



YASNAC PC NC Programming Manual

Version: Beta 1.0

SAFETY INFORMATION

PRECAUTIONS

1. Read this instruction manual in its entirety before using the operating functions available in the YASNAC PCNC.
2. The following warning symbols are used to indicate precautions that the user must be aware of to safely use this equipment. Failure to follow these precautions can result in serious or possibly even fatal injury and damage to products or related equipment or systems.

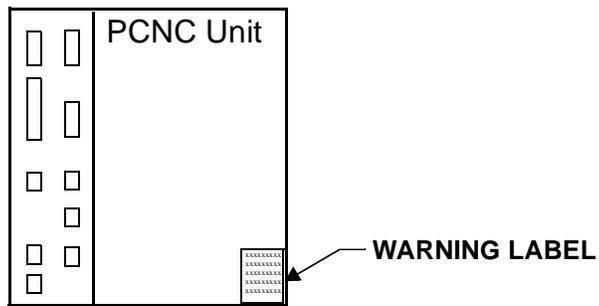


This symbol indicates the presence of a potentially *hazardous condition* which, if not avoided, could result in serious personal injury or death.



This precautionary symbol appears in labels attached to YASNAC products to alert the user to conditions requiring concern for safety.

SPECIAL SAFETY NOTE: This symbol indicates that **ELECTRICAL SHOCK HAZARD** condition exists. **DO NOT TOUCH** any electrical connection terminals when the power is on, and for at least 5 minutes **after** switching off the power supply. Warning label is located on the PCNC



NOTICE

Printed _____. 1999. The information contained within this document is the proprietary property of Yaskawa Electric America, Inc., and may not be copied, reproduced or transmitted to other parties without the expressed written authorization of Yaskawa Electric America, Inc.

No patent liability is assumed with respect to the uses of the information contained herein. Moreover, because Yaskawa is constantly improving its high quality product, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this document.

INFORMATION INDICATORS

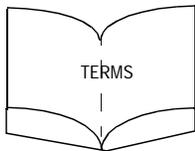
The following symbols are used in this operating manual to emphasize particular information to the user:



Indicates important information to be remembered, i.e., precautionary alarm displays to prevent damaging devices.



Indicates supplementary material.



Indicates definitions of terminology that has not been explained before.

NOTES REGARDING SAFE OPERATION

It is important that the user should read this manual before installing, operating, performing any maintenance or inspecting the *YASNAC PC NC*. Also, the functions and performance of a NC machine tool are not determined by the CNC unit itself, therefore thoroughly read and familiarize yourself with the machine builder's documentation concerning the safe and most efficient ways to use the machine tool. Nevertheless, Yasakawa assumes no responsibility for damages resulting from the use of the information contained within this publication.

Table of Contents

1.	PROGRAMMING BASICS	
1.1	FUNDAMENTALS OF PROGRAMMING TERMINOLOGY	1-2
1.1.1	Numerically Controlled Axes and the Number of Simultaneously Controllable Axes	1-2
1.1.2	Least Input Increment and Least Output Increment	1-6
1.1.3	Maximum Programmable Values for Axis Movement	1-8
1.1.4	Optional Block Skip (/1), (/2 to /9) *	1-11
1.1.5	Buffer Register and Multi-active Register	1-12
1.2	BASICS OF FEED FUNCTION	1-13
1.2.1	Rapid Traverse	1-13
1.2.2	Cutting Feed (F Command)	1-13
1.2.3	F1-Digit Feed*	1-16
1.2.4	Feed per Minute Function (G94)	1-18
1.2.5	Solid Tap Mode (G93, G94) *	1-18
1.2.6	Automatic Acceleration and Deceleration	1-19
2.	COMMAND CALLING AXIS MOVEMENTS	
2.1	INTERPOLATION COMMANDS	2-2
2.1.1	Positioning (G00, G06, G60)	2-2
2.1.2	Linear Interpolation (G01)	2-4
2.1.3	Circular Interpolation (G02, G03)	2-5
2.1.4	Helical Interpolation (G02, G03)*	2-11
2.2	REFERENCE POINT RETURN	2-13
2.2.1	Automatic Return to Reference Point (G28)	2-13
2.2.2	Reference Point Return Check (G27)	2-17
2.2.3	Return from Reference Point Return (G29)	2-18
2.2.4	Second to Fourth Reference Point Return (G30)	2-22

3. MOVEMENT CONTROL COMMANDS

3.1	SETTING THE COORDINATE SYSTEM	3-3
3.1.1	Selecting the Coordinate System	3-3
3.1.2	Base Coordinate System (G92)	3-3
3.1.3	Workpiece Coordinate System (G54 to G59)*	3-5
3.1.4	Local Coordinate System (G52 Q2)*	3-12
3.1.5	Machine Coordinate System (G53)	3-13
3.1.6	Rotation of Coordinate System (G68, G69)*	3-16
3.1.7	Plane Selection (G17, G18, G19)	3-18
3.2	DETERMINING THE COORDINATE VALUE INPUT MODES	3-19
3.2.1	Absolute/Incremental Designation (G90, G91)	3-19
3.2.2	Inch/Metric Input Designation (G20, G21)	3-21
3.2.3	Scaling (G50, G51) *	3-22
3.3	TIME-CONTROLLING COMMANDS	3-26
3.3.1	Dwell (G04)	3-26
3.3.2	Exact Stop (G09)	3-26
3.3.3	Exact Stop Mode (G61, G64)	3-26
3.4	TOOL OFFSET FUNCTIONS	3-28
3.4.1	Tool Offset Data Memory	3-28
3.4.2	Tool Length Offset (G43, G44, G49)	3-29
3.4.3	Tool Position Offset (G45 to G48)	3-32
3.4.4	Radius Offset C Function (G40, G41, G42) *	3-40
3.4.5	3-Dimensional Tool Offset Function (G40, G41, G42)*	3-78
3.5	S, T, M, AND B Functions	3-85
3.5.1	Spindle Function (S Function)	3-85
3.5.2	Tool Function (T Function)	3-86
3.5.3	Miscellaneous Function (M Function)	3-87
3.5.4	Second Miscellaneous Function (B Function) *	3-89

4.1	PROGRAM SUPPORT FUNCTIONS (1)	4-3
4.1.1	Canned Cycles (G73 to G89, G181 to G189) *	4-3
4.1.2	Hole Machining Pattern Cycles (G70, G71, G72) *	4-32
4.1.3	Solid Tap Function (G84, G74) *	4-36
4.1.4	Deep-hole Solid Tap Function (G184, G174)*	4-46
4.1.5	Circle Cutting Function (G12, G13)	4-57
4.1.6	Mirror Image ON/OFF (M94, M95) *	4-61
4.1.7	Programmable Data Input (G10) *	4-64
4.1.8	Subprogram Call Up Function (M98, M99)	4-67
4.2	PROGRAM SUPPORT FUNCTIONS (2)	4-69
4.2.1	Program Copy (G25)*	4-69
4.2.2	Automatic Corner Override (G106) *	4-72
4.2.3	Stored Stroke Limit B and C (G22, G23) *	4-77
4.2.4	Break Point Function	4-82
4.2.5	High-speed Cutting *	4-82
4.2.6	Chamfering and Corner Rounding Commands *	4-85
4.2.7	Corner Feedrate Designation (G107, G108)'	4-89
4.3	AUTOMATING SUPPORT FUNCTIONS	4-102
4.3.1	Skip Function (G31) *	4-102
4.3.2	Program interrupt Function (M90, M91)*	4-105
4.3.3	Tool Life Control Function *	4-107
4.4	MACROPROGRAMS	4-114
4.4.1	Differences from Subprograms	4-114
4.4.2	Microprogram Call (G65, G66, G67)	4-115
4.4.3	Variables	4-126
4.4.4	Operation Instructions	4-155
4.4.5	Control Instructions	4-157
4.4.6	Registering the Microprogram	4-163
4.4.7	Microprogram Alarm Numbers	4-164
4.4.8	Examples of Microprograms	4-165

USING THIS MANUAL

This manual describes the procedures for operating the *YASNAC PC NC*.

RELATED INFORMATION SOURCES

For additional information, refer to the following manuals:

TITLE OF DOCUMENT	CONTENTS
YASNAC PCNC Programming Manual (YEA-SIA-C844-2.2)	PCNC Program creation instructions
YASNAC PCNC/PLC Programming Manual (YEA-SIA-C844-0.1)	PLC Program creation instructions
YASNAC PCNC I/O Signal Manual (YEA-SIA-C844-2.3)	Describes functions between PCNC and PLC
YASNAC PCNC Connection Manual (YEA-SIA-C844-0.2)	Instructions for connecting PCNC with machines, machine interface and peripheral equipment
YASNAC PCNC Maintenance Manual (YEA-SIA-C844-2.9)	Describes service and maintenance procedures.

CAUTIONS

This manual describes all the option functions (identified by the “*” symbol) but some of these may not be available with your YASNAC PCNC. To determine the option functions installed in your PCNC, refer to the specification document or manuals published by the machine tool builder.

Unless otherwise specified, the following conditions apply in programming explanations and programming examples.

- Metric system for input and metric system for output / movement
- Zero point in the base coordinate system
- Reference point

Yaskawa has made every effort to describe individual functions and their relationships to other functions as accurately as possible. However, there are many things that cannot or must not be performed and it is not possible to describe all of these. Accordingly, readers are requested to understand that unless it is specifically stated that something can be performed, it should be assumed that it cannot be performed.

Also, bear in mind that the performance and functions of an PCNC machine tool are not determined solely by the PCNC unit. The entire control system consists of the mechanical system, then machine operation panel and other machine related equipment in addition to the PCNC. Therefore, read the manuals published by the machine tool builder for detailed information relating to the machine.

1

Programming Basics

Chapter 1 describes the basic terms used in programming and the feed functions.

1.1	FUNDAMENTALS OF PROGRAMMING TERMINOLOGY	1-2
1.1.1	Numerically Controlled Axes and the Number of Simultaneously Controllable Axes	1-2
1.1.2	Least Input Increment and Least Output Increment	1-6
1.1.3	Maximum Programmable Values for Axis Movement	1-8
1.1.4	Optional Block Skip (/1), (/2 to /9) *	1-11
1.1.5	Buffer Register and Multi-active Register	1-12
1.2	BASICS OF FEED FUNCTION	1-13
1.2.1	Rapid Traverse	1-13
1.2.2	Cutting Feed (F Command)	1-13
1.2.3	F1-Digit Feed*	1-16
1.2.4	Feed per Minute Function (G94)	1-18
1.2.5	Solid Tap Mode (G93, G94) *	1-18
1.2.6	Automatic Acceleration and Deceleration	1-19

1.1 FUNDAMENTALS OF PROGRAMMING TERMINOLOGY

This section describes the “basic terms used in programming.

1.1.1 Numerically Controlled Axes and the Number of Simultaneously Controllable Axes

(1) Numerically Controlled Axes and Axes Names

The numerically controlled axes and the axis names are indicated in Table 1.1.1.1.

Table 1.1.1.1: Numerically Controlled Axes

Controlled Axis	Axis Name	Model Name Description
Basic axes	X, Y, Z	Represents the coordinate position or distance in or along an axis indicated by X, Y, and Z.
* 4th and 5th axes	A, B, C or U, V, W	Represents the commands of the fourth and fifth axes. For rotary motion, address characters A, B, and C are used and for parallel (linear) motion, address characters U, V, and W are used.

(2) Number of Simultaneously Controllable Axes

With the standard specification, up to three axes can be controlled simultaneously. This number can be increased optionally to four and five axes.

(a) Number of simultaneously controllable axes with the 3-axis control function

The number of simultaneously controllable axes is indicated in Table 1.2.

Table 1.1.1.2: The Number of Simultaneously Controllable Axes with 3-axis Control Function

	Number of Simultaneously Controllable Axes
Positioning (G00)	3 axes (X-, Y-, and Z-axis)
Linear interpolation (G01)	3 axes (X-, Y-, and Z-axis)
Circular interpolation (G02, G03)	2 axes (X- and Y-axis, Y- and Z-axis, or Z- and X-axis)
*Circle cutting (G12, G13)	2 axes (X- and Y-axis)
*Helical interpolation (G02, G03)	2 axes (circular interpolation in XY plane) 1 axis (linear interpolation, Z-axis) } Simultaneous See 2.1.4, “Helical Interpolation (G02, G03)”. } 3-axis control
Manual operation	3 axes (X-, Y-, and Z-axis)

Note 1: The plane in which circular interpolation is executed is determined by the plane selection G code (G 17 to G19) which is presently valid. For details, see 2.1.3, “Circular Interpolation (G02, G03)”.

2: With a manual pulse generator, simultaneous control is possible in either one or three axes.

(b) Number of simultaneously controllable axes with the 4-axis control function*

The fourth axis can be selected optionally. In this manual, the fourth axis is referred to as “a-axis” and represents any of six axes – A, B, C, U, V, and W. Which address characters should be used for the fourth axis is set for parameters pm1109, pm1110, and pm1111, and pm1151, pm1152, and pm1153. The number of simultaneously controllable axes is indicated in Table 1.3.

Table 1.1.1.3: The Number of Simultaneously Controllable Axes with 4-axis Control Function

	Number of Simultaneously Controllable Axes
Positioning (G00)	4 axes (X-, Y-, Z-, and a-axis)
Linear interpolation (G01)	4 axes (X-, Y-, Z-, and a-axis)
Circular interpolation (G02, G03)	2 axes (X- and Y-axis, Y- and Z-axis, or Z- and X-axis) 2 axes (X- and a-axis, Y- and a-axis, or Z- and a-axis)
*Circle cutting (G12, G13)	2 axes (X- and Y-axis)
*Helical interpolation (G02, G03)	2 axes (circular interpolation in XY plane) 1 axis (linear interpolation, Z-axis) } Simultaneous See 2.1.4, “Helical Interpolation (G02, G03)”. } 3-axis control
Manual operation	4 axes (X-, Y-, and a-axis)

Note 1: If “a” is included in circular interpolation, it must be a linear axis (U, V, or W). The plane in which circular interpolation is executed is determined by the plane selection G code (G17 to G19) which is presently valid. For details, see 2.1.3, “Circular Interpolation (G02, G03)”.

2: With a manual pulse generator, simultaneous control is possible in either one or three axes.

For the a-axis, either a rotary axis or a linear axis can be selected.

- A rotary axis (A-, B-, or C-axis) is defined as indicated in Table 1.4.

Table 1.1.1.4: Rotary Axes

Rotary Axis	Definition
A-axis	Rotary axis around an axis which is parallel to X-axis
B-axis	Rotary axis around an axis which is parallel to Y-axis
C-axis	Rotary axis around an axis which is parallel to Z axis

Note 1: The unit of output increment (motion increment) and input increment for a rotary axis is “degrees” instead of “mm” which is used for a linear axis (X-, Y-,Z-axis). With the exception of the unit, a rotary axis can be treated in the same manner as a linear axis. (Metric system) (The NC circulates feedrate assuming 0.001 deg. as 0.001 mm.)

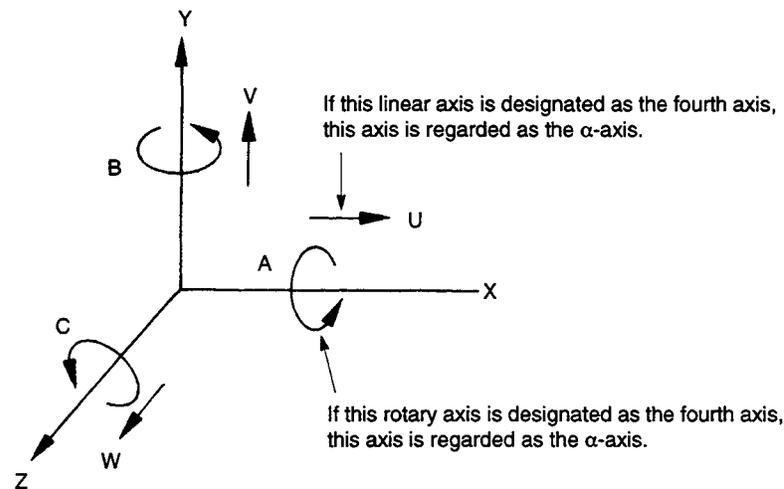
2: Even if the dimensions are changed to inches by using the inch/mm selection function, the unit system for a rotary axis remains unchanged (degrees).

- A linear axis (U-, V-, or W-axis) is defined as indicated in Table 1.5.

Table 1.1.1.5: Linear Axes

Linear Axis	Definition
U-axis	A linear axis parallel to X-axis.
V-axis	A linear axis parallel to Y-axis.
W-axis	A linear axis parallel to Z-axis.

- Note 1: The unit of output increment (motion increment) and input increment of linear axis is the same as other linear axes (X-, Y-, and Z-axis). The linear axes indicated above can be treated in exactly the same manner as other linear axes.
- 2: When the inch system is selected by using inch/mm selection function, dimensions must be input in units of inches as with other axes (X-, Y-, and Z-axis)

**Fig. 1.1.1.1 Fourth Axis (α -axis) in the Right-hand Coordinate System**

(c) Number of simultaneously controllable axes with the 5-axis control function*

The fifth axis can be selected optionally. In this manual, the fifth axis is referred to as "b-axis" and represents any of six axes – A, B, C, U, V, and W. Which address characters should be used for the fifth axis is set for parameters pm1112, pm1113, and pm1114, and pm1154, pm1155, and pm1156. The number of simultaneously controllable axes is indicated in Table 1.6.

Table 1.1.1.6: The Number of Simultaneously Controllable Axes with 5-axis Control Function

	Number of Simultaneously Controllable Axes
Positioning (G00)	5 axes (X-, Y-, Z-, α -, and β -axis)
Linear interpolation (G01)	5 axes (X-, Y-, Z-, α -, and β -axis)
Circular interpolation (G02, G03)	2 axes (X- and Y-axis, Y- and Z-axis, or Z- and X-axis) 2 axes (X- and a-axis, Y- and α -axis, or Z- and α -axis) 2 axes (X- and b-axis, Y- and β -axis, or Z- and β -axis)
*Circle cutting (G12, G13)	2 axes (X- and Y-axis)
*Helical interpolation (G02, G03)	2 axes (circular interpolation in XY plane) 1 axis (linear interpolation, Z-axis) See 2.1.4, "Helical Interpolation (G02, G03)". } Simultaneous 3-axis control
Manual operation	5 axes (X-, Y-, Z-, and α -axis)

Note 1: Circular interpolation is possible only when a- and b-axis are linear axes. The plane in which circular interpolation is executed is determined by the plane selection G code (G17 to G19) which is presently valid. For details, see 2.1.3, "Circular interpolation (G02, G03)".

2: With a manual pulse generator, simultaneous control is possible in either one or three axes.

For the b-axis, either a rotary axis or a linear axis can be selected.

- A rotary axis (A-, B-, or C-axis) is defined as indicated in Table 1.1.1.7.

Table 1.1.1.7: Rotary Axes

Rotary Axis	Definition
A-axis	Rotary axis around an axis which is parallel to X-axis
B-axis	Rotary axis around an axis which is parallel to Y-axis
C-axis	Rotary axis around an axis which is parallel to Z axis

Note 1: The unit of output increment (motion increment) and input increment for a rotary axis is "degrees" instead of "mm" which is used for a linear axis (X-, Y-, Z-axis). With the exception of the unit, a rotary axis can be treated in the same manner as a linear axis. (Metric system) (The NC calculates feedrate assuming 0.001 deg. as 0.001mm.)

2: Even if the dimensions are changed to inches by using the inch/mm selection function, the unit system for a rotary axis remains unchanged (degrees).

- A linear axis (U-, V-, or W-axis) is defined as indicated in Table 1.1.1.8.

Table 1.1.1.8: Linear Axes

Linear Axis	Definition
U-axis	A linear axis parallel to X-axis.
V-axis	A linear axis parallel to Y-axis.
W-axis	A linear axis parallel to Z-axis.

- Note 1: The unit of output increment (motion increment) and input increment of a linear axis is the same as other linear axes (X-, Y-, and Z-axis). The linear axes indicated above can be treated in exactly the same manner as other linear axes,
- 2: When the inch system is selected by using inch/nm selection function, dimensions must be input in units of inches as with other axes (X-, Y-, and Z-axis).

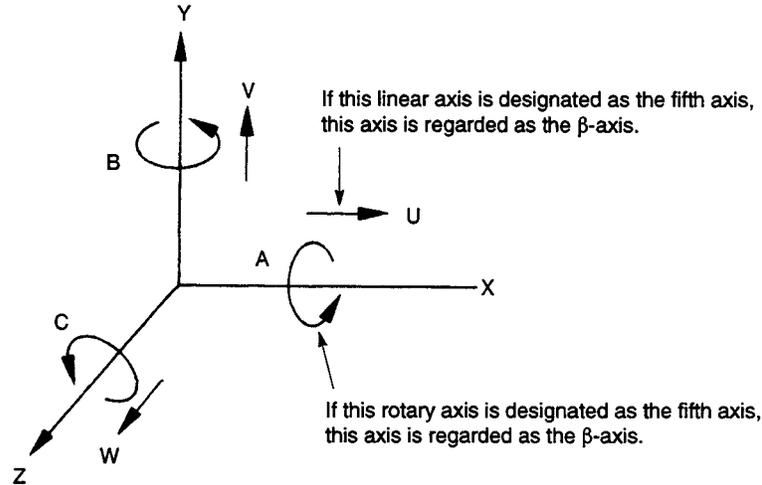


Fig. 1.1.1.2 Fifth Axis (b-axis) in the Right-hand Coordinate System

1.1.2 Least Input Increment and Least Output Increment

The least input and output increments vary depending on the type of controlled axis whether it is a rotary axis or a linear axis.

(1) Least Input Increment

The least input increment to express axis movement distance that is input by using punched tape or manual data input switches is indicated in Tables 1.9, 1.10, and 1.11.

Table 1.1.2.1: Least Increment (Standard)

	Linear Axis	*Rotary Axes
Metric Input	0.001 mm	0.001 deg.
Inch Input	0.0001 inch	0.001 deg.

Table 1.1.2.2: Least Increment (Sub Microns)

	Linear Axis	*Rotary Axes
Metric Input	0.0001 mm	0.001 deg.
Inch Input	0.00001 inch	0.001 deg.

Table 1.1.2.3: Least Increment (Sub Sub-microns)

	Linear Axis	*Rotary Axes
Metric Input	0.00001 mm	0.001 deg.
Inch Input	0.000001 inch	0.001 deg.

Note: Selection of “mm-input” and “inch-input” is made by the setting parameter pm0007 D0.

(2) Least Output Increment

The least output increment indicates the “minimum unit” of axis movement that is determined by the mechanical system. By selecting the option, it is possible to select the output unit system between “mm” and “inches”.

Table 1.1.2.4: Least Output Unit (Standard)

	Linear Axis	*Rotary Axes
Metric Output	0.001 mm	0.001 deg.
Inch Output	0.0001 inch	0.001 deg.

Table 1.1.2.5: Least Increment (Sub Microns)

	Linear Axis	*Rotary Axes
Metric Output	0.0001 mm	0.001 deg.
Inch Output	0.00001 inch	0.001 deg.

Table 1.1.2.6: Least Increment (Sub Sub-microns)

	Linear Axis	*Rotary Axes
Metric Output	0.00001 mm	0.001 deg.
Inch Output	0.000001 inch	0.001 deg.

1.1.3 Maximum Programmable Values for Axis Movement

The maximum programmable values that can be designated for a move command are indicated in Tables 1.15, 1.16, and 1.17. The maximum programmable values indicated in these tables are applicable to addresses I, J, K, R, and Q which are used for designating “distance” in addition to the move command addresses X, Y, Z, a, and b.

Table 1.1.3.1: Maximum Programmable Values for Axis Movement (Standard)

		Linear Axis	*Rotary Axes
Metric Output	Metric Input	±999999.999 mm	±999999.999 deg.
	Inch Input	±39370.0787 mm	±999999.999 deg.
Inch Output	Metric Input	±999999.999 mm	±999999.999 deg.
	Inch Input	±999999.999 mm	±999999.999 deg.

Table 1.1.3.2: Maximum Programmable Values for Axis Movement (Sub-microns)

		Linear Axis	*Rotary Axes
Metric Output	Metric Input	±999999.999 mm	±999999.999 deg.
	Inch Input	±39370.0787 mm	±999999.999 deg.

Table 1.1.3.3: Maximum Programmable Values for Axis Movement (Sub Sub-microns)

		Linear Axis	*Rotary Axes
Metric Output	Metric Input	±999999.999 mm	±999999.999 deg.
Inch Output	Inch Input	±39370.0787 mm	±999999.999 deg.

In incremental programming, the values to be designated must not exceed the maximum programmable values indicated above. In absolute programming, the move distance of each axis must not exceed the maximum programmable values indicated above. In addition to the notes indicated above, it must also be taken into consideration that the cumulative values of move command must not exceed the values indicated in Tables 1.18, 1.19, and 1.20.

Table 1.1.3.4: Maximum Cumulative Values (Standard)

	Linear Axis	*Rotary Axes
Metric Input	±999999.999 mm	±999999.999 deg.
Inch Input	±999999.999 inch	±999999.999 deg.

Table 1.1.3.5: Maximum Programmable Values for Axis Movement (Sub-microns)

	Linear Axis	*Rotary Axes
Metric Input	±999999.999 mm	±999999.999 deg.

Table 1.1.3.6: Maximum Programmable Values for Axis Movement (Sub Sub-microns)

	Linear Axis	*Rotary Axes
Metric Input	±999999.999 mm	±999999.999 deg.

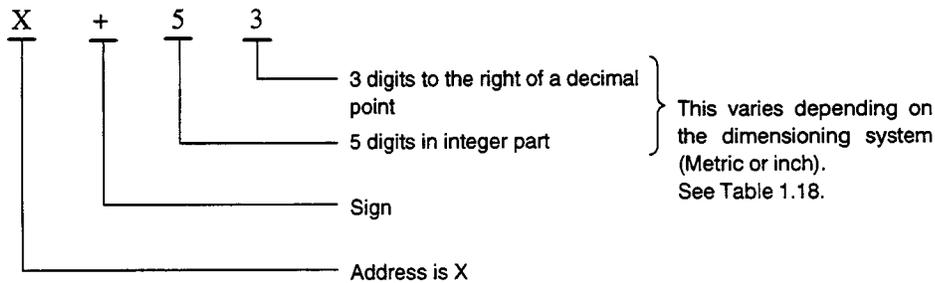
Note: The values indicated above do not depend on the “least output increment”.

(1) Programmable Range (Input Format)

This model of NC adopts the variable block format which complies with JIS B6313.

Programmable range of individual addresses is indicated in Table 1.1.3.7. The numbers given in this table indicate the allowable maximum number of digits.

An example of input format is given below.



Input data should be entered without a decimal point. If a decimal point is used, the entered values is treated in a different manner. Leading zeros and the “+” (plus) sign can be omitted for all kinds of address data including sequence number. Note that, however, the “-” (minus) sign cannot be omitted.

Table 1.1.3.7: Input Format (Standard)

Address	Metric Output		Inch Output		B: Basic O: Option	
	Metric Input	Metric Input	Metric Input	Metric Input		
Program number	O5		O5		B	
Sequence number	N5		N5		B	
G function	G3		G3		B	
Coordinate words	Linear axis	a+63	a+54	a+63	a+54	B
	Rotary axis	b+63	b+63	b+63	b+63	O
Feed per minute (mm/min)	F60	F41	F60	F51	B	
Feed per minute (mm/min) 1/10 function	F61	F42	F61	F52	B	
S function	S 5		S5		B	
T function	T 2		T2		B	
	T 4		T4		O	

M function	M 3	M3	B
Tool offset number	H4 or D4	H4 or D4	B
B function	B 3	B 3	O
Dwell	P 63	P 63	B
Program number designation	P 5	P 5	B
Sequence number designation	P 4	P 4	B
Number of repetitions	L 9	L 9	B

Table 1.1.3.8: Input Format (Sub Microns)

Address		Metric Output		B: Basic O: Option
		Metric Input	Metric Input	
Program number		O5		B
Sequence number		N5		B
G function		G3		B
Coordinate words	Linear axis	a+54	a+45	B
	Rotary axis	b+54	b+54	O
Feed per minute (mm/min)		F 51	F 32	B
Feed per minute (mm/min) 1/10 function		F 52	F 33	B
S function		S 5		B
T function		T 2		B
		T 4		O
M function		M 3		B
Tool offset number		H4 or D4		B
B function		B 3		O
Dwell		P 63		B
Program number designation		P 5		B
Sequence number designation		P 4		B
Number of repetitions		L 9		B

Table 1.1.3.9: Input Format (Sub Sub-microns)

Address		Metric Output		B: Basic O: Option
		Metric Input	Metric Input	
Program number		O5		B
Sequence number		N5		B
G function		G3		B
Coordinate words	Linear axis	a+45	a+36	B
	Rotary axis	b+45	b+45	O
Feed per minute (mm/min)		F 42	F 23	B

Feed per minute (mm/min) 1/10 function	F 43	F 24	B
S function	S 5		B
T function	T 2		B
	T 4		O
M function	M 3		B
Tool offset number	H4 or D4		B
B function	B 3		O
Dwell	P 63		B
Program number designation	P 5		B
Sequence number designation	P 4		B
Number of repetitions	L 9		B

1.1.4 Optional Block Skip (/1), (/2 to /9) *

If a block containing the slash code “/n (n = 1 to 9)” is executed with the external optional block skip switch corresponding to the designated number set ON, the commands in the block following the slash code to the end of block code are disregarded. The slash code “/n” can be designated at any position in a block.

Example:

```
/ 2 N 1234 G01X100 / 3 Y200;
```

If the “/2” switch is ON, the entire block is disregarded, and

if “/3” switch is ON, this block indicates the following.

```
N 1234 G01 X100;
```



1. “1” can be omitted for “1”.
2. The optional block skip function is processed when a part program is read to the buffer register from either the tape or memory. If the switch is set ON after the block containing the optional block skip code is read, the block is not skipped.
3. The optional block skip function is disregarded for program reading (input) and punch out (output) operation.

1.1.5 Buffer Register and Multi-active Register

By using the buffer register and multi-active register, the NC ensures smooth control of the machine by reading the blocks of data into the buffer register.

(1) Buffer Register

In normal operation, two blocks of data are buffered to calculate the offset and other data that are necessary for the succeeding operation.

In the tool radius offset C mode (option), two blocks of data (a maximum of four blocks of data, if necessary) are buffered to calculate the offset data that are necessary for the succeeding operation. In both of the normal operation mode and tool radius offset C mode, the data capacity of one block is a maximum of 128 characters, including the EOB code.

(2) Multi-active Registers *

With a part program enclosed by M93 and M92, a maximum of seven blocks of data are buffered. If the time required for automatic operation of these seven buffered blocks is longer than the time required for the buffering and calculation of the offset data for the next seven blocks, the program can be executed continuously without a stop between blocks.

Table 1.1.5.1: M92 and M93 Codes

Linear Axis	Definition
M92	Multi-active registers OFF
M93	Multi-active registers ON

1.2 BASICS OF FEED FUNCTION

This section describes the feed function that specifies feedrate (distance per minute, distance per revolution) of a cutting tool.

1.2.1 Rapid Traverse

Rapid traverse is used for positioning (G00) and manual rapid traverse (RAPID) operation. In the rapid traverse mode, each axis moves at the rapid traverse rate set for the individual axes; the rapid traverse rate is determined by the machine tool builder and set for the individual axes by using parameters. Since the axes move independently of each other, the axes reach the target point at different time. Therefore, the resultant tool paths are not a straight line generally.

The rapid traverse override function can adjust the set rapid traverse rate to F_0 , 25%, 50%, and 100% where F_0 indicates a fixed feedrate set for parameter pm2447.



1. Rapid traverse rate is set in the following units for the individual axes.

Setting units of rapid traverse rate	1 mm/min
	0.1 inch/min
	1 deg./min

2. The upper limit of the rapid traverse rate is 240,000 mm/min. Since the most appropriate value is set conforming to the machine capability, refer to the manuals published by the machine tool builder for the rapid traverse rate of your machine.

1.2.2 Cutting Feed (F Command)

The feedrate at which a cutting tool should be moved in the linear interpolation (G01) mode or circular interpolation (G02, G03) mode is designated using address character F.

- With a 6-digit numeral specified following address character F, feedrate of a cutting tool can be designated in units of “mm/min”.
- The feedrate specified using an F code can be multiplied by 1/10 by changing the setting for parameter pm2004 D0. The programmable feedrate range is indicated in Table 1.2.2.1

Table 1.2.2.1: Programmable Range of F Code

Input System	Normal Mode (pm2004 DO=0)		F-command 1/10 Function (pm2004 DO = 1)	
	Format	Programmable Range (Feed per Minute)	Format	Programmable Range (Feed per Minute)
Microns	F60	F1.0-F 24000 mm/min	F61	F0.1-F 24000.0 mm/min
Sub Microns	F51	F0.1-F 24000.0 mm/min	F52	F0.01-F 24000.00 mm/min
Sub Sub-microns	F42	F0.01-F 2400.00 mm/min	F43	F0.001-F 2400.000 mm/min
Inches	F41	F0.1-F 9448.8 inch/min	F42	F0.01-F 9448.81 mm/min

- The upper limit of feedrates indicated in Table 1.2.2.1 could be restricted by the servo system and the mechanical system. In this case, the allowable upper limit is set for parameter (pm2800) and if a feedrate command exceeding this limit value is specified, the feedrate is clamped at the set allowable upper limit.
- An F command specified in the simultaneous 2-axis linear interpolation mode or in the curricular interpolation mode represents the feedrate in the tangential direction.

Example of Programming

With the following program:

G91 (incremental programming)

G01 X40. Y30. F500;

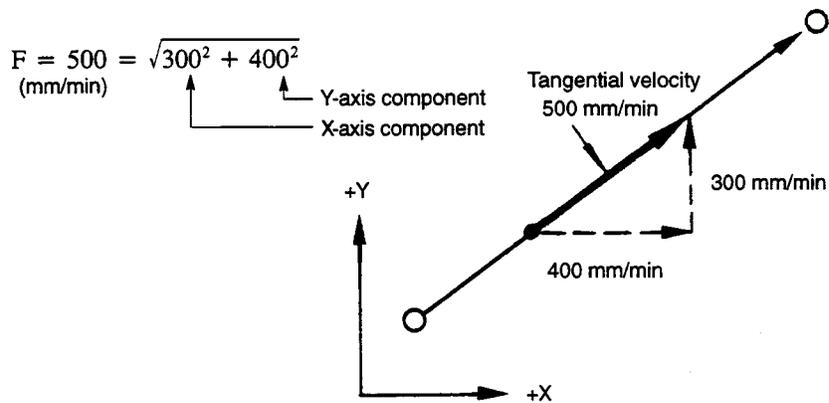


Fig. 1.2.2.1 F command in Simultaneous 2-axis Control Linear Interpolation

Example of Programming

With the following program:
G91 (incremental programming)
G03 X ... Y ... I ... F200;

$$F = 200 = \sqrt{F_x^2 + F_y^2}$$

(mm/min)

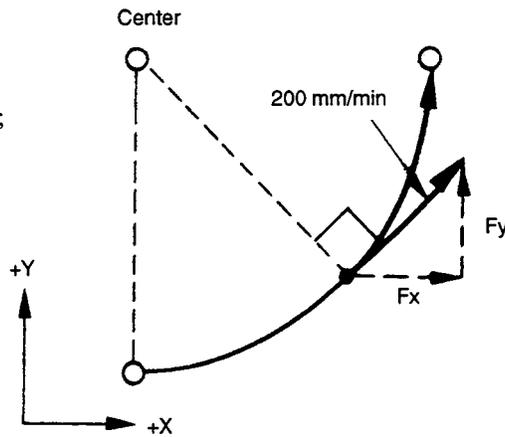


Fig. 1.2.2.2 F command in the Simultaneous 2-axis Control Circular Interpolation

- In the simultaneous 3-axis control linear interpolation, an F command indicates the tangential feedrate.

Example of Programming

With the following program:
 G01 X ... Y ... Z ... F400;
 $F = 400 = \sqrt{F_x^2 + F_y^2 + F_z^2}$
 (mm/min)

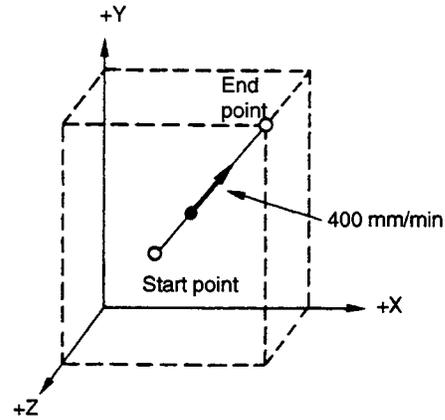


Fig. 1.2.2.3 F Command in Simultaneous 3-axis Control Linear Interpolation

- In the simultaneous 4-axis control* linear interpolation, an F command indicates the tangential feedrate.

$$F \text{ (mm/min)} = \sqrt{F_x^2 + F_y^2 + F_z^2 + F_\alpha^2}$$

- In the simultaneous 5-axis control* linear interpolation, an F command indicates the tangential feedrate.

$$F \text{ (mm/min)} = \sqrt{F_x^2 + F_y^2 + F_z^2 + F_\alpha^2 + F_\beta^2}$$



1. The F-command 1/10 function does not influence the feedrate called by an F1-digit command.
2. After changing the setting for parameter pm2004 D0, the new setting becomes valid when the NC is reset.
3. During solid tapping, the F-command 1/10 function does not influence the feedrate called by an F command.
4. The feedrate specified by an E code in a canned cycle is influenced by the F-command 1/10 function. The command format of an E command is the same as with an F command.
5. When the F-command 1/10 function is used, the minimum unit of the system variables used for E and F commands is made one decimal place smaller. In metric input, if the least increment of the F command system variable is 1 mrm/min, for example, it becomes 0.1 mm/min when the F-command 1/10 function is used.
6. When the F-command 1/10 function is used, designation of the macro system variables of E and F commands and the arguments (E, F) used for calling a macro program requires entry of a decimal fraction increased by one digit. In metric input, the command of “G65 PI F1234”, for example, is expressed as “#9=123.4”.
7. If “F0” is specified, alarm “0370” occurs.
8. For an F command, a minus value must not be specified. If a minus value is specified for an F command, correct operation cannot be guaranteed.

1.2.3 F1-Digit Feed*

It is possible to select a feedrate by specifying a 1-digit numeral (1 to 9) following address F. With this manner of designation of an F command, the feedrate preset for the specified numeral is selected. The feedrate to be selected in response to the designation of F1 to F9 should be set for the parameters indicated in Table 1.2.3.1.

Table 1.2.3.1: Parameter Numbers Used for Presetting F1-digit Feedrates

F Command	Parameter Numbers
F1	pm0820
F2	pm0821
F3	pm0822
F4	pm0823
F5	pm0824
F6	pm0825
F7	pm0826
F8	pm0827
F9	pm0828

Note: Value1= 0.1 mm/min, or 0.01 inch/min

When using the F1-digit feed function, it is possible to optimize the selected feedrate by turning the manual pulse generator while the F1-DIGIT switch is ON. Increase or decrease of increments per pulse (F1-digit multiply) should be set for the parameters indicated in Table 1.2.3.2.

Table 1.2.3.2: Parameter Numbers Used for Setting F1-digit Multiply for the para

F Command	Parameter Numbers
F1	pm2111
F2	pm2112
F3	pm2113
F4	pm2114
F5	pm2115
F6	pm2116
F7	pm2117
F8	pm2118
F9	pm2119

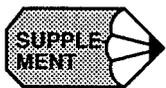
Note: Value “1”= 0.1 mm/min per pulse

If increase/decrease increments per pulse is set for these parameters, the value set for the parameters in Table 1.31 is updated in response to the manual pulse generator operation.

Parameters indicated in Table 1.33 are used to set the upper limits of the feedrate for F1-digit feedrate selection. If a value larger than the allowable maximum feedrate set for parameter pm2800 is set, it is disregarded and replaced with the value set for pm2800.

Table 1.2.3.3: Parameters pm2865 and pm2866

Parameter Numbers	Description
pm2865	Allowable maximum feedrate for F1 to F4
pm2866	Allowable maximum feedrate for F5 to F9



1. When the 1-digit numerals are set to the parameters pm0802 to pm0828, and pm2004 D0 = 0, feedrate on the screen is displayed as “0”. However, the machine moves in units of 0.1 to 0.9mm/min or 0.01 to 0.09 inch/min.

pm2004 D0 = 0	Feedrate at the deceleration of 0.001mm or 0.0001 inch is F6.0 mm/min or F4.1 inch/min.
pm2004 D0 = 1	Feedrate at the deceleration of 0.001mm or 0.0001 inch is F6.1 mm/min or F4.2 inch/min.

2. If “F0” is specified, alarm “0370” occurs.

3. When the DRY RUN switch is ON, feed commands are all executed at the feedrate set for the dry run operation.
4. The feed override function is invalid for the feedrate selected by the F1-digit command.
5. The feedrate set for the parameter is retained in memory if the power is turned OFF.
6. It is possible to designate an F1-digit command by specifying a variable in a macro program.
7. With the inch specification, feedrates are set in units of inch/min. However, the allowable maximum feedrates can be set only in units of mm/min.

1.2.4 Feed per Minute Function (G94)

When G94 is designated, a feedrate specified following address F is executed in units of “mm (inch)/min”.

1.2.5 Solid Tap Mode (G93, G94) *

The following G codes are used to indicate that tapping should be executed by using the solid tap function.

Table 1.2.5.1: Solid Tap Mode G Codes

G code	Description	Group
G93	Solid tap mode	05
G94	Solid tap mode cancel	05

G93 and G94 are modal G codes. When the power is turned ON or when the NC is reset, the G94 mode is automatically set.

(1) Solid Tap Mode (G93)

In the G93 solid tap mode, solid tapping is executed for the tapping cycles called by G84 or G74. Axis feed is controlled in the “feed per revolution” mode. In the solid tap mode, only solid tapping is allowed and no other operation is possible.

(2) Solid Tap Mode Cancel (G94)

When G94 is executed, the solid tap mode is canceled. After the cancellation of the solid tap mode, conventional type tapping is executed in which axis feed is controlled in the “feed per minute” mode.

1.2.6 Automatic Acceleration and Deceleration

Automatic acceleration/deceleration control is provided for rapid traverse and cutting feed operation, respectively.

(1) Acceleration and Deceleration for Rapid Traverse and Manual Axis Feed Operation

For positioning (G00), manual rapid traverse (RAPID), manual continuous feed (JOG), and manual handle feed (HANDLE), linear pattern automatic acceleration/deceleration is applied. Rapid traverse rate and acceleration/deceleration time constant for rapid traverse are set for following parameters.

Table 1.2.6.1: Parameters Used for Setting Rapid Traverse Rate and Acceleration/Deceleration Time Constant

G code	X-axis	Y-axis	Z-axis	4th-axis	5th-axis
Rapid traverse rate	pm2801	pm2802	pm2803	pm2804	m2805
Acceleration/deceleration time constant	pm2461	pm2462	prn2463	pm2464	pm2465

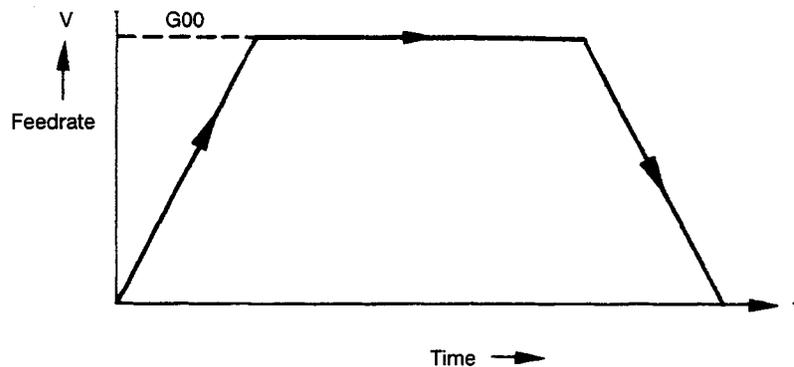


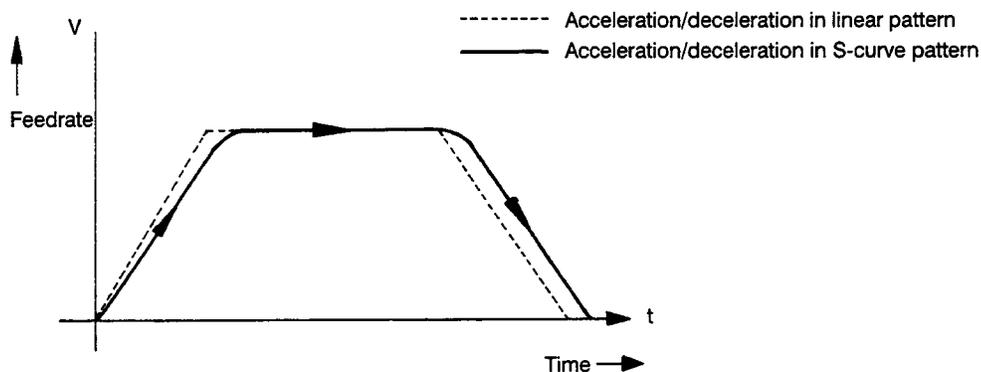
Fig. 1.2.6.1 Automatic Acceleration/Deceleration in Linear Pattern

(2) Acceleration and Deceleration in S-curve Pattern *

For positioning operation (G00 mode positioning), S-curve pattern can be selected for the acceleration/deceleration pattern instead of the linear pattern. By using the S-curve pattern, positioning is possible at a high acceleration/deceleration rate without applying shock to the machine. The S-curve pattern for rapid traverse is defined by the following parameters.

Table 1.2.6.2: S-curve Pattern Defining Parameters (for Rapid Traverse)

Parameter	Description	Setting range
pm2591	For rapid traverse of X-axis	0 to 20
pm2592	For rapid traverse of Y-axis	0 to 20
pm2593	For rapid traverse of Z-axis	0 to 20
pm2594	For rapid traverse of 4th-axis	0 to 20
pm2595	For rapid traverse of 5th-axis	0 to 20

**Fig. 1.2.6.2 Acceleration/Deceleration in S-curve Pattern**

For the S-curve pattern acceleration/deceleration, time constant is provided for the individual axes and setting is possible in the range from 0 to 20.

Table 1.2.6.3: Time Constant for S-curve Pattern Control

Setting Value	Explanation
0	Feedrate is controlled in the same pattern as acceleration/deceleration in the linear pattern.
1 to 20 (N)	The S-curve pattern having the time constant of "4 x N" is obtained. (Maximum time constant 60 msec) I
Less than 0	Regarded as "0".
Greater than 20	Regarded as "20".

(3) Acceleration and Deceleration for Cutting Feed

For cutting feed (G01 to G03 mode), feedrate is controlled by the automatic acceleration/deceleration in the exponential pattern.

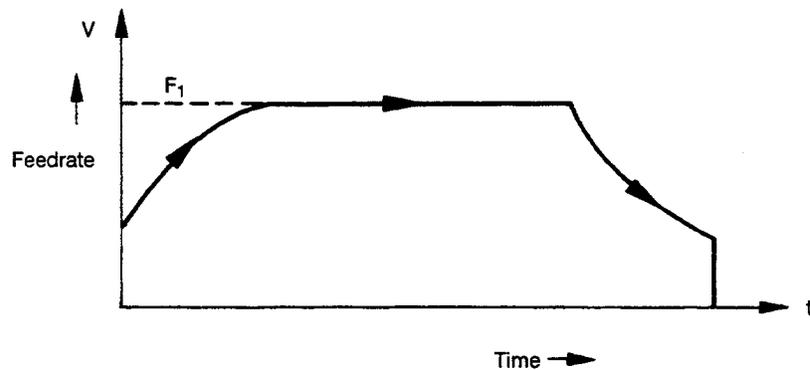


Fig. 1.2.6.3 Acceleration/Deceleration in Exponential Pattern

Time constant for cutting feed and feedrate bias are set for parameters. For tapping, time constant and feedrate bias can be set independently.

Table 1.2.6.4: Parameters for Tapping

G code	X-axis	Y-axis	Z-axis	4th-axis	5th-axis
Feedrate time constant	pm2501	pm2502	pm2503	pm2504	pm2505
Feedrate bias	pm2821	pm2822	pm2823	pm2824	pm2825
Tapping time constant	pm2511	pm2512	pm2513	pm2514	pm2515
Tapping feedrate bias	pm2831	pm2832	pm2833	pm2834	pm2835



1. For the parameters indicated above, the most optimum values are set for respective machines. Do not attempt to change the setting unless necessary.

2

Commands Calling Axis Movements

Chapter 2 describes the interpolation commands and the reference point return commands.

2.1	INTERPOLATION COMMANDS	2-2
2.1.1	Positioning (G00, G06, G60)	2-2
2.1.2	Linear Interpolation (G01)	2-4
2.1.3	Circular Interpolation (G02, G03)	2-5
2.1.4	Helical Interpolation (G02, G03)*	2-11
2.2	REFERENCE POINT RETURN	2-13
2.2.1	Automatic Return to Reference Point (G28)	2-13
2.2.2	Reference Point Return Check (G27)	2-17
2.2.3	Return from Reference Point Return (G29)	2-18
2.2.4	Second to Fourth Reference Point Return (G30)	2-22

2.1 INTERPOLATION COMMANDS

This section describes the positioning commands and the interpolation commands that control the tool path along the specified functions such as straight line and arc.

2.1.1 Positioning (G00, G06, G60)

In the absolute programming mode (G90), the axes are moved to the specified point in a work-piece coordinate system, and in the incremental programming mode (G91), the axes move by the specified distance from the present position at a rapid traverse rate.

For calling the positioning, the following G codes can be used,

Table 2.1.1.1 G Codes for Positioning

G code	Description	Group
G00	Positioning in the error detect ON mode	01
G06	Positioning in the error detect OFF mode	*
G60	Unidirectional positioning	01

(1) Positioning in the Error Detect ON Mode (G00)

When “G00X...Y...Z...;” is designated, positioning is executed in the “error detect ON” mode, in which the program advances to the next block only when the number of lag pulses due to servo lag are checked after the completion of pulse distribution has reduced to the permissible value.

In the G00 mode, positioning is made at a rapid traverse rate in the simultaneous 3-axis (*5-axis) control mode. The axes not designated in the G00 block do not move. In positioning operation, the individual axes move independently of each other at a rapid traverse rate that is set for each axis. The rapid traverse rates set for the individual axes differ depending on the machine. For the rapid traverse rates of your machine, refer to the manuals published by the machine tool builder.

Example of Programming

G00 X40. Y40. Z40.;

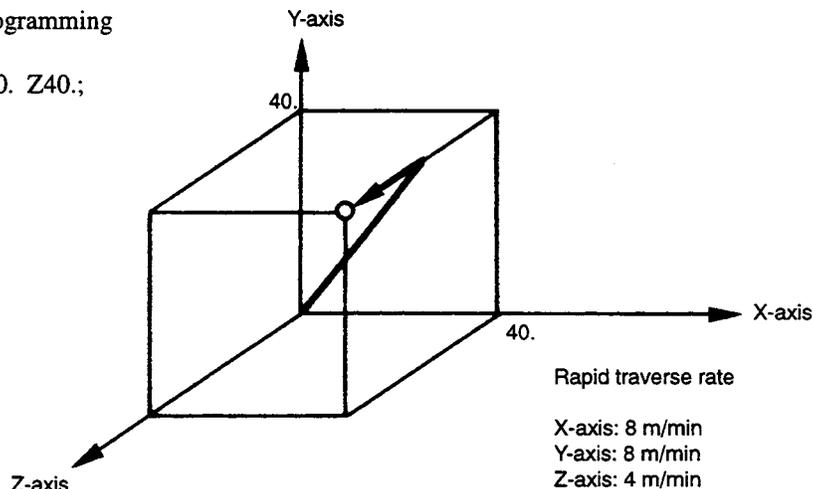


FIGURE 2.1.1.1 Positioning in Simultaneous 3-axis Control Mode



In the G00 positioning mode, since the axes move at a rapid traverse rate set for the individual axes independently, the tool paths are not always a straight line. Therefore, positioning must be programmed carefully so that a cutting tool will not interfere with a workpiece or fixture during positioning.

(2) Positioning in the Error Detect OFF Mode (G06)

When “G06X ••• Y ••• Z ••• (*a ••• b •••);” is specified, positioning is executed in the “error detect OFF” mode. The G06 command is valid only in the designated block.

In the G06 mode, program advances to the next block immediately after the completion of pulse distribution. The tool paths at a corner are therefore rounded.

(3) Unidirectional Positioning (G60) *

With the commands of “G60 X ••• Y ••• Z ••• (*a ••• b •••);”, movement is made to the designated position.

If positioning is made in the direction set by parameter pm4014, the axes overtravel the specified end point once and then return to be positioned at the end point specified in the G60 block. The overtravel distance is set for parameters pm4461 to pm4465 for the individual axes (X-axis to 5th-axis). The unidirectional positioning mode is provided to execute accurate positioning.

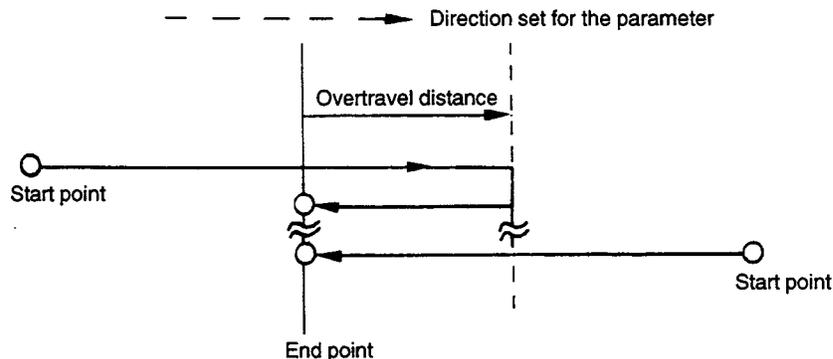


FIGURE 2.1.1.2 Unidirectional Positioning (G60)

2.1.2 Linear Interpolation (G01)

With the commands of “G01 X•••Y•••Z•••(*a•••b•••)F•••;”, linear interpolation is executed in the simultaneous 3-axis (*5-axis) control mode. The axes not designated in the G01 block do not move. For the execution of the linear interpolation, the following commands must be specified.

(1) Command Format

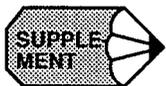
To execute the linear interpolation, the commands indicated below must be specified.

(a) Feedrate

Feedrate is designated by an F code. The axes are controlled so that vector sum (tangential velocity in reference to the tool moving direction) of feedrate of the designated axes will be the specified feedrate..

$$F \text{ (mm/min)} = \sqrt{F_x^2 + F_y^2 + F_z^2 + (F_\alpha^2 + F_\beta^2)}$$

(F_x: feedrate in the X-axis direction)



If no F code is designated in the block containing G01 or in the preceding blocks, execution of a G01 block causes alarm “0370”.

If the optional 4th- and 5th-axis are rotary axes (A-, B-, or C-axis), feedrates of basic three axes (X-, Y-, and Z-axis) and the optional 4th- and 5th-axis are determined as indicated in Table 2.1.2.1.

Table 2.1.2.1 Feedrates of Basic Three Axes and Rotary Axes (F Command)

F Function			Minimum F Command Unit	
			Feedrate of Basic Three Axes	Feedrate of Rotary Axes
Metric Output	Metric input	F60	1 mm/min	1 deg/min
	Inch input	F41	0.1 inch/min	2.54 deg/min
Inch Output	Metric input	F60	1 mm/min	0.3937 deg/min
	Inch input	F51	0.1 inch/min	1 deg/min

Note: If the 4th- and 5th-axis are linear axes, the feedrates of these axes are the same as the feedrates of basic three axes.

(2) End Point

The end point can be specified in either incremental or absolute values corresponding to the designation of G90 or G91. (For details, see 3.2.1, “Absolute/Incremental Programming”.)

Example of Programming

```
G01 X40. Y40. Z40. F100;
```

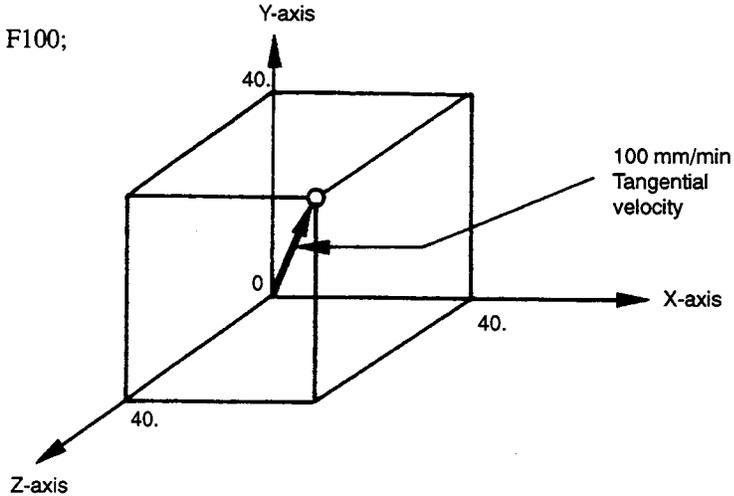


FIGURE 2.1.2.1 Linear Interpolation

2.1.3 Circular Interpolation (G02, G03)

(1) Command Format

To execute the circular interpolation, the commands indicated in Table 2.1.3.1 must be specified.

Table 2.1.3.1 Commands Necessary for Circular Interpolation

Plane Designation	G17		Circular arc in the XY plane
	G18		Circular arc in the ZX plane
	G19		Circular arc in the YZ plane
Direction of Rotation	G02		Clockwise (CW)
	G03		Counterclockwise (CCW)
Position of End Point	G90	Two axes among X, Y, and Z	End point position in a workpiece coordinate system
	G91	Two axes among X, Y, and Z	Signed distance from the start point to the end point
Distance from the Start Point to the Center	Two axes among I, J, and K		Signed distance from the start point to the center
Radius of circular arc	R		Radius of circular arc
Feedrate	F		Velocity along the circular arc

(a) Plane designation

With the commands indicated below, a cutting tool moves along the specified circular arc in the XY plane, ZX plane, or YZ plane so that the feedrate specified by the F command will be the tangential velocity of the arc.

- In the XY Plane

G17 G02 (or G03) X ••• Y ••• R ••• (or I ••• J •••) F •••;

- In the ZX Plane

G18 G02 (or G03) Z ••• X ••• R ••• (or K ••• I •••) F •••;

- In the YZ Plane

G19 G02 (or G03) Y ••• Z ••• R ••• (or J ••• K •••) F •••;

- To designate the circular interpolation mode (G02, G03), the plane of interpolation should be selected first by specifying the G17, G18, or G19. For the 4th- and 5th-axis, circular interpolation is allowed only when they are linear axes.

The G code designated to select the plane in which circular interpolation is executed also selects the plane where tool radius offset (G41/G42) is executed.

When the power is turned ON, the XY plane (G17) is automatically selected.

G17	XY plane, or $X\alpha$ or $X\beta$ plane
G18	ZX plane, or $Z\alpha$ or $Z\beta$ plane
G19	YZ plane, or $Y\alpha$ or $Y\beta$ plane

- If an optional linear 4th-axis is selected, circular interpolation is possible in the X_a , Z_a , or Y_a plane which includes the 4th-axis in addition to the XY, YZ, and ZX planes. (a = U, V, or W)
 - Circular interpolation in X_a plane

G17 G02 (or G03) X ••• a ••• R ••• (or I ••• J •••) F •••;
 - Circular interpolation in Y_a plane

G18 G02 (or G03) Z ••• a ••• R ••• (or K ••• I •••) F •••;
 - Circular interpolation in Z_a plane

G19 G02 (or G03) Y ••• a ••• R ••• (or J ••• K •••) F •••;

- If an optional linear 5th-axis is selected, circular interpolation is possible in the Xb, Zb, or Yb plane which includes the 5th-axis in addition to the XY, YZ, and ZX planes. (b = U, V, or W)
 - Circular interpolation in Xb plane
 G17 G02 (or G03) X ••• b ••• R ••• (or I ••• J •••) F •••;
 - Circular interpolation in Zb plane
 G18 G02 (or G03) Z ••• b ••• R ••• (or K ••• I •••) F •••;
 - Circular interpolation in Yab plane
 G19 G02 (or G03) Y ••• b ••• R ••• (or J ••• K •••) F •••;
- If address characters which represent the 4th- and 5th-axis are omitted as with the commands of “G17 G02 X ••• R ••• (or I ••• J •••) F •••;” the XY plane is automatically selected for the interpolation plane. Circular interpolation with the 4th- or 5th-axis is not possible if these additional axes are rotary axes.

(b) Rotation direction

The direction of arc rotation should be specified in the manner indicated in Fig. 2.1.3.1.

G02	Clockwise direction (CW)
G03	Counterclockwise direction (CCW)

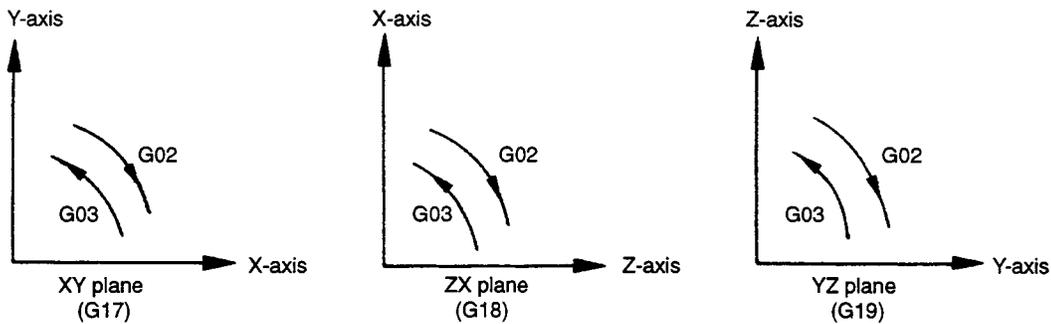


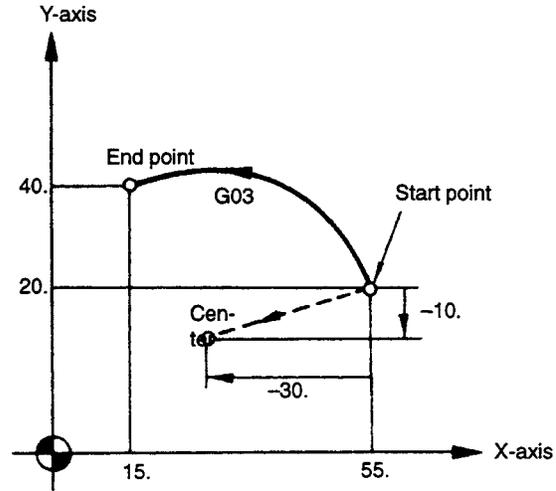
FIGURE 2.1.3.1 Rotation Direction of Circular Arc

(c) End point

The end point can be specified in either incremental or absolute values corresponding to the designation of G90 or G91.

Example of Programming

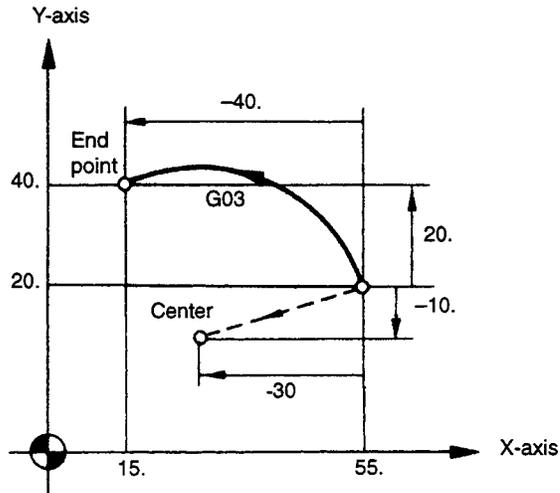
G17 G90 G03 X15. Y40. I-30. J-10. F150;



(a) Absolute programming (G90)

Example of Programming

G17 G91 G03 X-40. Y20. I-30. J-10. F150;



(b) Incremental programming (G91)

FIGURE 2.1.3.2 End Point of Circular Arc

If the specified end point is not on the specified arc, the arc radius is gradually changed from the start point to the end point to generate a spiral so that the end point lies on the specified arc.

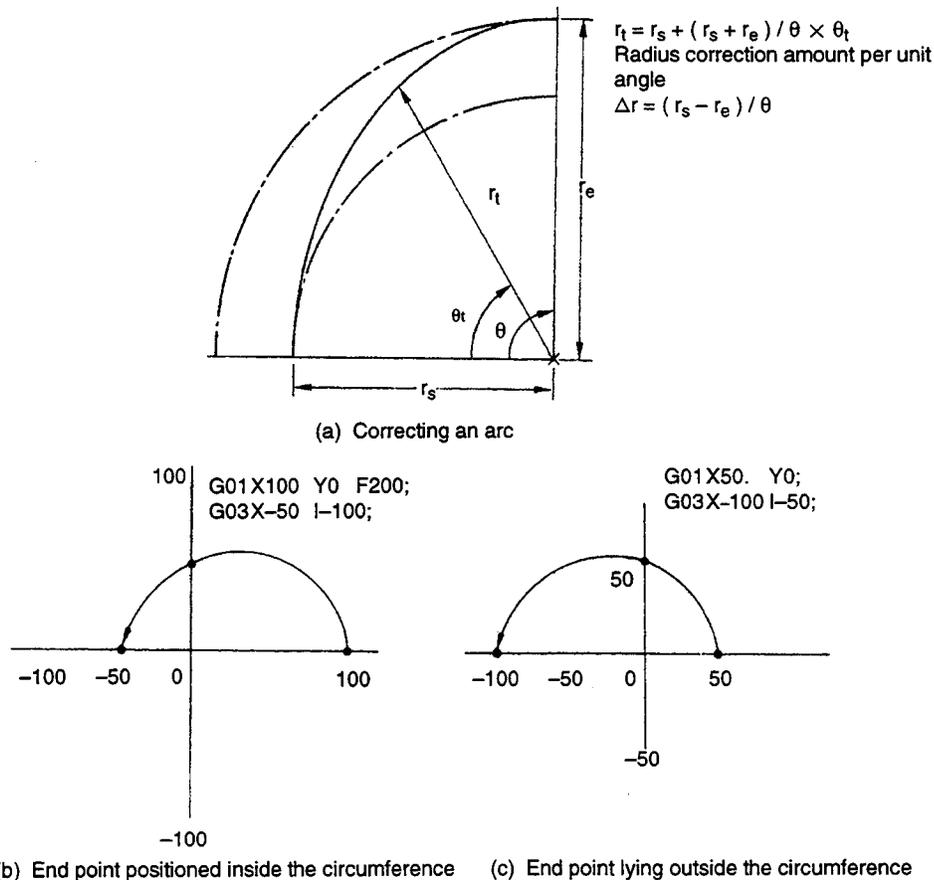


FIGURE 2.1.3.3 Interpolation with End Point off the Specified Arc

(d) Center of arc

The center of arc can be specified in two methods – designation of the distance from the start point to the center of the arc and designation of the radius of the arc.

- Specifying the distance from the start point to the center

Independent of the designated dimensioning mode (G90 or G91), the center of an arc must be specified in incremental values referenced from the start point.

- Specifying the radius

When defining an arc, it is possible to specify the radius by using address R instead of specifying the center of the arc by addresses I, J, or K. This is called “circular interpolation with R designation” mode.

- For the circular arc with the central angle of 180 deg. or smaller, use an R

value of “ $R > 0$ ”.

- For the circular arc with the central angle of 180 deg. or larger, use an R value of “ $R < 0$ ”.

Example of Programming

```
G17 G02 X ··· Y ··· R ± ··· F ···;
```

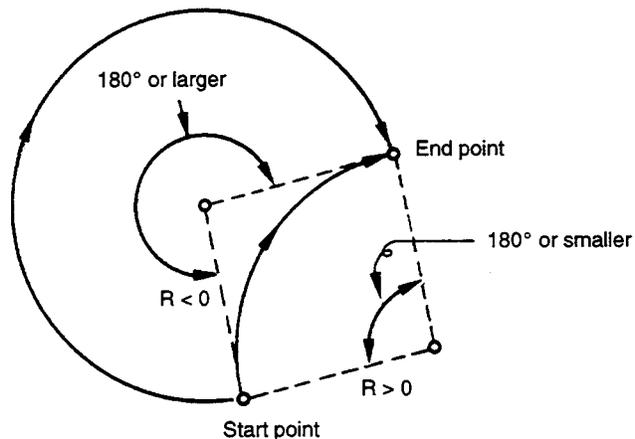


FIGURE 2.1.3.4 Circular Interpolation with Radius R Designation

(e) Feedrate

In the circular interpolation mode, the feedrate can be specified in the same manner as in the linear interpolation mode. Refer to 2.1.2 “**Linear Interpolation (G01)**”.

(2) Supplements to Circular Interpolation

- A circular arc extending to multiple quadrants can be defined by the commands in a single block. It is also possible to specify a full circle.

Example of Programming

```
G00 X0 Y0;  
G02 X0 Y0 I10 J0 F100;
```

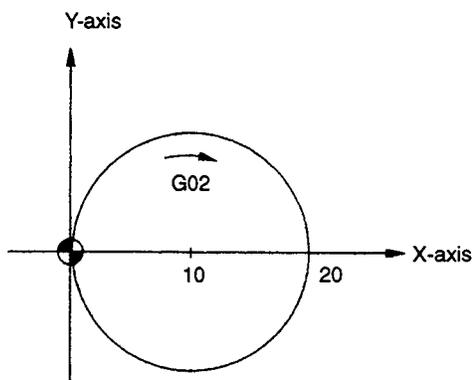


FIGURE 2.1.3.5 Full Circle

- With the commands of “G17 G02 (or G03) I ••• J ••• F ••• Ln;”, full-circle interpolation is repeated by n times. If address L is omitted, interpolation is executed once. Execution of the commands with the single-block function ON causes full-circle interpolation to be interrupted after the execution of one full-circle interpolation.
- In circular interpolation, if the distance between the start and end points is smaller than the value set for parameter pm4450, these two points are connected by a straight line.

2.1.4 Helical Interpolation (G02, G03)*

It is possible to execute linear interpolation in synchronization with circular interpolation with the axis which is not included in the circular interpolation plane. This is called helical interpolation. The command format is indicated below.

- In the XY plane
G17 G02 (or G03) X •• Y •• R •• (or I •• J ••) Z (a, b) •• F ••;
- In the ZX plane
G18 G02 (or G03) Z •• X •• R •• (or K •• I ••) Y (a, b) •• F ••;
- In the YZ plane
G19 G02 (or G03) Y •• Z •• R •• (or J •• K ••) X (a, b) •• F ••;
- In the Xa plane
G17 G02 (or G03) X •• a •• R •• (or I •• J ••) Z (b) •• F ••;
- In the Za plane
G18 G02 (or G03) Z •• a •• R •• (or K •• I ••) Y (b) •• F ••;
- In the Ya plane
G19 G02 (or G03) Y •• a •• R •• (or J •• K ••) X (b) •• F ••;
- In the Xb plane
G17 G02 (or G03) X •• b •• R •• (or I •• J ••) Z (a) •• F ••;
- In the Zb plane
G18 G02 (or G03) Z •• b •• R •• (or K •• I ••) Y (a) •• F ••;
- In the Yb plane
G19 G02 (or G03) Y •• b •• R •• (or J •• K ••) X (a) •• F ••;

Where, a and b are the linear 4th and 5th axes respectively, each representing any of U-, V-, and W-axis. If no 4th- or 5th-axis is specified as the end point command of the arc, any of the command format is selected among the commands in the XY plane, ZX plane, and YZ plane.

Example of Programming

```
G17 G03 X0 Y100. R100 Z90. F10;
```

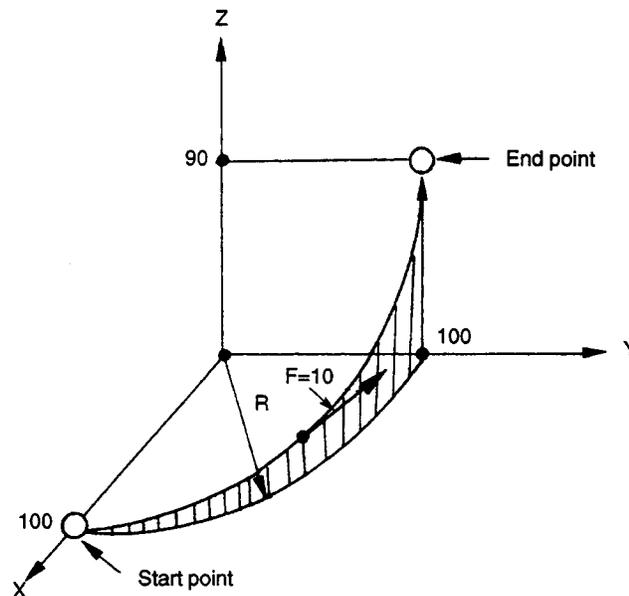


FIGURE 2.1.4.1 Helical Interpolation



1. An arc must be programmed within 360° range.
2. The feedrate specified with an F command indicates the tangential velocity in the three dimensional space constituted by the circular interpolation plane and the linear axis perpendicular to the interpolation plane.
3. Tool radius offset C* is valid for the axes included in the circular interpolation plane.

2.2 REFERENCE POINT RETURN

2.2.1 Automatic Return to Reference Point (G28)

With the commands of “G28 X ··· Y ··· Z ··· (*a ··· b ···);”, the numerically controlled axes are returned to the reference point. The axes are first moved to the specified position at a rapid traverse rate and then to the reference point automatically. This reference point return operation is possible in up to simultaneous 3-axis (* 5-axis) control. The axes not designated in the G28 block are not returned to the reference point.

Example of Programming:

```
(G90/G91) G28 X ··· Y ··· Z ···;
```

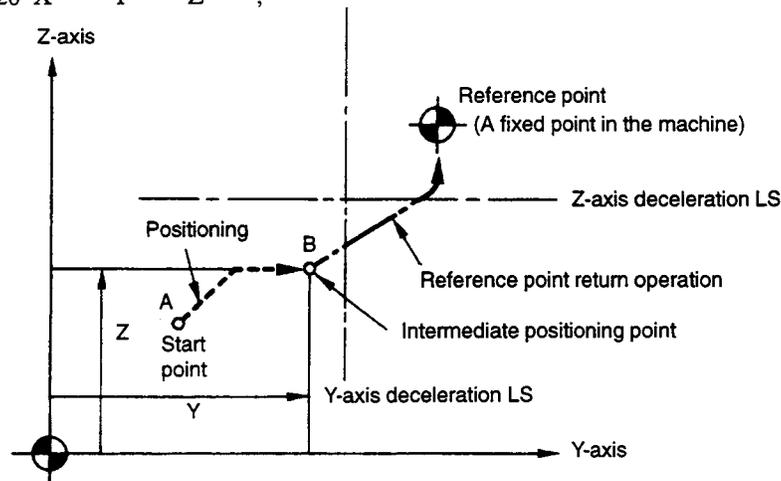


FIGURE 2.2.1.1 Automatic Reference Point Return

(1) Reference Point Return Operation

Reference point return operation is the series of operations in which the axes return to the reference point after the reference point return operation has been started manually.

Reference point return is accomplished in two ways:

(a) Low-speed reference point return

In low-speed reference point return operation, a deceleration limit switch is used. In high-speed reference point return operation, the first return operation is executed in the low-speed type using a deceleration limit switch; the reference point data are stored after the completion of the first reference point return and in subsequent reference point return operations is executed without using a deceleration limit switch.

(b) High-speed reference point return

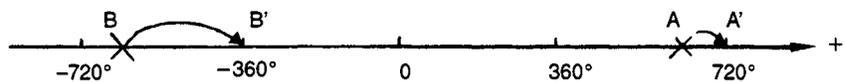
See parameter pm4003 D6 and D7.

It is possible to use the “high-speed reference point return” in place of the “automatic reference point return”. In this case, the reference point return is executed in the following manner.

- After the positioning at the intermediate positioning point B, the axes return directly to the reference point at a rapid traverse rate. The axes can be returned to the reference point in a shorter time compared to the normal reference point return operation that uses a deceleration limit switch for the individual axes.
- Even if point B is located outside the area in which reference point return is allowed, the high-speed reference point return specification allows the axes to return to the reference point.
- High-speed reference point return is enabled only for the axes for which normal reference point return has been completed either manually (manual reference point return) or by executing the G28 command after turning ON the power.
- If an axis for which low-speed reference point return has not been completed either manually or by executing the G28 command after power-ON is included in the G28 block, low-speed reference point return is executed for all axes designated in the G28 block.
- High-speed automatic reference point return is valid only when reference point return is called by G28, and it does not influence manual reference point return operation.

(2) Automatic Reference Point Return for Rotary Axes

With a rotary axis, it is possible to execute the automatic reference point return the same as with a linear axis. With a rotary axis, if it has been moved by more than $\pm 360.000^\circ$ from the reference point established first, reference point return is executed to the closest reference point in the preset direction of reference point return. The illustration below shows how the reference point return is executed from points A and B. (The reference point return direction is determined by the setting for pm4002 D3 and D4.)



(Reference point return: Negative direction is selected for the reference point return direction)

(3) Supplements to the Automatic Reference Point Return Commands

(a) Tool radius offset and canned cycle

If G28 is specified in the tool radius offset mode (G41, G42) or in a canned cycle, alarm “0170” occurs. G28 must not be specified in the tool radius offset mode (G41, G42) or in a canned cycle.

(b) Tool position offset

If G28 is specified in the tool position offset mode, positioning at the intermediate positioning point is made with the offset data valid. However, for the positioning at the reference point, the offset data are invalid and positioning is made at the absolute reference point. Whether or not the tool length offset function is disregarded after the positioning at the reference point can be determined by the setting for the parameters as indicated below.

Table 2.2.1.1 G28 Command in the Tool Position Offset Mode

Parameter pm4011	Operation
D 1 = 0	Pm4010 D6 = 0: As programmed D6 = 1: Offset valid
D 1 = 1	Pm4010 D7 = 0: As programmed D7 = 1: Offset valid

(c) Tool length offset

It is possible to cancel the tool length offset mode by G28 by changing the setting for a parameter. If the setting is so made to cancel the tool length offset mode by the execution of G28, it is valid only when a Z-axis command is specified with G28 in the same block. Although cancellation of the tool length offset mode is possible by G28, the tool length offset mode should be canceled before the designation of G28.

Table 2.2.1.2 Canceling Tool Length Offset Mode

Parameter pm4010 D7	Operation
0	Tool length offset mode is canceled when the NC is reset or at the execution of the reference point return. The H code is cleared to "0". In this case, the tool length offset G code is retained.
1	When the NC is reset or at the execution of the reference point return, the tool length offset mode is not canceled. Both the H code and the tool length offset G code are retained.

(d) Mirror image

If G28 is specified while the mirror image mode (M95) is called up, reference point return is executed in the manner indicated in Table 2.2.1.3.

Table 2.2.1.3 G28 Command in the Mirror Image Mode

Parameter pm4001 D7	Operation
D2 = 0	Mirror image is applied to the intermediate positioning point. Movement to the reference point is not influenced by the mirror image function.
D2 = 1	Alarm "0127" occurs.

(e) Machine lock intervention

Concerning machine lock intervention, there are two types of operation: turning ON the machine lock after suspending axis movement by using the feed hold function, and turning OFF the machine lock after suspending axis movement again by using the feed hold function. Table 2.2.1.4 shows how the machine operates according to the machine lock intervention.

Table 2.2.1.4 Machine Operation according to Machine Lock Intervention

		Machine Lock Intervention during Positioning to Intermediate Positioning Point	Machine Lock Intervention during Positioning to Reference Point
Machine Lock OFF → ON	Low speed type	Although positioning is continued to the intermediate positioning point (position data display only), movement to the reference point is not executed. Display data are not updated, either.	Display data are infinitely updated. Although positioning is made at the reference point after the detection of the actuation of the deceleration limit switch, this cannot be detected due to machine lock and, therefore, the display data are infinitely updated.
	High speed type		In response to the machine lock intervention, the axes stops moving. After that, the display data (position data in the workpiece coordinate system) are updated until the reference point return is completed. (without axis movement)
Machine Lock OFF → ON → OFF	Low speed type	The axes move in the workpiece coordinate system up to the intermediate positioning point. After that, positioning is executed at the reference point in the machine coordinate system. Accordingly, the reference point given in the workpiece coordinate system is offset. However, the reference point in the machine coordinate system is not offset.	The axes move to the reference point.
	High speed type		Actual axis position is displayed due to the intervention of machine lock. Accordingly, although the display data (position data in the workpiece coordinate system) agree with the reference point, the axes are not located at the reference point.

(4) Supplements to the Automatic Reference Point Return Commands

- It is possible to select valid/invalid of reference point return for each axis. If the axis for which “reference point return invalid” has been set is specified in the G28 block, alarm “0241” occurs. Refer to parameter pm4002 D0 to D4.
- It is possible to display alarm “0411” (X-axis) to “0415” (5th-axis) if the cycle start switch is pressed without carrying out reference point return for all axes after turning ON the power. Whether or not such alarm display should be given is determined by the setting for parameter pm4001 D6. For the axes for which the setting is “D6 = 1 (to generate alarm)”, setting is possible whether or not the reference point return is necessary for a parameter. Refer to parameter pm4018 D0 to D4 (X-axis to 5th-axis).
- “It is possible to display alarm “0411” (X-axis) to “0415” (5th-axis) when an axis move command other than G28 is executed without completing reference point return after turning ON the power. Whether or not such alarm display should be given is determined by the setting for a parameter. Refer to parameter pm4004 D0 to D4 (X-axis to 5th-axis).
- “The absolute coordinate values of the axes specified in the G28 block are saved to memory as the intermediate positioning point. For the axes not specified in the G28 block, the intermediate positioning point saved in the previous reference point return operation remains valid.
- If M, S, T, and/or B command is specified with G28 in the same block, the axes continue moving to the reference point disregarding whether or not the FIN processing is completed before the positioning of an axis at the intermediate positioning point. Therefore, DEN is output at the reference point.
- The deceleration limit switch position must be carefully attended to when executing the reference point return for the first time after turning ON the power. For details, refer to 2.4.2, “Manual Reference Point Return” of the Operating Manual.

2.2.2 Reference Point Return Check (G27)

This function checks whether the axes are correctly returned to the reference point at the completion of the part program which is created so that the program starts and ends at the reference point in the machine by specifying the commands of “G27 X ••• Y ••• Z ••• (*a ••• b •••);”.

In the G27 mode, the function checks whether or not the axes positioned by the execution of these commands in the simultaneous 3-axis (* 5-axis) control mode are located at the reference point. For the axes not specified in this block, and not moved although the axis command specified, positioning and check are not executed.

(1) Operation after the Check

When the position reached after the execution of the commands in the G27 block agrees with the reference point, the reference point return complete lamp lights. The automatic operation is continuously executed when all of the specified axes are positioned at the reference point. If there is an axis that has not been returned to the reference point, reference point return check error (alarm “0421” (X-axis) to “0425” (5th-axis)) occurs and the automatic operation

is interrupted. In this case, the cycle start lamp goes OFF.

(2) Supplements to the Reference Point Return Check Command and Other Operations

- If G27 is specified in the tool offset mode, positioning is made at the position displaced by the offset amount and the positioning point does not agree with the reference point. It is necessary to cancel the tool offset mode before specifying G27. Note that the tool position offset and tool length offset functions are not canceled by the G27 command.
- Check is not made if G27 is executed while the machine lock state is valid even for one axis. For example, if an X-axis movement command is specified in the G27 block while in the Z-axis neglect state, X-axis position is not checked.
- The mirror image function is valid to the direction of axis movement in the reference point return operation called by G27. To avoid a position unmatched error, the mirror image function should be canceled by specifying M94 before executing G27.

2.2.3 Return from Reference Point Return (G29)

The commands of “G29 Y ··· Z ···;” the axes, having been returned to the reference point by the execution of the automatic reference point return function (G28, G30), to the intermediate positioning point by back tracing the paths along which the reference point return has been executed.

Example of Programming

G28 Y ··· Z ···; Point A → B → C (Reference point)

·
·
·
Point B

G29 Y ··· Z ···; Point C → B → D

·
·
·
Point D

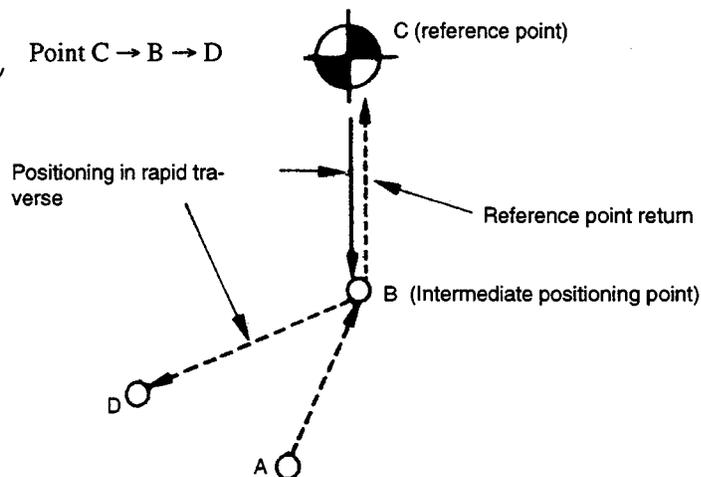


FIGURE 2.2.3.1 Return from Reference Point

(1) Intermediate Positioning Point

- It is not possible to specify the intermediate positioning point in the G29 block. The axes return to the previous point at a rapid traverse rate along the paths taken in the return to the reference point. Note that the axes not specified in the G29 block do not move.
- If G28 or G30 (see 2.2.4, “Second to Fourth Reference Point Return (G30)”) has been executed several times before the execution of G29, point B to be set for the execution of B29 is established at the intermediate positioning point set in the last G28 or G30 operation. The following program written in absolute commands explains how point B is set for the return operation from the reference point.

Coordinate values of intermediate positioning point

N21	G90;			X	Y	Z
N22	G28	Z10.	Y20.		(0, 20., 10.)	
N23	G28	X30. ;			(30., 20., 10.)	
N24	G29	<u>X-40.</u>	<u>Y-50. ;</u>			

↑ End point

Operation called by the commands in N24 block

G00	<u>X30. X20. ;</u>				Intermediate positioning point
G00	<u>X-40. Y-50. ;</u>				End point

Example of Programming

```

N31 G91;
N32 G28 Z...;
N33 G28 X20. Y40.;
N34 M06;
N35 G29 X40. Y-40.;

```

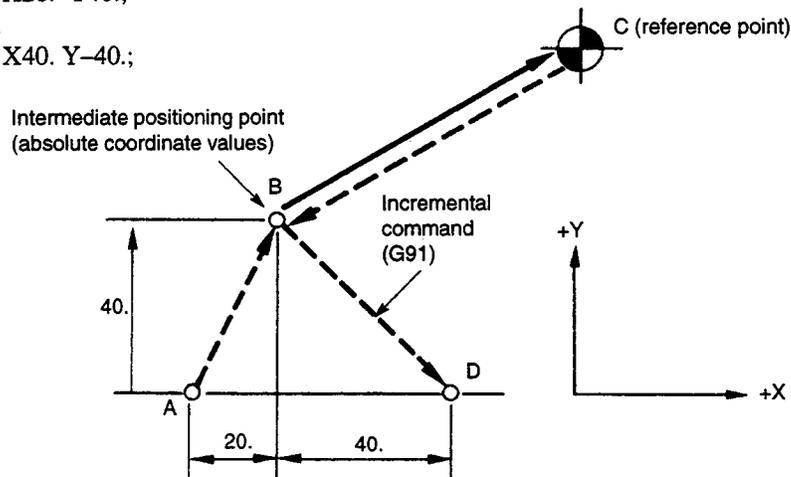


FIGURE 2.2.3.2 Coordinate Values of Point B for G29 Operation

- In the following cases, the intermediate positioning point used for the execution of G29 does not agree with the intermediate positioning point specified for the execution of G28 or G30. Therefore, do not specify such commands or attempt such operation.
 - Execution of the following before the execution of G29 after the completion of G28:
 - Coordinate system setting (G92 or coordinate system setting operation in POS. job)
 - Intervention of machine lock
 - Intervention of manual operation with manual absolute OFF
 - Execution of G28, or G30 or G29 in a block specified after the cancellation of the mirror image (M94) at a position different from the position where the mirror image was started.
 - Execution of G28, or M30 or M29 after the intervention of manual operation with the manual absolute OFF.
- If G29 is executed in the G91 incremental mode, it is valid for the commands in the G29 block, in which the distance from the intermediate positioning point is specified, and the position data of the intermediate positioning point are not influenced by this mode.

(2) Supplements to the Return Command from the Reference Point Return

(a) Automatic reference point return

If G29 is specified without the execution of G28 or G30 after turning ON the power, alarm “0240” occurs.

(b) Tool radius offset and canned cycle

If G29 is specified in the tool radius offset mode (G41, G42) or in a canned cycle (G73, G74, G76, G77, G81 to G89), alarm “0170” or “0182” occurs.

(c) Tool position offset

It is necessary to cancel the tool position offset function before specifying G28, G30, or G29. If these G codes are executed in the offset mode, the intermediate positioning point B' is also offset, causing the tool to move to point B. Note that the tool position offset and tool length offset functions are not canceled by G29.

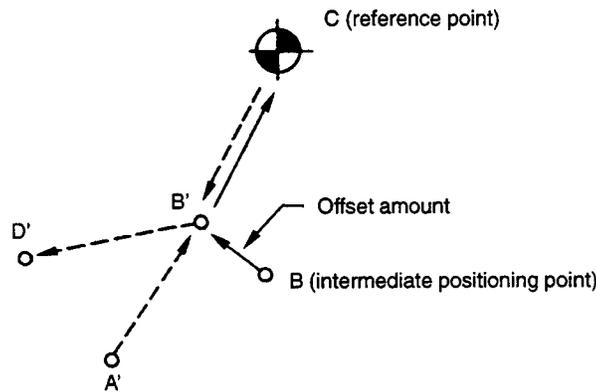


FIGURE 2.2.3.3 G29 Operation Executed in the Tool Position Offset Mode

(d) Mirror image

Table 2.2.3.1 indicates how G29 is executed if it is specified in the mirror image (M95) mode.

Table 2.2.3.1 G29 Operation Executed in Mirror Image Mode

Parameter pm4011 D2	Operation
0	Mirror image is applied to the intermediate positioning point and the command values specified in the G29 block.
1	Alarm “0127” occurs.

2.2.4 Second to Fourth Reference Point Return (G30)

With the commands of “G30 Pn X ••• Y ••• Z (*a ••• b •••);”, the axes are moved to P2 (second reference point), P3 (third reference point*), or P4 (fourth reference point*) in the simultaneous 3-axis (* 5-axis) control mode after the positioning at the specified intermediate positioning point. If “G30 P3 X30. Y50.;;” “N specified, the X- and Y-axis return to the third reference point. If “Pn” is omitted, the second reference point is selected. The axes not specified in the G30 block do not move.

(1) Reference Point Positions

The position of each reference point is determined in reference to the first reference point. The distance from the first reference point to each of the reference points is set for the following parameters.

Table 2.2.4.1 Reference Point Positions

	X-axis	Y-axis	Z-axis	4th-axis	5th-axis
2nd reference point	pm	pm	pm	pm	pm
3rd reference point	pm	pm	prn	pm	pm
4th reference point	pm	pm	pm	pm	pm

(2) Supplements to the 2nd to 4th Reference Point Return Commands

- For the points to be considered to for the execution of G30, refer to the supplements in 2.2.1, “Automatic Return to Reference Point (G28)”.
- If G29 is specified after G30, positioning is made at the point specified with G29 after passing the intermediate positioning point specified with G30. Only the coordinate value of intermediate positioning point of the axis specified with G30 is updated.
- For the execution of G30, reference point return must have been completed after power-ON either manually or by the execution of G28. If an axis for which reference point return has not been completed is included in the axes specified in the G30 block, alarm “0240” occurs.

3

Movement Control Commands

Chapter 3 describes the procedure used for setting and selecting the coordinate system and the programming for controlling the movement of a cutting tool.

3.1	SETTING THE COORDINATE SYSTEM	3-3
3.1.1	Selecting the Coordinate System	3-3
3.1.2	Base Coordinate System (G92)	3-3
3.1.3	Workpiece Coordinate System (G54 to G59)*	3-5
3.1.4	Local Coordinate System (G52 Q2)*	3-12
3.1.5	Machine Coordinate System (G53)	3-13
3.1.6	Rotation of Coordinate System (G68, G69)*	3-16
3.1.7	Plane Selection (G17, G18, G19)	3-18
3.2	DETERMINING THE COORDINATE VALUE INPUT MODES	3-19
3.2.1	Absolute/Incremental Designation (G90, G91)	3-19
3.2.2	Inch/Metric Input Designation (G20, G21)	3-21
3.2.3	Scaling (G50, G51) *	3-22
3.3	TIME-CONTROLLING COMMANDS	3-26
3.3.1	Dwell (G04)	3-26
3.3.2	Exact Stop (G09)	3-26
3.3.3	Exact Stop Mode (G61, G64)	3-26
3.4	TOOL OFFSET FUNCTIONS	3-28
3.4.1	Tool Offset Data Memory	3-28
3.4.2	Tool Length Offset (G43, G44, G49)	3-29
3.4.3	Tool Position Offset (G45 to G48)	3-32
3.4.4	Radius Offset C Function (G40, G41, G42) *	3-40

3.4.5	3-Dimensional Tool Offset Function (G40, G41, G42)*	3-78
3.5	S, T, M, AND B Functions	3-85
3.5.1	Spindle Function (S Function)	3-85
3.5.2	Tool Function (T Function)	3-86
3.5.3	Miscellaneous Function (M Function)	3-87
3.5.4	Second Miscellaneous Function (B Function) *	3-89

3.1 SETTING THE COORDINATE SYSTEM

3.1.1 Selecting the Coordinate System

There are four types of coordinate systems such as the base coordinate system, workpiece coordinate system*, local coordinate system*, and machine coordinate system, and it is possible to select the plane to be used in each of the coordinate systems.

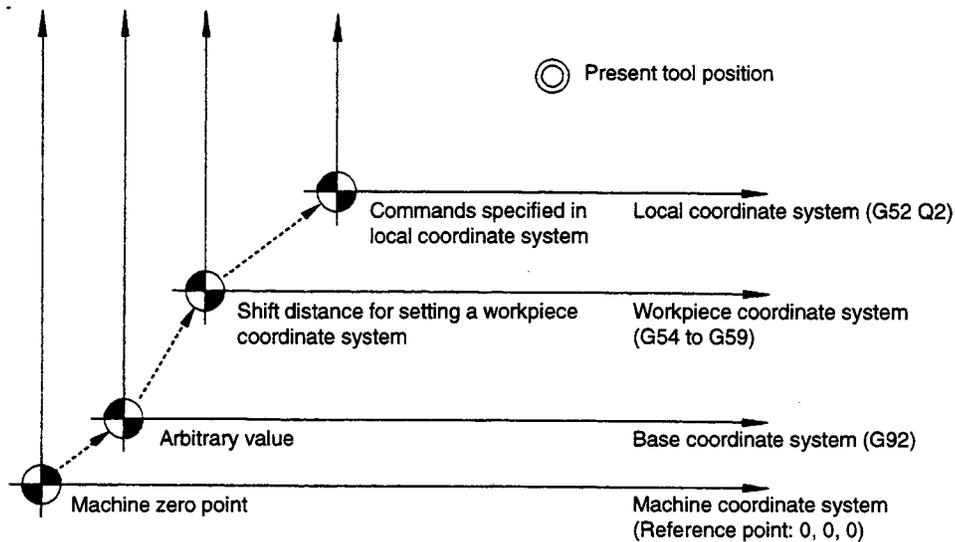


FIGURE 3.1.1.1: Coordinate Systems

3.1.2 Base Coordinate System (G92)

Before programming axis movement, a coordinate system must be set. When a coordinate system is set, a single absolute coordinate system is determined and absolute move commands specified after the setting of a coordinate system are all executed in it. The G92 command sets the position of the origin of a coordinate system used for programming.

(1) Commands

(a) Features of G92

G92 is a non-modal G code that is valid only in the specified block. In the G92 block, no other G codes can be specified.

(b) Command Format

With the commands of “G92 X ••• Y ••• Z ••• (*a ••• b •••):”, a coordinate system is set so that the present tool position has the absolute coordinate values specified in the G92 block (X, Y, Z, a *, b *). In other words, the addresses in the G92 block specify the distance from the point that should be set as the origin (0, 0, 0) of the coordinate system used for programming to the present tool position. Axis movement commands can be specified for up to three axes (* five axes) simultaneously. Note that the axes not specified the G92 block do not move.

Example of Programming

```
G92 X500. Y300. Z400.;
```

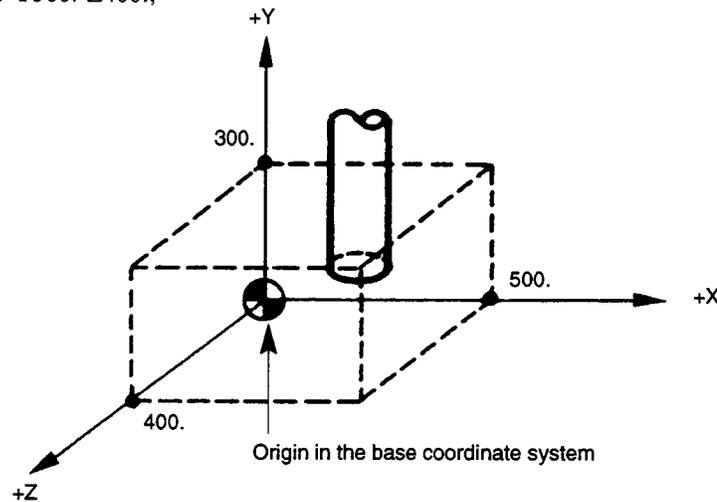


FIGURE 3.1.2.1: Setting the Base Coordinate System

(2) Supplements to the Base Coordinate System Commands

- G92 should be specified in the state where tool offset has been canceled.
- With the system equipped with an incremental encoder, coordinate system is established when the power is turned ON so that the present tool position will be (0, 0, 0). Therefore, it is necessary to set the required coordinate system before starting operation.
- Once set, the coordinate system is not influenced by the reset operation. To reset a coordinates system, perform any of the following operations.
 - Setting “0” on the UNIVERSAL (COORD-SET) screen
 - Setting “0” for the coordinate system in the MDI mode

```
G92 X0 Y0 Z0 (*a0 b0);
```

3.1.3 Workpiece Coordinate System (G54 to G59)*

It is possible to set a workpiece coordinate system by shifting the base coordinate system. When a G code (G54 to G59) is specified, shift distance corresponding to the specified G code is called and the workpiece coordinate system is set by shifting the base coordinate system by the called shift distance. Shift distance is set on the WORK SHIRT SET screen corresponding to G54 to G59. It is also possible to rotate a coordinate system around the origin (0, 0, 0) by the set angle.

Table 3.1.3.1 G Codes Used to Call Workpiece Coordinate Systems

G code	Description	Group
G54	Shift to workpiece coordinate system 1	12
G55	Shift to workpiece coordinate system 2	12
G56	Shift to workpiece coordinate system 3	12
G57	Shift to workpiece coordinate system 4	12
G58	Shift to workpiece coordinate system 5	12
G59	Shift to workpiece coordinate system 6	12

(1) Number of Workpiece Coordinate Systems

The number of workpiece coordinate system sets can be increased optionally.

Table 3.1.3.2 Number of Workpiece Coordinate System Sets

Option 1	6 sets
Option 2	54 sets
Option 3	162 sets

(2) Command Format

Table 3.1.3.3 Setting Workpiece Coordinate System

Option 1	G54 J1 (P1)	G55 J1 (P1)	G56 J1 (P1)	G57 J1 (P1)	G58 J1 (P1)	G59 J1 (P1)
Option 2	G54 J1 (P1)	G55 J1 (P1)	G56 J1 (P1)	G57 J1 (P1)	G58 J1 (P1)	G59 J1 (P1)
	-	-	-	-	-	-
Option 3	G54 J9 (P9)	G55 J9 (P9)	G56 J9 (P9)	G57 J9 (P9)	G58 J9 (P9)	G59 J9 (P9)
	G54 J1 (P1)	G55 J1 (P1)	G56 J1 (P1)	G57 J1 (P1)	G58 J1 (P1)	G59 J1 (P1)
	-	-	-	-	-	-
	G54 J27 (P27)	G55 J27 (P7)	G56 J27 (P7)	G57 J27 (P7)	G58 J27 (P7)	G59 J27 (P7)

Note: An alarm occurs if an improper value is specified for address J.

Option 1 J2 or larger

Option 2 J10 or larger

Option 3 J28 or larger

- By specifying “G54 (G55, G56, G57, G58, or G59) J1 (J2 ••••••• J27)”, the subsequent programs are executed in the selected workpiece coordinate system. Address P can be used instead of address J, by changing the setting for the following parameter. G54”, “G54 J1”, and “G54 J0” have the same meaning.

pm4012 D7 = 0	Address J
pm4012 D7 = 1	Address P

- By specifying “G53 (or G90) G00 X ••• Y ••• Z ••• (*a ••• b •••);”, it is possible to specify temporary shift of a coordinate system in the machine coordinate system.
- By specifying “G52;” the workpiece coordinate system selected currently is canceled and the base coordinate system is called.
- If workpiece coordinate system shift distance is changed after the workpiece coordinate system is selected by the execution of G54 to G59, whether or not the new shift distance is made valid without specifying G54 to G59 again can be selected by the setting for parameter (pm4021 D0).

(3) Rotation of Workpiece Coordinate System

By entering the address calling rotation of a workpiece coordinate system in the G54 to G59 block, rotation of the workpiece coordinate system is executed at the same time when the workpiece coordinate system is called.

Example of Programming

G92 X0 Y0 Z0;

.? The base coordinate system is set.

.
.

G54; (G68 X0 Y0 R •••)

.? A workpiece coordinate system is set by G54. At the same time, the workpiece coordinate system is rotated around (0, 0) it by the angle specified following address R.

G55; (G69)

.(G68 X0 Y0 R •••)

.? The workpiece coordinate system set by G54 is canceled and a new workpiece coordinate system is set according to the shift distance and rotation angle specified in the G55 block.

G52; (G69)

.? G52 returns the coordinate system to the one set by G92. Rotation of a coordinate system is canceled at the same time.

M30;

Note: The commands in () indicate the program which uses the coordinate rotation function.

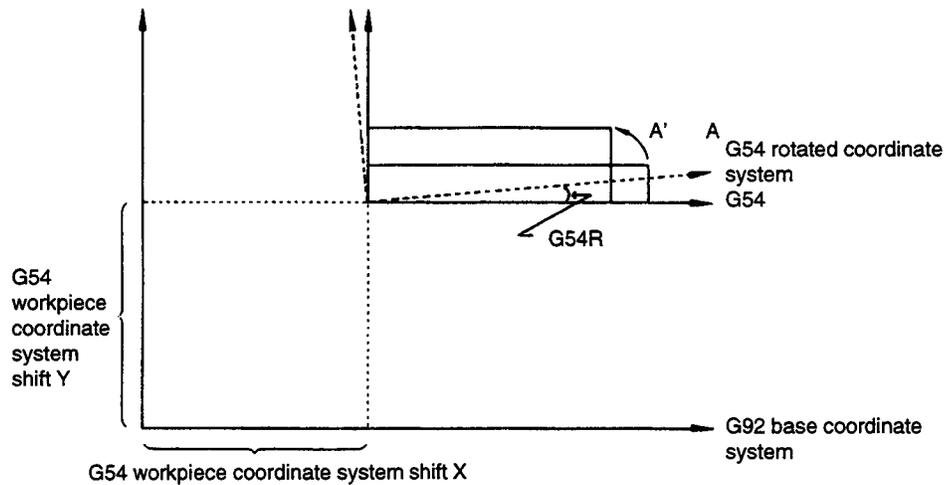


FIGURE 3.1.3.1: Rotation of Workpiece Coordinate System

- If coordinate rotation is executed when calling a workpiece coordinate system by G54, rotation is also applied to the intermediate positioning point when G28 or G30 is executed. However, this is not applied to the reference point.
- The coordinate rotation function is valid only in the G17 plane.
- To cancel the coordinate rotation executed by G54, either specify G52 or reset the shift distance of the called workpiece coordinate system to “0”, then specify G54 again.
- When the coordinate rotation is executed, the subsequent program paths will be varied depending on whether or not the intervention of reset is valid. This difference is caused by the following two specifications.
 - By the reset operation, the present position is taken into the NC as the command values to be re-set.
 - The coordinate: rotation function generates axis move distance by applying rotation calculation to the NC commands in reference to the center of rotation.



If an address different from the one selected by the setting for parameter pm4012 D7 (J when the selection is P, or P when the selection is J) is specified with G54, such an address is disregarded for the execution of G54.

(Example) pm4012 D7 = 1 (Selection of P) (for option 2 or 3)

G54 J2 X0 Y0; ? Executed as “G54 P1 X0 Y0;

(4) Workpiece Coordinate System Shift Distance

Fig. 3.1.3.2 shows how positioning is executed by setting a workpiece coordinate system.

Example of Programming

```

N1    G90    X100.  Y200.;
N2    G54;
N3    X100.  Y300.;
N4    X300.  Y200.;
N5    G52;
N6    X0.    Y0.;

```

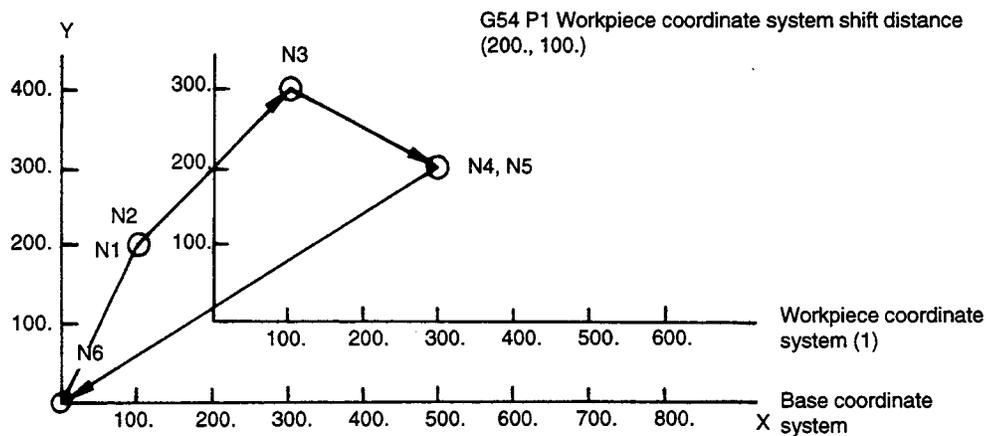


FIGURE 3.1.3.2 Workpiece Coordinate System Shift Distance

The shift distance of a workpiece coordinate system can be corrected by the following three methods.

- To use a microprogram
- To use external data input
- To use G10 command

(a) Correcting workpiece coordinate system shift distance by microprogram

The workpiece coordinate system shift distance (and the external workpiece coordinate system correction amount) can be read by entering a system variable in the right side of the operation expression. It is possible to change these values by entering a system variable in the left side of the operation expression.

Example of programming

- #116 = #2501;
G54 (P1) X-axis workpiece coordinate shift distance is entered to common variable #116.
- #2511 = #4;
G54 (J2) X-axis workpiece coordinate system shift distance is cleared and the contents of local variable #4 are set. For the correspondence between the system variables and the workpiece coordinate system shift distances, refer to Tables 4.41, 4.42, and 4.43.

(b) Correcting workpiece coordinate system shift distance by external data input

The workpiece coordinate system shift distance can be corrected by the external data input signal.

The shift correction amount is added to the present shift distances for G54 (J1) to G59 (J27) and the sum of externally input shift correction amount and the present shift distance is used as a new shift distance. In this case, the workpiece coordinate shift distance is not directly changed but the external workpiece coordinate system shift distance is changed.

Workpiece coordinate system shift distance in actual operation

= External workpiece coordinate system correction amount
+ Workpiece coordinate system shift distance

Concerning the angle of rotation, the workpiece coordinate system shift distance should be corrected directly.

(c) Correcting the workpiece coordinate system shift distance by programming (G10)

It is possible to correct the workpiece coordinate system shift distance by using the G10.

By specifying "G10 Q2 (or L2) Pm Jn X... Y... Z... (*a... b...);", the workpiece coordinate system shift distance of the specified workpiece coordinate system is corrected. A combination of Pm and Jn specifies the workpiece coordinate system for which the shift distance should be corrected.

- The correspondence between a P number and G code is:
P1 = G54, P2 = G55, etc.
- Jn: Selects J1 - J27.
(Example) If "P2 J3" is specified, it is equal to "G55 J3".
- If J is omitted or "J0" is specified, it is assumed that "J1" is specified.

- If an incorrect value is specified for “m” or “n”, alarm “0202” occurs.

(5) Supplements to Commands Used for Setting the Workpiece Coordinate System

- When the power is turned ON, the position where the tool is presently positioned is set as (0, 0, 0). For the system not equipped with absolute position encoders, the present tool position is temporarily set as the origin of the coordinate system until an appropriate coordinate system is set.
- In the state where G92 (base coordinate system setting) is not specified, a workpiece coordinate system is established in reference to the origin of the machine coordinate system.
- In the state where G92 (base coordinate system setting) has been specified, a workpiece coordinate system is established in reference to the origin of the base coordinate system.
- If G92 is specified in the state where a workpiece coordinate system has been set, the base coordinate system is set so that the present tool position takes the coordinate values specified in the G92 block. At the same time, the workpiece coordinate system is defined in reference to the origin of the newly set base coordinate system.
- If G54 to G59 is executed in the tool length offset or tool position offset mode, present offset is not canceled. Tool length offset or tool position offset should be canceled before specifying G54 to G59.

Example of Programming

G43 Z0 H01; ? Position in workpiece coordinate system Z 100.

G54;

G90 Z1000. ? Position in workpiece coordinate system Z1100.

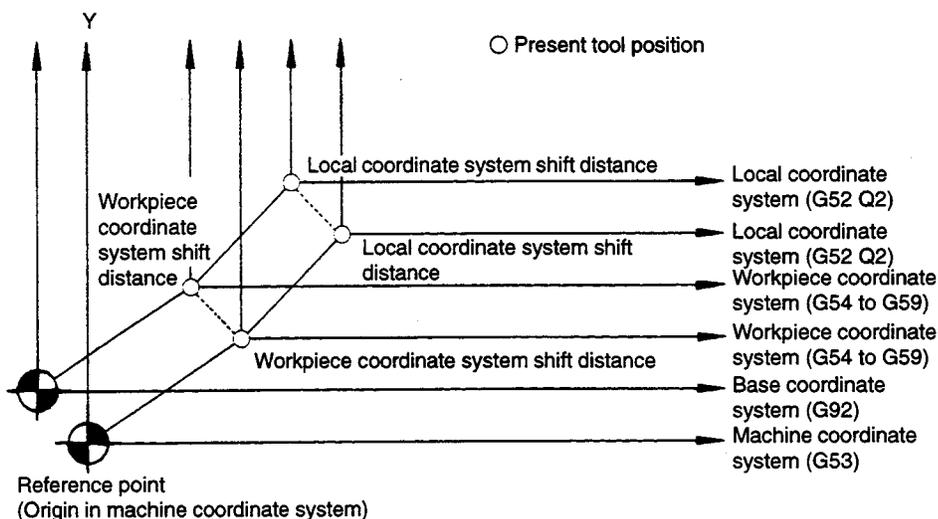
Actual Z-axis movement distance is 1400.

G54 shift distance Z = 300.

Offset H01 = 100.

- If G92 is specified while a program is executed in a workpiece coordinate system set by G54 to G59, all workpiece coordinate systems (G54 to G59) and the base coordinate system are shifted so that the coordinate values of the present position will be the coordinate values specified in the G92 block. Therefore, G92 must not be specified in the G54 to G59 mode.
- To change a workpiece coordinate system by G54 to G59, select the G90 mode before calling the new coordinate system and select the G90 mode again before returning to the base coordinate system.
- G54 to G59 must be specified in the G00 or G01 mode. If these G codes are specified in another mode, alarm “0322” occurs.

- If the workpiece coordinate system cancel (G52) is executed, the coordinate system returns to the base machine coordinate system that is used as the reference of the workpiece coordinate system.



Note: The reference point and the origin of the machine coordinate system are the same point.

FIGURE 3.1.3.3 Correcting the Workpiece Coordinate System Shift Distance

- The machine coordinate system, base coordinate system, and workpiece coordinate systems that have been set are not influenced by the reset operation. To reset the coordinate systems that have been set, operations meeting the type of coordinate system to be reset are required.
- Axis move commands can be specified with the workpiece coordinate system setting command in the same block. In this case, the move commands are executed in that workpiece coordinate system.

3.1.4 Local Coordinate System (G52 Q2)*

The coordinate system established by shifting the workpiece coordinate system by the specified shift distance is called the local coordinate system. Once a local coordinate system is set, a cutting tool moves in that system. A local coordinate system is valid only after the setting of a workpiece coordinate system. If local coordinate system setting commands are specified although a workpiece coordinate system has not been set, alarm “0321” occurs.

(1) Command Format

- By specifying “G52 Q2X ••• Y ••• Z ••• (*a ••• b •••);”, the coordinate system that is shifted from the present workpiece coordinate system by the specified shift distance can be set.
- By specifying “G52 Q2 X0 Y0 Z0 (* a0 b0);”, the local coordinate system is canceled and the previously selected workpiece coordinate system is recovered.
- By specifying “G52;” independently in a block, the base coordinate system is called. The base coordinate system is called with G52 specified independently in a block.

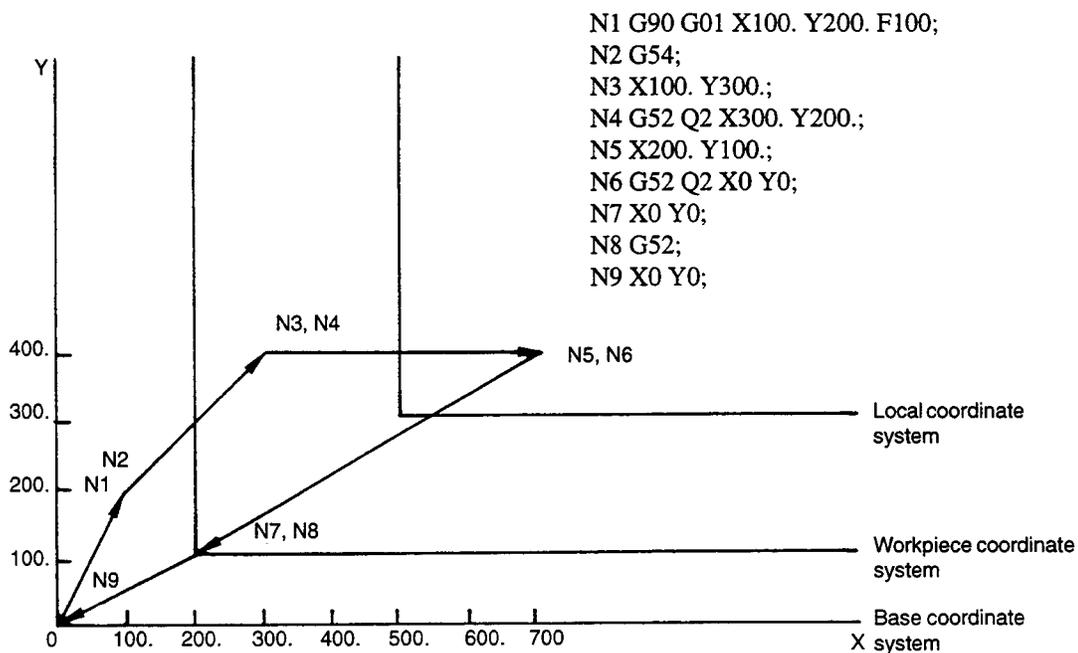


FIGURE 3.1.4.1: Local Coordinate System
[Workpiece Coordinate System Shift Distance (200., 100.)]

(2) Supplements to Local Coordinate System Setting Commands

- In the state where a workpiece coordinate system and local coordinate system have been set, G92 command must not be executed. An operation to set a coordinate system at the NC operation panel must also be avoided. Also refer to supplements indicated in “Base Coordinate System (G92)”.
- If G52 is executed while this optional function is not selected, it cancels the workpiece coordinate system (G54 to G59).
- The local coordinate system is canceled by the reset operation.

3.1.5 Machine Coordinate System (G53)

The machine coordinate system, specific to the machine, is determined by the execution of reference point return operation. It has the origin (0, 0, 0) at the reference point.

(1) Command Format

By specifying “(G90) G53 X...Y...Z...;”, axis move commands in this are executed in the machine coordinate system. G53 is a non-modal G code.

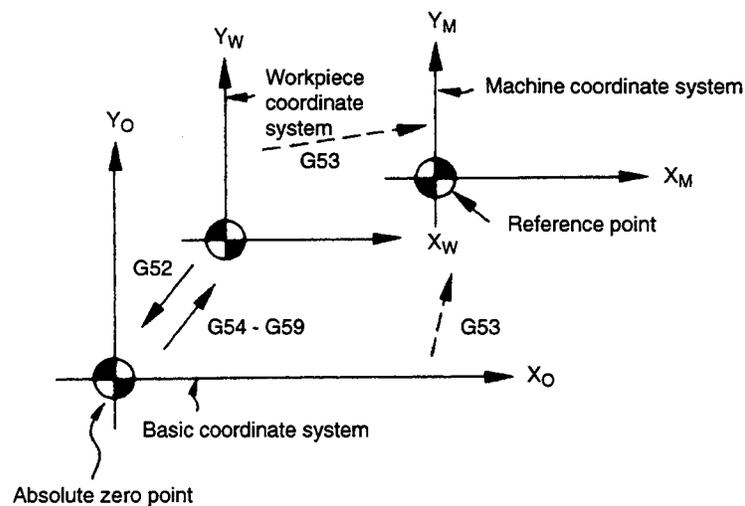


FIGURE 3.1.5.1: Machine Coordinate System

(2) Example of Programming

How the coordinate systems are defined by the program created based on the machine coordinate system is shown in Fig. 3.1.5.2.

Example of Programming

```

N1  G92  X200.  Y100.;
N2  G54  G90  X100.  Y200.;
N3  G53  X300.  Y100.;
N4  X300.  Y0;
N5  G52;
N6  X0  Y0;

```

Workpiece coordinate system (1) shift distance (300., 300.)

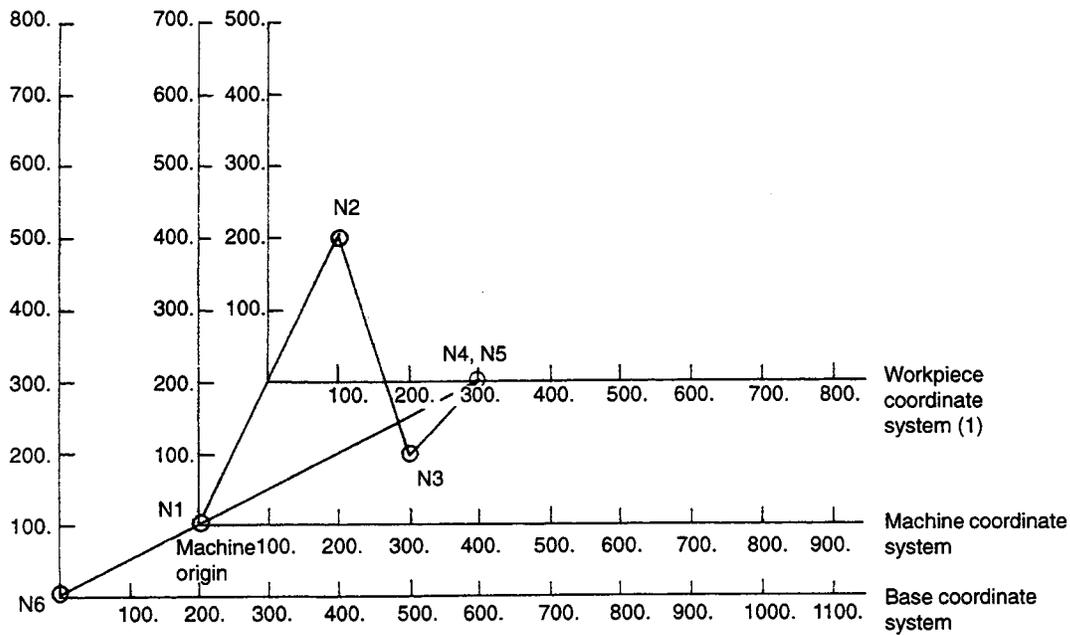


FIGURE 3.1.5.2 Example of Program in the Machine Coordinate System (G53)

(3) Supplements to the Machine coordinate System Commands

- The G53 command can be executed when all of the following conditions are satisfied. If it is executed before all of them are satisfied, alarm “0322” occurs.
 - Not in the mirror image mode
 - Not in a canned cycle mode or tool radius offset C mode
 - The immediately preceding 01-group G code is any of G00, G01, and G60.
 - G53 and G00 are not specified in the same block.
- If G53 is executed with machine lock ON, move command is processed to update the present position display until the value to which the move command must be processed if machine lock were OFF, and then the processing is completed. Accordingly, if machine lock ON/OFF setting is switched over while the commands in the G53 block are being executed, positioning cannot be made correctly. However, if all commands in the G53 block are executed in the machine lock OFF state, positioning can be completed correctly even if machine lock status is switched over beforehand.
- G53 should be executed in the G90 mode. If it is executed in the G91 mode, the commands in the G53 block are executed assuming the G90 mode.
- If G53 is executed in the tool length offset or tool position offset mode, the commands in the G53 block are executed in a manner to cancel the offset mode. Therefore, the tool length offset or tool position offset mode should be canceled before specifying the G53 command.

3.1.6 Rotation of Coordinate System (G68, G69)*

(1) Using the G68 and G69 Commands

(a) Features of G68 and G69

For the rotation of a coordinate system, the following G codes are used.

Table 3.1.6.1 Coordinate System Rotation G Codes

G code	Function	Group
G68	Coordinate system rotation	18
G69	Cancel of coordinate system rotation	18

G68 and G69 are modal G codes belonging to 18-group. When the power is turned ON and when the NC is reset, G69 is automatically selected.

The G68 and G69 blocks must not include other G codes.

The coordinate system rotation which is called by G68 must be canceled by G69.

(b) Command format

- By specifying “G17 (or G18, G19) G68a ••••• b ••••• R •••••;”, the commands specified in the following blocks are rotated by the angle specified with R around the point (a, b). Rotation angle can be specified in units of 0.001 degree.

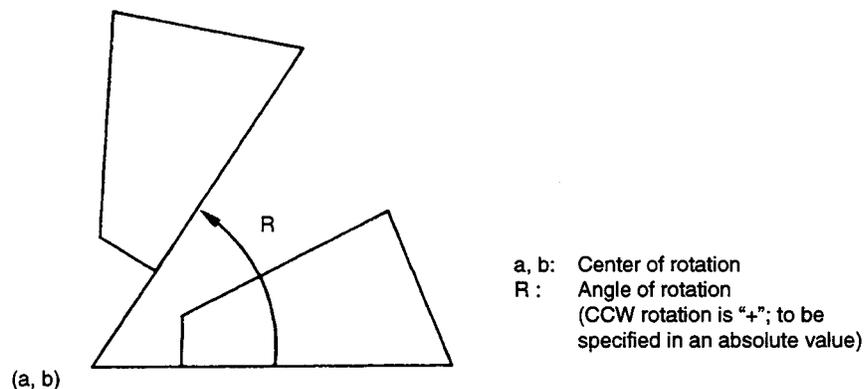


FIGURE 3.1.6.1: Rotation of Coordinate System

- By specifying “G69;”, the coordinate system rotation mode is canceled by “G69”.
- The G68 command is executed in the plane that has been selected when the G68 command is specified. The 4th- and 5th-axis must be linear axes.

G17: XY plane or Xa, Xb plane

G18: ZX plane or Za, Xb plane

G19: YZ plane or Ya, Xb plane

(2) Supplements to the Coordinate System Rotation Commands

- If “a” and “b” are omitted, the present position when the G68 block is executed is taken as the center of rotation. Note that R must not be omitted.
- When the coordinate system is rotated, position data are given in the rotated coordinate system.
- The workpiece coordinate system rotation, mirror image, and scaling functions must be specified in the predetermined order in a program. If this order is disregarded, alarm “0285” occurs.
- Workpiece coordinate > Mirror image > Scaling > Coordinate system
 system rotation (G54) (M94) (G51) rotation (G68)
- If the coordinate system rotation is executed in the mirror image mode, mirror image is applied to the center of rotation and also in the direction of rotation.
- If the coordinate system rotation is executed in the scaling mode, scaling is applied to the center of rotation. In this case, however, scaling is not applied to the angle of rotation.
- The coordinate system rotation (G68) command cannot be specified in the tool radius offset C mode. If it is specified in the tool radius offset C mode, alarm “03 10” occurs.
- Usually, the coordinate system rotation is turned ON before the start of approach motion and turned OFF after the completion of machining. The workpiece cannot be machined correctly if it is turned ON during machining.
- The alarm codes related to the coordinate system rotation are indicated in Table 3.1.6.2.

Table 3.1.6.2 Alarm Codes

Alarm No.	Description
0310 G code not allowed in the G68 mode	A G code that cannot be specified in the G68 mode is specified. G 68 is specified in the tool radius offset C mode.
0311 Format error	The G68 or G69 block includes a format error.

3.1.7 Plane Selection (G17, G18, G19)

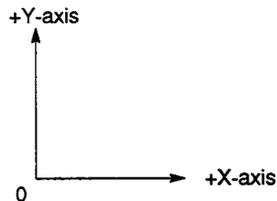
The plane where circular interpolation, tool radius offset, and coordinate system rotation are executed is selected by specifying the following G code.

Table 3.1.7.1 Plane Selection G Codes

G code	Function	Group
G17	XY plane	02
G18	ZX plane	02
G19	YZ plane	02

A plane is defined in the following manner (in the case of X Y plane):

The horizontal axis in the first quadrant is “+X-axis” and the vertical axis in the same quadrant “+Y-axis”.



- When the power is turned ON, the XY plane (G17) is selected.
- Axis move command of a single axis can be specified independent of the selection of plane by G 17, G18, and G 19. For example, the Z-axis can be moved by specifying “G17 Z •••;”.
- Execution of a canned cycle is possible only in the G17 plane (hole machining axis: Z-axis).
- If the 4th-axis * is selected, the following planes are added.
 - G17: XY plane or Xa plane
 - G18: ZX plane or Za plane
 - G19: YZ plane or Ya plane
- If the 5th-axis * is selected, the following planes are added.
 - G17: XY plane or Xb plane
 - G18: ZX plane of Zb plane
 - G19: YZ plane or Yb plane
- The plane on which the tool radius offset is executed by the G41 or G42 command is determined by the designation of G 17, G18 or G 19; the plane that includes the rotary 4th- or 5th-axis cannot be selected as the offset plane.

3.2 DETERMINING THE COORDINATE VALUE INPUT MODES

This section describes the commands used to input coordinate values.

3.2.1 Absolute/Incremental Designation (G90, G91)

These G codes specify whether dimension values specified following an axis address are given in an absolute value or incremental value.

(1) Using the G90/G91 Command

(a) Features of G90 and G91

Table 3.2.1.1 Absolute/incremental Designation G Codes

G code	Function	Group
G90	Absolute designation	03
G91	Incremental designation	03

- G90 and G91 are modal G code belonging to 03-group. If G90 and G91 are specified in the same block, the one specified later is valid.
- The power-ON state, that is the G90 or G91 mode that is valid when the power is turned ON, can be set for parameter pm4000 D1.

pm4000 D1 = 0	Power-ON state is G90.
pm4000 D1 = 1	Power-ON state is G91.

- The reset-state, that is the G90 or G91 mode that is valid when the NC is reset, can be set for parameter pm4000 D6.

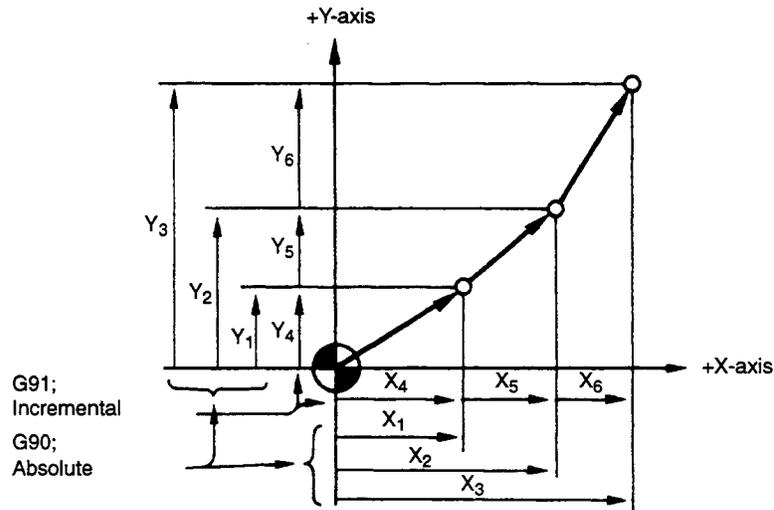
pm4000 D6 = 0 and pm4000 D 1 = 0: Reset-state is G90.

pm4000 D6 = 0 and pm4000 D 1 = 1: Reset-state is G91.

pm4000 D6 = 1: The G code (G90 or G91) that was valid before reset, disregarding of the setting for pm4000 D1.

(b) Command format

- For the commands specified in and after the G90 block, the dimension values specified following an address of X, Y, Z, (* a, b) are treated as absolute values.
- For the commands specified in and after the G91 block, the dimension values are treated as incremental values.

**FIGURE 3.2.1.1: Absolute/Incremental Commands (G90, G91)**

3.2.2 Inch/Metric Input Designation (G20, G21)

It is possible to select the dimension unit for the input data between “mm” and “inches”. For this selection, the following G codes are used.

Table 3.2.2.1 Dimension Unit Selection G Codes

G code	Function	Group
G20	Input in “inch” system	06
G21	Input in “mm” system	06

(1) Command Format

G20 and G21 should be specified at the beginning of a program in a block without other commands. When the G code which selects the input dimension unit is executed, the following values are processed in the selected dimension unit: subsequent programs, offset amount, a part of parameters, a part of manual operation, and display.

Example of Programming

ER

CR

01234;

G20; ? Designating the input in “inch” system

•
•
•

(2) Supplements to the Dimension Unit Designation Commands

- A parameter is used to select “inch/mm”. Therefore, the state when the power is turned ON is determined by the setting for this parameter.
- If the dimension unit system should be switched over during the execution of a program, the following processing must be accomplished in advance.
 - If a workpiece coordinate system (G54 to G59) is being used, return it to the base coordinate system.
 - Cancel all tool offsets (G41 to G48).
- After switching over the dimension unit system between G20 and G21, the following processing must be accomplished.
 - Execute G92 (coordinate system setting) for all axes before specifying axis move commands.
 - If position data are displayed in a workpiece coordinate system, or when an

external position data display unit is used, reset the present position data to “0”.

- The tool offset amounts stored in memory are treated in a different manner between the G20 and G21 modes.

Table 3.2.2.2 Tool Offset Amounts in G20 and G21 Modes

Stored Offset Amount	In the G20 (Inch System) Mode	In the G21 (mm System) Mode
150000	5.9055	150.000 mm

3.2.3 Scaling (G50, G51) *

The shape defined by a part program can be enlarged or reduced according to a required scale. For the scaling processing, the following G codes are used.

Table 3.2.3.1 Scaling G Codes

G code	Function	Group
G50	Scaling OFF	15
G51	Scaling ON	15

The G50 and G51 blocks must be specified in the manner as indicated above without other commands entered in these blocks. If other commands such as other G codes and X, Y, or Z are entered in these blocks, alarm “0281” occurs.

The scaling function which is called by G51 must be canceled by G50. If G51 is specified in the scaling mode, it is disregarded.

(1) Using the G50 and G51 Commands

- By specifying “G51 I ••• J ••• K ••• P •••;”, a program is executed on the shape enlarged or reduced by the ratio specified with address P. The center of scaling is defined by I, J, and K (representing X-axis, Y-axis and Z-axis, respectively). To establish the present position as the center of scaling, specify “G51 I0 J0 K0 P •••;”.
- The scaling mode can be canceled by the command of “G50;”.
- Scaling ratio can be specified in the following range.
- Scaling range: 0.000001 to 99.999999 times
- Command unit of P is “1 = 0.000001”.

If a decimal point is used, the value of P is assumed to have six digits to the right of the decimal point only in the G51 block.

(Example)

P 0.999999‡ 0.999999 times

P 2.0 ‡ 2 times

P2 ‡ 0.000002 times

- If P is omitted, the scale factor determined by the setting for setting parameters pm0803 and pm0804 is used. It is necessary to set the scale factor within the allowable range.

Scale factor = pm0803/pm0804

(Example) If pm0803 = 3 and pm0804 = 100,

Scale factor = 3/100= 0.03 times

- If I, J, and/or K is specified in the G51 block, scaling is valid only for the specified axes. Note that scaling is not possible for additional axes (4th- and 5th-axis).
- For I, J, and K in the G51 block, specify the distance from the origin of the workpiece coordinate system to the center of scaling if the workpiece coordinate system has been selected.

Example of Programming 1

```
N1 G92 X0 Y0;
N2 G00 G90 X200. Y100.;
N3 G51 I200. J200.;
```

.

.

.

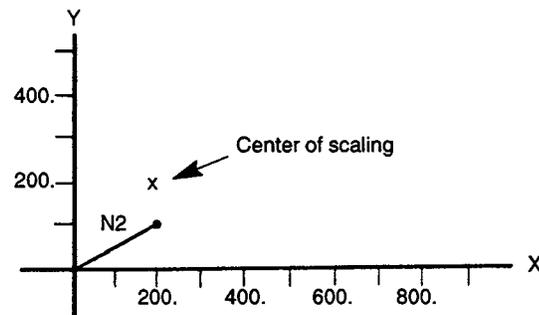


FIGURE 3.2.3.1: Example of Programming 1

Example of Programming 2

```

N1 G54
N2 G00 G90 X200. Y100.;
N3 G51 I200. J200.;

```

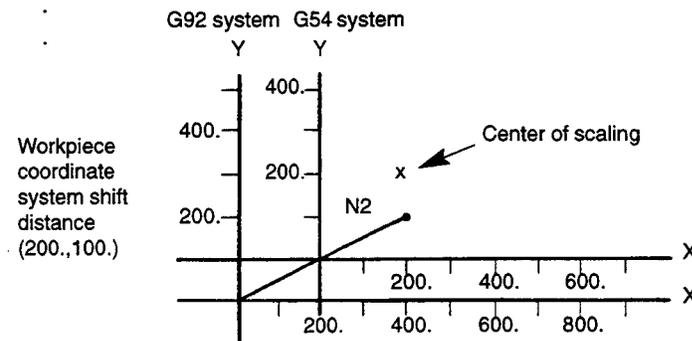


FIGURE 3.2.3.2: Example of Programming 2

(2) Supplements to the G50 and G51 Commands

- The scaling mode should be turned ON in the approach operation to start machining and turned OFF after retracting a cutting tool from the workpiece. If it is turned ON after the start of machining, the shape cannot be machined accurately.
- Scaling is applied to the two axes in the plane for machining. If scaling is applied to only one axis, the circular arc will be distorted. Therefore, an alarm occurs when circular interpolation is going to be executed if scaling is applied to only one axis.
- If a scale factor greater than “1” is set, make sure that the commands after scaling will not exceed the maximum programmable value.
- Scaling factor of “0” cannot be specified. If specified, alarm “0281” occurs.
- For offset amounts, scaling is not applied.
- It is not allowed to execute a canned cycle with scaling applied to the Z-axis. Otherwise, alarm “0280” occurs.
- The scaling mode is canceled (G50) by the reset operation (pressing the [RE-SET] key, M02, M03, or external reset).
- In the scaling mode, command values and present position data display are all processed after scaling.
- In the scaling mode, the following G codes cannot be specified. If specified, alarm “0280” occurs.
G27, G28, G29, G30, G31, G52Q2, G53, G92
- Scaling (G51) command cannot be specified in the tool radius offset C mode. Otherwise, alarm “0280” occurs.
- Alarm code relating to the scaling function are indicated in Table 3.2.3.2.

3.3 TIME-CONTROLLING COMMANDS

3.3.1 Dwell (G04)

It is possible to suspend the execution of axis move commands specified in the next block for the specified length of time (dwell period).

By specifying “G04 P•••;”, execution of programmed commands is suspended for the length of time specified by address P.

- Command unit of address P is “1 = 0.001 sec”. For example, a dwell period of 2.5 seconds is specified by “G04 P2500;”. The block used to specify dwell must not include commands other than G04 and P commands.
- The maximum programmable value with address P is indicated in Table 3.3.1.1.

Table 3.3.1.1 Dwell Period (Programmable Range of P)

Format	Programmable Range of Dwell Period (P)
P63	0 to 999999.999 sec

Note: The value is independent of the input and output unit systems.

- The timing at which the dwell should start can be selected by the setting for parameter (pm4015 D3) from “execution when the remaining number of pulses which have been distributed in the previous block has reduced to the allowable limit” or “execution immediately after the completion of pulse distribution”.
- Address X can be used instead of address P.

3.3.2 Exact Stop (G09)

If a block containing G09 is executed during cutting, and it calls the error detection ON mode; the program advances to the next block in the error detection ON mode. This function is used to cut a corner with a sharp angle. G09 is a non-modal G code and valid only in the specified block.

3.3.3 Exact Stop Mode (G61, G64)

The exact stop mode is called by G61. Once G61 is executed, the following blocks are all executed in the error detection ON mode until the exact stop mode is canceled by G64.



◆ Error Detection ON Mode

In the error detection ON mode, the number of lag pulses due to servo delay is checked after the completion of pulse distribution in linear and circular interpolation, and the program advances to the next block only after the number of lag pulses has been reduced to the preset value.



◆ Error Detection OFF Mode

In the normal linear and circular interpolation, which is called in other than the G09 or G61 mode, the program advances to the next block immediately after the completion of pulse distribution, causing the corners to be rounded due to servo lag. This operation mode is called the error detection OFF mode.



◆ Error Detection ON/OFF for Rapid Traverse Operation

In rapid traverse operation, error detection ON/OFF are controlled only by G00 and G06 and the G codes explained above are ignored.

3.4 TOOL OFFSET FUNCTIONS

The following three kinds of tool offset functions are provided: tool length offset function, tool position offset function (used for simplified tool radius offset), and tool radius offset C function* (used for tool radius offset for complicated shape).



1. If the tool wear offset function is selected optionally, the offset amount used in the tool length offset function and the tool radius offset function is the sum of the tool geometry offset amount and the tool wear offset amount.
2. Two types of tool offset number designation methods are provided and which method should be used can be selected by the setting for a parameter.

pm4029D0 = 0: H/D used in common

D0 = 1: H/D used for individual functions

An H code is used for the tool length offset function and a D code for the tool position offset and tool radius offset functions.

3.4.1 Tool Offset Data Memory

To execute the tool offset functions, the tool offset amount must be written to the tool offset data memory in advance. Depending on the selected specifications, the following number of memories is available.

Table 3.4.1.1 Tool Offset Data Memory

	Tool Offset Data Memory
Basic specifications	99
Addition of tool offset data memory (option 1)	299
Addition of tool offset data memory (option 2)	999
Addition of tool offset data memory (option 3)	1199

Table 3.4.1.2 indicates the setting range of tool offset amount.

Table 3.4.1.2 Setting Range of Tool Offset Amount

G code	Linear Axis	*Rotary Axis
Metric input	0 to ± 999.999 mm	0 to ± 999.999 deg.
Inch input	0 to ± 999.999 inch	0 to ± 999.999 deg.

Note: These values are not influenced by the output unit.

3.4.2 Tool Length Offset (G43, G44, G49)

The tool length offset function adds or subtracts the amount stored in the tool offset data memory to or from the Z coordinate values specified in a program to offset the programmed paths according to the length of a cutting tool.

(1) Commands

In the execution of the tool length offset function, addition or subtraction of the offset data is determined by the specified G code and the direction of offset by the H code.

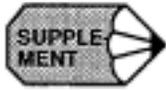
(a) G Codes used for tool length offset

The tool length offset function is called by the following G codes.

Table 3.4.2.1 G Codes Used for Tool Length Offset

G code	Function	Group
G43	Add	08
G44	Subtract	08
G49	Cancel	08

- G43 and G44 are modal and, once executed, they remain valid until canceled by G49. G49 cancels the tool length offset mode. H00 also cancels the tool length offset mode.
- By specifying “G43 (or G44) Z ••• H •••;”, tool offset amount specified by the H code is added to or subtracted from the specified Z-axis position, and the Z-axis moves to this offset target position. That is, the target position of Z-axis movement specified in the program is offset by the tool offset amount.
- By specifying “(G01) Z •••; G43 (or G44) H ••• 0;”, the Z-axis moves by the distance corresponding to the tool offset amount which is specified by the H code.
- By specifying “G43 (or G44) Z ••• H •••; H •••;”, the Z-axis moves by the distance which is equivalent to the difference between the previous tool offset amount and the new tool offset amount.



G43, G44, and G49 can be specified only in the mode called by the G code (G00, G01, or G60) in 01 group. If they are specified in other modes such as G02 or G03 mode, an error occurs.

(b) H code for designating the direction of offset

The direction of offset is determined by the sign of tool offset amount, specified by an H code, and a G code.

Table 3.4.2.2 Sign of Tool Offset Amount and Direction of Offset

	Sign of Tool Offset Amount (H code)	
	Positive	Negative
G43	Offset in the positive direction	Offset in the negative direction
G44	Offset in the negative direction	Offset in the positive direction

Example of Programming

```

H10 ..... Offset amount  -3.0
H11 ..... Offset amount   4.0
                                     Position data display
                                     including offset amount
                                     (Z-axis only)
N101 G92 Z0;                          0.000
N102 G90 G00X1.0Y2.0;                  0.000
N103 G43 Z-20. H10;                    -23.000
N104 G01 Z-30. F1000;                  -33.000
N105 G00 Z0H00;                        0.000
.
.
.
N201 G00 X-2.0Y-2.0;
N202 G44 Z-30 H11;                      -34.000
N203 G01 Z-40 F1000;                   -44.000
N204 G00 Z0 H00;                        0.000
    
```

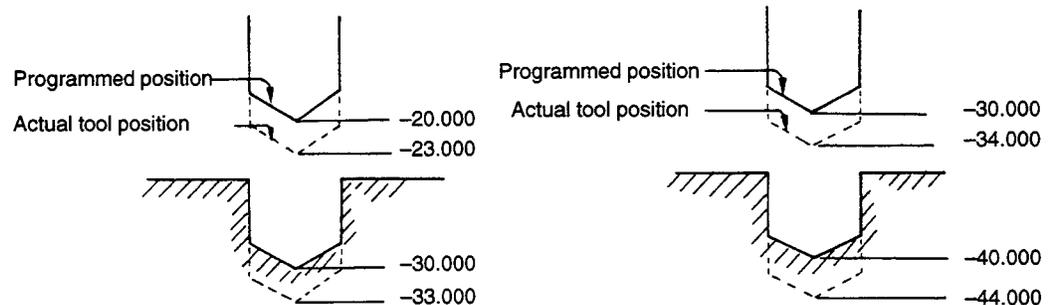


FIGURE 3.4.2.1: Tool Position Offset Function

(2) Supplements to the Tool Length Offset Function

- If the offset amount is changed by MDI operation, the new offset amount becomes valid from the block in which an H code is specified.
- It is possible to call up the tool position offset or tool radius offset C function in the tool length offset mode.
- It is not allowed to specify G43, G44, or G49 in a canned cycle mode.
- If G92 is specified with a Z command while in the tool length offset mode, the tool length offset mode is canceled. However, G92 should be specified in the state that the tool length offset mode has been canceled.
- During automatic operation, it is possible to display the tool offset number (H code number) that is presently valid.
- G43, G44, and G49 can be specified only in the G00 or G01 mode. Designation of these G codes in the G02 or G03 mode is not allowed.
- For the execution of the reference point return command, including two axes, in the tool length offset mode, whether the H code and the tool length offset amount should be saved or canceled can be specified by the setting for a parameter.

pm4010 D = 0	Both the H code and the tool length offset amount are canceled.
pm4010 D = 1	Both the H code and the tool length offset amount are saved.

- For the reset operation in the tool length offset mode, whether the H code and the tool length offset amount should be saved or canceled can be specified by the setting for a parameter. It is also possible to specify whether a G code (G49) in 08-group should be saved or canceled.

pm4010 D = 0	Both the H code and the tool length offset amount are canceled.
pm4010 D = 1	Both the H code and the tool length offset amount are saved.

3.4.3 Tool Position Offset (G45 to G48)

The tool position offset function extends or contracts axis movement distance specified in a program by the distance set in the tool offset data memory, and is primarily used for compensating for the tool radius when cutting a square shape. It is not necessary to use this function if the optional tool radius offset C function (G40, G41, G42) is selected.

(1) Commands

In the execution of the tool position offset function, extension or contraction of axis movement distance is determined by the specified H code and the direction of offset by the D code.

(a) G Codes used with the tool position offset function

Table 3.4.3.1 G Codes Used with the Tool Position Offset Function

G code	Description	Group
G45	Extension/Contraction	*
G46	Contraction/Extension	*
G47	Double-extension/Double-contraction	*
G48	Double-contraction/Double-extension	*

- G45 to G48 extends or contracts axis move distance specified in the block that contains G45 to G48 in the direction of axis movement. Extension or contraction is valid only for the axis move distance specified with G45 to G48 and in other blocks, the axes move as programmed. Therefore, to return the axis with which position has been offset by the execution of G45 to G48 to the programmed position, it is necessary to execute G45 to G48 that extends or contracts the axis move distance in the opposite direction.
- To use the tool position offset function, the program must be written in the incremental mode (G91). If it is written in the absolute mode (G90), the function extends or contracts the distance (incremental value) from the end point reached in the block preceding the offset block to the target point specified in absolute values in the offset block along the axis move direction.
- To execute the tool position offset function, specify a G code calling up the tool position offset function (G45 to G48), axis address and the tool offset data memory number by using a D code in the same block. If a D code is used to select the tool radius offset data, it can be omitted since D code is modal. Store the radius of a cutting tool in the tool offset data memory.

Example of Programming

```

G91
① G00 G46 X·····Y·····D01;      Contraction
② G01 G47 Y·····(D01) F·····;  Double-extension
③   G47 X·····(D01);           Double-extension
④   G47 Y·····(D01);           Double-extension
⑤   G47 X·····(D01);           Double-extension
⑥ G00 G46 X·····Y·····(D01);  Contraction
    
```

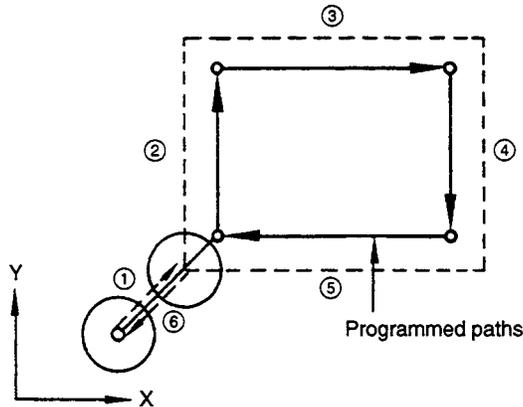


FIGURE 3.4.3.1: Tool Position Offset Function

(b) G code for specifying the direction of offset

Whether the tool position offset is processed in extension or contraction is determined by the sign of the tool offset amount specified by a D code and the G code specified to call up the tool position offset function as indicated in the table below.

However, the tool offset amount to be used in the G45 to G48 function should be stored with the positive sign.

Table 3.4.3.2 Sign Assigned to Tool Offset Data

	Sign of Tool Offset Amount (H code)	
	Positive	Negative
G45	Extension	Contraction
G46	Contraction	Extension
G47	Double-extension	Double-contraction
G48	Double-contraction	Double-extension

(2) Example of Programming

(a) Extension and contraction amounts

The relationship between the axis move distance specified in a program and the actual tool move distance is given below. The axis move distance specified in an incremental value is extended or contracted by one time or two times the specified tool offset amount.

Example of Programming

```
G91 G00 G47 X60. D10; D10 = 20.
```

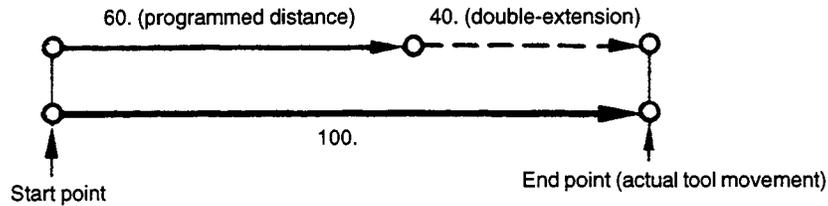


FIGURE 3.4.3.2: Extension and Contraction of Programmed Tool Move Distance

(b) Execution of extension/contraction from offset start point

If a program block is executed in the state where the start point of that block is offset by the execution of extension or contraction in the preceding block, although the total axis move distance is equal to the axis move distance specified in a program, the axis moves the specified distance starting from the offset start point

Example of Programming

```
G91 G00 G47 X60. D10; (D10 = 20.)
```

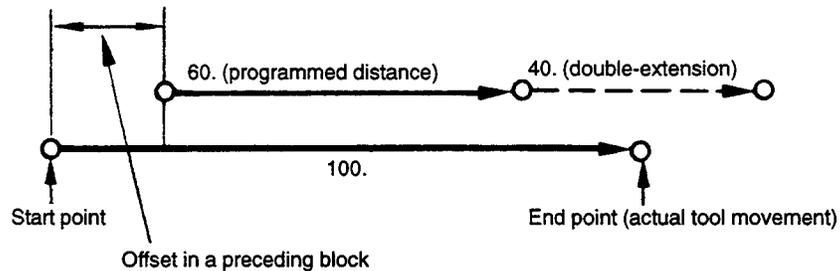


FIGURE 3.4.3.3 Execution of Commands from Offset Start Point

(c) Reversed axis movement direction due to execution of function

If the specified tool offset amount is larger than the axis move distance specified in a program, axis movement direction might be reversed as the result of extension or contraction. If this occurs, the axis moves in the direction opposite to the direction specified in the pro-

gram.

Example of Programming

```
G46 X10. D10; D10 = 20.
```

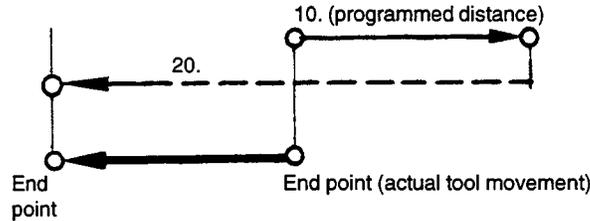


FIGURE 3.4.3.4 Reversed Axis Movement Direction

(d) I, J, and/or Kin G45 to G48 blocks

If I, J, and K are specified in the G45 to G48 block, each of these addresses calls up extension or contraction processing in the same direction as X-, Y-, and Z-axis, respectively. Therefore, tool radius offset is possible only for a one-quarter (1/4) circle, three-quarter (3/4) circle, or a full (4/4) circle. Note that a half-circle should be defined by combining two one-quarter (1/4) circles. In practical operation, the offset should be applied from the block preceding the circular interpolation block to obtain the arc correctly offset. See Fig. 3.4.3.6 below.

Example of Programming

```
G91
G45 G02 X50. Y50. I50. D10; D10 = 20.
```

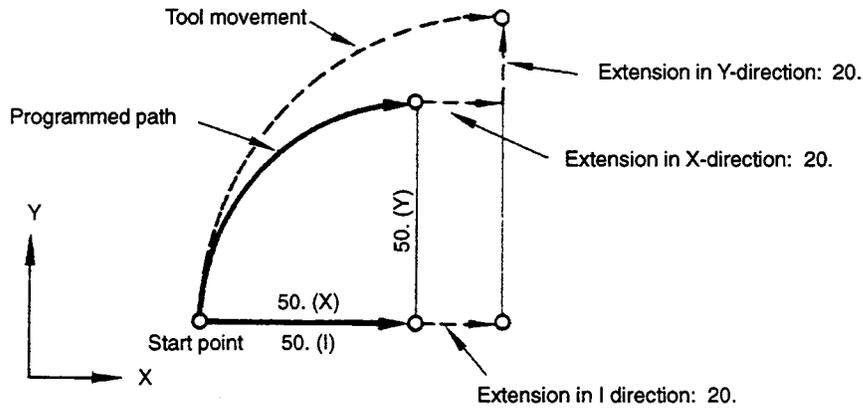


FIGURE 3.4.3.5 Tool Radius Offset for a 1/4 Circle

Example of Programming

```

G91 G01 F ··· ;
G46 X ··· Y ··· D10;
G45 Y ··· ;
G45 G02 X ··· Y ··· I ··· ;
G01 X;

```

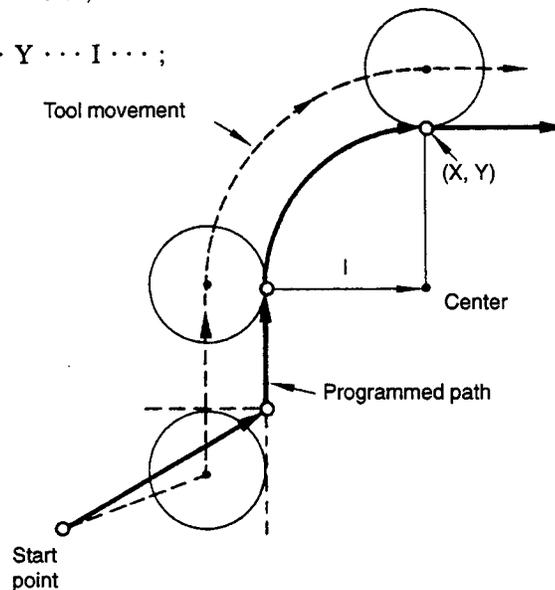


FIGURE 3.4.3.6 Offset Applied from the Preceding Block

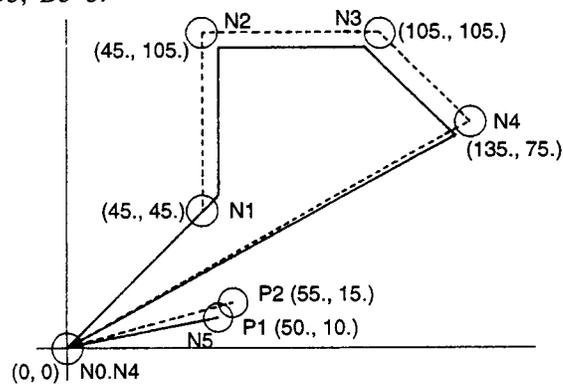
(e) Canceling G45 to G48 by G28 (or G30)

Example of Programming (G91 mode)

```

N0 G92 X0 Y0;
N1 G46 G00 X50. Y50. D5; D5=5.
N2 G47 Y50.;
N3 G47 X50.;
N4 G28 X30. Y-30.;
N5 X50. Y10.;

```



- Note
- 1: Although offset is applied to the intermediate positioning point specified in N4, the machine origin is not offset.
 - 2: Axis movement specified in N5 is executed in the following manner according to the setting for pm4010.
 Offset cancel is selected by pm4010: Movement to P1
 Offset not cancel is selected by pm401C: Movement to P2

FIGURE 3.4.3.7 Canceling G45 to G48 by G28 (G30)

(f) Canceling G45 to G48 by G92

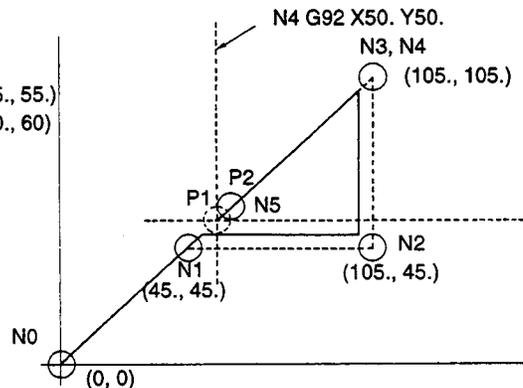
Example of Programming (G91 mode)

```

N0 G92 X0 Y0;
N1 G46 G00 X50. Y50. D5; D5 = 5.
N2 G47 X50.;
N3 G47 Y50.;
N4 G92 X50. Y50.;
N5 X-50. Y-50.;

```

P1 (55., 55.)
P2 (60., 60.)



Note: Axis movement specified in N5 is executed in the following manner according to the setting for pm4010.

Offset cancel is selected by pm4010: Movement to P1

Offset not-cancel is selected by pm4010: Movement to P2

FIGURE 3.4.3.8 Canceling G45 to G48 by G92

(3) Supplements to the Tool Offset Function

- In a G45 to G48 block, only a G code in 01-group (G00, G01, G02, G03, G60) can be specified with G45 to G48. If a G code in other groups is specified, alarm “0365” occurs.
- To move an axis only by the offset amount in the incremental mode (G91), specify “0” for the axis address. In this case, axis movement direction cannot be reversed even if “0” is preceded by a sign.

• Example of Programming 1

```
G91 G01 G45 X0 Y0 D10 F...;
```

Both the X- and Y-axis move in the positive direction by the offset amount called up by D10.

• Example of Programming 2

```
G91 G00 G46 X0 D11;
```

The X-axis moves in the negative direction by the offset amount called up by D11.

- For the designation of an offset number in the tool position offset mode (G45 to G48), either H or D code can be used. Which address characters should be used can be

selected by the setting for a parameter.

pm4011 D1 = 1	Only H codes are valid.
pm4011 D1 = 0	Only D codes are valid.

- If G45 to G48 is specified for simultaneous two-axis movement commands, the axis move distance is extended or contracted for both axes. Therefore, if the tool position offset function is used in such a manner, the workpiece may be cut excessively or an uncut portion may be left.

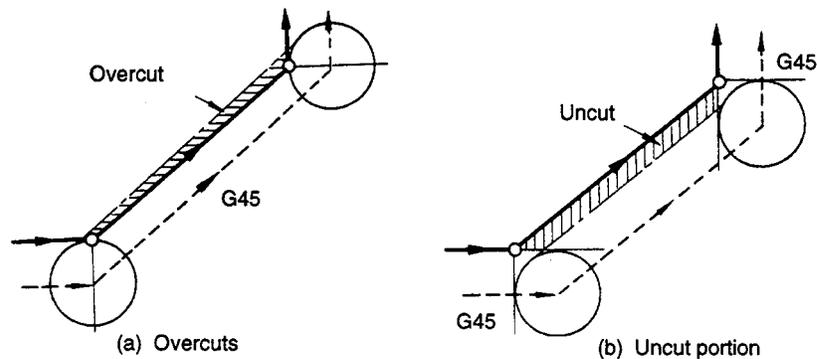


FIGURE 3.4.3.9 Calling G45 to G48 Mode for Simultaneous Two-axis Operation

- If an offset amount is changed by using the switches on the NC operation panel, the offset amount presently called is not changed immediately. The new offset amount becomes valid when G45 to G48 is executed next.
- The tool position offset function can be called up in the tool length offset mode.
- The mirror image function can be called up in the tool position offset mode. In other words, mirror image machining is possible using the offset tool paths.
- The tool position offset function can be executed independent of the plane selection G code (G17, G18, G19).
- It is not allowed to call up the tool position offset function (G54 to G58) in a canned cycle mode. If G54 to G58 is specified in the canned cycle mode, alarm “0170” occurs.
- If G92 is specified in the offset mode, the coordinate system is set after canceling the offset for the axes specified in the G92 block. The axes should be returned to the position not including offset by executing extension or contraction commands in the direction opposite to the presently offset direction before specifying G92. The axis move commands given in the next block are executed in the following manner according to the setting for parameters.

When parameter pm4011 D1 = 0:

pm4010 D6 = 0	Moves as programmed.
pm4010 D6 = 1	Offset applied.

When parameter pm4011 D1 = 1

pm4010 D7 = 0	Moves as programmed.
pm4010 D7 = 1	Offset applied.

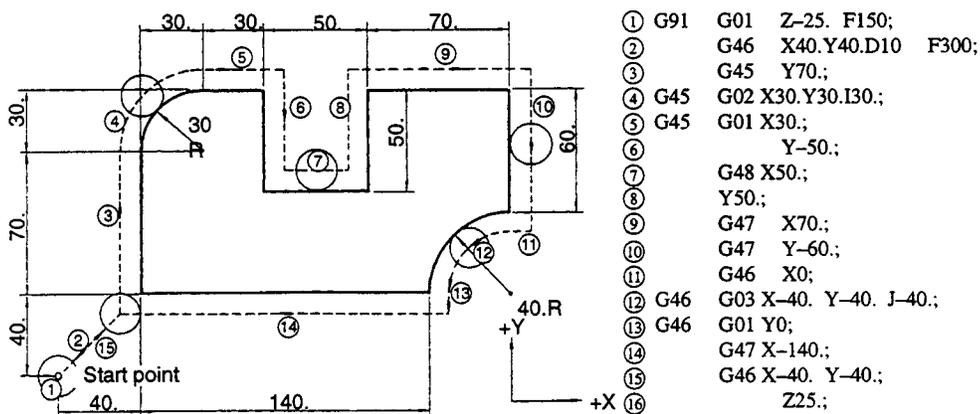


FIGURE 3.4.3.10 Example of Programming 1

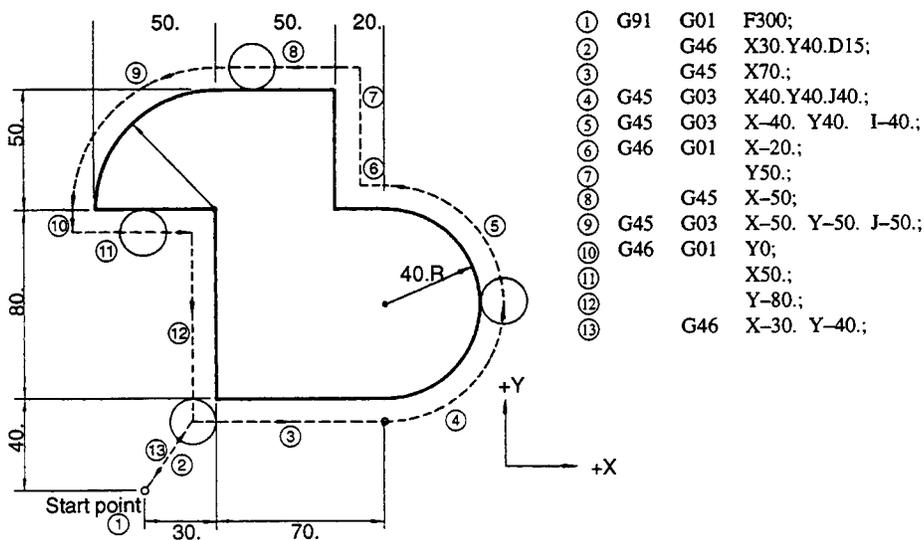


FIGURE 3.4.3.11 Example of Programming 2

3.4.4 Radius Offset C Function (G40, G41, G42) *

The tool radius offset function automatically offsets the programmed tool paths by specifying the radius of the cutting tool to be used. The distance to be offset (radius of cutting tool) can be stored to the tool offset data memory by using the switches on the NC operation panel. In a program, the offset data are called up by specifying the number of the tool offset data memory using a D code.

(1) Commands

To call up the tool radius offset C function, the following G codes are used.

Table 3.4.4.1 G Codes Used to Call Up the Tool Radius Offset C Function

G code	Function	Group
G40	Tool radius offset C mode cancel	07
G41	Tool radius offset C (offset to the left)	07
G42	Tool radius offset C (offset to the right)	07

The tool radius offset C function is called up by the execution of G41 or G42 and canceled by G40. Direction of offset is determined by the designated G code (G41, G42) and the offset amount is selected by the D code which is specified with the G code designated to call up the tool radius offset mode. When the power is turned ON, the G40 mode is set.

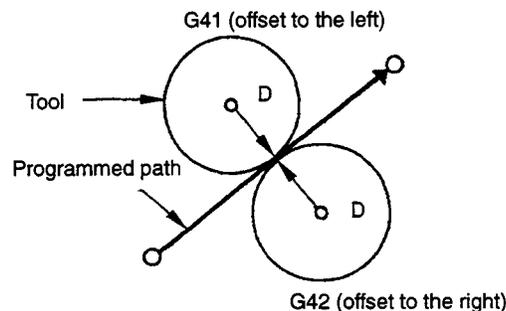


FIGURE 3.4.4.1: Tool Radius Offset C

- If a negative value is set in the tool offset data memory specified by the D code, the offset direction is reversed. The D code must be specified with G41 or G42 in the same block or in a preceding block. If D00 is specified, it specifies the tool radius of “0”.
- The tool radius offset plane is selected by the designation of G17, G18, or G19. The G code used to select the plane must be specified with G41 or G42 in the same block or in a block preceding the G41 or G42 block.

Table 3.4.4.2 Plane Selection G Codes

G code	Function	Group
G17	XY plane selection	02
G18	ZX plane selection	02
G19	YZ plane selection	02

- It is not allowed to change the selected plane in the offset mode. If a plane selection G code is specified in the offset mode, alarm “0183” occurs.
- The tool radius offset C function is not valid for the 4th- and 5th-axis. These axes are treated as an axis not in the selected offset plane.

(2) Entering the Offset Mode

There are two types of offset mode entry methods (types A and B) and which method should be used can be selected by the setting for a parameter.

pm4012 D6 = 0	Type A
pm4012 D6 = 1	Type B

(a) Type A: pm4012 D6 = 0

Start-up motion (axis movements for the start-up of offset) is not executed if there are no axis movement commands in the G41 or G42 block.

- Start-up motion is executed when an axis move command is specified.
- If there are no move commands of the axes in the offset plane in the block next to the G41 or G42 block, one more block is buffered to execute the start-up motion.
- In the offset mode, specification of up to two consecutive blocks that do not contain axis move (commands in the offset plane is allowed. However, if no axis move command in the offset plane is specified in three or more consecutive blocks, offset motion is interrupted and the offset block is not generated.

(b) Type B: pm4012 D6 = 1

Start-up motion (axis movements for the start-up of offset) is executed even when an axis move command is not specified in the G41 or G42 block.

- In response to the designation of G41 or G42, the cutting tool moves to the position offset by the radius. The offset is made to the left or right side in reference to the tool advancing direction and in the normal direction at the start point of the next block.
- If there are no axis move commands in the offset plane in the G41 or G42 block, the axes move by the offset amount.
- If there are no move commands of the axes in the offset plane in the block next to the G41 or G42 block, one more block is buffered to execute the start-up motion.
- In the offset mode, specification of up to two consecutive blocks that do not contain

axis move commands in the offset plane is allowed. If no axis move commands in the (offset plane) are specified in three or more consecutive blocks, tool radius offset starts from the next block that contains axis move commands.

(3) Start-up of Offset

Since the offset start-up is executed with the offset taken into account, the G code in 01-group must be either G00 or G01. If a G code other than G00 or G01 is specified, alarm “0180” occurs. If the offset starts in the G00 mode, the axes move to the offset point at their individual rapid traverse rates. Therefore, be aware of possible interference of a cutting tool with the workpiece. There are two types of start-up such as start-up at inside corner and start-up at outside corner.

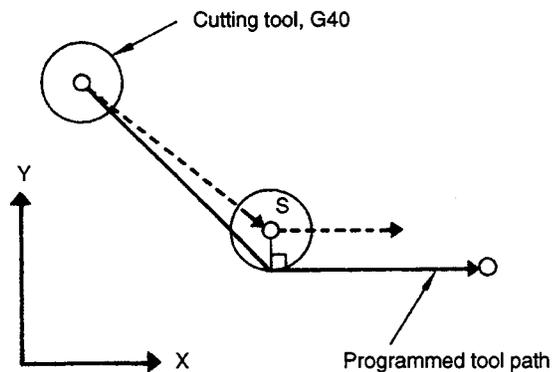
(a) Start-up at inside corner (180° or less)

The cutting tool moves to the offset point (on the normal start point of the vector of movement called up in the block next to the start-up block).

- From straight-line to straight-line

Example of Programming

```
G17 G01 F ..... ;
G41 D ..... X ..... Y ..... ;
X ..... ;
```



Note: “S” indicates the single-block stop point.

FIGURE 3.4.4.2 Offset Start-up (Straight-Line to Straight-Line)

- From straight-line to arc

Example of programming

```
G17 G01 F ..... ;
G41 D ..... X ..... Y ..... ;
G02 X ..... Y ..... J ..... ;
```

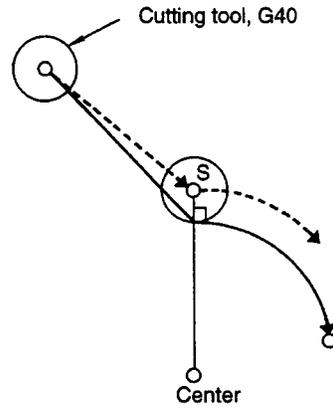


FIGURE 3.4.4.3 Offset Start-up (Straight Line to Arc)

- (b) Start-up at outside corner (180° or larger)

In this case, two kinds of start-up modes (types A and B) are provided and the mode to be used can be selected by the setting for a parameter.

pm4013 D0 = 1	Type A
pm4013 D0 = 0	Type B

- Type A: pm4013 D0 = 1

The cutting tool moves to the offset point (on the normal start point of the vector of the block next to the start-up block).

- From straight-line to straight line at outside corner (180° to 270°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
X ..... ;
```

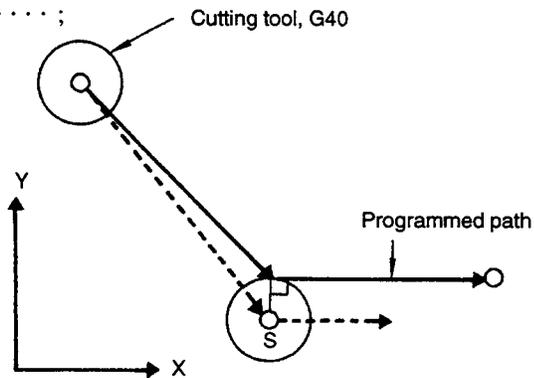


FIGURE 3.4.4.4 Offset Start-up (Straight-line to Straight-line (1))

- Straight-line to Straight-line at outside corner (270° to 360°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
X ..... ;
```

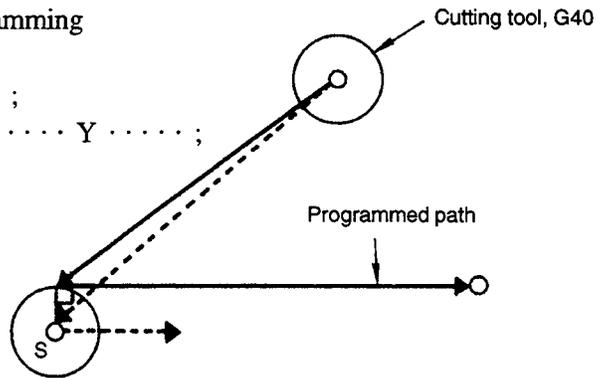


FIGURE 3.4.4.5 Offset Start-up (Straight-line to Straight-line (2))

- Straight-line to arc at outside corner (180° to 270°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
G02 X ..... Y ..... J ..... ;
```

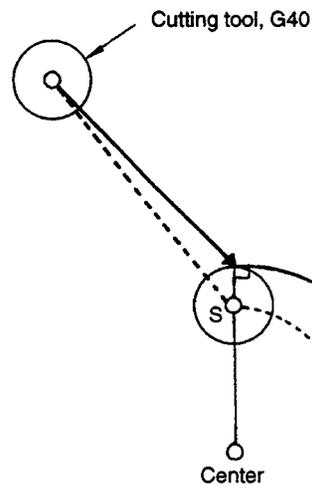


FIGURE 3.4.4.6 Offset Start-up (Straight-line to Arc (1))

- Straight-line to straight-line at outside corner (180° to 270°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
G02 X ..... Y ..... J ..... ;
```

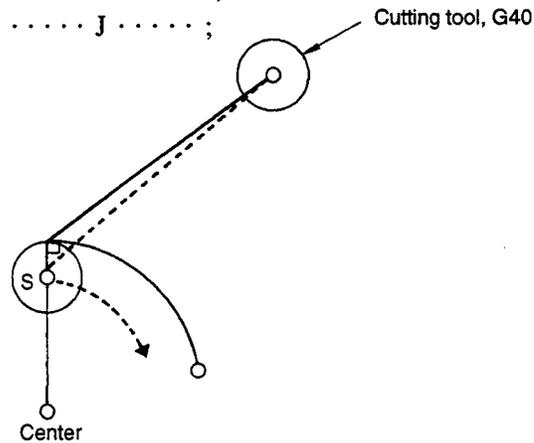


FIGURE 3.4.4.7 Offset Start-up (Straight-line to Arc (2))

- Type B: pm4013 D0 = 0

The cutting tool moves to the position obtained by the calculation for point of intersection using the commands in the start-up block and the next block.

- Straight-line to straight-line at outside corner (180° to 270°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
X ..... ;
```

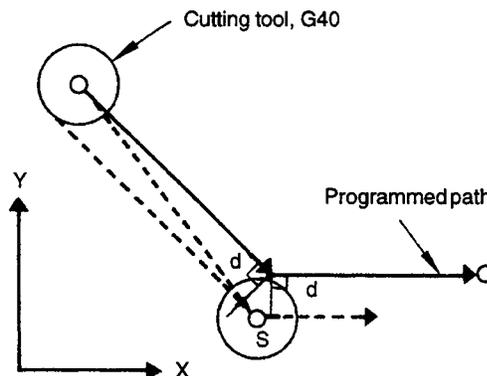


FIGURE 3.4.4.8 Offset Start-up (Straight-line to Straight-line (1))

- Straight-line to straight-line at outside corner (270° to 360°);

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
X ..... ;
```

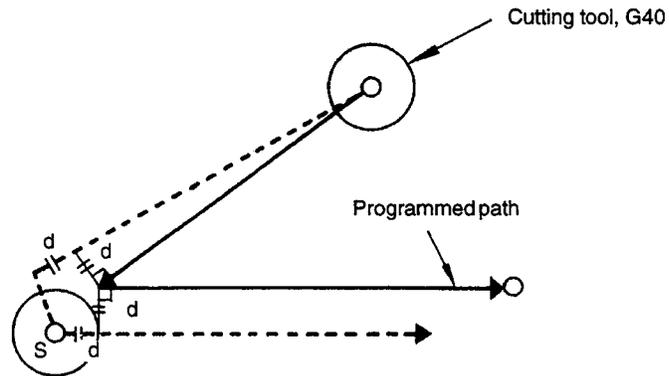


FIGURE 3.4.4.9 Offset Start-up (Straight-line to Straight-line (2))

- Straight-line to arc at outside corner (180° to 270°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
G02 X ..... Y ..... J ..... ;
```

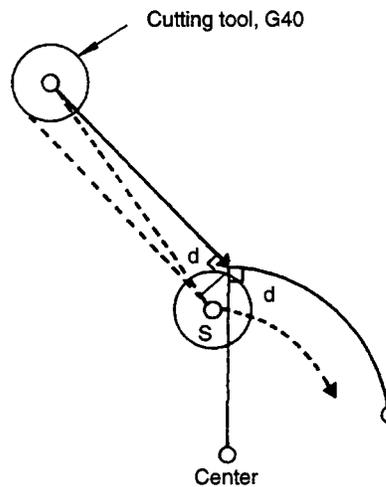


FIGURE 3.4.4.10 Offset Start-up (Straight-line to Arc (1))

- Straight-line to arc at outside corner (270° to 360°)

Example of Programming

```
G17 G01 F ..... ;
G42 D ..... X ..... Y ..... ;
G02 X ..... Y ..... J ..... ;
```

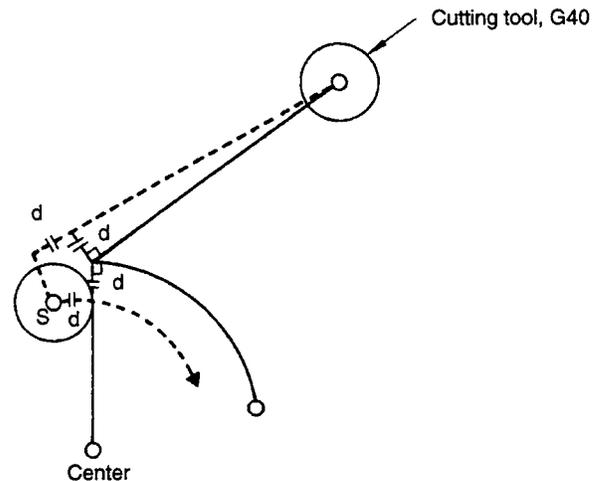


FIGURE 3.4.4.11 Offset Start-up (Straight-line to Arc (2))

(4) Axis Movements in the Offset Mode

Once the tool radius offset mode is set by the execution of G41 or G42, the cutting tool moves along the paths offset from the programmed paths until the tool radius offset mode is canceled by G40. Since the offset paths are automatically generated by the NC, the program should simply define the shape to be machined. The tool paths are controlled according to the angle made between the specified programmed paths.

(a) Inside corner (smaller than 180°)

The cutting tool moves to the position obtained by the calculate on for the point of intersection.

- Straight-line to straight-line

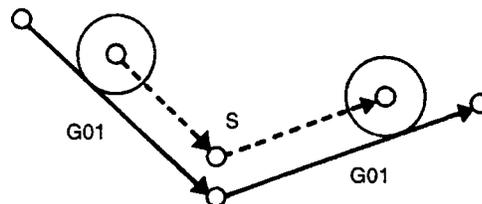


FIGURE 3.4.4.12 Straight-line to Straight-line

- Straight-line to Arc

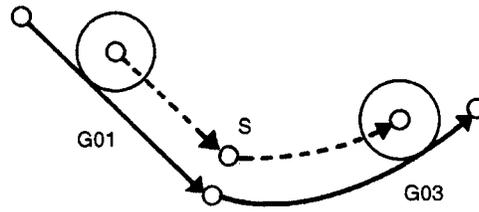


FIGURE 3.4.4.13 Straight-line to Arc

- Arc to arc

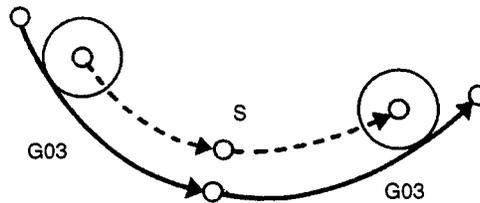


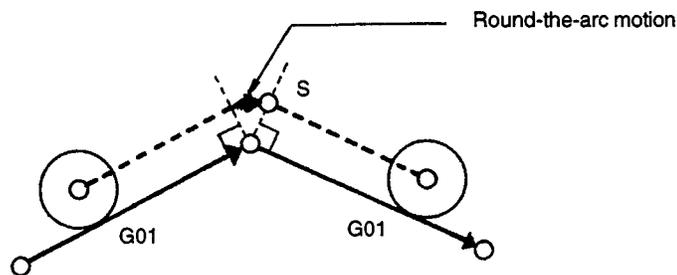
FIGURE 3.4.4.14 Arc to Arc

- (b) Outside corner (larger than 180°)

For this offset, two types of offset modes are provided and the offset mode to be used can be selected by the designation of an M code.

M96	Tool radius offset round-the-arc ON
M97	Tool radius offset round-the-arc OFF (calculation is executed to obtain the point of intersection)

- Tool movements in the M96 (tool radius offset round-the-arc ON) mode
 - Straight-line to straight-line



Note: In this case, round-the-arc motion of a cutting tool is included in the preceding block.

FIGURE 3.4.4.15 Round-the-arc Motion (Straight-line to Straight-line)

- Straight-line to arc

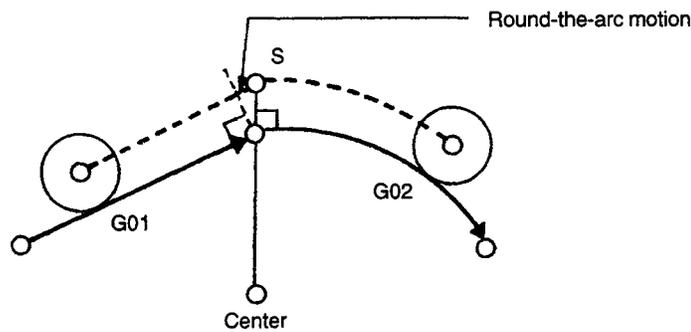


FIGURE 3.4.4.16 Round-the-arc Motion (Straight-line to Arc)

- Arc to arc

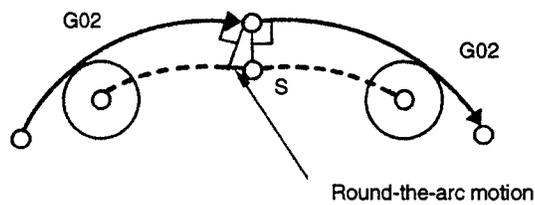


FIGURE 3.4.4.17 Round-the-arc Motion (Arc to Arc)

- Tool movements in the M97 (tool radius offset round-the-arc OFF) mode
 - Straight-line to straight-line at outside corner (180° to 270°)

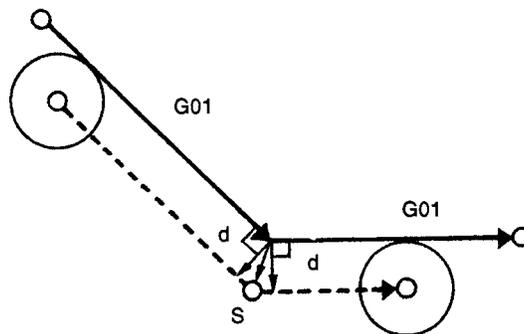


FIGURE 3.4.4.18 Offset Motion (Straight-line to Straight-line)

- Straight-line to arc at outside corner (180° to 270°)

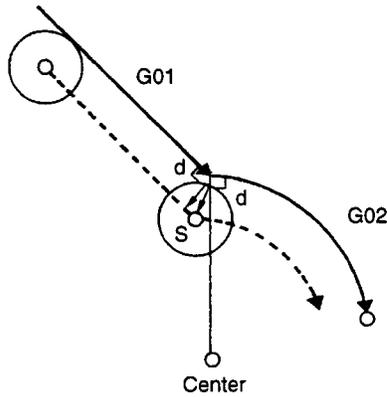


FIGURE 3.4.4.19 Offset Motion (Straight-line to Arc)

- Arc to arc at outside corner (180° to 270°)

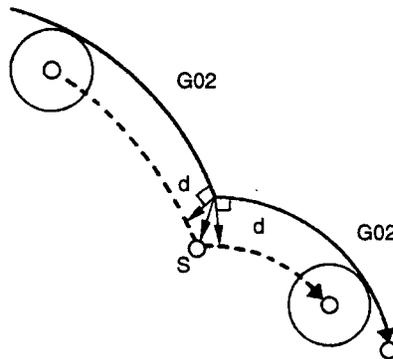


FIGURE 3.4.4.20 Offset Motion (Arc to Arc)

- Straight-line to straight-line at outside corner (270° to 360°)

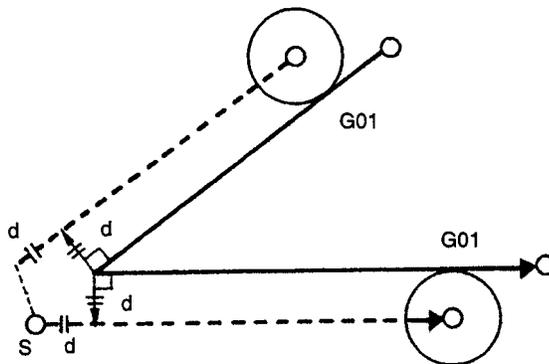


FIGURE 3.4.4.21 Offset Motion (Straight-line to Straight-line)

- Straight-line to arc at outside corner (270° to 360°)

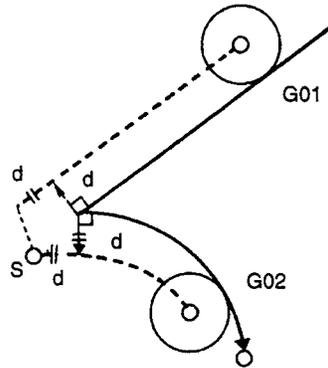


FIGURE 3.4.4.22 Offset Motion (Straight-line to Arc)

- Arc to arc at outside corner (270° to 360°)

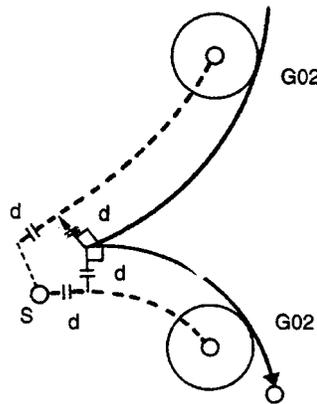


FIGURE 3.4.4.23 Offset Motion (Arc to Arc)

(c) Offset motion in the G00 mode

In the offset motion specified in the G00 block, the axes move to the offset position independently at their individual rapid traverse rates. Therefore, attention must be paid to the tool path so that interference will not be caused during this movement.

- Offset motion at inside corner (smaller than 180°)

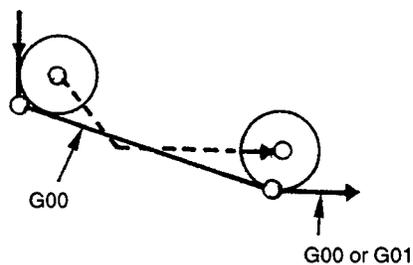


FIGURE 3.4.4.24 Offset Motion at Inside Corner

- Offset motion at outside corner (180° to 360°)

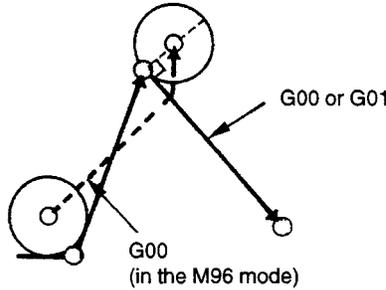


FIGURE 3.4.4.25 Offset Motion at Outside Corner

(5) Blocks not Including Axis Move Commands in the Offset Mode

In the tool radius offset mode, the NC generates the tool paths by buffering the data of two blocks. If a block not including axis move commands is read, the NC reads one more block to generate the offset tool paths. Designation of such a block which does not include axis move commands is allowed in the tool radius offset mode for up to two consecutive blocks.

After the designation of G41, there must not be three or more consecutive blocks that do not include the movement commands of the axes in the offset plane.

(a) Consecutive three or more blocks not including axis move commands

If three or more blocks not containing axis move commands in the offset plane are given consecutively, the cutting tool is moved to the position offset normally by the specified offset amount at the end point of the block immediately preceding such blocks.

Example of Programming

```
G17 G01 G41 X ..... Y ..... D ..... F ..... ;
X ..... Y ..... ;
.
.
X ..... Y ..... ;
G04 P1000 ;
X ..... Y ..... ;
.
.
X ..... Y ..... ;
Z ..... ;
Z ..... ;
X ..... Y ..... ;
.
.
X ..... Y ..... ;
G40 X ..... Y ..... ;
```

Blocks not including axis movement commands in the offset plane (If such blocks continue up to two blocks, the NC can generate tool paths without a problem.)

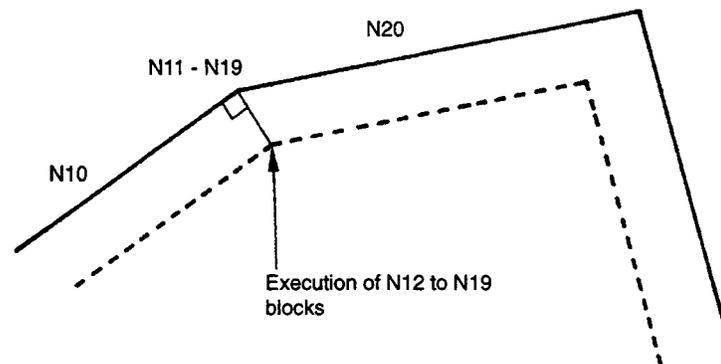


FIGURE 3.4.4.26 Offset Motion if Three or More Blocks not Including Axis Move Commands are Specified in the Offset Mode

(b) Insertion of dummy block

If there are no axis move commands in three consecutive blocks, the cutting tool is positioned on the normal end point of the block immediately preceding such blocks. If it is impossible to specify move commands of the axes in the offset plane due to the retraction motion of the third axis or other reasons and if normal positioning is not desirable, a dummy block that includes I, J, or K can be inserted in the program. The dummy block does not call up actual axis movements, but it only gives the data necessary for the calculation to generate the offset tool paths. In the example program given below, a dummy block specifying the same movements as given in the block (N020) where the axis movement restarts in the XY plane after the Z-axis movement is inserted in a program; addresses I and J are used in the dummy block.

Example of Programming

```

N001 G17 G01 G41 X ..... Y ..... D ..... F ..... ;
N002 X ..... Y ..... ;
.
.
.
N010 X ..... Y ..... ;
N011 I ..... J ..... ;
N012 Z ..... ;
.
.
.
N019 Z ..... ;
N020 X ..... Y ..... ;
.
.
.
N029 X ..... Y ..... ;
N030 G40 X ..... Y ..... ;

```

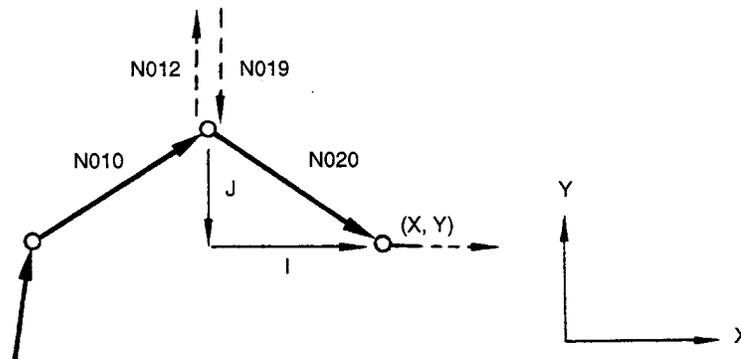


FIGURE 3.4.4.27 Insertion of Dummy Blocks

- In a dummy block, addresses I, J, and K are used corresponding to X-, Y-, and Z-axis. Specify these addresses meeting the plane which has been selected as the offset plane. Note that in dummy blocks, commands should be given in incremental commands. With the example program indicated above, if “X Y in” N20 are specified in absolute values, change them to equivalent incremental values.
- If the object of the dummy block is circular interpolation, enter the dummy block as shown in the example program given below. Insert the dummy block in which the straight line expressing the tangential direction at the start point of circular interpolation is specified as shown in Fig. 3.4.4.28. The cutting tool moves to point A as shown in Fig. 3.4.4.29 by the execution of the dummy block so that the following circular interpolation can be executed.

Example of Programming

```

N050 G01 X ..... Y ..... ;
N051 G01 I (b) J (-a) ; ← Dummy block
N052 Z ..... ;
.
.
.
N059 Z ..... ;
N060 G03 X ..... Y ..... I (a) J (b) ; ← Circular interpolation
N061 G01 X ..... Y ..... ;

```

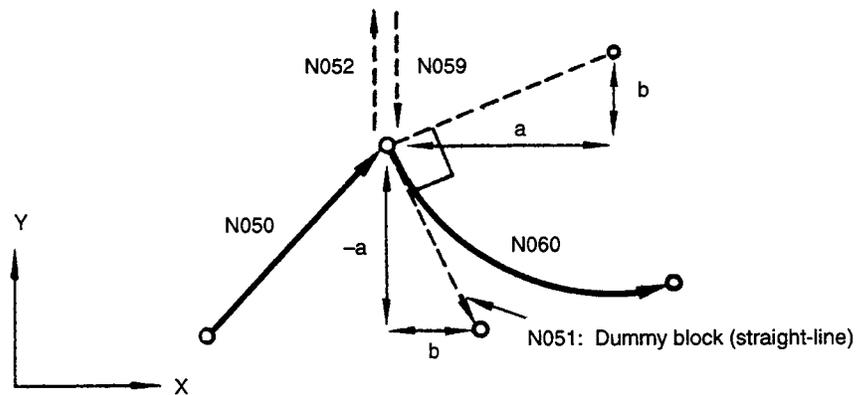


FIGURE 3.4.4.28 Insertion of Dummy Block

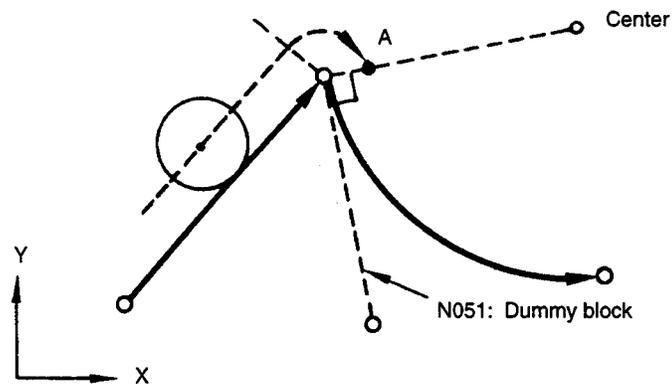


FIGURE 3.4.4.29 Movement to Point A by Execution of Dummy Block

- If I, J, or K is specified when canceling the offset mode, the offset position is corrected from point ① to point ② according to the direction specified by these addresses.

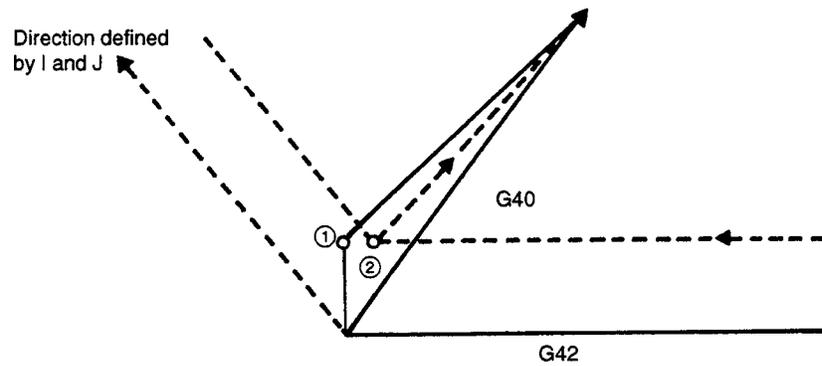


FIGURE 3.4.4.30 Correction of Offset Position

(6) Switching the G41 and G42 in the Offset Mode

The direction of offset (left side and right side) can be directly switched without canceling the offset mode. There are two kinds of G41/G42 switching methods (types A and B) and the method to be used can be selected by the setting for a parameter.

pm4013 D1 = 1	Type A
pm4013 D1 = 0	Type B

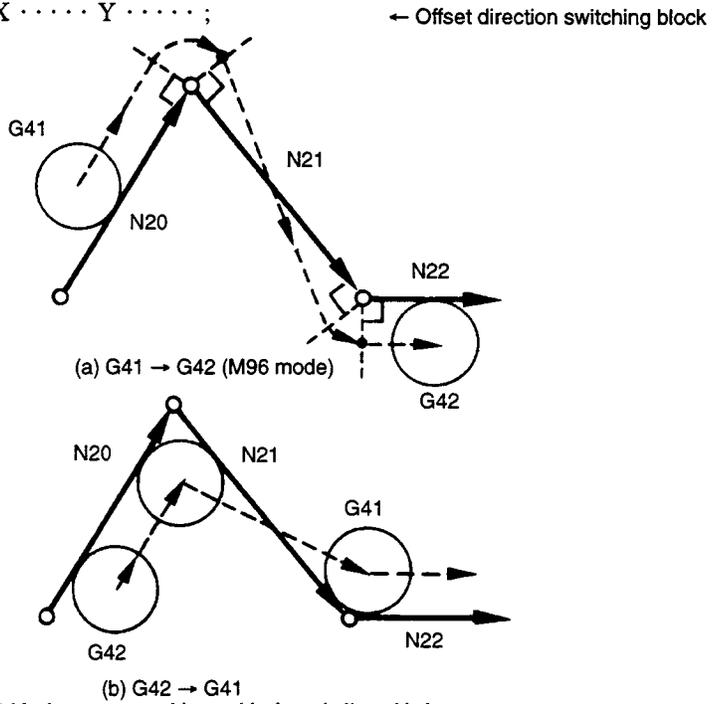
(a) Type A: pm4013 D1 = 1

The offset direction is switched at the start and end of the block in which the switching of the offset direction is specified.

Example of Programming

```

N10 G17 G01 F ..... ;
N11 G41 (G42) D ..... ;
.
.
.
N20 G01 X ..... Y ..... F ..... ;
N21 G42 (G41) X ..... Y ..... ;
N22 X ..... ;
    
```



Note: If the contents of N21 block are expressed in two blocks as indicated below
 G42 (or G41);
 X Y
 the offset direction is switched in the same manner.

FIGURE 3.4.4.31 Switching the Offset Direction at the Start and End of the Block

(b) Type B: pm4013D1 = 0

Direction of offset is switched at the point of intersection of the offset tool paths.

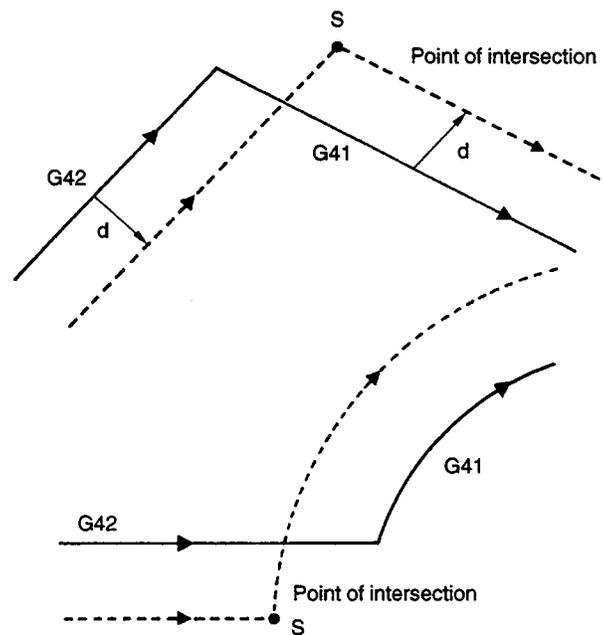


FIGURE 3.4.4.32 Switching of the Offset Direction at Point of Intersection of Offset

If there is no point of intersection, the offset direction is switched according to type A.

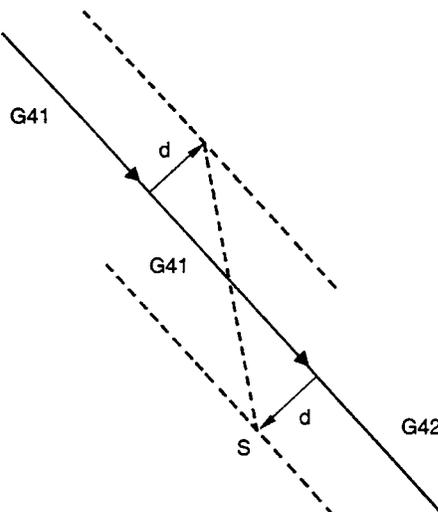


FIGURE 3.4.4.33 Switching of the Offset Direction when There Intersection is No Point of Intersection

(7) Changing the Tool Offset Amount in the Offset Mode

There are two kinds of offset amount changing methods (types A and B), and the method to be used can be selected by the setting for a parameter.

pm4013 D2 = 0	Type A
pm4013 D2 = 1	Type B

(a) Type A: pm4013 D2 = 0

When a new D code is specified, the new offset data are calculated from the axis move commands given in the block including the new D code and the next block.

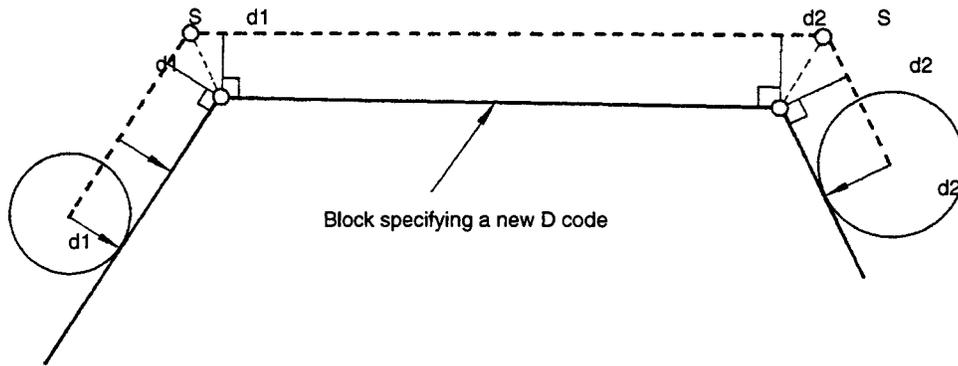


FIGURE 3.4.4.34 Calculating the New Offset Data from the Axis Move Commands in the New D Code Specifying Block and the Next Block

(b) Type B: pm4013 D2 = 1

When a new D code is specified, the new offset data are calculated from the axis move commands given in the block including the new D code and the preceding block.

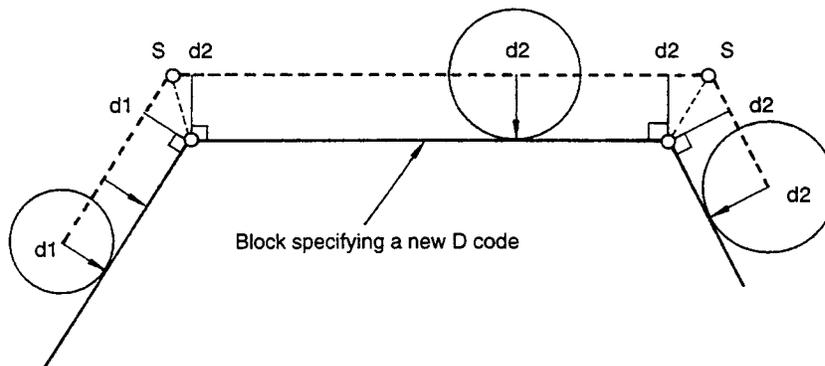


FIGURE 3.4.4.35 Calculating the New Offset Data from the Axis Move Commands in the New D Code Specifying Block and the Previous Block

(8) Canceling the Offset Mode

There are two types of offset mode cancellation methods (types A and B), and the method to be used can be selected by the setting for a parameter.

pm4012 D6 = 0	Type A
pm4012 D6 = 1	Type B

- Type A: pm4012 116 = 0

Offset mode cancellation movement is not executed in the G40 block if no axis move commands are given. The offset mode is canceled by the first axis move command given in a block specified following the G40 block. Axis move commands should be specified with G40 in the same block.

- Type B: pm4012 D6 = 1

Offset mode cancellation movement is executed in the G40 block even if there are no axis move commands given. The cutting tool moves normally to the offset position at the end point of the block immediately before the G40 block. Since G40 calls up (offset mode cancellation axis movements, it must be specified in the G00 or G01 mode. If it is specified in a mode other than G00 or G01, alarm “0181” occurs.

(a) Canceling the offset mode at inside corner (smaller than 180°)

- Straight-line to straight-line

Example of Programming

```
G41
.
.
.
G01 X ..... F ..... ;
G40 X ..... Y ..... ;
```

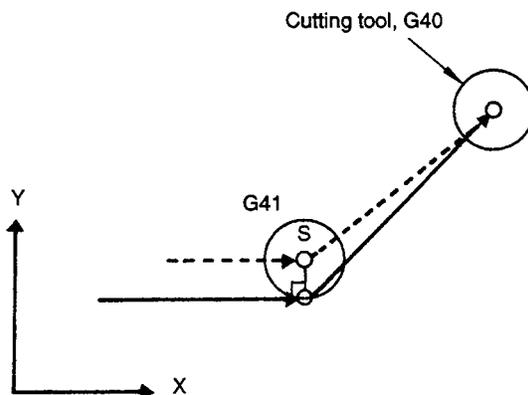


FIGURE 3.4.4.36 Canceling the Offset Mode at Inside Corner (Straight-line to Straight-line)

- Arc to straight-line

Example of Programming

```
G41  
.  
.  
.  
G02 X ..... Y ..... I ..... J ..... ;  
G01 G40 ..... X ..... Y ..... ;
```

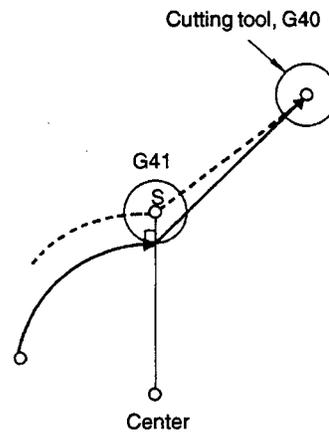


FIGURE 3.4.4.37 Canceling the Offset Mode at Inside Corner (Arc to Straight-line)

(b) Canceling the offset mode at outside corner (larger than 180°)

There are two types (types A and B) of offset mode cancellation axis movement patterns, and the pattern to be used can be selected by the setting for a parameter. For this selection, the same parameter as used to select the start-up mode is used.

pm4013 D0 = 1	Type A
pm4013 D0 = 0	Type B

- Type A: pm4013 D0 = 1

The cutting tool is moved normally to the offset position at the end point of the block immediately before the offset mode cancellation block and then to the end point specified in the program.

- Straight-line to straight-line at outside corner (180° to 270°)

Example of Programming

```
(G42)
.
.
.
G01 X ..... F ..... ;
G40 X ..... Y ..... ;
```

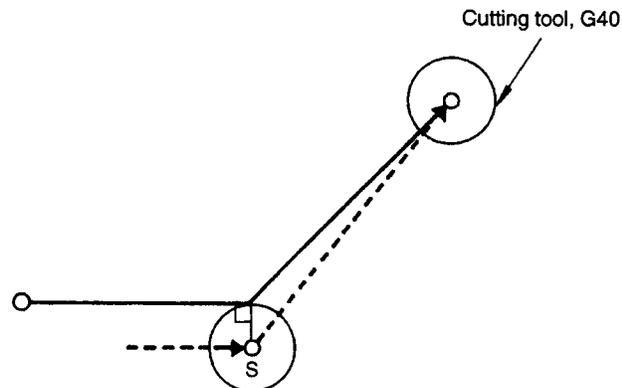


FIGURE 3.4.4.38 Straight-line to Straight-line at Outside Corner

- Straight-line to straight-line at outside corner (270° to 360°)

Example of Programming

```
(G42)
.
.
G01 X ..... F ..... ;
G40 X ..... Y ..... ;
```

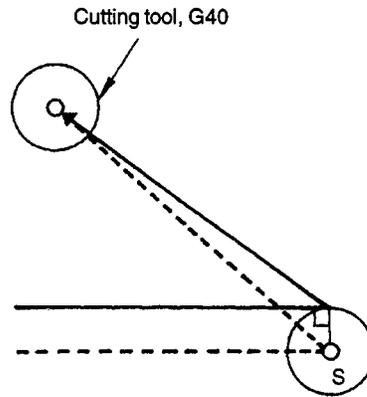


FIGURE 3.4.4.39 Straight-line to Straight-line at Outside Corner

- From arc to straight-line at outside corner (180° to 270°)

Example of Programming

```
G42
.
.
G02 X ..... Y ..... I ..... J ..... ;
G01 G40 ..... X ..... Y ..... ;
```

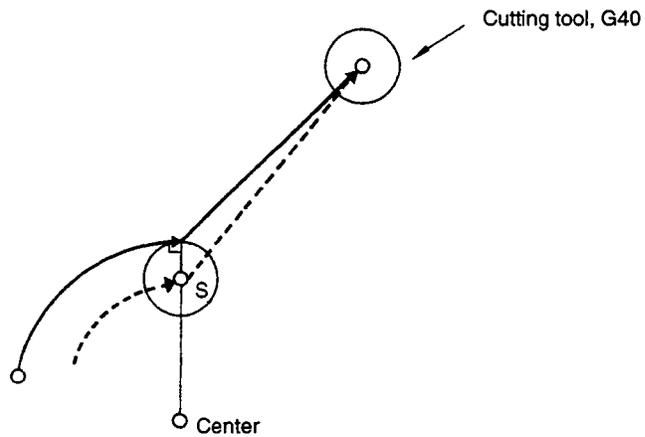


FIGURE 3.4.4.40 From Arc to Straight-line at Outside Corner

- Arc to straight-line at outside corner (270° to 360°)

Example of Programming

G42

.

.

G02 X Y I J ;

G01 G40 X Y ;

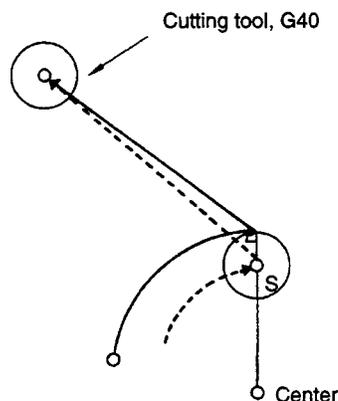


FIGURE 3.4.4.41 Arc to Straight-line at Outside Corner

- Type B: pm4013 D = 0

The cutting tool moves to the point obtained by the calculation of the point of intersection using the axis move commands in the offset mode cancel block and those in the block immediately before this block, and then to the point specified in the program.

- Straight-line to straight-line at outside corner (180° to 270°)

Example of Programming

G42

.

.

G01 X F ;

G40 X Y ;

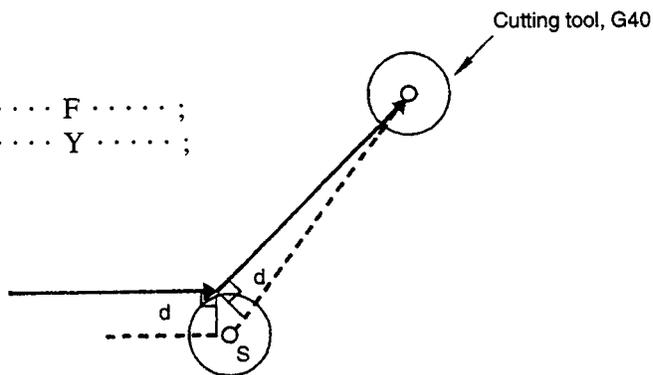


FIGURE 3.4.4.42 Straight-line to Straight-line at Outside Corner

- Straight-line to straight-line at outside corner (270° to 360°)

Example of Programming

```
G42
.
.
G01 X ..... F ..... ;
G40 X ..... Y ..... ;
```

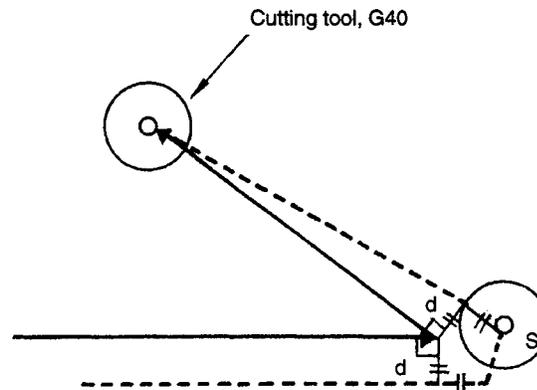


FIGURE 3.4.4.43 Straight-line to Straight-line at Outside Corner

- Arc to straight-line at outside corner (180° to 270°)

Example of Programming

```
G42
.
.
G02 X ..... Y ..... I ..... J ..... ;
G01 G40 ..... X ..... Y ..... ;
```

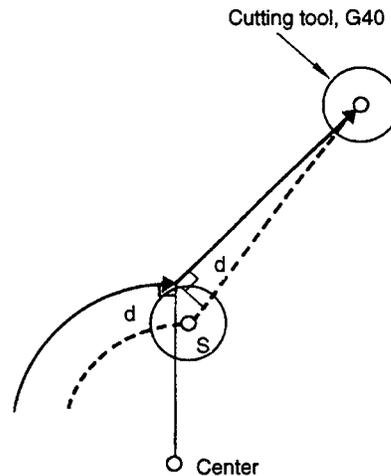


FIGURE 3.4.4.44 Arc to Straight-line at Outside Corner

- Arc to straight-line at outside corner (270° to 360°)

Example of Programming

G42

.

.

.

G02 X Y I J ;

G01 G40 X Y ;

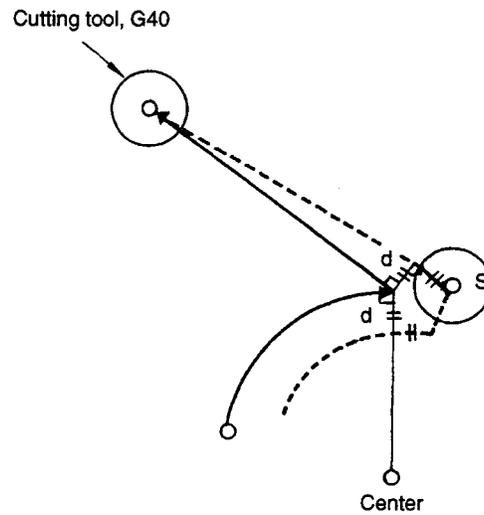


FIGURE 3.4.4.45 Arc to Straight-line at Outside Corner

(9) Interference Check

The interference check function prevents the cutting tool from cutting into or interfering with the workpiece. However, this check is not made at the start-up of the tool radius offset mode. The process to be taken in case of interference is detected from the blocks read into the buffer memory can be selected by the setting for a parameter as indicated below.

Type A	pm4013 D4 = 1	Alarm occurs and operation stops.
Type B	pm4013 D4 = 0	Tool paths are corrected.

Whether or not the interference check is executed is also determined by the setting for a parameter.

pm4013 D3 = 0	Interface check is not made.
pm4013 D3 = 1	Interface check is made.

The illustration in Fig. 3.4.4.46 shows how the interference check function operates. In reference to the programmed paths, offset tool paths are generated according to the set tool radius offset amount. With tool radius offset amount D_x , tool paths $f_1 \textcircled{1} \rightarrow f_2 \textcircled{1} \rightarrow f_3 \textcircled{1} \rightarrow f_4 \textcircled{1} \rightarrow f_5 \textcircled{1} \rightarrow f_6$ is generated and with D_y tool paths $f_1' \rightarrow f_2' \rightarrow f_3' \rightarrow f_4' \rightarrow f_5' \rightarrow f_6'$ are generated. However, in the tool paths generated with the offset amount of D_y , path $f_3' \rightarrow f_4'$ shows 180° reversed movement from the correct programmed path direction $f_3 \textcircled{1} \rightarrow f_4$. The function assumes this interference and generates an alarm.

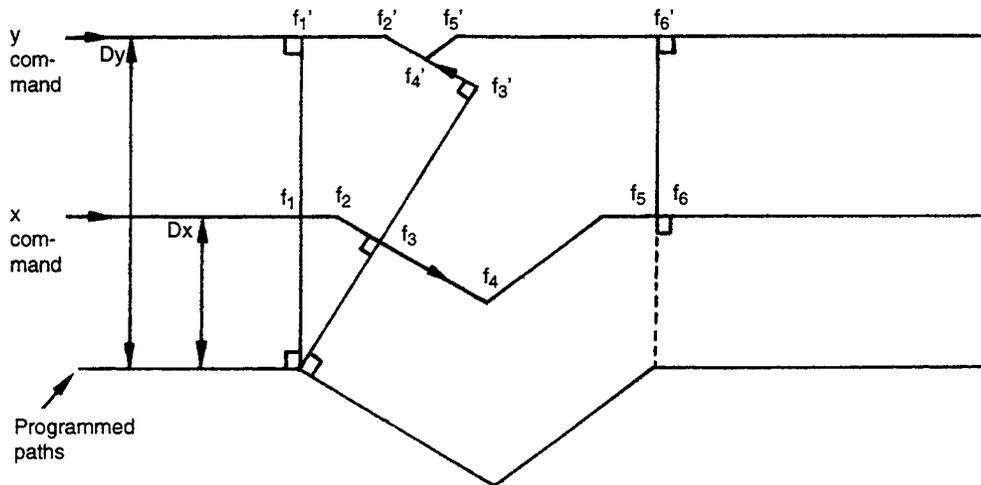


FIGURE 3.4.4.46 Definition of Interference

(a) Type A: pm4013 D4 = 1 Generation of alarm

The following programs give examples in which the function determines that interference (overcuts) will occur due to considerable differences between the programmed paths and the tool paths generated after offset.

- Example program 1

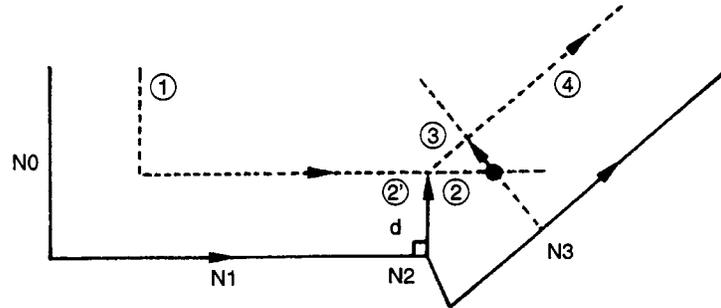
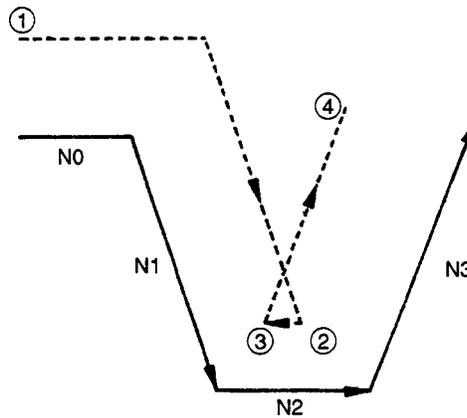


FIGURE 3.4.4.47 Example Program 1

- Example program 2



Note: Since the cutting tool cuts into the workpiece excessively at the end point ② of block N1, alarm "0187" occurs.

The operation stops when the cutting tool reaches the end point of block N0.

FIGURE 3.4.4.48 Example Program 2

(b) Type B: pm4013 D4 = 0 Correcting the tool paths

If the function detects possible interference after the calculation of the offset tool paths, the function clears the tool paths that might cause interference and generates the paths that are free of interference.

- Generating interference-free paths for straight-line to straight-line motion

For the programmed paths as shown in Fig. 3.4.4.49, three points f_1 , f_2 , and f_3 are generated at the joint of blocks N1 and N2 according to the tool radius offset function.

Point f_4 is also generated at the joint of N2 and N3. Interference check is made using these four points f_1 to f_4 and the points causing interference are erased one by one until the tool paths that are free of interference are generated.

Checked for $f_3 - f_4$: Erasing f_3 since interference occurs.

Checked for $f_2 - f_4$: Erasing f_2 since interference occurs.

Checked for $f_1 - f_4$: No interference Tool paths are generated as ① → f_1 → f_4 → ②.

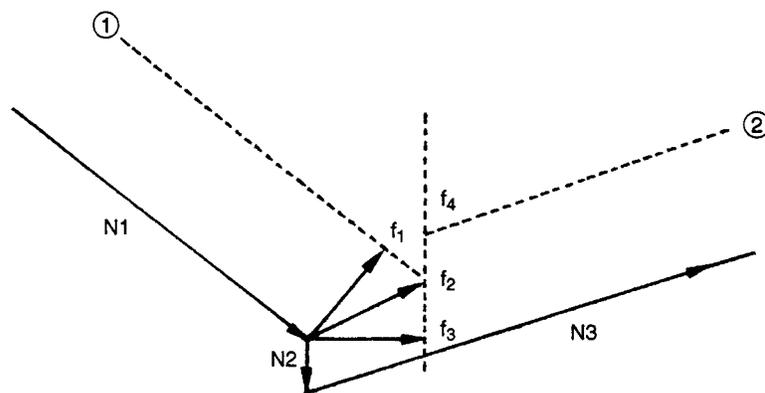


FIGURE 3.4.4.49 Generating Tool Paths without Interference (Straight-line to Straight-line)

- Generating interference-free paths for arc to arc motion

For the programmed paths as shown in Fig. 3.4.4.50, four points f_1 , f_2 , f_3 , and f_4 are generated at the joint of N 1 and N2 according to the tool radius offset function. At the joint of N2 and N3, another four points f_5 to f_8 are generated. Interference check is made using these eight points f_1 to f_8 and the points causing interference are erased one by one until the tool paths that are free of interference are generated.

Checked for $f_4 - f_5$: f_4 and f_5 are erased since interference occurs.

Checked for $f_3 - f_6$: f_3 and f_6 are erased since interference occurs.

Checked for $f_2 - f_7$: No interference

Tool paths are generated as $f_1 \rightarrow f_2 \rightarrow f_7 \rightarrow f_8$.

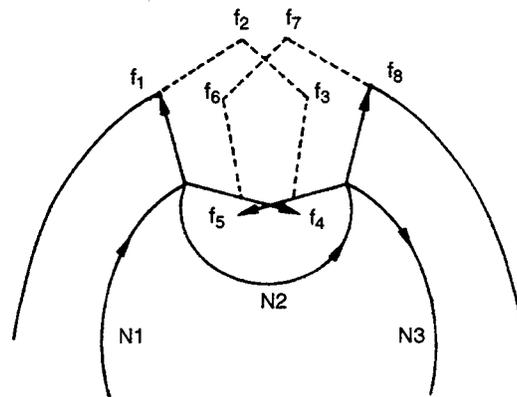


FIGURE 3.4.4.50 Generating Interference-free Paths for Arc to Arc Motion

- Example where interference-free tool paths cannot be generated

For the programmed paths as shown in Fig. 3.4.4.51, three points f_1 , f_2 , and f_3 are generated at the joint of N 1 and N2 according to the tool radius offset function. At the joint of N2 and N3, another three points f_4 to f_6 are generated. Interference check is made using these six points f_1 to f_6 and the points causing interference are erased one by one until the tool paths that are free of interference are generated.

Checked for $f_3 - f_4$: f_3 and f_4 are erased since interference occurs.

Checked for $f_2 - f_5$: f_2 and f_5 are erased since interference occurs.

Checked for $f_1 - f_6$: f_1 and f_6 are erased since interference occurs.

Occurrence of alarm (“0188”): Operation stops when the cutting tool is positioned at the start point of N1 block.

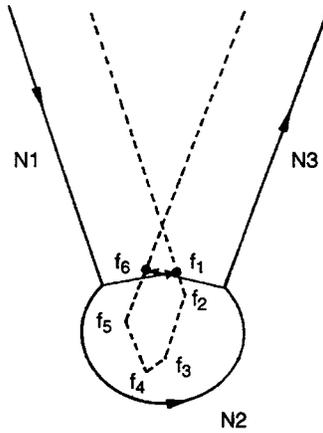


FIGURE 3.4.4.51 Example where Interference-free Tool Paths cannot be Generated

(10)Supplements to the Tool Radius Offset Commands

- In tool radius offset C mode, the maximum programmable values specified in Table 3.4.4.3 also apply.
- Alarm “0184” occurs if the following shapes are specified.
 - Circular arc: Radius (r) of arc + 5 £ Tool radius (d)

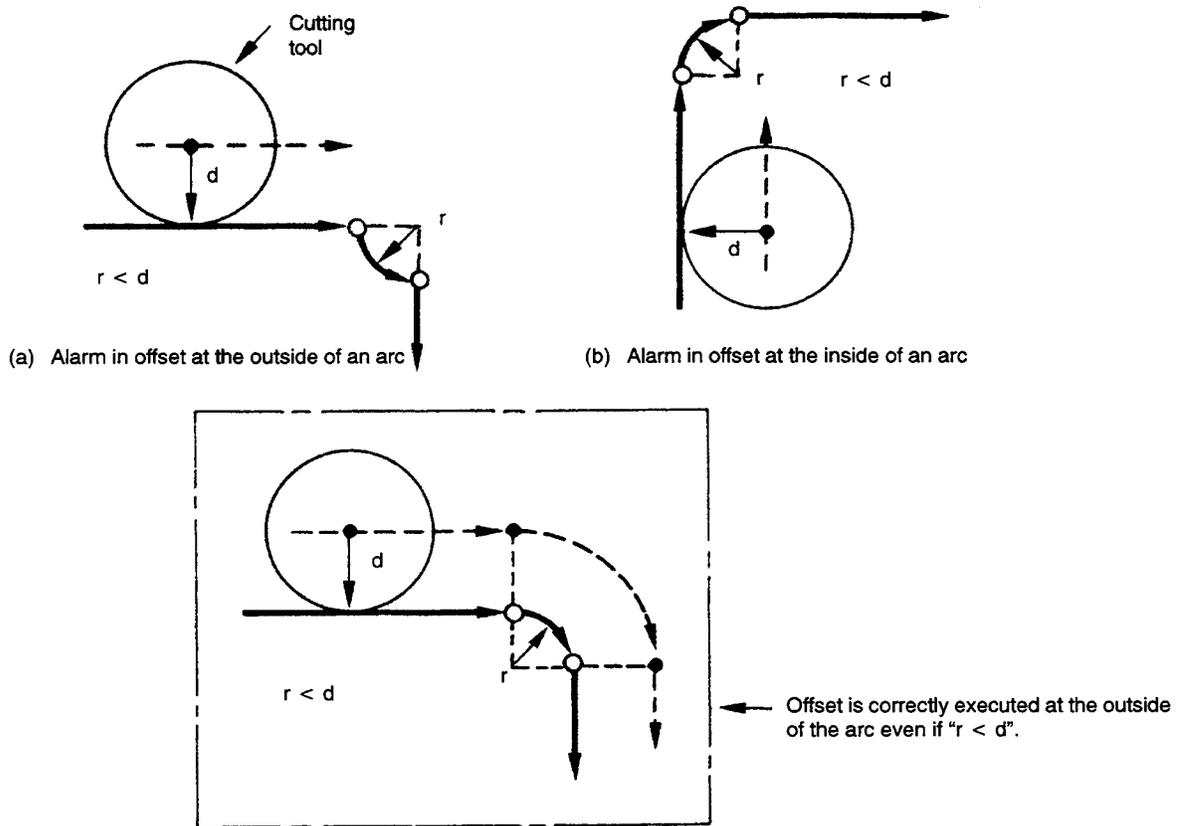


FIGURE 3.4.4.52 Programmed Paths Causing an Alarm

- Point of intersection does not lie on the tool paths

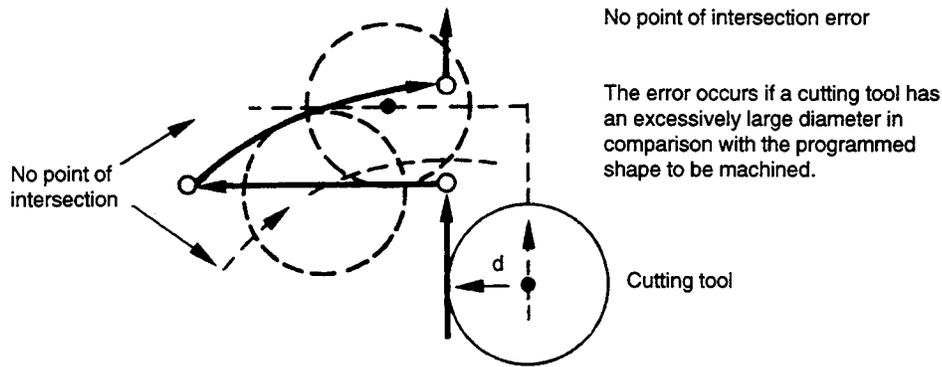


FIGURE 3.4.4.53 Programmed Paths Causing an Alarm C

- If any of the following G codes specified in the offset alarm “0182” occurs.
If the NC is reset in the offset mode, the offset mode is canceled and the G40 mode is set.

Table 3.4.4.3 Prohibited G Codes

	Prohibited G Codes
G codes that cause an alarm	G12, G13 (G17 to G19) G28, G29 G73, G74, G76, G77, G81 to G89 G92

1
1

- The tool radius offset C function is applied to the tool paths offset by the tool length offset and the tool position offset function. The tool radius offset function should not be applied to the tool paths for which the tool position offset function is used as the tool radius offset operation. If such offset is applied, it could complicate the programming.
- When specifying G41, G42, or G40, G00 or G01 and an F code must be specified with G41, G42, or G40 in the same block or a block that precedes the G40/G41/G42 block.
- If a plane selection G code (G17 to G19) is specified in the offset mode, alarm “0183” occurs.
- If the circular interpolation function is specified in a plane other than the offset plane specified by G17, G 18, or G19 while in the offset mode, alarm “0183” occurs.

- Circular cut (G12, G13) and a canned cycle (G73, G74, G76, G77, G80 to G89) must be called up in the tool radius offset cancel mode. The circle cut function incorporates the tool radius offset function. If circle cut or a canned cycle is called up in the offset mode, alarm “0182” occurs.
- The tool radius offset C function is valid for the circular interpolation commands which include a radius command.
- Subprogram call (M98, M99) commands can be specified in the offset mode.
- If simultaneous 3-axis (* 5-axis) movement is specified, offset is applied to the shape projected to the offset plane specified by G17, G 18, or G 19.

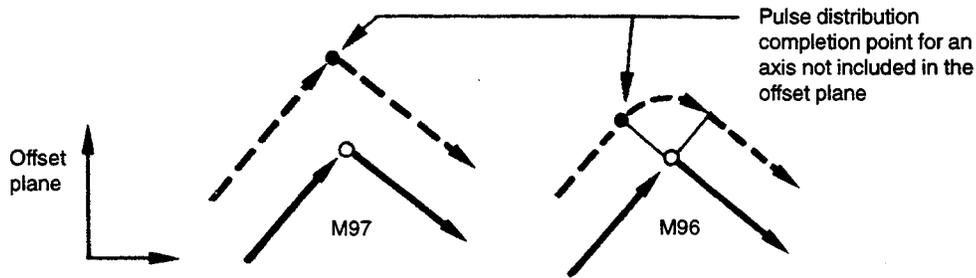


FIGURE 3.4.4.54 Simultaneous Three Axes Movement Specified in the Offset Mode

- It is possible to correct the offset position temporarily by specifying a dummy block including addresses I, J, or K.

Example of Programming

```
(G42)
.
.
.
N100 G01 X ..... Y .....;
N101 I ..... J .....;
N102 X .....;
```

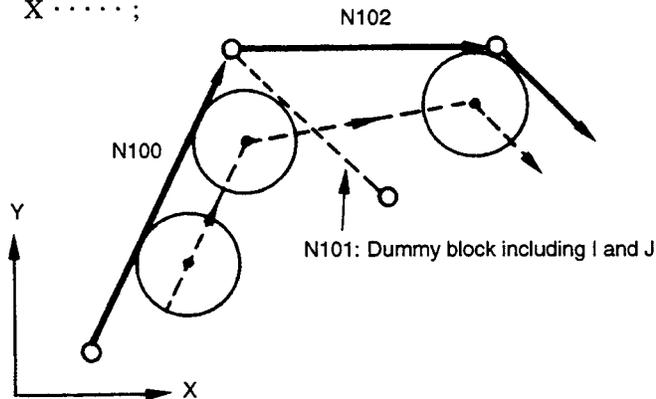


FIGURE 3.4.4.55 Correcting the Offset Position Temporarily

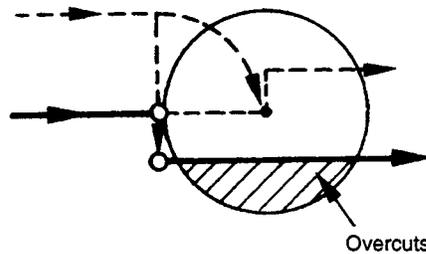
- If the M00, M01 (or M02, M30) command is specified in the offset mode, correct offset processing is interrupted since such M codes prohibit buffering of blocks. To avoid this, insert a dummy block including I, J, or K immediately before the block that includes M00 or M01. This allows continuous correct offset processing.

Example of Programming

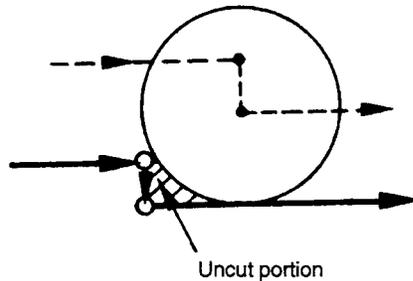
```
(G41)
N200 G01 X ..... Y ..... ;
N201 I ..... J ..... ;
N202 M00 ;
N203 X ..... Y ..... ;
```

Specify the axis move commands in N203 block using I, J, and K.

- The tool offset data memory capacity varies depending on whether the option is selected or not. The maximum value that can be set as the tool radius offset value is ±999.999 mm (or ±99.9999 inch).
- If compensation is made in the M96 mode for a step which is smaller than the tool radius, overcuts occur. If M97 mode is selected for the same operation, an uncut portion is left. However, in actual cutting, it is recommended to select the M97 mode.



(a) M96 mode



(b) M97 mode

FIGURE 3.4.4.56 Offsetting at Step Smaller than Tool Radius

- Even in the M96 mode, if the tangential angle made between the axis movements specified in the program is smaller than 30° or if g is smaller than a certain value as shown in Fig. 3.4.4.57, round-the-arc motion is not made, but the cutting tool directly moves to point B. Value of g is set for parameter pm4450. Note that since the value set for pm4450 is also used as the minimum distance for the execution of circular interpolation, the setting for this parameter has influence on normal circular interpolation.

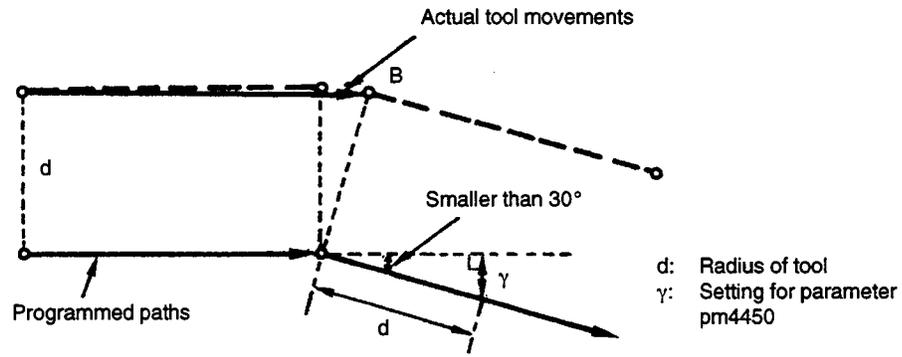


FIGURE 3.4.4.57 Tool Movements with Small Tangential Angle of Programmed Paths

- If G39 is specified in the tool radius offset C mode (G41/G42), an arc can be inserted at comers.

G39 is a non-modal G code. G39 must be specified in a block without other commands. If other commands are specified with G39 in the same block, those specified after G38 are all disregarded.

If the commands are specified in the G39 block in the manner as “G39 I_ J_ S_ F_ ;”, I, J, S, and F codes are disregarded.

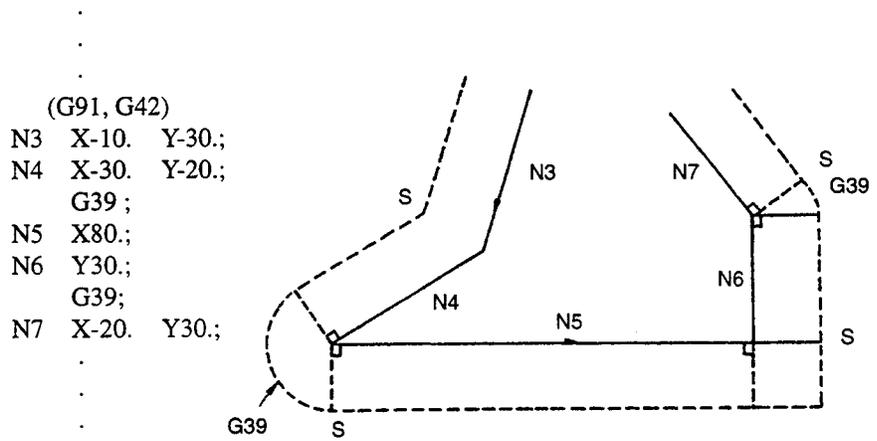


FIGURE 3.4.4.58 Program for the Around-the-arc Offset

- Upon completion of the execution of G39 block, the mode (M96/M97) selected previously is recovered.
- In the single-block mode, the cursor does not stop at the G39 block, but it stops at the next block.
- If interference is detected by the interference check function in the paths generated by G39, G39 is disregarded and the tool paths are automatically corrected.
- In the offset mode, intervention of MDI operation is not allowed. However, MDI operation with one line of program is possible.

3.4.5 3-Dimensional Tool Offset Function (G40, G41, G42)*

By specifying G41 and G42, it is possible to shift the tool paths in the 3D vector direction.

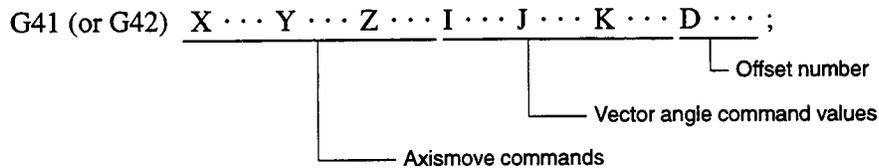
(1) Commands

(a) Features of G40, G41, and G42

Table 3.4.5.1 G Codes Used for the Tool Offset Function

G Code	Function	Group
G40	3-dimensional tool offset cancel	07
G41	3-dimensional tool offset, left	07
G42	3-dimensional tool offset, right	07

(b) Programming format



With the commands indicated above, the NC enters the 3-D tool offset mode and start-up processing begins. The plane used for the 3-D tool offset is fixed to the XYZ plane. Therefore, offset is not valid for additional axes. With G42, offset is made in the direction opposite to the offset motion called by G41.



1. The start-up block (G41, G42) must contain all of I, J, K, and D. If any one of POINT these addresses is not specified in the start-up block, the 2-D tool radius offset C is executed.
 - If a D command is not specified, alarm 0162, “LACK OF ADDRESS” occurs. Alarm 0184 “RADIUS OFS CAL ERROR” occurs if “0” is set for all of I, J, and K.

- If the offset amount specified “by D is “0”, the tool paths are generated as programmed.
2. It is not necessary to specify X, “Y, and Z at the start-up of the 3-D tool offset.
 3. The 3-D tool offset start-up must be made in either the G00 or the G01 mode.
 - G40 X ••• Y ••• Z •••;

└──────────┬──────────┘
Axis move commands

With the commands indicated above or “G01 (or G00) X ••• Y ••• Z ••• D00;”, the 3-D tool offset mode is canceled toward the end point specified in this block. If such a block does not contain any of these axes, the offset cancel motion is executed for such axis by the offset amount.



1. If D00 is specified to cancel the 3-D tool offset mode, the NC execute processing assuming that G40 is specified. To call up the 3-D tool offset mode after that, specify G41 or G42 again.
2. When the 3-D tool offset mode is canceled, I, J, and K values are cleared to “0”. Note that the command value specified by address D is not cleared and remains valid as a modal value for executing the program after that.
3. It is not necessary to specify X, Y, and Z when canceling the 3-D tool offset.
4. To cancel the 3-D tool offset mode, set the G00 or G01 mode.
5. Although an additional axis command such as A and B may be specified in the same block with the 3-D tool offset commands, the function is not valid for the additional axis.

(c) Description of the addresses

- X, Y, Z, I, J, and K: The same format as used when specifying these addresses as normal axis commands.

The number of digits are indicated in Table 3.4.5.2.

Table 3.4.5.2 Number of Command Digits

	Integer Section	Decimal Function
Metric	6	3
Inch	5	4

If a command not conforming to the specified format is given, it cause alarm 0104 “NUMERIC DATA OVERFLOW”. Values for I, J, and K are cleared to “0” when the power is turned ON, the NC is reset, or the 3-D tool offset is canceled.

- D: For the D command, the number of digits must meet the number of offset pairs. If the specified value exceeds the number of offset pairs, alarm 9150 “OFFSET NO. OVER” occurs. If a decimal point is included, the command value does not have a sign (+/-) or a command value other than a positive value is specified, alarm 0102 ““.” MISUSE” occurs.

(2) Operation

(a) Programmed paths

In the 3-D tool offset mode, the end point of each block is determined based on the vector commands I, J, and K specified in that block.

Example of Programming

```
G01 X5. Y10.
G42 X40. Y22. I-1. J5.5 K0. D1 (D1 = 2.0)
G01 X65. Y15. I-1.5 J5.5 K0
G01 X90. Y35.
G40 X110. Y10. (G01X110. Y10. D00)
```

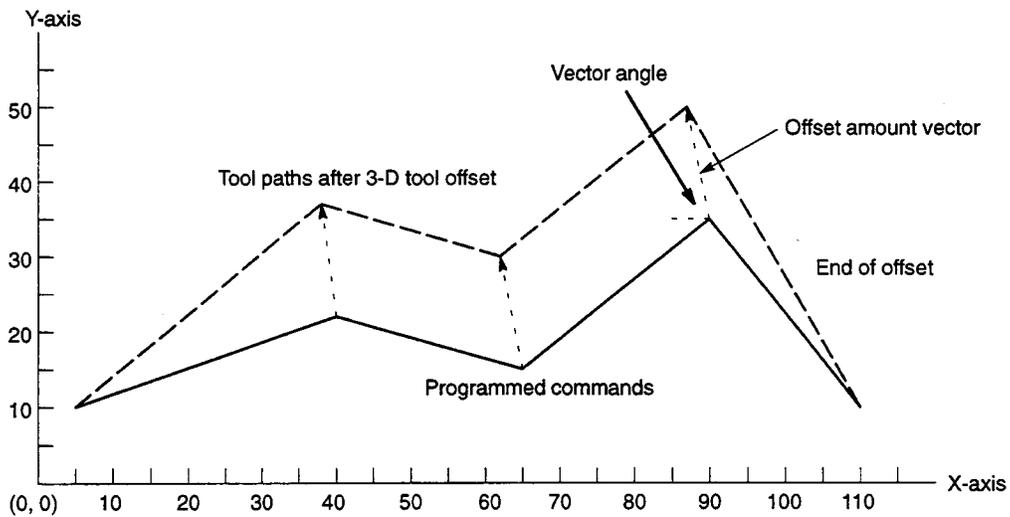


Fig. 3.4.5.1 Tool Path in 3-D Tool Offset and Programmed Tool Path

- The offset vectors of the individual axes are specified in the following manner.

$$x = \frac{I \times d}{R}$$

$$y = \frac{J \times d}{R}$$

$$z = \frac{K \times d}{R}$$

I, J, K : I, J, and K command values

d : Specified value for the specified tool offset number

R : If pm 6850 is "0"

$$R = \sqrt{I \times I + J \times J + K \times K}$$

If pm6850 is other than "0"

$$R = \text{pm6850} \text{ (1 = Least setting unit)}$$

- The end point of a block is calculated in the following manner.

$$X'C = X + x$$

$$Y' = Y + y$$

$$Z' = Z + z$$

X, Y, Z: End point specified in the program

X', Y', Z': End point after offset

(b) Blocks not including I, J, and K

The I, J, and K commands specified in the 3-D tool offset mode are retained as modal values and used in the following program. Therefore, the vector component of the block in which I, J, and K commands are not specified are those generated in the previous block.

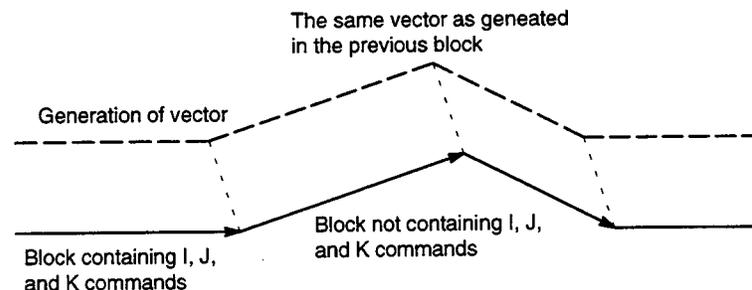
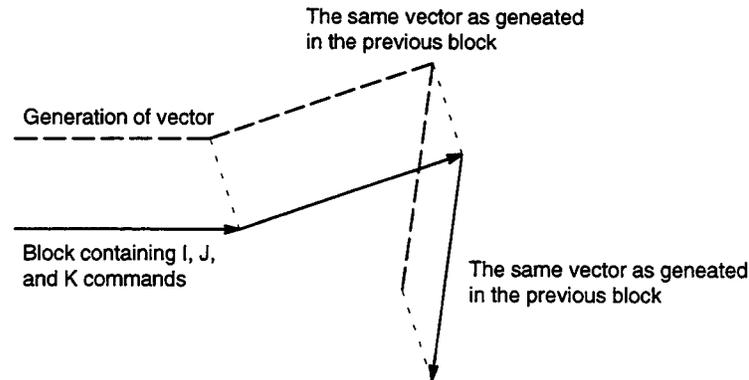


FIGURE 3.4.5.1 Tool Paths after 3-D Offset (No I, J, and K Commands)

There are cases that overcuts occur as shown in Fig. 3.4.5.3 as the result of offset depending on the relationship between the commands and the offset angle.



**FIGURE 3.4.5.2 An Example of 3-D Offset Causes Overcuts
(Continuous Blocks without I, J, and K Commands)**

(c) Changing the angle and size of vector

The vector angle can be changed by specifying I, J, and K commands in the 3-D tool offset mode. Concerning the size of vector, there are two methods to change it,

- Changing the vector size by changing the tool offset number.
- Changing the vector size by specifying the tool offset number and I, J, and K commands after setting other than “0” for the 3-D vector size setting parameter pm6850

(3) Error Processing

(a) Reset in the 3-D tool offset mode

If the NC is reset while it is in the 3-D tool offset mode, the 3-D tool offset mode is canceled by the same processing as the execution of G40. In this cancel processing, axis movement usually executed to cancel the offset amount is not executed although the I, J, and K values are cleared to “0”.

(b) Vector change error in the 3-D tool offset mode

Although vector angle/component can be changed in the 3-D tool offset mode, alarm 0162 “LACK OF ADDRESS” occurs if any of I, J, and K addresses is not specified even in one block causing the operation to be stopped.

- (c) Specifying the vector command that does not allow calculation of 3-D tool offset amount

The 3-D tool offset is executed based on the calculation that is performed using the 3-D vector size in the denominator. Therefore, if the I, J, and K vector commands are “0”, 3-D offset calculation becomes impossible. In this case, alarm 0184 “RADIUS OFS CAL ERROR” occurs and the NC stops operating.

- (d) Starting the 3-D tool offset in the 2-D tool offset mode

It is not allowed to switch the offset mode from the 2-D tool offset mode to the 3-D tool offset mode.

If the command to call up the 3-D tool offset mode is specified while in the 2-D tool offset mode, alarm 0189 “OFS ABNORMAL ERROR” occurs.

- (e) Changing the offset direction in the 3-D tool offset mode

While in the 3-D tool offset mode, it is not allowed to change the offset direction (G4 ‡ G42, or G42 ‡ G41). If such designation is made, alarm 0189 “OFS AB-NORMAL ERROR” occurs. To change the offset direction, the 3-D tool offset mode must be canceled once and the new offset direction must be specified when the 3-D tool offset mode is called up next. Note that it is allowed to change the tool offset direction by using the I, J, K, and D commands.

- (f) Changing the plane in the 3-D tool offset mode

If a plane selection G code (G 17, G 18, G19) is specified in the 3-D tool offset mode to change the plane, alarm 0183 “SURFACE UNMATCH IN OFS MODE” occurs.

(4) Relationship between 3-D Tool Offset and Machine Operations

- (a) Relationship with the G codes that use address I, J, and K

If the G code that allows I, J, and K codes to be specified in the same block is specified in the 3-D tool offset mode, the I, J, and K codes specified in such a block are not regarded as the vector value of the 3-D tool offset. Therefore, the I, J, and K commands specified with the G codes indicated in Table 3.24 are processed as the information that is related with the specified G code and the vector angle/component is not changed even if these commands are executed. If a D command is specified with the G codes that allow the destination of D code, such D command is used to change the offset number instead of changing the vector angle/component. The offset amount is changed immediately after the execution of the D command.

Table 3.4.5.3 G Codes That Use I, J, and K Addresses in the Same Block

G Code	Function Name
G02, G03	Circular interpolation /Circle cut
G106	Automatic corner override
G10	Programmable data input
G22	Stored stroke limit B, C
G65, G66, G67	Macroprogram call

(b) Relationship with tool radius offset C

If the G codes that are not allowed in the 3-D tool offset mode or modal G codes that must not be existent at the start of the tool radius offset C are specified in the 3-D tool offset mode, an alarm occurs. Also note that addresses I, J, and K are used in the 3-D tool offset mode to define the vector angle. Due to this these addresses must not be used as the dummy block commands (incremental commands) like in the tool radius offset C.

(c) Relationship with the high-speed cutting function

If high-speed cutting function is called in the 3-D tool offset mode, alarm 0189 “OFS ABNORMAL ERROR” occurs. To execute high-speed cutting, it is necessary to cancel the 3-D tool offset mode. If the 3-D tool offset mode is called during high-speed cutting operation, normal tool radius offset C is executed instead of the 3-D tool offset.

3.5 S, T, M, AND B Functions

3.5.1 Spindle Function (S Function)

A spindle speed can be directly specified by entering a 5-digit number following address S (S00000). The unit of spindle speed is “r/min”. The specified S value becomes valid from the moment the S command completion input signal (SFIN) is turned ON. If an S command is specified with M03 (spindle forward rotation) or M04 (spindle reverse rotation), the program usually advances to the next block only after the spindle has reached the speed specified by the S command. For details, refer to the instruction manuals published by the machine tool builder.

Example of Programming
S1000 M03;

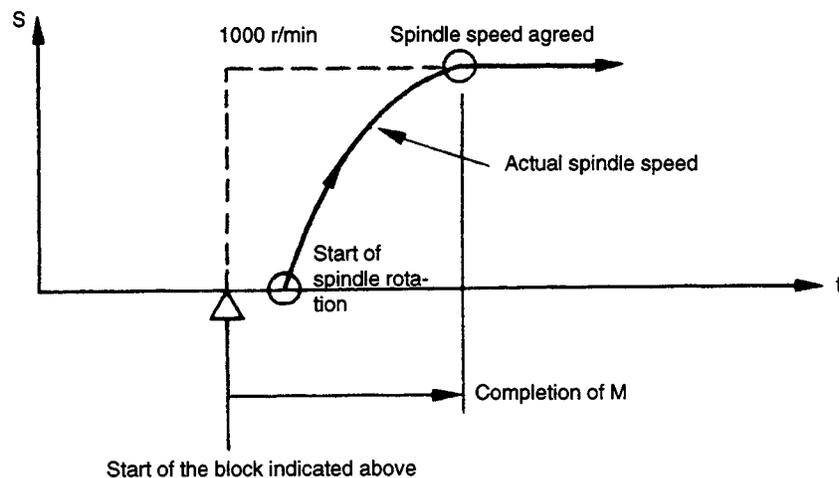


FIGURE 3.5.1.1: Spindle Speed Command

- An S command is modal and, once specified, it remains valid until another S command is given next. If the spindle is stopped by the execution of M05, the S command value is retained. Therefore, if M03 or M04 is specified without an S command in the same block, the spindle can start by using the S command value specified before.
- If a spindle speed is changed while the spindle is rotating by the execution of M03 or M04, pay attention to the selected spindle speed gear range. For details, refer to the instruction manuals published by the machine tool builder.
- The lower limit of an S command (S0 or an S command close to S0) is determined by the spindle drive motor and spindle drive system, and it varies with each machine. Do not use a negative value for an S command. For details, refer to the instruction manuals published by the machine tool builder.

3.5.2 Tool Function (T Function)

The tool function has two command designation types as T2-digit commands and T4-digit commands.

(1) T2-digit Commands

A tool number is specified by a 2-digit number following address T (Too). In a T command, leading zeros may be omitted.

- T commands are modal and, once a T command is executed, it remains valid until another T command: specified next.
- T commands are normally used to select the tool number of the tool to be selected next by the automatic tool changer (ATC). A T command can be specified independent of the G and H or D code that is used for tool length and tool radius offset of the presently selected tool.
- The range of tool numbers that can be specified with address T varies with each machine. For details, refer to the instruction manuals published by the machine tool builder. If a T code is specified with axis move commands in the same block, whether the T command is executed with the axis move commands simultaneously or it is executed after the completion of the axis move commands varies with each machine. For details, refer to the instruction manuals published by the machine tool builder.

(2) T4-digit Command*

A tool number is specified by a 4-digit number following address T (Toooo). In a T command, leading zeros maybe omitted. In comparison to the T2-digit commands, only the number of digits is increased and functions and other details are the same as T2-digit commands.

3.5.3 Miscellaneous Function (M Function)

The miscellaneous function is specified by a maximum of a three-digit number (Mooo) following address M. With the exception of specific M codes, the functions of M00 to M89 codes are defined by the machine tool builder. Therefore, for details of the M code functions, refer to the instruction manuals published by the machine tool builder.

The M codes specific to the NC are described below.

(1) M codes Relating to Stop Operation (M00, M01, M02, M30)

When an M code relating to stop is executed, the NC stops buffering. Whether spindle rotation, coolant discharge or another operation stops in response to the execution of such as M code is determined by the machine tool builder. For details refer to the instruction manuals published by the machine tool builder. For these M codes, a code signal is output independently in addition to M2-digit BIN code.

- M00 (Program Stop)

If M00 is specified during automatic operation is interrupted after the completion of the commands specified with M00 in the same block the M00R signal is output. The interrupted automatic operation can be restarted by pressing the cycle start switch.

- M01 (Optional Stop)

If M01 is executed with the optional stop switch ON, the same operation as with M00 is executed. If the optional stop switch is OFF, M01 is disregarded.

- M02 (End of Program)

M02 should be specified at the end of a program. When M02 is executed during automatic operation, automatic operation ends after the commands specified with M02 in the same block have been completed. The NC is reset. The state after the end of a program varies with each machine. For details, refer to the instruction manuals published by the machine tool builder.

- M30 (End of Tape)

Normally, M30 is specified at the end of tape. When M30 is executed during automatic operation, automatic operation ends after the commands specified with M30 in the same block have been completed. The NC is reset and the tape is rewound. The state after the execution of M30 varies with each machine. For details, refer to the instruction manuals published by the machine tool builder.

(2) Internally Processed M Codes

M codes in the range of M90 to M99 and M190 to M199 are processed by the NC internally and the corresponding output signal (BIN code and decode output) is not output even when these M codes are executed.

Table 3.5.3.1 Internally Processed M Codes

M Code	Function	Setting at Power-ON
*M90	Program interruption function OFF	○
*M91	Program interruption function ON	
*M92	Multi-active registers OFF	○
*M93	Multi-active registers ON I	
*M94	Mirror image OFF I o I	○
*M95	Mirror image ON	
*M96	Tool radius offset C, round-the-arc mode	○
*M97	Tool radius offset C, point of intersection calculation mode	
M98	Subprogram call	
M99	End of subprogram	
*M191	Comment output function	
*M192	Tool life control count	

Note 1: M190 to M199 are used for extension M codes.

2: When the power is turned ON, the M code mode indicated by “○” symbol is set. This is not influenced by the reset operation.

(3) Other General M Codes

The functions of the M codes other than the specific M codes are determined by the machine tool builder. The representative use of several general M codes is given below. For details, refer to the instruction manuals published by the machine tool builder. If an M code is specified with axis move commands in the same block, whether the M code is executed with the axis move commands simultaneously or it is executed after the completion of the axis move commands is determined by the machine tool builder. For details, refer to the instruction manuals published by the machine tool builder.

Table 3.5.3.2 Other General M Codes

M Code	Function	Remarks
M03	Spindle start, forward direction	Generally, M state between M03 and M04 cannot be switched directly. To change the M code state, execute M05 once.
M04	Spindle start, reverse direction	
M05	Spindle stop	
M08	Coolant ON	
M09	Coolant OFF	

3.5.4 Second Miscellaneous Function (B Function) *

For positioning of an index table and other functions, B codes (B000) are used; a B code consists of a three-digit number and address B. If a B code is specified with axis move commands in the same block, whether the B code is executed with the axis move commands simultaneously or it is executed after the completion of the axis move commands is determined by the machine tool builder. For details, refer to the instruction manuals published by the machine tool builder. The B function and the T4-digit specification * are not selectable at the same time.

- B commands are modal and, once a B command is executed, it remains valid until another B command is specified next.
- The standard interface of the B function outputs the signals to the machine in “3-digit BIN” format.
- “B” is used to specify the address for B codes. If the B function is selected, B-axis cannot be used to represent the 4th- or 5th-axis.

4

ENHANCED LEVEL COMMANDS

Chapter 4 describes the program support functions, automation support functions, and macro programs.

4.1	PROGRAM SUPPORT FUNCTIONS (1)	4-3
4.1.1	Canned Cycles (G73 to G89, G181 to G189) *	4-3
4.1.2	Hole Machining Pattern Cycles (G70, G71, G72) *	4-32
4.1.3	Solid Tap Function (G84, G74) *	4-36
4.1.4	Deep-hole Solid Tap Function (G184, G174)*	4-46
4.1.5	Circle Cutting Function (G12, G13)	4-57
4.1.6	Mirror Image ON/OFF (M94, M95) *	4-61
4.1.7	Programmable Data Input (G10) *	4-64
4.1.8	Subprogram Call Up Function (M98, M99)	4-67
4.2	PROGRAM SUPPORT FUNCTIONS (2)	4-69
4.2.1	Program Copy (G25)*	4-69
4.2.2	Automatic Corner Override (G106) *	4-72
4.2.3	Stored Stroke Limit B and C (G22, G23) *	4-77
4.2.4	Break Point Function	4-82
4.2.5	High-speed Cutting *	4-82
4.2.6	Chamfering and Corner Rounding Commands *	4-85
4.2.7	Corner Feedrate Designation (G107, G108)'	4-89
4.3	AUTOMATING SUPPORT FUNCTIONS	4-102
4.3.1	Skip Function (G31) *	4-102
4.3.2	Program interrupt Function (M90, M91)*	4-105
4.3.3	Tool Life Control Function *	4-107
4.4	MACROPROGRAMS	4-114

4.4.1	Differences from Subprograms	4-114
4.4.2	Microprogram Call (G65, G66, G67)	4-115
4.4.3	Variables	4-126
4.4.4	Operation Instructions	4-155
4.4.5	Control Instructions	4-157
4.4.6	Registering the Microprogram	4-163
4.4.7	Microprogram Alarm Numbers	4-164
4.4.8	Examples of Microprograms	4-165

4.1 PROGRAM SUPPORT FUNCTIONS (1)

4.1.1 Canned Cycles (G73 to G89, G181 to G189) *

Canned cycles (G73, G74, G76, G77, G80 to G89, G181, G182, G185, G186, G187, and G189) can define specific movements that usually require several blocks of single-block commands. 19 kinds of canned cycles are provided and G80 cancels the called up canned cycle program.

(1) G Codes Calling Canned Cycles and Axis Movement Patterns of Canned Cycles

Canned cycles are largely classified into normal hole-machining cycles and 2-step hole-machining canned cycles.

(a) Normal hole-machining canned cycles

For calling normal hole-machining canned cycles, the following G codes are used.

Table 4.1.1.1 Table of Normal Hole-machining Canned Cycles

G Code	Axis Feed	Processing at Hole Bottom	Retraction	Application
G73	Intermittent feed (dwell at each peck feed possible)	—	Rapid traverse	High-speed deep hole drilling
G74	Cutting feed	Spindle rotation in the reverse direction after dwell	Cutting feed → Dwell → Spindle reverse rotation	Reverse tapping
G76	Cutting feed	Spindle indexing → Shift	Rapid traverse → Shift, Spindle start	Boring
G77	Spindle indexing → Shift → Rapid traverse → Shift → Spindle start → Cutting feed	Dwell	Rapid traverse → Spindle indexing, Shift → Rapid traverse → Shift, Spindle start	Back boring
G80	—	—	—	Cancel
G81	Cutting feed	—	Rapid traverse	Drilling
G82	Cutting feed	Dwell	Rapid traverse	Spot facing
G83	Intermittent feed	—	Rapid traverse	Deep hole drilling
G84	Cutting feed	Spindle start in the reverse direction after dwell	Cutting feed → Dwell → Spindle reverse rotation	Tapping
G85	Cutting feed	—	Cutting feed	Boring

G86	Cutting feed	Spindle stop	Rapid traverse → Spindle start	Boring
G87	Cutting feed	Spindle stop	Manual return → Spindle start	Boring
G88	Cutting feed	Spindle stop after dwell	Manual return → Spindle start	Boring
G89	Cutting feed	Dwell	Cutting feed	Boring

(b) 2-step Hole-machining Canned Cycles

The following G codes are used to call 2-step hole-machining canned cycles

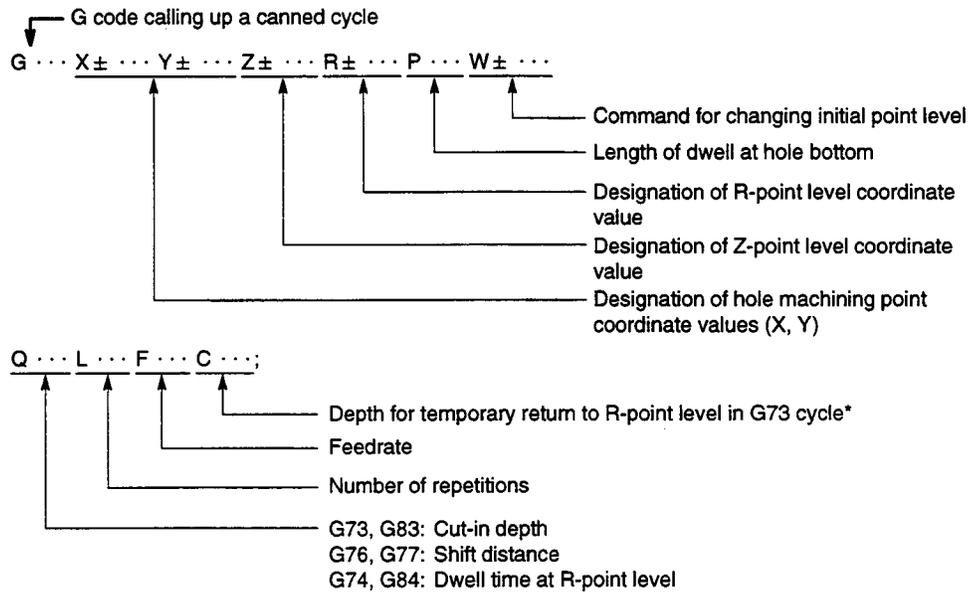
Table 4.1.1.2 Table of 2-step Hole-machining Canned Cycles

G Code	Axis Feed	Processing at Hole Bottom	Retraction	Application
G181	Cutting feed → Rapid traverse → Cutting feed	—	Rapid traverse	2-step hole drilling
G182	Cutting feed → Rapid traverse → Cutting feed	Dwell	Rapid traverse	2-step hole spot fac- ing
G185	Cutting feed → Rapid traverse → Cutting feed	—	Rapid traverse 2 → Rapid traverse → Cutting feed 2	2-step hole boring
G186	Cutting feed → Rapid traverse → Cutting feed	Spindle stop	Rapid traverse → Spindle start	2-step hole boring
G187	Cutting feed 1 → Cutting feed 2 → Cutting feed 1	Spindle stop after dwell	Cutting feed 2	2-step hole boring
G188	Cutting feed 1 → Rapid traverse → Cutting feed 1	Dwell	Rapid traverse 2 → Rapid traverse → Cutting feed 2	2-step hole boring

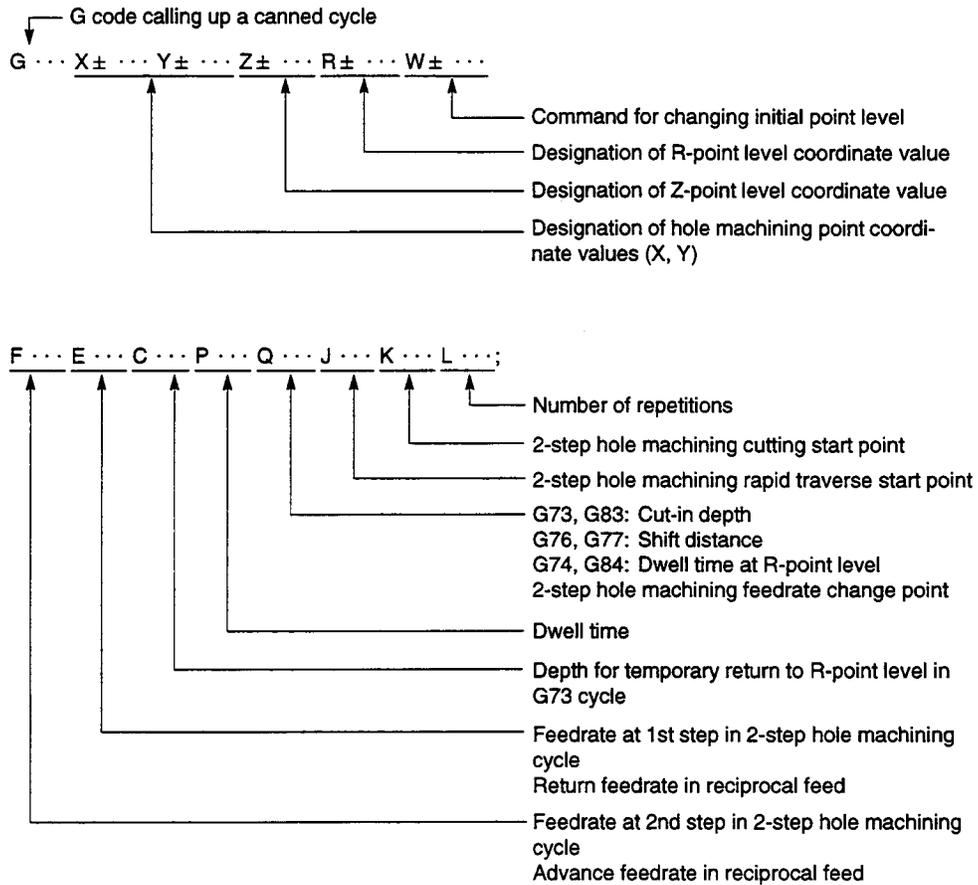
(2) Programming Format

The programming format used for a 2-step hole-machining canned cycle differs from that used for a normal hole-machining canned cycle as indicated below.

(a) Example of programming for normal hole-machining canned cycles



(b) Example of programming for optional special canned cycles



The following four steps are executed as one cycle with the commands indicated above.

- Positioning at the hole machining position
- Rapid traverse to R-point level
- Hole machining up to Z-point level
- Return to R-point or initial point level

(c) Description of the addresses

Table 4.1.1.3 Description of the Addresses

Address	Description
<ul style="list-style-type: none"> • X, Y (Positioning axes) 	The hole machining position is specified by either incremental or absolute values. For positioning operation, G00 or G01 mode can be selected by the setting for a parameter. If the G01 mode is selected, positioning at the hole machining position is executed at a feedrate specified by an F code.
<ul style="list-style-type: none"> • Z (hole machining axis) 	The position of the hole bottom is specified by either an absolute value or an incremental value referenced to the R-point level. Axis move from the R-point level to the bottom of the hole is controlled in the G01 mode using the feedrate specified by an F code. With some types of canned cycles, G00 operation is included (intermittent feed, for example). Return motion from the bottom of the hole to the R-point level is controlled in the G00 or G01 mode.
<ul style="list-style-type: none"> • R (hole machining feed start level) 	The position of R-point level is specified by either an absolute value or an incremental value. The feed axis is the hole machining axis. Return operation from the R-point level to the initial point level is controlled in the G00 mode.
<ul style="list-style-type: none"> • L (number of repetitions) 	The number of repetitions is specified by address L. If designation of address L is omitted, "L1" is assumed. If "L = 0" is specified, only positioning at (X, Y) is executed.

**t Initial Point Level**

The initial point level is the absolute position of the point where the hole machining axis is located when the NC mode enters the canned cycle mode from the canned cycle cancel state. The initial point level is not changed if a canned cycle is executed in the G99 (R-point level return) mode.

(d) Changing the initial-point level

The initial point level can be changed by using address W; the return operation to the new level is called the W-point level return.

- With the W command specified in the G91 mode (an incremental command), W-point level is set in reference to the initial point level specified in the preceding block.
- In the G98 mode (initial point level return), single-block stop does not occur at the R-point level, but at the W-point level or the initial point level.
- If the W-axis is set for an additional axis, the W command cannot be used to change the initial point level. In this case, if address W is specified, it is regarded as an axis command.

Example of Programming

```

N1 G99 G81 X1 ... Y1 ... R1 ... Z1 ... F ... ;
N2 G98 X2 ... Y2 ... R2 ... Z2 ... W1 ... ;
N3 X3 ... Y3 ... R3 ... Z3 ... W2 ... ;
N4 X4 ... Y4 ... ;
    
```

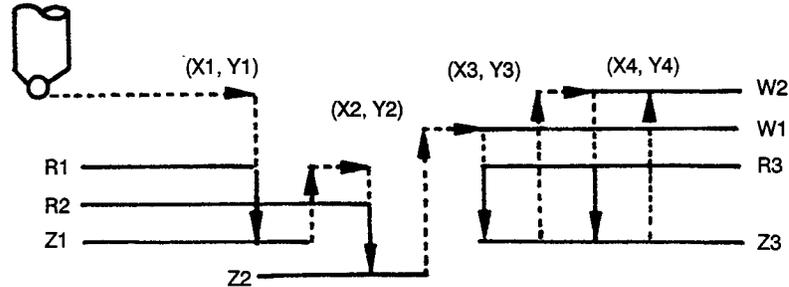


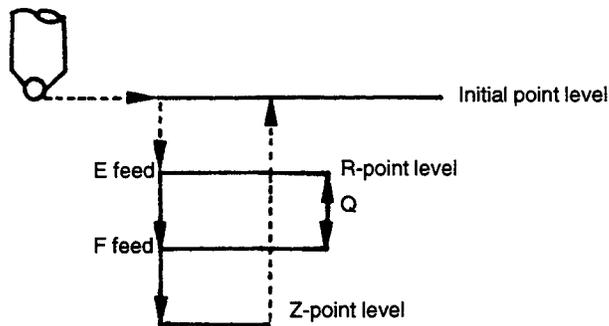
FIGURE 4.1.1.1 W-point Level Command



If the initial point level should be changed, it is not necessary to cancel the canned cycle.

(e) 2-step feed *

In drilling operation, it is possible to execute drilling at an accurate position without executing centering operation if feedrate is reduced near the level where the drill engages the workpiece.



Address E: 1st-step feedrate
 Address Q: Distance to be moved at 1st-step feedrate (unsigned value)

FIGURE 4.1.1.2 2-step Feed

(f) Reciprocal feed *

For tapping, boring and other similar operations, individual feedrates can be specified for advance and return feed.

By using this feature, tapping operation which will be completed before the spring in a tapper is fully extended can be performed. In addition, this mode provides various advantages for different types of operation. In boring operation, cycle time can be shortened. In reaming operation, return movement of the cutting tool does not leave scratches on the finished surface.

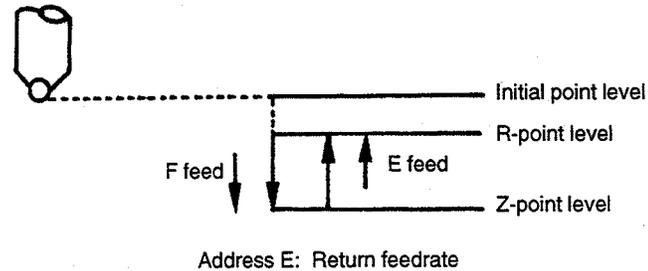


FIGURE 4.1.1.3 Reciprocal Feed

(3) Table of Operation

(a) Normal hole-machining canned cycles (including new initial point level return)

Table 4.1.1.4 Table of Normal Hole-machining Canned Cycles

		● Dwell ○ Single-block stop	
	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
<p>G73 (constant pitch)</p> <p>High-speed deep hole drilling</p>	<p>G73 X...Y...Z...R...Q...L</p>	<p>...F...C*...P...W...;</p>	
	<p>Dwell time: P or setting for pm0401 pm4017 D4 = 0: pm0401 invalid, 1: pm0401 valid δ: Setting for pm0870 Q: Unsigned incremental value C: Unsigned incremental value</p>		
<p>G73 (variable pitch)</p> <p>High-speed deep hole drilling</p>	<p>G73 X...Y...Z...R...I...J...</p>	<p>K...L...F...C*...P...W</p>	<p>...;</p>
	<p>Dwell time: P or setting for pm0401 pm4017 D4 = 0: pm0401 invalid, 1: pm0401 valid δ: Setting for pm0870 Q: Unsigned incremental value C: Unsigned incremental value</p>		
<p>G74</p> <p>Reverse tapping</p>	<p>G74 X...Y...Z...R...P...L</p>	<p>...F...Q*...W...;</p>	
	<p>P: Dwell time at Z-point level Q: Dwell time at R-point level</p>		

Table 4.1.1.4 Table of Normal Hole-machining Canned Cycles (cont'd)

● Dwell
○ Single-block stop

	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
<p>G76 (fixed shift distance)</p> <p>Boring</p>	<p>G76 X ... Y ... Z ... R ... Q ... L</p>	<p>... F ... P ... W ... ;</p>	
	<p>Q: Shift distance (signed incremental value) Shifting feedrate: pm2864 Dwell time: P or setting for pm0400 pm4017 D3 = 0: pm0400 invalid, 1: pm0400 valid</p>		
<p>G76 (variable shift distance)</p> <p>Boring</p>	<p>G76 X ... Y ... Z ... R ... I ... J</p>	<p>... L ... F ... P ... W ... ;</p>	
	<p>i: X-axis shift distance (signed incremental value) j: Y-axis shift distance (signed incremental value) Shifting feedrate: pm2864 Dwell time: P or setting for pm0400 pm4017 D3 = 0: pm0400 invalid, 1: pm0400 valid</p>		
<p>G77 (fixed shift distance)</p> <p>Back boring</p>	<p>G77 X ... Y ... Z ... R ... Q ... L</p> <p>Not used</p>	<p>... F ... P ... W ... ;</p>	
	<p>Q: Shift distance (signed incremental value) Shifting feedrate: pm2864 Dwell time: P or setting for pm0400 pm4017 D3 = 0: pm0400 invalid, 1: pm0400 valid</p>		
<p>G77 (variable shift distance)</p> <p>Back boring</p>	<p>G77 X ... Y ... Z ... R ... I ... J</p> <p>Not used</p>	<p>L ... F ... P ... W ... ;</p>	
	<p>i: X-axis shift distance (signed incremental value) j: Y-axis shift distance (signed incremental value) Shifting feedrate: pm2864 Dwell time: P or setting for pm0400 pm4017 D3 = 0: pm0400 invalid, 1: pm0400 valid</p>		

Table 4.1.1.4 Table of Normal Hole-machining Canned Cycles (cont'd)

● Dwell
○ Single-block stop

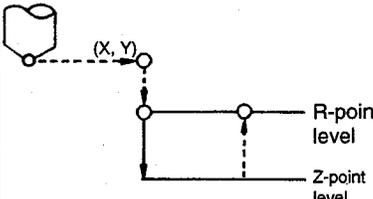
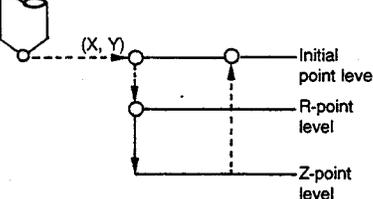
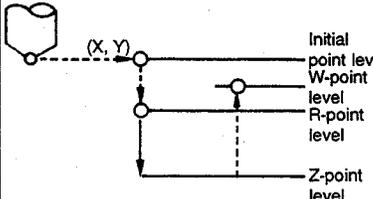
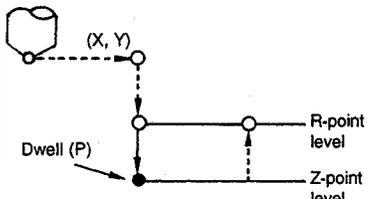
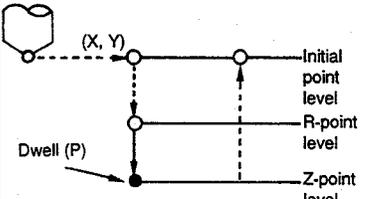
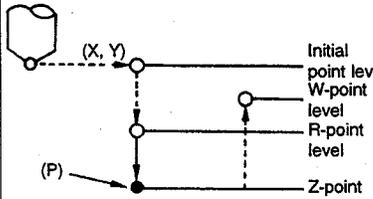
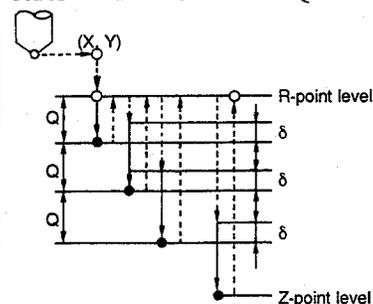
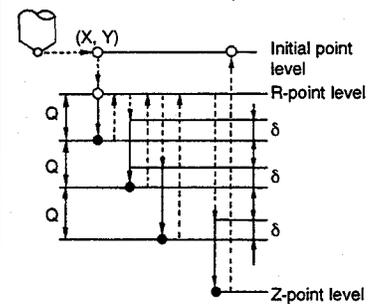
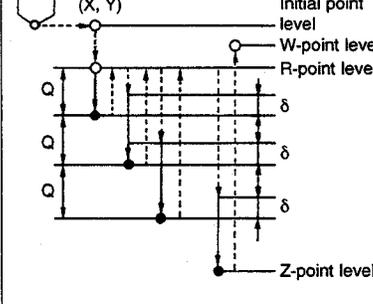
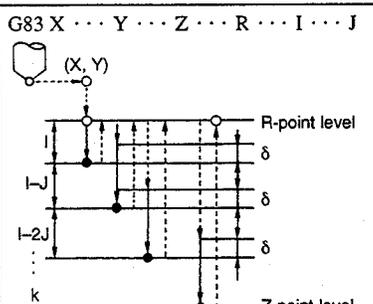
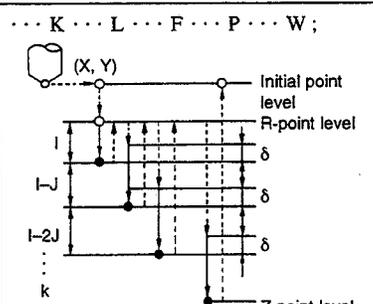
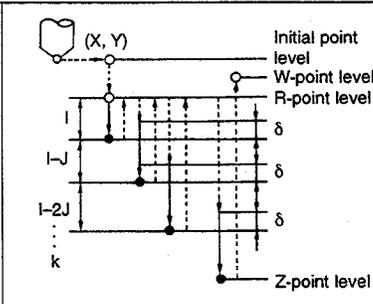
	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
G80 Cancel	G80:		
Drilling	G81 X ... Y ... Z ... R ... L ... F 	... W ... ; 	 
G82	G82 X ... Y ... Z ... R ... P ... L 	... F ... W ... ; 	 
G83 (constant pitch) Deep hole drilling	G82 X ... Y ... Z ... R ... Q ... L 	... F ... P ... W ... ; 	 
<p>δ: Setting for pm0871 Dwell time: P or setting for pm0401 pm4017 D4 = 0: pm0401 invalid, 1: pm0401 valid</p>			
G83 (variable pitch) Deep hole drilling	G83 X ... Y ... Z ... R ... I ... J 	... K ... L ... F ... P ... W ; 	 
<p>I: Initial value J: Decrement value K: Final value (incremental value) δ: Setting for pm0871 Dwell time: P or setting for pm0401 pm4017 D4 = 0: pm0401 invalid, 1: pm0401 valid</p>			

Table 4.1.1.4 Table of Normal Hole-machining Canned Cycles (cont'd)

● Dwell
○ Single-block stop

	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
G84	G84 X ... Y ... Z ... R ... P ... L	... F ... Q * ... W ... ;	
Tapping			
<p>P: Dwell time at Z-point level Q: Dwell time at R-point level</p>			
G85	G85 X ... Y ... Z ... R ... L ... F	... W ... ;	
Boring			
G86	G86 X ... Y ... Z ... R ... L ... F	... W ... ;	
Boring			
G87	G87 X ... Y ... Z ... R ... L ... F	... W ... ;	
Boring			

Table 4.1.1.4 Table of Normal Hole-machining Canned Cycles (cont'd)

● Dwell
○ Single-block stop

	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
G88	G88 X ... Y ... Z ... R ... P ... L	... F ... W ... ;	
Boring			
G89	G89 X ... Y ... Z ... R ... P ... L	... F ... W ... ;	
Boring			

(b) 2-step hole-machining canned cycles with special options*
(includes new initial point return)

Table 4.1.1.5 Table of 2-step Hole-machining Canned Cycles with Special Options

		● Dwell ○ Single-block stop		
	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode	
Reverse tapping	G74 X...Y...Z...R...P...L 	... F...E...Q...W...; 		
	P: Dwell time at Z-point level Q: Dwell time at R-point level E: Return feedrate in reciprocal feed (Z-point level → R-point level)			
Drilling	G81 X...Y...Z...R...L...F 	... E...Q...W...; 		
	E: 2-step feed feedrates (1st step: E, 2nd step: F) Q: Feedrate switchover point (unsigned incremental value)			
Spot facing	G82 X...Y...Z...R...P...L 	... F...E...Q...W...; 		
	E: 2-step feed feedrates (1st step: E, 2nd step: F) Q: Feedrate switchover point (unsigned incremental value)			
Tapping	G84 X...Y...Z...R...P...L 	... F...E...Q...W...; 		
	P: Dwell time at Z-point level Q: Dwell time at R-point level E: Return feedrate in reciprocal feed (Z-point level → R-point level)			

Table 4.1.1.5 Table of 2-step Hole-machining Canned Cycles with Special Options (cont'd)

		● Dwell ○ Single-block stop		
	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode	
Boring	G85 X ... Y ... Z ... R ... L ... F 	... E ... W ... ; 		
	E: Return feedrate in reciprocal feed (Z-point level → R-point level)			
Boring	G86 X ... Y ... Z ... R ... L ... F 	... E ... Q ... W ... ; 		
	E: Return feedrate in reciprocal feed (1st step: E, 2nd step: F) Q: Feedrate switchover point (unsigned incremental value)			
Boring	G89 X ... Y ... Z ... R ... P ... L 	... F ... E ... W ... ; 		
	E: Return feedrate in reciprocal feed (Z-point level → R-point level)			

Table 4.1.1.5 Table of 2-step Hole-machining Canned Cycles with Special Options (cont'd)

● Dwell
 ○ Single-block stop

	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
<p>G181</p> <p>2-step hole drilling</p>	<p>G181 X ... Y ... Z ... R ... L ...</p>	<p>F ... J ... K ... E ... Q ... W ...</p>	
<p>J: 2-step hole machining rapid traverse start point (signed absolute/incremental value) K: 2-step hole machining cutting feed start point (signed absolute/incremental value) E: 2-step feed feedrates (1st step: E, 2nd step: F) Q: Feedrate switching point (unsigned incremental value)</p>			
<p>G182</p> <p>2-step hole spot facing</p>	<p>G182 X ... Y ... Z ... R ... P ...</p>	<p>L ... F ... J ... K ... E ... Q ... W ...</p>	<p>... ;</p>
<p>J: 2-step hole machining rapid traverse start point (signed absolute/incremental value) K: 2-step hole machining cutting feed start point (signed absolute/incremental value) E: 2-step feed feedrates (1st step: E, 2nd step: F) Q: Feedrate switching point (unsigned incremental value)</p>			
<p>G185</p> <p>2-step hole boring</p>	<p>G185 X ... Y ... Z ... R ... L ...</p>	<p>F ... J ... K ... E ... W ... ;</p>	
<p>J: 2-step hole machining rapid traverse start point (signed absolute/incremental value) K: 2-step hole machining cutting feed start point (signed absolute/incremental value) E: Return feedrate in reciprocal feed (Z-point level → K-point level, J-point level → R-point level)</p>			

Table 4.1.1.5 Table of 2-step Hole-machining Canned Cycles with Special Options (cont'd)

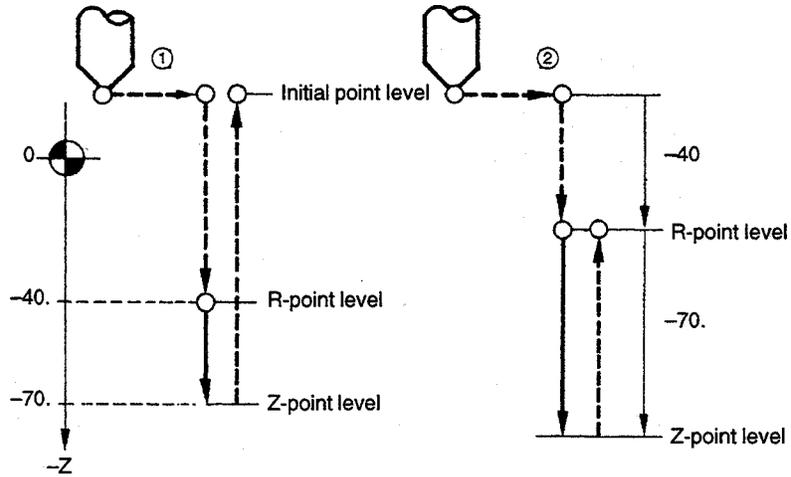
● Dwell
○ Single-block stop

	G99 (R-point Level Return) Mode	G98 (Initial Point Level Return) Mode	G98 (W-point Level Return) Mode
2-step hole boring	<p>G186 X ... Y ... Z ... R ... L ...</p>	<p>F ... J ... K ... E ... Q ... W ... ;</p>	
	<p>J: 2-step hole machining rapid traverse start point (signed absolute/incremental value) K: 2-step hole machining cutting feed start point (signed absolute/incremental value) E: 2-step feed feedrates (1st step: E, 2nd step: F) Q: Feedrate switching point (unsigned incremental value)</p>		
2-step hole boring	<p>G187 X ... Y ... Z ... R ... L</p>	<p>F ... P ... J ... K ... E ... W ... ;</p>	
	<p>J: 2-step hole machining rapid traverse start point (signed absolute/incremental value) K: 2-step hole machining cutting feed start point (signed absolute/incremental value) E: Return feedrate in reciprocal feed (J-point level → K-point level)</p>		
2-step hole boring	<p>G189 X ... Y ... Z ... R ... P ...</p>	<p>L ... F ... J ... K ... E ... W ... ;</p>	
	<p>J: 2-step hole machining rapid traverse start point (signed absolute/incremental value) K: 2-step hole machining cutting feed start point (signed absolute/incremental value) E: Return feedrate in reciprocal feed (Z-point level → K-point level)</p>		

(4) Using the Absolute/Incremental Designation Commands (G90/G91)

(a) Example of programming in the G90/G91 modes

- Program for returning to the initial point level in the G90 mode
G98 G90 G81 X ... Y ... Z-70. R-40. F ...;
- Program for returning to the R-point level in the G91 mode
G99 G91 G81 X ... Y ... Z-70. R-40. F ...;



Returning to the initial point level
(absolute programming mode)

(b) Returning to the R-point level
(incremental programming mode)

FIGURE 4.1.1.4 Programming in the Absolute/Incremental Programming Mode

(b) Switching the G90/G91 mode

As shown in the program given below, only addresses specified later are updated as in blocks N2 ‡ N3, including the switching of G code from G90 to 91. For addresses not specified, the position data specified in the preceding blocks are retained.

Example of Programming

```

N1  G92  X0 Y0 Z0 ;
N2  G98  G90 G81 X10. Y10. Z-50. R-20. F100 ; .....
N3  G91  X20. R-30. ; .....
N4          Z-50. ; .....
N5  G99  Z-40. R-50. ; .....
N6  G80  G00 Z50. ; .....

```

Table 4.1.1.6 Absolute Position of R- and Z-point Levels

Program	Absolute Position		Remarks
	Z =	R =	
	-50.	-20.	Starting a canned cycle
	-50.	-30.	Changing the data of X and R. Z-point level specified in the block immediately before this block is retained.
	-80.	-30.	Changing the data of Z. R-point level specified in the block immediately before this block is retained.
	-90.	-50.	Changing the data of R and Z
	Cutting tool moves to Z = 0. (absolute position)		Canceling the canned cycle

(c) Changing the R- and Z-point level in the G91 mode

If R is specified without Z while a canned cycle is executed in the G91 mode, Z-point level is recalculated as incremental distance from R-point level.

Example of Programming

```
G92 X0 Y0 Z0
G98 G91 G81 X ··· Y ··· Z-10.0 R-5.0 F ··· ; .....
X ··· R-7.0; .....
X ··· Z-3.0; .....
R-4.0 Z-11.0 .....
```

Table 4.1.1.7 Changes in Incremental Distance from R-point Level to Z-point

Program	R-point Level	Z-point Level
	-5.0	-15.0
	-7.0	-17.0 (Note 1)
	-7.0 (Note 2)	-10.0
	-4.0	-15.0

Note 1: Z-point position is commanded to be-10.0 in the G91 mode in advance.

Note 2: R-point position is commanded to be-7.0 in the G91 mode in advance.

(5) Specifying the High-Speed Deep Hole Drilling (G73) and Deep Hole Drilling (G83) Codes*

(a) Variable pitch peck feed command

In a deep hole drilling cycle called by G73 and G83, it is possible to specify a variable pecked feed stroke by using addresses I, J, or K instead of specifying address Q that specifies a constant peck feed stroke.

I: Initial value of peck feed stroke

J: Decrement value for the second and later peck feeds

K: Final value of peck feed stroke

d: Return stroke at each cut

Example of Programming

```
G91 G73X ... Y ... R-30. Z-55. I10. J1. K4. F ...;
```

Peck feed stroke

1st peck feed— 10 mm ? I10.

2nd peck feed— 9 mm

3rd peck feed— 8 mm

4th peck feed— 7 mm

5th peck feed— 6 mm

6th peck feed— 5 mm

7th peck feed— 4 mm ? K4.

8th peck feed— 4 mm

9th peck feed— 2 mm ? K'

Total 55.00 mm ? -Z -55.

(b) Dwell at each peck feed

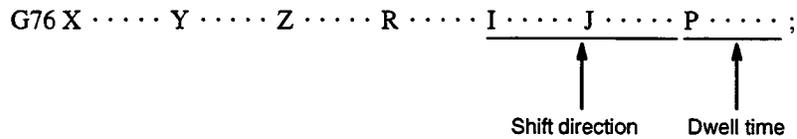
In a deep hole drilling cycle called by G73 or G83, load applied to a drill can be reduced by giving dwell for a short time after reaching the bottom in each peck feed. Dwell time can be set for parameter pm0401 or specified by address P.

- If peck dwell is specified in a program using address P, it supercedes the setting for parameter pm0401. If a P command is not specified in a program, the setting for parameter pm0401 (1 = 1ms) is used for peck dwell. If P0 is specified, the setting for parameter pm0401 is used.
- Default value for dwell at the bottom of a hole can be set by using parameter pm0401.
- The setting for parameter pm0401 is disregarded by setting appropriate value for parameter pm4017 D4. If pm0401 is disregarded, peck dwell is not executed unless a P command is specified.

pm4017 D4 = 0	pm0401 Invalid
pm4017 D4 = 1	pm0401 Valid

(6) Programming Format for Boring (G76) and Back Boring (G77)*

The direction of shift motion in the G76/G77 cycle can be specified by “Q command + angle (setting parameter pm0805)” as well as the commands in a program. For the designation of dwell time before shifting, it is possible to specify it by the commands in a program in addition to the setting for parameter pm0400.



(a) Boring command (G76)

- If addresses I or J are specified with address Q in the same block, addresses I and J are disregarded.
- For the dwell at the bottom of hole, the setting for pm0400 is given priority to the commands specified in a program.
- Parameter pm0400 is valid if “pm4017 D3 = 1”.
- Axis shift feedrate is set for parameter pm2864. If “pm2864 = 0”, feedrate specified by an F command is used.
- Since address Q is modal, if a Q command is specified before I or J commands, these commands are disregarded. To make I and J commands valid, specify them after canceling a canned cycle by G80.

• Programming for fixed shift distance

G76 X_ Y_ Z_ R_ Q_ L_ F_ P_ W_ ;

• Programming for variable shift distance

G76 X_ Y_ Z_ R_ I_ J_ L_ F_ P_ W_ ;

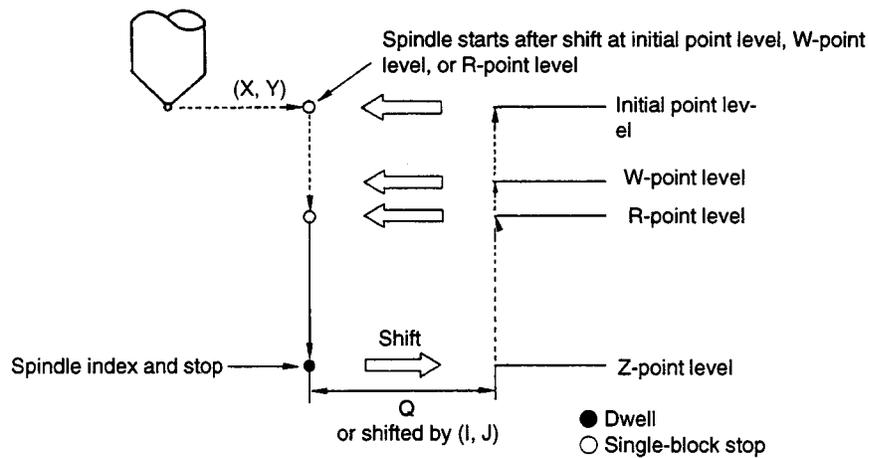


FIGURE 4.1.1.7 Commands of Boring Cycle (G76)

(b) Back boring command (G77)

- If addresses I or J are specified with address Q in the same block, addresses I and J are disregarded.
- For the dwell at the bottom of a hole, the setting for pm0400 is given priority to the commands specified in a program.
- Parameter pm0400 is valid if “pm4017 D3 = 1”.
- Axis shift feedrate is set for parameter pm2864. If “pm2864 = 0”, feedrate specified by an F command is used.
- Since address Q is modal, if a Q command is specified before I or J commands, these commands are disregarded. To make I and J commands valid, specify them after canceling a canned cycle by G80.

- Programming for fixed shift distance

G77_X_Y_Z_R_Q_L_F_P*_W*_;

- Programming for variable shift distance;

G77_X_Y_Z_R_I_J_L_F_P*_W*_;

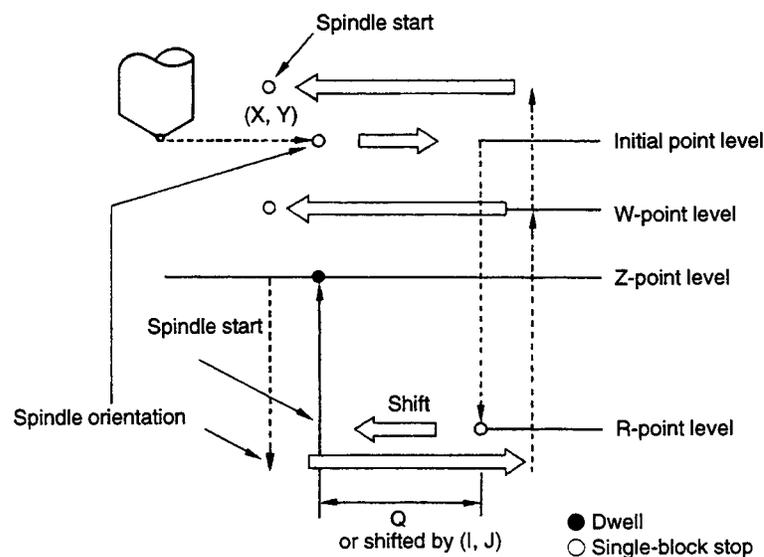


FIGURE 4.1.1.8 Commands of Back Boring Cycle (G77)

(7) Programming Format for Reverse Tapping (G74) and Tapping (G84)*

In a tapping cycle, dwell can be specified after the return to the R-point and initial point levels in addition to the dwell at the bottom of the hole to absorb elongation and contraction of the taper spring.

G74 and G84 call up the reverse tapping and tapping cycles, respectively; there are no differences in the basic control for these two patterns of tapping cycles. If the tapping cycle starts with the spindle rotation of M03, M04 is output at the bottom of the hole for the return operation and, conversely, if a cycle starts with the spindle rotation of M04, M03 is output at the bottom of the hole for the return operation. If G84 is specified without starting the spindle, M04 is output at the bottom of the hole, and if G74 is specified without starting the spindle, M03 is output at the bottom of the hole.

(a) Reverse tapping command (G74)

- Example of Programming

```
G74 X_Y_Z_R_P_L_F_E_Q_W_;
```

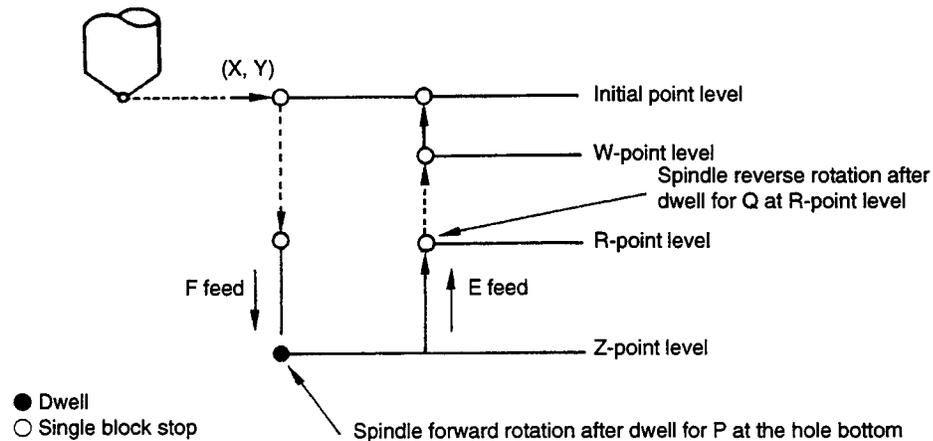


FIGURE 4.1.1.9 Reverse Tapping Commands (G74)

(b) Tapping command (G84)

- Example of Programming

G84 X_Y_Z_R_P_L_F_E_Q_W_;

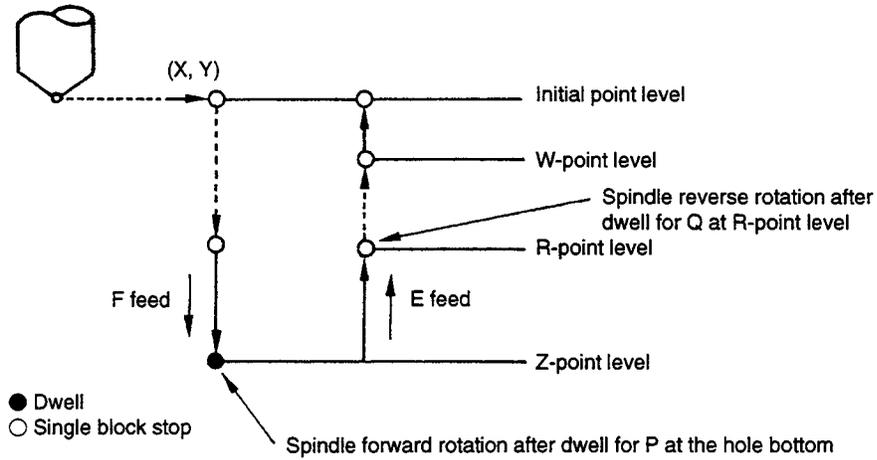


FIGURE 4.1.1.10 Tapping Cycle Command (G84)

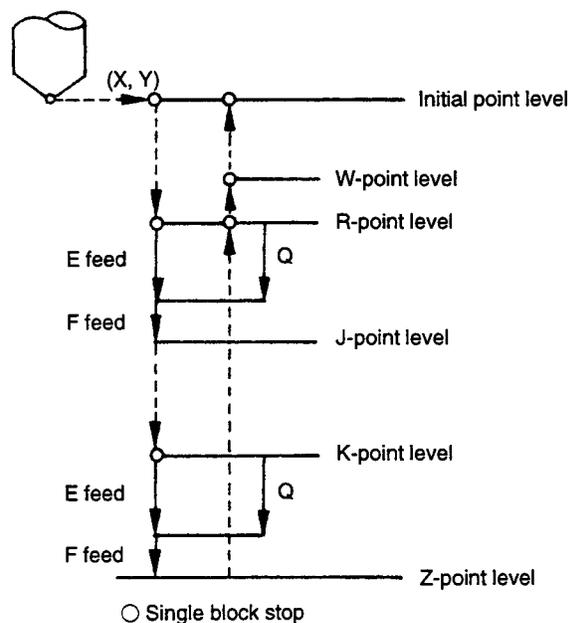
(8) Programming for 2-step Hole Machining Canned Cycles*

To execute a hole machining cycle in a hollow workpiece such as a cast workpiece, 2-step hole machining canned cycles are provided. For each type of canned cycle, called by a G code of 80s, 2-step hole machining canned cycle is available. To call up a 2-step hole machining canned cycle, add "100" to a G code number used to call up a hole machining canned cycle. The following G codes are used to call up the 2-step hole machining canned cycle: G181, G182, G185, G186, G187, and G189.

(a) 2-step hole drilling command (G181)

Example of Programming

```
G181 X_Y_Z_R_L_F_J_K_E_Q_W_;
```



- J: 2-step drilling cycle rapid traverse start point (absolute/incremental)
- K: 2-step drilling cycle cutting feed start point (absolute/incremental)
- E: 2-step feed feedrates (1st step: E, 2nd step: F)
- Q: Feedrate switching point (unsigned incremental value)
- W: New initial point (absolute/incremental)

Note: To specify J or K in the G91 (incremental) mode, specify the distance in reference to the R-point level.

FIGURE 4.1.1.11 2-step Hole Drilling Cycle Command (G181)

(9) Example of Canned Cycle

Example of Programming

```

N10 C92 X0 Y0 Z0 ;
N11 G90 G98 ;
N12 G81 X30. Y40. R-20. Z-30. F200 ;
N13 M98 P400 ;
N14 G00 X0 Y0 ;
N15 T05 ;
N16 M06 ;
N17 G84 X30. Y40. R-20. Z-30. F2000 ;
N18 M98 P400 ;
N19 G00 X0 Y0 ;

```

→ Absolute mode, Initial point level return
 → Drilling cycle
 → Subprogram call
 → Selection of a taper
 → Tool change cycle
 → Tapping cycle
 → Subprogram call

```

O400
N100 G91 X40. L3 ;
N101 Y30. ;
N102 X-40. L3 ;
N103 Y30. ;
N104 X40. L3 ;
N105 G90 G80 ;
N106 M99 ;

```

Subprogram defining hole position pattern

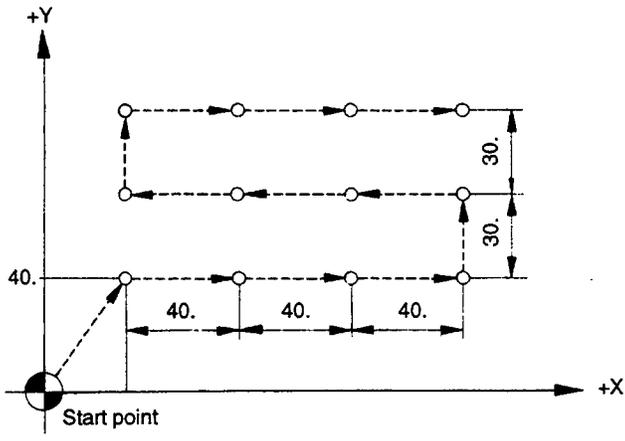


FIGURE 4.1.1.12 Example of Canned Cycle

(10)Supplements to Canned Cycle Functions

- When a canned cycle is executed with the SINGLE-BLOCK switch set ON, the cycle is suspended at the following points and the FEED HOLD lamp lights.
 - After positioning at (X, Y)
 - After positioning at R-point level
 - After the completion of one cycle if an L command is specified
- Upon completion of a canned cycle, the single-block stop takes place as in operations other than canned cycles and, in this case, the FEED HOLD lamp does not light.
- Before entering the canned cycle mode, define the R-point and Z-point levels newly by designating R and Z. The R-point and Z-point data is cleared when the canned cycle mode is canceled.
- If a canned cycle is executed after changing the address data, one of the following address commands must be specified in that block; otherwise, the canned cycle is not executed. X, Y, Z, a, b, R
- When M, S, T, and/or B * is specified in a block which calls up a canned cycle, M, S, T, and/or B * signal is output at the first positioning operation. If the number of repetitions (L) is specified, M, S, T, and/or B * signal is output at the first positioning operation, too. M, S, T, and B * commands should be specified in a block without other commands.
- If the following G codes are specified in the canned cycle mode, alarm “0170” occurs. It is necessary to cancel the canned cycle mode before specifying these G codes.
G code in *-group with an exception of G04,G70,G71, and G72
 - G code in 07-group (G41, G42)
 - It is possible to insert the block that calls for dwell during the execution of a canned cycle. In this case, dwell function is executed correctly.
- If a canned cycle is called up in the tool radius offset C (G41, G42) mode, alarm “0182” occurs.
- Before entering a canned cycle, the spindle must have been started in automatic operation by executing M03 or M04. It is not allowed to enter a canned cycle after changing the spindle rotation direction manually.

- It is possible to call a subprogram while in the canned cycle mode by specifying the subprogram call command (M98). The canned cycle can be continuously executed in the called subprogram. When a subprogram is called, the P command (dwell time) used for a canned cycle is temporarily destroyed by the P command (jump destination program number) specified with M98. However, the previously specified P value is recovered when jump to the sub-program is completed.

The limit on the nesting level (up to 4 levels) of M98, command from punched tape, etc, and other requirements are the same as those applicable to the modes other than the canned cycle mode. An alarm occurs if M98 and canned cycle commands are specified in the same block.

- An L command specifying the number of repetitions of a canned cycle is non-modal. However, there is a special case in which an L command is saved temporarily. As in the example of programming given below, if only an L command is specified in a block, it is saved until it is actually executed.

Example of Programming

G91 G81 X10. R-20. Z-30. F100;

L3; The canned cycle is not executed since this block does not include X, Y, Z, a, or R.

L3 is saved.

X20; G81 is executed three times according to the L3 command which has been saved. After the execution of the G81 cycle three times, L3 is deleted.

- A canned cycle is canceled either when G80 or a G code in 01-group is specified. If a G code of 01-group is specified with a G code calling up a canned cycle, alarm "0170" occurs. However, if G80 is specified as a G code to call up the canned cycle, alarm does not occur, but the specified operation is executed.
- An F command specified in a canned cycle remains valid even after the cancellation of the canned cycle.
- If an address search is executed during the execution of a canned cycle while a cycle is suspended (block stop state), an alarm occurs. Address search during block stop state specified in a program is allowed. If more than one cycle is executed by the designation of an L command, address search attempted at the completion of a single cycle causes an alarm.
- It is possible to call up a canned cycle while in another canned cycle already called up. If address data are omitted in a newly specified canned cycle calling block, the modal data of that address are used.

4.1.2 Hole Machining Pattern Cycles (G70, G71, G72) *

The hole machining pattern cycle permits hole machining position data to be entered by the radius and angle. The function converts the data into the position data in the Cartesian coordinate system to execute positioning. The function supports three pattern cycles; bolt-hole circle, arc, and line-at-angle. These pattern cycles can be used in the canned cycle mode.

Positioning at the coordinate position specified by the radius and angle is executed in the rapid traverse (G00) mode.

Table 4.1.2.1 G Codes Used for Calling Hole Machining Pattern Cycles

G code	Function	Group
G70	Bolt hole circle command	*
G71	Arc command	*
G72	Line-at-angle command	*

(1) Bolt Hole Circle Command (G70)

By using the bolt hole circle pattern, hole machining, positions are defined on the circle of radius “I” and having the center at (X, Y). The positions are arranged on this circle-equally divided by L and starting from a point at angle J measured from the X-axis.

G70X... Y... I... J... L...;

Table 4.1.2.2 Description of Addresses

G code	Function
X, Y	Used to define the center of bolt hole circle. The values to be assigned to addresses X and Y should conform to the specified dimensioning mode, G90 or G91.
I	Used to specify the radius of the bolt hole circle to be defined. The value must be positive and can be set in units of least input.
J	Used to set the angle of the first positioning point. The value should be set in units of 0.001 degree; the angle measured in the counterclockwise direction is expressed by using a positive value.
L	Used to set the number of equally divided positions on the circle. If a positive value is used, positioning is executed in the counterclockwise direction and if a negative value is used, positioning is executed in the clockwise direction.

Example of Programming

```
G81 G98 G90 Z-50. R-20. F20 L0 ;
G70 X90. Y30. I40. J20. L6 ;
G80 G00 X0 Y0 ;
```

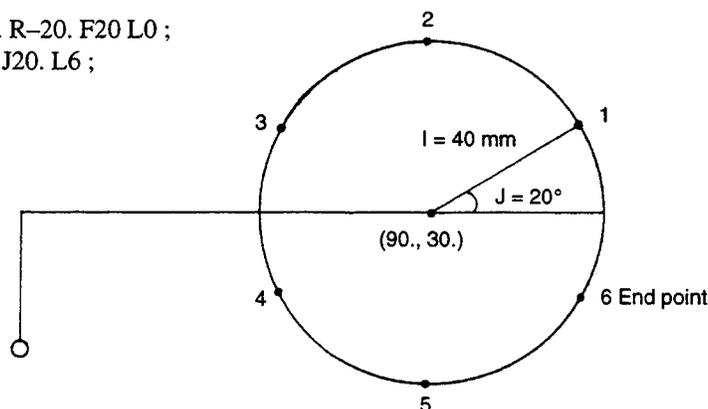


FIGURE 4.1.2.1 Bolt Hole Circle Command (G70)

(2) Arc Command (G71)

By using the arc pattern, hole machining positions are defined on the arc of radius “I” and having the center at (X, Y). The positions are arranged on this arc – L, equally spaced in angular intervals of K, and starting from a point at angle J measured from the X-axis.

G71X...Y...I...J...L...;

Table 4.1.2.3 Description of Addresses

G code	Function
X, Y	Used to define the center of the arc. The values to be assigned to addresses X and Y should conform to the specified dimensioning mode, G90 or G91.
I	Used to specify the radius of the arc to be defined. The value must be positive and can be set in units of least input.
J	Used to set the angle of the first positioning point. The value should be set in units of 0.001 degree; the angle measured in the counterclockwise direction is expressed by using a positive value.
K	Used to set the angular intervals. The values should be set in units of 0.001 degree, and positioning is executed, in the counterclockwise direction when a positive value is specified.
L	Used to set the number of positioning times. A positive value must be set.

Example of Programming

```
G81 G98 G90 Z-50. R-20. F20 L0 ;
G71 X70. Y30. I110. J20. K15.2 L7 ;
G80 G00 X0 Y0 ;
```

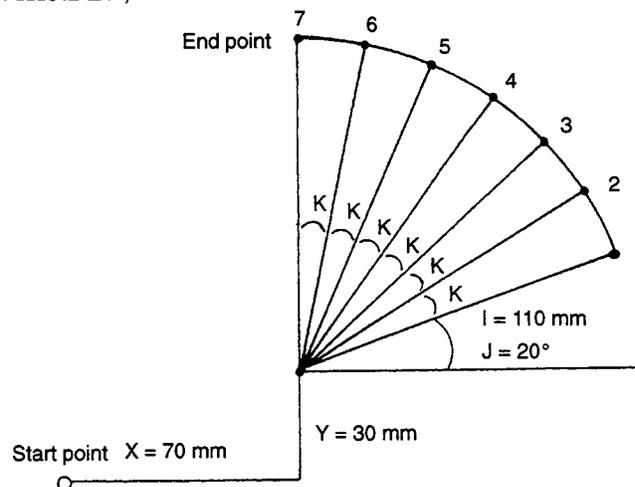


FIGURE 4.1.2.2 Arc Command (G71)

(3) Line-at-angle Command (G72)

By using the line-at-angle pattern, hole machining positions are defined on the line having an angle of J from the X-axis. The positions are arranged on this line – L, equally spaced in intervals of I, and starting from at the point defined by (X, Y).

G72X... Y... I... J... L...;

Table 4.1.2.4 Description of Addresses

G code	Function
X, Y	Used to define the coordinate values of the start point. The values to be assigned to addresses X and Y should conform to the specified dimensioning mode, G90 or G91.
I	Used to set the interval. If a negative value is set, positioning is executed in the symmetrical direction in reference to the start point. The value can be set in units of least input.
J	Used to set the angle of the line measured from the X-axis. The value should be set in units of 0.001 degree and the angle measured in the counterclockwise direction is expressed by a positive value.
L	Used to set the number of positioning times. A positive value must be set.

Example of Programming

```
G81 G98 G90 Z-50. R-20. F20 L0 ;
G71 X70. Y30. I25. J15.5 L6 ;
G80 G00 X0 Y0 ;
```

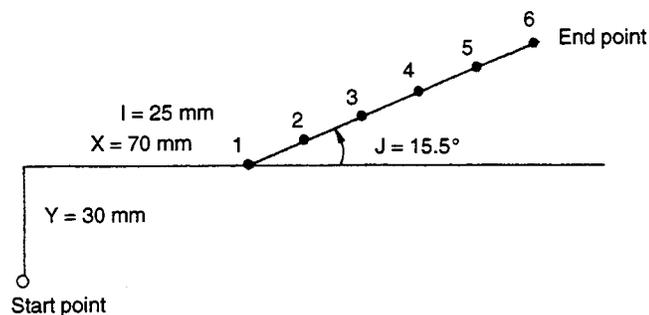


FIGURE 4.1.2.3 Line-at-angle Command (G72)

(4) Supplements to the Hole Machining Pattern Cycle Function

- To execute a hole machining cycle using G70, G71, or G72, a canned cycle (G73, G74, G76, G77, G81 - G89) should be specified with “L = 0” in the preceding block as shown in the Example of Programming of Fig. 4.15. By the designation of “L = 0”, the hole machining position data are saved without executing the canned cycle.
- It is allowed to specify a G code calling a canned cycle with G70, G71, or G72 in the same block. However, the canned cycle (G73 and G83) which uses addresses I, J, and K cannot be specified with G70, G71, or G72 in the same block. To use G73, G83, or 2-step hole machining canned cycles (G181, G182, G185, G186, G187, G189), either use address Q or specify I, J, or K in the preceding block.
- The hole machining canned cycle executed using G70, G71, or G72 ends after the retraction from the hole machined last. To execute positioning to the next position, move distance must be calculated if the cycle is programmed in the G91 (incremental) mode. Since the use of the G90 (absolute) mode does not require such calculation, it can be used more conveniently than the G91 mode.
- “Upon completion of G70, G71, or G72, the canned cycle mode remains valid. It is necessary to execute G80 to cancel the canned cycle mode.
- If G70, G71, or G72 is specified in the tool radius offset mode (G41, G42), alarm “0182” occurs.

4.1.3 Solid Tap Function (G84, G74) *

The solid tap function executes tapping by synchronizing Z-axis feed with spindle rotation. If tapping is executed by using this function, a floating chuck is not necessary any more and, at the same time, accurate tapping is made possible at a high speed. Conventional tapping operation can also be programmed.

(1) Requirements for Adding the Solid Tap Function

To add the solid tap function, the following conditions are required.

- To install the S5-digit analog output option.
- To use Yaskawa’s VS-626VM3 or VS-626M5 for the spindle driver.
- To fix the gear ratio, to be used for solid tapping, by installing a PG to the spindle or the spindle drive motor. The best performance can be expected if the spindle is directly coupled to the spindle drive motor.

(2) Commands Used for Solid Tap Cycle

To execute solid tapping, change the mode to the solid tap mode and then specify the solid tap cycle.

(a) Solid tap mode commands

The following G codes are provided to determine which of the tapping mode is called up, solid tap or conventional tapping cycle. These G codes are modal. When the power is

turned ON or the NC is reset, G94 mode is set.

- Solid tap mode command (G93)

Once G93 is executed, the tapping cycle commands (G84/G74) are executed in the solid tap mode. In this mode, Z-axis feed is controlled in the “feed per revolution” mode. In the solid tap mode, no machining other than solid tapping is allowed.

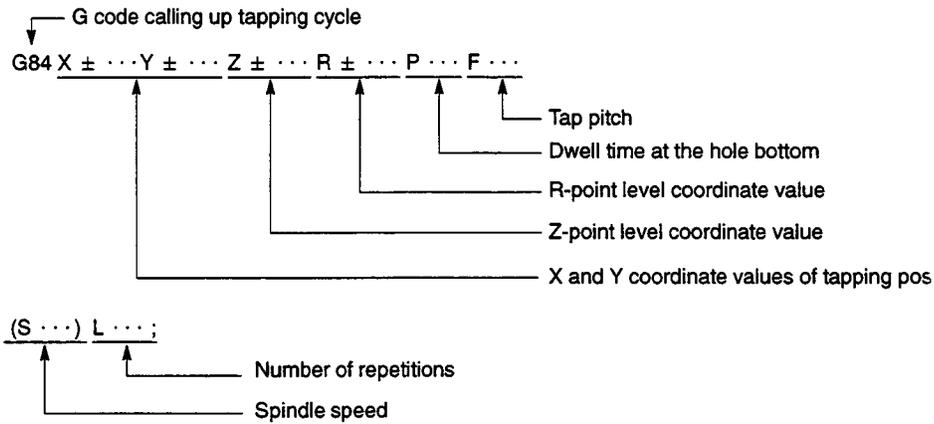
- Solid tap mode cancel command (G94)

The solid tap mode is canceled and the conventional tapping mode is called up. Once G94 is executed, tapping cycles are executed in the conventional mode, in which the Z-axis feed is controlled in the “feed per minute” mode.

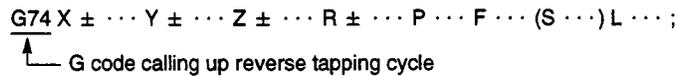
(b) Programming for the solid tap cycle

After executing G93, solid tapping is enabled by specifying the commands indicated below.

- Programming for tapping cycle



- Programming for reverse tapping cycle



For an F code, specify Z-axis feed pitch (mm/rev, inch/rev) per one revolution of the spindle.

The unit of command and programmable range of F and S codes are indicated below. Note that, even within the specified programmable range, the values exceeding the specified limit cannot be used: “F x S = 24000.0 mm/min or 944.8 inch/ min”.

F command (Metric input)	1 = 0.0001 mm/rev Range: 0.0001 to 200.0000 mm/rev
F command (Inch input)	1 = 0.000001 inch/rev Range: 0.00000 I to 7.874015 inch/rev
S command	1 = 1 rev Range: 1 to 4500 rev

(3) Part Program Using Solid Tap Commands

(a) M** command

This command selects the gear range used for solid tap cycle. If no such M code is specified, A gear is selected.

(b) G93 command

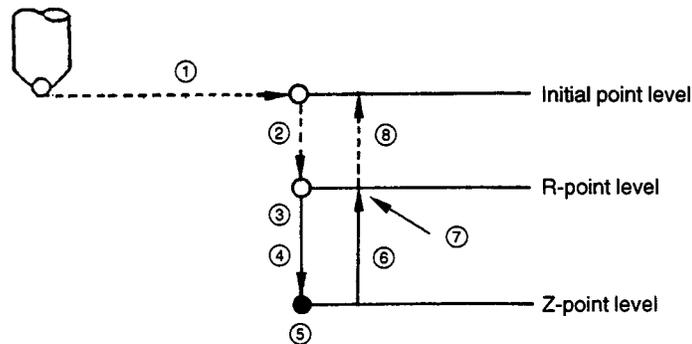
When the G93 command is executed, the spindle stops and the solid tap mode is established with the position loop set for the control of the spindle.

It is also possible to execute spindle indexing to position the spindle at a fixed position before establishing the solid tap mode after the spindle has been stopped. To execute spindle indexing, change the setting for parameter (pm 1053 D2 = 1). Note that spindle indexing to the fixed position is possible only when the spindle and the spindle PG rotate at a 1 : 1 ratio.

The solid tap mode is canceled by G94.

(c) G84 command

How the tapping cycle is executed in the solid tap mode is described below.



Note: For spindle motion, linear pattern acceleration/deceleration is automatically applied. The time constant is set for parameter pm2471.

FIGURE 4.1.3.1 Tapping Cycle Executed in the G84 Mode

- ① Positioning at the position specified by X and Y
- ② Positioning at the R-point level
- ③ Spindle rotating in the forward direction, and cutting to the Z-point level at the specified feedrate (at the start of this block, the number of servo lag pulses is checked for the spindle and the Z-axis whether it is within the allowable number error detect ON)
- ④ Spindle stop (the number of servo lag pulses is checked for the spindle.)
- ⑤ Dwell (if P command is specified.)
- ⑥ Spindle rotating in the reverse direction, and cutting to the R-point level at the specified feedrate

- ⑦ Spindle stop
- ⑧ Positioning at the initial point level (in the G98 mode)
- (d) G74 command

If G74 is specified instead of G84, the same operation is executed with the exception that the spindle rotates in the reverse direction instead of h-ward rotation in step É and that the spindle rotates in the forward direction instead of reverse rotation in stepÛ.

- (e) G94 command

The G93 solid tap mode is canceled when the G94 command is executed. In the solid tap canceled mode, the G84/G74 cycle is executed in the conventional tapping mode.

Example of programming and description of operation

Example of Programming

```

NI G92 X0 Y0 Z0;
(M**); ..... ①
N2 G93;..... ②
N3 G98 G90; ..... ③
N4 G84 X100. Y100. Z-50 R-20. F1. S3000 ..... ④
N5 X200. Y200; ..... ⑤
N6 X10. Y10.; ..... ⑥
N7 G80; ..... ⑦
N8 G94; ..... ⑧
    
```

Operation

Spindle drive gear range is selected (A/B). If the transmission gears are not provided, A gear is selected.

Solid tap mode ON

Modal information of the canned cycle

Tapping in the solid tap mode at (100., 100.) in 1 mm pitches.

Tapping in the solid tap mode at (200., 200.) in 1 mm pitches.

Tapping in the solid tap mode at (10., 10.) in 1 mm pitches,

Canceling the canned cycle

Solid tap mode OFF

(4) Relationship between the Solid Tap and Other Operation

(a) Dry run

Whether the solid tap cycle or the conventional tapping should be executed when G93 is executed with the DRY RUN switch ON can be selected by the setting for a parameter.

pm4016 D6 = 0	Conventional tapping cycle
pm4016 D6 = 1	Solid tap cycle

If a tapping cycle is called up with the setting of “pm4016 D6 = 0”, the cycle is executed in the following manner.

If G93 is executed with the DRY RUN switch set ON, G93 is invalid and the G84/G74 command called up in the solid tap mode is processed as the G84/G74 command called up in the conventional tapping mode. The feedrate for this operation is determined by the setting of the JOG switch. Once the solid tap mode is entered with the DRY RUN switch ON, the G84/G74 command is processed as if the command were called up in the conventional tapping mode even if the DRY RUN switch setting is changed from ON to OFF during the execution of the cycle. Therefore, the spindle does not start even when G93 is executed.

For the execution of the G93 block, whether the DRY RUN switch is ON or not is judged when the G93 code is read. Usually, this state is determined while the commands in the preceding block are executed. Therefore, when G93 is operated with DRY RUN ON for program check, etc., be sure that the DRY RUN switch is ON from the beginning and do not change it before completion.

(b) MST function lock

If G93 is executed with the MST FUNCTION LOCK switch set ON, G93 is invalid and the G84/G74 cycle is executed in the conventional tapping mode. Spindle position is not controlled, either.

Whether the solid tap cycle is executed with the MST function lock state or not is judged when the G93 code is read. Therefore, if the G93 program should be checked in the MST function lock state, the MST LOCK switch must be set ON from the beginning of operation so that it will not be turned ON during the operation.

(c) Machine lock, axis-dependent machine lock

If tapping cycle is executed in the solid tap mode with the MACHINE LOCK switch or Z-AXIS LOCK switch of axis-dependent machine lock switches set ON, although the spindle rotates, Z-axis does not move but only the position data are updated.

(d) Feedrate override and spindle override

During tapping cycle in the solid tap mode (G84 or G74), feed override is fixed at 100%. Note that override for rapid traverse is valid. Spindle override is also invalid during tapping cycle in the solid tap mode and fixed at 100%.

(e) Feed hold

During tapping cycle in the solid tap mode, feed hold is invalid.

(f) Mode change

During tapping cycle in the solid tap mode, mode change is invalid.

(g) Program re-start

If the program is restarted from a block in the solid tap mode, G93 is not executed. Therefore, to restart a program from a block in the solid tap mode, it is necessary to enter and execute G93 in the MDI mode.

(h) Special options for canned cycles

Although an E command for reciprocal feed does not cause an alarm, it is disregarded. The return feedrate is set for parameter pm 1252.

(5) Supplements to the Solid Tap Function

- In the G93 block, only S, F, and N codes can be specified. If other commands are specified in the G93 block, alarm “0250” occurs.
- In the G93 mode, an S code is processed as the S command for solid tap operation.
- In the G93 mode, only G codes indicated below can be specified. If a G code not indicated below is specified, alarm “0250” occurs. Concerning G01, although it can be specified in the G93 mode, axis move commands cannot be specified.

G codes that can be specified in the G93 mode

G00, G01, G04, G70, G71, G72, G74, G80, G84, G90, G91, G98, G99

- Spindle indexing called in the G93 mode is the indexing to the fixed position in reference to the zero point pulse (C-axis) that is output from the spindle PG.
- If the G93 mode is established once, the spindle will fail to start even if the G93 mode is canceled by G94 unless M03 or M04 is specified. Note that this feature is determined by the processing sequence of the machine. Although an S command remains, gear selection signal is not output in the G93 mode. Accordingly, it is necessary to specify an S command again when specifying M03 or M04.
- If there is more than one gear range, it is necessary to select the gear range to be used for solid tap before specifying G93. Execute G93 only after the appropriate gear has been selected.
- Concerning the R-point and Z-point level commands, G98/G99, G90/G91, L command, etc., all the precautions described for conventional tapping operation apply.
- The output voltage for spindle forward rotation/reverse rotation in G84/G74 cycle will have a different sign according to the setting for the parameter. If “pm1001 D0 = 0” “+” for the forward rotation, and if “pm 1001 D0 = 1”, “-” “for the forward rotation. For reversing the spindle rotation direction, the NC automatically reverses the sign without using SINV signal.
- To specify G94 after the completion of solid tap, make sure to cancel the canned cycle

by specifying G80.

- When G94 is specified after the completion of solid tap, an F command value is reset to “0”. Make sure to specify an F command when a program including cutting feed is specified after the designation of G94.

(6) Functions Related to Solid Tap

If the options related to the solid tap function are selected, the following functions are added or modified.

(a) Display of synchronization error in solid tapping

- While in the solid tap mode, the servo lag error display screen displays the following data.

X-axis . . . Spindle servo lag error

Y-axis . . . Synchronization error between the spindle and Z-axis

Note that the value indicated for Y-axis is not accurate if the operation is executed continuously (single block OFF).

- If “pm4015 D6 = 1”, it is possible to display the peak value of the number of synchronization error pulses for X-axis and Z-axis. (X-axis: Positive peak value; Z-axis: Negative peak value)

(b) Error detect in the solid tap mode

By setting “1” for parameter pm4015 D5, it is possible to set the error detect OFF mode for rapid traverse (X- and Y-axis positioning, Z-axis positioning) during solid tap.

By this setting, cycle time can be reduced. In this case, the program must be made carefully since the program advances to the Z-axis block immediately after the completion of pulse distribution for the positioning of the X- and Y-axis.

(c) High-speed return speed in solid tap

- By setting a numeral “n” for parameter pm1252, the cutting speed is controlled during solid tap so that the feedrate in the return motion will be “n” times (0.1 £ n £ 25.5) the feedrate applied in the cutting motion.

pm1252 Programmable range:0 to 255

Setting: 1 = 0.1 times (If “pm 1252 = 0”, setting of
 “1” is equivalent to 1 time.)

- For spindle speed, the value of “S command value × multiplication ratio” is clamped at the maximum speed for solid tap.

(d) Alarm code

Table 4.1.3.1 List of Alarm

Alarm Code	Description	Causes
0250	A command not allowed in the solid tap mode	<ul style="list-style-type: none"> In the G93 block, a command other than G93, S, F, and N is specified. G93 is specified in other than the G00 or G011 mode established by the 01-group G code. A G code not allowed in the G93 mode is specified. Pitch F specified for solid tap operation is greater than the allowable maximum value (200 mm/rev).
2191	Solid tap input/output error	<ul style="list-style-type: none"> When solid tap operation is executed, the position control loop is not established for the spindle. When spindle indexing is executed, SLPC is turned OFF before the completion of indexing.

(7) Parameters and input/Output Signals Related to Solid Tap Function

(a) List of parameters

Table 4.1.3.2 List of Parameter

Item	Parameter	Setting and Setting Range
Spindle speed for checking-Spindle stop after indexing	pm1225: 1st spindle gear A pm 1 226: 1st spindle gear B	Setting 11 r/ min Range 1 - 255
Error detect ON area with spindle position loop established	pm1331: 1st spindle gear A pm1332: 1st spindle gear B	Setting 1 = 1 least detect- able increment Range 1 - 255
Servo error area with spindle position loop ON	pm1351: 1st spindle gear A pm1352: 1st spindle gear B	Range 1 = 1% Setting 0 - 200
Spindle speed for command voltage of 10 V with the gear used for solid tap	pm 1415: 1st spindle gear A pm 1435: 1st spindle gear B	Range 1 = 1 r/min Setting 1 - 32767

Maximum spindle speed for sol-id tap	pm1416: 1st spindle gear A pm1436: 1st spindle gear B	Setting r/min Range 32767	1 = 1 1 -
Spindle position loop gain for solid tap	pm1417: 1st spindle gear A Pm1437: 1st spindle gear B	Setting 0.01 (1/s) Range 32767	1 = 1 -
In-position width for solid tap servo axis to be accelerated to the target point	pm1500: Solid tap A gear pm1501: Solid tap B gear	Setting pulse Range 32767	1 = 1 0 -
Solid tap synchronization compensation parameter (k1)	pm1502: Solid tap A gear pm1504: Solid tap B gear	Range 32767 - 32727	-

Table 4.1.3.2 List of Parameters (cont'd)

Item	Parameter	Setting and Setting Range
Solid tap synchronization compensation parameter (k2)	pm1503: Solid tap A gear pm1505: Solid tap B gear	Range 32767 - 32727 -
Spindle gear ration for solid tapping	pm1510: 1st spindle gear A:No.of teeth at the spindle side pm1511: 1st spindle gear A:No.of teeth of intermediate gear (Spindle) pm1512: 1st spindle gear A:No. of teeth of intermediate gear (motor) pm1513: 1st spindle gear A:No. of teeth at the motor side pm1514: 1st spindle gear B:No. of teeth at the spindle side pm1515: 1st spindle gear B:No. of teeth of intermediate gear (spindle) pm1516: 1st spindle gear B:No. of teeth of intermediate gear (motor) pm1517: 1st spindle gear B:No. of teeth at the motor side	Setting 1 = 1 tooth Range 0 - 32768
Linear pattern spindle acceleration/deceleration time constant for solid tapping	pm2471: 1st spindle gear A pm2472: 1st spindle gear B	Setting 1 = 1 ms Range 0 - 32767
Spindle indexing start speed	pm2541: 1st spindle gear A pm2542: 1st spindle gear B	Setting 1 = 1 r/min Range 0- 32767
Creep speed for spindle indexing	pm1546: 1st spindle gear A Pm1547: 1st spindle gear B	Setting 1 r/ min Range 0 - 32767

(b) List of input/output

Table 4.1.3.3 List of input/Output Signal

Item	Address	Setting and Setting Range
Solid spindle selection input (STGR1)	#31155	0: Gear A selection 1: Gear B selection

4.1.4 Deep-hole Solid Tap Function (G184, G174)*

When carrying out large-diameter hole tapping or deep-hole tapping operation, it is difficult to finish tapping in a single step of tapping operation due to chips accumulated inside the hole and cutting load that increases as tapping cycle progresses. In such cases, tapping cycle has to be repeated until the thread is cut to the bottom of the hole.

The tapping cycle can be repeated by a single block of commands by specifying the G 184 or G 174 command in the solid tap mode (G93).

(1) Commands

The G codes used for deep-hole solid tapping are indicated below.

Table 4.1.4.1 G Codes Used for Deep-hole Solid Tapping

G code	Function	Group
G184	Deep-hole solid tapping	09
G174	Reverse deep-hole solid tapping	09

(a) Addresses

Basically, the same addresses as used in the G84/G74 solid tapping cycle are used in the G184/G174 deep-hole solid tapping operation in the same format. The addresses that are used specially for the deep-hole solid tapping are indicated below.

Table 4.1.4.2 Addresses Used for Deep-hole Solid Tapping

Address	Description	Unit	Programmable Range
Q	Used to specify the depth of feed for each infeed cycle in an unsigned incremental values,	1 = 0.001 mm 0.0001 inch	0.001 to 999999.999 mm
I	Used to specify the depth of feed for the first infeed cycle in an unsigned incremental value.	1 = 0,001 mm 0,0001 inch	0.001 to 999999.999 mm, 0.0001 to 99999.9999 mm
J	Used to specify increment/decrement amount of depth of feed in an unsigned incremental value.	1 = 0,001 mm 0,000 inch	0.001 to 999999.999 mm, 0.0001 to 99999.9999 inch
K	Used to specify the depth of feed for the last infeed cycle in an unsigned incremental value. When “I -(n - 1) x J < K”, the value specified for K is taken as the depth of feed.	1 = 0,001 mm 0,0001 inch	0.001 to 999999.999 mm, 0.0001 to 99999.9999 inch
E	Used to specify the feedrate multiplication ratio to be applied when moving up the Z-axis (retraction operation). If no E command is specified, the value set for pm1252	1 = 0.1 ^{(Note)1, 2}	0.1 to 25.5
C	Used to specify the levels (tool retraction start levels) where the tool is retracted to the R-point level in an unsigned incremental value. These levels (C-point levels) are set in intervals of the value set for address C starting from the R-point level in the direction to the Z-point level.	1 = 0.001 mm, 0.0001 inch	0.001 to 999999.999 mm, 0.0001 to 99999.9999 inch
δ	Used to specify the chamfering amount for G184/G174 cycle by parameter pm0873. Note that the value must always be input in “mm” even if the tapping cycle is programmed in the inch mode.	1 = 0.001 mm ^{(Note)3}	0 to 999999999

Note 1.: An E command specified in a canned cycle is usually treated as the command for E feed (mm/min, inch/min). However, if an E command is specified in the G184/G174 mode, it is treated as the feedrate multiplying command and, therefore, the unit of command differs from the unit applied to normal E commands.

2: Although the E command is modal if it is used in a canned cycle, the modal value before the change of a G code becomes invalid if a G code of 09 group is changed to G184/G174 from a G code other than these.

3: With the G174 command, the same tapping operation is executed as with the cycle called by the G184 command with the exceptions that the tapping is carried out in the reverse rotation or the spindle and that the spindle rotates in the forward direction for tap retraction operation.

(2) Operation

There are two types of deep-hole solid tapping operations as indicated below.

- R-point return type (type A)
- Pecking cycle type (type B)

Which of the types should be used can be selected by the setting for parameter pm4016.

Type A	pm4016 D4 = 0
Type B	pm4016 D4 = 1

(a) R-point return type (type A)

The R-point return type includes the two types of commands such as fixed pitch command and variable pitch command.

- Fixed pitch command

Example of Programming

G93;

G184 (or G174) X± •• Y± •• Z± •• R± •• W± •• Q •• P •• F •• E •• S •• L ••;

X •• Y ••;

G94; .

G93/G94	Solid tap mode ON, G94: Solid tap mode OFF
G184/G174	Deep-hole solid tap, G174 Reverse deep-hole solid tap
X, Y	Hole position (mm, inch)
Z	Distance from the R-point level to the bottom of the hole, or the position of the hole bottom (mm, inch)
R	R-point level (mm, inch)
W	W-point return level (mm, inch)
Q	Depth of feed in each infeed operation (mm, inch)
P	Dwell time at the Z-point level (bottom of the hole) (see)
F	Thread pitch (mm/rev, inch/rev)
E	Feedrate multiplication ratio for retraction to the R-point level
S	Spindle speed (r/min)
L	Number of repetitions

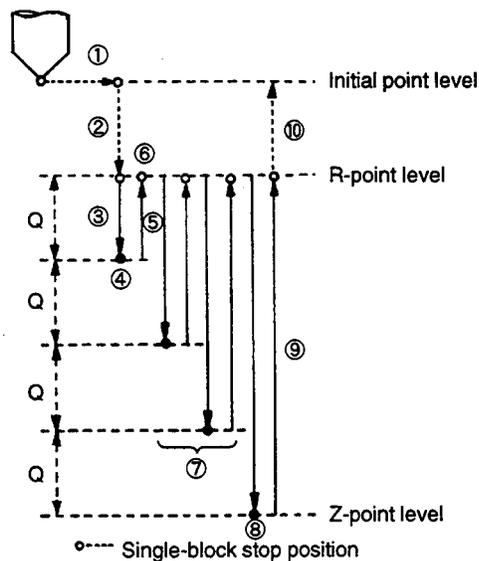


FIGURE 4.1.4.1 Tapping Cycle with Fixed Pitch Command

Operation (G184 mode)

- ① Positioning at the hole position specified by X, Y
- ② Positioning at the R-point level
- ③ Starting the spindle in the forward direction; Z-axis feed by the distance specified by Q at the specified feedrate
- ④ Stopping the spindle
- ⑤ Starting the spindle in the reverse direction; Z-axis retraction to the R-point level at the specified feedrate
- ⑥ Stopping the spindle
- ⑦ Repeats steps ③ to ⑥ by changing tapping feed distance to “Q x n”.
- ⑧ When the Z-point level is reached, the spindle stops and dwell is executed (if a P command is specified).
- ⑨ Starting the spindle in the reverse direction; Z-axis retraction to the R-point level at the specified feedrate
- ⑩ In the G98 mode: Positioning at the initial point level
 In the G98 mode with a W command: Positioning at the W-point level
 In the G99 mode: Cycle completes at the R-point level.

- Variable pitch command

Example of Programming

G93;

G184 (or G174) X±··Y±··Z±··R±··W±··I··J··K··P··F··E··S··L··;
X··Y····;

G94;

G93/G94	Solid tap mode ON, G94: Solid tap mode OFF
G184/G174	Deep-hole solid tap, G174 Reverse deep-hole solid tap
X, Y	Hole position (mm, inch)
Z	Distance from the R-point level to the bottom of the hole, or the position of the hole bottom (mm, inch)
R	R-point level (mm, inch)
W	W-point return level (mm, inch)
I	Initial value of feed distance (mm, inch)
J	Decrement value of feed distance (mm, inch)
K	Last value of feed distance (mm, inch)
P	Dwell time at the Z-point level (bottom of the hole) (see)
F	Thread pitch (mm/rev, inch/rev)
E	Feedrate multiplication ratio for retraction to the R-point level
S	Spindle speed (r/min)
L	Number of repetitions

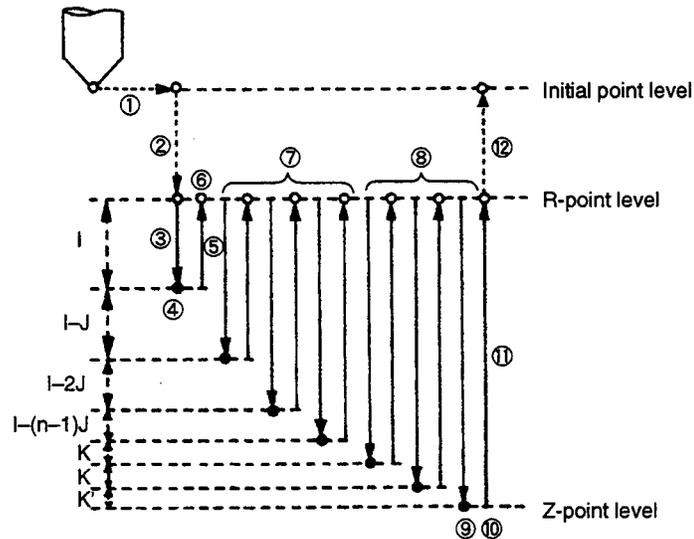


FIGURE 4.1.4.2 Tapping Cycle with Variable Pitch Command

Operation (G 184 mode)

- ① Positioning at the hole position specified by X, Y
- ② Positioning at the R-point level
- ③ Starting the spindle in the forward direction; Z-axis feed by the distance specified by I at the specified feedrate
- ④ Stopping the spindle
- ⑤ Starting the spindle in the reverse direction; Z-axis retraction to the R-point level at the specified feedrate
- ⑥ Stopping the spindle
- ⑦ Repeats steps ③ to ⑥ by increasing tapping feed distance by “I – (n – 1) J” from the previous feed distance until “Increased amount of feed distance £ K.

$$\begin{aligned}
 \text{Feed distance in the nth cycle} &= I + (I - J) + (I - 2J) + (I - 3J) \dots \\
 &\quad + (I - (n - 1) J) \\
 &= n \times I - \left(\sum_{i=1}^n (i - 1) \right) \times J \\
 &= n \times I - \frac{n(n - 1)}{2} \times J
 \end{aligned}$$

- ⑧ Infeed to the Z-point level by taking the feed distance that satisfies “Increased amount of feed distance < K” as value K.
- ⑨ When the Z-point level is reached, the spindle stops.

- ⑩ Dwell is executed according to the P command,
- ⑪ Starting the spindle in the reverse direction; Z-axis retraction to the R-point level at the specified feedrate
- ⑫ In the G98 mode: Positioning at the initial point level
 In the G98 mode with a W command: Positioning at the W-point level
 In the G99 mode: Cycle completes at the R-point level.



The tapping operation is executed in the fixed pitch cycle if I = 0, or there is no J command. If K = 0, or there is no K command, alarm 0162 “LACK OF ADDRESS” occurs.

(b) Pecking cycle type (type B)

The pecking cycle type includes the two types of commands such as fixed pitch command and variable pitch command.

- Fixed pitch command

Example of Programming

G93;

G184 (or G174) X±··Y±··Z±··R±··W±··Q··C··P··F··E··S··L··;
 X··Y··;

G94;

G93/G94	Solid tap mode ON, G94: Solid tap mode OFF
G184/G174	Deep-hole solid tap, G174 Reverse deep-hole solid tap
X, Y	Hole position (mm, inch)
Z	Distance from the R-point level to the bottom of the hole, or the position of the hole bottom (mm, inch)
R	R-point level (mm, inch)
W	W-point return level (mm, inch)
Q	Depth of feed in each infeed operation (mm, inch)
P	Dwell time at the Z-point level (bottom of the hole) (see)
F	Thread pitch (mm/rev, inch/rev)
E	Feedrate multiplication ratio for retraction to the R-point level
S	Spindle speed (r/min)
L	Number of repetitions
C	C-point level (mm, inch)

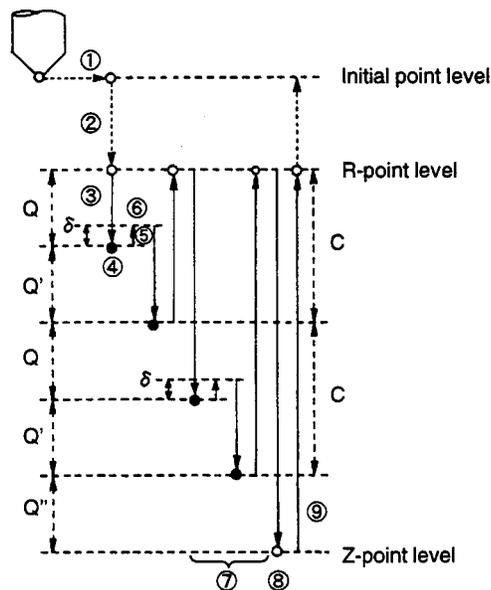


FIGURE 4.1.4.3 Tapping Cycle with Fixed Pitch Command

Operation (G184 mode)

- ① Positioning at the hole position specified by X, Y
- ② Positioning at the R-point level
- ③ Starting the spindle in the forward direction; Z-axis feed by the distance specified by Q at the specified feedrate
- ④ Stopping the spindle
- ⑤ Starting the spindle in the reverse direction; Z-axis retraction by “d” at the specified feedrate
- ⑥ Stopping the spindle
- ⑦ Starting the spindle in the forward direction; Z-axis feed by “Q + d” at the specified feedrate. This is repeated until the Z-position level is reached.
- ⑧ When the Z-point level is reached, the spindle stops and dwell is executed according to the P command.
- ⑨ Starting the spindle in the reverse direction; Z-axis retraction to the R-point level at the specified feedrate
- ⑩ In the G98 mode: Positioning at the initial point level
 In the G98 mode with a W command: Positioning at the W-point level
 In the G99 mode: Cycle completes at the R-point level.



1. If a C command is specified, the following operation is executed when the cut level reaches the C-point level.
 - The spindle stops and then starts rotating in the reverse direction; Z-axis retracts to the R-point level at the specified feedrate.
 - The spindle stops.
 - The spindle starts rotating in the forward direction; Z-axis is fed to “C + Q” level at the specified feedrate.
2. The E command is valid for the retraction operation from the C-point level to the R-point level and from the Z-point level to the R-point level. It is not valid for the Z-axis retraction by “d”.
 - Variable pitch command

Example of Programming

G93;

G184 (or G174) X±••Y±••Z±••R±••W±••I••J••K••C••P••F••E••S••L;
X••Y••;

G94;

G93/G94	Solid tap mode ON, G94: Solid tap mode OFF
G184/G174	Deep-hole solid tap, G174 Reverse deep-hole solid tap
X, Y	Hole position (mm, inch)
Z	Distance from the R-point level to the bottom of the hole, or the position of the hole bottom (mm, inch)
R	R-point level (mm, inch)
W	W-point return level (mm, inch)
I	Initial value of feed distance (mm, inch)
J	Decrement value of feed distance (mm, inch)
K	Last value of feed distance (mm, inch)
P	Dwell time at the Z-point level (bottom of the hole) (see)
F	Thread pitch (mm/rev, inch/rev)
E	Feedrate multiplication ratio for retraction to the R-point level
S	Spindle speed (r/min)
L	Number of repetitions
C	C-point level (mm, inch)

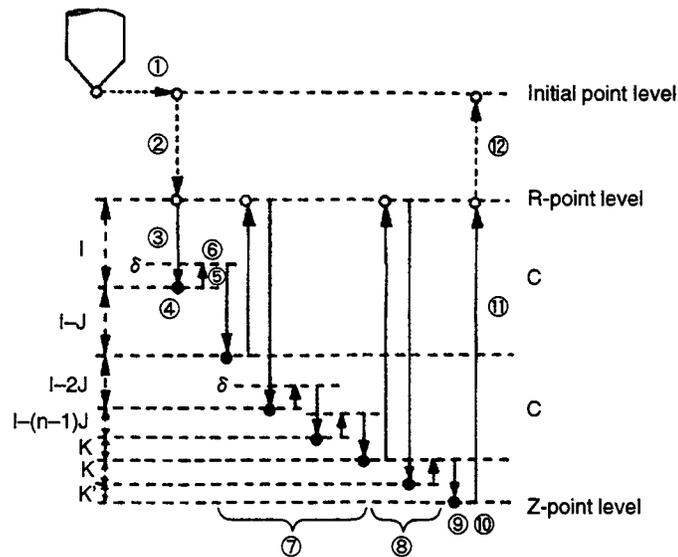


FIGURE 4.1.4.4 Tapping Cycle with Variable Pitch Command

Operation (G184 mode)

- ① Positioning at the hole position specified by X, Y
- ② Positioning at the R-point level
- ③ Starting the spindle in the forward direction; Z-axis feed by the distance specified by I at the specified feedrate
- ④ Stopping the spindle
- ⑤ Starting the spindle in the reverse direction; Z-axis retraction by “d” at the specified feedrate
- ⑥ Stopping the spindle
- ⑦ Repeats steps ③ to ⑥ by changing the feed distance to “I – (n – 1) J + d” until “I – (n – 1) J ≤ K” is satisfied.
- ⑧ If “Feed distance < K”, the value of “K + d” is taken as the feed distance and the cycle is repeated until the Z-position level is reached.
- ⑨ When the Z-point level is reached, the spindle stops.
- ⑩ Dwell is executed according to the P command.
- ⑪ Starting the spindle in the reverse direction; Z-axis retraction to the R-point level at the specified feedrate.
- ⑫ In the G98 mode: Positioning at the initial point level
 In the G98 mode with a W command: Positioning at the W-point level
 In the G99 mode: Cycle completes at the R-point level.



1. If a C command is specified, the following operation is executed when the cut level reaches the C-point level.
 - The spindle stops and then starts rotating in the reverse direction; Z-axis retracts to the R-point level at the specified feedrate.
 - The spindle stops.
 - The spindle starts rotating in the forward direction; Z-axis is fed to “C + I + (n – 1) J” level at the specified feedrate.
2. The E command is valid for the retraction operation from the C-point level to the R-point level and from the Z-point level to the R-point level. It is not valid for the Z-axis retraction by “d”.
3. The tapping operation is executed in the fixed pitch cycle if J = 0, or there is no J command.

If K = 0, or there is no K command, alarm 0162 “LACK OF ADDRESS” occurs.

(3) Supplements to the Deep-hole Solid Tapping Function

- If the G184/G174 command is specified in the G94 mode, the solid tapping is executed in the G84/G74 mode using the addresses of X, Y, Z, F, S, R, and P.
- In the G184/G174 mode, feed override and spindle override are fixed at 100%. Note that rapid traverse override is valid.
- During the execution of a tapping cycle, single-block stop is not executed and only the cycle is suspended at the R-point level. Concerning the feed-hold stop, the points where the cycle execution is suspended differ depending on the selected type of tapping cycle.

Type A (R-point return type)	Feed-hold stop is not executed during tapping. If the FEED HOLD switch is pressed during the execution of a tapping cycle, the operation stops after the completion of chamfering. (The same position as the single-block stop position.)
Type B (pecking cycle type)	If the feed hold switch is pressed during tapping, Z-axis retracts to the R-point level instead of retraction by “δ”. If the feed hold switch is pressed during the retraction by “δ”, Z-axis retracts to the R-point level after the completion of the next feed. After the operation is suspended due to feed hold, the remaining tapping cycle is executed by restart operation.

4.1.5 Circle Cutting Function (G12, G13)

The circle cutting function allows a series of circle cutting operation to be executed by the commands specified in a single block.

(1) Programming Format

The following programming format is used to specify circle cutting.

G12 (or G13) I...D...F...;

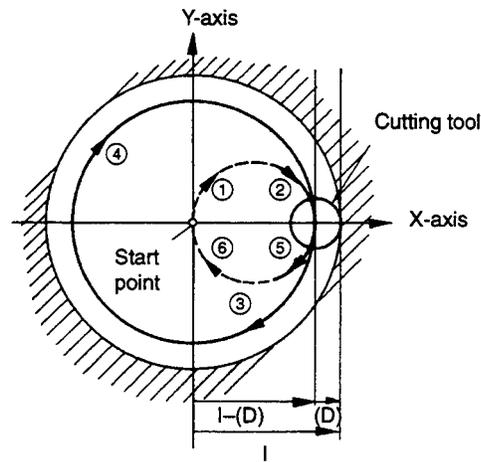


FIGURE 4.1.5.1 Cutting Commands (G12, G13)

(a) Description of the addresses

- High-speed feed interval designation R

By specifying “G12 (or G13) I...R...D...F...;”, circle cutting is executed in the manner shown in Fig. 4.21. The interval in which a cutting tool is moved at a high speed can be specified by R. In Fig. 4.1.5.2, the intervals $\overset{\circ}{A}$ and $\overset{\circ}{U}$ indicate the high-speed feed intervals.

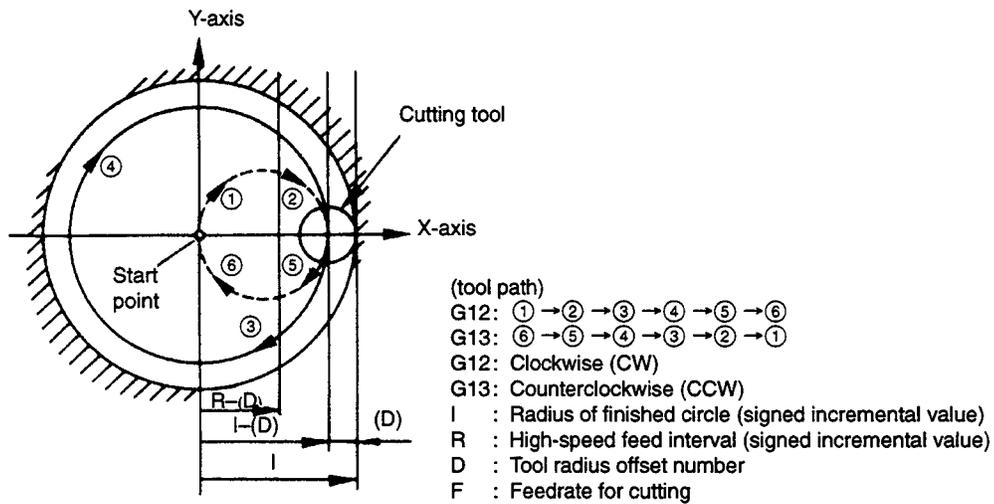


FIGURE 4.1.5.2 High-speed Feed Interval Designation by R

- Designation of circle section repetition by L
 By specifying “G12 (or G13) I ••• D ••• L ••• F •••;”, circle cutting is executed by repeating cutting at the circle section by L times.
- Spiral circle cutting designation by Q and K
 By specifying “G12 (or G13) I ••• D ••• K ••• Q ••• F •••;”, the cycle to cut a circle by controlling the spiral cutting tool path shown in Fig. 4.1.5.3 is defined.

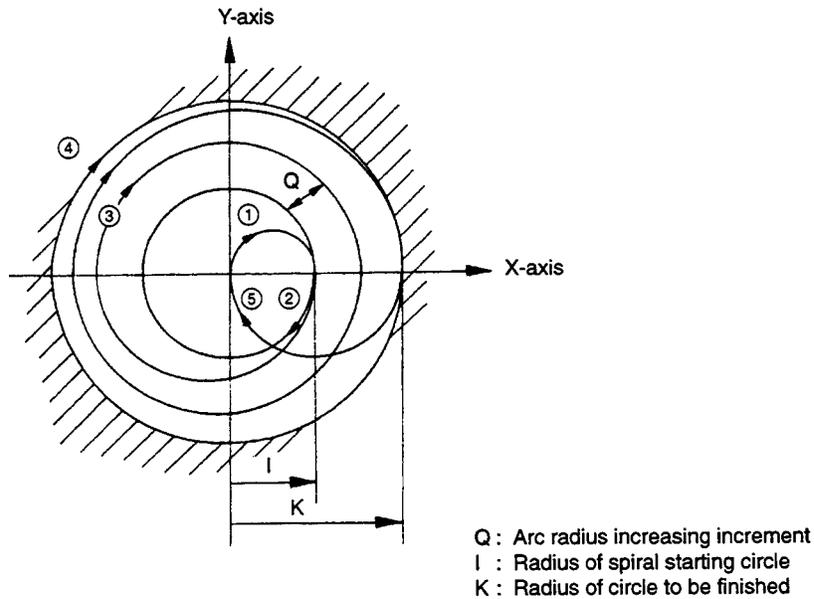


FIGURE 4.1.5.3 Spiral Circle Cut Designation by Q and K

- Combination of designation of rapid traverse section, repeat command, and circle cutting in spiral

By specifying “G12 (or G13) I...R... (or J...) K...Q...L...F...;”, it is possible to combine the addresses indicated above.

- High-speed feed interval automatic calculation designation by J

By specifying “G12 (or G13) I...J...D...F...;”, the high-speed feed intervals can be calculated automatically. To show how the spiral path is generated, the illustration is given assuming tool radius (D) of 0.

Example of Programming

```
G00 Z-40.;
```

```
G13 I-50.J-7. D15 F300;
```

```
G00 Z40.;
```

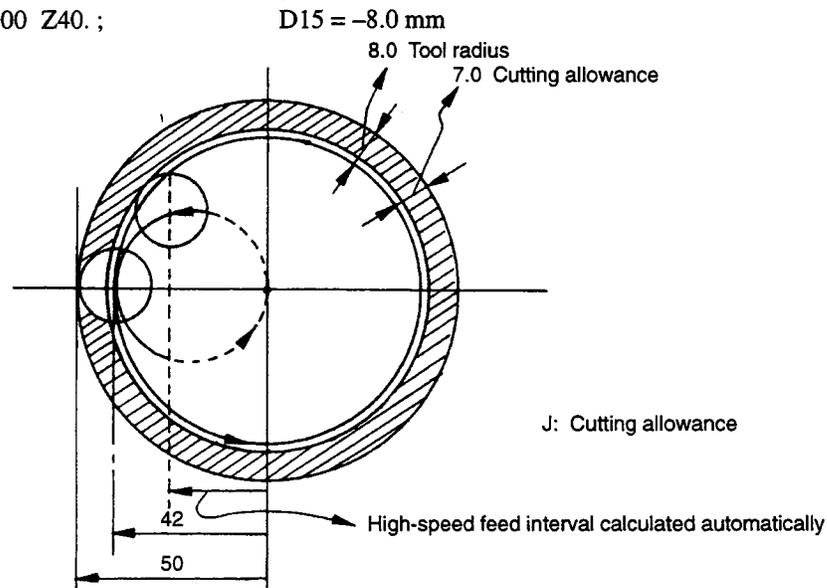


FIGURE 4.1.5.4 High-speed Feed Interval Automatic Calculation Designation by J

(2) Notes and Supplements to the Circle Cutting Function

- Circle cutting is possible only in the XY plane. If circle cutting is specified in a plane other than the XY plane, alarm “0361” occurs.
- The feedrate to be selected for axis move in the high-speed feed intervals is set for parameter pm2862. It is possible to override this feedrate. However, when the DRY RUN switch is ON, the dry run feedrate is used.
- In the circle cutting mode (G12, G 13), the tool paths are generated by compensating for the tool radius independent of the designation for the tool radius offset function called up by G41 and G42.

G12 and G13 must be specified in the G40 (tool radius offset cancel) mode. If these G commands are specified in other than G40 mode, alarm “0182” occurs.

- Movements in the “X-” direction (symmetrical to Y-axis) are possible by changing the signs of I, J, K, and D. In the example as shown in Fig. 4.1.5.5, a negative value should be assigned for I, R, (and D). Note that cutting is not possible by Y-axis movements.

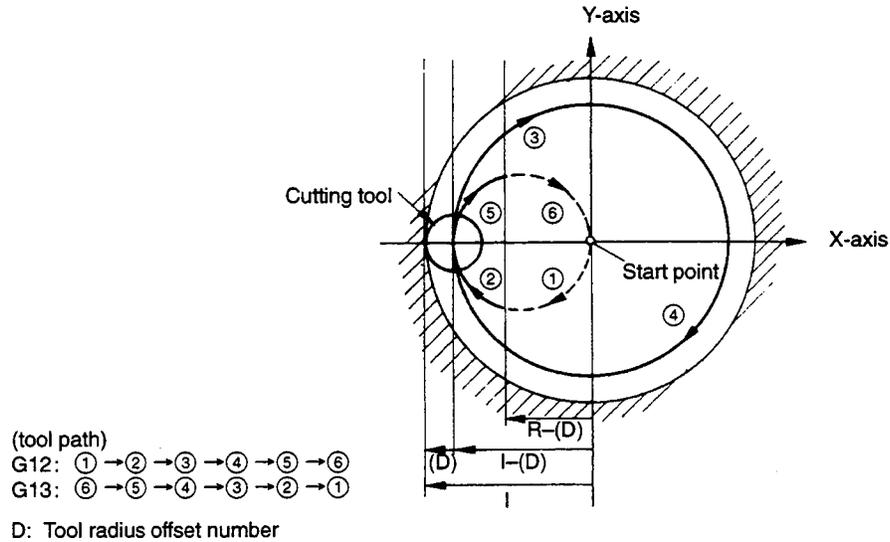


FIGURE 4.1.5.5 Movements in the “X-” Direction (Symmetrical to Y-axis)

- I, J, K, R, Q, and L specified in the circle cutting block are all valid only in the specified block. Among these addresses, at least I must be specified. If none of these addresses is specified, alarm “0362” occurs.
- The restriction indicated below applies for I (radius of circle to be finished) and R (high-speed feed interval). If this restriction is not satisfied, high-speed feed is not executed.

$$|R-d| < |I-d|$$

Also note that high-speed feed is not possible if the signs of “R-d” and “I-d” differ from each other.

- Before specifying G12 or G13, specify the tool radius offset number D. If this is omitted, the offset motion is determined by the setting for a parameter.

pm4010 D1 = 0	Offset is applied using the D number specified previously.
pm4010 D1 = 1	Offset is not applied.

4.1.6 Mirror Image ON/OFF (M94, M95) *

The mirror image function can be turned ON and OFF by specifying the following M codes in a program.

Table 4.1.6.1 Mirror Image ON/OFF M Codes.

M Code	Function
M94	Mirror image OFF
M95	Mirror image ON

M94 and M95 are modal. When the power is turned ON and when the NC is reset, M94 is set. M94 and M95 must be specified in a block without other commands.

(1) Specifying the Mirror Image Application Axes

The axes for which the mirror image is applied are set using the parameters pm0002 D0 to pm0003 D0 or by using the mirror image axis setting switches on the machine operation panel.

The mirror image axis setting switches are used to set the axes for which symmetrical machining should be executed. Set parameter “pm5001 D1 = 1” and turn ON the switches corresponding to the axes. For the designated axes, the mirror image function is applied from the block next to the one in which M95 is specified; mirror image remains valid until M94 is specified. Note that the mirror image application axes must not be changed in the M95 mode.

(2) Mirror Image ON (M95)

Mirror image is applied to the specified axes from the block next to the one in which M95 is specified. In the mirror image mode, the specified axes move in the direction opposite to the direction specified in the program.

The mirror image function is executed in the same manner independent of the dimensioning mode (absolute or incremental). The position at which M95 is specified is taken as the mirror-point,

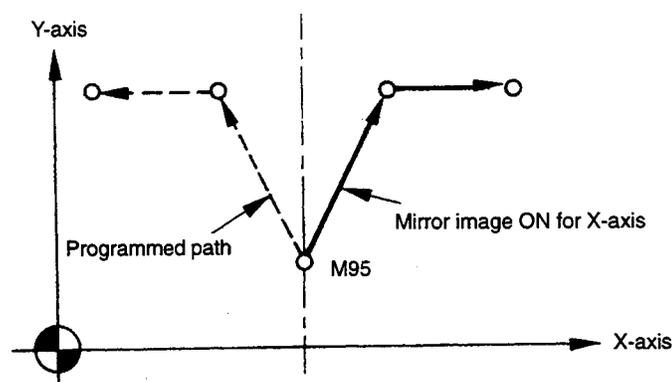


FIGURE 4.1.6.1 Mirror Image ON (M95)

(3) Mirror Image OFF (M94)

For the blocks after the block in which M94 is specified, mirror image is canceled.

The mirror image function must be called up and canceled at the same point. If mirror image is canceled at a point different from the point where mirror image has been called up, axis movements after the cancellation of mirror image are offset from the programmed paths by the amount equivalent to the difference between these two points.

Example of Programming

```

N01 G92 X0 Y0 ;
N02 M95 ; ← Mirror image ON
N03 G90 G01 X40. Y40. F300 ;
N04 X120. ;
N05 Y100. ;
N06 X40. ;
N07 Y40. ;
N08 X0 Y0 ; ← If "M94;" is entered here, the commands in the following
N09 M94 ; ← Mirror image canceled correctly blocks are executed in the manner as shown by dotted lines.
N10 X20. Y-40. ;
N11 X60. ;
    
```

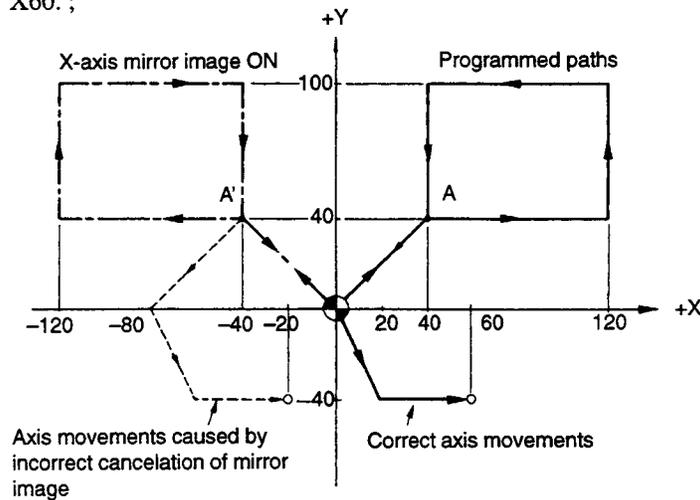


FIGURE 4.1.6.2 Disagreement between Mirror Image Call Up and Cancel Positions

(4) Mirror Image External Input Function

In addition to the conventional mirror image function, it is possible to set the mirror image execution mode when the power is turned ON or the NC is reset by appropriate parameter setting.

If the parameter setting is made so that the M95 mode is set when the power is turned ON or when the NC is reset, it is possible to select whether or not the mirror image function is called up at the intermediate positioning point for G28 operation.

The following indicates how this function should be used:

- When the power is turned ON or when the NC is reset.

pm4000 D1 = 0	M94 mode (mirror image OFF)
pm4000 D1 = 1	M95 mode (mirror image ON)

To specify G28 or G29 with the parameter setting of “pm4001 D0 = 0”, it is necessary to turn OFF the mirror image function by specifying M94. If G28 or G29 is specified without turning OFF the mirror image function, alarm “0127” occurs.

- When the M95 mode is set at power ON (pm4001 D0 = 1)

pm4001 D2 = 0	Mirror image is applied to the intermediate positioning point of G28.
pm4001 D2 = 1	Mirror image is not applied to the intermediate positioning point of G28.

Example of Programming

```
G90 G00 X0 Y0 ;
G00 X100. Y50. ;
G28 X40. Y100. ;
```

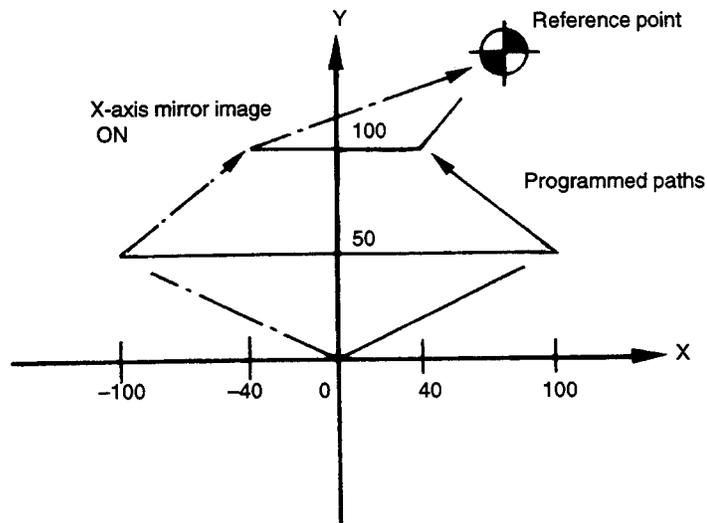


FIGURE 4.1.6.3 Example of Mirror Image Application (X-axis) to G28 Intermediate Positioning Point

(5) Supplements to the Mirror Image ON/OFF Function

- Before specifying G28 or G29 to execute tool change or to end machining, make sure to turn OFF the mirror image function by specifying M94. If G28 or G29 is specified without canceling the mirror image function, alarm “0127” occurs.
- The mirror image function is invalid for offset distance used by the tool length offset function. In the mirror image mode, the present position data (POS) display the actual tool movements.
- Using the mirror image function with the workpiece coordinate rotation, scaling, and/or coordinate rotation function

The functions must be specified in a predetermined order. If they are specified by disregarding the predetermined order, alarm “0285” occurs.

Workpiece coordinate rotation (G54) > Mirror image (M94) >
Scaling (G51) > Coordinate rotation (G68)

- If the scaling function is called up in the mirror image mode, it is applied to the center point of scaling.
- If the coordinate rotation function is called up in the mirror image mode, it is applied to the center of rotation and also in the direction of rotation.
- If the axes for which mirror image is applied are specified using the M code, buffering of the M codes which set and reset the axes must be disabled by the setting for appropriate parameters (pm4400 to pm4409).

4.1.7 Programmable Data Input (G10) *

By using the G10 command, it is possible to write the tool offset amount and the work coordinate system.

(1) Setting the Tool Offset Amount

The tool offset amount can be set by using the specially provided screen. By using a program, the tool offset amount is set in either of the following two types (H and D used in common, and H and D used independently, and tool wear offset function provided/not provided).

(a) When H and D are used in common

By specifying “G10 L10P ••• R 00 •••; (P: tool offset number, R: tool offset amount, L10: omissible)”, the tool offset data of the specified tool offset number can be changed to the required amount.

- If G10 is executed in the G90 mode, the value set for R is saved as the tool offset amount.
- If G10 is executed in the G91 mode, the value set for R is added to the existing offset amount.
- Tool wear offset function provided.

- Tool geometry offset can be changed by the following format:
G10 L10 P...R...;
- Tool wear offset can be changed by the following format:
G10 L11 P...R...;

(b) H and D used for different offset data

The format used to write the offset amount differs H code and D code.

- For H code
G10 L10P...R...; (L10: omissible)
- For D code
G10 L12...P...R...;
or
G10 Q1P...R...;
- Tool wear offset provided
G10 L10 P...R...;Tool geometry offset at H code
G10 L11 P...R...;Tool wear offset at H code
G10 L12 P...R...;Tool geometry offset at D code
G10 L13 P...R...;Tool wear offset at D code

(2) Changing Workpiece Coordinate System *

The workpiece coordinate system shift amount is set for G54 to G59 by using the specially provided screen. By using a program, the workpiece coordinate system shift distance is set in the following manner.

G10 Q2 Pm Jn X...Y...Z''...(*a...b...);

or

G10 L2 Pm Jn X...Y...Z''...(*a...b...);

- Q2 and L2 are used to discriminate the setting of the workpiece coordinate system shift distance from the setting of the tool offset amount.
- Pm (m: 1 to 6) corresponds to the workpiece coordinate system number (m) to be set.

Without P or P1	G54
P2	G55
P3	G56
P4	G57

P5	G58
P6	G59

- Jn (n: 1 to 27) corresponds to the expanded workpiece coordinate systems.

Without J, J0 or J1	G54 J1
J2	G54 J2
•	•
•	•
•	•
J27	G54 J27

The programmable data input function is summarized in the table below.

Table 4.1.7.1 Programmable Data Input Format

Type			Format				
Offset	Tool wear offset not provided	H/D used in common	G10 L10 G10	P•••R•••; P•••R•••;			
		H/D used for different offset functions	H	G10 L10 G10	P•••R•••; P•••R•••;		
			D	G10 L12 G10 Q1	P•••R•••; P•••R•••;		
	Tool wear offset provided	H/D used in common	Geometry offset	G10 L10 G10	P•••R•••; P•••R•••;		
			Wear offset	G10 L11	P•••R•••;		
		H/D used for different offset functions	H	Geometry offset	G10 L10 G10	P•••R•••; P•••R•••;	
				Wear offset	G10 L11	P•••R•••;	
		D	Geometry offset	G10 L12 G10 Q1	P•••R•••; P•••R•••;		
			Wear offset	G10 L13	P•••R•••;		
		Work coordinate system shift distance	6 pairs	G10 Q2 distance; G10 L2 distance;	P1 - P6 (without J, J0, J1)Axis shift distance; P1 - P6 (without J, J0, J1)Axis shift distance;		
54 pairs	G10 Q2 G10 L2		P1 - P6 J1 - J9Axis shift distance; P1 - P6 J1 - J9Axis shift distance;				
162 pairs	G10 Q2 G10 L2		P1 - P6 J1 - J27Axis shift distance; P1 - P6 J1 - J27Axis shift distance;				

Note: When the workpiece coordinate system shift distance is changed, the specified value is saved as it is independent of the G90 or G91 mode.

4.1.8 Subprogram Call Up Function (M98, M99)

This function can be used when subprograms are stored in the part program memory. Sub-programs registered to the memory with program numbers assigned early be called up and executed as many times as required.

The created subprograms should be stored in the part program memory before they are called up.

```

O ..... ? Program number
.....;
.....;
.....;
M99;      ? Program number

```

(1) Commands

The M codes indicated in Table 4.19 are used.

Table 4.1.8.1 Subprogram Call M Code

M Code	Function
M98	Subprogram call up
M99	End of subprogram

(a) Subprogram call (M98)

- By specifying “M98P...L...;”, the subprogram of the program number that is specified by P is called up and executed by the number specified by L. If L is omitted, the subprogram is executed once. If the specified program number is not found, alarm “0390” occurs.
- Nesting of subprograms is possible - the allowable nesting level is four. If the nesting level exceeds this limit, an alarm occurs.
- By specifying “M98P...L...Q...;” the subprogram specified by the P command is executed from the block specified by the Q command. A sequence number must be specified by a numeral of up to five digits.

(b) End of Subprogram Code (M99)

At the end of a subprogram, M99 must be specified in a block without other commands. Upon execution of M99, the program automatically returns to the block in the main program next to the one where the subprogram has been called up. Fig. 4.1.8.1 shows how a subprogram called up from the main program is executed.

Example of Programming

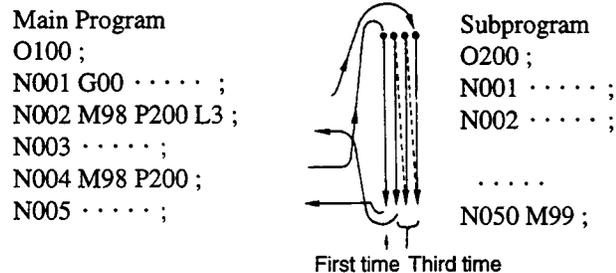


FIGURE 4.1.8.1 Execution of A Subprogram

- By specifying “M99P;”, the program returns to the block specified by the P command. in the main program instead of the block next to the one where the subprogram has been called up.
- If M99 is specified in a main program, the program returns to the beginning of that main programming and the program is repeatedly executed.

4.2 PROGRAM SUPPORT FUNCTIONS (2)

4.2.1 Program Copy (G25)*

By specifying “G25 P (p1 p2) ••• Q (q1 q2) ••• L •••;”, the programs can be repeatedly executed by L times in the range from the block of sequence number “q1” in the program of program number “p1” to the block of sequence number “q2” in the program of program number “p2”.

Table 4.2.1.1 Description of Addresses

Address		Description
P	p1	Program number of copy start program (5-digit O number)
	p2	Copy start sequence number (4-digit N number)
Q	q1	Program number of copy end program (5-digit O number) (p 1 = q1)
	q2	Copy end sequence number (4-digit N number)
L		Number of repetitions (omission: L1)



1. For both “p1” and “q1” commands, leading zeros can be omitted. For “p2” and “q2”, omission of leading zeros is not allowed. If more than four digits are specified for “p2” or “q2”, the four digits from the least significant digit are taken for a sequence number.
2. The O number (q1) to be specified later can be omitted. In this case, the same O number as specified for “p1” is assumed.
3. The sequence numbers specified for “p2” and “q2” are searched from the first block of the specified program. Therefore, the sequence numbers used for the objective block for search must not be assigned to more than one block. If there are two or more blocks that are assigned the same sequence number, the block appearing first is searched.
4. If the specified block or program cannot be searched, an alarm occurs; alarm “0390” if “p1” cannot be searched, alarm “0393” if “p2” or “q2” cannot be searched, and alarm “0260” if none of “p1”, “p2” and “q2” can be searched.

G25 can be executed in the tape, MDI, or memory mode operation. However, for the execution of G25, it is necessary to store the program to be copied to the memory. If more than one block is assigned the same program number, the block appearing first is valid for search operation.

(1) Program Copy Nesting

It is allowed to specify G25 in the program that is copied. This is called program copy nesting and up to four levels of program nesting are allowed. If the nesting level exceeds four, or the total nesting (program copy nesting and subprogram call nesting) exceeds four, alarm “0261” occurs.

If G25 is used with macro program call up (G65, G66), program copy nesting is possible for up to four levels independent of macro program call up.

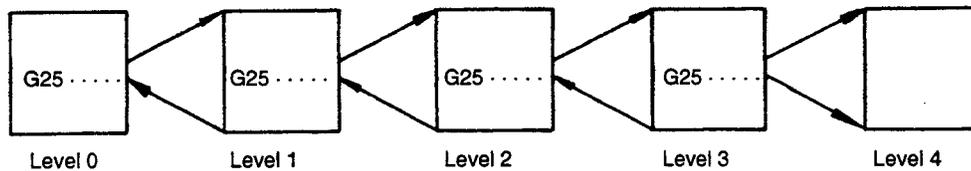
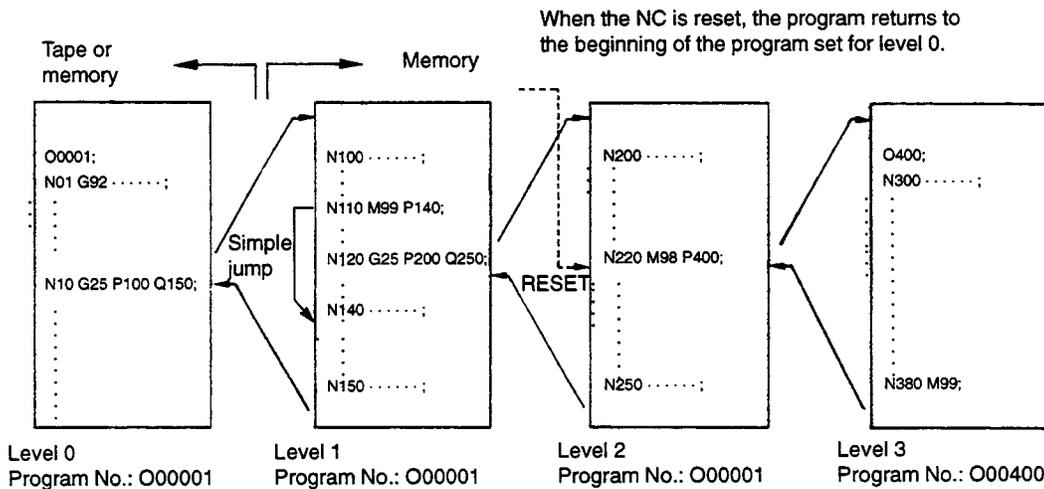


FIGURE 4.2.1.1 Program Copy Nesting

(a) Example 1 (copying in the same program)

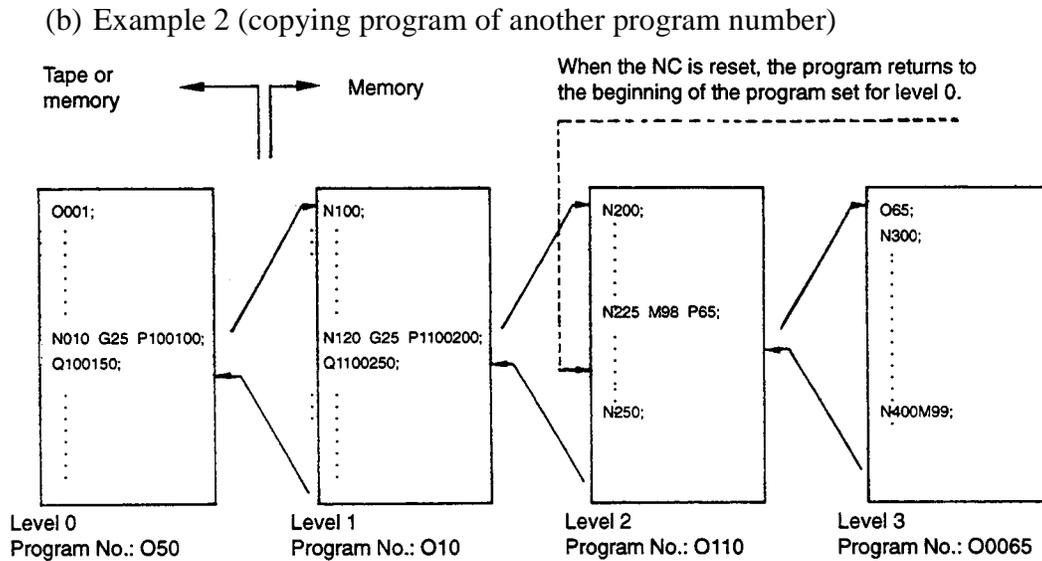
Designation of a program number is not necessary when copying in a program in the same program.



Note: To give G25 in tape, the program number must always be specified with P.

(Example) N10 G25 P100100 Q150;
 ↑
 Program Number

FIGURE 4.2.1.2 Copying in the Same Program Number



Note: If M99 is specified to execute jump to another level, program enters endless loop and cannot exit this loop.

Note: If M99 is specified to execute jump to another level, program enters endless loop and cannot exit this loop.

FIGURE 4.2.1.3 Copying Another Program Number

(2) Supplements to the Program Copy Function

- If the NC is forcibly reset due to the occurrence of an error or M30 or M02 is executed to reset the NC during the execution of G25, the program returns to the beginning of the program in level 0.
- If a numeral of four digits or less is specified for P or Q, it is regarded as a sequence number and the sequence number search is executed in the program of the present program number.
- If M98 is specified in the G25 block, or if M99 is specified in the block designated by “p2” and “q2”, alarm “0602” occurs.
- If M99 is specified in a block in the range of blocks beginning with “p2” block and ending with “q2” block, alarm “0264” occurs.
- When calling up the program copy function in the canned cycle mode, it is not allowed to specify the program copy commands with the canned cycle call up command in the same block. If specified, the alarm “0170” occurs.
- The program copy commands can be specified in the canned cycle mode (G70 to G72, G73, G76, G81 to G89).

4.2.2 Automatic Corner Override (G106) *

When cutting the inside of a workpiece, load applied to the cutting tool increases gradually during the cutting at corners since the area of the cutting tool that is in contact with the workpiece increases as the tool moves near the corner.

The automatic corner override function automatically changes the cutting feedrate of the cutting tool at a point where load applied to the cutting tool changes. The function calculates such a point from the allowance specified in the program and the radius of the cutting tool presently used and overrides the programmed feedrate. The override values are set for parameters.

(1) Programming Format

G106 G01 X... Y... I (or J) ... F...;

I: Allowance in the X-axis direction

J: Allowance in the Y-axis direction

(I>0 and J>0 must be satisfied.)

With the commands indicated above, automatic corner override is executed. In the G 106 block, axis commands of other than X- and Y-axis are not allowed. If an axis other than X- or Y-axis is specified or none of X- or Y-axis is specified in the G 106 block, an alarm occurs.

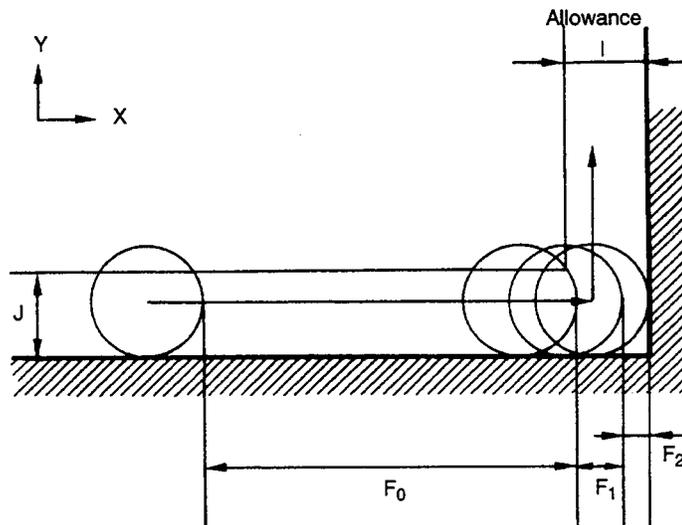


FIGURE 4.2.2.1 Automatic Corner Override

(2) Parameters

Override values are set for parameters for each of the override steps. If allowance is greater than the tool radius, the value for pm2120 is always assumed to be 100.

Table 4.2.2.1 Automatic Corner override Parameters

Feedrate	Parameter	Description
F1	pm2120	Override value applied to the range where an axis moved at F ₁ (1 to 100%)
F2	pm2121	Override value applied to the range where an axis moved at F ₂ (1 to 100%)

The override values must satisfy the following conditions.

- Setting range is from 1 to 100 (%). If a value greater than 100 is set, such a value is disregarded and replaced with 100 (clamped at 100). If “0” is set, override is disregarded.
- The values for F₀, F₁, and F₂ must be “F₀ > F₁ > F₂”. If the value of F₂ is greater than the value of F₁, the value of F₂ is replaced with the value of F₁.

(3) Operation

As explained above, the function calculates the point where load is applied to the cutting tool from the radius of tool and the allowance. For this calculation, the following three cases are taken into account according to the position where load is applied.

- Tool radius is not greater than 1/2 of allowance ($D \leq I (J)/2$)
- Tool radius is greater than the allowance ($D > I (J)$)
- Tool radius is greater than a half the allowance and not greater than the allowance ($I (J)/2 < D \leq I (J)$)

Note that corner override is valid only in one axis direction. If both X- and Y-axis are specified, the direction in which override is applied is determined from the specification of allowance (I, J).

The allowance (I, J) must always be specified. If neither of I or J is specified, or both I and J are specified, an alarm occurs.

(a) Tool radius is not greater than 1/2 of allowance

In this case, override is: not applied independent of the setting for parameters and feed-rate is clamped at F_0 .

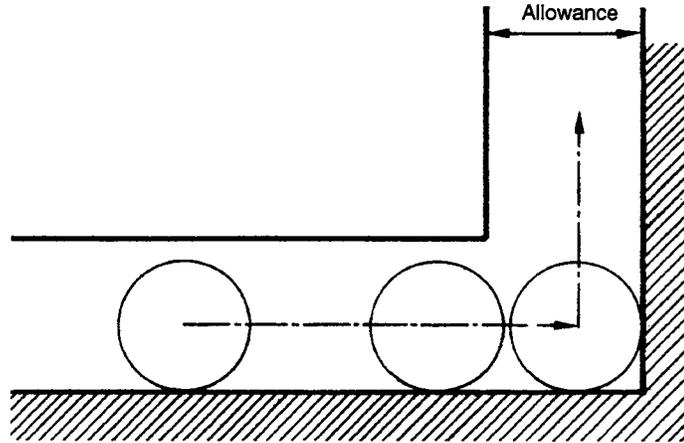
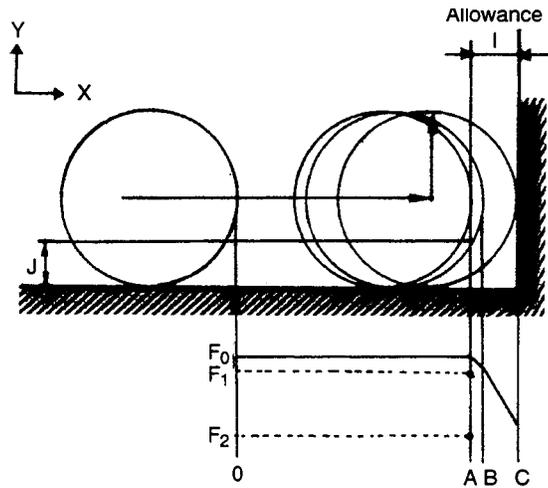


FIGURE 4.2.2.2 Automatic Corner Override – Tool Radius \leq Allowance/2

(b) Tool radius is greater than allowance



$$F_0 = (\text{Programmed feedrate}) \times (\text{Override \%})$$

$$F_1 = F_0 \times (\text{pm2120})$$

$$F_2 = F_0 \times (\text{pm2121})$$

FIGURE 4.2.2.3 Automatic Corner Override – Tool Radius $>$ l/J (Allowance)

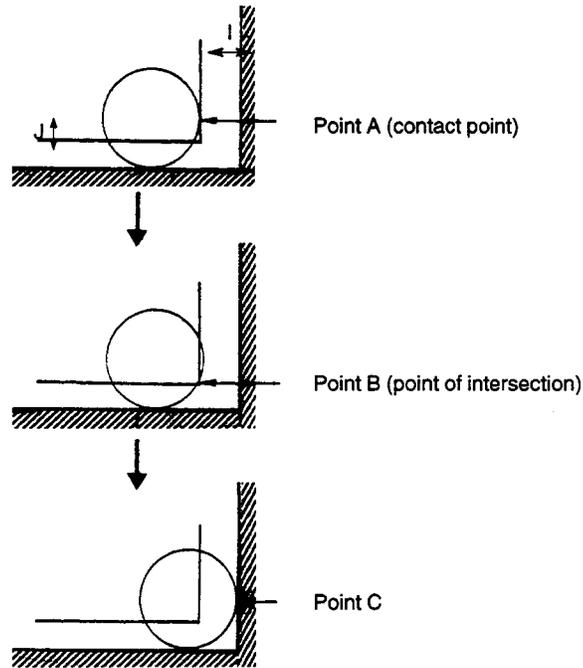


FIGURE 4.2.2.4 Feedrate Change Points

(c) Tool radius is greater than a half the allowance and not greater than allowance.

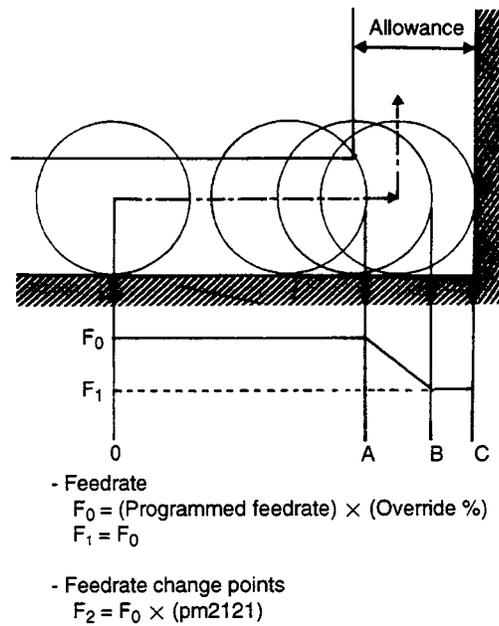


FIGURE 4.2.2.5 Automatic Corner Override - $I (\text{Allowance})/2 < \text{Tool Radius} \leq I (\text{Allowance})$

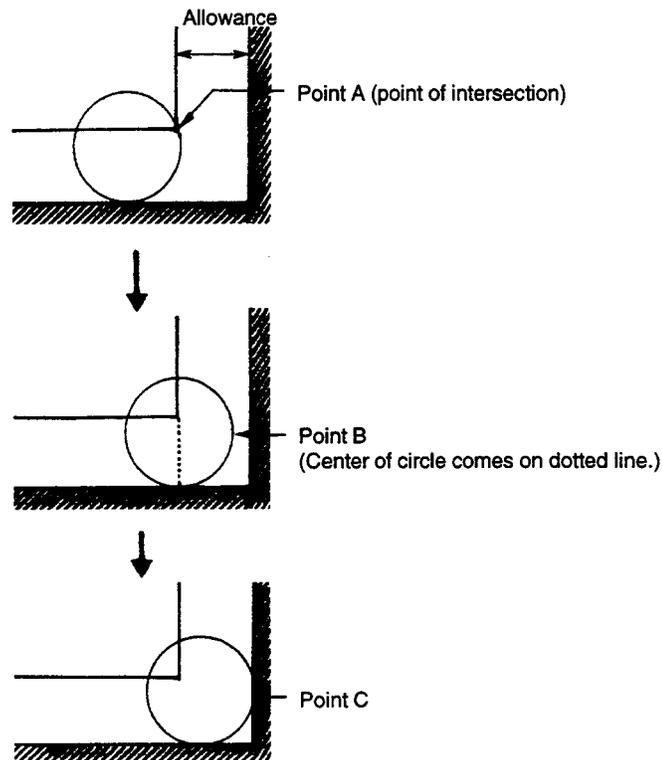


FIGURE 4.2.2.6 Feedrate Change Points

(4) Supplements to the Automatic Corner Override

- When G106 is specified, the G code mode set by 01-group G code must be G01.
If a G code other than G01 is specified, an alarm occurs.
- It is not allowed to specify G106 in the canned cycle mode. To specify G 106, it is necessary to cancel the canned cycle mode.
- If I and J address characters are specified with minus sign, and alarm occurs.

4.2.3 Stored Stroke Limit B and C (G22, G23) *

The stored stroke limit function checks whether the present position of axes operated manually or automatically enters the stored stroke limit (entry prohibited area) which is set by G22. The No. 2 entry prohibited area is called the stored stroke limit B and the No.3 to No.5 entry prohibited area the stored stroke limit C. If an axis has entered the stroke end limit, operation is stopped and alarm occurs.

(1) Programming Format

G22 X... Y... Z... I... J... K... P...;

↑

(Upper limit of points
A2 - A5)

↑

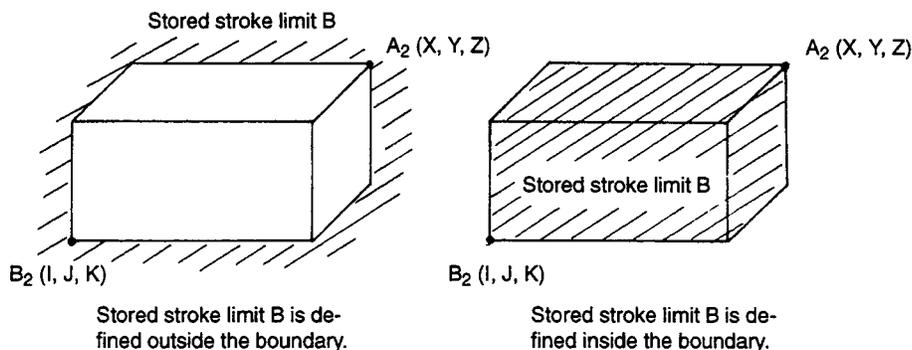
(Lower limit of points
B2 - B5)

↑

(Area designation: No. 2 - No. 5)

With the commands indicated above, the function checks the entry of axes into the entry prohibited area specified by P.

The commands of "G23 P...;" cancel the function to check entry into the prohibited area specified by P.



Note: If P is Omitted, P2 is assumed.

FIGURE 4.2.3.1 Stored Stroke Limit B, C

(a) Setting the boundary and area

The parameter numbers used for area designation are indicated in Table 4.2.3.1

Table 4.2.3.1 Parameter Numbers Used for Area Designation.

Boundary	Check Axis No.		
	No. 1	No. 2	No. 3
No. 2 prohibited area (+): A2 point	pm0831	pm0832	pm0833
No. 2 prohibited area (-): B2 point	pm0834	pm0835	pm0836
No. 3 prohibited area (+): A3 point	pm0837	pm0838	pm0839
No. 3 prohibited area (-): B3 point	pm0840	pm0841	pm0842
No. 4 prohibited area (+): A4 point	pm0843	pm0844	pm0845
No.4 prohibited area (-): B4 point	pm0846	pm0847	pm0848
No. 5 prohibited area (+): A5 point	pm0849	pm0850	pm0851
No. 5 prohibited area (-): B5 point	pm0852	pm0853	pm0854

(b) Designation of check axes

The axes for which stored stroke limit B, C (No. 2 to No. 5 entry prohibited areas) is checked is designated by using parameters (maximum of three axes).

Table 4.2.3.2 Stored Stroke Limit Check Axis Numbers for No. 2 to No. 5 Entry Prohibit Areas

Boundary	Check Axis No.		
	No. 1	No. 2	No. 3
No. 2 entry prohibited area	pm6111	pm6112	pm6113
No. 3 entry prohibited area	pm6114	pm6115	pm6116
No. 4 entry prohibited area	pm6117	pm6118	pm6119
No. 5 entry prohibited area	pm6120	pm6121	pm6122

Note: Setting: 1 = X-axis, 2 = Y-axis, 3 = Z-axis

(c) Parameters used for setting the entry prohibited area outside/inside of the boundary

Table 4.2.3.3 indicates the parameter numbers used for setting the entry prohibited area outside or inside the boundary.

Table 4.2.3.3 Outside/Inside Designation of Entry Prohibited Area (No. 2 to No. 5)

Check Area	Parameter No.	Description
No. 2 entry prohibited area	pm0008 D4	0: Inside the specified area 1: Outside the specified area
No. 3 entry prohibited area	pm0008 D5	
No. 4 entry prohibited area	pm0008 D6	
No. 5 entry prohibited area	pm0008 D7	

- Whether the boundary is included or not into the stored stroke limit is determined as indicated below.

Table 4.2.3.4

Entry prohibited Area	Boundary	Include/Not Included
Prohibited Area Defined Outside	Positive direction	Included
	Negative direction	Not included
Prohibited Area Defined Inside	Positive direction	Not included
	Negative direction	Included



FIGURE 4.2.3.2 Definition of Whether Boundary Is Included in Entry Prohibited Area

(d) Turning ON/OFF the Stored Stroke Limit Check

Whether or not the entry to No. 2 to No. 5 entry prohibited area should be checked can be designated by the setting for the parameters or by using appropriate input signals. If input signals are used to set, only stored stroke limit C can be turned ON/OFF.

If parameters are used for this setting, check for two entry prohibited area can be turned ON simultaneously in stored stroke limit B, C (No. 2 to No. 5 entry prohibited areas).

Table 4.2.3.5 Turning ON/OFF the Stored Stroke Limit C

Check Area	Parameter No.	Description
No. 2 entry prohibited area	pm0008 D0	0: Invalid (OFF) 1: Valid (ON)
No. 3 entry prohibited area	pm0008 D1	
No. 4 entry prohibited area	pm0008 D2	
No. 5 entry prohibited area	pm0008 D3	

In addition to the setting for the parameters, it is possible to use the input signals to turn ON/OFF the stored stroke limit C. The check becomes valid as the OR condition between the setting by parameter and the input signal.

Table 4.2.3.6 Turning ON/OFF the Stored Stroke Limit C

#3012	Description
0 0	Stored stroke limit check invalid for No. 3 to No. 5 entry prohibited area
0 1	Stored stroke limit check valid for No. 3 entry prohibited area
1 0	Stored stroke limit check valid for No. 4 entry prohibited area
1 1	Stored stroke limit check valid for No. 5 entry prohibited area

(2) Stored Stroke Limit A

Stored stroke limit A is always executed independent of the axis types whether the objective axis is linear or rotary. However, for a rotary axis, parameters are provided to set whether the stored stroke limit check is executed or not. Usually, the setting is for “not checked”.

Table 4.2.3.7 Parameters for Setting Servo Axis Types

Axis Name	X	Y	Z	4	5
Parameter	pm6030 D6	pm6031 D6	pm6032 D6	pm6033 D6	pm6034 D6

Note: D6 = 0: Linear axis, 1: Rotary axis

Table 4.2.3.8 Parameters for Setting Valid/Invalid of Stored Stroke Limit for Rotary Axes

Axis Name	X	Y	Z	4	5
Parameter	pm6004 D0	pm6004 D51	pm6004 D2	pm6004 D3	pm6004 D4

Note: Stored stroke limit check is executed or not executed according to the setting for the bits.

0: Limit check is not executed.

1: Limit check is executed.

(3) Supplements to the Stored Stroke Limit B and C Functions

- It is possible to set two areas in stored stroke limit B, C with stored stroke limit A.
- Whether the stored stroke limit function is valid or invalid is judged immediately after the return to the reference point from the setting for the corresponding parameter bits. If the reference point is set in the entry prohibited area, an alarm occurs; in this case, set the parameter bit so that the stored stroke limit check function becomes valid after the completion of reference point return of all axes after power ON.
- If the axis to be checked is in the entry prohibited area, an error occurs immediately when the stored stroke limit check function becomes valid. In this case, turn OFF the No. 2 to No. 5 entry prohibited areas by the setting operation and either change the data or move the axes out of the entry prohibited area manually.
- If an alarm occurs due to the entry of an axis into the entry prohibited area, only the axis movement in the opposite direction is allowed.
- If the machine lock function is ON, stored stroke limit check is not possible.
- With the axes not specified with G22 command, the stored stroke limit check is executed with the values set for parameters.
- The stored stroke limits B and C allows the designation of two areas at the same time. If more than two areas are made valid at the same time, an alarm occurs.
- When setting the coordinate values for defining a stored stroke limit area width, set the values which are greater than the value obtained by the calculation using feedrates. If the stored stroke limit area width is set smaller than the calculated value, there is a case that an alarm cannot be detected.

$$\text{Feedrate (mm/min)} \div 60 (\text{sec}) \times \underset{\substack{\uparrow \\ \text{Coefficient for detection}}}{6} (\text{msec})$$

Example

Feedrate (m/min)	Stored Stroke Limit Area Width (mm)
4	0.4
26	2.4
36	3.6

4.2.4 Break Point Function

The function displays “BREAK POINT!” message and suspends program execution in the block stop mode when the block assigned with the sequence number that is set for a parameter is executed. The message “BREAKPOINT!” can be cleared by the cycle start or reset operation.

The sequence number to call up this function is set for the following parameters:

pm0801	Break point sequence number 1
pm0802	Break point sequence number 2

4.2.5 High-speed Cutting *

By converting the linear interpolation mode (G01) or circular interpolation mode (G02, G03) commands in the part program memory into the axis move distance per unit time (to be referred to as segment data), high-speed cutting is made possible.

(1) Programming Format

Enclose the part of program to be operated at high-speed cutting mode with “HON” and “HOF”.

Example of Programming

(G91 F10000;)

.....

HON;

X100;

Y100;

X50;

X50 Y50;

HOF;

.....

(2) Parameters

(a) Additional axis address

For additional axes, address characters can be selected from A, B, C, U, V, or W by the setting for parameters pm1201 and pm1202. The address characters to be used to express basic axes are restricted to X, Y, and Z.

(b) Segment data conversion start and end codes

In this manual, “HON” and “HOF” are used for the start and end codes. However, the codes are not restricted to these and can be specified in the following range.

- Three alphanumerics (beginning with an alphabet)
- Those which are not reserved for use in the NC language such as HON, HOF, and G05.

Start designation code: Parameters pm4161, pm4162, and pm4163

End designation code: Parameters pm4164, pm4165, and pm4166

(c) Segment data type

For parameter pm4167, always set “4”.

(d) Selecting the conversion mode

For parameter pm4160, always set “1”.

(e) Number of axes used for high-speed cutting

Set the number of axes for parameter pm 1200. The setting range is from 3 to 5; “3” for 3 axes, “4” for 4 axes, and “5” for 5 axes.

(f) Accuracy at corners

Use parameter pm4800 for setting set acceleration/deceleration to be applied for cutting at corners.

If “0” is set, setting of “500 (mm/sec²)” is assumed.

(g) Acceleration/deceleration

User parameter pm4810 for setting acceleration/deceleration to be applied for high-speed cutting with exception of corners.

If “0” is set, setting of “500 (mm/sec²)” is assumed.

(h) Allowable acceleration for circular arc blocks

Use parameter pm4823 for setting the allowable limit of acceleration for circular arc blocks where high-speed cutting is executed.

If “0” is set, setting of “500 (mm/sec²)” is assumed.

(i) Shape offset coefficient

The shape offset coefficient is a constant used to compensate for follow-up error generated due to servo lag.

Table 4.2.5.1 Shape Offset Coefficient

	No.1 Offset Coefficient	No.1 Offset Coefficient
X-axis	pm1801	pm1781
Y-axis	pm1802	pm1782
Z-axis	pm1803	pm1783
4th-axis	pm1804	pm1784
5th-axis	pm1805	pm1785

(3) Input Units

Set 0.001 mm or 0.0001 inch for parameter pm0007. The unit of output is fixed at 0.001 mm.

(4) Supplements to the High-speed Cutting Function

- For blocks including only feed value (F command), high-speed cutting is not possible since segment conversion is not possible for such blocks.
- During high-speed cutting, “SHC” or “GHC” is displayed at the operating status display area on the screen.
- The single block function is invalid in the area enclosed by “HON” and “HOF” codes.
- During high-speed cutting, switching of the dimensioning mode between “mm” and “inch” systems is not possible by using G20 and G21.
- During high-speed cutting, scaling, coordinate rotation, and other functions are not allowed.
- The dry run function cannot be called if the DRY RUN switch is turned ON during the execution of commands in the area enclosed by “HON” and “HOF” codes.
- In the area enclosed by “HON” and “HOF” codes, external deceleration is not possible.
- In the area enclosed by “HON” and “HOF” codes, reset key, external reset, and feed hold are valid.
- The subprogram call up code (M98) can be used.
- In the area enclosed by “HON” and “HOF” codes, the feedrate specified in the program is displayed on the PROGRAM EXECUTION screen since special processing is executed in the NC.
- In the area enclosed by “HON” and “HOF” codes, processing is executed only for the basic axis (X, Y, Z) and those set for parameters pm1200 to pm1202.

Therefore, if the parameters are not set correctly, operation is not performed as programmed. If axis move commands are specified for the axis not set for parameters, they are disregarded. This does not cause an alarm.

- G41, G42, and G40 should be specified in the area enclosed by “HON” and “HOF” codes. If the high-speed cutting mode is started with G41, G42, or G40 specified before such area, alarm 0189, “OFS ABNORMAL ERROR” (tool radius offset error) or alarm 0181, “RADIUS OFS END ERROR” (tool radius offset end error) occurs.
- For the G41, G42, or G40 program specified in the high-speed cutting mode, only the following G codes can be specified.

G00, G01, G02, G03, G90, and G91

If G codes other than these G codes are specified, alarm “0181” occurs. The following G codes can be specified in high-speed cutting: G01, G02, G03, G41, G42, G90 and G91. In addition to these G codes, G00 specified in the tool radius offset mode is treated as G01 to execute high-speed cutting.

Axis move commands specified with a G code other than those indicated above are not converted into segment data and thus they are processed in normal mode. Note that designation of G20, G21, G25, G50, G51, G50R, G66, G67, G68 and G69 causes alarm “0163”.

- When the setting for a parameter is made so that “the tool radius offset C cancel is made”, alarm “0189” occurs if the HOF code is specified immediately after the designation of G40. Therefore, to specify G40 in the high speed cutting mode, either insert a G01 mode block between the G40 and HOF blocks or enter an axis move command in the G40 block.

4.2.6 Chamfering and Corner Rounding Commands *

For the continuous linear interpolation blocks or circular interpolation blocks, chamfering or corner rounding are executed by specifying “P” or “Q” command. If P and “Q” are specified in the same block, chamfering is executed.

(1) Programming Format

Both P and Q commands are non-modal commands and should be specified with an absolute value. If a minus sign is used, processing is executed based on the absolute value of the specified value.

If chamfering or corner rounding command is specified in the incremental mode (G91), the programmed commands are treated as if the chamfering or corner rounding commands were not specified.

(a) Chamfering command

G01(or G02, G03) ••• P_;

(P: distance between the chamfering start/end point and assumed point of intersection)

With the commands indicated above, chamfering at an arbitrary angle is executed.

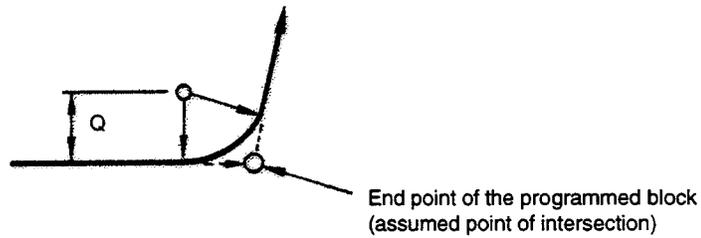


FIGURE 4.2.6.1 Chamfering at Arbitrary Angles

(b) Corner rounding command

G01 (or G02, G03) ••• Q_;

(Q: Radius of arc contacting the two continuous cutting blocks)

With the commands indicated above, rounding is possible at arbitrary angle corners.

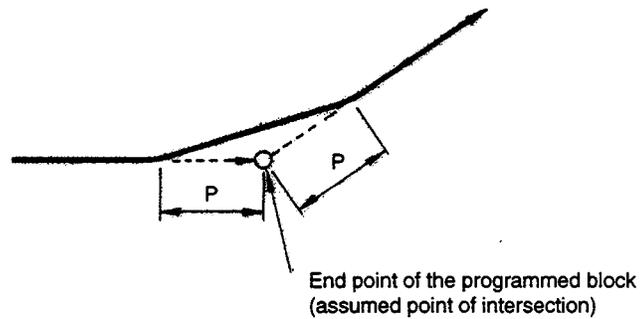


FIGURE 4.2.6.2 Corner Rounding at Arbitrary Angle Corners

(2) Operation

Chamfering and corner rounding commands are valid only when the cutting feed blocks continue. If the block specified immediately after the designation of chamfering or corner rounding is not a linear or circular interpolation block, alarm "0198" occurs. Alarm "0198" also occurs if there are no axis move commands in the next block.

In the chamfering and corner rounding blocks, the following G codes can be specified: G01, G02, G03, G07, G10, and G13

(a) Chamfering command

Example of Programming

```
G17 G54 G90 G00 X0 Y10.;
N1 G01 X10. F150;
N2 G03 X60. Y60. R50. P10.;
N3 G01 X70.;
N4 X80. Y30.;
```

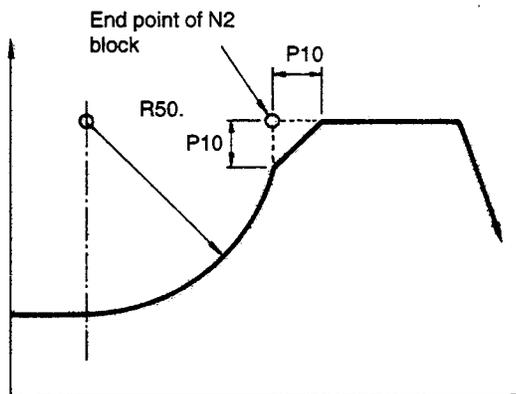


FIGURE 4.2.6.3 Chamfering Command

(b) Corner rounding command

Example of Programming

```
G17 G54 G90 G00 X0 Y10.;
N1 G01 X10. F150;
N2 G03 X60. Y60. R50. Q20.;
N3 G01 X80.;
N4 X80. Y30.;
```

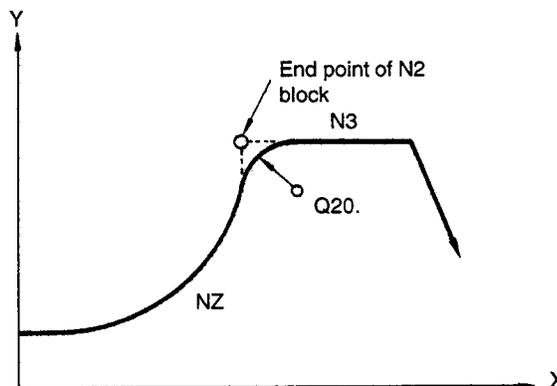


FIGURE 4.2.6.4 Corner Rounding Command

(3) Alarms

If there is a program error in the chamfering or corner rounding commands, the following alarms occur.

(a) Alarm 0196 CHAM/RND COMMAND ERROR

(b) Alarm 0197 CHAM/RND PLANE ERROR

Arbitrary angle chamfering and corner rounding are executed in the specified plane. If the plane specified for the execution of chamfering or corner rounding differs from the plane specified in the next block, alarm “0197” occurs. An axis move command of the axis not included in the specified plane also causes alarm “0197”.

(c) Alarm 0198 CHAM/RND CANNOT COMMAND

If an axis moves beyond the axis move range specified in the program as the result of insertion of a chamfering or corner rounding block, alarm “0198” occurs. If the chamfering or corner rounding block includes commands for which buffering is not allowed, the chamfering or corner rounding cannot be executed and alarm “0198” occurs.

(4) Supplements to the Chamfering/Corner Rounding Commands

- “If chamfering or corner rounding operation is executed, the single block stop point is changed to the end point of inserted chamfering or corner rounding operation.
- If chamfering or corner rounding is inserted between the blocks for which the difference in the tangential angles is within $\pm 1^\circ$, chamfering or corner rounding is invalid. Note that this does not cause an alarm. With a circular interpolation block, the difference in angles is obtained based on the tangent at the point of intersection.
- If an interpolation command block includes the subprogram call up command (M98 P_ Q_) or the macro program call up command (G65 P_ Q_), the P command is assumed as address “P” used for subprogram call up, and chamfering or corner rounding is not executed.
- If M, S, and/or T command is specified in the chamfering or corner rounding block, the NC waits for the finish signal at the end point of chamfering or corner rounding.
- Even if the programmed shape is distorted due to the execution of chamfering or corner rounding, these commands are executed. For example, although the chamfering command between the arc block and linear interpolation block causes a path which cuts the inside of the programmed arc as shown in Fig. 4.48, chamfering is executed as programmed.

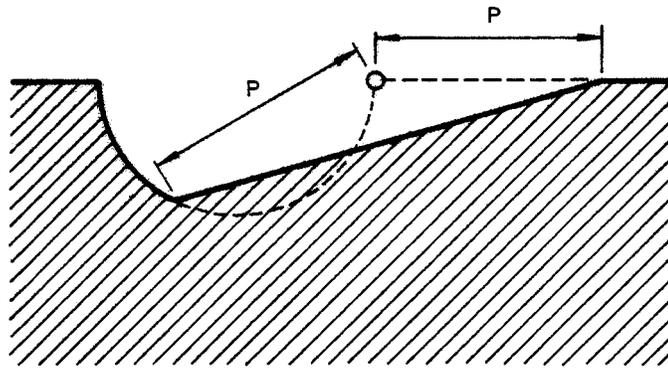


FIGURE 4.2.6.5 Chamfering between Arc Block and Linear Interpolation Block

- If the chamfering/corner rounding option is not selected, addresses P and Q specified in the program are disregarded.
- If chamfering or corner rounding is executed in the scaling mode, scaling is not executed for the chamfering or corner rounding.
- If a chamfering or corner rounding command is specified in the block immediately before the start or end block of the tool radius offset block, chamfering or corner rounding is executed as programmed. The offset start and end movements are started from the point where the chamfering or corner rounding is completed.
- If a chamfering or corner rounding command is specified in the start or end block of the tool radius offset block, offset start and end movements are executed at the start point of chamfering or corner rounding.
- If a chamfering or corner rounding command is specified in the tool radius offset mode, tool radius offset is executed after the generation of the chamfering or corner rounding block.

4.2.7 Corner Feedrate Designation (G107, G108)

With the automatic corner override function which is conventionally available, feedrate is decelerated only at the entry to the inside corner. The new corner feedrate designation function decelerates feedrate both at the entry and exit of the inside and outside corners so that an error between the actual tool paths and the programmed tool paths caused by servo lag is minimized.

Axis movement specified in a G01 block is divided in three sections such as decelerated feedrate section 1, specified feedrate section, and decelerated feedrate section 2.

Concerning axis movements specified in the G02 and G03 modes, specified decelerated feedrate is applied.

(1) Commands

The corner feedrate designation function is turned ON/OFF by the following G codes.

Table 4.2.7.1 G Codes Used for Turning ON/OFF the Corner Feedrate Designation

G code	Function	Group
G107	<ul style="list-style-type: none"> • Corner feedrate designation mode ON • Setting the decelerated feedrate, deceleration, distance, and tolerance. 	*
G108	<ul style="list-style-type: none"> • Corner feedrate designation mode cancel. 	*

(a) Corner feedrate designation mode ON command (G107)

G107 P...L...Q...;

G107 P...L...;

G107 P...Q...;

G107 L...Q...;

G107 P...;

G107 L...;

G107 Q...;

G107;...;

P: Sets the decelerated feedrate.

Setting unit: 1 = 1 mm/min or 0.1 inch/min

L: Sets the deceleration distance in the tangential direction

Setting unit: 1 = 0.001 mm or 0.0001 inch

Q: Sets the arc tolerance

Setting unit 1 = 0.001 mm or 0.0001 inch

Setting range: 0 to 32767

If address P, L, and/or Q is not specified, the values set for the following parameters are used:

Decelerated feedrate pm0860, pm0862

Deceleration distance pm0861, pm0863, pm2585

Arc tolerance pm0416, pm0417

(b) Corner feedrate designation mode cancel command (G108)

The G108 command must be specified in a block without other commands.



1. The setting ranges for P and L are the same as applied for normal F commands and axis commands.
2. The Q command becomes invalid when the NC is reset either by reset operation or a command in a program.
3. When the NC is reset, it is placed in the G108 mode.
4. The values set for P and L are displayed at the lower area of command value displayscreen in the [RUN] process. The display is blank while in the G108 mode.

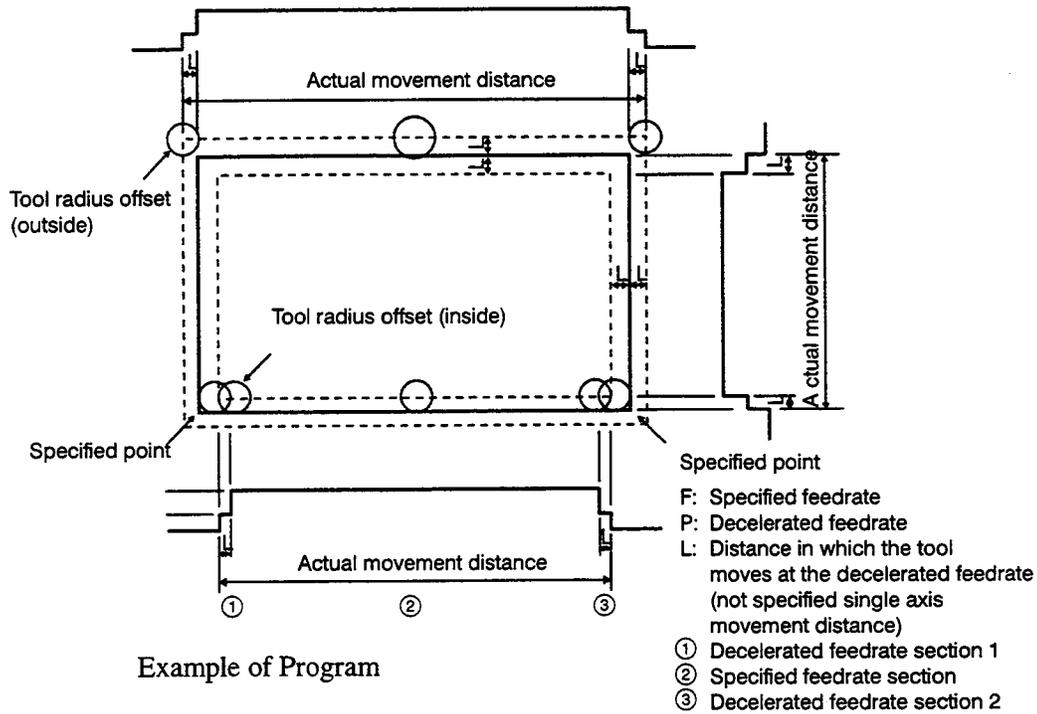
(2) Operations

- In the G01 mode

Axis movement specified in a block is divided into three sections, including the acceleration section and the deceleration section.

- In the G02/G03 mode

Axis movement is executed at the decelerated feedrate.



Example of Program

```

G107 P ... L ... ← Mode ON
G01 ← Start of axis feed
G02
G01
... ← Specify G107 to change P and/or L during the
... execution of the commands in the G107 mode.
G03
G01
...
G01 ← End of axis feed
G108 ← Mode OFF
    
```

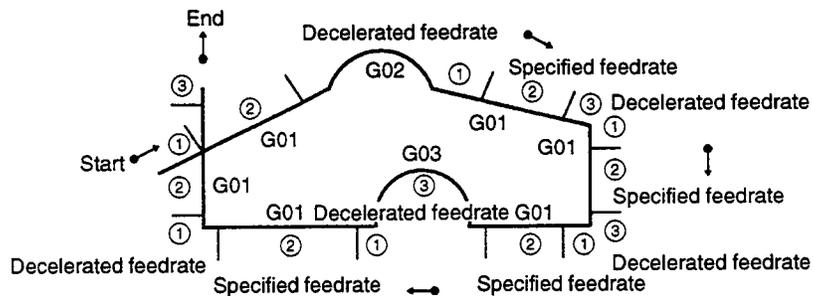


FIGURE 4.2.7.1 Example of Programming



1. In the decelerated feedrate sections 1 and 2, the tool is moved at the feedrate specified by the P command in the G 107 block.

In the following conditions, the tool is moved at the decelerated feedrate in the entire sections.

- If $F/150000 \geq L$, the tool moves at the decelerated feedrate in the entire section up to the distance calculated by the following formula.

$$\text{Distance} = L \times 2 + F \times 2/15000$$

- If $F/150000 < L$, the tool moves at the decelerated feedrate in the entire section up to the distance calculated by the following formula.

$$\text{Distance} = L \times 2 + F/15000$$

2. In the G02 and G03 modes, the tool moves at the decelerated feedrate.
3. Deceleration distance L is the length of tangential line and differs from the lengths along the axes.
4. For the length of the line, the actual movement distance in which the tool radius offset, tool length offset, and position offset are included is used instead of the specified line.
5. The blocks inserted due to the execution of the tool radius offset are processed as the normal blocks.
6. The G107 and G108 commands are not always specified in the same block. The G108 command is used to change the decelerated feedrate and deceleration distance after the entry to the G107 mode. Usually, it is used with circular interpolation operation that will take a relatively long time if the decelerated feedrate which is specified in the previous block is applied. In such a case, the decelerated feedrate is increased temporarily by specifying the G108 command to shorten the cutting time.
7. If the feedrate specified by the F command is slower than the decelerated feedrate specified with the G107 command, the axis movement commands are executed at the specified F command.
8. The P and L commands specify the decelerated feedrate and deceleration distance which are valid only for the commands specified with the G107 command and they are different from the normal P and L commands used to specify the feedrate and number of repetition times.
9. While the tool is moving in the decelerated feedrate section, "F" flashes at the operation status display area.

(3) Corner Feedrate B Specification

When the corner feedrate B specification is optionally selected, the following three functions can be used.

- Automatic arc-corner feedrate designation function
- Automatic deceleration distance designation function
- Start-up G107 mode selection function

(a) Automatic arc-corner feedrate designation function

The function automatically determines the feedrate to be applied to movements along arc-corners so that the tolerance (error between the programmed arc and the actual tool path) at the arc shaped (G02, G03) corners will be equal to the value set for the parameter.

This function is valid when both of the following requirements are satisfied.

- Parameter pm4031 D2 = 1
- A value equal to or greater than “1” is set for arc tolerance parameter pm0416 (“mm” mode) or pm0417 (“inch” mode).

In the case of the example of the program indicated below, feedrates to be applied to the movements along the arc-corner and the leading and trailing sections are shown in Fig. 4.2.7.2. The broken lines in Fig. 4.2.7.2 represent the feedrate waveform.

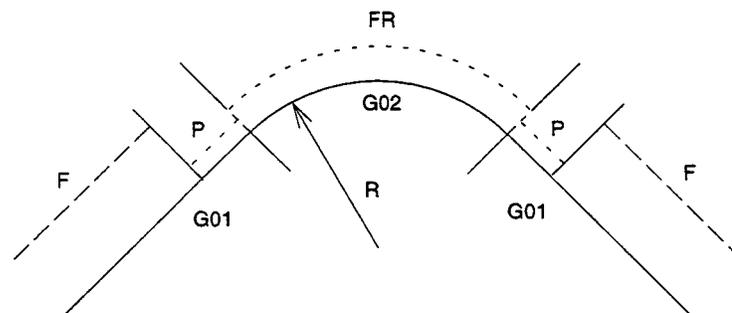
Example of Program

- Setting parameters

```
G91 G107P ... L ...
G01 X ... Y ... F ...
G02 X ... Y ... R ...
G01 X ... Y ...
```

- Program commands

```
G91 G107P ... L ... Q ...
G01 X ... Y ... F ...
G02 X ... Y ... R ...
G01 X ... Y ...
```



F Feedrate specified in the program
 P Decelerated feedrate specified using the corner feedrate designation function
 FR Feedrate determined by the automatic arc-corner feedrate designation function
 Note: If the calculated value of FR exceeds the value specified by F, FR = F.

FIGURE 4.2.7.2 Feed rate at Arc-Corner



1. To make this function invalid, set “0” for pm0416 (for “mm” mode) or pm0417 (for “inch” mode). If a Q command is specified in a program, the function is valid even if “0” is set for these parameters.
2. While the tool is moving in the arc-corner section, “F” flashes in the operation mode display area as with the normal feed by an F command.

(b) Automatic decelerating distance designation function

The deceleration distance (tangential direction) which is specified by address L in the G107 block is automatically determined according to the P command to be specified in the G 107 block. The deceleration distance L is calculated in the following manner: $P \times \Delta t \times N = L$.

This function is valid when all of the following requirements are satisfied.

- Parameter pm4031 D2 = 1
- A value equal to or greater than “1” is set for deceleration distance multiplication parameter pm2585.
- Address L is not specified in the G107 block, or G107 is not specified with the parameter setting of “pm4031 D 1 = 1”. For details of pm4031, refer to item (c).

If all of the requirements indicated above are satisfied, the deceleration distance L is calculated by the following formula.

$$L' = P \times \Delta t \times n \times 600$$

(where, $\Delta t = T_s + T_p$)

where,

L': Deceleration distance to be calculated (mm)

P: Decelerated feedrate specified by the corner feedrate designation function (mm/min)
If address P is not specified, the value set for pm0860 is used.

Ts: Acceleration/deceleration time constant of X-axis cutting feedrate
(see) ($T_s = \text{pm2501} \times 0.001$)

Tp: X-axis position loop gain (see) ($T_p = 100 \times \text{pm 1580}$)

n: Deceleration distance multiplication ratio set for pm2585
(1 = 0.1 time)

This function is invalid if n = 0.

If the deceleration distance multiplication ratio is 4 times ($pm2585 = 40$), for example, deceleration distance L' is equivalent to the area of the shaded parts in Figs. 4.2.7.3 and 4.2.7.4. Fig. 4.2.7.3 shows how the feedrate changes when G01 blocks continue and Fig. 4.2.7.4 show the changes in feedrate when tool movement mode changes from the G01 mode to the G02 mode. Note that broken lines in Figs. 4.2.7.3 and 4.2.7.4 represent the feedrate waveform including the delay due to "TS + Tp".

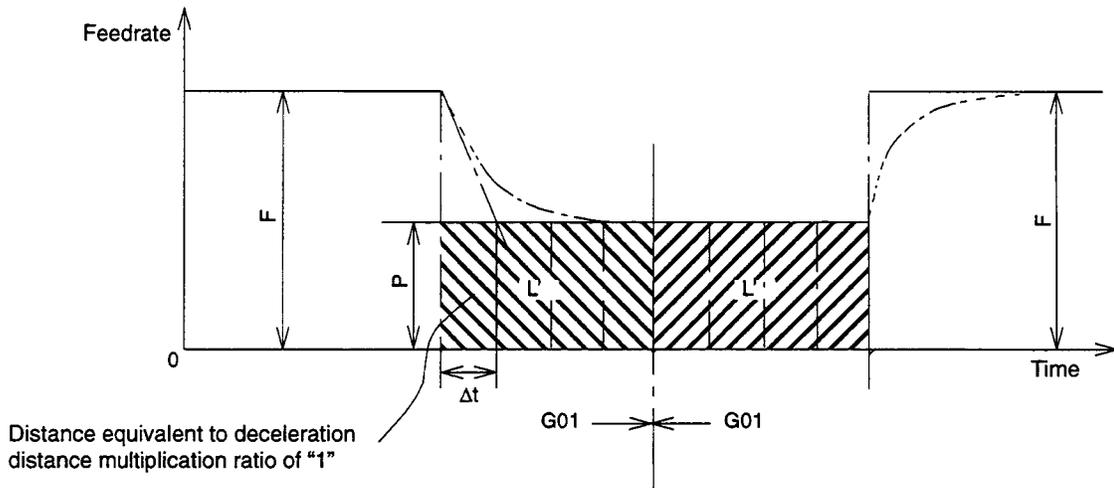


FIGURE 4.2.7.3 Deceleration Distance (Continuous G01 Blocks)

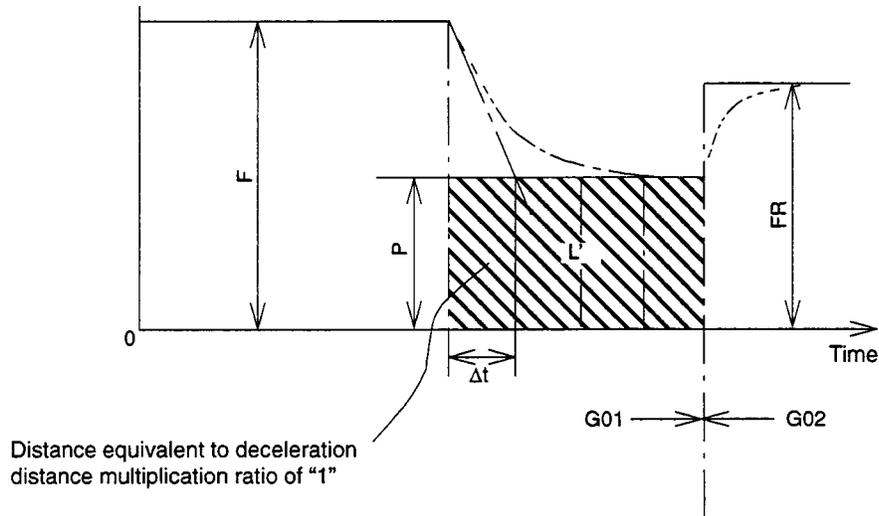


FIGURE 4.2.7.4 Deceleration Distance (G01 to G02 Block)



1. The cutting feed acceleration/deceleration time constant and position loop gain must be the same for all axes.
2. To make this function invalid, set “0” for pm2585. For the setting conditions under which the deceleration distance is set, refer to Table 4.2.7.2 in item (c).

(c) Start-up G107 mode selection function

This function establishes the G107 modal mode when the power is turned ON. In addition to this, the functions described in items (a) and (b) above are automatically set valid at the same time when the necessary requirements are satisfied.

This function is valid when all of the following requirements are satisfied.

- Parameter pm4031 D1 = 1
- Parameter pm4031 D2 = 1

When this function is made valid, the corner feedrate is automatically designated for G01, G02, and G03 blocks. Concerning the deceleration at corners, arc-corner feedrate, and deceleration distance, they are specified as shown in Table 4.2.7.2 according to the part program and the NC parameter setting conditions.

Table 4.2.7.2 Decelerated Feedrate and Arc-corner Feedrate

Conditions			Specified Value	
G107	Address P	pm0416 or Address Q	Decelerated Feedrate	Feedrate at Arc-corner
Used	Used	0	Specified by address P	Specified by address P
Used	Used	≥1	Specified by address P	Calculated FR value
Used	Not used	0	Specified by pm0860	Specified by pm0860
Used	Not used	≥1	Specified by pm0860	Calculated FR value
Not used	Not used	0	Specified by pm0860	Specified by pm0860
Not used	Not used	≥1	Specified by pm0860	Calculated FR value

Note I: The table above is for “mm” mode. The parameter numbers are changed as indicated below for “inch” mode.

pm0416 + pm0417

pm0860 + pm0862

- 2: If address Q is specified, priority is given to the value set for address Q.

Table 4.2.7.3 Deceleration Distance

Conditions			Specified Value
G107	Address L	pm2585	Decelerated Distance
Used	Used	0	Specified by address L
Used	Used	≥ 1	Calculate L' value
Used	Not used	0	Specified by pm0861
Used	Not used	≥ 1	Calculate L' value
Not used	Not used	0	Specified by pm0861
Not used	Not used	≥ 1	Calculate L' value

Note: The table above is for “mm” mode. The parameter number is changed as indicated below for “inch” mode.

pm0861 † pm0863



If the G107 command is specified while this function is valid, the corner feedrate is specified at the condition specified by G107 until the G108 command is specified. When the G108 command is specified, the corner feedrate designation function is turned OFF and it is turned ON again when the G107 command is specified once again. If the G107 command is specified independently without other commands, the status to be established at power ON is recovered.

To make this function invalid, either set “pm4031 D1 = 0” or specify the G107/G108 command as indicated in Tables 4.2.7.2 and 4.2.7.3.

(4) The Relationship between the Corner Feedrate Designation and Operations

(a) Override

The override function by the I/O input is valid in the section where the tool is moved at the specified feedrate (specified feedrate section). In the decelerated feedrate sections, axis movement stops only if the override signal is “0”.

(b) Override cancel

By canceling the override, 100% override is applied to the specified feedrate. The decelerated feedrate specified “by the G107 command is not influenced by override.

(c) Dry run

In the dry run mode, the corner feedrate designation is invalid and the dry run feedrate is used.

(d) Canned cycle

The G74 and G84 commands (tapping cycle) and the G93 command (solid tapping) cannot be specified with the G107 command in the same block. If specified, an alarm occurs. The G107 command is valid in canned cycles other than G74, G84, and G93 cycles. If the G81 drilling cycle is executed in the G107 mode, for example, axis feed in the section from R-point level to the Z-point level is controlled in the G107 mode.

(e) High-speed cutting

If high-speed cutting is called in the G107 mode, the G107 mode is canceled temporarily and it is recovered after the completion of high-speed cutting.

(f) Inch/mm switching

The values for addresses P and L are generated at the execution of the G107 command. If dimension mode is switched between “inch” and “mm” in the G107 mode, the values for addresses P and L are not updated according to this switching and, therefore, the program is not executed correctly.

(g) Cycle start in the G107 mode after feed hold

In the decelerated feedrate sections 1 and 2, cutting restarts at the decelerated feedrate. In the specified feedrate section, cutting restarts at the specified feedrate.

(h) Others

The G107 command can be specified with the G68/G69 (coordinate rotation) command, subprogram, and microprogram functions in the same block.

(5) Related Parameters

Table 4.2.7.4 Table of Parameters

Parameter		Description	Setting Range	Setting Unit
pm4031	D2 = 1	Corner feedrate designation function B ON		
	D2 = 0	Corner feedrate designation function B OFF		
pm4031	D1 = 1	The corner feedrate designation function is made valid at power ON or after reset		
	D1 = 0	The corner feedrate designation function is made valid at the execution of G107		
pm0416		Arc-corner tolerance for the corner feedrate designation function (“mm” mode)	0 to 32767	1 = 0.001 mm
pm0417		Arc-corner tolerance for the corner feedrate designation function (“inch” mode)	0 to 32767	1 = 0.0001 inch
pm0860		Decelerated feedrate for the corner feedrate designation function (“mm” mode)	0 to 999999	1 = 1 mm/min
pm0861		Deceleration distance for the corner feedrate designation function (“mm” mode)	0 to 999999999	1 = 0.001 mm
pm0862		Decelerated feedrate for the corner feedrate designation function (“inch” mode)	0 to 999999	1 = 0.001 inch/min
pm0863		Deceleration distance for the corner feedrate designation function (“inch” mode)	0 to 999999999	1 = 0.0001 inch
pm2585		Deceleration distance multiplication ratio	0 to 32767	1 = 1 time

Note: To change the setting for the parameters indicated above, resetting operation is necessary.

4.3 AUTOMATING SUPPORT FUNCTIONS

4.3.1 Skip Function (G31) *

By specifying “G31 X ••• Y ••• Z ••• (*a ••• b •••) F •••;”, special linear interpolation is executed. If a skip signal is input during the execution of linear interpolation, linear interpolation is interrupted and the program advances to the next block without executing the remaining linear interpolation.

Delay from the input of the skip signal to the start of processing corresponding to the input signal is shorter than 0.5 msec; this is processed at extremely high speed.

Example of Programming

```
N100 G90 G31 X100. Y50.;
N200 G01 X80. Y15.;
```

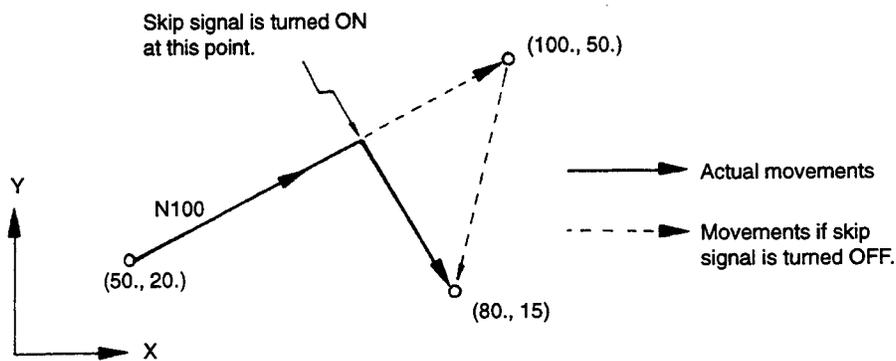


FIGURE 4.3.1.1 Example of Skip Function

(1) Programming Format

(a) Feedrate

For the execution of the G31 block, feedrate can be selected from the following two methods according to the setting for parameter pm2001 D0.

pm2001 D0 = 0	To specify the feedrate with F as another ordinary program.
pm2001 D0 = 1	To use the feedrate preset for parameter pm2440

(b) If skip signal is turned ON

When the skip signal is input, the coordinate values of the point where the skip signal is input are automatically saved to the parameters. Therefore, the coordinate values of the skip point can be used as the coordinate data in macro programs.

pm0811	Saving the X-axis coordinate value
pm0812	Saving the Y-axis coordinate value
pm0813	Saving the 4th-axis coordinate value
pm0814	Saving the Z-axis coordinate value
pm0815	Saving the 5th-axis coordinate value

Note: The coordinate values of the point where the skip signal is input are saved. This point differs from the point where axis moves are stopped.

(c) If skip signal is not turned ON

If the skip signal is not turned ON during the execution of the commands specified in the G31 block, the operation stops upon completion of these commands and alarm “049 1” occurs. Note that G31 is a non-modal G code.

Note that G31 is a non-modal code.

pm0007 D2 = 0	An alarm occurs if the skip signal is not input until the completion of the G31 block.
pm0007 D2 = 1	An alarm does not occur if the skip signal is not input until the completion of the G31 block. The program advances to the next block.

(2) Operation

(a) Designating the skip signal polarity

Designate the polarity of the skip signal by using parameter pm5011 D0.

pm5011 D0 = 0	Valid when the skip signal goes from “1” to “0” (negative logic).
pm5011 D0 = 1	Valid when the skip signal goes from “0” to “1” (positive logic).

If the designated skip signal polarity and the input status of the skip signal do not agree, the program proceeds to the next block at the start of the G31 block.

(b) G codes that do not allow the use of the skip function

The skip function cannot be used in the following states. If the skip function is called up, alarm “0160” or “0161” occurs.

- In the canned cycle mode (G12, G13, G81 to G89, G181 to G189)
- In the solid tap mode (G93)
- In the tool radius offset mode (G41, G42)
- In the tool position offset mode (G45 to G48)
- In the coordinate rotation mode (G68)
- If specified with scaling (G51) in the same block

- While the interrupt program, where jump has been made in response to the input of the interrupt signal during the execution of a program in the M91 mode, is executed, another interrupt signal is invalid.
- It is possible to specify the sequence number of the block where the interrupt program should start by using a Q command in the M91P block.

(b) End of interruption (M90)

The program interrupt function is canceled by specifying “M90;”.

(2) Supplements to the Program Interrupt Function

- In the program that is called up after interrupting the execution of another program, it is not allowed to specify M90 or M91. If specified, alarm “0114” occurs.
- The M91 command can be specified in a subprogram or macro program. Jump to an interrupt program is not counted as a nesting level. Therefore, the level saved to the macro local variable does not change.
- By the execution of M99 specified in the interrupt program, the program returns to the block next to the one where the interrupt program has been called up. It is also possible to specify the return block by specifying a P command with M99. When returning to the previous program by the execution of M99, the modal information which was valid before the interruption, is recovered. However, if M99P is used to return to the previous program, the modal information changed during the execution of the interrupt program is used for the execution of the previous program.
- If the interrupt signal is input during the block stop state, the program jumps to the interrupt program when the operation is started by depressing of the cycle start switch.
- The program interrupt signal is invalid if input during the execution of high-speed cutting.
- If the program interrupt signal is input during the execution of the G28 block (low-speed pattern), the program interrupt signal is disregarded. It is valid if it is input during the execution of high-speed pattern reference point return.
- If the program interrupt signal is input during the execution of G31 (skip), the skip mode is canceled and the program interrupt function is executed.
- If the program interrupt signal is input during the execution of a block including M, S, T, or B command, the program jumps to the interrupt program. Before jumping to the interrupt program, axis move is stopped after deceleration if the interrupt signal is input during axis move. If the M or T function is being executed when the interrupt signal is input, the program does not jump until the M or T function completion signal is input.
- If the program interrupt signal is input during the execution of tapping in the solid tap mode, execution of the interrupt program starts only after the completion of the solid tap block.

4.3.3 Tool Life Control Function *

When the tool life control function is used, the tools are controlled in groups and the service life of the tool is set for individual tools. Tools are selected by tool group and if a selected tool is used to the preset life, it is replaced with another tool in the same tool group according to the preset tool selection order.

(1) Tool Life Control Specifications

(a) Number of tool life controlled tools

With the tool life control function, tool life can be controlled for up to 256 tools. The tools are grouped for this function. Parameter pm0009 is used to set the number of tool groups and the number of tools in each group.

Table 4.3.3.1 Number of Tool Life Controlled Tools

pm0009		Number of Groups	Number of Tools in Group
D7	D6		
0	0	128	2
0	1	64	4
1	0	32	8
1	1	16	16



Do not change the number of tool groups during the execution of the tool life management function.

(b) Tool life control data

The data used by the tool life control function is stored as the tool group file. The data in this file is retained even after the power is turned OFF.

Table 4.3.3.2 Tool Life Control Data

Title	Description
T NO (tool number)	The number assigned to the tool for which tool life is controlled.
H NO (tool length offset number (H))	The tool length offset number corresponding to the set tool number
D NO (tool diameter offset number (D))	The tool radius offset number corresponding to the set tool number
LIFE (tool life)	The service life of the tool set for the tool number
USED (tool used data)	Courted used data of the tool set for the tool number
STS (status)	Status of the life of the tool set for the tool number OVER/SKIP
.LIFE TYPE	Tool life control data for each tool group TIME: 1 to 9999 COUNT 1 to 9999 DIST 1 to 9999 DRILL: 1 to 9999

(2) Specifying the Tool Life Control

(a) Specifying in a program

In a program, the T codes, H/D codes, and M codes indicated below are used to specify the tool life control function.

Table 4.3.3.3 Tool Life Control Commands Specified in a Program

Program Block	Operation (with the setting of pm4110 = 1)
TI005;	Specifies tool group number “5”. Tools are selected from the specified tool group in the preset order.
M06;	Places the status of the selected tool in “selected”.
H999;	Selects the H number registered for the tool.
D999;	Selects the D number registered for the tool.
D0;	Cancels D.
H0;	Cancels H
T0010;	Returns the T command to a normal T command.
M06;	Changes the tool to a not life-controlled tool; ends the tool life control mode which has been called up by the preceding M06.
M30;	End of program

(b) Counting the tool life

Tool life is counted for the life-controlled tools while they are actually called up and used in the tool life control mode. It is possible to set the tool life counting type for each of tool groups.

Table 4.3.3.4 Counting the Tool Life

Count Type	Counting Method
Controlled by time	Length of time the cutting tool is actually used in the cutting mode is counted in units of seconds. The life count data are increased by "1" in units of minutes. While the NC is in the single-block stop, feed hold, rapid traverse, or dwell state, the time is not counted.
Controlled by distance	Length of distance the cutting tool is actually used in the cutting mode is counted every second. The life count data are increased by "1" in cutting of every meter or foot.
Controlled by the number of uses	Life count is increased by the set number at each execution of M02, M30, or M192. It is also possible to count the life data in response to the execution of T9999. When "T9999L***;" is executed, the value set with L is added to the life count data. If no L command is given, L1 is assumed. It is not possible to use an M command and T command for counting the life count data at the same time. Which of the M or T command should be used can be determined by the setting for parameter pm0007 D4. pm0007D4 = 0: Counting at M02/M30/M 192 pm0007D4 = 1: Not counting at M02/M30/M 192
Controlled by the number of holes	The life count data is increased by the number of execution of hole machining cycles. The life count data is counted by the L command (number of repetitions) for the canned cycles and the hole pattern cycles.

Since tool life is counted in the buffering processing, if life exceeded occurs during the execution of the next one block, the life exceeded status is triggered before the execution of that block.



During the operation, registration or deletion of a tool file is not allowed.



1. It is possible to select the tool life counting (number of uses) objective groups by using the parameter indicated below.

pm4029 D3 = 0	Counting only for the specified tool groups
pm4029 D3 = 1	Counting for all registered tool groups

2. If the last tool in a group is skipped, the one previously used is called. If the previous tool has been skipped, the one used before the previous tool is called.

(C) Selecting a tool

When a tool group is specified, the tools that have used up the life or those in the skip state are disregarded and the tool is selected from the remaining tools in the order of registration. If the life expires during the use of the tool, machining is continuously performed. When the same tool group is specified next, another tool is selected among the available tools in that tool group. If the life of all the tools registered in a group has expired or placed in the skip state, the tool change signal is output. If machining is continued in this state, the tool used last is called up and machining is performed continuously.

(d) H999/D999 command

During the execution of tool life control function, H999/D999 is used to specify the offset number for the tool length offset/tool radius offset function. If H999/D999 is specified, the tool length offset number/tool radius offset number of the tool number of the tool presently called up is read from the tool life control data. By the H0/D0 command, the offset data (tool length offset data/tool radius offset data) are canceled.

(e) Input/output signals

Table 4.3.3.5 Input/Output Signals

Contents of Signal	Description
Tool-life count disregarded input (TLCTN = #30127)	If this input signal is “ON” during the execution of the tool life control function, counting of the tool life is suspended.
Tool skip input (TLSKP = #30126)	If this input signal state changes from “OFF” to “ON” during the execution of the tool life control function, the tool life status is forcibly skipped. pm4029 D1 = 0: For the tool presently selected pm4029 D1 = 1: For the tools in the specified group (TLGRP)
Tool group number input (TGN = #30250 to #30257)	Specifies the tool group number that is used for the tool skip input.
Tool change signal output (TLCHA = #35087)	When service life of all tool in a tool group has expired, the tool change signal status is set to “closed”.
New tool selection signal output (TLCHB = #35086)	When a new tool is selected in a tool group, if the life count data of the new tool are “0”, this signal is output with a T code at the same time.

(f) Display and setting of tool life control data

The tool life control data can be displayed by depressing the [T-LIFE] (f3) job soft key in the [SET] process. On this screen, the tool life control data (tool numbers in a tool group, offset number, life, number of use, etc.) can be changed by directly writing the data. The data can be changed only while the automatic operation is suspended (reset status, single block stop status) and not in the edit prohibited status.

(g) Specifying tool group

Tool group is specified by a T code. Tool group number is specified by the value set for parameter pm4110 \times 1000 + 1 to + 128. These numbers correspond to group 1 to group 128.

If the setting is “pm4110 = 5”: Tool groups T5001 to T5128 are set.

(3) Life count multiplication command

For counting the tool use data (distance, time), it is possible to specify the multiplying factor in a program by specifying “G125 P***; [P: Life count multiplying factor (one digit to the right of the decimal point)]”.

- G125 is a non-modal G code. Once “G125 P***” is specified, the life count multiplying factor specified as P*** remains valid until the NC is reset or G125 is specified again. The life count multiplying factor is returned to “1” when no P command, PO or P1 is specified with G125.
- The life count multiplying factor is valid only when counting tool life in distance or

time. Therefore, it is not valid for the number of use or the number of holes.

Example of Programming

G125 P1.5;

With the commands indicated above, the life count data (distance, time) are counted in 1.5 times the normal counting mode.

Distance: 2 m ‡ 3 counts

Time: 2 minutes ‡ 3 counts

4.4 MACROPROGRAMS

The NC has a set of instructions that can be used by the machine tool builders and the users to implement the original functions. The program created by using these instructions is called a microprogram, which can be called and executed by the commands specified in a block with G65 or G66.

A microprogram provides the following.

- Variables can be used.
- Arithmetic and logical operations using variables and constants are possible.
- Control commands for branch and repeat can be used.
- Commands to output messages and data can be used.
- Arguments can be specified.

This makes it possible to create a program in which complicated operations and operations requiring conditional judgment are included.

4.4.1 Differences from Subprograms

Differences between microprograms and subprograms are indicated below.

- With microprogram call up commands (G65, G66), arguments can be specified. However, with subprogram call up command (M98), it is not possible to use arguments.
- If commands other than P, Q, and L are specified in the M98 block, the program jumps to the specified subprogram after executing these commands. With G65 and G66, commands other than P and L are regarded as argument specification and the program jumps to the specified microprogram immediately. In this case, however, the commands specified preceding G65 and G66 are executed normally.
- With a microprogram, local programs at the same level as the level of the microprogram are used. However, with subprograms, levels of local variables are not changed.

In other words, local variables in a microprogram are different before and after the call up of the microprogram and those in a subprogram remain the same before and after the call up of the subprogram.

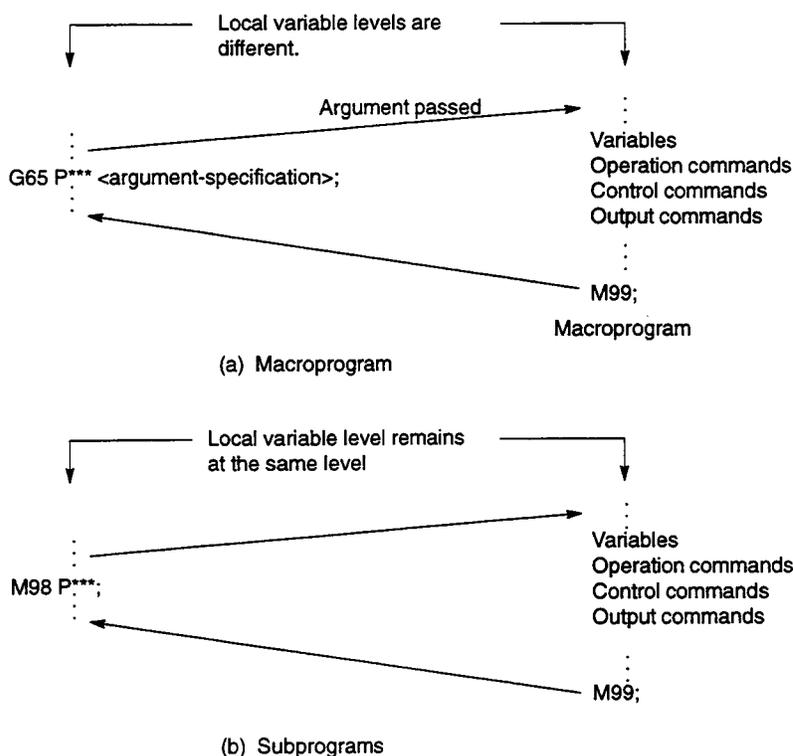


Figure. 4.4.1.1 Differences between Microprograms and Subprograms

4.4.2 Microprogram Call (G65, G66, G67)

Microprograms are usually executed after being called up.

The procedure used for calling up a microprogram is indicated in Table 4.4.2.1.

Table 4.4.2.1 Microprogram Calling Format

Calling Up Method	Command Code	Remarks
Simple call up	G65	
Modal call up (a)	G66	Canceled by G67:
G code call up	G***	G command: 3 digits
M code call up	M***	M command: 2 or 4 digits
S code call up	S***	S command: 6 digits
T code call up	T***	T command: 4 digits
B code call up	B***	B command: 4 digits

(1) Simple Call Up (G65)

By specifying “G65 P•••L•••<argument specification>”, the microprogram which is assigned the program number specified with P is called up and executed L times.

If it is necessary to pass arguments to the called up microprogram, these arguments can be specified in this block.

Table 4.4.2.2 P and L Commands

Address	Description	Number of Digits
P	Program number	5 digits
L	Number of repetitions	9 digits



t Argument Specification

A real number is assigned to the local variable that corresponds to the level of the called up microprogram. When specifying arguments, G65 must be placed before all arguments.

Commands specified before G65 are processed as normal commands and the program jumps to the called up microprogram after the completion of these commands.

For details, refer to item (9) “Specifying Argument”.

(2) Modal Call Up (G66, G67)

The modal call up commands set the mode for calling up a microprogram. The specified microprogram is called up and executed when the specified conditions are satisfied.

- By specifying “G66 P•••L•••<argument-specification>;”; the mode for calling up the microprogram is set. Once this block is executed, the microprogram which is assigned the program number specified with P is called up and executed L times after the completion of move commands.

If an argument is specified, the argument is passed to the microprogram each time it is called up as with the simple call up of a microprogram. The correspondence between the address of argument and local variables is the same as in the case of simple call up (G65).

- G67 cancels the G66 mode. When arguments are specified, G66 must be specified before all arguments. If G66 is specified, G67 must be specified in the same program corresponding to it.

Table 4.4.2.3 Modal Call Up Conditions

Call Up Conditions	Mode Setting Code	Mode Cancel Code
After the execution of move command	G66	G67

(3) Microprogram call up by G code

By specifying “G*** <argument-specification>;”, the microprogram/subprogram of the program number that corresponds to the number specified by G code is called up and executed.

For the G code used to call up a macroprogram/subprogram, a maximum of 24 pairs of G codes can be set; each G code has a maximum of 3 numerical digits that are not used by the NC. The program numbers of programs to be called up can be set corresponding to the set G codes.

Table 4.4.2.4 Parameters for Setting the Correspondence

Number of Pairs	Macroprogram Call Up G Code	Program No. to be Called Up
1	pm4480	pm4840
2	(max. 3 digits)	(max. 5 digits)
•	•	•
•	•	•
•	•	•
23	•	•
24	pm4503	pm4863

Max. 24 pairs

(4) Microprogram Call Up by M Codes**(a) Microprogram call up format**

By specifying “M*** <argument >;”, the microprogram of the program number that corresponds to the specified M code is called up and executed.

In this case, if a move command is specified in the same block, the microprogram is executed after the completion of the axis move command.

For the M code used to call up a microprogram/subprogram, a maximum of 24 pairs of M codes can be set excluding such M codes as M00, M01, M02, M30, and those used for internal processing. The program numbers of programs to be called up can be set corresponding to the set M codes.

Table 4.4.2.5 Parameters for Setting the Correspondence

Number of Pairs	Macroprogram Call Up M Code	Program No. to be Called Up
-----------------	-----------------------------	-----------------------------

1	pm4504	pm4864
2	(max. 4 digits)	(max. 5 digits)
•	•	•
•	•	•
•	•	•
23	•	•
24	pm4527	pm4887

Max. 24 pairs

(b) Specifying arguments

It is possible to specify arguments in the M code macroprogram call up block. In this case, it is not allowed to specify axis move commands in the same block.

pm4020 D5 = 0	Call up without argument specification
pm4020 D5 = 1	Call up with argument specification

If more than one M code is specified in a single block, the first M code is checked whether it is for microprogram call up. Concerning the second and later M codes, if the setting for parameter pm4020 D5 is “call up with argument specification”, the use of them is determined by the setting for a parameter whether they are treated as a normal M code or as M code used for specifying an argument.

pm4020 D6 = 0	Normal M code
pm4020 D6 = 1	M code for specifying argument

When an M code used for microprogram call up is executed, M code or MF which is output for normal M code is not output.

(5) Microprogram Call Up by S Code

(a) Microprogram call up format

By specifying “S*****”, it is possible to determine whether the specified S command should be treated as a normal S command or a microprogram call up S command by the setting for parameter pm4888. If “pm4888 = 0”, the S command is treated as a normal S command.

When using an S command for calling up a microprogram, one required program number can be set. In this case, the S command value is used as the argument of common variable #147. Designation of other arguments is not allowed.

Table 4.4.2.6 Parameter Used for Microprogram Call Up by S Code

Common Selection	Call Up Program Number
Normal S command	pm4888 = 0
Macroprogram call up S command	pm4888 max. 5 digits

When more than one S command is specified in a block, the first S command is checked whether it is a microprogram call up S command or not. For the second and later S commands, it is possible to select whether they should be treated as normal S commands or disregarded by the setting for a parameter.

pm4020 D7 = 0	Treated as normal S command
pm4020 D7 = 1	Disregarded

When a microprogram call up S command is executed, S code and SF are not output as a normal S code. The S command is a 6-digit command.

(6) Microprogram Call Up by T Code

By specifying “T*****”, it is possible to determine whether the specified T command should be treated as a normal T command or a microprogram call up T command by the setting for parameter pm4889. If “pm4889 = 0”, the T command is treated as a normal T command.

When using a T command for calling up a microprogram, one required program number can be set. In this case, the T command value is used as the argument of common variable #149. Designation of other arguments is not allowed.

Table 4.4.2.7 Parameter Used for Microprogram Call Up by T Code

Common Selection	Call Up Program Number
Normal T command	pm4889 = 0
Macroprogram call up T command	pm4889 max. 5 digits

When a microprogram call up T command is executed, T code and TF are not output as a normal T code. The T command is a 2-digit or 4-digit command.

(7) Microprogram Call Up by B Code (Second Miscellaneous Function)

By specifying “B*****”, it is possible to determine whether the specified B command should be treated as a normal B command or a microprogram call up B command by the setting for parameter pm4890. If “pm4890 = 0”, the B command is treated as a normal B command.

When using a B command for calling up a microprogram, one required program number can be set. In this case, the B command value is used as the argument of common variable #146. Designation of other arguments is not allowed.

Table 4.4.2.8 Parameter Used for Microprogram Call Up by B Code

Common Selection	Call Up Program Number
Normal B command	pm4890 = 0
Macroprogram call up B command	pm4890 max. 5 digits

When a microprogram call up B command is executed, B code and BF are not output as a normal B code. The B command is a 4-digit command.



When microprogram call up is specified with G, M, S, T, and/or B entered in one block, the execution priority is “G > M > S > T > B”.

- G • • • R • • • T • • • S • • • B • • •;

If the commands are specified as indicated above, only “G • • •” is valid and M, S, T, and B are disregarded. This does not cause an alarm.

- B • • • S • • • T • • • M • • • G • • •;

This block of commands is processed in the same manner. That is, the G command is given priority for the execution of the program independent of the order of the specified microprogram call up commands.

(8) Nesting of Macroprogram Call Up

As with subprograms, it is possible to call up a microprogram from another microprogram. In this type of call up, nesting level increases one each time a microprogram call up is executed by G65, G66, G, M, S, T, or B code. The allowable maximum nesting level of microprogram call up is four.

(a) Nesting level of microprogram call up

With a microprogram called up by G, M, S, T, and B codes, the allowable nesting level is one. In other words, from a microprogram called up by G, M, S, T, or B code, call up of another microprogram using G, M, S, T, or B code again is not allowed. If G, M, S, T, or B code is specified in a microprogram which has been called up by the execution of microprogram call up G, M, S, T, or B code, an alarm occurs in the case of a G code and with other codes (M, S, T, and B), they are executed as normal M, S, T, and B codes.

(b) Modal call up (G66)

In the modal call up mode, the specified microprogram is called up and executed at each execution of a move command. If more than one G66 is specified in the same program, the prior G66 command specified is valid during the execution of a microprogram called up by the G66 command given later. Therefore, after the execution of a move command given in the microprogram called up by G66 specified later, the microprogram specified with the previous G66 is also executed. In other words, the macroprograms are executed

specified last.

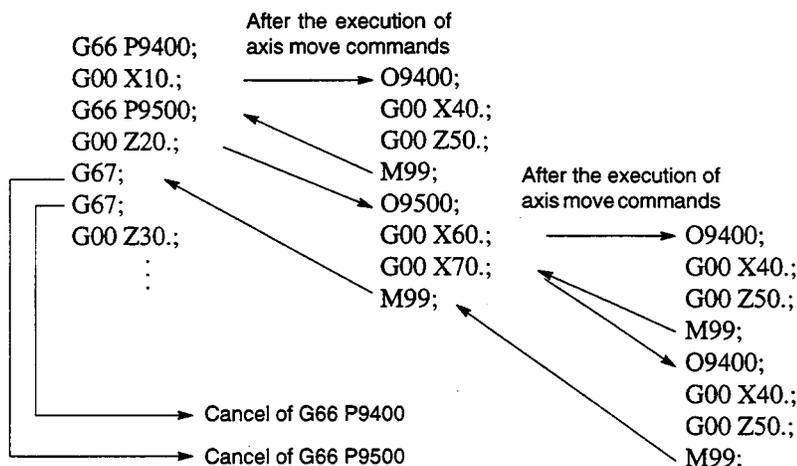
Example of Programming

```

: /
G66 P9400;          O9400;          O9500;
G00 X10.;          G00 X40.;          G00 X60.;
G66 P9500;          G00 Z50.;          G00 Z70.;
G00 Z20.;          M99;            M99;
G67;
G67;
G00 Z30.;
:

```

Execution order of programs



Note: If microprogram call up is nested by specifying more than one G66, cancel code G67, cancels G66 sequentially beginning with the one specified last. It is not allowed to specify G66 in the macroprograms which is called up by G66.

Figure 4.4.2.1 Nesting of Macroprogram Call

(9) Specifying Argument

The term “to specify argument” means “assigning a real number” for local variables used in a microprogram. There are two types of argument specifications: type I and type II. These types can be used as required, including a combination of the two types.

(a) Correspondence between addresses and local variables (Type 1)

Table 4.4.2.9 Address – Variable Correspondence and Usable Address for Call Up Commands (Type 1)

Address-Variable Correspondence		Call Up and Usable Address
Address in Type 1	Local Variable	G65, G66
A	#1	°
B	#2	°
C	#3	°
D	#7	°
E	#8	°
F	#9	°
G	#10	×
H	#11	°
I	#4	°
J	#5	°
K	#6	°
L	#12	×
M	#13	°
N	#14	×
O	#15	×
P	#16	×
Q	#17	°
R	#18	°
S	#19	°
T	#20	°
U	#21	°
V	#22	°
W	#23	°
X	#24	°
Y	#25	°
Z	#26	°

Note: °: Can be used, ×: Cannot be used

(b) Correspondence between addresses and local variables (Type II)

To use I, J, and K, they must be specified in the order of I, J, and K. Suffixes 1 to 10 specified in the table below indicate the order they are used in a set, and the suffix is not written in actual commands.

For addresses for which argument specification is not required, the commands can be omitted. In this case, local variables corresponding the addresses without commands are <empty>.

Table 4.4.2.10 Address – Variable Correspondence and Usable Addresses for Call Up Commands (Type II)

Address-Variable Correspondence		Call Up and Usable Address
Address in Type II	Local Variable	G65, G66
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	°
K 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	°

Table 4.4.2.10 Address – Variable Correspondence and Usable Addresses for Call Up Commands (Type II) (cont'd)

Address-Variable Correspondence		Call Up and Usable Address
Address in Type II	Local Variable	G65, G66
I ₇	#22	°
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	°

- Note
1. °: Can be used
 2. If more than one address is specified for one variable number, the one specified later is valid.
 3. If more than one set of I, J, or K is specified, the order of sets is determined for each I/J/K set, so that variable numbers are determined corresponding to that order.

(c) Example of argument specification

When arguments are specified, the macroprogram call up code must always be specified before the specification of arguments. If argument specification is given before the microprogram call up code, an alarm occurs. The value of argument specification can include a sign and decimal point independent of the address.

If no decimal point is used, the value is saved to the variable as the value with a decimal point according to the normal number of digits of that address. .

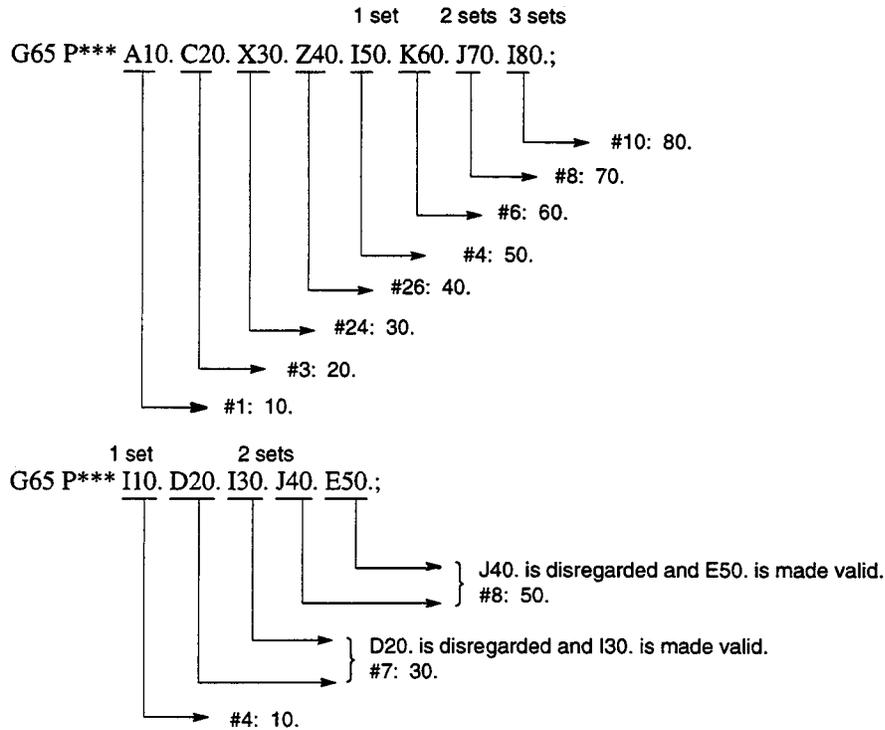


FIGURE 4.4.2.1 Example of Argument Specification

(d) Decimal point position in argument

An argument is usually specified with a sign and a decimal point. If a decimal point is not specified, the decimal point position is assumed at the position indicated Table 4.4.2.11.

Table 4.4.2.11 Decimal Point Position in Argument

Address for Argument Specification	Input in mm	Input in inches
A, C	3	3
B (without B3-digit option)	3	3
B (with B3-digit option)	0	0
D, H	0	0
E, F	0 (1)	1 (2)
I, J, K	3	4
M, S, T	0	0
Q, R	3	4
U, V, W	3	4
X, Y, Z	3	4

- Note 1: The number indicates the position of the decimal point counted from the lowest position.
 2: Numbers in () indicate the number of digits right to the decimal point when the setting of parameter “pm2004 D0 = 1”.

4.4.3 Variables

Three types of variables are provided: local variables, common variables and system variables.

(1) Local Variables (#1 to #33)

Local variables are used locally for each microprogram. Each time a microprogram is called up, new local variables (#1 to #33) are secured independently for that microprogram. For the local variables, values specified using arguments are saved or the results of operation executed in the microprogram are saved.

For those for which an argument is passed, the value is saved and those for which argument is not passed, the contents are <empty>. When execution of a program returns from the called up microprogram by the execution of M99, the local variables secured, for that microprogram

become <empty>. They are also <empty> when the power is turned ON or the NC is reset.

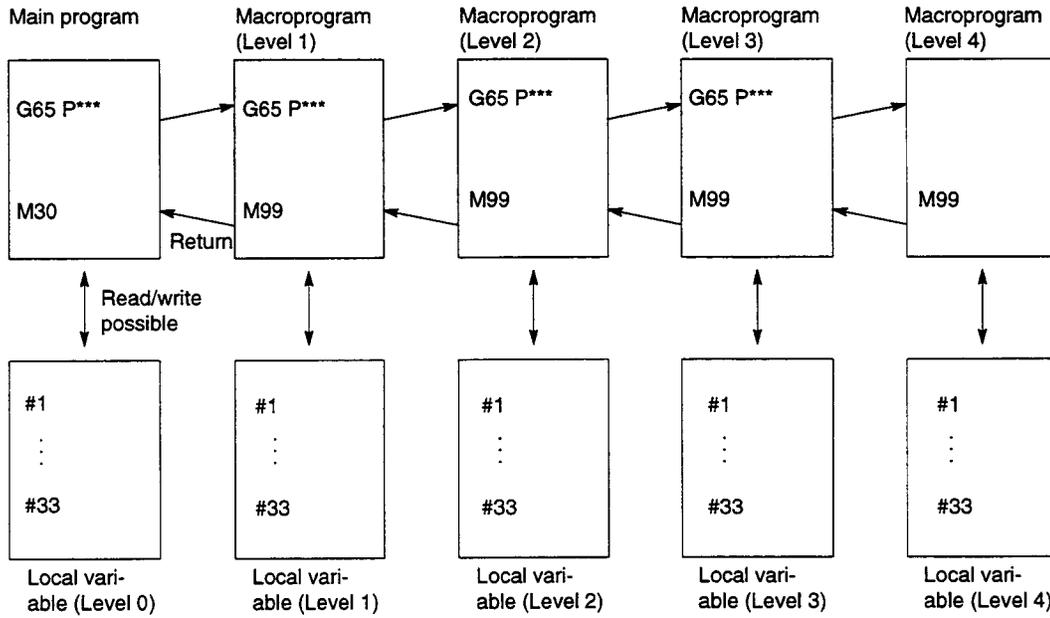


Fig. 4.4.3.1 Local Variables

- ① Local variables of level 0 are secured for the main program. For microprograms, local variables are secured corresponding to the level (level 1 to level 4) of the called up microprogram.
- ② If a microprogram is called up by G65, for example, the local variables used for the program where microprogram call up is executed are saved and the local variables are secured for the called up microprogram corresponding to its level. In this case, the arguments are passed to the called up microprogram.
Consequently, even with the same microprogram, the local variables of the level of that microprogram have different values if the microprogram is called up in different timing.
- ③ When the execution of a microprogram returns to the microprogram one level above by the execution of M99, the local variables of the previous microprogram level are reset to <empty> and the local variables having been saved are recovered.
- ④ You should not change the contents of local variables while a macroprogram is being executed. If they are changed after interrupting the operation by single block stop, make sure that the new contents do not cause problems before restarting the operation.
- ⑤ Local variables can be used in a subprogram. In this case, the local variables of the present macroprogram level are used. Argument specification is not allowed when calling up a subprogram. The contents of the local variables are not reset to <empty>, when the execution of a program returns from the subprogram by the execution of M99.

(2) Common Variables (#100 to #299, #500 to #999)

Common variables means the variables that can be used in common in main programs, sub-programs, microprograms, and those called up in nesting. Therefore, the common variable where the result of an operation executed in a macroprogram is saved can be used in another microprogram. For common variables, argument specification is not allowed.

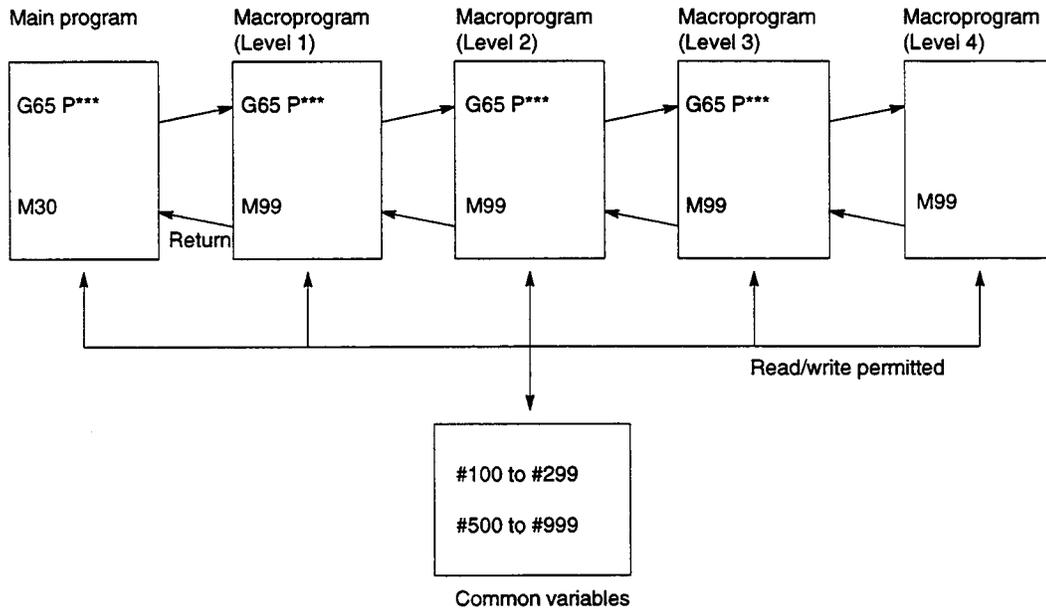


FIGURE 4.4.3.1 Common Variables

- Common variables are classified into two types according to the state they are in when the NC is reset.

Table 4.4.3.1 Common Variables

#100 to #299	The content is <empty:> when the power is turned ON or the NC is reset. By setting parameter “pm4009 D] = 1”, the content is not cleared to <empty>.
#500 to #999	The content is saved and not cleared to <empty> when the power is turned ON or the NC is reset.

- The number of sets of the common variables can be optionally expanded.

Table 4.4.3.2 Option Type and Expanded Common Variables

Option Type	Number of Sets
a	#100 to #149 (50 sets) #500 to #559 (60 sets)
b	#100 to #199 (100 sets) #500 to #599 (100 sets)
c	#100 to #199 (100 sets) #500 to #699 (200 sets)
d	#100 to #299 (200 sets) #500 to #999 (500 sets)

(3) System Variables

With the system variables, their use is predetermined as indicated in Table 4.4.3.3.

Table 4.4.3.3 System Variables

Type of System Variable	System Variable Number
Interface input signals	#1000 to #1031, #1032
Interface output signals	#1100 to #1131, #1132
Tool offset amount work coordinate system shift distance	#2001 to #2299 #12001 to #13199
Alarm message display	#3000
Clock	#3001, #3002
Control for single-block stop, and miscellaneous function complete wait	#3003
Control for feed hold, feedrate override, and exact stop	#3004
RS-232C data output	#3100
Modal information	#4000 to #4999
Position information	#5000 to #5999

(a) Interface input signals

- By entering system variables #1000 to #1031 in the right side of an operation expression, it is possible to read the ON/OFF state of each of 32-point input signal exclusively used for a microprogram.

The relationship between the input signals and system variables is indicated in Table 4.4.3.4.

Table 4.4.3.4 Interface Input Signals and System Variables

System Variables	#1007	#1006	#1005	#1004	#1003	#1002	#1001	#1000
Input Signals	UI 7 2^7	UI 6 2^6	UI 5 2^5	UI 4 2^4	UI 3 2^3	UI 2 2^2	UI 1 2^1	UI 0 2^0
System Variables	#1015	#1014	#1013	#1012	#1011	#1010	#1009	#1008
Input Signals	UI 15 2^{15}	UI 14 2^{14}	UI 13 2^{13}	UI 12 2^{12}	UI 11 2^{11}	UI 10 2^{10}	UI 09 2^9	UI 08 2^8
System Variables	#1023	#1022	#1021	#1020	#1019	#1018	#1017	#1016
Input Signals	UI 23 2^{23}	UI 22 2^{22}	UI 21 2^{21}	UI 20 2^{20}	UI 19 2^{19}	UI 18 2^{18}	UI 17 2^{17}	UI 16 2^{16}
System Variables	#1031	#1030	#1029	#1028	#1027	#1026	#1025	#1024
Input Signals	UI 31 2^{31}	UI 30 2^{30}	UI 29 2^{29}	UI 28 2^{28}	UI 27 2^{27}	UI 26 2^{26}	UI 25 2^{25}	UI 24 2^{24}

The value read to the system variables indicated above is either “1.0” or “0.0” according to the ON/OFF state of the corresponding input signals.

Table 4.4.3.5 Value of Variables

Input Signal State	Variable Value
ON	1.0
OFF	0.0

- By entering system variable #1032 in the right side of an operation expression, it is possible to read the ON/OFF state of all of 32 points of input signals (U10 to U131) collectively as a positive decimal value.

$$\#1032 = \sum_{i=0}^{31} \#[1000 + i] \times 2^i$$

- Note that it is not possible to enter a value by entering a system variable (#1000 to #1032) in the right side of an operation expression.

(b) Interface output signals

- By entering system variables #1100 to #1131 in the right side of an operation expression, it is possible to output the ON/OFF state to each of 32-point output signal exclusively used for a microprogram. The relationship between the output signals and system variables is indicated in Table 4.4.3.6.

Table 4.4.3.6 Interface Output Signals and System Variables

System Variables	#1107	#1106	#1105	#1104	#1103	#1102	#1101	#1100
Output Signals	UO 7 2^7	UO 6 2^6	UO 5 2^5	UO 4 2^4	UO3 2^3	UO 2 2^2	UO 1 2^1	UO 0 2^0
System Variables	#1115	#1114	#1113	#1112	#1111	#1110	#1109	#1108
Output Signals	UO 15 2^{15}	UO 14 2^{14}	UO 13 2^{13}	UO 12 2^{12}	UO 11 2^{11}	UO 10 2^{10}	UO 09 2^9	UO 08 2^8
System Variables	#1123	#1122	#1121	#1120	#1119	#1118	#1117	#1116
Output Signals	UO 23 2^{23}	UO 22 2^{22}	UO 21 2^{21}	UO 20 2^{20}	UO 19 2^{19}	UO 18 2^{18}	UO 17 2^{17}	UO 16 2^{16}
System Variables	#1131	#1130	#1129	#1128	#1127	#1126	#1125	#1124
Output Signals	UO 31 2^{31}	UO 30 2^{30}	UO 29 2^{29}	UO 28 2^{28}	UO 27 2^{27}	UO 26 2^{26}	UO 25 2^{25}	UO 24 2^{24}

By entering “1.0” or “0.0” to the system variables indicated in **Table 4.4.3.7**, the corresponding signals are output in the ON or OFF state.

Table 4.4.3.7 Value of Variables

Output Signal State	Variable Value
ON	1.0
OFF	0.0

- If a value other than “1.0” or “0.0” is set for variables #1100 to #1131, it is treated as indicated below.

<empty> or less than 0.5: 0.0

Other than above: 1.0

- By entering system variable #1132 in the left side of an operation expression, it is possible to output the ON/OFF state to the 32 point output signals (U00 to U031) collectively. In this case, a positive decimal value set for #1132 is output after converted into a binary 32-bit value.

$$\#1132 = \sum_{i=0}^{31} \#[1100 + i] \times 2^i$$

- By entering system variables #1100 to #1132 in the right side of an operation expression, it is possible to read the ON/OFF state (1.0, 0.0, positive decimal value) output last can be read.

(c) Offset amount and workpiece coordinate system distance

Tool offset amount can be read by entering system variables #12001 to #13199 in the right side an operation expression.

Workpiece coordinate system shift distance (and workpiece coordinate system correction amount) can be read by entering system variables #5201 to #5327, #7001 to #7947, and #17821 to #19987 in the right side an operation expression.

By entering the system variables indicated above in the left side of an operation expression, it is possible to update the offset values.

Example of Programming

#116 = #12016: Enters the content of tool offset number 16 to common variable #116.

#5321 = #4: Clears the workpiece coordinate system shift distance of X-axis and sets the content of local variable #4.

(d) Correspondence between the tool offset numbers

The correspondence between the tool offset numbers and the system variables is indicated in Table 4.4.3.8.

Table 4.4.3.8 Tool Offset Numbers and System Variables

	Basic		299-set Option		999-set Option		1199-set Option	
	Offset No.	System Variables	Offset No.	System Variables	Offset No.	System Variables	Offset No.	System Variables
		PCNC		PCNC		PCNC		PCNC
H/D (used in common)	H(D)01	#12001	H(D)01	#12001	H(D)01	#12001	H(D)01	#12001
	H(D)02	#12002	H(D)02	#12002	H(D)02	#12002	H(D)02	#12002
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
	H(D)99	#12099	H(D)299	#12299	H(D)999	#12999	H(D)1199	#13199
H/D (used for individ- ual func- tions)	H01	#12001	H01	#12001	H01	#12001	H01	#12001
	H02	#12002	H02	#12002	H02	#12002	H02	#12002
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
	H49	#12049	H149	#12149	H499	#12499	H599	#12549
	D01	#12051	D01	#12151	D01	#12501	D01	#12601
	D02	#12052	D02	#12152	D02	#12502	D02	#12602
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
	•	•	•	•	•	•	•	•
D49	#12099	D149	#12299	D999	#12999	D1199	#13199	

(e) Tool wear offset correspondence table

The correspondence between the tool wear offset numbers and the system variables is indicated in Table 4.4.3.9 for the NC equipped with the tool wear offset option.

Table 4.4.3.9 Tool Offset Numbers and System Variables

		Basic		299-set Option		999-set Option		1199-set Option	
		Offset No.	System Variables	Offset No.	System Variables	Offset No.	System Variables	Offset No.	System Variables
			PCNC		PCNC		PCNC		PCNC
H/D (used in common)	Geom-etry	H(D)01	#12001	H(D)01	#12001	H(D)01	#12001	H(D)01	#12001
		•	•	•	•	•	•	•	•
		•	•	•	•	•	•	•	•
		•	•	•	•	•	•	•	•
	H(D)49	#12049	H(D)149	#12149	H(D)499	#12499	H(D)599	#12599	
	Wear	H(D)50	#12050	H(D)150	#12150	H(D)500	#12500	H(D)600	#12600
		•	•	•	•	•	•	•	•
		•	•	•	•	•	•	•	•
•		•	•	•	•	•	•	•	
H(D)99	#12099	H(D)299	#12299	H(D)999	#12999	H(D)1199	#13199		
H/D (used for individual functions)	Geom-etry	H01	#12001	H01	#12001	H01	#12001	H01	#12001
		•	•	•	•	•	•	•	•
		•	•	•	•	•	•	•	•
	H24	#12024	H74	#12074	H249	#12249	H299	#12299	
	Wear	H01	#12026	H01	#12076	H01	#12251	H01	#12301
		•	•	•	•	•	•	•	•
		•	•	•	•	•	•	•	•
	H24	#12049	H74	#12149	H249	#12499	H299	#12599	
	Geom-etry	D01	#12051	D01	#12151	D01	#12501	D01	#12601
		•	•	•	•	•	•	•	•
		•	•	•	•	•	•	•	•
	D24	#12074	D74	#12224	D249	#12749	D299	#12899	
Wear	D01	#12076	D01	#12226	D01	#12751	D01	#12701	
	•	•	•	•	•	•	•	•	
	•	•	•	•	•	•	•	•	
D24	#12099	D74	#12299	D249	#12999	D299	#13199		

(f) Correspondence between work coordinate system shift distance and system variables

The correspondence between the workpiece coordinate system shift distance and the system variables is indicated in Table 4.4.3.10.

Table 4.4.3.10 Workpiece Coordinate Shift Distance and System Variables (6-set Basic)

System Variables			System Variables		
		PCNC			PCNC
Correction amount	X	#5201	G59 J1	X	#5321
	Y	#5202		Y	#5322
	Z	#5203		Z	#5323
	4	#5204		4	#5324
	5	#5205		5	#5325
	R	#5207		R	#5327
G54 J1	X	#5221			
	Y	#5222			
	Z	#5223			
	4	#5224			
	5	#5225			
	R	#5227			
G55 J1	X	#5241			
	Y	#5242			
	Z	#5243			
	4	#5244			
	5	#5245			
	R	#5247			
G56 J1	X	#5261			
	Y	#5262			
	Z	#5263			
	4	#5264			
	5	#5265			
	R	#5267			
G57 J1	X	#5281			
	Y	#5282			
	Z	#5283			
	4	#5284			
	5	#5285			
	R	#5327			
G58 J1	X	#5301			
	Y	#5302			
	Z	#5303			
	4	#5304			
	5	#5305			
	R	#5307			

**Table 4.4.3.11 Workpiece Coordinate Shift Distance and System Variables
[54-set Option (6-set Basic+ Below)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J2	X	#7001	G54 J3	X	#7121	G54 J4	X	#7241
	Y	#7002		Y	#7122		Y	#7242
	Z	#7003		Z	#7123		Z	#7243
	4	#7004		4	#7124		4	#7244
	5	#7005		5	#7125		5	#7245
	R	#7007		R	#7127		R	#7247
G55 J2	X	#7021	G55 J3	X	#7141	G55 J4	X	#7261
	Y	#7022		Y	#7142		Y	#7262
	Z	#7023		Z	#7143		Z	#7263
	4	#7024		4	#7144		4	#7264
	5	#7025		5	#7145		5	#7265
	R	#7027		R	#7147		R	#7267
G56 J2	X	#7041	G56 J3	X	#7161	G56 J4	X	#7281
	Y	#7042		Y	#7162		Y	#7282
	Z	#7043		Z	#7163		Z	#7283
	4	#7044		4	#7164		4	#7284
	5	#7045		5	#7165		5	#7285
	R	#7047		R	#7167		R	#7287
G57 J2	X	#7061	G57 J3	X	#7181	G57 J4	X	#7301
	Y	#7062		Y	#7182		Y	#7302
	Z	#7063		Z	#7183		Z	#7303
	4	#7064		4	#7184		4	#7304
	5	#7065		5	#7185		5	#7305
	R	#7067		R	#7187		R	#7307
G58 J2	X	#7081	G58 J3	X	#7201	G58 J4	X	#7321
	Y	#7082		Y	#7202		Y	#7322
	Z	#7083		Z	#7203		Z	#7323
	4	#7084		4	#7204		4	#7324
	5	#7085		5	#7205		5	#7325
	R	#7027		R	#7207		R	#7327
G59 J2	X	#7101	G59 J3	X	#7221	G59 J4	X	#7341
	Y	#7102		Y	#7222		Y	#7342
	Z	#7103		Z	#7223		Z	#7343
	4	#7104		4	#7224		4	#7344
	5	#7105		5	#7225		5	#7345
	R	#7107		R	#7227		R	#7347

**Table 4.4.3.11 Workpiece Coordinate Shift Distance and System Variables
[54-set Option (6-set Basic+ Below) (cont'd)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J5	X	#7361	G54 J6	X	#7481	G54 J7	X	#7601
	Y	#7362		Y	#7482		Y	#7602
	Z	#7363		Z	#7483		Z	#7603
	4	#7364		4	#7484		4	#7604
	5	#7365		5	#7485		5	#7605
	R	#7367		R	#7487		R	#7607
G55 J5	X	#7381	G55 J6	X	#7501	G55 J7	X	#7621
	Y	#7382		Y	#7502		Y	#7622
	Z	#7383		Z	#7503		Z	#7623
	4	#7384		4	#7504		4	#7624
	5	#7385		5	#7505		5	#7625
	R	#7387		R	#7507		R	#7627
G56 J5	X	#7401	G56 J6	X	#7521	G56 J7	X	#7641
	Y	#7402		Y	#7522		Y	#7642
	Z	#7403		Z	#7523		Z	#7643
	4	#7404		4	#7524		4	#7644
	5	#7405		5	#7525		5	#7645
	R	#7407		R	#7527		R	#7647
G57 J5	X	#7421	G57 J6	X	#7541	G57 J7	X	#7661
	Y	#7422		Y	#7542		Y	#7662
	Z	#7423		Z	#7543		Z	#7663
	4	#7424		4	#7544		4	#7664
	5	#7425		5	#7545		5	#7665
	R	#7427		R	#7547		R	#7667
G58 J5	X	#7441	G58 J6	X	#7561	G58 J7	X	#7681
	Y	#7442		Y	#7562		Y	#7682
	Z	#7443		Z	#7563		Z	#7683
	4	#7444		4	#7564		4	#7684
	5	#7445		5	#7565		5	#7685
	R	#7447		R	#7567		R	#7687
G59 J5	X	#7461	G59 J6	X	#7581	G59 J7	X	#7701
	Y	#7462		Y	#7582		Y	#7702
	Z	#7463		Z	#7583		Z	#7703
	4	#7464		4	#7584		4	#7704
	5	#7465		5	#7585		5	#7705
	R	#7467		R	#7587		R	#7707

**Table 4.4.3.11 Workpiece Coordinate Shift Distance and System Variables
[54-set Option (6-set Basic+ Below) (cont'd)]**

System Variables			System Variables		
		PCNC			PCNC
G54 J8	X	#7721	G54 J9	X	#7841
	Y	#7722		Y	#7842
	Z	#7723		Z	#7843
	4	#7724		4	#7844
	5	#7725		5	#7845
	R	#7727		R	#7847
G55 J8	X	#7741	G55 J9	X	#7861
	Y	#7742		Y	#7862
	Z	#7743		Z	#7863
	4	#7744		4	#7864
	5	#7745		5	#7865
	R	#7747		R	#7867
G56 J8	X	#7761	G56 J9	X	#7881
	Y	#7762		Y	#7882
	Z	#7763		Z	#7883
	4	#7764		4	#7884
	5	#7765		5	#7885
	R	#7767		R	#7887
G57 J8	X	#7781	G57 J9	X	#7901
	Y	#7782		Y	#7902
	Z	#7783		Z	#7903
	4	#7784		4	#7904
	5	#7785		5	#7905
	R	#7787		R	#7907
G58 J8	X	#7801	G58 J9	X	#7921
	Y	#7802		Y	#7922
	Z	#7803		Z	#7923
	4	#7804		4	#7924
	5	#7805		5	#7925
	R	#7807		R	#7927
G59 J8	X	#7821	G59 J9	X	#7941
	Y	#7822		Y	#7942
	Z	#7823		Z	#7943
	4	#7824		4	#7944
	5	#7825		5	#7945
	R	#7827		R	#7947

**Table 4.4.3.12 Workpiece Coordinate Shift Distance and System Variables
[162-set Option (54-set Option + Below)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J10	X	#17821	G54 J11	X	#17941	G54 J12	X	#18061
	Y	#17822		Y	#17942		Y	#18062
	Z	#17823		Z	#17943		Z	#18063
	4	#17824		4	#17944		4	#18064
	5	#17825		5	#17945		5	#18065
	R	#17827		R	#17947		R	#18067
G55 J10	X	#17841	G55 J11	X	#17961	G55 J12	X	#18081
	Y	#17842		Y	#17962		Y	#18082
	Z	#17843		Z	#17963		Z	#18083
	4	#17844		4	#17964		4	#18084
	5	#17845		5	#17965		5	#18085
	R	#17847		R	#17967		R	#18087
G56 J10	X	#17861	G56 J11	X	#17981	G56 J12	X	#18101
	Y	#17862		Y	#17982		Y	#18102
	Z	#17863		Z	#17983		Z	#18103
	4	#17864		4	#17984		4	#18104
	5	#17865		5	#17985		5	#18105
	R	#17867		R	#17987		R	#18107
G57 J10	X	#17881	G57 J11	X	#18001	G57 J12	X	#18121
	Y	#17882		Y	#18002		Y	#18122
	Z	#17883		Z	#18003		Z	#18123
	4	#17884		4	#18004		4	#18124
	5	#17885		5	#18005		5	#18125
	R	#17887		R	#18007		R	#18127
G58 J10	X	#17901	G58 J11	X	#18021	G58 J12	X	#18141
	Y	#17902		Y	#18022		Y	#18142
	Z	#17903		Z	#18023		Z	#18143
	4	#17904		4	#18024		4	#18144
	5	#17905		5	#18025		5	#18145
	R	#17907		R	#18027		R	#18147
G59 J10	X	#17921	G59 J11	X	#18041	G59 J12	X	#18161
	Y	#17922		Y	#18042		Y	#18162
	Z	#17923		Z	#18043		Z	#18163
	4	#17924		4	#18044		4	#18164
	5	#17925		5	#18045		5	#18165
	R	#17927		R	#18047		R	#18167

**Table 4.4.3.12 Workpiece Coordinate Shift Distance and System Variables
[162-set Option (54-set Option + Below) (cont'd)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J13	X	#18181	G54 J14	X	#18301	G54 J15	X	#18421
	Y	#18182		Y	#18302		Y	#18422
	Z	#18183		Z	#18303		Z	#18423
	4	#18184		4	#18304		4	#18424
	5	#18185		5	#18305		5	#18425
	R	#18187		R	#18307		R	#18427
G55 J13	X	#18201	G55 J14	X	#18321	G55 J15	X	#18441
	Y	#18202		Y	#18322		Y	#18442
	Z	#18203		Z	#18323		Z	#18443
	4	#18204		4	#18324		4	#18444
	5	#18205		5	#18325		5	#18445
	R	#18207		R	#18327		R	#18447
G56 J13	X	#18221	G56 J14	X	#18341	G56 J15	X	#18461
	Y	#18222		Y	#18342		Y	#18462
	Z	#18223		Z	#18343		Z	#18463
	4	#18224		4	#18344		4	#18464
	5	#18225		5	#18345		5	#18465
	R	#18227		R	#18347		R	#18467
G57 J13	X	#18241	G57 J14	X	#18361	G57 J15	X	#18481
	Y	#18242		Y	#18362		Y	#18482
	Z	#18243		Z	#18363		Z	#18483
	4	#18244		4	#18364		4	#18484
	5	#18245		5	#18365		5	#18485
	R	#18247		R	#18367		R	#18487
G58 J13	X	#18261	G58 J14	X	#18381	G58 J15	X	#18501
	Y	#18262		Y	#18382		Y	#18502
	Z	#18263		Z	#18383		Z	#18503
	4	#18264		4	#18384		4	#18504
	5	#18265		5	#18385		5	#18505
	R	#18267		R	#18387		R	#18507
G59 J13	X	#18281	G59 J14	X	#18401	G59 J15	X	#18521
	Y	#18282		Y	#18402		Y	#18522
	Z	#18283		Z	#18403		Z	#18523
	4	#18284		4	#18404		4	#18524
	5	#18285		5	#18405		5	#18525
	R	#18287		R	#18407		R	#18527

**Table 4.4.3.12 Workpiece Coordinate Shift Distance and System Variables
[162-set Option (54-set Option + Below) (cont'd)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J16	X	#18541	G54 J17	X	#18661	G54 J18	X	#18781
	Y	#18542		Y	#18662		Y	#18782
	Z	#18543		Z	#18663		Z	#18783
	4	#18544		4	#18664		4	#18784
	5	#18545		5	#18665		5	#18785
	R	#18547		R	#18667		R	#18787
G55 J16	X	#18561	G55 J17	X	#18681	G55 J18	X	#18801
	Y	#18562		Y	#18682		Y	#18802
	Z	#18563		Z	#18683		Z	#18803
	4	#18564		4	#18684		4	#18804
	5	#18565		5	#18685		5	#18805
	R	#18567		R	#18687		R	#18807
G56 J16	X	#18581	G56 J17	X	#18701	G56 J18	X	#18821
	Y	#18582		Y	#18702		Y	#18822
	Z	#18583		Z	#18703		Z	#18823
	4	#18584		4	#18704		4	#18824
	5	#18585		5	#18705		5	#18825
	R	#18587		R	#18707		R	#18827
G57 J16	X	#18601	G57 J17	X	#18721	G57 J18	X	#18841
	Y	#18602		Y	#18722		Y	#18842
	Z	#18603		Z	#18723		Z	#18843
	4	#18604		4	#18724		4	#18844
	5	#18605		5	#18725		5	#18845
	R	#18607		R	#18727		R	#18847
G58 J16	X	#18621	G58 J17	X	#18741	G58 J18	X	#18861
	Y	#18622		Y	#18742		Y	#18862
	Z	#18623		Z	#18743		Z	#18863
	4	#18624		4	#18744		4	#18864
	5	#18625		5	#18745		5	#18865
	R	#18627		R	#18747		R	#18867
G59 J16	X	#18641	G59 J17	X	#18761	G59 J18	X	#18881
	Y	#18642		Y	#18762		Y	#18882
	Z	#18643		Z	#18763		Z	#18883
	4	#18644		4	#18764		4	#18884
	5	#18645		5	#18765		5	#18885
	R	#18647		R	#18767		R	#18887

**Table 4.4.3.12 Workpiece Coordinate Shift Distance and System Variables
[54-set Option (6-set Basic+ Below) (cont'd)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J19	X	#18921	G54 J20	X	#19041	G54 J21	X	#19161
	Y	#18922		Y	#19042		Y	#19162
	Z	#18923		Z	#19043		Z	#19163
	4	#18924		4	#19044		4	#19164
	5	#18925		5	#19045		5	#19165
	R	#18927		R	#19047		R	#19167
G55 J19	X	#18941	G55 J20	X	#19061	G55 J21	X	#19181
	Y	#18942		Y	#19062		Y	#19182
	Z	#18943		Z	#19063		Z	#19183
	4	#18944		4	#19064		4	#19184
	5	#18945		5	#19065		5	#19185
	R	#18947		R	#19067		R	#19187
G56 J19	X	#18961	G56 J20	X	#19081	G56 J21	X	#19201
	Y	#18962		Y	#19082		Y	#19202
	Z	#18963		Z	#19083		Z	#19203
	4	#18964		4	#19084		4	#19204
	5	#18965		5	#19085		5	#19205
	R	#18967		R	#19087		R	#19207
G57 J19	X	#18981	G57 J20	X	#19101	G57 J21	X	#19221
	Y	#18982		Y	#19102		Y	#19222
	Z	#18983		Z	#19103		Z	#19223
	4	#18984		4	#19104		4	#19224
	5	#18985		5	#19105		5	#19225
	R	#18987		R	#19107		R	#19227
G58 J19	X	#19001	G58 J20	X	#19121	G58 J21	X	#19241
	Y	#19002		Y	#19122		Y	#19242
	Z	#19003		Z	#19123		Z	#19243
	4	#19004		4	#19124		4	#19244
	5	#19005		5	#19125		5	#19245
	R	#19007		R	#19127		R	#19247
G59 J19	X	#19021	G59 J20	X	#19141	G59 J21	X	#19261
	Y	#19022		Y	#19142		Y	#19262
	Z	#19023		Z	#19143		Z	#19263
	4	#19024		4	#19144		4	#19264
	5	#19025		5	#19145		5	#19265
	R	#19027		R	#19147		R	#19267

**Table 4.4.3.12 Workpiece Coordinate Shift Distance and System Variables
[162-set Option (54-set Option + Below) (cont'd)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J22	X	#19281	G54 J23	X	#19401	G54 J24	X	#19521
	Y	#19282		Y	#19402		Y	#19522
	Z	#19283		Z	#19403		Z	#19523
	4	#19284		4	#19404		4	#19524
	5	#19285		5	#19405		5	#19525
	R	#19287		R	#19407		R	#19527
G55 J22	X	#19301	G55 J23	X	#19421	G55 J24	X	#19541
	Y	#19302		Y	#19422		Y	#19542
	Z	#19303		Z	#19423		Z	#19543
	4	#19304		4	#19424		4	#19544
	5	#19305		5	#19425		5	#19545
	R	#19307		R	#19427		R	#19547
G56 J22	X	#19321	G56 J23	X	#19441	G56 J24	X	#19561
	Y	#19322		Y	#19442		Y	#19562
	Z	#19323		Z	#19443		Z	#19563
	4	#19324		4	#19444		4	#19564
	5	#19325		5	#19445		5	#19565
	R	#19327		R	#19447		R	#19567
G57 J22	X	#19341	G57 J23	X	#19461	G57 J24	X	#19581
	Y	#19342		Y	#19462		Y	#19582
	Z	#19343		Z	#19463		Z	#19583
	4	#19344		4	#19464		4	#19584
	5	#19345		5	#19465		5	#19585
	R	#19347		R	#19467		R	#19587
G58 J22	X	#19361	G58 J23	X	#19481	G58 J24	X	#19601
	Y	#19362		Y	#19482		Y	#19602
	Z	#19363		Z	#19483		Z	#19603
	4	#19364		4	#19484		4	#19604
	5	#19365		5	#19485		5	#19605
	R	#19367		R	#19487		R	#19607
G59 J22	X	#19381	G59 J23	X	#19501	G59 J24	X	#19621
	Y	#19382		Y	#19502		Y	#19622
	Z	#19383		Z	#19503		Z	#19623
	4	#19384		4	#19504		4	#19624
	5	#19385		5	#19505		5	#19625
	R	#19387		R	#19507		R	#19627

**Table 4.4.3.12 Workpiece Coordinate Shift Distance and System Variables
[162-set Option (54-set Option + Below) (cont'd)]**

System Variables			System Variables			System Variables		
		PCNC			PCNC			PCNC
G54 J25	X	#19641	G54 J26	X	#19761	G54 J27	X	#19881
	Y	#19642		Y	#19762		Y	#19882
	Z	#19643		Z	#19763		Z	#19883
	4	#19644		4	#19764		4	#19884
	5	#19645		5	#19765		5	#19885
	R	#19647		R	#19767		R	#19887
G55 J25	X	#19661	G55 J26	X	#19781	G55 J27	X	#19901
	Y	#19662		Y	#19782		Y	#19902
	Z	#19663		Z	#19783		Z	#19903
	4	#19664		4	#19784		4	#19904
	5	#19665		5	#19785		5	#19905
	R	#19667		R	#19787		R	#19907
G56 J25	X	#19681	G56 J26	X	#19801	G56 J27	X	#19921
	Y	#19682		Y	#19802		Y	#19922
	Z	#19683		Z	#19803		Z	#19923
	4	#19684		4	#19804		4	#19924
	5	#19685		5	#19805		5	#19925
	R	#19687		R	#19807		R	#19927
G57 J25	X	#19701	G57 J26	X	#19821	G57 J27	X	#19941
	Y	#19702		Y	#19822		Y	#19942
	Z	#19703		Z	#19823		Z	#19943
	4	#19704		4	#19824		4	#19944
	5	#19705		5	#19825		5	#19945
	R	#19707		R	#19827		R	#19947
G58 J25	X	#19721	G58 J26	X	#19841	G58 J27	X	#19961
	Y	#19722		Y	#19842		Y	#19962
	Z	#19723		Z	#19843		Z	#19963
	4	#19724		4	#19844		4	#19964
	5	#19725		5	#19845		5	#19965
	R	#19727		R	#19847		R	#19967
G59 J25	X	#19741	G59 J26	X	#19861	G59 J27	X	#19981
	Y	#19742		Y	#19862		Y	#19982
	Z	#19743		Z	#19863		Z	#19983
	4	#19744		4	#19864		4	#19984
	5	#19745		5	#19865		5	#19985
	R	#19747		R	#19867		R	#19987

(g) Alarm message display

By specifying “#3000 = <Alarm-number> (<alarm-message>);”, the NC can be placed in the alarm state. The timing the NC is placed in the alarm state is after the completion of the commands in the block immediately preceding the block including the commands indicated above.

<Alarm-number>: A 4-digit alarm number not used by NC.

Use of a variable is allowed.

(Alarm number range: 5000 to 5999)

<Alarm-message>: ASCII character string with 32 or less characters

(alphanumeric and special characters)

(h) Clock

It is possible to read time by entering the system variable used for the clock in the right side of an operation expression. If such a system variable is entered in the left side of an operation expression, it is possible to preset the time.

Table 4.4.3.13 System Variables Used for Clock Function

Type	System Variable	Unit	At Power ON	Count Conditions
Clock 1	#3001	1 msec	Preset to “0”	Always
Clock 2	#3002	1 sec	The state immediately before the power was turned OFF is retained	When the STL signal goes ON

(i) Control for single-block stop and waiting for completion of miscellaneous function

By setting an appropriate number for system variable #3003, the following control is possible:

- To make valid/invalid the SINGLE-BLOCK switch setting for the succeeding blocks.
- To advance the program to the next block without waiting for the input of the miscellaneous function (M, S, T, B) completion signal (FIN).

If a miscellaneous function is specified with the setting that the input of the completion signal (FIN) is not checked, the distribution complete signal (DEN) is not output and the program advances to the next block without waiting for the input of the FIN signal although the output of the M, S, T, or B code and M, S, T, or B read output is executed as normal. When the block in which the setting is made for system variable #3003 so that the state for checking the input of the miscellaneous function complete signal is executed after that, the NC outputs DEN signal and waits for the input of the FIN signal.

When M, S, T, or B is specified in the state that the miscellaneous function complete is not checked, the first appearance of individual codes is recognized and executed.

Then, the second and later appearance are disregarded until the setting is changed to

the state in which the complete signal is checked. In the state that the miscellaneous function complete signal is not checked, the program advances to the next block without waiting for the input of the complete signal even when the buffering stop M code (M00, M01, M02, M30) is specified.

When the NC is reset, the setting for #3003 is “0”.

Table 4.4.3.14 Control for Single-Block Stop and Miscellaneous Function Completion Waiting

#3003	SINGLE-BLOCK Switch	Miscellaneous Function Complete Signal (FIN)
0	Valid	Checked
1	Invalid	Checked
2	Valid	Not checked
3	Invalid	Not checked

Example of Programming

```

M10;      ← M10 code: MF output, DEN output, FIN checked
  ⋮
  #3003 = 3
  ⋮
M20;      ← M20 code: MF not output, FIN not checked
  ⋮
M25;      ← Disregarded
  ⋮
  #3003 = 0 ← DEN output, FIN checked
  ⋮
M40;      ← M40 code: MF output, DEN output, FIN checked
    
```

Single-block stop FIN check not made {

} "M" on display

(j) Setting for feed hold, feedrate override, and positioning completion control

For the control of feed hold, feedrate override, and positioning completion, system variable #3004 is provided and by setting an appropriate value for this system variable, it is possible to make these functions valid or invalid.

When the NC is reset, the setting is reset to “#3004 = 0”

Table 4.4.3.15 Control for Feed Hold, Feedrate Override, and Positioning Completion Functions

#3004	Feed Hold	Feedrate Override	Positioning Completion
0	Valid	Valid	Valid
1	Invalid	Valid	Valid
2	Valid	Invalid	Valid
3	Invalid	Invalid	Valid
4	Valid	Valid	Invalid
5	Invalid	Valid	Invalid
6	Valid	Invalid	Invalid
7	Invalid	Invalid	Invalid

- For the feed hold function

The feed hold function is invalid in the following blocks.

From the block where 1, 3, 5, or 7 is set for #3004

To the block where 0, 2, 4, or 6 is set for #3004

The blocks for which the feed hold is made invalid are not accepted and the feed hold signal is not output.

- For feedrate override

The setting of feedrate override is disregarded in the following blocks.

From the block where 2, 3, 6, or 7 is set for #3004

To the block where 0, 1, 4, or 5 is set for #3004

- For the positioning completion function

The check is not made for the completion of positioning.

From the block where 4, 5, 6, or 7 is set for #3004

To the block where 0, 1, 2, or 3 is set for #3004

Example of Programming for Tapping Cycle (incremental mode)

Main program

```
G91;
G65 P9601 L... R... Z...;
```

Designation of argument for #26

Designation of argument for #18

Macroprogram

```
O9601;
#3003 = 1;
G00 Z#18;
#3004 = 7
G01 Z#26;
M05;
M04;
#3004 = 0;
G00 Z - #18
M05;
M03;
#3003 = 0;
M99;
```

SINGLE-BLOCK switch made invalid

Feed hold
Feedrate override
Exact stop } Made invalid

(k) Modal information

By entering the system variables indicated below in the right side of an operation expression, it is possible to read the modal value given in blocks up to the immediately preceding block. Note that these system variables cannot be entered to the left side of an operation expression.

Table 4.4.3.16 Modal Values and System Variables

Modal Command	System Variable
G code (01-group) to (*-group)	#4001 to #4024
J (P) code	Extended workpiece coordinate system #4080
B code	#4102
D code	#4107
E code	#4108
F code	#4109
H code	#4111
Sequence number	#4114
Program number	#4115
S code	#4119
T code	#4120

- Note
- 1: Since an M code is non-modal information, it is not possible to read M codes using system variables.
 - 2: Concerning E (#4108) and F (#4109), either the E or F command specified immediately before the specification of the system variable is saved. Therefore, system variables #4108 and #4109 hold the same value.

(l) Position information

By specifying the system variables indicated below, it is possible to read the position information.

Note that these system variables cannot be specified in the left side of an operation expression.

Table 4.4.3.17 Position Information and System Variables

Position Information	System Variable	Reading during Operation
X-axis block end point position (ABSIO) to 5th-axis block end point position (ABSIO)	#5001 to #5005	Possible
X-axis position in the machine coordinate system (ABSMT) to 5th-axis position in the machine coordinate system (ABSMT)	#5021 to #5025	Possible (note)
X-axis POS.ABS position (ABSOT) to 5th-axis POS.ABS position (ABSOT)	#5041 to #5045	Possible (note)
X-axis skip signal input position (ABSKP) to 5th-axis skip signal input position (ABSKP)	#5061 to #5065	Possible
Z-axis offset	#5083	Possible
X-axis servo position error to 5th-axis servo position error	#5101 to #5105	Possible (note)

Note: When the system variable indicated by (note) is specified, the position information is read after the completion of the commands specified in the immediately preceding block

Table 4.4.3.18

Abbreviation	ABSIO	ABSMT	ABSOT	ABSKP
Description	End point position of the immediately preceding block	Present position of the command (the same value as the coordinate value in the machine coordinate system)	Present position of the command (the same value as the present position data)	The position where the skip signal is turned ON (G31 block)
Coordinate system	Workpiece coordinate system	Machine coordinate system	Workpiece coordinate system	Workpiece coordinate system
Tool offset amount	Not included	-	Included	Included

Note: The unit of the position information is the specified mm or inch input unit.

Table 4.4.3.19

	Unit	
	Microns	Sub Microns
Input in mm	0.001mm	0.0001mm
Input in inches	0.0001 inch	0.00001 inch
Input in degrees	0.001 deg	0.0001 deg

Note: If the skip signal is turned ON during the execution of the G31 block, the end point position of this block is the skip signal input position.

If the skip signal is not turned ON, the skip signal input position is the end point of the G31 block.

(m) NC parameters and battery back-up memory

It is possible to read the contents of the NC parameters and the battery back-up memory.

To read them, the system variable memory number to be specified is “50000 + NC parameter number/memory number”,

(Example)

System Variable Type	System Variable No.
NC parameter	#50000 to #56949
Battery back-up memory	#57100 to #57999

Note: To read the value saved at parameter pm6000, specify as indicated below,
#100 to #56000

(4) Expression of Variables

Variables are expressed by variable numbers or alphanumeric specified following #.

- Specifying a variable number directly
#i (i: variable number)
(Example) #1, #101, #501, #2001
- Specifying an expression as a variable number
[<expression>]
(Example) # [#101], # [#501+1], # [#1/2]

(5) Assigning Variables

A numeric value specified following an address can be replaced with a variable.

By specifying “<address> #i or <address> – #i”, the value of the specified variable or its negative value (complement) can be taken as the command value of that address.

(Example)

G#30, #30 = 1.0	Equivalent to G01
X#101, #101 = 100.	Equivalent to X100.
Z#101 #103 = 300.	Equivalent to Z300.
F#140, #140 = 0.3	Equivalent to G0.3

- For the following addresses, it is not allowed to assign a variable.

(Example)

/#5	It is not allowed to use a variable for “n” of “/n” (n= 1 to 9).
#100	It is not allowed to use a variable for O number (program number).
N #200	If it not allowed to use a variable for N number (sequence number).

- It is not allowed to use a variable to express a variable number.
- When replacing “10” in #10 with #20, for example, expression of ##20 is not allowed. This must be written by # [#20].
- If a variable is used as the address data, values below the minimum input unit are rounded off.

(Example)

X#1, #1 = 45.2346	X45.235 mm (0.001 mm input unit)
F#2, #2 = 0.255	F0.26 m/rev (F32 format)
G04 P#4, #3 = 5.37672	G04 P5.377 sec
M#4, #4 = 2.7236	M03
G#4, #4= 2.7236	G03

- It is possible to use <expression> instead of a numeric value to be assigned to an address.

By specifying “<address> [<expression>], or <address> – [<expression>]”, the value or negative value (complement) of the <expression> can be used as the command value for that address.

- The constant used in [] without a decimal point is assumed to have a decimal point at the end.

(6) Undefined Variables

Variables which have not been defined yet are called undefined variables, and their values are <empty>. The following variables are treated as undefined variables.

- Local variables and common variables (#100 to #299) when the power is turned ON or the NC is reset.
- Local variables for which arguments are not specified where a microprogram is called up.
- Local variables which belong to the level of the microprogram from which the execution of program returns by the execution of M99.
- Local variables and common variables where no values have been set in a microprogram.
- Common variables where no values have been set at the NC operation panel.
- Variable “#0”. (This is always treated as <empty> and must not be entered in the left side of an operation expression.)

(a) Meaning of <empty>

- If an undefined variable is assigned, the address itself for which it is assigned is disregarded.

#2 = <empty>	G00X#2; is equivalent to G00;
#2 = 0	G00X#2; is equivalent to G00X0;

- If an undefined variable is used in an operation expression, it is treated to have the variable value of “0” with the exception that it is replaced with <empty>.

#2 = <empty>	#3 = #2; indicates #3 = <empty>.
#2 = <empty>	#3 = # [#2+#2]; indicates #3 = #0 = <empty>.
#2 = <empty>	#3 = #1*#2; indicates #3 = <empty>.
#2 = <empty>	#3 = #2+#2; indicates #3 = <empty>.
#2 = <empty>	#3 = #2/#2; indicates #3 = <empty>.
#2 = <empty>	#3 = 5*#2; indicates #3 = <empty>.
#2 = <empty>	#3 = 2-#2; indicates #3 = 2.
#2 = <empty>	#3 = 5/#2 causes division error.

- If an undefined variable is used in a conditional expression, it is treated to have the variable value of “0” with an exception of EQ and NE.

#3 = 0, #2 = <empty>	#3EQ #2: Not satisfied
#3 = 0, #2 = <empty>	#3NE #2: Satisfied
#3 = 0, #2 = <empty>	#3GE #2: Satisfied
#3 = 0, #2 = <empty>	#3LT #2: Not satisfied.

4.4.4 Operation Instructions

By performing general arithmetic operations in which local variables, common variables, system variables, and constants are connected with operators and functions, it is possible to set the result of operation to the given variable.

The variables used in the arithmetic operation read the required data from the internal variable data area. The result of the operation is set to a variable to write the result of the operation to the internal variable data area. The write cycle is completed when the execution of one block is completed.

The basic formula of arithmetic operation is “#i = <expression>”. The following operations and functions can be used.

(1) Definition and Setting of Variables

#i = ##j	Definition or setting
#i = # [#j = #k]	Indirect designation

(2) Addition Type Operations

#i = #j + #k	Sum
#i = #j - #k	Difference
#i = #j OR #k	Logical sum (for each bit in 32-bit binary)
#i = #j XOR #k	Exclusive logical sum (for each bit in 32-bit binary)

(3) Multiplication Type Operation

#i = #j*#k	Product
#i = #j/#k	Quotient
#i = #j AND #k	Logical product (for each bit in 32-bit binary)
#i = #j MOD #k	Remainder (With #j and #k, remainder is obtained after rounding the values to an integer. If #j is negative, #i is also negative.)

(4) Function

#i = SIN [#j]	Sine (in units of degrees)
#i = COS [#j]	Cosine (in units of degrees)
#i = TAN [#j]	Tangent (in units of degrees)
#i = ATAN [#j] or #i = ATAN [#j/#k]	Arctangent
#i = SQRT [#j]	Square root
#i = ABS [#j]	Absolute value
#i = BIN [#j]	Conversion from BCD to binary
#i = BCD [#j]	Conversion from binary to BCD
#i = ROUND [#j]	Conversion into integer by rounding off
#i = FIX [#j]	Cutting off decimal fractions
#i = FUP [#j]	Rounding off decimal fractions I
#i = ASIN [#j]	Arcsine
#i = ACOS [#j]	Arccosine
#i = LN [#j]	Natural logarithm
#i = EXP [#j]	Exponent with e (= 2.718 •••) as a base

(5) Combination of Operations

It is possible to combine the operations and functions explained in items (1) to (4) above.

In this case, the priority of operation is in the order of functions, multiplication type operation and addition type operation.

(Example) #i =#j + #k * SIN [#l]
 ① ② ③

(6) Changing the Order of Operations by []

By enclosing a part of an expression by brackets ([]), that part is given priority for calculation. The brackets can be nested in up to five levels including the brackets used in functions.

(Example) #i =SIN [[#j+#k] * #l + #m]* #n
 ⑤ ① ② ③ ④

(7) Supplements to the Operation Instructions

- A constant used in <expression> without a decimal point is assumed to have a decimal point at the end. The allowable range of the constant is + 99999999.99999999.
- Function ROUND converts a value into integer by rounding off processing. This processing is executed at the digit indicated below.
 - If used in an operation instruction, conditional expression IF or WHILE, decimal fractions are rounded off.
 - If used in address data, the value is rounded off at the digit one place below the minimum input unit of the address.

(Example 1) When #10 = 12.3758,

#1 = ROUND [#10] ? #1 = 12.0

ROUND [#10] in IF [#10 GT ROUND [#10] ‡12.0

(Example 2) When #10 = 12.3758,

G00X [ROUND [#10] is equivalent to G00X12.376

(minimum input unit: 0.001 mm).

- Numerical values treated in microprograms are floating point type values.
 - M*2^E M: One sign bit + 52-bit binary data
 - E: One sign bit + 10-bit binary data
- With an operation instruction, whether the NC operation stops in the single block mode or not when the single block input (SKB) is ON is determined by the setting for parameter pm0007 D 1.

pm0007 D1 = 0	Does not stop in the signal block mode
pm0007 D1 = 1	Stops in the single block mode.

4.4.5 Control Instructions

To control the program flow of microprograms, the following two instructions are provided: branch instruction and repetition instruction.

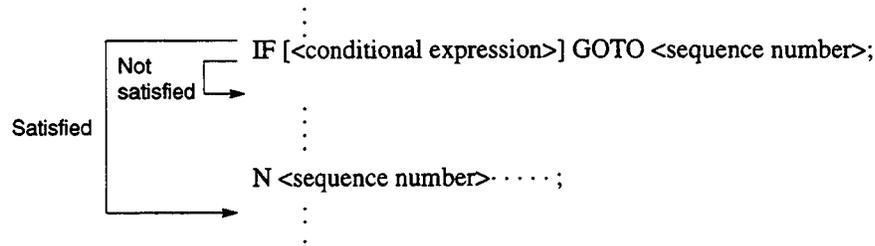
- Branch instruction
- Repetition instruction

(1) Branch Instruction

By specifying “IF [< conditional expression>] GOTO <sequence number>”, the program jumps to the block of the specified sequence number in the same program if <conditional expression> is satisfied.

If <conditional expression>, is not satisfied, the program advances to the next block.

<sequence number> should be placed at the beginning of a block. Even if it is not placed at the beginning of a block, the commands in the block are executed from the beginning. When branch occurs, branch in the reverse direction takes a longer time than branch in the forward direction.



<sequence number: 5-digit positive integer, variable, [<expression>]

Figure 4.4.5.1 Branch Instruction (Conditional Expression Satisfied, Not Satisfied)

It is possible to omit “IF [<conditional expression>]. In this case, the block indicates a simple jump instruction.

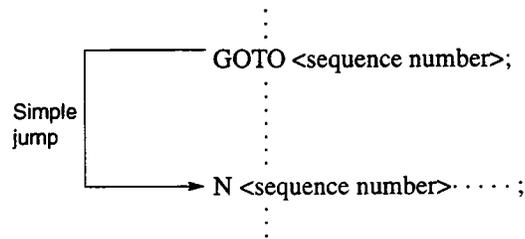


FIGURE 4.4.5.1 Branch Instruction (Simple Jump Instruction)

Instead of “GOTO <sequence number>”, an NC statement or macro program statement can be specified in one block. However, the following macroprogram statements cannot be used due to restrictions.

- Control instructions
- RS-232C data output 2
- Q Status monitoring instruction

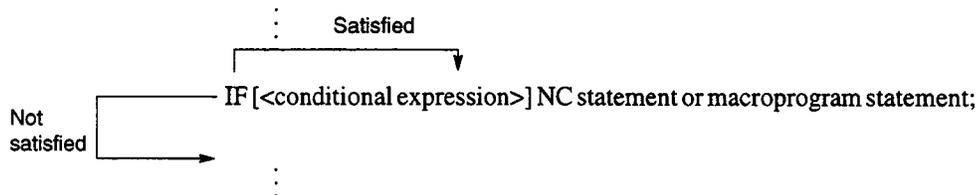


FIGURE 4.4.5.2 Branch Instruction (1-Block Instruction)

The <conditional expression> includes those indicated in Table 4.4.5.1.

Table 4.4.5.1 Types of Conditional Expressions

Conditional Expression	Description
#i EQ #j	#i = #j
#i NE #j	#i ≠ #j
#i GT #j	#i > #j
#i LT #j	#i < #j
#i GE #j	#i ≥ #j
#i LE #j	#i ≤ #j
A OR B	Logical sum of A and B
A AND B	Logical product of A and B
A XOR B	Exclusive logical sum of A and B

Note: Constants and <expression> can be used instead of #i and #j.

(2) Repeat Instructions

```

WHILE [<conditional expression>] DO<number>;
      ⋮
      ⋮
      ⋮
END <number>;

```

<number> = 1, 2, 3

With the commands indicated above, blocks between the block next to the DO block and the END block are repeatedly executed while the <conditional expression> is satisfied.

If the <conditional expression> is not satisfied, the program jumps to the block next to the END block. It is possible to omit “WHILE [<conditional expression>]”. In this case, the block between the DO and END blocks is continually repeated.

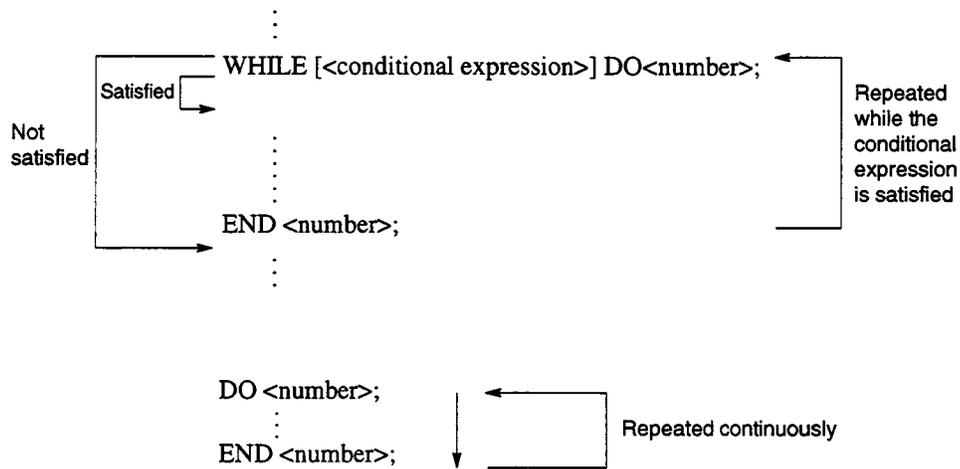


FIGURE 4.4.5.3 Repeat Instruction

- DO must be specified before END.

O1 - END1	END1 - DO1
◦	×

Note: ◦: Correct, ×: Incorrect

- The <number> in “DO number” and “END <number>” must be the same number, and DO and END must be specified as a pair.

DO1 - END1 ... DO2 - END2	DO1 - END2	DO1 - DO1 - END1	DO1 - END1 - END1
○	×	×	×

Note: ○: Correct, ×: Incorrect

- The same <number> can be used as many times as required. However, the range of repetition must not overlap.

DO1 - END1 ... DO1 - END1	DO1 - DO1 - END1 - END1	DO1 - DO1 - END1 - END1	DO1 - DO2 - END1 - END2
○	×	×	×

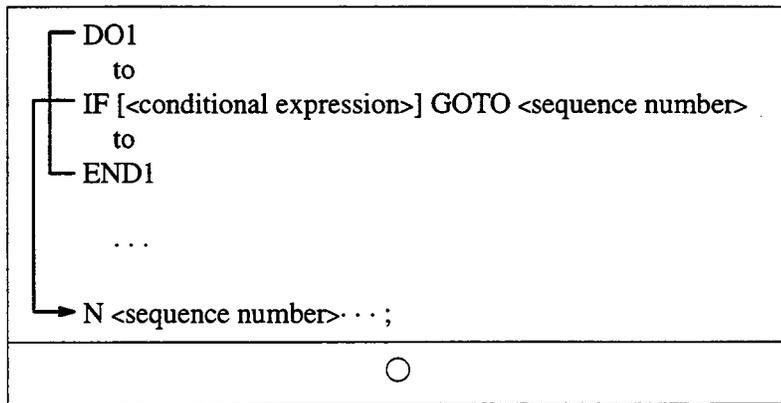
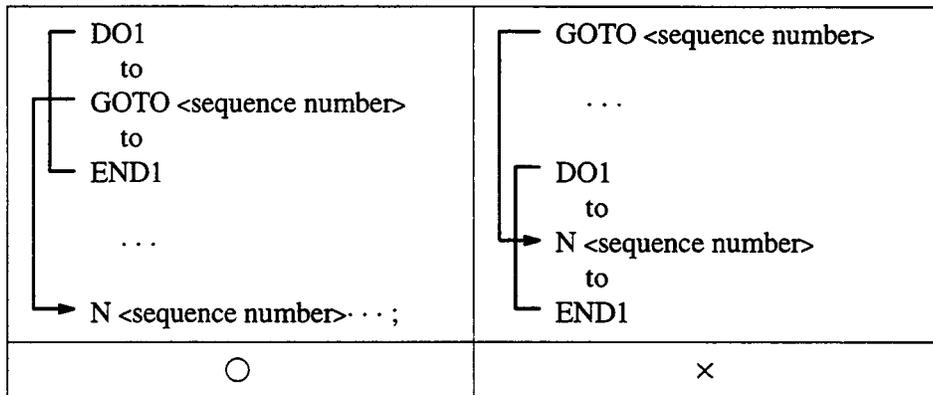
Note: ○: Correct, ×: Incorrect

- Nesting of DO to END loop is allowed for up to three levels in a microprogram or sub-program. From the DO to END loop, it is possible to call up a microprogram or sub-program. In the call up program, nesting of the DO to END loop is also allowed for up to three levels.

DO1 - DO2 - DO3 - END3 - END2 - END1	DO1 - G65P*** - M98P*** - END1	DO1 - DO2 - DO3 - END3 - END2 - END1
○	○	

Note: ○: Correct

- By specifying “GOTO <sequence number>”, it is possible to jump from the DO to END loop to a block outside the loop. However, jump into the DO to END loop by using “GOTO <sequence number>” is not possible.



Note; ○: Correct, ×: Incorrect



For the execution of an operation instruction, operation stops or does not stop in the single block stop mode if the single block input (SKB) is ON according to the setting for parameter pm0007 D1.

pm0007 D1 = 0	Does not stop in the signal-block mode
pm0007 D1 = 1	Stops in the single-block mode.

4.4.6 Registering the Microprogram

Microprograms can be registered and edited in entirely the same manner as registering and editing normal NC programs and subprograms.

For this registration, there are no limits in the size of microprograms; NC programs, subprograms and microprograms can be stored to the limit of the memory capacity.

The program numbers to be used when registering microprograms are classified as indicated in Table 4.4.6.1 according to their applications.

In addition to the classification indicated below, the program numbers to be used specially for microprograms can be set to clearly identify them from NC programs and subprograms. Whether the program number range should be secured for microprograms or not can be set by using a parameter.

Table 4.4.6.1 Classification of Macroprograms

Program No.	Classification by Applications	Protect
O1 to O7999	There are no restrictions on registration, deletion or editing.	
O8000 to O8999	Macroprograms can be protected from edit and display; edit protect and display protect can be set independently by using a parameter.	Protect 1
O9000 to O9999	Macroprograms can be protected from edit and display; edit protect and display protect can be set collectively by using a parameter.	Protect 2
O10000 to O99999	There are no restrictions on registration, deletion or editing	

- To disable all of edit, input/output, and display of O8000 to O8999: pm0020 D0 = 1
- To disable edit and output/input and enable display of O8000 to O8999: pm0020 D0 = 0 and pm0021 D0 = 1
- To disable all of edit, input/output, and display of O9000 to O9999: pm3004 D0 = 1 and pm0022 D0 = 0
- To disable edit and output/input and enable display of O9000 to O9999: pm3004 D0 = 0 and pm0022 D0 = 1

For O9000 to O9999, an option is provided to disable all of edit, input/output, and display always disregarding of the setting for pm3004 and pm0022.

4.4.7 Microprogram Alarm Numbers

Alarm numbers related with microprograms and the cause of them are indicated in Table 4.4.7.1.

Table 4.4.7.1 Macroprogram Alarm Numbers

Alarm No.	Description	Alarm No.	Description
0210	CONSTANT DATA OUT OF RANGE	0221	0 DIVIDE IN MACRO
	In a macroprogram, specified constant is outside the allowable range.		In a microprogram, division by "0" is executed.
0211	UNMATCH G67 COMMAND	0222	ROOT VALUE NEGATIVE
	The number of G67 commands is greater than the number of G65 and G66 commands.		A negative value is specified for square root operation.
0212	MACRO FORMAT ERROR	0223	FLOATING DATA OUT OF RANGE
	There is an error in macroprogram format		Floating point data exceed the allowable range.
0213	UNDEFINED # NO	0224	G66-M99 PROG ERROR
	A value not defined as a variable number is used.		An axis move command is spechified with M99 in the modal call (G66) mode.
0214	ILL LEFT SIDE # NO.	0225	MACRO SYSTEM ERROR
	A variable that cannot be used is set in the left side of operation expression.		Overflow with operation stack.
0215	[] LIMIT OVER	0226	ASIN, ACOS, LN, SQRT ERROR
	Nesting level of brackets [] exceeds the limit.		The result of function operation (ASIN, ACOS, LN, SQRT) is outside the allowable range.
0216	MACRO CALL LIMIT OVER	0227	EXCHANG OVERFLOW
	Macroprogram call up nesting level exceeds the limit.		Overflow during conversion into integer.
0217	DO-END FORMAT ERROR	0228	BCD INPUT DATA OVERRFLOW
	DO and END instructions are not specified in pairs.		Overflow of input data for BCD function.
0218	[] UNMATCH	0229	BIN FORMAT ERROR
	The numbers of left parentheses [and right blacket] do not match.		There is an error in format with the BIN function.
0219	DO-END NO.OUT OF RANCE	0230	EXP OUTPUT DATAOVERFLOW
	In "DO m" command, "m" is not in the range of $1 \leq m \leq 3$.		Overflow with the EXP function.
0220	GOTO NO.FORMAT ERROR		
	'The value of "n" in the "GOTO n" command is outside the allowable range, or the specified "n" is not found.		

4.4.8 Examples of Microprograms

Some examples of microprograms are explained below.

(1) Microprogram in a Canned Cycle

(a) Canned cycle with G82

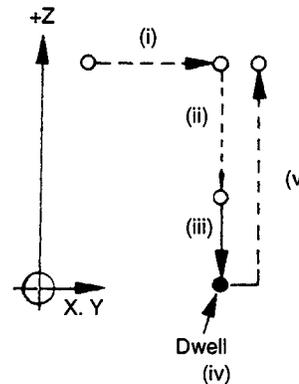
Example of Programming (P1)

```
G91;
G82X100. Y50. R-80. Z-40. P3.0 F250;
```

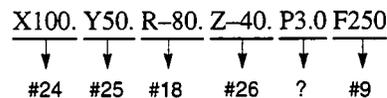
The commands indicated above are executed according to the following processing in the NC.

Example of Programming (P2)

```
G91;
(i) G00W100. Y50.;
(ii) G00Z-80.;
(iii) G01Z-40. F250;
(iv) G04P3.; (dwell)
(v) G00Z- ( (-80) + (-40) );
```

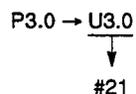


- ① All axis move distances should be replaced with variables (local variables: #1 to #33).
- ② For local variables, type I and type II variables are provided. When the number of local variables to be handled is small, it is recommended to use type I local variables which allow the use of X, Y, and Z, thus facilitating assigning of arguments.
- ③ By using type I local variables, local variables are assigned to address characters as indicated below.



Here, address P cannot be used for assigning argument. Therefore, address P should be replaced with another address.

- ④ Since the dwell is a timer function, it can be replaced with address U.



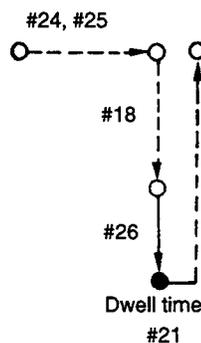
- ⑤ By using these variables, the example program (P2) can be written in the following manner. To carry out operations by using the variables used as the address data, the data must first be rounded off to a number that meets the minimum input unit.

In this example, when returning the Z-axis to the start point, the axis move amount specified in the axis return block must be 120.002 if the contents of #18 and #26 are -80.0006 and -40.0006 , respectively with a minimum setting unit being 0.001 mm. Otherwise the Z-axis cannot be returned to the start point.

If ROUND function is not used, rounding off is executed for the result of operation; that means the Z-axis move distance in the return block is calculated to be: 120.001”.

Example of Programming (P3)

```
G91;
G00X#24, Y#25;
G00Z#18;
G01Z#26 F#9;
G04P#21;
G00Z-[ROUND [#18] + [ROUND [#26] ] ;
M99 ;
```



(b) Calling up a microprogram by using G65

Example of Microprogram Call Up Program (P4)

```
G91;
G65 P9082 X100. Y50. R-80. Z-40.
      U3.0 F250.;
```

- ① Use the example program (P4) as the macroprogram and assign program number “O9082” to this macroprogram.

```
O9082;
G00X#24 Y#25;
G00Z#18;
G01Z#26 F#9;
G04P#21;
G00Z- [ROUND [#18] + ROUND [#26] ];
M99;
```

- ② With the microprogram indicated above, it is necessary to specify the R-point and Z-point levels for each execution of the macroprogram. Therefore, another microprogram should be written which specifies the position of the R-point and Z-point levels.

```
O9000;
#100 = #18;
#101 = #26;
#102 = #21;
#103 = #9;
M99;
O9082;
G00X#24 Y#25;
G00Z#100;
G01Z#101 F#9;
G04P#102;
G01Z- [ROUND [#100] + ROUND [#101] ];
M99;
```

- ③ The program used to call the macroprograms indicated in item ② above is indicated below.

```
G91;
G65 P9000 R-80. Z-40. U3.0 F250.;
G65 P9082 X100. Y50.;
G65 P9082 X ••• Y •••;
•••••
```

(c) Example of control instructions using modal information

With the programs indicated above, the Z-axis returns to the initial point level. When giving the commands for initial point level return (G98) and R-point level return (G99), it is possible to change the command that specifies the tool return level according to the present mode (G98 or G99); the present mode is found by checking the modal information. For this check, modal information variable can be used.

- System variable #4010 should be used since G98 and G99 are G code of 10-group.
- To change the tool return level command, use “IF ••• GOTO ••• command.
- It is possible to retain the G code of 0-group, which is valid before the execution of a macroprogram, after the execution of the macroprogram. “G codes of 01-group” means G00, G01, G02, G03.

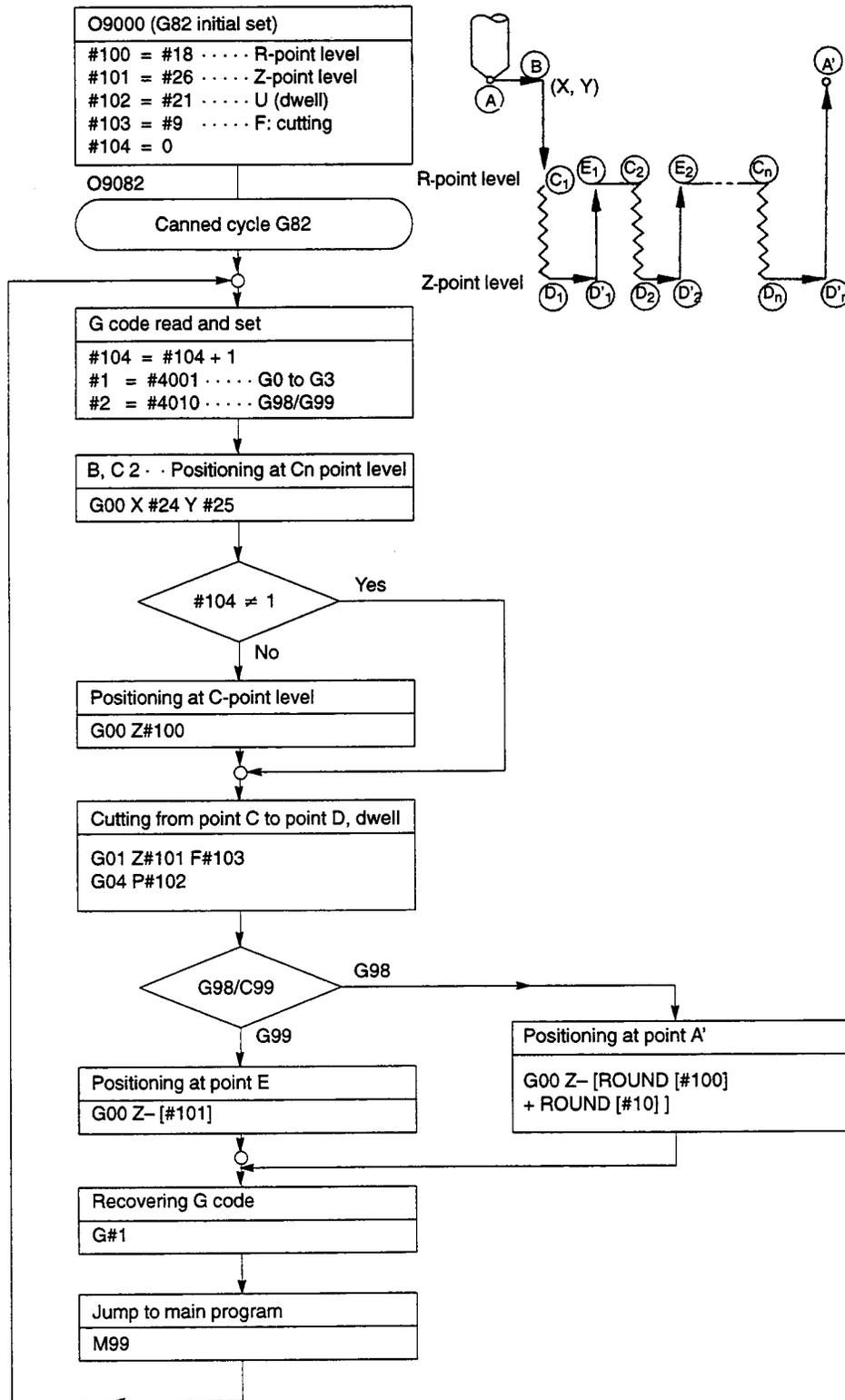
① Example of macroprogram call up program

```
G91 G99;
G65 P9000 R-80. Z-40. U3.0 F250;
G65 P9082 X-100 Y50.;
G65P9082X ••••• Y •••••;
•••••
G98;
G65P9082X ••••• Y •••••;
```

② Example of macroprogram

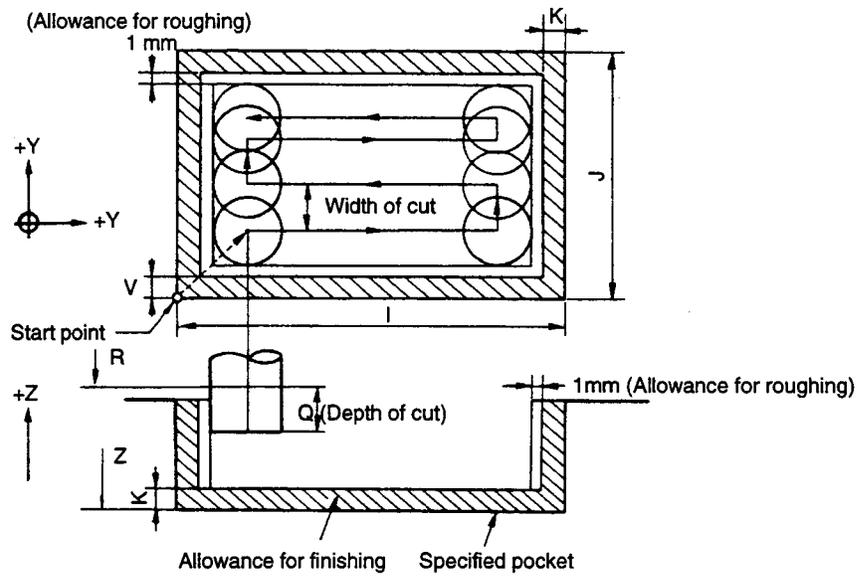
```
O9000
#100 = #18;
#101 = #26;
#102 = #21;
#103 = #9;
#104 = 0;
-----
O9082;
#104 = #104 + 1;
#1 = #4001;..... G0 to G3
#2 = #4010;..... G98/G99
G00 X#24 Y#25;
IF [#104 NE1] GOTO1;
G00 Z#100;
N1 G01 Z#101 F#103;
G04 P#102;
IF [#2 EQ98] GOTO2;
G00 Z-[#101];
GOTO3;
N2 G00 Z-[ROUND [#100] + ROUND [#101]];
N3 G#1;          ..... Recovering the G code
M99;. .
```

③ Execution order of example program



(2) Example of Pocket Milling

The following gives an explanation on a macroprogram used for executing pocket milling as shown below.



- ① Macroprogram call up is specified as indicated below.

G65 P9061 X...Y...Z...R...I...J...K...T...Q...D...F...E
...; where,

X, Y: Absolute coordinate value of start point (lower left end of pocket)

Z: Absolute position of the bottom of pocket

R: Absolute position (level) for tool return at rapid traverse

I, J: Length, of pocket in the X- and Y-axis direction (unsigned)

K: Allowance for finishing (allowance to be left, unsigned) "0" if not specified.

T: Width of cut ratio (in % of tool diameter)

Width of cut = Tool diameter \times T/100

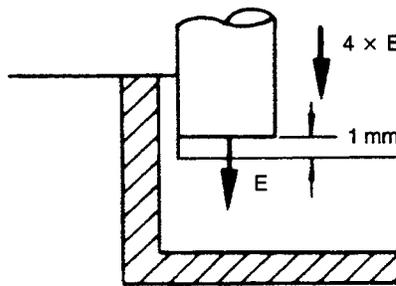
Q: Depth of cut per cycle (unsigned)

D: Tool of offset number

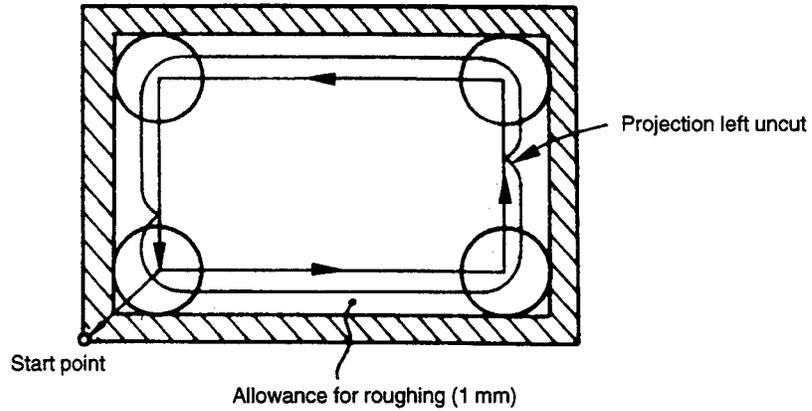
F: Feedrate in the XY plane

E: Feedrate for Z-axis cut-in

(Up to a level 1 mm above the previously cut level, "E \times 4")



- ② In the final process, allowance for roughing (1 mm width), left on the allowance for finishing, is removed in one cycle inside the pocket. After that the axes return to the start point to end the microprogram.

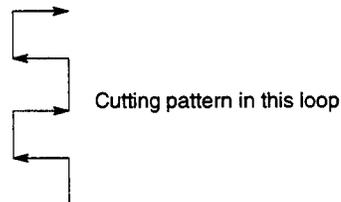


- ③ The macroprogram is indicated below. Refer to the supplemental explanation given for each block.

```

O9061;
#10 = # [2000+#7] ————— Radius of cutting tool
#11 = #6+1.0+#10;
#12 = #5-2*#10;
#13 = 2*#10*#20/100 ————— Width of cut
#14 = FUP [#12/#13] ————— Number of cuts (X-axis direction) - 1
#27 = #24+#11; ————— Start point of machining
#28 = #25+#11; ————— X and Y coordinate values
#29 = #26+#6; ————— Z coordinate value of bottom level
#30 = #24+#4-#11;
#15 = #4003 ————— Reading of G90/G91
G90; ————— Designation of absolute mode
G00 X#27 Y#28;
G00 Z#18;
#32 = #18; ————— #32: Bottom of present cutting
D01;
#32 = #32-#17;
IF [#32 GT#29] GOTO1;
#32 = #29;
N1 G01 Z#32 F#8;
G01 X#30 F#9;
#33 = 1;
WHILE [#33 LE #14] DO2;
IF [#33 EQ #14] GOTO2;
G01Y [#28+#33*#13] F#9; _
GOTO3;
N2 G01 Y [#25+#5-#11] ;
N3 IF [#33 AND 1 EQ 0] GOTO4;
G01 X#27;
GOTO5;
N4 G01 X#30;
N5 #33 = #33+1;
END2;

```



```

G00 Z#18;
IF [#32 LE#29] GOTO6;
G00 X#27 Y#28;
G01 Z [#32+1.0] F [4*#8] ;
END1:
N6 #11 = #11-1.0;
#27 = #27-1.0;
#28 = #28-1.0;
#30 = #30-1.0;
#31 = #25+#5-#11
G00 X#27 Y#28;
G01 Z#32 F#8;
G01 X#30 Y#9;
Y#31;
X#27;
Y#28;
G00 Z#18;
G00 X#24 Y#25; ————— Return to the start point.
G#15; ————— Recovering G90/G91
M99;

```

Rough finishing cycle

APPENDIX 1

G CODE TABLE

Appendix 1 describes the G code and the functions.

1.1 G CODE TABLE

APPENDIX 1.1 G CODE TABLE

Appendix Table 1.1G Code Table

G Code	Group	Function	Refer to
G00 #1	01	Positioning (rapid traverse):	2.1.1
G01 #1		Linear interpolation	2.1.2
G02		Circular interpolation CW, helical interpolation CW	2.1.3, 2.1.4
G03		Circular interpolation CCW, helical interpolation CCW	2.1.3, 2.1.4
G04	*	Dwell	
G06		positioning (error detect OFF)	
G09		Exact stop	
G10		Programmable data input	
G12		Circle cutting CW	
G13		Circle cutting CCW	
G17 #2		02	Plane selection (XY)
G18	Plane selection (ZX)		
G19	Plane selection (YZ)		
G20	06	Input unit system designation (inch)	
G21		Input unit system designation (mm)	
G22	04	Stored stroke limit ON	
G23		Stored stroke limit OFF	
G25	*	Program copy	
G27		Reference point return check	
G28		Automatic reference point return	
G29		Rectum from reference point	
G30		Second, third, and fourth reference point return	
G31		Skip function	
G40 #1	07	Tool radius offset, cancel	
		Tool-dimensional tool radius offset, cancel	
G41		Tool radius offset, left	
G42		Three-dimensional tool radius offset, left	
		Tool radius offset, right	
G43 #2	08	Tool length offset, +	
G44 #2		Tool length offset, -	
G49 #2		Tool length offset, cancel 3	

Appendix Table 1.1 G Code Table (cont'd)

G Code	Group	Function	Refer to
G45	01	Tool position offset, extension	2.1.1
G46		Tool position offset, reduction I	2.1.2
G47		Tool position offset, double-extension	2.1.3, 2.1.4
G48		Tool position offset, double-reduction	2.1.3, 2.1.4
G50 #1	08	Scaling OFF	
G51		Scaling ON	
G52 #2	12	Local coordinate system	
G53	*	Machine coordinate system	
G54	12	Shift to workpiece coordinate system 1	
G55		Shift to workpiece coordinate system 2	
G56		Shift to workpiece coordinate system 3	
G57		Shift to workpiece coordinate system 4	
G58		Shift to workpiece coordinate system 5	
G59		Shift to workpiece coordinate system 6	
G60	01	Unidirectional positioning	
G61	13	Exact stop mode	
G64 #2		Exact stop mode, cancel	
G65	*	Microprogram simple call	
G66	14	Microprogram modal call	
G67 #1		Microprogram modal call, cancel	
G68	18	Coordinate rotation ON	
G69 #1		Coordinate rotation OFF	
G70	*	Bolt-hole circle	
G71		Arc	
G72		Line-at-angle	
G73	09	Canned cycle (high-speed deep hole drilling)	
G74		Canned cycle (reverse tapping)	
G76		Canned cycle (boring)	
G77		Canned cycle (back boring)	
G80 #1		Canned cycle, cancel	
G81		Canned cycle (drilling), external operation function	
G82		Canned cycle (spot facing)	
G83		Canned cycle (deep hole drilling)	
G84		Canned cycle (tapping)	
G85		Canned cycle (boring)	
G86		Canned cycle (boring)	
G87		Canned cycle (boring)	
G88		Canned cycle (boring)	
G89		Canned cycle (boring)	

Appendix Table 1.1 G Code Table (cont'd)

G Code	Group	Function	Refer to
G90 #2	03	Absolute programming	
G91		Incremental programming	
G92 #2	*	Coordinate system setting	
G93	05	Solid tap mode	
G94 #1		Feed per minute mode, solid tap mode cancel	
G98 #2	10	Canned cycle, initial level return	
G99		Canned cycle, R-point level return	
G106	*	Automatic corner override	
G107		Corner cutting feedrate designation	
G108		Corner cutting feedrate designation cancel	
G174	09	Deep hole solid tap	
G181		Canned cycle (2-step drilling) 4.1.1	
G182		Canned cycle (2-step spot facing)	
G184		Deep hole solid tap cancel	
G185		Canned cycle (2-step boring)	
G186		Canned cycle (2-step boring)	
G187		Canned cycle (2-step boring)	
G189		Canned cycle (2-step boring)	

- Note1: The NC establishes the G code modes, identified by #1, when the power turned ON or when the NC is reset.
 2. The NC establishes the G code modes, identified by #2, when the power is turned ON.
 3: If both #1 and #2 are indicated in the same G code group, the G codes which should be valid when the power is turned ON can be selected by the setting for the appropriate parameters.

APPENDIX 2

INDEX

In Appendix 2, technical terms specific to NC and J300M are arranged in alphabetical order.

Please use this index when looking for descriptions using the technical term as the key code.



YASKAWA ELECTRIC AMERICA, INC.

Chicago-Corporate Headquarters 2121 Norman Drive South, Waukegan, IL 60085, U.S.A.
Phone: (847) 887-7000 Fax: (847) 887-7310 Internet: <http://www.yaskawa.com>

MOTOMAN INC.

805 Liberty Lane, West Carrollton, OH 45449, U.S.A.
Phone: (937) 847-6200 Fax: (937) 847-6277

YASKAWA ELECTRIC CORPORATION

New Pier Takeshiba South Tower, 1-16-1, Kaigan, Minatoku, Tokyo, 105-0022, Japan
Phone: 81-3-5402-4511 Fax: 81-3-5402-4580 Internet: <http://www.yaskawa.co.jp>

YASKAWA ELETRICO DO BRASIL COMERCIO LTDA.

Avenida Fagundes Filho, 620 Bairro Saude Sao Paulo-SP, Brasil CEP: 04304-000
Phone: 55-11-5071-2552 Fax: 55-11-5581-8795 E-mail: yaskawabrasil@originet.com.br

YASKAWA ELECTRIC EUROPE GmbH

Am Kronberger Hang 2, 65824 Schwalbach, Germany
Phone: 49-6196-569-300 Fax: 49-6196-888-301 Internet: <http://www.yaskawa.de>

MOTOMAN ROBOTICS AB

Box 504 S38525, Torsas, Sweden
Phone: 46-486-48800 Fax: 46-486-41410

MOTOMAN ROBOTEC GmbH

Kammerfeldstraße 1, 85391 Allershausen, Germany
Phone: 49-8166-900 Fax: 49-8166-9039

YASKAWA ELECTRIC UK LTD.

1 Hunt Hill Orchardton Woods Cumbernauld, G68 9LF, Scotland, United Kingdom
Phone: 44-12-3673-5000 Fax: 44-12-3645-8182

YASKAWA ELECTRIC KOREA CORPORATION

Paik Nam Bldg. 901 188-3, 1-Ga Euljiro, Joong-Gu, Seoul, Korea
Phone: 82-2-776-7844 Fax: 82-2-753-2639

YASKAWA ELECTRIC (SINGAPORE) PTE. LTD.

Head Office: 151 Lorong Chuan, #04-01, New Tech Park Singapore 556741, SINGAPORE
Phone: 65-282-3003 Fax: 65-289-3003

TAIPEI OFFICE (AND YATEC ENGINEERING CORPORATION)

10F 146 Sung Chiang Road, Taipei, Taiwan
Phone: 886-2-2563-0010 Fax: 886-2-2567-4677

YASKAWA JASON (HK) COMPANY LIMITED

Rm. 2909-10, Hong Kong Plaza, 186-191 Connaught Road West, Hong Kong
Phone: 852-2803-2385 Fax: 852-2547-5773

BEIJING OFFICE

Room No. 301 Office Building of Beijing International Club,
21 Jianguomanwai Avenue, Beijing 100020, China
Phone: 86-10-6532-1850 Fax: 86-10-6532-1851

SHANGHAI OFFICE

27 Hui He Road Shanghai 200437 China
Phone: 86-21-6553-6600 Fax: 86-21-6531-4242

SHANGHAI YASKAWA-TONJI M & E CO., LTD.

27 Hui He Road Shanghai 200437 China
Phone: 86-21-6533-2828 Fax: 86-21-6553-6677

BEIJING YASKAWA BEIKE AUTOMATION ENGINEERING CO., LTD.

30 Xue Yuan Road, Haidian, Beijing 100083 China
Phone: 86-10-6232-9943 Fax: 86-10-6234-5002

SHOUGANG MOTOMAN ROBOT CO., LTD.

7, Yongchang-North Street, Beijing Economic & Technological Development Area,
Beijing 100076 China

Phone: 86-10-6788-0551 Fax: 86-10-6788-2878