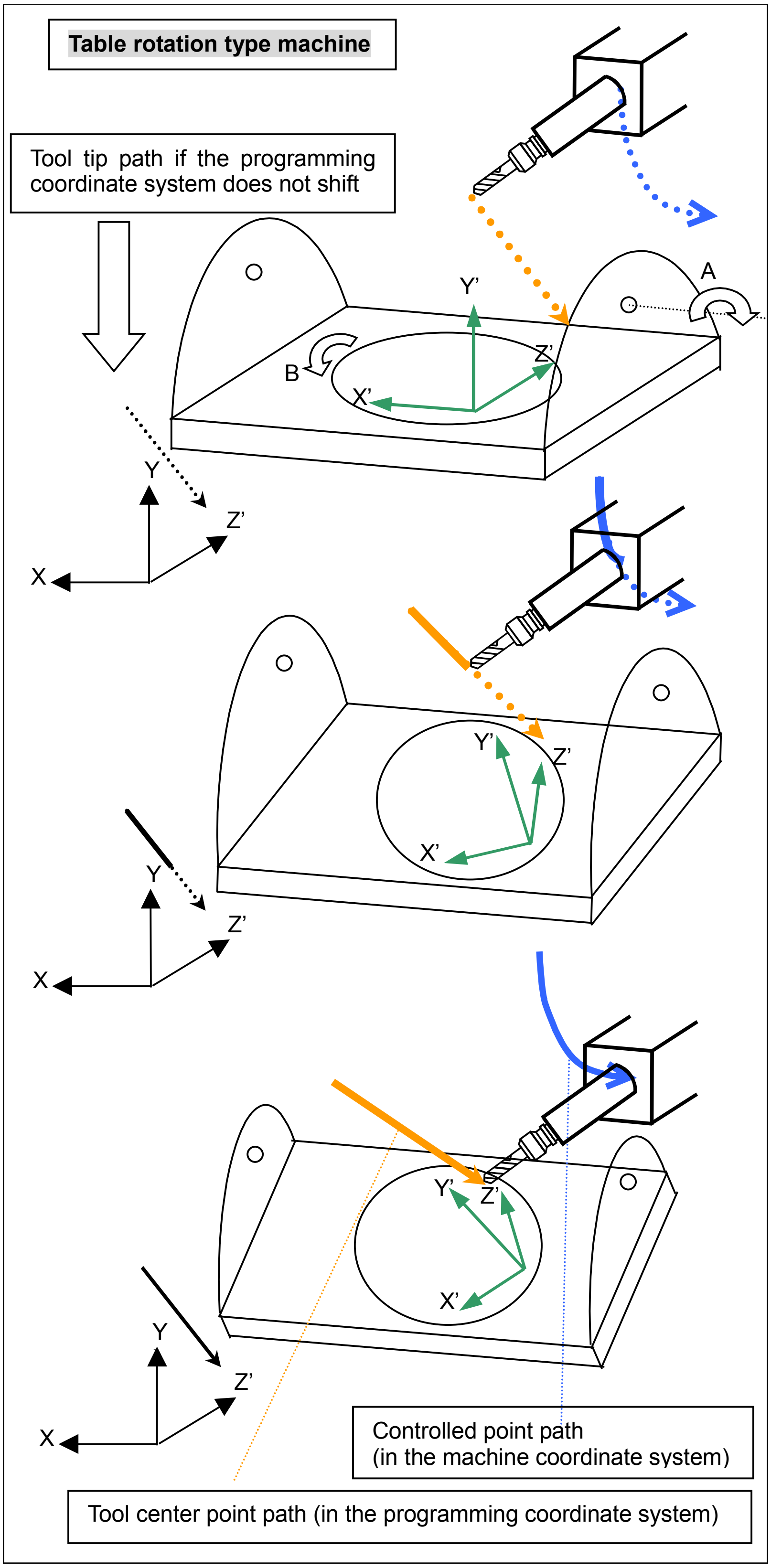


Table rotation type machine
Tool tip path if the programming coordinate system does not shift
A
B
Y'
Z'
X'
Y
Z'
X
Y'
Z'
X'
Y
Z'
X
Y'
Z'
X'
Y
Z'
X
Controlled point path (in the machine coordinate system)
Tool center point path (in the programming coordinate system)

Fig.14.16.1 (g)

- Mixed type

Explained below is a mixed-type machine configuration with one table rotary axis (X-axis) and one tool rotary axis (Y-axis). (See Fig.14.16.1(h).)

If commands for a rotary axis for moving a rotary table and for a tool rotary axis and a command for linear interpolation for the X-, Y-, and Z-axes of the table coordinate system fixed at the rotary table are issued, control is performed in such a way that as the rotary table and tool rotate, the tool center point moves on a specified straight line in the programming coordinate system.
Also speed control is performed in such a way that the tool center point moves at the specified speed in the programming coordinate system.
O200 is sample program 2.

```
O200 (Sample Program2) ;
N1 G00 G90 A0 B0 ;
N2 G55 ;
    Gets the programming coordinate system ready
N3 G43.4 H01 ;
    Starts tool center point control.
    H01 is a tool compensation number.
N4 G00 X200.0 Y150.0 Z20.0 ;
    Moves the tool to the start point.
N5 G01 X5.0 Y5.0 Z50.0 A60.0 B45.0 F500;
    Linear interpolation.
N6 G49 ;
    Cancels tool center point control.
N7 M30;
```

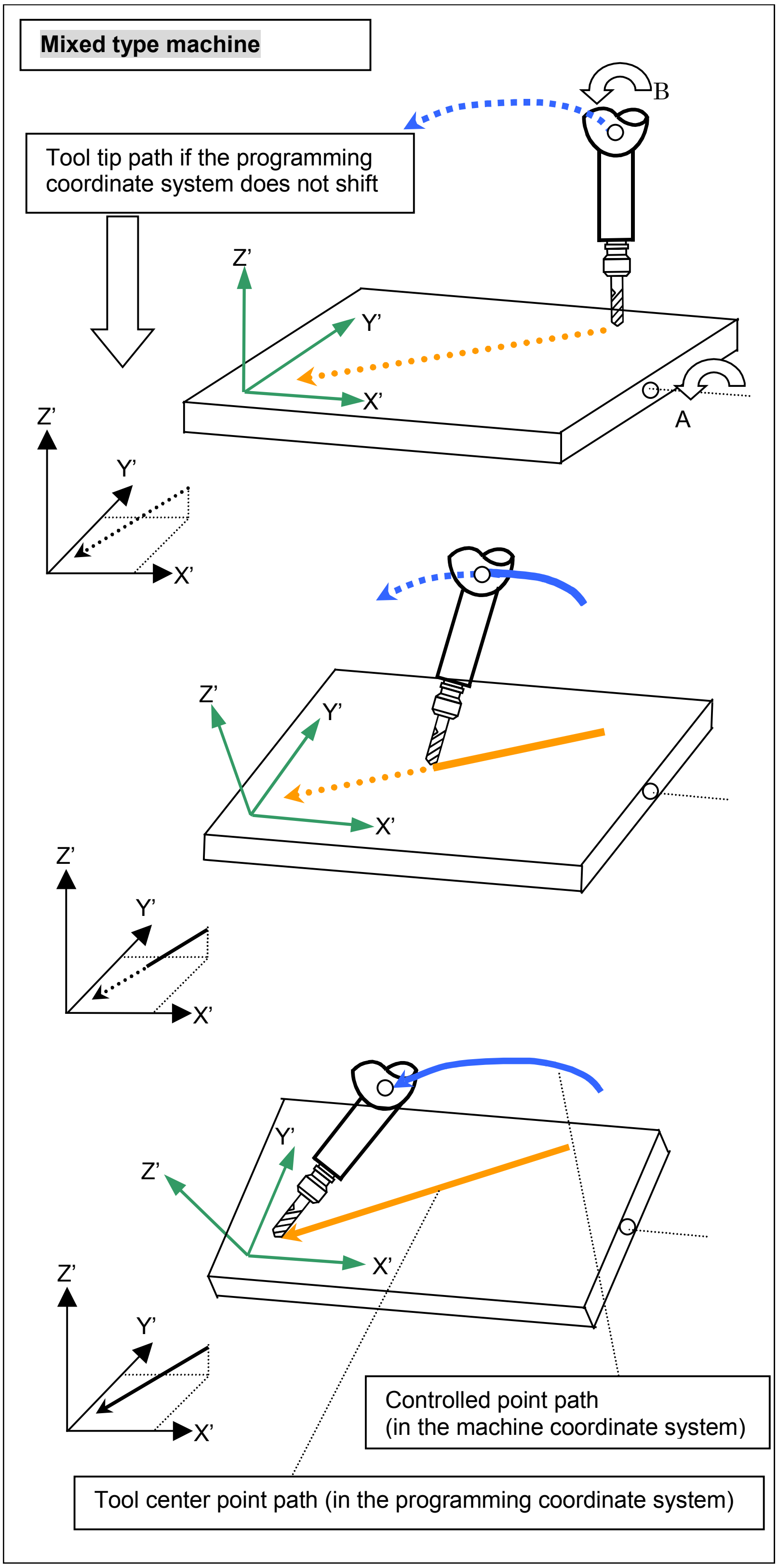

Mixed type machine
B
Tool tip path if the programming coordinate system does not shift
Z'
Y'
X'
A
Z'
Y'
X'
Z'
Y'
X'
Z'
Y'
X'
Y'
Z'
X'
Z'
Y'
X'
Controlled point path (in the machine coordinate system)
Tool center point path (in the programming coordinate system)

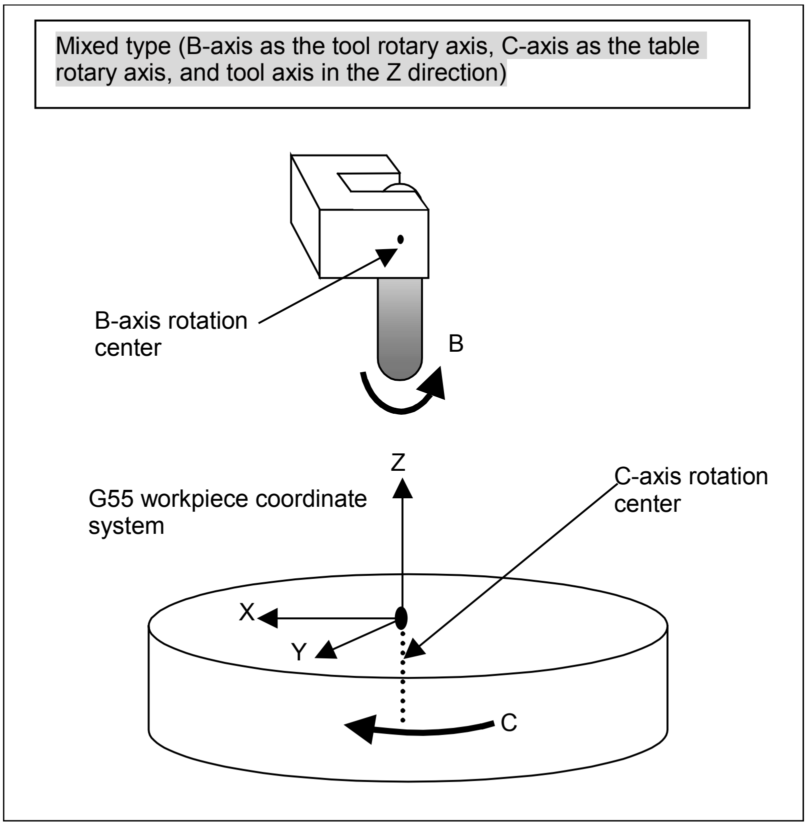
Fig.14.16.1 (h)

Examples

O300 is sample program 3.

In this example, each side, 100 mm long, of an equilateral triangle is created with the B-axis set, respectively, to 0, 30 to 60, and 60 degrees.

```
O300 (Sample Program3) ;
N10 G55 ;
    Gets the programming coordinate system ready
N20 G90 X50.0 Y-70.0 Z300.0 B0 C0 ;
    Moves the tool to the initial position.
N30 G01 G43.4 H01 Z20.0 ;
    Starts tool center point control.
    Moves the tool to an approach position.
    H01 is a tool compensation number.
N40 X28.868 Y-50.0 Z10.0 B30.0 ;
    Set the height of the surface to be cut to 10.0 mm along the Z-axis.
N50 Y50.0 ;
N60 B45.0 C120.0 ;
N70 X-57.735 Y0 B60.0 C180.0 ;
     Moves the tool along the X- and Y-axes while keeping the B- and C-axes
    operating.
N80 C240.0 ;
N90 X28.868 Y-50.0 ;
N100 X50.0 Y-70.0 Z20.0 B0 C360.0 ;
    Set the X, Y, and Z coordinates of the approach position.
    Moves the rotary axes to the previous position.
N110 G49 Z300.0 ;
    Cancels tool center point control.
    Moves the tool to the initial position along the Z-axis.
N120 M30;
```



Mixed type (B-axis as the tool rotary axis, C-axis as the table rotary axis, and tool axis in the Z direction)

Fig.14.16.1 (i)

Fig.14.16.1(j) shows the attitude of the workpiece and the attitude of the tool head relative to the workpiece as viewed in the +Z direction on the assumption that the table rotary axis C stands still.

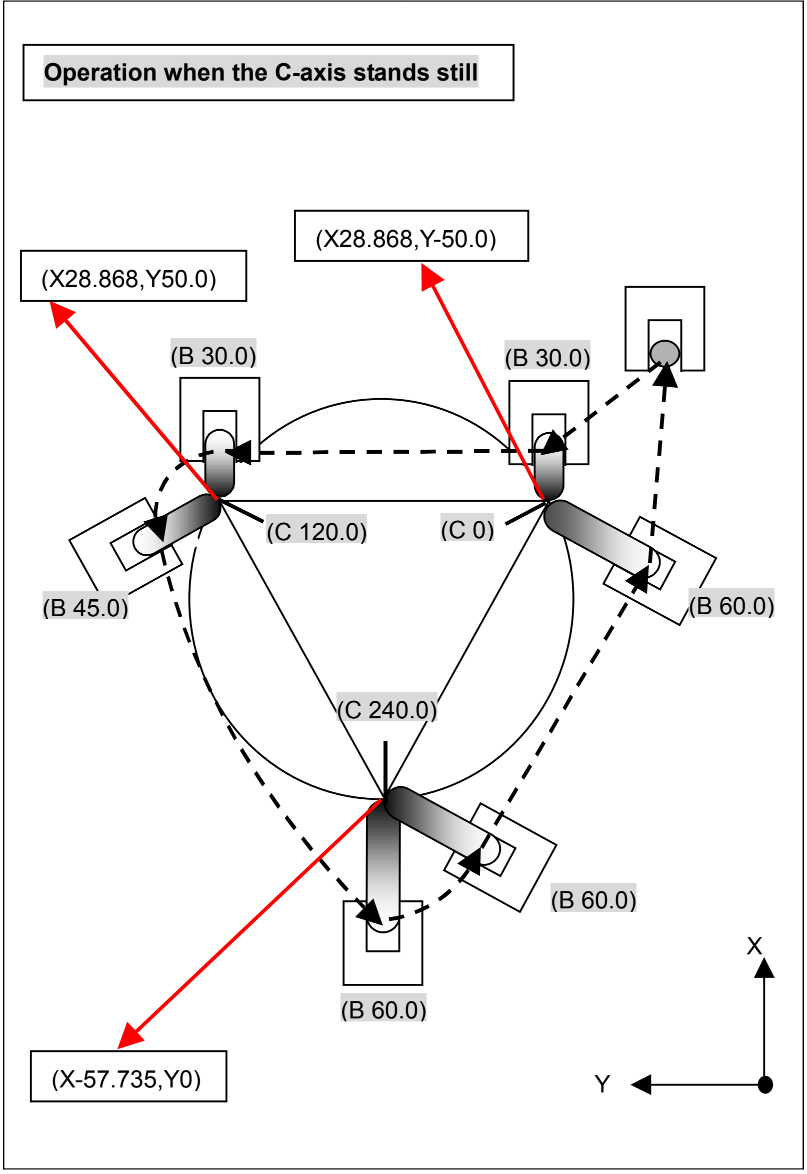
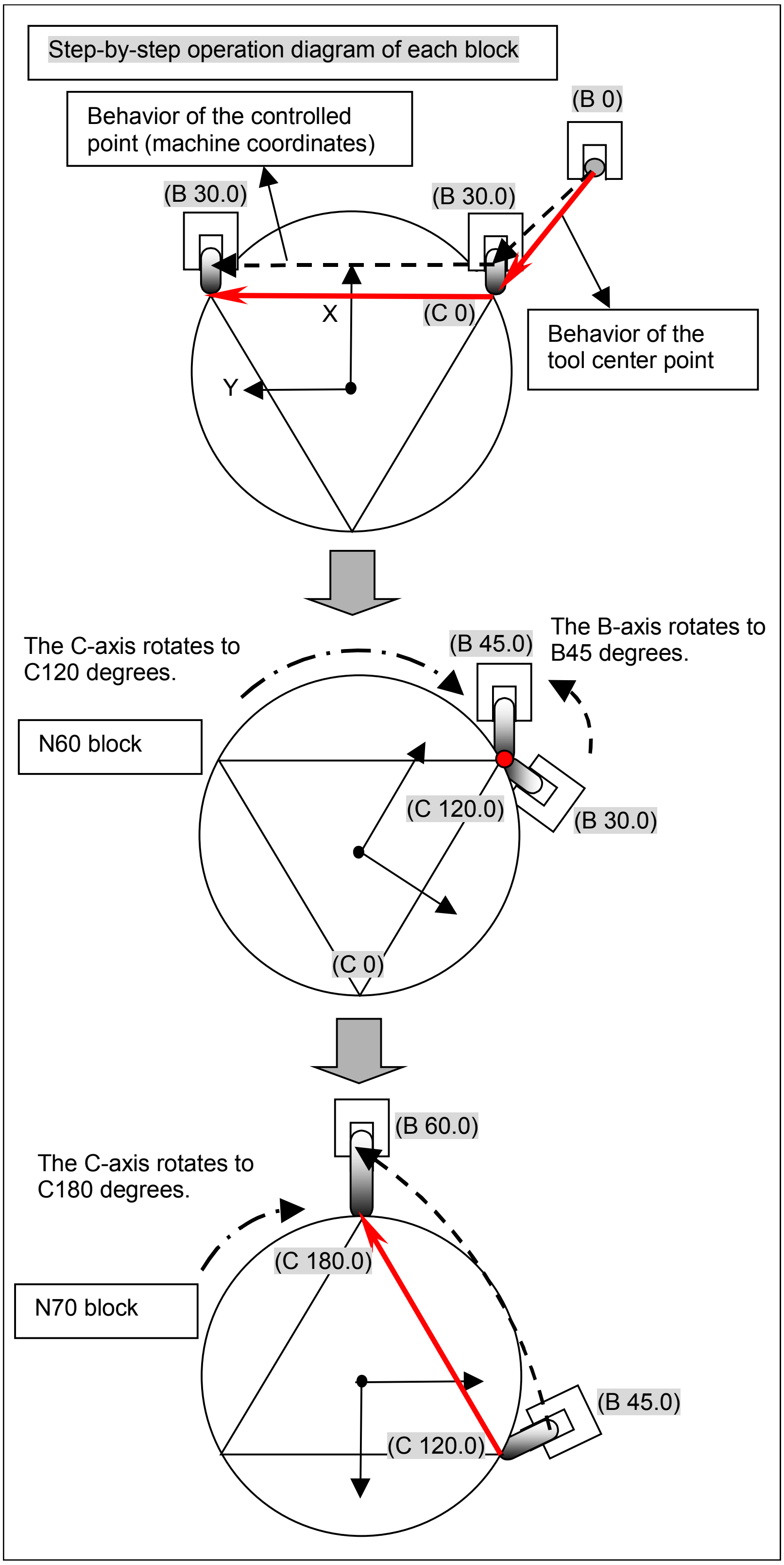


Fig.14.16.1 (j)


Step-by-step operation diagram of each block
Behavior of the controlled point (machine coordinates)
(B 0)
(B 30.0)
(B 30.0)
X
(C 0)
Y
Behavior of the tool center point
The C-axis rotates to C120 degrees.
(B 45.0)
The B-axis rotates to B45 degrees.
N60 block
(C 120.0)
(B 30.0)
(C 0)
(B 60.0)
The C-axis rotates to C180 degrees.
(C 180.0)
N70 block
(B 45.0)
(C 120.0)

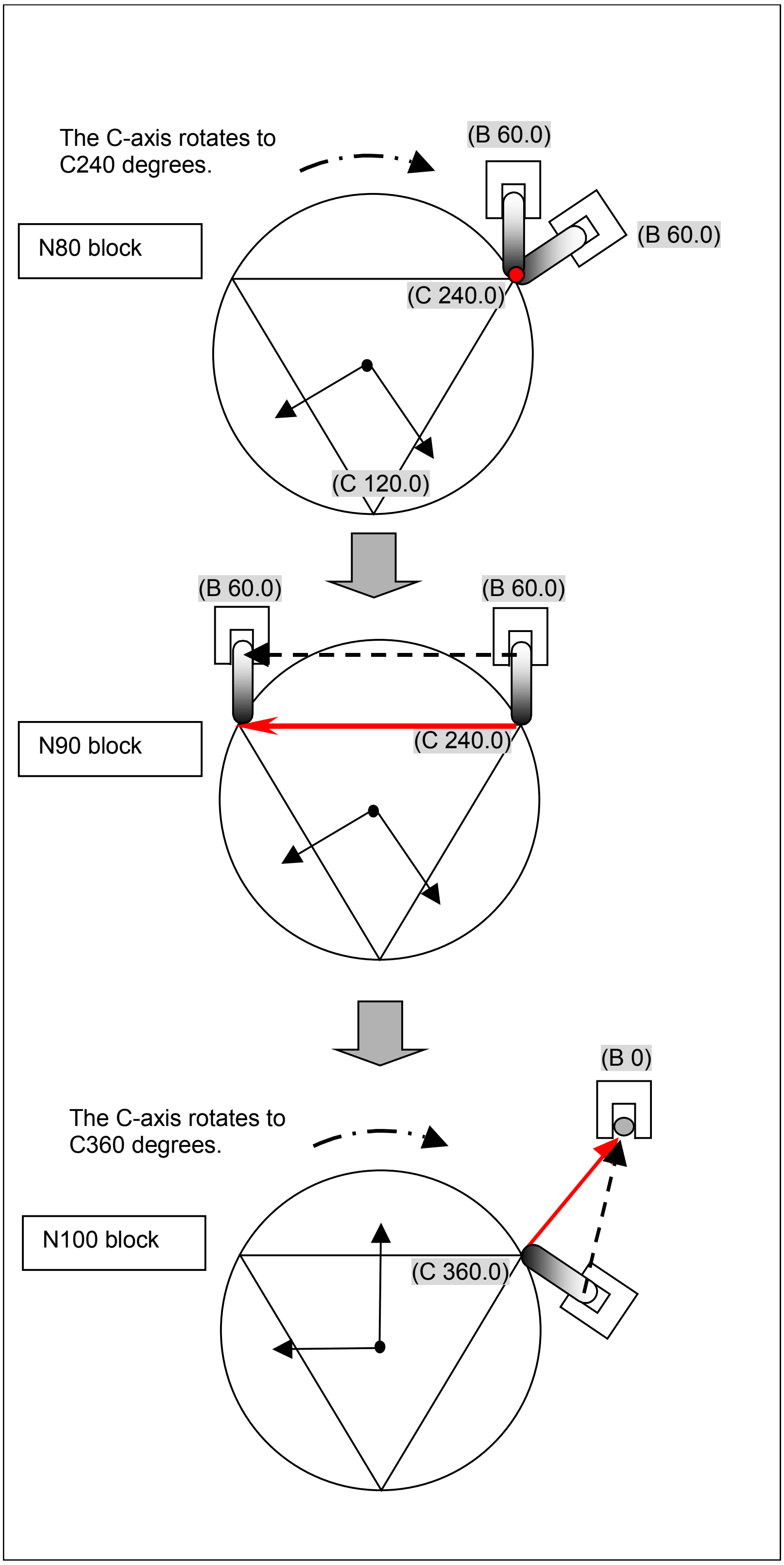

The C-axis rotates to C240 degrees.
(B 60.0)
N80 block
(B 60.0)
(C 240.0)
(C 120.0)
(B 60.0)
(B 60.0)
N90 block
(C 240.0)
(B 0)
The C-axis rotates to C360 degrees.
N100 block
(C 360.0)

Fig.14.16.1 (k)

Restrictions

- Deceleration at a corner

When tool center point control is in use, the controlled point may move on a curved line even if a straight-line command is issued. Some commands may cause the tool center point to make a sharp turn.

For this reason, the tool may be decelerated if a low value is set as a permissible speed difference (parameter No. 1478) or a permissible acceleration (parameter No. 1663) for a corner.

- Manual intervention

If manual intervention is made to a rotary axis with the manual absolute switch at the OFF position, a compensation vector is obtained using the position where the tool was before the manual intervention.

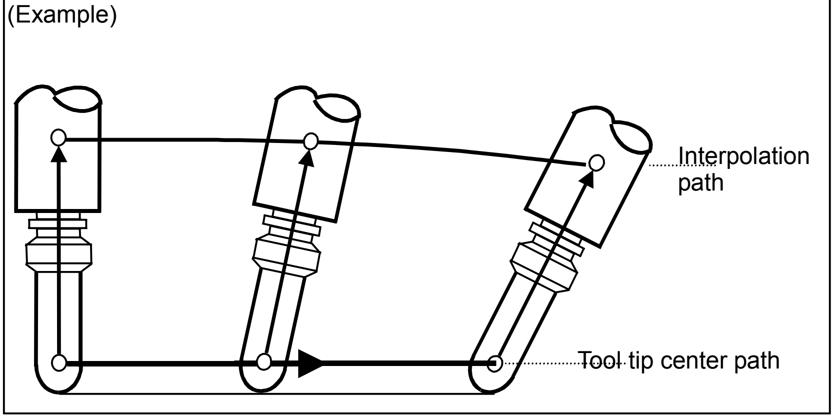
(Example)

> If a manual intervention of 1.0 degree is made to the C-axis at a position of C = 30.0 with the manual absolute switch at the OFF position, the C-axis is set to 31.0 in the machine coordinate system. When automatic operation is resumed, the C-axis is set to 30.0 in the absolute coordinate system. Therefore, C = 30.0 is used in compensation vector calculation.

- Look-ahead acceleration/deceleration before interpolation

When using tool center point control, basically, use look-ahead acceleration/deceleration before interpolation also. When look-ahead acceleration/deceleration before interpolation is not in use, an alarm (OT0553) is raised if the tool center point control results in the tool trying to move at a speed higher than the maximum cutting feedrate.

(Example)



In the above example, speed control is performed in such a way that the tool center point moves at a specified speed, so it can move faster than the specified speed along the interpolation path. If look-ahead acceleration/deceleration before interpolation is in use, the actual speed is clamped in such a way that the maximum cutting feedrate or rapid traverse rate will not be exceeded on the interpolation path. If look-ahead acceleration/deceleration before interpolation is not in use, however, an alarm (OT0553) is raised if the tool tries to move at a speed higher than the maximum cutting feedrate on the interpolation path.

- Hypothetical axis as the table rotary axis

If a hypothetical axis is used as the table rotary axis, tool center point control is performed with the table rotary axis set to 0 degrees.

- Unusable functions

Do not use the following functions in tool center point control mode, because they do not work normally.

- Commands for axes not related to tool center point control
- The following group 01 G functions
  - Circular interpolation, Helical interpolation, Spiral/conical interpolation (G02,G03)
  - Circular threading B (G2.1,G3.1)
  - Exponential interpolation (G2.3,G3.3)
  - Involute interpolation (G2.2,G3.2)
  - Three-dimensional circular interpolation (G2.4,G3.4)
  - Spline interpolation (G6.1)
  - Smooth interpolation (G5.1Q2)
  - NURBS interpolation (G6.2)
  - Threading (G33)
- Cylindrical interpolation (G7.1)
- Polar coordinate interpolation (G12.1)
- Skip function (G31 to G31.9)
- Normal-direction control (G41.1,G42.1)
- Single direction positioning (G60)
- Workpiece coordinate system (G54 to G59)
- Canned cycle (G73 to G80)
- Rigid tapping(G84.2,G84.3)
- Rotary table dynamic fixture offset (G54.2)
- Workpiece coordinate system setting (G92)
- Inverse time feed (G93)
- Feed per revolution (G95)
- Reference position return check (G27)
- Automatic reference position return check command (G28,G29,G30)
- Positioning in the machine coordinate system (G53)
- Three-dimensional cutter compensation (G41.2, G42.2)
- Three-dimensional coordinate conversion (G68)
- Tilted working plane command (G68.2)

Alarm And message

Number	Message	Contents
PS0217	ILLEGAL OFFSET VALUE	The offset number is incorrect.
PS1100	ILLEGAL PARAMETER OF MACHINE COMPONENT	A parameter (No. 6161 to No. 6195 or No. 7540 to No. 7548) for configuring the machine is incorrect.
OT553	EXCESS VELOCITY IN G43.4/G43.5	Tool center point control resulted in an axis trying to move faster than the maximum cutting feed rate.

14.17 CONTROL POINT COMPENSATION OF TOOL LENGTH COMPENSATION ALONG TOOL AXIS AND TOOL CENTER POINT CONTROL

Normally, the control point of tool length compensation along the tool axis and tool center point control is the point of intersection of the centers of two rotation axes. The machine coordinates also indicate this control point.

This section explains the compensation that is performed when the centers of the two rotation axes do not intersect and also explains how to place the control point at a convenient position on the machine.

Description

- Compensation of the rotation centers of two rotation axes

Compensation is performed when the rotation centers of two rotation axes do not match.

The length from the tool mounting position to the first rotation axis center is set as the tool holder offset value in parameter No. 7548.

The vector from the first rotation axis center to the second rotation axis center is set as the rotation center compensation vector in parameter No. 7519. Since parameter No. 7519 is an axis type parameter, the compensation amount for three axes (X, Y, and Z) can be set in this parameter.

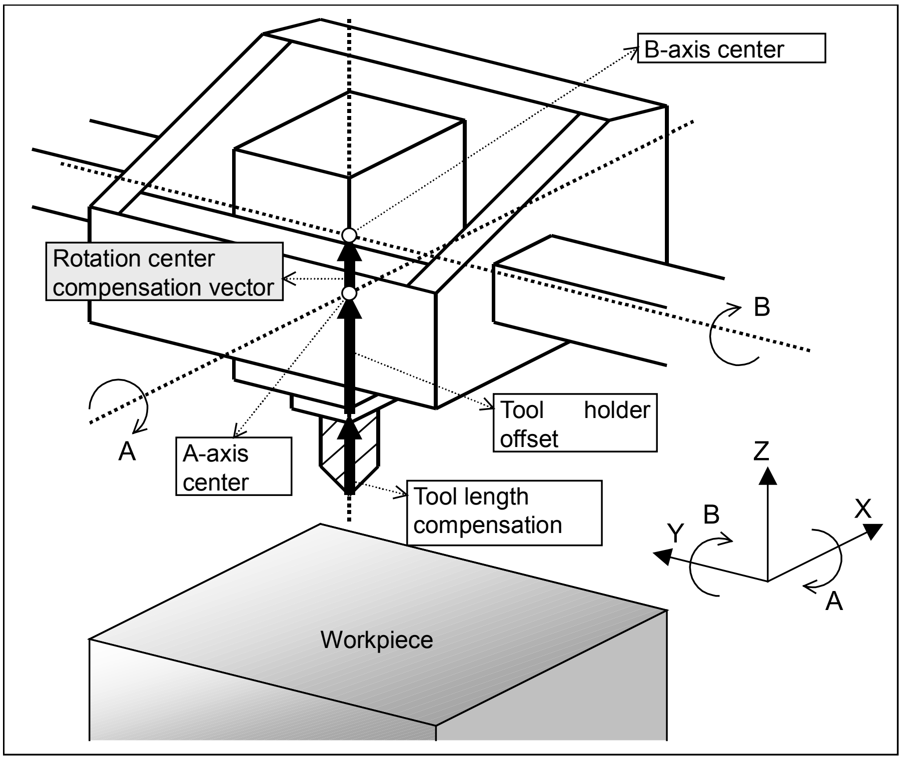


Fig. 14.17 (a) Compensation of Rotation Centers of Two Rotation Axes

According to the machine type, set the values listed in the following table:

Table 14.17 (a) Setting the Tool Holder Offset and Rotation Center Compensation Vector

Machine type	Tool holder offset Parameter No. 7548	Rotation center compensation vector Parameter No. 7519
(1) A- and C- axes. Tool axis is Z-axis.	Length from tool mounting position to A-axis center	Vector from A-axis center to C-axis center
(2) B- and C-axes. Tool axis is Z-axis.	Length from tool mounting position to B-axis center	Vector from B-axis center to C-axis center
(3) A- and B-axes. Tool axis is X-axis.	Length from tool mounting position to B-axis center	Vector from B-axis center to A-axis center
(4) A- and B-axes. Tool axis is Z-axis. B-axis is master.	Length from tool mounting position to A-axis center	Vector from A-axis center to B-axis center
(5) A- and B-axes. Tool axis is Z-axis. A-axis is master.	Length from tool mounting position to B-axis center	Vector from B-axis center to A-axis center

NOTE

When using the spindle center compensation described below, set the length from the tool mounting position to the spindle center as the tool holder offset.

- Spindle center compensation

Compensation of the spindle center is performed.

The amount of spindle center compensation is set in parameter No. 7520. Since parameter No. 7520 is an axis type parameter, the compensation amount for three axes (X, Y, and Z) can be set in this parameter.

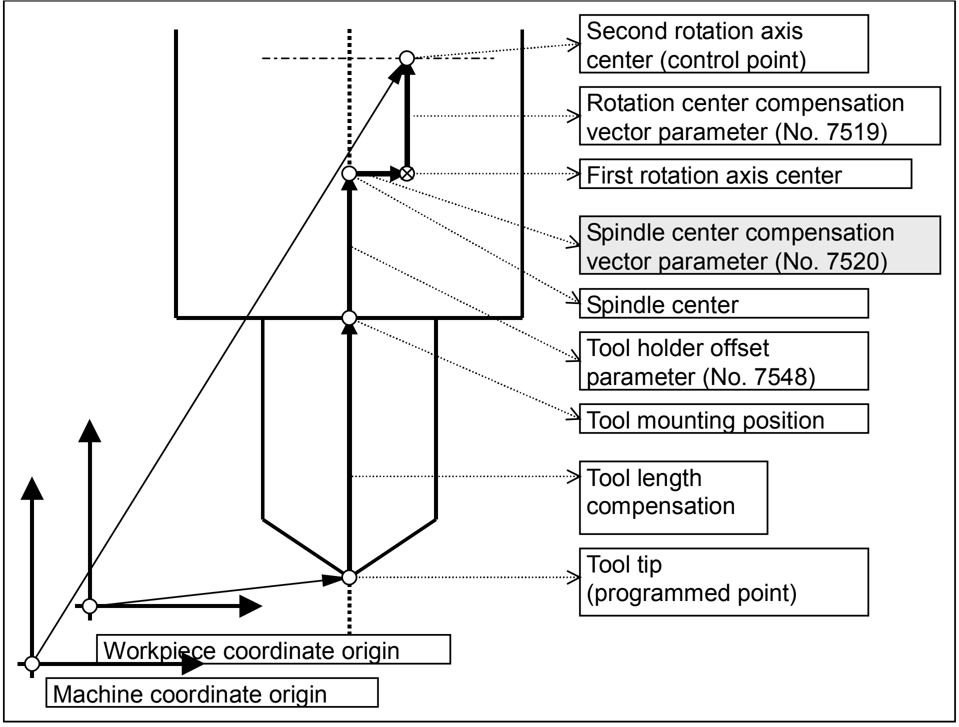

Second rotation axis center (control point)
Rotation center compensation vector parameter (No. 7519)
First rotation axis center
Spindle center compensation vector parameter (No. 7520)
Spindle center
Tool holder offset parameter (No. 7548)
Tool mounting position
Tool length compensation
Tool tip (programmed point)
Workpiece coordinate origin
Machine coordinate origin

Fig. 14.17 (b) Spindle Center Compensation

Shifting the control point

Conventionally, the center of a rotation axis was used as the control point. The control point can now be shifted as shown in the figure below.

Then, when the rotation axis is at the 0-degree position also in tool length compensation along the tool axis (G43.1), the control point can be set to the same position as in ordinary tool length compensation (G43).

The control point here is indicated by machine coordinates.

When linear interpolation is specified, for example, this control point moves linearly.

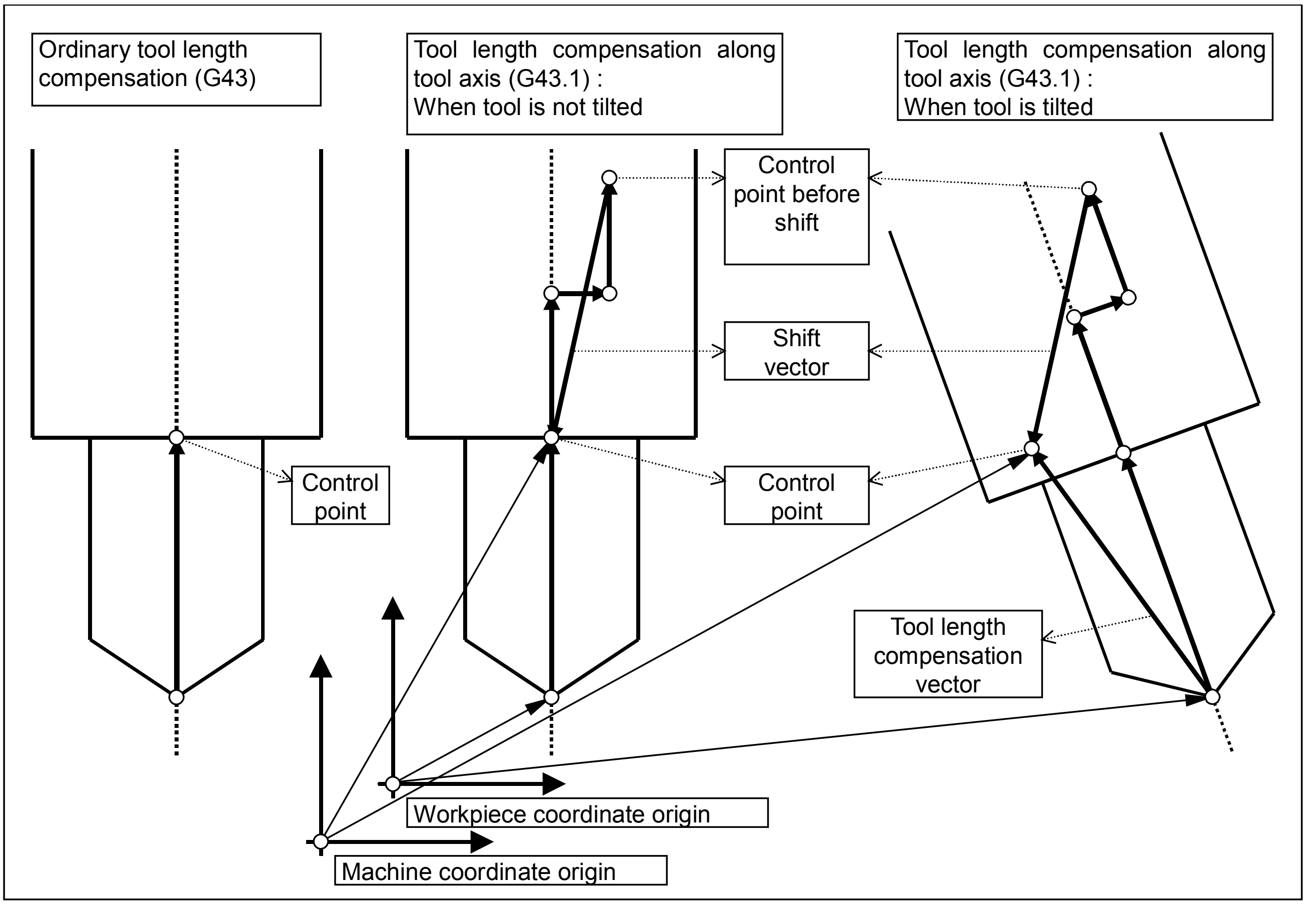

Ordinary tool length compensation (G43)
Tool length compensation along tool axis (G43.1) : When tool is not tilted
Tool length compensation along tool axis (G43.1) : When tool is tilted
Control point before shift
Shift vector
Control point
Control point
Tool length compensation vector
Workpiece coordinate origin
Machine coordinate origin

Fig. 14.17 (c) Shift of Control Point

The method for shifting the control point can be selected using the following parameters:

Table 14.17 (b) Methods of Shifting the Control Point

Bit 5 (SVC) of parameter No. 7540	Bit 4 (SBP) of parameter No. 7540	Shift of control point
0	-	As normal, the control point is not shifted.
1	0	The control point is shifted, and the shift vector is calculated automatically, as follows: - (rotation center compensation vector (parameter No. 7519)) + spindle center compensation vector (parameter No. 7520)) + tool holder offset along tool axis (parameter No. 7548)))
1	1	The control point is shifted, and the shift vector is the vector set in parameter No. 7745.

- Tool center point control

Control point compensation of tool length compensation along the tool axis is also enabled for tool center point control.

Reference

This Manual	II.14.12	TOOL LENGTH COMPENSATION ALONG TOOL AXIS
	II.14.16	TOOL CENTER POINT CONTROL

14.18 GRINDING WHEEL WEAR COMPENSATION

On a specified compensation plane, a compensation vector is created on an extension of a straight line starting from a specified point (compensation center) toward a command end point.

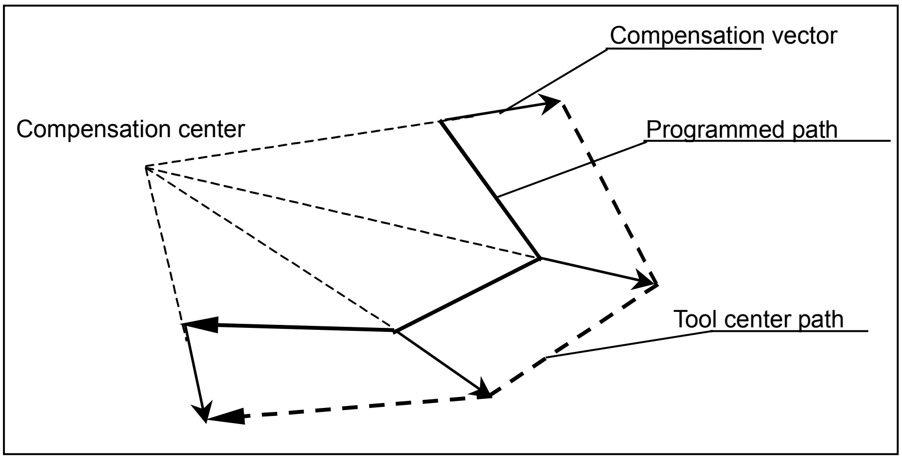


Fig. 14.18 (a) Grinding Wheel Wear Compensation

Format

- Grinding wheel wear compensation (start of grinding wheel wear compensation)

G41 P_ D_ ;

P_ : Number of compensation center position
P1 (Selection of first compensation center)
P2 (Selection of second compensation center)
P3 (Selection of third compensation center)
D_ : Non-zero offset number

- Canceling the compensation vector

D0;

Cancels only the compensation vector.

- Maintaining the compensation vector

G40 ;

When a D code with a non-zero value has been specified, specifying G40 sets the mode in which the compensation vector is maintained.

- Canceling grinding wheel wear compensation

G40 D0 ;

Description

- Grinding wheel wear compensation (start of grinding wheel wear compensation)

Up to three compensation center positions can be set. Set the coordinates (in the workpiece coordinate system) of these compensation center positions in parameter Nos. 6050 to 6055.

With the Pn (n = 1, 2, or 3) command, select the number of a compensation center position.

P1	:	Selects the first compensation center.
P2	:	Selects the second compensation center.
P3	:	Selects the third compensation center.

When G41P_ (selection of a compensation center position) and a D code with a non-zero value are specified, a compensation vector is created, and movement takes place even if there is no move command specified in the block.

The D code may be specified in a block before or after G41P_.

Example

G41 P1 ;	Selects the first compensation center.
D2 ;	Creates a compensation vector and performs movement.

NOTE

1 Specify the P_ command and G41 command at the same time. If they are not specified at the same time, or if a value other than 1 to 3 is specified in the P_ command, alarm PS618 is issued.

2 When the selected compensation center position and absolute coordinate match on an axis, the compensation vector component on that axis becomes 0.

- Canceling the compensation vector

Specifying D0 cancels the compensation vector.

When D0 is specified in a block, a movement due to the cancellation takes place even if the block contains no move command.

Then, when a D code with a non-zero value is specified again, a compensation vector is created, and movement takes place.

- Maintaining the compensation vector

When a D code with a non-zero value has been specified, specifying G40 sets the mode that maintains the created compensation vector.

When compensation vector maintenance mode has been entered, the command end position is shifted by the amount of the maintained vector.

To create a new compensation vector again, specify the grinding wheel wear compensation start command (G41 P_ D_). To cancel the maintained vector, specify D0. Specifying D0 cancels grinding wheel wear compensation, therefore, thus canceling the maintained vector. In this case, movement due to the cancellation occurs.

- Canceling grinding wheel wear compensation

When G40 and D0 are specified at the same time, the compensation vector is canceled, movement due to the cancellation occurs, and then grinding wheel wear compensation is canceled.

When D0 has been canceled, specifying G40 cancels grinding wheel wear compensation without causing movement due to the compensation vector cancellation. This is because the compensation vector has already been canceled by the D0 command.

- Grinding wheel wear compensation status

G code	D code	Mode and its meaning
G41	Other than 0	Grinding wheel wear compensation mode (A compensation vector is created.)
	0	Compensation vector cancel mode (No compensation vector is created.)
G40	Other than 0	Compensation vector maintenance mode (A compensation vector is maintained.)
	0	Grinding wheel wear compensation cancel mode (Grinding wheel wear compensation is canceled.)

- Compensation plane

In grinding wheel wear compensation mode or compensation vector maintenance mode, a compensation vector is always created for the axes on the compensation plane determined by parameter Nos. 6056 and 6057.

If an axis on the compensation plane is changed in this mode, alarm PS619 is issued in a block for obtaining the movement end position.

- Compensation vector

A compensation vector is created only on the plane (compensation plane) of the axes (compensation axes) set in parameter Nos. 6056 and 6057.

On an extension of a straight line starting from the compensation center toward the command end position, a compensation vector is created, where the length of the vector equals the offset value specified by the offset number in a D code. (See Fig. 14.18(a).)

If the offset value is negative, a compensation vector whose direction is from the compensation center toward the command end position is added to the command end position. (See Fig. 14.18(b).)

If the offset value is positive, a compensation vector whose direction is from the command end position toward the compensation center is added to the command end position. (See Fig. 14.18(c).)

The direction of the compensation vector can be changed by bit 3 (WCD) of parameter No. 6008.

> ⚠ **CAUTION**
> The offset value is used as a radius. It can also be used as a diameter by setting bit 0 (ODI) of parameter No. 6008 to 1.

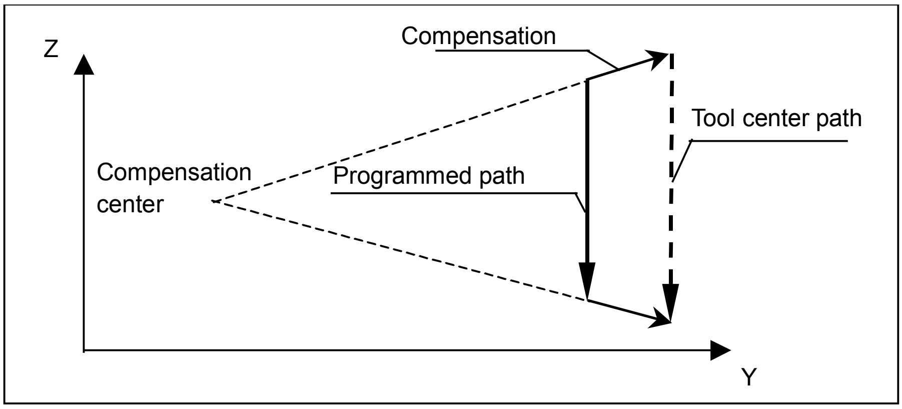


Fig. 14.18 (b) Offset Value (-)

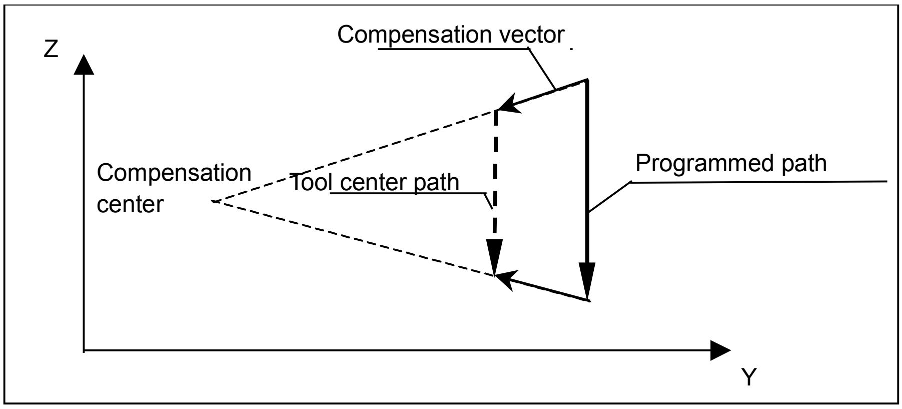


Fig. 14.18 (c) Offset Value (+)

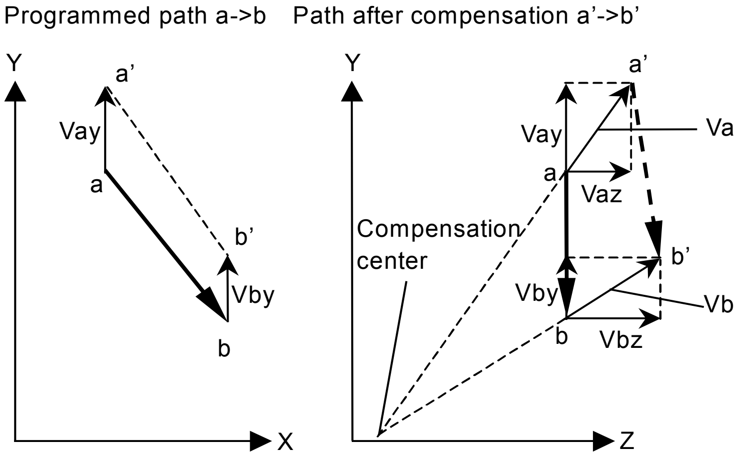
- Compensation plane and plane selection by G17/G18/G19

The creation of a compensation vector is not related to plane selection by G17/G18/G19.

For example, while circular interpolation is being performed on the XY (G17) plane, compensation can be applied to a compensation plane (such as the YZ plane). In grinding wheel wear compensation mode, if the creation of a compensation vector results in a change in the vector component, movement along a compensation axis is performed even when no move command is specified for that axis.

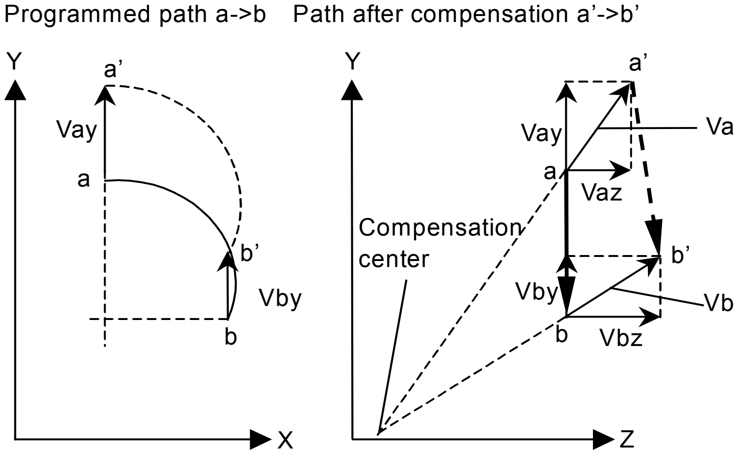
Example 1

With the Y- and Z-axes set as the compensation axes, a linear interpolation command is executed for the X- and Y-axes.



Example 2

With the Y- and Z- axes set as the compensation axes, an arc command is executed for the X- and Y- axes.



- Circular interpolation/helical interpolation

When circular interpolation (G02/G03) is specified in grinding wheel wear compensation mode, the radius at the start point of an arc differs from the radius at the end point, which prevents a correct arc from being formed, unless the compensation center and arc center match. As a result, a spiral is formed.
Similarly, this occurs in helical interpolation.

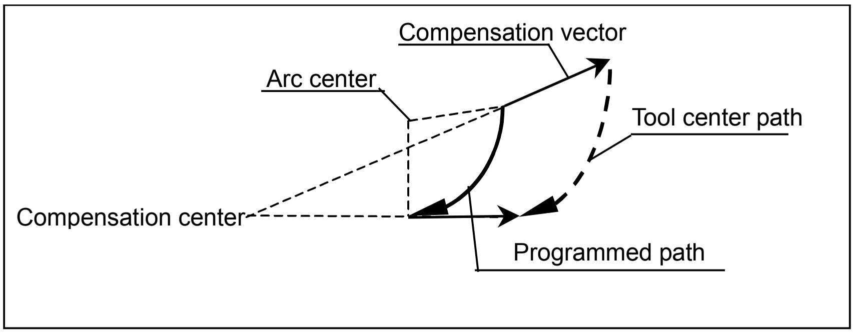


Fig. 14.18 (d) Arc Interpolation in Grinding Wheel Wear Compensation Mode

Also, for the value obtained after compensation, a check of the arc radius error limit (parameter No. 2410) is performed.

When arc interpolation is specified in compensation vector maintenance mode, the end position and arc center are shifted by the amount of the compensation vector. As a result, an arc is formed instead of a spiral.

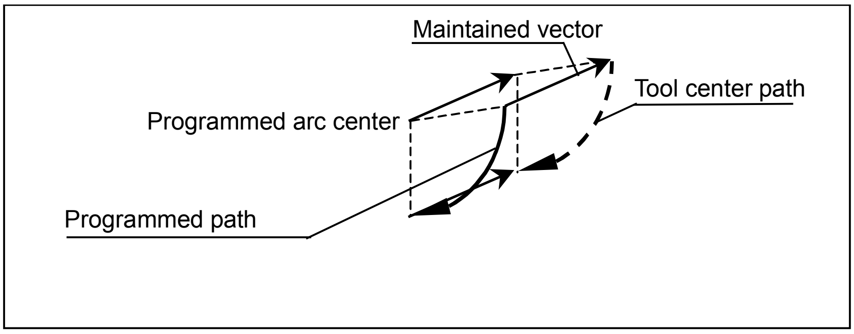


Fig. 14.18 (e) Arc Command in Compensation Vector Maintenance Mode (G40)

- Exponential interpolation

When exponential interpolation is specified in grinding wheel wear compensation mode, a compensation vector is created for each of the points approximating a straight line in exponential interpolation.

- Power-up and reset

When a power-up or reset operation is performed, grinding wheel wear compensation cancel mode is set.

- Available compensation functions

The commands listed below can be used in grinding wheel wear compensation mode. In these command modes, grinding wheel wear compensation can also be used.
- Tool length compensation (G43, G44, G49)
- Position offset (G45 to G48)

Restrictions

- Changing a compensation axis

Before changing a compensation axis, set compensation vector cancel mode, or cancel grinding wheel wear compensation.

- Changing the coordinate system

Before changing the coordinate system, set compensation vector cancel mode, or cancel grinding wheel wear compensation.

- Commands related to reference position return

Before issuing the following commands, set compensation vector cancel mode, or cancel grinding wheel wear compensation:
- Reference position return check (G27)
- Reference position return (G28)
- Second reference position return (G30)
- Return from the reference position (G29)
- Floating reference position return (G30.1)

- Relation with other compensation functions

For a system having the grinding wheel wear compensation function, the following functions cannot be added:

- Functions belonging to G code group 07, except this function Cutter compensation (G40, G41, G42), three-dimensional tool compensation (G40, G41), three-dimensional cutter compensation (G41.3, G41.2, G42.2), tool length compensation in a specified direction (G41), and so forth

The grinding wheel wear compensation function cannot be used when the tool offset function by tool number is valid. Before using the grinding wheel wear compensation function, disable the tool offset function by tool number by setting bit 5 (NOT) of parameter No. 0011 to 1.

The commands listed below are invalid for the coordinates of the compensation center of grinding wheel wear compensation. When these commands are executed in grinding wheel wear compensation mode, a compensation vector is created on an extension of the line connecting the compensation center and the command end position of the following functions:
- Scaling (G51)
- Coordinate system rotation (G68)

- Relation with compensation functions

The commands listed below cannot be used in grinding wheel wear compensation function mode. Before using these commands, cancel grinding wheel wear compensation. Also, grinding wheel wear compensation cannot be used in a mode of these commands.

- Three-dimensional circular interpolation (G02.4, G03.4)
- Hypothetical axis interpolation (G07)
- Polar coordinate interpolation (G12.1, G13.1)
- Cylindrical interpolation (G07.1)
- Spline interpolation (G06.1)
- Smooth interpolation (G05.1)
- NURBS interpolation (G06.2)

- Relation with functions that simplify programming

The following command is invalid for the coordinates of the compensation center of grinding wheel wear compensation:

- Programmable mirror image (G51.1)
  A compensation vector is created on an extension of the line connecting the compensation center and the command end position to which a programmable mirror image is applied.

The commands listed below cannot be used in grinding wheel wear compensation function mode. Before using these commands, cancel grinding wheel wear compensation. Also, in a mode of these commands, grinding wheel wear compensation cannot be used.

- Chamfering at an arbitrary angle, corner rounding
- Figure copy (G72.1, G72.2)
- Normal direction control (G40.1, G41.1, G42.1)
- Three-dimensional coordinate conversion (G68, G69)

- Relation with measurement functions

The commands listed below cannot be used in grinding wheel wear compensation function mode. Before using these commands, cancel grinding wheel wear compensation.

- Skip function/multiple-command skip function (G31)
- Multistage skip (G31.1 to G31.4)
- Automatic tool measurement (G37)

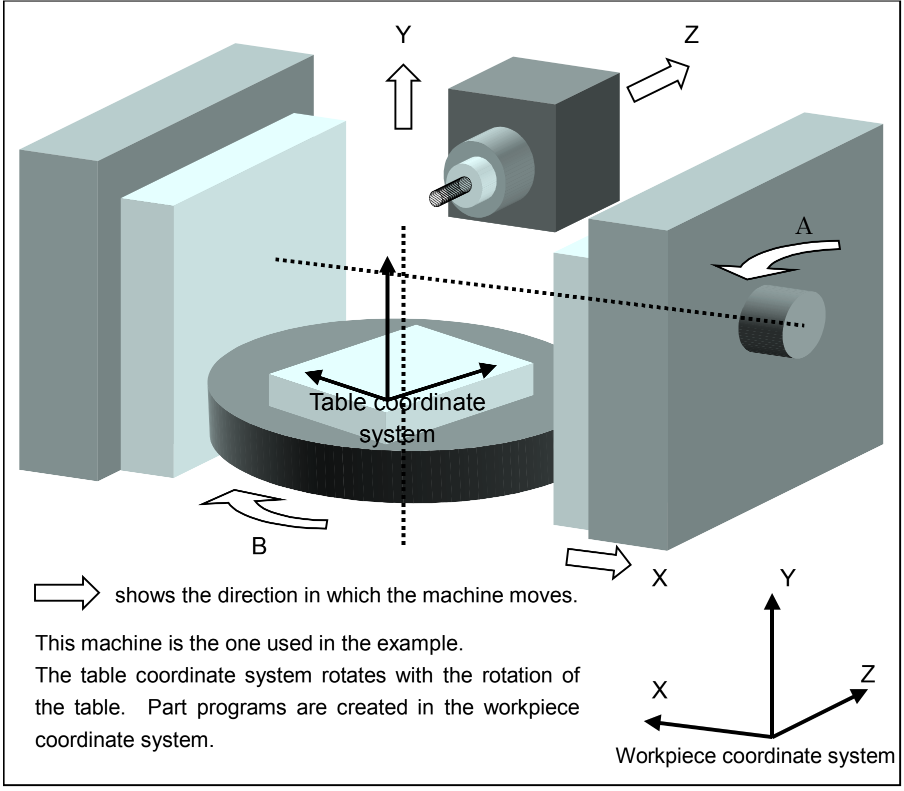
- Relation with other functions

- Background drawing
  A program producing a spiral tool center path cannot be drawn correctly.
- Binary input operation by a remote buffer
  This function cannot be used in grinding wheel wear compensation function mode. Before using this function, cancel grinding wheel wear compensation.

14.19 CUTTER COMPENSATION FOR ROTARY TABLE

Overview

For machines having a rotary table, such as that shown in the figure below, cutter compensation can be performed.



shows the direction in which the machine moves.

This machine is the one used in the example.
The table coordinate system rotates with the rotation of the table. Part programs are created in the workpiece coordinate system.

Fig.14.19 (a) Machine having a rotary table

Format

- Startup (cutter compensation start)

G41.4 (or G42.4) IP_ D_ ;

G41.4 : Cutter compensation, left (group 07)
G42.4 : Cutter compensation, right (group 07)
IP_ : Value specified for moving an axis
D_ : Code specifying the cutter compensation amount (1 to 3 digits)

- Cutter compensation cancellation

G40 IP_ ;

G40 : Cutter compensation cancellation (group 07)
IP_ : Value specified for axis moving

- Selection of an offset plane

Offset plane	Plane selection command	IP_
XpYp	G17 ;	Xp_Yp_
ZpXp	G18 ;	Xp_Zp_
YpZp	G19 ;	Yp_Zp_

The selected plane, or two axes, must be included in the three linear axes (parameters Nos. 6140 to 6142) handled by this function.
Select the plane vertical to the tool (XpYp plane in Fig.14.19 (a)).

Description

- Cutter compensation

The cutter compensation for Rotary table basically performs operations in conformance with cutter compensation. The operations different from those of cutter compensation are described below. For a description of the specifications and cautions not mentioned here, see the description of cutter compensation.

- Startup

If the cutter compensation for Rotary table (G41.4 or G42.4, a command with a dimension word other than 0 and a D code other than D0 on the offset plane) is issued in offset cancellation mode, the CNC enters offset mode.
Startup is specified with positioning (G00) or linear interpolation (G01).
If circular interpolation (G02, G03) or involute interpolation (G02.2, G03.2) is specified, alarm PS0270 is issued.

- Offset mode

In offset mode, compensation is performed for positioning (G00) and linear interpolation (G01).

> **NOTE**
> In G41.4 or G42.4 mode, circular interpolation (G02, G03) or involute interpolation (G02.2, G03.2) cannot be specified.

- Offset mode cancellation

If a block satisfying either of the following conditions is executed in offset mode, the CNC enters offset cancellation mode.

1. G40 is specified.
2. 0 is specified for the code for specifying the cutter compensation amount (D code).

If offset cancellation is to be performed, circular interpolation (G02, G03) or involute interpolation (G02.2, G03.2) must not be specified Otherwise, alarm PS0270 is issued.

Example

- Parameter specification example

On the machine shown in Fig.14.19 (a), parameters must be specified as follows:
The axis numbers are assumed as follows: X = 1, Y = 2, Z = 3, A = 4, B = 5

Parameter No.	Setting	Description
6140	1 (X)	Axis number of linear axis 1
6141	2 (Y)	Axis number of linear axis 2
6142	3 (Z)	Axis number of linear axis 3
6143	-5 (B)	Axis number of rotation axis (first set)
6144	2 (Y)	Axis number of the linear axis corresponding to the rotation axis (first set)
6145	-4 (A)	Axis number of rotation axis (second set)
6146	1 (X)	Axis number of the linear axis corresponding to the rotary axis (second set)
6150	0.0	Reference angle for the rotation axis (first set)
6151	0.0	Reference angle for the rotation axis (second set)

- Formulas

With the above settings, the cutter compensation vector at end point N2 in the following program is calculated as follows:

(1) Part program

A part program is created in the workpiece coordinate system.

G42.4 D1
N1 Xx_1 Yy_1 Zz_1 Aa_1 Bb_1
N2 Xx_2 Yy_2 Zz_2 Aa_2 Bb_2
N3 Xx_3 Yy_3 Zz_3 Aa_3 Bb_3

(2) Definitions

$P_1 = (x_1, y_1, z_1)$, $P_2 = (x_2, y_2, z_2)$, $P_3 = (x_3, y_3, z_3)$

P_0: Origin of the table coordinate system (parameter No. 6154)

(3) Calculation of the matrix for conversion from the workpiece coordinate system to the table coordinate system

Reference angle conversion matrix

$$M_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos a_0 & \sin a_0 \\ 0 & -\sin a_0 & \cos a_0 \end{bmatrix} \begin{bmatrix} \cos b_0 & 0 & -\sin b_0 \\ 0 & 1 & 0 \\ \sin b_0 & 0 & \cos b_0 \end{bmatrix}$$

where a_0 and b_0 are the reference angles specified for parameters Nos. 6150 and 6151.

P_1 conversion matrix

$$M_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos a_1 & \sin a_1 \\ 0 & -\sin a_1 & \cos a_1 \end{bmatrix} \begin{bmatrix} \cos b_1 & 0 & -\sin b_1 \\ 0 & 1 & 0 \\ \sin b_1 & 0 & \cos b_1 \end{bmatrix}$$

P_2 conversion matrix

$$M_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos a_2 & \sin a_2 \\ 0 & -\sin a_2 & \cos a_2 \end{bmatrix} \begin{bmatrix} \cos b_2 & 0 & -\sin b_2 \\ 0 & 1 & 0 \\ \sin b_2 & 0 & \cos b_2 \end{bmatrix}$$

P_3 conversion matrix

$$M_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos a_3 & \sin a_3 \\ 0 & -\sin a_3 & \cos a_3 \end{bmatrix} \begin{bmatrix} \cos b_3 & 0 & -\sin b_3 \\ 0 & 1 & 0 \\ \sin b_3 & 0 & \cos b_3 \end{bmatrix}$$

(3) Calculation of three points P_1', P_2', P_3' used to calculate cutter compensation

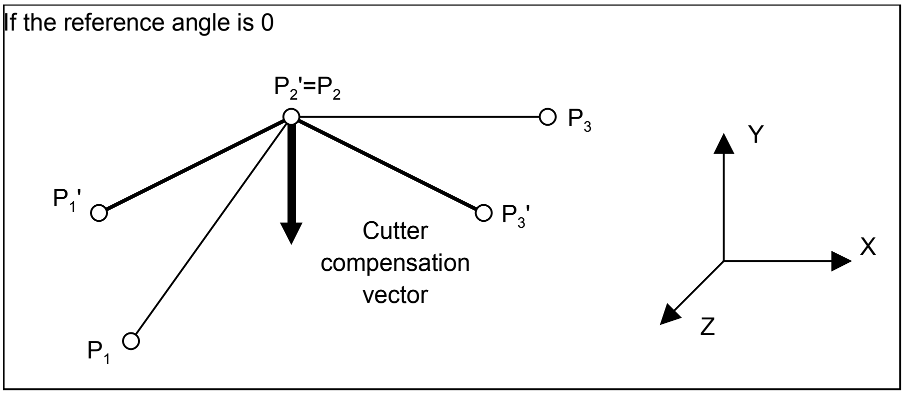
P_1, P_2, and P_3 are converted to P_1', P_2', and P_3' using the following formulas:

$$P_1' = M_2^{-1} M_1 M_0^{-1} (P_1 - P_0) + P_0$$

$$P_2' = M_0^{-1} (P_2 - P_0) + P_0$$

$$P_3' = M_2^{-1} M_3 M_0^{-1} (P_3 - P_0) + P_0$$

The cutter compensation vector is calculated using these three points.



Restrictions

- Interference check

In G41.4 or G42.4 mode, an interference check is performed using a specified position in the workpiece coordinate system and a compensation vector. The interference check avoidance function cannot be used.

- Corner arc (G39)

In G41.4 or G42.4 mode, G39 cannot be specified. Specifying G39 causes alarm PS1062 to be issued.

- Manual intervention

In G41.4 or G42.4 mode, manual intervention must not be performed on the rotation axis.

Alarm and message

No.	Message	Description
PS1062	ILLEGAL USE OF G41.4/G42.4	(1) Any of the parameters Nos. 6140 to 6146, related to the cutter compensation for Rotary table, is not correct. (2) At the start of the rotary table support of cutter compensation (G41.4/G42.4), axes other than those specified for parameters Nos. 6140 to 6142 were selected as a plane. (3) In the mode of the cutter compensation for Rotary table (G41.4/G42.4), a move command other than G00/G01 (such as G02) was issued.

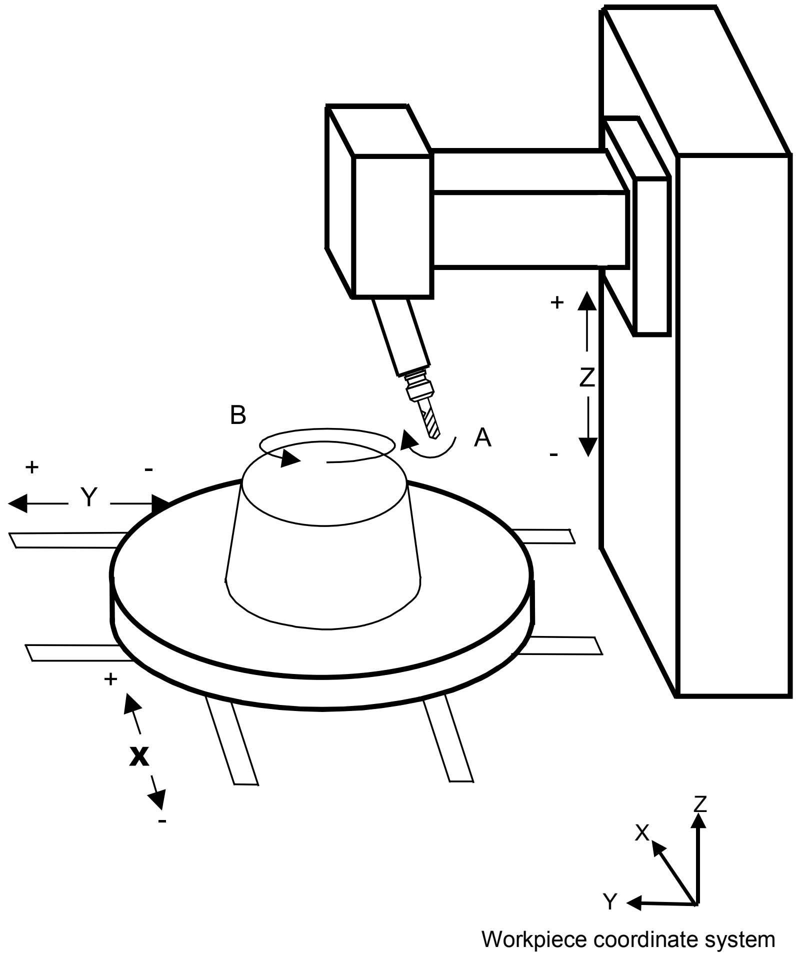
Reference item

FANUC Series 15*i*/150*i*-MB	PARAMETER MANUAL (B-63790EN)	4.22	TOOL OFFSET PARAMETERS

14.20 THREE-DIMENSIONAL CUTTER COMPENSATION FOR ROTARY TABLE

Overview

This function allows three-dimensional cutter compensation to be performed on a 5-axis machine having a rotary table and a rotation tool axis, such as that shown in the figure below.
The machine example shown in the figure below is a 5-axis machine having the tool axis A about the X-axis (the orientation of the tool axis being on the Z-axis) and the table rotation axis C about the Z-axis.
Three-dimensional cutter compensation can be performed on such a machine.



Workpiece coordinate system:
Coordinate system with which a program is to be created

Fig. 14.20 (a) Machine having tool rotation axis and table rotation axis

Format

- Startup (start of three-dimensional cutter compensation rotary table) (tool side offset)

G41.5 (or G42.5) IP_ D_ ;

G41.5	:	Cutter compensation, left (group 07)
G42.5	:	Cutter compensation, right (group 07)
IP_	:	Specified amount of axial movement
D_	:	Code specifying the amount of cutter compensation (1 to 3 digits)

- Cutter compensation cancellation

G40 IP_ ;

G40	:	Cutter compensation cancellation (group 07)
IP_	:	Specified amount of axial movement

Explanation of operations

- Cutter compensation

Basically, the operations of the function of three-dimensional cutter compensation for rotary table conform to those of three-dimensional cutter compensation. The following explanation mainly covers operations different from those of three-dimensional cutter compensation.
Thus, for those specifications and notes that are not mentioned here, refer to the explanation of three-dimensional cutter compensation.

- Startup

In offset cancel mode, issuing a command of three-dimensional cutter compensation for rotary table (G41.5 or G42.5 with a D code other than D0) causes the CNC to enter offset mode.
Startup may be specified with positioning (G00) or linear interpolation (G01).
If, during startup, an attempt is made to issue a G code other than G00 and G01, such as circular interpolation (G02 or G03) or involute interpolation (G02.2 or G03.2), an alarm will be issued.

- Offset mode

In offset mode, positioning (G00) and linear interpolation (G01) are subject to compensation.

NOTE

If, during startup, an attempt is made to issue a G code other than G00 and G01, such as circular interpolation (G02 or G03) or involute interpolation (G02.2 or G03.2), an alarm will be issued.
If, in G41.5 or G42.5 mode, a G code other than G00 and G01, such as circular interpolation (G02 or G03) or involute interpolation (G02.2 or G03.2), is issued, compensation will not be performed properly.

- Offset mode cancellation

In offset mode, executing a block satisfying either or both of the following conditions causes the CNC to enter offset cancel mode:

1 G40 is specified.
2 0 is specified as the code for specifying the amount of cutter compensation (D code).

The command used to perform offset cancellation must be G00 or G01, not any other G code such as a circular interpolation command (G02 or G03) or involute interpolation command (G02.2 or G03.2). Otherwise, an alarm will be issued.

- Formulas

The three-dimensional cutter compensation vector at the N2 end point in the following program can be calculated as follows:

1. Part program
   A part program may be created with a workpiece coordinate system.
   G42.5 D1;
   N1 Xx_1 Yy_1 Zz_1 Aa_1 Cc_1;
   N2 Xx_2 Yy_2 Zz_2 Aa_2 Cc_2;
   N3 Xx_3 Yy_3 Zz_3 Aa_3 Cc_3;

2. Definition of symbols
   $P = (x_1, y_1, z_1)$
   $Q = (x_2, y_2, z_2)$
   $R = (x_3, y_3, z_3)$
   P0: Origin of the table coordinate system (parameter No. 6154)
   A: Tool rotation axis
   C: Table rotation axis

3. Conversion of program coordinates using the table rotation axis
   (1) Conversion from the workpiece coordinate system to the table coordinate system using the table rotation axis

   The table coordinate system is the one that is fixed to the table.

   The table coordinate system will move with the rotation of the table coordinate system.

   Determining the vector at Q requires that P and R in the same state as that of Q (state at the N2 end point) be determined.

   The matrixes for conversion from the workpiece coordinate system to the table coordinate system are as follows:

   The conversion matrix for P is $M_1^{-1} = R_c(c_1)^{-1}$

   The conversion matrix for Q is $M_2^{-1} = R_c(c_2)^{-1}$

   The conversion matrix for R is $M_3^{-1} = R_c(c_3)^{-1}$

   where $R_c(c) = \begin{bmatrix} \cos c & -\sin c & 0 \\ \sin c & \cos c & 0 \\ 0 & 0 & 1 \end{bmatrix}$

   (2) Calculation of the three points P', Q', and R' used for the calculation of three-dimensional cutter compensation

   The conversion matrix for the reference angle is given as follows:

   $M_0 = R_c(c_0)$ (c_0: parameter No. 6150)

   Conversion from the table coordinate system to the workpiece coordinate system in the state of the N2 end point is given by M_2.

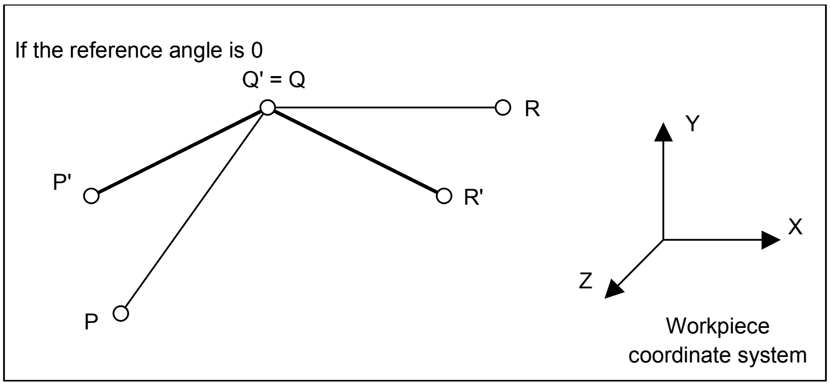
   Let the values of the three points P, Q, and R in the state of the N2 end point be P', Q', and R', then

   P’ = $M_2 M_1^{-1} M_0^{-1}$(P-P$_0$) + P$_0$

   Q’ = $M_2 M_2^{-1} M_0^{-1}$(Q-P$_0$) + P$_0$ = M_0^{-1}(Q-P$_0$) + P$_0$

   R’ = $M_2 M_3^{-1} M_0^{-1}$(R-P$_0$) + P$_0$

   Using these three points P', Q', and R', three-dimensional cutter compensation may be calculated.



Limitations

- Interference check

In G41.5 or G42.5 mode, an interference check is performed using a specified position in the workpiece coordinate system and a compensation vector. The interference check avoidance function cannot be used.

- Corner arc (G39)

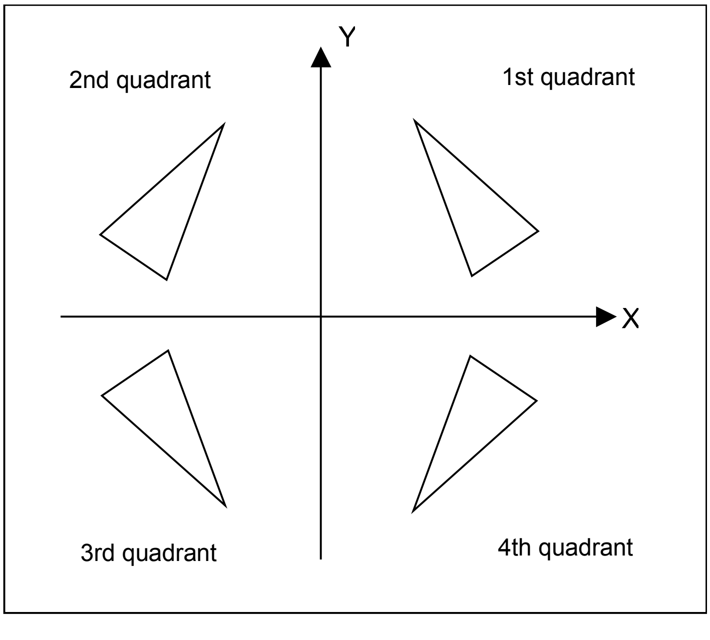
In G41.5 or G42.5 mode, G39 cannot be issued. Otherwise, an alarm will be issued.

- Manual intervention

In G41.5 or G42.5 mode, manual intervention must not be performed on the rotation axis.

- Programmable mirror image (G50.1 or G51.1)

The same mirror must be applied to both the linear and rotation axes.



For example, suppose machining on the XY plane using a composite machine with the tool rotation axis about the X-axis being assumed the A-axis and the table rotation axis about the Z-axis being assumed the C-axis.

If the basic figure in the first quadrant is to be converted into the figures in the second, third, and fourth quadrants using a programmable mirror image, the programmable mirror image must be used as follows:

	X	Y	Z	A	C
1st quadrant	Normal	Normal	Normal	Normal	Normal
2nd quadrant	Mirror	Normal	Normal	Normal	Mirror centered on 0 degrees
3rd quadrant	Mirror	Mirror	Normal	Mirror centered on 0 degrees	Mirror centered on 0 degrees
4th quadrant	Normal	Mirror	Normal	Mirror centered on 0 degrees	Normal

Alarm and message

No.	Message	Description
PS1070	ILLEGAL USE OF G41.5/G42.5	The parameters related to the three-dimensional cutter compensation for rotary table are not specified properly. An attempt was made to issue the G39 command in the mode of three-dimensional cutter compensation for rotary table(G41.5/G42.5). An attempt was made to issue a move command other than G00 and G01, such as G02, at the startup of three-dimensional cutter compensation for rotary table(G41.5/G42.5).

Reference

FANUC Series 15*i*/150*i*-MB	OPERATOR'S MANUAL(PROGRAMMING) (B-63784EN)	II.14.14	Three-dimensional Cutter Compensation

15 PROGRAMMABLE PARAMETER INPUT (G10)

General

The values of parameters can be entered in a lprogram. This function is used for setting pitch error compensation data when attachments are changed or the maximum cutting feedrate or cutting time constants are changed to meet changing machining conditions.

Format

Format	
G10L52 ;	**Parameter entry mode setting**
N_R_ ;	**For parameters other than the axis type**
N_P_R_ ;	**For axis type parameters**
:	
:	
G11 ;	**Parameter entry mode cancel**
Meaning of command	
N_ :	Parameter No. (4digids) or compensation position No. for pitch errors compensation +10,000 (5digid)
R_ :	Parameter setting value (Leading zeros can be omitted.)
P_ :	Axis No. 1 to 8 (Used for entering axis type parameters)

Explanation

- Parameter setting value (R_)

Do not use a decimal point in a value set in a parameter (R_).
a decimal point cannot be used in a custom macro variable for R_either.

- Axis No.(P_)

Specify an axis number (P_) for an axis type parameter. The control axes are numbered in the order in which they are displayed on theCNC display.
For example, specity P2 for the control axis which is displayed second.

⚠ **WARNING**

1 Before changing the pitch error compensation data or backlash compensation data, disable pitch error compensation or backlash compensation (return to the machine zero point). If the data is changed while compensation is enabled, the machine position will shift.

2 If changing of the grid shift is specified, the machine moves by the difference between the new grid shift and previous grid shift.

NOTE

1 Other NC statements cannot be specified while in parameter input mode.

2 In the parameter input mode, a fixed-point number cannot be specified in address R

Example

1. Set bit 0 (CIP) of bit type parameter No.1000

```
G10L52 ;    Parameter entry mode
N1000 R00000001 ;    SBP setting
G11 ;    cancel parameter entry mode
```

2. Change the values for the Z-axis (3rd axis) and A-axis (4th axis) in axis type parameter No. 5222(the coordinates of stored stroke limit 2 in the positive direction for each axis).

```
G10L52 ;    Parameter entry mode
N5222 P3 R45.0 ; Modify Z axis
N5222 P4 R12.0 ; Modify A axis
G11 ;    cancel parameter entry mode
```

Cautions (Compatibility with Series 15-B)

⚠ **CAUTION**

1 Some parameters for the Series 15*i* are not compatible with the Series 15-B. Before specifying parameters to use this function, be sure to refer to the description on the parameters for the Series 15*i*.

2 G10L50 (format for the Series 15-B) can be used for those parameters that are fully compatible with the Series 15-B.

16 MEASUREMENT FUNCTIOM

16.1 SKIP FUNCTION (G31)

Linear interpolation can be commanded by specifying axial move following the G31 command, like G01. If an external skip signal is input during the execution of this command, execution of the command is interrupted and the next block is executed. The skip function is used when the end of machining is not programmed but specified with a signal from the machine, for example, in grinding. It is used also for measuring the dimensions of a workpiece.

Format

G31 IP_ ;

G31 : One-shot G code (If is effective only in the block in which it is specified)

Explanation

The coordinate values when the skip signal is turned on can be used in a custom macro because they are stored in the custom macro system variable #5061 to #5080, as follows : (In a system with more than 20 axes, #100151 through #100174 are used.)

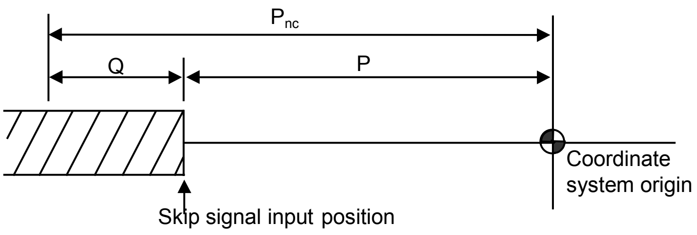
#5061	1st axis coordinate value
#5062	2nd axis coordinate value
#5063	3rd axis coordinate value
:	
#5080	20th axis coordinate value

The feedrate of the block in which G31 is specified may be one of the following, depending on the setting of the bit3 (SKF) in parameter No. 1400 :

(1) Feedrate specified by F (The feedrate may be specified either in or before the block containing the G31 code.)
(2) Feedrate set in parameter No. 1428

- Servo delay compensation

When a skip signal is entered, the CNC internally stores the current position. However, the current position stored in the CNC includes a servo system delay, so that the current position is shifted from the machine position by the servo system delay. This shift amount can be found from the positional deviation value held in the servo system and the number of remaining pulses due to a feedrate increase/decrease made internally by the CNC. When this shift amount is considered, the need to include a servo system delay in a measurement error is eliminated.



P_{nc} : Current position in the CNC when the skip signal is turned on (mm or inch)
P : Distance to be measured (mm or inch)
Q : Servo delay (mm or inch)

When bit 7 (SEB) of parameter No. 7300 is set, the CNC internally makes the following calculation :

$$P = P_{nc} - Q$$

The distance P to be measured can be read using a macro variable. So, a measurement error consists of only skip signal detection variations.

⚠ CAUTION

To increase tool position precision when a skip signal is entered, automatic acceleration/deceleration is not performed if a parameter-set feedrate is used as the feedrate used with the G31 command.

NOTE

If G31 is specified when cutter compensation is applied, an alarm is issued. Cancel cutter compensation before specifying G31.

Example

- The next block to G31 is an incremental command

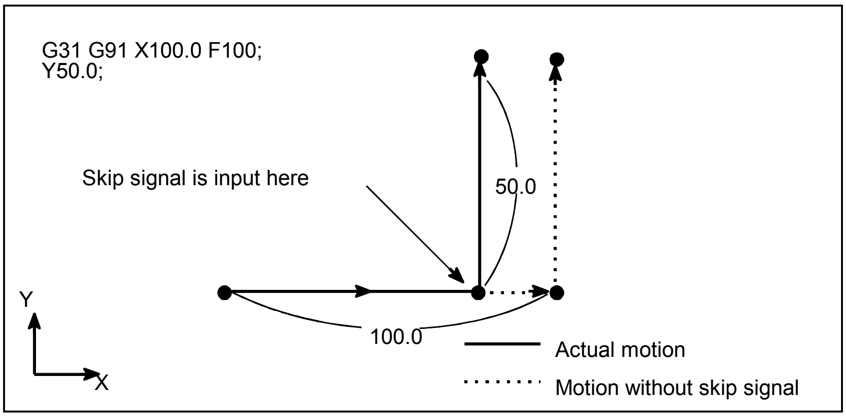


Fig.16.1 (a) The next block is an incremental command

- The next block to G31 is an absolute command for 1 axis

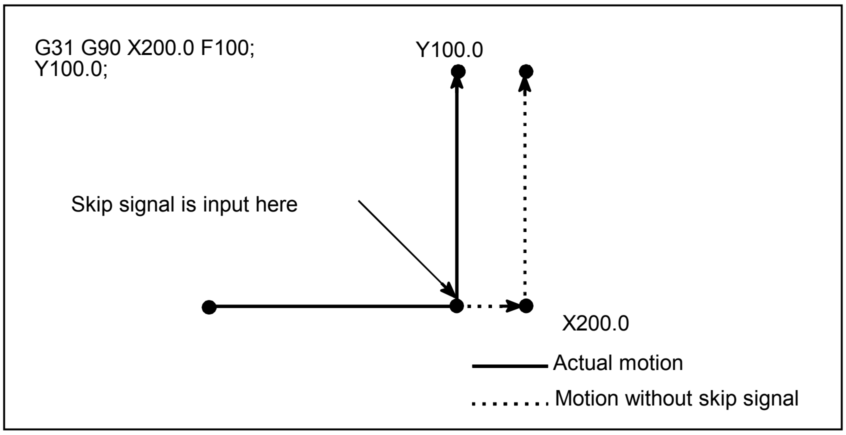


Fig.16.1 (b) The next block is an absolute command for 1 axis

- The next block to G31 is an absolute command for 2 axes

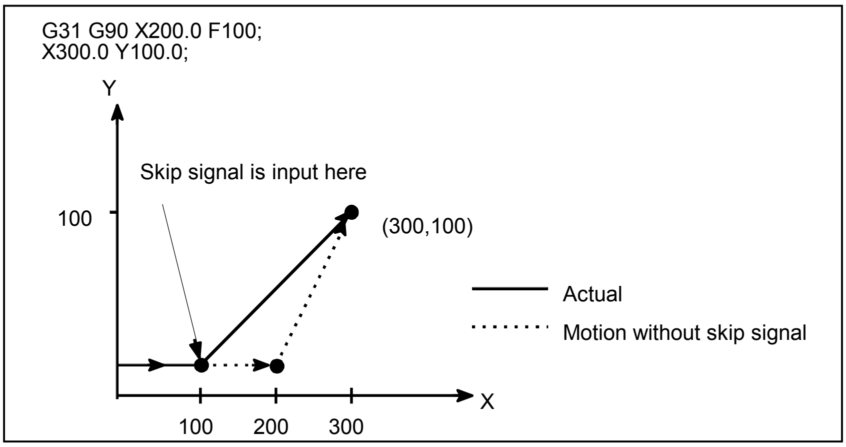


Fig.16.1 (c) The next block is an absolute command for 2 axes

16.2 SKIPPING THE COMMANDS FOR SEVERAL AXES

Move commands can be specified for several axes at one time in a G31 block. If an external skip signal is input during such commands, the command is canceled for all specified axes and the next block is executed.

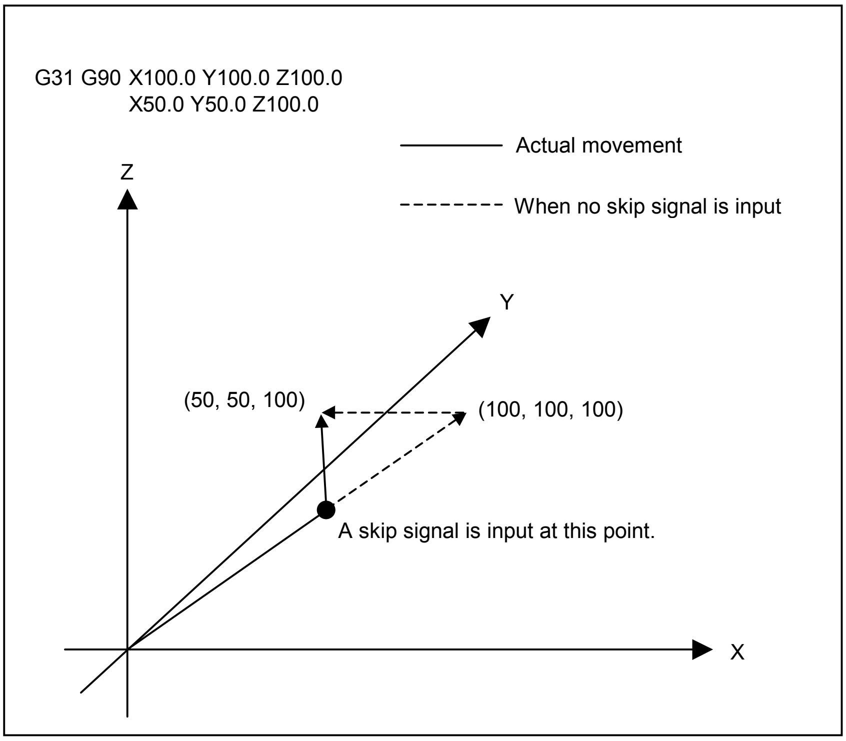
The position for each specified axis where a skip signal is input is set in the macro variable for the axis(#5061 through #5080. #100151 through #100174 for a system with more than 20 axes

Format

G31 IP_ ;

G31 : One-shot G code (valid only in the block in which it is specified)

Example


G31 G90 X100.0 Y100.0 Z100.0
X50.0 Y50.0 Z100.0
Actual movement
When no skip signal is input
Z
Y
X
(50, 50, 100)
(100, 100, 100)
A skip signal is input at this point.

16.3 HIGH SPEED SKIP SIGNAL (G31)

The skip function operates based on a high-speed skip signal (connected directly to the NC; not via the PMC) instead of an ordinary skip signal. In this case, up to eight signals can be input.
Delay and error of skip signal input is 0 - 2 msec at the NC side (not considering those at the PMC side). This high-speed skip signal input function keeps this value to 0.1 msec or less, thus allowing high precision measurement.
For details, refer to the appropriate manual supplied from the machine tool builder.

Format

G31 IP_ ;

G31 : One-shot G code (If is effective only in the block in which it is specified

16.4 MULTISTAGE SKIP (G31.1 TO G31.4)

The multistage skip function can be used for a block specifying G31.1 to G31.4. The function stores, in the custom macro variable, the coordinates when four normal skip signals or eight high-speed skip signals are turned on. Then, the function skips the remaining amount of movement. In a block specifying a dwell command (G04), the function skips the dwell when the skip signal is turned on.

In a block specifying G31.1 to G31.4, the multistage skip function is enabled. The correspondence between a G code and skip signal varies from one machine to another. So, refer to the manual provided by the machine tool builder.

A skip signal from equipment such as a fixed-dimension size measuring instrument can be used to skip programs being executed.

In plunge grinding, for example, a series of operations from rough machining to spark-out can be performed automatically by applying a skip signal each time rough machining, semi-fine machining, fine-machining, or spark-out operation is completed.

Format

```
Move command
  G31.1 (G31.2, G31.3, G31.4) IP_ F_ ;
     IP_  :  End point
     F_  :  Feedrate
Dwell
  G04 X(,P)_ ;
    X(,P)_  :  Dwell time
```

Explanation

In a block specifying G31.1 to G31.4, the multistage skip function is enabled. The correspondence between a G code and skip signal varies from one machine to another. Refer to the manual provided by the machine tool builder.

In a block that also specifies the dwell command (G04), the multistage skip function is enabled. The skip signal that can be used with the dwell command varies from one machine to another. So, refer to the manual provided by the machine tool builder.

In a multistage skip block, one of the following feedrates is used according to the setting of bit 5 (SFN) in parameter No. 7200 :

(a) Feedrate specified by F (which may be specified either before or after a multistage skip block)

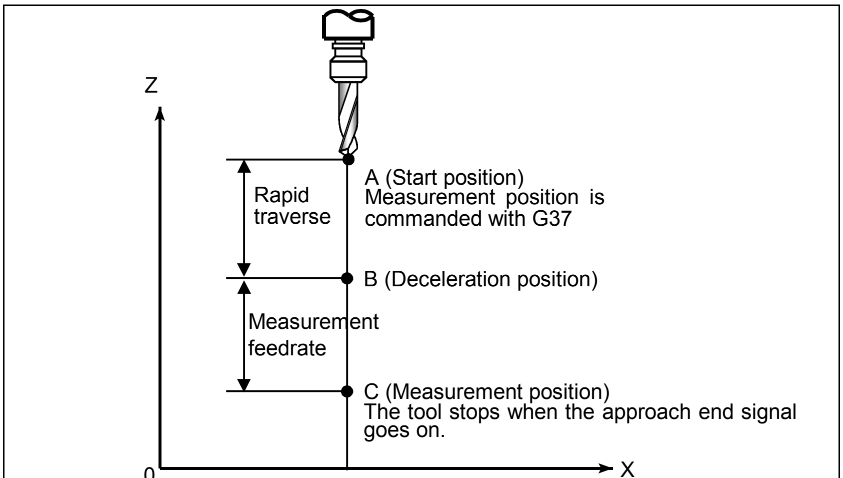
(b) Feedrate specified in one of parameter No. 7211 to No. 7214

- Correspondence to skip signals

Parameter Nos. 7205 to 7208 can be used to specify whether the 4-point or 8-point skip signal is used (when a high-speed skip signal is used). Specification is not limited to one-to-one correspondence. It is possible to specify that one skip signal correspond to two or more G codes. Also, of parameter No. 7209 can be used to specify dwell.

16.5 AUTOMATIC TOOL LENGTH MEASUREMENT (G37)

By issuing G37 the tool starts moving to the measurement position and keeps on moving till the approach end signal from the measurement device is output. Movement of the tool is stopped when the tool tip reaches the measurement position.
When the tool reaches the measurement position, the difference between the coordinate value of that position and the coordinate value specified by G37 is added to the current tool length compensation data.


Z
A (Start position)
Measurement position is commanded with G37
Rapid traverse
B (Deceleration position)
Measurement feedrate
C (Measurement position)
The tool stops when the approach end signal goes on.
0
X

Compensation data = (Current compensation data) + [(Measurement position reach signal detection position) - (Coordinate value specified by G37)]
when the MDC bit (bit 0 of parameter 7300) is set to 1

Fig.16.5 (a) Automatic tool length measurement

Format

G92 IP_ ; Sets the workpiece coordinate system. (It can be set with G54 to G59. See Chapter 7, "Coordinate System.")
Hxx ; Specifies an offset number for tool length offset.
G90 G37 IP_ ; Absolute command
G37 is valid only in the block in which it is specified.
IP_ indicates the X_, Y_, Z_ or fourth axis.

Explanations

- Setting the workpiece coordinate system

Set the workpiece coordinate system so that a measurement can be made after moving the tool to the measurement position. The coordinate system must be the same as the workpiece coordinate system for programming.

- Specifying G37

Specify the absolute coordinates of the correct measurement position. Execution of this command moves the tool at the rapid traverse rate toward the measurement position, reduces the federate halfway, then continuous to move it until the approach end signal from the measuring instrument is issued. When the tool tip reaches the measurement position, the measuring instrument sends an approach end signal to the CNC which stops the tool.

- Changing the offset value

When the MDC bit (bit 0 of parameter 7300) is set to 1 (default value), the difference between the coordinate value of the tool reaching the measurement position and the coordinate value specified by G37 is added to the current tool compensation data.
Compensation data = (Current compensation data) + [(Measurement position reach signal detection position) - (Coordinate value specified by G37)]
If the MDC bit (bit 0 of parameter 7300) is set to 0, subtraction can be performed here. The compensation data can be manually changed to a desired value by MDI.

- Alarm

When automatic tool length measurement is executed, the tool moves as shown in Fig. 14.2 (b). If the approach end signal goes on while the tool is traveling from point B to point C, an alarm occurs. Unless the approach end signal goes on before the tool reaches point F, the same alarm occurs. The P/S alarm number is 080.

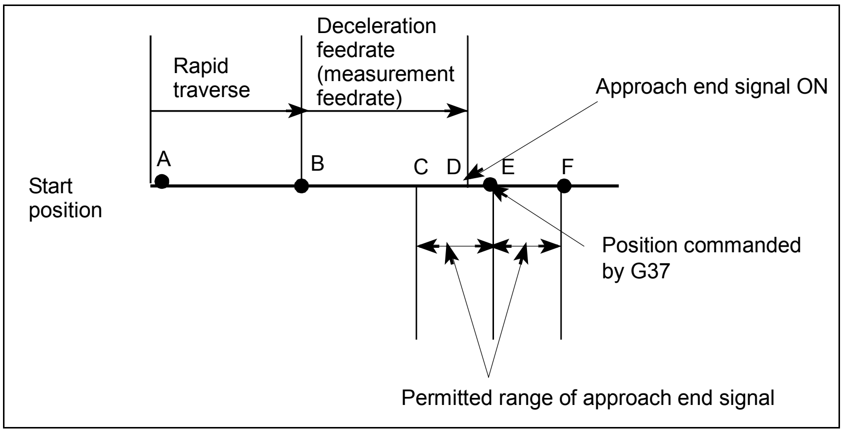

Rapid traverse
Deceleration feedrate (measurement feedrate)
Approach end signal ON
Start position
A
B
C
D
E
F
Position commanded by G37
Permitted range of approach end signal

Fig.16.5 (b) Tool movement to the measurement position

> ⚠ **WARNING**
> When a manual movement is inserted into a movement at a measurement federate, return the tool to the position before the inserted manual movement for restart.

NOTE

1 When an H code is specified in the same block as G37, an alarm is generated. Specify H code before the block!of G37.

2 The measurement speed (parameter No. 7311), deceleration position (parameter No. 7321), and permitted range of the approach end signal (parameter No. 7331) are specified by the machine tool builder.

3 Enter data that satisfies the following condition: deceleration position > allowable measurement position arrival signal range.

4 When offset memory A is used, the offset value is changed.
When offset memory B is used, the tool wear compensation value is changed.
When offset memory C is used, the tool wear compensation value for the H code is changed.

5 The approach end signal is monitored usually every 2 ms. When the high-speed measurement position arrival signal is used, monitoring is performed at intervals of 0.1 ms or less. The following measuring error is generated

$$ERR_{max} = F_m \times \frac{1}{60} \times \frac{T_S}{1000}$$

T_S :Sampling period, for usual 2(ms)
ERR_{max}:maximum measuring error (mm)
F_m :measurement federate (mm/min)

For example, when F_m=100mm/min Fm = 1000 mm/min., ERRmax= 0.003m

6 The tool stops a maximum of 16 ms after the approach end signal is detected. But the value of the position!at which the approach end signal was detected (note the value when the tool stopped) is used to determine the offset amount. The overrun for 16 ms is:

$$Q_{max} = F_m \times \frac{1}{60} \times \frac{16}{1000}$$

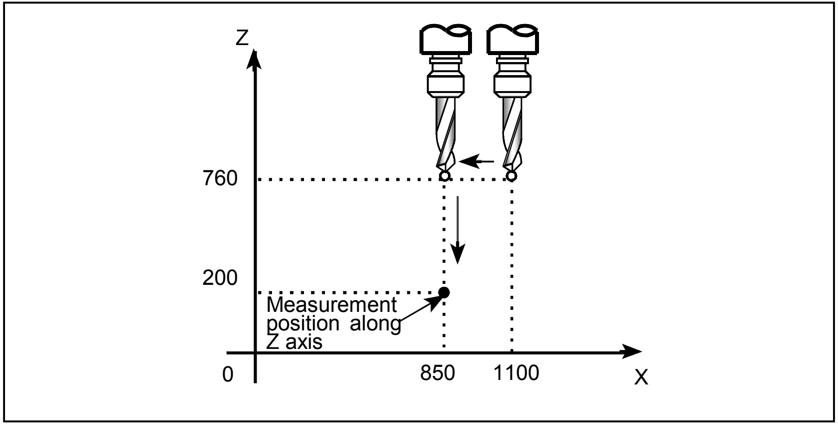
Q_{max} :maximum overrun (mm)
F_m :measurement federate (mm/min)

7 When using this function, set bit 0 (EV0) of parameter No. 6000 to 0. (If the offset is changed, the change becomes effective for all blocks starting with that specifying the next D code or H code (or T code for a lathe system).

Examples

G92 Z760.0 X1100.0 ;	Sets a workpiece coordinate system with respect
G00 G90 X850.0 ;	Moves the tool to X850.0. That is the tool is moved to a position that is a specified distance from the measurement position along the Z-axis.
H01 ;	Specifies offset number 1
G37 Z200.0 ;	Moves the tool to the measurement position
G00 Z204.0 ;	Retracts the tool a small distance along the Z-axis.

For example, if the tool reaches the measurement position with Z198.0;, the compensation value must be corrected. Because the correct measurement position is at a distance of 200 mm, the compensation value is lessened by 2.0 mm (198.0 - 200.0 = -2.0).



16.6 TORQUE LIMIT SKIP

If a move command is specified after G31 P99 (or G31 P98) when the servo motor torque limit(*1) is overridden, the same cutting feed as that achieved by linear interpolation (G01) is possible. If the servo motor torque reaches the torque limit (overridden servo motor torque limit) due to pressing, or if a skip signal (or high-speed skip signal) is entered during movement according to this move command, the remaining part of the move command is cancelled, and the next block is executed. (An operation that cancels the remaining part of the move command, and which executes the next block is referred to as a skip operation.) The following specification methods enable the servo motor torque limit to be overridden :

(1) Specify a torque limit override command in the PMC window.
(2) Specify address Q in a block in which G31 P99 (or G31 P98) is specified.

*1 : The servo motor torque limit corresponding to the setting of a motor type is automatically set.

Format

G31 P98 Q_ α_ F_
G31 P99 Q_ α _ F_

G31 : Skip command (one-shot G code)

P98 : A skip operation is performed when the servo motor torque reaches the torque limit.

P99 : A skip operation is performed when the torque of the servo motor reaches the torque limit or if a skip signal is entered.

Q :Torque limit override value
Specifiable range : 1 to 99 (%)
The Q command can be omitted. When omitting the Q command, however, specify a torque limit in the PMC window beforehand. When the Q command is omitted, a PS alarm (PS0151) is issued if a torque limit override is not set beforehand.
If a value that is not within the specifiable range is specified, a PS alarm (PS0150) is output.
A specified override value is valid only in the block in which the override value is specified. At the end of the skip operation, the override value returns to the value being used immediately before G31 was specified.

α :Axis address of an arbitrary axis

F :Feedrate

Explanation

- Skip operation condition

Condition	Command	
	G31P98	G31P99
When a torque limit is reached	A	A
When a skip signal is entered	B	A

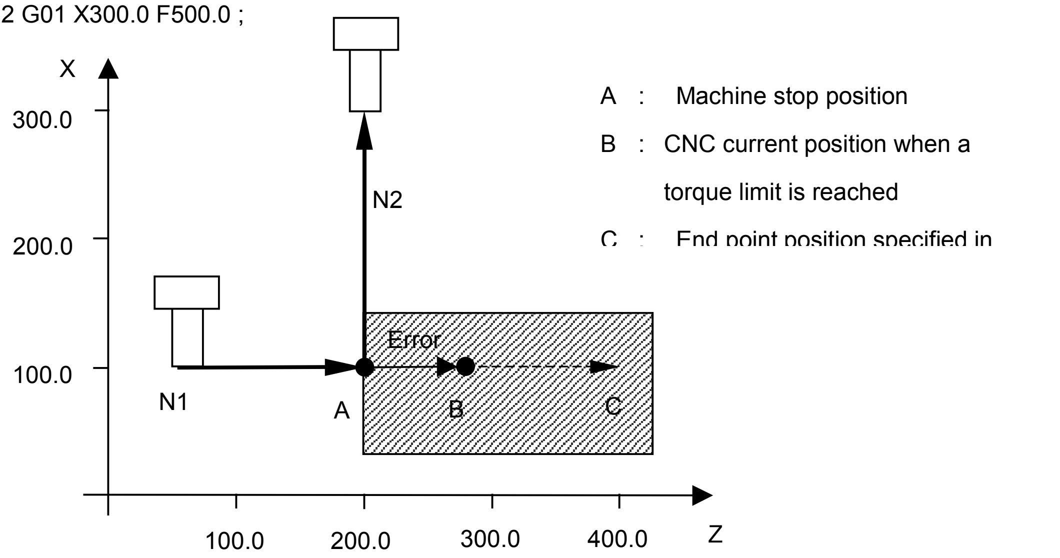
A : Skip operation is performed.
B : Skip offset is not performed.

- Torque limit skip operation

In torque limit skip operation, a specified axis is pressed against a workpiece when a torque limit has been specified with the servo motor. Then, skip operation is performed when the servo motor reaches the torque limit. Skip operation is performed when a torque limit detected by the servo motor is reached, so that it is not necessary to enter a skip signal by using a separate sensor as in the case of the normal skip function.

(Example)

```
N1 G31 P99 Q50 Z400.0 F100.0 ;
N2 G01 X300.0 F500.0 ;
```


X
300.0
200.0
100.0
N2
N1
Error
A
B
C
100.0
200.0
300.0
400.0
Z
A : Machine stop position
B : CNC current position when a torque limit is reached
C : End point position specified in

(1) At point A, the machine touches a measurement object, then stops. At this point, the torque limit is not reached, so that skip operation is not performed, but move command output continues, updating the current CNC position.

(2) Move command output continues, but the machine is stopped, so a displacement (error) occurs between the current CNC position and the machine position, and a torque is applied to the servo motor.

(3) When the torque limit is reached, skip operation is performed starting at the machine stop position (point A), and the N2 command is executed. Suppose that the current CNC position when the torque limit is reached is point B. Then, when torque limit skip is performed, the error is (A - B).

- Torque limit command

If a torque limit skip command specifies no torque limit override value in address Q, and no torque limit is specified using the PMC window, a PS alarm (PS151) is output.

No torque limit is specified when a torque limit override value of 0% or 100% is specified.

When a block containing a torque limit skip command specifies no torque limit override value in address Q, specify a torque limit as coded in the following program :

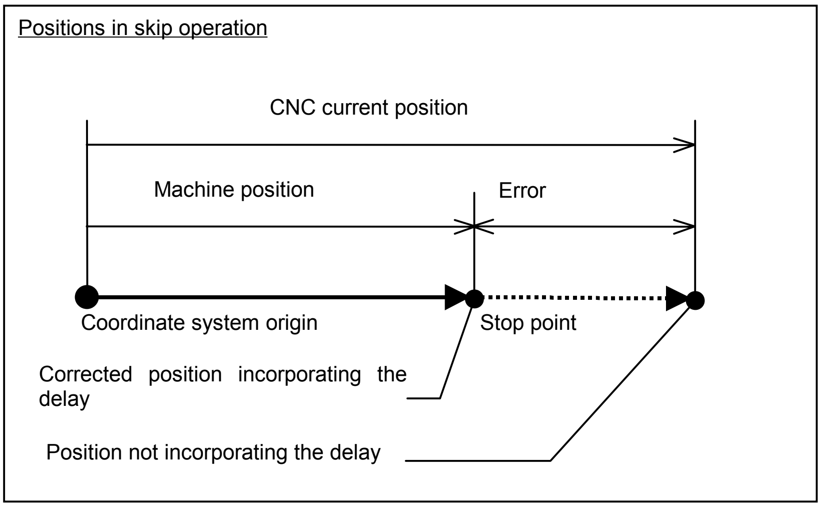
```
(Sample program)
O0012
  :
Mxx (Specify a torque limit in the PMC window.)
  :
 G31 P99 X200. F100. (Torque limit skip command)
  :
 G01 X100. F500. (Move command with the torque limit imposed)
  :
 Myy (The torque limit is cleared by the PMC)
  :
 M30
```

- Positional deviation limit when a torque limit is specified

During torque limit command execution, a positional deviation limit check based on the setting of parameter No. 1828 or No. 1829 is not made. Instead, a positional deviation limit check is made with the setting of parameter No. 1843. When a positional deviation limit is exceeded, an SV alarm (SV0109) is output to stop the machine immediately.

- Custom macro variables

When a torque limit skip command is issued, the custom macro variables (#5061 to #5080 : Skip signal positions; #100151 to #100174 are used for a system with 20 axes or more) hold the coordinates at the time of skip termination. When skip operation is actually performed, a servo system delay arises between the machine position and current CNC position. This error can be determined from the servo positional deviation value. By setting bit 1 (TSE) of parameter No. 7203, the user can choose whether the skip signal position data held in the system variables represents a position that either incorporates or does not incorporate a servo system error (positional deviation value).


Positions in skip operation
CNC current position
Machine position
Error
Coordinate system origin
Stop point
Corrected position incorporating the delay
Position not incorporating the delay

NOTE

1 Specify a torque limit skip command for one axis only. If no axis is specified, or if multiple axes are specified, a PS alarm (PS0150) is output.

2 Never specify a torque limit skip command in three-dimensional coordinate conversion mode or for parallel axes. Otherwise, a PS alarm (PS0151) is output.

3 Never specify a torque limit skip command in G41 or G42 mode. Otherwise, a PS alarm (PS0151) is output.

4 The torque limit reach signal is output, regardless of whether a torque limit skip command is issued.

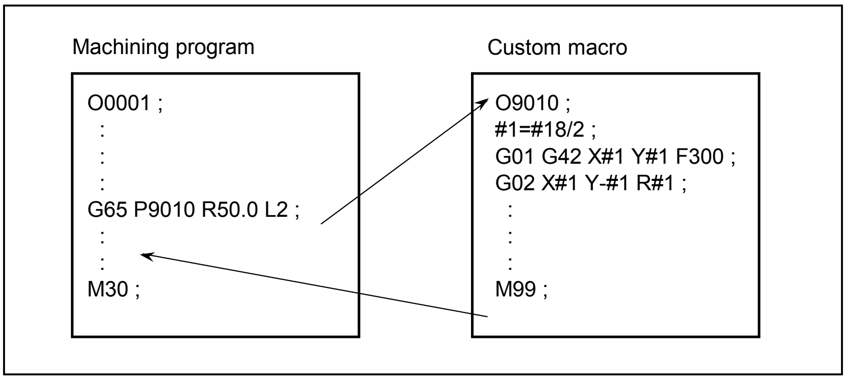
5 Never specify a torque limit skip command for those axes synchronized by synchronization control, twin table control, or an electronic gear box.

6 Never specify a torque limit skip command in successive blocks.

7 As the feedrate is increased, a greater error arises between the position at which the machine stops and the position at which a skip operation is actually detected. Moreover, a greater error arises when the feedrate is changed during movement. So, ensure that the feedrate is not changed by the application of an override.

17 CUSTOM MACRO

Although subprograms are useful for repeating the same operation, the custom macro function also allows use of variables, arithmetic and logic operations, and conditional branches for easy development of general programs such as pocketing and user-defined canned cycles.A machining program can call a custom macro with a simple command, just like a subprogram.



```
O0001 ;
  :
  :
  :
G65 P9010 R50.0 L2 ;
  :
  :
M30 ;
```

```
O9010 ;
#1=#18/2 ;
G01 G42 X#1 Y#1 F300 ;
G02 X#1 Y-#1 R#1 ;
  :
  :
  :
M99 ;
```

17.1 VARIABLES

An ordinary machining program specifies a G code and the travel distance directly with a numeric value; examples are G100 and X100.0. With a custom macro, numeric values can be specified directly or using a variable number. When a variable number is used, the variable value can be changed by a program or using operations on the MDI panel.

```
#1=#2+100 ;
G01 X#1 F300 ;
```

Explanation

- Variable representation

Specify a variable number after a sharp (#).

```
  #i(i=1,2,3,4,.....)
[Example]   #5
            #109
            #1005
```

A variable may also be represented by an <expression>, described below in "Operation command," as shown below :

```
  #[<Expression>]
[Example]   #[#100]
            #[#1001-1]
            #[#6/2]
```

Variable #i in the subsequent description can be replaced by variable #[<expression>].

- Types of variables

Variables are classified, according to their variable number, as local variables, common variables, or system variables. Each group of variables has different applications and attributes. In addition, read-only system constants are supported.

- Local variables #1 to #33

The local variable is a variable locally used in the macro. That is, a local variable #i used in the macro and called at one point in time, is different from #i used in the macro (whether it is the same macro or not) called at another point in time. Accordingly, when macro B is called from macro A, as in a multiplex call, a local variable used in macro A is not destroyed by being used in macro B

A local variable is used for an argument transfer. For information on the correspondence to the argument address, refer to section 17.1 for Macro Call Command. A local variable without a transferred argument is vacant in its initial status and can be used freely. The local variables have the read/write attribute.

- Common variables #100 - #199, #500 - #999

Just as a local variable is used locally in the macro, a common variable is in common use throughout the main program, throughout each subprogram called from the main program, and throughout each macro. That is, #i used in a certain macro is the same as #i used in another macro. Accordingly, the calculated value of a common variable #i in a certain macro can be used in another macro. The common variables basically have the read/write attribute. However, the common variable of a variable number specified by a parameter (No. 7036 to 7039) can be protected (for read-only). The application of common variables is not specified by the system, but can be specified as necessary by the user. The user can choose any of the common variables described below.

(a) 600 common variables (usable only with the custom macro options)
Common variables 100 to 199 and 500 to 999 can be used.
When the power is turned off, common variables 100 to 199 are cleared, but common variables 500 to 999 are retained.

(b) 900 common variables (usable with the custom macro options and 900 common variable options)
Common variables #100 to #199, #200 to #499, and #500 to #999 can be used. The values of #100 to #199 are cleared when the power is turned off. The values of #200 to #499 and #500 to #999 are held even when the power is turned off.

- System Variables

The applications of the system variables are determined by the system. Each system variable has one of three attributes : read-only, write-only, and read/write.

- System Constants

System constants can be referenced in the same way as variables. System constants have the read-only attribute.

- Omission of the decimal point

When a variable value is defined in a program, the decimal point can be omitted.

[Example]
When #1=123;is defined,the actual value of variable #1 is 123.000.

- Referencing variables

A number following an address can be replaced with a variable. When <address>#i or <address>-#i is programmed, the value of the variable or the complement of the value of the variable is used as the specified value of the address.

[Example]F#33:When #33 = 1.5, this is equivalent to specifying F1.5.
Z-#18 : When #18 = 20.0, this is equivalent to specifying Z-20.0.
G#130 : When #130 = 3.0, this is equivalent to specifying G3.

Variables cannot be referenced by address/, : , O and N.

[Example] O#27 and N#1 cannot be programmed.

n (1 to 9) of optional block skip/n cannot be replaced with a variable.

No variable number can be specified directly using a variable.

[Example] When replacing 5 of #5 with #30, specify #[#30] instead of ##30.

Values exceeding the maximum allowable number for each address cannot be specified.

[Example] When #140 = 120, G#140 exceeds the maximum allowable number.

When a variable is used as address data, the value of the variable is automatically rounded to the significant number of decimal places for each address.

[Example] If G00 X#1; is executed when #1 = 12.3456 on a machine for which the increment system is 1/1000 mm (IS-B), the result is G00 X12.346;.

By using <expression> described later, the number after an address can be replaced with <expression>. When the following is programmed :

<Address>[<expression>] or <Address>-[<expression>]

the value of <expression> or the compliment of the value of <expression> is used as the value of the address. Note that a constant with no decimal point used in [ ] is assumed to have a decimal point at the end.

[Example] X[#24+#18*COS[#1]]
Z-[#18+#26]

- Undefined variable

When the value of a variable is not defined, such a variable is referred to as a "null" variable. Variable #0 is always a null variable. It cannot be written to, but it can be read.

(a) Quotation

When an undefined variable is quotated, the address itself is also ignored.

Original command	G90 X100 Y#1
Equivalent command when #1 = <null>	G90 X100
Equivalent command when #1 = 0	G90 X100 Y0

(b) Definition/replacement, addition-type, multiplication-type operations

If a local variable or common variable is replaced directly with <null>, the result becomes <null>. If a system variable is replaced directly with <null>, or the result of an operation in which <null> is used is replaced, it is treated in the same way as for a variable assigned with 0.

Original arithmetic expression (example of local variable)	#2=#1	#2=#1*5	#2=#1+#1
Replacement result (if #1=<null>)	<null>	0	0
Replacement result (if #1=0)	0	0	0

Original arithmetic expression (example of common variable)	#100=#1	#100=#1*5	#100=#1+#1
Replacement result (if #1=<null>)	<null>	0	0
Replacement result (if #1=0)	0	0	0

Original arithmetic expression (example of system variable)	#2001=#1	#2001=#1*5	#2001=#1+#1
Replacement result (if #1=<null>)	0	0	0
Replacement result (if #1=0)	0	0	0

(c) Conditional expressions
<vacant> differs from 0 only for EQ and NE

Conditional expression	#1 EQ #0	#1 NE 0	#1 GE #0	#1 GT 0	#1 LE #0	#1 LT 0
When #1 = <null>	**Valid** (True)	**Valid** (True)	**Valid** (True)	**Invalid** (False)	**Valid** (True)	**Invalid** (False)
When #1 = 0	**Invalid** (False)	**Invalid** (False)	**Valid** (True)	**Invalid** (False)	**Valid** (True)	**Invalid** (False)

- Naming of system variables (constants)

A system variable (constant) is specified using a variable number. A system variable (constant) can also be specified using a preassigned system variable (constant) name. A system variable (constant) name starts with an underscore (_), and consists of up to eight characters including uppercase letters, numeric characters, and underscores (_). For variables such as axis-dependent variables (such as coordinates) and variables (such as tool compensation values) that may involve many items of similar data, [n] (n : integer) can be specified as the suffix of each name. In this case, n can be specified for <expression> or in arithmetic expression format. When specifying a system variable name, use the format [# system variable name] indicated below :

[#_DATE]

[Example]

[#_DATE] = 19980817; : Assigns 1998.08.17 to #3011 Year/month/day).

[#_TIME] = 161705; : Assigns 16 : 17 : 05 to #3012 (hours : minutes : seconds).

#101=[#_ABSMT[1]]; : Reads #5021 (machine coordinate of the first axis), and assigns the read value to #101.

#102=[#_ABSKP[#500*2]]; : Reads #506x (skip position of the [#500*2]-th axis, and assigns the read value to #102.

If a non-integral value is specified as suffix n, the value of n is rounded off to the nearest integer to reference the variable value.

[Example]

[#_ABSIO[1.4999999]] : [#_ABSIO[1]], namely, #5001 is assumed.
[#_ABSIO[1.5000000]] : [#_ABSIO[2]], namely, #5002 is assumed.

NOTE

1 If an unregistered variable name is specified, a PS0098 alarm is issued.

2 If an invalid value (such as a negative value) is specified as suffix n, a PS0099 alarm is issued.

- Naming of common variables

By specifying a variable name set with the SETVN command described later, common variable read and write operations are enabled. When specifying a common variable name, use the format [# common variable name] for [#VAR500].

[Example]

X[#POS1] Y[#POS2]; : Position specification using variable names

[#POS1] = #100+#101; : Assignment statement execution using a variable name

#[100+[#ABS]] = 500; : Same as above (variable number specification)

#500 = [1000+[#POS2]*10]; : Reading of a variable by variable name specification

17.2 SYSTEM VARIABLES

System variables can be used to read and write internal CNC data such as tool compensation values and current position data. System variables are essential for automation and general-purpose program development.

System variables/constants

n represents a suffix.

R, W, and R/W are the attributes of variables. R means read-only, W means write-only, and R/W means that both read and write operations are possible.

- Interface signals

System variable number	System variable name	Attribute	Description
#1000 to #1031	[#_UI[n]]	R	Interface input signals (bit), UI000 to UI031 Note) The suffix n represents a bit position (0 to 31).
#1032 to #1035	[#_UIL[n]]	R	Interface input signals (long), UI000 to UI031/UI100 to UI131/UI200 to UI231/UI300 to UI331 Note) Suffix n ranges from 0 to 3 and is used as follows : 0 = UI000 to UI031, 1 = UI100 to UI131, 2 = UI200 to UI231, and 3 = UI300 to UI331.
#1100 to #1131	[#_UO[n]]	R/W	Interface output signals (bit), UO000 to UO031 Note) Suffix n represents a bit position (0 to 31).
#1132 to #1135	[#_UOL[n]]	R/W	nterface output signals (long), UO000 to UO031/UO100 to UO131/UO200 to UO231/UO300 to UO331 Note) Suffix n ranges from 0 to 3 and is used as follows : 0 = UO000 to UO031, 1 = UO100 to UO131, 2 = UO200 to UO231, and 3 = UO300 to UO331.

- Tool offset values

System variable number	System variable name	Attribute	Description
#2001 to #2200 #10001 to #10999	[#_OFS[n]]	R/W	Tool compensation values in compensation memory A Note) Suffix n represents a compensation number (1 to 200). These numbers can also be used when the number of compensation pairs is greater than 200. Note) Suffix n represents a compensation number (1 to 999).
#2001 to #2200 #10001 to #10999	[#_OFSG[n]]	R/W	Tool compensation values (geometric) in compensation memory B Note) Suffix n represents a compensation number (1 to 200). These numbers can also be used when the number of compensation pairs is greater than 200. Note) Suffix n represents a compensation number (1 to 999).

System variable number	System variable name	Attribute	Description
#2201 to #2400 #11001 to #11999	[#_OFSW[n]]	R/W	Tool compensation values (wear) in compensation memory B Note) Suffix n represents a compensation number (1 to 200). These numbers can also be used when the number of compensation pairs is greater than 200. Note) Suffix n represents a compensation number (1 to 999).
#2001 to #2200 #10001 to #10999	[#_OFSHG[n]]	R/W	Tool compensation values (H code, geometric) in compensation memory C Note) Suffix n represents a compensation number (1 to 200). Tool compensation values (H code, wear) in compensation memory C Note) Suffix n represents a compensation number (1 to 200).
#2201 to #2400 #11001 to #11999	[#_OFSHW[n]]	R/W	Tool compensation values (H code, wear) in compensation memory C Note) Suffix n represents a compensation number (1 to 200). Tool compensation values (H code, wear) in compensation memory C Note) Suffix n represents a compensation number (1 to 200).
#2401 to #2600 #12001 to #12999	[#_OFSDG[n]]	R/W	Tool compensation values (D code, geometric) in compensation memory C Note) Suffix n represents a compensation number (1 to 200). Tool compensation values (H code, wear) in compensation memory C Note) Suffix n represents a compensation number (1 to 200).
#2601 to #2800 #13001 to #13999	[#_OFSDW[n]]	R/W	Tool compensation values (D code, wear) of compensation memory Note) Suffix n represents a compensation number (1 to 200). Tool compensation values (H code, wear) in compensation memory C Note) Suffix n represents a compensation number (1 to 200).

- Automatic operation and so forth

System variable number	System variable name	Attribute	Description
#3000	[#_ALM]	W	Macro alarm
#3001	[#_CLOCK1]	R/W	Clock 1 (Units : milliseconds)
#3002	[#_CLOCK2]	R/W	Clock 2 (Units : hours)
#3003	[#_CNTL1]	R/W	Single block stop disabled/enabled. Auxiliary function completion signal awaited/not awaited.
#3003 bit0	[#_M_SBK]	R/W	Single block stop disabled/enabled

System variable number	System variable name	Attribute	Description
#3003 bit1	[#_M_FIN]	R/W	Auxiliary function completion signal awaited/not awaited
#3004	[#_CNTL2]	R/W	Feed hold enabled/disabled. Feedrate override enabled/disabled. Exact stop check enabled/disabled.
#3004 bit0	[#_M_FHD]	R/W	Feed hold enabled/disabled
#3004 bit1	[#_M_OV]	R/W	Feedrate override enabled/disabled
#3004 bit2	[#_M_EST]	R/W	Exact stop check enabled/disabled
#3006	[#_MSGSTP]	W	Operation stopped with a message
#3007	[#_MRIMG]	R	Mirror image state (DI and setting)
#3008	[#_PRSTR]	R	Program not being restarted/program being restarted

- Time information

System variable number	System variable name	Attribute	Description
#3011	[#_DATE]	R	Year/month/day
#3012	[#_TIME]	R	Hours/minutes/seconds

- Cutting time

System variable No.	System variable name	Attrib-ute	Description
#3016	[#_CUTTM]	R/W	Reads and presets the cumulative cutting time parameters (No. 103, No. 104).

- Number of parts

System variable number	System variable name	Attribute	Description
#3901	[#_PRTSA]	R	Cumulative number of machined parts
#3902	[#_PRTSN]	R	Number of required parts

- Main program number

System variable number	System variable name	Attribute	Description
#4000	[#_MAINO]	R	Main program number

- Modal information

System variable number	System variable name	Attribute	Description
#4001 to #4030	[#_BUFG[n]]	R	Modal information of blocks up to the immediately preceding block (G code) Note) Suffix n represents a G code group number.
#4102	[#_BUFB]	R	Modal information of blocks up to the immediately preceding block (B code)
#4107	[#_BUFD]	R	Modal information of blocks up to the immediately preceding block (D code)
#4108	[#_BUFE]	R	Modal information of blocks up to the immediately preceding block (E code)
#4109	[#_BUFF]	R	Modal information of blocks up to the immediately preceding block (F code)
#4111	[#_BUFH]	R	Modal information of blocks up to the immediately preceding block (H code)
#4113	[#_BUFM]	R	Modal information of blocks up to the immediately preceding block (M code)
#4114	[#_BUFN]	R	Modal information of blocks up to the immediately preceding block (sequence number)
#4115	[#_BUFO]	R	Modal information of blocks up to the immediately preceding block (program number)
#4119	[#_BUFS]	R	Modal information of blocks up to the immediately preceding block (S code)
#4120	[#_BUFT]	R	Modal information of blocks up to the immediately preceding block (T code)
#4130	[#_BUFWZP]	R	Modal information of blocks up to the immediately preceding block (additional workpiece coordinate system number)
#4201 to #4230	[#_ACTG[n]]	R	Modal information of the block currently being executed (G code) Note) Suffix n represents a G code group number.
#4302	[#_ACTB]	R	Modal information of the block currently being executed (B code)
#4307	[#_ACTD]	R	Modal information of the block currently being executed (D code)
#4308	[#_ACTE]	R	Modal information of the block currently being executed (E code)
#4309	[#_ACTF]	R	Modal information of the block currently being executed (F code))
#4311	[#_ACTH]	R	Modal information of the block currently being executed (H code)
#4313	[#_ACTM]	R	Modal information of the block currently being executed (M code)
#4314	[#_ACTN]	R	Modal information of the block currently being executed (sequence number)
#4315	[#_ACTO]	R	Modal information of the block currently being executed (program number)
#4319	[#_ACTS]	R	Modal information of the block currently being executed (S code)
#4320	[#_ACTT]	R	Modal information of the block currently being executed (T code)

System variable number	System variable name	Attribute	Description
#4330	[#_ACTWZP]	R	Modal information of the block currently being executed (additional workpiece coordinate system number)
#4401 to #4430	[#_INTG[n]]	R	Modal information of an interrupted block (G code) Note) Suffix n represents a G code group number.
#4502	[#_INTB]	R	Modal information of an interrupted block (B code)
#4507	[#_INTD]	R	Modal information of an interrupted block (D code)
#4508	[#_INTE]	R	Modal information of an interrupted block (E code)
#4509	[#_INTF]	R	Modal information of an interrupted block (F code)
#4511	[#_INTH]	R	Modal information of an interrupted block (H code)
#4513	[#_INTM]	R	Modal information of an interrupted block (M code)
#4514	[#_INTN]	R	Modal information of an interrupted block (sequence number)
#4515	[#_INTO]	R	Modal information of an interrupted block (program number)
#4519	[#_INTS]	R	Modal information of an interrupted block (S code)
#4520	[#_INTT]	R	Modal information of an interrupted block (T code)
#4530	[#_INTWZP]	R	Modal information of an interrupted block (additional workpiece coordinate system number)

- Position information

System variable number	System variable name	Attribute	Description
#5001 to #5020	[#_ABSIO[n]]	R	End point of the immediately preceding block (workpiece coordinate system) Note) Suffix n represents an axis number (1 to 20)
#5021 to #5040	[#_ABSMT[n]]	R	Specified current position (machine coordinate system) Note) The suffix represents an axis number (1 to 20).
#5041 to #5060	[#_ABSOT[n]]	R	Specified current position (workpiece coordinate system) Note) The suffix represents an axis number (1 to 20).
#5061 to #5080	[#_ABSKP[n]]	R	Skip signal position (workpiece coordinate system) Note) Suffix n represents an axis number (1 to 20).

- Tool length compensation values

System variable number	System variable name	Attribute	Description
#5081 to #5100	[#_TOFS[n]]	R	Tool length compensation value Note) Suffix n represents an axis number (1 to 20).

- Servo positional deviation values

System variable number	System variable name	Attribute	Description
#5101 to #5120	[#_SVERR[n]]	R	Servo positional deviation value Note) Suffix n represents an axis number (1 to 20).

- Manual handle interrupt values

System variable number	System variable name	Attribute	Description
#5121 to #5140	[#_MIRTP[n]]	R	Manual handle interrupt value Note) Suffix n represents an axis number (1 to 20).

- Workpiece origin offsets

System variable number	System variable name	Attribute	Description
#5201 to #5220	[#_WZCMN[n]]	R/W	Common workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#5221 to #5240	[#_WZG54[n]]	R/W	G54 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#5241 to #5260	[#_WZG55[n]]	R/W	G55 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#5261 to #5280	[#_WZG56[n]]	R/W	G56 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#5281 to #5300	[#_WZG57[n]]	R/W	G57 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#5301 to #5320	[#_WZG58[n]]	R/W	G58 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#5321 to #5340	[#_WZG59[n]]	R/W	G59 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#7001 to #7020	[#_WZP1[n]]	R/W	G54.1P1 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
#7021 to #7040	[#_WZP2[n]]	R/W	G54.1P2 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).
: :	: :	: :	: :
#7941 to #7960	[#_WZP48[n]]	R/W	G54.1P48 workpiece origin offset Note) Suffix n represents an axis number (1 to 20).

- Reference fixture offset values

System variable number	System variable name	Attribute	Description
#15001 to #15020	[#_FOFS1[n]]	R/W	Reference fixture offset value (first pair) Note) Suffix n represents an axis number (1 to 20).
#15021 to #15040	[#_FOFS2[n]]	R/W	Reference fixture offset value (second pair) Note) Suffix n represents an axis number (1 to 20).
#15041 to #15060	[#_FOFS3[n]]	R/W	Reference fixture offset value (third pair) Note) Suffix n represents an axis number (1 to 20).
#15061 to #15080	[#_FOFS4[n]]	R/W	Reference fixture offset value (fourth pair) Note) Suffix n represents an axis number (1 to 20).
#15081 to #15100	[#_FOFS5[n]]	R/W	Reference fixture offset value (fifth pair) Note) Suffix n represents an axis number (1 to 20).
#15101 to #15120	[#_FOFS6[n]]	R/W	Reference fixture offset value (sixth pair) Note) Suffix n represents an axis number (1 to 20).
#15121 to #15140	[#_FOFS7[n]]	R/W	Reference fixture offset value (seventh pair) Note) Suffix n represents an axis number (1 to 20).
#15141 to #15160	[#_FOFS8[n]]	R/W	Reference fixture offset value (eighth pair) Note) Suffix n represents an axis number (1 to 20).

- Dynamic reference tool compensation values

System variable number	System variable name	Attribute	Description
#16001 to #16020	[#_DOFS1[n]]	R/W	Dynamic reference tool compensation value (first pair) Note) Suffix n represents an axis number (1 to 20).
#16021 to #16040	[#_DOFS2[n]]	R/W	Dynamic reference tool compensation value (second pair) Note) Suffix n represents an axis number (1 to 20).
#16041 to #16060	[#_DOFS3[n]]	R/W	Dynamic reference tool compensation value (third pair) Note) Suffix n represents an axis number (1 to 20).
#16061 to #16080	[#_DOFS4[n]]	R/W	Dynamic reference tool compensation value (fourth pair) Note) Suffix n represents an axis number (1 to 20).
#16081 to #16100	[#_DOFS5[n]]	R/W	Dynamic reference tool compensation value (fifth pair) Note) Suffix n represents an axis number (1 to 20).
#16101 to #16120	[#_DOFS6[n]]	R/W	Dynamic reference tool compensation value (sixth pair) Note) Suffix n represents an axis number (1 to 20).
#16121 to #16140	[#_DOFS7[n]]	R/W	Dynamic reference tool compensation value (seventh pair) Note) Suffix n represents an axis number (1 to 20).
#16141 to #16160	[#_DOFS8[n]]	R/W	Dynamic reference tool compensation value (eighth pair) Note) Suffix n represents an axis number (1 to 20).

- System constants

System constant number	System constant name	Attribute	Description
#0 , #3100	[#_EMPTY]	R	Null value
#3101	[#_PI]	R	Circle ratio π = 3.14159265358979323846
#3102	[#_E]	R	Base of natural logarithm e = 2.71828182845904523536

Explanations

R, W, and R/W are the attributes of variables. R means read-only, W means write-only, and R/W means that both read and write operations are possible.

- Interface signals #1000 to #1031, #1032, #1033 to #1035 (Attribute : R) #1100 to #1115, #1132, #1133 to #1135 (Attribute : R/W)

[Input signals]

By reading the system variables, #1000 to #1032, for reading interface signals, the states of the interface input signals can be checked.

Variable number	Variable name	Number of points	Interface input signal
#1000	[#_UI[0]]	1	UI000 (2^0)
#1001	[#_UI[1]]	1	UI001 (2^1)
#1002	[#_UI[2]]	1	UI002 (2^2)
#1003	[#_UI[3]]	1	UI003 (2^3)
#1004	[#_UI[4]]	1	UI004 (2^4)
#1005	[#_UI[5]]	1	UI005 (2^5)
#1006	[#_UI[6]]	1	UI006 (2^6)
#1007	[#_UI[7]]	1	UI007 (2^7)
#1008	[#_UI[8]]	1	UI008 (2^8)
#1009	[#_UI[9]]	1	UI009 (2^9)
#1010	[#_UI[10]]	1	UI010 (2^{10})
#1011	[#_UI[11]]	1	UI011 (2^{11})
#1012	[#_UI[12]]	1	UI012 (2^{12})
#1013	[#_UI[13]]	1	UI013 (2^{13})
#1014	[#_UI[14]]	1	UI014 (2^{14})
#1015	[#_UI[15]]	1	UI015 (2^{15})
#1016	[#_UI[16]]	1	UI016 (2^{16})
#1017	[#_UI[17]]	1	UI017 (2^{17})
#1018	[#_UI[18]]	1	UI018 (2^{18})
#1019	[#_UI[19]]	1	UI019 (2^{19})
#1020	[#_UI[20]]	1	UI020 (2^{20})
#1021	[#_UI[21]]	1	UI021 (2^{21})
#1022	[#_UI[22]]	1	UI022 (2^{22})
#1023	[#_UI[23]]	1	UI023 (2^{23})
#1024	[#_UI[24]]	1	UI024 (2^{24})
#1025	[#_UI[25]]	1	UI025 (2^{25})
#1026	[#_UI[26]]	1	UI026 (2^{26})
#1027	[#_UI[27]]	1	UI027 (2^{27})
#1028	[#_UI[28]]	1	UI028 (2^{28})
#1029	[#_UI[29]]	1	UI029 (2^{29})
#1030	[#_UI[30]]	1	UI030 (2^{30})
#1031	[#_UI[31]]	1	UI031 (2^{31})
#1032	[#_UIL[0]]	32	UI000 to UI031
#1033	[#_UIL[1]]	32	UI100 to UI131
#1034	[#_UIL[2]]	32	UI200 to UI231
#1035	[#_UIL[3]]	32	UI300 to UI331

Variable value	Input signal
1.0	Contact closed
0.0	Contact open

A read variable value is 1.0 or 0.0, regardless of the increment system, so that the increment system must be considered when macros are created. By reading a system variable from #1032 to #1035, 32 input signals can be read at a time.

$$\#1032 = \sum_{i=0}^{30} \#[1000 + i] \times 2^{i} - \#1031 \times 2^{31}$$

$$\#[1032 + n] = \sum_{i=0}^{30} \left\{ 2^{i} \times V_{i} \right\} - 2^{31} \times V_{31}$$

where, Vi = 0 when UIni is 0
Vi = 1 when UIni is 1
n : 0 to 3

[Output signals]
By assigning values to system variables #1100 to #1132, for outputting interface signals, interface output signals can be output.

Variable number	Variable name	Number of points	Interface input signal
#1100	[#_UO[0]]	1	UO000 (2^0)
#1101	[#_UO[1]]	1	UO001 (2^1)
#1102	[#_UO[2]]	1	UO002 (2^2)
#1103	[#_UO[3]]	1	UO003 (2^3)
#1104	[#_UO[4]]	1	UO004 (2^4)
#1105	[#_UO[5]]	1	UO005 (2^5)
#1106	[#_UO[6]]	1	UO006 (2^6)
#1107	[#_UO[7]]	1	UO007 (2^7)
#1108	[#_UO[8]]	1	UO008 (2^8)
#1109	[#_UO[9]]	1	UO009 (2^9)
#1110	[#_UO[10]]	1	UO010 (2^{10})
#1111	[#_UO[11]]	1	UO011 (2^{11})
#1112	[#_UO[12]]	1	UO012 (2^{12})
#1113	[#_UO[13]]	1	UO013 (2^{13})
#1114	[#_UO[14]]	1	UO014 (2^{14})
#1115	[#_UO[15]]	1	UO015 (2^{15})
#1116	[#_UO[16]]	1	UO016 (2^{16})
#1117	[#_UO[17]]	1	UO017 (2^{17})
#1118	[#_UO[18]]	1	UO018 (2^{18})
#1119	[#_UO[19]]	1	UO019 (2^{19})
#1120	[#_UO[20]]	1	UO020 (2^{20})
#1121	[#_UO[21]]	1	UO021 (2^{21})
#1122	[#_UO[22]]	1	UO022 (2^{22})
#1123	[#_UO[23]]	1	UO023 (2^{23})
#1124	[#_UO[24]]	1	UO024 (2^{24})
#1125	[#_UO[25]]	1	UO025 (2^{25})
#1126	[#_UO[26]]	1	UO026 (2^{26})
#1127	[#_UO[27]]	1	UO027 (2^{27})
#1128	[#_UO[28]]	1	UO028 (2^{28})
#1129	[#_UO[29]]	1	UO029 (2^{29})
#1130	[#_UO[30]]	1	UO030 (2^{30})
#1131	[#_UO[31]]	1	UO031 (2^{31})

Variable number	Variable name	Number of points	Interface input signal
#1132	[#_UOL[0]]	32	UO000 to UO031
#1133	[#_UOL[1]]	32	UO100 to UO131
#1134	[#_UOL[2]]	32	UO200 to UO231
#1135	[#_UOL[3]]	32	UO300 to UO331

Variable value	Input signal
1.0	Contact closed
0.0	Contact open

By writing to a system variable from #1132 to #1135, 32 output signals can be written at a time. Similarly, 32 signals can be read at a time.

$$\#1132 = \sum_{i=0}^{30} \#[1100 + i] \times 2^{i} - \#1131 \times 2^{31}$$

$$\#[1132 + n] = \sum_{i=0}^{30} \left\{ 2^{i} \times V_{i} \right\} - 2^{31} \times V_{31}$$

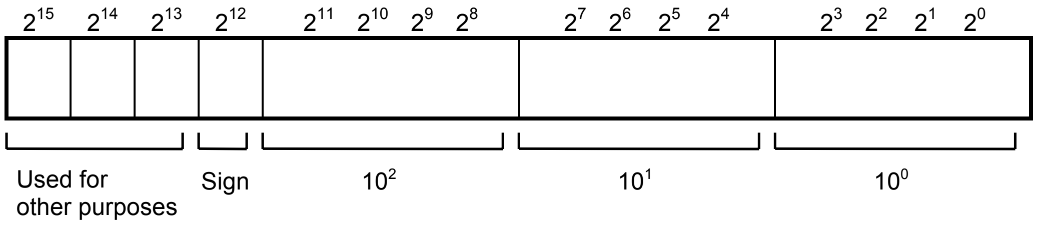
where, Vi = 0 when UIni is 0
Vi = 1 when UIni is 1
n : 0 to 3

NOTE

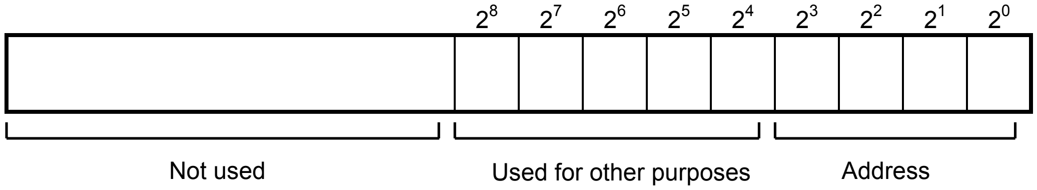
If a value other than 1.0 or 0.0 is assigned to #1100 to #1115, the following is assumed:
<Null> is assumed to be 0.
A value other than <null> is assumed to be 1.
However, a value of less than 0.00000001 is handled as an undefined value.

[Example]

DI configuration



DO configuration



(1) Signed three BCD digits based on address switching are read.

Macro call command

```
G65 P9100 D(address);
```

A custom macro body is generated as follows :

```
O9100 ;
#1132= #1132 AND 496 OR#7  ;      : Outputs address.
G65 P9101 T60 ;                   : Timer macro
#100 = BIN[#1032 AND 4095] ;      : Reads three BCD digits.
IF [#1012 EQ  0] GOTO  9100 ;     : Assigns a sign.
#100 = -#100
N9100 M99
```

(2) Eight types of signed six BCD digits (three digits in integer part + three digits in fractional part) based on address switching are read into #101.

Configuration on the machine

When DO 20 = 0 : Data of 3 digits in the fractional part
When DO 20 = 1 : Data of 3 digits in the integer part
When DO 23 to 21 = 000 : No. 1 data
When DO 23 to 21 = 001 : No. 2 data

:

= 111 : No. 8 data

Macro call command

```
G65 P9101 D(data number);
```

A custom macro body is generated as follows :

```
O9101 ;
G65 P9101 D[#7 * 2 + 1] ;
#101 = #100  ;
G65 P9100 D[#7 * 2] ;
#101 = #101 + #100/1000 ;
M99 ;
```

- Tool compensation values #2000 to #2800, #10001 to #13999 (Attribute : R/W)

Compensation values can be checked by reading system variables #2001 to #2800 and #10001 to #13999. Compensation values can be changed by assigning desired values to the system variables.

(1) Tool compensation memory A

- When the number of compensation values does not exceed 200

Compensation number	Variable number	Variable name
1	#2001	[#_OFS[1]]
2	#2002	[#_OFS[2]]
3	#2003	[#_OFS[3]]
:	:	:
199	#2199	[#_OFS[199]]
200	#2200	[#_OFS[200]]

- When the number of compensation values exceeds 200 (The values of the compensation numbers up to 200 can also be referenced using #2001 to #2200.)

Compensation number	Variable number	Variable name
1	#10001	[#_OFS[1]]
2	#10002	[#_OFS[2]]
3	#10003	[#_OFS[3]]
:	:	:
998	#10998	[#_OFS[998]]
999	#10999	[#_OFS[999]]

(2) Tool compensation memory B

- When the number of compensation values does not exceed 200

Compensation number	Geometric		Wear	
	Variable number	Variable name	Variable number	Variable name
1	#2001	[#_OFSG[1]]	#2201	[#_OFSG[1]]
2	#2002	[#_OFSG[2]]	#2202	[#_OFSG[2]]
3	#2003	[#_OFSG[3]]	#2203	[#_OFSG[3]]
:	:	:	:	:
199	#2199	[#_OFSG[199]]	#2399	[#_OFSG[199]]
200	#2200	[#_OFSG[200]]	#2400	[#_OFSG[200]]

- When the number of compensation values exceeds 200 (The values of the compensation numbers up to 200 can also be referenced using #2001 to #2400.)

Compensation number	Geometric		Wear	
	Variable number	Variable name	Variable number	Variable name
1	#10001	[#_OFSG[1]]	#11001	[#_OFSW[1]]
2	#10002	[#_OFSG[2]]	#11002	[#_OFSW[2]]
3	#10003	[#_OFSG[3]]	#11003	[#_OFSW[3]]
:	:	:	:	:
998	#10998	[#_OFSG[998]]	#11998	[#_OFSW[998]]
999	#10999	[#_OFSG[999]]	#11999	[#_OFSW[999]]

(3) Tool compensation memory C

- When the number of compensation values does not exceed 200

H code				
Compensation number	Geometric		Wear	
	Variable number	Variable name	Variable number	Variable name
1	#2001	[#_OFSHG[1]]	#2201	[#_OFSHW[1]]
2	#2002	[#_OFSHG[2]]	#2202	[#_OFSHW[2]]
3	#2003	[#_OFSHG[3]]	#2203	[#_OFSHW[3]]
:	:	:	:	:
199	#2199	[#_OFSHG[199]]	#2399	[#_OFSHW[199]]
200	#2200	[#_OFSHG[200]]	#2400	[#_OFSHW[200]]

D code				
Compensation number	Geometric		Wear	
	Variable number	Variable name	Variable number	Variable name
1	#2401	[#_OFSDG[1]]	#2601	[#_OFSDW[1]]
2	#2402	[#_OFSDG[2]]	#2602	[#_OFSDW[2]]
3	#2403	[#_OFSDG[3]]	#2603	[#_OFSDW[3]]
:	:	:	:	:
199	#2599	[#_OFSDG[199]]	#2799	[#_OFSDW[199]]
200	#2600	[#_OFSDG[200]]	#2800	[#_OFSDW[200]]

- When the number of compensation values exceeds 200 (The values of the compensation numbers up to 200 can also be referenced using #2001 to #2800.)

H code				
Compensation number	Geometric		Wear	
	Variable number	Variable name	Variable number	Variable name
1	#10001	[#_OFSHG[1]]	#110001	[#_OFSHW[1]]
2	#10002	[#_OFSHG[2]]	#11002	[#_OFSHW[2]]
3	#10003	[#_OFSHG[3]]	#11003	[#_OFSHW[3]]
:	:	:	:	:
998	#10998	[#_OFSHG[998]]	#11998	[#_OFSHW[998]]
999	#10999	[#_OFSHG[999]]	#11999	[#_OFSHW[999]]

D code				
Compensation number	Geometric		Wear	
	Variable number	Variable name	Variable number	Variable name
1	#12001	[#_OFSDG[1]]	#13001	[#_OFSDW[1]]
2	#12002	[#_OFSDG[2]]	#13002	[#_OFSDW[2]]
3	#12003	[#_OFSDG[3]]	#13003	[#_OFSDW[3]]
:	:	:	:	:
998	#12998	[#_OFSDG[998]]	#13998	[#_OFSDW[998]]
999	#12999	[#_OFSDG[999]]	#13999	[#_OFSDW[999]]

- Alarm #3000 (Attribute : R/W)

When an error is detected in a macro, the machine can be placed in the alarm state. By assigning an alarm number to system variable #3000, the alarm lamp is turned on, and the machine is placed in the macro alarm state (MCxxxx) as soon as processing of the previous block ends.

Variable number	Variable name	Description
#3000	[#_ALM]	Macro alarm

[Example] #3000 = n (ALARM MESSAGE) ; (n ≤ 4095)

An alarm message not longer than 26 characters, enclosed in a control-out code and control-in code, can be programmed. When n = 123, the alarm screen displays the following :

MC0123 ALARM MESSAGE

- Clocks #3001, #3002 (Attribute : R/W)

By reading the system variables for clocks #3001 and #3002, the times of the clocks can be checked. The time of a clock can be preset by assigning a desired value to the system variable.

Type	Variable number	Variable name	Units	Upon power-up	Count condition
Clock 1	#3001	[#_CLOCK1]	1msec	Reset to 0	At all times
Clock 2	#3002	[#_CLOCK2]	Hours	Same as power-off	When the STL signal is turned on

The clock precision is 16 msec. Clock 1 returns to 0 once 2147483648 msec have elapsed. Clock 2 returns to 0 once 9544.37176 hours have elapsed.

[Example] Timer

```
Macro call command
G65 P9101 T (wait time) msec;
A macro is to be generated as follows :
O9101 ;
#3001 = 0 : Initialization
WHILE [#3001 LE #20] DO1 : Waits a specified period of
                           time.
END1 ;
M99 ;
```

- Control upon single block stop and waiting for the auxiliary function completion signal #3003 (Attribute : R/W)

By assigning a value indicated in the table below to system variable #3003, single block stop can be disabled in the subsequent blocks, or the next block can be executed without waiting for the completion signal (FIN) of an auxiliary function (M, S, T, or B). When the completion signal is not awaited, the distribution completion signal (DEN) is not output. Be careful not to specify an additional auxiliary function without first waiting for the completion signal.

Variable number or variable name	Value	Single block stop	Completion of an auxiliary function
#3003 [#_CNTL1]	0	Enabled	To be awaited
	1	Disabled	To be awaited
	2	Enabled	Not to be awaited
	3	Disabled	Not to be awaited

By using the variable names listed below, single block stop and waiting for the auxiliary function completion signal can be controlled separately.

Variable number or variable name	Value	Single block stop	Completion of an auxiliary function
[#_M_SBK]	0	Enabled	_
	1	Disabled	_
[#_M_FIN]	0	_	To be awaited
	1	_	Not to be awaited

[Example]

Drilling cycle (incremental programming)
(Equivalent to G81)
Macro call command
G65 P9081 L number-of-repeats R point-R Z point-Z;
A custom macro body is generated as follows :

```
O9081 ;
#3003 = 1 ;
G00 Z#18 ;
G01 Z#23 ;
G00 Z-[ ROUND[#18] + ROUND[#26] ] ;
 #3003 = 0  :
 M99 ;
```

Single block stop is disabled.
#18 corresponds to R, and #26 corresponds to Z.

NOTE
#3003 is cleared upon a reset.

- Disabling feed hold, feedrate override, and exact stop check #3004 (Attribute:R/W)

By assigning any of the values listed in the table below to system variable #3004, feed hold or feedrate override can be disabled in a subsequent block. An exact stop in G61 mode or by G09 can also be disabled.

Variable number Variable name	Value	Feed hold	Feedrate override	Exact stop
#3004 [#_CNTL2]	0	Enabled	Enabled	Enabled
	1	Disabled	Enabled	Enabled
	2	Enabled	Disabled	Enabled
	3	Disabled	Disabled	Enabled
	4	Enabled	Enabled	Disabled
	5	Disabled	Enabled	Disabled
	6	Enabled	Disabled	Disabled
	7	Disabled	Disabled	Disabled

By using the following variable names, feed hold, feedrate override, and exact stop in G61 mode or by G09 can be individually enabled or disabled.

Variable number Variable name	Value	Feed hold	Feedrate override	Exact stop
[#_M_FHD]	0	Enabled	_	_
	1	Disabled	_	_
[#_M_OV]	0	_	Enabled	_
	1	_	Disabled	_
[#_M_EST]	0	_	_	Enabled
	1	_	_	Disabled

NOTE

1 This system variable is provided to ensure compatibility with conventional NC programs. It is recommended that functions based on G codes such as G63, G09, and G61 be used to control the feed hold, feedrate override, and exact stop check functions.

2 If the feed hold button is pressed while a block disabling the feed hold function is being executed, the following results :

1) If the feed hold button is held down, single block stop occurs. No single block stop occurs, however, if the single block stop function is disabled.

2) If the feed hold button is released after being pressed, the feed hold lamp is turned on. However, the tool does not stop immediately, but stops at the end point of the next block that enables the feed hold function.

3 #3004 is cleared upon a reset.

4 Variable #3004, set to disable an exact stop, does not affect the original position of exact stop between cutting feed and positioning block. Variable #3004 can only temporarily disable an exact stop in G61 mode or by G09 between two cutting feed sessions.

- Stop with message #3006 (Attribute : W)

By specifying #3006 = 1 (MESSAGE) in a macro, operation can be stopped after executing up to the immediately preceding block. A message not longer than 26 characters enclosed in a control-out code and control-in code is displayed on the external operator message screen if the message is programmed in the same block.

Value number	Value name	Description
#3006	[#_MSGSTP]	Stop with a message

- Mirror image state #3007 (Attribute : R)

By reading #3007, the mirror image (setting or DI) state at that time can be checked for each axis.

Value number	Value name	Description
#3007	[#_MRIMG]	Mirror image state

In the binary representation below, each bit corresponds to an axis.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
n-th axis	-	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

When a bit is set to 0, mirror image is disabled for the corresponding axis. When a bit is set to 1, mirror image is enabled for the corresponding axis.

[Example] When #3007 is set to 3, mirror image is enabled for the first and second axes.

NOTE

Programmable mirror image state is not reflected in this variable.

- Program restart state #3008 (Attribute : R)

By reading #3008, whether the program is being restarted at that time can be checked.

Value number	Value name	Description
#3008	[#_PRSTR]	0 : Program not being restarted 1 : Program being restarted

- Time information #3011, #3012 (Attribute : R)

By reading system variables #3011 and #3012, year, month, day, hours, minutes, and seconds information can be checked. It is not possible to write to these variables. To update the year, month, day, hours, minutes, and seconds information, use the timer screen.

Value number	Value name	Description
#3011	[#_DATE]	Year/month/day
#3012	[#_TIME]	Hours/minutes/seconds

[Example] Seventeen minutes five seconds past four PM on May 20, 1998
#3011 = 19980520
#3012 = 161705

- Cutting time #3016 (Attribute: R/W)

By using the custom macro system variable #3016, the cumulative cutting time parameters (No. 103, No. 104) can be read and preset. The range of values is 0.0 to 1666666.65, and the unit is hours. When presetting is performed, a value less than one minute is treated as 0. The clock precision is 16 ms.

Variable No.	Variable name	Description
#3016	[#_CUTTM]	Reads and presets the cumulative cutting time parameters (No. 103, No. 104).

Example 1

Reading of a cutting time:

If #3016 is read when the cutting time is 5 hours, 45 minutes, and 18 seconds, 5.755 is read.

Example 2

Presetting of a cutting time:

If 5.755 is preset in #3016, 5.75 is actually set in #3016 because a value less than one minute is treated as 0. In parameter No. 103, 0 is set. In parameter No. 104, 345 is set.

- Cumulative number of machined parts and number of required parts #3901, #3902 (Attribute : R/W)

With the run time and part count display function, the number of required parts and the cumulative number of machined parts can be displayed on the screen. When the cumulative number of machined parts reaches the number of required parts, a signal is sent to the machine (PMC) to report the fact. The system variables can be used to read and write the cumulative number of machined parts and the number of required parts.

Value number	Value name	Description
#3901	[#_PRTSA]	Cumulative number of machined parts
#3902	[#_PRTSN]	Number of required parts

- Main program number #4000 (Attribute : R)

System variable #4000, even when placed in a subprogram of any level, can be used to read the main program number.

Value number	Value name	Description
#4000	[#_MAINO]	Main processing number

NOTE

1 The main processing number is the number of a program that was started first.

2 If an O number is specified by MDI during main program execution, or a second O number is specified in DNC mode, the value of #4000 is changed to the specified O number. If no program is registered or no O number is specified in DNC mode, the value of #4000 is 0.

3 Even if #4000 is used when drawing is performed in the background, the main program number is read in the background.

- Modal information #4001 to #4130, #4201 to #4330, #4401 to #4530 (Attribute : R)

By reading system variables #4001 to #4130, the modal information specified in the currently buffered block immediately preceding a macro statement that is also buffered and specified to read a system variable from #4001 to #4130 can be checked.
By reading system variables #4201 to #4330, the modal information of the block currently being executed can be checked.
By reading system variables #4401 to #4530, the modal information specified in the blocks up to the block that was interrupted by an interrupt-type custom macro can be checked.
The units used when the specification was made are used.

(Classification; (1) Immediately preceding block, (2) Block currently being executed, (3) Interrupted block)

Classification	Value number	Value name	Description
(1) (2) (3)	#4001 #4201 #4401	[#_BUFG[1]] [#_ACTG[1]] [#_INTG[1]]	Modal information (G code : Group 1)
(1) (2) (3)	#4002 #4202 #4402	[#_BUFG[2]] [#_ACTG[2]] [#_INTG[2]]	Modal information (G code : Group 2)
: :	: :	: :	: :
(1) (2) (3)	#4030 #4230 #4430	[#_BUFG[30]] [#_ACTG[30]] [#_INTG[30]]	Modal information (G code : Group 30)
(1) (2) (3)	#4102 #4302 #4502	[#_BUFB] [#_ACTB] [#_INTB]	Modal information (B code)
(1) (2) (3)	#4107 #4307 #4507	[#_BUFD] [#_ACTD] [#_INTD]	Modal information (D code)
(1) (2) (3)	#4108 #4308 #4508	[#_BUFE] [#_ACTE] [#_INTE]	Modal information (E code)
(1) (2) (3)	#4109 #4309 #4509	[#_BUFF] [#_ACTF] [#_INTF]	Modal information (F code)
(1) (2) (3)	#4111 #4311 #4511	[#_BUFH] [#_ACTH] [#_INTH]	Modal information (H code)
(1) (2) (3)	#4113 #4313 #4513	[#_BUFM] [#_ACTM] [#_INTM]	Modal information (M code)
(1) (2) (3)	#4114 #4314 #4514	[#_BUFN] [#_ACTN] [#_INTN]	Modal information (sequence number N)
(1) (2) (3)	#4115 #4315 #4515	[#_BUFO] [#_ACTO] [#_INTO]	Modal information (program number O)

Classifi-cation	Value number	Value name	Description
(1) (2) (3)	#4119 #4319 #4519	[#_BUFS] [#_ACTS] [#_INTS]	Modal information (S code)
(1) (2) (3)	#4120 #4320 #4520	[#_BUFT] [#_ACTT] [#_INTT]	Modal information (T code)
(1) (2) (3)	#4130 #4330 #4530	[#_BUFWZP] [#_ACTWZP] [#_INTWZP]	Modal information (additional workpiece coordinate system number P)

NOTE

Note on "immediately preceding block" and "block currently being executed"
The CNC reads, in advance, those blocks following the block currently being executed, so that the block currently being executed usually differs from the block that is buffered. In a program, the immediately preceding block is that block that is buffered by the CNC that immediately precedes a buffered block that specifies a system variable from #4001 to #4130.

```
[Example] O1234;
          N10 G00 X200. Y200. ;
          N20 G01 X1000. Y1000. F10. ;
                :
                :
          N50 G00 X500. Y500. ;
          N60 #1 = #4001 ;
```

Suppose that the CNC is currently executing N20, and that the CNC has buffered the blocks up to N60 (as is usually the case in multibuffer mode). The block currently being executed is N20, and the immediately preceding block is N50. So, the group 1 modal information of the block currently being executed is G01, and the group 1 modal information of the immediately preceding block is G00.
When N60 #1=#4201; is specified, #1 = 1.
When N60 #1=#4001; is specified, #1 = 0.

- Position information #5001 to #5080 (Attribute : R)

By reading system variables #5001 to #5080, the end point positions of the immediately preceding block, the currently specified positions (machine coordinate system, workpiece coordinate system), and the skip signal positions can be checked.

Variable number	Variable name	Position information	Coordinate system	Tool offset/ tool length compensation/ cutter compensation	Read during travel
#5001 #5002 : #5020	[#_ABSIO[1]] [#_ABSIO[2]] : [#_ABSIO[20]]	1st axis block end point position 2nd axis block end point position : 20th axis block end point position	Workpiece coordinate system	Not included	Possible
#5021 #5022 : #5040	[#_ABSMT[1]] [#_ABSMT[2]] : [#_ABSMT[20]]	1st axis current position 2nd axis current position : 20th axis current position	Machine coordinate system	Included	Impossible
#5041 #5042 : #5060	[#_ABSOT[1]] [#_ABSOT[2]] : [#_ABSOT[20]]	1st axis current position 2nd axis current position : 20th axis current position	Workpiece coordinate system	Included	Impossible
#5061 #5062 : #5080	[#_ABSKP[1]] [#_ABSKP[2]] : [#_ABSKP[20]]	1st axis skip signal position 2nd axis skip signal position : 20th axis skip signal position	Workpiece coordinate system	Included	Possible

NOTE

1 A variable value greater than the number of controlled axes is undefined.

2 The block end point position (ABSIO) of the skip function (G31) is that position at which the skip signal is turned on; if the skip signal is not turned on, ABSIO is the end point position of a block in which G31 is specified.

3 "Read during travel is impossible" means that the reading of a correct value during travel is not guaranteed.

- Tool length compensation #5081 to #5100 (Attribute : R)

By reading system variables #5081 to #5100, the tool length compensation value of each axis in the block currently being executed can be checked.

Variable number	Variable name	Position information	Read during travel
#5081	[#_TOFS[1]]	1st axis tool length compensation value	Impossible
#5082	[#_TOFS[2]]	2nd axis tool length compensation value	
:	:	:	
#5100	[#_TOFS[20]]	20th axis tool length compensation value	

NOTE
A variable value greater than the number of controlled axes is undefined.

- Servo positional deviation #5101 to #5120 (Attribute : R)

By reading system variables #5101 to #5120, the servo positional deviation of each axis can be checked.

Variable number	Variable name	Position information	Read during travel
#5101	[#_SVERR[1]]	1st axis servo positional deviation	Impossible
#5102	[#_SVERR[2]]	2nd axis servo positional deviation	
:	:	:	
#5120	[#_SVERR[20]]	20th axis servo positional deviation	

NOTE
A variable value greater than the number of controlled axes is undefined.

- Manual handle interrupt values #5121 to #5140 (Attribute : R)

By reading system variables #5121 to #5135, the manual handle interrupt value of each axis can be checked.

Variable number	Variable name	Position information	Read during travel
#5121	[#_MIRTP[1]]	1st axis manual handle interrupt value	Impossible
#5122	[#_MIRTP[2]]	2nd axis manual handle interrupt value	
:	:	:	
#5140	[#_MIRTP[20]]	20th axis manual handle interrupt value	

NOTE
A variable value greater than the number of controlled axes is undefined.

- Workpiece origin offsets #5201 to #5340, #7001 to #7960 (Attribute : R)

Workpiece origin offsets can be checked by reading system variables #5201 to #5340 and #7001 to #7960. Workpiece origin offsets can be updated by assigning desired values to the system variables.

Variable number	Variable name	Controlled axes	Workpiece coordinate system
#5201	[#_WZCMN[1]]	1st axis common workpiece origin offset	Common workpiece origin offset (applicable to all coordinate systems)
#5202	[#_WZCMN[2]]	2nd axis common workpiece origin offset	
:	:	:	
#5220	[#_WZCMN[20]]	20th axis common workpiece origin offset	
#5221	[#_WZG54[1]]	1st workpiece origin offset	G54
#5222	[#_WZG54[2]]	2nd workpiece origin offset	
:	:	:	
#5240	[#_WZG54[20]]	20th workpiece origin offset	
#5241	[#_WZG55[1]]	1st workpiece origin offset	G55
#5242	[#_WZG55[2]]	2nd workpiece origin offset	
:	:	:	
#5260	[#_WZG55[20]]	20th workpiece origin offset	
#5261	[#_WZG56[1]]	1st workpiece origin offset	G56
#5262	[#_WZG56[2]]	2nd workpiece origin offset	
:	:	:	
#5280	[#_WZG56[20]]	20th workpiece origin offset	
#5281	[#_WZG57[1]]	1st workpiece origin offset	G57
#5282	[#_WZG57[2]]	2nd workpiece origin offset	
:	:	:	
#5300	[#_WZG57[20]]	20th workpiece origin offset	
#5301	[#_WZG58[1]]	1st workpiece origin offset	G58
#5302	[#_WZG58[2]]	2nd workpiece origin offset	
:	:	:	
#5320	[#_WZG58[20]]	20th workpiece origin offset	
#5321	[#_WZG59[1]]	1st workpiece origin offset	G59
#5322	[#_WZG59[2]]	2nd workpiece origin offset	
:	:	:	
#5340	[#_WZG59[20]]	20th workpiece origin offset	

The workpiece origin offsets of additional workpiece coordinate systems can be handled as system variables as with a standard workpiece coordinate system. The system variable numbers are as follows :

Variable number	Variable name	Controlled axes	Additional workpiece coordinate system number
#7001 #7002 : #7020	[#_WZP1[1]] [#_WZP1[2]] : [#_WZP1[20]]	1st axis workpiece origin offset 2nd axis workpiece origin offset : 20th axis workpiece origin offset	1 (G54.1 P1)
#7021 #7022 : #7040	[#_WZP2[1]] [#_WZP2[2]] : [#_WZP2[20]]	1st axis workpiece origin offset 2nd axis workpiece origin offset : 20th axis workpiece origin offset	2 (G54.1 P2)
#7041 #7042 : #7060	[#_WZP3[1]] [#_WZP3[2]] : [#_WZP3[20]]	1st axis workpiece origin offset 2nd axis workpiece origin offset : 20th axis workpiece origin offset	3 (G54.1 P3)
: :	: :	: :	: :
#7941 #7942 : #7960	[#_WZP48[1]] [#_WZP48[2]] : [#_WZP48[20]]	1st axis workpiece origin offset 2nd axis workpiece origin offset : 20th axis workpiece origin offset	48 (G54.1 P48)

System variable number = 7000 + (coordinate system number - 1) x 20
+ axis number

Coordinate system number : 1 to 48
Axis number : 1 to 20

NOTE

A variable value greater than the number of controlled axes is undefined.

- Reference fixture offset values #15001 to #15160 (Attribute : R/W)

By reading system variables #15001 to #15160, the reference fixture offset values used with the rotary table dynamic fixture offset function can be checked. Reference fixture offset values can be updated by assigning desired values to the system variables.

Variable number	Variable name	Controlled axes	Fixture offset value
#15001 #15002 : #15020	[#_FOFS1[1]] [#_FOFS1[2]] : [#_FOFS1[20]]	1st axis reference fixture offset value 2nd axis reference fixture offset value : 20th axis reference fixture offset value	1 (G54.2 P1)
#15021 #15022 : #15040	[#_FOFS2[1]] [#_FOFS2[2]] : [#_FOFS2[20]]	1st axis reference fixture offset value 2nd axis reference fixture offset value : 20th axis reference fixture offset value	2 (G54.2 P2)
#15041 #15042 : #15060	[#_FOFS3[1]] [#_FOFS3[2]] : [#_FOFS3[20]]	1st axis reference fixture offset value 2nd axis reference fixture offset value : 20th axis reference fixture offset value	3 (G54.2 P3)
#15061 #15062 : #15080	[#_FOFS4[1]] [#_FOFS4[2]] : [#_FOFS4[20]]	1st axis reference fixture offset value 2nd axis reference fixture offset value : 20th axis reference fixture offset value	4 (G54.2 P4)
#15081 #15082 : #15100	[#_FOFS5[1]] [#_FOFS5[2]] : [#_FOFS5[20]]	1st axis reference fixture offset value 2nd axis reference fixture offset value : 20th axis reference fixture offset value	5 (G54.2 P5)
:	:	:	:
#15141 #15142 : #15160	[#_FOFS8[1]] [#_FOFS8[2]] : [#_FOFS8[20]]	1st axis reference fixture offset value 2nd axis reference fixture offset value : 20th axis reference fixture offset value	8 (G54.2 P8)

NOTE

A variable value greater than the number of controlled axes is undefined.

- Dynamic reference tool compensation values #16001 to #16160 (Attribute : R/W)

By reading system variables #16001 to #16160, the dynamic reference tool compensation values used with the rotary head dynamic tool compensation function can be checked. Dynamic reference tool compensation values can be updated by assigning desired values to the system variables.

Variable number	Variable name	Controlled axes	Dynamic tool compensation number
#16001	[#_DOFS1[1]]	1st axis dynamic reference tool compensation value	1 (G43.2H1)
#16002	[#_DOFS1[2]]	2nd axis dynamic reference tool compensation value	
:	:	:	
#16020	[#_DOFS1[20]]	20th axis dynamic reference tool compensation value	
#16021	[#_DOFS2[1]]	1st axis dynamic reference tool compensation value	2 (G43.2H2)
#16022	[#_DOFS2[2]]	2nd axis dynamic reference tool compensation value	
:	:	:	
#16040	[#_DOFS2[20]]	20th axis dynamic reference tool compensation value	
#16041	[#_DOFS3[1]]	1st axis dynamic reference tool compensation value	3 (G43.2H3)
#16042	[#_DOFS3[2]]	2nd axis dynamic reference tool compensation value	
:	:	:	
#16060	[#_DOFS3[20]]	20th axis dynamic reference tool compensation value	
#16061	[#_DOFS4[1]]	1st axis dynamic reference tool compensation value	4 (G43.2H4)
#16062	[#_DOFS4[2]]	2nd axis dynamic reference tool compensation value	
:	:	:	
#16080	[#_DOFS4[20]]	20th axis dynamic reference tool compensation value	
#16081	[#_DOFS5[1]]	1st axis dynamic reference tool compensation value	5 (G43.2H5)
#16082	[#_DOFS5[2]]	2nd axis dynamic reference tool compensation value	
:	:	:	
#16100	[#_DOFS5[20]]	20th axis dynamic reference tool compensation value	
:	:	:	:
#16141	[#_DOFS8[1]]	1st axis dynamic reference tool compensation value	8 (G43.2H8)
#16142	[#_DOFS8[2]]	2nd axis dynamic reference tool compensation value	
:	:	:	
#16160	[#_DOFS8[20]]	20th axis dynamic reference tool compensation value	

NOTE

A variable value greater than the number of controlled axes is undefined.

- System constants #0, #3100 to #3102 (Attribute : R)

Fixed values or constants used with the system can be handled in the same way as system variables. These constants are referred to as system constants. The following system constants are supported :

Constant number	Constant name	Description
#0 , #3100	[#_EMPTY]	Null value
#3101	[#_PI]	Circle ratioπ = 3.14159265358979323846
#3102	[#_E]	Base of natural logarithm e = 2.71828182845904523536

- Common variable name setting and specification

With the following command, a name not longer than 8 characters can be assigned to 50 common variables (#500 to #549) :

SETVN n[VAR500,VAR501,VAR502, - - -] ;

n is the start number of common variables to be named. VAR500, VAR501, VAR502, and so forth are the variable names of variable number n, n+1, n+2, and so forth. Character strings are separated from each other by a comma (,). All codes that the system can recognize as significant information can be used except for control-in and control-out codes ([ ]), EOB, EOR, and : (used with a program number). Each common variable name must start with a letter. No common variable name is deleted when the power is turned off. Set common variable names are displayed on the custom macro screen in the same way as with system variable names.

By specifying a set common variable name, the common variable can be both read and written. When specifying a common variable name, use the [# common variable name] format for [#VAR500].

[Example]

SETVN 510[TOOL_NO,WORK_NO,COUNTER1,COUNTER2];

This command assigns the following variable names :

Variable	Name
#510	TOOL_NO
#511	WORK_NO
#512	COUNTER1
#513	COUNTER2

A variable name thus assigned can be used in a program. When assigning 10 to #510, for example, the user can specify [#TOOL_NO]=10; instead of #510=10;.

17.3 ARITHMETIC COMMANDS

A variety of arithmetic operations can be performed on variables. An arithmetic command must be specified the same as in general arithmetic expressions.

<Formula>

<Formula>, the right-hand-side of an arithmetic command is a combination of constants, variables, functions and operators. A constant can be used instead of #i, #j, and #k. A constant without a decimal point used in <Formula> is considered to have a decimal point at the end.

Table17.3 (a) Operation Commands

Operation type	Operation command	Description
(1)Definition, replacement	#i=#j	Definition or replacement of a variable
(2)Addition operation	#i=#j+#k	Addition
	#i=#j-#k	Subtraction
	#i=#j OR #k	Logical OR. (Each pair of bits is handled separately, each argument being 32 bits in length.)
	#i=#j XOR #k	Exclusive-OR. (Each pair of bits is handled separately, each argument being 32 bits in length.)
(3)Multiplication operation	#i=#j*#k	Multiplication
	#i=#j/#k	Division
	#i=#j AND #k	Logical AND. (Each pair of bits is handled separately, each argument being 32 bits in length.)
	#i=#j MOD #k	Remaindering. (The remainder is determined after #j and #k are rounded to integers. If #j is negative, #i is also negative.)
(4)Function	#i=SIN[#j]	Sine (angle in degrees)
	#i=COS[#j]	Cosine (angle in degrees)
	#i=TAN[#j]	Tangent (angle in degrees)
	#i=ASIN[#j]	Arcsine
	#i=ACOS[#j]	Arccosine
	#i=ATAN[#j]	Arctangent (one argument). Can be shortened to ATN.
	#i=ATAN[#j]/[#k]	Arctangent (two arguments). Can be shortened to ATN.
	#i=ATAN[#j,#k]	Same as above.
	#i=SQRT[#j]	Square root. Can be shortened to SQR.
	#i=ABS[#j]	Absolute value
	#i=BIN[#j]	Convert from BCD to BINNARY
	#i=BCD[#j]	Convert from BINARY to BCD
	#i=ROUND[#j]	Rounding. Can be shortened to RND.
	#i=FIX[#j]	Rounding off to an integer
	#i=FUP[#j]	Rounding up to an integer
	#i=LN[#j]	Natural logarithms
	#i=EXP[#j]	Exponentiate e (2.718...).
	#i=POW[#j,#k]	Exponentiation (#j to the #k-th power)
	#i=ADP[#j]	Add decimal point

Explanation

- Angle units

The units of angles used with the SIN, COS, ASIN, ACOS, TAN, and ATAN functions are degrees. For example, 90 degrees and 30 minutes is represented as 90.5 degrees.

- ARCSIN #i = ASIN[#j];

- The solution ranges from -90 to 90 deg.
- # When #j is beyond the range of -1 to 1, alarm PS0119 is issued.
- A constant can be used instead of the #j variable.

- ARCCOS #i = ACOS[#j];

- The solution ranges from 180 to 0deg.
- When #j is beyond the range of -1 to 1, alarm PS0119 is issued.
- A constant can be used instead of the #j variable.

- Arctangent #i = ATAN[#j]/[#k]; (two arguments)

- Can be written as ATAN[#j, #k].
- This function returns the value of the arctangent for the angle formed by a point (#k, #j) on the X-Y plane.
- A constant can be used instead of variable #j.
- The range of the answer is as follows :
  Parameter ATN (bit 1 of No. 7003) = 0 : -180 to 180deg.
  [Example] If #1 = ATAN [-1]/[-1];, #1 is equal to -135.0.
  Parameter ATN (bit 1 of No. 7003)= : 0 to 360deg.
  [Example] If #1 = ATAN [-1]/[-1];, #1 is equal to 225.0.

- Arctangent #i = ATAN[#j]; (one argument)

- When specified with one argument only, ATAN returns the main arctangent value (-90 deg. ≤ ATAN[#j] ≤ 90 deg.). ATAN in this case has exactly the same function as that on a calculator.
- When using the value of this function as the dividend in division, enclose the function in brackets ([]). Otherwise, the function is regarded as being ATN[#j]/[#k].

[Example]#100 = [ATAN[1]]/10 ; : Divides ATAN with one argument by 10.

#100 = ATAN[1]/[10] ; : Executes ATAN as ATAN with two arguments.

#100 = ATAN[1]/10 ; : Regards ATAN as being ATAN with two arguments, but causes alarm PS0131 to be issued, because the X coordinate is not enclosed in [].

- Natural logarithm #i = LN[#j];

- When the antilogarithm (#j) is zero or smaller, alarm PS0119 is issued.
- A constant can be used instead of the #j variable.

- Exponential function #i = EXP[#j];

- When the result (j) of the operation exceeds about 709, an overflow occurs and alarm PS0111 is issued.
- A constant can be used instead of the #j variable.

- ROUND function

- When the ROUND function is included in an arithmetic or logic operation command, IF statement, or WHILE statement, the ROUND function rounds off at the first decimal place.

[Example]When #1=ROUND[#2]; is executed where #2 holds 1.2345, the value of variable #1 is 1.0.

- When the ROUND function is used in NC statement addresses, the ROUND function rounds off the specified value according to the least input increment of the address.

[Example]Creation of a drilling program that cuts according to the values of variables #1 and #2, then returns to the original position Suppose that the increment system is 1/1000 mm, variable #1 holds 1.2345, and variable #2 holds 2.3456. Then,

```
G00 G91 X-#1;   Moves 1.235 mm.
G01 X-#2 F300;  Moves 2.346 mm.
G00 X[#1+#2];   Since 1.2345 + 2.3456 = 3.5801, the travel
                distance is 3.580, which does not return
                the tool to the original position.
```

This difference comes from whether addition is performed before or after rounding off.

G00X-[ROUND[#1]+ROUND[#2]] must be specified to return the tool to the original position.

- ADP (Add Decimal Point)

- By using ADP[#n] (where n = 1 to 33), a subprogram can add the decimal point to an argument passed without the decimal point.

[Example] In the subprogram called by G65 P_X10;, the value of ADP[#24] is 10., with a decimal point added at the end of the argument. This can be used when the subprogram does not consider the increment system. If, however, 1 is set for parameter CVA (bit 4 of No. 7000), the ADP function cannot be used because the argument is converted to 0.01 as soon as it is passed.

NOTE

To maintain program compatibility, it is recommended that the decimal point be added by the argument specification at the time of a macro call, instead of the ADP function.

- Rounding up and down to an integer

With CNC, when the absolute value of the integer produced by an operation on a number is greater than the absolute value of the original number, such an operation is referred to as rounding up to an integer. Conversely, when the absolute value of the integer produced by an operation on a number is less than the absolute value of the original number, such an operation is referred to as rounding down to an integer. Be particularly careful when handling negative numbers.

[Example]

Suppose that #1=1.2 and #2=-1.2.
When #3=FUP[#1] is executed, 2.0 is assigned to #3.
When #3=FIX[#1] is executed, 1.0 is assigned to #3.
When #3=FUP[#2] is executed, -2.0 is assigned to #3.
When #3=FIX[#2] is executed, -1.0 is assigned to #3.

- Abbreviations of arithmetic and logic operation commands

When a function is specified in a program, the first two characters of the function name can be used to specify the function.

[Example] ROUND → RO
FIX → FI

- Priority of operations

(1)Functions
(2)Operations such as multiplication and division (*, /, AND)
(3)Operations such as addition and subtraction (+, -, OR, XOR)

Example) $\#1=\underbrace{\#2+\underbrace{\#3*\underbrace{SIN[\#4]}_{(1)}}_{(2)}}_{(3)};$

(1), (2) and (3) indicate the order of operations.

- Bracket nesting

Brackets are used to change the order of operations. Brackets can be used to a depth of five levels including the brackets used to enclose a function. When a depth of five levels is exceeded, P/S alarm No. 118 occurs.

Example) $\#1=\underbrace{SIN\ \underbrace{[\ [\ \underbrace{[\underbrace{\#2+\#3}_{(1)}]\ *\#4}_{(2)}\ +\#5}_{(3)}]\ *\#6}_{(4)}]}_{(5)}\ ;$

(1)to(5) indicate the order of operations.

Limitation

- Data type

The numeric data handled by custom macros are double-precision real data, as laid down in the applicable IEEE standard. Any errors associated with the execution of operations conform to the standard.

- Cautions on reduced precision

- Addition and subtraction

In addition or subtraction, if the absolute value is used as a subtrahend, the relative error cannot be held within 10^{-15}.

For example, let us assume that the actual values of #1 and #2 are as follows in an operation. (These values are given only as examples. In reality, they cannot be specified from a program.)

```
#1=9876543210.987654321
#2=9876543210.987657777
```

Operation #2 - #1 does not produce the following :

```
#2-#1=0.000003456
```

The reason for this is as follows : Custom macro variables are decimal numbers with a precision of 15 digits. Thus, #1 and #2 can only be as precise as the following :

```
#1=9876543210.123450000
#2=9876543210.123460000
```

(Strictly speaking, the internal values differ slightly from the above values because they are binary.) Thus, the result will be

```
#2-1=0.000010000
```

This causes a large error.

- Relational operators

EQ, NE, GT, GE, and LE are basically the same as addition and subtraction operators. Therefore, beware of errors. For example, in the example above, the following cannot always determine whether #1 is equal to #2, because of the possible error :

```
IF [#1 EQ #2]
```

It is necessary to allow for the possible error and determine that #1 is equal to #2 if the difference is within the error, by using, for example, the following :

```
IF [ABS [#1_#2]LT 0.1]
```

- Trigonometric functions

In the trigonometric functions, the absolute error is guaranteed. Because, however, the relative error cannot be held within 10^{-15}, multiplication and division must be used with care after trigonometric function operations.

- FIX function

Pay careful attention to the precision when applying the FIX function to the result of an operation.

For example, #3 will not necessary be equal to 2 if the following are performed :

```
N10 #1=0.002 ;
N20 #2=#1*1000 ;
N30 #3=FIX[#2] ;
```

The reason for this is that, because of the possible error in the operation of N20, #2 may not be equal to

#2=2.0000000000000000 but instead be equal to a slightly smaller value, such as

#2=1.9999999999999997

To prevent this from occurring, change the N30 line as follows :

N30 #3=FIX[#2+0.001];

In general,

FIX[expression] must be changed to FIX[expression±ε]

(where ε must be +ε if the value of the expression is positive, or -ε if it is negative. It must be 0.1, 0.01, 0.001, and so on, depending on the application).

- Brackets

Brackets ([,]) are used to enclose an expression. Note that parentheses are used for comments.

- Divisor

If the denominator is equal to "0" in division, alarm PS0112 is issued.

17.4 MACRO STATEMENTS AND NC STATEMENTS

The following blocks are referred to as macro statements :

- Blocks containing an arithmetic or logic operation (=)
- Blocks containing a control statement (such as GOTO, DO, END)
- Blocks containing a macro call command (such as macro calls by G65, G66, G67, or other G codes, or by M codes)

Any block other than a macro statement is referred to as an NC statement.

Explanation

- Differences from NC statements

- Even when single block mode is on, the machine does not stop. Note, however, that the machine stops in the single block mode when bit 5 of parameter SBM No. 0010 is 1.
- Macro blocks are not regarded as blocks that involve no movement in the cutter compensation mode.

- NC statements that have the same property as macro statements

- NC statements that include a subprogram call command (such as subprogram calls by M98 or other M codes, or by T codes) and not include other command addresses except an O,N or L address have the same property as macro statements.
- The blocks not include other command addresses except an O,N,P or L address have the same property as macro statements.

17.5 BRANCH AND REPETITION

In a program, the flow of control can be changed using the GOTO statement and IF statement. Three types of branch and repetition operations are used:

Branch and repetition
- GOTO statement (unconditional branch)
- IF statement (conditional branch: if ..., then...)
- WHILE statement (repetition while ...)

17.5.1 Unconditional Branch (GOTO Statement)

A branch to sequence number n occurs. When a sequence number outside of the range 1 to 99999999 is specified, alarm PS0128 occurs. A sequence number can also be specified using an expression.

GOTOn ; n : Sequence number (1 to 99999999)

```
[Example]  GOTO 1;
           GOTO #10;
```

⚠ **WARNING**

Within a single program, do not specify multiple blocks having the same sequence number. Otherwise, when the GOTO statement causes a branch, there is no guarantee that the program will branch to the intended point. This is very dangerous.

NOTE

1 Backward branching takes longer than forward branching.

2 In the block having sequence number n to which GOTOn causes a branch, the sequence number must appear at the beginning of the block. Otherwise, the branch fails.

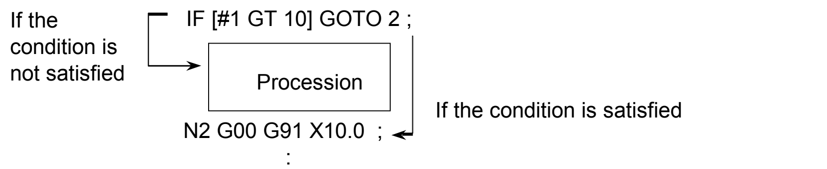
17.5.2 Conditional Branch (IF Statement)

A <conditional expression> is specified after IF.

IF[<conditional expression>]GOTOn

If the <conditional expression> is satisfied (true), the processing branches to sequence number n. If the conditional expression is not satisfied, a subsequent block is executed.

If the value of variable #1 is greater than 10, a branch to sequence number N2 occurs.



```
IF [#1 GT 10] GOTO 2 ;
N2 G00 G91 X10.0 ;
:
```

If the condition is not satisfied
Procession
If the condition is satisfied

IF[<conditional expression>]THEN

If the <conditional expression> is satisfied (true), a macro statement specified after THEN is executed. Only one macro statement can be executed.

If the values of #1 and #2 are the same, 0 is assigned to #3.

```
IF[#1 EQ #2] THEN#3=0 ;
```

If #1 equals #2 and if #3 equals #4, 0 is assigned to #5.

```
IF[[#1 EQ #2] AND [#3 EQ #4]] THEN#5=0 ;
```

If #1 equals #2 or if #3 equals #4, 0 is assigned to #5.

```
IF[[#1 EQ #2] OR [#3 EQ #4]] THEN#5=0 ;
```

Explanation

- <Conditional expression>

A <conditional expression> can be a <simple conditional expression> or a <compound conditional expression>. A <simple conditional expression> has a relational operator as listed in Table 1.1.2 (a) between two variables or between a variable and a constant. An <expression> can be specified instead of a variable. A <compound conditional expression> ANDs (logical product), ORs (logical sum) or XORs (exclusive logical sum) the results of multiple <simple conditional expressions>.

- Relational operator

A relational operator consists of two alphabetic characters as shown in the table below and is used to judge whether an operand is greater, smaller, or equal. The equal (=), greater than (>), and less than (<) signs cannot be used as relational operators.

Table17.5.2 Operators

Operator	Meaning
EQ	Equal to (=)
NE	Not equal to (≠)
GT	Greater than (>)
GE	Greater than or equal to (≥)
LT	Less than (<)
LE	Less than or equal to (≤)

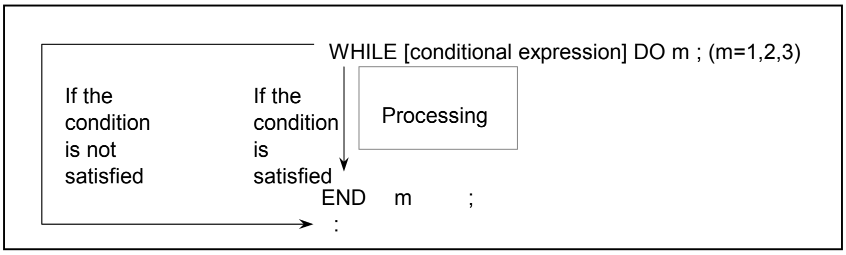
Sample program

The sample program below finds the total of numbers 1 to 10.

```
O9500;
#1=0;................................Initial value of the variable to
                                     hold the sum
#2=1;................................Initial value of the variable as
                                     an addend
N1 IF[#2 GT 10] GOTO 2; Branch to N2 when the
                                     addend is greater than 10
#1=#1+#2; .........................Calculation to find the sum
#2=#2+1; ...........................Next addend
GOTO 1;............................Branch to N1
N2 M30;.............................End of program
```

17.5.3 Repetition (While Statement)

Specify a conditional expression after WHILE. While the specified condition is satisfied, the program from DO to END is executed. If the specified condition is not satisfied, program execution proceeds to the block after END.

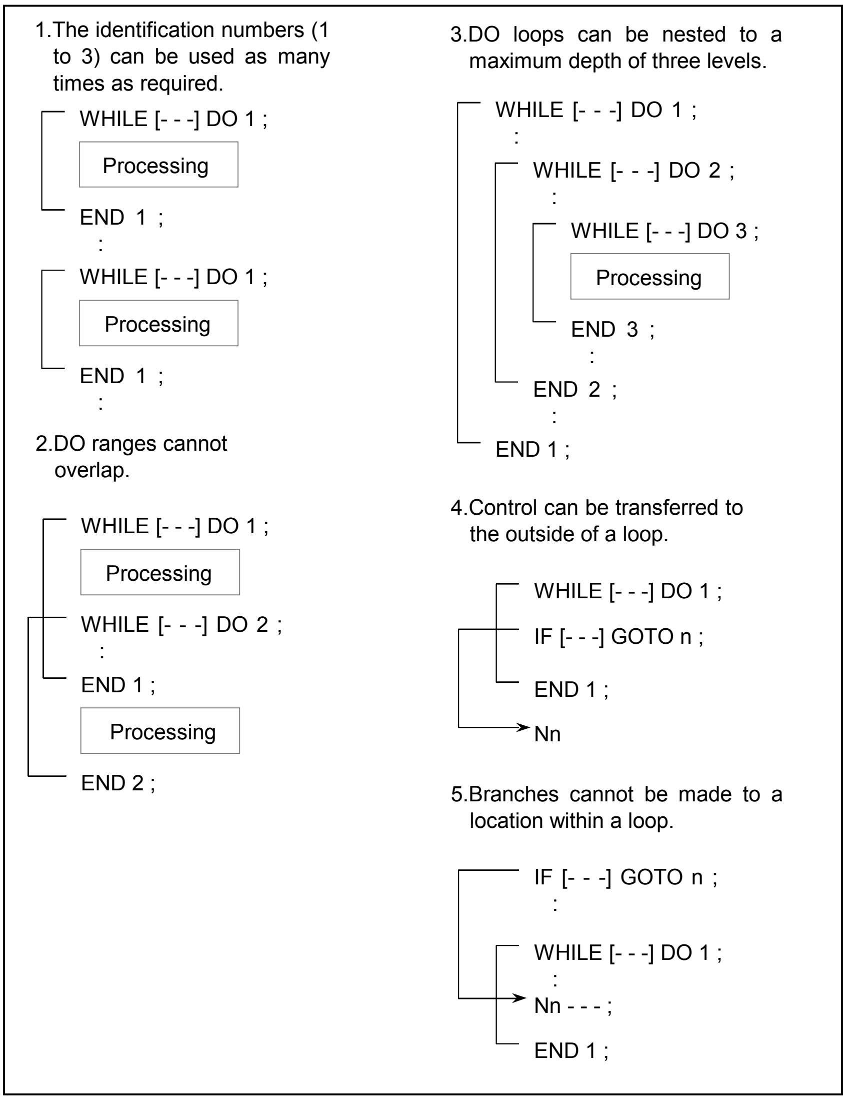

WHILE [conditional expression] DO m ; (m=1,2,3)
If the condition is not satisfied
If the condition is satisfied
Processing
END m ;
:

Explanation

While the specified condition is satisfied, the program from DO to END after WHILE is executed. If the specified condition is not satisfied, program execution proceeds to the block after END. The same format as for the IF statement applies. A number after DO and a number after END are identification numbers for specifying the range of execution. The numbers 1, 2, and 3 can be used. When a number other than 1, 2, and 3 is used, alarm PS0126 occurs.

- Nesting

The identification numbers (1 to 3) in a DO-END loop can be used as many times as desired. Note, however, when a program includes crossing repetition loops (overlapped DO ranges), alarm PS0124 occurs.



1.The identification numbers (1 to 3) can be used as many times as required.

```
WHILE [- - -] DO 1 ;
  Processing
END 1 ;
  :
WHILE [- - -] DO 1 ;
  Processing
END 1 ;
  :
```

2.DO ranges cannot overlap.

```
WHILE [- - -] DO 1 ;
  Processing
WHILE [- - -] DO 2 ;
  :
END 1 ;
  Processing
END 2 ;
```

3.DO loops can be nested to a maximum depth of three levels.

```
WHILE [- - -] DO 1 ;
  :
  WHILE [- - -] DO 2 ;
    :
    WHILE [- - -] DO 3 ;
      Processing
    END 3 ;
    :
  END 2 ;
  :
END 1 ;
```

4.Control can be transferred to the outside of a loop.

```
WHILE [- - -] DO 1 ;
IF [- - -] GOTO n ;
END 1 ;
Nn
```

5.Branches cannot be made to a location within a loop.

```
IF [- - -] GOTO n ;
  :
WHILE [- - -] DO 1 ;
  :
Nn - - - ;
END 1 ;
```

Limitation

- Infinite loops

When DOm is specified without specifying the WHILE statement, an infinite loop ranging from DO to END is produced.

- Processing time

When a branch to the sequence number specified in a GOTO statement occurs, the sequence number is searched for. For this reason, processing in the reverse direction takes a longer time than processing in the forward direction. Using the WHILE statement for repetition reduces processing time.

- Undefined variable

In a conditional expression that uses EQ or NE, a <vacant> and zero have different effects. In other types of conditional expressions, a <vacant> is regarded as zero.

Sample program

The sample program below finds the total of numbers 1 to 10.

```
O0001;
#1=0;
#2=1;
WHILE[#2 LE 10]DO 1;
#1=#1+#2;
#2=#2+1;
END 1;
M30;
```

17.6 MACRO CALL

A macro program can be called using the following methods:

- Macro call
  - Simple call (G65)
  - modal call (G66, G66.1, G67)
  - Macro call with G code
  - Macro call with M code
  - Subprogram call with M code
  - Subprogram call with T code
  - Subprogram call with S code
  - Subprogram call with 2nd auxiliary function

Limitation

- Differences between macro calls and subprogram calls

Macro call (G65) differs from subprogram call (M98) as described below.

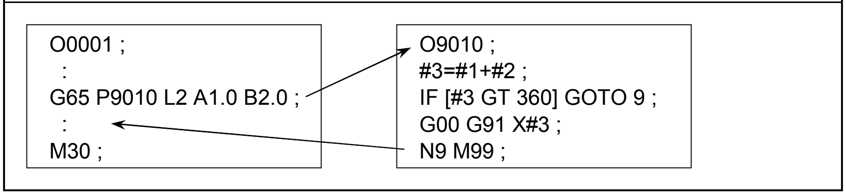
- With G65, an argument (data passed to a macro) can be specified. M98 does not have this capability.
- If a G65 block contains another NC command (for example, G01 X100.0 G65 Pp), an alarm "PS0090 NC statement/macro statement duplicated" is issued.
  When an M98 block contains another NC command (for example, G01 X100.0 M98Pp), the subprogram is called after the command is executed. On the other hand, G65 unconditionally calls a macro.
- A G65 block does not cause the machine to stop in a single-block mode.
  If an M98 block contains another NC command (for example, G01 X100.0 M98 Pp), the machine stops in a single-block mode.
- With G65, the level of local variables changes. With M98, the level of local variables does not change.

17.6.1 Simple Call (G65)

When G65 is specified, the custom macro specified at address P is called. Data (argument) can be passed to the custom macro program.

G65 Pp L Lambda < argument-specification > ;

P : Number of the program to call
Lambda : Repetition count (1 by default)
Argument : Data passed to the macro



```
O0001 ;
  :
G65 P9010 L2 A1.0 B2.0 ;
  :
M30 ;
```

```
O9010 ;
#3=#1+#2 ;
IF [#3 GT 360] GOTO 9 ;
G00 G91 X#3 ;
N9 M99 ;
```

Explanation

- Call

- After G65, specify at address P the program number of the custom macro to call
- When a number of repetitions is required, specify a number from 1 to 99999999 after address L. When L is omitted, 1 is assumed.
- By using argument specification, values are assigned to corresponding local variables.

- Argument specification

Two types of argument specification are available. Argument specification I uses letters other than G, L, O, N, and P once each. Argument specification II uses A, B, and C once each and also uses I, J, and K up to ten times. The type of argument specification is determined automatically according to the letters used.

- Argument specificationI

Address	Variable number
A	#1
B	#2
C	#3
D	#7
E	#8
F	#9
H	#11

Address	Variable number
I	#4
J	#5
K	#6
M	#13
Q	#17
R	#18
S	#19

Address	Variable number
T	#20
U	#21
V	#22
W	#23
X	#24
Y	#25
Z	#26

- Addresses G, L, N, O, and P cannot be used in arguments.
- Addresses that need not be specified can be omitted. Local variables corresponding to an omitted address are set to null.

- Argument specificationII

Argument specification II uses A, B, and C once each and uses I, J, and K up to ten times. Argument specification II is used to pass values such as three-dimensional coordinates as arguments.

Address	Variable number
A	#1
B	#2
C	#3
I_1	#4
J_1	#5
K_1	#6
I_2	#7
J_2	#8
K_2	#9
I_3	#10
J_3	#11

Address	Variable number
IK_3	#12
I_4	#13
J_4	#14
K_4	#15
I_5	#16
J_5	#17
K_5	#18
I_6	#19
J_6	#20
K_6	#21
I_7	#22

Address	Variable number
J_7	#23
K_7	#24
I_8	#25
J_8	#26
K_8	#27
I_9	#28
J_9	#29
K_9	#30
I_{10}	#31
J_{10}	#32
K_{10}	#33

- Subscripts of I, J, and K for indicating the order of argument specification are not written in the actual program.

Limitation

- Format

G65 must be specified before any argument.

- Mixture of argument specifications I and II

The CNC internally identifies argument specification I and argument specification II. If a mixture of argument specification I and argument specification II is specified, the type of argument specification specified later takes precedence.

- Position of the decimal point

The units used for argument data passed without a decimal point correspond to the least input increment of each address. The value of an argument passed without a decimal point may vary according to the system configuration of the machine. It is good practice to use decimal points in macro call arguments to maintain program compatibility.

The number of decimal positions in an argument if it is specified without a decimal point is as follows:

Address		Metric input	Inch input
D,E[*1],H,M,S,T		0	0
I,J,K,Q,R,U,V,W,X,Y,Z		α [*2]	α+1 [*2]
A,B,C		α [*2]	α+1(α) [*2] [*3]
F	G95 mode		4
	G94 mode	0(1) [*4]	2
Second auxiliary function address		β [*5]	β [*5]

NOTE

1 If address E is used as an axis name, using the program axis name expansion option, *2 and *3 apply.

2 The value of α differs with the increment system of the axis for which the address is set, as follows:
IS-A:2, IS-B:3, IS-C:4, IS-D:5, IS-E:6
If the input unit is multiplied by 10, i.e., parameter CIP (bit 0 of No.1000) is specified, the value of α is equal to the above value minus 1 for the IS-B, IS-C, IS-D, and IS-E axes. If the address is not an axis address, it conforms to the increment system of the reference axis.

3 If the address is used as an axis address of a rotation axis, the position of the decimal point is the value enclosed in parentheses ().

4 If 1 is set for parameter F41 (bit 1 of No. 2400) (the unit of the F code of feed per minute of metric input is 0.1 mm/min), the position of the decimal point is the value enclosed in parentheses ().

5 An option for the second auxiliary function is supported. When the address (specified by parameter No. 1030 A, B, C, U, V, or W) is used as the address of the second auxiliary function, the number of decimal places, β, specified by parameter No. 2428 is used. When parameter No. 2428 is not set to 0 and the input unit is multiplied by 10, i.e., parameter IMO (bit 0 of No. 1013) is used, the value of β is one less than the above value.

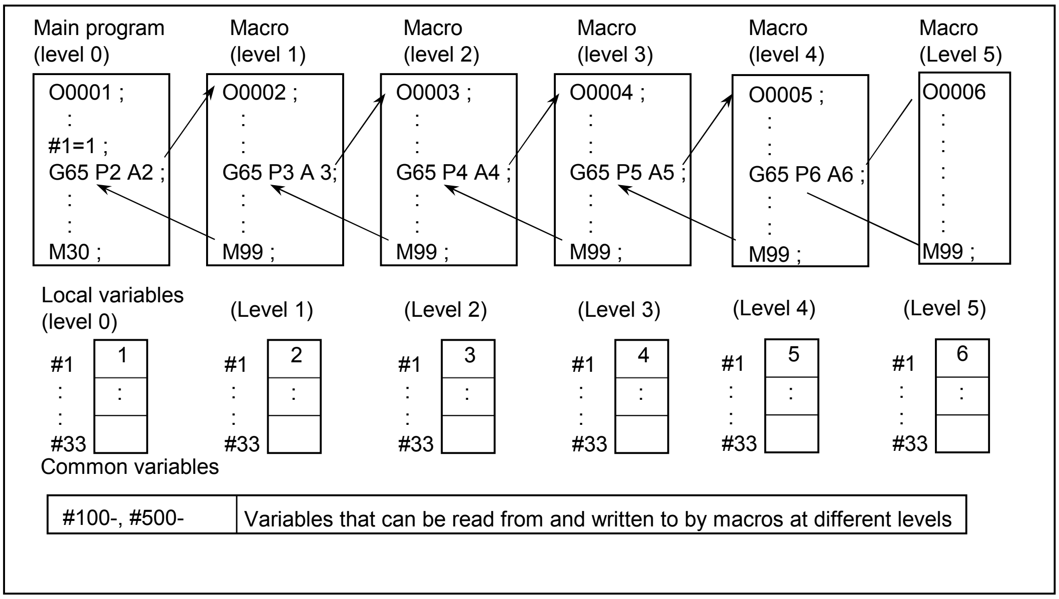
6 If 1 is set for parameter DPI (bit 0 of No. 2400) (calculator-type decimal point input), the number of decimal positions is zero.

- Call nesting

Calls can be nested to a depth of five levels including simple calls (G65) and modal calls (G66). This does not include subprogram calls (M98).
The multiplicity of subprogram calls (M98) is 10, including the number of macro calls.

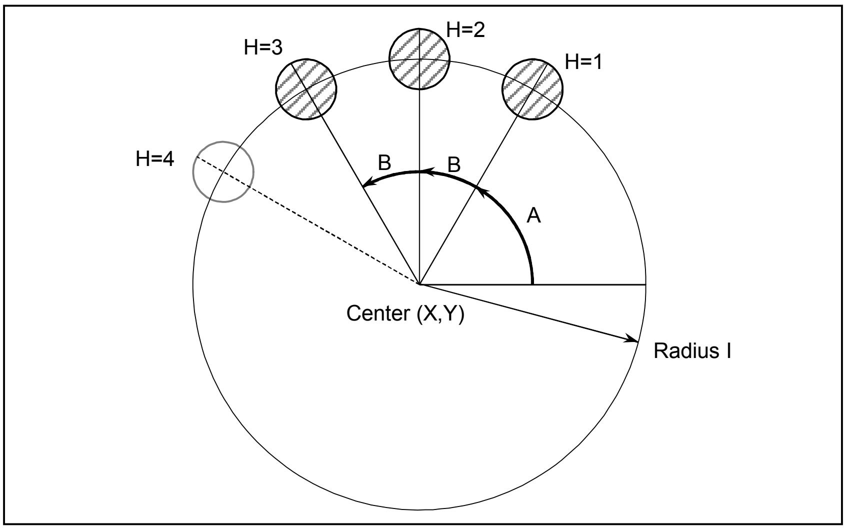
- Local variable levels

- Local variables from level 0 to 5 are provided for nesting.
- The level of the main program is 0.
- Each time a macro is called (with G65, G66 or G66.1), the local variable level is incremented by one. The values of the local variables at the previous level are saved in the CNC.
- When M99 is executed in a macro program, control returns to the calling program. At that time, the local variable level is decremented by one; the values of the local variables saved when the macro was called are restored.


Main program (level 0)
Macro (level 1)
Macro (level 2)
Macro (level 3)
Macro (level 4)
Macro (Level 5)
O0001 ; : #1=1 ; G65 P2 A2 ; : : M30 ;
O0002 ; : : G65 P3 A 3; : : M99 ;
O0003 ; : : G65 P4 A4 ; : : M99 ;
O0004 ; : : G65 P5 A5 ; : : M99 ;
O0005 ; : : G65 P6 A6 ; : : M99 ;
O0006 : : : : : M99 ;
Local variables (level 0)
(Level 1)
(Level 2)
(Level 3)
(Level 4)
(Level 5)
#1 1
#1 2
#1 3
#1 4
#1 5
#1 6
#33
Common variables
#100-, #500-
Variables that can be read from and written to by macros at different levels

Sample program (bolt hole circle)

A macro is created which drills H holes at intervals of B degrees after a start angle of A degrees along the periphery of a circle with radius I. The center of the circle is (X,Y). Commands can be specified in either the absolute or incremental mode. To drill in the clockwise direction, specify a negative value for B.



- Calling format

```
G65 P9100 Xx Yy Zz Rr Ff Ii Aa Bb Hh ;
```

X : X coordinate of the center of the circle
(absolute or incremental specification)..................................(#24)

Y : Y coordinate of the center of the circle
(absolute or incremental specification)..................................(#25)

Z : Hole depth..(#26)

R : Coordinates of an approach point..(#18)

F : Cutting feedrate ....(#9)

I : Radius of the circle ....(#4)

A : Drilling start angle ....(#1)

B : Incremental angle (clockwise when a negative value is specified)(#2)

H : Number of holes...(#11)

- Program calling a macro program

```
O0002;
G90 G92 X0 Y0 Z100.0;
G65 P9100 X100.0 Y50.0 R30.0 Z-50.0 F500 I100.0 A0 B45.0 H5;
M30;
```

- Macro program (called program)

```
O9100;
#3=#4003;   .........................Stores G code of group 3.
G81 Z#26 R#18 F#9 L0; .....Drilling cycle.
IF[#3 EQ 90]GOTO 1;.........Branches to N1 in the G90 mode.
#24=#5001+#24;..................Calculates the X coordinate of the center.
#25=#5002+#25;..................Calculates the Y coordinate of the center.
N1 WHILE[#11 GT 0]DO 1;..Until the number of remaining holes
                            reaches 0
#5=#24+#4*COS[#1];..........Calculates a drilling position on the X-
                           axis.
#6=#25+#4*SIN[#1];...........Calculates a drilling position on the Y-
                           axis.
G90 X#5 Y#6;......................Performs drilling after moving to the
                           target position.
#1=#1+#2;............................Updates the angle.
#11=#11-1;...........................Decrements the number of holes.
END 1;
G#3 G80; .............................Returns the G code to the original state.
M99;
```

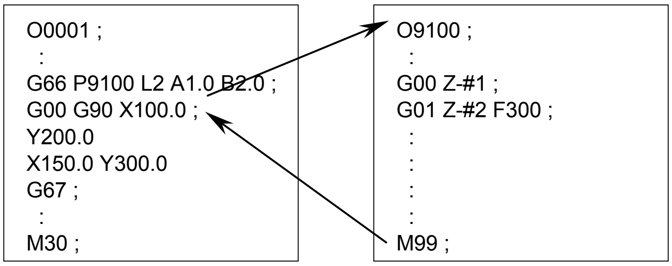
Meaning of variables:
- #3 : Stores the G code of group 3.
- #5 : X coordinate of the next hole to drill
- #6 : Y coordinate of the next hole to drill

17.6.2 Modal Call : Move Command Call (G66)

Once G66 is issued to specify a modal call a macro is called after a block specifying movement along axes is executed. This continues until G67 is issued to cancel a modal call.

```
G66 Pp Lλ <argument-specification > ;
```

P	:	Number of the program to call
λ	:	Repetition count (1 by default)
Argument	:	Data passed to the macro



```
O0001 ;
  :
G66 P9100 L2 A1.0 B2.0 ;
G00 G90 X100.0 ;
Y200.0
X150.0 Y300.0
G67 ;
  :
M30 ;
```

```
O9100 ;
  :
G00 Z-#1 ;
G01 Z-#2 F300 ;
  :
  :
  :
  :
M99 ;
```

Explanation

- Call

- After G66, specify at address P a program number subject to a modal call.
- When a number of repetitions is required, a number from 1 to 999999999 can be specified at address L.
- As with a simple call (G65), data passed to a macro program is specified in arguments.

- Cancellation

When a G67 code is specified, modal macro calls are no longer performed in subsequent blocks.

- Call nesting

Calls can be nested to a depth of four levels including simple calls (G65) and modal calls (G66). This does not include subprogram calls (M98). The multiplicity of subprogram calls (M98) is 10, including the number of macro calls.

- Modal call nesting

Modal calls can be nested by specifying another G66 code during a modal call.

Limitation

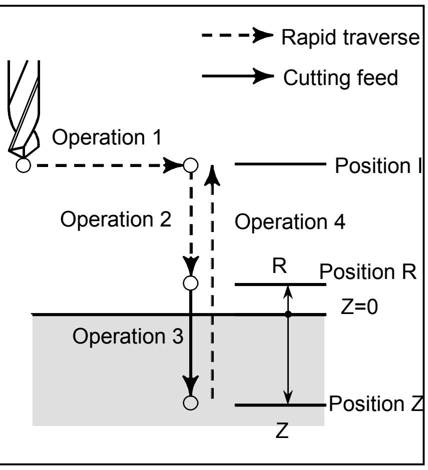
- In a G66 block, no macros can be called. Local variables (arguments) can, however, be set.
- 66 needs to be specified before any arguments.
- No macros can be called in a block which contains a code such as a miscellaneous function that does not involve movement along an axis.
- Local variables (arguments) can only be set in G66 blocks. Note that local variables are not set each time a modal call is performed.

Sample program

The same operation as the drilling canned cycle G81 is created using a custom macro and the machining program makes a modal macro call. For program simplicity,all drilling data is specified using absolute values.

The canned cycle consists of the following basic operations:

Operation 1 : Positioning along the X-axis and Y-axis
Operation 2 : Rapid traverse to point R
Operation 3 : Cutting feed to point Z
Operation 4 : Rapid traverse to point R or I



- Calling format

```
G66 P9110 Zz Rr Ff Ll ;
```

Z : Coordinates of position Z (absolute specification only)........(#26)
R : Coordinates of position R (absolute specification only) ....(#18)
F : Cutting feedrate ....(#9)
L : Repetition count

- Program that calls a macro program

```
O0001;
G28 G91 X0 Y0 Z0;
G92 X0 Y0 Z50.0;
G00 G90 X100.0 Y50.0;
G66 P9110 Z-20.0 R5.0 F500;
G90 X20.0 Y20.0;
X50.0;
Y50.0;
X70.0 Y80.0;
G67;
M30;
```

- Macro program (program called)

```
O9110;
    #1=#4001; ................................... Stores G00/G01.
    #3=#4003; ................................... Stores G90/G91.
    #4=#4109; ................................... Stores the cutting feedrate.
    #5=#5003; ................................... Stores the Z coordinate at the
                                                  start of drilling.
    G00 G90 Z#18; ............................... Positioning at position R
    G01 Z#26 F#9;................................ Cutting feed to position Z
    IF[#4010 EQ 98]GOTO 1;....................... Return to position I
    G00 Z#18; ................................... Positioning at position R
    GOTO 2;
N1 G00 Z#5;    ................................. Positioning at position I
N2 G#1 G#3 F#4;................................. Restores modal information.
    M99;
```

17.6.3 Modal Call : Per-Block Call (G66.1)

In this macro call mode, a specified macro is called unconditionally in each NC command block. All the commands in each block are regarded as being arguments, without being executed, except the O, N, and G codes. (For the G codes, only the code specified last is regarded as being an argument.)
Thus, each NC command block appears as if G65P were specified at the beginning of the block if no O or N code exists, or in the next block if an O or N code exists.

[Example]

```
G66.1 P100 ; mode,
N001 G01 G91 X100 Y200 D1 R1000 ; is equivalent to
N001 G65 P100 G01 G91 X100 Y200 D1 R1000 ;
```

Limitation

- G66.1 blocks
  (a) Even in G66.1 blocks, calls can be made.
  (b) The correspondence between argument addresses and variables is the same as that for simple calls.

- Blocks subsequent to a G66.1 block in which calls are made (excluding G66.1 blocks)
  (a) G, P, and L are also regarded as being arguments. G corresponds to #10, L to #12, and P to #16. The data is subject to the restrictions imposed on the input format of normal NC commands. For example, ;G1000.P0.12 L-4 is not allowed.
  (b) When multiple G codes appear, only the last one is regarded as being an argument. The O and N codes are inherited by the subsequent blocks as modal information, as are G codes in groups other than the 00 group.

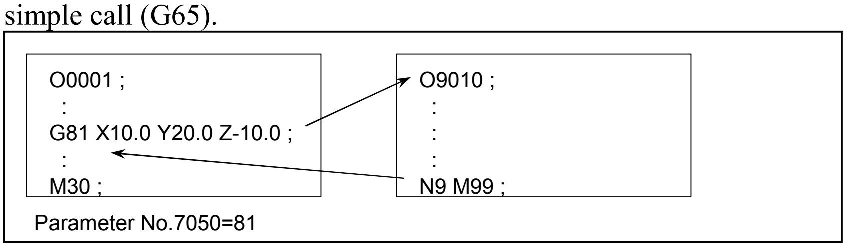
NOTE

1 Each block is regarded as being an NC command block. A per-block call is made when an address other than O or N is specified. If an N code is specified after an address other than O or N, the N code is regarded as being an argument. In this case, the N code corresponds to variable #14 and the number of decimal positions is 0.

2 In G66.1 mode, S, T, and secondary auxiliary function addresses cannot be specified.
[Example]
If G66.1 P1000; T12; is specified when 1 is set for parameter TCS (bit 0 of No. 7000) (subprogram call with the T code), alarm PS0093 is issued.

17.6.4 Macro Call Using G Code

By setting a G code number used to call a macro program in a parameter, the macro program can be called in the same way as for a simple call (G65).



```
O0001 ;
  :
G81 X10.0 Y20.0 Z-10.0 ;
  :
M30 ;
```

```
O9010 ;
  :
  :
  :
N9 M99 ;
```

Parameter No.7050=81

Explanation

By setting a G code number from -999 to 999 used to call a custom macro program (O9010 to O9019) in the corresponding parameter (N0.7050 to No.7059), the macro program can be called in the same way as with G65.
If a call is to be made with a G code with the decimal point, custom macroprogram O9040 to O9049 can be called by setting the code for parameter No. 7060 to 7069. The G code, for which the number of decimal positions is 1, must be multiplied by 10.

[Example] When 234 is set for parameter No. 7060, O9040 is called with G23.4.

If a negative G code is set, the call is regarded as being modal. In this case, whether the G code is equivalent to G66 or G66.1 can be specified using parameter MGE (bit 3 of No. 7000).
For example, when a parameter is set so that macro program O9010 can be called with G81, a user-specific cycle created using a custom macro can be called without modifying the machining program.

- Correspondence between parameter numbers and program numbers

G code without the decimal point

Program number	Parameter number
O9010	7050
O9011	7051
O9012	7052
O9013	7053
O9014	7054
O9015	7055
O9016	7056
O9017	7057
O9018	7058
O9019	7059

G code with the decimal point

Program number	Parameter number
O9040	7060
O9041	7061
O9042	7062
O9043	7063
O9044	7064
O9045	7065
O9046	7066
O9047	7067
O9048	7068
O9049	7069

- Repetition

As with a simple call, a number of repetitions from 1 to 999999999 can be specified at address L.

- Argument specification

As with a simple call, two types of argument specification are available: Argument specificationIand argument specification II. The type of argument specification is determined automatically according to the addresses used.

Limitation

- Nesting of calls using G codes

In a program called with a G code when parameter GMP (No.7000#5) is 1, no macros can be called using a G code. A G code in such a program is treated as an ordinary G code. In a program called as a subprogram with an M, S, T, second auxiliary function code when parameter GMP (No.7000#5) is 1, no macros can be called using a G code. A G code in such a program is also treated as an ordinary G code.

17.6.5 Macro Calls with G Codes (Specification of Multiple G Codes)

By setting the first G code to be used for a macro program call, the number of the first program to be called, and the number of code and call combinations, macro calls can be defined with multiple G codes.

Explanation

Using n G codes that start with that set for parameter No. 7090, n custom macros with the program numbers that start with that set for parameter No. 7091 can be called, n being the number set for parameter No. 7092. To disable the calls, set 0 for parameter No. 7092.
If a negative G code is set for parameter No. 7090, the calls are modal calls. In this case, whether the G code is equivalent to G66 or G66.1 can be specified using parameter MGE (bit 3 of No. 7000). The iteration and argument specifications are the same as those for a macro call with a G code.

[Example]

If No.7090=900, No.7091=10000000, and No.7092=100,

G900 → O10000000
G901 → O10000001
G902 → O10000002
:
G999 → O10000099

the above 100 combinations of G codes and custom macro calls (simple calls) are defined. If No. 7090 is changed to -900, the same combinations of G codes and custom macro calls (modal calls) are defined.

NOTE

1 In the following cases, all calls defined with these settings are nullified:
1)If a value outside the valid data range is set for a parameter.
2)If (No.7091+No.7092-1)>99999999

2 Simple and modal calls cannot be specified at the same time.

3 If a G code specified as described in this section is also specified for parameter No. 7050 to 7059, the call with parameter No. 7050 to 7059 will take precedence.

17.6.6 Macro Calls with G Codes with the Decimal Point (Specification of Multiple G Codes)

By setting the first G code with the decimal point to be used for a macro program call, the number of the first program to be called, and the number of code and call combinations, macro calls can be defined with multiple G codes with the decimal point.

Explanation

Using n G codes with the decimal point that start with that set for parameter No. 7093, n custom macros with the program numbers that start with that set for parameter No. 7094 can be called, n being the number set for parameter No. 7095. To disable the calls, set 0 for parameter No. 7095.
If a negative G code is set for parameter No. 7093, the calls are modal calls. In this case, whether the G code is equivalent to G66 or G66.1 can be specified using parameter MGE (bit 3 of No. 7000).
The iteration and argument specifications are the same as those for a macro call with a G code.
[Example]

If No.7093=900, No.7094=20000000, and No.7095=100,

```
G90.0 → O20000000
G90.1 → O20000001
G90.2 → O20000002
  :
G99.9 → O20000099
```

the above 100 combinations of G codes and custom macro calls (simple calls) are defined. If No. 7093 is changed to -900, the same combinations of G codes and custom macro calls (modal calls) are defined.

NOTE

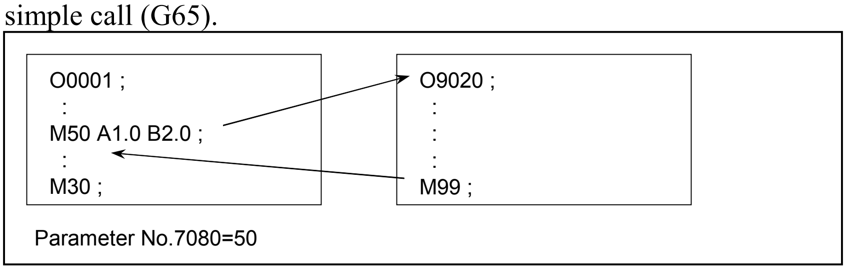
1 In the following cases, all calls defined with these settings are nullified:
   1) If a value outside the valid data range is set for a parameter.
   2) If (No.7094+No.7095-1)>99999999

2 Simple and modal calls cannot be specified at the same time.

3 If a G code specified as described in this section is also specified for parameter No. 7060 to 7069, the call with parameter No. 7060 to 7069 will take precedence.

17.6.7 Macro Call Using an M Code

By setting an M code number used to call a macro program in a parameter, the macro program can be called in the same way as with a simple call (G65).



```
O0001 ;
  :
M50 A1.0 B2.0 ;
  :
M30 ;
```

```
O9020 ;
  :
  :
  :
M99 ;
```

Parameter No.7080=50

Explanation

By setting an M code number from 3 to 99999999 used to call a custom macro program (O9020 to O9029) in the corresponding parameter (No.7080 to No.7089), the macro program can be called in the same way as with G65.

- Correspondence between parameter numbers and program numbers

Program number	Parameter number
O9020	7080
O9021	7081
O9022	7082
O9023	7083
O9024	7084
O9025	7085
O9026	7086
O9027	7087
O9028	7088
O9029	7089

- Repetition

As with a simple call, a number of repetitions from 1 to 999999999 can be specified at address L.

- Argument specification

As with a simple call, two types of argument specification are available: Argument specificationIand argument specification II. The type of argument specification is determined automatically according to the addresses used.

Limitation

- An M code used to call a macro program must be specified at the start of a block.
- In a macro called with a G code or in a program called as a subprogram with an M, S, T, 2nd auxiliary function code when parameter GMP (No.7000#5) is 1, no macros can be called using an M code. An M code in such a macro or program is treated as an ordinary M code.

17.6.8 Macro Calls with M Codes with the Decimal Point (Specification of Multiple G Codes)

By setting the first M code with the decimal point to be used for a macro program call, the number of the first program to be called, and the number of code and call combinations, macro calls can be defined with multiple M codes with the decimal point.

Explanation

Using n G codes with the decimal point that start with that set for parameter No. 7099, n custom macros with the program numbers that start with that set for parameter No. 7100 can be called, n being the number set for parameter No. 7101. To disable the calls, set 0 for parameter No. 7101.
The iteration and argument specifications are the same as those for a macro call with a M code.
[Example]

If No.7099=90000000, No.7100=40000000, and
No.7101=100,

```
M90000000 → O40000000
M90000001 → O40000001
M90000002 → O40000002
        :
M90000099 → O40000099
```

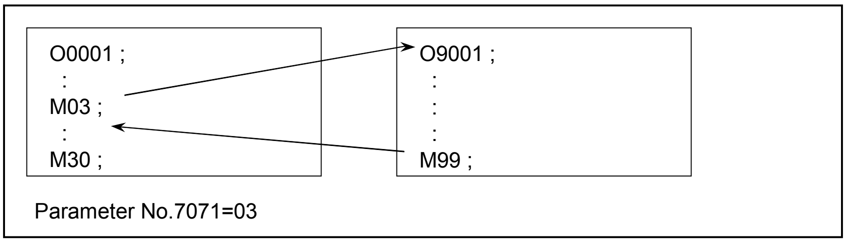
the above 100 combinations of M codes and custom macro calls (simple calls) are defined.

NOTE

1 In the following cases, all calls defined with these settings are nullified:
   1) If a value outside the valid data range is set for a parameter.
   2) If (No.7100+No.7101-1)>99999999
2 If a M code specified as described in this section is also specified for parameter No. 7080 to 7089, the call with parameter No. 7080 to 7089 will take precedence.

17.6.9 Subprogram Call Using an M Code

By setting an M code number used to call a subprogram (macro program) in a parameter, the macro program can be called in the same way as with a subprogram call (M98).



```
O0001 ;
  :
M03 ;
  :
M30 ;
```

```
O9001 ;
  :
  :
  :
M99 ;
```

Parameter No.7071=03

Explanation

By setting an M code number from 3 to 99999999 used to call a subprogram in a parameter (No.7071 to No. 7079), the corresponding custom macro program (O9001 to O9009) can be called in the same way as with M98.

- Correspondence between parameter numbers and program numbers

Program number	Parameter number
O9001	7071
O9002	7072
O9003	7073
O9004	7074
O9005	7075
O9006	7076
O9007	7077
O9008	7078
O9009	7079

- Repetition

As with a simple call, a number of repetitions from 1 to 999999999 can be specified at address L.

- Argument specification

Argument specification is not allowed.

- M code

An M code in a macro program that has been called is treated as an ordinary M code.

Limitation

In a macro called with a G code or in a program called with an M, S, T, 2nd auxiliary function code when parameter GMP (No.7000#5) is 1, no subprograms can be called using an M code. An M code in such a macro or program is treated as an ordinary M code.

17.6.10 Subprogram Call Using an M Code (Specification of Multiple G Codes)

By setting the first M code to be used for a subprogram call, the number of the first program to be called, and the number of code and call combinations, subprogram calls can be defined with multiple M codes.

Explanation

Using n G codes with the decimal point that start with that set for parameter No. 7096, n custom macros with the program numbers that start with that set for parameter No. 7097 can be called, n being the number set for parameter No. 7098. To disable the calls, set 0 for parameter No. 7098.

[Example]

If No.7096=80000000, No.7097=30000000, and No.7098=100,

```
M80000000 → O30000000
M80000001 → O30000001
M80000002 → O30000002
      :
M80000099 → O30000099
```

the above 100 combinations of M codes and subprogram calls are defined.

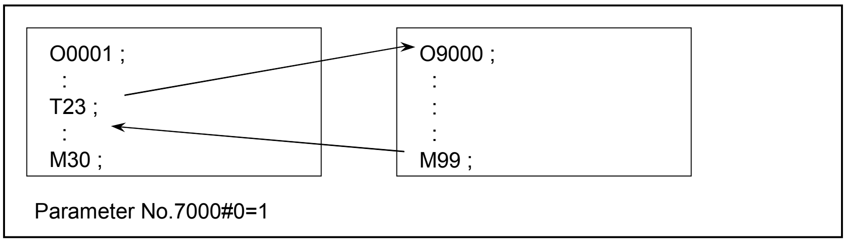
NOTE

1 In the following cases, all calls defined with these settings are nullified:
  1) If a value outside the valid data range is set for a parameter.
  2) If (No.7097+No.7098-1)>99999999

2 If a M code specified as described in this section is also specified for parameter No. 7071 to 7079, the call with parameter No. 7071 to 7079 will take precedence.

17.6.11 Subprogram Calls Using a T Code

By enabling subprograms (macro program) to be called with a T code in a parameter, a macro program can be called each time the T code is specified in the machining program.



```
O0001 ;
  :
T23 ;
  :
M30 ;
```

```
O9000 ;
  :
  :
  :
M99 ;
```

Parameter No.7000#0=1

Explanation

- Call

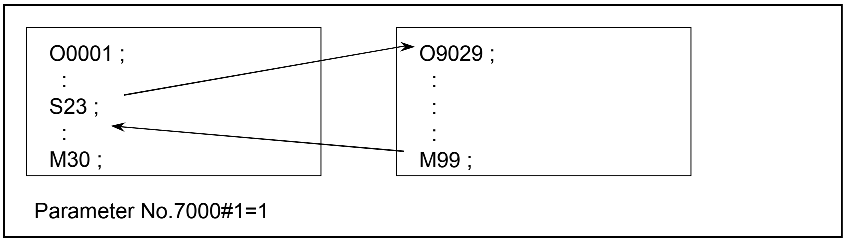
By setting bit 0 of parameter TCS No.7000 to 1, the macro program O9000 can be called when a T code is specified in the machining program. A T code specified in a machining program is assigned to common variable #149.

Limitation

In a macro called with a G code or in a program called with an M, S, T, 2nd auxiliary function code when parameter GMP (No.7000#5) is 1, no subprograms can be called using a T code. A T code in such a macro or program is treated as an ordinary T code.

17.6.12 Subprogram Calls Using a S Code

By enabling subprograms (macro program) to be called with a S code in a parameter, a macro program can be called each time the S code is specified in the machining program.



Explanation

- Call

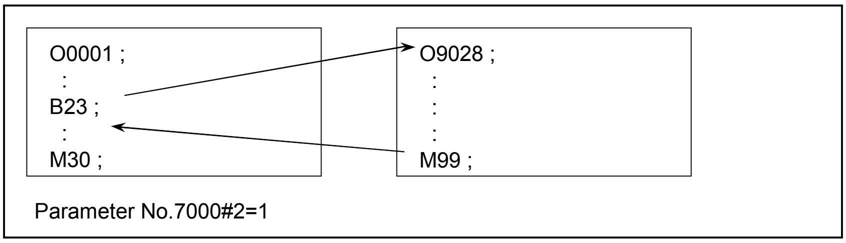
By setting bit 1 of parameter SCS No.7000 to 1, the macro program O9029 can be called when a S code is specified in the machining program. A S code specified in a machining program is assigned to common variable #147.

Limitation

In a macro called with a G code or in a program called with an M, S, T, 2nd auxiliary function code when parameter GMP (No.7000#5) is 1, no subprograms can be called using a S code. A S code in such a macro or program is treated as an ordinary S code.

17.6.13 Subprogram Calls Using a 2nd Auxiliary Function Code

By enabling subprograms (macro program) to be called with a 2nd auxiliary function code in a parameter, a macro program can be called each time the 2nd auxiliary function code is specified in the machining program.



```
O0001 ;
  :
B23 ;
  :
M30 ;
```

```
O9028 ;
  :
  :
  :
M99 ;
```

Parameter No.7000#2=1

Explanation

- Call

By setting bit 2 of parameter BCS No.7000 to 1, the macro program O9028 can be called when a 2nd auxiliary function code is specified in the machining program. A 2nd auxiliary function code specified in a machining program is assigned to common variable #146.

Limitation

In a macro called with a G code or in a program called with an M, S, T, 2nd auxiliary function code when parameter GMP (No.7000#5) is 1. no subprograms can be called using a 2nd auxiliary function code. A 2nd auxiliary function code in such a macro or program is treated as an ordinary 2nd auxiliary function code.

17.6.14 Sample Program

By using the subprogram call function that uses M codes, the cumulative usage time of each tool is measured.

- Conditions

- The cumulative usage time of each of tools T01 to T05 is measured.
  No measurement is made for tools with numbers greater than T05.
- The following variables are used to store the tool numbers and measured times:

#501	Cumulative usage time of tool number 1
#502	Cumulative usage time of tool number 2
#503	Cumulative usage time of tool number 3
#504	Cumulative usage time of tool number 4
#505	Cumulative usage time of tool number 5

- Usage time starts being counted when the M03 command is specified and stops when M05 is specified. System variable #3002 is used to measure the time during which the cycle start lamp is on. The time during which the machine is stopped by feed hold and single block stop operation is not counted, but the time used to change tools and pallets is included.

Operation check

- Parameter setting

Set 3 in parameter No.7071, and set 5 in parameter No.7072.

- Variable value setting

Set 0 in variables #501 to #505.

- Program that calls a macro program

```
O0001;
T01 M06;
M03;
  :
M05;..........................................Changes #501.
T02 M06;
M03;
  :
M05;..........................................Changes #502.
T03 M06;
M03;
  :
M05;..........................................Changes #503.
T04 M06;
M03;
  :
M05;..........................................Changes #504.
T05 M06;
M03;
  :
M05;..........................................Changes #505.
M30;
```

Macro program (program called)

```
O9001(M03);  Macro to start counting
    M01;
    IF[#4120 EQ 0]GOTO 9;.................No tool specified
    IF[#4120 GT 5]GOTO 9;.................Out-of-range tool number
    #3002=0; Clears the timer.
N9 M03; ........................ Rotates the spindle in the forward direction
    M99;

O9002(M05);  Macro to end counting
    M01;
    IF[#4120 EQ 0]GOTO 9;.................No tool specified
    IF[#4120 GT 5]GOTO 9;.................Out-of-range tool number
    #[500+#4120]=#3002+#[500+#4120];
            .........................................Calculates cumulative time.

N9 M05;     .........................................Stops the spindle.
    M99;
```

17.7 PROCESSING MACRO STATEMENTS

For smooth machining, the CNC prereads the NC statement to be performed next. This operation is referred to as buffering.

In multi-buffer mode, which is specified by setting MBF (bit 6 of parameter No. 2401) to 1 or assumed in G5.1Q1 mode, buffering is performed for the next NC statement, as well as multiple statements in subsequent blocks according to the settings; for example, statements in subsequent 5, 15, or 100 blocks are read.

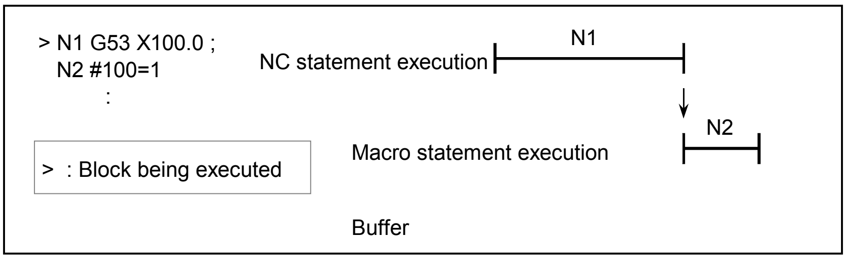
In cutter compensation mode (G41, G42), the NC statements in at least three subsequent blocks are read to calculate the point of intersection.

Macro statements for arithmetic expressions or conditional branches are, however, processed as soon as they are buffered (that is, they are read into the buffer). Therefore, macro statements are not always executed in the specified order.

Conversely, when a block contains M00, M01, M02, M30, an M code that prevents buffering set with any of parameters No. 2411 to No. 2418 and No. 2450 to No. 2453, or a G code (such as G53) that prevents buffering, subsequent blocks are not subjected to read-ahead processing. This ensures that this M or G code is executed before subsequent macro statements are executed.

Explanations

- When the next block is not buffered (M codes that prevent buffering, G53, etc.)

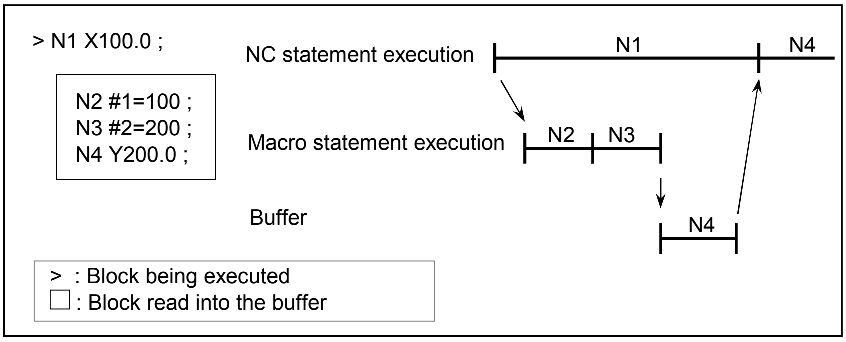


```
> N1 G53 X100.0 ;
  N2 #100=1
     :
```

⚠ **CAUTION**

To enable a macro statement to be executed upon completion of the execution of the previous NC statement, an M or G code that prevents buffering must be specified immediately before that macro statement as shown above. Specify this M or G code as necessary, especially when system variables for controlling signals, coordinates, offsets, and other elements are read or written, because these system variable values may vary depending on when NC statements are executed.

- Buffering the next block in other than cutter compensation mode (G41, G42)

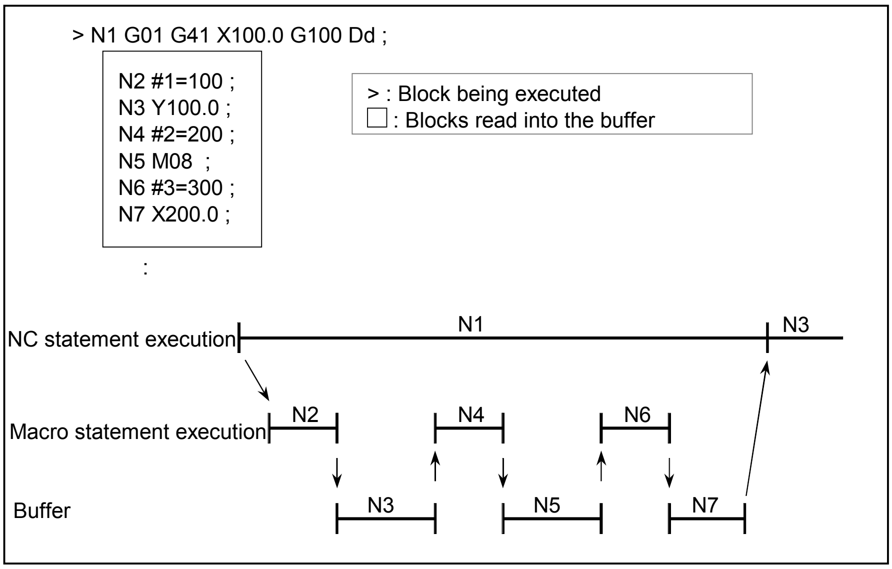


```
> N1 X100.0 ;
  N2 #1=100 ;
  N3 #2=200 ;
  N4 Y200.0 ;
```

When N1 is being executed, the next NC statement (N4) is read into the buffer.

The macro statements (N2, N3) between N1 and N4 are processed during execution of N1.

- In cutter compensation (G41, G42) mode



```
> N1 G01 G41 X100.0 G100 Dd ;
  N2 #1=100 ;
  N3 Y100.0 ;
  N4 #2=200 ;
  N5 M08  ;
  N6 #3=300 ;
  N7 X200.0 ;
     :
```

When the NC1 block is being executed, the NC statements in the next two blocks (up to N7) are read into the buffer. The macro statements (N2, N4, and N6) between N1 and N7 are processed during execution of N1.

17.8 REGISTERING CUSTOM MACRO PROGRAMS

Custom macro programs are similar to subprograms. They can be registered and edited in the same way as subprograms.
The storage capacity is determined by the total length of tape used to store both custom macros and subprograms.

17.9 CODES AND RESERVED WORDS USED IN CUSTOM MACROS

The following codes can be used in custom macro programs, in addition to the codes used in ordinary programs.

Explanation

- Codes

(1) ISO codes (represented by hole patterns in tape)

Meaning	8	7	6	5	4		3	2	1
[	O	O		O	O			O	O
]	O	O		O	O	O	O		O
#	O		O			O		O	O
*	O		O		O	O		O	
=	O		O	O	O	O	O		O
?			O	O	O	O	O	O	O
@	O	O				O			
&	O		O			O	O	O	
_		O		O	O	O	O	O	O
O	O	O			O	O	O	O	O

(2) EIA codes

Meaning	Codes
[	Code set for parameter No. 7010
]	Code set for parameter No. 7011
#	Code set for parameter No. 7012
*	Code set for parameter No. 7013
=	Code set for parameter No. 7014
?	Code set for parameter No. 7015
@	Code set for parameter No. 7016
&	Code set for parameter No. 7017
_	Code set for parameter No. 7018

O, the same code O as in the program number, must be used. The hole pattern for [,], #, * and = in EIA code must be set as parameters (Nos. 7010-7018).
However, the character with no punched hole cannot be used. Note that alphabetic codes can be used, but when used as these codes, they are not used in their proper sense.

- Reserved words

The following reserved words can be used in custom macros:

AND, OR, XOR, MOD, EQ, NE, GT, LT, GE, LE , SIN, COS, TAN, ASIN, ACOS, ATAN, ATN, SQRT, SQR, ABS, BIN, BCD, ROUND, RND, FIX, FUP, LN, EXP, POW, ADP, IF, GOTO, WHILE, DO, END, BPRNT, DPRNT, POPEN, PCLOS, SETVN

System variable (constant) names and registered common variable names are also regarded as being reserved words.

17.10 WRITE-PROTECTING COMMON VARIABLES

By setting variable numbers for parameters Nos. 7029 to 7032, multiple common variables (#500 to #999 or #200 to #499) can be protected, with their attributes changed to READ-only. This protection is effective against input/all clear from the macro screen using MDI and writing with macro programs. If an attempt is made to execute WRITE (on the left side) on the write-protected variables, using an NC program, alarm PS0116 is issued.

17.11 DISPLAYING A MACRO ALARM AND MACRO MESSAGE IN JAPANESE

Explanation

Kanji, katakana and hiragana characters as well as alphanumeric characters and special characters can be displayed on the alarm screen and external operator message screen using system variables 3000 and 3006.

To display alphanumeric and special characters, directly enter the alphanumeric and special characters. To display Japanese characters, enclose the internal codes of the characters in @ symbols. This is called internal code input. (The internal codes of kanji and hiragana characters are represented by four hexadecimal digits. The internal code of a katakana character is represented by two hexadecimal digits.)

If, for example, the following macro message is to be displayed

″MADE IN 日本-ﾌｧﾅｯｸLTD.製造″

specify the following:

```
#3006=1(MADE IN @867C 8B5C@-@CC A7 C5 AF
B8@LTD.@803D 8224@) ;
日   本    ﾌ  ｧ  ﾅ  ｯ  ｸ       製   造
```

Special character "@" cannot be used in the message, because it is used to identify Japanese internal codes.

In the internal code input state, text display is interrupted if:

- A character that cannot be recognized as an internal code is entered.
- A control-in signal is received.

In the internal code input state, the state terminates if:

- An odd number of hexadecimal characters are enclosed in @.

For an explanation of the internal codes for Kanji, hiragana, and katakana characters, see the FANUC internal code table.

17.12 EXTERNAL OUTPUT COMMANDS

In addition to the standard custom macro commands, the following macro commands are available. They are referred to as external output commands.

- BPRNT
- DPRNT
- POPEN
- PCLOS

These commands are provided to output variable values and characters through the reader/punch interface.

Explanations

Specify these commands in the following order:

Open command: POPEN

Before specifying a sequence of data output commands, specify this command to establish a connection to an external input/output device.

Data output command: BPRNT or DPRNT

Specify necessary data output.

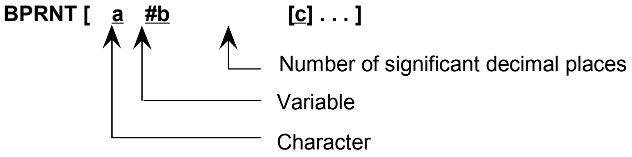
Close command: PCLOS

When all data output commands have completed, specify PCLOS to release a connection to an external input/output device.

- Open command POPEN

POPEN establishes a connection to an external input/output device. It must be specified before a sequence of data output commands. The CNC outputs a DC2 control code.

- Data output command BPRNT


BPRNT [a #b [c] . . .]

The BPRNT command outputs characters and variable values in binary.

(i) Specified characters are converted to the codes according to the parameter EIA(No.0000 #4) that is output at that time.
Specifiable characters are as follows:
- Letters (A to Z)
- Numbers
- Special characters (*, /, +, -, @ , ? , &)

> **NOTE**
> 1 An asterisk (*) is output by a space code.
> 2 Set the parameter EIA(No.0000 #4) to 0 (ISO code system is used for punch codes), when the special characters (@,?,&) are output,

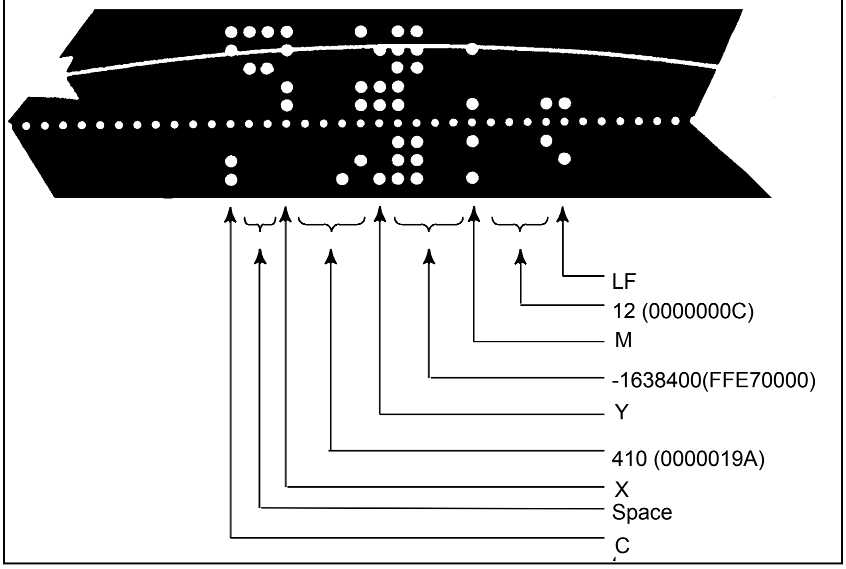
(ii) All variables are stored with a decimal point. Specify a variable followed by the number of significant decimal places enclosed in brackets. A variable value is treated as 2-word (32-bit) data, including the decimal digits. It is output as binary data starting from the highest byte.

(iii) When specified data has been output, an EOB code is output according to the parameter EIA(No.0000 #4).

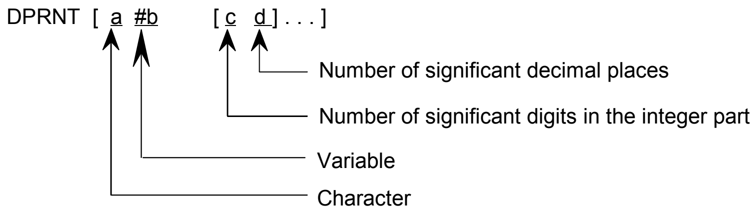
(iv) Null variables are regarded as 0.

Example)

```
BPRNT [ C** X#100 [3] Y#101 [3] M#10 [0] ]
  Variable value
   #100=0.40956
   #101=-1638.4
   #10=12.34
```



- Data output command DPRNT


DPRNT [a #b [c d] . . .]
Number of significant decimal places
Number of significant digits in the integer part
Variable
Character

The DPRNT command outputs characters and each digit in the value of a variable according to the code set in the settings (ISO).

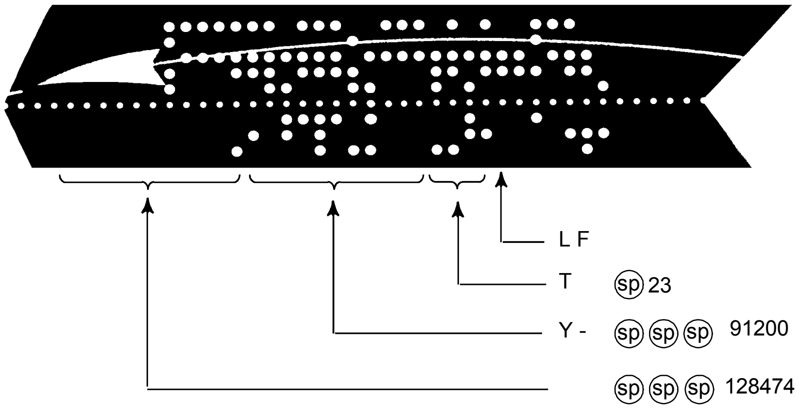
(i) For an explanation of the DPRNT command, see Items (i), (iii), and (iv) for the BPRNT command.

(ii) When outputting a variable, specify # followed by the variable number, then specify the number of digits in the integer part and the number of decimal places enclosed in brackets.
One code is output for each of the specified number of digits, starting with the highest digit. For each digit, a code is output . The decimal point is also output using a code.
Each variable must be a numeric value consisting of up to eight digits. When high-order digits are zeros, these zeros are not output if PRT (bit 7 of parameter 7000) is 1. If parameter PRT is 0, a space code is output each time a zero is encountered.
When the number of decimal places is not zero, digits in the decimal part are always output. If the number of decimal places is zero, no decimal point is output.
When PRT (bit 7 of parameter 7000) is 0, a space code is output to indicate a positive number instead of +; if parameter PRT is 1, no code is output.

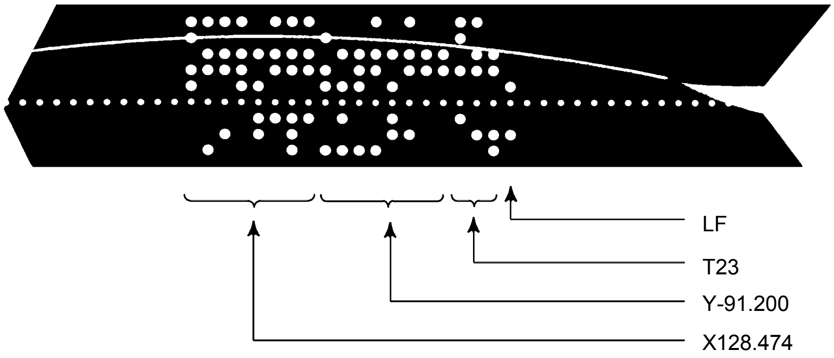
Example)

```
DPRNT [ X#2 [53] Y#5 [53] T#30 [20] ]
  Variable value
  #2=128.47398
  #5=-91.2
  #30=123.456
```

(1) Parameter PRT(No.7000#7)=0



(2) Parameter PRT(No.7000#7)=0



- Close command PCLOS

The PCLOS command releases a connection to an external input/output device. Specify this command when all data output commands have terminated. DC4 control code is output from the CNC.

- Required setting

Specify the specification number use for I/O device specification number . According to the specification of this data, set data items (such as the baud rate) for the reader/punch interface.
Never specify the output device FANUC Cassette or Floppy for punching. When specifying a DPRNT command to output data, specify whether leading zeros are output as spaces (by setting PRT (bit 7 of parameter 7000) to 1 or 0).

NOTE

1 It is not necessary to always specify the open command (POPEN), data output command (BPRNT, DPRNT), and close command (PCLOS) together. Once an open command is specified at the beginning of a program, it does not need to be specified again except after a close command was specified.

2 Be sure to specify open commands and close commands in pairs. Specify the close command at the end of the program. However, do not specify a close command if no open command has been specified.

3 When a reset operation is performed while commands are being output by a data output command, output is stopped and subsequent data is erased. Therefore, when a reset operation is performed by a code such as M30 at the end of a program that performs data output, specify a close command at the end of the program so that processing such as M30 is not performed until all data is output.

17.13 LIMITATIONS

- Sequence number search

A custom macro program cannot be searched for a sequence number.

- Single block

Even while a macro program is being executed, blocks can be stopped in the single block mode.

A block containing a macro call command (G65, G66,G66.1, or G67) does not stop even when the single block mode is on.

Blocks containing arithmetic operation commands and control commands can be stopped in single block mode by setting SBM (bit 5 of parameter 0010) to 1. Single block stop operation is used for testing custom macro programs. By setting 1 for parameter SB8 (bit 4 of No. 0010), parameter SB7 (bit 3 of No. 0010), and parameter SB9 (bit 2 of No. 2201), single-block stop can be performed for limited program numbers, using macro statements.

Note that when a single block stop occurs at a macro statement in cutter compensation mode, the statement is assumed to be a block that does not involve movement, and proper compensation cannot be performed in some cases. (Strictly speaking, the block is regarded as specifying a movement with a travel distance 0.)

- Optional block skip

A / appearing in the middle of an <expression> (enclosed in brackets [ ] on the right-hand side of an arithmetic expression) is regarded as a division operator; it is not regarded as the specifier for an optional block skip code.

- Operation in EDIT mode

By setting NE8 (bit 0 of parameter 0011) and NE9 (bit 0 of parameter 2201) to 1, deletion and editing are disabled for custom macro programs and subprograms with program numbers 8000 to 8999 and 9000 to 9999. This prevents registered custom macro programs and subprograms from being destroyed by accident. When the entire memory is cleared, the contents of memory such as custom macro programs are deleted.

- Program display in modes other than EDIT mode

By setting 1 for parameter ND8 (bit 1 of No. 0011) and parameter ND9 (bit 1 of No. 2201), custom macros or subprograms with program numbers 8000 to 8999 and 9000 to 9999 will not be subject to program display in other than EDIT mode. Thus, custom macros or subprograms called during execution are not displayed on the program screen or the program check screen.

- Reset

With a reset operation, local variables and common variables #100 to #199 are cleared to null values. They can be prevented from clearing by setting, CLV (bit 6 of parameter 7000). System variables #1000 to #1132 are not cleared.
A reset operation clears any called states of custom macro programs and subprograms, and any DO states, and returns control to the main program.

- Display of the PROGRAM RESTART

As with M98, the M and T codes used for subprogram calls are not displayed.

- Feed hold

When a feed hold is enabled during execution of a macro statement, the machine stops after execution of the macro statement. The machine also stops when a reset or alarm occurs.

- DNC operation

Control commands (such as GOTO and WHILE-DO) cannot be executed during DNC operation.

- Usable variable values

Maximum: Approx. $\pm 10^{308}$
Minimum: Approx.$\pm 10^{-308}$

- Constant values that can be used in <expression>

+0.00000000001 to +999999999999
-999999999999 to -0.00000000001
The number of significant digits is 12 (decimal). If this range is exceeded, alarm PS0012 occurs.

17.14 INTERRUPTION TYPE CUSTOM MACRO

When a program is being executed, another program can be called by inputting an interrupt signal (UINT) from the machine. This function is referred to as an interruption type custom macro function. Program an interrupt command in the following format:

Format

```
M96Pxxxxxxxx ;      Enables custom macro interrupt
M97;     Disables custom macro interrupt
```

Explanations

Use of the interruption type custom macro function allows the user to call a program during execution of an arbitrary block of another program. This allows programs to be operated to match situations which vary from time to time.

(1) When a tool abnormality is detected, processing to handle the abnormality is started by an external signal.
(2) A sequence of machining operations is interrupted by another machining operation without the cancellation of the current operation.
(3) At regular intervals, information on current machining is read. Listed above are examples like adaptive control applications of the interruption type custom macro function.

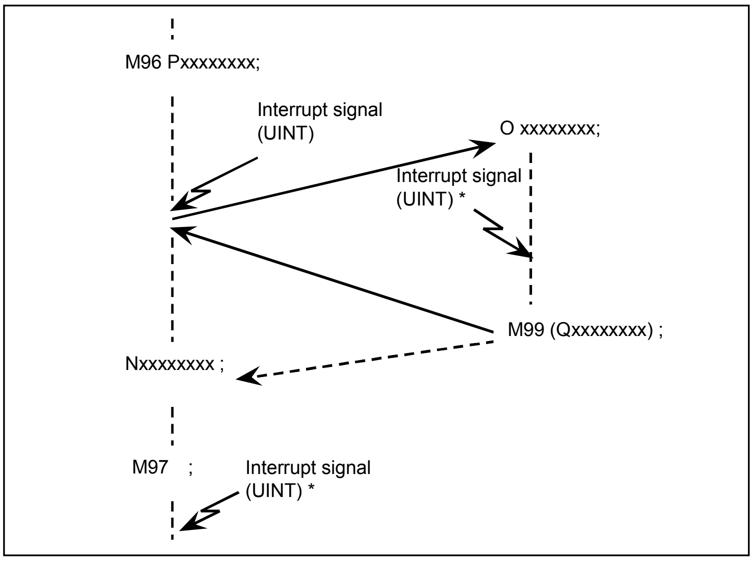


Fig.17.14 Interruption type custom macro function

When M96Pxxxx is specified in a program, subsequent program operation can be interrupted by an interrupt signal (UINT) input to execute the program specified by Pxxxx.

When the interrupt signal (UINT, marked by * in Fig. 17.14(a) is input during execution of the interrupt program or after M97 is specified, it is ignored.

17.14.1 Specification Method

Explanations

- Interrupt conditions

A custom macro interrupt is available only during program execution. It is enabled under the following conditions

- When memory operation, DNC operation, or MDI operation is selected
- When STL (start lamp) is on
- When a custom macro interrupt is not currently being processed

- Specification

Generally, the custom macro interrupt function is used by specifying M96 to enable the interrupt signal (UINT) and M97 to disable the signal. Once M96 is specified, a custom macro interrupt can be initiated by the input of the interrupt signal (UINT) until M97 is specified or the NC is reset. After M97 is specified or the NC is reset, no custom macro interrupts are initiated even when the interrupt signal (UINT) is input. The interrupt signal (UINT) is ignored until another M96 command is specified.

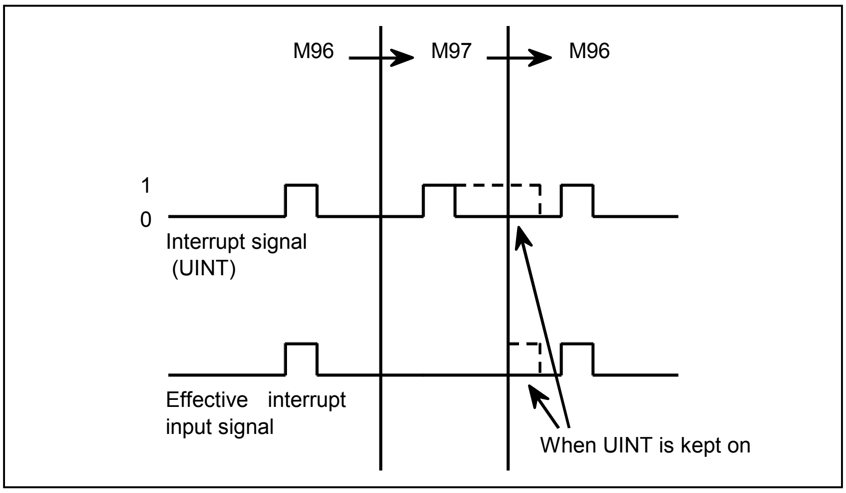


Fig. 17.14.1 M96 and M97, and Interval of the Interrupt Signal (UINT)

The interrupt signal (UINT) becomes valid after M96 is specified. Even when the signal is input in M97 mode, it is ignored. When the signal input in M97 mode is kept on until M96 is specified, a custom macro interrupt is initiated as soon as M96 is specified (only when the status-triggered scheme is employed); when the edge-triggered scheme is employed, the custom macro interrupt is not initiated even when M96 is specified.

> **NOTE**
> For the status-triggered and edge-triggered schemes, see Item "Custom macro interrupt signal (UINT)" of II-17.14.2.

17.14.2 Details of Functions

Explanations

- Subprogram-type interrupt and macro-type interrupt

There are two types of custom macro interrupts: Subprogram-type interrupts and macro-type interrupts. The interrupt type used is selected by MSB (bit 1 of parameter 7002).

(a) Subprogram-type interrupt: MSB(bit 0 of parameter 7002) is 1
An interrupt program is called as a subprogram. This means that the levels of local variables remain unchanged before and after the interrupt. This interrupt is not included in the nesting level of subprogram calls.

(b) Macro-type interrupt: MSB(bit 1 of parameter 7002) is 0
An interrupt program is called as a custom macro. This means that the levels of local variables change before and after the interrupt. The interrupt is not included in the nesting level of custom macro calls. When a subprogram call or a custom macro call is performed within the interrupt program, this call is included in the nesting level of subprogram calls or custom macro calls. Arguments cannot be passed from the current program even when the custom macro interrupt is a macro-type interrupt.

- M codes for custom macro interrupt control

In general, custom macro interrupts are controlled by M96 and M97. However, these M codes, may already being used for other purposes (such as an M function or macro M code call) by some machine tool builders. For this reason, MPR (bit 2 of parameter 7002) is provided to set M codes for custom macro interrupt control.

When specifying this parameter to use the custom macro interrupt control M codes set by parameters, set parameters 7033 and 7034 as follows: Set the M code to enable custom macro interrupts in parameter 7033, and set the M code to disable custom macro interrupts in parameter 7034. When specifying that parameter-set M codes are not used, M96 and M97 are used as the custom macro control M codes regardless of the settings of parameters 7033 and 7034.

The M codes used for custom macro interrupt control are processed internally (they are not output to external units). However, in terms of program compatibility, it is undesirable to use M codes other than M96 and M97 to control custom macro interrupts.

- Custom macro interrupts and NC statements

When performing a custom macro interrupt, the user may want to interrupt the NC statement being executed, or the user may not want to perform the interrupt until the execution of the current block is completed. MIN (bit 4 of parameter 7002)is used to select whether to perform interrupts even in the middle of a block or to wait until the end of the block.

Type I (when an interrupt is performed even in the middle of a block)

(i) When the interrupt signal (UINT) is input, any movement or dwell being performed is stopped immediately and the interrupt program is executed.

(ii) If there are NC statements in the interrupt program, the command in the interrupted block is lost and the NC statement in the interrupt program is executed. When control is returned to the interrupted program, the program is restarted from the next block after the interrupted block.

(iii) If there are no NC statements in the interrupt program, control is returned to the interrupted program by M99, then the program is restarted from the command in the interrupted block.

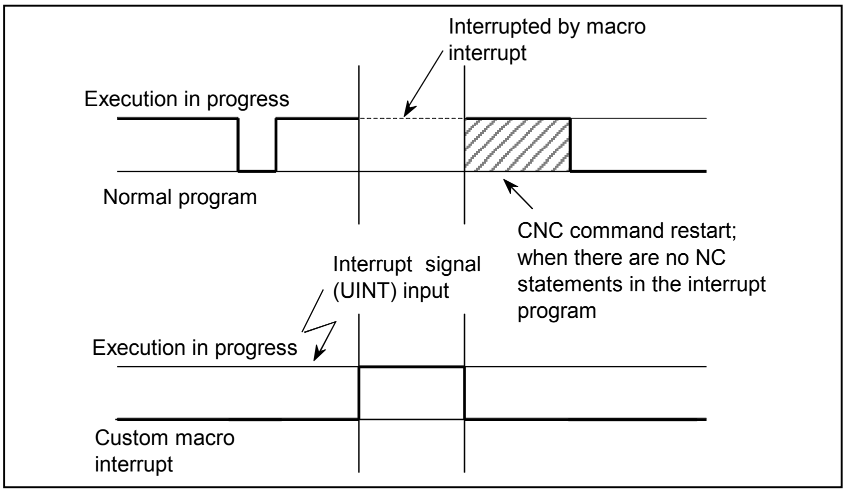

Interrupted by macro interrupt
Execution in progress
Normal program
CNC command restart; when there are no NC statements in the interrupt program
Interrupt signal (UINT) input
Execution in progress
Custom macro interrupt

Fig.17.14.2 (a) Custom macro interrupts and NC statements(type-I)

Type II (when an interrupt is performed at the end of the block)

(i) If the block being executed is not a block that consists of several cycle operations such as a drilling canned cycle and automatic reference position return (G28), an interrupt is performed as follows:
When an interrupt signal (UINT) is input, macro statements in the interrupt program are executed immediately unless an NC statement is encountered in the interrupt program. NC statements are not executed until the current block is completed.

(ii) If the block being executed consists of several cycle operations, an interrupt is performed as follows:
When the last movement in the cycle operations is started, macro statements in the interrupt program are executed unless an NC statement is encountered. NC statements are executed after all cycle operations are completed.

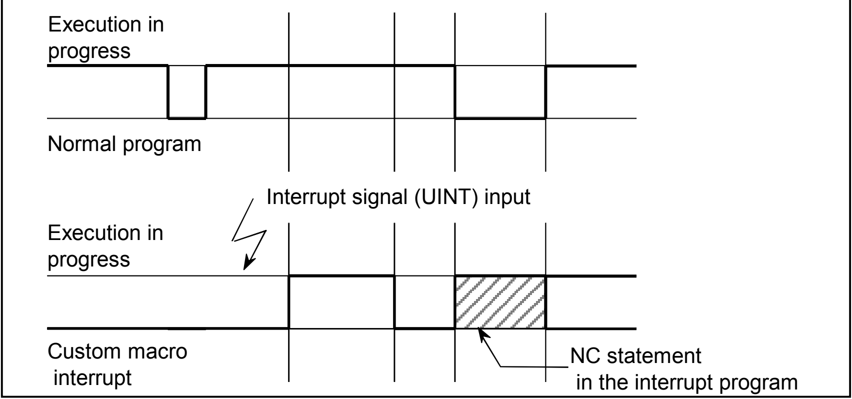


Fig.17.14.2 (b) Custom macro interrupts and NC statements(type-II)

- Conditions for enabling and disabling the custom macro interrupt signal

The interrupt signal becomes valid after execution starts of a block that contains M96 for enabling custom macro interrupts. The signal becomes invalid when execution starts of a block that contains M97. While an interrupt program is being executed, the interrupt signal becomes invalid. The signal become valid when the execution of the block that immediately follows the interrupted block in the main program is started after control returns from the interrupt program. In type I, if the interrupt program consists of only macro statements, the interrupt signal becomes valid when execution of the interrupted block is started after control returns from the interrupt program.

- Custom macro interrupt during execution of a block that involves cycle operation

For type I

Even when cycle operation is in progress, movement is interrupted, and the interrupt program is executed. If the interrupt program contains no NC statements, the cycle operation is restarted after control is returned to the interrupted program. If there are NC statements, the remaining operations in the interrupted cycle are discarded, and the next block is executed.

For type II

When the last movement of the cycle operation is started, macro statements in the interrupt program are executed unless an NC statement is encountered. NC statements are executed after cycle operation is completed.

- Custom macro interrupt signal (UINT)

There are two schemes for custom macro interrupt signal (UINT) input: The status-triggered scheme and edge- triggered scheme. When the status-triggered scheme is used, the signal is valid when it is on. When the edge triggered scheme is used, the signal becomes valid on the rising edge when it switches from off to on status.
One of the two schemes is selected with TSE (bit 3 of parameter 6003).
When the status-triggered scheme is selected by this parameter, a custom macro interrupt is generated if the interrupt signal (UINT) is on at the time the signal becomes valid. By keeping the interrupt signal (UINT) on, the interrupt program can be executed repeatedly.
When the edge-triggered scheme is selected, the interrupt signal (UINT) becomes valid only on its rising edge. Therefore, the interrupt program is executed only momentarily (in cases when the program consists of only macro statements). When the status-triggered scheme is inappropriate, or when a custom macro interrupt is to be performed just once for the entire program (in this case, the interrupt signal may be kept on), the edge-triggered scheme is useful.
Except for the specific applications mentioned above, use of either scheme results in the same effects. The time from signal input until a custom macro interrupt is executed does not vary between the two schemes.

In the Fig.17.14.2 (c), an interrupt is executed four times when the status triggered scheme is used; when the edge- triggered scheme is used, the interrupt is executed just once.

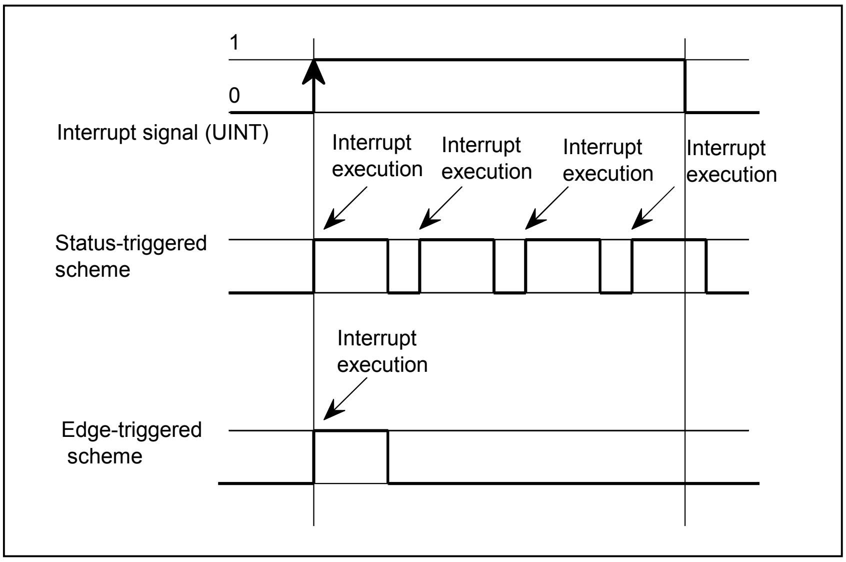


Fig.17.14.2 (c) Custom macro interrupt signal

- Return from a custom macro interrupt

To return control from a custom macro interrupt to the interrupted program, specify M99. A sequence number in the interrupted program can also be specified using address Q(or P). If this is specified, the program is searched from the beginning for the specified sequence number. Control is returned to the first sequence number found.

When a custom macro interrupt program is being executed, no interrupts are generated. To enable another interrupt, execute M99. When M99 is specified alone, it is executed before the preceding commands terminate. Therefore, a custom macro interrupt is enabled for the last command of the interrupt program. If this is inconvenient, custom macro interrupts should be controlled by specifying M96 and M97 in the program.

When a custom macro interrupt is being executed, no other custom macro interrupts are generated; when an interrupt is generated, additional interrupts are inhibited automatically. Executing M99 makes it possible for another custom macro interrupt to occur. M99 specified alone in a block is executed before the previous block terminates. In the following example, an interrupt is enabled for the Gxx block of O1234. When the signal is input, O1234 is executed again. O5678 is controlled by M96 and M97. In this case, an interrupt is not enabled for O5678 (enabled after control is returned to O1000).

NOTE

To specify the sequence number to be returned with M99, Q is used. To maintain compatibility with existing programs, however, P can be used instead. In the following explanation, Q is used.

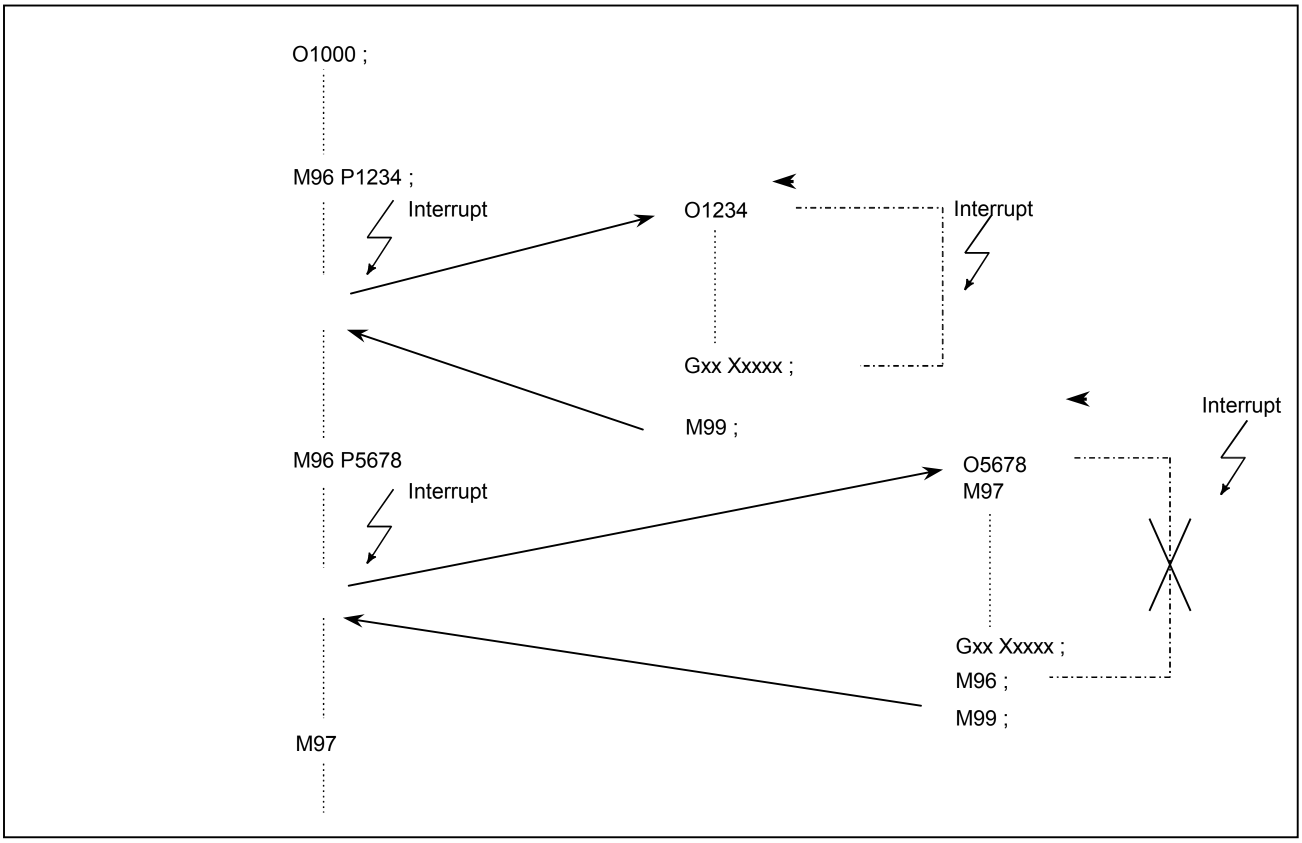


Fig.17.14.2 (d) Return from a custom macro interrupt

NOTE

When an M99 block consists only of address O, N, P, L, or M, this block is regarded as belonging to the previous block in the program. Therefore, a single-block stop does not occur for this block. In terms of programming, the following 1) and 2) are basically the same. (The difference is whether Gff is executed before M99 is recognized.)

```
1)Gxx Xyyy ;
  M99 ;
2)Gxx Xyyy M99 ;
```

- Custom macro interrupt and modal information

A custom macro interrupt is different from a normal program call. It is initiated by an interrupt signal (UINT) during program execution. In general, any modifications of modal information made by the interrupt program should not affect the interrupted program.
For this reason, even when modal information is modified by the interrupt program, the modal information before the interrupt is restored when control is returned to the interrupted program by M99.
When control is returned from the interrupt program to the interrupted program by M99 Pxxxx, modal information can again be controlled by the program. In this case, the new continuous information modified by the interrupt program is passed to the interrupted program. Restoration of the old modal information present before the interrupt is not desirable.

In this case, perform the following processing as required:

1) The interrupt program provides modal information to be used after control is returned to the interrupted program.
2) After control is returned to the interrupted program, modal information is specified again as necessary.

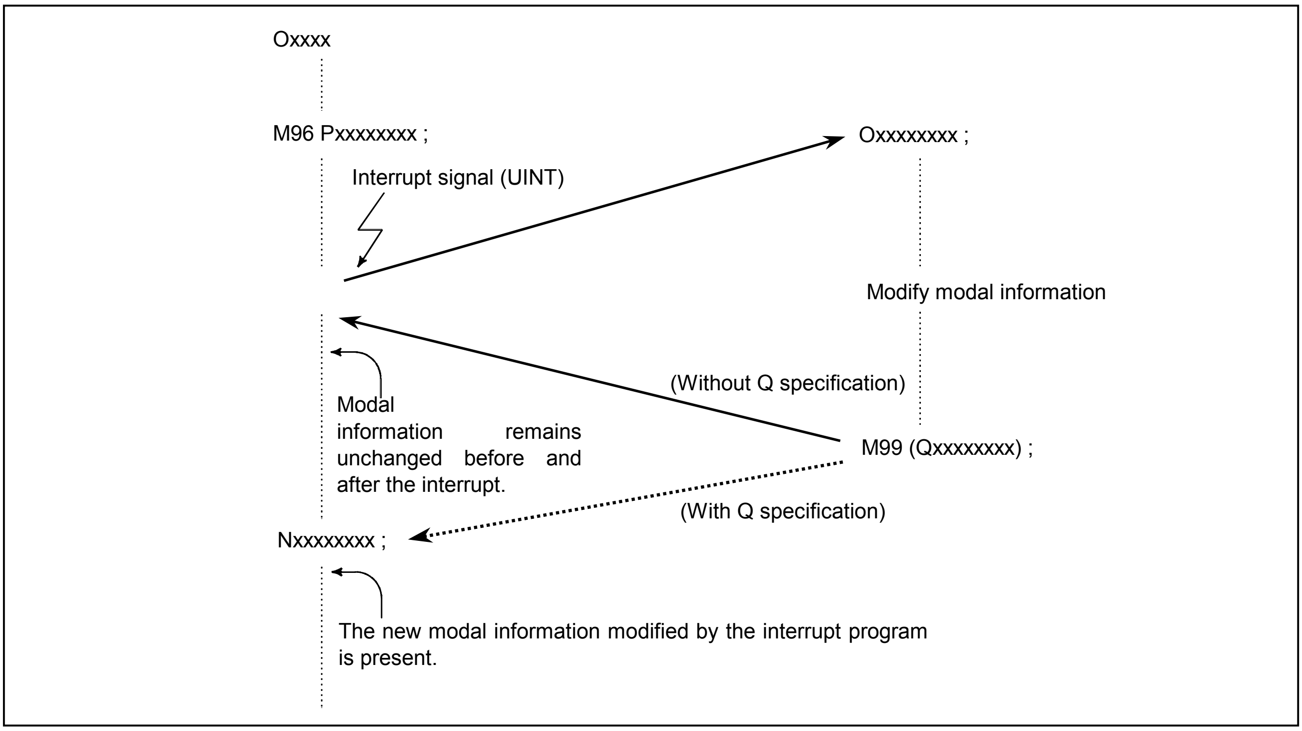

Oxxxx
M96 Pxxxxxxxx ;
Interrupt signal (UINT)
Oxxxxxxxx ;
Modify modal information
(Without Q specification)
Modal information remains unchanged before and after the interrupt.
M99 (Qxxxxxxxx) ;
(With Q specification)
Nxxxxxxxx ;
The new modal information modified by the interrupt program is present.

Fig.17.14.2 (e) Custom macro interrupt and modal information

Modal information when control is returned by M99

The modal information present before the interrupt becomes valid. The new modal information modified by the interrupt program is made invalid.

Modal information when control is returned by M99 Qxxxxxxxx

The new modal information modified by the interrupt program remains valid even after control is returned.

Modal information for blocks in which interrupts have occurred

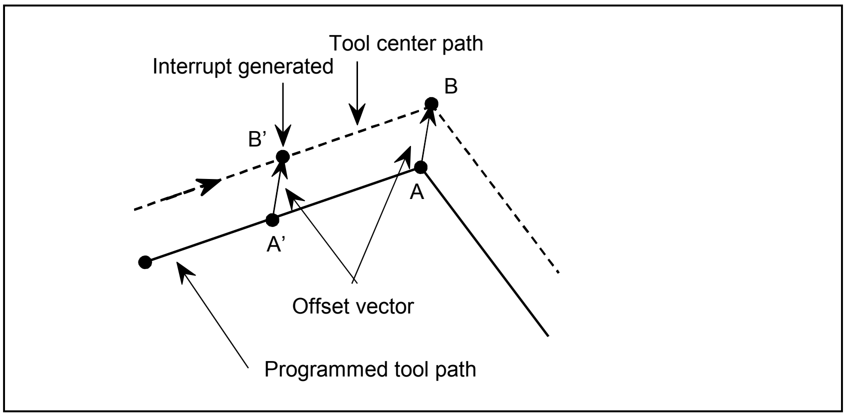
Using custom macro system variables #4401 to #4530, the modal information for the blocks in which interrupts have occurred can be read.

System variables	Modal information at the time of a custom macro interrupt
#4401	G code (group 01)
to	to
#4421	G code (group 02)
#4502	B code
#4507	D code
#4508	E code
#4509	F code
#4511	H code
#4513	M code
#4514	Sequence number
#4515	Program number
#4519	S code
#4520	T code
#4530	Additional workpiece coordinate system number

- System variable (position information values) for the interrupt program

Position information can be read as follows:

Macro variable	Condition	Position information value
#5001or later	Until the first NC statement appears.	Coordinates of point A
	After an NC statement without a move command appears.	Coordinates of point A'
	After an NC statement with a move command appears.	Coordinates of the end point specified with the move command
#5021 or later		Machine coordinates of point B'
#5041 or later		Workpiece coordinates of point B'



- Custom macro interrupt and custom macro modal call

When the interrupt signal (UINT) is input and an interrupt program is called, the custom macro modal call is canceled (G67). However, when G66 is specified in the interrupt program, the custom macro modal call becomes valid. When control is returned from the interrupt program by M99, the modal call is restored to the state it was in before the interrupt was generated. When control is returned by M99 Qxxxxxxxx;, the modal call in the interrupt program remains valid.

NOTE

In the following modes, only those programs consisting of macro programs only can generate interrupts. If programs containing NC statements attempt to generate an interrupt, alarm PS0101 is issued.
1) Canned cycle
2) Programmable mirror image
3) Coordinate system rotation
4) Scaling

Additional functions for Interruption Type Custom Macro

- Function that makes an interruption type custom macro valid when the power is turned on (if bit 2 of parameter No. 7004 is 1).
- Function that prevents an interruption type custom macro from being made invalid (if bit 2 of parameter No. 7004 is 1).
- Function that ensures that the modal information in an interruption type custom macro program is inherited even if control returns from the last M99 in that interruption type custom macro program to the original program (if bit 3 of parameter No. 7004 is 1).
- Function that, when a programmable mirror image is effective, allows execution to continue without issuing an alarm if an interruption type custom macro is called and the custom macro program contains an NC statement (if bit 4 of parameter No. 7004 is 1).

Cautions

⚠ **CAUTION**

1 If the setting of bit 3 of parameter No. 7004 and the settings of parameter No. 7102 are changed, the changes will take effect the next time the power is turned on.

2 To perform programmable mirroring using these functions, set bits 3 and 4 of parameter No. 7004 to 1.

18 HIGH-SPEED CUTTING FUNCTICNS

18.1 MULTIBUFFER (G05.1)

While executing a block, the CNC usually calculates the next block to convert it to an applicable data form for execution (executable form). This feature is called buffering. The multi-buffer function increases the number of blocks buffered to fifteen. The number of blocks can be increased to 100 with an option. Consequently, even if two or more small blocks are in succession, an interruption in the pulse distribution between blocks is prevented

Format

G05.1;	**Multibuffer made on**
G05.1 P1;	**Multibuffer made off**

Explanation

The multibuffer mode is set on and off by using single-shot G code G05.1 as follows. A block for specifying this G code must not contain any
other command.

When multi-buffer mode is set on in a block, the fifteen blocks (100 blocks when the option is used) following the block are buffered.

Whether the multibuffer mode is set on or off at power-on or immediately after a clear operation can be selected by bit 6 (MBF) in a parameter No. 2401.

- Restrictions on the multibuffer

Some G codes and M codes are not buffered even in the multibuffer mode. Thus, when such a G or M code is specified in a block, buffering for the following blocks is suppressed until execution of that block is completed. This occurs not only in the multibuffer mode, but also in normal automatic operation.
The following lists the G codes and M codes that suppress buffering.

Code	Function
G10	Data setting
G10.1	PMC data setting
G11	Data setting mode cancel
G20	Imperial input (inch)
G21	Metric input (mm)
G22	Stored stroke check on
G23	Stored stroke check off
G31	Skip function
G37.1 to G37.3	Automatic tool offset
G37	Automatic tool length measurement
G28	Reference position return
G30	Return to second, third, or fourth reference position
G53	Machine coordinate system selection

Code	Function
G52	Local coordinate system setting[*1]
M00	Program stop
M01	Optional stop
M02	End of program
M30	End of program

In addition, M codes to suppress buffering can be set with parameters. (No.2411-2418)

*1 To specify G52 as a G code that permits buffering, set O52 *bit 2 of parameter No.2409(to 1. In this case, the first move command after G52 must be specified in absolute mode.

Example

```
N1 G92 X0 Y0 G01
N2 G05.1 (multibuffer mode on)
N3 G42 G90 X1000 Y1000 F1000 D01
N4 G68 R-3000
...
N98 G05.1 P1 (multibuffer mode off)
N99 G49 G40 G90 X0 Y0 M30
```

NOTE

1 Macro statements do not count as blocks mentioned here. A macro statement specified immediately before NC statements to be converted to an executable form is executed when the CNC statements are converted to an executable form.
(Example)

```
N10 G01 X100.0 F100
N20 G90 X200.0
N21 #1 = 100.0
N22 #2 = 200.0
N30 X300.0
N31 #3 = 300.0
N40 X400.0
N50 X500.0
N60 X600.0
N61 #4 = 400.0
N70 X700.0
```

When the N10 block is being executed, blocks up to N60 have been buffered, and the N21, N22, and N031 macro statements have been executed. When execution of the N20 block starts, the N61 macro statement is executed, and N70 is buffered.

NOTE

2 If many small blocks are specified in succession, an interruption in pulse distribution may occur between blocks. Such an interruption can be prevented if the time for executing blocks read in advance is longer than the time required for advance reading of the following block.

3 The blocks mentioned above include cycles created internally in the CNC. Blocks including canned cycles and cycles created by cutter compensation are buffered.

4 Note the following when using system variables 5021 to 5055 in the multibuffer mode: When current position information is read through a system variable in the multibuffer mode, specify G53 alone in the previous block
(Example)

```
#3 = 5003
G00 X#24 Y#25
G53 (To read #5021 to #5023, stop buffering.))
G91 X[10.0 - #5021] Y[20.0 - #5022] Z[30.0 - #5023]
```

5 Specifying G53 suppresses advance reading both in the multibuffer mode and in normal operation.
Indication of a canned cycle
On the program screen or program check screen, the block being executed is normally marked by > at the left of the block. When a canned cycle with L specified is executed in the multibuffer mode, the mark > is sometimes not indicated. This only concerns the display, however, and it does not affect operation.

NOTE

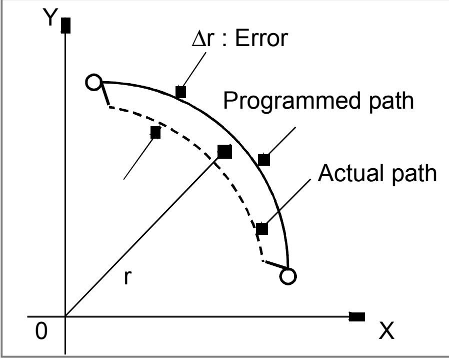
6 Processing performed at buffering
The following processes performed at buffering are also performed at buffering in the multibuffer mode
(1) Tool selection according to tool life management
(2) Input of the park signal
(Example)

```
N1 G01 G91 X100.0
N2 Y100.0
N3 X100.0
N4 Y100.0
N5 X100.0
N6 M06 T101
```

Suppose that blocks up to N6 have been buffered during execution of the N1 block. In this case, even if the end of the useful life of group 01 is reached while the N1 to N5 blocks are being executed, the next tool is not used for N6 because N6 is already buffered.

7 In the multibuffer mode as well as other modes, the single-block stop, feed hold stop, and restart operations are enabled.

18.2 DECELERATION BASED ON ACCELERATION DURING CIRCULAR INTERPOLATION



Δr : Maximum radial error (mm)
v : Feedrate (mm/sec)
r : Arc radius (mm)
a : Acceleration (mm/sec^2)
T_1 : Time constant (s) for exponential acceleration/deceleration of cutting feed(sec)
T_2 : Time constant of the servo motor (sec)

When cutting is performed at high speed for circular, helical, or spiral interpolation, the actual tool path will vary slightly from that intended.. This error in circular interpolation can be approximated by the formula given below:

$$\Delta r = \frac{1}{2}(T_1^2 + T_2^2)\frac{v^2}{r} = \frac{1}{2}(T_1^2 + T_2^2)\cdot a \text{ .... (Equation 1)}$$

When actual machining is performed, radius r of the arc to be machined and permissible error Δr are given. Then, maximum allowable acceleration a (mm/sec^2) is determined from the above expression.
The function for clamping the feedrate by the acceleration automatically clamps the feedrate of arc cutting to the value set in a parameter. This function is effective when the specified feedrate may cause the radial error for an arc with a programmed radius to exceed the permissible degree of error.
When the permissible maximum acceleration for each axis is set for a parameter, and the larger of the permissible maximum accelerations for the two axes used for circular interpolation is assumed to be A, the permissible maximum feedrate v at the radius r specified by the program is

$$v = \sqrt{A \times r} \text{ .... (Equation 2)}$$

If the specified feedrate exceeds feedrate v, determined with equation 2, the feedrate is automatically clamped to the determined feedrate.
The following parameters are used to specify the permissible maximum accelerations:
In fine HPCC mode : Parameter 1663
In a mode other than fine HPCC mode(in normal mode) : Parameter 1665

NOTE

In fine HPCC mode, an optimum feedrate that causes the accelerations on individual axes to fall within the range of permissible acceleration is calculated even if the permissible accelerations specified for the axes are different.
In a mode other than fine HPCC mode (in normal mode), the smaller of the permissible accelerations on the two axes of circular interpolation is used. If either of the values of the two interpolation axes is 0, the non-zero value is used as the permissible acceleration. If both values are 0, deceleration is not performed.

If the radius of the arc is small, the calculated deceleration speed v may become very small. To prevent the feedrate from becoming too low, the minimum feedrate can be specified in a parameter. The following parameters are used to specify the minimum feedrate:
In fine HPCC mode: Parameter 1483
In a mode other than fine HPCC mode (in normal mode): Parameter 1491

Explanations

- Linear acceleration/deceleration after interpolation in cutting feed and bell-shaped acceleration/deceleration

If the function for linear acceleration/deceleration in cutting feed is applied, an error in circular cutting can be approximated using formula 3.

$$\Delta r = (\frac{1}{24}T_1^2 + \frac{1}{2}T_2^2)\frac{v^2}{r} = (\frac{1}{12}T_1^2 + \frac{1}{2}T_2^2)\cdot a \text{ ....(Formula 3)}$$

If the function for bell-shaped acceleration/deceleration is applied, an error in circular cutting can be approximated using formula 4.

$$\Delta r = (\frac{1}{48}T_1^2 + \frac{1}{2}T_2^2)\frac{v^2}{r} = (\frac{1}{24}T_1^2 + \frac{1}{2}T_2^2)\cdot a \text{ ....(Formula 4)}$$

Because the relationship of formula 2 holds, as indicated by formula 3 and 4, deceleration by linear acceleration/deceleration after interpolation can be based on the acceleration during circular interpolation.

- Actual error

Formulas 1 and 3 provide only the theoretical approximate errors obtained by the CNC. Those values are not errors in actual machining. The error in actual machining r all is given by the following formula:
Δr all =Δr NC +Δr machine...................................(Formula 5)
Δr machine : Error caused by the machine
Δr NC : Error resulting from acceleration/deceleration time constant (T_1) and servo motor time constant (T_2)
This function keeps only the first term on the right side, which is the error resulting from the acceleration/deceleration time constant (T_1) and servo motor time constant (T_2), constant and is unrelated to the error caused by the machine.
Formulas 1 and 3 are expressions of approximations. The precision of the approximation decreases with a decrease in the radius of the arc. Even if the speed is clamped to the maximum permissible speed v obtained by formula 2, the error may not be permissible.

18.3 ADVANCED PREVIEW CONTROL(G05.1)

With the FANUC Series 15*i*, the look-ahead acceleration/deceleration before interpolation function is used for high-speed, high-precision machining, instead of advanced preview control.
The look-ahead acceleration/deceleration before interpolation function supports functions equivalent to the linear acceleration/deceleration before interpolation function and the automatic corner deceleration function supported by advanced preview control.

18.4 LOOK-AHEAD ACCELERATION/DECELERATION BEFORE INTERPOLATION (G05.1)

This function is designed to achieve high-speed, high-precision machining with a program including a combination of straight lines and arcs, like those used for parts machining. This function can suppress both servo system delay and the delay caused by acceleration/ deceleration that increase together with the feedrate.
Thus, this function can be used to ensure that the tool strictly traces the specified values, thus minimizing machining profile errors
To perform high-precision machining involving a sequence of very short straight lines and NURBS curved lines, like that used for metal die machining, the fine HPCC function is required.

Format

G05.1 Q1 :
Look-ahead acceleration/deceleration before interpolation mode on (Multi-buffer mode is set at the same time.)

G05.1 Q0 :
Look-ahead acceleration/deceleration before interpolation mode off (Multi-buffer mode remains set.)

G05.1 P1 Q0 :
Look-ahead acceleration/deceleration before interpolation mode off (Multi-buffer mode is cleared.)

Specify only G05.1 in a single block.
Look-ahead acceleration/deceleration before interpolation mode can be automatically turned on at power-up and upon a reset if the MBF bit (bit 6 of parameter 2401) is specified accordingly.

Explanation

- Functions enabled

In look-ahead acceleration/deceleration before interpolation mode, the following functions are enabled:

1) Linear acceleration/deceleration before interpolation or bell-shaped acceleration/deceleration before interpolation
2) Deceleration function based on feedrate differences at corners
3) Advanced feed-forward function
4) Nano interpolation

For details of these functions, refer to the Connection Manual (FUNCTION) (B-63783EN-1).
Dedicated parameters are supported for each function.

- Fine high precision contour control (fine HPCC)

When the fine HPCC option is selected, fine HPCC mode is also set when look-ahead acceleration/deceleration before interpolation mode is set.

- Dry run

If the dry run signal switches from 0 to 1 or 1 to 0 during movement along an axis, the feedrate is increased or decreased to a specified feedrate. It is not decreased to 0. The deceleration function based on feedrate differences at corners is enabled even in dry run mode.

- Gradual stop

If a block specifying no movement or a one-shot G code such as G04 is specified in look-ahead acceleration/deceleration before interpolation mode, a gradual stop occurs in the previous block.

- Override to a determined feedrate

The following gives the specifications of override to a feedrate determined by functions such as deceleration based on feedrate difference in look-ahead acceleration/deceleration before interpolation and deceleration based on acceleration rate in fine HPCC.

- When bit 6 (OVR) of parameter No. 1403 = 0
  The conventional specifications apply.
  Override is invalid for deceleration functions such as deceleration based on feedrate difference and deceleration based on acceleration rate.
- When bit 6 (OVR) of parameter No. 1403 = 1
  Override is valid for deceleration functions such as deceleration based on feedrate difference and deceleration based on acceleration rate.

When bit 6 (OVR) of parameter No. 1403 is 1, the following feedrates can be overridden:

- Feedrate decelerated by deceleration based on feedrate difference in look-ahead acceleration/deceleration before interpolation
- Feedrate decelerated by deceleration based on acceleration rate in fine HPCC.
- Feedrate decelerated by deceleration based on acceleration rate in circular interpolation
- Feedrate decelerated by acceleration clamp in involute interpolation
- Minimum feedrate for deceleration based on acceleration rate in fine HPCC and circular interpolation
- Maximum feedrate of fine HPCC

Even when the feedrate is overridden, the resulting feedrate does not exceed the maximum cutting feedrate (parameter No. 1422).

Limitation

- Condition for performing look-ahead acceleration/deceleration before interpolation

Even if look-ahead acceleration/deceleration before interpolation mode is specified, look-ahead acceleration/deceleration before interpolation is not performed in the following cases:

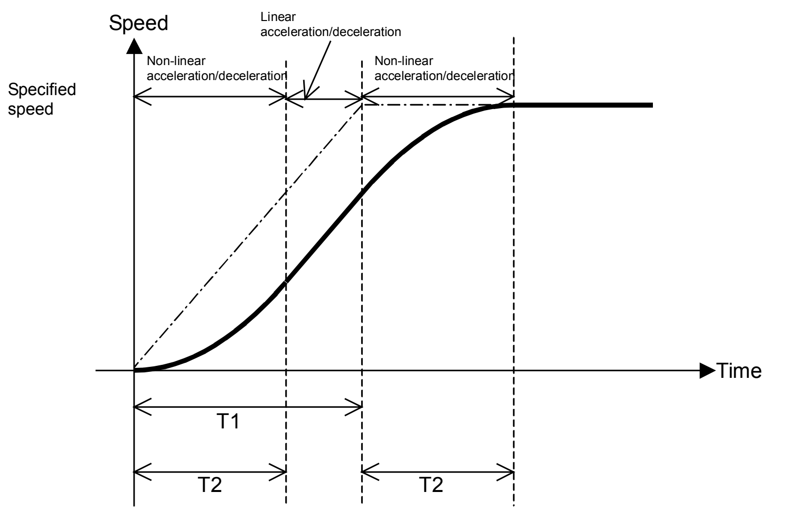
- In threading mode
- In feed per revolution mode (G95)
- In tapping mode (G63)
- During a canned cycle (other than G80)
- When no move command is specified
- During movement for spindle positioning
- When a one-shot G code other than G38, G45, G46, G47, G48, G39, or G09 is specified
- When positioning (G00, G60) is performed
- In a block in which an alarm is issued

In look-ahead acceleration/deceleration before interpolation mode, the functions listed below cannot be specified. Clear the mode before using any of these functions, then set the mode again.

- Cs contour control function
- Hypothetical axis interpolation function
- Electronic gear box function (EGB)

18.4.1 Bell-Shaped Acceleration/Deceleration Time Constant Change

In Look-ahead bell-shaped acceleration/deceleration before interpolation, the speed during acceleration/deceleration is as shown in the figure below.



T1: Time determined from the specified speed and the specified acceleration (specified speed/acceleration (parameter 1660))
T2: Acceleration change time (parameter 1656)

Fig.18.4.1(a)

The time T_1, shown above, varies with the feedrate. If the feedrate is low, the speed will be as shown below, causing linear acceleration/ deceleration not reaching the specified acceleration.
In the example below, T_2 is extended for ease of understanding, as compared with that in the above example.

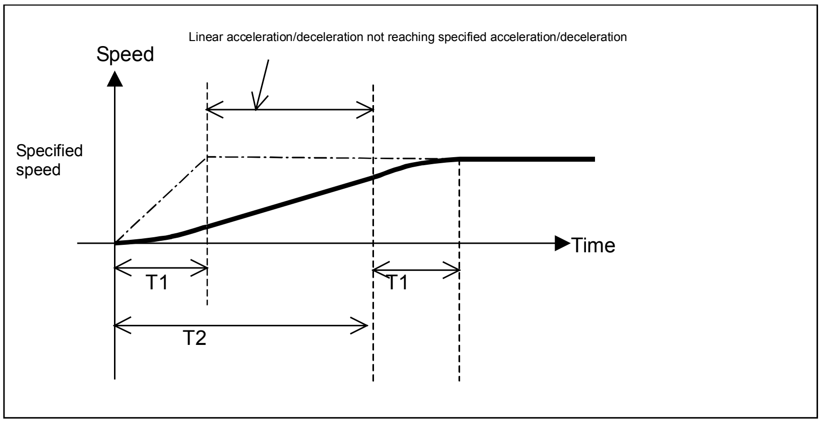


Fig.18.4.1(b)

If linear acceleration/deceleration not reaching the specified acceleration occurs as shown above, this function shortens the acceleration/deceleration time by changing the internal acceleration for acceleration/deceleration before interpolation and the bell-shaped time constant in order to generate an acceleration/deceleration pattern as close as possible to that that permits optimum bell-shaped acceleration/ deceleration before interpolation for the specified speed.
Optimum bell-shaped acceleration/deceleration before interpolation, as mentioned here, refers to bell-shaped acceleration/deceleration before interpolation in which if $T_2 > T_1$, T_1 and T_2 are changed to T_1' and T_2' as shown in the figure below so that linear acceleration/ deceleration not reaching the specified acceleration/deceleration does not occur.

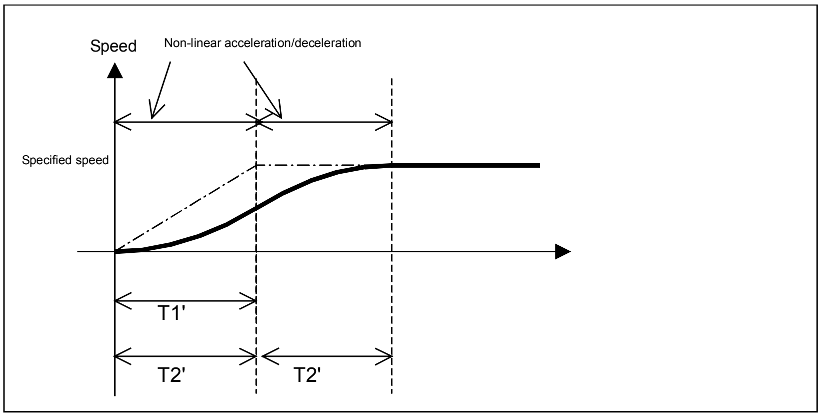


Fig.18.4.1(c)

Methods of specifying the acceleration/deceleration reference speed

The acceleration/deceleration reference speed refers to the feedrate used as the reference for calculating optimum acceleration. In Fig.18.4.1(c), it is equivalent to the specified speed used to determine T_1' and T_2'.
There are three methods for specifying the acceleration/deceleration reference speed.
Specifying a value close to the actually specified speed as the acceleration/deceleration reference speed will be effective.

(1) Specifying the speed using an F in a G05.1 Q1 block
(2) Setting the speed on the high-speed high-precision machining setting screen
(3) Setting the speed specified with the F command issued at the start of cutting as the reference speed

NOTE
This function is effective if BCG (No. 1603 bit 6) is 1.

Explanation

- Specifying the speed in a G05.1 Q1 block

If an F command is used in a G05.1Q1 block, the speed specified with the F command is assumed the acceleration/deceleration reference speed.
This reference speed is not stored in any of parameters Nos. 1473, 1539, 1559, and 1579, and cleared upon a reset. After the reference speed is cleared upon a reset or after the power is turned off and then on again, the reference speed specified for parameter No. 1473 will be used. (Method (2), described later)
If the reference speed specified for the parameter is 0, the feedrate assumed at the start of cutting will be assumed the reference speed. (Method (3), described later)

(Program example)

```
G05.1 Q1 R3 F9000 ;  Selects machining mode 3 (roughing).
      Sets the reference speed to 9000 mm/min.
G05.1 Q1 R2 F6000 ;  Selects machining mode 2 (semifinishing).
      Sets the reference speed to 6000 mm/min.
G05.1 Q1 R1 F3000 ;  Selects machining mode 1 (finishing).
      Sets the reference speed to 3000 mm/min.
G05.1 Q1 F5000 ;    Leaves the machining mode unchanged.
      Sets the reference speed to F5000 mm/min.
```

The F command used in a G05.1 block is used to specify the acceleration/deceleration reference speed, and is also used as a normal F command.

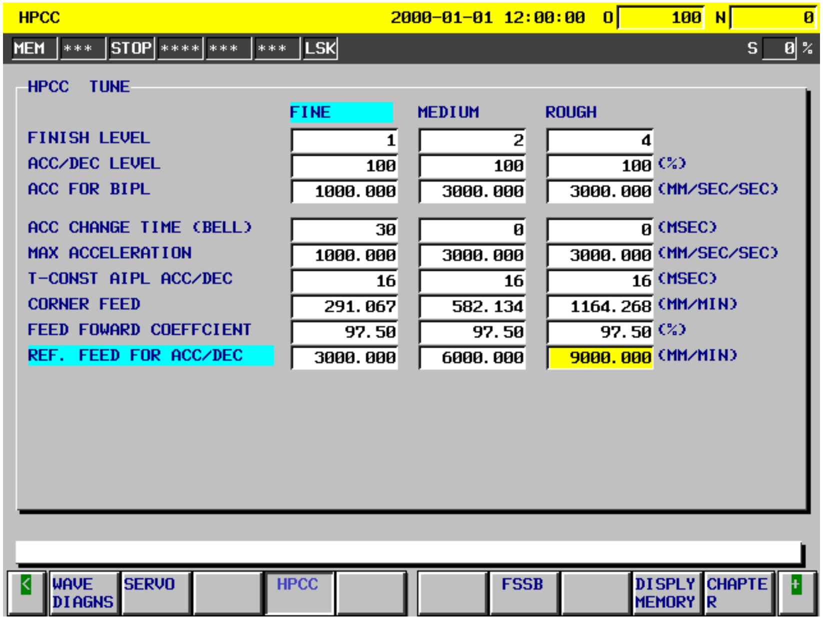
Even if the feedrate is changed during the execution of the machining program, the reference speed specified with the above command remains in effect. If this occurs, the machining time may become longer because machining is performed at the feedrate different from the reference speed.
For this reason, the reference speed to be specified with the above command should be as close as possible to the actual machining speed.

NOTE

The G05.1Q1RxFxxxx command must be issued in feed per minute (G94) mode.
If this command is issued in another mode, the speed specified with this command will not be used as the reference speed.

- Setting the speed on the High-speed High-precision Machining Setting Screen

If the reference speed for each machining mode is set on the High-speed High-precision Machining Setting Screen, the acceleration/ deceleration reference speed for the machining mode selected with the G05.1Q1Rx; command or selected by clicking the Active selection key on the MDI will be assumed the reference speed.

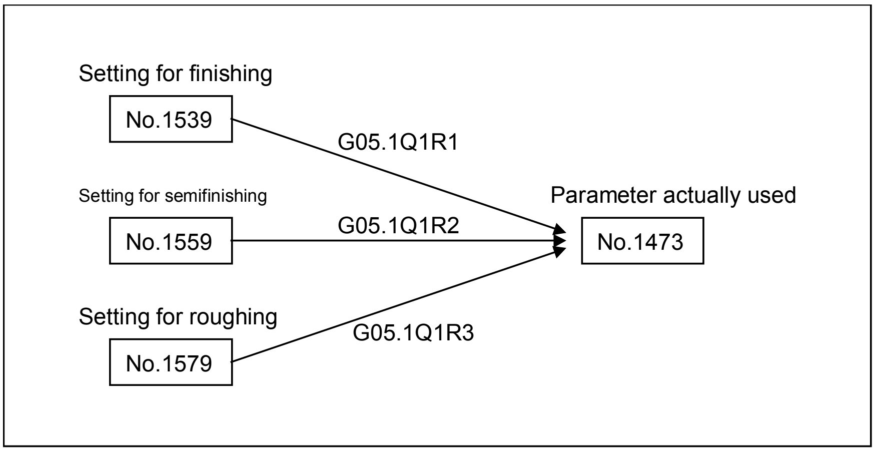


```
HPCC                                  2000-01-01 12:00:00  O    100  N     0
MEM  ***  STOP ****  ***  ***  LSK                                  S   0 %
 HPCC  TUNE
                             FINE       MEDIUM      ROUGH
 FINISH LEVEL                     1           2           4
 ACC/DEC LEVEL                  100         100         100 (%)
 ACC FOR BIPL              1000.000    3000.000    3000.000 (MM/SEC/SEC)

 ACC CHANGE TIME (BELL)          30           0           0 (MSEC)
 MAX ACCELERATION          1000.000    3000.000    3000.000 (MM/SEC/SEC)
 T-CONST AIPL ACC/DEC            16          16          16 (MSEC)
 CORNER FEED                291.067     582.134    1164.268 (MM/MIN)
 FEED FOWARD COEFFCIENT       97.50       97.50       97.50 (%)
 REF. FEED FOR ACC/DEC     3000.000    6000.000    9000.000 (MM/MIN)
```

The acceleration/deceleration reference speed set for each mode is stored in the parameter for that machining mode, and is transferred to the parameter to be actually used when:

(1) The setting for the selected mode is changed.

(2) That machining mode is selected with the G05.1Q1Rx; command or selected by clicking the Active selection key on the MDI.



The acceleration/deceleration speed is displayed only when BCG (bit 6 of parameter No. 1603) is set to 1.

- Using the speed specified with the F command issued at the start of cutting as the reference speed

The speed specified with the F command issued when a cutting block group (such as G01 and G02) starts is assumed the acceleration/deceleration reference speed, and an acceleration/deceleration pattern is generated so that optimum look-ahead bell-shaped acceleration/deceleration before interpolation is performed.

This method is used if the G05.1Q1 block does not have an F command and the reference speed set in the parameter is 0.

```
(G00)
G01 X--- Y--- Z--- F*** ;  ┐
X--- Y--- Z---             │
X--- Y--- Z---             ├  Cutting block group
X--- Y--- Z---             │
G00 ;                      ┘
```

Even if an F command is issued before a cutting block group and the F command is effective to the cutting block group modally, the speed specified with the modal F effective at the start of cutting will be assumed the reference speed.

- Examples in which three methods are used

(Example 1)

The reference speed specified for a parameter (No. 1473, 1539, 1559, or 1579) is used (the value specified for the parameter is not 0)

```
G05.1 Q1 R3 F9000 ;   Selects machining mode 3 (roughing).
                      Sets the reference speed to 9000 mm/min.
G05.1 Q1 R2 ;         Selects machining mode 2 (semifinishing).
                      The value set on the High-speed High-
                      precision Machining Setting Screen (stored in
                      parameter No. 1559) is assumed the
                      reference speed.
→ A reset is performed here.
G05.1 Q1 ;
G01 G91 X100. F2000 ; Cutting restarts.  (Machining mode:  2)
                       Because a non-zero value is set as the
                       acceleration/deceleration reference speed
                       for machining mode 2, the value set for
                       machining mode 2 is used as the reference
                       speed, not the specified speed 2000.
```

(Example 2)

The reference speed specified for a parameter (No. 1473, 1539, 1559, or 1579) is used (the value specified for the parameter is not 0)

```
 G05.1 Q1 R3 F9000 ;  Selects machining mode 3 (roughing).
                      Sets the reference speed to 9000 mm/min.
→ A reset is performed here.
 G05.1 Q1 ;
G01 G91 X100. F2000 ; Cutting restarts.  (Machining mode:  3)
                       Because a reset is performed, 9000,
                       previously specified with the F command in
                       the G05.1Q1 block, is cleared.  Because a
                       non-zero value is set as the
                       acceleration/deceleration reference speed
                       for machining mode 3, the value set for
                       machining mode 3 is used as the
                       reference mode, not the specified speed
                       2000.
```

(Example 3)

The reference speed specified for a parameter (No. 1473, 1539, 1559, or 1579) is not used (the value specified for the parameter is 0)

```
 G05.1 Q1 R3 F9000 ;  Selects machining mode 3 (roughing).
                      Sets the reference speed to 9000 mm/min.
→ A reset is performed here.
 G05.1 Q1 ;
G01 G91 X100. F2000 ; Cutting restarts.
                       Machining is performed in machining
                       mode 3.
                       Because the acceleration/deceleration
                       reference speed for machining mode 3 is
                       0, the specified speed 2000 is assumed
                       the reference speed.
```

- Acceleration/deceleration parameter calculation method

If linear acceleration/deceleration not reaching the specified acceleration occurs as shown above, this function shortens the acceleration/deceleration time by changing the internal acceleration for acceleration/deceleration before interpolation and the bell-shaped time constant in order to generate an acceleration/deceleration pattern as close as possible to that permits optimum bell-shaped acceleration/ deceleration before interpolation for the specified speed.

Optimum bell-shaped acceleration/deceleration before interpolation, as mentioned here, refers to bell-shaped acceleration/deceleration before interpolation in which linear acceleration/deceleration not reaching the specified acceleration/deceleration does not occur if $T_2 > T_1$. Calculation is performed as described below.

(1) If the bell-shaped acceleration/deceleration before interpolation time constant T_2' is calculated under the condition that the bell-shaped acceleration/deceleration before interpolation on the reference axis must not have a linear portion,

$$T_2' = \sqrt{\frac{T_2 * F}{A}}$$

where

T_2 : Acceleration change time specified for bell-shaped acceleration/deceleration before interpolation

F : Reference speed

A : Acceleration for the acceleration/deceleration before interpolation on the reference axis

(2) A proper acceleration for each axis is determined under the condition that the acceleration change on each axis must be about the same as the setting so that parameter changes do not cause considerable shock to the machine, that is:

$$\frac{\text{Acceleration for that axis after the change}}{\text{Acceleration change time after the change}} = \frac{\text{Acceleration setting for that axis}}{\text{Acceleration change time setting}}$$

The acceleration change time is regarded to be the sum of the acceleration change time of bell-shaped acceleration/deceleration and the time constant of acceleration/deceleration after interpolation, and the acceleration A'(n) for the acceleration/ deceleration before interpolation for each axis is determined as follows:

$$A'(n) = A(n) * \frac{T2' + Tc(n)}{T2 + Tc(n)}$$

where

n : Axis number

A(n) : Acceleration for the acceleration/deceleration before interpolation on each axis

Tc(n) : Time constant of acceleration/deceleration after interpolation for each axis

Acceleration/deceleration is performed using T_2' and A'(n), determined as described above.

T_1' is defined as follows:

$$T_1' = \frac{F}{A'(n)}$$

NOTE

- Depending on the value of Tc(n), the actual pattern may slightly differ from the acceleration/deceleration pattern shown in Fig.18.4.1(c).
- For an axis other than the reference axis, the actual pattern may slightly differ from the acceleration/deceleration pattern shown in Fig. 3.

18.5 FINE HPCC (G05.1)

This function is designed to achieve high-speed, high-precision machining with a program involving a sequence of very small straight lines and NURBS curved lines, like those used for metal die machining. This function can suppress the servo system delay and delay caused by acceleration/deceleration that increase as a higher feedrate is used.

Format

Thus, this function can be used to ensure that the tool strictly traces the specified values, thus minimizing machining profile errors for higher-speed, higher-precision machining.

G05.1 Q1 :
Look-ahead acceleration/deceleration before interpolation mode on (Multi-buffer mode is set at the same time.)

G05.1 Q0 :
Look-ahead acceleration/deceleration before interpolation mode off (Multi-buffer mode remains set.)

G05.1 P1 Q0 :
Look-ahead acceleration/deceleration before interpolation mode off (Multi-buffer mode is cleared.)

Specify G05.1 alone in a block.
By setting bit 6 (MBF) of parameter No. 2401, the look-ahead acceleration/deceleration before interpolation mode is set can also be turned on automatically upon power-up or a reset.
Fine HPCC mode can be turned on and off by G05P10000 and G05P0, which specify the application of high precision contour control by the 64-bit RISC processor of the FANUC Series 15-MB. For details, see "Specifying G05P10000/G05P0," below.

Explanation

- Functions enabled

In fine HPCC mode, the look-ahead acceleration/deceleration before interpolation function becomes effective. Therefore, the following functions are enabled:

1) Linear acceleration/deceleration before interpolation or bell-shaped acceleration/deceleration before interpolation
2) Deceleration function based on feedrate differences at corners
3) Advanced feed-forward function
4) Nano interpolation

With the fine HPCC function, the additional functions listed below can be used to achieve high-speed, high-precision machining for very small straight lines and NURBS curved lines:

1) Feedrate determination based on acceleration on each axis
2) Deceleration function based on Z-axis fall angle
3) Fifteen-block multi-buffer function
   (The number of blocks can be optionally increased to a maximum of 100 blocks.)

For details of these functions, refer to the Connection Manual (FUNCTION) (B-63783EN-1). Dedicated parameters are supported for each function.

- Specifying G05P10000/G05P0

Fine HPCC mode can be turned on and off by G05P10000 and G05P0, which are used to specify the application of high-precision contour control by the 64-bit RISC processor of the FANUC Series 15-MB.
Fine HPCC mode differs from the high-precision contour control applied by the 64-bit RISC processor of the FANUC Series 15-MB in the following points:

1. Restrictions on specifiable G codes and the like differ from those of high-precision contour control by RISC. See the restrictions imposed on fine HPCC and look-ahead acceleration/deceleration before interpolation for the FANUC Series 15*i*.
2. If the MSU bit (bit 1 of parameter 8403) of the FANUC Series 15-MB is set to 1 while cutter compensation is used, the cutter compensation vectors formed at the end of a block of a positioning or auxiliary function and the preceding block are the same as the cutter compensation vector formed at the end of the block preceding the two blocks.
   The FANUC Series 15*i* performs the calculation in the same way as in normal cutter compensation. Accordingly, the tool path in high-precision contour control by the RISC processor of the Series 15-MB may differ from the tool path in fine HPCC mode of the Series 15*i*.
3. In high-precision contour control by the RISC processor of the Series 15-MB, if the SG0 bit (bit 7 of parameter 8403) is set to 1, positioning in HPCC mode is always carried out by linear interpolation. With the Series 15*i*, however, the type of positioning in fine HPCC mode is determined by the LRP bit (bit 4 of parameter 1400), as in normal positioning.

NOTE

1 Always specify G05P10000 and G01P0 as a pair. Fine HPCC mode, after being turned on by G05.1Q1, cannot be turned off by G05P0. Fine HPCC mode, after being turned on by G05P10000, cannot be turned off by G05.1Q0.

2 Setting the MBF bit (bit 6 of parameter 2401) to 1 to automatically turn on HPCC mode is regarded as being equivalent to specifying G05.1Q1. To turn off fine HPCC mode in this case, specify G05.1Q0.

3 When the mode is cancelled by G05P0, the multi-buffer is left enabled.

Limitation

This function uses look-ahead acceleration/deceleration before interpolation. For details of the imposed limitations, see Section II-18.4.

18.6 MACHINING TYPE IN HPCC SCREEN PROGRAMMING (G05.1 OR G10)

General

The high-speed high-precision machining setting screen supports three machining parameter sets (FINE, MEDIUM, and ROUGH). The parameter set to use can be selected in MDI mode. Selecting one parameter set causes the working parameters to be rewritten. If FINE is selected, for example, the values of the finish machining parameters listed below are set in the working parameters.

	Finish machining parameter	Working parameter
ACC/DEC LEVEL	1533	(None)
ACC FOR BIPL	1534	1660(*1)
ACC CHANGE TIME (BELL)	1536	1656
MAX ACCELERATION	1535	1663(*2)
T-CONST AIPL ACC/DEC	1522	1635
CORNER FEED	1524	1478
FEED FORWARD COEFFICIENT	1529	1985-3344(*3)

*1,*2 These values have been overridden on the acceleration/ deceleration level.

*3 These values apply only when parameter TSP (bit 0 of parameter No. 8403) is set to 1.

This function enables a machining parameter set to be selected by a command in a program during operation as well as in MDI mode. The selected machining parameters are reflected on the high-speed high-precision machining setting screen.

Format

1. To enter multibuffer mode and select machining-specific parameters at the same time

```
        G05.1 Q1 R_ ;
              R1 : Transfers "finish machining" parameters.
              R2 : Transfers "semifinish machining" parameters.
              R3 : Transfers "rough machining" parameters.
```

2. To select machining-specific parameters

```
        G10 L80 R_ ;
              R1 : Transfers "finish machining" parameters.
              R2 : Transfers "semifinish machining" parameters.
              R3 : Transfers "rough machining" parameters.
```

To issue the G10L80R_ command, the programmable data input option is required.

- Program example

```
O12345678

G05.1 Q1 R3
               Operation is performed with "rough machining" settings.
G05.1 Q1 R2
               Operation is performed with "semifinish machining" settings.
G10 L80 R1
               Operation is performed with "finish" machining.
G05.1 Q0

M30
```

- Retaining the defaults of high-speed high precision parameters

If processing-specific parameters are selected with the G05.1 Q1 Rx or G10 L80 Rx command, execution parameters are changed so that machining can be performed with the settings for each mode. Usually, the execution parameters do not return to the previous settings when the mode is turned off. If ATDF (bit 5 of parameter 1517) is set to 1, the execution parameters return to the previous settings (defaults) after machining in each mode.

- Restoration

Execution data is restored from default data when:
1. G05.1Q1 is issued.
2. G05.1Q0 is issued.
3. The CNC is reset.

- Setting defaults

Default data is updated when execution parameters are changed with an event other than the selection of processing-specific parameters (MDI input, window input, and G10 parameter input).

- Parameter numbers

The defaults of the following parameters can be retained and restored.
1473,1478,1635,1656,1660,1663,1985

NOTE

If ATDF is set to 1 so that execution parameters are restored to their defaults at the end of machining with machining-mode-specific parameters, the machining mode cannot be changed by operating on MDI keys on the high-speed high precision machining setup screen. ("WRITE PROTECT(DATA NUMBER 1517#5 IS 1)" warning is issued.)
The G05 P10000 and G05 P0 command do not cause the parameters to be restored to their defaults.

18.7 JERK CONTROL

Overview

Look-ahead acceleration/deceleration before interpolation and fine HPCC, which are high-speed, high-precision machine functions, perform speed control in such a way that the rate of change of acceleration (jerk) will be smooth.
As a result, vibration and shocks on the machine are reduced, leading to machining at higher precision.
There are two types of jerk control functions as described below.

<1> Speed control based on changes to acceleration for each axis
Speed control for portions where there is a large change to acceleration, for example, a portion where a specified shape changes from a straight line to a curved line, is performed using look-ahead acceleration/deceleration before interpolation by obtaining a feed rate that can keep a change to acceleration for each axis from exceeding their permissible acceleration change level.
As a result, vibration in a servo section where there can be a large change to acceleration is suppressed, leading to an improved precision for cut surfaces.

<2> Look-ahead smooth bell-shaped acceleration/deceleration before interpolation
Look-ahead bell-shaped acceleration/deceleration before interpolation performs control in such a way that tangential changes to acceleration in the acceleration change time will be smooth.
As a result, ordinary bell-shaped acceleration/deceleration can become smoother, leading to a reduction in shocks to the machine at acceleration/deceleration.
If the bell-shaped acceleration/deceleration is specified as an acceleration/deceleration type for the following functions performed in acceleration/deceleration before interpolation, the same effect as stated above can be achieved because this function makes changes to acceleration smoother.

- Acceleration/deceleration before interpolation for linear rapid traverse
- Optimum-torque acceleration/deceleration

Limitations

For explanations about the restrictions of this function, see the descriptions about look-ahead acceleration/deceleration before interpolation, because the function uses it.

Reference item

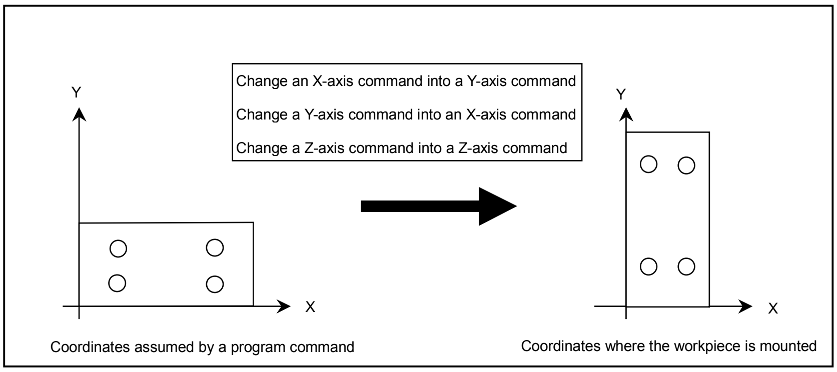
Series15*i*/150*i*-B	Connection Manual (this manual) (B-63783EN-1)	7.2.2	Look-ahead acceleration/deceleration before interpolation
FANUC Series 15*i*/150*i*-MB	Operator's Manual (Programming) (B-63784EN)	II-18.4	Look-ahead acceleration/deceleration before interpolation

19 AXIS CONTROL FUNCTIONS

19.1 AXIS INTERCHANGE

The machine axis on which the tool actually moves with the X, Y, or Z command specified by memory, DNC, or MDI operation can be changed by using the setting data (No. 1049) or the switches on the machine operator's panel.

This is useful when the coordinates assumed by a program command differ from those where the workpiece is actually mounted, as in the following example:


Y
X
Change an X-axis command into a Y-axis command
Change a Y-axis command into an X-axis command
Change a Z-axis command into a Z-axis command
Y
X
Coordinates assumed by a program command
Coordinates where the workpiece is mounted

Explanation

- Axis interchange number

This function allows six patterns of axis interchange. The table below shows the correspondence between program addresses X, Y, and Z and machine axes x, y, and z.

Axis interchange number	Program address		
	X	Y	Z
0	x	y	z
1	x	z	y
2	y	x	z
3	y	z	x
4	z	x	y
5	z	y	x

NOTE

Axis interchange number 0 means that the axis is not changed.

- Specifying axis interchange

(1) Specification with the setting data
Set the desired axis interchange number (0 to 5) for the setting data.

(2) Specification with the switches on the machine operator's panel
For an explanation of using the panel switches, refer to the manual provided by the machine tool builder.

The relationships between the specification with the setting data and that with the switches on the machine operator's panel are as given below.

	Setting data (No. 1049)	Switch on machine operator's panel	Valid specification
1	0	0	Axis interchange is disabled.
2	1 to 5	0	Specification with the setting data
3	0	1 to 5	Specification with switches on the machine operator's panel
4	1 to 5	1 to 5	Specification with switches on the machine operator's panel

When axis interchange need not be performed, set both the setting data and the switch on the machine operator's panel to "0."

- Cases in which axis interchange is disabled

For the following commands, axis interchange is disabled.

(1) Manual operation commands

(2) Move command used to move the tool to a particular machine position, command related to machine coordinates, and coordinate system setting command

(a) Automatic reference position return
G28, G30

(b) Floating reference position return
G30.1

(c) Return from the reference position
G29
By setting 1 for parameter RPC (bit 0 of No. 1001), axis interchange is enabled for the G29 command.

(d) Stored stroke limit
G22, G23

(e) Coordinate system setting
G92

(f) Offset setting
G10

(g) Dwell
G04

(h) Machine coordinate positioning
G53

Example

When axis interchange is performed, the addresses specified with a program command are changed according to the axis interchange number before the command is executed.

Example) If interchange number 4 is specified

Command block : `G00 X100.0 Y200.0 Z30.0 ;`

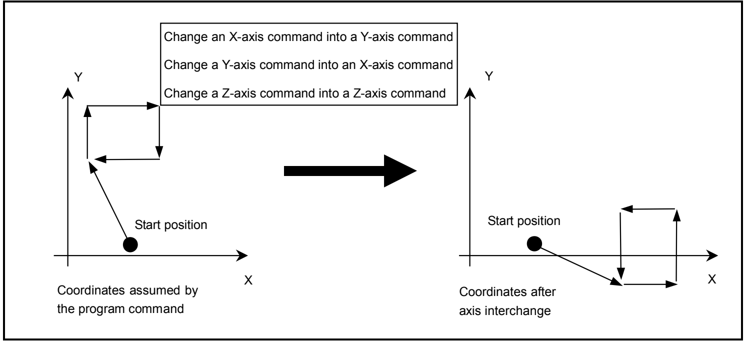
Interpretation of the command after axis interchange : `G00 X200.0 Y300.0 Z100.0 ;`

NOTE

1 If the same program is used with and without axis interchange, the amount of travel and the direction assumed with axis interchange differ from those assumed without axis interchange, depending on the start position and whether the command is an incremental or absolute command.

(Example)

Case in which the start position is the same and the direction of travel for the incremental command differs with axis interchange


Change an X-axis command into a Y-axis command
Change a Y-axis command into an X-axis command
Change a Z-axis command into a Z-axis command
Y
X
Start position
Coordinates assumed by the program command
Y
X
Start position
Coordinates after axis interchange

2 The drilling axis in a canned cycle or in tool length compensation and the axis subject to tool length compensation can be fixed to the Z-axis, using parameter LXY (bit 4 of No. 6000) and parameter FXY (bit 0 of No. 6200). Even when these axes are fixed to the Z-axis, if program address Z is changed to machine axis x or y using axis interchange, the drilling axis and the axis subject to tool length compensation are also changed to x or y.

19.2 TWIN TABLE CONTROL

Two specified axes can be switched to synchronous, independent, or normal operation, using the appropriate switches on the machine operator's panel.

The following operating modes are applicable to machines having two tables driven independently by separate control axes. The following example is of a machine with two tables driven independently by the Y axis and V axis. If the axis names and axis sets that are actually being used differ from those in the example, substitute the actual names for those below.

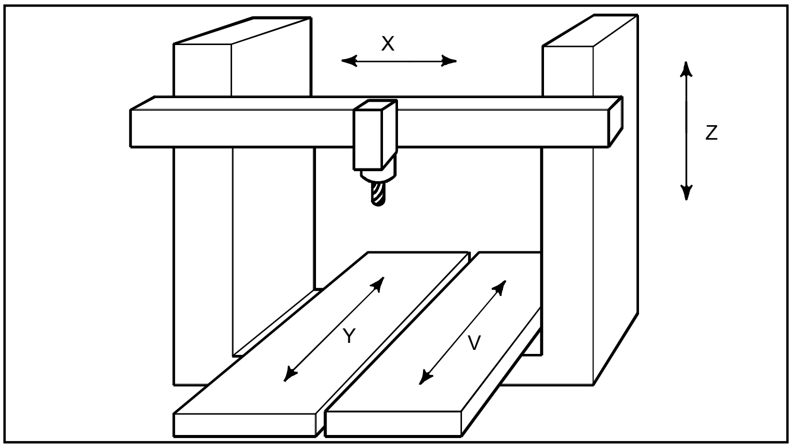


Fig.19.2 Example of axis configuration of the machine operated by Twin table control

Explanation

- Synchronous operation

This mode is used for, for example, machining large workpieces that extend over two tables.

While operating one axis with a move command, it is possible to synchronously move the other axis. In the synchronous mode, the axis to which the move command applies is called the master axis, and the axis that moves synchronously with the master axis is called the slave axis. In this example, it is assumed that Y axis is the master axis and V axis is the slave axis. Here, the Y axis and the V axis move synchronously in accordance with program command Yyyyy issued to the Y axis (master axis).

In this synchronous operation, the move command specified for the master axis is simultaneously provided for the servo motors of the master and slave axes. The system does not carry out continuous detection of the difference between the two servo motors or compensate for the difference by adjusting the servo motor of the slave axis. It does not detect the synchronization error alarm.

The synchronous operation can be executed in automatic operation, continuous manual feed, manual handle feed, and incremental feed. It cannot be executed in manual return to the reference position.

- Independent operation

This mode is used to machine a small workpiece on either of the two tables. The move command specified for the master axis can determine the movement along the master or slave axis.

(1) Command Yyyyy specified for the master determines the movement along the Y-axis.

(2) Command Yyyyy specified for the master axis determines the movement along the V-axis.

The command specified for the master axis can be used even when only the slave axis is operated. An identical command can be used, irrespective of the table on which the workpiece is placed.

Independent operation can be executed in automatic operation. In automatic operation mode, independent operation can be performed. In manual operation mode, synchronous or normal operation should be performed.

- Normal operation

This operating mode is used for machining different workpieces on each table. The operation is the same as in ordinary CNC control, where the movement of the master axis and slave axis is controlled by the independent axis address (Y and V). It is possible to issue the move commands to both the master axis and slave axis in the same block.

(1) The Y axis moves normally according to program command Yyyyy issued to the master axis.

(2) The V axis moves normally according to program command Vvvvv issued to the slave axis.

(3) The Y axis and the V axis move simultaneously according to program command YyyyyVvvvv.

Both automatic and manual operations are the same as in ordinary CNC control.

- Switching to synchronous, independent, or normal operation

For an explanation of switching to synchronous, independent, or normal operation, refer to the manual provided by the machine tool builder.

- Automatic reference position return

When the automatic reference position return command (G28) and the 2nd/3rd/4th reference position return command (G30) are issued during synchronous operation, the V axis follows the same movement as the Y axis returns to the reference position. If the V axis is positioned at the reference position after the return movement is complete, the reference position return complete signal of the V axis goes on when that of the Y axis goes on.

As a rule, commands G28 and G30 must be issued in the normal operating mode.

- Automatic reference position return check

When the automatic reference position return check command (G27) is issued during synchronous operation, the V axis and Y axis move in tandem. If both the Y axis and the V axis have reached their respective reference positions after the movement is complete, the reference position return complete signals go on. If either axis is not at the reference position, an alarm is issued. As a rule, command G27 must be issued in the normal operating mode.

- Specifying the slave axis

When a move command is issued to the slave axis during synchronous operation, an alarm (PS 0011) is issued.

- Master axis and slave axis

The axis to be used as the master axis is set in parameter No.7702.

- Display of the actual cutting feedrate on the master axis only

It is possible to hide the feedrate on any slave axis on the actual cutting feedrate display, using the switches on the machine operator's panel. Refer to the manual issued by the machine tool builder for details.

Limitation

- Coordinate system setting

When the command to specify the origin of the workpiece coordinate system (G92) or of the local coordinate system (G52) or another command which does not move the tool along an axis is specified as command Yyyyy for the master axis, the axis to which the command applies depends on the control mode. When the axes are controlled in synchronization, the command is applied to the Y-axis. When the slave axis is independently controlled, the command is applied to the V-axis.

- External deceleration, interlock, and machine lock

When the slave axis is independently controlled, the external deceleration, interlock, and machine lock signals of only the slave axis are validated. When the axes are controlled in synchronization, the identical signals of only the master axis are validated and those of the slave axis are ignored.

- Pitch error compensation

In synchronous or independent operation, pitch error or backlash compensation is performed on each of the master and slave axes, independently of the other axes, according to the settings made for that axis.

- Manual absolute

Turn on (ABS=1) the manual absolute switch during synchronous operation. If it is off, the slave axis may not move correctly.

- Synchronous deviation compensation

Synchronous deviation compensation cannot be performed. This would constantly monitor the master axis and a slave axis for any servo position deviation difference and compensate the servo motor of the slave axis to reduce the difference.

- Manual reference position return

Manual return to the reference position must be executed in the normal operation mode.

- Plane selection command

Be sure to specify a plane selection code (G17, G18 or G19) immediately after switching to synchronous operation, single operation, or normal operation.
The plane selection command is executed together with the command on the axes to constitute the plane after switching to synchronous, independent, or normal operation.

- Synchronous control

When the synchronous control is used, the twin table control can not be used.

- Tool retraction and return

When the X-, Y-, Z-, V-, and W-axes are set as the controlled axes, and a master-to-slave relationship exists between the Y- and V-axes and Z- and W-axes during twin table control, the following operations can be performed:

Twin-table mode	When retraction displacement is set in G10.6	When retraction is performed manually
Synchronization mode	G10.6 X_ Y_ Z_ Sets the retraction displacement for the master axis.	The tool can be retracted and returned for both the master and slave axes.
Slave mode	G10.6 X_ V_ W_ If the retraction displacement is set for the slave axis only, master axis setting is unnecessary(*2).	Since slave mode is not supported for manual operation in the twin table system, the same operation as that performed in normal mode takes place. (All SYNC signals are off.)
Normal mode	An axis can be specified for the retraction displacement set in G10.6.	Normal tool retraction and return is performed.

*1 Suppose that the retraction displacement in G10.6 is specified in normal mode.

*2 The axis along which the tool is to be retracted in slave mode is that axis specified in G10.6. Therefore, if G10.6 Y10. ; is specified, for example, the tool is retracted along the Y-axis. In this case, the tool is not retracted along the V-axis.

19.2.1 Tool Length Compensation in Tool Axis Direction with Twin Table Control

For a machine that applies twin table control to two heads, tool length compensation along the tool axis can be performed simultaneously for both heads (synchronous operation) or for each head (independent operation).
In synchronous operation, the compensation value calculated using the rotation axis positions of the master head is applied to the linear axes of both the master and slave heads.
In independent operation, the compensation value calculated using the rotation axis positions of the selected head is applied to the linear axes of the selected head.

- Sample machine configuration

This function is explained using the following axis configuration as an example.

Table19.2.1 (a) Sample Axis Configuration

Axis	Description
X	Axis common to heads 1 and 2
Y	Head 1 axis (master axis)
V	Head 2 axis (slave axis of the Y axis)
Z	Head 1 axis (master axis)
W	Head 2 axis (slave axis of the Z axis)
B	Head 1 axis (master axis)
J	Head 2 axis (slave axis of the B axis)
C	Head 1 axis (master axis)
K	Head 2 axis (slave axis of the C axis)

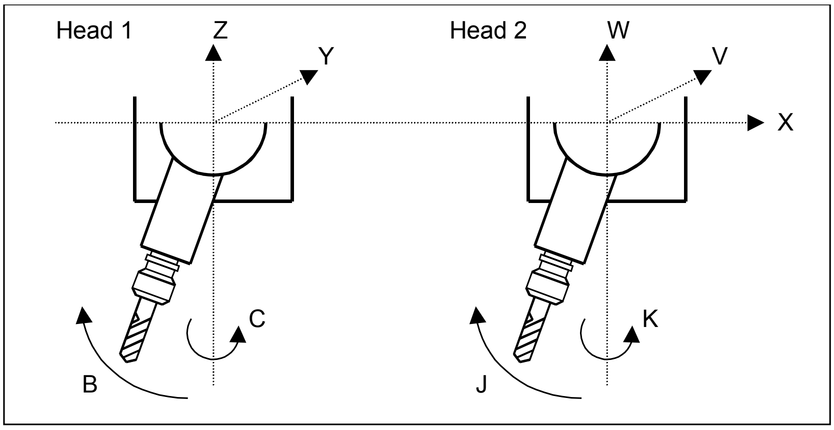


Fig. 19.2.1 Sample Machine Configuration

- Switching between synchronous and independent operation

(1) Synchronous operation

Tool length compensation along the tool axis performs simultaneously for both heads.

The compensation value calculated using the positions of the master rotation axes (B and C) is applied to the master linear axes (X, Y, and Z) and slave linear axes (V and W).

(2) Independent operation

Tool length compensation along the tool axis performs for only one head.

When head 1 is selected, the compensation value calculated using the positions of the head 1 rotation axes (B and C) is applied to the head 1 linear axes (X, Y, and Z).

When head 2 is selected, the compensation value calculated using the positions of the head 2 rotation axes (J and K) is applied to the head 2 linear axes (X, V, and W).

- Switching between synchronous and independent operation using the miscellaneous function

Specify the program command miscellaneous function in a single block to switch between synchronous and independent operation.

Set an M code to be used in the corresponding parameter, listed below.

Table 19.2.1 (b) M Code Parameter Settings

Number	Description	
7633	M code which turns synchronization off (master axis independent operation)	Mmm
7634	M code which turns synchronization on (synchronous operation)	Mpp
7642	M code which switches to slave axis independent operation	Mss

[Sample program]

```
G43.1 X_ Y_ Z_ H_ B_ C_   When no M code is specified, tool length
                          compensation is performed along the master
                          axes.
  :
Mss                       Tool length compensation is performed along the
                          slave axes.
  :
Mmm                       Tool length compensation is performed along the
                          master axes.
  :
Mpp                       Tool length compensation is performed along the
                          master and slave axes.
```

Restrictions

- Changing the tool length compensation value along the tool axis

The tool length compensation value along the tool axis can be changed for both synchronous and independent operation by three-dimensional handle interruption. In synchronous operation, the compensation value for head 1 is used as the compensation value for head 2, however.

19.3 SYNCHRONIZATION CONTROL

When one axis is driven by two servo motors as in the case of a large gantry machine, a command for one axis can drive two motors synchronously. Moreover, for synchronization error compensation, feedback information from each motor allows a positional difference (synchronization error) between the two motors to be detected. If a synchronization error beyond a set value occurs, an alarm is issued to stop movement along the axes.

The reference axis for synchronization error compensation is referred to as a master axis (M axis), while the axis to which a compensation is applied is referred to as a slave axis (S axis).

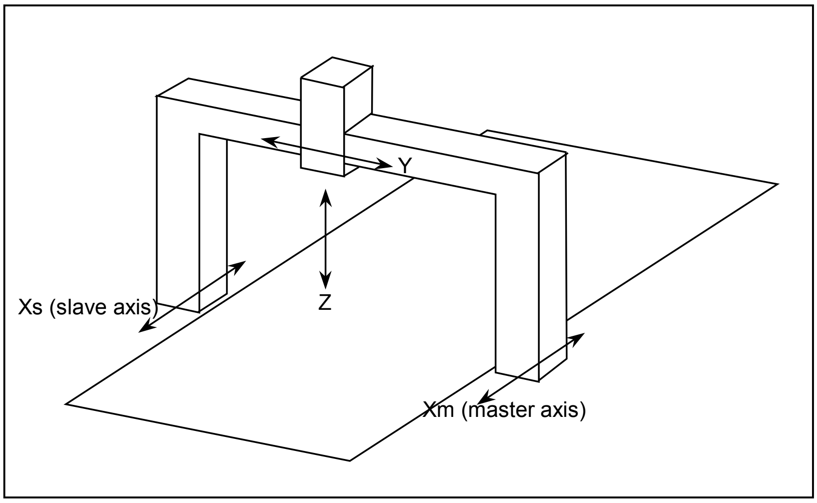


Fig. 19.3 Example Machine with Synchronization Axes Xm and Xs

19.4 TANDEM CONTROL

When enough torque for driving a large table cannot be produced by only one motor, two motors can be used for movement along a single axis.

Positioning is performed by the main motor only. The sub motor is used only to produce torque. With this tandem control function, the torque produced can be doubled.

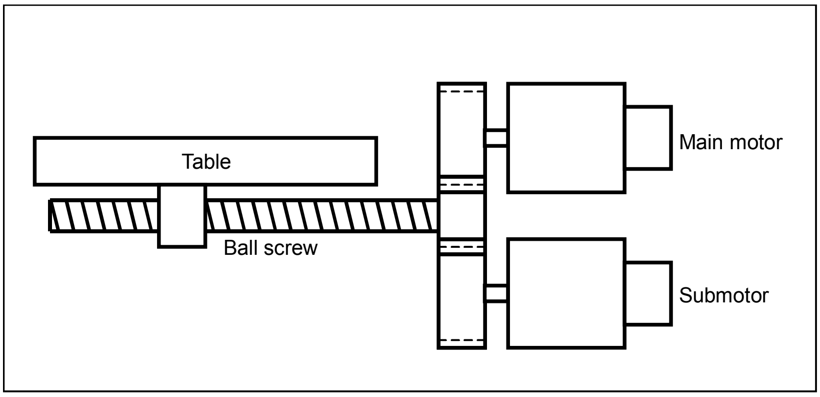


Fig.19.4 Example of operation

In general, the NC regards tandem control as being performed for one axis. However, for servo parameter management and servo alarm monitoring, tandem control is regarded as being performed for two axes.

For details, refer to the relevant manual published by the machine tool builder.

19.5 CHOPPING FUNCTION (G80,G81.1)

When contour grinding is performed, the chopping function can be used to grind the side face of a workpiece. By means of this function, while the grinding axis (the axis with the grinding wheel) is being moved vertically, a contour program can be executed to instigate movement along other axes.

The chopping function can be executed with a program command or by setting the switches on the machine operator's panel. For an explanation of chopping with the switch on the machine operator's panel, refer to the manual provided by the machine tool builder.

Format

G81.1 Z_Q_R_F_;

Z : Upper dead point
(When the axis for grinding is other than the Z-axis, specify the axis address.)

Q : Distance between the upper dead point and lower dead point
(Specify the distance as an incremental value, relative to the upper dead point.)

R : Distance from the upper dead point to point R
(Specify the distance as an incremental value, relative to the upper dead point.)

F : Feedrate during chopping
The unit for the Q and R commands is equal to the increment system of the grinding axis.
The F command must be specified without the decimal point.

G80 ; Cancels chopping

Explanation

- Basic operation

Chopping follows the steps described below:

(1) The tool is positioned at point R at the rapid traverse feedrate.
(2) The tool is moved to the bottom dead center at the feedrate specified by F.
(3) The tool is returned to the top dead center at the feedrate specified by F.
(4) The tool travels between the top dead center and bottom dead center at the chopping feedrate specified by F. The chopping feedrate can be overridden by pressing a switch on the machine operator's panel.

Chopping is continued even when the mode is switched to the manual mode or the automatic operation is halted by the feed hold function.

The chopping operation is stopped with the G80 command or a reset, after which the tool returns to point R and then stops.

If an emergency stop or servo alarm occurs, the chopping operation stops immediately.

- Chopping with the switch on the machine operator's panel

Before starting chopping, set the chopping axis, reference position, top dead point, bottom dead point, and chopping feedrate from the parameter screen.
Refer to the manual provided by the machine tool builder for details.

- Feedrate up to point R

When chopping is started, the tool moves at the rapid traverse rate (parameter No. 1420) up to point R.
The chopping feedrate override signal is enabled.
If, however, the chopping feedrate override is set to 100% or more, it is clamped to 100%.

- Feedrate from point R

When chopping is started, the tool moves at the chopping feedrate specified by the program or that specified for parameter No. 1195 from point R until chopping is canceled. If the feedrate exceeds the maximum chopping feedrate (parameter No. 1197), it is clamped to the maximum chopping feedrate. If the maximum chopping feedrate is 0, the chopping feedrate is clamped to the rapid traverse rate.
Overriding can be performed using the override switch dedicated to chopping. Refer to the manual provided by the machine tool builder for details.

- Setting chopping data

Set the following chopping data:

- Chopping axis:	Parameter No.1191
- Reference point (point R) :	Parameter No.1192
- Upper dead point:	Parameter No.1193
- Lower dead point:	Parameter No.1194
- Chopping feedrate:	Parameter No.1195
- Maximum chopping feedrate:	Parameter No.1197

- Changing chopping data

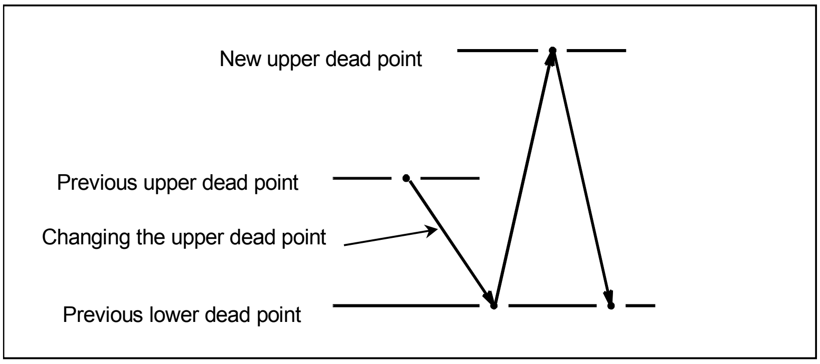
During chopping, point R, top dead point, bottom dead point, and chopping feedrate can be changed. The chopping data set for parameters is also changed.

- Chopping after the upper dead point or lower dead point has been changed

When the upper dead point or lower dead point is changed while chopping is being performed, the tool moves to the position specified by the old data. Then, chopping is continued using the new data.
When movement according to the new data starts, the chopping delay compensation function stops the chopping delay compensation for the old data, and starts the chopping delay compensation for the new data.

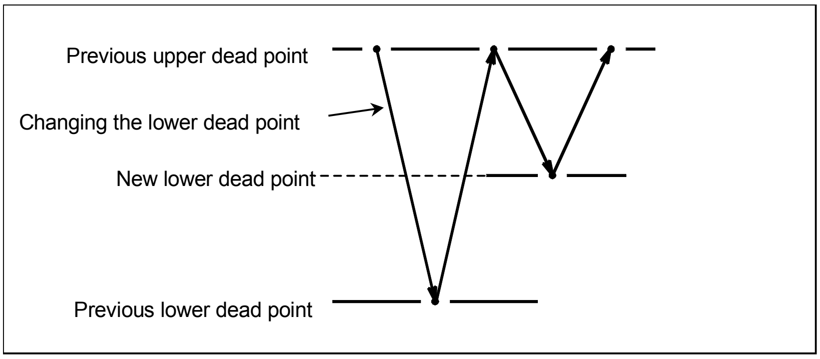
The following describes the operations performed after the data has been changed.

(1) When the upper dead point is changed during movement from the upper dead point to the lower dead point


New upper dead point
Previous upper dead point
Changing the upper dead point
Previous lower dead point

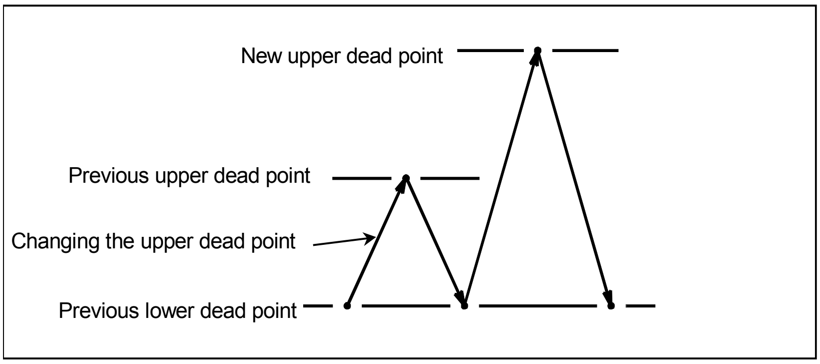
The tool first moves to the lower dead point, then to the new upper dead point.
Once movement to the lower dead point has been completed, the previous chopping delay compensation is set to 0, and chopping delay compensation is performed based on the new data.

(2) When the lower dead point is changed during movement from the upper dead point to the lower dead point


Previous upper dead point
Changing the lower dead point
New lower dead point
Previous lower dead point

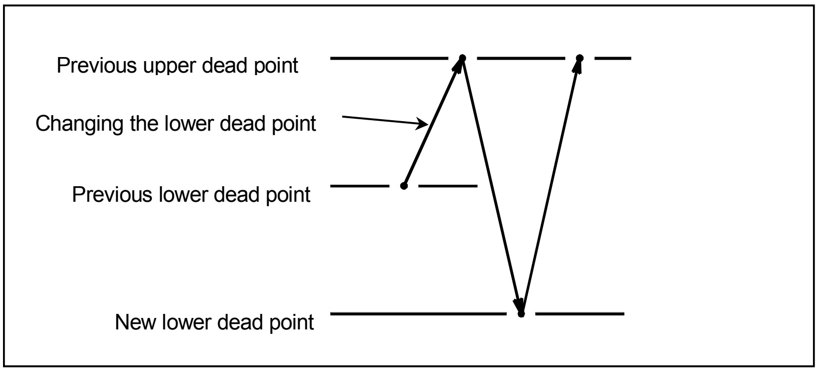
The tool first moves to the previous lower dead point, then to the upper dead point, and finally to the new lower dead point.
Once movement to the upper dead point has been completed, the previous chopping delay compensation is set to 0, and chopping delay compensation is performed based on the new data.

(3) When the upper dead point is changed during movement from the lower dead point to the upper dead point


New upper dead point
Previous upper dead point
Changing the upper dead point
Previous lower dead point

The tool first moves to the previous upper dead point, then to the lower dead point, and finally to the new upper dead point.
Once movement to the lower dead point has been completed, the previous chopping delay compensation is set to 0, and chopping delay compensation is performed based on the new data.

(4) When the lower dead point is changed during movement from the lower dead point to the upper dead point


Previous upper dead point
Changing the lower dead point
Previous lower dead point
New lower dead point

The tool first moves to the upper dead point, then to the new lower dead point.
Once movement to the upper dead point has been completed, the previous chopping delay compensation is set to 0, and chopping delay compensation is performed based on the new data.

- Chopping delay compensation function

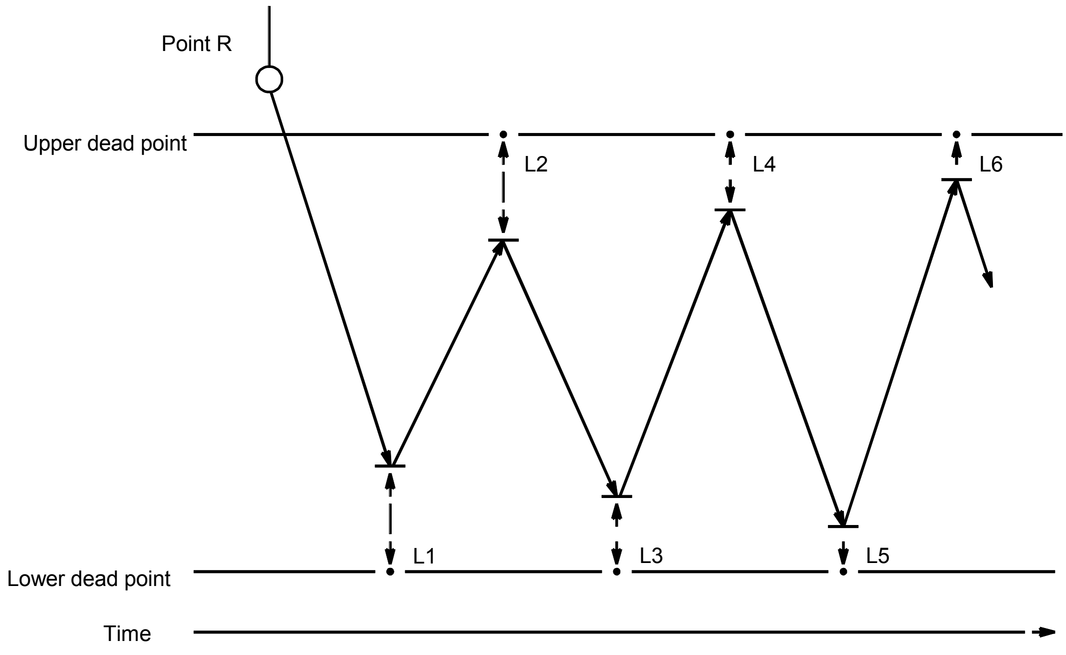
When high-speed chopping is performed with the grinding axis, a servo delay and acceleration/deceleration delay occur. These delays prevent the tool from actually reaching the specified position.

The CNC measures the difference between the specified position and the actual tool position, and performs chopping delay compensation to compensate for the difference.

To compensate for this displacement, an amount of travel equal to the distance between the upper and lower dead points, plus an appropriate compensation amount, is specified. When a chopping command is specified, the feedrate is determined so that the chopping count per unit time equals the specified count. When the difference between the displacement of the tool from the upper dead point and the displacement of the tool from the lower dead point becomes smaller than the setting of parameter No. 1881, after the start of chopping, the control unit performs compensation.

When compensation is applied, the chopping axis moves beyond the specified upper dead point and lower dead point, and the chopping feedrate increases gradually.

When the difference between the actual machine position and the specified position becomes smaller than the effective area setting (parameter No. 1827), the control unit no longer applies compensation, allowing the tool to continue moving at its current feedrate.



Displacement between the tool and the upper dead point: L2, L4, L6
Displacement between the tool and the lower dead point: L1, L3, L5

Compensation starts when:
|L3 - L2| < (parameter No.1881)
When the following condition is satisfied, compensation is no longer applied, and the tool continues to move at its current feedrate:
|L6| < effective area setting (parameter No.1827)

- Mode switching during chopping

If the mode is changed during chopping, chopping does not stop. In manual mode, the chopping axis cannot be moved manually. It can, however, be moved manually by means of the manual interrupt.

- Reset during chopping

When a reset is performed during chopping, the tool immediately moves to point R, after which chopping mode is canceled.
If an emergency stop or servo alarm occurs during chopping, mode is canceled, and the tool stops immediately.

- Stopping chopping

The following table lists the operations and commands that can be used to stop chopping, the positions at which chopping stops, and the operation performed after chopping stops:

Operation/command	Stop position	Operation after chopping stops
G80	Point R	Canceled
Chopping start switch OFF(*)	The tool moves to the lower dead point, then to point R.	Canceled
Chopping half switch ON(*)	Point R	Restart after chopping half switch goes OFF
Reset	Point R	Canceled
Emergency stop	The tool stops immediately.	Canceled
Servo alarm	The tool stops immediately	Canceled

(*) For details, see the manual provided by the machine tool builder.

- Single block signal

Even if the single block signal SBK is input during chopping, chopping continues without stopping.

Limitation

- Move command during chopping

If, during chopping, an attempt is made to execute a move command on the chopping axis, a P/S alarm (PS472) is issued.

- Canned cycle

Canned cycle commands cannot be executed during chopping.

- Cutter compensation

The chopping command (G81.1) and the cancel command (G80) suppress buffering (read-ahead). If, therefore, these commands are executed in cutter compensation mode, compensation vectors may be created differently.

- Three-dimensional coordinate conversion

This function is not effective in three-dimensional coordinate conversion mode. Before starting this function, therefore, cancel three-dimensional coordinate system conversion mode.

- PMC axis

When a chopping axis is selected as a PMC axis, chopping cannot be started.

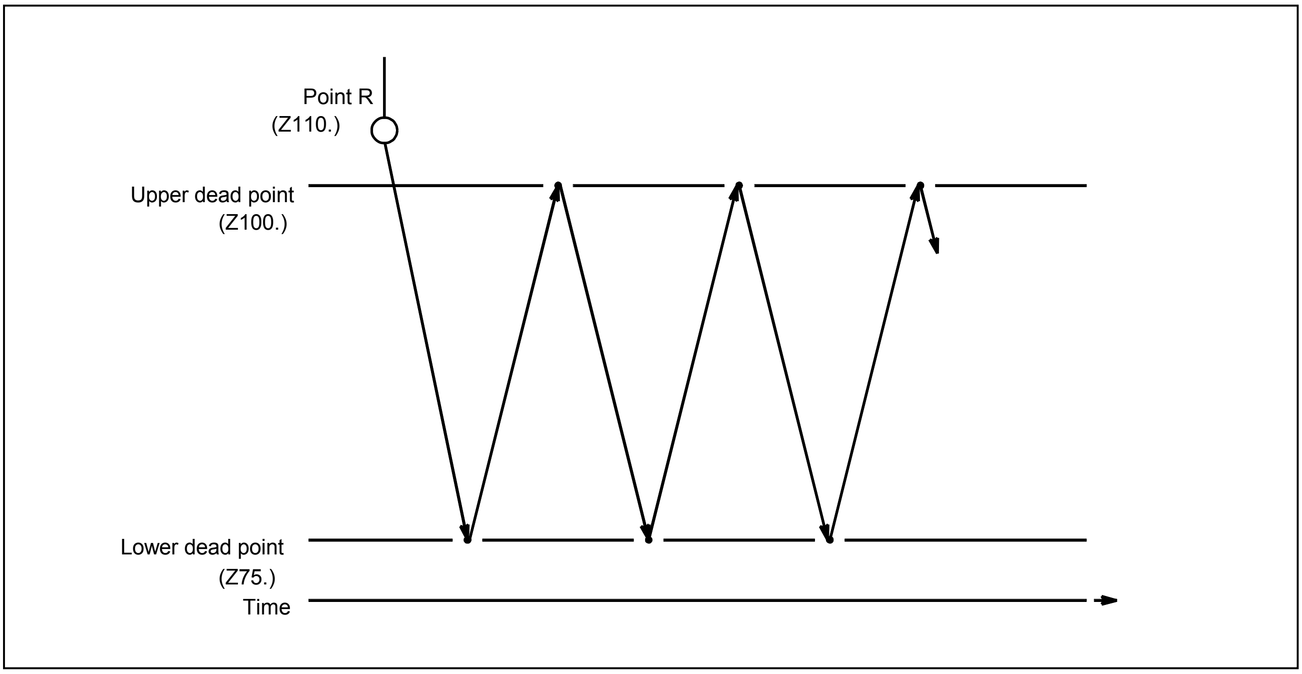
- Look-ahead acceleration/deceleration before interpolation

Look-ahead acceleration/deceleration before interpolation cannot be applied to a chopping axis.

Example

G90 G81.1 Z100. Q-25. R10. F3000 ;

- Perform rapid traverse to position the tool to Z110. (point R).
- Then, perform reciprocating movement along the Z-axis between Z100. (upper dead point) and Z75. (lower dead point) at 3000 mm/min. Chopping override is enabled.



To cancel chopping, specify the following command:
G80 ;

- The tool stops at point R.

19.6 PARALLEL AXIS CONTROL

When a machine having two or more heads or tables is used to simultaneously machine two or more identical workpieces, parallel operation is executed. In parallel operation, the move command specified for a programmed axis simultaneously controls two or more control axes having the same name.

Either the move command specified for one programmed axis or the command of one address simultaneously controls multiple axes in parallel. These axes are called parallel axes.

This function can be used when the automatic or MDI operation is executed or when the manual numerical command is specified. This function cannot be used in normal manual operation. If this function is attempted in normal manual operation, the control axes are operated independently.

In parallel operation, the control axes represented by a single programmed axis are generally operated in a specified manner. When an external signal is used, one or more of the parallel axes can be selected and operated in a different manner (parking).

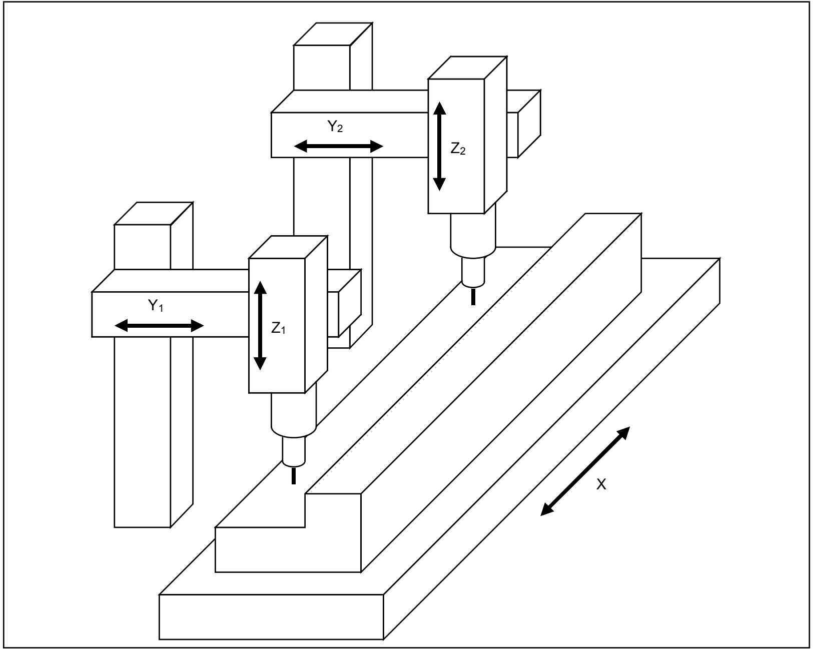


Fig.19.6 (a)

In the figure shown above, Y1 and Y2 are parallel axes and are operated in parallel by the command of a single address, address Y.

Z1 and Z2 are also parallel axes and are operated in parallel by the command of a single address, address Z.

There are two types of travel on each parallel axis. The desired type can be selected with the switches on the machine operator's panel. Refer to the manual provided by the machine tool builder for an explanation of using the panel.

- Normal (parking off) : The axis is operated as specified by the command.
- Parking (parking on) : The command is ignored and the axis is not operated.

Explanation

- Selection of the coordinate system in parallel axes

An individual offset from the workpiece reference point can be specified for each of the control axes represented by a single programmed axis. The coordinate systems of the control axes can be programmed independently.

When a machine has two heads and each head has two control axes X and Y, for example, the tools are operated as shown below.

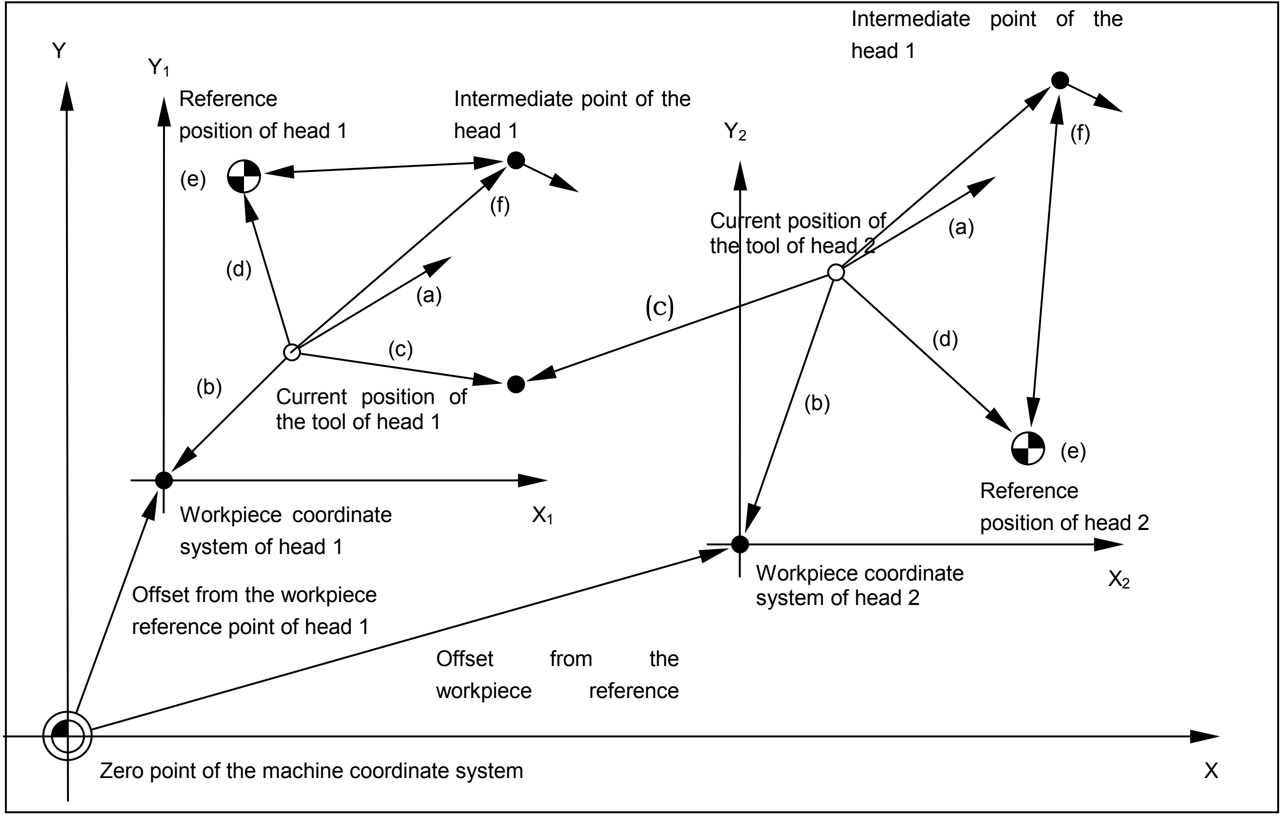

Intermediate point of the head 1
Y
Y1
Reference position of head 1
Intermediate point of the head 1
Y2
(f)
(e)
(f)
Current position of the tool of head 2
(a)
(d)
(a)
(c)
(c)
(d)
(b)
Current position of the tool of head 1
(b)
(e)
Reference position of head 2
Workpiece coordinate system of head 1
X1
Workpiece coordinate system of head 2
X2
Offset from the workpiece reference point of head 1
Offset from the workpiece reference
Zero point of the machine coordinate system
X

Fig.19.6 (b)

(a) Incremental move command
(Example)G91 X_ Y_ ;

(b) Absolute move command
(Example)G90 X0 Y0;

(c) Move command on the machine coordinate system
The tool is moved to the point determined by the axes of the machine coordinate system.
(Example)G90 G53 X_ Y_ ;

(d) Automatic return to the reference position (G28, G30)
The tools are moved to the reference positions of the control axes, which are individually specified in the parameters.
(Example)G91 G28 X0 Y0;

(e) Reference position return check(G27)
Checks whether the tool has returned to the reference position on each axis.
(Example)G91 G27 X0 Y0;

(f) Automatic return from the reference position(G29)
Positions the tool to the specified position on each axis via the mid point.
(Example)G91 G29 X30. Y50.;

- Tool length compensation and tool offset in parallel axes

Tool length compensation can be executed for tools of individual axes when an H code number and the difference between the H code number and the corresponding offset data number, or a bias value, are specified for each axis in the parameter. In the same manner, tool offset can be executed.
Set the bias for tool length compensation for parameter No. 6021.
Set the bias for tool position offset for parameter No. 6020.

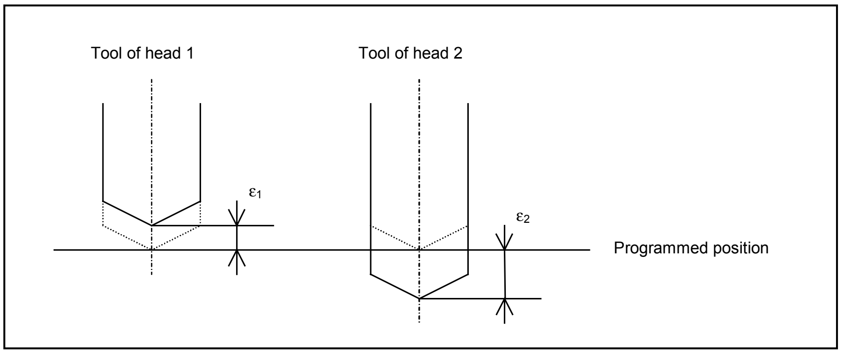


Fig.19.6 (c)

(Example)

Head	Specified offset number	Bias value	Offset data number used	Offset value
Head1	07	10	17	ε_1
Head2	07	20	27	ε_2

For example, when the third and fourth axes are called as Z1 and Z2 and handled as parallel axes, specify 10 as the bias value of the Z1-axis and 20 as that of the Z2-axis in parameter No. 6021. Then enter G43H07; as the command of tool length compensation. Tool length compensation for the tool of head 1 is executed with the offset value obtained from offset data number 17 (07 + 10). The tool length compensation for the tool of head 2 is executed with the offset value obtained from offset data number 27 (07 + 20).

NOTE
The offset value corresponding to offset number 00, or H00, is always 0 and does not depend on the bias value.

- Amounts of travel on parallel axes

The amounts of travel on parallel axes differ depending on whether the command is incremental or absolute.

(1) For an incremental command

- Rapid traverse and linear interpolation
  The amounts of travel on all parallel axes are the same.
- Circular interpolation and helical interpolation
  The amounts of travel on all parallel axes are the same. Thus, interpolation is performed on the same, multiple arcs at the same time.

(2) For an absolute command

- Rapid traverse and linear interpolation
  The absolute coordinates of the end points on all parallel axes are the same.
  Thus, if the start positions differ, the amounts of travel on the parallel axes differ.
- Circular interpolation and helical interpolation
  Interpolation is performed using the data for the controlled axis having the smallest number that is not parked. The same data as that resulting from interpolation is output for the other parallel axes.
  Thus, even for an absolute command, if the start positions on the other axes differ from that on the axis for which interpolation is performed, the tool moves to positions other than the specified end points.

- Calculation of the feedrate

In linear interpolation, the feedrate of parallel operation is calculated from the data of the axis along which the largest movement is made of all control axes. The data of the other axes is not reflected in the calculation of the feedrate.

(Example)

For parallel X axes X_1 and X_2,
Start positions
X_1 : 0.0
X_2 : 5.0
Y : 0.0
Command
G01 G90 X10. Y20. F500

The amount of travel on X_1 is 10.0 and that on X_2 is 5.0. Thus, using the greater amount of travel on X_1, the feedrate is calculated as follows:
Let

$$L = \sqrt{10*10+20*20}$$

then the feedrate on each axis is:
X_1:500 * 10 / L
X_2:500 * 5 / L
Y :500 * 20 / L

Limitation

- Synchronous control and twin table control

Of the parallel axes with the same axis name, that having the smallest controlled axis number is called the master axis.

Axes other than the master axis are called slave axes.

When synchronous control and twin table control are applied at the same time, the following restrictions are imposed:

(1) Synchronous control and twin table control cannot be used for slave parallel axes.

(2) A slave axis subject to synchronous control and twin table control cannot be used as the master parallel axis.

- Manual operation

Even for parallel axes, manual operation is performed on each of the axes independently of the others.

The manual numeric command, however, performs parallel operation in the same way as in automatic operation.

- Cutter compensation

Cutter compensation cannot be applied to a single parallel axis independently of the others. Compensation is applied to all parallel axes in the same way.

NOTE

1 The independent signals for each axis (such as over travel and interlock signals) are effective for that axis only, even if the axis is a parallel axis.

2 The following commands are not effective for parked axes:
(Example)
G92, G52, G53, G92.1, G10
Coordinate origin/preset operations on the position display screen

3 To perform circular or cutter compensation, set the basic coordinate system for each axis, using parameter No. 1022. Set the same basic coordinate system for parallel axes.
(Example)
For parallel X axes X_1 and X_2, set 1 for parameter No. 1022 for both axes.

19.7 ROTARY AXIS ROLL-OVER

The roll-over function prevents the coordinate of the rotary axis from overflowing because it converts the coordinate to a rotation angle of less than 360 degrees.
The roll-over function is enabled by setting bit 2 of parameter RDA 1009 to 1.
When the rotation axis rollover function is used, the absolute coordinates are rolled over by the amount of travel per rotation that is set for parameter No. 1260.

Explanation

In the incremental mode, a specified value directly indicates an angular displacement. In the absolute mode, the specified value is converted to the remainder obtained by dividing the specified value by 360 degrees. The difference between the converted value and the current value indicates the angular displacement. The movement by angular displacement is always made in the shorter direction. That is, if the difference between the converted value and the current value is greater than 180 degrees, the movement to the specified position is made in the opposite direction. If the difference is 180 degrees, the movement is made in the normal direction.

Example

When the roll-over function is used for rotation axis A with the following program, the following movements are made about the rotation axis:

```
G90 A0 ;
N1 G90 A-150.0 ;
N2 G90 A540.0 ;
N3 G90 A-620.0 ;
N4 G91 A380.0 ;
N5 G91 A-840.0 ;
```

Sequence number	Actual movement value	Absolute coordinate value after movement end
N1	-150	210
N2	-30	180
N3	-80	100
N4	+380	120
N5	-840	0

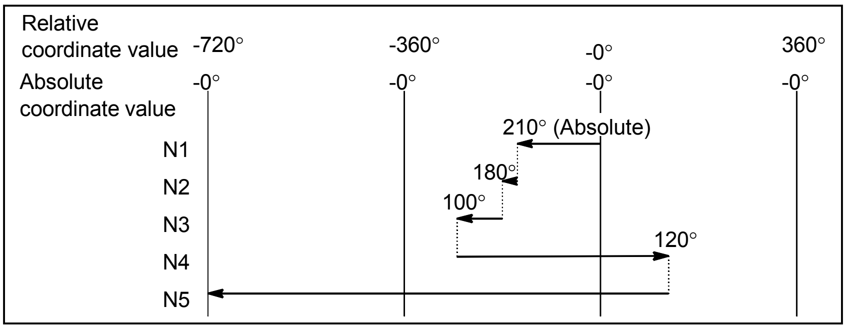


Fig.19.7 (a)

- Example of rollover with a manual intervention amount

When this function is used in the absolute mode, if the manual absolute switch is turned on to make a manual intervention during automatic operation the manual intervention is converted to the remainder obtained by dividing the intervention by 360 degrees. Then, a movement is made in the shorter direction.

[Example]
G90 A0 ;
G90 A180. ; - - - After a manual intervention of 700 degrees is made, the movement is restarted.

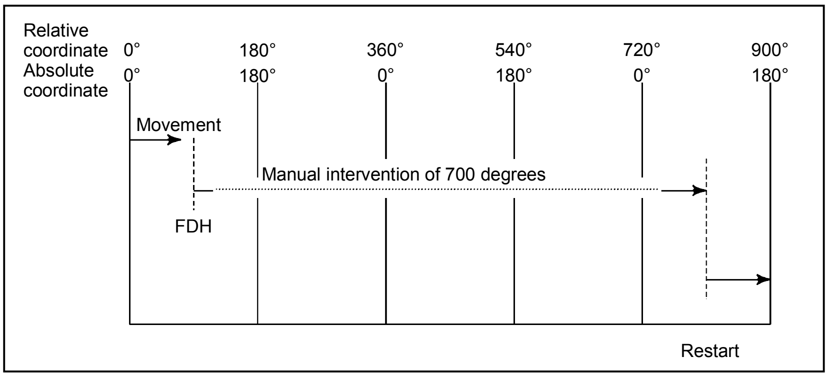


Fig.19.7 (b)

NOTE
1 This function is effective for a rotation axis only.
2 When this function is executed, the index table indexing function or rotary control function cannot be used.

19.8 MULTIPLE ROTARY CONTROL AXIS FUNCTION

Explanation

A rotary axis is specified in the ROT bit (bit 1 of parameter 1008). When incremental programming is specified for the rotary axis, a specified value directly determines the travel distance. When absolute programming is selected, either of the following two operations can be specified by the RSR bit (bit 2 of parameter 1007).

(1) The NC unit rounds down the value specified in the absolute command (G90) to a value indicating a full turn or less. The difference between the rounded value and the current position is the angular displacement. When the bit 5(INC) in parameter No. 1007 is specified, short-cut control can be executed. When the difference between the rounded value and the current position indicates more than a half turn, the short-cut control function allows reverse rotation up to the specified position.

(2) The sign of the specified value indicates the direction of rotation. (The plus sign (+) indicates counterclockwise rotation and the minus sign (-) clockwise rotation). The absolute value of the specified value indicates the destination.

- Position display

An absolute value of a position relative to the rotary axis is represented by an equivalent value within the amount of travel for a single revolution as specified in parameter 1260.

Example

When the rotary axis is called the B-axis, the following sample programs control the rotary axis as shown below:

(1) When both the RSR bit (bit 2 of parameter 1007) and the INC bit (bit 5 of parameter 1007) are set to 0
The NC internally converts the specified value to an equivalent value within a single revolution. The difference between the converted value and the value of the current position is regarded as being the amount of travel.
[Example]

G90B0 ;	Movement to the 0-degree position
G90B380. ;	Rotation by 20 degrees in the positive direction. The destination is the 20-degree position.
G90B-90. ;	Rotation by 250 degrees in the positive direction. The destination is the 270-degree position.
G90B60. ;	Rotation by 210 degrees in the negative direction. The destination is the 60-degree position.

(2) When the RSR bit (bit 2 of parameter 1007) is set to 0 and the INC bit (bit 5 of parameter 1007) is set to 1
The shortest way to make the movement of (1) is selected. [Example]

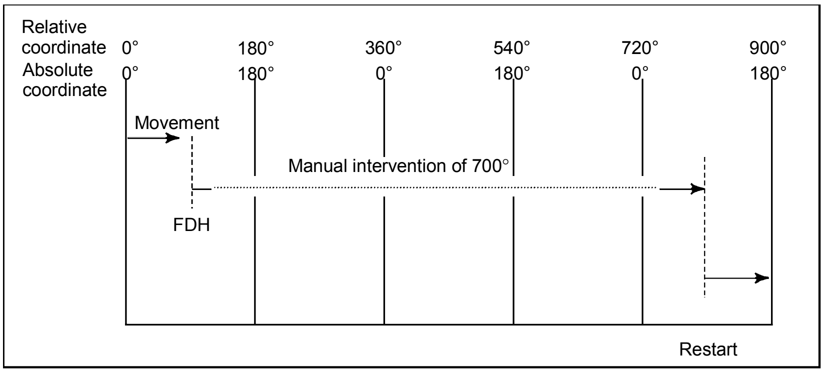
G90B0 ;	Movement to the 0-degree position
G90B380. ;	Rotation by 20 degrees in the positive direction. The destination is the 20-degree position.
G90B-90. ;	Rotation by 110 degrees in the negative direction. The destination is the 270-degree position.
G90B60. ;	Rotation by 150 degrees in the positive direction. The destination is the 60-degree position.

(3) When the RSR bit (bit 2 of parameter 1007) is set to 1 and the INC bit (bit 5 of parameter 1007) is set to 0
The sign added to the specified value determines the direction of rotation. The absolute value of the specified value is converted to an equivalent value within a single rotation. The movement is made to the position determined from the converted value.
[Example]

G90B0 ;	Movement to the 0-degree position
G90B380. ;	Rotation by 20 degrees in the positive direction. The destination is the 20-degree position.
G90B-90. ;	Rotation by 290 degrees in the negative direction. The destination is the 90-degree position.
G90B60. ;	Rotation by 330 degrees in the positive direction. The destination is the 60-degree position.

- Manual intervention

If the manual absolute switch is set on to carry out manual intervention while this function is executed in automatic operation in the absolute mode, the amount of manual intervention is rounded down to a value indicating a full turn or less and the manual intervention is offset according to the rounded value.
(Example)
G90 B0 ;
G90 B180. ; .... Manual intervention of 700 degrees is carried out during traveling, then operation is resumed.



NOTE
This function is effective only for a rotary axis.

19.9 ELECTRONIC GEAR BOX (G80, G81, G80.5, G81.5)

The Electronic Gear Box is a function for rotating a workpiece in sync with a rotating tool, or to move a tool in sync with a rotating workpiece. With this function, the high-precision machining of gears, threads, and the like can be implemented. A desired synchronization ratio can be programmed. Up to two sets of axes can be synchronized. A gear grinding machine can be controlled, for instance, by using one axis for rotating the workpiece in sync with the tool and another axis for performing dressing in sync with the tool.

The specification method differs depending on the configuration of the machine. For details, refer to the manual supplied by the machine tool builder.

The Electronic Gear Box is hereafter referred to as the EGB function.

19.9.1 Command Specification (G80.5, G81.5)

Format

$$\textbf{G81.5}\quad \begin{Bmatrix} \text{T}\,t \\ \text{P}\,p \end{Bmatrix}\quad \begin{Bmatrix} \beta j \\ \beta 0\ \text{L}l \end{Bmatrix}\ ;$$ Synchronization started

Amount of travel relative to the master axis — Amount of travel relative to the slave axis

G80.5 β0 ; Synchronization canceled

Explanation

- Master axis and slave axis

The reference axis of synchronization is referred to as the master axis, while the axis of synchronized movement is referred to as the slave axis. Taking a hobbing machine, which moves a workpiece in sync with the rotating tool, as an example, the tool axis is the master axis and the workpiece axis is the slave axis.

The master axis and slave axis of a machine depend on the configuration of the machine. For details, refer to the manual supplied by the machine tool builder.

- Start of synchronization

Synchronization is started by specifying the ratio of the amount of slave axis movement to the amount of master axis movement.

The amount of master axis movement is to be specified using one of the methods described below.

1. Master axis speed
   Tt: Specify the master axis speed in t(1 ≤ t ≤ 1000)
2. Number of pulses for the master axis
   P p : Specify the number of pulses for the master axis in p
   (1 ≤ p ≤ 999999999)
   Specify this number with four pulses equaling one A/B phase cycle.

The amount of slave axis movement is to be specified using one of the methods described below.

1. Amount of slave axis movement
   βj : Specify the address of the slave axis in β.
   Specify the amount of slave axis movement in i with the least command increment. (The range of valid settings is the same as for movement along an ordinary axis.)
   When j = 0 is specified, use of method (2) below is assumed In this case, an alarm is issued if L is not specified.
2. Slave axis speed
   β0 L±*l* : Specify the address of the slave axis in β.
   Specify the slave axis speed in *l*(1 ≤ *l* ≤ 21)

> ⚠ **CAUTION**
>
> 1 During synchronization, movements for the slave axis and other axes can be specified by programming. Note, however, that a move command must be specified in incremental mode.
>
> 2 G27, G28, G29, G30, G30.1, G33, or G53 cannot be specified for the slave axis in synchronization mode.
>
> 3 The controlled axis cancel function cannot be used for the master axis or slave axis.

> **NOTE**
>
> 1 During synchronization, the slave axis and other axes can be subjected to manual handle interruption.
>
> 2 The maximum feedrates along the master axis and slave axis depend on the position detector used.
>
> 3 In the synchronization mode, the inch/metric conversion commands (G20, G21, G70, and G71) cannot be specified.
>
> 4 In the synchronization mode, only the machine position relative to the slave axis is updated. For the absolute position referred to the slave axis, the amount of synchronized movement is converted to an equivalent value within 360 and added when synchronization is canceled.

- End of synchronization

1. Axis-by-axis cancellation of synchronization using a command
   The command "G80.5 β0" cancels synchronization.
   In β, specify the address of a slave axis. The synchronization of the slave axis specified in β is canceled. One block allows the synchronization of only one axis to be canceled. When β0 is not specified, the synchronization of all synchronized axes is canceled. The absolute coordinates are updated according to the amount of synchronous movement. For a rotation axis, the amount of synchronous movement (modulus with respect to 360 degrees) is added to the absolute coordinates.
2. Cancellation of synchronization by reset
   The synchronization of an axis is canceled by a reset when the RSH bit of parameter 7612 is set to 0 (cancel the synchronization of the axis by a reset). Whether the absolute coordinates are updated depends on the manual absolute signal present at that time.
3. Other causes of synchronization cancellation
   Synchronization is automatically canceled when:
   1) Emergency stop is applied
   2) A servo alarm is issued
   3) An overtravel (hard/soft) alarm is issued for the EGB axis
   4) A PW000 alarm (requiring power-off) is issued
   5) A PC alarm or IO alarm is issued

19.9.2 Command Specification Compatible with Hobbing Machine (G80,G81)

Synchronization can be specified in the same way as the operation of a hobbing machine is specified.

When the canned cycle option is specified, this specification method cannot be used.

If two sets of axes are specified for synchronization by this method, the EGB axis subjected to synchronization is determined by parameter 5995.

Format

G81 T_ (L_) (Q_ P_) ; Synchronization started

G80 ; Synchronization canceled

T : Number of teeth(Range of valid settings: 1 to 1000)

L : Number of hob threads
(Range of valid settings: -21 to +21 excluding 0).
Specify the direction of rotation about the C-axis with the sign of L:
When the sign of L is positive, rotation about the C-axis is in the forward (+) direction.
When the sign of L is negative, rotation about the C-axis is in the reverse (-) direction.

Q : Module or diametral pitch
Specify a module for metric input.
(Unit: 0.00001 mm, Range of valid settings: 0.1 to 25.0)
Specify a diametral pitch for inch input.
(Unit: 0.0001 inch-1, Range of valid settings: 0.1 to 25.0)

P : Gear helix angle
(Unit: 0.0001 degree, Range of valid settings: -90.0 degrees to +90 degrees)

* Decimal places can be specified in the Q and P settings.

Explanation

- Start of synchronization

P and Q are specified when using helical gear compensation. If only P or Q is specified, alarm PS0594 is output.

When G81 is specified to start synchronization mode, synchronization control of the spindle and the workpiece rotation axis (C-axis) starts.

During synchronization, the ratio of the rotation about the C-axis to the rotation of the spindle is made equal to the following ratio:

T (number of teeth) : L (hob threads)

If another G81 command is specified during synchronization without canceling synchronization, the synchronization factor is updated by T and L if specified. If only P and Q are specified and T and L are not, helical gear compensation is applied with the synchronizing factor remaining unchanged. Thus, a helical gear and spur gear can be machined in succession.

NOTE

While synchronization specified by the method compatible with hobbing machine is in progress, feed per revolution is performed according to the rotation speed about the slave axis of synchronization.

- End of synchronization

The synchronization of all synchronized axes is canceled. The absolute coordinates of slave axis are updated according to the amount of synchronous movement. For a rotation axis, the amount of synchronous movement (modulus with respect to 360 degrees) is added to the absolute coordinates. In a G80 block, no address other than O or N can be specified.

- Compensation for helical gears

For a helical gear, a compensation movement along the C-axis is made with respect to movement along the Z-axis according to the helix angle of the gear.

Helical gear offset is executed according the next formula.

$$\text{Compensation angle} = \frac{Z \times \sin(P)}{\pi \times T \times Q} \times 360 \text{ (for metric input)}$$

$$\text{Compensation angle} = \frac{Z \times Q \times \sin(P)}{\pi \times T} \times 360 \text{ (for inch input)}$$

Compensation angle : Absolute value (degrees) with a sign

Z : Amount of Z-axis movement after G81 (mm or inch). All movements along the Z-axis including automatic and manual movements.

P : Gear helix angle (plus or minus value in degrees)

π : Ratio of the circumference of a circle to its diameter

T : Number of teeth

Q : Module (mm) or diametral pitch (1/inch)

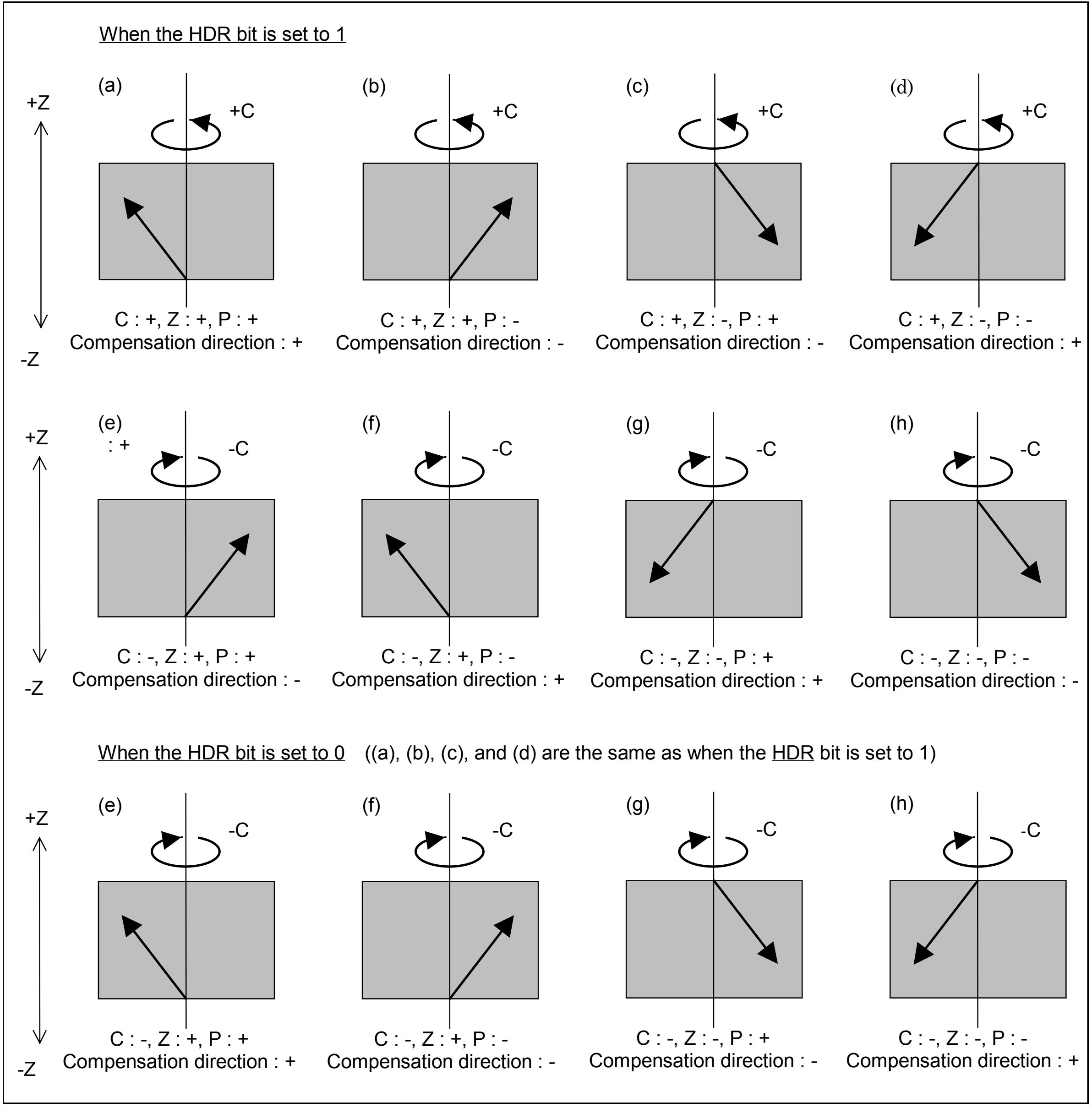
The values of P, T, and Q are those specified in a G81 block.

NOTE

The axial feed axis is an axis along the central axis of rotation about the C-axis and is usually the Z-axis. For convenience, this manual uses the Z-axis. When specifying an axis other than the Z-axis as the axial feed axis, specify the axis number in parameter 5994.

- Direction of helical gear compensation

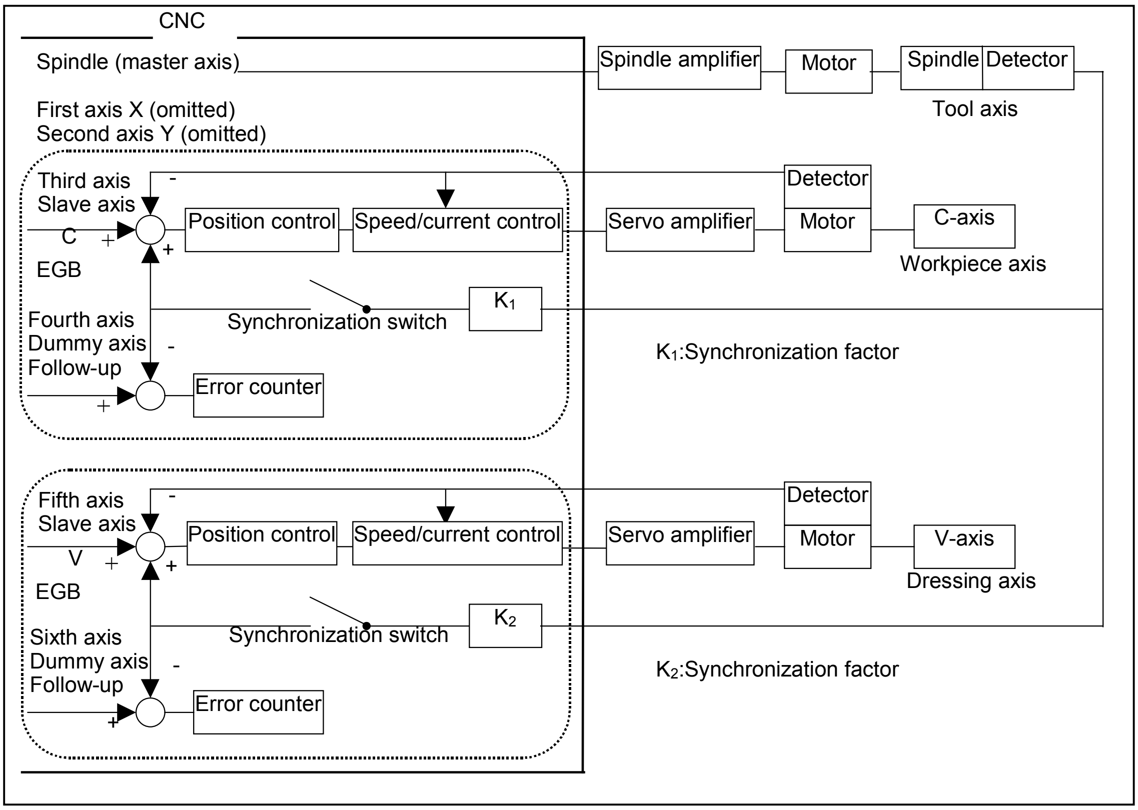
About HDR bit (bit 2 of parameter 7612)


When the HDR bit is set to 1
+Z
-Z
(a)
+C
C : +, Z : +, P : +
Compensation direction : +
(b)
+C
C : +, Z : +, P : -
Compensation direction : -
(c)
+C
C : +, Z : -, P : +
Compensation direction : -
(d)
+C
C : +, Z : -, P : -
Compensation direction : +
+Z
-Z
(e)
: +
-C
C : -, Z : +, P : +
Compensation direction : -
(f)
-C
C : -, Z : +, P : -
Compensation direction : +
(g)
-C
C : -, Z : -, P : +
Compensation direction : +
(h)
-C
C : -, Z : -, P : -
Compensation direction : -
When the HDR bit is set to 0 ((a), (b), (c), and (d) are the same as when the HDR bit is set to 1)
+Z
-Z
(e)
-C
C : -, Z : +, P : +
Compensation direction : +
(f)
-C
C : -, Z : +, P : -
Compensation direction : -
(g)
-C
C : -, Z : -, P : +
Compensation direction : -
(h)
-C
C : -, Z : -, P : -
Compensation direction : +

19.9.3 Example of Controlled Axis Configuration

- Gear grinder

Spindle : EGB master axis: Tool axis
First axis : X
Second axis : Y
Third axis : C-axis (EGB slave axis: Workpiece axis)
Fourth axis : C-axis (EGB dummy axis: Not usable as an ordinary controlled axis)
Fifth axis : V-axis (EGB slave axis: Dressing axis)
Sixth axis : V-axis (EGB dummy axis: Not usable as an ordinary controlled axis)



NOTE

With a sampling frequency of 1 ms, a feedback pulse signal is read from the master axis, the number of synchronization pulses for the slave axis is calculated from the synchronization factor K, and the calculated value is used for slave axis position control.

19.9.4 Sample Programs

- When the master axis is the spindle, and the slave axis is the C-axis

1. G81.5 T10 C0 L1 ;
   Synchronization between the master axis and C-axis is started at the ratio of one rotation about the C-axis to ten rotations about the master axis.

2. G81.5 T10 C0 L-1 ;
   Synchronization between the master axis and C-axis is started at the ratio of one rotation about the C-axis to ten rotations about the master axis. In this case, however, the direction of rotation is opposite to that of (a) above.

3. G81.5 T1 C3.26 ;
   Synchronization between the master axis and C-axis is started at the ratio of a 3.26-degree rotation about the C-axis per one rotation about the master axis.

4. G81.5 P10000 C-0.214 ;
   Synchronization between the master axis and C-axis is started at the ratio of a -0.214-degree rotation about the C-axis to 10,000 feedback pulses from the pulse coder of the master axis.

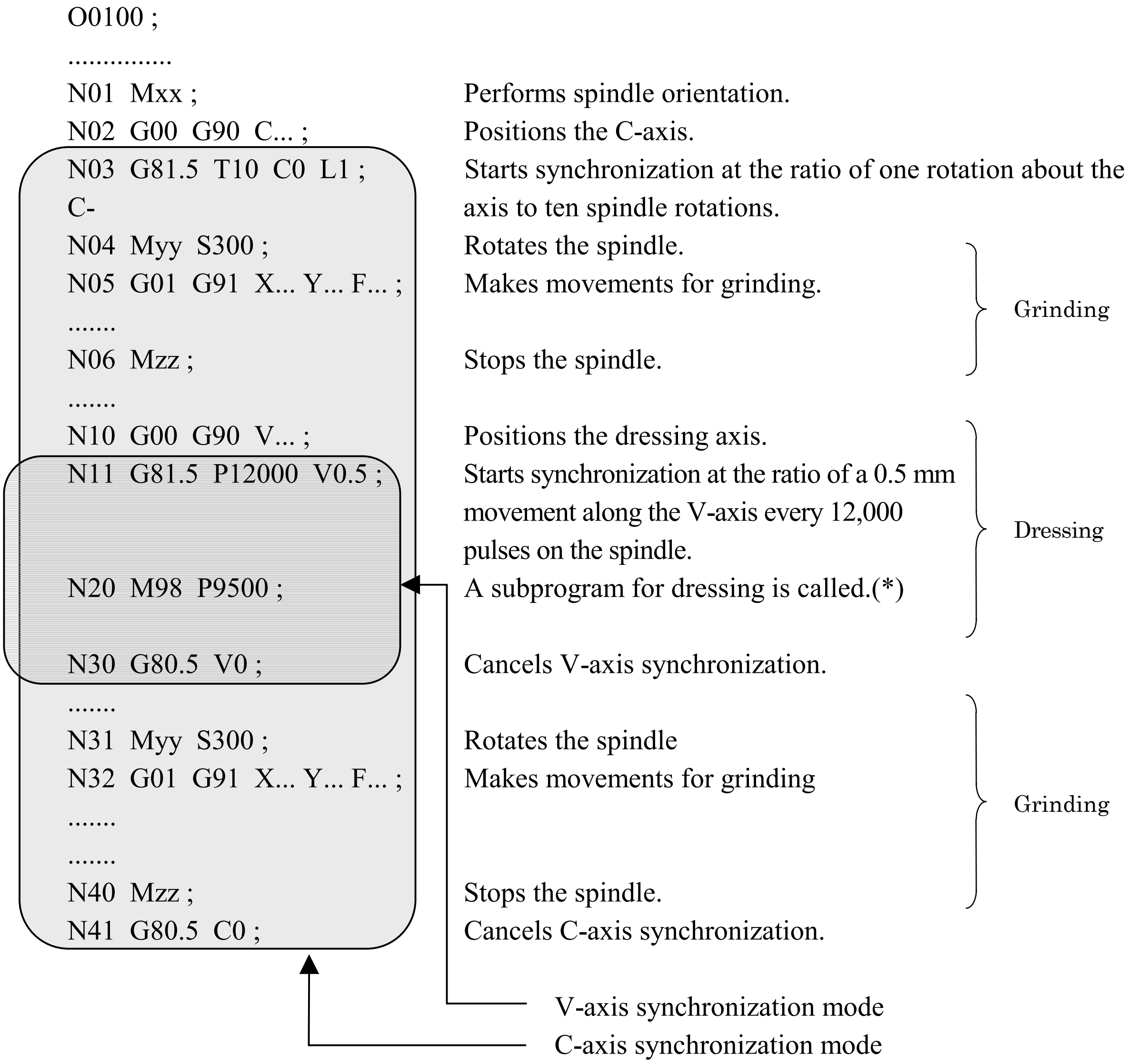
- When the master axis is the spindle, the slave axis is the V-axis (linear axis), and inch/metric conversion is performed

1. For a millimeter machine and metric input
   G81.5 T1 V1.0 ;
   Synchronization between the master axis and V-axis is started at the ratio of a 1.00 mm movement along the V-axis per rotation about the master axis.

2. For a millimeter machine and inch input
   G81.5 T1 V1.0 ;
   Synchronization between the master axis and V-axis is started at the ratio of a 1.0 inch movement (25.4 mm) along the V-axis per rotation about the master axis.

- When two groups of axes are synchronized simultaneously

Based on the controlled axis configuration described in II-1.1.3, the sample program below synchronizes the spindle with the V-axis while the spindle is synchronized with the C-axis.



```
O0100 ;
...............
N01  Mxx ;                        Performs spindle orientation.
N02  G00  G90  C... ;             Positions the C-axis.
N03  G81.5  T10  C0  L1 ;         Starts synchronization at the ratio of one rotation about the
C-                                axis to ten spindle rotations.
N04  Myy  S300 ;                  Rotates the spindle.                           ⎫
N05  G01  G91  X... Y... F... ;   Makes movements for grinding.                  ⎬ Grinding
.......                                                                          ⎪
N06  Mzz ;                        Stops the spindle.                             ⎭
.......
N10  G00  G90  V... ;             Positions the dressing axis.                   ⎫
N11  G81.5  P12000  V0.5 ;        Starts synchronization at the ratio of a 0.5 mm ⎪
                                  movement along the V-axis every 12,000         ⎬ Dressing
                                  pulses on the spindle.                         ⎪
N20  M98  P9500 ;                 A subprogram for dressing is called.(*)        ⎭

N30  G80.5  V0 ;                  Cancels V-axis synchronization.
.......                                                                          ⎫
N31  Myy  S300 ;                  Rotates the spindle                            ⎪
N32  G01  G91  X... Y... F... ;   Makes movements for grinding                   ⎬ Grinding
.......                                                                          ⎪
.......                                                                          ⎪
N40  Mzz ;                        Stops the spindle.                             ⎭
N41  G80.5  C0 ;                  Cancels C-axis synchronization.
```

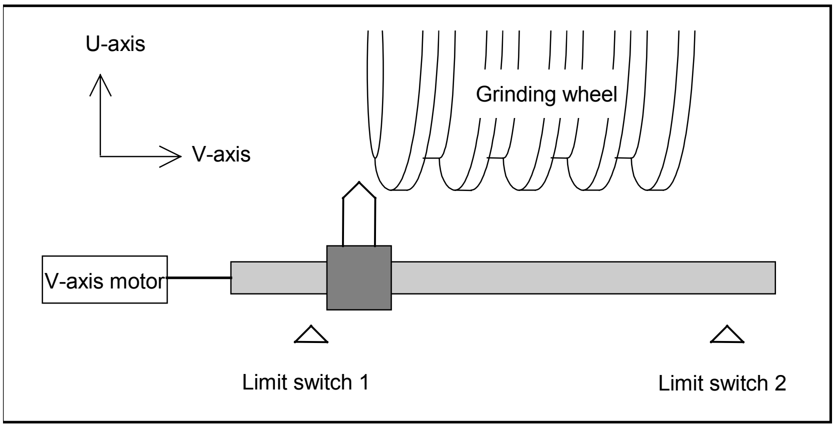
V-axis synchronization mode

C-axis synchronization mode

Thus, the synchronizations of two groups can be started and canceled independently of each other.

- Dressing

Dressing on a gear grinding machine configured as illustrated below



O9500 ;	
N01 G01 G91 U_ F100 ;	Approach along the dressing axis
N02 Maa S100 ;	With Maa, the PMC rotates the grinding wheel in the positive direction. Accordingly, movement in the positive direction is made on the V-axis. When the position of limit switch 2 is reached on the V-axis, the PMC stops the grinding wheel and returns FIN.
N02 U_ V_ ;	Movement to the next dressing position
N03 Mbb S100 ;	With Mbb, the PMC rotates the grinding wheel in the negative direction. Accordingly, movement in the negative direction is made on the V-axis. When the position of limit switch 1 is reached on the V-axis, the PMC stops the
grinding	wheel and returns FIN.
N04 U_ V_ ;	Movement to the next dressing position

If further dressing is required, N02 to N04 are repeated.

....
....

M99 ;

- Command specification for hobbing machines

Based on the controlled axis configuration described in Section 19.9.5, the sample program below sets the C-axis (in parameter 5995) for starting synchronization with the spindle according to the command specification method for hobbing machines.

O1234 ;	
....	
....	
N01 G81 T20 L1 ;	Starts synchronization with the spindle and C-axis at the ratio of a 1/20 rotation about the C-axis to one spindle rotation.
N02 Mxx S300 ;	Rotates the spindle at 300 min^{-1}.
N03 X... F... ;	Makes a movement along the X-axis (for cutting).
N04 Y... F... ;	Makes a movement along the Y-axis (for grinding). Axes such as the C-axis, X-axis, and Y-axis can be specified as required.
N05 X... F... ;	Makes a movement along the X-axis (for retraction).
N06 Mzz ;	Stops the spindle.
N07 G80 ;	Cancels the synchronization between the spindle and C-axis.

19.9.5 Synchronization Ratio Specification Range

The programmed ratio (synchronization ratio) of a movement along the slave axis to a movement along the master axis is converted to a detection unit ratio inside the NC. If such converted data (detection unit ratio) exceeds a certain allowable data range in the NC, synchronization cannot be established correctly, and an alarm (PS0596) is issued.
Even when a programmed master axis movement and a programmed slave axis movement are within specifiable ranges, a detection unit ratio obtained by conversion can exceed the allowable range, thus resulting in an alarm.
Let K be a synchronization ratio. The internal data corresponding K is the amount of slave axis movement (Kn) represented in the detection unit divided by the amount of master axis movement (Kd) represented in the detection unit; this fraction is represented as Kn/Kd (reduced to its lowest terms) as indicated below.

K=Kn/Kd=
Amount of slave axis movement represented in the detection unit / Amount of master axis movement represented in the detection unit
Kn and Kd must lie within the following ranges:
$-2147483648 \leq Kn \leq 2147483647$
$1 \leq Kd \leq 65535$

When Kn or Kd exceeds its allowable range above, an alarm is issued.
In conversion to the detection unit, when the CMR (command multiplication: parameter 1820) is a fraction or when inch/millimeter conversion is used, the fraction is directly converted without modification so that no error can occur in the conversion of specified amounts of movement.
During conversion, the amount of movement is multiplied by 254/100 for inch input on a millimeter machine, and 100/254 for metric input on an inch machine. Thus, Kn and Kd can become large numbers. If a synchronization ratio cannot be reduced to its lowest terms, an alarm condition is likely to occur.

- Based on the controlled axis configuration described in Subsec. II-19.9.3, suppose that the spindle and V-axis are as follows:

Spindle pulse coder :
72000pulse/rev(4 pulses for one A/B phase cycle)

C-axis least command increment	: 0.001 degree
C-axis CMR	: 5
V-axis least command increment	: 0.001mm
V-axis CMR	: 5

Then, the C-axis detection unit is 0.0002 degree. The V-axis detection unit is 0.0002 mm. In this case, the synchronization ratio (Kn, Kd) is related with a command as indicated below. Here, let Pm and Ps be the amounts of movements represented in the detection unit on the master axis and slave axis specified in a synchronization start command, respectively.

(1) When the master axis is the spindle, and the slave axis is the C-axis

(a) Command : G81.5 T10 C0 L1 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of one rotation about the C-axis to ten spindle rotations.

Pm : (Number of pulses per spindle rotation) ×10 rotations → 72000 × 10

Ps : (Amount of movement per rotation about the C-axis) × CMR × (one rotation) → 360000 × 5 × 1

$$\frac{Kn}{Kd}=\frac{360000\times5\times1}{72000\times10}=\frac{5}{2}$$

Both Kn and Kd are within the allowable range. No alarm is output.

(b) Command : G81.5 T10 C0 L-1 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of one rotation about the C-axis to ten spindle rotations. In this case, however, the direction of rotation is opposite to that of (a) above.

Pm : (Number of pulses per spindle rotation) × 10 revolutions → 72000 × 10

Ps : (Amount of movement per rotation about the C-axis) × CMR × (one rotation) → -360000 × 5 × 1

$$\frac{Kn}{Kd}=\frac{-360000\times5\times1}{72000\times10}=\frac{-5}{2}$$

Both Kn and Kd are within the allowable range. No alarm is output.

(c) Command : G81.5 T1 C3.263 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of a 3.263-degree rotation about the C-axis to one spindle rotation.

Pm : (Number of pulses per spindle rotation) × 1 rotation → 72000 × 1

Ps : (Amount of C-axis movement) × CMR → 3263 × 5

$$\frac{Kn}{Kd}=\frac{3263\times5}{72000\times1}=\frac{3263}{14400}$$

Both Kn and Kd are within the allowable range. No alarm is output.

In this sample program, when T1 is specified for the master axis, the synchronization ratio (fraction) of the CMR of the C-axis to the denominator Kd can always be reduced to lowest terms, thus Kd falls in the allowable range. So, the specifiable range of C is as follows:

$-99999999 \le C \le 99999999$

(d) Command : G81.5 T10 C3.263 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of a 3.263-degree rotation about the C-axis to ten spindle rotations.

Pm : (Number of pulses per spindle rotation) × 10 rotations → 72000 × 10

Ps : (Amount of the C-axis movement)×CMR → 3263×5

$$\frac{Kn}{Kd}=\frac{3263\times5}{72000\times10}=\frac{3263}{144000}$$

In this case, an alarm is issued because Kd exceeds the specifiable range.

(e) Command : G81.5 P10000 C-0.214 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of a -0.214- degree rotation of the C-axis to 10,000 feedback pulses from the pulse coder of the spindle.

Pm : (Specified number of feedback pulses from the pulse coder of the spindle) → 10000

Ps : (Amount of C-axis movement) × CMR → -214 × 5

$$\frac{Kn}{Kd}=\frac{-214\times5}{10000}=\frac{-107}{1000}$$

Both Kn and Kd are within the allowable range. No alarm is output.

(2) When the master axis is the spindle, the slave axis is the V-axis (linear axis), and inch/metric conversion is performed

(a) For a millimeter machine and metric input

Command : G81.5 T1 V1.0 ;

Operation : Synchronization between the spindle and V-axis is started at the ratio of a 1.00 mm movement along the V-axis per spindle rotation.

Pm : (Number of pulses per spindle rotation) × 1 rotation → 72000 × 1

Ps : (Amount of V-axis movement) × CMR → 1000 × 5

$$\frac{Kn}{Kd}=\frac{1000\times5}{72000}=\frac{5}{72}$$

Both Kn and Kd are within the allowable range. No alarm is output.

(b) For a millimeter machine and inch input

Command :G81.5 T1 V1.0 ;

Operation : Synchronization between the spindle and V-axis is started at the ratio of a 1.0 inch movement (25.4 mm) along the V-axis per spindle rotation.

Pm : (Number of pulses per spindle rotation) × 1 revolution → 72000 × 1

Ps : (Amount of V-axis movement) × CMR × 254 ÷ 100 → 10000 × 5 × 254 ÷ 100

$$\frac{Kn}{Kd}=\frac{10000\times5\times254}{72000\times100}=\frac{127}{72}$$

Both Kn and Kd are within the allowable range. No alarm is output.

(c) For a millimeter machine and inch input

Command : G81.5 T1 V0.0013 ;

Operation : Synchronization between the spindle and V-axis is started at the ratio of a 0.0013 inch (0.03302 mm) movement along the V-axis per spindle rotation.

Pm : (Number of pulses per spindle rotation) × 1 rotation → 72000 × 1

Ps : (Amount of V-axis movement) × CMR × 254 ÷ 100 → 13 × 5 × 254 ÷ 100

$$\frac{Kn}{Kd}=\frac{13\times5\times254}{72000\times100}=\frac{1651}{720000}$$

In this case, an alarm is issued because Kd exceeds the specifiable range.

- Based on the controlled axis configuration described in Subsec. II-19.9.1, suppose that the spindle and V-axis are as follows:

Spindle pulse coder	: 72000 pulses/rotation (4 pulses for one A/B phase cycle)
C-axis least command increment	: 0.001 degree
C-axis CMR	: 1/2
V-axis least command increment	: 0.001mm
V-axis CMR	: 1/2

Then, the C-axis detection unit is 0.002 degree. The V-axis detection unit is 0.002 mm. In this case, the synchronization ratio (Kn, Kd) is related with a command as indicated below. Here, let Pm and Ps be the amounts of movements represented in the detection unit for the master axis and slave axis specified in a synchronization start command, respectively.

(1) When the master axis is the spindle, and the slave axis is the C-axis

(a) Command : G81.5 T1 C3.263 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of a 3.263-degree rotation about the C-axis per spindle rotation.

Pm : (Number of pulses per spindle rotation) × 1 rotation → 72000 × 1

Ps : (Amount of C-axis movement) × CMR → 3263 × 1 ÷ 2

$$\frac{Kn}{Kd}=\frac{3263\times1}{72000\times2}=\frac{3263}{144000}$$

In this case, an alarm is issued because Kd exceeds the specifiable range.

(b) Command : G81.5 T1 C3.26 ;

Operation : Synchronization between the spindle and C-axis is started at the ratio of a 3.26-degree rotation about the C-axis per spindle rotation.

Pm : (Number of pulses per spindle rotation) × 1 revolution → 72000 × 1

Ps : (Amount of C-axis movement) × CMR → 3260 × 1 ÷ 2

$$\frac{Kn}{Kd}=\frac{3260\times1}{72000\times2}=\frac{163}{7200}$$

(a) causes an alarm to be output because the values cannot be abbreviated. (b) causes no alarm because the ratio of the travel distances can be abbreviated to a simple ratio.

19.9.6 Retract Function

- Retract function by an external signal

When the retract switch on the machine operator's panel is turned on, retraction and feedrate are made by the amount specified in parameter 7796 and 7795. When the retract amount is set to 0 for an axis, no movement is made on that axis.

For details of the retract switch, refer to the manual supplied by the machine tool builder.

- Retraction upon a servo/spindle alarm

If a CNC alarm is output upon the occurrence of an error on the servo axis or spindle, retraction is made according to the direction and feedrate specified in the parameters 7796 and 7795.

This function prevents the tool or workpiece from being damaged in the event of a servo alarm.

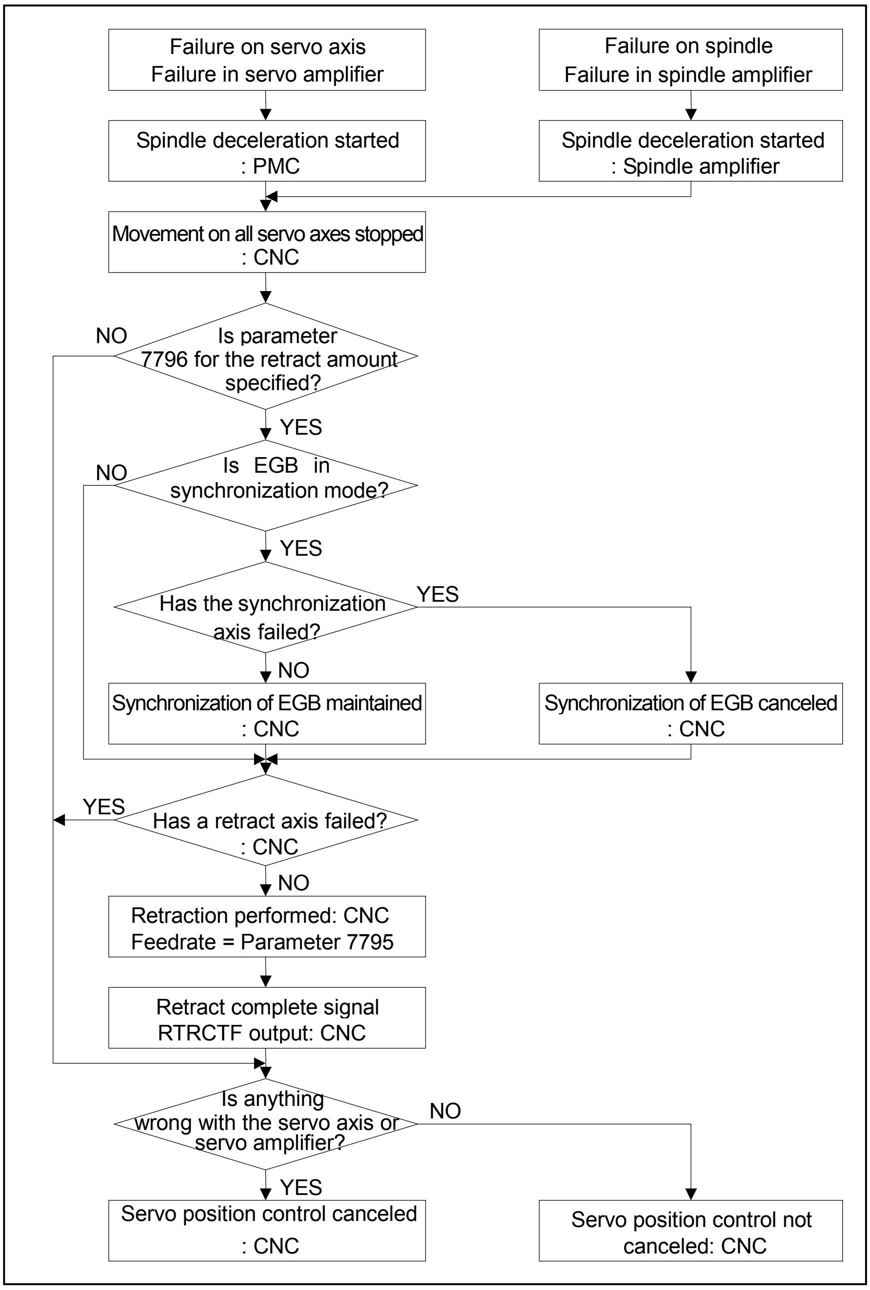
⚠ **CAUTION**

1 Feed hold cannot be applied to retraction.
2 In this case, the feedrate override capability is disabled.

NOTE

1 When the retract signal goes on during automatic operation, retract operation is performed, and automatic operation is stopped.
2 Automatic operation cannot be performed in retraction.
When a cycle is started, this warning is displayed: CANNOT START (RETRACTING).
3 Retraction is performed on the basis of positioning by linear interpolation.
4 Servo position control stops 400 milliseconds after retraction by a servo alarm.

- Processing of the retract function by a servo/spindle alarm


Failure on servo axis
Failure in servo amplifier
Failure on spindle
Failure in spindle amplifier
Spindle deceleration started
: PMC
Spindle deceleration started
: Spindle amplifier
Movement on all servo axes stopped
: CNC
NO
Is parameter
7796 for the retract amount
specified?
YES
NO
Is EGB in
synchronization mode?
YES
Has the synchronization
axis failed?
YES
NO
Synchronization of EGB maintained
: CNC
Synchronization of EGB canceled
: CNC
YES
Has a retract axis failed?
: CNC
NO
Retraction performed: CNC
Feedrate = Parameter 7795
Retract complete signal
RTRCTF output: CNC
Is anything
wrong with the servo axis or
servo amplifier?
NO
YES
Servo position control canceled
: CNC
Servo position control not
canceled: CNC

19.9.7 Electronic Gear Box Automatic Phase Synchronization

When synchronization start or cancellation is specified, the EGB (Electronic Gear Box) function does not immediately start or cancel synchronization. Instead, it performs acceleration or deceleration. Synchronization can be started or canceled without stopping the rotation of the spindle.

When synchronization starts, automatic phase matching is performed so that the position relative to the C-axis matches the position of the one-rotation signal on the spindle. This operation is similar to the operation at the beginning of synchronization by a one-rotation signal in hob synchronization with conventional functions for hobbing machine.

Format

- Acceleration/deceleration type

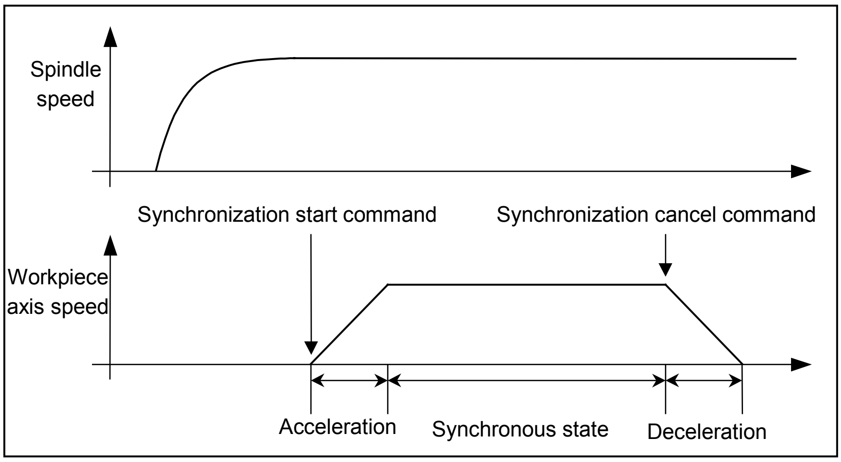
```
G81 T_ L_ R1 ;         Starting synchronization
G80 R1 ;               Canceling synchronization

 T :  Number of gear teeth (valid range: 1 to 1000)
 L :  Number of hob threads (valid range: -21 to +21, except 0)
      When L is a positive value:
        The C-axis turns in the positive direction (+)
      When L is a negative value:
        The C-axis turns in the negative direction (-)
```

- Acceleration/deceleration and automatic phase synchronization

```
G81 T_ L_ R2 ;         Starting synchronization
G80 R2 ;               Canceling synchronization

 T : Number of gear teeth (valid range: 1 to 1000)
 L : Number of hob threads (valid range: -21 to +21, except 0)
      When L is a positive value:
        The C-axis turns in the positive direction (+)
      When L is a negative value:
        The C-axis turns in the negative direction (-)
```

Explanation

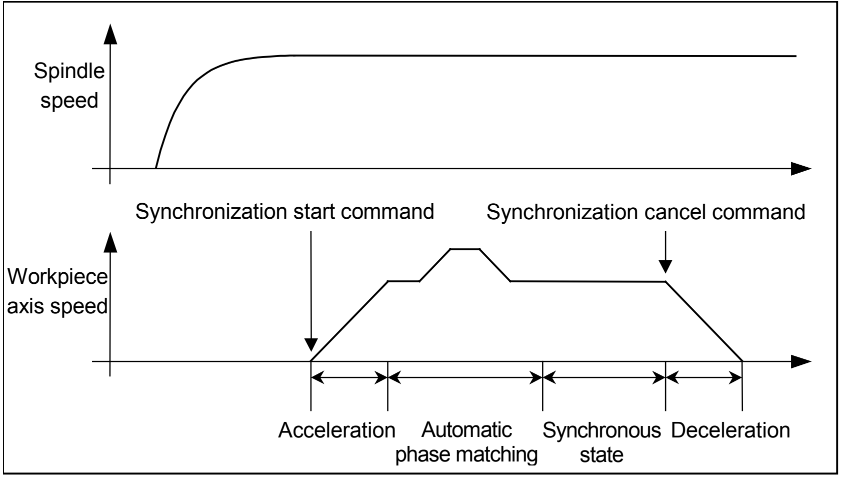
- Acceleration/deceleration type



1. Starting synchronization
   When synchronization is started, the workpiece axis speed is accelerated according to the acceleration rate set in the parameter (No. 7729).

2. Canceling synchronization
   Deceleration starts according to the acceleration rate set in the parameter (No. 7729).

- Acceleration/deceleration and automatic phase synchronization



1. Starting synchronization
   When synchronization starts, the travel speed of the workpiece increases by the acceleration specified in parameter 7729.
   Once the synchronization speed is attained, phase synchronization is performed automatically. In phase synchronization, the workpiece axis is moved so that the workpiece axis position at the time of synchronization start command issue matches the one-rotation signal position of the spindle.

2. Canceling synchronization
   Deceleration starts according to the acceleration rate set in the parameter (No. 7729).

⚠ **CAUTION**

The automatic phase matching speed is specified in parameter 5984 while the travel direction is specified in the PHD bit (bit 1 of parameter 7712).
In phase synchronization, rapid traverse linear acceleration/deceleration is performed(time constant is set by parameter No.1620).The speed in the workpiece axis is determined by superimposing the automatic phase synchronization speed on the speed synchronism with the spindle rotation. Specify the limit of position deviation in parameter 1828 to allow for superimposing.

NOTE

1 In automatic phase synchronization, the reference position used for phase synchronization for the workpiece axis can be shifted from the one-rotation signal position by parameter setting (data No. 5985).

2 When a synchronization command is next issued while the synchronous state is set, automatic phase synchronization causes workpiece axis motion such that the spindle one-rotation signal matches the same workpiece axis position as that when the previous G81R2 synchronization start command was executed.

3 In automatic phase synchronization, the workpiece axis is moved in the parameter-set phase synchronization move direction to the phase position nearest to the current position.

4 The acceleration/deceleration performed at synchronization start and cancellation is linear.

5 Acceleration/deceleration and automatic phase synchronization can be performed by setting the appropriate parameter (No. 7712#0, PHS). In this case, the R2 command need not be specified in either block G81 or G80.

6 When synchronization is canceled automatically as a result of any of the following, deceleration is performed, after which synchronization is canceled:
1) Reset
2) PW000 (POWER MUST BE OFF)
3) IO alarm

Example

- Acceleration/deceleration type

```
M03 ;              Clockwise spindle rotation command
G81  T_  L_  R1 ;  Synchronization start command
G00  X_ ;          Positions the workpiece at the machining position.
Machining in the synchronous state
G00  X_ ;          Retract the workpiece from the tool.
G80  R1  ;         Synchronization cancel command
```

- Acceleration/deceleration and automatic phase synchronization

```
M03 ;              Clockwise spindle rotation command
G00  G90  C_ ;     C-axis positioning
G81  T_  L_  R2 ;  Synchronization start command
G00  X_ ;          Positions the workpiece at the machining position.
Machining in the synchronous state
G00  X_ ;          Retract the workpiece from the tool.
G80  R2  ;         Synchronization cancel command
```

Alarm

Number	Message	Contents
PS0593	EGB PARAMETER SETTING ERROR	Erroneous EGB parameter setting (1) The setting of SYN parameter No. 1955#0 is incorrect. (2) The slave axes set by the G81 code are not set to rotary axes. (ROT and ROS parameter Nos. 1006#0, #1) (3) The number of pulses (parameter Nos. 5596, 5597) per rotation is not set. (4) Parameter No. 5995 is not set by the hobbing machine compatible instruction.
PS0594	EGB FORMAT ERROR	Format error in the block in which EGB was specified (1) Nothing is specified to the master axis or slave axes in the G81.5 block. (2) Out of range data is specified to the master axis or slave axes in the G81.5 block. (3) Number of threads T is not specified in the G81 block. (4) Out of range data is specified by one of T, L, P or Q codes in the G81 block. (5) Only one of P or Q codes is specified in the G81 block.
PS0595	ILL-COMMAND IN EGB MODE	An illegal instruction was issued during synchronization by EGB. (1) Slave axis was specified by G27, G28, G29, G30, G30.1, G33 and G53 G codes. (2) Inch/metric conversion was specified by G20 or G21 G codes.
PS0596	EGB OVERFLOW	An overflow occurred during calculation of the synchronization coefficient.
PS0597	EGB AUTO PHASE FORMAT ERROR	Format error in block in which G80 or G81 was specified by EGB automatic phase alignment (1) R is data outside of the instruction range.
PS0598	EGB AUTO PHASE PARAMETER SETTING ERROR	Erroneous parameter settings relating to EGB automatic phase alignment (1) The acceleration/deceleration parameter is invalid. (2) The automatic phase alignment parameter is invalid.

19.10 SKIP FUNCTION FOR EGB AXIS (G31.8)

This function validates a skip signal or high-speed skip signal (both referred to as the skip signal) for the EGB slave axis in the synchronization mode set by the EGB (Electronic Gear Box) function.
This function has these features.

1. The block with this function is not interrupted until the skip signal input has been counted to the commanded times.
2. The movement by synchronization of EGB axis is not stopped by skip signal input.
3. The value of machine coordinate is stored in the commanded variable of custom macro when the skip signal is input. And the total number of times of skip signal inputs is stored in other commanded variable.

Format

```
G81 T_ L_ ;                    (EGB mode on)
G31.8 G91 α0 P_ Q_ ( R_ ) ;    (EGB skip command)
```

α : EGB axis.
P : The top number of custom macro variables in which the value of machine coordinate is set when skip signal is input.
Q : The total times of skip signal input during execution of the block with G31.8. (range of command value : 1–900)
R : The number of custom macro variable in which the total number of times of skip signal inputs is set. This data is usually the same with the data of the variable specified by Q. Therefore this is not necessarily specified. Specify R when the total number of times of skip signal inputs should be confirmed.

Explanation

G31.8 is an one- shot G code.
After the execution of G31.8, values of machine coordinate which are gotten at every time of skip signal input are set in custom macro variables. The numbers of variables are used from the top number commanded by P to the number added with the amount of times commanded by Q.
And the total times of skip signal input is set in custom macro variable whose number is commanded by R.

Example

```
G81  T200  L2 ;                       (EGB mode on)
X_ ;
Z_ ;
G31.8  G91  A0  P500  Q200  R1 ;      (EGB skip command)
```

After 200 times of skip signal inputs, 200 skip positions of A axis corresponding to each skip signal input are set in the custom macro variables whose numbers are from 500 to 699. And the times of skip signal input is set in the custom macro variable whose number is 1 and whose value is 200.

NOTE

1 In this block, only one axis should be commanded for EGB axis. If no axis is commanded or, 2 or more axes are commanded, an alarm (PS152) will occur.

2 If P is not commanded in this block, an alarm (PS152) will occur.

3 If R is not commanded in this block, the times of skip signal input is not set in the custom macro variable.

4 The numbers of custom macro variables commanded by P or R should be specified within usable numbers. If the number is specified without usable number, an alarm (PS114) will occur. And when variables become lacking, an alarm (PS114) will occur.

5 Whether this function uses the conventional skip signal or high-speed skip signal is specified in the HSS bit (bit 4 of parameter 7200). When the high-speed skip signal is used, the individual high-speed signal to be used is specified in the 9S1 to 9S4 bits (bits 0 to 3 of parameter 7210).

6 If the EGB master axis is a controlled axis other than the spindle, the master axis must be a travel axis under PMC axis control.

7 The skip position is determined by the value calculated from the feedback pulse of the machine. The value indicates the machine position. Compensation based on a position error or servo time constant (loop gain) is not performed.

Alarm

Number	Message	Contents
PS0114	VARIABLE NO. OUT OF RANGE	An illegal No. was specified in a local variable, common variable or a system variable in a custom macro. A non-existent custom macro variable No. was specified in the EGB axis skip function (G31.8), or there are not enough custom macro variables for storing the skip position.
PS0152	G31.9/G31.8 FORMAT ERROR	The format of the G31.9 or G31.8 block is erroneous in the following cases: - The axis was not specified in the G31.9 or G31.8 block. - Multiple axes were specified in the G31.9 or G31.8 block. - The P code was specified in the G31.9 or G31.8 block.

19.11 TOOL WITHDRAWAL AND RETURN (G10.6)

To replace the tool damaged during machining or to check the status of machining, the tool can be withdrawn from a workpiece. The tool can then be advanced again to restart machining efficiently.

The tool withdrawal and return operation consists of the following four steps:

- Retract

  The tool is retracted to a predefined position using the TOOL WITHDRAW switch.

- Withdrawal

  The tool is moved to the tool-change position manually.

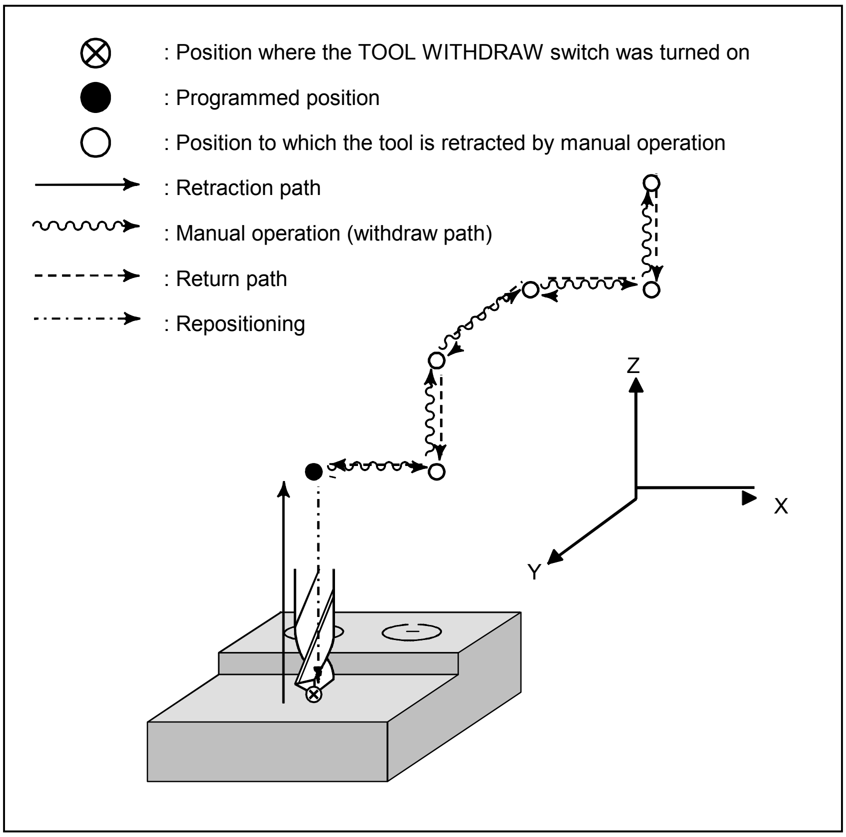
- Return

  The tool returns to the retract position.

- Repositioning

  The tool returns to the interrupted position.

For the tool withdrawal and return operations, see Operation II-4.10.


⊗ : Position where the TOOL WITHDRAW switch was turned on
● : Programmed position
○ : Position to which the tool is retracted by manual operation
: Retraction path
: Manual operation (withdraw path)
: Return path
: Repositioning
Z
X
Y

Format

Specify a retraction axis and distance in the following format:

```
Specify the amount of retraction, using G10.6.
G10.6 IP_;
  IP_ : In incremental mode, retraction distance from the
        position where the retract signal is turned on
        In the absolute mode, retraction distance to an absolute
        position
        The specified amount of retraction is effective until
        G10.6 is next executed.  To cancel the retraction,
        specify the following:
  G10.6 : (as a single block containing no other commands)
```

Explanation

- Retraction

When the TOOL WITHDRAW switch on the machine operator's panel is turned on during automatic operation or in the automatic operation stop or hold state, the tool is retracted the length of the programmed retraction distance. This operation is called retraction. The position at which retraction is completed is called the retraction position. Upon completion of retraction, the RETRACT POSITION LED on the machine operator's panel goes on.

When the TOOL WITHDRAW switch is turned on during execution of a block in automatic operation, execution of the block is interrupted immediately and the tool is retracted. After retraction is completed, the system enters the automatic operation hold state.

If the retraction distance and direction are not programmed, retraction is not performed. In this state, the tool can be withdrawn and returned.

When the TOOL WITHDRAW switch is turned on in the automatic operation stop or hold state, the tool is retracted, then the automatic operation stop or hold state is entered again.

When the TOOL WITHDRAW switch is turned on, the tool withdraw mode is set. When the tool withdraw mode is set, the TOOL BEING WITHDRAWN LED on the machine operator's panel goes on.

- Withdrawal

When the manual mode is set, the tool can be moved manually (Manual continuous feed or manual handle feed) to replace the tool or measure a machined workpiece. This operation is called a withdrawal. The tool withdrawal path is automatically memorized by the CNC.

- Return

When the mode is returned to automatic operation mode and the TOOL RETURN switch on the machine operator's panel is turned off, the CNC automatically moves the tool to the retraction position by tracing the manually-moved tool path backwards. This operation is called a return. Upon completion of a return to the retraction position, the RETRACTIONS POSITION LED comes on.

- Repositioning

When the cycle start button is pressed while the tool is in the retraction position, the tool moves to the position where the TOOL WITHDRAW switch was turned on. This operation is called repositioning. Upon completion of repositioning, the TOOL BEING WITHDRAWN LED is turned off, indicating that the tool withdrawal mode has terminated. Operation after completion of repositioning depends on the automatic operation state when the tool withdrawal mode is set.

(1) When the tool withdrawal mode is set during automatic operation, operation is resumed after completion of repositioning.

(2) When the tool withdrawal mode is set when automatic operation is held or stopped, the original automatic operation hold or stop state is set after completion of repositioning. When the cycle start button is pressed again, automatic operation is resumed.

Limitation

- offset

If the origin, presetting, or workpiece origin offset value (or External workpiece origin offset value) is changed after retraction is specified with G10.6 in absolute mode, the change is not reflected in the retraction position. After such changes are made, the retraction position must be respecified with G10.6.

When the tool is damaged, automatic operation can be interrupted with a tool withdrawal and return operation in order to replace the tool. Note that if the tool offset value is changed after tool replacement, the change is ignored when automatic operation is resumed from the start point or other point in the interrupted block.

- Machine lock, mirror image, and scaling

When withdrawing the tool manually in the tool withdrawal mode, never use the machine lock, mirror-image, or scaling function.

- Reset

Upon reset, the retraction data specified in G10.6 is cleared. Retraction data needs to be specified again.

- Retraction command

The tool withdrawal and return function is enabled even when the retraction command is not specified. In this case, retraction and repositioning are not performed.

⚠ **WARNING**

The retraction axis and retraction distance specified in G10.6 need to be changed in an appropriate block according to the figure being machined. Be very careful when specifying the retraction distance; an incorrect retraction distance may damage the workpiece, machine, or tool.

19.12 HIGH SPEED HRV MODE

Overview

Higher speed and higher precision HIGH SPEED HRV control can be performed by using the servo control card, the servo amplifier, and Separate Detector I/F Unit supporting HIGH SPEED HRV control.

Format

G05.4 Q1 ;	**Turns HIGH SPEED HRV mode on.**
G05.4 Q0 ;	**Turns HIGH SPEED HRV mode off.**

For details of HIGH SPEED HRV mode, refer to the "FANUC AC SERVO MOTOR α*i* series PARAMETER MANUAL (B-65270EN)."

- Sample program

```
O12345678
  :
G00 X...
G05.1 Q1           Turn fine HPCC mode on.
G01 X...Y...F...
G01 X...Y...
G00 X...Y...
G05.4 Q1           Turn HIGH SPEED HRV mode on.
G01 X...Y...
G02 I...J...R...
G00 X...Y...
G01 X...Y...
  :
G05.4 Q0           Turn HIGH SPEED HRV mode off.
G01 X...Y...
  :
G05.1 Q0           Turn fine HPCC mode off.
```

Restrictions

HIGH SPEED HRV mode is disabled under any of the following conditions, even if an attempt is made to turn it on:

- Automatic operation is stopped
- PMC axis control axis
- Axis on which a chopping operation is in progress
- Axis for which the setup type of the simultaneous auto/manual operation is selected

NOTE

1 While automatic operation is stopped, HIGH SPEED HRV mode is disabled. When automatic operation is resumed without a reset, the mode is enabled again.
If a reset is performed, HIGH SPEED HRV mode remains disabled until G5.4Q1 is issued.

2 If the servo is in the HIGH SPEED HRV disabled state (as determined with the status of the servo software, servo control card, servo amplifier, and Separate Detector I/F Unit), issuing G05.4Q1 causes "PS0010 invalid G code" alarm to be issued.
To determine whether HIGH SPEED HRV is enabled, check bit 1 (HIGH SPEED HRVOK) of the following numbers displayed on the diagnosis screen:

First axis	: Number 3022
Second axis	: Number 3042
Third axis	: Number 3062

(n-th axis: 3022 + 20 x (n – 1)

APPENDIX

A TAPE CODE LIST

IBC Code										EIA Code										Meaning	
Character	**8**	**7**	**6**	**1**	**4**		**3**	**2**	**1**	**Character**	**8**	**7**	**6**	**5**	**4**		**3**	**2**	**1**		
0			O			·				0			O			·					Number 0
1	O		O	O		O			O	1						·			O		Number 1
2	O		O			·		O		2						·		O			Number 2
3			O	O		·		O	O	3				O		·		O	O		Number 3
4	O		O			·	O			4						·	O				Number 4
5			O	O		·	O		O	5				O		·	O		O		Number 5
6			O			·	O	O		6				O		·	O	O			Number 6
7	O		O	O		·	O	O	O	7						·	O	O	O		Number 7
8	O		O		O	·				8					O	·					Number 8
9			O	O	O	·			O	9				O	O	·			O		Number 9
A		O		O		·			O	a		O	O			·			O		Address A
B		O				·		O		b		O	O			·		O			Address B
C	O	O		O		·		O	O	c		O	O	O		·		O	O		Address C
D		O				·	O			d		O	O			·	O				Address D
E	O	O		O		·	O		O	e		O	O	O		·	O		O		Address E
F	O	O				·	O	O		f		O	O	O		·	O	O			Address F
G		O		O		·	O	O	O	g		O	O			·	O	O	O		Address G
H		O			O	·				h		O	O		O	·					Address H
I	O	O		O	O	·			O	i		O	O	O	O	·			O		Address I
J	O	O		O	O	·		O		j		O		O		·		O	O		Address J
K		O			O	·		O	O	k		O		O		·		O			Address K
L	O	O		O	O	·	O			l		O				·		O	O		Address L
M		O			O	·	O		O	m		O		O		·	O				Address M
N		O		O	O	·	O	O		n		O				·	O		O		Address N
O	O	O			O	·	O	O	O	o		O				·	O	O			Address O
P		O		O		·				p		O		O		·	O	O	O		Address P
Q	O	O				·			O	q		O		O	O	·					Address Q
R	O	O		O		·		O		r		O			O	·			O		Address R
S		O				·		O	O	s			O	O		·		O			Address S
T	O	O		O		·	O			t			O			·		O	O		Address T
U		O				·	O		O	u			O	O		·	O				Address U
V		O		O		·	O	O		v			O			·	O		O		Address V
W	O	O				·	O	O	O	w			O			·	O	O			Address W
X	O	O		O	O	·				x			O	O		·	O	O	O		Address X
Y		O			O	·			O	y			O	O	O	·					Address Y
Z		O		O	O	·		O		z			O		O	·			O		Address Z

ISO code										EIA code										Meaning	
Character	8	7	6	1	4		3	2	1	Character	8	7	6	5	4		3	2	1		
DEL	O	O	O	O	O	·	O	O	O	Del		O	O	O	O	·	O	O	O		Delete (deleting a mispunch)
NUL						·				Blank						·					No punch. With EIA code, this code cannot be used in a significant information section.
BS	O				O	·				BS			O		O	·		O		*	Backspace
HT					O	·			O	Tab			O	O	O	·	O	O		*	Tabulator
LF or NL					O	·		O		CR or EOB	O					·				?	End of block
CR	O				O	·	O		O											*	Carriage return
SP	O		O			·				SP				O		·				*	Space
%	O		O	O		·	O		O	ER					O	·		O	O		Absolute rewind stop
(			O		O	·				(2-4-5)				O	O	·		O			Control out (start of comment)
)	O		O		O	·			O	(2-4-7)		O			O	·		O		?	Control in (end of comment)
+			O		O	·		O	O	+		O	O	O		·				*	Plus sign
-			O		O	·	O		O	-		O				·					Minus sign
:			O		O	·		O													Colon (address O)
/	O		O	O	O	·	O	O	O	/			O	O		·			O		Optional block skip
.			O	O	O	·	O	O		.		O	O		O	·		O	O		Period (decimal point)
#	O		O			·		O	O											?	Sharp
$			O			·	O													*	Dollar sign
&	O		O			·	O	O		&					O	·	O	O		?	Ampersand
‘			O			·	O	O	O											*	Apostrophe
*	O		O		O	·		O												?	Asterisk
,	O		O	O	O	·	O			,			O	O	O	·		O	O	?	Comma
;	O		O		O	·		O	O											*	Semicolon
<			O		O	·	O													*	Left angle bracket
=	O		O		O	·	O		O											*	Equal sign
>	O		O		O	·	O	O												*	Right angle bracket
?			O		O	·	O	O	O											?	Question mark
@	O	O				·														?	Commercial at mark
“			O					O												*	Quotation mark
[	O	O			O	·		O	O											?	Left square bracket
]	O	O			O	·	O		O											?	Right square bracket
_		O			O	·	O	O	O											?	Under score

NOTE

1 *:Codes with an asterisk that are entered in a comment area are read into memory. When entered in a significant data area, these codes are ignored.
x: Codes with an x are ignored.
?:Codes with a question mark are ignored when entered in a significant data area that comes before the program number. These codes are read into memory when entered in any other significant data area or comment area. However, when the custom macro option is used, the following codes are read into memory even when entered in a significant data area that comes before the program number.
When ISO codes are used: #, &, *, =, [and]
When EIA codes are used: (codes set by parameter)

4 Codes not shown in this table are ignored as long as their parity is correct.

5 Codes with incorrect parity cause a TH alarm to be generated (except when found in a comment area, in which case they are ignored.)

6 When EIA codes are used, the code with all eight holes punched is treated as a special case and does not generate a parity alarm.

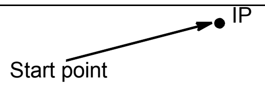
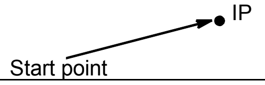
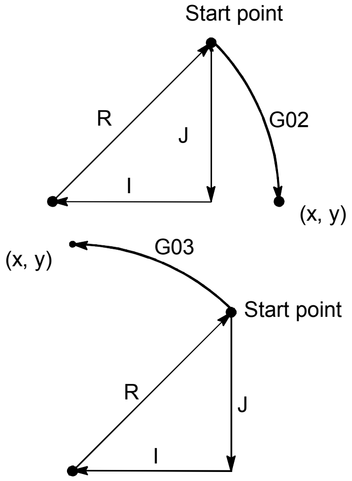
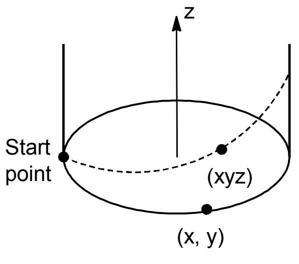
B LIST OF FUNCTION AND TAPE FORMAT

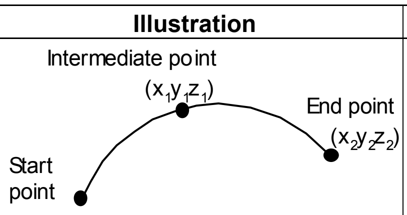
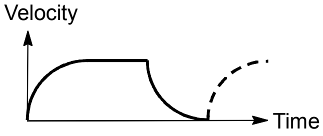
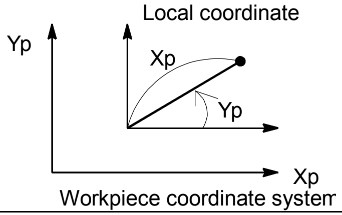
The symbols in the list represent the following.

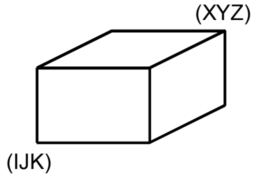
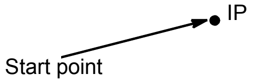
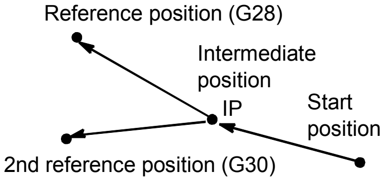
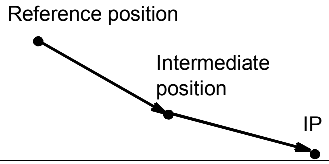
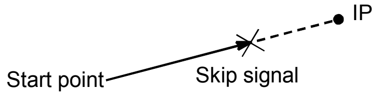
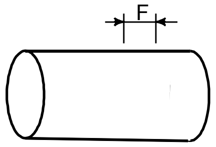
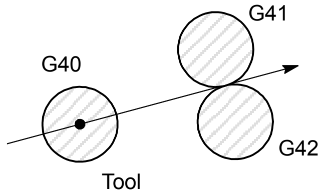
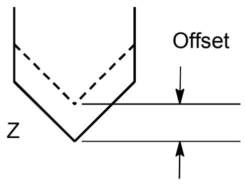
IP _ : X _ Y _ Z _ A _

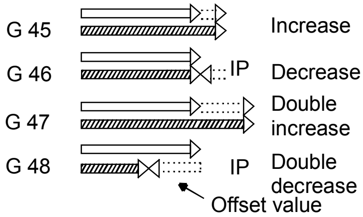
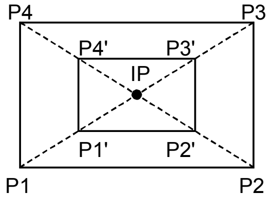
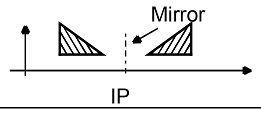
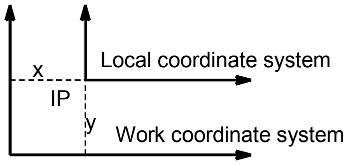
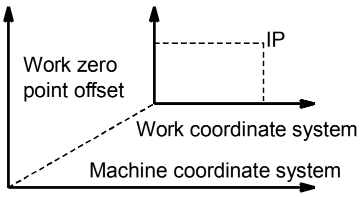
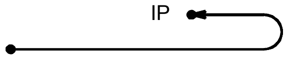
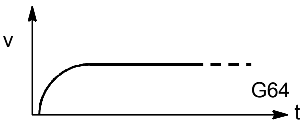
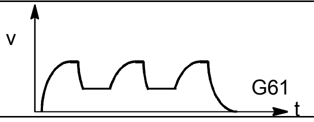
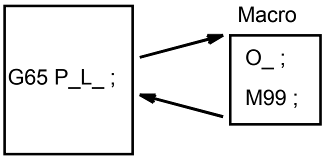
As seen above, the format consists of a combination of arbitary axis addresses among X, Y, Z, A, B, C, U, V, and W

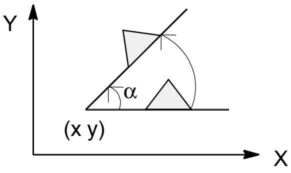
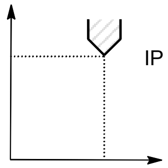
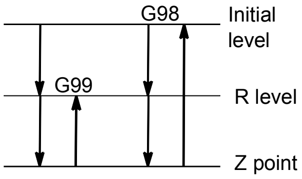
x	:	First basic axis (X usually)
y	:	Second basic axis (Y usually)
z	:	Third basic axis (Z usually)
a	:	One of arbitary addressed
b	:	One of arbitary addressed
Xp	:	X axis or its parallel axis
Yp	:	Y axis or its parallel axis
Zp	:	Z axis or its parallel axis

Functions	Illustration	Tape format
Positioning (G00)		G00 IP_ ;
Linear interpolation (G01)		G01 IP_ F_;
Circular interpolation (G02, G03)	 Start point R J G02 I (x, y) (x, y) G03 Start point R J I	$G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Xp\_\ Yp\_ \begin{Bmatrix} R\_ \\ I\_\ J\_ \end{Bmatrix} F\_ ;$ $G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Xp\_\ Zp\_ \begin{Bmatrix} R\_ \\ I\_\ K\_ \end{Bmatrix} F\_ ;$ $G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Yp\_\ Zp\_ \begin{Bmatrix} R\_ \\ J\_\ K\_ \end{Bmatrix} F\_ ;$
Helical interpolation (G02, G03)	 (In case of X-Y plane)	$G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Xp\_\ Yp\_ \begin{Bmatrix} R\_ \\ I\_\ J\_ \end{Bmatrix} \alpha\_F\_ ;$ $G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Xp\_\ Zp\_ \begin{Bmatrix} R\_ \\ I\_\ K\_ \end{Bmatrix} \alpha\_F\_ ;$ $G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Yp\_\ Zp\_ \begin{Bmatrix} R\_ \\ J\_\ K\_ \end{Bmatrix} \alpha\_F\_ ;$ α : Any axis other than circular interpolation axes.

Functions	Illustration	Tape format
Helical interpolation (G02, G03)	 Intermediate point ($x_1y_1z_1$) End point ($x_2y_2z_2$) Start point	G02.4 Xp_ Yp_ Zp_ α_ β_ ; (Intermediate point) Xp_ Yp_ Zp_ α_ β_ ; (End point) or G03.4 Xp_ Yp_ Zp_ α_ β_ ; (Intermediate point) Xp_ Yp_ Zp_ α_ β_ ; (End point) α, β : Any axis other than circular interpolation axes.
Dwell (G04)		Per second dwell $G04 \begin{Bmatrix} X\_ \\ P\_ \end{Bmatrix} ;$ Per revolution dwell $G04 \begin{Bmatrix} X\_ \\ P\_ \end{Bmatrix} ;$
Cylindrical interpolation (G07.1)		G07 IP_ R_ ; Cylindrical interpolation R : Radius of cylinder G07 IP0 ; Cylindrical interpolation cancel
Exact stop (G09)		$G09 \begin{Bmatrix} G01 \\ G02 \\ G03 \end{Bmatrix} IP\_ ;$
Change of offset value by program (G10)		Tool offset memory A G10 L11 P_ R_ ; Tool offset memory B G10 L11 P_ R_ ; (Wear offset value) G11 L11 P_ R_ ; (Geometry offset value) Tool offset memory C G10 L10 P_ R_ ; (Wear offset value/H) G10 L11 P_ R_ ; (Geometry offset value/H) G10 L12 P_ R_ ; (Wear offset value/D) G10 L13 P_ R_ ; (Geometry offset value/D) Workpiece zero point offset value G10 L2 P_ IP_ ;
Polar coordinate (G15, G16)	 Local coordinate Yp Xp Yp Xp Workpiece coordinate system	G17 G16 Xp_ Yp_ ; G18 G16 Zp_ Xp_ ; G19 G16 Yp_ Zp_ ; G15 ; Cancel
Plane section (G17, G18, G19)		G17 Xp_ Yp_ ; G18 ;Zp_ Xp_ G19 Yp_ Zp_ ;
Inch/millimeter conversion (G20, G21)		G20 ; Inch input G21 ; Millimeter input

Functions	Illustration	Tape format
Stored stroke check (G22, 23)		G22 X_ Y_ Z_ I_ J_ K_ ; G23 ; Cancel
Reference position return check (G27)		G27 IP_ ;
Reference position return (G28) 2nd, reference position return (G30)	 Reference position (G28) Intermediate position IP Start position 2nd reference position (G30)	G28 IP_ ; G30 IP_ ;P
Return from reference position to start point (G29)		G29 IP_ ;
Skip function (G31)		G31 IP_ F_;
Thread cutting (G33)		Thread cutting G33 IP_ F_ Q_ ; F_ : Larger component of lead Q_ : Angle by which the threading start angle is shifted Inch thread cutting G33 IP_ E_ Q_ ; E_ : Number of thread ridges per inch Q_ : Number of thread ridges per inch at threading start angle
Cutter compensation C (G40 to G42)		$\left\{\begin{array}{c}\text{G17}\\\text{G18}\\\text{G19}\end{array}\right\}\left\{\begin{array}{c}\text{G41}\\\text{G42}\end{array}\right\}$ D_ ; D : Tool offset G40 : Cancel
Normal direction control (G40.1, G41.1, G42.1)		G41.1 Normal direction control (left) G42.1 Normal direction control (right) G40.1 Normal direction control cancel
Tool length offset (G43, G44, G49)		$\left\{\begin{array}{c}\text{G43}\\\text{G44}\end{array}\right\}$ Z_ H_ ; $\left\{\begin{array}{c}\text{G43}\\\text{G44}\end{array}\right\}$ H_ ; H : Tool offset G49 : Cancel

Functions	Illustration	Tape format
Tool offset (G45 to G48)	 G 45 Increase G 46 IP Decrease G 47 Double increase G 48 IP Double decrease Offset value	G45 G46 IP_ D_ ; G47 G48 D : Tool offset number
Scaling (G50, G51)		G51 IP_ P_ ; P, I, J, K: Scaling magnification X, Y,Z : Center coordinate of scaling G50 ; Cancel
Programmable mirror image (G50.1, G51.1)		G51.1 IP_ ; G50.1 ; Cancel
Setting of local coordinate system (G52)	 Local coordinate system x IP y Work coordinate system	G52 IP_ ;
Command in machine coordinate system (G53)		G53 IP_ ;
Selection of work coordinate system (G54 to G59)	 IP Work zero point offset Work coordinate system Machine coordinate system	G54 : G59 IP_ ;
Single direction positioning (G60)		G60 IP_ ;
Cutting mode Exact stop mode Tapping mode		G64_ ; Cutting mode G61_ ; Exact stop mode G63_ ; Tapping mode
Automatic corner override		G62_ ; Automatic corner override
Custom macro (G65, G66, G67)		One-shot call G65 P_ L_ <Argument assignment>; P : Program No. L : Number of repetition Modal call G66 P_ L_ <Argument assignment>; G66.1 P_ L_ <Argument assignment>; G67; Cancel

Functions	Illustration	Tape format
Coordinate system rotation (G68, G69)	 (In case of X-Y plane)	$G68 \left\{\begin{array}{l} G17\ Xp\_\ Yp\_ \\ G18\ Zp\_\ Xp\_ \\ G19\ Yp\_\ Zp\_ \end{array}\right\}\quad R\ \underline{\alpha}\ ;$ G69 ; Cancel
Canned cycles (G73, G74, G80 to G89)	Refer to II.13. FUNCTIONS TO SIMPLIFY PROGRAMMING	G80 ; Cancel G73 $\left.\begin{array}{l} G74 \\ G76 \\ G81 \\ : \\ G89 \end{array}\right\}$ X_Y_Z_P_Q_R_F_K_;
Absolute/incremental programming (G90/G91)		G90_ ; Absolute command G91_ ; Incremental command G90_ G91_ ; Combined use
Change of workpiece coordinate system (G92)		G92 IP_ ;
Workpiece coordinate system preset (G92.1)		G92.1 IP 0 ;
Inverse time feed (G93)	1/min	G93 F_ ;
Feed per minute (G94)	mm/min inch/min	G94 F_ ;
Feed per revolution (G94, G95)	mm/rev inch/rev	G95 F_ ;
Constant surface speed control (G96, G97)		G96 S_ ; G97 S_ ;
Initial point return / R point return (G98, G99)		G98_ ; G99_ ;

C RANGE OF COMMAND VALUE

Linear axis

- in case of metric thread for feed screw and metric input

	Increment system				
	IS-A	IS-B	IS-C	IS-D	IS-E
Least input increment (mm)	0.01	0.001	0.0001	0.00001	0.000001
Least command increment (mm)	0.01	0.001	0.0001	0.00001	0.000001
Max. programmable dimension (mm)	±999,999.99	±999,999.999	±99,999.9999	±9,999.99999	±999.999999
Max. rapid traverse (mm/min)[*1]	2,400,000	240,000	99,999	9,999	999
Feedrate range(mm/min)[*1]	0.0001 - 2,400,000	0.0001 - 240,000	0.0001 – 99,999	0.00001 – 9,999	0.000001 - 999
Incremental feed (mm/step)[*2]	0.01 0.1 1.0 10.0 100.0 1,000.0	0.001 0.01 0.1 1.0 10.0 100.0	0.0001 0.001 0.01 0.1 1.0 10.0	0.00001 0.0001 0.001 0.01 0.1 1.0	0.000001 0.00001 0.0001 0.001 0.01 0.1
Tool compensation(mm)[*3]	0 - ±9,999.99	0 - ±9,999.999	0 - ±9,999.9999	0 - ±9,999.99999	0 - ±999.999999
Backlash compensation(pulse)[*4]	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999
Dwell time(sec)[*5]	0 - 999,999.99	0 - 999,999.999	0 - 99,999.9999	0 - 9,999.99999	0 - 999.999999

- in case of metric threads for feed screw and inch input

	Increment system				
	IS-A	IS-B	IS-C	IS-D	IS-E
Least input increment (inch)	0.001	0.0001	0.00001	0.000001	0.0000001
Least command increment (inch)	0.001	0.0001	0.00001	0.000001	0.0000001
Max. programmable dimension (inch)	±39,370.078	±39,370.0787	±3,937.00787	±393.700787	±39.3700787
Max. rapid traverse (mm/min)*1	2,400,000	240,000	99,999	9,999	999
Feedrate range (inch/min)*1	0.00001 - 96,000	0.00001 - 9,600	0.00001 - 4,000	0.000001 - 400	0.0000001 - 40
Incremental feed (inch/step)*2	0.001 0.01 0.1 1.0 10.0 100.0	0.0001 0.001 0.01 0.1 1.0 10.0	0.00001 0.0001 0.001 0.01 0.1 1.0	0.000001 0.00001 0.0001 0.001 0.01 0.1	0.0000001 0.000001 0.00001 0.0001 0.001 0.01
Tool compensation(inch)*3	0 - ±999.999	0 - ±999.9999	0 - ±999.99999	0 - ±999.999999	0 - ±99.9999999
Backlash compensation(pulse)*4	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999
Dwell time(sec)*5	0 - 999,999.99	0 - 999,999.999	0 - 99,999.9999	0 - 9,999.99999	0 - 999.999999

- in case of inch threads for feed screw and inch input)

	Increment system				
	IS-A	IS-B	IS-C	IS-D	IS-E
Least input increment *inch)	0.001	0.0001	0.00001	0.000001	0.0000001
Least command increment (inch)	0.001	0.0001	0.00001	0.000001	0.0000001
Max. programmable dimension (inch)	±99,999.999	±99,999.9999	±9,999.99999	±999.999999	±99.9999999
Max. rapid traverse (inch/min)*1	240,000	24,000	9,999	999	99
Feedrate range (inch/min)*1	0.00001 - 240,000	0.00001 - 24,000	0.00001 – 9,999	0.000001 - 999	0.0000001 - 99
Incremental feed (inch/step)*2	0.001 0.01 0.1 1.0 10.0 100.0	0.0001 0.001 0.01 0.1 1.0 10.0	0.00001 0.0001 0.001 0.01 0.1 1.0	0.000001 0.00001 0.0001 0.001 0.01 0.1	0.0000001 0.000001 0.00001 0.0001 0.001 0.01
Tool compensation(inch)*3	0 - ±999.999	0 - ±999.9999	0 - ±999.99999	0 - ±999.999999	0 - ±99.9999999
Backlash compensation(pulse)*4	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999
Dwell time(sec)*5	0 - 999,999.99	0 - 999,999.999	0 - 99,999.9999	0 - 9,999.99999	0 - 999.999999

- in case of inch thread for feed screw and metric input)

	Increment system				
	IS-A	IS-B	IS-C	IS-D	IS-E
Least input increment (mm)	0.01	0.001	0.0001	0.00001	0.000001
Least command increment (mm)	0.01	0.001	0.0001	0.00001	0.000001
Max. programmable dimension (mm)	±999,999.99	±999,999.999	±99,999.9999	±9,999.99999	±999.999999
Max. rapid traverse (inch/min)[*1]	240,000	24,000	9,999	999	99
Feedrate range(mm/min)[*1]	0.0001 - 2,400,000	0.0001 - 240,000	0.0001 - 99,999	0.00001 - 9,999	0.000001 - 99
Incremental feed (mm/step)[*2]	0.01 0.1 1.0 10.0 100.0 1000.0	0.001 0.01 0.1 1.0 10.0 100.0	0.0001 0.001 0.01 0.1 1.0 10.0	0.00001 0.0001 0.001 0.01 0.1 1.0	0.000001 0.00001 0.0001 0.001 0.01 0.1
Tool compensation(mm)[*3]	0 - ±9,999.99	0 - ±9,999.999	0 - ±9,999.9999	0 - ±9,999.99999	0 - ±999.999999
Backlash compensation(pulse)[*4]	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999
Dwell time(sec)[*5]	0 - 999,999.99	0 - 999,999.999	0 - 99,999.9999	0 - 9,999.99999	0 - 999.999999

Rotary axis

	Increment system				
	IS-A	IS-B	IS-C	IS-D	IS-E
Least input increment (deg)	0.01	0.001	0.0001	0.00001	0.000001
Least command increment (deg)	0.01	0.001	0.0001	0.00001	0.000001
Max. programmable dimension (deg)	±999,999.99	±999,999.999	±99,999.9999	±9,999.99999	±999.999999
Max. rapid traverse (deg/min)[*1]	2,400,000	240,000	99,999	9,999	999
Feedrate range(deg/min)[*1]	0.0001 - 2,400,000	0.0001 - 240,000	0.0001 - 99,999	0.00001 - 9,999	0.000001 - 999
Incremental feed (deg/step)[*2]	0.01 0.1 1.0 10.0 100.0 1000.0	0.001 0.01 0.1 1.0 10.0 100.0	0.0001 0.001 0.01 0.1 1.0 10.0	0.00001 0.0001 0.001 0.01 0.1 1.0	0.000001 0.00001 0.0001 0.001 0.01 0.1
Tool compensation(deg)[*3]	0 - ±9,999.99	0 - ±9,999.999	0 - ±9,999.9999	0 - ±9,999.99999	0 - ±999.999999
Backlash compensation(pulse)[*4]	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999	0 - ±9,999
Dwell time(sec)[*5]	0 - 999,999.99	0 - 999,999.999	0 - 99,999.9999	0 - 9,999.99999	0 - 999.999999

NOTE

*1 The feed rate range shown above are limitations depending on CNC interpolation capacity. When regarded as a whole system, limitations, depending on the servo system, must also be considered.

*2 Incremental feed amount can be specified by setting amount (parameter setting.)

*3 The values are applied when the offset extension function is used.
When input is switched between inch input and metric input, note the following. The maximum compensation value that can be set in inch input mode is (maximum compensation value) x 1/25.4.
If a compensation value beyond this limit is input in inch input mode, the value cannot be converted correctly to a metric value in metric input mode.

*4 The unit of backlash compensation is detection unit.

*5 Will depend on the unit system of the axis on address X.

D NOMOGRAPHS

D.1 INCORRECT THREADED LENGTH

The leads of a thread are generally incorrect in δ_1 and δ_2, as shown in Fig. D.1 (a), due to automatic acceleration and deceleration.
Thus distance allowances must be made to the extent of δ_1 and δ_2 in the program.

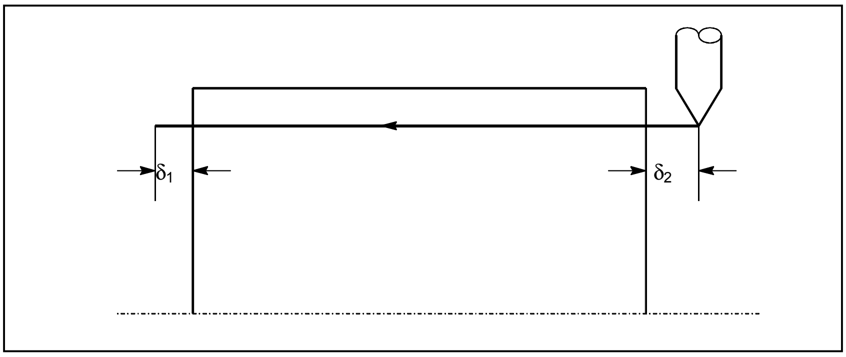


Fig. D.1(a) Incorrect thread position

Explanations

- How to determine δ_2

$$\delta_2 = T_1 V(\text{mm}) \quad \text{............(1)}$$

$$V = \frac{1}{60} RL$$

T_1: Time constant of servo system (sec)
V : Cutting speed (mm/sec)
R : Spindle speed (min^{-1})
L : Thread feed (mm)

Time constant T_1 (sec) of the servo system:
Usually 0.033 s.

- How to determine δ_1

$$\delta_1 = \left\{ t - T_1 + T_1 \exp\left(-\frac{t}{T_1} \right) \right\} V \quad \text{............(2)}$$

$$\alpha = \exp\left(-\frac{t}{T_1} \right) \quad \text{............(3)}$$

T_1 : Time constant of servo system (sec)
V : Cutting speed (mm/sec)

Time constant T_1 (sec) of the servo system: Usually 0.033 s.

The lead at the beginning of thread cutting is shorter than the specified lead L, and the allowable lead error is DL. Then as follows.

$$\alpha = \frac{\Delta L}{L}$$

When the value of HaI is determined, the time lapse until the thread accuracy is attained. The time HtI is substituted in (2) to determine δ_1: Constants V and T_1 are determined in the same way as for δ_2. Since the calculation of d1 is rather complex, a nomography is provided on the following pages.

- How to use nomograph

First specify the class and the lead of a thread. The thread accuracy, a, will be obtained at (1), and depending on the time constant of cutting feed acceleration/ deceleration, the d1 value when V = 10mm / s will be obtained at (2). Then, depending on the speed of thread cutting, d1 for speed other than 10mm/ s can be obtained at (3).

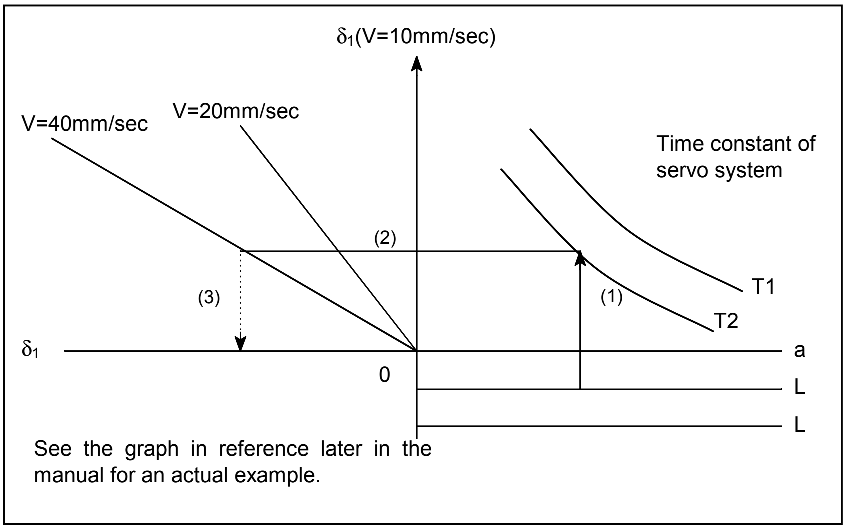

δ1(V=10mm/sec)
V=40mm/sec
V=20mm/sec
Time constant of servo system
(2)
(3)
(1)
T1
T2
δ1
a
0
L
L
See the graph in reference later in the manual for an actual example.

Fig. D.1(b) Nomograph

NOTE

The equations for δ_1, and δ_2 are for when the acceleration/ deceleration time constant for cutting feed is 0.

D.2 SIMPLE CALCULATION OF INCORRECT THREAD LENGTH

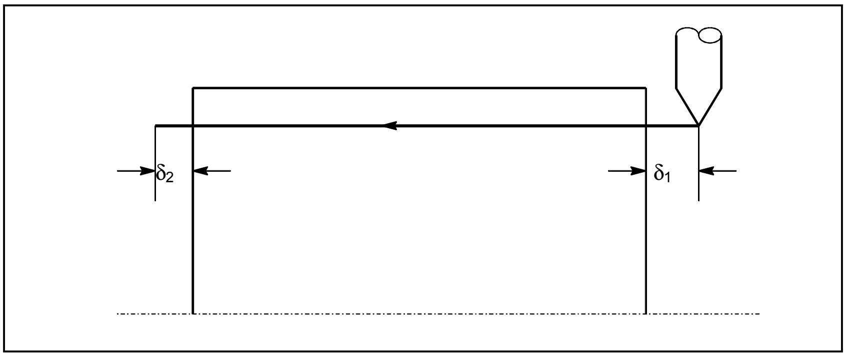


Fig. D.2(a) Incorrect threaded portion

Explanations

- How to determine δ2

$$\delta_2 = \frac{LR}{1800*}(\mathrm{mm})$$

R : Spindle speed (min^{-1})
L : Thread lead (mm)
* When time constant T of the servo system is 0.033 s.

- How to determine δ1

$$\delta_2 = \frac{LR}{1800*}(-1 - \ln a)(\mathrm{mm})$$
$$= \delta_2(-1 - \ln a)(\mathrm{mm})$$

R : Spindle speed (min^{-1})
L : Thread lead (mm)
* When time constant T of the servo system is 0.033 s.

Following a is a permited value of thread.

a	-1-lna
0.005	4.298
0.0	3.605
0.015	3.200
0.0	2.912

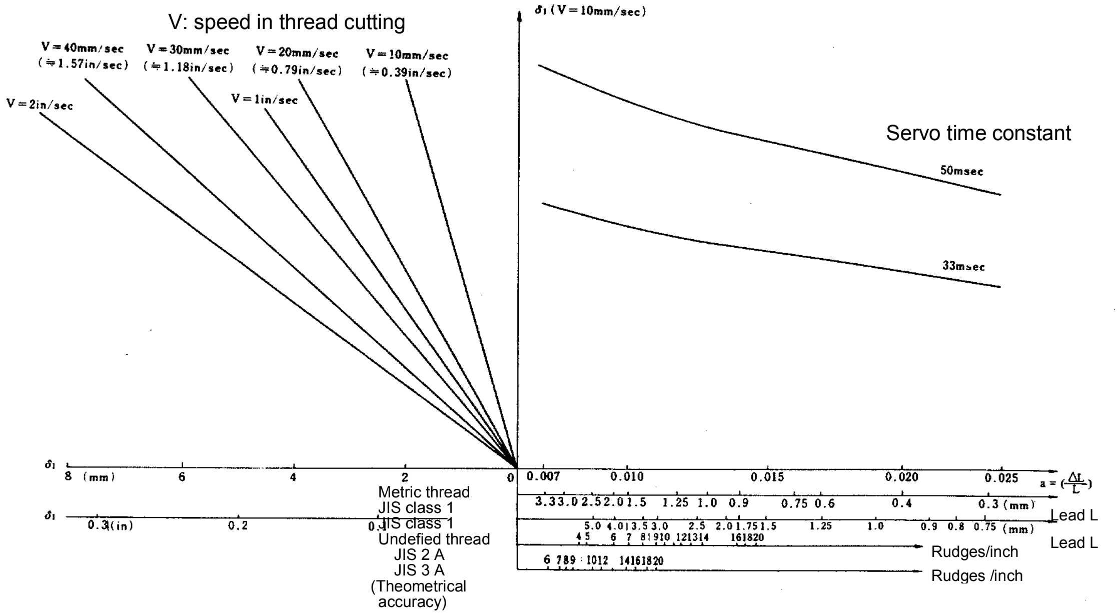
Examples

R=350 min^{-1}
L=1mm
a=0.01 then

$$\delta_2 = \frac{350 \times 1}{1800} = 0.194(\mathrm{mm})$$

$$\delta_1 = \delta_2 \times 3.605 = 0.701(\mathrm{mm})$$

- Reference



Nomograph for approach distance δ_1

Fig. D.2(b) Nomograph for obtaining approach distance δ_1

D.3 TOOL PATH AT CORNER

When servo system delay (by exponential acceleration/deceleration at cutting or caused by the positioning system when a servo motor is used) is accompanied by cornering, a slight deviation is produced between the tool path (tool center path) and the programmed path as shown in Fig. D.3 (a).
Time constant T_1 of the exponential acceleration/deceleration is fixed to 0.

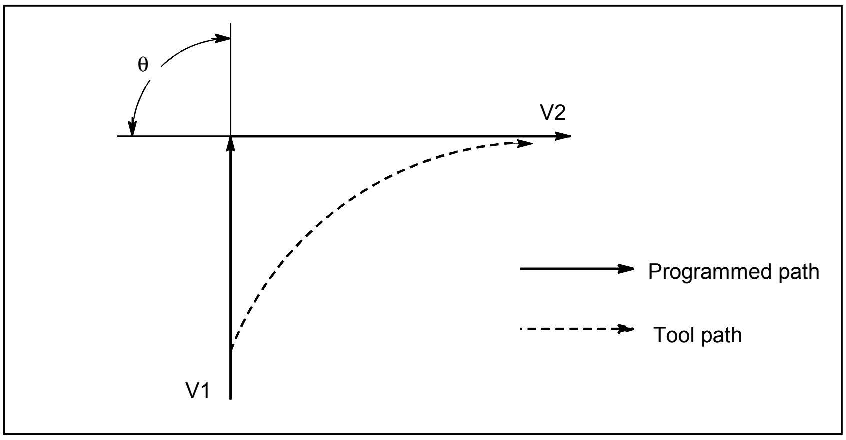


Fig. D.3(a) Slight deviation between the tool path and the programmed path

This tool path is determined by the following parameters:

- **Feedrate (V_1, V_2)**
- **Corner angle (θ)**
- **Exponential acceleration / deceleration time constant (T_1) at cutting (T_1=0)**
- **Presence or absence of buffer register.**

The above parameters are used to theoretically analyze the tool path and above tool path is drawn with the parameter which is set as an example.
When actually programming, the above items must be considered and programming must be performed carefully so that the shape of the workpiece is within the desired precision.
In other words, when the shape of the workpiece is not within the theoretical precision, the commands of the next block must not be read until the specified feedrate becomes zero. The dwell function is then used to stop the machine for the appropriate period.

Analysis

The tool path shown in Fig. D.3 (b) is analyzed based on the following conditions:
Feedrate is constant at both blocks before and after cornering.
The controller has a buffer register. (The error differs with the reading speed of the tape reader, number of characters of the next block, etc.)

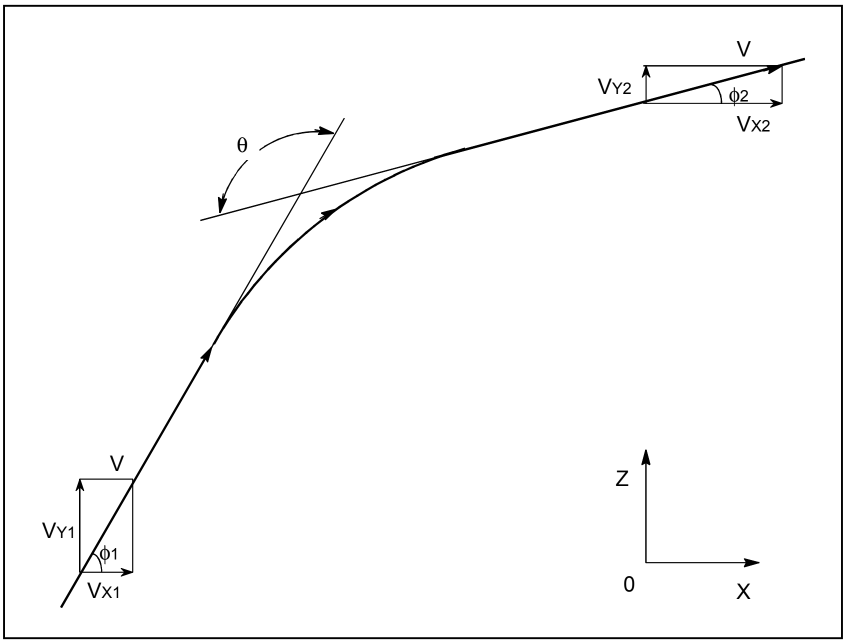


Fig. D.3(b) Example of tool path

- Description of conditions and symbols

$V_{X1} = V \cos \phi_1$
$V_{Y1} = V \sin \phi_1$
$V_{X2} = V \cos \phi_2$
$V_{Y2} = V \sin \phi_2$

V : Feedrate at both blocks before and after cornering
V_{X1} : X-axis component of feedrate of preceding block
V_{Y1} : Y-axis component of feedrate of preceding block
V_{X2} : X-axis component of feedrate of following block
V_{Y2} : Y-axis component of feedrate of following block
θ : Corner angle
ϕ_1 : Angle formed by specified path direction of preceding block and X-axis
ϕ_2 : Angle formed by specified path direction of following block and X-axis

- Initial value calculation

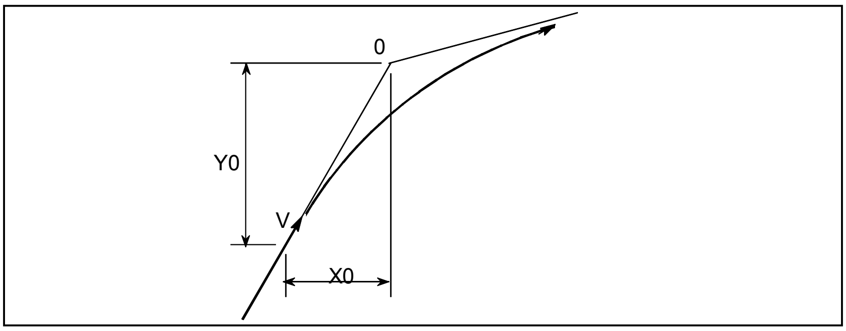


Fig. D.3(c) Initial value

The initial value when cornering begins, that is, the X and Y coordinates at the end of command distribution by the controller, is determined by the feedrate and the positioning system time constant of the servo motor.

$$X_0 = V_{X1}(T_1 + T_2)$$

$$Y_0 = V_{Y1}(T_1 + T_2)$$

T_1 : Exponential acceleration/deceleration time constant. (T=0)

T_2 : Time constant of positioning system (Inverse of position loop gain)

- Analysis of corner tool path

The equations below represent the feedrate for the corner section in X-axis direction and Y-axis direction.

$$V_X(t) = (V_{X2} - V_{X1})\left[1 - \frac{V_{X1}}{T_1 - T_2}\left\{T_1 \exp\left(-\frac{t}{T_1}\right) - T_2 \exp\left(-\frac{t}{T_2}\right)\right\} + V_{X1}\right]$$

$$= V_{X2}\left[1 - \frac{V_{X1}}{T_1 - T_2}\left\{T_1 \exp\left(-\frac{t}{T_1}\right) - T_2 \exp\left(-\frac{t}{T_2}\right)\right\}\right]$$

$$V_Y(t) = \frac{V_{Y1} - V_{Y2}}{T_1 - T_2}\left\{T_1 \exp\left(-\frac{t}{T_1}\right) - T_2 \exp\left(-\frac{t}{T_2}\right)\right\} + V_{Y1}$$

Therefore, the coordinates of the tool path at time t are calculated from the following equations:

$$X(t) = \int_0^1 V_X(t)dt - X_0$$

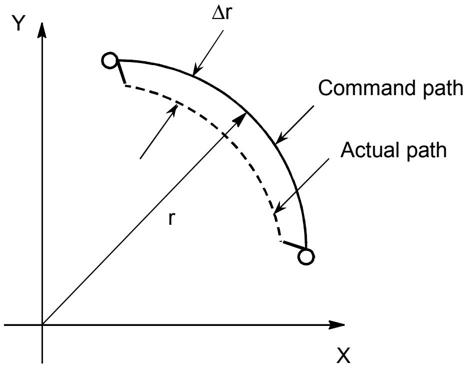
$$= \frac{V_{X2} - V_{X1}}{T_1 - T_2}\left\{T_1{}^2 \exp\left(-\frac{t}{T_1}\right) - T_2{}^2 \exp\left(-\frac{t}{T_2}\right)\right\} - V_{X2}\left(T_1 + T_2 - t\right)$$

$$Y(t) = \int_0^1 V_Y(t)dt - Y_0$$

$$= \frac{V_{Y2} - V_{Y1}}{T_1 - T_2}\left\{T_1{}^2 \exp\left(-\frac{t}{T_1}\right) - T_2{}^2 \exp\left(-\frac{t}{T_2}\right)\right\} - V_{Y2}\left(T_1 + T_2 - t\right)$$

D.4 RADIUS DIRECTION ERROR AT CIRCLE CUTTING

When a servo motor is used, the positioning system causes an error between input commands and output results. Since the tool advances along the specified segment, an error is not produced in linear interpolation. In circular interpolation, however, radial errors may be produced, sepecially for circular cutting at high speeds.
This error can be obtained as follows:



Δr : Maximum radius error (mm)
V : Feedrate (mm/s)
R : Circle radius (mm)
T_1 : Exponential acceleration/deceleration time constant (sec) at cutting (T=0)
T_2 : Time constant of positoning system (sec).
(Inverse of positon loop gain)

An approximation of this radius error can be obtained with the following expression:
In the case of exponential acceleration/deceleration

$$\Delta r = (\frac{1}{2}T_1^{\,2} + \frac{1}{2}T_2^{\,2})\frac{V^2}{r}$$

In the case of leanear acceleration/deceleration after interpolation

$$\Delta r = (\frac{1}{24}T_1^{\,2} + \frac{1}{2}T_2^{\,2})\frac{V^2}{r}$$

In the case of bell-shaped acceleration/deceleration

$$\Delta r = (\frac{1}{48}T_1^{\,2} + \frac{1}{2}T_2^{\,2})\frac{V^2}{r}$$

Thus, the radius error in the case of bell-shaped acceleration/ deceleration and linear acceleration/deceleration after interpolation is smaller than in case of exponential acceleration/deceleration by a factor of 12 or 24, excluding any error caused by a servo loop time constant.

Since the machining radius r (mm) and allowable error Δr (mm) of the workpiece is given in actual machining, the allowable limit feedrate v (mm /sec) is determined by equation (1).
Since the acceleration/deceleration time constant at cutting which is set by this equipment varies with the machine tool, refer to the manual issued by the machine tool builder.

E TABLE OF KANJI AND HIRAGANA CODES

- Table of Katakana codes

ア	B1	イ	B2	ウ	B3	エ	B4	オ	B5
カ	B6	キ	B7	ク	B8	ケ	B9	コ	BA
サ	BB	シ	BC	ス	BD	セ	BE	ソ	BF
タ	C0	チ	C1	ツ	C2	テ	C3	ト	C4
ナ	C5	ニ	C6	ヌ	C7	ネ	C8	ノ	C9
ハ	CA	ヒ	CB	フ	CC	ヘ	CD	ホ	CE
マ	CF	ミ	D0	ム	D1	メ	D2	モ	D3
ヤ	D4			ユ	D5			ヨ	D6
ラ	D7	リ	D8	ル	D9	レ	DA	ロ	DB
ワ	DC							ヲ	A6
ン	DD								
ァ	A7	ィ	A8	ゥ	A9	ェ	AA	ォ	AB
				ッ	AF				
ャ	AC			ュ	AD			ョ	AE
゛	DE	゜	DF	。	A1	「	A2	」	A3
、	A4	・	A5	～	A0	ー	B0		

- Table of Kanji and Hiragana codes

ア	ぁ	6421	あ	6422	阿	7024	哀	7025	愛	7026	挨	7027	逢	7029	悪	702D
	握	702E	旭	7030	圧	7035	斡	7036	扱	7037	宛	7038	安	7042	暗	7045
	案	7046	闇	7047	鞍	7048										
イ	ぃ	6423	い	6424	以	704A	伊	704B	位	704C	依	704D	偉	704E	囲	704F
	委	7051	威	7052	意	7055	慰	7056	易	7057	椅	7058	為	7059	異	705B
	移	705C	維	705D	緯	705E	胃	705F	萎	7060	衣	7061	違	7063	遺	7064
	医	7065	井	7066	域	7068	育	7069	一	706C	稲	7070	印	7075	員	7077
	因	7078	引	707A	飲	707B	院	7121	陰	7122	隠	7123				
ウ	ぅ	6425	う	6426	右	7126	宇	7127	羽	7129	雨	712B	渦	7132	嘘	7133
	唄	7134	浦	313A	瓜	713B	噂	713D	運	713F	雲	7140				
エ	ぇ	6427	え	6428	営	7144	影	7146	映	7147	栄	7149	永	714A	泳	714B
	洩	714C	英	7151	衛	7152	鋭	7154	液	7155	益	7157	駅	7158	越	715B
	閲	715C	円	715F	園	7160	宴	7163	延	7164	援	7167	沿	7168	演	7169
	炎	716A	煙	716C	縁	716F	遠	7173	鉛	7174	塩	7176				
オ	ぉ	6429	お	642A	汚	7178	凹	717A	央	717B	奥	717C	往	717D	応	717E
	押	7221	横	7223	欧	7224	王	7226	黄	722B	岡	722C	沖	722D	億	722F
	屋	7230	憶	7231	臆	7232	牡	7234	乙	7235	恩	7238	温	7239	穏	723A
	音	723B														

カ	か	642B	が	642C	下	723C	化	723D	仮	723E	何	723F	価	7241
	佳	7242												
	加	7243	可	7244	夏	7246	家	7248	科	724A	暇	724B	果	724C
	架	724D												
	歌	724E	河	724F	火	7250	稼	7254	箇	7255	花	7256	荷	7259
	華	725A												
	菓	725B	課	725D	貨	725F	過	7261	我	7266	牙	7267	画	7268
	芽	726A												
	賀	726C	雅	726D	介	7270	会	7271	解	7272	回	7273	壊	7275
	廻	7276												
	快	7277	怪	7278	懐	727B	拐	727D	改	727E	械	7323	海	7324
	灰	7325												
	界	7326	皆	7327	絵	7328	開	732B	階	732C	貝	732D	劾	732E
	外	7330												
	害	7332	慨	7334	概	7335	涯	7336	街	7339	該	733A	垣	7340
	各	7346												
	拡	7348	格	734A	核	734B	殻	734C	獲	734D	確	734E	穫	734F
	覚	7350												
	角	7351	較	7353	郭	7354	閣	7355	隔	7356	革	7357	学	7358
	楽	735A												
	額	735B	掛	735D	笠	735E	潟	7363	割	7364	括	7367	活	7368
	渇	7369												
	滑	736A	株	7374	刈	7422	乾	7425	冠	7427	寒	7428	刊	7429
	勧	742B												
	巻	742C	喚	742D	完	7430	官	7431	寛	7432	干	7433	幹	7434
	患	7435												
	感	7436	慣	7437	換	7439	敢	743A	歓	743F	汗	7440	漢	7441
	環	7444												
	甘	7445	監	7446	看	7447	管	7449	簡	744A	緩	744B	缶	744C
	肝	744E												
	観	7451	貫	7453	還	7454	鑑	7455	間	7456	閑	7457	関	7458
	陥	7459												
	韓	745A	館	745B	丸	745D	含	745E	岸	745F	眼	7463	岩	7464
	顔	7469												
	願	746A												

ク	く	642F	ぐ	6430	九	7665	句	7667	区	7668	矩	766B	苦	766C	駆	766E
	具	7671	愚	7672	空	7675	偶	7676	遇	7678	隅	7679	屑	767D	屈	767E
	掘	7721	靴	7724	熊	7727	繰	772B	君	772F	訓	7731	群	7732	軍	7733
	郡	7734														
ケ	け	6431	げ	6432	係	7738	傾	7739	刑	773A	兄	773B	啓	773C	型	773F
	契	7740	形	7741	径	7742	慶	7744	憩	7746	掲	7747	携	7748	敬	7749
	景	774A	系	774F	経	7750	継	7751	茎	7754	計	7757	警	7759	軽	775A
	芸	775D	迎	775E	劇	7760	撃	7762	激	7763	隙	7764	桁	7765	傑	7766
	欠	7767	決	7768	潔	7769	穴	776A	結	776B	血	776C	月	776E	件	776F
	倹	7770	健	7772	兼	7773	券	7774	剣	7775	圏	7777	堅	7778	嫌	7779
	建	777A	憲	777B	懸	777C	拳	777D	検	7821	権	7822	犬	7824	献	7825
	研	7826	絹	7828	県	7829	肩	782A	見	782B	謙	782C	軒	782E	鍵	7830
	険	7831	験	7833	元	7835	原	7836	厳	7837	幻	7838	弦	7839	減	783A
	源	783B	現	783D	言	7840	限	7842								
コ	こ	6433	ご	6434	個	7844	古	7845	呼	7846	固	7847	己	784A	庫	784B
	弧	784C	戸	784D	故	784E	湖	7850	狐	7851	誇	7858	雇	785B	顧	785C
	五	785E	互	785F	午	7861	娯	7864	後	7865	御	7866	語	786C	誤	786D
	護	786E	交	7872	侯	7874	候	7875	光	7877	公	7878	功	7879	効	787A
	勾	787B	厚	787C	口	787D	向	787E	喉	7922	坑	7923	好	7925	孔	7926
	孝	7927	工	7929	巧	792A	幸	792C	広	792D	康	792F	弘	7930	抗	7933
	拘	7934	控	7935	攻	7936	更	7939	校	793B	構	793D	江	793E	洪	793F
	港	7941	溝	7942	甲	7943	硬	7945	稿	7946	紅	7948	絞	794A	綱	794B
	耕	794C	考	794D	肯	794E	航	7952	荒	7953	行	7954	衡	7955	講	7956
	貢	7957	購	7958	郊	7959	鉱	795B	鋼	795D	降	795F	項	7960	香	7961
	高	7962	剛	7964	号	7966	合	7967	克	796E	刻	796F	告	7970	国	7971
	穀	7972	酷	7973	黒	7975	腰	7978	骨	797C	込	797E	此	7A21	頃	7A22
	今	7A23	困	7A24	婚	7A27	根	7A2C	混	7A2E						
サ	さ	6435	ざ	6436	左	7A38	差	7A39	査	7A3A	砂	7A3D	鎖	7A3F	座	7A42
	挫	7A43	債	7A44	催	7A45	再	7A46	最	7A47	妻	7A4A	彩	7A4C	才	7A4D
	採	7A4E	栽	7A4F	済	7A51	災	7A52	砕	7A55	祭	7A57	細	7A59	菜	7A5A
	裁	7A5B	載	7A5C	際	7A5D	剤	7A5E	在	7A5F	材	7A60	罪	7A61	財	7A62
	坂	7A64	阪	7A65	咲	7A69	崎	7A6A	作	7A6E	削	7A6F	咋	7A72	柵	7A74
	策	7A76	索	7A77	錯	7A78	桜	7A79	冊	7A7D	刷	7A7E	察	7B21	搾	7B22
	撮	7B23	擦	7B24	札	7B25	殺	7B26	雑	7B28	皿	7B2E	三	7B30	傘	7B31
	参	7B32	山	7B33	撒	7B35	散	7B36	産	7B3A	算	7B3B	讃	7B3E	賛	7B3F
	酸	7B40	残	7B44												

シ	し	6437	じ	6438	仕	7B45	伺	7B47	使	7B48	刺	7B49	史	7B4B	四	7B4D
	士	7B4E	始	7B4F	姉	7B50	姿	7B51	子	7B52	市	7B54	師	7B55	志	7B56
	思	7B57	指	7B58	支	7B59	施	7B5C	旨	7B5D	枝	7B5E	止	7B5F	死	7B60
	私	7B64	糸	7B65	紙	7B66	紫	7B67	脂	7B69	至	7B6A	視	7B6B	詞	7B6C
	詩	7B6D	試	7B6E	誌	7B6F	資	7B71	歯	7B75	事	7B76	似	7B77	字	7B7A
	寺	7B7B	持	7B7D	時	7B7E	次	7C21	治	7C23	磁	7C27	示	7C28	耳	7C2A
	自	7C2B	辞	7C2D	式	7C30	識	7C31	軸	7C34	七	7C37	失	7C3A	室	7C3C
	湿	7C3E	質	7C41	実	7C42	芝	7C47	縞	7C4A	写	7C4C	射	7C4D	捨	7C4E
	斜	7C50	煮	7C51	社	7C52	者	7C54	謝	7C55	車	7C56	借	7C5A	尺	7C5C
	釈	7C61	若	7C63	弱	7C65	主	7C67	取	7C68	守	7C69	手	7C6A	殊	7C6C
	狩	7C6D	種	7C6F	趣	7C71	酒	7C72	首	7C73	受	7C75	寿	7C77	授	7C78
	樹	7C79	需	7C7B	収	7C7D	周	7C7E	就	7D22	修	7D24	秀	7D28	秋	7D29
	終	7D2A	習	7D2C	臭	7D2D	舟	7D2E	衆	7D30	襲	7D31	蹴	7D33	週	7D35
	集	7D38	住	7D3B	充	7D3C	十	7D3D	従	7D3E	柔	7D40	渋	7D42	縦	7D44
	重	7D45	宿	7D49	祝	7D4B	縮	7D4C	熟	7D4F	出	7D50	術	7D51	述	7D52
	春	7D55	瞬	7D56	準	7D60	盾	7D62	純	7D63	巡	7D64	順	7D67	処	7D68
	初	7D69	所	7D6A	暑	7D6B	緒	7D6F	署	7D70	書	7D71	諸	7D74	助	7D75
	叙	7D76	女	7D77	序	7D78	除	7D7C	傷	7D7D	勝	7D21	商	7E26	唱	7E27
	奨	7E29	将	7E2D	小	7E2E	少	7E2F	尚	7E30	床	7E32	承	7E35	招	7E37
	掌	7E38	昇	7E3A	昭	7E3C	消	7E43	渉	7E44	焼	7E46	焦	7E47	照	7E48
	省	7E4A	称	7E4E	章	7E4F	笑	7E50	紹	7E52	衝	7E57	訟	7E59	証	7E5A
	詳	7E5C	象	7E5D	賞	7E5E	鐘	7E62	障	7E63	上	7E65	乗	7E68	剰	7E6A
	城	7E6B	場	7E6C	壌	7E6D	常	7E6F	情	7E70	条	7E72	浄	7E74	状	7E75
	蒸	7E78	錠	7E7B	飾	7E7E	植	7F22	織	7F25	職	7F26	色	7F27	触	7F28
	食	7F29	伸	7F2D	信	7F2E	侵	7F2F	唇	7F30	寝	7F32	審	7F33	心	7F34
	振	7F36	新	7F37	森	7F39	浸	7F3B	深	7F3C	申	7F3D	真	7F3F	神	7F40
	紳	7F42	芯	7F44	親	7F46	診	7F47	身	7F48	辛	7F49	進	7F4A	針	7F4B
	震	7F4C	人	7F4D	刃	7F4F	尽	7F54	陣	7F58						
ス	す	6439	ず	643A	須	7F5C	酢	7F5D	図	7F5E	吹	7F61	垂	7F62	推	7F64
	水	7F65	粋	7F68	遂	7F6B	酔	7F6C	錐	7F6D	数	7F74	据	7F78	杉	7F79
	裾	7F7E	澄	8021	寸	8023										

セ	せ	643B	ぜ	643C	世	8024	瀬	8025	是	8027	制	8029	勢	802A	征	802C
	性	802D	成	802E	政	802F	整	8030	星	8031	晴	8032	正	8035	清	8036
	生	8038	盛	8039	精	803A	聖	803B	声	803C	製	803D	西	803E	誠	803F
	誓	8040	請	8041	青	8044	静	8045	税	8047	席	804A	昔	804E	析	804F
	石	8050	積	8051	籍	8052	績	8053	責	8055	赤	8056	跡	8057	切	805A
	接	805C	折	805E	設	805F	節	8061	説	8062	雪	8063	絶	8064	舌	8065
	先	8068	千	8069	占	806A	宣	806B	専	806C	尖	806D	川	806E	戦	806F
	扇	8070	栓	8072	泉	8074	浅	8075	洗	8076	染	8077	潜	8078	旋	807B
	線	807E	繊	8121	船	8125	選	812A	銑	812D	鮮	812F	前	8130	善	8131
	漸	8132	然	8133	全	8134	繕	8136								
ソ	そ	643D	ぞ	643E	遡	813A	礎	8143	粗	8146	素	8147	組	8148	訴	814A
	阻	814B	創	814F	双	8150	倉	8152	奏	8155	層	8158	想	815B	捜	815C
	掃	815D	挿	815E	操	8160	早	8161	巣	8163	争	8168	相	816A	窓	816B
	総	816D	草	8170	装	8175	走	8176	送	8177	騒	817B	像	817C	増	817D
	臓	8221	蔵	8222	贈	8223	造	8224	促	8225	側	8226	則	8227	即	8228
	息	8229	束	822B	測	822C	足	822D	速	822E	俗	822F	属	8230	族	8232
	続	8233	卒	8234	其	8236	揃	8237	存	8238	尊	823A	損	823B	村	823C
タ	た	643F	だ	6440	他	823E	多	823F	太	8240	詑	8242	堕	8244	妥	8245
	惰	8246	打	8247	体	824E	対	8250	耐	8251	帯	8253	待	8254	怠	8255
	態	8256	戴	8257	替	8258	滞	825A	袋	825E	貸	825F	退	8260	隊	8262
	代	8265	台	8266	大	8267	第	8268	題	826A	滝	826C	卓	826E	宅	8270
	択	8272	拓	8273	濯	8275	託	8277	濁	8279	諾	827A	叩	8231	達	8323
	奪	8325	脱	8326	棚	832A	谷	832B	誰	832F	単	8331	嘆	8332	担	8334
	探	8335	旦	8336	淡	8338	炭	833A	短	833B	端	833C	誕	8342	団	8344
	弾	8346	断	8347	暖	8348	段	834A	男	834B	談	834C				
チ	ち	6441	ぢ	6442	値	834D	知	834E	地	834F	恥	8351	池	8353	置	8356
	致	8357	遅	8359	馳	835A	築	835B	畜	835C	竹	835D	筑	835E	秩	8361
	茶	8363	着	8365	中	8366	仲	8367	宙	8368	忠	8369	抽	836A	昼	836B
	柱	836C	注	836D	虫	836E	鋳	8372	駐	8373	貯	8379	丁	837A	兆	837B
	帳	8422	庁	8423	張	8425	彫	8426	徴	8427	懲	8428	挑	8429	朝	842B
	町	842E	脹	8431	腸	8432	調	8434	超	8436	跳	8437	長	8439	頂	843A
	鳥	843B	直	843E	沈	8440	珍	8441	賃	8442						
ツ	っ	6443	つ	6444	づ	6445	墜	8446	追	8449	痛	844B	通	844C	塚	844D
	爪	845E	吊	845F	釣	8460										

テ	て	6446	で	6447	低	8463	停	8464	定	846A	底	846C	庭	846D	廷	846E
	抵	8471	提	8473	程	8478	締	8479	訂	847B	釘	8523	泥	8525	摘	8526
	敵	8528	滴	8529	的	852A	笛	852B	適	852C	撤	8531	鉄	8534	典	8535
	天	8537	展	8538	店	8539	添	853A	貼	853D	転	853E	点	8540	伝	8541
	殿	8542	田	8544	電	8545										
ト	と	6448	ど	6449	吐	8547	塗	8549	徒	854C	登	8550	途	8553	都	8554
	砥	8556	努	8558	度	8559	土	855A	怒	855C	倒	855D	党	855E	冬	855F
	凍	8560	刀	8561	島	8567	投	856A	東	856C	盗	8570	湯	8572	灯	8574
	当	8576	等	8579	答	857A	筒	857B	糖	857C	統	857D	到	857E	藤	8623
	討	8624	踏	8627	逃	8628	透	8629	陶	862B	頭	862C	闘	862E	働	862F
	動	8630	同	8631	堂	8632	導	8633	胴	8639	道	863B	銅	863C	峠	863D
	得	8640	徳	8641	特	8643	督	8644	毒	8647	独	8648	読	8649	凸	864C
	突	864D	届	864F	畳	865E	鈍	865F								
ナ	な	644A	内	8662	謎	8666	鍋	8669	馴	866B	縄	866C	南	866E	軟	8670
	難	8671														
ニ	に	644B	二	8673	匂	8677	肉	8679	日	867C	乳	867D	入	867E	尿	8722
	任	8724	認	8727												
ヌ	ぬ	644C														
ネ	ね	644D	熱	872E	年	872F	念	8730	燃	8733	粘	8734				
ノ	の	644E	悩	873A	濃	873B	納	873C	能	873D	脳	873E	農	8740		
ハ	は	644F	ば	6450	ぱ	6451	把	8744	覇	8746	波	8748	派	8749	破	874B
	馬	874F	廃	8751	拝	8752	排	8753	敗	8754	杯	8755	背	8758	肺	8759
	配	875B	倍	875C	媒	875E	買	8763	売	8764	博	876E	拍	876F	泊	8771
	白	8772	舶	8775	薄	8776	爆	877A	縛	877B	麦	877E	箱	8822	肌	8829
	畑	882A	八	882C	発	882F	髪	8831	罰	8833	抜	8834	閥	8836	伴	883C
	判	883D	半	883E	反	883F	搬	8842	板	8844	汎	8846	版	8847	犯	8848
	班	8849	繁	884B	般	884C	販	884E	範	884F	飯	8853	番	8856	盤	8857
ヒ	ひ	6452	び	6453	ぴ	6454	否	885D	彼	8860	悲	8861	扉	8862	批	8863
	比	8866	泌	8867	疲	8868	皮	8869	秘	886B	肥	886E	被	886F	費	8871
	避	8872	非	8873	飛	8874	備	8877	尾	8878	微	8879	美	887E	鼻	8921
	匹	8924	菱	8929	必	892C	筆	892E	百	8934	俵	8936	標	8938	氷	8939
	票	893C	表	893D	評	893E	描	8941	病	8942	秒	8943	品	894A	浜	894D
	貧	894F	敏	8952												

フ	ふ	6455	ぶ	6456	ぷ	6457	不	8954	付	8955	夫	8957	婦	8958	富	8959
	布	895B	府	895C	怖	895D	敷	895F	普	8961	浮	8962	父	8963	符	8964
	腐	8965	負	8969	武	8970	舞	8971	部	8974	封	8975	風	8977	伏	897A
	副	897B	復	897C	幅	897D	服	897E	福	8A21	腹	8A22	複	8A23	払	8A27
	沸	8A28	仏	8A29	物	8A2A	分	8A2C	噴	8A2E	憤	8A30	奮	8A33	粉	8A34
	紛	8A36	文	8A38	聞	8A39										
ヘ	へ	6458	べ	6459	ぺ	645A	丙	8A3A	併	8A3B	兵	8A3C	弊	8A3E	平	8A3F
	柄	8A41	並	8A42	閉	8A44	米	8A46	頁	8A47	壁	8A49	癖	8A4A	別	8A4C
	偏	8A50	変	8A51	片	8A52	編	8A54	辺	8A55	返	8A56	便	8A58	勉	8A59
	弁	8A5B														
ホ	ほ	645B	ぼ	645C	ぽ	645D	保	8A5D	捕	8A61	歩	8A62	補	8A64	募	8A67
	墓	8A68	慕	8A69	母	8A6C	簿	8A6D	倣	8A6F	包	8A71	報	8A73	宝	8A75
	崩	8A78	捧	8A7B	放	8A7C	方	8A7D	法	8B21	泡	8B22	縫	8B25	胞	8B26
	芳	8B27	訪	8B2C	豊	8B2D	飽	8B30	乏	8B33	亡	8B34	傍	8B35	剖	8B36
	妨	8B38	帽	8B39	忘	8B3A	忙	8B3B	房	8B3C	暴	8B3D	望	8B3E	棒	8B40
	紡	8B42	肪	8B43	膨	8B44	防	8B49	北	8B4C	僕	8B4D	撲	8B50	釦	8B55
	没	8B57	本	8B5C	翻	8B5D										
マ	ま	645E	摩	8B60	磨	8B61	魔	8B62	枚	8B67	毎	8B68	幕	8B6B	膜	8B6C
	末	8B76	迄	8B78	万	8B7C	満	8B7E								
ミ	み	645F	味	8C23	未	8C24	魅	8C25	密	8C29	脈	8C2E	妙	8C2F	民	8C31
ム	む	6460	務	8C33	夢	8C34	無	8C35	矛	8C37	霧	8C38				
メ	め	6461	名	8C3E	命	8C3F	明	8C40	盟	8C41	迷	8C42	鳴	8C44	滅	8C47
	免	8C48	綿	8C4A	面	8C4C										
モ	も	6462	模	8C4F	茂	8C50	毛	8C53	盲	8C55	網	8C56	耗	8C57	木	8C5A
	黙	8C5B	目	8C5C	戻	8C61	問	8C64	紋	8C66	門	8C67				
ヤ	ゃ	6463	や	6464	冶	8C6A	夜	8C6B	野	8C6E	矢	8C70	役	8C72	約	8C73
	薬	8C74	訳	8C75	躍	8C76										
ユ	ゅ	6465	ゆ	6466	油	8C7D	諭	8D21	輸	8D22	優	8D25	勇	8D26	友	8D27
	有	8D2D	由	8D33	誘	8D36	遊	8D37	郵	8D39	融	8D3B				
ヨ	ょ	6467	よ	6468	予	8D3D	余	8D3E	与	8D3F	誉	8D40	預	8D42	幼	8D44
	容	8D46	揚	8D48	揺	8D49	曜	8D4B	様	8D4D	洋	8D4E	溶	8D4F	用	8D51
	葉	8D55	要	8D57	踊	8D59	陽	8D5B	養	8D5C	抑	8D5E	浴	8D61	翼	8D63
ラ	ら	6469	螺	8D66	裸	8D67	来	8D68	頼	8D6A	雷	8D6B	絡	8D6D	落	8D6E
	乱	8D70	卵	8D71	欄	8D73	覧	8D77								

リ	り	646A	利	8D78	理	8D7D	裏	8E22	里	8E24	離	8E25	陸	8E26	律	8E27
	率	8E28	立	8E29	略	8E2C	流	8E2E	留	8E31	粒	8E33	隆	8E34	慮	8E38
	旅	8E39	虜	8E3A	了	8E3B	両	8E3E	寮	8E40	料	8E41	療	8E45	稜	8E47
	良	8E49	量	8E4C	領	8E4E	力	8E4F	緑	8E50	林	8E53	臨	8E57	輪	8E58
	隣	8E59														
ル	る	646B	塁	8E5D	涙	8E5E	累	8E5F	類	8E60						
レ	れ	646C	令	8E61	例	8E63	冷	8E64	励	8E65	礼	8E69	鈴	8E6B	隷	8E6C
	霊	8E6E	暦	8E71	歴	8E72	列	8E73	劣	8E74	烈	8E75	裂	8E76	恋	8E78
	練	8E7D	連	8F22												
ロ	ろ	646D	路	8F29	労	8F2B	浪	8F32	漏	8F33	老	8F37	郎	8F3A	六	8F3B
	録	8F3F	論	8F40												
ワ	ゎ	646E	わ	646F	和	8F42	話	8F43	歪	8F44	脇	8F46	惑	8F47	枠	8F48
	詫	8F4D	湾	8F51	腕	8F53										
ヲ	を	6472														
ン	ん	6473														
Special symbol	α	6641	β	6642	→	6F40	↗	6F41	↑	6F42	↖	6F43	←	6F44	↙	6F45
	↓	6F46	↘	6F47	↻	6F48	↺	6F49	⌒	6F4A	∩	6F4B	▮	6F4C	▿	6F50
	▿▿	6F51	▿▿▿	6F52	▿▿▿▿	6F53										

F ALARM LIST

F.1 PS ALARM (ALARMS RELATED TO PROGRAM)

Number	Message	Contents
PS0001	AXIS CONTROL MODE ILLEGAL	Illegal axis control mode
PS0003	TOO MANY DIGIT	Data entered with more digits than permitted in the NC instruction word. The number of permissible digits varies according to the function and the word.
PS0006	ILLEGAL USE OF MINUS SIGN	A minus sign (-) was specified at an NC instruction word or system variable where no minus signal may be specified.
PS0007	ILLEGAL USE OF DECIMAL POINT	A decimal point (.) was specified at an address where no decimal point may be specified, or two decimal points were specified.
PS0010	IMPROPER G-CODE	An illegal Code was specified.
PS0011	IMPROPER NC-ADDRESS	An illegal address was specified, or parameter 1020 is not set.
PS0012	INVALID BREAK POINT OF WORDS	NC word(s) address + numerical value not in word format. This alarm is also generated when a custom macro does not contain a reserved word, or does not conform to the syntax.
PS0013	ILLEGAL POS. OF PROGRAM NO.	Address O or N is specified in an illegal location (e.g. after a macro statement).
PS0014	ILLEGAL FORMAT OF PROGRAM NO.	Address O or N is not followed by a number.
PS0015	TOO MANY WORD IN ONE BLOCK	The number of words in a block exceeds the maximum. The maximum is 26 words. However, this figure varies according to NC options. Divide the instruction word into two blocks.
PS0016	EOB NOT FOUND	EOB (End of Block) code is missing at the end of a program input in the MDI mode.
PS0017	ILLEGAL MODE FOR GOTO/WHILE/DO	A GOTO statement or WHILE-DO statement was found in the main program in the MDI or DNC mode.
PS0059	COMMAND IN BUFFERING MODE	The manual intervention compensation request signal MIGET became "1" when a previous block was found during automatic operation. To input the manual intervention compensation during automatic operation, a sequence for manipulating the manual intervention compensation request signal MIGET is required in an M code instruction without buffering.
PS0060	SEQUENCE NUMBER NOT FOUND	The specified sequence No. was not found during sequence number search. The sequence No. specified as the jump destination in GOTO-- and M99P-- was not found.
PS0076	PROGRAM NOT FOUND	The specified program is not found in the subprogram call, macro call or graphic copy. The M, G, T or S codes are called by a P instruction other than that in an M98, G65, G66, G66.1 or interrupt type custom macro, and a program is called by a 2nd auxiliary function code. This alarm is also generated when a program is not found by these calls.
PS0077	PROGRAM IN USE	An attempt was made in the foreground to execute a program being edited in the background. The currently edited program cannot be executed, so end editing and restart program execution.
PS0079	PROGRAM FILE NOT FOUND	The program of the specified file No. is not registered in an external device. (external device subprogram call)
PS0080	DUPLICATE DEVICE SUB PROGRAM CALL	Another external device subprogram call was made from a subprogram after the subprogram called by the external device subprogram call.
PS0081	EXT DEVICE SUB PROGRAM CALL MODE ERROR	The external device subprogram call is not possible in this mode.

Number	Message	Contents
PS0090	DUPLICATE NC,MACRO STATEMENT	An NC statement and macro statement were specified in the same block.
PS0091	DUPLICATE SUB-CALL WORD	More than one subprogram call instruction was specified in the same block.
PS0092	DUPLICATE MACRO-CALL WORD	More than one macro call instruction was specified in the same block.
PS0093	DUPLICATE NC-WORD & M99	An address other than O, N, P or L was specified in the same block as M99 during the macro modal call state.
PS0094	USE 'G' AS ARGUMENT	Address G was used as the argument of a custom macro call. Address G can be used as the argument in individual block call (G66.1).
PS0095	TOO MANY TYPE-2 ARGUMENT	More than ten sets of I, J and K arguments were specified in the type-II arguments (A, B, C, I, J, K, I, J, K, . . .) for custom macros.
PS0096	ILLEGAL VARIABLE NAME	An illegal variable name was specified. A code that cannot be specified as a variable name was specified. [#_OFSxx] does not match the tool offset memory option configuration.
PS0097	TOO LONG VARIABLE NAME	The specified variable name is too long.
PS0098	NO VARIABLE NAME	The specified variable name cannot be used as it is not registered.
PS0099	ILLLEGAL SUFFIX [ ]	A suffix was not specified to a variable name that required a suffix enclosed by [ ]. A suffix was specified to a variable name that did not require a suffix enclosed by [ ]. The value enclosed by the specified [ ] was out of range.
PS0100	CANCEL WITHOUT MODAL CALL	Call mode cancel (G67) was specified even though macro continuous-state call mode (G66) was not in effect.
PS0101	ILLEGAL CNC STATEMENT IRT.	An interrupt was made in a state where a custom macro interrupt containing a move instruction could not be executed.
PS0110	OVERFLOW :INTEGER	An integer went out of range during arithmetic calculations.
PS0111	OVERFLOW :FLOATING	A decimal point (floating point number format data) went out of range during arithmetic calculations.
PS0112	ZERO DIVIDE	An attempt was made to divide by zero in a custom macro.
PS0114	VARIABLE NO. OUT OF RANGE	An illegal No. was specified in a local variable, common variable or a system variable in a custom macro. A non-existent custom macro variable No. was specified in the EGB axis skip function (G31.8), or there are not enough custom macro variables for storing the skip position.
PS0115	READ PROTECTED VARIABLE	An attempt was made in a custom macro to use on the right side of an expression a variable that can only be used on the left side of an expression.
PS0116	WRITE PROTECTED VARIABLE	An attempt was made in a custom macro to use on the left side of an expression a variable that can only be used on the right side of an expression.
PS0118	TOO MANY BRACKET NESTING	Too many brackets "[]" were nested in a custom macro. The nesting level including function brackets is 5.
PS0119	ARGUMENT VALUE OUT OF RANGE	The value of an argument in a custom macro function is out of range.
PS0120	ILLEGAL ARGUMENT FORMAT	The specified argument in the argument function (ATAN, POW) is in error.
PS0121	OO MANY SUB,MACRO NESTING	The total number of subprogram and macro calls exceeds the permissible range. Another subprogram call was executed during an external memory subprogram call.
PS0122	TOO MANY MACRO NESTING	Too many macro calls were nested in a custom macro. The nesting level is 5.
PS0123	MISSING END STATEMENT	The END instruction corresponding to the DO instruction was missing in a custom macro.

Number	Message	Contents
PS0124	MISSING DO STATEMENT	The DO instruction corresponding to the END instruction was missing in a custom macro.
PS0125	ILLEGAL EXPRESSION FORMAT	The format used in an expression in a custom macro statement is in error. The parameter tape format is in error.
PS0126	ILLEGAL LOOP NUMBER	DO and END Nos. in a custom macro are in error, or exceed the permissible range (valid range: 1 to 3).
PS0128	SEQUENCE NUMBER OUT OF RANGE	The jump destination sequence No. in a custom macro statement GOTO instruction was out of range (valid range: 1 to 99999999).
PS0131	MISSING OPEN BRACKET	The number of left brackets ([) is less than the number of right brackets (]) in a custom macro statement.
PS0132	MISSING CLOSE BRACKET	The number of right brackets (]) is less than the number of left brackets ([) in a custom macro statement.
PS0133	MISSING '='	An equal sign (=) is missing in the arithmetic calculation instruction in a custom macro statement.
PS0134	MISSING ','	A delimiter (,) is missing in a custom macro statement.
PS0135	MACRO STATEMENT FORMAT ERROR	The format used in a macro statement in a custom macro is in error.
PS0137	IF STATEMENT FORMAT ERROR	The format used in the IF statement in a custom macro is in error.
PS0138	WHILE STATEMENT FORMAT ERROR	The format used in the WHILE statement in a custom macro is in error.
PS0139	SETVN STATEMENT FORMAT ERROR	The format used in the SETVN statement in a custom macro is in error.
PS0141	ILLEGAL CHARACTER IN VAR. NAME	The SETVN statement in a custom macro contacts a character that cannot be used in a variable name.
PS0142	TOO LONG V-NAME (SETVN)	The variable name used in a SETVN statement in a custom macro exceeds 8 characters.
PS0143	BPRNT/DPRNT STATEMENT FORMAT ERROR	The format used in the BPRINT statement or DPRINT statement is in error.
PS0144	G10 FORMAT ERROR	The G10 L No. contains no relevant data input or corresponding option. Data setting address P or R is not specified. An address not relating to the data setting is specified. Which address to specify varies according to the L No. The sign, decimal point or range of the specified address are in error.
PS0145	G10.1 TIME OUT	The response to a G10.1 instruction was not received from the PMC within the specified time limit.
PS0146	G10.1 FORMAT ERROR	The G10.1 instruction format is in error.
PS0150	G31 FORMAT ERROR	• No axis is specified or tow or more axes are specified in the torque limit switch instruction (G31P98/P99). • The specified torque Q value in the torque limit switch instruction is out of range. The torque Q range is 1 to 99.
PS0151	CANNOT USE G31	The alarm occurs in the following cases: • The G31 code cannot be specified. The G code in group 07 (e.g. cutter compensation) is not canceled. • The torque limit was not specified by the torque limit skip instruction (G31 P98/P99). Either specify the torque limit in the PMC window, or specify torque limit override by address Q.
PS0152	G31.9/G31.8 FORMAT ERROR	The format of the G31.9 or G31.8 block is erroneous in the following cases: • The axis was not specified in the G31.9 or G31.8 block. • Multiple axes were specified in the G31.9 or G31.8 block. • The P code was specified in the G31.9 or G31.8 block.
PS0153	CANNOT USE G31.9	G31.9 cannot be specified in this modal state. This alarm is also generated when G31.9 is specified when a group 07 G code (e.g. cutter compensation) is not canceled.

Number	Message	Contents
PS0160	COMMAND DATA OVERFLOW	An overflow occurred in the storage length of the CNC internal data. This alarm is also generated when the result of internal calculation of scaling, coordinate rotation and cylindrical interpolation overflows the data storage. It also is generated during input of the manual intervention amount.
PS0180	ALL PARALLEL AXES IN PARKING	All of the axis specified for automatic operation are parked.
PS0181	ZERO RETURN NOT FINISHED	A move instruction was issued to an axis in which the zero return instruction was instructed once after the power was turned ON. Execute operation after zero return by manual operation or the G28 code. This alarm can be suppressed for the machine-locked axis by setting parameter No. 1200#6 is set to "1".
PS0182	CIRCLE CUT IN RAPID (F0)	F0 (rapid traverse in inverse feed or feed specified by an F code with 1-digit number) was specified during circular interpolation (G02, G03) or involute interpolation (G02.2, G03.2).
PS0183	TOO MANY SIMULTANEOUS AXES	A move command was specified for more axes than can be controlled by simultaneous axis control. Either add on the simultaneous axis control extension option, or divide the number of programmed move axes into two blocks.
PS0184	TOO LARGE DISTANCE	Due to compensation, point of intersection calculation, interpolation or similar reasons, a movement distance that exceeds the maximum permissible distance was specified. Check the programmed coordinates or compensation amounts.
PS0185	ZERO RETURN CHECK (G27) ERROR	The axis specified in G27 has not returned to zero. Reprogram so that the axis returns to zero.
PS0186	ILLEGAL PLANE SELECT	The plane selection instructions G17 to G19 are in error. Reprogram so that same 3 basic parallel axes are not specified simultaneously. This alarm is also generated when an axis that should not be specified for plane machining is specified, for example, for circular interpolation or involute interpolation. To enable programming of 3 or more axes, the helical interpolation option must be added to each of the relevant axes.
PS0187	FEED ZERO (COMMAND)	The cutting feedrate instructed by an F code has been set to 0. This alarm is also generated if the F code instructed for the S code is set extremely small in a rigid tapping instruction as the tool cannot cut at the programmed lead.
PS0188	PARAMETER ZERO (DRY RUN)	The dry run feedrate parameter No. 1410 or maximum cutting feedrate parameter No. 1422 for each axis has been set to 0.
PS0190	PARAMETER ZERO (CUT MAX)	The maximum cutting feedrate parameter No. 1422 has been set to 0.
PS0191	OVER TOLERANCE OF RADIUS	An arc was specified for which the difference in the radius at the start and end points exceeds the value set in parameter No. 2410. Check arc center codes I, J and K in the program. The tool path when parameter No. 2410 is set to a large value is spiral.
PS0193	ILLEGAL OFFSET NUMBER	An illegal offset No. was specified. This alarm is also generated when the tool shape offset No. exceeds the maximum number of tool offset sets in the case of tool offset memory B.

Number	Message	Contents
PS0194	ZERO RETURN END NOT ON REF	The axis specified in automatic zero return was not at the correct zero point when positioning was completed. Perform zero return from a point whose distance from the zero return start position to the zero point is 2 or more revolutions of the motor. Other probable causes are: - The positional deviation after triggering the deceleration dog is less than 128. - Insufficient voltage or malfunctioning pulse coder.
PS0195	ILLEGAL AXIS SELECTED (G96)	An illegal value was specified in P in a G96 block or parameter No. 5844.
PS0196	ILLEGAL DRILLING AXIS SELECTED	An illegal axis was specified for drilling in a canned cycle for drilling. If the zero point of the drilling axis is not specified or parallel axes are specified in a block containing a G code in a canned cycle, simultaneously specify the parallel axes for the drilling axis.
PS0200	PULSCODER INVALID ZERO RETURN	The grid position could not be calculated during grid reference position return using the grid system as the one-revolution signal was not received before leaving the deceleration dog. This alarm is also generated when the tool does not reach a feedrate that exceeds the servo error amount preset to parameter No. 1841 before the deceleration limit switch is left (deceleration signal *DEC returns to "1").
PS0202	NO F COMMAND AT G93	F codes in the inverse time specification mode (G93) are not handled as modal, and must be specified in individual blocks.
PS0213	ILLEGAL USE OF G12.1/G13.1	The axis No. of plane selection parameter No. 1032 (linear axis) and No. 1033 (axis of rotation) in the polar coordinate interpolation mode is out of range (1 to number of controlled axes).
PS0214	ILLEGAL USE OF G-CODE	The modal G code group contains an illegal G code in the polar coordinate interpolation mode or when a mode was canceled. Only the following G codes are allowed: G40, G50, G69.1 An illegal G code was specified while in the polar coordinate interpolation mode. The following C codes are not allowed: G27, G28, G30, G30.1, G31 to G31.4, G37 to G387.3, G52, G92, G53, G17 to G19, G81 to G89, G68 In the 01 group, G codes other than G01, G02, G03, G02.2 and G03.2 cannot be specified.
PS0215	ILLEGAL COMMAND IN G10.6	The retract command was specified in the long axis for threading when retract was started by the threading block.
PS0217	ILLEGAL OFFSET VALUE	Illegal offset No.
PS0223	ILLEGAL SPINDLE SELECT	An attempt was made to execute an instruction that uses the spindle although the spindle to be controlled has not been set correctly.
PS0270	CRC:START_UP /CANCEL BY CIRCLE	An attempt was made to execute the cutter compensation startup or cancel block in the circular interpolation mode or involute interpolation mode.
PS0271	CRC:PLANE CHANGE	An attempt was made to change the plane in the cutter compensation mode. To change the plane, cancel the cutter compensation mode.

Number	Message	Contents
PS0272	CRC:INTERFERENCE	The depth of the cut is too great during cutter compensation. Check the program. The criteria for judging interference are as follows: (1) The direction of movement of the programmed block differs from the direction of movement of the corresponding tool center path block by 90° or more or 270° or less. The check in this case can be disabled by setting CNC parameter No. 6001#1 to "1". (2) In the case of an arc, the difference in angle between the start and end points of the programmed block differs by 180° or more with the difference in angle between the start and end points of the corresponding tool center path block.
PS0273	CRC:MOTION IN G39	Corner circular interpolation (G93) during cutter compensation has been specified not as an individual instruction but together with a move instruction.
PS0275	CRC:MDI MODE	Cutter compensation has been specified in the MDI mode. This alarm is also generated when the AIM parameter AIM No. 6005#1 is set to 1.
PS0276	CRC:NO INTERSECTION	There is not point of intersection of the compensated tool center path during cutter compensation.
PS0277	CRC:NO AVOIDANCE	Interference cannot be avoided as there no interference avoidance vector has been specified for the interference check avoidance function during cutter compensation.
PS0278	RC:DANGEROUS AVOIDANCE	Danger was judged when avoidance operation was executed by the interference check avoidance function during cutter compensation.
PS0279	CRC:INTERFERENCE TO AVD.	Interference occurred again even though the interference avoidance vector has been calculated by the interference check avoidance function during cutter compensation.
PS0280	ILLEGAL COMMAND IN SPIRAL	R was specified in the program block; 4 or more axes were specified in the program block; or the direction of the acceleration/deceleration vector on the specified radius is opposite to the direction of the end point.
PS0281	OVER TOLERANCE OF END POINT IN SPIRAL	The distance between the positions of the specified end point and the end point calculated from the programmed block exceeds the value set to parameter No. 2511. Either change the end point to be specified, or set a larger value to parameter No. 2511.
PS0282	ILLEGAL COMMAND IN 3-D OFFSET	An illegal G code is specified in three-dimensional tool compensation mode.
PS0283	ILLEGAL IJK IN 3-D OFFSET	When bit 0 (ONI) of parameter No. 6029 is set to 1, I, J, and K commands are specified without the decimal point in three-dimensional tool compensation mode.
PS0298	ILLEGAL INCH/METRIC CONVERSION	An error occurred during inch/metric switching.
PS0299	ILLEGAL ZERO RETURN COMMAND	A value other than 2 to 4 was specified to address P in the 2nd to 4th reference position return instruction.
PS0300	ILLEGAL ADDRESS	The axis No. address was specified even though the parameter is not an axis-type while loading parameters or pitch error compensation data from a tape or by entry of the G10 parameter. Axis No. cannot be specified in pitch error compensation data.
PS0301	MISSING ADDRESS	The axis No. was not specified even though the parameter is an axis-type while loading parameters or pitch error compensation data from a tape or by entry of the G10 parameter. Or, data No. address N, or setting data address P or R are not specified.

Number	Message	Contents
PS0302	ILLEGAL DATA NUMBER	A non-existent data No. was found while loading parameters or pitch error compensation data from a tape or by entry of the G10 parameter. This alarm is also generated when illegal word values are found. An invalid value was specified for address R in a pattern program command specific to a particular machining purpose on the high-speed, high-precision setting screen.
PS0303	ILLEGAL AXIS NUMBER	An axis No. address exceeding the maximum number of controlled axes was found while loading parameters from a tape or by entry of the G10 parameter.
PS0304	TOO MANY DIGIT	Data with too many digits was found while loading parameters or pitch error compensation data from a tape.
PS0305	DATA OUT OF RANGE	Out-of-range data was found while loading parameters or pitch error compensation data from a tape. The values of the data setting addresses corresponding to L Nos. during data input by G10 was out of range. This alarm is also generated when NC programming words contain out-of-range values.
PS0306	MISSING AXIS NUMBER	A parameter which requires an axis to be specified was found without an axis No. (address A) while loading parameters from a tape.
PS0307	ILLEGAL USE OF MINUS SIGN	Data with an illegal sign was found while loading parameters or pitch error compensation data from a tape, or by entry of the G10 parameter. A sign was specified to an address that does not support the use of signs.
PS0308	MISSING DATA	An address not followed by a numeric value was found while loading parameters or pitch error compensation data from a tape.
PS0330	ILLEGAL SPINDLE NUMBER	An spindle No. address exceeding the maximum number of controlled spindles was found while loading parameters from a tape or by entry of the G10 parameter.
PS0332	DATA WRITE LOCK ERROR	Could not load data while loading parameters, pitch error compensation data and work coordinate data from tape.
PS0333	DATA WRITE ERROR	Could not write data while loading data from tape.
PS0360	PARAMETER OUT OF RANGE (TLAC)	Illegal parameter setting. (Set value is out of range.)
PS0361	PARAMTER SETTING ERROR 1 (TLAC)	Illegal parameter setting. (axis of rotation setting)
PS0362	PARAMETER SETTING ERROR 2 (TLAC)	Illegal parameter setting (tool axis setting)
PS0370	PARAMETER SETTING ERROR (DM3H-1)	Out-of-range data was set during setting of the 3-dimensional handle feed parameter.
PS0371	PARAMETER SETTING ERROR (DM3H-2)	An illegal axis of rotation was set during setting of the 3-dimensional handle feed parameter.
PS0372	PARAMETAR SETTING ERROR (DM3H-3)	An illegal master axis was set during setting of the 3-dimensional handle feed parameter.
PS0373	PARAMETER SETTING ERROR (DM3H-4)	An illegal parallel axis or twin table was set during setting of the 3-dimensional handle feed parameter.
PS0400	PROGRAM NOT MATCH	The program in memory does not match the program stored on tape. Multiple programs cannot be matched continuously when parameter No. 2200#3 is set to "1". Set parameter No. 2200#3 to "0" before executing a match.
PS0410	G37 IMPROPER AXIS COMMAND	No axis or 2 or more axes were specified in the automatic tool measurement instruction (G37) block.
PS0411	G37 SPECIFIED WITH H CODE	An H code was specified in the same block as an automatic tool-length measurement instruction (G37).
PS0412	G37 OFFSET NO. UNASSIGNED	No H code was specified in the automatic tool measurement instruction (G37) block.

Number	Message	Contents
PS0415	G37 MEASURING POSITION REACHED SIGNAL IS NOT PROPERLY INPUT	The measurement position arrival signal became "1" before or after the area specified by parameter No. 7331 in automatic tool length measurement (G37).
PS0421	SETTING COMMAND ERROR	The instruction for setting tool data (G10L70 to G11, G10L71 to G11) is in error.
PS0422	NOT FOUND TOOL DATA	The specified tool No., pot No. , for the tool No. , tool length compensation data, and cutter compensation data has not been set.
PS0423	TOOL DATA WRITE ERROR	Writing occurred simultaneously on tool set data by a tool No.
PS0424	OVER MAXIMUM TOOL DATA	An attempt was made to set tool data exceeding the maximum number of tool data sets.
PS0425	NOT DELETE TOOL DATA IN OPERATION	Tool data cannot be deleted.
PS0429	MISSING VALUE AT CNR,CHF	The specified movement distance is less than the specified chamfering or corner rounding amount in a block in which chamfering or corner rounding is specified. Review the program instructions.
PS0431	MISSING MOVE AFTER CNR,CHF	The move direction or distance in the block following the block in which chamfering or cornering is specified is illegal. Review the program instructions.
PS0437	ILLEGAL LIFE GROUP NUMBER	A tool group number exceeded the maximum value. The tool group No. (P after G10 L3:) or the group No. assigned by the T code instruction for tool life management in the machining program exceeded the maximum value.
PS0438	GROUP NOT FOUND AT LIFE DATA	A tool group specified in the machining program was not set in the tool life management data.
PS0439	OVER MAXIMUM TOOL NUMBER	The number of tool specified in a single tool group exceeds the maximum allowed number.
PS0440	T COMMAND NOT FOUND	No T command was specified in a program which sets a tool group. The same block as M06 in the machining program does not contain a T command in tool change method D (parameters CT2=2, CT1=1: No. 7401#1/#0).
PS0441	NOT USING TOOL IN LIFE GROUP	An H99 instruction, D99 instruction or the H/D code specified in parameters (No. 7443, 7444) has been specified even though no tool that belongs to a group was being used.
PS0442	ILLEGAL T COMMAND AT M06	The tool group of the tool specified in the tool instruction (return tool group) after the M06 instruction in the machining program does not match the current tool group in tool change method A (parameters CT2=0, CT1=0: No. 7401#1/#0). This alarm can be suppressed by setting ABT parameter No. 7400#6 to "1".
PS0443	P,L COMMAND NOT FOUND	No P (group No.) or L (tool life) was specified at the beginning of a program that sets a tool group when tool data in a group was being loaded during loading tool life management data.
PS0444	OVER MAXIMUM LIFE GROUP	The number of blocks in which P (group No.) or L (tool life) is instructed exceeding the maximum number of groups was found while loading tool life management data.
PS0445	ILLEGAL L COMMAND	An illegal range instructed in the L (tool life) instruction was found while loading tool life management data.
PS0446	ILLEGAL H,D,T COMMAND	A value specified by T (tool No.), D (cutter compensation No.) or H (tool length compensation No.) exceeding the maximum value was found while loading tool life management data.
PS0448	ILLEGAL FORMAT AT LIFE DATA	An illegal address was specified in a program (G10 L3: to G11;) that sets a tool group. Allowable addresses are P (group No.), L (tool life) and T (tool No.).

Number	Message	Contents
PS0449	ILLEGAL TOOL LIFE DATA	Tool life management data is damaged for some reason. Reload the tool group and the corresponding tool data by G10 L3; or MDI input.
PS0450	IN PMC AXIS MODE	The PMC axis control mode, the CNC issued a move instruction for the PMC axis. A move instruction can be issued on one axis from only either the PMC or the CNC. This alarm is generated when a move instruction is issued from both. This alarm can be suppressed by setting NPA of parameter No. 2405#5 to "1".
PS0470	G40.1-G42.1 PARAMETER MISS	The normal line direction control axes parameter settings are erroneous. - Set the normal line direction control axis setting in NDC parameter No. 1006#6. Only one axis can be set to "1"; this is not possible for multiple axes. - Set the axis for which NDC parameter No. 1006#6 is set to "1" to axis of rotation (ROT parameter No. 1006#6=1, and parameter No. 1022=2). - Set the feedrate when rotation of the normal line direction control axis is set to parameter No. 1472 within the range 1 to 5000 degrees/min.
PS0472	ILL-COMMAND IN G81.1 MODE	Either G81.1 was instructed again while in the chopping mode, or a move instruction was issued to the chopping axis. To change the chopping conditions, cancel G80 and specify G81.1 again.
PS0508	DUPLICATE M-CODE (INDEX TABLE REVERSING)	A function to which the same code as this M code is set exists. (indexing of index table)
PS0509	DUPLICATE M-CODE (SPOS AXIS ORIENTATION)	A function to which the same code as this M code is set exists. (spindle positioning, orientation)
PS0510	DUPLICATE M-CODE (SPOS AXIS POSITIONING)	A function to which the same code as this M code is set exists. (spindle positioning, positioning)
PS0511	DUPLICATE M-CODE (SPOS AXIS RELEASE)	A function to which the same code as this M code is set exists. (spindle positioning, mode cancel)
PS0531	ILLEGAL USE OF DECIMAL POINT (F-CODE)	When the feedrate instruction contains valid data below the decimal point, the alarm is set and the F code contains valid data below the decimal point.
PS0532	ILLEGAL USE OF DECIMAL POINT (E-CODE)	When the feedrate instruction contains valid data below the decimal point, the alarm is set and the E code contains valid data below the decimal point.
PS0533	ADDRESS F UNDERFLOW (G95)	The feedrate for the hole drilling axis calculated from the F and S codes is too slow in the feed per single rotation mode (G95).
PS0534	ADDRESS F OVERFLOW (G95)	The feedrate for the hole drilling axis calculated from the F and S codes is too fast in the feed per single rotation mode (G95).
PS0535	ADDRESS E UNDERFLOW (G95)	The feedrate for the hole drilling axis calculated from the E and S codes is too slow in the feed per single rotation mode (G95).
PS0536	ADDRESS E OVERFLOW (G95)	The feedrate for the hole drilling axis calculated from the E and S codes is too fast in the feed per single rotation mode (G95).
PS0537	ADDRESS F UNDERFLOW (OVERRIDE)	The speed obtained by applying override to the F instruction is too slow.
PS0538	ADDRESS F OVERFLOW (OVERRIDE)	The speed obtained by applying override to the F instruction is too fast.
PS0539	ADDRESS E UNDERFLOW (OVERRIDE)	The speed obtained by applying override to the E instruction is too slow.

Number	Message	Contents
PS0540	ADDRESS E OVERFLOW (OVERRIDE)	The speed obtained by applying override to the E instruction is too fast.
PS0541	S-CODE ZERO	"0" has been instructed as the S code.
PS0542	FEED ZERO (E-CODE)	"0" has been instructed as the feedrate (E code).
PS0543	ILLEGAL GEAR SETTING	The gear ratio between the spindle and position coder, or the set position coder number of pulses is illegal in the spindle position function and the rigid tapping function.
PS0548	ILLGAL AXIS MODE	The spindle positioning axis/Cs contour control axis was specified during switching of the controlled axis mode.
PS0551	DUPLICATE SPOS AXIS COMMAND	Two or more axes were specified on a single spindle positioning axis. (e.g. positioning by an M code or positioning by an axis address)
PS0552	SPOS AXIS - OTHER AXIS SAME TIME	The spindle positioning axis and another axis are specified in the same block.
PS0553	SPOS AXIS ILLEGAL SEQUENCE NUMBER	Operation sequence pattern setting parameter No. 5895 for spindle positioning is out of range.
PS0555	SPOS AXIS DUPLICATE AXIS COMMAND	The spindle positioning instruction has been issued to a spindle that is currently moving or which the instructed spindle positioning sequence has not completed.
PS0561	ILLEGAL INDEXING ANGLE	The specified angle of rotation is not an integer multiple of the minimum indexing angle.
PS0562	INDEX TABLE AXIS COMMAND CAN NOT START	The specified index table indexing sequence is illegal.
PS0563	EXCESS ERROR OF INDEXING ANGLE	The specified index table indexing angle is illegal.
PS0564	INDEX TABLE AXIS - OTHER AXIS SAME TIME	The index table indexing axis and another axis have been specified in the same block.
PS0566	INDEX TABLE AXIS DUPLICATE AXIS MODE	A move instruction has been issued to an axis that is not an index table indexing axis.
PS0567	INDEX TABLE AXIS DUPLICATE AXIS COMMAND	Index table indexing was specified during axis movement or on an axis for which the index table indexing sequence was not completed.
PS0571	CS AXIS DUPLICATE AXIS MODE	The Cs contour control axis has been specified on an axis that is not in the Cs contour control mode.
PS0572	CS AXIS DUPLICATE AXIS COMMAND	A currently moving axis has been specified as the Cs contour control axis.
PS0580	ENCODE ALARM (PSWD&KEY)	When an attempt was made to read a program, the specified password did not match the password on the tape and the password on tape was not equal to 0. When an attempt was made to punch an encrypted tape, the password was not in the range 1 to 99999999. The password parameter is No. 2210.
PS0581	ENCODE ALARM (PARAMETER)	When an attempt was made to punch an encrypted tape, the punch code parameter was set to EIA. Set EIA parameter No. 0000#4 to "0". An incorrect instruction was specified for program encryption or protection. This alarm is also generated when the protected range is edited or deleted in a program-locked state. The protected range is defined from the program No. preset by parameter No. 2212 up to the program No. preset to parameter No. 2213. When both parameters are set to "0", the protected range becomes O9000 to O9999.
PS0590	TH ERROR	A TH error was detected during reading from an input device. The read code that caused the TH error and how many statements it is from the block can be verified in the diagnostics screen.
PS0591	TV ERROR	An error was detected during the single-block TV error. The TV check can be suppressed by setting TVC parameter No. 0000#0 to "0".

Number	Message	Contents
PS0592	END OF RECORD	The EOR (End of Record) code is specified in the middle of a block. This alarm is also generated when the percentage at the end of the NC program is read.
PS0593	EGB PARAMETER SETTING ERROR	Erroneous EGB parameter setting (1) The setting of SYN parameter No. 1955#0 is incorrect. (2) The slave axes set by the G81 code are not set to rotary axes. (ROT and ROS parameter Nos. 1006#0, #1) (3) The number of pulses (parameter Nos. 5596, 5597) per rotation is not set. (4) Parameter No. 5995 is not set by the hobbing machine compatible instruction.
PS0594	EGB FORMAT ERROR	Format error in the block in which EGB was specified (1) Nothing is specified to the master axis or slave axes in the G81.5 block. (2) Out of range data is specified to the master axis or slave axes in the G81.5 block. (3) Number of threads T is not specified in the G81 block. (4) Out of range data is specified by one of T, L, P or Q codes in the G81 block. (5) Only one of P or Q codes is specified in the G81 block.
PS0595	ILL-COMMAND IN EGB MODE	An illegal instruction was issued during synchronization by EGB. (1) Slave axis was specified by G27, G28, G29, G30, G30.1, G33 and G53 G codes. (2) Inch/metric conversion was specified by G20 or G21 G codes.
PS0596	EGB OVERFLOW	An overflow occurred during calculation of the synchronization coefficient.
PS0597	EGB AUTO PHASE FORMAT ERROR	Format error in block in which G80 or G81 was specified by EGB automatic phase alignment (1) R is data outside of the instruction range.
PS0598	EGB AUTO PHASE PARAMETER SETTING ERROR	Erroneous parameter settings relating to EGB automatic phase alignment (1) The acceleration/deceleration parameter is invalid. (2) The automatic phase alignment parameter is invalid.
PS0610	ILLEGAL G07.1 AXIS	An axis which cannot perform cylindrical interpolation was specified. More than one axis was specified in a G07.1 block. An attempt was made to cancel cylindrical interpolation for an axis that was not in the cylindrical interpolation mode. For the cylindrical interpolation axis, set not "0" but one of 5, 6 or 7 (parallel axis specification) to parameter No. 1022 to instruct the arc with axis of rotation (ROT parameter No. 1006#1 is set to "1" and parameter No. 1260 is set) ON.
PS0611	ILLEGAL G-CODE USE(G07.1 MODE)	A G code was specified that cannot be specified in the cylindrical interpolation mode. This alarm also is generated when an 01 group G code was in the G00 mode or code G00 was instructed. Cancel the cylindrical interpolation mode before instructing code G00.
PS0618	ILLEGAL P-DATA (WHEEL WEAR COMPENSATION)	The P data for selecting the compensation center in grinding wheel wear compensation is illegal.
PS0619	ILLEGAL AXIS (WHEEL WEAR COMPENSATION)	A compensation axis has been changed in grinding wheel wear compensation mode or compensation vector maintenance mode. Alternatively, the settings in parameter Nos. 6056 and 6057 that determine the target axes for grinding wheel wear compensation are illegal.

Number	Message	Contents
PS0625	TOO MANY G68 NESTING	3-dimensional coordinate conversion was specified more than twice. Cancel 3-dimensional coordinate conversion before executing new coordinate conversion.
PS0626	G68 FORMAT ERROR	There is a format error in the 3-dimensional coordinate conversion block. The alarm occurs in the following cases: (1) When I, J or K is missing from the block in which 3-dimensional coordinate conversion is specified (when the coordinate rotation option is not available) (2) When I, J and K specified in the block in which 3-dimensional coordinate conversion is specified are all "0" (3) When angle of rotation R is not specified in the block in which 3-dimensional coordinate conversion is specified
PS645	TOO MANY G68.2 NESTING	A feature coordinate system set command was issued more than once. To newly set a feature coordinate system, cancel the previous commands, then newly issue a feature coordinate system command.
PS646	G68.2 FORMAT ERROR	In a feature coordinate system set command block, the I, J, and K commands are all 0s.
PS647	ILLEGAL USE OF G53.1	(1) A tool axis direction control command was issued before a feature coordinate system set command. Issue the tool axis direction control command in a block after the feature coordinate system set command. (2) There is no angle solution for setting the tool axis direction to the +Z-axis direction of the feature coordinate system. Specify a feature coordinate system that can produce an angle solution.
PS0710	ILLEGAL COMMAND IN 3-D CIR	In a modal state in which a 3-dimensional circular interpolation (G02.4 or G03.4) cannot be commanded, an attempt for such interpolation was made. An invalid code was commanded while 3-dimensional circular interpolation mode is in progress.
PS0712	G02.4/G03.4 FORMAT ERROR	Three-dimensional circular interpolation (G02.4 or G03.4) was commanded incorrectly.
PS0713	MANUAL INTERVENTION IN G02.4/G03.4 (ABS ON)	While 3-dimensional circular interpolation mode (G02.4 or G03.4) is in progress, manual operation was performed with the manual absolute switch turned on.
PS0805	ILLEGAL COMMAND	[I/O Device] An attempt was made to specify an illegal command during I/O processing on an I/O device. [G30 Zero Return] The P address Nos. for instructing 2nd to 4th reference position return are each out of the range 2 to 4. [Single Rotation Dwell] The specified spindle rotation is "0" when single rotation dwell is specified. [3-dimensional Tool Offset] A G code that cannot be specified was specified in the 3-dimensional tool offset mode. Scaling instruction G51, skip cutting G31 and automatic tool length measurement G37 were specified.
PS0806	DEVICE TYPE MISS MATCH	An operation not possible on the I/O device that is currently selected in the setting was specified. This alarm is also generated when file rewind is instructed even though the I/O device is not a FANUC Cassette.

Number	Message	Contents
PS0807	PARAMETER SETTING ERROR	An I/O interface option that has not yet been added on was specified. The external I/O device and baud rate, stop bit and protocol selection settings are erroneous.
PS0808	DEVICE DOUBLE OPENED	An attempt was made to open a device that is being accessed.
PS0809	ILLEGAL COMMAND IN G41/G42	Specified direction tool length compensation parameters are incorrect. A move instruction for a axis of rotation was specified in the specified direction tool length compensation mode.
PS0895	ILLEGAL PARAMETER IN G02.3/G03.3	The parameter setting that specifies the axis on which to execute exponential interpolation is incorrect. Parameter No. 7636: Linear axis No. on which exponential interpolation is executed Parameter No. 7637: Rotation axis No. on which exponential interpolation is executed The setting is 1 to the number of controlled axes. The same axis Nos. must not be set.
PS0896	ILLEGAL FORMAT IN G02.3/G03.3	The format for specifying exponential interpolation is incorrect. Addresses I slope angle, J excessive torsion and setting R in exponential interpolation are not specified, or are set to "0". The setting ranges for I and J are -89.0 to +89.0 (excluding "0"). This alarm is also generated when addresses I and J are outside of this range. When CBK parameter No. 7610#7 is set to "0", the linear axis span value is assigned by parameter No. 7685. When the CBK parameter is set to "1", the span value is specified by address K in the G02.3/G03.3 block. This alarm is also generated when these span values are "0".
PS0897	ILLEGAL COMMAND IN G02.3/G03.3	An illegal value was specified in exponential interpolation. The natural logarithm parameter fell to less than "0" during exponential interpolation calculation. Review the exponential interpolation instruction.
PS0898	ILLEGAL PARAMETER IN G54.2	An illegal parameter (Nos. 6068 to 6076) was specified for fixture offset.
PS0899	ILLEGAL PARAMETER IN G43.2	An illegal parameter was specified for dynamic tool offset. Set the controlled axis number in order from the 1st to the 3rd set in parameter Nos. 6059 to 6067. If setting up to the 3rd set is not required. set parameter No. 6065 to "0".
PS0900	G72.1 NESTING ERROR	G72.1 was specified again during G72.1 rotation copying.
PS0901	G72.2 NESTING ERROR	G72.2 was specified again during G72.2 parallel copying.
PS0935	ILLEGAL FORMAT IN G02.2/G03.2	The end point of an involute curve on the currently selected plane, or the center coordinate instruction I, J or K of the corresponding basic circle, or basic circle radius R was not specified.
PS0936	ILLEGAL COMMAND IN G02.2/G03.2	An illegal value was specified in the involute curve. The coordinate instruction I, J or K of the basic circle on the currently selected plane or the basic circle radius R is "0", or the start and end points are not inside the basic circle.
PS0937	OVER TOLERANCE OF END POINT	The end point is not positioned on the involute curve that passes through the start point, and this error exceeds the permissible error limit (parameter No. 2510).
PS0990	SPL:ILLEGAL AXIS COMMAND	An illegal axis was specified for spline interpolation or smooth interpolation. This alarm is also generated when an axis other than those used for spline interpolation is specified. An "axis used for spline interpolation" refers to the axis specified in the block following the block in which the G06.1 G code is specified. When smooth interpolation is executed, the axis specified by the G5.1Q2 G code is illegal.

Number	Message	Contents
PS0991	SPL:ILLEGAL COMMAND	A G06.1 code was specified in a G code mode in which the instruction is not supported.
PS0992	SPL:ILLEGAL AXIS MOVING	Movement was specified for an axis other than those used for spline interpolation. This alarm is also generated, for example, when movement is specified along the Z-axis when spline interpolation along the 2 axes, X and Y, is executed in the 3-dimensional tool offset mode in which the 3 axes, X, Y and Z, are set as the offset vector components.
PS0993	SPL:CAN'T MAKE VECTOR	A 3-dimensional tool offset vector cannot be generated. - In generation of the 3-dimensional tool offset vector from P2 onwards, the previous point and the following point are on the same line, and that line is parallel with the 3-dimensional tool offset vector for the previous point. - The end point and the 2 previous point are the same in generation of the 3-dimensional tool offset vector by the end point for smooth interpolation and spline interpolation.
PS0995	ILLEGAL PARAMETER IN G41.2/G42.2	The parameter settings (parameter Nos. 6080 to 6089) for determining the relationship between the axis of rotation and the rotation plane are incorrect.
PS0996	G41.3/G40 FORMAT ERROR	(1) A move instruction was specified in a block in which the G41.3 or G40 code is specified. (2) A G or M code which suppresses buffering was specified in the block in which the G41.3 code was specified.
PS0997	ILLEGAL COMMAND IN G41.3	(1) A G code other than G00 or G01 in group 01 was specified in the G41.3 mode. (2) An offset (G code in group 07) was specified in the G41.3 mode. (3) The block following the block in which G41.3 (startup) was specified did not contain a move command.
PS0998	G41.3 ILLEGAL START_UP	(1) The G41.3 G code (startup) was specified in a group 01 mode for other than G00 and G01. (2) The angle formed by the tool direction vector and the movement direction vector was 0° or 180° degrees at startup.
PS0999	ILLEGAL PARAMETER IN G41.3	The parameter settings (parameter Nos. 6080 to 6089) for determining the relationship between the axis of rotation and the rotation plane are incorrect.
PS1001	ILLEGAL ORDER (NURBS)	The specified number of levels is incorrect.
PS1002	NO KNOT COMMAND (NURBS)	Knot has not been specified, or a block not related to NURBS interpolation was specified in the NURBS interpolation mode.
PS1003	ILLEGAL AXIS COMMAND (NURBS)	An axis not specified as a control point was specified in the 1st block.
PS1004	ILLEGAL KNOT	There is an insufficient number of knot individual blocks.
PS1005	ILLEGAL CANCEL (NURBS)	The NURBS interpolation mode was turned OFF even though NURBS interpolation was not completed.
PS1006	ILLEGAL MODE (NURBS)	A mode that cannot be paired with the NURBS interpolation mode was specified.
PS1007	ILLEGAL MULTI-KNOT	Nested knots for each level can be specified for the start and end points.
PS1008	ILLEGAL KNOT VALUE (NURBS)	Knot is not increased monotonously.
PS1009	ILLEGAL 1ST CONTROL POINT (NURBS)	The 1st control point is erroneous, or there is no continuity with the previous block.
PS1010	ILLEGAL RESTART (NURBS)	An attempt was made to resume NURBS interpolation after manual intervention with manual absolute ON.

Number	Message	Contents
PS1070	ILLEGAL USE OF G41.5/G42.5	The parameters related to three-dimensional cutter compensation for rotary table are not specified properly. An attempt was made to issue the G39 command in the mode of three-dimensional cutter compensation for rotary table(G41.5/G42.5). An attempt was made to issue a move command other than G00 and G01, such as G02, at the startup of three-dimensional cutter compensation for rotary table(G41.5/G42.5).
PS1100	ILLEGAL PARAMETER OF MACHINE COMPONENT	A parameter (No. 6161 to No. 6195 or No. 7540 to No. 7548) for configuring the machine is incorrect.

F.2 BG ALARM (ALARMS RELATED TO BACKGROUND EDIT)

Number	Message	Contents
BG0590	TH ERROR	A TH error was detected during reading from an input device. The read code that caused the TH error and how many statements it is from the block can be verified in the diagnostics screen.
BG0591	TV ERROR	An error was detected during the single-block TV error. The TV check can be suppressed by setting TVC parameter No. 0000#0 to "0".
BG0592	END OF RECORD	The EOR (End of Record) code is specified in the middle of a block. This alarm is also generated when the percentage at the end of the NC program is read.
BG0805	ILLEGAL COMMAND	An attempt was made to specify an illegal command during I/O processing on an I/O device.
BG0806	DEVICE TYPE MISS MATCH	An operation not possible on the I/O device that is currently selected in the setting was specified. This alarm is also generated when file rewind is instructed even though the I/O device is not a FANUC Cassette.
BG0807	PARAMETER SETTING ERROR	An I/O interface option that has not yet been added on was specified. The external I/O device and baud rate, stop bit and protocol selection settings are erroneous.
BG0808	DEVICE DOUBLE OPENED	An attempt was made to open a device that is being accessed.
BG0820	DR OFF(1)	The data set ready input signal DR of the I/O device connected to reader/punch interface 1 turned OFF.
BG0822	OVERRUN ERROR(1)	The next character was received from the I/O device connected to reader/punch interface 1 before it could read a previously received character.
BG0823	FRAMING ERROR(1)	The stop bit of the character received from the I/O device connected to reader/punch interface 1 was not detected.
BG0824	BUFFER OVERFLOW(1)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 1 even though the NC sent a stop code (DC3) during data reception.
BG0830	DR OFF(2)	The data set ready input signal DR of the I/O device connected to reader/punch interface 2 turned OFF.
BG0832	OVERRUN ERROR(2)	The next character was received from the I/O device connected to reader/punch interface 2 before it could read a previously received character.
BG0833	FRAMING ERROR(2)	The stop bit of the character received from the I/O device connected to reader/punch interface 2 was not detected.
BG0834	BUFFER OVERFLOW(2)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 1 even though the NC sent a stop code (DC3) during data reception.
BG0840	DR OFF(3)	The data set ready input signal DR of the I/O device connected to reader/punch interface 3 turned OFF.
BG0842	OVERRUN ERROR(3)	The next character was received from the I/O device connected to reader/punch interface 3 before it could read a previously received character.
BG0843	FRAMING ERROR(3)	The stop bit of the character received from the I/O device connected to reader/punch interface 3 was not detected.
BG0844	BUFFER OVERFLOW(3)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 3 even though the NC sent a stop code (DC3) during data reception.
BG0850	DR OFF(4)	The data set ready input signal DR of the I/O device connected to reader/punch interface 4 turned OFF.

Number	Message	Contents
BG0852	OVERRUN ERROR(4)	The next character was received from the I/O device connected to reader/punch interface 4 before it could read a previously received character.
BG0853	FRAMING ERROR(4)	The stop bit of the character received from the I/O device connected to reader/punch interface 4 was not detected.
BG0854	BUFFER OVERFLOW(4)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 4 even though the NC sent a stop code (DC3) during data reception.
BG0910	DEVICE DRIVER ERROR (UNDEFINED)	An error occurred during device driver control.
BG0911	V-DEVICE DRIVER ERROR (DEVICE)	An error occurred during device driver control.
BG0912	V-DEVICE DRIVER ERROR (OPEN)	An error occurred during device driver control.
BG0913	V-DEVICE DRIVER ERROR (COMMAND)	An error occurred during device driver control.
BG0914	V-DEVICE DRIVER ERROR (RANGE)	An error occurred during device driver control.
BG0915	V-DEVICE DRIVER ERROR (TEST)	An error occurred during device driver control.
BG0950	DRIVER ERROR (MEMORY CARD)	An error occurred in the memory card device driver.
BG0960	ACCESS ERROR (MEMORY CARD)	Illegal memory card accessing This alarm is also generated during reading when reading is executed up to the end of the file without detection of the EOR code.
BG0961	NOT READY (MEMORY CARD)	The memory card is not ready.
BG0962	CARD FULL (MEMORY CARD)	The memory card has run out of space.
BG0963	CARD PROTECTED (MEMORY CARD)	The memory card is write-protected.
BG0964	NOT MOUNTED (MEMORY CARD)	The memory card could not be mounted.
BG0965	DIRECTORY FULL (MEMORY CARD)	The file could not be generated in the root directory for the memory card.
BG0966	FILE NOT FOUND (MEMORY CARD)	The specified file could not be found on the memory card.
BG0967	FILE PROTECTED (MEMORY CARD)	The memory card is write-protected.
BG0968	ILLEGAL FILE NAME (MEMORY CARD)	Illegal memory card file name
BG0969	ILLEGAL FORMAT (MEMORY CARD)	Check the file name.
BG0970	ILLEGAL CARD (MEMORY CARD)	This memory card cannot be handled.
BG0971	ERASE ERROR (MEMORY CARD)	An error occurred during memory card erase.
BG0972	BATTERY LOW (MEMORY CARD)	The memory card battery is low.
BG0973	FILE ALREADY EXIST	A file having the same name already exists on the memory card.

F.3 SR ALARM

Number	Message	Contents
SR0125	ILLEGAL EXPRESSION FORMAT	The description of the custom macro statement is erroneous. The format of the parameter data is erroneous.
SR0160	COMMAND DATA OVERFLOW	An overflow in the CNC internal positional data occurred. This error is also generated when the target position exceeds the maximum stroke as a result of calculating coordinate conversion, offset and manual intervention compensation inputs.
SR0421	SETTING COMMAND ERROR	The specified tool data setting (G10 L70 to G11, G10 L71 to G11) is erroneous.
SR0422	NOT FOUND TOOL DATA	The tool No. (pot No.) to which tool data delete was specified cannot be found.
SR0423	TOOL DATA WRITE ERROR	A write error occurred simultaneously on tool offset data by a tool No.
SR0424	OVER MAXIMUM TOOL DATA	An attempt was made to set tool data that exceeds the maximum number of tool data sets.
SR0425	NOT DELETE TOOL DATA IN OPERATION	An attempt was made to delete currently selected tool data.
SR0580	ENCODE ALARM (PSWD&KEY)	When an attempt was made to read a program, the specified password did not match the password on the tape and the password on tape was not equal to 0. When an attempt was made to punch an encrypted tape, the password was not in the range 0 to 99999999. The password parameter is No. 2210.
SR0581	ENCODE ALARM (PARAMETER)	When an attempt was made to punch an encrypted tape, the punch code parameter was set to EIA. Set EIA parameter No. 0#4 to "0". An incorrect instruction was specified for program encryption or protection. This alarm is also generated when the protected range is edited or deleted in a program-locked state. The protected range is defined from the program No. preset by parameter No. 2212 up to the program No. preset to parameter No. 2213. When both parameters are set to "0", the protected range becomes O9000 to O9999.
SR0590	TH ERROR	A TH error was detected during reading from an input device. The read code that caused the TH error and how many statements it is from the block can be verified in the diagnostics screen.
SR0591	TV ERROR	An error was detected during the single-block TV error. The TV check can be suppressed by setting TVC parameter No. 0000#0 to "0".
SR0592	END OF RECORD	The EOR (End of Record) code is specified in the middle of a block. This alarm is also generated when the percentage at the end of the NC program is read. This alarm is also generated when the specified block is not found by the program restart function.
SR0600	PARAMETER OF RESTART ERROR	An illegal value is set to parameter No. 7110 that specifies the order in which axes move when machining is restarted in the dry run. The setting range is 1 to the number of controlled axes.
SR0805	ILLEGAL COMMAND	An attempt was made to specify an illegal command during I/O processing on an I/O device.
SR0806	DEVICE TYPE MISS MATCH	An operation not possible on the I/O device that is currently selected in the setting was specified. This alarm is also generated when file rewind is instructed even though the I/O device is not a FANUC Cassette.

Number	Message	Contents
SR0807	PARAMETER SETTING ERROR	An I/O interface option that has not yet been added on was specified. The external I/O device and baud rate, stop bit and protocol selection settings are erroneous.
SR0808	DEVICE DOUBLE OPENED	An attempt was made to open a device that is being accessed.
SR0820	DR OFF(1)	The data set ready input signal DR of the I/O device connected to reader/punch interface 1 turned OFF.
SR0822	OVERRUN ERROR(1)	The next character was received from the I/O device connected to reader/punch interface 1 before it could read a previously received character.
SR0823	FRAMING ERROR(1)	The stop bit of the character received from the I/O device connected to reader/punch interface 1 was not detected.
SR0824	BUFFER OVERFLOW(1)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 1 even though the NC sent a stop code (DC3) during data reception.
SR0830	DR OFF(2)	The data set ready input signal DR of the I/O device connected to reader/punch interface 2 turned OFF.
SR0832	OVERRUN ERROR(2)	The next character was received from the I/O device connected to reader/punch interface 2 before it could read a previously received character.
SR0833	FRAMING ERROR(2)	The stop bit of the character received from the I/O device connected to reader/punch interface 2 was not detected.
SR0834	BUFFER OVERFLOW(2)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 2 even though the NC sent a stop code (DC3) during data reception.
SR0840	DR OFF(3)	The data set ready input signal DR of the I/O device connected to reader/punch interface 3 turned OFF.
SR0842	OVERRUN ERROR(3)	The next character was received from the I/O device connected to reader/punch interface 3 before it could read a previously received character.
SR0843	FRAMING ERROR(3)	The stop bit of the character received from the I/O device connected to reader/punch interface 3 was not detected.
SR0844	BUFFER OVERFLOW(3)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 3 even though the NC sent a stop code (DC3) during data reception.
SR0850	DR OFF(4)	The data set ready input signal DR of the I/O device connected to reader/punch interface 4 turned OFF.
SR0852	OVERRUN ERROR(4)	The next character was received from the I/O device connected to reader/punch interface 4 before it could read a previously received character.
SR0853	FRAMING ERROR(4)	The stop bit of the character received from the I/O device connected to reader/punch interface 4 was not detected.
SR0854	BUFFER OVERFLOW(4)	The NC received more than 10 characters of data from the I/O device connected to reader/punch interface 4 even though the NC sent a stop code (DC3) during data reception.
SR0910	DEVICE DRIVER ERROR (UNDEFINED)	An error occurred during device driver control.
SR0911	V-DEVICE DRIVER ERROR (DEVICE)	An error occurred during device driver control.
SR0912	V-DEVICE DRIVER ERROR (OPEN)	An error occurred during device driver control.
SR0913	V-DEVICE DRIVER ERROR (COMMAND)	An error occurred during device driver control.
SR0914	V-DEVICE DRIVER ERROR (RANGE)	An error occurred during device driver control.
SR0915	V-DEVICE DRIVER ERROR (TEST)	An error occurred during device driver control.
SR0950	DRIVER ERROR (MEMORY CARD)	An error occurred in the memory card device driver.

Number	Message	Contents
SR0960	ACCESS ERROR (MEMORY CARD)	Illegal memory card accessing This alarm is also generated during reading when reading is executed up to the end of the file without detection of the EOR code.
SR0961	NOT READY (MEMORY CARD)	The memory card is not ready.
SR0962	CARD FULL (MEMORY CARD)	The memory card has run out of space.
SR0963	CARD PROTECTED (MEMORY CARD)	The memory card is write-protected.
SR0964	NOT MOUNTED (MEMORY CARD)	The memory card could not be mounted.
SR0965	DIRECTORY FULL (MEMORY CARD)	The file could not be generated in the root directory for the memory card.
SR0966	FILE NOT FOUND (MEMORY CARD)	The specified file could not be found on the memory card.
SR0967	FILE PROTECTED (MEMORY CARD)	The memory card is write-protected.
SR0968	ILLEGAL FILE NAME (MEMORY CARD)	Illegal memory card file name
SR0969	ILLEGAL FORMAT (MEMORY CARD)	Illegal memory card format
SR0970	ILLEGAL CARD (MEMORY CARD)	This memory card cannot be handled.
SR0971	ERASE ERROR (MEMORY CARD)	An error occurred during memory card erase.
SR0972	BATTERY LOW (MEMORY CARD)	The memory card battery is low.
SR0973	FILE ALREADY EXIST	A file having the same name already exists on the memory card.
SR1030	NOT READY (DATA SERVER)	The board is not ready in the Ethernet/data server board function.
SR1031	BUSY (DATA SERVER)	The board is busy in the Ethernet/data server board function.
SR1032	BOARD ERROR (DATA SERVER)	An error was returned from the board in the Ethernet/data server board function.
SR1033	TIME OVER (DATA SERVER)	The CNC registered a time-out in the Ethernet/data server board function.
SR1034	CNC ERROR (DATA SERVER)	An internal CNC error occurred by the Ethernet/data server board function.
SR1035	SEQUENCE ERROR (DATA SERVER)	A contradiction occurred in the CNC by the Ethernet/data server board function.
SR1036	MODE ERROR (DATA SERVER)	The mode could not be changed in the Ethernet/data server board function.
SR1037	DOUBLE OPEN (DATA SERVER)	An attempt was made to open the file twice in the Ethernet/data server board function.

F.4 SW ALARM (ALARMS RELATED TO PARAMETER WRITING)

Number	Message	Contents
SW0000	PARAMETER ENABLE SWITCH ON	The parameter setting is enabled (PWE, one bit of parameter No. 8000 is set to "1"). To set the parameter, turn this parameter ON. Otherwise, set to OFF.

F.5 SV ALARM (ALARMS RELATED TO SERVO)

Number	Message	Contents
SV0008	EXCESS ERROR (STOP)	Position deviation during a stop is larger than the value set in parameter No. 1829. Check the value of the position deviation error limit in parameter No. 1829.
SV0009	EXCESS ERROR (MOVING)	The position deviation during movement is larger than the value set in parameter No. 1828.
SV0011	LSI OVERFLOW	The position deviation counter overflowed.
SV0012	MOTION VALUE OVERFLOW	A movement speed exceeding the limit was commanded. The limit is as follows: When acceleration/deceleration mode before pre-read interpolation is enabled: When fine acceleration/deceleration (FAD) and feed forwarding are disabled in other than acceleration/ deceleration mode before pre-read interpolation: 32,767 pulses/msec (detection unit) When fine acceleration/deceleration (FAD) is enabled in other than acceleration/deceleration mode before pre-read interpolation: 16,383 pulses/msec (detection unit) When fine acceleration/deceleration (FAD) is disabled but feed forwarding is enabled in other than acceleration/ deceleration mode before pre-read interpolation: 4095 pulses/msec (detection unit) When parameter FFN (No. 1801#1) is set to 1, feed forwarding can be enabled only in acceleration/ deceleration mode before pre-read interpolation. In this state, the movement speed limit can be kept at 32,767 pulses/msec.
SV0013	IMPROPER V_READY OFF	The speed control ready signal (VRDY) turned OFF even though the position control ready signal (PRDY) was ON.
SV0014	IMPROPER V_READY ON	The speed control ready signal (VRDY) turned ON even though the position control ready signal (PRDY) was OFF.
SV0024	SYNC EXCESS ERROR ALARM 2	The synchronous error amount is greater than the value set to parameter No. 1913 by the synchronize adjust function. When synchronization alignment has not completed after the power is turned ON, this error is judged by the value obtained of multiplying the value of parameter No. 1913 by the value of parameter No. 1910.
SV0025	V_READY ON (INITIALIZING)	During servo control, the speed control ready signal (VRDY) turned ON even though it is supposed to be OFF.
SV0026	ILLEGAL AXIS ARRANGE	The parameter for specifying the arrangement of the servo axes is set incorrectly. A minus value, duplicated value, or a value larger than the number of controlled axes has been set to parameter No. 1023 (servo No. axis for each axis).
SV0027	ILL DGTL SERVO PARAMETER	The setting of the digital servo parameter is illegal.
SV0030	EMERGENCY STOP	Emergency stop occurred. Whether or not this alarm is generated is determined by ENR parameter No. 2001#0). 1: The control is reset by an emergency stop. 0: This alarm is generated without reset by an emergency stop. An emergency stop is canceled by resetting the control.
SV0055	ILLEGAL TANDEM AXIS	The setting of parameter No. 1023 is illegal in tandem control.
SV0056	ILLEGAL TANDEM PAIR	The setting of parameter No. 1020 or TDM parameter No. 1817#6 is illegal in tandem control.

Number	Message	Contents
SV0060	FSSB:OPEN READY TIME OUT	The FSSB was not in a ready to open state during initialization. A probable cause is an axis card malfunction.
SV0061	FSSB:ERROR MODE	The FSSB entered the error mode. Probable causes are an axis card or amplifier malfunction.
SV0062	FSSB:NUMBER OF AMP. IS INSUFFICIENT	The number of amplifiers identified by the FSSB is less than the number of controlled axes. The set number of axes or the amplifier connection is erroneous.
SV0063	FSSB:CONFIGURATION ERROR	The FSSB configuration error occurred. Or, there is a difference in the type of connected amplifier and FSSB setting.
SV0064	FSSB:AXIS SETTING NOT COMPLETE	Setting of axes was not completed in the automatic setting mode.
SV0065	FSSB:OPEN TIME OUT	FSSB did not open even though FSSB initialization was not completed. Or, the connection between the CNC and amplifier is incorrect.
SV0066	FSSB:ID DATA NOT READ	Amplifier information cannot be read as FSSB is not assigned. Or, the connection between the CNC and amplifier is incorrect.
SV0067	FSSB:CONFIGURATION ERROR(SOFT)	The FSSB configuration error occurred. (Detected in software). Or, there is a difference in the type of connected amplifier and FSSB setting.
SV0070	FSSB DISCONNECT	FSSB communications was broken; the FSSB communications cable was disconnected, or broken; the amplifier power supply turned OFF; or, the low voltage alarm occurred on the amplifier.
SV0071	ILLEGAL AMP. INTERFACE	The axes of a 2-axis amplifier were assigned to a fast-type interface; or, the axis setting was incorrect.
SV0072	SEND CNC DATA FAILED	An FSSB communications error prevented correct data from being sent or received by the slave.
SV0073	SEND SLAVE DATA FAILED	An FSSB communications error prevented correct data from being sent or received by the servo software.
SV0074	READ ID DATA FAILED	Failed to read amplifier ID information during power ON.
SV0075	MOTOR/AMP. COMBINATION	The maximum current of the motor is different from the maximum current of the amplifier; the connection between the axis and amplifier is incorrect; or, the parameter setting is incorrect.
SV0076	ILLEGAL SETTING OF AXIS	The setting is incorrect even though functions requiring hardware for 2 axes are built into one axis.
SV0100	S-COMP. VALUE OVERFLOW	The value set for straightness compensation has exceeded maximum value 32,767.
SV0101	DATA ERROR(ABS PCDR)	A correct machine position cannot be obtained as the absolute pulse coder has malfunctioned or the machine moved too far during power ON.
SV0109	EXCESS ERROR (G31)	The position deviation during execution of torque limit switch operation exceeds the limit setting of parameter No. 1843.
SV0119	ABNORMAL TORQUE(1ST SPDL)	An abnormal load was detected at the No. 1 spindle motor. This alarm is canceled by resetting the control.
SV0120	ABNORMAL TORQUE(2ND SPDL)	An abnormal load was detected at the No. 2 spindle motor. This alarm is canceled by resetting the control.
SV0125	EXCESS VELOCITY IN TORQUE	In torque control, the specified permissible speed was exceeded.
SV0126	EXCESS ERROR IN TORQUE	In torque control, the maximum accumulated speed specified in parameters was exceeded.
SV0127	ABNORMAL TORQUE(3RD SPDL)	An abnormal load was detected at the No. 3 spindle motor. This alarm is canceled by resetting the control.
SV0128	ABNORMAL TORQUE(4TH SPDL)	An abnormal load was detected at the No. 24 spindle motor. This alarm is canceled by resetting the control.

Number	Message	Contents
SV0350	EXCESS SYNC TORQUE	The difference in torque between the master axis and slave axis exceeded the value set in the parameter (No. 1716) during synchronous control. This alarm is generated only for the master axis.
SV0360	ABNORMAL CHECKSUM(INT)	The checksum alarm occurred on the built-in pulse coder.
SV0361	ABNORMAL PHASE DATA(INT)	The phase data abnormal alarm occurred on the built-in pulse coder.
SV0363	ABNORMAL CLOCK(INT)	The clock alarm occurred on the built-in pulse coder.
SV0364	SOFT PHASE ALARM(INT)	The digital servo software detected abnormal data on the built-in pulse coder.
SV0365	BROKEN LED(INT)	Abnormal LED on the built-in pulse coder
SV0366	PULSE MISS(INT)	A pulse error occurred on the built-in pulse coder.
SV0367	COUNT MISS(INT)	A count error occurred on the built-in pulse coder.
SV0368	SERIAL DATA ERROR(INT)	The communications data could not be received from the built-in pulse coder.
SV0369	DATA TRANS. ERROR(INT)	A CRC error or stop bit error occurred in the communications data from the built-in pulse coder.
SV0380	BROKEN LED(EXT)	Standalone detector error
SV0381	ABNORMAL PHASE (EXT LIN)	An abnormal alarm in the position data occurred on the standalone linear scale.
SV0382	COUNT MISS(EXT)	A count error occurred on the standalone detector.
SV0383	PULSE MISS(EXT)	A pulse error occurred on the standalone detector.
SV0384	SOFT PHASE ALARM(EXT)	The digital servo software detected abnormal data on the standalone detector.
SV0385	SERIAL DATA ERROR(EXT)	The communications data could not be received from the standalone detector.
SV0386	DATA TRANS. ERROR(EXT)	A CRC error or stop bit error occurred in the communications data from the standalone detector.
SV0409	DETECT ABNORMAL TORQUE	An abnormal load was detected on the servo motor, or during Cs axis or spindle positioning.
SV0421	EXCESS ERROR(SEMI-FULL)	The difference between the feedback from the semi and full sides exceeded the setting of parameter No.1729.
SV0430	SV MOTOR OVERHEAT	The servo motor has overheated.
SV0431	CNV. OVERLOAD	PSM: Overheat B series SVU: Overheat
SV0432	CNV.LOWVOLT CON./POWFAULT	PSM : The control power supply voltage has dropped. PSMR : The control power supply voltage has dropped. β SVU : The control power supply voltage has dropped.
SV0433	CNV. LOW VOLT DC LINK	PSM: Missing phase in input voltage PSMR: Low DC link voltage A series SVU: Low DC link voltage B series SVU: Low DC link voltage
SV0434	LOW VOLT CONTROL	SVM: Low control power voltage
SV0435	INV. LOW VOLT DC LINK	SVM: Low DC link voltage
SV0436	SOFTTHERMAL(OVC)	The digital servo software detected a software thermal (OVC).
SV0437	CNV. OVERCURRENT POWER	PSM: Overcurrent on input circuit section.
SV0438	INV. ABNORMAL CURRENT	SVM: Motor overcurrent A series SVU: Motor overcurrent B series SVU: Motor overcurrent
SV0439	CNV. OVERVOLT POWER	PSM : The DC link voltage is too high. PSMR : The DC link voltage is too high. β SVU : The DC link voltage is too high.
SV0440	CONV. EX. DECELERATION POW.	PSMR: Excessive generative discharge A series SVU: Excessive generative discharge, or abnormal error in generative power circuit
SV0441	ABNORMAL CURRENT OFFSET	The digital servo software detected an abnormality in the motor current detection circuit.

Number	Message	Contents
SV0442	CNV. CHARGE FAULT/INV. DB	PSM : The spare charge circuit for the DC link is abnormal. PSMR : The spare charge circuit for the DC link is abnormal.
SV0443	CNV. COOLING FAN FAILURE	PSM: Internal cooling fan failure. PSMR: Internal cooling fan failure B series SVU: Internal cooling fan failure
SV0444	INV. COOLING FAN FAILURE	SVM: Internal cooling fan failure
SV0445	SOFT DISCONNECT ALARM	The digital servo software detected a disconnected pulse coder.
SV0446	HARD DISCONNECT ALARM	The hardware detected a disconnected built-in pulse coder.
SV0447	HARD DISCONNECT(EXT)	The hardware detected a disconnected standalone detector.
SV0448	UNMATCHED FEEDBACK ALARM	The sign of the feedback signal from the standalone detector is opposite to that from the feedback signal from the built-on pulse coder.
SV0449	INV. IPM ALARM	SVM: The IPM (Intelligent Power Module) detected an alarm. A series SVU: The IPM (Intelligent Power Module) detected an alarm.
SV0600	INV. DC LINK OVER CURRENT	SVM : The DC link current is too high. β SVU : The DC link current is too high.
SV0601	INV. RADIATOR FAN FAILURE	SVM : The heat sink cooling fan is defective. β SVU : The heat sink cooling fan is defective.
SV0602	INV. OVERHEAT	SVM : The servo amplifier is overheated.
SV0603	INV. IPM ALARM(OH)	SVM : The intelligent power module (IPM) has detected an overheat alarm condition. β SVU : The intelligent power module (IPM) has detected an overheat alarm condition.
SV0604	AMP. COMMUNICATION ERROR	Communication between the SVM and PSM is abnormal.
SV0605	CNV. EX. DISCHARGE POW.	PSMR : The motor regenerative power is too high.
SV0606	CNV. RADIATOR FAN FAILURE	PSM : The fan for cooling the external heat sink is defective. PSMR : The fan for cooling the external heat sink is defective.
SV0607	CNV. SINGLE PHASE FAILURE	PSM : One of the input power phases is abnormal. PSMR : One of the input power phases is abnormal.

F.6 OT ALARM

Number	Message	Contents
OT0001	+ OVERTRAVEL (SOFT 1)	The tool entered the prohibited area of stored stroke check 1 during movement in the positive direction.
OT0002	- OVERTRAVEL (SOFT 1)	The tool entered the prohibited area of stored stroke check 1 during movement in the negative direction.
OT0003	+ OVERTRAVEL (SOFT 2)	The tool entered the prohibited area of stored stroke check 2 during movement in the positive direction.
OT0004	- OVERTRAVEL (SOFT 2)	The tool entered the prohibited area of stored stroke check 1 during movement in the negative direction.
OT0007	+ OVERTRAVEL (HARD)	The stroke limit switch in the positive direction was triggered. This alarm is generated when the machine reaches the stroke end. When this alarm is not generated, feed of all axes is stopped during automatic operation. During manual operation, only the feed of the axis on which the alarm occurred is stopped.
OT0008	- OVERTRAVEL (HARD)	The stroke limit switch in the negative direction was triggered. This alarm is generated when the machine reaches the stroke end. When this alarm is not generated, feed of all axes is stopped during automatic operation. During manual operation, only the feed of the axis on which the alarm occurred is stopped.
OT0021	+ OVERTRAVEL (PRE-CHECK)	The tool exceeded the limit in the negative direction during the stroke check before movement.
OT0022	- OVERTRAVEL (PRE-CHECK)	The tool exceeded the limit in the positive direction during the stroke check before movement.
OT0030	SYNC EXCESS ERROR ALARM 1	The synchronous error amount is greater than the value set to parameter No. 1914 by the synchronize adjust function.
OT0031	SYNCHRONIZE ADJUST MODE	The system is in the synchronize adjust mode.
OT0032	NEED ZRN(ABS PCDR)	The reference position and the absolute pulse coder counter value do not match.
OT0034	BATTERY ZERO(ABS PCDR)	The battery voltage of the absolute position detector has fallen to "0", or power is supplied to the pulse coder for the first time.
OT0035	IMPOSSIBLE ZRN(SERIAL)	An attempt was made to create correspondence between the reference position and the absolute position detector when the origin cannot be established.
OT0036	BATTERY DOWN (ABS PCDR)	Low absolute position detector battery voltage
OT0120	UNASSIGNED ADDRESS (HIGH)	The upper 4 bits (EIA4 to EIA7) of an external data I/O interface address signal are set to an undefined address (high bits).
OT0121	UNASSIGNED ADDRESS (LOW)	The lower 4 bits (EIA0 to EIA3) of an external data I/O interface address signal are set to an undefined address (low bits).
OT0122	TOO MANY MESSAGE	Requests were made to display more than 4 external operator messages or external alarm messages at the same time.
OT0123	MESSAGE NUMBER NOT FOUND	An external operator message or external alarm message cannot be canceled as no message No. is specified.
OT0124	OUTPUT REQUEST ERROR	An output request was issued during external data output, or an output request was issued for an address that has no output data.
OT0125	TOO LARGE NUMBER	A numerical value outside the range 0 to 4095 was specified as the No. for the external operator message or an external alarm message.

Number	Message	Contents
OT0126	SPECIFIED NUMBER NOT FOUND	[External data I/O] The No. specified for a program No. or sequence No. search could not be found. There was an I/O request issued for a pot No. or offset (tool data), but either no tool numbers have been input since power ON or there is no data for the entered tool No. [External workpiece No. search] The program corresponding to the specified workpiece No. could not be found.
OT0127	DI.EIDHW OUT OF RANGE	The numerical value input by external data input signals EID32 to EID47 has exceeded the permissible range.
OT0128	DI.EIDLL OUT OF RANGE	The numerical value input by external data input signals EID0 to EID31 has exceeded the permissible range.
OT0129	NEGATE POS CODER 1 REV ON	The CPU of the position coder or a peripheral circuit is malfunctioning. Replace the CPU or peripheral circuit.
OT0130	SEARCH REQUEST NOT ACCEPTED	No requests can be accepted for a program No. or a sequence No. search as the system is not in the memory mode or the reset state.
OT0131	EXT-DATA ERROR (OTHER)	[External Data I/O] An attempt was made to input tool data for tool offset by a tool No. during loading by the G10 code.
OT0132	NOT ON RETURN_POINT	The tool did not arrive at the stored return position along the axis, or the position may have deviated by machine lock or mirror image operation during zero return.
OT0150	A/D CONVERT ALARM	A/D converter malfunction
OT0151	A/D CONVERT ALARM	A/D converter malfunction
OT0184	PARAMETER ERROR IN TORQUE	An invalid parameter was set for torque control. The torque constant parameter is set to "0".
OT0449	ZERO RETURN NOT FINISHED	The interval of the actual match mark does not match the interval of the match mark set to the parameter on the match-marked linear scale.
OT0450	ZERO RETURN NOT FINISHED	No. 1 zero return (CDxX7 to CDxX0: 17h (Hex)) was specified when the manual reference zero return was not executed with the reference zero return function enabled (ZRN parameter No. 1005#0 set to "0").
OT0451	IMPROPER PMC AXIS COMMAND	The PMC axes cannot be controlled in this state.
OT0512	EXCESS VELOCITY	The feedrate of the linear axis during polar coordinate interpolation exceeded the maximum cutting feedrate.
OT0513	SYNC EXCESS ERROR	The difference in the machine coordinate value during synchronize interpolation is equal to or greater than the synchronize error limit set in parameter No. 7723. Or, the offset during synchronize alignment by the machine coordinate values was equal to or greater than the maximum offset set in parameter No. 7724.
OT553	EXCESS VELOCITY IN G43.4/G43.5	Tool center point control resulted in an axis trying to move faster than the maximum cutting feed rate.
OT0710	ILLEGAL ACC. PARAMETER (OPTIMUM TORQUE ACC/DEC)	There are errors in the parameters of permissible acceleration for Optimum Torque Acceleration/Deceleration. One of the following is the cause. The ratio of the acceleration for deceleration to the acceleration for the acceleration is lower than the limited value. The time to decelerate to 0 is larger than the maximum.

F.7 IO ALARM

Number	Message	Contents
IO0001	FILE ACCESS ERROR	The resident-type file system could not be accessed as an error occurred in the resident-type file system.
IO0002	FILE SYSTEM ERROR	The file could not be accessed as an error occurred in the CNC file system.
IO0030	CHECK SUM ERROR	The checksum of the CNC part program storage memory is incorrect.
IO0032	MEMORY ACCESS OVER RANGE	Accessing of data occurred outside the CNC part program storage memory range.

F.8 PW ALARM (POWER MUST BE TURNED OFF THEN ON AGAIN)

Number	Message	Contents
PW0000	POWER MUST BE OFF	A parameter was set for which the power must be turned OFF then ON again.
PW0100	ILLEGAL PARAMETER (S-COMP.)	The parameter for setting straightness compensation is incorrect.
PW0102	ILLEGAL PARAMETER (I-COMP.)	The parameter for setting slope compensation is incorrect. This alarm occurs in the following cases: - When the number of pitch error compensation points on the axis on which slope compensation is executed exceeds 128 between the most negative side and most positive side - When the size relationship between the slope compensation point Nos. is incorrect - When the slope compensation point is not located between the most negative side and most positive side of pitch error compensation - When the compensation per compensation point is too small or too great.
PW0103	ILLEGAL PARAMETER (S-COMP.128)	The parameter for setting 128 straightness compensation points or the parameter compensation data is incorrect,

F.9 SP ALARM (ALARMS RELATED TO SPINDLE)

Number	Message	Contents
SP0001	SSPA:01 MOTOR OVERHEAT	An alarm (AL-01) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0002	SSPA:02 EX DEVIATION SPEED	An alarm (AL-02) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0003	SSPA:03 DC-LINK FUSE IS BROKEN	An alarm (AL-03) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0004	SSPA:04 POWER SUPPLY ERROR	An alarm (AL-04) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0005	SSPA:XX DECODED ALARM	An alarm (AL-05) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0006	SSPA:XX DECODED ALARM	An alarm (AL-06) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0007	SSPA:07 OVER SPEED	An alarm (AL-07) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0008	SSPA:XX DECODED ALARM	An alarm (AL-08) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0009	SSPA:09 OVERHEAT MAIN CIRCUIT	An alarm (AL-09) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0010	SSPA:10 LOW VOLT INPUT POWER	An alarm (AL-10) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0011	SSPA:11 OVERVOLT POWER CIRCUIT	An alarm (AL-11) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0012	SSPA:12 OVERCURRENT POWER CIRCUIT	An alarm (AL-12) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0013	SSPA:13 CPU DATA MEMORY FAULT	An alarm (AL-13) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0014	SSPA:XX DECODED ALARM	SP0014 An alarm (AL-14) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0015	SSPA:15 SPINDLE SWITCHING FAULT	An alarm (AL-15) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0016	SSPA:16 RAM ERROR	An alarm (AL-16) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0017	SSPA:XX DECODED ALARM	An alarm (AL-17) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0018	SSPA:18 SUMCHECK ERROR PROGRAM ROM	SP0018 An alarm (AL-18) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0019	SSPA:19 EXCESS OFFSET CURRENT U	An alarm (AL-19) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0020	SSPA:20 EXCESS OFFSET CURRENT V	An alarm (AL-20) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0021	POS SENSOR POLARITY ALARM	An alarm (AL-21) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0022	SSPA:XX DECODED ALARM	An alarm (AL-22) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0023	SSPA:XX DECODED ALARM	An alarm (AL-23) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0024	SSPA:24 SERIAL TRANSFER ERROR	An alarm (AL-24) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0025	SSPA:25 SERIAL TRANSFER STOP	An alarm (AL-25) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0026	SSPA:26 DISCONNECT CS VELOCITY DETECTOR	An alarm (AL-26) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0027	SSPA:27 DISCONNECT POSITION CODER	An alarm (AL-27) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0028	SSPA:28 DISCONNECT CS POSITION DETECTOR	An alarm (AL-28) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.

Number	Message	Contents
SP0029	SSPA:29 OVERLOAD	An alarm (AL-29) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0030	SSPA:30 OVERCURRENT INPUT CIRCUIT	An alarm (AL-32) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0031	SSPA:31 MOTOR LOCK OR DISCONNECT DETECTOR	An alarm (AL-31) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0032	SSPA:32 SIC-LSI RAM FAULT	An alarm (AL-32) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0033	SSPA:33 SHORTAGE POWER CHARGE	An alarm (AL-33) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0034	SSPA:34 ILLEGAL PARAMETER	An alarm (AL-34) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0035	SSPA:35 ILLEGAL GEAR RATIO PARAMETER	An alarm (AL-35) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0036	SSPA:36 OVERFLOW ERROR COUNTER	An alarm (AL-36) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0037	SSPA:37 ILLEGAL SETTING VELOCITY DETECTOR	An alarm (AL-37) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0038	SSPA:XX DECODED ALARM	An alarm (AL-38) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0039	SSPA:39 ILLEGAL 1REV SIGN OF CS DETECTOR	An alarm (AL-39) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0040	SSPA:40 NO 1REV SIGN OF CS DETECTOR	An alarm (AL-40) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0041	SSPA:41 ILLEGAL 1REV SIGN OF POSITION CODER	An alarm (AL-41) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0042	SSPA:42 NO 1REV SIGN OF POSITION CODER	An alarm (AL-42) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0043	SSPA:43 DISCONNECT POSITION CODER DEF. SPEED	An alarm (AL-43) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0044	SSPA:44 ILLEGAL A/D CONVERT	An alarm (AL-44) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0045	SSPA:XX DECODED ALARM	An alarm (AL-45) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0046	SSPA:46 ILLEGAL 1REV SIGN OF SCREW CUT	An alarm (AL-46) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0047	SSPA:47 ILLEGAL SIGNAL OF POSITION CODER	An alarm (AL-47) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0048	SSPA:XX DECODED ALARM	An alarm (AL-48) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0049	SSPA:49 DEF. SPEED IS OVER VALUE	An alarm (AL-49) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0050	SSPA:50 SYNCRONOUS VALUE IS OVER SPEED	An alarm (AL-50) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0051	SSPA:51 LOW VOLT POWER CIRCUIT	An alarm (AL-51) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0052	SSPA:52 ITP FAULT 1	An alarm (AL-52) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0053	SSPA:53 ITP FAULT 2	An alarm (AL-53) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0054	SSPA:54 OVERCURRENT	An alarm (AL-54) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0055	SSPA:55 ILLEGAL POWER LINE	An alarm (AL-55) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0056	COOLING FAN FAILURE	An alarm (AL-56) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0057	CONV.EX.DECELERATION POW.	An alarm (AL-57) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0058	CNV. OVERLOAD	An alarm (AL-58) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0059	CNV.COOLING FAN FAILURE	An alarm (AL-59) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.

Number	Message	Contents
SP0062	MOTOR VCMD OVERFLOWED	An alarm (AL-62) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0066	COM. ERROR BETWEEN SP AMPS	An alarm (AL-66) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0073	MOTOR SENSOR DISCONNECTED	An alarm (AL-73) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0074	CPU TEST ERROR	An alarm (AL-74) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0075	CRC ERROR	An alarm (AL-75) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0079	INITIAL TEST ERROR	An alarm (AL-79) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0080	ALARM AT THE OTHER SP AMP.	An alarm (AL-80) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0081	1-ROT MOTOR SENSOR ERROR	An alarm (AL-81) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0082	NO 1-ROT MOTOR SENSOR	An alarm (AL-82) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0083	MOTOR SENSOR SIGNAL ERROR	An alarm (AL-83) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0084	SPNDL SENSOR DISCONNECTED	An alarm (AL-84) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0085	1-ROT SPNDL SENSOR ERROR	An alarm (AL-85) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0086	NO 1-ROT SPNDL SENSOR	An alarm (AL-86) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0087	SPNDL SENSOR SIGNAL ERROR	An alarm (AL-87) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0088	COOLING RADI FAN FAILURE	An alarm (AL-88) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0097	OTHER SPINDLE ALARM	An alarm (AL-97) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0098	OTHER CONVERTER ALARM	An alarm (AL-98) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0110	AMP COMMUNICATION ERROR	An alarm (AL-B0) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0111	CONV. LOW VOLT CONTROL	An alarm (AL-B1) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0112	CONV. EX. DISCHARGE POW.	An alarm (AL-B2) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0113	CNV. COOLING FAN FAILURE	An alarm (AL-B3) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0120	COMMUNICATION DATA ERROR	An alarm (AL-C0) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0121	COMMUNICATION DATA ERROR	An alarm (AL-C1) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0122	COMMUNICATION DATA ERROR	An alarm (AL-C2) occurred on the spindle amplifier unit For details, refer to the Serial Spindle User's Manual.
SP0201	MOTOR NUMBER DUPLICATE	Two or more of the same motor Nos. other than "0" were set in parameter No. 5841.
SP0202	SPINDLE SELECT ERROR	A spindle No. exceeding the number of spindles were set in parameter No. 5850.
SP0220	NO SPINDLE AMP.	Either the cable connected to a serial spindle amplifier is broken, or the serial spindle amplifier is not connected.

Number	Message	Contents
SP0221	ILLEGAL MOTOR NUMBER	The spindle No. and the motor No. are incorrectly matched.
SP0222	CAN NOT USE ANALOG SPINDLE	The machine tool does not support analog spindles.
SP0223	CAN NOT USE SERIAL SPINDLE	The machine tool does not support digital spindles.
SP0224	ILLEGAL SPINDLE-POSITION CODER GEAR RATIO	The spindle-position coder gear ratio was incorrect.
SP0225	CRC ERROR (SERIAL SPINDLE)	A CRC error (communications error) occurred in communications between the CNC and the serial spindle amplifier.
SP0226	FRAMING ERROR (SERIAL SPINDLE)	A framing error occurred in communications between the CNC and the serial spindle amplifier.
SP0227	RECEIVING ERROR (SERIAL SPINDLE)	A receive error occurred in communications between the CNC and the serial spindle amplifier.
SP0228	COMMUNICATION ERROR (SERIAL SPINDLE)	A communications error occurred between the CNC and the serial spindle amplifier.
SP0229	COMMUNICATION ERROR SERIAL SPINDLE AMP.	A communications error occurred between serial spindle amplifiers (motor Nos. 1 and 2, or motor Nos. 3-4).
SP0230	MOTOR NUMBER OUT OF RANGE	The setting of parameter No. 5841 is out of range.
SP0231	SPINDLE EXCESS ERROR (MOVING)	The position deviation during spindle rotation was greater than the value set in parameters.
SP0232	SPINDLE EXCESS ERROR (STOP)	The position deviation during spindle stop was greater than the value set in parameters.
SP0233	POSITION CODER OVERFLOW	The error counter/speed instruction value of the position coder overflowed.
SP0234	GRID SHIFT OVERFLOW	Grid shift overflowed.
SP0235	ORIENTATION COMMAND OVERFLOW	The orientation speed is too fast.
SP0236	DUPLICATE SPINDLE CONTROL MODE (CHANGING)	An attempt was made to change the spindle mode during spindle mode switching.
SP0237	DUPLICATE SPINDLE CONTROL MODE (SPOS)	An attempt was made to change the spindle mode during the spindle positioning mode.
SP0238	DUPLICATE SPINDLE CONTROL MODE (RIGID TAP)	An attempt was made to change the spindle mode during the rigid tapping mode.
SP0239	DUPLICATE SPINDLE CONTROL MODE (CS)	An attempt was made to change the spindle mode during the Cs contour control mode.
SP0240	DISCONNECT POSITION CODER	The analog spindle position coder is broken.
SP0241	D/A CONVERTER ERROR	The D/A converter for controlling analog spindles is erroneous.
SP0242	OVERHEAT	An overheat was detected on the spindle due to spindle speed variations. - Reduce the number of cutting requirements when performing heavy cutting. - Check whether the tip of the cutting tool is sharp. - Check whether there is a problem with the spindle amplifier.
SP0245	COMMUNICATION DATA ERROR	A communication data error was detected on the CNC.
SP0246	COMMUNICATION DATA ERROR	A communication data error was detected on the CNC.
SP0247	COMMUNICATION DATA ERROR	A communication data error was detected on the CNC.
SP0968	SSPA:XX DECODED ALARM	An alarm occurred in the spindle amplifier unit for the serial spindle. For details, refer to the Serial Spindle User's Manual.
SP0969	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0970	SPINDLE CONTROL ERROR	Initialization of spindle control ended in error.
SP0971	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0972	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0973	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0974	ANALOG SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0975	ANALOG SPINDLE CONTROL ERROR	An position coder error was detected on the analog spindle.
SP0976	SERIAL SPINDLE COMMUNICATION ERROR	The amplifier No. could not be set to the serial spindle amplifier.

Number	Message	Contents
SP0977	SERIAL SPINDLE COMMUNICATION ERROR	An error occurred in the spindle control software.
SP0978	SERIAL SPINDLE COMMUNICATION ERROR	A time-out was detected during communications with the serial spindle amplifier.
SP0979	SERIAL SPINDLE COMMUNICATION ERROR	The communications sequence was no longer correct during communications with the serial spindle amplifier.
SP0980	SERIAL SPINDLE AMP. ERROR	Defective SIC-LSI on serial spindle amplifier
SP0981	SERIAL SPINDLE AMP. ERROR	An error occurred during reading of the data from SIC-LSI on the analog spindle amplifier side.
SP0982	SERIAL SPINDLE AMP. ERROR	An error occurred during reading of the data from SIC-LSI on the serial spindle amplifier side.
SP0983	SERIAL SPINDLE AMP. ERROR	Could not clear on the spindle amplifier side.
SP0984	SERIAL SPINDLE AMP. ERROR	An error occurred during re-initialization of the spindle amplifier.
SP0985	SERIAL SPINDLE CONTROL ERROR	Failed to automatically set parameters
SP0986	SERIAL SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0987	SERIAL SPINDLE CONTROL ERROR	Defective SIC-LSI on the CNC
SP0988	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0989	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0996	ILLEGAL SPINDLE PARAMETER SETTING	Illegal spindle and spindle motor setting
SP0997	SPINDLE CONTROL ERROR	The newly selected spindle No. by the spindle selection function could not be reflected in parameters.
SP0998	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.
SP0999	SPINDLE CONTROL ERROR	An error occurred in the spindle control software.

F.10 OH ALARM (ALARMS RELATED TO OVERHEAT)

Number	Message	Contents
OH0001	LOCKER OVERHEAT	CNC cabinet overheat
OH0002	FAN MOTOR STOP	OCB cooling fan motor abnormality

INDEX

<u>Revision Record</u>

FANUC Series 15*i*/15*i*-MB OPERATOR'S MANUAL (PROGRAMMING) (B-63784E)

01	Jan., 2002	—			
Edition	Date	Contents	Edition	Date	Contents