

CNC SYSTEM

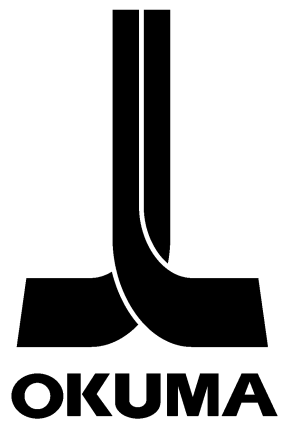
OSP-P200M/P200MA/P20M

OSP-P200M-R/P200MA-R/P20M-R

PROGRAMMING MANUAL

(10th Edition)

Pub No. 5228-E-R9 (ME33-018-R10) Oct. 2010



SAFETY PRECAUTIONS

The machine is equipped with safety devices which serve to protect personnel and the machine itself from hazards arising from unforeseen accidents. However, operators must not rely exclusively on these safety devices: they must also become fully familiar with the safety guidelines presented below to ensure accident-free operation.

This instruction manual and the warning signs attached to the machine cover only those hazards which Okuma can predict. Be aware that they do not cover all possible hazards.

1. Precautions Relating to Installation

(1) Please be noted about a primary power supply as follows.

- Do not draw the primary power supply from a distribution panel that also supplies a major noise source (for example, an electric welder or electric discharge machine) since this could cause malfunction of the CNC unit.
- If possible, connect the machine to a ground not used by any other equipment. If there is no choice but to use a common ground, the other equipment must not generate a large amount of noise (such as an electric welder or electric discharge machine).

(2) Installation Environment

Observe the following points when installing the control enclosure.

- Make sure that the CNC unit will not be subject to direct sunlight.
- Make sure that the control enclosure will not be splashed with chips, water, or oil.
- Make sure that the control enclosure and operation panel are not subject to excessive vibrations or shock.
- The permissible ambient temperature range for the control enclosure is 5 to 40°C.
- The permissible ambient humidity range for the control enclosure is relative humidity 50% or less at 40°C (no condensation).
- The maximum altitude at which the control enclosure can be used is 1000 m (3281ft.).

2. Points to Check before Turning on the Power

- (1) Close all the doors of the control enclosure and operation panel to prevent the entry of water, chips, and dust.
- (2) Make absolutely sure that there is nobody near the moving parts of the machine, and that there are no obstacles around the machine, before starting machine operation.
- (3) When turning on the power, turn on the main power disconnect switch first, then the CONTROL ON switch on the operation panel.

3. Precautions Relating to Operation

- (1) After turning on the power, carry out inspection and adjustment in accordance with the daily inspection procedure described in this instruction manual.
- (2) Use tools whose dimensions and type are appropriate for the work undertaken and the machine specifications. Do not use badly worn tools since they can cause accidents.
- (3) Do not, for any reason, touch the spindle or tool while spindle indexing is in progress since the spindle could rotate: this is dangerous.
- (4) Check that the workpiece and tool are properly secured.
- (5) Never touch a workpiece or tool while it is rotating: this is extremely dangerous.
- (6) Do not remove chips by hand while machining is in progress since this is dangerous. Always stop the machine first, then remove the chips with a brush or broom.
- (7) Do not operate the machine with any of the safety devices removed. Do not operate the machine with any of the covers removed unless it is necessary to do so.
- (8) Always stop the machine before mounting or removing a tool.
- (9) Do not approach or touch any moving part of the machine while it is operating.
- (10) Do not touch any switch or button with wet hands. This is extremely dangerous.
- (11) Before using any switch or button on the operation panel, check that it is the one intended.

4. Precautions Relating to the ATC

- (1) The tool clamps of the magazine, spindle, etc., are designed for reliability, but it is possible that a tool could be released and fall in the event of an unforeseen accident, exposing you to danger: do not touch or approach the ATC mechanism during ATC operation.
- (2) Always inspect and change tools in the magazine in the manual magazine interrupt mode.
- (3) Remove chips adhering to the magazine at appropriate intervals since they can cause misoperation. Do not use compressed air to remove these chips since it will only push the chips further in.
- (4) If the ATC stops during operation for some reason and it has to be inspected without turning the power off, do not touch the ATC since it may start moving suddenly.

5. On Finishing Work

- (1) On finishing work, clean the vicinity of the machine.
- (2) Return the ATC, APC and other equipment to the predetermined retraction position.
- (3) Always turn off the power to the machine before leaving it.
- (4) To turn off the power, turn off the CONTROL ON switch on the operation panel first, then the main power disconnect switch.

6. Precautions during Maintenance Inspection and When Trouble Occurs

In order to prevent unforeseen accidents, damage to the machine, etc., it is essential to observe the following points when performing maintenance inspections or during checking when trouble has occurred.

- (1) When trouble occurs, press the emergency stop button on the operation panel to stop the machine.
- (2) Consult the person responsible for maintenance to determine what corrective measures need to be taken.
- (3) If two or more persons must work together, establish signals so that they can communicate to confirm safety before proceeding to each new step.
- (4) Use only the specified replacement parts and fuses.
- (5) Always turn the power off before starting inspection or changing parts.
- (6) When parts are removed during inspection or repair work, always replace them as they were and secure them properly with their screws, etc.
- (7) When carrying out inspections in which measuring instruments are used - for example voltage checks - make sure the instrument is properly calibrated.
- (8) Do not keep combustible materials or metals inside the control enclosure or terminal box.
- (9) Check that cables and wires are free of damage: damaged cables and wires will cause current leakage and electric shocks.
- (10) Maintenance inside the Control Enclosure
 - a. Switch the main power disconnect switch OFF before opening the control enclosure door.
 - b. Even when the main power disconnect switch is OFF, there may be some residual charge in the MCS drive unit (servo/spindle), and for this reason only service personnel are permitted to perform any work on this unit. Even then, they must observe the following precautions.
 - MCS drive unit (servo/spindle)
The residual voltage discharges two minutes after the main switch is turned OFF.

- c. The control enclosure contains the NC unit, and the NC unit has a printed circuit board whose memory stores the machining programs, parameters, etc. In order to ensure that the contents of this memory will be retained even when the power is switched off, the memory is supplied with power by a battery. Depending on how the printed circuit boards are handled, the contents of the memory may be destroyed and for this reason only service personnel should handle these boards.

(11) Periodic Inspection of the Control Enclosure

- a. **Cleaning the cooling unit**
The cooling unit in the door of the control enclosure serves to prevent excessive temperature rise inside the control enclosure and increase the reliability of the NC unit. Inspect the following points every three months.
 - Is the fan motor inside the cooling unit working?
The motor is normal if there is a strong draft from the unit.
 - Is the external air inlet blocked?
If it is blocked, clean it with compressed air.

7. General Precautions

- (1) Keep the vicinity of the machine clean and tidy.
- (2) Wear appropriate clothing while working, and follow the instructions of someone with sufficient training.
- (3) Make sure that your clothes and hair cannot become entangled in the machine. Machine operators must wear safety equipment such as safety shoes and goggles.
- (4) Machine operators must read the instruction manual carefully and make sure of the correct procedure before operating the machine.
- (5) Memorize the position of the emergency stop button so that you can press it immediately at any time and from any position.
- (6) Do not access the inside of the control panel, transformer, motor, etc., since they contain high-voltage terminals and other components which are extremely dangerous.
- (7) If two or more persons must work together, establish signals so that they can communicate to confirm safety before proceeding to each new step.

8. Symbols Used in This Manual

The following warning indications are used in this manual to draw attention to information of particular importance. Read the instructions marked with these symbols carefully and follow them.



indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.



indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.



indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury.



indicates a potentially hazardous situation which, if not avoided, may result in damage to your property.



indicates general instructions for safe operation.

INTRODUCTION

Thank you very much for choosing our NC system. This NC system is an expandable CNC with various features. Major features of the NC system are described below.

(1) Compact and highly reliable

The CNC system has become compact and highly reliable because of advanced hardware technology, including the computer boards equipped with high-speed microprocessors, I/O link, and servo link. The "variable software" as a technical philosophy of the OSPs supported by a flash memory. Functions may be added to the CNC system as required after delivery.

(2) NC operation panels

The following types of NC operation panels are offered to improve the user-friendliness.

- Thin color operation panels (horizontal)
 - Thin color operation panels (vertical)
- One or more of the above types may not be used for some models.

(3) Machining management functions

These functions contribute to the efficient operation of the CNC system and improve the profitability from small quantity production of multiple items and variable quantity production of variations. Major control functions are described below.

a. Reduction of setup time

With increase in small-volume production, machining data setting is more frequently needed. The simplified file operation facilitates such troublesome operation. The documents necessary for setup, such as work instructions, are displayed on the CNC system to eliminate the necessity of controlling drawings and further reduce the setup time.

b. Production Status Monitor

The progress and operation status can be checked on a real-time basis on the screen of the CNC system.

c. Reduction of troubleshooting time

Correct information is quickly available for troubleshooting.

(4) Help functions

When an alarm is raised, press the help key to view the content of the alarm.

This helps take quick action against the alarm.

To operate the CNC system to its maximum performance, thoroughly read and understand this instruction manual before use.

Keep this instruction manual at hand so that it will be available when you need a help.

Screens

Different screens are used for different models. Therefore, the screens used on your CNC system may differ from those shown in this manual.

TABLE OF CONTENTS

SECTION 1 PROGRAM CONFIGURATIONS	1
1. Program Types and Extensions.....	1
2. Program Name	2
3. Sequence Name	3
4. Program Format.....	4
4-1. Word Configuration.....	4
4-2. Block Configuration	4
4-3. Program.....	5
4-4. Programmable Range of Address Characters.....	5
5. Mathematical Operation Functions	6
6. Optional Block Skip.....	8
7. Program Branch Function (Optional).....	9
8. Comment Function (Control OUT/IN)	9
9. Message Function (Optional).....	10
10. Operation Methods and Program Storage Memory Capacity.....	10
SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS	13
1. Coordinate System	13
1-1. Coordinate Systems and Values	13
1-2. Machine Zero and Machine Coordinate System	13
1-3. Work Coordinate System.....	14
1-4. Local Coordinate System	14
2. Coordinate Commands.....	15
2-1. Numerically Controlled Axes	15
2-2. Unit Systems	16
2-3. Travel Limit Commands (G22, G23) (Optional).....	21
2-4. Home Position Command (G30)	23
2-5. Absolute and Incremental Commands (G90, G91)	24
2-6. Coordinate Recalculation Command (G97).....	25
SECTION 3 FEED FUNCTIONS.....	26
1. Rapid Feed	26
2. Cutting Feed	26
2-1. Feed per Minute (G94)	26
2-2. Feed per Revolution (G95).....	26
2-3. F1-digit Feed Function (Optional).....	27
2-4. F0 Command During Cutting Feed.....	28
3. Exact Stop Check Function (G09, G61, G64).....	29
4. Automatic Acceleration and Deceleration.....	30

5. Following Error Check	31
6. Positioning (G00)	32
7. Uni-directional Positioning (G60)	33
8. Linear Interpolation (G01)	34
9. Plane Selection (G17, G18, G19)	35
10. Circular Interpolation (G02, G03)	37
11. Helical Cutting (G02, G03) (Optional)	40
SECTION 4 PREPARATORY FUNCTIONS	41
1. Dwell Command (G04)	41
2. Programmable Mirror Image (G62) (Optional)	42
3. Work Coordinate System Selection (G15, G16)	44
4. Work Coordinate System Change (G92)	45
5. Unit System Check (G20, G21) (Optional)	45
6. Coordinate System Conversion Functions	46
6-1. Parallel Shift and Rotation of Coordinate Systems (G11, G10)	46
6-2. Copy Function (COPY, COPYE)	48
7. Workpiece Geometry Enlargement / Reduction Function (G51, G50) (Optional)	50
SECTION 5 S, T, AND M FUNCTIONS	52
1. S Code Function	52
2. T Code Function	52
3. M Code Function	53
3-1. Examples of M Codes	53
SECTION 6 OFFSET FUNCTIONS	56
1. Tool Length Offset Function (G53 - G59)	56
2. Cutter Radius Compensation (G40, G41, G42)	57
2-1. Cutter Radius Compensation Function	57
2-2. Tool Movement in Start-up	59
2-3. Tool Movement in Cutter Radius Compensation Mode	62
2-4. Tool Movement when Cutter Radius Compensation is Canceled	67
2-5. Changing Compensation Direction in Cutter Radius Compensation Mode	71
2-6. Cutter Radius Compensation Type A	74
2-7. Notes on Cutter Radius Compensation	81
3. Cutter Radius Compensation Mode Override Function	90
3-1. Automatic Override at Corners	90
3-2. Circular Arc Inside Cutting Override	92
4. Tool Radius Compensation G39 Command	93
4-1. Parameter	93
4-2. Corner Circular Interpolation	94
4-3. Corner Circular Interpolation Command Automatic Insertion	96

5. Three-dimensional Tool Offset (G43, G44) (Optional).....	98
5-1. Three-dimensional Tool Offset Start-up	98
5-2. Three-dimensional Tool Offset Vector	99
5-3. Canceling Three-dimensional Tool Offset	101
5-4. Actual Position Data Display And Feedrate.....	101
5-5. Relationship with Other G Functions	102
5-6. Relationship to Other Tool Offset Functions.....	102
 SECTION 7 FIXED CYCLES	 103
1. Table of Fixed Cycle Functions	104
2. Fixed Cycle Operations	105
2-1. Determining the Positioning Plane and the Cycle Axis.....	106
2-2. Controlling the Return Level	107
2-3. Fixed Cycle Mode	107
2-4. Cycle Operation Conditions	108
3. General Rules for Programming Fixed Cycles	109
3-1. Programming Format (General Command Format)	109
3-2. Command Items Necessary for Fixed Cycle Function Commands	111
3-3. Absolute Programming Mode and Incremental Programming Mode.....	112
3-4. Positional Relationship among Return Point Level, Point R Level and Point Z Level	113
3-5. Axis Shift.....	113
3-6. Z-axis G01 Mode Return Function	114
3-7. Relationships between Fixed Cycle Functions and Other Functions	115
3-8. Notes for Programming a Fixed Cycle.....	116
4. Specification of Return-point Level (G71).....	117
5. High Speed Deep Hole Drilling Cycle (G73).....	118
6. Reverse Tapping Cycle (G74)	119
7. Fine Boring (G76)	120
8. Fixed Cycle Cancel (G80).....	121
9. Drilling Cycle (G81, G82).....	122
10. Deep Hole Drilling Cycle (G83).....	123
11. Tapping Cycle (G84).....	125
12. Boring Cycle (G85, G89)	126
13. Boring Cycle (G86)	127
14. Back Boring Cycle (G87)	128
 SECTION 8 COORDINATE CALCULATION FUNCTION (PATTERN FUNC- TION)	 129
1. Table of Functions	129
2. General Rules of Coordinate Calculation	130
2-1. Programming Format for Coordinate Calculation	130
2-2. Plane on Which Coordinate Calculation is Performed, and Motion Axes	132

2-3. Positioning at Calculated Pattern Points	132
2-4. Others	132
3. Omit (OMIT).....	133
4. Restart (RSTRT).....	134
5. Line at Angle (LAA).....	135
6. Grid (GRDX, GRDY).....	136
7. Double Grid (DGRDX, DGRDY)	137
8. Square (SQRX, SQRY)	139
9. Bolt Hole Circle (BHC).....	141
10.Arc (ARC)	142
SECTION 9 AREA MACHINING FUNCTIONS.....	143
1. List of Area Machining Functions.....	143
2. Area Machining Operations	143
2-1. Basic Operations	143
2-2. Tool Movements	144
3. Area Machining Plane and Cycle Axis.....	146
4. General Rules.....	147
4-1. Programming Format (General Command Format)	147
4-2. Area Machining Functions and Commands to be Used	148
4-3. Data Entry in Incremental/Absolute Mode	149
4-4. Relationship among Present Point, Point R Level, and Finish Surface Level	149
4-5. Definition of Machining Area (I, J)	150
4-6. Notes on Area Machining	150
5. Face Milling Functions (FMILR, FMILF)	151
6. Pocket Milling (PMIL, PMILR).....	156
6-1. Zigzag Pattern Pocket Milling Function (PMIL)	156
6-2. Spiral Pattern Pocket Milling Function (PMILR)	160
7. Round Milling Functions (RMILO, RMILI).....	164
SECTION 10SUBPROGRAM FUNCTIONS.....	170
1. Overview.....	170
1-1. Calling a Subprogram.....	170
2. Simple Call (CALL)	173
3. Subprogram Call after Axis Movement (MODIN, MODOUT).....	175
4. G and M Code Macro Functions.....	180
5. Program Call Function Using Variables.....	183
5-1. Outline	183
5-2. Program Call function by Variables	183
5-3. Program Registration Function	185
SECTION 11USER TASK.....	186

1. User Task 1	186
1-1. Branch Function	186
1-2. Variable Function.....	189
1-3. Math Functions	195
1-4. System Variables.....	196
2. User Task 2	234
2-1. I/O Variables	234
2-2. Math Functions	240
SECTION 12 SCHEDULE PROGRAMS	243
1. Overview	243
2. PSELECT Block.....	244
3. Branch Block.....	247
4. Variables Setting Block.....	248
5. Schedule Program End Block.....	248
SECTION 13 OTHER FUNCTIONS	249
1. Table Index Specification.....	249
1-1. 5-Degree Index Commands	249
1-2. 1-Degree Index Commands	250
1-3. 0.001 Degree Commands (Optional).....	252
2. Angular Commands	254
3. Manual Shift Amount Cancel Command.....	255
4. Print Format Function	258
SECTION 14 FILE MANAGEMENT	259
1. Files	259
2. Various Files	260
SECTION 15 APPENDIX	261
1. G Code Table (Including Optional Functions).....	261
2. Table of Mnemonic Codes (Including Optional Functions).....	265
3. M Code Table	268
4. Table of Reserved Local Variable Words	277
5. Table of System Variables	278

SECTION 1 PROGRAM CONFIGURATIONS

1. Program Types and Extensions

For OSP-E100M/E10M, four kinds of programs are used: schedule programs, main programs, subprograms, and library programs. The following briefly explains these four kinds of programs.

Schedule Program

When more than one type of workpiece is machined using a pallet changer or other loading and unloading equipment, multiple main programs are used. A schedule program is used to specify the order in which the main programs are executed and the number of times the individual main program is executed. Using a schedule program makes it possible to carry out untended operation easily.

It is not necessary to assign a program name. The END code must be specified at the end of a schedule program. For details, refer to SECTION 12, "SCHEDULE PROGRAMS".

Main Program

A main program contains a series of commands to machine one type of workpiece. Subprograms can be called from a main program to simplify programming.

A main program begins with a program name which begins with address character "O" and ends with M02 or M30.

Subprogram

A subprogram can be called from a main program or another subprogram. There are two types of subprograms: those written and supplied by Okuma (maker subprogram), and those written by the customer (user subprogram).

The program name, which must start with "O", is required at the beginning of the subprogram. The RTS command must be specified at the end of the subprogram. For details, refer to SECTION 10, "SUBPROGRAM FUNCTIONS".

Library Program

Subprograms and G code macros which are used frequently may be stored as library programs. Since library programs are automatically stored in the operation buffer area when the power is turned on, they can be accessed at any time.

When a library program is stored in the operation buffer area, both a file name and an extension are stored. The file name format is shown below.

- Program file format

Main file name: Begins with alphabetic characters (max. 16 characters)



ME33018R1000300010001

- Extensions

SDF: Schedule program file

MIN: Main program file

MSB: Maker subprogram file

SSB: System subprogram file

SUB: User subprogram file

LIB: Library program file

2. Program Name

All programs are assigned a program name or a program number, and a desired program can be called and executed by simply specifying the program name or number.

A program name that contains only alphabetic characters is called a program label and the one that contains only numbers is called a program number. In this manual, both of them are referred to as a program name.

Program Name Designation

- Enter letters of the alphabet (A to Z) or numbers (0 to 9) following address character “O”. Note that no space is allowed between “O” and a letter of the alphabet or a number. Similarly, no space is allowed between letters of the alphabet and numbers.
- Up to four characters can be used.
- An alphabetic character can only be used in a program name if it begins with an alphabetic character. Although a program beginning with an alphabetic character can contain a number in it, one that begins with a number cannot contain an alphabetic character.
- Although all of the four characters may be numeric, program names of the type “OO***” (***: alphanumeric) cannot be used since this kind of program name is used for system operation, automating functions, etc.
- A block which contains a program name must not contain other commands.
- A program name may not be used for a schedule program.
- The program name assigned to a main program / subprogram must begin with address character “O”.
- Since program names are handled in units of characters, the following names are judged to be different program names.
 - O0123 and O123
 - O00 and O0
- All program names must be unique.
If program name “O1” is used for more than one program, the operation to call program “O1” may call a program differing from the desired one.

3. Sequence Name

All blocks in a program are assigned a sequence name that begins with address character “N” followed by an alphanumeric sequence.

Functions such as a sequence search function, a sequence stop function and a branching function can be used for blocks assigned a sequence name.

A sequence name that contains only alphabetic characters is called a sequence label and the one that contains only numbers is called a sequence number. In this manual, both of them are referred to as a sequence name.

Sequence Name Designation

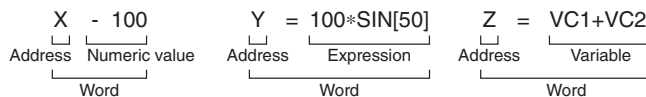
- Enter letters of the alphabet (A to Z) or numbers (0 to 9) following address character “N”.
- Up to five characters can be entered in succession to the address.
- Both alphabetic characters and numbers may be used in a sequence name. If an alphabetic character is used in a sequence name, however, the sequence name must begin with an alphabetic character.
- Although a sequence name must be specified at the beginning of a block, an optional block skip code may be placed before a sequence name.
- Sequence numbers may be specified in any order.
- Since sequence names are handled in units of characters, the following names are judged to be different sequence names.
 - N0123 and N123
 - N00 and N0
- When a sequence label is used, place a space or a tab after the sequence label.

4. Program Format

4-1. Word Configuration

A word is defined as an address character followed by a group of numeric values, an expression, or a variable name. If a word consists of an expression or a variable, the address character must be followed by an equal sign “=”.

Examples:



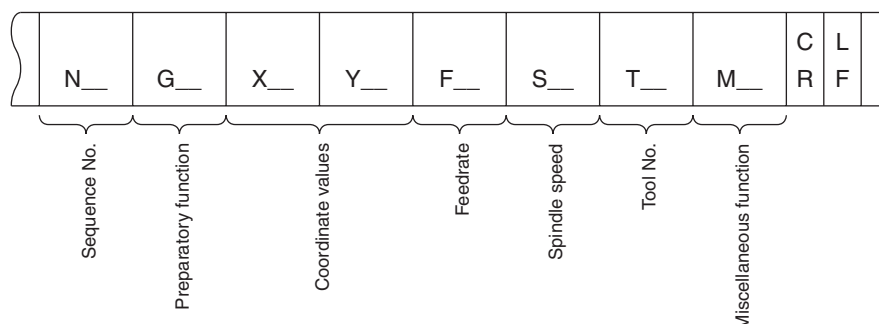
ME33018R1000300040001

- An address character is one of the alphabetic characters A through Z and defines the meaning of the entry specified following it. In addition, an extended address character, consisting of two alphabetic characters, may also be used.
- Refer to SECTION 11, “Variable Function” for more information on variables.
- Hexadecimals may be used for numeric values.
Example: X#1000H (same as X4096)

4-2. Block Configuration

A group consisting of several words is called a block, and a block expresses a command. Blocks are delimited by an end of block code.

- The end of block code differs depending on the selected code system, ISO or EIA:
ISO: LF
EIA: CR
- A block comprises several words.
- A block may contain up to 158 characters.
A block consists of the following commands, for example.



ME33018R1000300050001

4-3. Program

A program consists of several blocks.

4-4. Programmable Range of Address Characters

The programmable ranges of numerical values of individual address characters are shown in the following table.

Address	Function	Programmable Range		Remarks
		Metric	Inch	
O	Program name	0000 - 9999	Same as metric	Alphabetic characters available
N	Sequence name	00000 - 99999	Same as metric	Alphabetic characters available
G	Preparatory function	0 - 399	Same as metric	Mnemonics available
X, Y, Z, U, V, W	Coordinate values (linear axis)	±99999.999mm	±9999.9999inch	
I, J, K	Coordinate values of center of arc	±99999.999mm	±9999.9999inch	
R	Radius of arc	±99999.999mm	±9999.9999inch	
A, B, C	Coordinate values of rotary axis	±360.0000deg	Same as metric	Multi-turn specification ±9999.9999deg
F	Feed per minute	0.1 - 24000.0 mm/min	0.01 - 2400.00 inch/min	
	Feed per revolution	0.001 - 500.000 mm/rev	0.0001 - 50.0000 inch/rev	
	Dwell time period	0.001 - 99999.999 sec	Same as metric	
S	Spindle speed	0 - 65535	Same as metric	
T	Tool number	1 - 9999	Same as metric	
M	Miscellaneous function	0 - 511	Same as metric	
H	Tool length offset number	1 to maximum tool data number	Same as metric	
D	Cutter radius compensation number	1 to maximum tool data number	Same as metric	
P	Dwell time period (during fixed cycle)	0.001 - 99999.999 sec	Same as metric	
Q	Second dwell time period (during fixed cycle)	0.001 - 99999.999 sec	Same as metric	
	Depth of cut (during fixed cycle)	0 - 99999.999 mm	0 - 9999.9999inch	
	Repetition time (schedule program)	1 - 9999	Same as metric	
R	Cut starting level (during fixed cycle)	±99999.999mm	±9999.9999inch	

*: An alarm occurs when any of the following addresses is specified more than once within a block:
X, Y, Z, U, V, W, A, B, C, F.

5. Mathematical Operation Functions

Mathematical operation functions are used to convey logical operations, arithmetic operations, and trigonometric functions. A table of the operation symbols is shown below. Operation functions can be used together with variables to control peripherals or to pass on the results of an operation.

Category	Operation	Operator	Remarks
Logical operation	Exclusive OR	EOR	0110 = 1010 EOR 1100 (See *3.)
	Logical OR	OR	1110 = 1010 OR 1100 (See *3.)
	Logical AND	AND	1000 = 1010 AND 1100 (See *3.)
	Negation	NOT	1010 = NOT 0101
Arithmetic operation	Addition	+	8 = 5 + 3
	Subtraction	-	2 = 5 - 3
	Multiplication	*	15 = 5 * 3
	Division	/ (slash)	3 = 15/5
Trigonometric functions, etc.	Sine	SIN	0.5 = SIN [30] (See *4.)
	Cosine	COS	0.5 = COS [60] (See *4.)
	Tangent	TAN	1 = TAN [45] (See *4.)
	Arctangent (1)	ATAN	45 = ATAN [1] (value range: -90° to 90°)
	Arctangent (2)	ATAN2	30 = ATAN 2 [1,(Square root 3)] (See *1.)
	Square root	SQRT	4 = SQRT [16]
	Absolute value	ABS	3 = ABS [-3]
	Decimal to hexadecimal conversion	BIN	25 = BIN [\$25] (\$ represents a hexadecimal number.)
	Hexadecimal to decimal conversion	BCD	\$25 = BCD [25]
	Integer implementation (rounding)	ROUND	128 = ROUND [1.2763 x 102]
	Integer implementation (truncation)	FIX	127 = FIX [1.2763 x 102]
	Integer implementation (raising)	FUP	128 = FUP [1.2763 x 102]
	Unit integer implementation (rounding)	DROUND	13.265 = DROUND [13.26462] (See *2.)
	Unit integer implementation (truncation)	DFIX	13.264 = DFIX [13.26462] (See *2.)
	Unit integer implementation (raising)	DFUP	13.265 = DFUP [13.26462] (See *2.)
Remainder	MOD	2 = MOD [17, 5]	

SECTION 1 PROGRAM CONFIGURATIONS

Category	Operation	Operator	Remarks
Brackets	Opening bracket	[Determines the order of calculation. (Expression in inner brackets is calculated first.)
	Closing bracket]	

- *1. The value of ATAN2 [b, a] is an argument (range: -180° to 180°) of the point that is expressed by coordinate values (a, b).
- *2. In this example, the setting unit is mm.
- *3. Blanks must be placed before and after the logical operation symbols (EOR, OR, AND, NOT).
- *4. Numbers after function operation symbols (SIN, COS, TAN, etc.) must be enclosed in brackets "[]". ("a", "b", and "c" are used to indicate the contents of the corresponding bits.)

Logical Operations

- Exclusive OR (EOR) $c = a \text{ EOR } b$
If the two corresponding values agree, EOR outputs 0.
If the two values do not agree, EOR outputs 1.

a	b	c
0	0	0
0	1	1
1	0	1
1	1	0

- Logical OR (OR) $c = a \text{ OR } b$
If both corresponding values are 0, OR outputs 0.
If not, OR outputs 1.

a	b	c
0	0	0
0	1	1
1	0	1
1	1	1

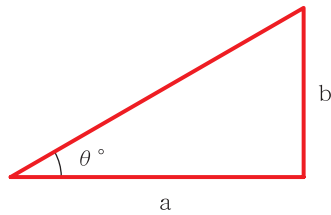
- Logical AND (AND) $c = a \text{ AND } b$
If both corresponding values are 1, AND outputs 1.
If not, AND outputs 0.

a	b	c
0	0	0
0	1	0
1	0	0
1	1	1

- Negation (NOT) $b = \text{NOT } a$
NOT inverts the value (from 0 to 1, and 1 to 0).

a	b
0	1
1	0

- Arc tangent (1) (ATAN)
 $\theta = \text{ATAN} [b/a]$
- Arc tangent (2) (ATAN2)
 $\theta = \text{ATAN2} [b/a]$



ME33018R1000300080001

- Integer implementation (ROUND, FIX, FUP)
 Converts a specified value into an integer (in units of microns) by rounding off, truncating, or raising the number at the first place to the right of the decimal point.

6. Optional Block Skip

[Function]

Blocks preceded by “/n” are ignored in automatic operation mode if the BLOCK SKIP switch on the machine panel is set ON. If the switch is OFF, these blocks are executed normally. The optional block skip function allows an operator to determine if a specific block should be executed or ignored in automatic mode operation.

When the block skip function is called, the entire block will be ignored.

[Details]

- In the standard specification, one optional block skip can be specified; as an option, up to nine are possible. These are distinguished in code as follows: “/1”, “/2”, “/3”. Note that “/” has the same meaning as “/1” when this option is selected.
- A slash code “/” must be placed at the start of a block. If it is placed in the middle of a block, an alarm is activated. A sequence name may precede a slash code “/”.
- A slash code “/” may not be contained in the program name block.
- Blocks which contain a slash code “/” are also subjected to the sequence search function, regardless of the BLOCK SKIP switch position.
- Sequence stop is not executed at a block which contains a slash code “/” in single block mode operation if the BLOCK SKIP switch is ON. The succeeding block is executed, and then the operation stops.
- This function is also available in the schedule program.

7. Program Branch Function (Optional)

[Function]

The program branch function executes or ignores the program branch command specified in a part program according to the ON/OFF setting of the PROGRAM BRANCH switch on the machine panel. The function corresponds to maximum two program branch switches, PROGRAM BRANCH 1 and PROGRAM BRANCH 2 (extended to maximum nine switches by additional option). If the switch is ON, the program branches when the following command is read.

- IF VPBR1 N*** The program branches to N*** block if the PROGRAM BRANCH 1 switch is ON.
- IF VPBR2 N*** The program branches to N*** block if the PROGRAM BRANCH 2 switch is ON.

Example:

```

    IF VPBR1 N100 ← Branching to N100 if PROGRAM BRANCH 1 switch is ON.
    G00 X100 Z100
N100 G00 Y100
    IF VPBR1 N200 ← Branching to N200 if PROGRAM BRANCH 2 switch is ON.
    G00 X200 Z200
N200 G00 Y200
    M02

```

ME33018R1000300100001

[Details]

- In operation method B (large-volume program operation mode), use a sequence label name to specify the branch destination.
- The program branch function has the same restrictions as the branch function of User Task 1.
- A program branch command (IF VPBR1 N*** or IF VPBR2 N***) must be specified in a block without other commands.

8. Comment Function (Control OUT/IN)

A program may be made easier to understand by using comments in parentheses.

- A comment must be parenthesized to distinguish it from general operation information. All information placed in parentheses is regarded by the machine as comments.
- Comments are displayed in the normal character size.

Example:

```

N100 G00 X200 (FIRST STEP)
           Comment

```

ME33018R1000300110001

9. Message Function (Optional)

[Function]

For conditional branching it may be necessary to display a message, depending on the processing at the destination of the branching. The message function is used in such cases, and the message is displayed in enlarged characters.

[Format]

MSG (message statement)

[Details]

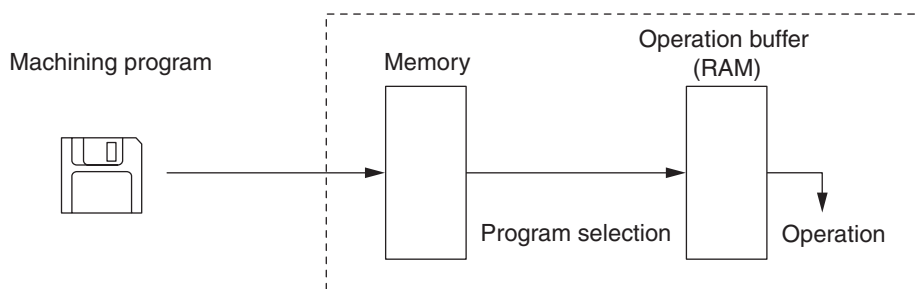
- The display of a message statement on the screen is twice the size of normal characters.
- If the MSG code is not followed by a message statement, the comment statement given last up to the present block will be displayed.
- Up to 128 characters may be used in a message statement.
- The message function is possible only during machine operation mode.
- The following code can be used in the program to return the screen to the previous status after the message has been displayed: NMSG

10. Operation Methods and Program Storage Memory Capacity

(1) Operation Capacity

The NC has a memory to store machining programs. The memory capacity is selected depending on the size of the user program. On execution of a program, the program is transferred from the memory to the operation buffer (RAM).

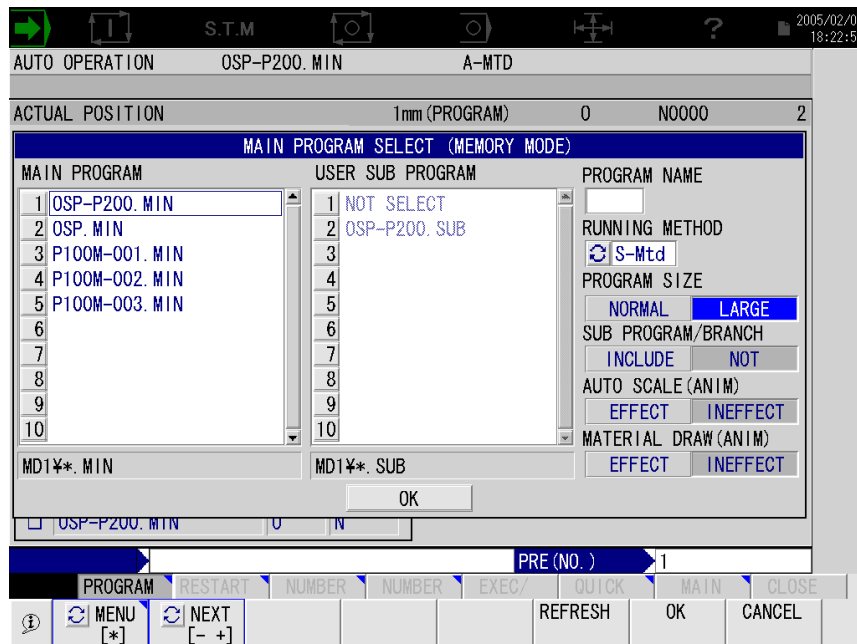
If the program size is larger than the operation buffer capacity, (for example, if the program size is larger than 320 m (1050 ft.) although the operation buffer capacity is 320 m (1050 ft.)), the program cannot be transferred from the memory to the operation buffer in batch (at one time). Depending on the size of a program in comparison to the operation buffer capacity, two types of operation methods are available (operation method A and operation method B), and restrictions apply in programming according to the operation method used.



ME33018R1000300130001

(2) Operation Methods

Select the operation method using the pop-up window MAIN PROGRAM SELECT (MEMORY MODE) that appears when calling a program to be run. The operation method can be also selected by the setting at the NC optional parameter (word) No. 11.



ME33018R1000300130002

- When A-Mtd is selected
Program running method A becomes effective.
The program to be executed is transferred to the operation buffer in batch.
This method is used when the program is smaller than the operation buffer capacity.
- When B-Mtd is selected
Program running method B becomes effective.
The program to be executed is called to the operation buffer in several segments.
This method is used when the program is larger than the operation buffer capacity.
Since schedule programs, subprograms, and library programs are generally called to the operation buffer in batch, these programs must be created with restriction placed on their capacities.
- When S-Mtd is selected
Program running method S becomes effective.
This method is used to execute a large program which does not use branch or subprogram call functions.

SECTION 1 PROGRAM CONFIGURATIONS

- When selecting an operation method, also select the program size and whether the program has a sub program branch or not (only in the case of operation A and B). The table below shows the relation between the operation method and the program size.

Item		Program of normal size	Large program		
Program running method		Method A	Method B	Method S	
Program size limit	Main program	Total program size is 2MB.	2GB		
	Sub program		-		
	Library program		Total program size is about 1.8 MB.	Total program size is about 1.8 MB.	
	Schedule program				
Sub program function		Usable	Usable	Unusable (alarm)	
Branch function		Usable	Usable	Unusable (alarm)	
Destination of a jump specified in branch command	Main program	Sequence label or sequence number	Sequence label or sequence number	-	
	Sub program				
	Library program				
	Schedule program				
Main program sequence label limit		Unlimited	Unlimited	Unlimited	
Program selection time		*1	*1	Completed immediately	

*1. Time varies with the selected program size.

(3) Programming Restrictions for the Operation Method

For details of restrictions that must be taken into consideration when writing a program, refer to SECTION 12, "PSELECT BLOCK".

(4) Others

- The maximum capacity for running the main program is about 2 GB when the operation method B is selected.
- The library program capacity is equivalent to the designated library program buffer size. This means that the library program buffer size is always contained in the operation capacity even if a library program is not registered.
- The number of subprograms and library programs stored in memory is independent of the operation buffer size. They are always 126 and 65, respectively.

SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS

1. Coordinate System

1-1. Coordinate Systems and Values

In order to move a cutting tool to a target position, a coordinate system must be established to specify the target position using coordinate values in the coordinate system.

The OSP-P200M uses three types of coordinate system (machine coordinate system, work coordinate system, and local coordinate system). These coordinate systems are briefly explained below.

- Machine coordinate system
The machine coordinate system is set by the machine tool manufactures. Although the setting may be changed by the user, machine dependent setting values such as pitch error compensation data and travel limit values must be changed accordingly.
- Work coordinate system
A work coordinate system is set by the user.
- Local coordinate system
A local coordinate system set temporarily by the commands in a program.
The user can select the coordinate system to be used as needed from the coordinate systems indicated above.
The coordinate value is represented by components of the axes which make up the coordinate system. Usually, a maximum of six axis components is used (the number differs depending on the NC unit specifications.)
Example:

X__Y__Z__W__A__C__

ME33018R1000400010001

The number of programmable axes, that is, the number of axis components used to define a coordinate value varies depending on the machine specifications. This manual, therefore, uses the following designation to indicate a coordinate value.

IP__

1-2. Machine Zero and Machine Coordinate System

The reference point specific to the individual machine is referred to as the machine zero and the coordinate system having the machine zero as the origin is referred to as the machine coordinate system.

The machine zero is set for each individual machine using system parameters.

Since the travel end limits and the home positions are set in the machine coordinate system, the user should not change the location of the machine zero at his/her own discretion.

A cutting tool may not always be moved to the machine zero.

1-3. Work Coordinate System

The coordinate system used to machine workpieces is referred to as the work coordinate system.

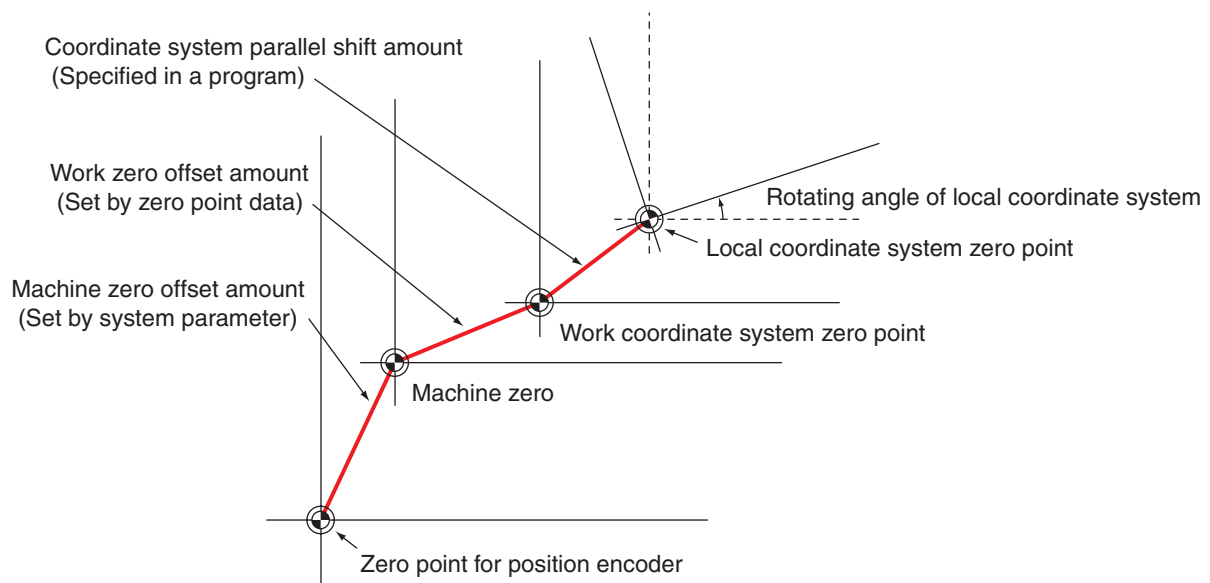
- Work coordinate systems are established and stored with work coordinate system numbers in the memory before starting operation. The desired work coordinate system may be called at the start of machining.
- Work coordinate systems are set by specifying the distance from the machine zero to the origin of a work coordinate system as an offset value (work zero offset).
- For details, see SECTION 4, "Selection of Work Coordinate System" and SECTION 4, "Change of Work Coordinate System".

1-4. Local Coordinate System

Programming the entire operation of a workpiece using only a work coordinate system may sometimes be difficult on some portions of the workpiece. In such cases, programming is facilitated by setting a new coordinate system appropriate for a specific workpiece portion.

The new coordinate system is referred to as a local coordinate system.

- The desired local coordinate system can be established by specifying the origin in reference to the origin of the presently selected work coordinate system and the angle of rotation on the specified plane about the origin of the local coordinate system to be set with G11. Once a local coordinate system has been established, all coordinate values are executed in the newly set local coordinate system.
To change the local coordinate system to another one, the position of the origin of the new local coordinate system and the angle of rotation about the origin should be specified with G11.
As explained above, a local coordinate system can be established only by specifying the coordinate values of the origin and the angle of rotation in a program.
- To designate coordinate values in the work coordinate system, cancel the local coordinate system by specifying G10.
- For details, refer to SECTION 4, "Parallel Shift and Rotation of Coordinates System".



2. Coordinate Commands

2-1. Numerically Controlled Axes

- The following table lists the addresses to be specified to control the axes.

	Address	Contents
Basic axis	X, Y, Z	Addresses corresponding to the three axes orthogonal to one another
Parallel axis	U, V, W	Addresses of three orthogonal axes parallel to the basic axes
Rotary axis	A, B, C	Addresses of rotary axis in a plane right angle to the basic axis
Circular interpolation parameters	I, J, K	Addresses specifying distances, parallel to an individual axis, from a start point to the center of an arc
	R	Addresses specifying the radius of an arc

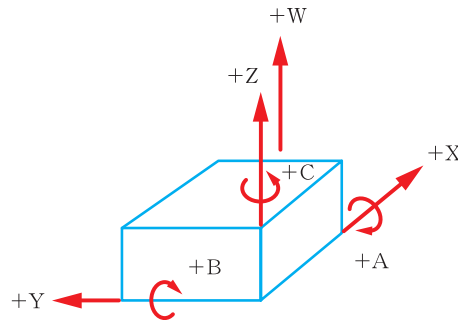
- An axis movement command consists of an axis address, a sign indicating the direction of the axis movement, and a numeric value which describes the axis movement. Refer to “Absolute and Incremental Commands” for the designation of numeric values.
- In this manual, to simplify the explanation for axis designation, “Xp”, “Yp”, and “Zp” are used instead of the actual axis addresses. They represent the axis as follows:
 Xp X-axis and the axis parallel to X-axis (U-axis)
 Yp Y-axis and the axis parallel to Y-axis (V-axis)
 Zp Z-axis and the axis parallel to Z-axis (W-axis)
- The maximum number of controllable axes is six. This capability varies depending on the NC model.
- The following table shows the number of simultaneously controllable axes in each of the axis movement modes.

	Number of Simultaneously Controllable Axes (“n” represents the number of controllable axes.)
Positioning	n
Linear interpolation	n
Circular interpolation	2
Helical cutting	3
Manual operation	1
Pulse handle operation	1

In pulse handle operation, the optional 3-axis control function is available.

SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS

- The positive directions of the linear and rotary axes are defined as follows:



ME33018R1000400050001

The definition of the coordinate axes and directions conforms to ISO R841.
ISO: International Organization of Standardization

2-2. Unit Systems

The unit systems that can be used in a program are described below. Note that the unit system selected for programming and the unit system used for setting data such as zero point, tool data, and parameters are independent of each other. The unit systems to be used for inputting the data are set at NC optional parameter (INPUT UNIT SYSTEM).

2-2-1. Minimum Input Unit

The minimum input unit is the smallest unit of a value that may be entered in a program. For a linear axis, the minimum input unit is 0.001 mm or 0.0001 inch.

For the NC with metric / inch switchable specification, the unit system can be selected by the setting at LENGTH UNIT SYSTEM of NC optional parameter (INPUT UNIT SYSTEM).

For a rotary axis, the minimum input unit is 0.001 degree or 0.0001 degree. Either 0.001 degree or 0.0001 degree can be selected by the setting at ANGLE of NC optional parameter (INPUT UNIT SYSTEM).

2-2-2. Basic Input Unit

The input unit may be changed to the "basic" unit by the setting at LENGTH of NC optional parameter (INPUT UNIT SYSTEM). The fundamental units are then 1 mm, 1 inch, 1 degree, and 1 second.

2-2-3. Numeric Values (inch / metric switchable as optional function)

As the unit for specifying program values, “mm”, “deg.”, “sec”, etc. are used. For these units, a decimal point may be used.

- Cautions on using a decimal point value
 - a. A decimal point value must not be used for addresses O, N, G, and M.
 - b. If a decimal point is not entered in a numeric value, the decimal point is assumed to exist at the end of the specified numeric value.
 - c. If a value is set below the specified minimum input unit, the data is processed in the following manner.
 - For addresses S, T, H, D, Q, etc. that require integer type data, the value below the minimum input unit is truncated.
 - For addresses that use real data, the value below the minimum input unit is rounded.
- The input unit of dimension commands is determined by the setting at NC optional parameter (INPUT UNIT SYSTEM) or NC optional parameter (bit) No. 3, bit 0 to bit 7 and No. 4, bit 0. How these bits set the input unit is shown below.
- NC optional parameter (INPUT UNIT SYSTEM) screen

PARAMETER OSP-P200. MIN A-MTD

2005/02/07 18:25:40

NC OPTIONAL PARAMETER

INPUT UNIT SYSTEM

	PART PROGRAM UNIT	DATA SET UNIT		Co
LENGTH UNIT SYSTEM	<input checked="" type="radio"/> mm <input type="radio"/> inch	<input type="radio"/> mm <input type="radio"/> inch	X	52.123
REAL NUMBER	<input type="radio"/> YES <input type="radio"/> NO	<input type="radio"/> YES <input type="radio"/> NO	Y	19.447
LENGTH	<input type="text" value="1"/> (mm)	<input type="text" value="1"/> (mm)	Z	10.000
ANGLE	<input type="text" value="1"/> (deg)	<input type="text" value="1"/> (deg)	(DISTANCE)	
FEEDRATE	<input type="text" value="1"/> (mm/min)		X	0.000
FEEDRATE	<input type="text" value="1"/> (mm/rev)		Y	0.000
DWELL TIME	<input type="text" value="1"/> (sec)		Z	0.000
ROUND, FIX, FUP ARE CONVERT	<input type="radio"/> INTEGER <input type="radio"/> DECIMAL POINT			

NC OPTIONAL PARAMETER BIT NO. 3 BIT0

PRE (NO.) 4

MENU [*] ITEM ↑ ITEM ↓ DISPLAY CHANGE

SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS

- NC optional parameter (bit) No. 3, bit 0 to bit 7 and No. 4, bit 0

Parameter No.	Bit No.	Contents	With Check Mark	Without Check Mark
3	0	Sets the unit system of length, "inch" or "mm". (*2)	inch	mm
	1	Sets the unit of 1 mm, 1 inch, 1 deg., and 1 sec.	Unit of 1 mm, 1 inch, 1 deg., and 1 sec is selected.	Conforms to the setting for bit 2 to bit 5 and bit 7 of No. 3 and bit 0 of No. 4.
	2	Sets unit of length, "0.01 mm" or "0.001 mm".	0.01 mm	0.001 mm
	3	Sets the unit of feedrate, 0.1 mm/min, 0.01 inch/min, or 1 mm/min, 0.1 inch/min	0.1 mm/min 0.01 inch/min	1 mm/min 0.1 inch/min
	4	Sets the unit of feedrate, 0.001 mm/rev, 0.0001 inch/rev, or 0.01 mm/rev, 0.001 inch/rev	0.001 mm/rev 0.0001 inch/rev	0.01 mm/rev 0.001 inch/rev
	5	Sets the unit of time, "0.01 sec" or "0.1 sec".	0.01 sec	0.1 sec
	6	Sets the unit place at "1 mm", "1 inch", "1 deg", and "1 sec" when decimal point input is selected.	Unit place is set at "1 mm", "1 inch", "1 deg", and "1 sec".	Conforms to the setting for bit 1 to bit 5, and bit 7.
4	7	Sets the unit time, "0.001 sec" or "0.1 sec".(*1)	0.001 sec	0.1 sec
	0	Sets the unit of angle, "0.001 deg" or "0.0001 deg".	0.0001 deg	0.001 deg

*1: The unit of time is always "0.01 sec" if "1" is set for bit 5.

*2: The setting for bit 0 is valid only when the inch/mm switchable specification is selected.

SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS

- Examples of parameter setting are given below.
(●: With check mark, ○: Without check mark)

- mm system

No.4 0	Parameter Bit No.3								Length (mm)	Angle (°)	Feedrate (mm/min)	Feedrate (mm/rev)	Time (sec)
	7	6	5	4	3	2	1	0					
*	*	*	*	*	*	*	●	○	1	1	1	1	1
○	○	*	○	○	○	○	○	○	0.001	0.001	1	0.01	0.1
○	○	*	○	○	○	●	○	○	0.01	0.001	1	0.01	0.1
○	○	*	○	○	●	○	○	○	0.001	0.001	0.1	0.01	0.1
●	○	*	○	●	○	○	○	○	0.001	0.0001	1	0.001	0.1
●	*	*	●	○	○	○	○	○	0.001	0.0001	1	0.01	0.01
●	●	*	○	○	○	○	○	○	0.001	0.0001	1	0.01	0.001

ME33018R1000400090002

- inch system

No.4 0	Parameter Bit No.3								Length (inch)	Angle (°)	Feedrate (inch/min)	Feedrate (inch/rev)	Time (sec)
	7	6	5	4	3	2	1	0					
*	*	*	*	*	*	*	●	●	1	1	1	1	1
○	○	*	○	○	○	*	○	●	0.001	0.001	0.1	0.01	0.1
○	○	*	○	○	○	*	○	●	0.001	0.001	0.01	0.01	0.1
○	○	*	○	○	●	*	○	●	0.001	0.001	0.1	0.01	0.1
○	○	*	○	●	○	*	○	●	0.001	0.001	0.1	0.001	0.1
●	*	*	●	○	○	*	○	●	0.001	0.0001	0.1	0.01	0.1
●	●	*	○	○	○	*	○	●	0.001	0.0001	0.1	0.01	0.001

ME33018R1000400090003

An asterisk (*) in the table indicates setting of “0” or “1” is allowed.

- μm / mm unit system

The unit system that handles the data in units of mm (inch) for real data and in units of microns (1/10000 inch) is called the “μm / mm unit system”. For this unit system, the unit is determined depending on whether or not a decimal point is used in the data when YES is selected at REAL NUMBER of NC optional parameter (INPUT UNIT SYSTEM). If a decimal point is used, the unit of “mm (inch)” is set and if a decimal point is not used, the unit of “microns (1/10000 inch)” is set.

Example 1:

X100. → 100mm
X100 → 100μm

ME33018R1000400090004

If an expression or a variable is used for the command of this unit system, the values are always treated as real data.

Example 2: Local variables

PX = 100 PX = 100.
X = PX → 100mm X = PX → 100mm

ME33018R1000400090005

(The value is not “100 μm”.)

SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS

The following is a comparison how a numeric value is interpreted according to whether or not a decimal point is used when “ μm / mm unit system” is selected.

Command Element		Value	“mm unit system” element
X	100	100 μm	–
X=	100	100 μm	–
X	100.	100 mm	Decimal point
X=	100.	100 mm	Decimal point
X	100+100	200 mm	Expression
X=	100.+100	200 mm	Expression
X=	100+100.	200 mm	Expression
X	100+100*2	300 mm	Expression
X=	100+100*2	300 mm	Expression
X=	100+100*2.5	350 mm	Expression
PK=	100	200 mm	Variable
X=	100+PK	200 mm	Variable
PK=	100.	100 mm	Variable
X=	200-PK	100 mm	Expression
X=	200-100	100 mm	Expression
X	-100	-100 μm	–
X	-100.	-100 mm	Decimal point
X	+100	100 μm	–
X	+100.	100 mm	Decimal point
X=	ROUND[100]	100 mm (*1)	Expression
X=	FIX[100.]	100 mm (*1)	Expression
X=	FUP[-100]	-100 mm (*1)	Expression
X=	ROUND[100.]	100 μm (*2)	–
X=	FIX[100.]	100 μm (*2)	–
X=	FUP[-100.]	-100 μm (*2)	–
LA1=4	F=FIX[LA1]	4 mm/min	Variable

(*1) Decimal point is selected for designation of ROUND/FIX/FUP real number command.

(*2) Integer is selected for designation of ROUND/FIX/FUP real number command.

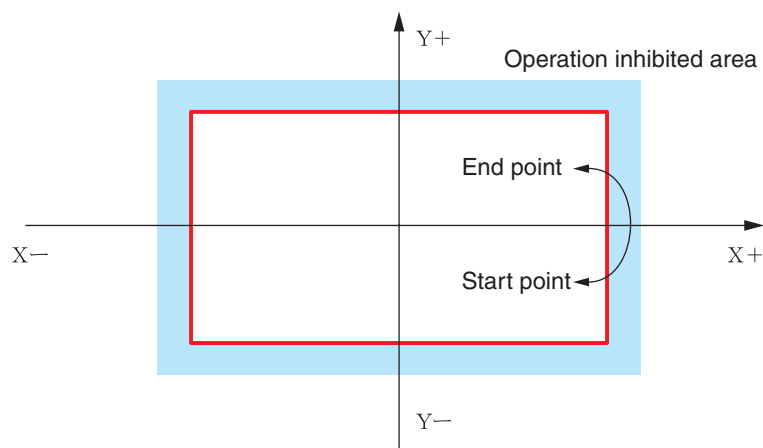
2-3. Travel Limit Commands (G22, G23) (Optional)

Since the NC is equipped with absolute position encoders, it is possible to set the travel limit with the software. That is, if the travel limit is set as an absolute value by the software, the limit switch usually used to detect the travel limit may not be used. If the travel limit is set in this manner, it is possible to change the travel limit position by changing the travel limit value in a program.

Note that two types of travel limit, one set by the manufacture (factory-set travel limit) and the other set by the user (user-set travel limit), are provided.

(1) Factory-Set Travel Limit (Soft-Limit)

- The travel limit is set in accordance with the maximum travel distance from the machine zero of each axis. The travel limits are set both in the positive (P) and negative (N) directions using the system parameters.
- The area inside of the set values (from the N direction travel limit to the P direction travel limit) is available for operation (operation permitted area). The outside area is called the operation inhibited area and axis movements into this area are not allowed.
- The travel limit function always monitors the programmed tool path. If the tool path enters the operation inhibited area, even if the end point lies in the operation permitted area, this function disables the tool movement.



ME33018R1000400100001

(2) User-Set Travel Limit (Programmable Limit) (Optional)

The travel limit may be set by the user either with user parameters or by programs using the programmable travel limit function. Since both settings (user parameter and programmed command) establish an identical area and since the data is stored in the same area, the data entered last becomes the valid data, updating the previously set data. For example, when the travel limits are set using a program after setting them with the user parameters, the travel limit setting data is replaced with the data set for the user parameters. When setting the travel limits, both positive (P) and negative (N) direction limit data must be set.

The area between the P and N travel limits is defined as the operation permitted area and that outside the travel limits is defined as the operation inhibited area.

SECTION 2 COORDINATE SYSTEMS AND COORDINATE COMMANDS

- Setting the travel limits by a program
[Programming format]

```
G22 X__Y__Z__α__β__γ__I__J__K__P__Q__R__
X ..... X Programmable limit in the P direction of X-axis
Y ..... Y Programmable limit in the P direction of Y-axis
Z ..... Z Programmable limit in the P direction of Z-axis
α ..... α Programmable limit in the P direction of 4th-axis
β ..... β Programmable limit in the P direction of 5th-axis
γ ..... γ Programmable limit in the P direction of 6th-axis
I ..... I Programmable limit in the N direction of X-axis
J ..... J Programmable limit in the N direction of Y-axis
K ..... K Programmable limit in the N direction of Z-axis
P ..... P Programmable limit in the N direction of 4th-axis
Q ..... Q Programmable limit in the N direction of 5th-axis
R ..... R Programmable limit in the N direction of 6th-axis
```

ME33018R1000400100002

The numeric values entered are processed as coordinate values in the work coordinate system.

“α”, “β”, and “γ” above do not represent an address. In actual programming, use axis addresses of the 4th to 6th axis (A, B, C, U, V, and W).

[Details]

- An alarm occurs if the command indicated above is executed for the machine equipped with a multi-turn type rotary axis.
- The data set using G22 is backed up and is therefore valid even after the power is turned off.
- If the setting data is outside the factory-set soft limits, an alarm will occur.
- Which of the travel limits - the limits set with the system parameters (soft-limit) or the limits set with user parameters or by a program (programmable limits) - becomes valid as the operation permitted area can be set by specifying an appropriate G code.
G22: Selects the travel limits set with user parameters or those newly set by G22 are as the travel limits and checks the program according to the selected operation permitted area.
G23: Cancels the G22 mode and selects the travel limits set with the system parameters. The program is checked according to the selected operation permitted area.
If G22 is specified independently, the programmable limit values set with user parameters become valid.
- For setting the travel limits with user parameters, refer to User Parameter, SECTION 4 PARAMETER in III DATA OPERATION of OPERATION MANUAL.
- The programmed tool path is checked for entry into the operation inhibited area even if the end point lies inside the operation permitted area.

2-4. Home Position Command (G30)

[Function]

The term “home position” refers to a particular position that can be set for individual machines. The home position command is used to move the axes to the preset home position.

The home position is used as the tool change position or the pallet change position.

[Programming format]

G30 P__

P: Home position number. Up to 32 home positions may be set.

Home positions are set with coordinate values in the machine coordinate system using system parameters.

[Supplement]

The operating sequence of the axes to move to the home position and the position of the home position are determined by the machine tool builder and differ according to machine. Before operating the machine, you must thoroughly understand the axis operating sequence and the position of the home position for each home position number.

For details on home positions, refer to SECTION 4 PARAMETER in III DATA OPERATION of OPERATION MANUAL.

How the individual axes move to the home position is determined according to the setting for NC optional parameter (bit) No. 46, bit 2, whether the path is generated along a straight line (linear interpolation mode) or not.

[Supplement]

After the execution of a home position command, it is necessary to execute positioning for all axes in the G90 mode (absolute command) before starting the next operation.

2-5. Absolute and Incremental Commands (G90, G91)

For the designation of axis movement distance, two types of commands (absolute commands, incremental commands) are available.

(1) Absolute Commands

G90 specifies the absolute dimensioning mode.

In this mode, the coordinate values in the selected work coordinate system are used to specify the movement of axes.

(2) Incremental Commands

G91 specifies the incremental dimensioning mode.

In this mode, the axis movement distance from the current position to the target position is used to specify the movement of axes.

[Details]

- It is not permissible to specify G90 and G91 in the same block.
- Either G90 or G91 is always valid.
- Which of G90 and G91 is made valid when the power is turned ON or when the NC is reset is determined by the setting for a parameter (NC optional parameter (bit) No. 18, bit 4).
- When an incremental command needs to be designated right after the completion of a fixed cycle, specify the movement of the cycle axis of the fixed cycle in the absolute mode before specifying incremental commands.

```
G81 X_Y_Z_R_F_
    X_Y_
    ⋮
    X_Y_
G80
G00 Z_
G91 X_Y_Z_
    ⋮
```

ME33018R1000400120001

- After executing a command such as G15, G16, or G92 that changes a coordinate system, it is necessary to execute positioning in the G90 mode for all axes. (After changing the coordinate system, a coordinate system must be established using absolute commands.)

2-6. Coordinate Recalculation Command (G97)

[Function]

After changing the home position (G30) or the coordinate system (G15, G16, G92, etc.), it is usually necessary to issue G90 (absolute command) to position each axis (to define the coordinates). The coordinate recalculation command eliminates such need.

[Programming format]

G97

[Supplement]

- 1) G97 is effective only in one block.
- 2) G97 cannot be specified with an axis movement command in the same block. If attempted, an alarm occurs.

SECTION 3 FEED FUNCTIONS

1. Rapid Feed

In the rapid feed mode, each of the axes moves at the specified rapid feedrate independently of other axes that are moved at the same time. Note that rapid feedrate differs depending on the machine specification. Consequently, the individual axes arrive at the target point at different times. Override is possible.

2. Cutting Feed

2-1. Feed per Minute (G94)

[Function]

This function sets the feedrate per minute of a cutting tool with a numeric value following address "F".

[Programming format]

G94

Setting unit:

Selection is possible from among 1 mm/min, 0.1 mm/min, 1 inch/min, 0.1 inch/min and 0.01 inch/min by setting NC optional parameter (INPUT UNIT SYSTEM).

Setting range: 0.1 ~ 24000.0 mm/min, 0.01 ~ 2400.00 inch/min

[Details]

- The allowable maximum feedrate that is called the "clamp feedrate" is set with NC optional parameter (long word) No. 10. If an axis is going to move beyond this limit, its feedrate is clamped at this clamp feedrate and the following alarm message is displayed at the alarm display line on the screen.
4204 ALARM-D Feedrate command limit over (replacing)
- The programmed feedrate can be overridden. The clamp feedrate is applied to the actual feedrate, or the overridden feedrate.

2-2. Feed per Revolution (G95)

[Function]

This function sets the feedrate per revolution of a cutting tool with a numeric value following address "F".

[Programming format]

G95

Setting unit:

Selection is possible from among 1 mm/rev, 0.01 mm/rev, 0.001 mm/rev, 1 inch/rev, 0.001 inch/rev or 0.0001 inch/rev by setting NC optional parameter (INPUT UNIT SYSTEM).

Setting range: 0.001 ~ 500.000 mm/rev, 0.0001 ~ 50.000 inch

[Details]

- Since the clamp feedrate is set in units of “mm/min” it is converted into a value in “mm/rev” units using the following formula:

$$f_m = f_r \times N$$
 where,
 N = spindle speed (rpm)
 f_m = feedrate (mm/min)
 f_r = feedrate (mm/rev)

2-3. F1-digit Feed Function (Optional)

The F1-digit feed function has two types of control:

Switch-type control:

In a program, feedrate commands are written with F1 to F8 and the actual feedrate is set with the corresponding setting switches (up to 8 sets) provided on the machine operation panel.

Parameter-type control:

The feedrate commands are written in a program in the same manner as with the switch-type control. Actual feedrates are set for 9 sets of parameters F1 through F9.

For details of feedrate setting procedure for the parameter-type control, refer to the SECTION 11 F1-digit Feed Command Function in Special Functions Manual.

- F1-digit feed designation is distinguished from F4-digit feed designation as follows.

(1) F1-digit Feed

Switch-type control: An integer in the range of 1 to 8

Parameter-type control: An integer in the range of 1 to 9

(2) F4-digit Feed

If a real number (including a variable) is specified following address F (F1. for example), the feedrate command is interpreted as an F4-digit command. Therefore, for the F1-digit feed function, a variable cannot be used to specify 1 to 8 (for switch-type control) or 1 to 9 (for parameter-type control).

Examples:

```
F1           The feedrate is determined by the setting of rotary switch F1.
F5.         Feedrate = 5 mm/min
           LA1 = 8
           Feedrate = 8 mm/min
           F = LA1
```

ME33018R1000500040001

- The selected feedrate code (F1 to F8 in the case of switch-type control and F1 to F9 in the case of parameter-type control) is not cleared even when the NC is reset. It is cleared if an F4-digit command is specified or when power is turned OFF and then back ON.
- A feedrate override setting is invalid if a feedrate is specified by an F1-digit feedrate command.
- If an F1-digit feedrate command is specified in the G95 mode (feed per revolution), an alarm occurs.

2-4. F0 Command During Cutting Feed

F0 commands during the cutting feed are activated. When "F0", "F0.0", or "F0." is specified, the NC operates considering the setting value of the optional parameter long word No. 62 as an F command value. Specified F0 command is activated in the succeeding cutting blocks.

- If F0 commands are inactivated, the following alarm appears. "2244 ALARM-B Data word: 'F' 2" Even if F0 commands are activated, an alarm appears as the current condition in the feed per revolution mode.
- If [F0], [F0.], or [F0.0] is specified in a block of cutting feed command (including in the NURBS mode, excluding synchronized tapping command), the value of the optional parameter long word No.62 is adopted as the F value of the command. However, F0 commands are available other than when specifying F commands directly.
- Not only X-, Y-, and Z-axes, commands are issued to the same block, regardless of whether W-axis command is specified.
Since the upper-limit speed of the W axis is considerably low compared to the basic axes, if a command is specified at the same time as the basic axes, the following alarm appears. "ALARM-A CON speed over" (Same as usual.)
If this alarm occurs, set a lower value to the setting value of the optional parameter long word No.62
- The speed is clamped at the upper limit feedrate (in the Super NURBS control parameter). If a desired speed cannot be achieved due to the upper limit feedrate, change the upper limit feedrate.
- The speed is changed by the cutting feed override.
However, when the setting value of the optional parameter long word No.62 is 10 m/min, the override is clamped at 10 m/min by the maximum feedrate value or upper limit feedrate even if the override is set 100 % or more.

a. Optional parameter long word

No.	Description	Details	
62	Feedrate value when specifying F0 command	Range	0 to 10000000
		Initial value	10,000,000
		Unit	μm/min

Set an F command value for the case when the F0 is specified. The setting is allowed up to 10 m/min.

If the setting value is "0", the initial value 10 m/min is automatically set on the initialization.

b. Optional parameter bit

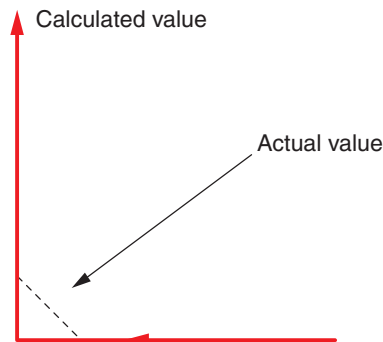
NO.	Bit	Description	<input checked="" type="checkbox"/>	<input type="checkbox"/>
78	1	F0 command is not set as an alarm.	Not set as alarm	Set as alarm (Initial value)

This parameter is not effective when turning on the power.

3. Exact Stop Check Function (G09, G61, G64)

[Function]

- During axis feed control, the calculated value always precedes the actual value when an axis is moving to the target point. Therefore, if the calculated value is at the target point, the actual value is behind the calculated value and is not at the target position. If the next block is executed at the time the calculated value reaches the target point, the actual position follows the calculated value, causing the tool path to stray from the programmed path at the join between two blocks.
- The exact stop function successfully eliminates errors caused by the axis control indicated above. With the exact stop function, operation for the next block does not start until the actual value arrives at the target point even if the operation of the current block has completed, so that the tool path exactly follows the programmed path.
The state where the current position has reached the target point is referred to as the “in-position” state. To establish the in-position state, the target point is defined with a band that is set using a system parameter.



ME33018R1000500060001

- The exact stop check mode may be either one-shot (valid only for a programmed block) or modal, as explained in the following.
Note that in the positioning mode (G00, G60), an exact stop check is always executed regardless of whether or not an exact stop check G code is specified.

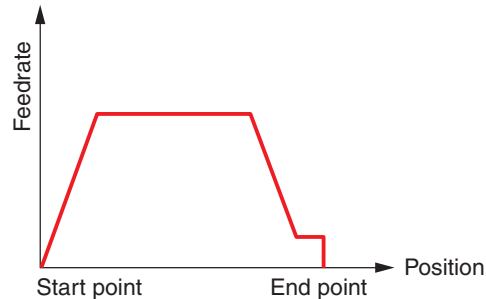
[Programming format]

- One-shot exact stop check command: G09 IP__
Exact stop check is executed only in the specified block.
- Modal exact stop check command: G61 IP__
Exact stop check is executed for all blocks until a cutting mode (G64) is specified.
- Cutting mode (cancel G61): G64 IP__
At a join between blocks containing cutting commands, the commands in the next block are executed immediately so that axis movements will not be decelerated at the join.
Even in the cutting mode, however, an exact stop check is executed in the positioning mode (G00, G60) or in a block containing the one-shot exact stop command (G09).
An exact stop check is also executed at blocks where cutting feed does not continue.

4. Automatic Acceleration and Deceleration

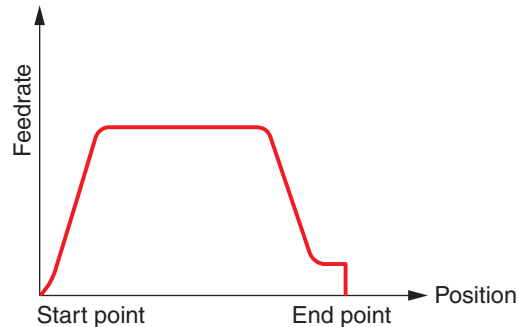
At the start and end of axis movements, axis feedrate is automatically accelerated and decelerated.

- (1) Automatic Acceleration/Deceleration in Positioning Mode / Manual Feed Mode
Axis feed is accelerated and decelerated in a linear pattern as shown below.



ME33018R1000500070001

- (2) Automatic Acceleration/Deceleration in the Cutting Feed Mode (G01, G02, G03)
In the cutting feed mode, axis feed is automatically accelerated and decelerated in the appropriate pattern as shown below.



ME33018R1000500070002

- (3) Processing Between Blocks

Type of New Block	Type of Old Block		
	Positioning	Cutting Feed	No Axis Movement
Positioning	O	O	O
Cutting Feed	O	X	O
No axis movement	O	O	O

O: Program advances to the next block after executing the in-position check.

X: Program advances to the next block without executing the in-position check.

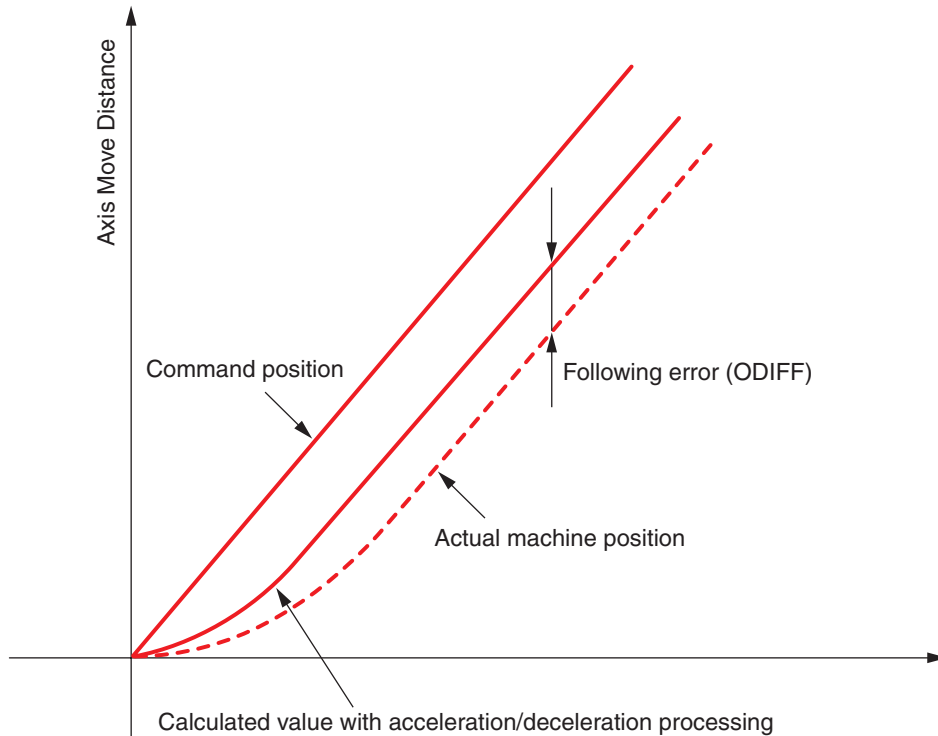
The term "in-position check" indicates the check if the actual axis position is within a predetermined width from the specified coordinate value. The in-position width is set for a system parameter.

- (4) Exact Stop (G61, G09)

As indicated in the table above, since in-position check is not performed when cutting feed blocks continue, the join between cutting feed blocks will be dulled or rounded. The exact stop function prevents the join between such blocks from being dulled or rounded. For details, refer to SECTION 3, "Exact Stop Check".

5. Following Error Check

Following error is defined as the difference between the command value output from the NC and the output of the position encoder. A DIFF over alarm occurs if a following error exceeds a certain value during rapid feed or cutting feed of an axis.



ME33018R1000500080001

6. Positioning (G00)

[Function]

The axes move from the present position to the target position at rapid feedrate. During this movement, axes are automatically accelerated and decelerated.

[Programming format]

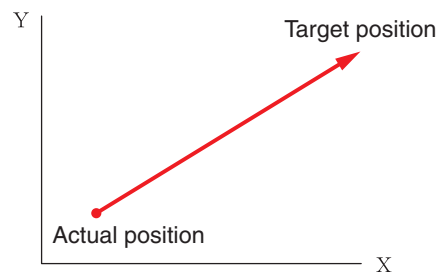
G00 IP__

In the positioning operation executed in the G00 mode, in-position check is executed. The commands in the next block are executed only after the in-position state is confirmed (in-position width is set for a system parameter).

[Details]

- Whether positioning is executed in the linear pattern or a non-linear pattern is determined by the setting for NC optional parameter (bit) No. 46, bit 0.
 - a. Linear interpolation pattern

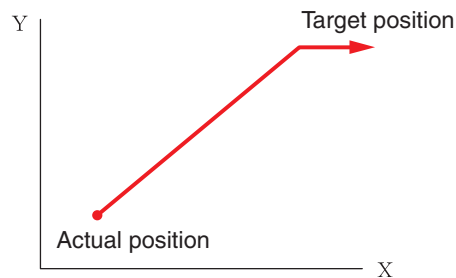
The tool path is generated along a straight line from the actual position to the target position. In this movement, the feedrates of the individual axes are determined within the individual rapid feedrates so that positioning time can be minimized.



ME33018R1000500090001

- b. Non-linear interpolation pattern

The individual axes move independently of each other at the individual rapid feedrates. Therefore, the resultant tool path is not always a straight line.



ME33018R1000500090002

- The rapid feedrate of the individual axes is set by the machine tool builder and cannot be changed.
- The in-position range is set for each axis using system parameters.

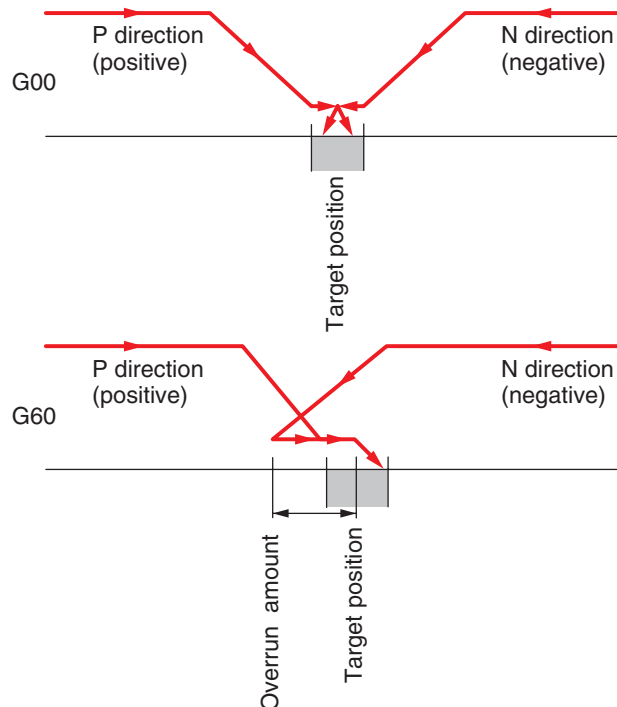
7. Uni-directional Positioning (G60)

[Function]

- In the positioning called by G00, positional error is unavoidable if positioning is executed in different directions due to backlash in the axis feed mechanism. If positioning is always executed in the same direction, the influence of backlash is eliminated and therefore high positioning accuracy can be obtained. The function to always execute positioning in the same direction is called the unidirectional positioning function.
- If the positioning is going to be executed in the direction opposite to the direction set at positioning direction of NC optional parameter (uni-directional positioning), the axis passes over the target point once and then moves back to the target point. The amount by which the axis passes beyond the target point (overrun amount) is set at either G60 overrun amount of NC optional parameter (uni-directional positioning) or user parameter.

[Programming format]

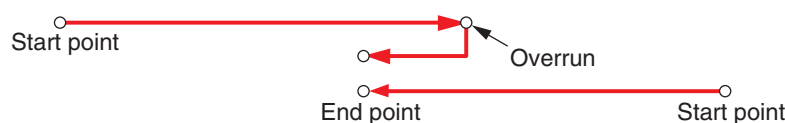
G60 IP__



ME33018R1000500100001

[Details]

- When the linear interpolation mode positioning specification is provided, whether or not the positioning is executed in the linear interpolation pattern is determined by the setting for NC optional parameter (bit) No. 46, bit 1.
- If the direction of the specified positioning agrees with the positioning direction set at positioning direction of NC optional parameter (uni-directional positioning), the axis does not pass over the end point.



ME33018R1000500100002

- G60 is a modal command.
- Uni-directional positioning is not valid for a cycle axis or shift movement in a fixed cycle.
- Uni-directional positioning is not valid on an axis for which no pass-over amount is set.
- Mirror image is not applied to the positioning direction.

8. Linear Interpolation (G01)

[Function]

In the G01 linear interpolation mode, axes move directly from the actual position to the specified target point at the specified feedrate.

[Programming format]

G01 IP__F__

IP: Target point (end point)

F: Feedrate. The specified feedrate remains valid until updated by another value.

[Details]

- A feedrate value specified with address "F" is cleared to zero when the NC is reset. Note that the F command value is saved when the NC is reset if a feedrate is specified in an F1-digit command.
- The feedrate for each axis is as indicated below. (For values X, Y, and Z, convert them into an incremental value.)

G01 XxYyZzFf

Calculation of feedrates:

$$\text{X-axis feedrate: } FX = \frac{x}{L} f$$

$$\text{Y-axis feedrate: } FY = \frac{y}{L} f$$

$$\text{Z-axis feedrate: } FZ = \frac{z}{L} f$$

Where, $L = \sqrt{x^2+y^2+z^2}$

ME33018R1000500110001

For the rotary axis, the unit of feedrate is regarded as indicated below:

1 mm/min = 1 deg/min

1 inch/min = 1 deg/min

In linear interpolation including a rotary axis, the feedrates are determined according to the formulas given above for the individual axes.

Example:

G91 G01 X10 C20 F30.0

<"mm" input>

$$\text{X-axis feedrate } FX = \frac{10}{\sqrt{10^2+20^2}} \times 30 \quad 13.41 \text{ mm/min}$$

$$\text{C-axis feedrate } FC = \frac{20}{\sqrt{10^2+20^2}} \times 30 \quad 26.83 \text{ deg/min}$$

ME33018R1000500110002

<"inch" input>

$$\text{X-axis feedrate FX} = \frac{10}{\sqrt{10^2+20^2}} \times 30 \quad 13.41 \text{ inch/min}$$

$$\text{C-axis feedrate FC} = \frac{20}{\sqrt{10^2+20^2}} \times 30 \quad 26.83 \text{ deg/min}$$

ME33018R1000500110003

- When the F command (F=1) to the rotary axis in the inch system is issued, whether to interpret "F1" as 1 deg/min or as 25.4 deg/min is set by NC optional parameter (bit) No. 15, bit 7.

9. Plane Selection (G17, G18, G19)

[Function]

Selecting a plane is necessary in order to perform the following functions:

- Circular interpolation (Helical cutting)
- Angle command (AG)
- Cutter radius compensation
- Coordinate rotation (Local coordinate system)
- Fixed cycle
- Coordinate calculation
- Area machining

The planes that can be selected are indicated below:

G17 : Xp-Yp plane	$\left\{ \begin{array}{l} \text{Xp ... X- or U-axis.} \\ \text{Yp ... Y- or V-axis.} \\ \text{Zp ... Z- or W-axis.} \end{array} \right.$
G18 : Zp-Xp plane	
G19 : Yp-Zp plane	

ME33018R1000500120001

[Programming format]

G17 Xp__ Yp__

G18 Zp__ Xp__

G19 Yp__ Zp__

ME33018R1000500120002

[Details]

- Whether a basic axis (X, Y, Z) or a parallel axis (U, V, W) is selected is determined by the axis addresses specified in the block containing G17, G18 or G19.

Examples:

```
G17 X _ Y _ XY plane
G17 U _ Y _ UY plane
G18 Z _ X _ ZX plane
G18 W _ X _ WX plane
G19 Y _ Z _ YZ plane
G19 Y _ W _ YW plane
```

ME33018R1000500120003

- In blocks where none of G17, G18, and G19 are specified, the selected plane remains unchanged even if axis addresses are changed.
- In blocks where G17, G18, or G19 is specified, if an axis address is omitted, the basic axis (X, Y, Z) is assumed to be omitted.

Examples:

```
G17      XY plane
G17 X _ XY plane
G17 U _ UY plane
G18      ZX plane
G18 W _ WX plane
```

ME33018R1000500120004

- If a command for an axis that does not exist in the plane selected in the G17, G18, or G19 block is issued, the axis movement is performed regardless of selected plane.
- The plane to be selected when the power is turned ON or the NC is reset can be designated by the setting at THE G CODE TO BE SET AUTOMATICALLY (PLANE) of the NC optional parameter (AUTO SET AT NC RESET/POWER ON).
- An alarm occurs if both the basic axis and its parallel axis are specified in a plane selection block.

10. Circular Interpolation (G02, G03)

[Function]

The circular interpolation function moves a tool from the actual position to the specified position along an arc at the specified feedrate.

[Programming format]

$$\begin{aligned} \text{Arc on } X_p\text{-}Y_p \text{ plane : } G17 & \left\{ \begin{array}{l} G02 \\ G03 \end{array} \right\} X_p_Y_p_ \left\{ \begin{array}{l} R_ \\ I_J_ \end{array} \right\} F_ \\ \text{Arc on } Z_p\text{-}X_p \text{ plane : } G18 & \left\{ \begin{array}{l} G02 \\ G03 \end{array} \right\} Z_p_X_p_ \left\{ \begin{array}{l} R_ \\ K_I_ \end{array} \right\} F_ \\ \text{Arc on } Y_p\text{-}Z_p \text{ plane : } G19 & \left\{ \begin{array}{l} G02 \\ G03 \end{array} \right\} Y_p_Z_p_ \left\{ \begin{array}{l} R_ \\ J_K_ \end{array} \right\} F_ \end{aligned}$$

ME33018R1000500130001

X_p = X-axis or U-axis

Y_p = Y-axis or V-axis

Z_p = Z-axis or W-axis

- G codes used for the circular interpolation function are indicated below.
 - G17 : Plane selection : Sets the circular arc in the X_p - Y_p plane.
 - G18 : Plane selection : Sets the circular arc in the Z_p - X_p plane.
 - G19 : Plane selection : Sets the circular arc in the Y_p - Z_p plane.
 - G02 : Direction of rotation : Sets the clockwise direction.
 - G03 : Direction of rotation : Sets the counterclockwise direction.

Two axes among X_p , Y_p , and Z_p , G90 mode:

Sets the end point in the work coordinate system

Two axes among X_p , Y_p , and Z_p , G91 mode:

Sets the position in reference to the start point with signed values.

Two axes among I, J, and K:

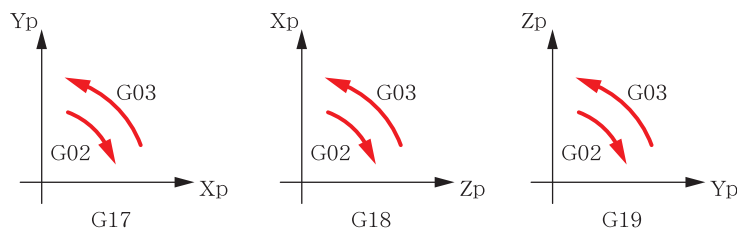
Sets the distance from the start point to the center with signed values.

R: Sets the radius of an arc.

F: Sets the feedrate.

[Details]

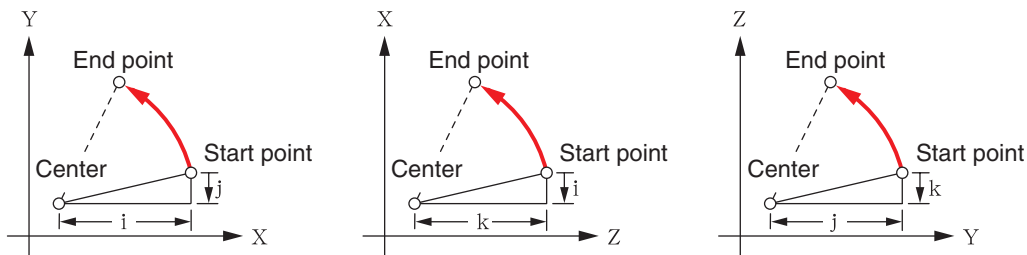
- Direction of rotation, clockwise or counterclockwise, is defined when viewing the plane from the positive direction of the Z_p -axis (Y_p -axis, X_p -axis) on the X_p - Y_p (Z_p - X_p , Y_p - Z_p) plane, as shown in the illustrations below.



ME33018R1000500130002

- The end point is defined by either an absolute value or an incremental value according to G90 or G91.

The center point of an arc is determined by the I, J, and K values which correspond to X_p , Y_p , and Z_p , respectively. Their coordinate values are always specified as incremental values, regardless of G90 or G91.



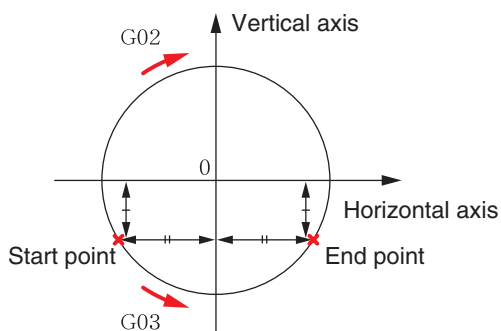
ME33018R1000500130003

A minus sign should be used for the I, J, and K values when necessary.

- The end point of an arc can be designated by specifying the coordinate value on one of the two axes.

If only one axis is specified, the processing may be selected from the following two methods.

- For axes with no command, the previous command value is used as the end point of the arc. (For this processing, set the value for axis not programmed (circular single-axis) of NC optional parameter (circular interpolation) as the current value.)



ex : X -70.711, Y -70.711

G02 X70.711 I70.711 J70.711

X -70.711, Y -70.711

G02 X10 I70.711 J70.711

When programming an arc as illustrated to the left, the end point of the arc can be designated with only the coordinate value of the horizontal axis, since the coordinate value of the vertical axis is the same at the start and end points. An alarm occurs if the end point does not lie on an arc.

The left program defines a clockwise arc:

Radius: 100

Center: (0, 0)

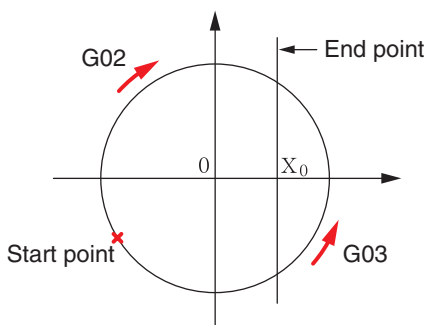
Start point: (-70.711, -70.711)

End point: (70.711, -70.711)

The left program will cause an alarm, since the end point (10, -70.711) is not on the arc.

ME33018R1000500130004

- For the omitted axis, the coordinate value is calculated using the coordinate value of the specified axis. For this processing, choose point on arc at command value for the axis not programmed (single-axis) of NC optional parameter (circular interpolation).



When programming an arc as illustrated in the left, the end point can be designated with only the horizontal axis coordinate value. The vertical axis coordinate value is calculated from the horizontal axis coordinate value.

ME33018R1000500130005

- If more than one end point is possible, the one which is reached first in the designated arc direction is selected.

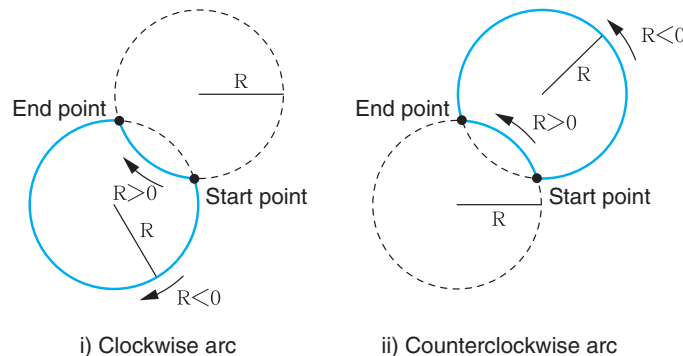
Example:

X-70.711 Y-70.711 F200 The program to the left defines a clockwise arc:
 G02 X10 I70.711 J70.711 Radius: 100
 Center: (0, 0)
 Start point: (-70.711, -70.711)
 End point: (10,99.499)

ME33018R1000500130006

The operations explained above also apply when designation of a vertical axis is omitted.

- The center of an arc can be defined by specifying the radius (R) of the arc instead of specifying I, J, and K. If an arc is specified by the radius, four arcs that pass the same start and end points are defined. To define a specific arc from among these four arcs, an R value is used in the manner indicated below.
 - Clockwise arc (G02)
 - An arc whose central angle is smaller than or equal to 180 degrees: Radius $R > 0$
 - An arc whose central angle is greater than 180 degrees: Radius $R < 0$
 - Counterclockwise arc (G03)
 - An arc whose central angle is smaller than or equal to 180 degrees: Radius $R > 0$
 - An arc whose central angle is greater than 180 degrees: Radius $R < 0$



ME33018R1000500130007

- The feedrate in circular interpolation is the feedrate component tangential to the arc.

[Supplement]

- If I, J, or K is omitted, it is regarded that "0" is specified.
- An arc with radius 0 ($R = 0$) cannot be specified.
- If the values for X_p , Y_p , and Z_p are omitted, an arc having the start and end points on the same point is defined in the following manner:
 - If the center is specified by I, J, and/or K, a 360-degree arc
 - If the radius is specified by R, a 0-degree arc
- It is not possible to specify R, and I, J, and K at the same time.
- It is not possible to specify any axis parallel to the axes which make up the selected plane. For example, designation of the W-axis is not allowed when the Z-X plane is selected.
- An alarm will occur if the difference in radius between the start point and the end point of an arc is greater than or equal to the value set at arc check data (difference in radius between start and end) of the NC optional parameter (circular interpolation).

11. Helical Cutting (G02, G03) (Optional)

[Function]

Helical cutting or helical interpolation may be executed by synchronizing circular interpolation with linear interpolation of the axis which intersects at right angles the plane in which the arc is defined.

[Programming format]

XpYp plane : G17 $\left\{ \begin{array}{l} G02 \\ G03 \end{array} \right\}$ Xp__Yp__ $\left\{ \begin{array}{l} R_ \\ I_J_ \end{array} \right\}$ α __F__

α : An axis not parallel to the axes comprising the arc plane

ME33018R1000500140001

[Details]

- Helical cutting may also be programmed on the Zp-Xp (G18) and Yp-Zp (G19) planes, using a format similar to that above.
- To program helical cutting, simply add the command of the axis which intersects the arc plane to the circular interpolation.
- Helical cutting is possible for an arc having a center angle of smaller than 360 degrees.
- The feedrate specified by an F command is valid for circular interpolation. Therefore, the feedrate in the direction of the linear axis is calculated by the following formula:

$$\text{Feedrate in the linear axis direction} = \frac{\text{Motion distance of the linear axis}}{\text{Arc length}} \times F$$

ME33018R1000500140002

- Tool length offset is valid for the axis at right angles to the arc plane.
- Cutter radius compensation is valid only for circular interpolation commands.

SECTION 4 PREPARATORY FUNCTIONS

G codes consisting of address character G and a three-digit number (00 to 399) set the mode that specifies how the commands are executed.

Instead of using address character G, some G codes are expressed by mnemonics. A mnemonic code consists of up to eight alphabetic characters (A to Z).

- Valid range of G codes
 - One-shot G codes:
Valid only in the specified block. Such G codes are automatically canceled when a program advances to the next block.
 - Modal G codes:
Once specified, such G codes remain valid until another G code in the same group is specified.
- Special G codes
 - Mnemonic codes used for subprogram call and those used as branch instructions are called special G codes.
 - Special G codes must be specified at the beginning of a block and entry of such codes at a middle of a block is not allowed. Note, however, that a slash "/" code (optional block skip code) or a sequence name may be placed before a special G code.
- For the tables of G codes and mnemonic codes, refer to "G Code Table" and "Table of Mnemonic Codes" in APPENDIX.

1. Dwell Command (G04)

[Function]

At the end of the specified block, the dwell function suspends the execution of a program for the specified length of time before proceeding to the next block.

[Programming format]

The following two programming formats may be used to specify the dwell function.

- G04 F__
 - F: Sets the length of dwell time
 - The unit of dwell time can be selected from 1, 0.1, 0.01 and 0.001 seconds by the NC optional parameter (INPUT UNIT SYSTEM).
 - The maximum programmable dwell time is 99999.999 seconds.
- G04 P__
 - P: Sets the length of dwell time
 - The unit of dwell time is selected in the same manner as when specified by F.

2. Programmable Mirror Image (G62) (Optional)

[Function]

The mirror image function creates a geometry which is symmetric around a specific axis. In addition to the mirror image switch on the machine panel, the programmable image function creates mirror images using programmed commands.

The axis which is in the mirror image mode is identified on the screen display; a dash "-" is added before the axis name on the ACTUAL POSITION screen.

[Programming format]

G62 IP $\left. \begin{matrix} 0 \\ 1 \end{matrix} \right\}$
 0 : Normal (Mirror image OFF)
 1 : Mirror image

ME33018R1000600030001

[Details]

- The actual state of the mirror image function based on the specification of G62 and the MIRROR IMAGE switch setting is displayed in the table below.

G62	Switch Setting	Actual State
Normal	Normal	Normal
Normal	Mirror image	Mirror image
Mirror image	Normal	Mirror image
Mirror image	Mirror image	Normal

ME33018R1000600030002

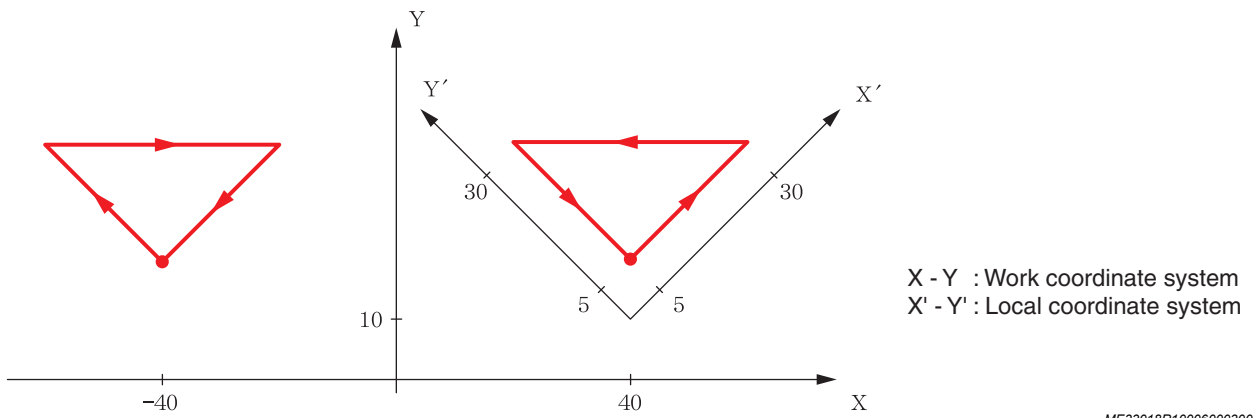
- A block in which G62 is specified must not contain any other commands.
- The mirror image function is modal.
- The axes not specified in the G62 block are assumed to be in the normal mode.
- All axes are in the normal mode when the power supply is turned on.
- Whether all axes will be set in the normal mode or not when the NC is reset can be set at AT AN NC RESET, CLEARS THE G62 MIRROR IMAGE FOR ALL AXES of NC optional parameter (MIRROR IMAGE).
- The coordinate system (local or work) in which the mirror image function will be active can be selected at local/work coordinate system select of NC optional parameter (MIRROR IMAGE).

Example:

```
G11 X40 Y10 P45
G01 X5 Y5 S__F__
      X30
      Y5 Y30
      Y5
G62 X1
```

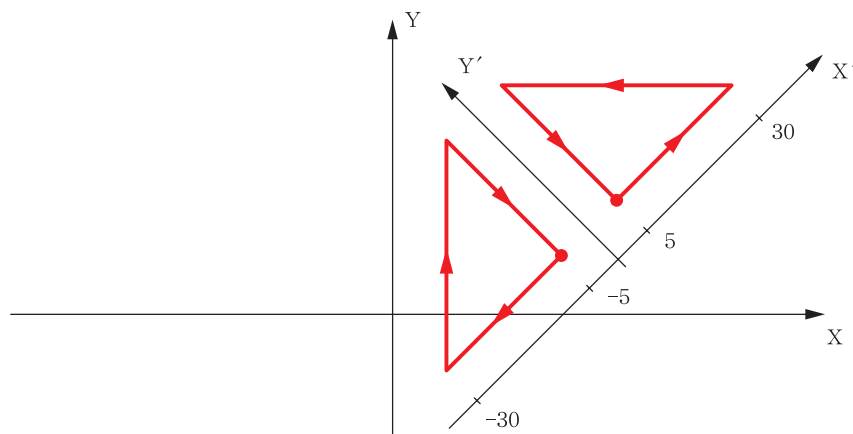
ME33018R1000600030003

- (1) If work is selected at local/work coordinate system select of NC optional parameter (MIRROR IMAGE)



ME33018R1000600030004

- (2) If local is selected at local/work coordinate system of NC optional parameter (MIRROR IMAGE)



ME33018R1000600030005

3. Work Coordinate System Selection (G15, G16)

[Function]

20 sets of work coordinate systems are supplied as a standard feature and this can be expanded to 50, 100 or 200 sets optionally.

[Programming format]

Modal G code: G15 Hn ($0 \leq n \leq 200$)

Once a new work coordinate system "n" is set using the modal G code, the coordinate values specified in the same and later blocks are interpreted as coordinate values in the selected work coordinate system "n".

One-shot G code: G16 Hn ($0 \leq n \leq 200$)

If a new work coordinate system "n" is set using the one-shot G code, only the coordinate values specified in the same block are interpreted as coordinate values in the selected work coordinate system "n".

[Details]

- For G15 and G16, the work coordinate system number between 1 and 200 is specified by "n" (1 to 200). If "0" is specified for "n", the machine coordinate system is selected.
- When the power supply is turned ON, and after the NC is reset, the work coordinate system previously selected by G15 is automatically selected.
- G15 and G16 may not be specified in the following modes:
 - Cutter radius compensation mode
 - Three-dimensional offset mode
 - Geometry enlargement/reduction mode
 - Coordinate system parallel shift/rotation mode

[Supplement]

Axis feed commands specified immediately after G15 must be specified in the absolute mode.
--

4. Work Coordinate System Change (G92)

[Function]

The work coordinate system change function changes the work coordinate system.

[Programming format]

G92 IP__

[Details]

- G92 automatically changes the work zero offset value of the presently selected work coordinate system so that the coordinate value of the present tool position will be the coordinate value specified as IP__.
- G92 changes only the work coordinate system that is selected at the time it is executed; it does not affect any other work coordinate systems.
- The coordinate value P specified in this block is always treated as an absolute value regardless of the specification of G90 (absolute mode) and G91 (incremental mode).
- For the axis not specified with the coordinate value P, the work zero offset value remains unchanged.
- G92 may not be specified in the following modes:
 - Cutter radius compensation mode
 - Three-dimensional offset mode
 - Geometry enlargement/reduction mode
 - Coordinate system parallel shift/rotation mode
 - Machine coordinate system selected mode

5. Unit System Check (G20, G21) (Optional)

[Function]

The unit system check function checks the unit system selected by the setting at LENGTH UNIT SYSTEM of NC optional parameter (INPUT UNIT SYSTEM). If the selected system does not agree with the unit system specified by G20 / G21, an alarm occurs.

[Programming format]

G20:Checking for the selection of the inch system

An alarm occurs if the metric system is selected by the setting for the parameter.

G21:Checking for the selection of the metric system

An alarm occurs if the inch system is selected by the setting for the parameter.

6. Coordinate System Conversion Functions

6-1. Parallel Shift and Rotation of Coordinate Systems (G11, G10)

[Function]

The parallel shift / rotation function shifts or rotates a work coordinate system. The new coordinate system defined by shifting or rotating a work coordinate system is called a local coordinate system. It is possible to cancel a local coordinate system.

[Programming format]

Parallel shift / rotation of coordinate system: G11 IP__ P__

- IP: Parallel shift amount to establish a local coordinate system
Specify the shift amount as an absolute value in reference to the origin of the work coordinate system, regardless of the selected dimensioning mode, absolute mode (G90), incremental mode (G91), or mirror image (G62).
- P: Rotation amount to establish a local coordinate system
Specify the angle of rotation in units of 1 degree, 0.001 degree, or 0.0001 degree in accordance with the selected unit system (LENGTH UNIT SYSTEM and ANGLE of NC optional parameter (INPUT UNIT SYSTEM)).
If "P0" is specified or a P command is not specified, only work coordinate system shift takes place, without rotation.
Rotation of a work coordinate system is executed in the plane (G17, G18, G19) that is active when G11 is specified, and it does not affect the axes not included in this plane. The direction of rotation is counterclockwise viewed from the positive direction of the axis not included in the rotation plane.
Specify the angle of rotation as an absolute value, regardless of the selected dimensioning mode (G90, G91).

- Cancellation of local coordinate system: G10
When G10 is specified, the parallel shift amount and angle of rotation are canceled.

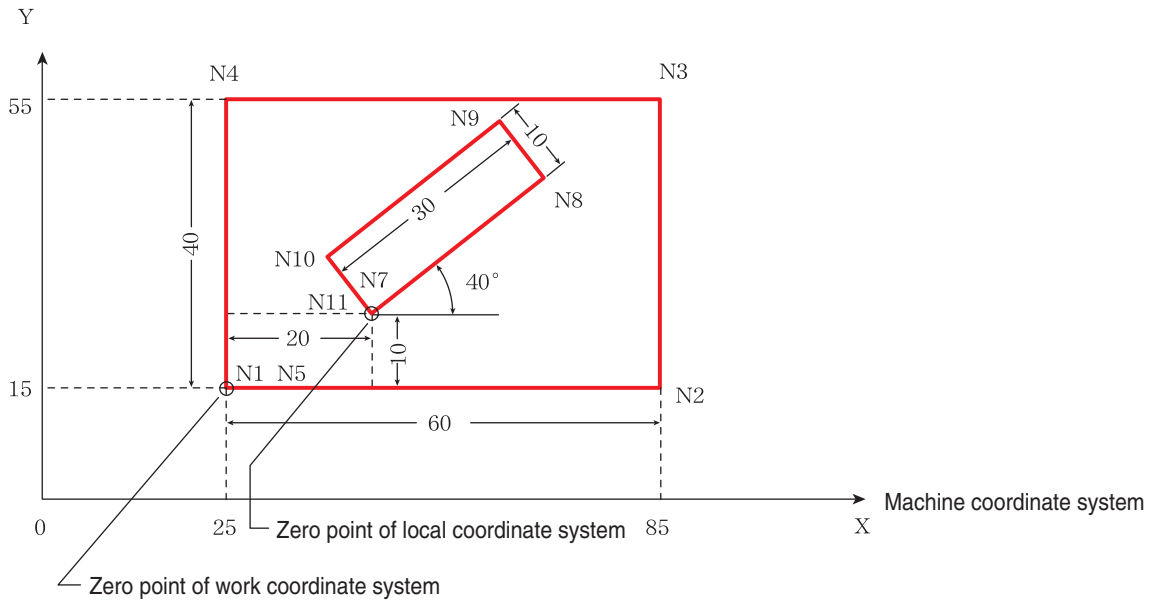
[Details]

- Once G11 is executed, the NC enters the state in which a local coordinate system is defined. If G11 is executed again in this state, it will change the previously defined local coordinate system. At the second designation of G11, if the designation of an axis address is omitted, the value designated in the first G11 is assumed to apply. The set values are cleared when the power supply is turned OFF / ON, the NC is reset, or G10 (local coordinate system cancel) is executed.
- A block which contains G10 or G11 must not contain any other G codes.
- G10 and G11 are modal. G10 is set when the power is turned ON or when the NC is reset.
- G11 must not be specified in the following modes:
 - Geometry enlargement/reduction mode
 - When the machine coordinate system (H00) is selected
 - Copy function mode

SECTION 4 PREPARATORY FUNCTIONS

[Example program]

If a local coordinate system is used, the example workpiece shown below would be programmed as indicated in the example program.



```

N1 G15 G90 G00 X0 Y0 H01..... Selecting work coordinate system 1
N2 G01 X60 F100
N3 Y40
N4 X0
N5 Y0
N6 G17 G11 X20 Y10 P40 ..... Setting a local coordinate system
N7 X0 Y0
N8 X30
N9 Y10
N10 X0
N11 Y0
N12 G10 ..... Canceling a local coordinate system
    
```

* : The zero offset values of work coordinate system 1 are : x= 25, y = 15

6-2. Copy Function (COPY, COPYE)

[Function]

The copy function is used to facilitate part machining by repeating the same pattern with parallel shift and rotation.

First, specify parallel shift and rotation of a local coordinate system using COPY instead of G11, then program the pattern to be repeated. Finally, specify the incremental value of parallel shift / rotation.

[Programming format]

Parallel shift/rotation of local coordinate system: COPY IP__ P__ Q__

- IP: Initial value of parallel shift component to establish a local coordinate system
Specify this as an absolute value in reference to the origin of the presently selected work coordinate system.
- P: Initial value of rotation component to establish a local coordinate system
Specify this value in units of 1 degree, 0.001 degree, or 0.0001 degree in accordance with the selected unit system ("LENGTH UNIT SYSTEM" and "ANGLE" of NC optional parameter (INPUT UNIT SYSTEM)).
If a P command is not specified in the G11 mode, the previous setting is valid.
- Q: The number of times the pattern should be repeated.
Setting range: 1 to 9999
Default value: 1

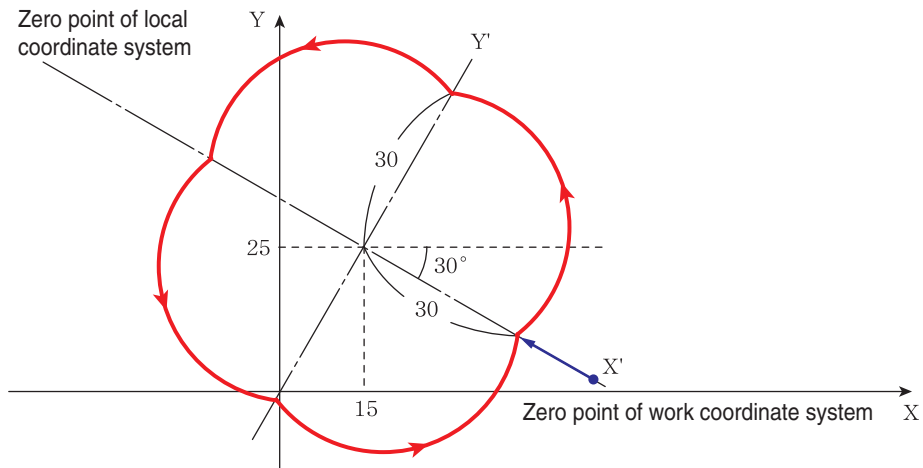
Repeated pattern program: COPYE IP__ P__

- IP: Incremental value for parallel shift of a local coordinate system.
Default value: 0
- P: Incremental value for rotation of a local coordinate system.
Default value: 0

[Details]

- Both G11 and COPY may be specified while a local coordinate system is established by the execution of G11. Once COPY is specified, however, an alarm occurs if G11 or COPY is specified again.
- If COPY is specified in the main program selected for operation method B (large-volume tape operation), designation of IF and GOTO is not permissible in the program that defines the pattern to be repeated.
The size of the program beginning with COPY and ending with COPYE must be within 10 m (33 ft) in tape length.

[Example program]



```

G11   X15  Y25  P - 30
G01   X30  F100 M03
COPY  Q4
G01   X30  Y0 ..... Start point of arc
G03   X0   Y30I - 20J10
G01   X0   Y30 ..... End point of arc
COPYE P90

```

* : Circular interpolation commands must not be specified in the block immediately after the COPY block and the one immediately before the COPYE block.

7. Workpiece Geometry Enlargement / Reduction Function (G51, G50) (Optional)

[Function]

The workpiece geometry enlargement / reduction function enlarges or reduces the geometry defined by a program in reference to the point specified in a local coordinate system. If a local coordinate system is not specified, a work coordinate system is used to specify the reference point for enlargement / reduction.

[Programming format]

Enlargement / reduction of geometry: G51 IP __ P__

IP: The center of the enlargement / reduction of geometry.

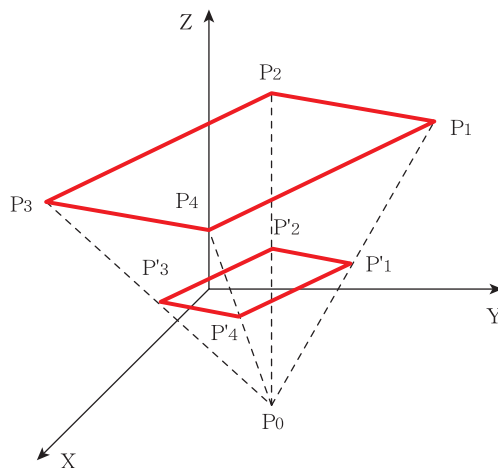
Specify this point in a local coordinate system.

For axes not specified in this block, the coordinate value (in the local coordinate system) of the point at which G51 is specified is assumed to apply.

P: Multiplication factor for enlargement or reduction.

Programmable range: 0.000001 to 99.999999

Default value: 1



P0 : Center of enlargement / reduction

P1 - P4 : Programmed geometry

P'1 - P'4 : Reduced geometry

Cancellation of enlargement / reduction : G50

ME33018R1000600090001

[Details]

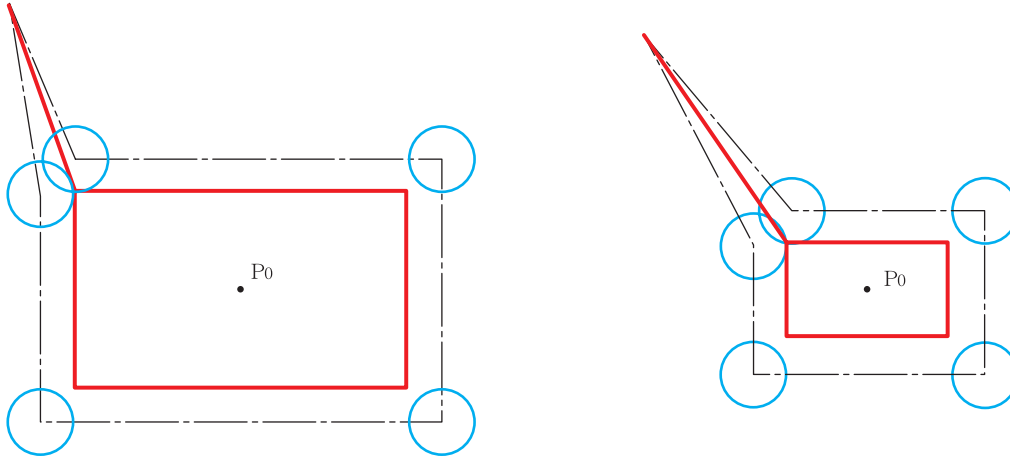
- The enlargement/reduction function is made valid or invalid on individual axes according to the setting for NC optional parameter (geometry enlargement/reduction). However, an alarm will occur in the radius check if the parameter setting differs among the axes in the plane specified for circular interpolation.
- The enlargement/reduction function does not affect the following:
 - a. Local coordinate system setting values (G11)
 - b. Cutter radius compensation values and three-dimensional offset values (G41, G42, G43)
 - c. Tool length offset values (G54 to G59)
 - d. Work coordinate system setting values (G92)

e. The following Z-axis movements in a fixed cycle:

- In-feed and retraction amounts in deep hole drilling cycle (G73, G83)
- X, Y shift amounts in fine boring or back boring (G76, G87)

Example:

Cutter radius compensation and enlargement and reduction of workpiece geometry



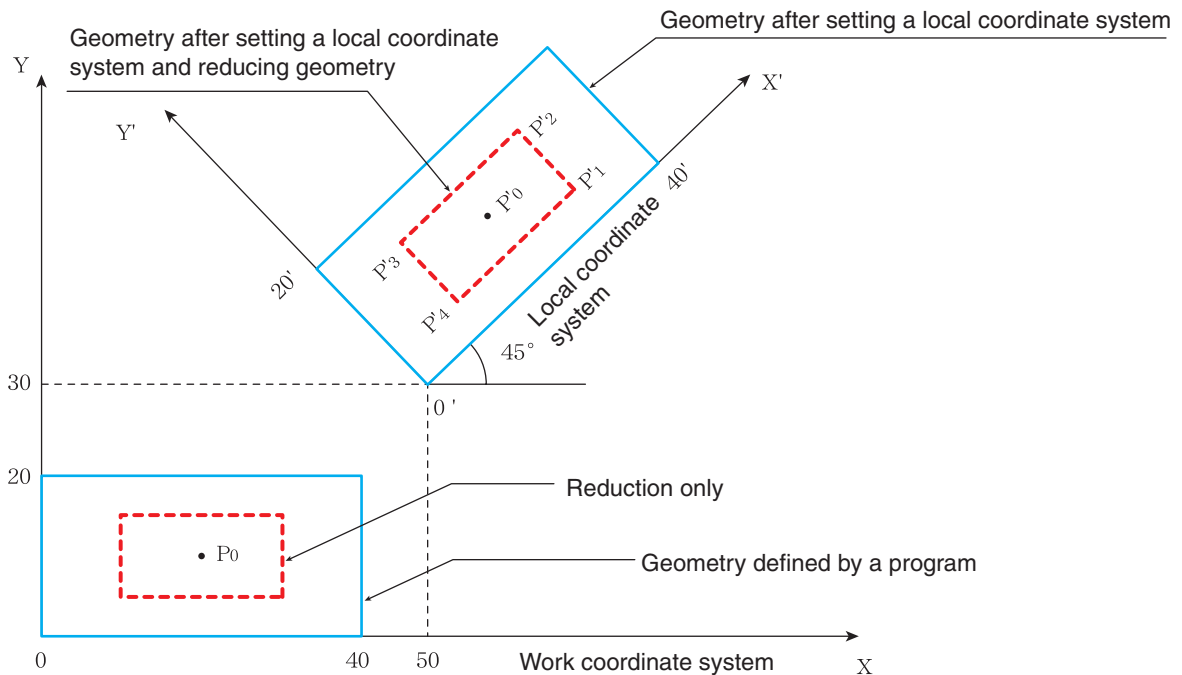
ME33018R1000600090002

[Example program]

An example program for setting a local coordinate system and enlarging/reducing workpiece geometry is shown below.

```

N1 G17 G11 X50 Y30 P45 Setting of local coordinate system
N2 G90 G51 X20 Y10 P0.5 Reduction of geometry
N3 G01 X40 Positioning at P'1
N4 Y20 Positioning at P'2
N5 X0 Positioning at P'3
N6 Y0 Positioning at P'4
    
```



ME33018R1000600090003

SECTION 5 S, T, AND M FUNCTIONS

This section describes the S, T, and M codes which specify necessary machine operations other than axis movement commands.

S: Spindle speed

T: Tool number for tool change cycle

M: Turning solenoids and other similar devices on and off

Only one of each of these types of code may be specified in one block.

If two or more commands of the same code type are issued to one block, the rightmost command of each code type will be executed.

<Example> M6 T1 T2-----M6 T2 will be executed.

1. S Code Function

[Function]

The spindle function specifies a spindle speed with a numeric value (up to five digits) entered following address S.

[Details]

- The desired spindle speed (min^{-1}) is directly specified by a numeric value following the address S.
Programmable range: 0 to 65535
- If an S command is specified with axis movement commands in the same block, the S command becomes valid at the same time axis movement commands are executed.
- Although an S command is not canceled when the NC is reset, it is cleared when the power supply is turned off.
- To execute a spindle rotation command (M03, M04), an S command must be specified in the same or a previous block.

2. T Code Function

[Function]

The tool function selects a tool in the machine with a numeric value (up to four digits) entered following address T.

[Details]

- The programmable range of a T command is indicated below.
Programmable range: 0 to 9999
- When a T code is executed, the next tool is prepared (indexing the next tool in the magazine, or taking the next tool out of the magazine and setting it in the ready station position).
- The actual tool change cycle is executed by M06.
- If a T command is specified with axis movement commands in the same block, the execution timing of the T code can be selected from the following two timings:
Executed simultaneously with axis movement commands
Executed after the completion of axis movement commands

3. M Code Function

[Function]

The M code function outputs an M code number, consisting of a three-digit number and address M, and the strobe to the PLC. The programmable range of M codes is from 0 to 511.

3-1. Examples of M Codes

The followings are examples of M codes.

(1) M02, M30 (End of Program)

These M codes indicate the end of a program.

When M02 or M30 is executed, the main program ends and reset processing is executed. The program is rewound to its start. (In the case of a schedule program, execution of M02 or M30 in the main program does not reset the NC.)

(2) M03, M04, M05 (Spindle CW/CCW and Stop)

These M codes control spindle rotation and stop; spindle CW (M03), spindle CCW (M04), and spindle stop (M05).

(3) M19 (Spindle Orientation)

The M19 command is used with machines equipped with the spindle orientation mechanism.

The spindle orientation function stops the spindle at a specified angular position.

- Multi-point spindle indexing

By specifying "RS=angle" following M19, it is possible to index the spindle at the specified angular position.

Although the following explanation uses M19 as an example, the same applies to M118 and M119.

M19 RS = θ

- θ represents the desired index angle and it is specified in units of 1° . If a value smaller than 1° is specified, it is truncated.
- Programmable range of θ : 0 to 360°
- θ specifies the desired index angle of the spindle, measured in the CW rotation angle in reference to the spindle orientation position.

[Supplement]

- If M19 (M118, M119) is specified without argument RS, ordinary spindle orientation is performed. That is, the called operation is the same as that called by "M19 RS=0".
- RS must always be specified in the same block as M19 (M118, M119).

(4) M52 (Fixed Cycle - Return to the Retract End)

In various fixed cycles, this command sets the return position of the cycle axis 0.1 mm away from the travel limit of the Z-axis in the positive direction.

For details, refer to SECTION 7, "Fixed Cycle Operations".

(5) M53 (Fixed Cycle - Return to the Specified Point)

In various fixed cycles, this command sets the return position of the cycle axis at the position specified by G71.

For details, refer to SECTION 7, "Fixed Cycle Operations".

- (6) M54 (Fixed Cycle - Return to Point R Level)
In various fixed cycles, this command sets the return position of the cycle axis at the position specified by R command.
For details, refer to SECTION 7, "Fixed Cycle Operations".
- (7) M132, M133 (Single Block Valid/Invalid)
These M codes set whether the single block function is made invalid (M132) or valid (M133) independently of the setting of the single block switch on the machine operation panel.
- (8) M201 to M210 (M Code Macro)
By setting the program names which correspond to M201 to M210 in the parameters, the sub programs can be executed by specifying the M codes.
For details of M code macro, refer to SECTION 10, "G and M Code Macro Functions".
- (9) M238, M239 (Soft-override Valid/Invalid)
These commands set whether or not the soft-override value (%) set for system variables <VFSOV> is valid (M238) or invalid (M239) for the cutting feedrate (F command × override value).
- (10) M00 (Program Stop)
After the execution of M00, the program stops. If the NC is started in this program stop state, the program restarts.
- (11) M01 (Optional Stop)
When M01 is executed while the optional stop switch on the machine operation panel is ON, the program stops. If the NC is started in this optional stop state, the program restarts.
- (12) M06 (Tool Change)
This M code is used with machines equipped with the tool change mechanism as the tool change cycle start command.
- (13) M15, M16 (Fourth Axis - Rotary Table CW, CCW)
These M codes are used with machines equipped with the rotary table as the fourth axis to specify the direction of rotary table rotation; CW (M15), CCW (M16).
For details of the rotary table control, refer to "Additional Axis (Rotary Axis) Function" is SPECIAL FUNCTIONS Manual No.2.
- (14) M115, M116 (Fifth Axis - Rotary Table CW, CCW)
These M codes are used with machines equipped with the rotary table as the fifth axis to specify the direction of rotary table rotation; CW (M115), CCW (M116)
For details of rotary table control, refer to "Additional Axis (Rotary Axis) Function" is SPECIAL FUNCTIONS Manual No.2.
- (15) M118, M119 (Spindle Index - CCW, Shorter Path)
These M codes are used with machines equipped with the spindle index mechanism as the spindle orientation direction specifying command.
[Programming format]
- M118 Spindle index (CCW)
 - M119 Spindle index (shorter path)
- (16) M130, M131 (For Cutting Feed, Spindle Rotation Condition Valid / Invalid)
Usually, in the G01, G02, and G03 modes, the spindle must be rotating to execute axis feed. These M codes are set to ignore this condition (M130) or validate it (M131).

- (17) M134, M135 (Spindle Speed Override Valid / Invalid)
Even in the status in which spindle speed override control from the PLC is valid, the spindle speed override function can be made invalid (M134) or valid (M135) with these commands.
- (18) M136, M137 (Axis Feed Override Valid / Invalid)
These M codes set whether the axis feed override function is made invalid (M136) or valid (M137) independently of the ON status of the axis feed override signal from the PLC.
- (19) M138, M139 (Dry Run Valid / Invalid)
These M codes set whether the dry run function is made invalid (M138) or valid (M139) independently of the setting of the dry run switch on the machine operation panel.
- (20) M140, M141 (Slide Hold Valid / Invalid)
These M codes set whether the slide hold function is made invalid (M140) or valid (M141) independently of the setting of the slide hold switch on the machine operation panel.
- (21) M234 to M237 (Gear Selection Range for Synchronized Tapping)
These M codes set the gear selection range for synchronized tapping.
For details, refer to "Torque Monitoring Function" in Synchronized Tapping of SPECIAL FUNCTIONS Manual.
- (22) M326, M327 (Torque Monitor ON / OFF for Synchronized Tapping)
These M codes turn ON (M326) and OFF (M327) the torque monitor mode for synchronized tapping.
- (23) M331, M332 (Sixth Axis - Rotary Table CW / CCW)
These M codes are used with machines equipped with the rotary table as the sixth axis to specify the direction of rotary table rotation; CW (M331), CCW (M332)
For details of rotary table control, refer to "Additional Axis (Rotary Axis) Function" in SPECIAL FUNCTIONS Manual No.2.
- (24) M396 to M399 (Gear Position Selection for Synchronized Tapping)
These are gear position commands, specially for synchronized tapping, introduced by the gear selection range specifying M codes (M234 to M237) and the S command. They are automatically generated by the NC.
- M396: 1st gear command for synchronized tapping
 - M397: 2nd gear command for synchronized tapping
 - M398: 3rd gear command for synchronized tapping
 - M399: 4th gear command for synchronized tapping

SECTION 6 OFFSET FUNCTIONS

1. Tool Length Offset Function (G53 - G59)

[Function]

The tool length offset function compensates for the position of a cutting tool so that the tip of the cutting tool is located at the programmed position.

Available G Codes

G Code	Function
G53	Cancel tool length offset
G54	Tool length offset, X-axis
G55	Tool length offset, Y-axis
G56	Tool length offset, Z-axis
G57	Tool length offset, 4th-axis
G58	Tool length offset, 5th-axis
G59	Tool length offset, 6th-axis

[Programming format]

{G54 - G59} IP__ H__

IP: Current position of tool tip after compensation

H: Tool offset number

The standard tool offset numbers are H00 to H100, and this can be expanded to H200 or H300.

The offset amount of H00 is always zero.

Offset data is set in the tool data setting mode.

Setting range: 0 to ± 999.999 mm (0 to ± 39.3700 inches)

[Details]

- The displayed actual tool position value always includes the tool length offset amount.
- The tool length offset cannot be applied to two or more axes at the same time or to the rotary axis.
- The tool length offset may be changed directly without having to cancel the previous command with G53.
- When the NC is reset, H00 is automatically set.

2. Cutter Radius Compensation (G40, G41, G42)

2-1. Cutter Radius Compensation Function

[Function]

The cutter radius compensation function automatically compensates for the cutter radius.

Programming the geometry of a workpiece as it is will not result in a correct final product because the size (diameter) of the tool is not taken into consideration. It would, however, be extremely complicated and difficult to develop a program which takes the tool diameter into account. This problem may be solved by a function called cutter radius compensation which automatically compensates for the tool diameter. If the cutter radius compensation function is used for programming, the correctly offset tool center path is automatically generated by programming the tool path along the geometry of workpiece to be machined.

[Programming format]

```
G17   G41   (G42)  Xp__ Yp__ D__
G18   G41   (G42)  Zp__ Xp__ D__
G19   G41   (G42)  Yp__ Zp__ D__
```

G40: Cancel cutter radius compensation (The mode automatically selected when the power is turned ON.)

For details, refer to "Tool Movement when Cutter Radius Compensation is Canceled".

G41: Cutting at left (Offset - the left side as seen from the tool moving direction; downward cutting)

For details, refer to "Changing Compensation Direction in Cutter Radius Compensation Mode".

G42: Cutting at right (Offset to the right side as seen from the direction of tool motion; upward cutting)

The cutter radius compensation mode is set when either G41 or G42 is specified and this mode is canceled by G40. For details, refer to "Changing Compensation Direction in Cutter Radius Compensation Mode".

G17: Xp-Yp plane selection Select the plane in the same manner as in the G02 or G03 mode.

G18: Zp-Xp plane selection Select the plane in the same manner as in the G02 or G03 mode.

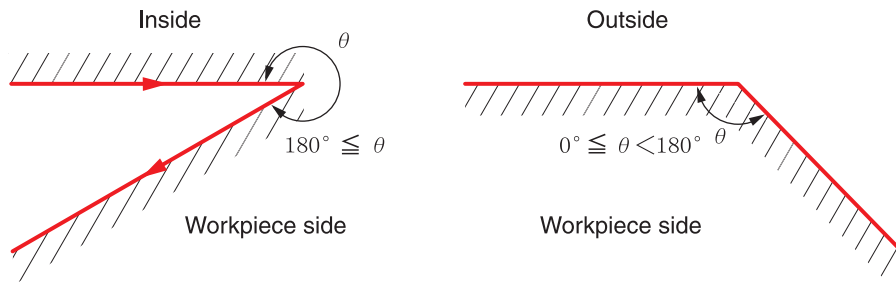
G19: Yp-Zp plane selection Select the plane in the same manner as in the G02 or G03 mode.

D**: Cutter radius compensation number. (For details, refer to "Notes on Cutter Radius Compensation".)

[Supplement]

- The explanation below assumes G17 (Xp-Yp plane), which is automatically set when power is turned ON. For the Zp-Yp plane and the Yp-Zp plane, the same explanation applies.
- Entry to the cutter radius compensation mode is allowed only in the G00 or G01 mode. An alarm occurs if the cutter radius compensation mode is called in other modes.
- The mode is changed to the cutter radius compensation mode in the first block that contains a command that actually causes axis movement after the designation of the cutter radius compensation command.

- The terms “inside” and “outside” are defined as follows:
The angle made between consecutive tool paths is measured at the workpiece side and “inside” and “outside” are defined by the magnitude of this angle. If the angle is larger than 180° , it is defined as “inside” and if the angle is in the range between 0 and 180° , it is defined as “outside”.



ME33018R1000800020001

- The symbols used in the illustrations in “Tool Movement in Start-up” to “Notes on Cutter Radius Compensation” have the following meaning:

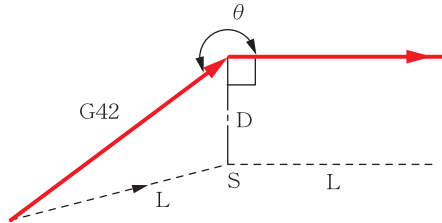
S : Single block stop point
 L : Linear motion
 C : Circular motion
 T : Tangent to an arc
 D : Cutter radius compensation amount
 θ : Angle at the workpiece side
 CP : Cross point, made when a programmed path (or the tangent to an arc) is shifted by a compensation amount
 \rightarrow : Programmed tool path
 $- \rightarrow$: Tool center path
 $---$: Auxiliary line

ME33018R1000800020002

2-2. Tool Movement in Start-up

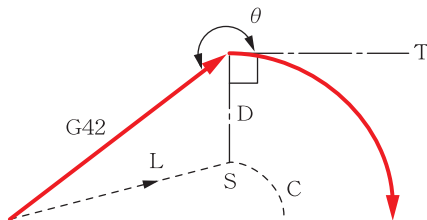
2-2-1. Inside Corner Cutting ($\theta \geq 180^\circ$)

(1) Straight line - Straight line



ME33018R1000800030001

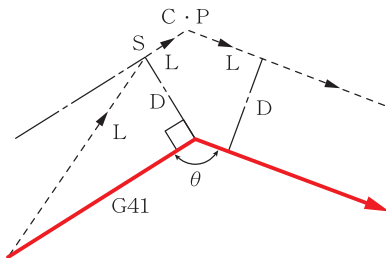
(2) Straight line - Arc



ME33018R1000800030002

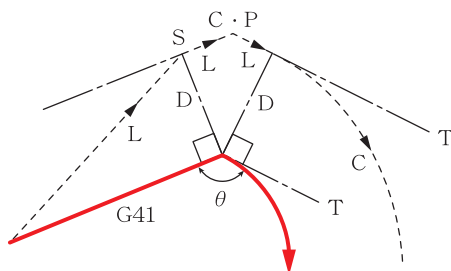
2-2-2. Obtusely Angled Corner - Outside Cutting ($90^\circ \leq \theta \leq 180^\circ$)

(1) Straight line - Straight line



ME33018R1000800040001

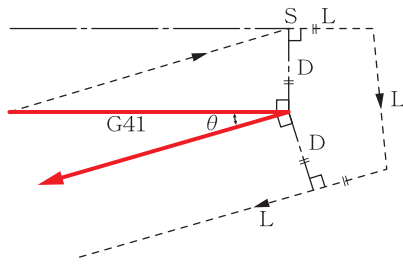
(2) Straight line - Arc



ME33018R1000800040002

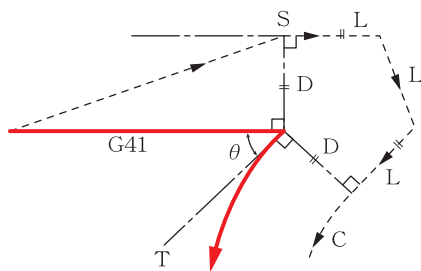
2-2-3. Acutely Angled Corner - Outside Cutting ($\theta < 90^\circ$)

- (1) Straight line - Straight line



ME33018R1000800050001

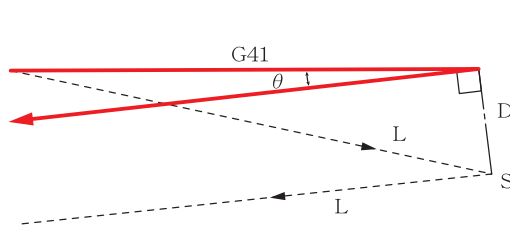
- (2) Straight line - Arc



ME33018R1000800050002

- (3) Exception

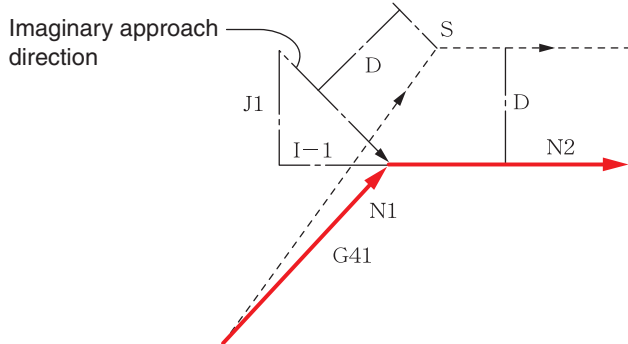
Outside cutting at an acute angle of 1° or less is considered to be "inside" as shown below.



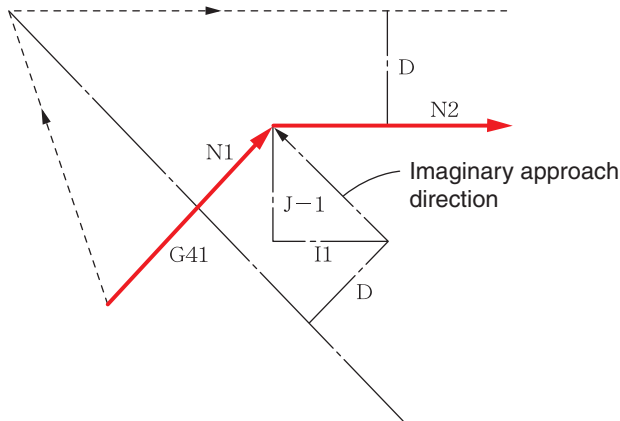
ME33018R1000800050003

2-2-4. Start-up with Imaginary Approach Direction

If the block which starts up the cutter radius compensation includes any I__, J__, or K__ belonging to the offset plane (I__, J__ in the case of G17 plane), the axes move to the target point specified in this block from the direction defined by I__ and/or J__. In this case, note that the cross point is always calculated regardless of whether the cutting is “inside” or “outside.”



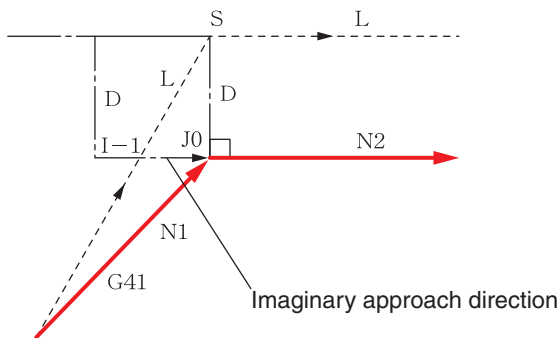
```
N1 G41 X5000 Y5000 I-1J1D1
N2 X100000
```



```
N1 G41 X5000 Y5000 I1J-1D1
N2 X100000
```

ME33018R1000800060001

If no cross point exists, positioning is executed to the point obtained by a vertical shift by the compensation amount from the target point specified in the G41 block.



```
N1 G41 X5000 Y5000 I-1J0
N2 X100000
```

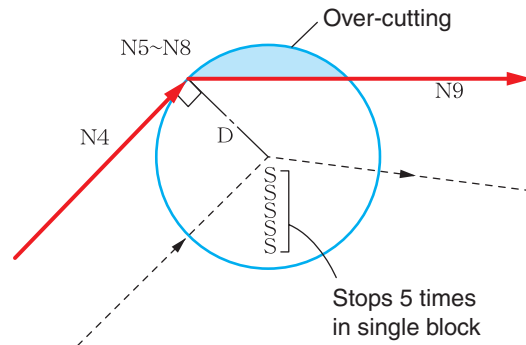
ME33018R1000800060002

2-3. Tool Movement in Cutter Radius Compensation Mode

[Supplement]

This section describes operations from the operation that begins after entering in the tool offset mode until just before canceling the cutter radius compensation mode.

Example: Consecutive 4 blocks (zero movements of the axes in the selected plane)



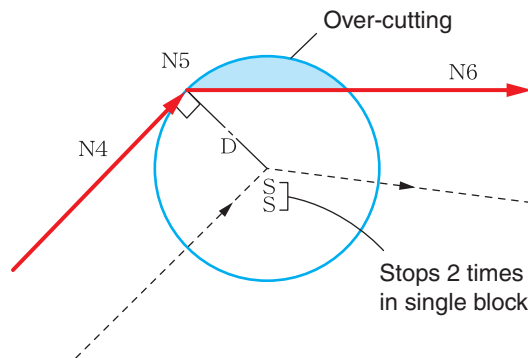
```

:
N4 X5000 Y5000
N5 Z5000
N6 F1000
N7 M01
N8 G04 F50
N9 X100000
:

```

ME33018R1000800070001

Example: One block (zero movement of the axes in the selected plane)



```

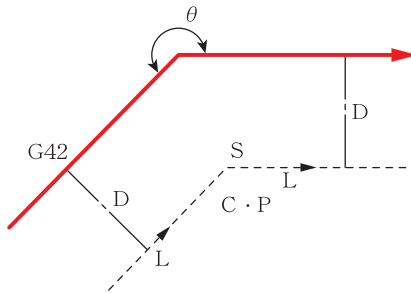
:
N4 G91 X5000 Y5000
N5 X0
N6 X5000
:

```

ME33018R1000800070002

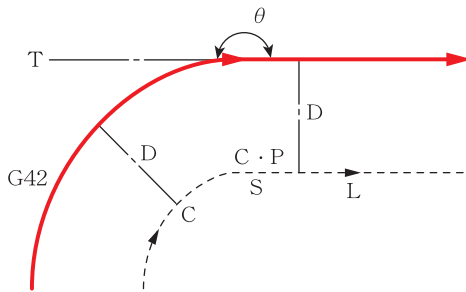
2-3-1. Inside Cutting ($\theta \geq 180^\circ$)

(1) Straight line - Straight line



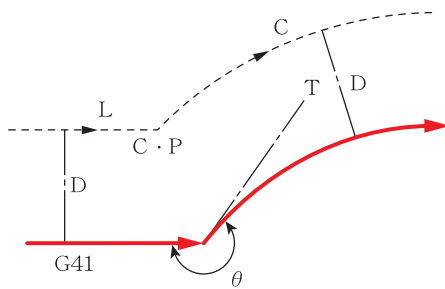
ME33018R1000800080001

(2) Straight line - Arc



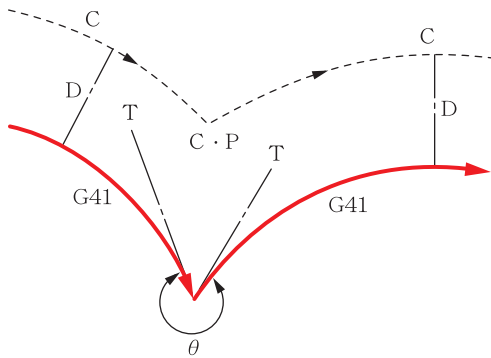
ME33018R1000800080002

(3) Arc - Straight line



ME33018R1000800080003

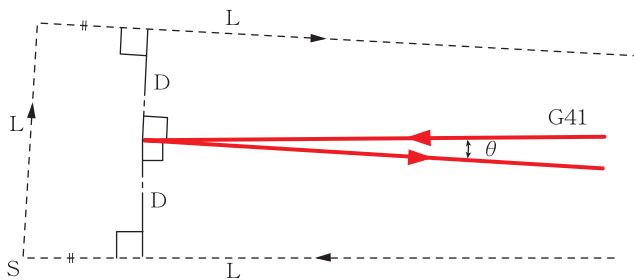
(4) Arc - arc



ME33018R1000800080004

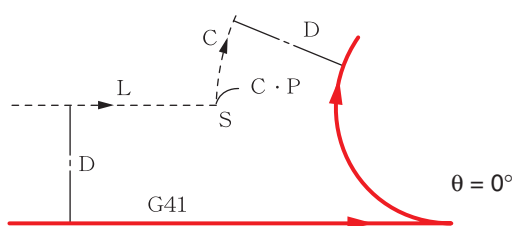
(5) Exception

There is an exception in processing where inside cutting at 1 degree or less for the straight line - straight line configuration is replaced by outside cutting (this is explained later) because the ordinary method of finding the cross point will deviate significantly from the command value. Straight line - straight line $\theta \leq 1^\circ$



ME33018R1000800080005

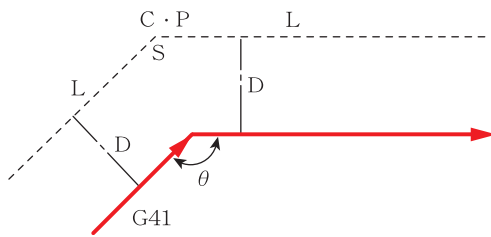
(6) The processing shown above is limited to the straight line - straight line configuration. In other cases, such as the straight line - arc shown below, the ordinary method is used.



ME33018R1000800080006

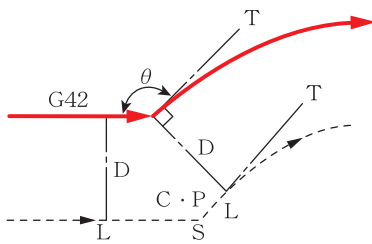
2-3-2. Obtusely Angled Corner - Outside Cutting ($90^\circ \leq \theta \leq 180^\circ$)

(1) Straight line - Straight line



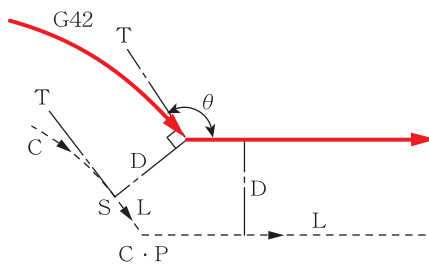
ME33018R1000800090001

(2) Straight line - Arc



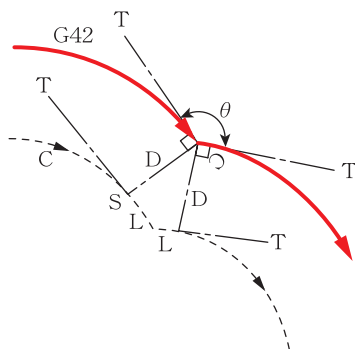
ME33018R1000800090002

(3) Arc - Straight line



ME33018R1000800090003

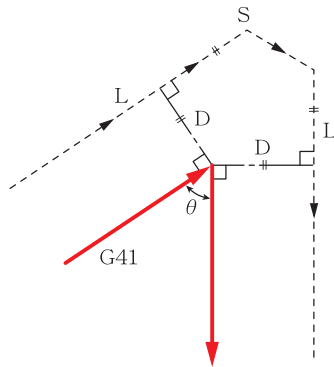
(4) Arc - arc



ME33018R1000800090004

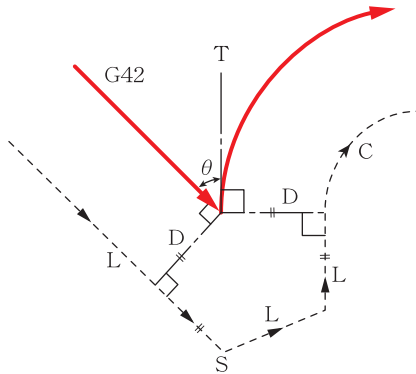
2-3-3. Acutely Angled Corner - Outside Cutting ($\theta < 90^\circ$)

(1) Straight line - Straight line



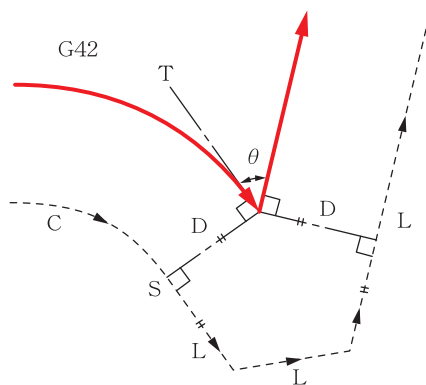
ME33018R1000800100001

(2) Straight line - Arc



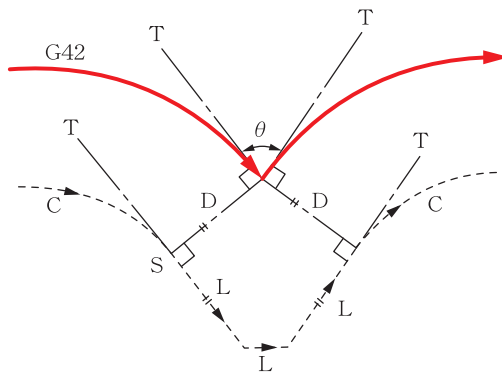
ME33018R1000800100002

(3) Arc - Straight line



ME33018R1000800100003

(4) Arc - arc

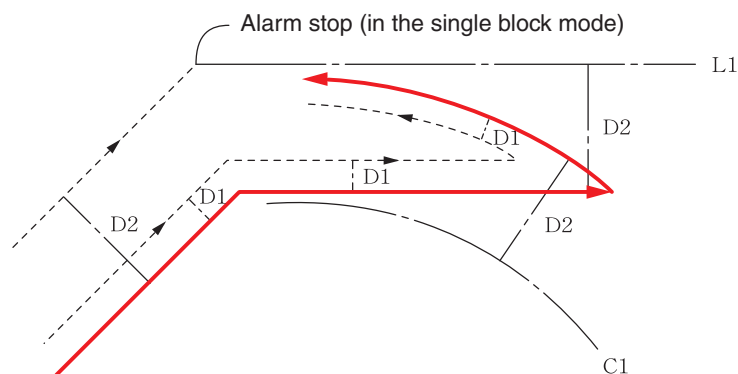


ME33018R1000800100004

2-3-4. Inside Cutting, with Failure to Find Cross Point

As shown in the illustration below, there may be situations in which a cross point exists with a small compensation amount (D1), but not with a large compensation amount (D2). In this case, an alarm occurs and operation stops.

In the single block mode, the alarm occurs in the block which precedes the one which will cause the alarm state. In other modes, the alarm occurs several blocks before the block causing the “no cross point” condition.



ME33018R1000800110001

2-4. Tool Movement when Cutter Radius Compensation is Canceled

[Function]

When the following commands are executed in the cutter radius compensation mode, the cutter radius compensation cancel mode is set.

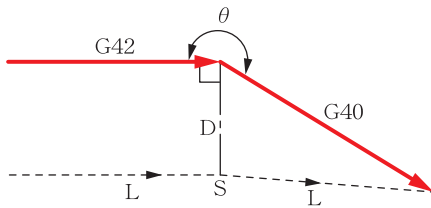
[Programming format]

G40 G00 (G01) Xp__ Yp__

The axis movement mode for canceling the cutter radius compensation mode must be either G00 or G01.

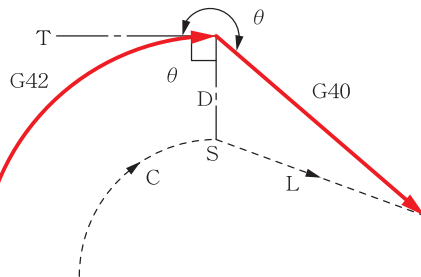
2-4-1. Inside Cutting ($\theta \geq 180^\circ$)

(1) Straight line - Straight line



ME33018R1000800130001

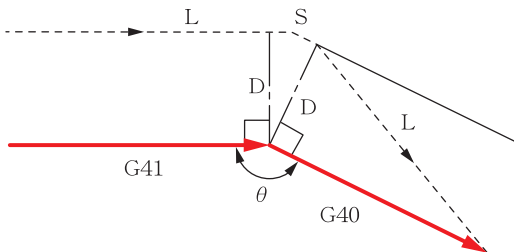
(2) Arc - Straight line



ME33018R1000800130002

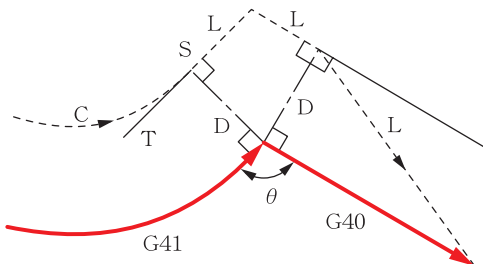
2-4-2. Obtusely Angled Corner - Outside Cutting ($90^\circ \leq \theta \leq 180^\circ$)

(1) Straight line - Straight line



ME33018R1000800140001

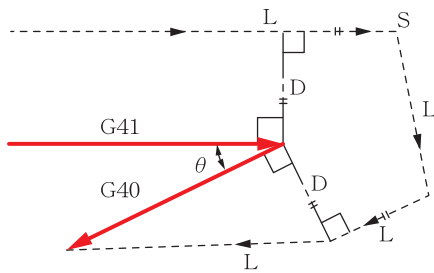
(2) Arc - Straight line



ME33018R1000800140002

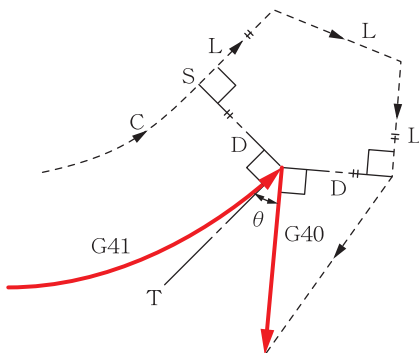
2-4-3. Acutely Angled Corner - Outside Cutting ($\theta < 90^\circ$)

- (1) Straight line - Straight line



ME33018R1000800150001

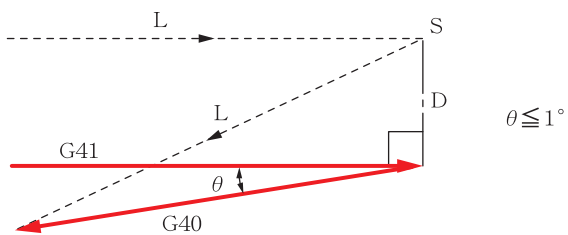
- (2) Arc - Straight line



ME33018R1000800150002

- (3) Exception

Outside cutting at an acute angle of 1 degree or less is considered to be "inside" as shown below.

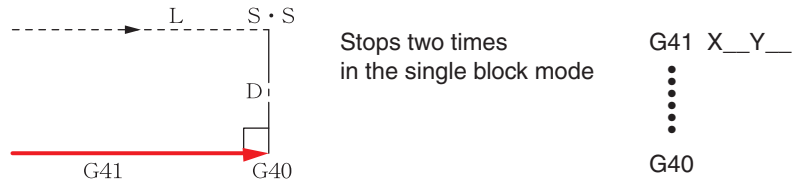


ME33018R1000800150003

2-4-4. Independent G40 Command

G40 given independently will position the axes at a point shifted in the vertical direction by an amount equivalent to the compensation amount (D) from the position specified in the preceding block.

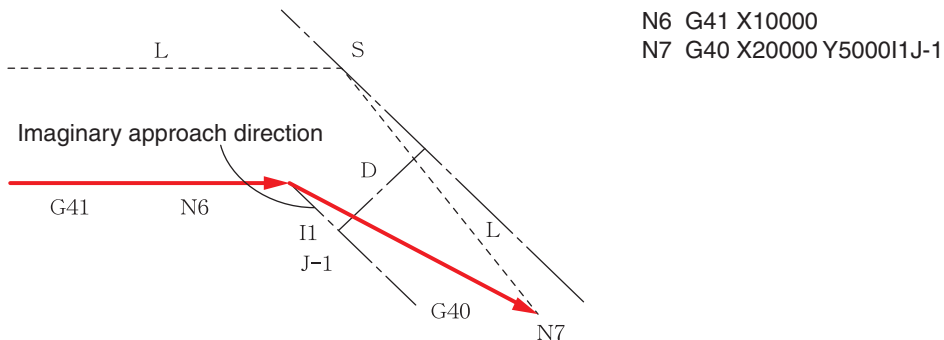
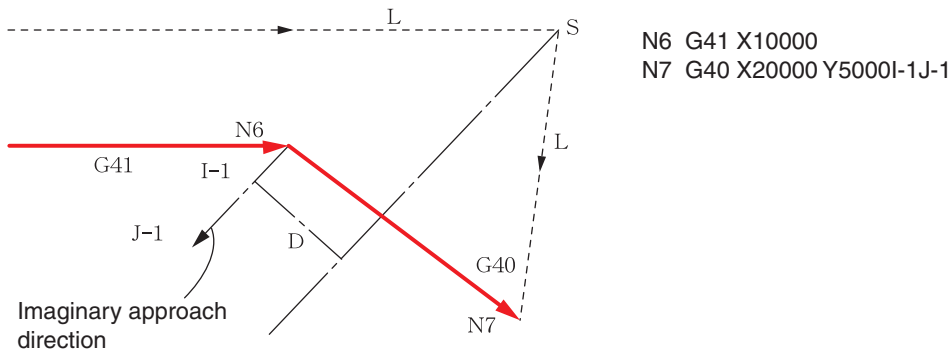
Straight line



ME33018R1000800160001

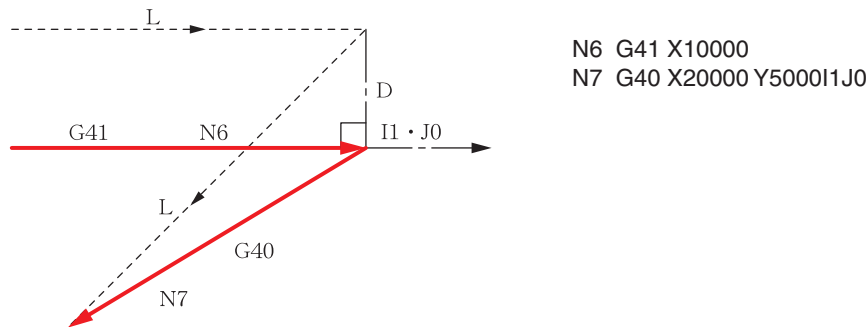
2-4-5. Cancel with Imaginary Approach Direction

If the block which cancels the cutter radius compensation mode includes any I__, J__, or K__ belonging to the offset plane (I__, J__ in the case of G17 plane), the axes move to the target point specified in this block from the direction defined by I__ and/or J__. In this case, note that the cross point is always calculated regardless of whether the cutting is “inside” or “outside”.



ME33018R1000800170001

If no cross point exists, positioning is executed to the point obtained by a vertical shift by the compensation amount from the target point specified in the block immediately preceding the G40 block.



ME33018R1000800170002

2-5. Changing Compensation Direction in Cutter Radius Compensation Mode

- The direction of compensation may be changed even in the cutter radius compensation mode by executing G41 or G42 or by reversing the sign (plus or minus) of the compensation amount.

G Code	Positive/Negative Sign	
	+	-
G41	Offset to left (cutting left side)	Offset to right (cutting right side)
G42	Offset to right (cutting right side)	Offset to left (cutting left side)

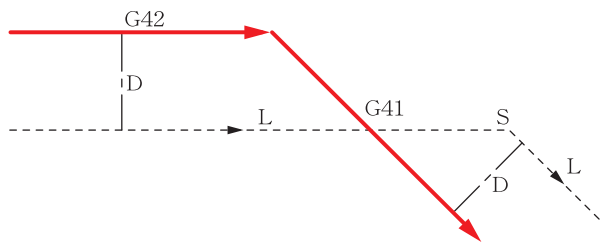
- Execution conditions

Mode	Command	Straight line - Straight line	Straight line - Arc	Arc - Straight line	Arc - Arc
G41	G41	Not valid (When the plus or minus sign of the offset amount is not changed)			
G42	G42				
G41	G42	Executable		Alarm if no cross point exists	
G42	G41				

When changing the offset direction, there are no distinctions between inside and outside cutting, but there are differences depending on whether or not a cross point exists. The following descriptions assume that the compensation amount is positive.

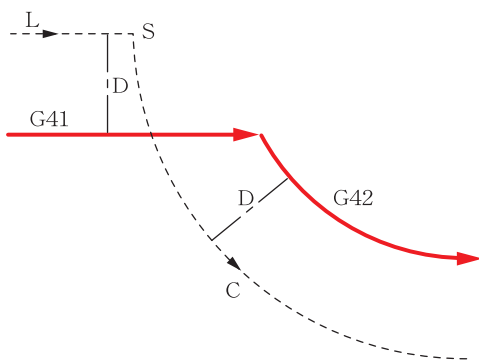
2-5-1. With Cross Point

- (1) Straight line - Straight line



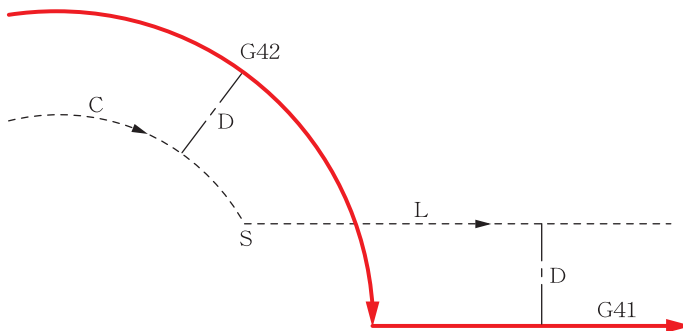
ME33018R1000800190001

- (2) Straight line - Arc



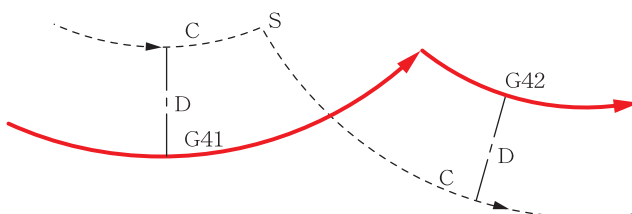
ME33018R1000800190002

- (3) Arc - Straight line



ME33018R1000800190003

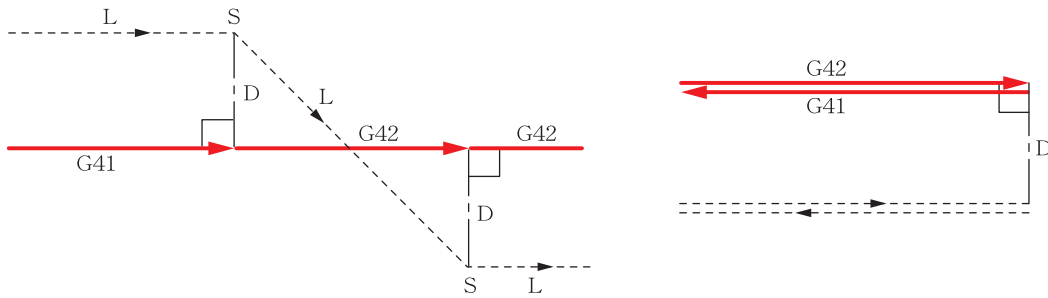
- (4) Arc - Arc



ME33018R1000800190004

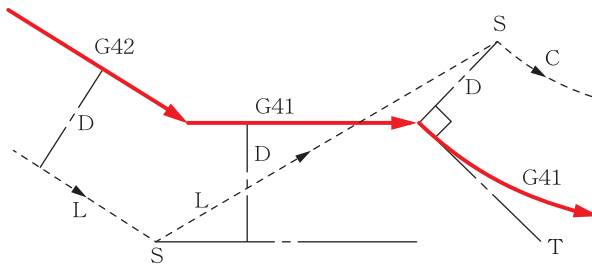
2-5-2. Without Cross Point

(1) Straight line - Straight line



ME33018R1000800200001

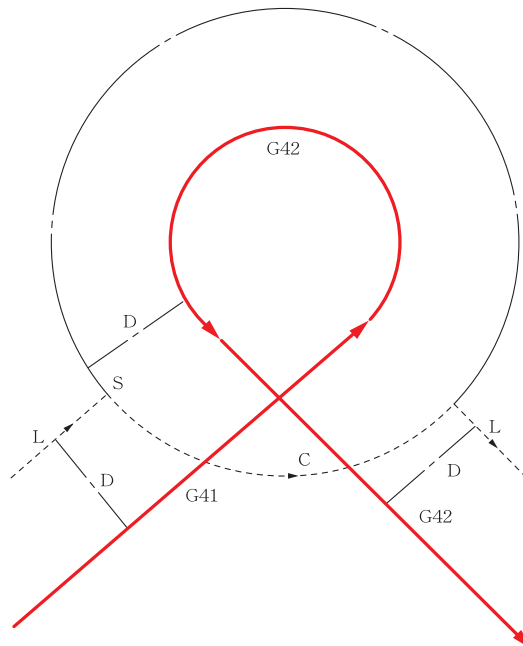
(2) Straight line - Arc



ME33018R1000800200002

2-5-3. Circular Arc Forming an Overlapping Circle

If an overlapping circle (exceeding a full circle) is generated as the result of offset direction change, the tool will move along a shorter arc to reach the target point. To avoid this, the circular arc should be divided.



ME33018R1000800210001

2-6. Cutter Radius Compensation Type A

2-6-1. Overview

Programs are often created using hypothetical cutter radius first and then used by setting the cutter radius compensation for the difference between the hypothetical and actual cutter radiuses. In the OSP system, cutting direction is determined by the sign for the cutter radius compensation value. However, over cutting error may result if cutting direction differs after difference between the hypothetical and actual cutter radiuses are set. This over cutting error is rather obvious at the beginning and the end of the compensation. Set the following parameter to prevent the cutting errors at the beginning and the end of the compensation.

2-6-2. Parameter

Conventional tool movement at the beginning of cutter radius compensation (G41/G42) and the end of it (G40) is referred to as "Type B". Tool movement at the beginning and end of cutter radius compensation in this section is referred to as "Type A". No difference between these two types occurs in the exception handling as in a case of outside cutting at an acute angle of 1° or less and machining with an imaginary approach direction. No difference occurs in the machine movement during cutter radius compensation mode as well.

Following NC optional parameter is to switch Type A and B.

NC optional parameter bit				
NO.	Bit	Contents	<input checked="" type="checkbox"/>	<input type="checkbox"/>
79	4	Selects type of tool movement at the beginning and end of cutter radius compensation.	Type A	Type B (Conventional specification)

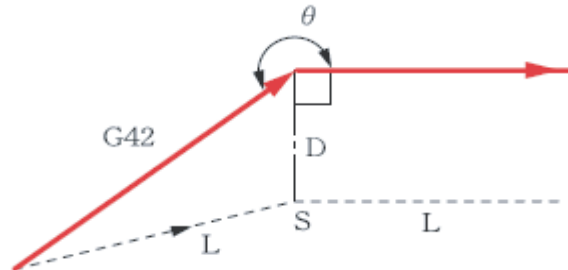
Default setting is Type B.

2-6-3. Tool Movement at the Beginning of Cutter Radius Compensation

Inside cutting $\theta \geq 180^\circ$

Tool moves to the position of the vector vertical to the next command's origin irrespective of cutter radius compensation types.

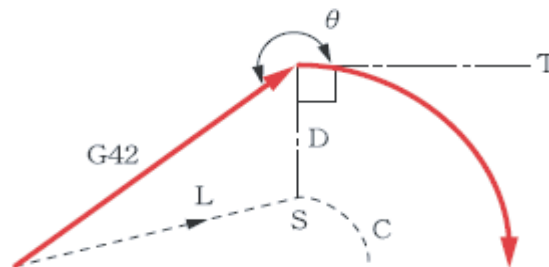
(1) Straight line - Straight line



- S : Single block stop point
- L : Linear motion
- C : Circular motion
- T : Tangent to an arc
- D : Cutter radius compensation amount
- θ : Angle at the workpiece side
- CP : Cross point, made when a programmed path (or the tangent to an arc) is shifted by a compensation amount
- \rightarrow : Programmed tool path
- $- \rightarrow$: Tool center path
- $---$: Auxiliary line

ME33018R1000800240001

(2) Straight line - Arc

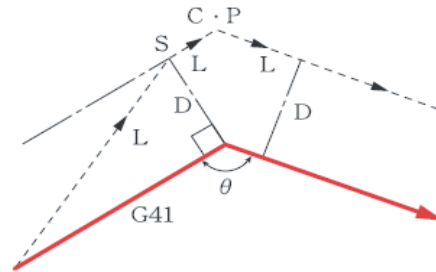


ME33018R1000800240002

Obtusely Angled Corner - Outside Cutting ($90^\circ \leq \theta < 180^\circ$)

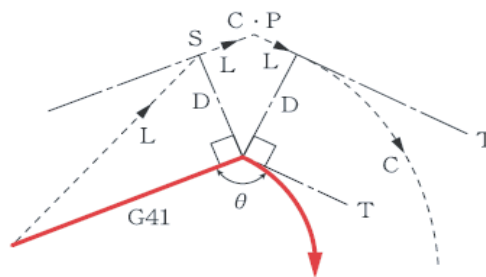
In Type B, as is conventionally done, the tool detours by calculating an extension point as shown below.

- (1) Straight line - Straight line



ME33018R1000800240003

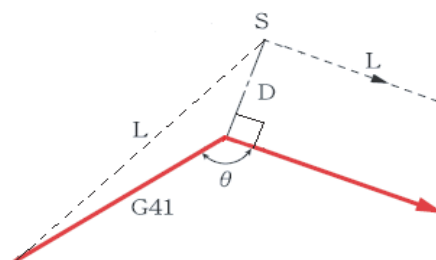
- (2) Straight line - Arc



ME33018R1000800240004

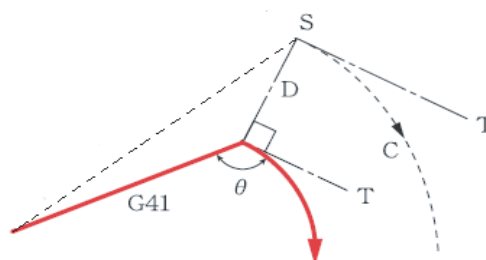
In Type A, the tool moves to the position of the vector vertical to the next command's origin as shown below.

- (1) Straight line - Straight line



ME33018R1000800240005

- (2) Straight line - Arc

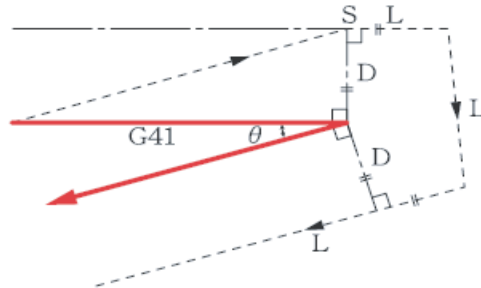


ME33018R1000800240006

Acutely Angled Corner - Outside Cutting ($\theta < 90^\circ$)

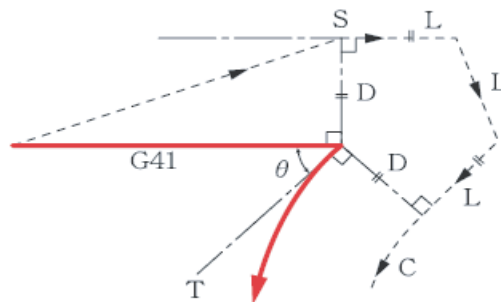
In Type B, as is conventionally done, the tool detours by calculating an extension point as shown below.

- (1) Straight line - Straight line



ME33018R1000800240007

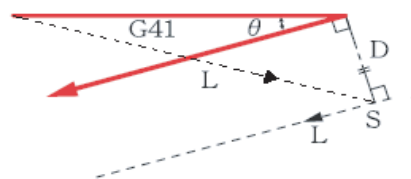
- (2) Straight line - Arc



ME33018R1000800240008

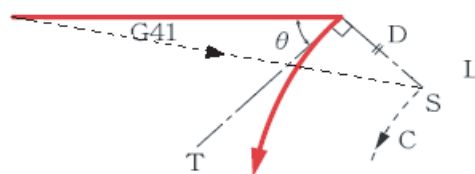
In Type A, the tool moves to the position of the vector vertical to the next command's origin as shown below.

- (1) Straight line - Straight line



ME33018R1000800240009

- (2) Straight line - Arc



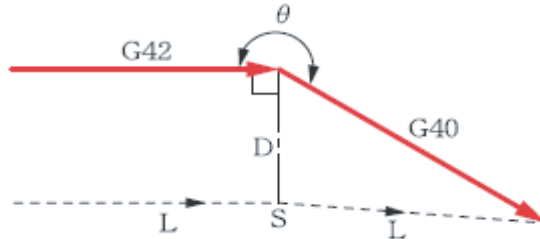
ME33018R1000800240010

2-6-4. Tool Movement at the End of Cutter Radius Compensation

Inside cutting $\theta \geq 180^\circ$

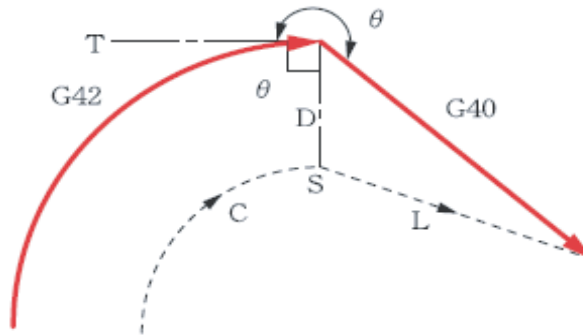
Tool moves to the position of the vector vertical to the previous command's end irrespective of cutter radius compensation types.

- (1) Straight line - Straight line



ME33018R1000800250001

- (2) Straight line - Arc

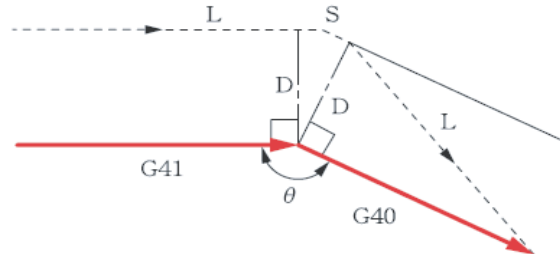


ME33018R1000800250002

Obtusely Angled Corner - Outside Cutting ($90^\circ \leq \theta < 180^\circ$)

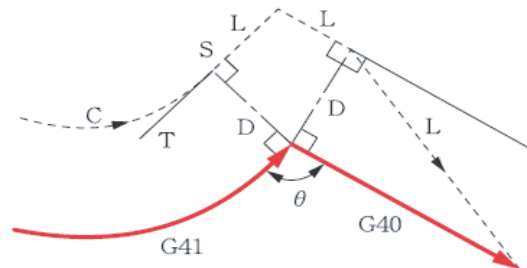
In Type B, as is conventionally done, the tool detours by calculating an extension point as shown below.

- (1) Straight line - Straight line



ME33018R1000800250003

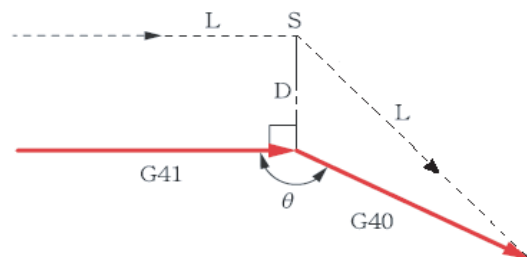
- (2) Straight line - Arc



ME33018R1000800250004

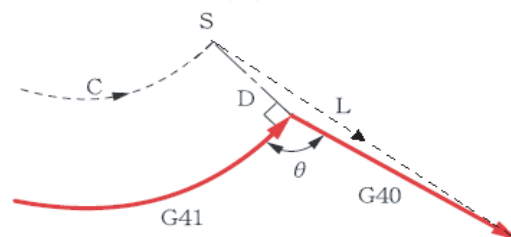
In Type A, the tool moves to the position of the vector vertical to the previous command's end as shown below.

- (1) Straight line - Straight line



ME33018R1000800250005

- (2) Straight line - Arc

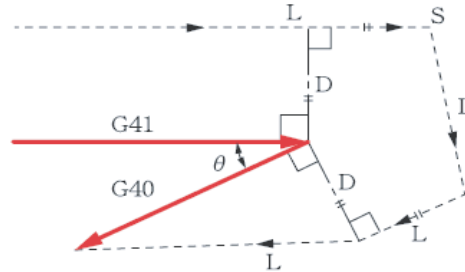


ME33018R1000800250006

Acutely Angled Corner - Outside Cutting ($\theta < 90^\circ$)

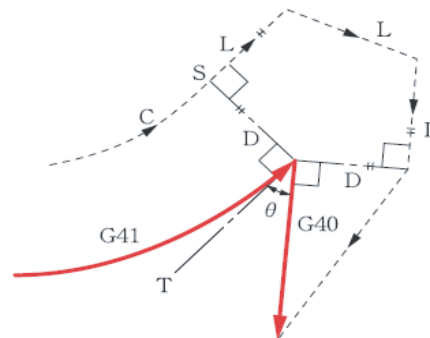
In Type B, as is conventionally done, the tool detours by calculating an extension point as shown below.

- (1) Straight line - Straight line



ME33018R1000800250007

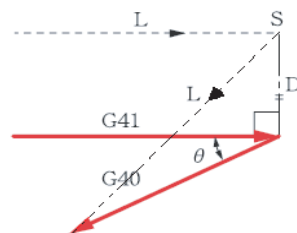
- (2) Straight line - Arc



ME33018R1000800250008

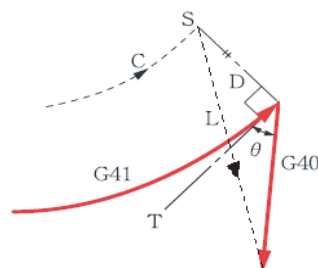
In Type A, the tool moves to the position of the vector vertical to the previous command's end as shown below.

- (1) Straight line - Straight line



ME33018R1000800250009

- (2) Straight line - Arc



ME33018R1000800250010

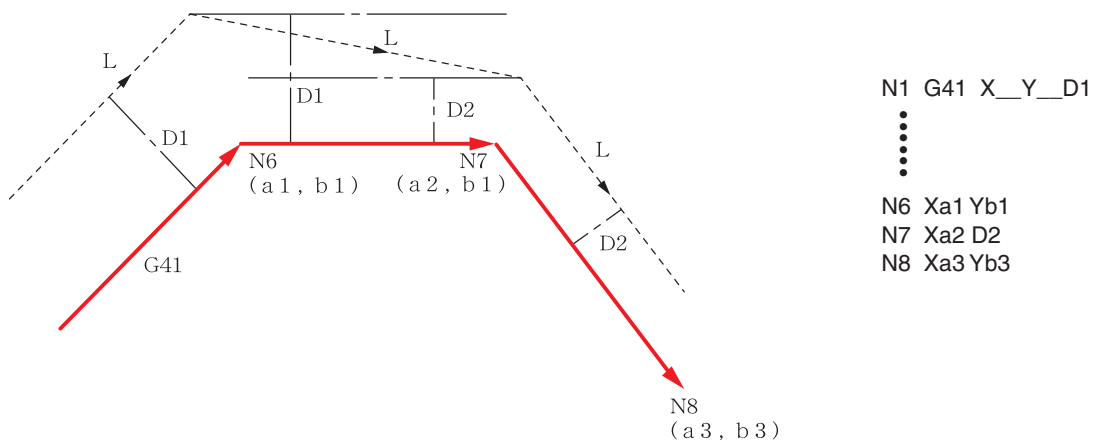
2-7. Notes on Cutter Radius Compensation

2-7-1. Specifying the Cutter Radius Compensation Amount

- The compensation amount is specified as a D command. A D command is usually specified with G41 or G42 in the same block. If no D command is included in a G41 or G42 block, the previously specified D command is used.
- The range of cutter radius compensation numbers is from D00 to D100 for the standard specification, and this can be expanded to D200 or D300. The compensation amount of D00 is "0". The compensation data is set in the tool data setting mode.

2-7-2. Changing the Compensation Amount

If the compensation amount is changed in the compensation mode, the new compensation amount becomes valid starting at the end of the block in which the new compensation amount is specified.



ME33018R1000800270001

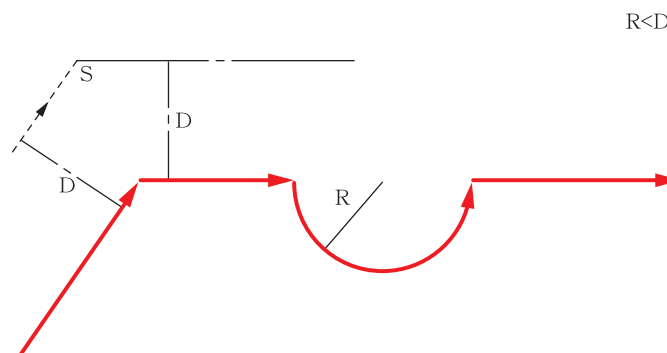
2-7-3. Actual Position Data Display

For the present position display, the coordinate value of the tool center is displayed.

2-7-4. Inside Cutting of an Arc Smaller than the Cutter Radius

An alarm occurs and operation stops if the inside of an arc that is smaller than the cutter radius is going to be cut.

In the single block mode, operation stops at the end point two blocks ahead of the block which specifies such an operation, and in other modes, operation stops several blocks ahead.

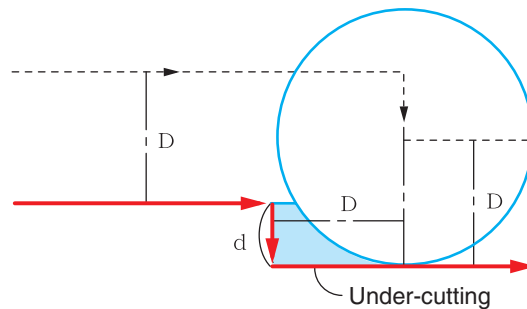


ME33018R1000800290001

2-7-5. Under-cutting

Under-cutting may occur when cutting a step with a height smaller than the cutter radius.

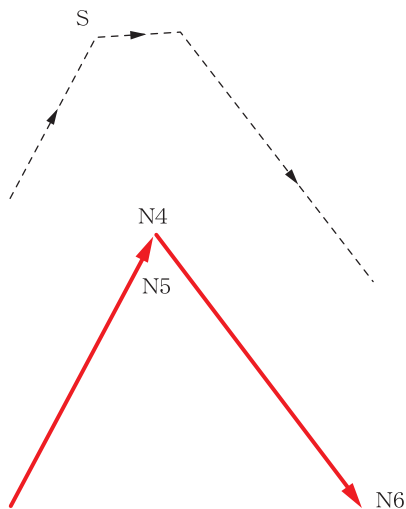
$$d < D$$



ME33018R1000800300001

2-7-6. Cautions on Corner Cutting

- When cutting an outside corner, a polygonal tool path is generated. The axis move mode and feedrate at the corners will follow the commands specified in the next block. If the interpolation mode in the next block is either G02 or G03, the tool moves in the G01 mode along the generated polygonal tool path.



Axis movement for this inserted path is controlled by the command (F800) specified in N5.

N4 X__Y__ F500

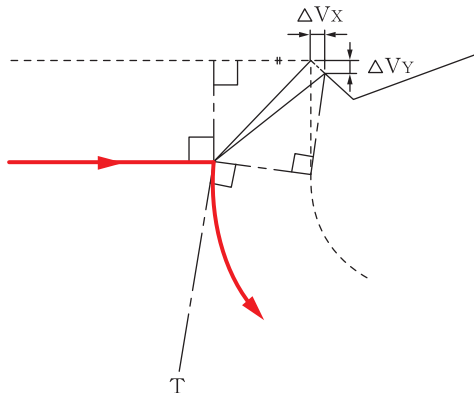
N5 Z__ F800

N6 X__Y__

Z-axis movement is executed at point S

ME33018R1000800310001

- If the tool path inserted to cut a corner is very small ($\Delta V_x \leq \Delta V$ and $\Delta V_y \leq \Delta V$ in the illustration), the second point defining this movement is disregarded.

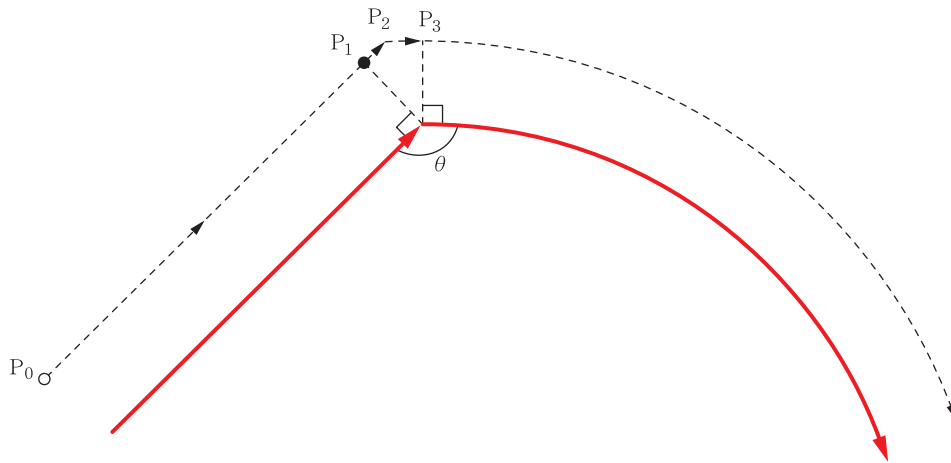


The second point defining extra movement is disregarded if $\Delta V_x \leq \Delta V$ and $\Delta V_y \leq \Delta V$.

ΔV value: Set for COMPENSATION VECTOR CHECK of NC optional parameter (CUTTER R COMPENSATION)

ME33018R1000800310002

In this manner, the additional minute axis movement may be reduced.
Note that this processing is not executed when the next block forms a full circle.



In the illustration shown above, correct movement should be as follows:

- 1) P0 - P1 - P2 Straight line
- 2) P2 - P3 Straight line
- 3) From point P3 Full circle

However, if the movement from point P2 to point P3 is disregarded due to the minute movement processing, the movement up to point P3 is as follows:

- 1) P0 - P1 - P2 Straight line
- 2) P2 - P3 Arc

Thus, the program generates a minute arc from P2 to P3 and disregards the full circle that should be generated after P3.

ME33018R1000800310003

2-7-7. Interference

[Supplement]

Interference refers to problems in which a cutting tool over-cuts or makes too deep a cut into a workpiece. The NC always monitors and checks the occurrence of interference.

The NC judges interference to have occurred in the following case:

When the difference between the direction of the programmed path and that of the path resulting from cutter radius compensation is between 90° and 270° .

It is therefore possible that conditions that do not cause interference are regarded as interference and conditions that actually cause interference are regarded as an interference-free state.

When a corner is cut along a polygonal tool path, each corner can be formed by up to four points.

To check for interference, two corners, P1, P2, P3, P4 and P5, P6, P7, P8, are evaluated.

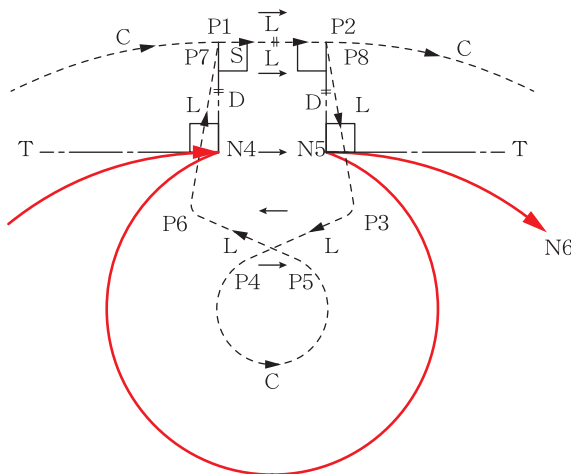
Interference checks are made sequentially; the first interference check is made between the last point of a corner (P4) and the first point of the next corner (P5). If an interference is found, the point is disregarded and the next point is checked. If no interference is found halfway through the procedure, the interference check is not executed for the later points. The movement mode during the check is straight line movement. For the circular interpolation block, axes move along the inserted polygonal path in the G01 linear interpolation mode.

If an interference remains after all points have been checked, an interference alarm occurs, but the very last point is not disregarded. As a result, over-cut can occur if the program is executed in the single block mode.

How the interference check is executed is explained below using several examples.

(1) Interference not found

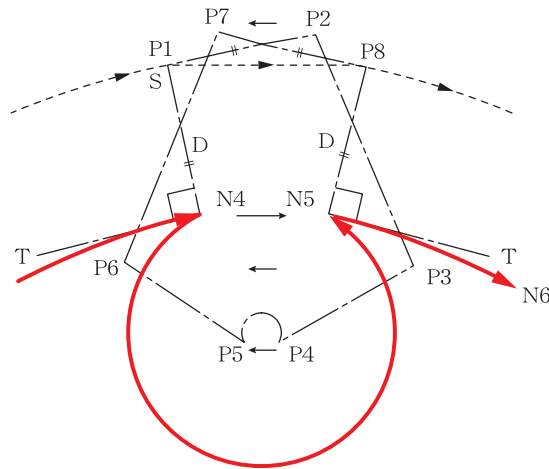
In this example, no interference is found in the first check ($N4 \rightarrow N5$ and $P4 \rightarrow P5$). Therefore, no checks are made on the later points and the interference is not discovered.



Although direction P3-P6 is reversed, this is not checked since there is no interference in the check on P4-P5.

(2) Interference check resulting in a path change

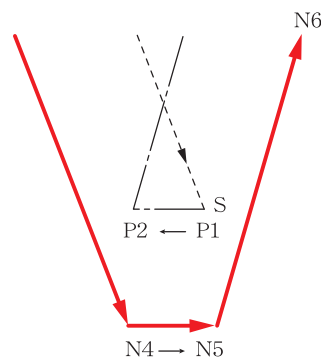
In this example, the following directions of movement are checked and disregarded because interference is discovered: N4 - N5, P4 - P5, P3 - P6 and P2 - P7. However, since interference is not found in the check on P1 - P8, the tool moves along this path (P1 - P8) in the G01 mode.



ME33018R1000800320002

(3) Interference check resulting in an alarm

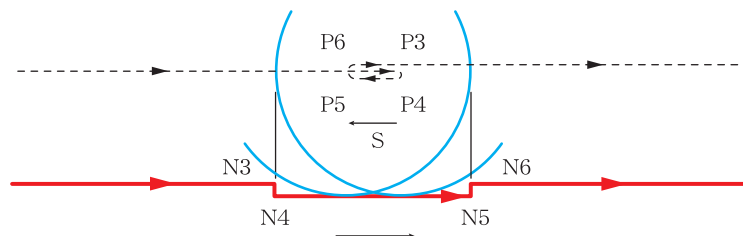
In this example, each corner has only one point and point P1 remains and is not disregarded. In the single block mode, an alarm occurs and operation stops after positioning is executed at P1. In other operation modes, an alarm occurs and operation stops several blocks ahead of the block causing positioning at P1.



ME33018R1000800320003

(4) Non-interference considered interference

In this example, if N4-N5 is smaller than the cutter diameter, no interference will take place. However, since the direction of P4-P5 is opposite to that of N4-N5, an interference alarm occurs.

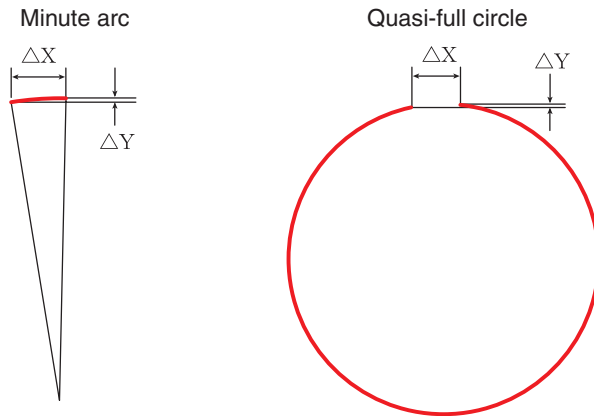


ME33018R1000800320004

(5) Minute arc and quasi-full circle

A minute arc is defined as an arc in which the horizontal and vertical distances from start to end point is smaller than the value set at ERROR DATA RESULTING FROM CUTTER R COMP. CAL. of NC optional parameter (cutter R compensation).

A quasi-full circle is defined as an arc which is close to a full circle; the horizontal and vertical distances of the break is smaller than the value set at ERROR DATA RESULTING FROM CUTTER R COMP. CAL. of NC optional parameter (cutter R compensation).



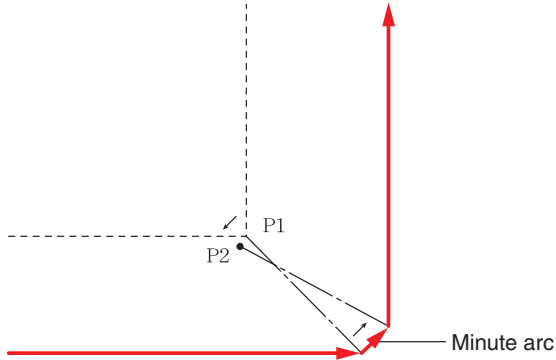
ME33018R1000800320005

Here,

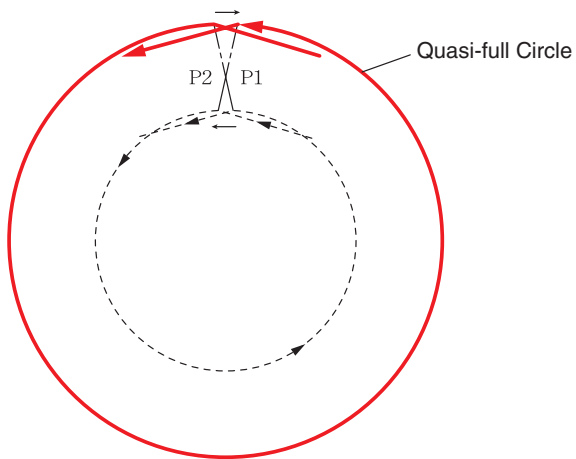
Assume that $\Delta X \leq \Delta Y$ and $\Delta Y \leq \Delta V$.

ΔV : Set at ERROR DATA RESULTING FROM CUTTER R COMP. CAL. of NC optional parameter (cutter R compensation).

For these two types of arcs, special interference checks are provided. "Problem" conditions detected in minute arcs and quasi-full circles by an interference check are not considered interference, but are regarded as operational errors. In the case of a minute arc, the end point is disregarded and the shape is regarded as a point; no movements along an arc are executed. In the case of a quasi-full circle, the end point is disregarded and the shape is processed as a full circle.



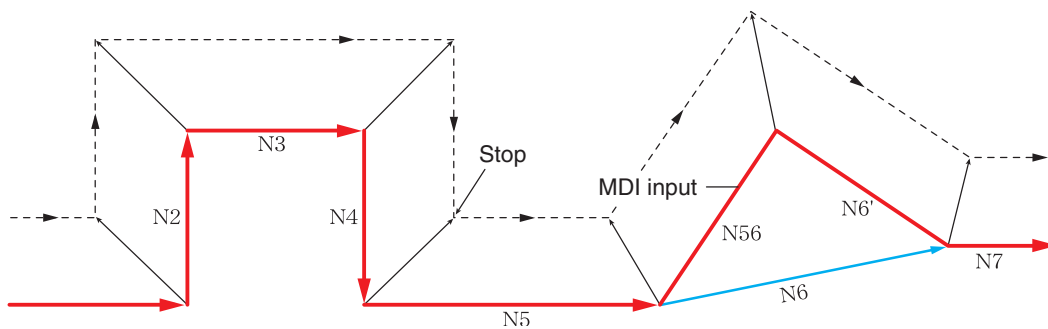
P2 is disregarded and circular interpolation is not executed.



P2 is disregarded and a full circle from P1 is formed.

2-7-8. Manual Data Input

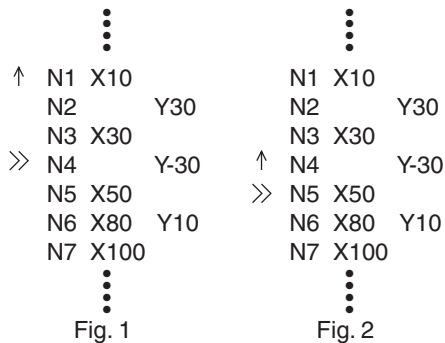
- If the cutter radius compensation mode is set while in the MDI mode, or if the MDI mode is set in the cutter radius compensation mode, execution of a block of commands including an axis movement command is not allowed immediately after their input from the keyboard. In this case, the commands of the next axis movement must be input before executing the presently input commands. Alternatively, instead of inputting the next axis movement commands, inputting four successive blocks of commands not including axis movements also allows the execution of the presently input commands.
- In automatic operation with single block function OFF, if the mode is changed to the MDI mode, the program is executed up to the block immediately ahead of the block that has been read to the buffer (the line identified by a ">>" symbol on the screen) and then operation stops. The commands input in the MDI mode are read next to the block in the buffer, then the cutter radius compensation function is executed.



ME33018R1000800330001

Example:

Suppose that the MDI mode is established while block N1 is being executed. If the screen displays the program shown in Fig. 1, operation stops after block N4 is executed. After the operation is stopped, the screen displays the program as shown in Fig. 2.



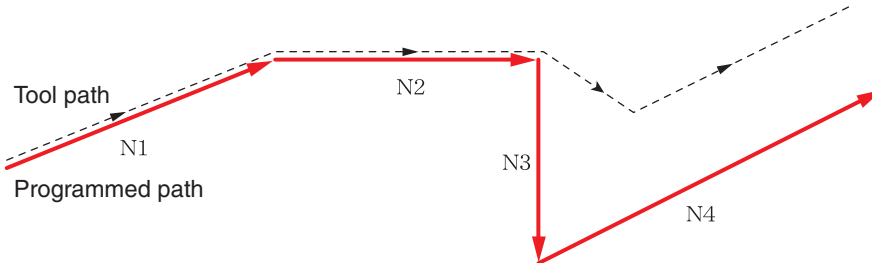
ME33018R1000800330002

When the commands of block N56 are input from the keyboard and the CYCLE START button is pressed, block N5 is executed and then operation stops. If the operation mode is returned to an automatic mode and the CYCLE START button is pressed, blocks are executed in the order N56, N6", then N7.

2-7-9. Zero Cutter Radius Compensation Amount

(1) During start-up

The cutter radius compensation mode is established when G41 or G42 is executed in the cancel mode, and the cutter radius compensation mode start-up operation is executed with a cutter radius compensation amount of zero. In this case, however, cutter radius compensation is not executed. When the cutter radius compensation number is changed to the one that calls a compensation amount other than zero in the cutter radius compensation mode, the processing described in “Changing the Compensation Amount” is executed.



```

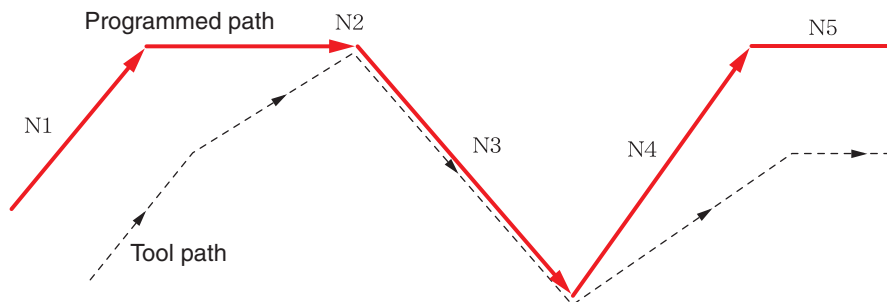
N1 G41 X__Y__ D01   Offset data in D01 = 0
N2   X__
N3   Y__ D02   Offset data in D02 = 0
N4   X__Y__

```

ME33018R1000800340001

(2) During compensation mode

Even if the cutter radius compensation number is changed to one that calls a compensation amount of “zero” while in the cutter radius compensation mode, the cancel processing is not executed, and neither is the cancel mode set. In this case, the processing described in “Changing the Compensation Amount” is executed. When the cutter radius compensation number is changed again to one that calls a compensation amount other than zero, the same processing as described in “Changing the Compensation Amount” is also executed.



```

N1 X__Y__   Offset data in D01 = 0
N2 X__ D01
N3 X__Y__   Offset data in D02 = 0
N4 X__Y__ D02
N5 X__

```

ME33018R1000800340002

3. Cutter Radius Compensation Mode Override Function

3-1. Automatic Override at Corners

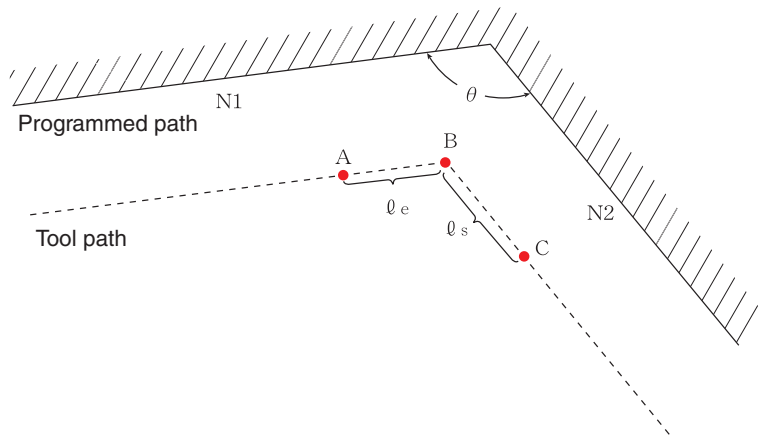
[Function]

In the cutter radius compensation mode, depth of cut may increase while cutting the inside of a corner, resulting in an increased tool load. To reduce the load applied to the tool, feedrates are automatically overridden.

[Setting values]

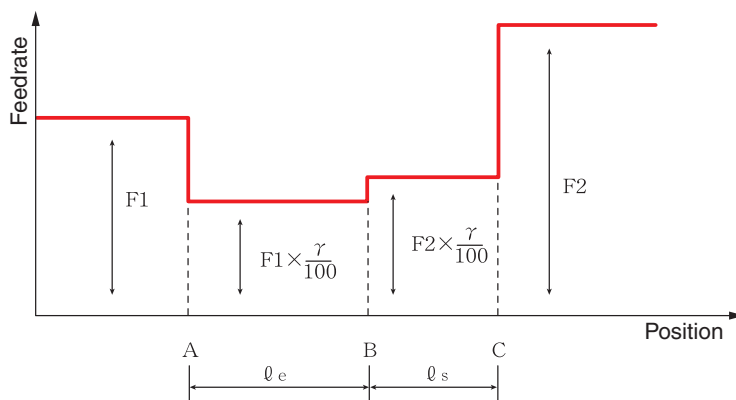
Set the following four parameters directly in the PARAMETER SET mode.

- *Slowdown distance at the end point of a corner
le: End Point Decelerating Distance of NC optional parameter (cutter R compensation) Setting range: 0 to 99999.999 mm or 0 to 3937.0078 inches
Default: 0
- Slowdown distance at the start point of a corner
ls: Starting Point Decelerating Distance of NC optional parameter (cutter R compensation) Setting range: 0 to 99999.999 mm or 0 to 3937.0078 inches
Default: 0
- Slowdown rate
 γ : Decelerating Rate of NC optional parameter (cutter R compensation) Setting range: 1 to 100%
Default: 100%
- Angle of inside cut
 θ : Corner's Internal Identification Angle of NC optional parameter (cutter R compensation) Setting range: 1 to 179°
Default: 90°



A : Override ON position
 B : Position at which
 positioning is carried
 out by the N1 block
 commands
 C : Override OFF position

N1 X__Y__F1000
 N2 X__Y__F2000



ME33018R1000800350001

- Requirements for turning ON the override function
 The override function will be turned ON if both of the two blocks that form a corner satisfy the following requirements.
 - The block is specified in the cutter radius compensation mode.
 - The axis movement mode is G01, G02, or G03.
 - The corner's inner angle is smaller than the value, θ , set for the "angle of inside cut".
 - No more than three blocks calling no axis movements are entered between these two blocks.
 - G40, G41, or G42 is not specified in these two blocks.
 - A rotary axis command is not specified in these two blocks.
 - The slowdown rate is not 100%.
 - The direction of compensation does not change.
 - The slowdown distance at the corner start point and end point is not "0".

The override function is also valid for F1-digit feed commands as long as the above requirements are satisfied.

The override function is invalid for dry run operation even if the above requirements are satisfied.

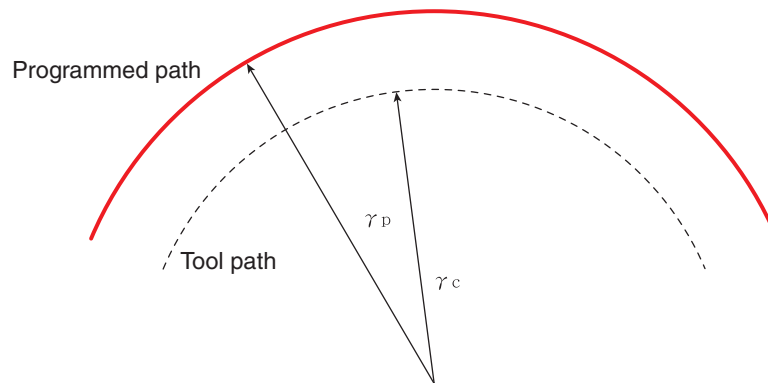
3-2. Circular Arc Inside Cutting Override

[Function]

In the cutter radius compensation mode, feedrate is normally controlled so that the feedrate on the tool path (the path along which the tool center moves) will be the specified feedrate. When cutting the inside of a circular arc, however, the feedrate is overridden so that the feedrate on the programmed path will be the feedrate specified in the program.

[Setting value]

Slowdown rate is set by the NC optional parameter (word) No. 5.



ME33018R1000800360001

[Details]

- When cutting the inside of an arc as shown in the illustration above, the actual feedrate is expressed by the formulas indicated below.

$$1. \frac{\gamma_c}{\gamma_p} \times 100 \geq \text{Parameter setting}$$

$$\text{Actual feedrate} = \text{Programmed feedrate} \times \frac{\gamma_c}{\gamma_p}$$

$$2. \frac{\gamma_c}{\gamma_p} \times 100 < \text{Parameter setting}$$

$$\text{Actual feedrate} = \text{Programmed feedrate} \times \text{Parameter setting}$$

ME33018R1000800360002

- Requirements for turning ON the circular arc inside cutting override function
The override function will be turned ON when the following requirements are satisfied.
 - The block is specified in the cutter radius compensation mode.
 - Inside cutting of a circular arc is being executed.
 - The above two override functions are effective for both F4-digit and F1-digit commands specified in the tool radius compensation mode.
 - If override functions (automatic corner override, arc inside cutting override, and standard F4-digit feed command override) overlap, the actual feedrate is obtained by the following calculation.

$$\text{Actual feedrate} \times \frac{\gamma_c}{\gamma_p} \times \frac{\text{F4-digit feed override}}{100} \times \frac{\text{Automatic corner}}{100}$$

ME33018R1000800360003

4. Tool Radius Compensation G39 Command

In the tool radius compensation, corner circular interpolation whose radius is compensation amount is possible by issuing G39 command during the offset mode. G39 command can be generated automatically in the NC by switching the parameter.

4-1. Parameter

<<Switching of corner circular interpolation>>

"Valid" of "corner circular interpolation" can be specified with NC optional parameter bit.

NC optional parameter bit				
NO.	Bit	Description	<input checked="" type="checkbox"/>	<input type="checkbox"/>
80	0	G39 (Corner circular interpolation) command in the tool radius compensation is valid.	Valid	Alarm

When G39 command is issued while this parameter is "invalid", the alarm occurs. In the G39 command ignoring spec. (optional), the command will be "ignored" regardless of this parameter.

<<Switching of corner circular interpolation command automatic insertion>>

"Valid" of "corner circular interpolation automatic insertion" can be specified with NC optional parameter bit.

NC optional parameter bit				
NO.	Bit	Description	<input checked="" type="checkbox"/>	<input type="checkbox"/>
80	1	G39 (Corner circular interpolation) command in the tool radius compensation is automatically inserted.	Automatic insertion	No automatic insertion

This parameter functions independently regardless of "corner circular interpolation" parameter.

- Not affected by the "corner circular interpolation" parameter.
- Not affected by the G39 command ignoring spec. (optional).

4-2. Corner Circular Interpolation

Corner circular interpolation whose radius is compensation amount is possible by issuing G39 command. G39 is a one-shot G code.

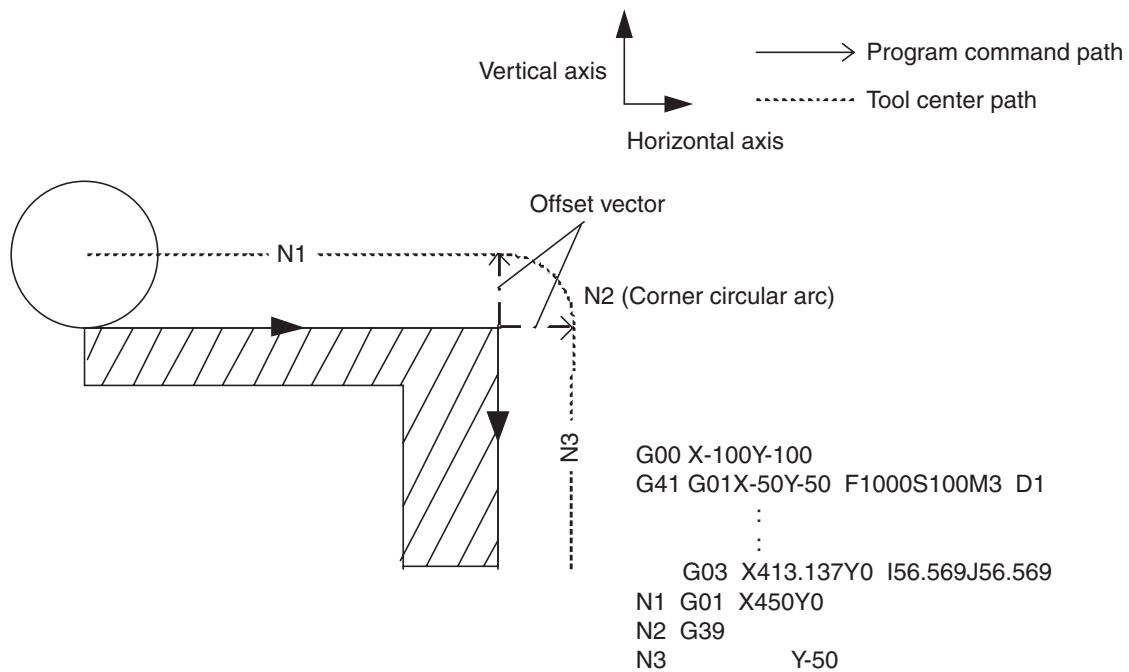
- Format

```
(During offset mode)
G39
or
G17 plane G39 I_J_
G18 plane G39 I_K_
G19 plane G39 J_K_
```

ME33018R1000800390001

- G39 without I, J, K command

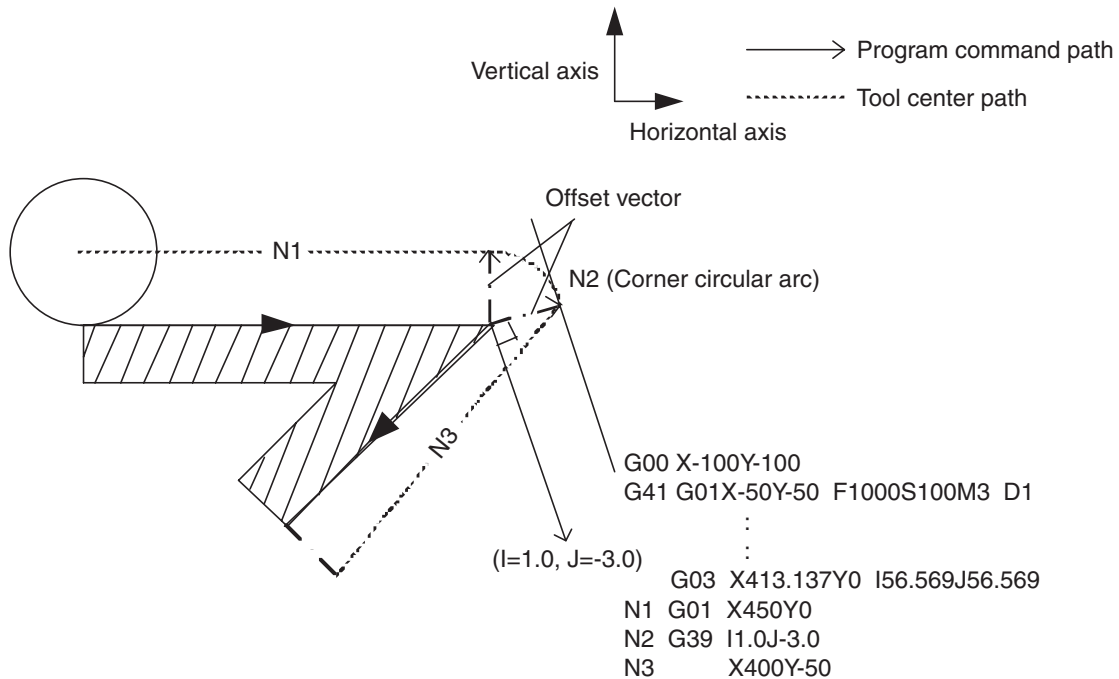
When G39 command is issued, the corner circular arc is drawn with its last vector vertical to the starting point of the next block.



ME33018R1000800390002

- G39 with I, J, K command

When I,J,K command is issued, the corner circular arc is drawn with its last vector vertical to the vector specified by I, J, K command.



ME33018R1000800390003

4-2-1. Restrictions

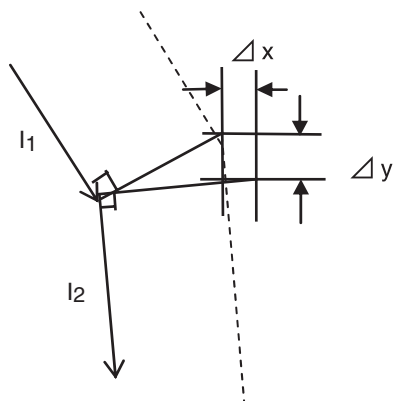
- A command for movement is impossible in G39 block. If issued, "Alarm B 2621 Corner circular interpolation command 3" occurs.
- When changing the compensation amount, it will become valid at the end of the block after drawing circular arc by G39.
- When the compensation amount is "0", the path as instructed by the program is drawn. Corner circular interpolation is not carried out.
- If G39 command is issued at the same point command (travel amount is "0"), G39 is ignored.
- If the number of blocks in which plane axis does not move is four or more in the tool radius compensation mode, G39 command is ignored until reaching the block in which the plane axis moves.
- Do not include I,J,K command in G39 which following approach block or preceding cancel block when approaching and canceling in tool radius compensation if there is a direction of imaginary target approach/exit to prevent the alarm.
- Avoid issuing a command which can be judged as inside while I,J,K command is included.
- Speed of corner circular arc
When the command of corner circular arc is issued by G39 in G00 mode, the speed of the block with corner circular arc is specified by F command for the following block. If G39 command is issued while no F command has been issued in programming, the alarm occurs.
- If the second figure is a circular arc in the tool radius compensation, G39 command including I,J,K command ignores I,J,K command.

4-3. Corner Circular Interpolation Command Automatic Insertion

G39 (corner circular interpolation) command can be generated automatically in the NC during offset mode of tool radius compensation. The last vector of the arc automatically inserted is vertical to the starting point of the next block same as G39 excluding I,J,K command.

4-3-1. Conditions for Corner Circular Interpolation Automatic Insertion

- Automatic insertion is not carried out if the preceding or following block is G39 command. Corner circular interpolation is carried out by G39 command in the program. When the compensation amount is "0", the path as instructed by the program is drawn. Automatic insertion is not carried out.
- Automatic insertion is not carried out at the same point command.
- If the preceding plane axis movement block is an approaching block of tool radius compensation with imaginary target approach, automatic insertion is not carried out.
- If the following plane axis movement block is a cancel block of tool radius compensation with imaginary target exit, automatic insertion is not carried out.
- Circular interpolation automatic insertion is not carried out if the distance from the starting point and end point to each axis is the set value of NC optional parameter long word No.7 or less (from starting point and end point on the tool plane to each axis during execution of tool side offset function) after G39 command insertion. In this case, G01 (linear interpolation) is inserted instead of circular interpolation.



When

$$\Delta x \leq \text{Set value,}$$

and

$$\Delta y \leq \text{Set value,}$$

corner circular interpolation is not carried out.

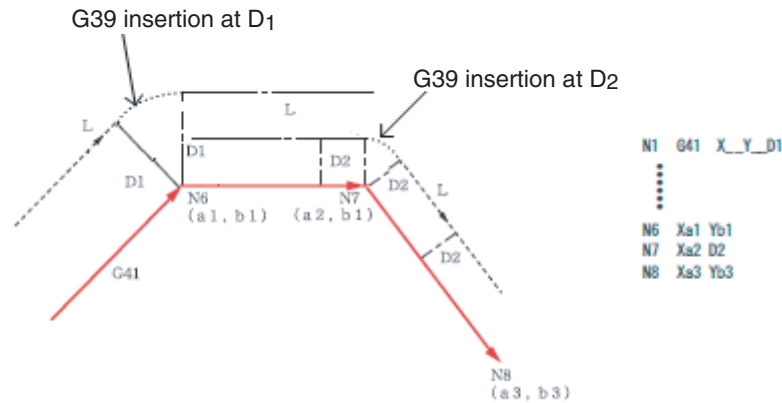
Operates the compensation point at l_1 end to the compensation point at l_2 end.

ME33018R1000800420001

- G39 command is generated automatically only when external cutting, but not when internal cutting. Automatic generation of G39 does not occur when approaching and canceling in the tool radius compensation type A (NC optional parameter bit No.79 Bit 4: Valid), since approaching and canceling operations are considered those when internal cutting.
- When changing compensation direction (G42 \Leftrightarrow G41 or change of data sign of radius compensation D), automatic generation does not occur and the operation is carried out in the same way as current cross point way.

4-3-2. Others

- When changing the compensation amount, it will become valid at the end of the block after drawing circular arc by G39.



ME33018R1000800430001

- Speed of corner circular arc
The speed of corner circular interpolation generated automatically is specified by F command for the following block.
- Single block is prohibited in the corner circular interpolation generated automatically and is handled together with the following block. Therefore, counting is not added in the block counter at the corner circular interpolation block automatically generated.
- When G39 is generated automatically, G39 command is displayed on the block data.

5. Three-dimensional Tool Offset (G43, G44) (Optional)

The three-dimensional tool offset function executes tool offset in three dimensions based on the axis move commands and the I, J, and K values which specify the tool offset direction.

5-1. Three-dimensional Tool Offset Start-up

[Programming format]

The three-dimensional offset mode will begin when the following command is executed during the cancel mode for tool offset (cutter radius compensation, three-dimensional tool offset):

G44 Xp__ Yp__ Zp__ I__ J__ K__ D__

[Details]

- The three-dimensional space where the offset will be applied is determined by the axis addresses (Xp, Yp, Zp) specified in this start-up block. If axis addresses are not specified in the start-up block, the default settings that apply are the X-, Y-, and/or Z-axes.

Example:

```
G44 X__ I__ J__ K__          X, Y, Z space
G44 U__ V__ Z__ I__ J__ K__ U, V, Z space
G44 W__ I__ J__ K__          X, Y, W space
```

Where

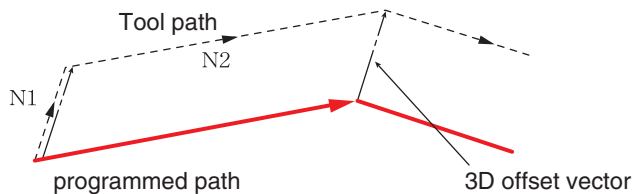
Xp is the X-axis or its parallel axis, U

Yp is the Y-axis or its parallel axis, V

Zp is the Z-axis or its parallel axis, W

ME33018R1000800450001

- If none of the X, Y, and Z axes are specified, the axes move by the tool offset amount.

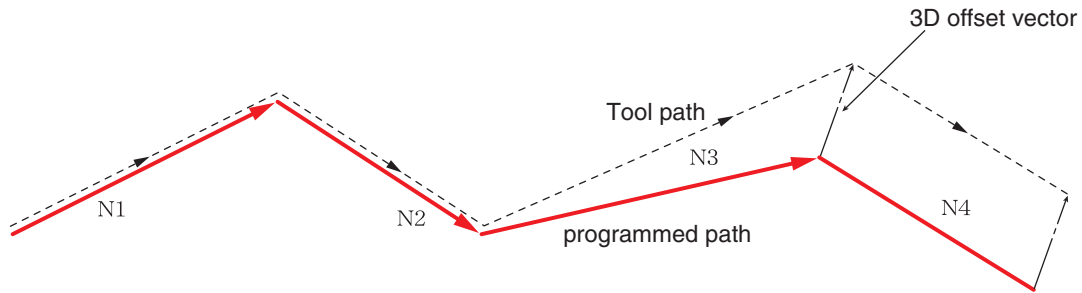


```
N1 G44 I__ J__ K__
N2 X__ Y__ Z__
```

ME33018R1000800450002

- At the start-up, if any of the values for I, J, and K is not specified, "0" is assumed for the address(es).
- The tool offset number, or compensation amount, is specified by a D command, just as with the cutter radius compensation function.
- The standard three-dimensional tool offset numbers are D00 to D100, and this can be expanded to D200 or D300.
- If no D command is specified, the previously specified D number will be used. If there is no previously specified D command, D00 will be assumed to apply.

- Even with a tool offset amount of zero (D00), the three-dimensional offset mode will be started, but no offset movement will take place.



```

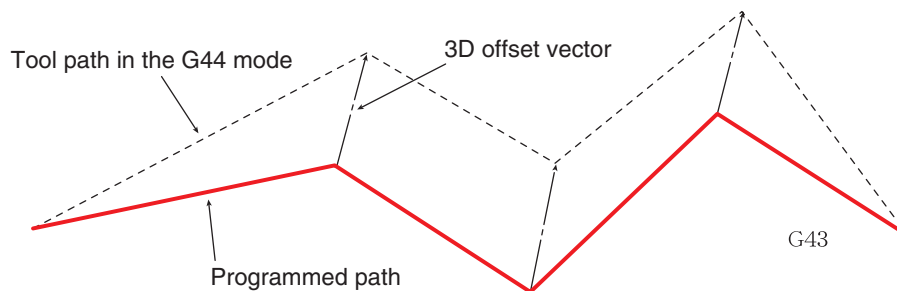
N1 G44 X__Y__Z__I__J__K__D00
N2     X__Y__Z__
N3     X__Y__Z__D01
N4     X__Y__Z__

```

ME33018R1000800450003

5-2. Three-dimensional Tool Offset Vector

In the three-dimensional tool offset mode, a three-dimensional offset vector is generated at the end of each block, as shown in the illustration below.



ME33018R1000800460001

[Function]

A three-dimensional offset vector is expressed as indicated below.

$$\text{Xp-axis vector component: } V_x = \frac{i \times \gamma}{p}$$

$$\text{Yp-axis vector component: } V_y = \frac{j \times \gamma}{p}$$

$$\text{Zp-axis vector component: } V_z = \frac{k \times \gamma}{p}$$

ME33018R1000800460002

i, j, k: Values specified by the addresses I, J, and K.

γ : Cutter radius compensation amount that corresponds to the cutter radius offset number specified by D

p: Value set for a parameter. When "0" is set for the parameter, the value of p is obtained by the following formula.

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

ME33018R1000800460003

Default: 0

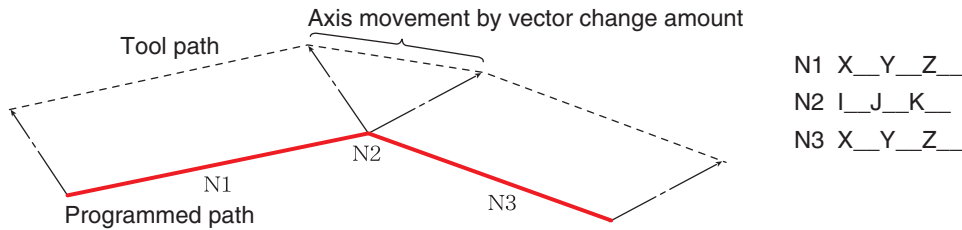
Setting range: 0 to ± 99999.999 mm or 0 to ± 3937.0078 inches

Parameter: NC optional parameter (long word) No. 7

[Details]

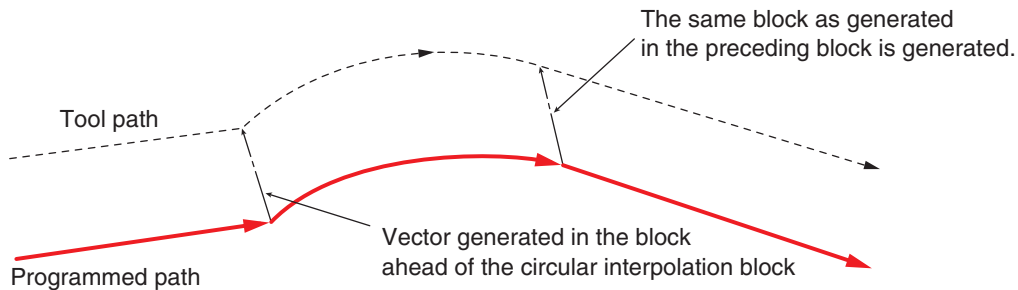
In a block where none of I, J, and K is specified, the same vector as the one generated in the previous block is generated.

- In a block where one of I, J, and K is not specified, a vector with a zero component in the omitted direction is generated.
- If no axis movement command (Xp, Yp, and Zp) is specified although I, J, and K are specified, only the vector is updated and the axes move according to the vector changes.



ME33018R1000800460004

- I, J, and K specified in a G02 or G03 block (arc or helical cutting) are used to indicate the coordinates of the center of that arc. In this case, the same vector as the one generated in the preceding block is generated.



ME33018R1000800460005

- At the start-up of the three-dimensional offset, if the axis name of a parallel axis is used for the axis names that define the three-dimensional space, the three-dimensional offset is not valid for the basic axis of the specified parallel axis. This designation does not cause an alarm to occur.

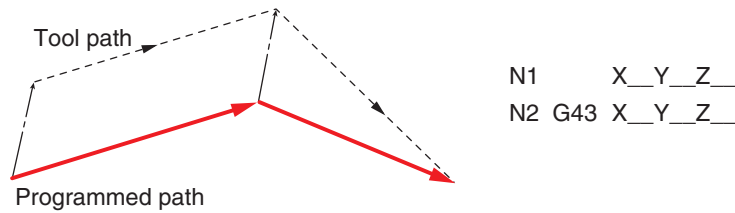
N1 G44 W__ I__ J__ K__ Specifies XYW space
N1 X__ Y__
N3 Z__ Z-axis is not offset.

ME33018R1000800460006

5-3. Canceling Three-dimensional Tool Offset

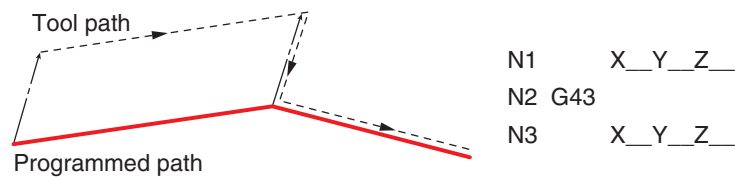
G43 is used to cancel the three-dimensional tool offset mode.

- a. Canceling in a block with axis commands



ME33018R1000800470001

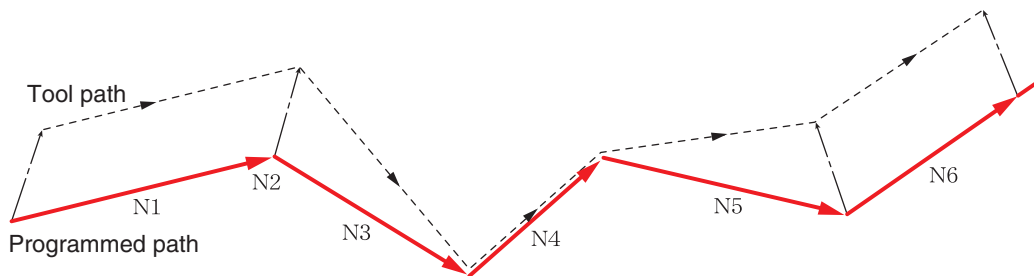
- b. Canceling in a block without other commands



ME33018R1000800470002

- c. Setting tool offset number to D00 (zero offset amount)

If an axis movement command is executed by changing the tool offset number to D00, which calls for zero offset amount, although the axes move in the same manner as when canceling the three-dimensional offset mode, the three-dimensional offset mode is not canceled.



```
N1 X__Y__Z__
N2 D00
N3 X__Y__Z__
N4 X__Y__Z__
N5 X__Y__Z__D01
N6 X__Y__Z__
```

ME33018R1000800470003

If D01 is specified independently in block N5, the axes will not move.

5-4. Actual Position Data Display And Feedrate

The coordinate values on the position display screen indicate the actual tool path after offset. The feedrate is controlled so that the axes move along the offset tool path at the programmed feedrate.

5-5. Relationship with Other G Functions

- G codes that must not be specified in the three-dimensional tool offset mode.
G15, G16, G40, G41, G42, G92
G codes for area machining
G codes for coordinate system parallel shift/rotation
G codes calling a fixed cycle
- The three-dimensional tool offset command must not be specified in the fixed cycle mode.

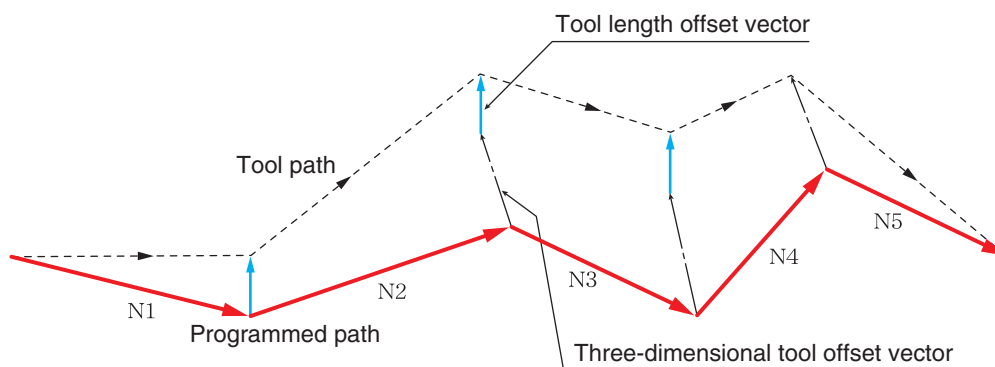
5-6. Relationship to Other Tool Offset Functions

(1) Tool length offset (G53 - G56)

The three-dimensional tool offset mode may be executed in the tool length offset mode; and the tool length offset mode may be executed in the three-dimensional tool offset mode. In these cases, tool length offset and three-dimensional tool offset are executed simultaneously.

Example:

Three-dimensional tool offset command is specified in the tool length offset mode.



N1	G56 X__Y__Z__H01	... Tool length offset mode
N2	G44 X__Y__Z__I__J__K__D02	... Three-dimensional tool offset mode
N3	X__Y__Z__	
N4	G53 X__Y__Z__	... Cancels tool length offset mode
N5	G43 X__Y__Z__	... Cancels three-dimensional tool offset mode

ME33018R1000800500001

(2) Cutter radius compensation (G40 to G42)

An alarm occurs if a cutter radius compensation command (G41 or G42) is specified in the three-dimensional tool offset mode, or if a three-dimensional tool offset command (G44) is specified in the cutter radius compensation mode. Therefore, these two offset functions must not be performed simultaneously.

SECTION 7 FIXED CYCLES

A fixed cycle refers to the function which can define a series of operations executed along the tool in-feed axis (hereafter referred to as the cycle axis), like drilling, boring and tapping, by one block of commands. When repeating the same operation at multiple positions, only the hole positions should be specified in a program. The following explanation assumes that the X-Y plane is the positioning plane and the Z-axis is the cycle axis. For operations performed on other planes using a different cycle axis, the principle described below is also applicable.

1. Table of Fixed Cycle Functions

G Code	Function	Spindle Rotation at Positioning Point	Hole Machining Operation	Operation at Hole Bottom Level	Retraction Operation	Spindle Rotation at Return Level
G71	Specifies the return level	—	—	—	—	—
G73	High speed deep hole drilling	CW	Pecking feed	Dwell	Rapid feed	CW
G74	Reverse tapping	CCW	Cutting feed	CW after dwell	Cutting feed	CW after dwell
G76	Fine boring	CW	Cutting feed	After the dwell, the tool bit is moved away from the bored surface. Then, the spindle stops at a specified position and shifts to the direction opposite that of the tool bit	Rapid feed	CCW after shifting to tool bit direction
G80	Cancel fixed cycle mode	—	—	—	—	—
G81 G82	Drilling	CW	Cutting feed	Dwell	Rapid feed	CW
G83	Deep hole drilling	CW	Pecking feed	Dwell	Rapid feed	CW
G84	Tapping	CW	Cutting feed	CCW after dwell	Cutting feed	CW after dwell
G85 G89	Boring	CW	Cutting feed	Dwell	Cutting feed	CW
G86	Boring	CW	Cutting feed	Stop after dwell	Rapid feed	CW
G87	Back boring	(*)	Cutting feed	After the dwell, the tool bit is moved away from the bored surface. Then, the spindle stops at a specified position and shifts to the direction opposite that of the tool bit	Rapid feed	CW after shift to tool bit direction

(*) After orientation, the spindle shifts to the direction opposite that of the tool bit and moves up to level R at a rapid feedrate. After shifting to the tool bit direction, the spindle rotates forward.

M codes used to select the return level:

M52 Return to the upper limit level

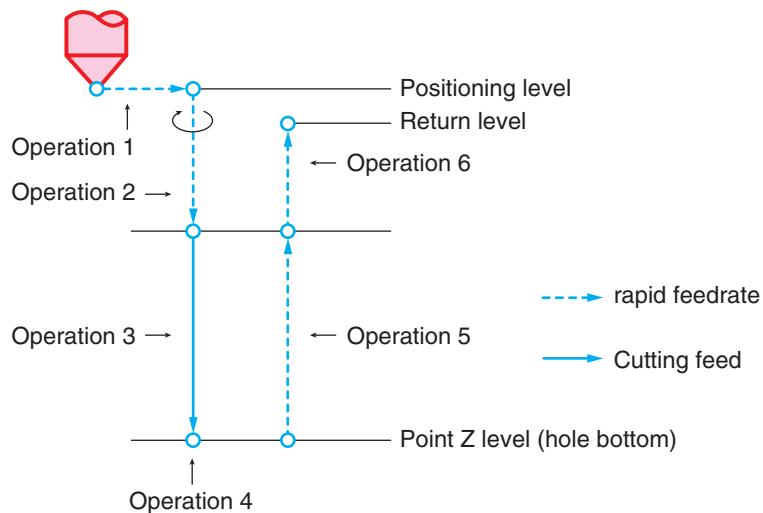
M53 Return to the specified point level set by G71

M54 Return to the point R level

2. Fixed Cycle Operations

All fixed cycle functions are composed of the following six operations:

- Operation 1 Positioning to hole machining position (rapid feedrate)
- Operation 2 Rapid feed to point R level
- Operation 3 Hole machining
- Operation 4 Operation at hole bottom position (point Z level)
- Operation 5 Retraction to point R Level
- Operation 6 Rapid feed to the return level (upper end or specified point level)



ME33018R1000900030001

Operation 1 is referred to as the positioning operation and operations 2 to 6 are referred to as the cycle axis operation.

Fixed cycles including a rotary axis may be executed.

2-1. Determining the Positioning Plane and the Cycle Axis

(1) Determining the positioning plane and the cycle axis by commands

The positioning plane may be determined by selecting a plane using G17, G18, and G19. The cycle axis is then chosen as the axis which is vertical to the selected positioning plane or the axis parallel to it.

G Code	Positioning Plane	Cycle Axis
G17	Xp-Yp plane	Zp
G18	Zp-Xp plane	Yp
G19	Yp-Zp plane	Xp

Xp = X- or U-axis

Yp = Y- or V-axis

Zp = Z- or W-axis

Due to the nature of the cycle axis as described above, once the positioning plane is determined, only two axes can be selected as the cycle axis. To determine the cycle axis to be used, specify the address of the desired axis in the block that contains a fixed cycle G code (G73 - G89).

(Wrong) G17 X__Y__

G81 X__Y__Z__W__R__F__

→ An alarm occurs since the cycle axis cannot be determined (two possible axes, Z and W, are specified).

(Correct) G17 X__Y__

G81 X__Y__U__V__W__R__

→ W-axis is selected as the cycle axis.

→ Movements in the positioning plane

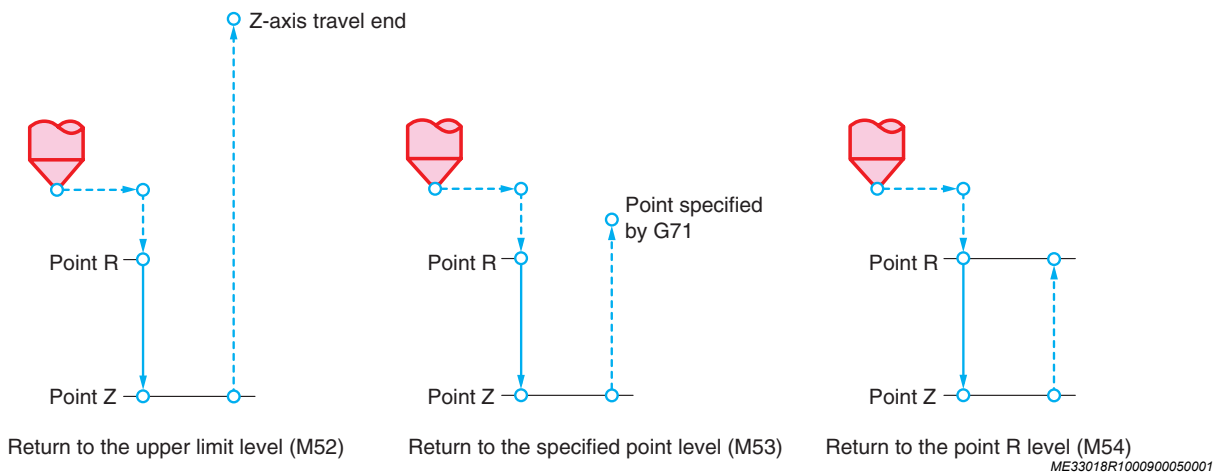
ME33018R1000900040001

(2) Method to determine by a parameter

It is possible to set the Z-axis as the cycle axis constantly by the setting of "Cycle axis" of the NC optional parameter (fixed cycle). In this case, only the command by G17 is effective for the positioning plane.

2-2. Controlling the Return Level

There are three different tool return levels when one fixed cycle operation ends; Return to the upper limit level (M52), Return to the specified point level (M53), Return to the point R level (M54).



- Selection of the upper limit level (M52) is possible only when the Z-axis is chosen as the cycle axis and when operation is performed in the negative direction of the Z-axis. If the upper limit is selected as the return level, the Z-axis returns to a point 0.1 mm (0.004 in.) away from the travel limit in the positive direction.
- To select the specified point level (M53), it is necessary to define this level in advance using G71.
- For back boring (G87) operation, a return to the point R level (M54) is not possible. If M54 is selected at this time, a tool returns to the starting point instead of point R.
- M52 is a one-shot command, which is valid only in a programmed block, and takes priority over M53 and M54. Since both M53 and M54 are modal, one of them is always valid. When the power is turned ON or after the NC is reset, the M54 mode (point R level return) is set.

2-3. Fixed Cycle Mode

- A fixed cycle mode is established by executing a hole machining definition command (G73 to G76 and G81 to G89). The fixed cycle mode is canceled when one of the following G codes is executed: G00, G01, G02, G03 and G80
- If a hole machining definition command is executed in the fixed cycle mode, the hole machining data is updated according to the newly executed hole executed hole machining command and the fixed cycle mode remains active.

- When the fixed cycle mode is canceled by G80, the interpolation mode (G00, G01, G02, G03, or G60) valid before entering the fixed cycle mode is restored and M05 is generated.

Example:

```

G17
G18 X__Y__
G81 X__Y__Z__R__F__
      X__
      X__Y__
      } Fixed cycle mode
G80                                     Fixed cycle mode cancel
XY..... Same as G01 X__ Y__

```

* : Axis movement commands, which are specified with G80 in the same block, are executed only after the fixed cycle mode has been canceled.

ME33018R100090060001

2-4. Cycle Operation Conditions

In the fixed cycle mode, the cycle axis operates in the following blocks:

- Hole machining definition command block which defines operations of the cycle axis.
If the hole position commands are omitted, the position where the axes are presently located is regarded as the hole position and the cycle axis will operate at this position.
- Blocks between the hole machining definition command block and the G80 block which contain the hole position data of at least one axis.

Example:

```

Fixed cycle mode {
G17 G00 X__Y__ ..... Hole position data is omitted, so drilling is carried out at the present position.
G18 Z__R__F__
      (empty block)
      } ..... Drilling cycle is not performed, since the above condition (2) is not met.
F__
M__
G04 F__
X__ ..... Drilling cycle is performed at this point, since the above condition (2) is not met.
G80 X__Y__ ..... Drilling cycle is not performed at this block, since the fixed cycle mode
                                     is canceled.

```

ME33018R100090070001

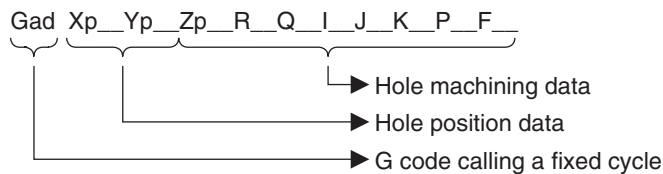
Even in the blocks that satisfy the above conditions (1) and (2), the cycle axis does not move in the following cases:

- NCYL (NO CYCLE)**
If NCYL is specified in the fixed cycle mode, positioning to the defined hole position is performed, but the cycle axis does not operate.
- NOEX (NO EXECUTE)**
If NOEX is specified in the fixed cycle mode, no axis movements may be performed. Both NCYL and NOEX are specified in conjunction with the coordinate computation function.

3. General Rules for Programming Fixed Cycles

This section describes the general rules on specifying hole machining data which is specified in blocks containing a fixed cycle call G code, G73 to G76 and G81 to G89. The following explanation assumes that the positioning plane is the Xp-Yp plane and the cycle axis is the Zp-axis.

3-1. Programming Format (General Command Format)



ME33018R1000900090001

(1) G Code Calling a fixed cycle

Once executed, a fixed cycle G code remains valid until it is canceled by a cycle cancel G code (G80, G00 - G03) or replaced by another fixed cycle G code. For details, refer to "Table of fixed cycle Functions".

(2) Hole Position Data

The hole position data specifies the position at which the fixed cycle (hole machining) is executed.

The addresses used for specifying the position are determined by the plane selection G code (G17, G18 and G19) which defines the positioning plane.

The hole position data may be given in the absolute (G90) or incremental mode (G91) as needed.

Axes move to the hole machining position at a rapid feed rate (G00) mode. If G60 (unidirectional positioning) mode is specified, positioning is made in the unidirectional positioning mode.

(3) Hole Machining Data

- Once programmed, the hole machining data remains valid until it is changed or the fixed cycle mode is canceled.
- Zp: Specifies the hole bottom position.
In the absolute programming mode (G90), the hole bottom position should be specified as an absolute value. In the incremental programming mode (G91), the distance between the point R level and the hole bottom should be specified.
- R: Specifies the point R level.
In the absolute programming mode (G90), the point R level should be specified as an absolute value. In the incremental programming mode (G91), the distance from the tool position where the fixed cycle mode starts to the point R level should be programmed.
- Q:
 - a. G73, G83 mode
Specifies the depth of cut per in-feed motion, as a positive, incremental value.
 - b. G76, G87 mode
Specifies the shift amount as a positive, incremental value.
(If a negative value is specified, the sign is disregarded.)

- c. G74, G84 mode
 Specifies the dwell period at the point R level.
 The relationship between the length of time and the value to be specified is the same as that for G04.

If a negative value is set in the above mode a. or b., the NC ignores the negative sign.

When changing the fixed cycle function, check if a Q value specified for another function remains.

I, J, K: Specifies the shift amount in the G76, G87 mode, in an incremental value.

The addresses used to specify a shift amount depend on the selected positioning plane as shown below.

G Code	Positioning Plane	Addresses
G17	Xp-Yp plane	I, J
G18	Zp-Xp plane	K, I
G19	Yp-Zp plane	J, K

P: Specifies the dwell time period.

The relationship between the length of time and the value to be specified is the same as that for G04.

F: Specifies the feedrate applied to the cutting feed areas in a cycle axis operation.

The programmed feedrate is valid in all interpolation commands and remains valid even after the fixed cycle mode is canceled until a new feedrate is specified.

[Supplement]

The positional relationship among the actual position, return point, point R level, and point Z level is as indicated below.

- In the M52 and M53 modes, the infeed direction is judged from point R and point Z levels, and it is checked whether or not the direction of retraction is the reverse of the infeed direction. (The relation to the actual position is not checked.)
- In the M54 mode, positional relationship is not checked.

3-2. Command Items Necessary for Fixed Cycle Function Commands

The table below shows the command items that must be specified for the individual fixed cycles.

Command Item		Hole Position		Point Z	Point R	Pecking	Shift Amount		Dwell Time		Feedrate	Retraction	
		ABS/INC		ABS/INC	ABS/INC	Amount	INC		-		-	INC	
		Function	Address	X	Y	Z	R	Q	Q	I,J,K	P	Q	F
G73	High speed, deep hole drilling	(A)	(A)	(B)	(B)	(C)				(C)		(B)	
G74	Reverse tapping	(A)	(A)	(B)	(B)					(C)	(C)	(B)	
G76	Fine boring	(A)	(A)	(B)	(B)			(C)	(C)	(C)		(B)	d2
G81 G82	Drilling	(A)	(A)	(B)	(B)					(C)		(B)	
G83	Deep hole drilling	(A)	(A)	(B)	(B)	(B)				(C)		(B)	
G84	Tapping	(A)	(A)	(B)	(B)					(C)	(C)	(B)	
G85 G89	Boring	(A)	(A)	(B)	(B)					(C)		(B)	
G86	Boring	(A)	(A)	(B)	(B)					(C)		(B)	
G87	Back boring	(A)	(A)	(B)	(B)			(C)	(C)	(C)		(B)	

ME33018R1000900100001

*: The positioning plane and the cycle axis are assumed to be the X-Y plane and Z-axis respectively.
[Details]

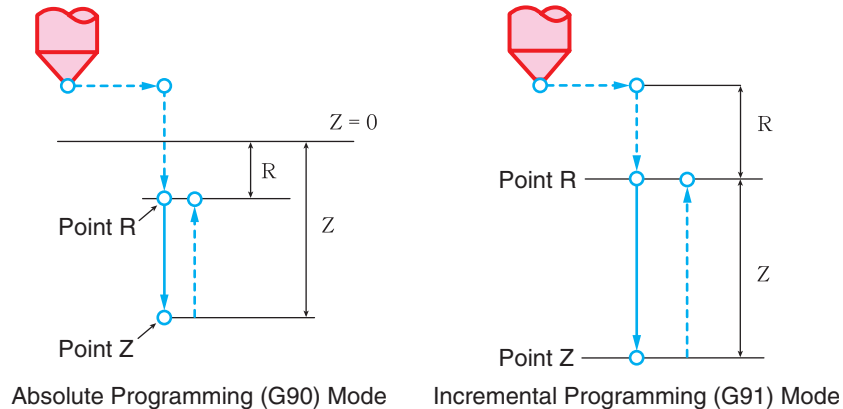
- “ABS/INC” indicates that either the absolute (G90) or the incremental command (G91) may be selected for specifying a value.
“INC” indicates that a value should be specified as an incremental (G91) amount, regardless of the specified dimensioning mode.
- “(A)” indicates that the command item may be omitted from the fixed cycle program block. The most recently entered values will be used as the default values.
- “(B)” indicates that the command item may be omitted from the fixed cycle program block, but must be specified at some point before cycle axis operation.
- “(C)” indicates that the command item may be omitted from the fixed cycle program block. However, if a command item has been specified in a previous fixed cycle program block, the previously specified value will remain in effect.
- The shift amount should be specified using Q, or a combination of I, J, and/or K. The amount specified by Q is valid only when the cycle axis is at the Z-axis, by parameter setting.
- If an address of a blank column is specified, it will be saved to be used as the hole machining data for other fixed cycles.
- All data with the exception of retraction amounts (d1 and d2), which are set for parameters, and feedrate (F), are valid only in the fixed cycle mode.
The retraction amount values, d1 and d2, are cleared when the fixed cycle mode is canceled or the NC is reset. The feedrate, F, is cleared when the NC is reset.
- The values for P, Q, I, J, and K must be specified in a block where a cycle axis operation is performed. If one of these values is specified in a block without a cycle axis operation, the specified value is not treated as modal data.

- The shift amount must be specified for the fixed cycle called by G76 and G87, otherwise an alarm occurs.

3-3. Absolute Programming Mode and Incremental Programming Mode

(1) Specifying point R and point Z

How the points R and Z are defined differs depending on the selected dimensioning mode, absolute (G90) or incremental (G91). The hole machining data should also be programmed in accordance with the dimensioning mode selected when it is defined.



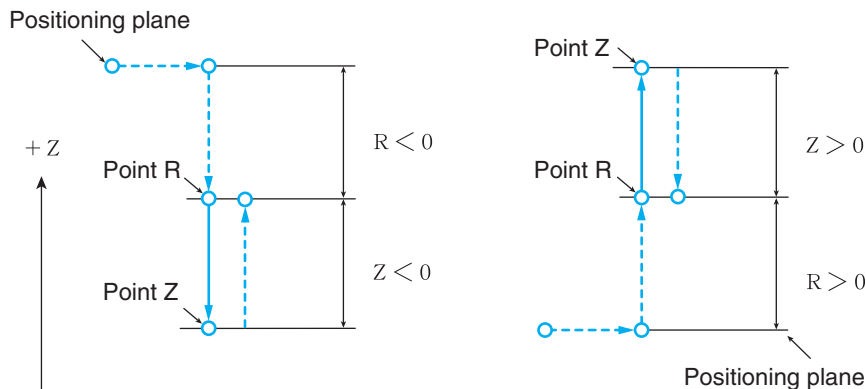
ME33018R1000900110001

(2) Interpretation of incremental commands

In the incremental programming mode, the sign of R and Z values has significant meaning. Generally, the cycle axis is first moved in the negative direction (infeed) and then in the positive direction (retraction). However, it is possible to move the Z-axis in the opposite direction in reference to the positioning plane. When programming a fixed cycle in the incremental programming mode, the direction of cycle axis movement can be determined by assigning the proper sign to the R and Z values.

Example:

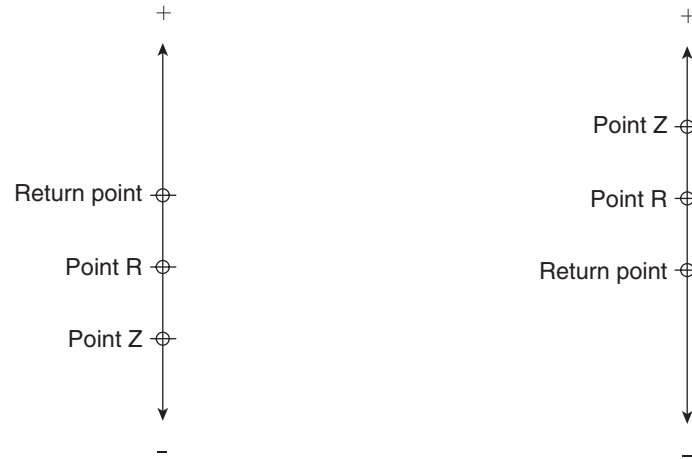
Cycle axis movement direction in the incremental programming mode



ME33018R1000900110002

3-4. Positional Relationship among Return Point Level, Point R Level and Point Z Level

The positional relationship among the three levels along the cycle axis direction must comply with one of the two cases shown below. (The only exception is G87 back boring, where the point R and point Z levels are exchanged.)



ME33018R1000900120001

3-5. Axis Shift

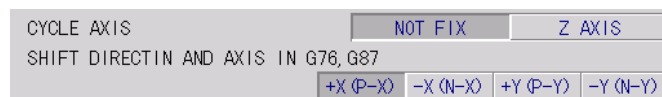
In the G76 (fine boring) and G87 (back boring) modes, an axis shift is executed at a rapid feedrate. The shift amount and direction may be set using one of the following two methods.

(1) Q Command

The shift amount and direction may be set using a Q command only when Z axis is chosen as the cycle axis at cycle axis of NC optional parameter (fixed cycle).

The value of Q is a positive, incremental value. Q commands are given priority over I, J, and K commands, explained below in item (2). Therefore, if a Q command is specified, the shift amount specified by I, J, and/or K commands is ignored. The direction of shift should be set in advance at shift direction and axis in G76, G87 of NC optional parameter (fixed cycle).

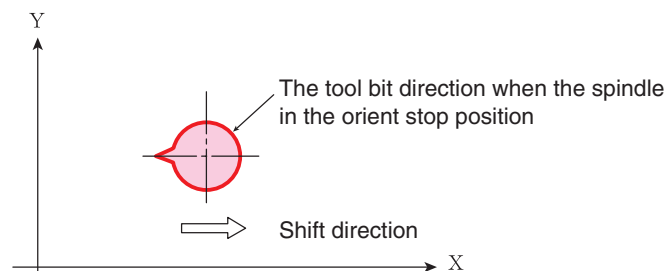
The shift direction is defined in the machine coordinate system and is the opposite of the direction of the bit when the spindle stops in the orient stop position.



ME33018R1000900130001

Example:

“+X” is set for the shift direction setting parameter



ME33018R1000900130002

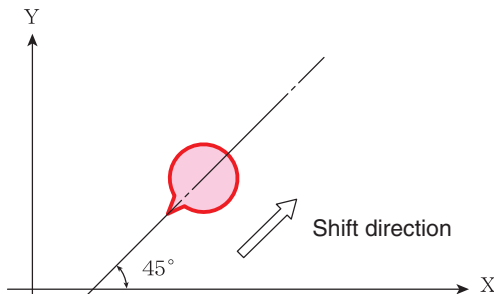
(2) I, J, and K Commands

I, J, and K commands are used when the cycle axis is not fixed by parameter.

The shift amount and direction of the tool should be specified using (I, J), (K, I), or (J, K) depending on the selected positioning plane. The shift direction is defined in the machine coordinate system.

Example:

If a value is specified for either of I, J, K command pair, the NC regards the other command value is 0.



If "I1.0 J1.0" is specified, the tool shifts in the direction indicated by the arrow symbol by $\sqrt{1^2+1^2} = 1.414$.

ME33018R1000900130003

3-6. Z-axis G01 Mode Return Function

In the high speed deep hole drilling cycle (G73) and the deep hole drilling cycle (G83), axis return operation in the Z-axis direction can be executed either in the G00 mode or the G01 mode by making the appropriate parameter setting.

(1) Switching between G00 and G01

Choose the Z-axis return mode between G00 and G01 by the setting at Z-AXIS RETURN OPERATION in G73, G83 of the NC optional parameter (FIXED CYCLE).

(2) Feedrate for Z-axis return in the G01 mode

Determine the Z-axis return feedrate in G01 mode by the setting at FEEDRATE FOR Z-AXIS RETURN IN THE G01 of the NC optional parameter (FIXED CYCLE).

Setting range	: 1 to 20000
Initial value	: 20000 (1 mm/min)
Unit	: mm/min

[Supplement]

The maximum value for the feedrate in the G01 mode differs according to the machine specification. Therefore, the following alarm may occur depending on the value set at FEEDRATE FOR Z-AXIS RETURN IN THE G01 of the NC optional parameter (FIXED CYCLE).

4204 ALARM-D Feedrate command limit over (replacing)

If this alarm occurs, set the value at FEEDRATE FOR Z-AXIS RETURN IN THE G01 of the NC optional parameter (FIXED CYCLE) again to match the specifications of the machine.

Since the cutting feed override is valid, the above alarm may also occur depending on the override value; if this happens the override value should be reduced.

3-7. Relationships between Fixed Cycle Functions and Other Functions

(1) Axis Movement Call Mode (MODIN, MODOUT)

If the fixed cycle mode and the axis movement call mode overlap, the MODIN command will call axis movements after the cycle axis operation has been completed.

(2) Uni-direction Positioning

In the uni-direction positioning mode (G60), the cycle axis operates as follows:

- Positioning in the positioning plane is executed in the uni-direction positioning mode.
- Positioning of the cycle axis is executed in the two-directional positioning mode.
G60 may be specified in a fixed cycle mode. In contrast to other interpolation mode call commands (G00 to G03), G60 does not cancel the active fixed cycle mode. The G60 mode specified in the fixed cycle mode is, however, canceled at the same time the fixed cycle mode is canceled by G80.

(3) Geometry Enlargement/Reduction Function

If a fixed cycle mode is specified in the geometry enlargement/reduction mode, the following factors are not subjected to the enlargement/reduction processing.

- Pecking amount (Q) and retraction amounts (d1 and d2) in the G73 and G83 modes
- Shift amount (Q, or I, J, and K) in the G76 and G87 modes

(4) The following items cannot be specified in a fixed cycle mode:

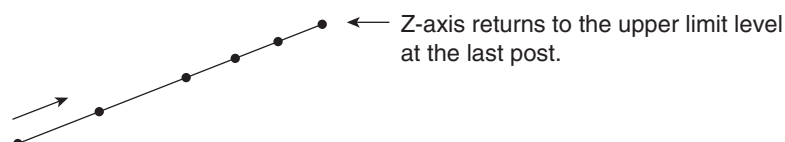
- Plane selection which changes the positioning plane
- Cutter radius compensation (G41, G42)
- Three-dimensional tool offset (G43)
- Tool change (M06)
- Area machining function
- Home position command (G30)

(5) Coordinate Calculation Function

If a coordinate calculation function command is specified with M52 in a fixed cycle mode, the Z-axis returns to the upper limit level at the last point obtained by coordinate calculation.

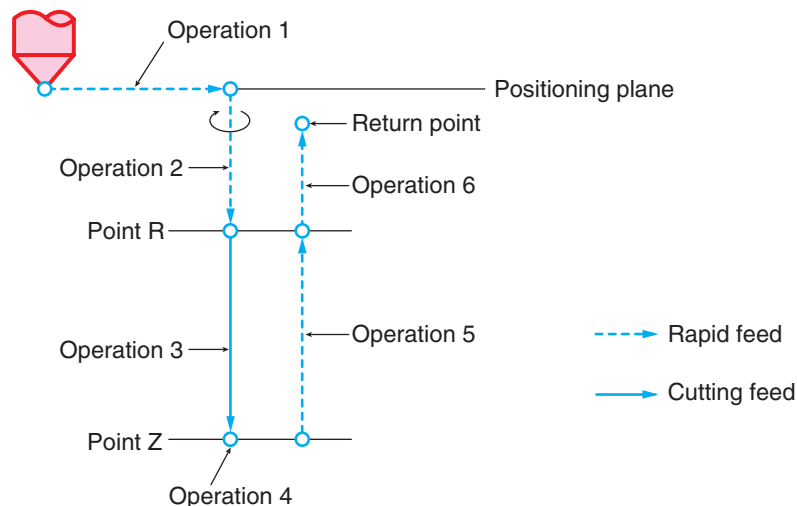
Example:

```
G81X__Y__Z__R__F__
LAAX__Y__I__K__I__K__J__M52
```



3-8. Notes for Programming a Fixed Cycle

- In a fixed cycle (G74, G84, G86) mode in which spindle rotation is controlled, if a hole machining cycle is consecutively executed for holes arranged in short intervals with short distance between the specified point level and the point R level, the spindle may fail to reach the programmed speed before starting the machining of hole. In this case, it is necessary to enter a dwell command (G04) in between the hole machining operations.
- A fixed cycle mode can be canceled by specifying G00 to G03 without using G80. If any of G00 to G03 is specified, the fixed cycle mode is canceled at the time the G code is read. However, if these G codes are specified with fixed cycle commands, the fixed cycle mode is not canceled. Canceling with Commands G00 through G03
Example:
G01 G73 X__ Y__ Z__ R__ P__ Q__ F__
(The specified fixed cycle is executed.)
- The fixed cycle mode and the programmed hole machining data are all cleared when the NC is reset.
- If a fixed cycle is executed in the single block mode, the cycle motion will stop at the end of operations 1 and 5 (M52, M53) or at the end of operation 6 (M54).



ME33018R1000900160001

It is also possible to stop the cycle after the completion of operation 2 in the illustration above, by selecting does at single block stop at point R level of NC optional parameter (fixed cycle). When selecting this type of operation, the following points must be taken into consideration.

- The point R level at which the Z-axis stops in single block mode operation is the point R level reached first after the start of a fixed cycle operation. It is not the point R level to which the Z-axis returns in each pecking operation in the G83 deep hole drilling cycle until the specified depth is reached.
- In the G87 back boring cycle, the point R level reached first is used as the start point for the hole bottom machining. The Z-axis stops at this point if the cycle is executed in the single block mode.
- If the point R level lies on the position plane, that is, if the positioning point and the point R level are at the same point, the Z-axis stops at the positioning point and the point R level. This means the Z-axis stops two times at the same point.

- If the slide hold function is turned on during a tapping cycle (G74, G84), cycle motion does not stop until the completion of operation 5, even though the SLIDE HOLD lamp lights immediately after pressing the SLIDE HOLD button. If it is pressed during operation 1, 2, or 6, however, cycle operation stops immediately.
- During the execution of G74 or G84 tapping cycle, cutting feedrate override is fixed at 100%. The rapid feed override is valid. For spindle operation, override value is also fixed at 100%.

4. Specification of Return-point Level (G71)

[Function]

The G71 command specifies the return-point level which is used for executing a fixed cycle in the M53 mode (return to the specified level).

[Programming format]

G71 Z__

Z: Indicates the cycle axis.

The coordinate value should be specified as an absolute value in the local coordinate system, regardless of whether the G90 or G91 mode (absolute or incremental) is selected.

[Supplement]

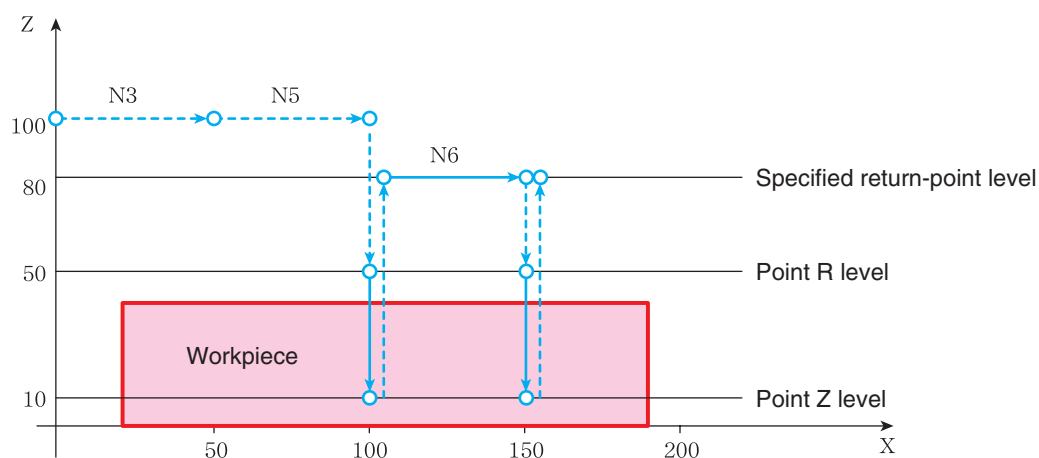
- 1) The return point level must be specified with G71 before M53 is specified.
- 2) When the NC is reset, the return point is undefined.
- 3) The return-point level value is modal for each axis, and is not changed until another value is set.
- 4) Even if the local and/or work coordinate system is changed, the specified return-point level value remains as it is.

[Example program]

```

N1 G90 X0 Y0 Z100.0
N2 G91
N3 G17 G00 X50.0 Y50.0
N4 G71 Z80.0
N5 G81 X50.0 Y50.0 Z-40.0 R-50.0 F100 M53
N6 X50.0 Y50.0
N7 G80

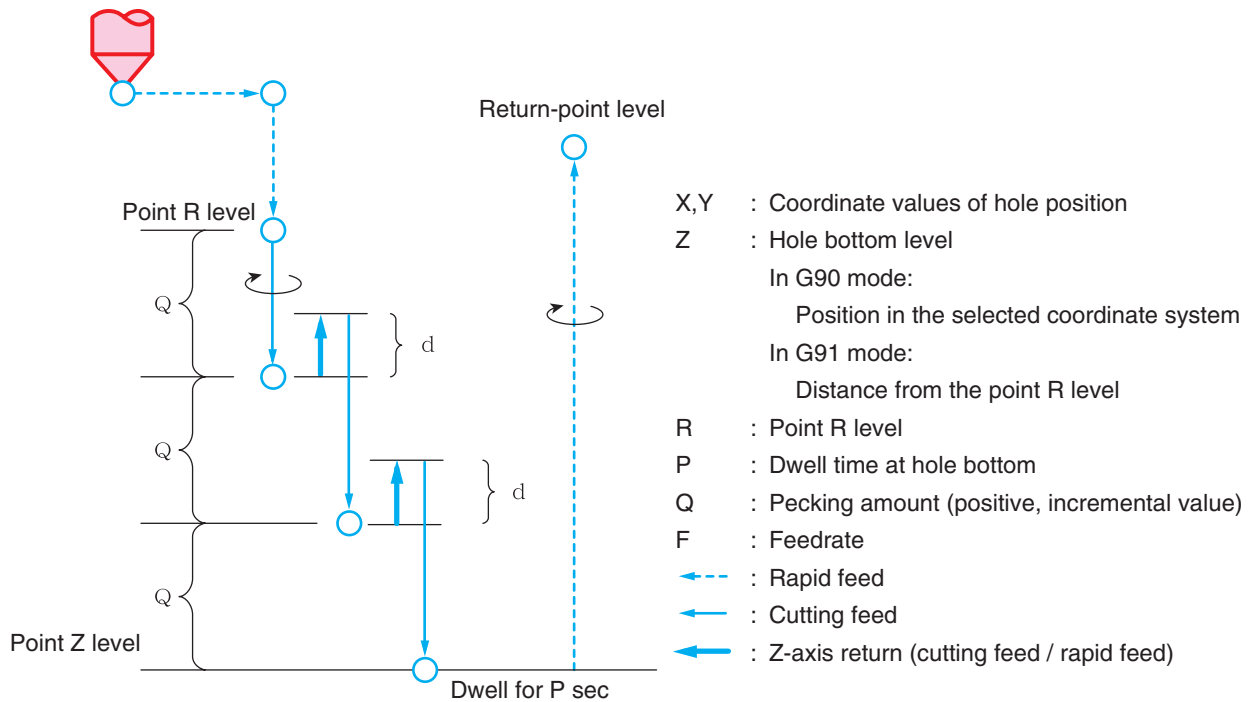
```



5. High Speed Deep Hole Drilling Cycle (G73)

[Programming format]

G73 X__Y__Z__R__P__Q__F__



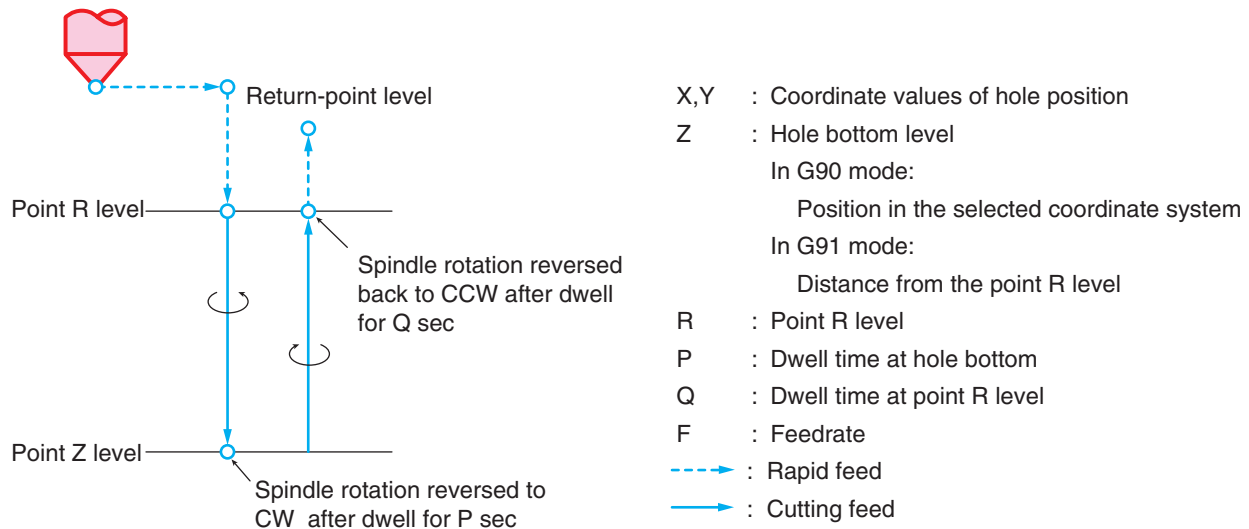
Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate
- (2) Positioning to the point R level at a rapid feedrate
- (3) Drilling by the pecking amount specified by Q at a cutting feedrate and with the spindle rotating in the forward direction
- (4) Cutting tool retraction by "d".
Set the retraction amount "d" at retraction in G73 cycle (high-speed deep hole) OR G83 CYCLE (DEEP HOLE) WITH I, J COMMAND of the NC optional parameter (fixed cycle).
- (5) Drilling to the point R level by repeating steps (3) and (4)
- (6) Dwelling at the point Z level for P seconds
- (7) Returning to the return-point level at a rapid feedrate

6. Reverse Tapping Cycle (G74)

[Programming format]

G74 X__Y__Z__R__P__Q__F__



ME33018R1000900190001

Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate
- (2) Positioning to the point R level at a rapid feedrate
- (3) Tapping to the point Z level at the specified cutting feedrate with the spindle rotating in the CCW direction.
- (4) Dwelling at the point Z level for P seconds, then reversal of the spindle rotating direction to the CW direction.
- (5) Returning to the point R level at a cutting feedrate
- (6) Dwelling at the point R level for Q seconds, then reversal of the spindle rotating direction back to the CCW direction.
- (7) Returning to the return-point level at a rapid feedrate.

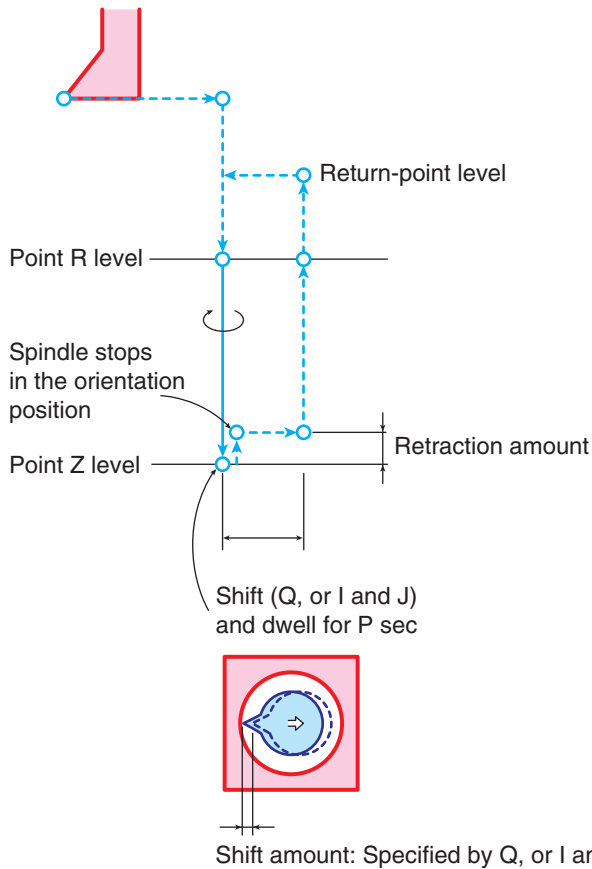
[Details]

- Dwell is not executed if a P and/or Q value is not specified.
The units of P and Q values are the same as used for the G04 mode dwell command.
- A feed override is disregarded during reverse tapping operation.
- If the SLIDE HOLD button is pressed during the return from the point Z level to the point R level, the cycle stops after the point R level is reached.
- If positioning to the next tapping point is executed at the point R level after the start of the spindle counterclockwise rotation but before the tapping tool is completely disengaged from the workpiece, enter a dwell at this level by specifying Q.
- Both the cutting feedrate override and the spindle speed override value are fixed at 100%.
A rapid feed override can be set.

7. Fine Boring (G76)

[Programming format]

G76 X__Y__Z__R__Q__(I__J__)P__F__



X,Y : Coordinate values of hole position
 Z : Hole bottom level
 R : Point R level
 Q : Shift amount } See the explanation below.
 I,J : Shift amount
 P : Dwell time at hole bottom
 F : Feedrate
 - - - : Rapid feed
 - - - : Cutting feed

ME33018R1000900200001

Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate
- (2) Positioning to the point R level at a rapid feedrate
- (3) Boring to the point Z level at the specified cutting feedrate with the spindle rotating in the forward direction
- (4) Dwelling at the point Z level for P seconds, retracting by the amount set at SHIFT DIRECTION AND AXIS IN G76, G87 of NC optional parameter (FIXED CYCLE), then spindle stop in the orientation position. After that, the tool shifts by the shift amount, Q, to the direction the tool bit moves away from the machined workpiece inner surface.
- (5) Returning to the return point level at a rapid feedrate
- (6) Tool shifts back in the bit direction by the shift amount, Q, then the spindle starts rotating in the clockwise direction.

[Details]

- Retraction amount at the point Z level
The amount the Z-axis retracts upward from the point Z level is set at SHIFT DIRECTION AND AXIS IN G76, G87 of the NC optional parameter (FIXED CYCLE).
- Shift amount
 - a. Q is used to specify the shift amount if the cycle axis is fixed as the Z-axis by the setting at SHIFT DIRECTION AND AXIS IN G76, G87 of the NC optional parameter (FIXED CYCLE). The value set must always be positive. The direction for shift motion, +X, -X, +Y, or -Y, should be set using a parameter beforehand. Note that a Q value is modal data and address Q is also used in the G73 and G83 cycles. A Q value is given priority over I and J values.
 - b. I and J are used to specify the shift amount when the plane is selected using G17, G18, or G19. The relationship between the plane selecting G code and the addresses to be used is shown below.
 - G17 I, J
 - G18 K, I
 - G19 J, K
 For addresses I, J, and K, all values are set as incremental values.
The shift direction is always defined in the machine coordinate system.
 - c. If the shift amount is not specified by Q, or I and J, an alarm occurs.

8. Fixed Cycle Cancel (G80)

[Function]

G80 cancels a fixed cycle mode (G73, G74, G76, G81 to G87, and G89). When G80 is executed, all hole machining defining commands including point R and point Z are canceled and the interpolation mode (G00 to G03, G60) valid before the fixed cycle mode was called is restored. At the same time, the M05 code (spindle stop command) is generated. If G00 or G01 is specified preceding the G80 block, however, M05 is not generated.

- Example program that does not generate M05


```
G81 X__Y__Z__R__F__
G00 X__Y__
G80
```
- Example program that generates M05


```
G81 X__Y__Z__R__F__
G80
G00 X__Y__
```

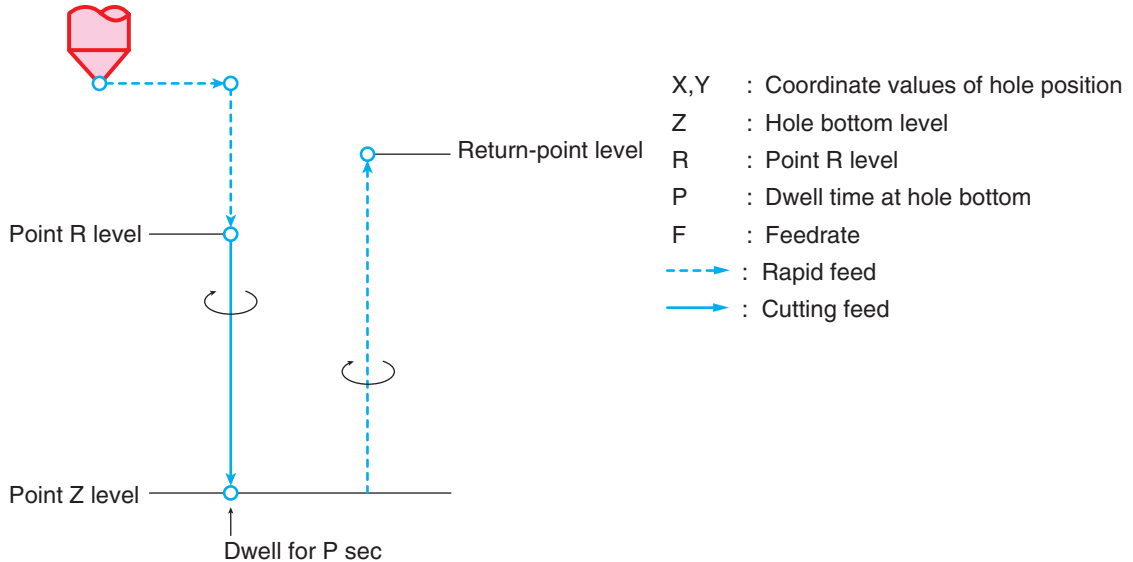
[Details]

- Fixed cycle modes are also canceled if an interpolation mode call G code (G00, G01, G02, G03), with the exception of G60, is specified.
- If axis movement commands are specified with G80 in the same block, the fixed cycle mode is canceled first and the axis movement commands are executed after that.

9. Drilling Cycle (G81, G82)

[Programming format]

```
G81 X__Y__Z__R__P__F__
G82 X__Y__Z__R__P__F__
```



ME33018R1000900220001

G81 and G82 are used in the same manner.

Machining Sequence

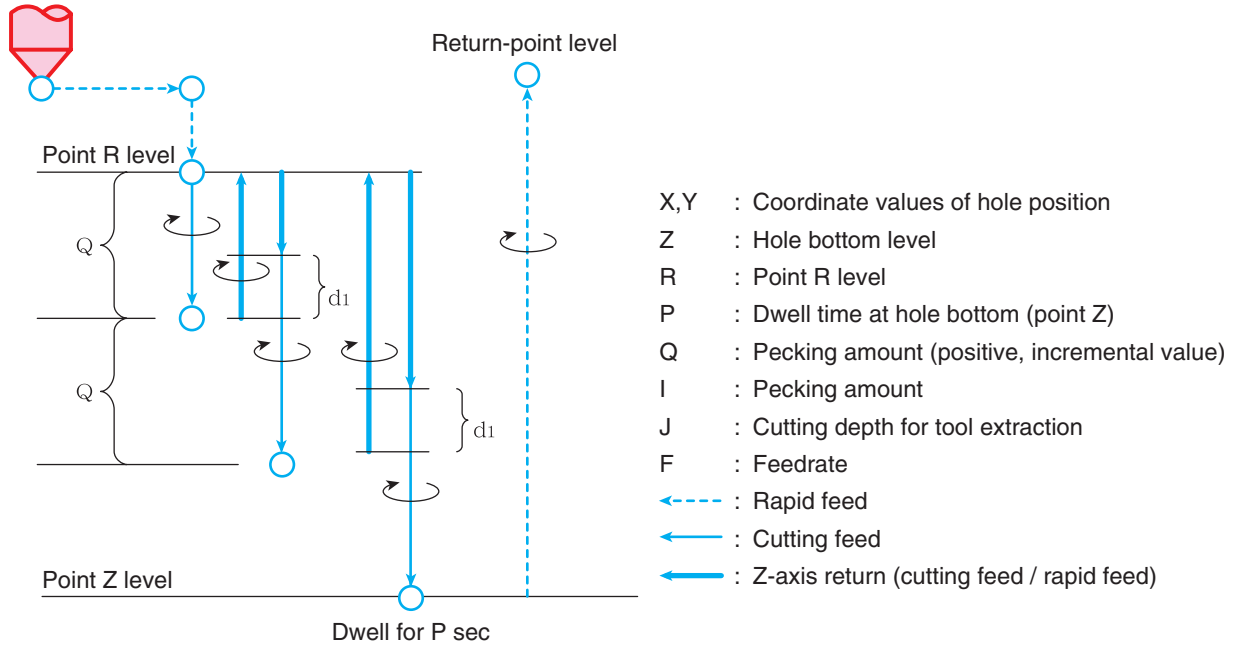
- (1) Positioning along the X- and Y-axis at a rapid feedrate
- (2) Positioning to the point R level at a rapid feedrate
- (3) Drilling to the point Z level at the specified cutting feedrate with the spindle rotating in the clockwise direction
- (4) Dwelling at the point Z level for P seconds.
- (5) Returning to the return point level at a rapid feedrate

10. Deep Hole Drilling Cycle (G83)

[Programming format]

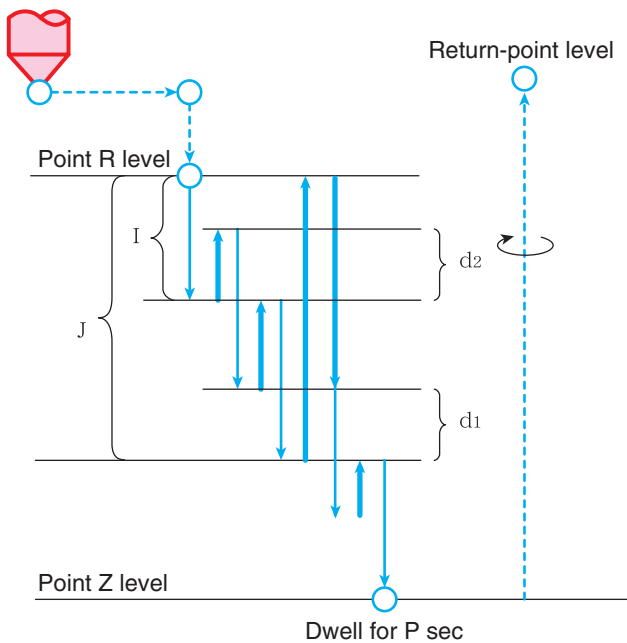
G83 X__Y__Z__R__Q__(I__J__)P__F__

- Programming using Q



ME33018R1000900230001

- Programming using I and J



ME33018R1000900230002

If a Q value is programmed in the same block as I and J values, the Q value will be given priority.

[Setting values]

Retraction amount d1:

Set at RETRACTION POSITIONING FROM LEVEL 'R' TO WORK IN G83 CYCLE (DEEP HOLE) of the NC optional parameter (fixed cycle).

Retraction amount d2:

Set at RETRACTION IN G73 CYCLE (HIGH-SPEED DEEP HOLE) OR G83 CYCLE (DEEP HOLE) WITH I, J COMMAND of the NC optional parameter (fixed cycle).

Machining Sequence

(1) Programming using Q

- a. Positioning along the X- and Y-axis at a rapid feedrate
- b. Positioning to the point R level at a rapid feedrate
- c. Drilling by pecking amount specified by Q at the specified cutting feedrate with the spindle rotating in the forward direction
- d. Returning to the point R level at a rapid feedrate
- e. In the second and later in-feed operations: Positioning at a level "d1" above the previously machined depth at a rapid feedrate and drilling by "Q + d1".
- f. Returning to the point R level at a rapid feedrate
- g. Drilling to the point Z level by repeating steps e). and f).

(2) Programming using I and J

- a. Positioning along the X- and Y-axis at a rapid feedrate
- b. Positioning to the point R level at a rapid feedrate
- c. Drilling by pecking amount I then retracting by d2. After that, drilling by "I + d2". The pecking and retraction cycle is repeated until depth J (tool extraction depth) is reached.
- d. Returning to the point R level at a rapid feedrate
- e. Positioning at a level "d1" above the previously machined depth at a rapid feedrate and drilling by "I + d1". Then step c). is repeated to machine by depth J.
- f. Returning to the point R level at a rapid feedrate
- g. Drilling to the point Z level by repeating steps e). and f).
- h. Dwelling at the point Z level by P seconds.
- i. Returning to the return point level at a rapid feedrate.

[Details]

According to the value of I, J, cutting is performed as follows. If either I or J is specified, the other one is considered as "0" is specified.

- No Q designation

J = 0 G73 cycle is executed, using pecking amount I.

I ≥ J G83 cycle is executed, using pecking amount J.

I < J Operation steps c). to i)., explained in item (2) above, are executed.

I = 0 An alarm occurs. (Alarm "Q")

ME33018R1000900230003

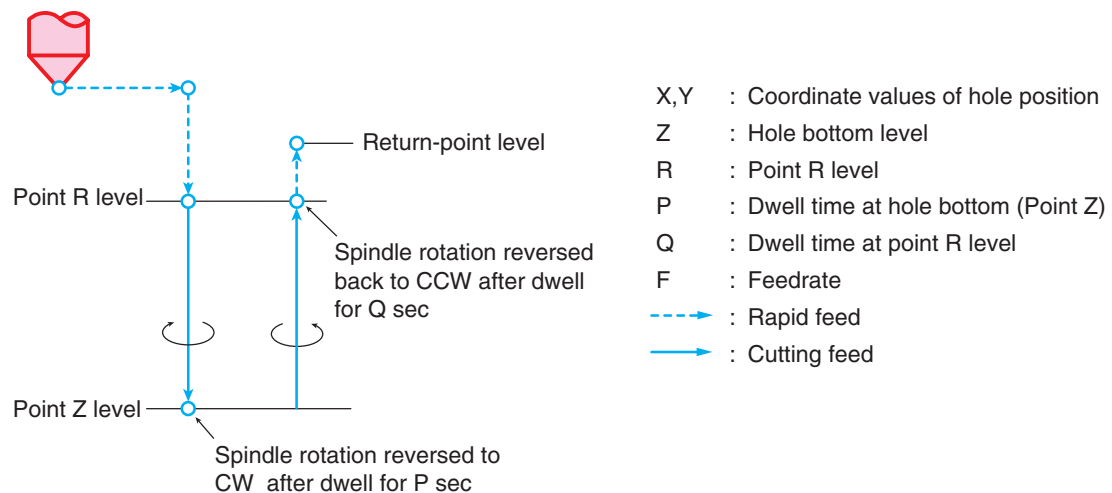
- Q designated with I and J in the same block

Operation steps c) to g), explained in (1) above, are executed disregarding I and J values.

11. Tapping Cycle (G84)

[Programming format]

G84 X__Y__Z__R__ (P__) (Q__) F__



ME33018R1000900240001

Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate
- (2) Positioning to the point R level at a rapid feedrate
- (3) Tapping to the point Z level at the specified cutting feedrate with the spindle rotating in the clockwise direction
- (4) Dwelling at the point Z level for P seconds, then reversal of the spindle rotating direction to the counterclockwise direction
- (5) Returning to the point R level at a cutting feedrate
- (6) Dwelling at the point R level for Q seconds, then reversal of the spindle rotating direction back to the clockwise direction
- (7) Returning to the return-point level at a rapid feedrate

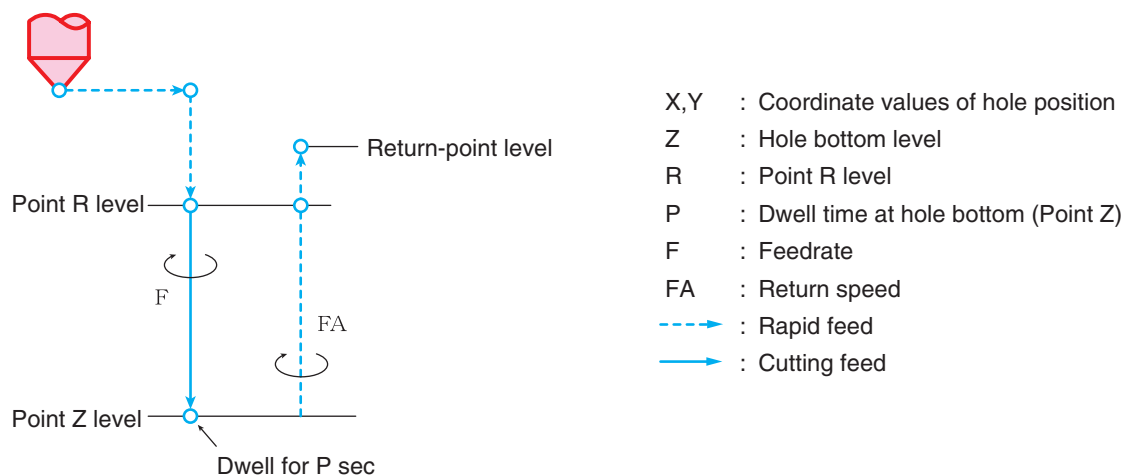
[Details]

- Dwell is not executed if a P and/or Q value is not specified.
Units of P and Q values are the same as used for the G04 mode dwell command.
- A feed override is disregarded during reverse tapping operation.
- If the SLIDE HOLD button is pressed during the return from the point Z level to the point R level, the cycle stops after the point R level is reached.
- If positioning to the next tapping point is executed at the point R level after the start of the spindle counterclockwise rotation but before the tapping tool is completely disengaged from the workpiece, enter a dwell at this level by specifying Q.
- Both the cutting feedrate override and the spindle speed override value are fixed at 100%.
A rapid feed override can be set.

12. Boring Cycle (G85, G89)

[Programming format]

G85 (G89) X__Y__Z__R__P__F__FA=__



ME33018R1000900250001

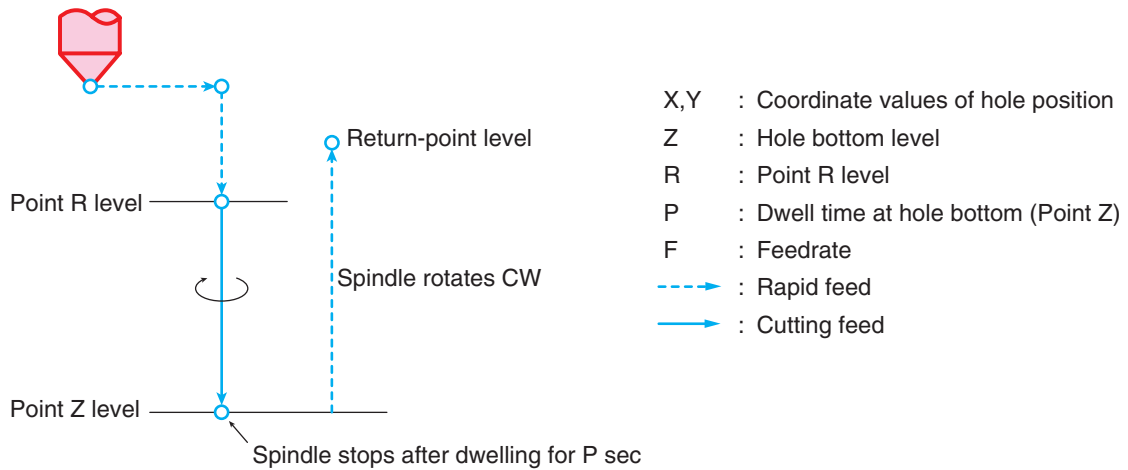
Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate
 - (2) Positioning to the point R level at a rapid feedrate
 - (3) Boring to the point Z level at the specified cutting feedrate with the spindle rotating in the clockwise direction
 - (4) Dwelling at the point Z level for P seconds
 - (5) Returning to the point R level at a return speed
 - (6) Returning to the return point level at a rapid feedrate
- If FA is not specified, F is applied for the return operation from the point R level to the return point level.

13. Boring Cycle (G86)

[Programming format]

G86 X__Y__Z__R__P__F__



ME33018R1000900260001

Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate
- (2) Positioning to the point R level at a rapid feedrate
- (3) Boring to the point Z level at the specified cutting feedrate with the spindle rotating in the clockwise direction
- (4) Dwelling at the point Z level for P seconds. Then, the spindle stops rotating.
- (5) Returning to the return point level at a rapid feedrate
- (6) Spindle rotation start in the clockwise direction.

[Details]

The difference between the G86 boring cycle and the G81/G82 drilling cycles is that the spindle stops at the hole bottom level in the G86 cycle.

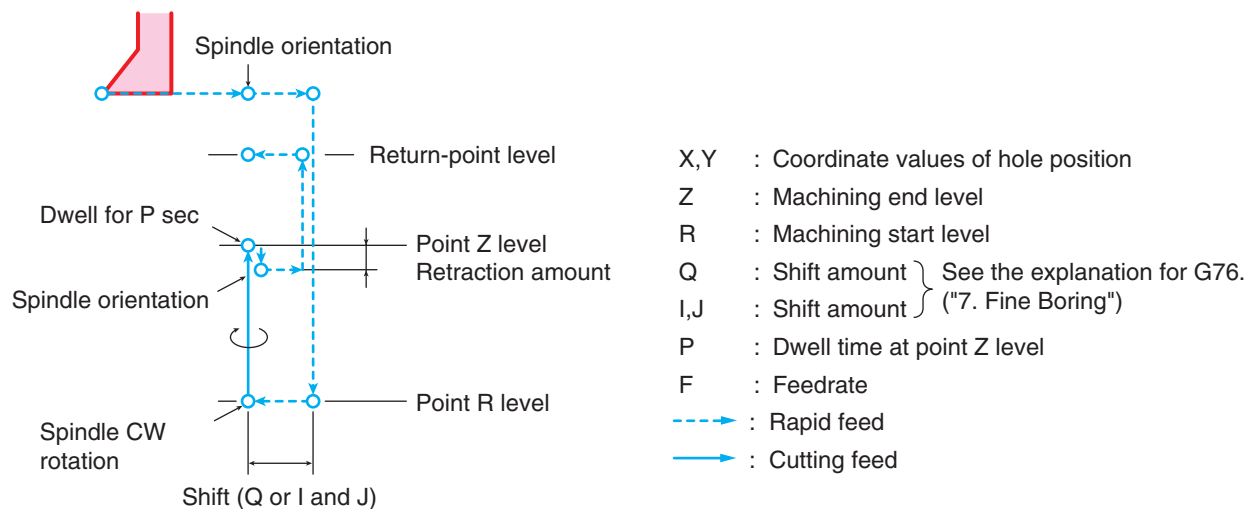
- If the distance from the return-point level where the spindle restarts rotating in the clockwise direction to the point R level of the next hole is short, the spindle may not reach the commanded speed. Therefore, it is recommended to enter a dwell command (G04) with an appropriate dwell time before the cycle for boring the next point starts.

14. Back Boring Cycle (G87)

Note that this cycle differs somewhat from other fixed cycles.

[Programming format]

G87 X__Y__Z__R__Q__ (I__J__) P__F__



[Setting values]

Retraction amount at the point Z level: Set at retraction for G76/G87 (fine boring/back boring) before orientation of NC optional parameter (fixed cycle).

The return point level may be specified from the following three levels using an M code.

- M52 Upper limit level
- M53 Specified point level (to be set in advance with G71)
- M54 Start point level

Machining Sequence

- (1) Positioning along the X- and Y-axis at a rapid feedrate. Then, the spindle stops (spindle orientation).
- (2) Shifting by shift amount Q in the direction opposite the tool bit direction
- (3) Positioning to the point R level at a rapid feedrate
- (4) At the point R level, the tool shifts back by the shift amount. Then, the spindle starts rotating in the clockwise direction.
- (5) Boring to the point Z level in the position direction of the Z-axis at the specified cutting feedrate
- (6) Dwelling at the point Z level for P seconds. After that the Z-axis moves back by the amount set for the parameter and the spindle stops (spindle orientation). The tool shifts by the shift amount Q in the direction opposite the tool bit direction.
- (7) Returning to the return point level at a rapid feedrate
- (8) Shifting back by shift amount Q

SECTION 8 COORDINATE CALCULATION FUNCTION (PATTERN FUNCTION)

The coordinate calculation function calculates the coordinate values of points on a line, grid, or circle using one command.

Combining this function with the fixed cycle function and the axis movement call function allows hole machining such as drilling to be conducted at points on a line, grid, or circle by designating one command.

1. Table of Functions

Item	Mnemonic Code	Function Outline
Omit	OMIT	Deletes coordinate calculation of a specified point.
Restart	RSTRT	Starts coordinate calculation from a certain point on a line, grid, or circle.
Line at angle	LAA	Outputs the coordinate values of points on a line which has an angle.
Grid X	GRDX	Outputs the coordinate values of points on a grid. (The calculating sequence is distinguished by X and Y.)
Grid Y	GRDY	
Double grid X	DGRDX	Outputs the coordinate values of points on two grids. (The calculating sequence is distinguished by X and Y.)
Double grid Y	DGRDY	
Square X	SQRX	Outputs the coordinate values of points on four sides of a rectangle, surrounding a grid. (The calculating sequence is distinguished by X and Y.)
Square Y	SQRY	
Bolt hole circle	BHC	Outputs the coordinate values of points on a circle.
Arc	ARC	Outputs the coordinate values of points on an arc.

2. General Rules of Coordinate Calculation

2-1. Programming Format for Coordinate Calculation

The programming format is as indicated below.

(Mnemonic code)

Hp__ Vp__ I__ J__ K__ P__ Q__ R__

Hp, Vp: Represent the coordinate values of the reference point where coordinate calculation starts.

I, J, K, P, Q, R: Parameters used for coordinate calculation.

(1) Reference point for starting coordinate calculation

The reference point is specified using the names of axes which constitute the currently selected plane. If the coordinate value of the reference point is not specified, the actual position value is taken as the coordinate value of the reference point.

The relation of plane selection to Hp (horizontal axis) and Vp (vertical axis) is as listed below.

Axis	Plane											
	G17				G18				G19			
	X-Y	X-V	U-Y	U-V	Z-X	Z-U	W-X	W-U	Y-Z	Y-W	V-Z	V-W
Hp (horizontal axis)	X	X	U	U	Z	Z	W	W	Y	Y	V	V
Vp (vertical axis)	Y	V	Y	V	X	U	X	U	Z	W	Z	W

The reference point is designated in a local coordinate system. The coordinate values to be specified vary according to the selected dimensioning mode - absolute dimensioning mode (G90) or incremental dimensioning mode (G91).

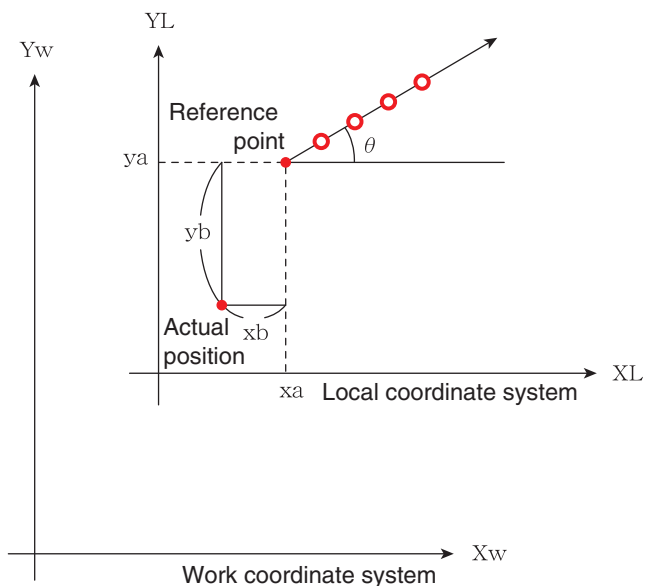
Example:

Absolute dimensioning mode

G90 LAA Xxa Yya

Incremental dimensioning mode

G91 LAA Xxb Yyb



SECTION 8 COORDINATE CALCULATION FUNCTION (PATTERN FUNCTION)

(2) Parameters used for coordinate calculation

Parameters used by a coordinate calculation function must be designated in the same block as the mnemonic code that specified the specific coordinate calculation function. These parameters are valid only in the block where they are specified and cleared after the completion of coordinate calculation.

The relationship between the coordinate calculation function and the parameters is indicated below.

Item	Mnemonic Code	Reference Point		Parameter						Remarks
		Hp	Vp	I	J	K	P	Q	R	
Omit	OMIT								(B)	More than one omit point may be specified.
Restart	RSTRT								(B)	Only one restart point may be specified.
Line at angle	LAA	(A)	(A)	(B)	(B)	(A1)				I and K must be specified in pairs. K may be omitted if "K = 1".
Arc	ARC	(A)	(A)	(B)	(B)	(A1)		(B)		Q and K must be specified in pairs. K may be omitted if "K = 1".
Grid	GRDX GRDY	(A)	(A)	(B)	(B)	(B)	(B)			
Double grid	DGRDX DGRDY	(A)	(A)	(B)	(B)	(B)	(B)	(A2)	(A3)	
Square	SQRX SQRY	(A)	(A)	(B)	(B)	(B)	(B)			
Bolt hole circle	BHC	(A)	(A)	(B)	(B)	(B)				

(A): May be omitted. If omitted, the actual position is regarded as the reference point.

(B): Must not be omitted. An alarm occurs if omitted.

(C): May be omitted.

If omitted, "1" is assumed to apply for (A1), I/2 for (A2), and J/2 for (A3).

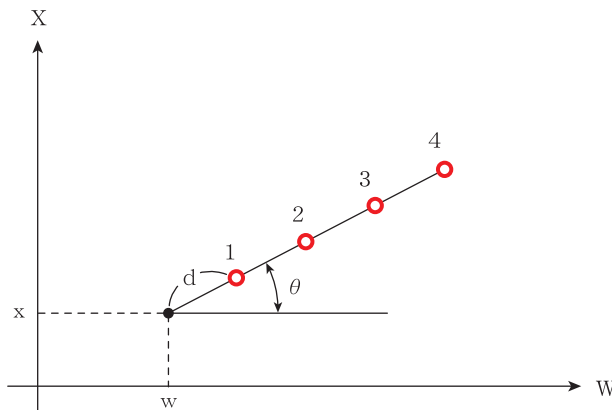
Data set for a parameter, shown as a blank column above, is disregarded.

2-2. Plane on Which Coordinate Calculation is Performed, and Motion Axes

Coordinate values are calculated on the plane which is selected when a pattern command is designated, and positioning at each calculated point is executed using the axes which are determined in accordance with the plane selection.

Example:

Pattern development on the WX plane



```
G90
G18 W__X__
NCYL G81 Y__R__F__
LAA Ww Xx Id Jθ K4
```

ME33018R1001000040001

- When the coordinate calculation function is used in combination with a fixed cycle function, the coordinate value of the axis which is perpendicular to the plane at the end of pattern operation is not always identical to the coordinate value which was active right before pattern designation.
- When return to the upper limit level (M52) is specified, the cycle axis returns to this level only after the last point of the pattern has been machined.

2-3. Positioning at Calculated Pattern Points

Positioning to a calculated point is executed in the G00, G60, or G01 mode, or in the mode determined by fixed cycle commands. An alarm occurs if positioning is attempted in the G02 or G03 mode.

2-4. Others

- Processing for single block mode operation
If the pattern function is executed in the single block mode, operation halts after positioning at the calculated pattern points has been completed.
- A maximum of 65535 points can be calculated by one pattern command.
- A pattern command cannot be designated while any of the following modes is valid.
 - Circular interpolation mode (G02, G03)
 - Cutter radius compensation mode (G41, G42)
 - 3-D tool offset mode
 - During coordinate calculation

If output of the coordinate values of the last point is deleted using the OMIT command, the upper limit level return is not executed.

3. Omit (OMIT)

[Function]

This function is normally used in combination with other coordinate calculation functions and deletes output of the coordinate value which is calculated using the coordinate calculation function.

[Programming format]

OMIT Rn1__ Rn2__ Rn3__.....Rnm__

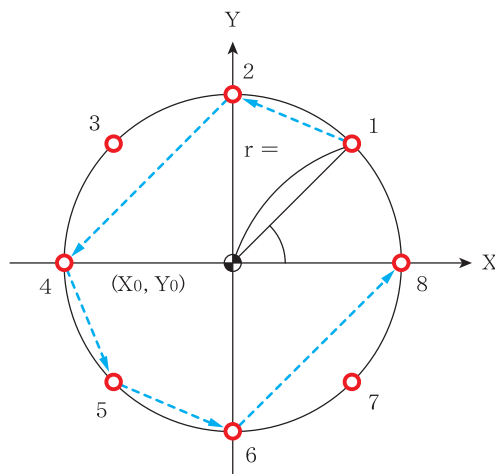
R: Coordinate point number (n) to be deleted ($m \leq 30$)

- (1) A maximum of 30 omit point numbers may be specified for one pattern command.
- (2) The maximum number usable with R is 65535.
If items 1. and 2. are not satisfied in a pattern, the pattern must be divided into two or more patterns.

[Details]

- Specify OMIT before a pattern command is specified. It is cleared after it is executed in the pattern command that appears next.
- Define the omit numbers according to the order of the points at which coordinate values are calculated. In the OMIT block, these numbers may not necessarily be specified in ascending order. If the specified omit number is larger than the maximum point number of the points calculated using the pattern command specified next, this omit number is disregarded.

[Example program]



```

NCYL G81 G56 R_Z_F_H04
OMIT R3 R7
BHC X0 Y0 I50 J45 K8

```

ME33018R1001000070001

- In this example, a drilling cycle called by G81 is executed at positions on the circle (BHC) taking (X0, Y0) as the reference point. The drilling cycle is not executed at the 3rd and 7th points due to the designation of R3 and R7.

4. Restart (RSTRT)

[Function]

This function restarts machining from a required point among the points for which coordinate values are calculated using the coordinate calculation function.

Generally, the restart data (RSTRT command and restart point number) is entered in the MDI mode.

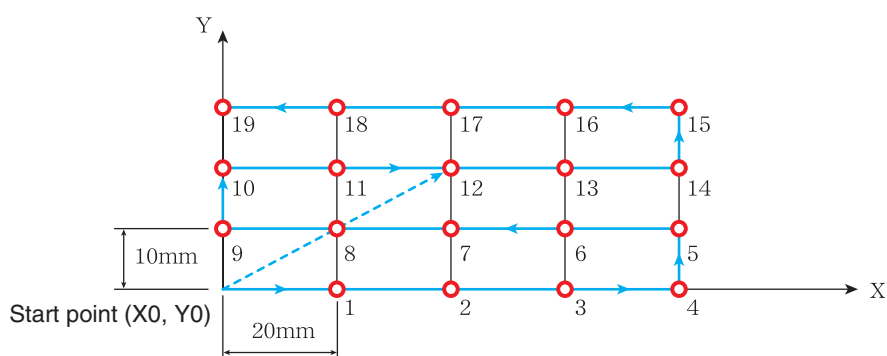
[Programming format]

RSTRT Rn

R: Coordinate point number (n) at which machining is restarted

[Details]

- If the specified omit number is larger than the maximum point number of the points calculated using the pattern command specified next, the next pattern command is disregarded.
- If more than one restart command is designated, the last restart command, or, in other words, the restart command designated right before the pattern command specified next, becomes valid.



ME33018R1001000080001

[Example program]

```
N011 G81 G56 Xo Yo R_Z_F_H
RSTRT R12 ← Enter the command in the MDI mode.
N012 GRDX I20 J10 K4 P3
```

ME33018R1001000080002

- In this example, a drilling cycle called by G81 is conducted at positions arranged in a grid pattern (GRDX) taking (X0, Y0) as the reference point. Designation of "RSTRT R12" starts drilling from hole No. 12 while skipping holes No. 1 to No. 11.

5. Line at Angle (LAA)

[Function]

This function calculates the coordinate values of points arranged at irregular intervals (d_1 , d_2 , and so forth) on a line which forms an angle θ to the horizontal axis. Here, the actual position or the point defined by the specified coordinate values is taken as the reference point.

When points are arranged at regular intervals, designate the number of these points (n).

[Programming format]

LAA Hp__Vp__I \pm d1Kn1 \pm d2 Km2 J \pm θ

Hp : Coordinate value of the reference point on the horizontal axis

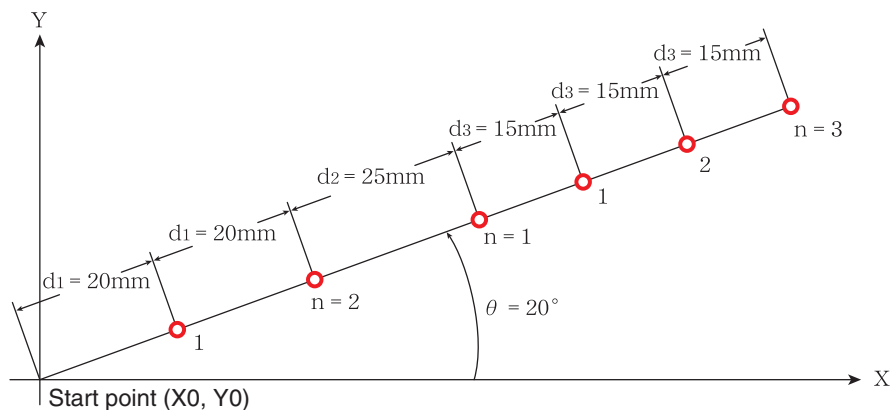
Vp : Coordinate value of the reference point on the vertical axis

I : Interval (d) When a negative value is specified, coordinate values are calculated on the line that extends in the symmetric direction in reference to the start point.

K : Number of points (n) arranged at regular intervals (d) The maximum number is 65535.

J : Angle of the line in reference to the horizontal axis (θ)

The angle is set in units of 1 degree, 0.001 degree, or 0.0001 degree in accordance with the selected unit system. The angle is measured in the CCW direction.



LAA X₀ Y₀ I20 K2 I25 (K1) I15 K3 J20

↑
May be omitted

ME33018R1001000090001

[Details]

- If address K is omitted, it is regarded as I is set.
- Up to ten intervals can be designated in one block.
- When one pair of I and K is designated with J, the LAA command can be used as a unit-pitch line at angle command.
- The coordinate values of the reference point are not output.
- The order for designating I, J, and K must be either J, I, K, I, K.... or I, K, I, K J.

6. Grid (GRDX, GRDY)

[Function]

This function calculates the coordinate values of points arranged in the grid pattern, composed of the points (nx) placed at an interval of (dx) in parallel with the horizontal axis and of the points (ny) placed at an interval of (dy) in parallel with the vertical axis. Here, the actual position or the point defined by the specified coordinate values is taken as the reference point.

- GRDX: The coordinate values are calculated along the horizontal axis starting from the reference point.
- GRDY: The coordinate values are calculated along the vertical axis starting from the reference point.

[Programming format]

GRDX(GRDY) Hp__Vp__I ± dx J ± dy Knx Pny

Hp : Coordinate value of the reference point on the horizontal axis

Vp : Coordinate value of the reference point on the vertical axis

I : Interval along the horizontal axis (dx)

When a positive value is specified for "dx", coordinate calculation is performed in the positive direction of the horizontal axis.

When a negative value is specified for "dx", coordinate calculation is performed in the negative direction of the horizontal axis.

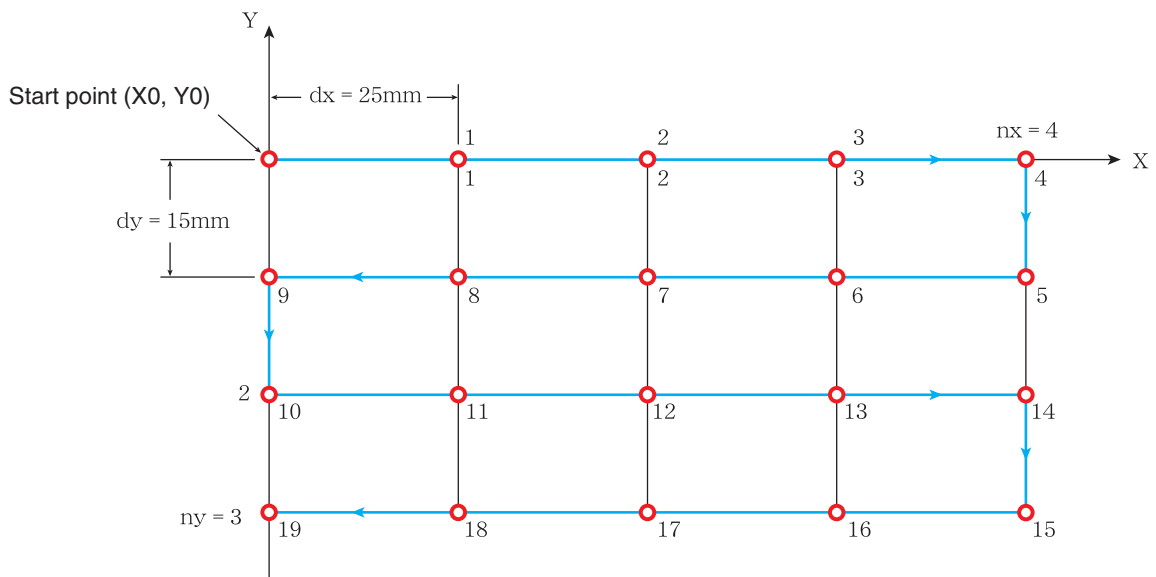
J : Interval along the vertical axis (dy)

When a positive value is specified for "dy", coordinate calculation is performed in the positive direction of the vertical axis.

When a negative value is specified for "dy", coordinate calculation is performed in the negative direction of the vertical axis.

K : Number of points arranged along the horizontal axis (nx)(Positive integer)

P : Number of points arranged along the vertical axis (ny)(Positive integer)



GRDX X0 Y0 I25 J - 15 K4 P3

[Details]

- The maximum number of points on a grid $((nx + 1) \times (ny + 1) - 1)$ is 65535.
- The number of the last point is $(nx + 1) \times (ny + 1) - 1$.
- The coordinate values of the reference point are not output.

7. Double Grid (DGRDX, DGRDY)

[Function]

This function calculates the coordinate values of points arranged in the double grid pattern, composed of the reference grid and the other grid defined by shifting the reference grid. The pattern of the reference grid is composed of the points (nx) placed at regular intervals (dx1) in parallel with the horizontal axis and of the points (ny) placed at regular intervals (dy,) in parallel with the vertical axis. The second grid pattern is defined by shifting the reference grid both in the X- and Y-axis directions by "dx2" and "dy2", respectively. Here, the actual position or the point defined by the specified coordinate values is taken as the reference point.

- DGRDX: The coordinate values are calculated along the horizontal axis starting from the reference point.
- DGRDY: The coordinate values are calculated along the vertical axis starting from the reference point.

[Programming format]

$$\left\{ \begin{array}{l} \text{DGRDX} \\ \text{DGRDY} \end{array} \right\} \text{Hp_Vp_ I} \pm \text{dx J} \pm \text{dy1 Knx Pny Q} \pm \text{dx2 R} \pm \text{dy2}$$

Hp : Coordinate value of the reference point on the horizontal axis

Vp : Coordinate value of the reference point on the vertical axis

I : Interval along the horizontal axis (dx1)

When a positive value is specified for "dx1", coordinate calculation is performed in the positive direction of the horizontal axis.

When a negative value is specified for "dx1", coordinate calculation is performed in the negative direction of the horizontal axis.

J : Interval along the vertical axis (dy1)

When a positive value is specified for "dy1", coordinate calculation is performed in the positive direction of the vertical axis.

When a negative value is specified for "dy1", coordinate calculation is performed in the negative direction of the vertical axis.

K : Number of points arranged along the horizontal axis (nx)(Positive integer)

P : Number of points arranged along the vertical axis (ny)(Positive integer)

Q : Shift interval in the horizontal-axis direction (dx2)

The sign used for "dx2" has the same meaning as for "dx1".

R : Shift interval in the vertical-axis direction (dy2)

The sign used for "dy2" has the same meaning as for "dy1".

SECTION 8 COORDINATE CALCULATION FUNCTION (PATTERN FUNCTION)

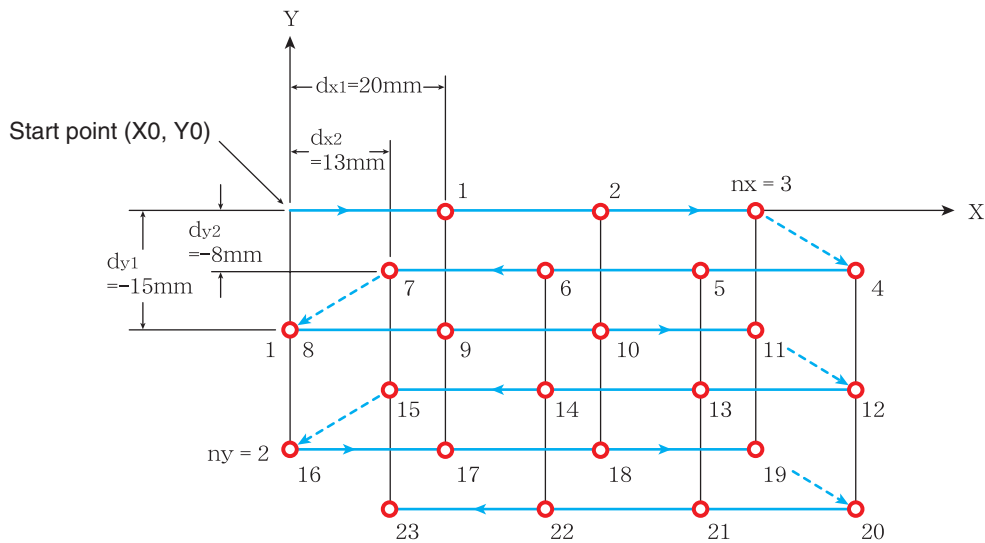
[Details]

- When “dx2” is equal to “dx1/2”, designation of Q can be omitted. Similarly, when “dy2” is equal to “dy1/2”, designation of R can be omitted. Note that the mark is the same as dx1, dy1.
- The maximum number of points on a double grid $(2(n_x + 1)(n_y + 1) - 1)$ is 65535.
- The number of points on the shift grid is the same as that on the reference grid (n_x, n_y) .
- The coordinate values of the reference point are not output.

[Example program]

Example 1:

Coordinate value calculation order for double grid defined by DGRDX

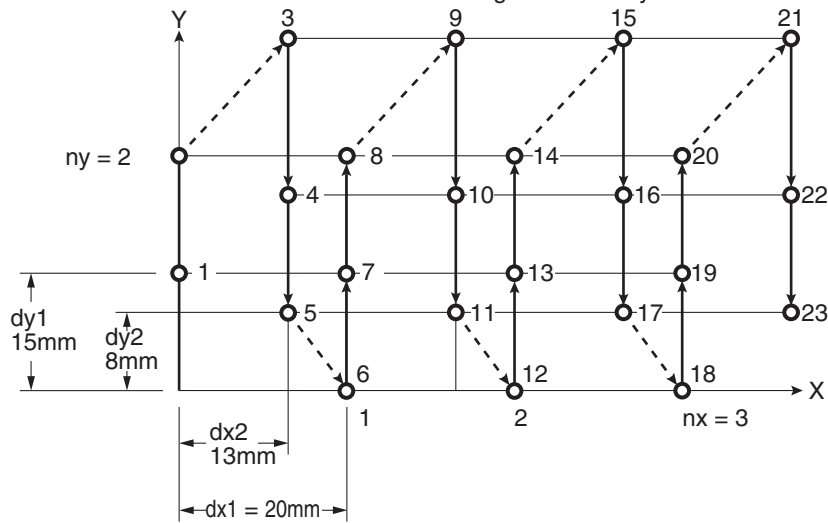


DGRDX X0 Y0 I20 J - 15 K3 P2 Q13 R - 8

SECTION 8 COORDINATE CALCULATION FUNCTION (PATTERN FUNCTION)

Example 2:

Coordinate value calculation order for double grid defined by DGRDY



DGRDY X0 Y0 I20 J15 K3 P2 Q13 R8

ME33018R1001000110003

8. Square (SQRX, SQRY)

[Function]

This function calculates the coordinate values of points arranged in the square pattern, composed of the points (nx) placed at an interval of (dx) parallel to the horizontal axis and of the points (ny) placed at an interval of (dy) parallel to the vertical axis. Here, the actual position or the point defined by the specified coordinate values is taken as the reference point.

- SQRX: The coordinate values are calculated along the horizontal axis starting from the reference point.
- SQRY: The coordinate values are calculated along the vertical axis starting from the reference point.

SECTION 8 COORDINATE CALCULATION FUNCTION (PATTERN FUNCTION)

[Programming format]

$$\left. \begin{array}{l} \text{SQRX} \\ \text{SQRY} \end{array} \right\} \text{Hp_Vp_ } I \pm dx \text{ } J \pm dy \text{ } K_{nx} \text{ } P_{ny}$$

Hp : Coordinate value of the reference point on the horizontal axis

Vp : Coordinate value of the reference point on the vertical axis

I : Interval along the horizontal axis (dx)

When a positive value is specified for "dx", coordinate calculation is performed in the positive direction of the horizontal axis.

When a negative value is specified for "dx", coordinate calculation is performed in the negative direction of the horizontal axis.

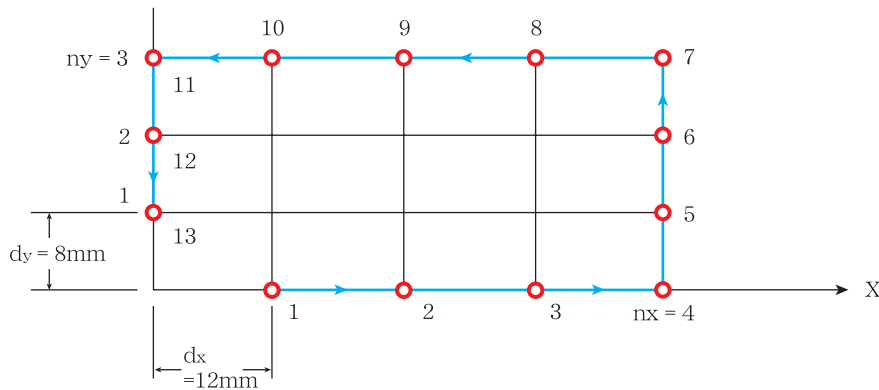
J : Interval along the vertical axis (dy)

When a positive value is specified for "dy", coordinate calculation is performed in the positive direction of the vertical axis.

When a negative value is specified for "dy", coordinate calculation is performed in the negative direction of the vertical axis.

K : Number of points arranged along the horizontal axis (nx)(Positive integer)

P : Number of points arranged along the vertical axis (ny)(Positive integer)



SQRX Xo Yo I12 J8 K4 P3

ME33018R1001000120001

[Details]

- The maximum number of points on a square $(2(nx + ny) - 1)$ is 65535.
- The coordinate values of the reference point are not output.

9. Bolt Hole Circle (BHC)

[Function]

This function calculates the coordinate values of the points arranged on the circumference of a circle that has its center at the actual position or a point defined by the specified coordinate values. The radius of the circle is "r" and "n" points are arranged at regular intervals starting from the position at angle θ to the horizontal axis.

[Programming format]

BHC Hp__Vp__ IrJ θ Kn

Hp : Coordinate value of the center (horizontal axis)

Vp : Coordinate value of the center (vertical axis)

I : Radius of the circle (r)(Positive value)

J : Start angle (θ)

The angle is set in units of 1 degree, 0.001 degree, or 0.0001 degree in accordance with the selected unit system.

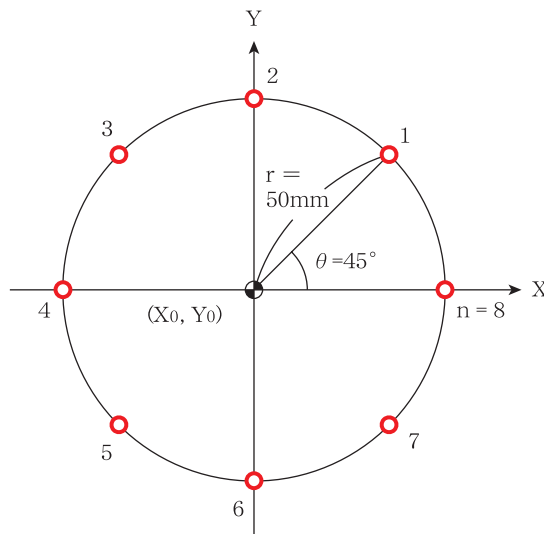
The angle is measured in the CCW direction.

K : Number of points (n)

Coordinate values are calculated in the CCW direction when a positive value is set for "n".

Coordinate values are calculated in the CW direction when a negative value is set for "n".

The maximum number of points is 65535.



BHC XXo YYo I50 J45 K8

10. Arc (ARC)

[Function]

This function calculates the coordinate values of the points arranged on the circumference of a circle that has its center at the actual position or a point defined by the specified coordinate values. The radius of the circle is "r" and the points are arranged at irregular intervals ($\Delta\theta_1, \Delta\theta_2, \dots$) starting from the position at angle θ to the horizontal axis.

If more than two points are arranged at regular intervals ($\Delta\theta$) consecutively, the number of such points should be specified.

[Programming format]

ARC Hp__Vp__ Ir Q $\Delta\theta_1$ Kn1 Q $\Delta\theta_2$ kn2 J θ

Hp : Coordinate value of the center (horizontal axis)

Vp : Coordinate value of the center (vertical axis)

I : Radius of the circle (r)(Positive value)

Q : Irregular angular interval ($\Delta\theta$)

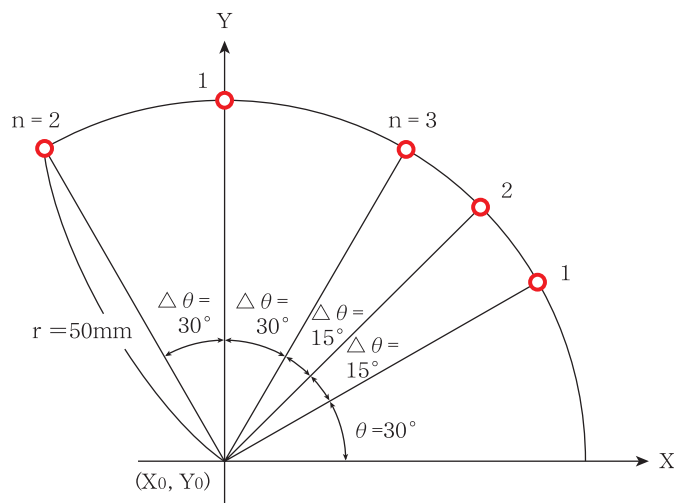
The angle is set in units of 1 degree, 0.001 degree, or 0.0001 degree in accordance with the selected unit system. The angle is measured in the CCW direction.

K : Number of points of the same angle sequence ($\Delta\theta$) (n)

The maximum number of points is 65535.

J : Start angle (θ)

The angle is set in units of 1 degree or 0.001 degree in accordance with the selected unit system. The angle is measured in the CCW direction.



ARC XXo YYo I50 Q15 K3 Q30 K2 J30

ME33018R1001000140001

[Details]

- If address K is omitted, it is regarded as I is set.
- Up to ten Q commands, specifying an irregular angular interval, can be specified in one block.
- When one pair of Q and K is specified with I and J, the ARC command can be used as a unit-pitch arc command.

SECTION 9 AREA MACHINING FUNCTIONS

Area machining functions are used to machine the top, periphery or inside surface of a rectangular area with a single command. The area to be machined must be formed by four straight lines which intersect at right angles. The direction of cutting is in the longitudinal direction of the rectangular area.

The explanation below assumes that the X-Y plane is the machining plane and the Z-axis is cycle axis. The explanation is similar for the other planes.

1. List of Area Machining Functions

Function	Mnemonic Code	Area to be Machined	Cutting Direction	Remarks
Face milling	FMILR	Top face of the area (surface)	Longitudinal direction of rectangular area	Tool-ON
	FMILF			Tool-OFF
Pocket milling	PMIL	Inside		Zigzag
	PMILR			Spiral
Round milling	RMILO	External periphery		-
	RMILI	Internal periphery		-

2. Area Machining Operations

2-1. Basic Operations

Area machining consists of the basic operations shown below.

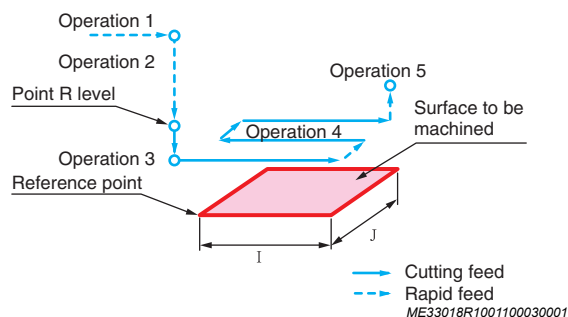
Operation 1 : Positioning in the currently selected plane to a position near the reference point

Operation 2 : Positioning of the cutting axis to the point R level

Operation 3 : Cutting in of the cycle axis

Operation 4 : Area machining

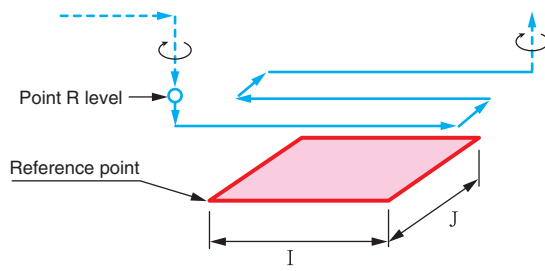
Operation 5 : Retraction of the cutting tool from the machined surface to the point R level



- Operations 3, 4, and 5 are repeated as necessary depending on the amount of stock to be removed. Infeed for the machining in the next level is always executed at the first positioning point.
- Area machining is executed in the currently selected plane, and the axis vertical to the machining plane is selected as the cycle axis.
- If M52 is specified in a program while area machining is being executed with the Z-axis as the cycle axis, the Z-axis will be returned to the upper limit level after the final cutting.

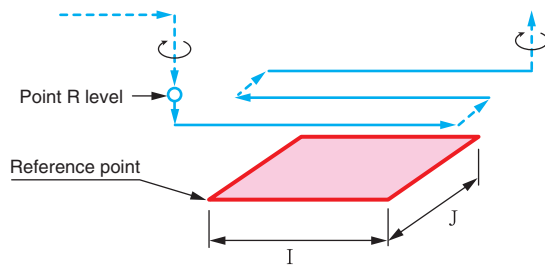
2-2. Tool Movements

(1) Face milling (FMILR)



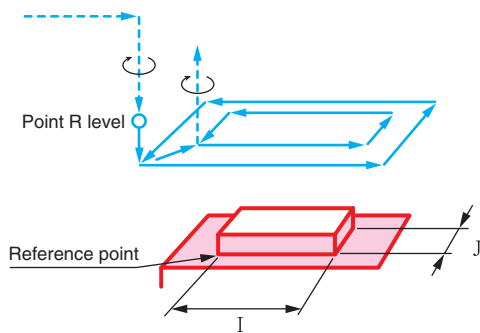
ME33018R1001100040001

(2) Face milling (FMILF)



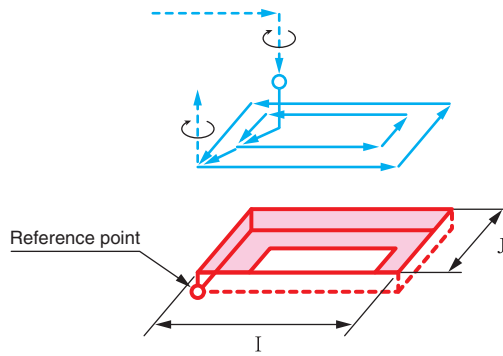
ME33018R1001100040002

(3) Round milling (RMILO)



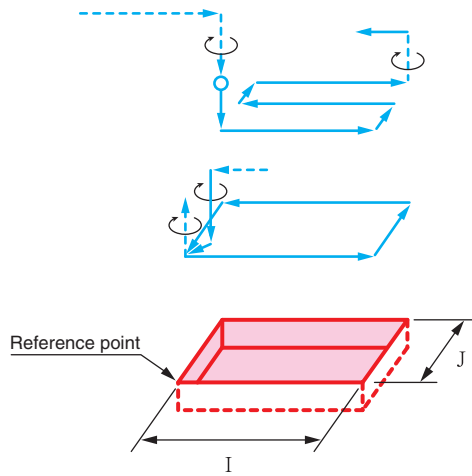
ME33018R1001100040003

(4) Round milling (RMIL)



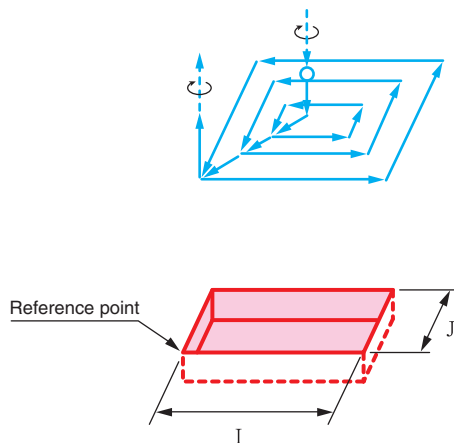
ME33018R1001100040004

(5) Pocket milling (PMIL)



ME33018R1001100040005

(6) Pocket milling (PMILR)



ME33018R1001100040006

3. Area Machining Plane and Cycle Axis

- The plane in which operations 1 and 4, defined in “Basic Operations”, are executed is determined by the designation of G17, G18, or G19 for plane selection. Hereafter, the selected plane is called the area machining plane.
- The cycle axis is the one vertical to the area machining plane and it is represented by X, Y, Z, or an axis parallel to them.

Plane Selection Code	Area Machining Plane	Cycle Axis	Plane Composing Axes	
			Horizontal Axis	Vertical Axis
G17	Xp-Yp plane	Zp	Xp	Yp
G18	Zp-Xp plane	Yp	Zp	Xp
G19	Yp-Zp plane	Xp	Yp	Zp

Xp: X- or U-axis

Yp: Y- or V-axis

Zp: Z- or W-axis

- The plane selection code automatically determines two cycle axes; the address specified in the area machining command block to define the finishing surface level determines the actual cycle axis.

4. General Rules

The following explanations assume that the area machining plane is the XY plane and that the infeed axis is the Z-axis. The explanation is similar for the other planes.

4-1. Programming Format (General Command Format)

(Mnemonic code)

Xp__ Yp__ Zp__ I__ J__ K__ P__ Q__ R__ D__ F__ FA =__ FB =__

Mnemonic code: Area machining function code

- Xp: X-coordinate value of the reference point (Horizontal axis coordinate value)
- Yp: Y-coordinate value of the reference point (Vertical axis coordinate value)
- Zp: Z-coordinate value of finish surface level (Cycle axis coordinate value)
- I: Length from the reference point in the X-axis direction (Horizontal length from the reference point)
- J: Length from the reference point in the Y-axis direction (Vertical length from the reference point)
- K: Finish allowance
- P: Percentage of cutting width
- Q: Depth of cut of each cycle
- R: Rapid retraction position (for cycle axis)
- D: Cutter radius compensation number

F, FA, FB: Feedrate

[Details]

- Xp, Yp, and Zp represent the X-axis or U-axis, the Y-axis or V-axis, and the Z-axis or W-axis, respectively.
- FA and FB are used in the round milling and pocket milling cycles.
- Xp and Yp should designate the axes which define the area machining plane.
- Zp should designate the cycle axis which is vertical to the plane defined by Xp and Yp.
- The K value must be such that the machining allowance is positive. The following requirements must be met, otherwise an alarm occurs.

In the G90 mode : $|K| < |Z|$

In the G91 mode : $|K| < |R - Z|$

4-2. Area Machining Functions and Commands to be Used

Item Address	Reference Point		Finish Surface Level	Area Definition		Finish Allowance	Cutting Width (%)	Depth of Cut Per Cycle	Level R	Cutter Radius Compensation No.	Feedrate		
	XP	Yp	Zp	I	J	K	P	Q	R	D	F	FA	FB
Face Milling	(A)	(A)	(B)	(B)	(B)	(C)	(C)	(C)	(B)	(B)	(B)		
Pocket Milling	(A)	(A)	(B)	(B)	(B)	(C)	(C)	(C)	(B)	(B)	(B)	(C)	(C)
Round Milling	(A)	(A)	(B)	(B)	(B)	(C)	(C)	(C)	(B)	(B)	(B)	(C)	

ME33018R1001100080001

[Explanation of the above table]

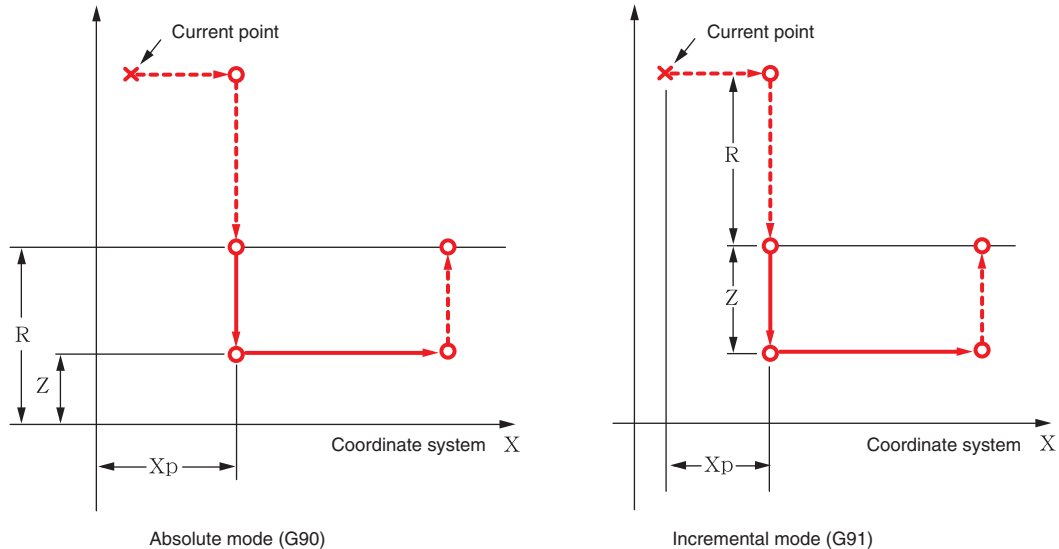
- Addresses designated with a single circle (A) may be omitted. If omitted, the coordinate value of the current position is used.
- Addresses designated with a double circle (B) must always be specified. An alarm occurs if any of these addresses is not specified. However, F may be omitted if it has been specified in a previous block.
- Addresses designated with a triangle (C) may be omitted. If any of these addresses is not specified, the value indicated below is regarded.

Address	Description	Default	Remarks
K	Finish allowance	0: No finish allowance	-
P	Cutting width ratio (%)	70: 70 percent	-
Q	Depth of cut in per cycle	The level "finish surface + finish allowance" is reached by a single cut.	When K is also omitted, cutting is performed up to the finishing level.
FA	Feedrate	$4 \times F$	Applies only to pocket milling and round milling.
FB	Feedrate	$F/4$	Applies only to pocket milling.

- D, D00 or a number which calls zero cutter radius must not be specified for the cutter radius compensation number.
- The setting range of the cutting width, P, is "1" to "100". Any value outside this range will result in an alarm.
- If a value is specified for a blank box address, it is disregarded.

4-3. Data Entry in Incremental/Absolute Mode

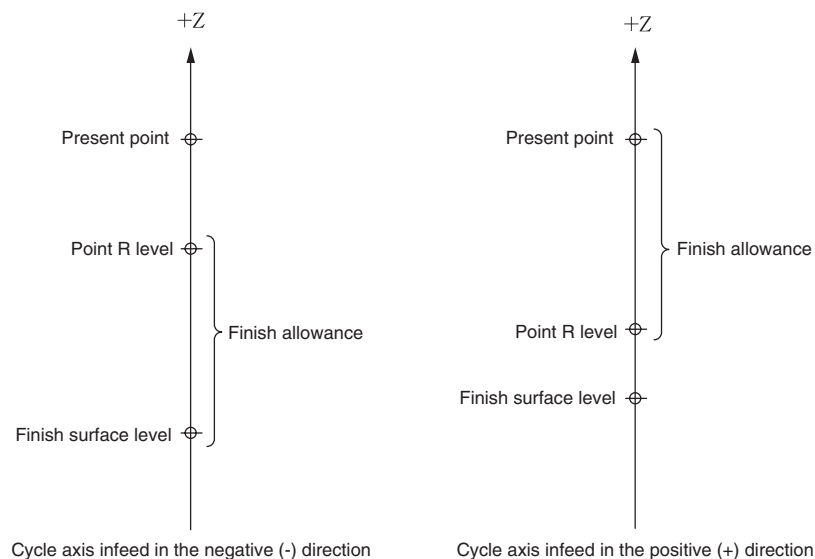
Four addresses for an area machining function must be specified depending on the selected dimensioning mode, incremental or absolute. They are: the coordinate values of the reference point (X_p , Y_p), the finish surface level (Z_p), and the rapid retraction level (R). How these values differ depending on the dimensioning mode is shown in the illustration below.



In the incremental mode, the direction of cycle axis infeed is determined by the sign specified preceding R and Z . The illustration above shows the operation when a negative value is specified for both R and Z ($R < 0$, $Z < 0$).

4-4. Relationship among Present Point, Point R Level, and Finish Surface Level

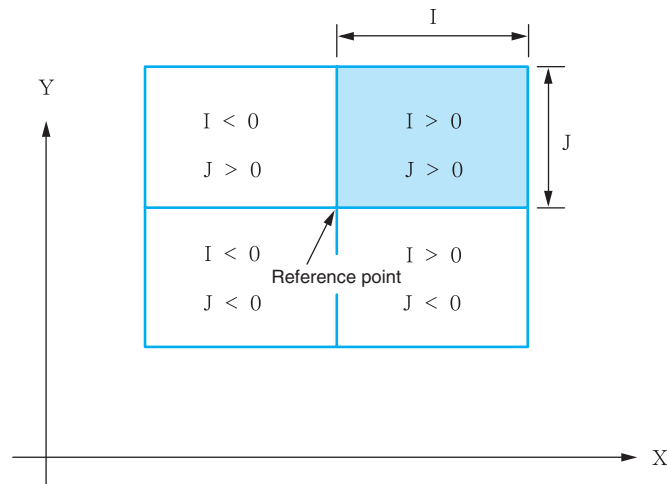
- The positional relationship among the present point, point R level, and finish surface level complies with one of the two cases shown below.
- The surface level on which the finish allowance is left must lie between the point R level and finish surface level.



4-5. Definition of Machining Area (I, J)

The machining area is defined by I and J values, and the signs of I and J values. How the areas are defined according to the signs specified preceding the I and J values is shown below.

The areas are defined independently of the direction of the cycle axis infeed described in "Relationship among Present Point, Point R Level, and Finish Surface Level".



ME33018R1001100110001

4-6. Notes on Area Machining

- The first positioning point will be located near the specified reference point (X_p , Y_p). The position varies depending on the specified Milling area machining mode. Refer to "Face Milling Functions", "Pocket Milling", and "Round Milling Functions".
- The finish allowance may be identical for both the side and top surface.
- An area machining cycle is always executed with the spindle rotating in the forward direction (M03). If the spindle is stopped or it is rotating in the reverse direction when an area machining cycle is specified, M03 is automatically generated and executed to rotate the spindle in the forward direction. Note that, even in such a case, the M03 mode remains valid after the completion of the cycle.
- The cutter diameter is calculated as twice the cutter radius compensation value. Since this value is treated as the actual cutter diameter when an area machining cycle is executed, special care must be taken if the compensation value stored in "Dnn" differs from the actual cutter radius.
- Restrictions
The area machining function cannot be specified in the following modes:
 - Circular interpolation, or helical cutting modes (G02, G03)
 - Cutter radius compensation mode (G41, G42)
 - Three-dimensional tool offset mode (G44)
 - Axis move call mode (MODIN)
 - Fixed cycle mode (G73 - G89)

5. Face Milling Functions (FMILR, FMILF)

[Function]

The face milling function uses the specified coordinate values as a reference point and cyclically machines the workpiece surface at a certain depth of cut (Q) over the range specified by the X- and Y-axis lengths (I and J) until the final finish allowance (K) remains on the finish surface level (Z).

There are two types of face milling functions, depending on the tool movement:

- FMILR in which the tool remains on the workpiece during operation
- FMILF in which the tool moves off of the workpiece to shift to the next cutting path

[Programming format]

$$\left. \begin{array}{l} \text{FMILR} \\ \text{FMILF} \end{array} \right\} X \pm x \ Y \pm y \ Z \pm z \ I \pm dx \ J \pm dy \ kfi \ P\% \ Qdp \ R \pm rz \ Dnn \ F_$$

X : X coordinate value (x) of reference point

If omitted, the X coordinate value of the current point is regarded as that of the reference point.

Y : Y coordinate value (y) of reference point

If omitted, the Y coordinate value of the current point is regarded as that of the reference point.

Z : Position of finish surface (z)

In the G90 mode: Height from the programming zero to the finish surface level

In the G91 mode: Distance from the point R level to the finish surface level

I : Length of the rectangle to be cut along the X-axis (dx)

Length referenced to the reference point (x)

J : Length of the rectangle to be cut along the Y-axis (dy)

Length referenced to the reference point (y)

K : Finish allowance (fl)

If omitted, "f1 = 0" is regarded.

P : Cutting width expressed in percent (%)

Ratio, in percentage terms, of the cutting width to the cutter diameter. Although the ratio is expressed as a percentage, the percent symbol (%) must not be specified.

If omitted, "P70" (70%) is assumed to apply.

As will be explained later, the command value is slightly different from the actual cutting width.

Q : Depth of cut (dp)

If omitted, the cutter reaches the surface "finish surface position + finish allowance (K)" in a single cut.

Number of cuts: The number of cuts repeated to reach the level indicated above is calculated as indicated below.

$$n = Fup \left(\frac{\text{Shorter side} + 5 \text{ mm}}{D \times 2 \times \frac{P}{100}} \right) \quad * : Fup \text{ indicates the processing to round up decimal fractions.}$$

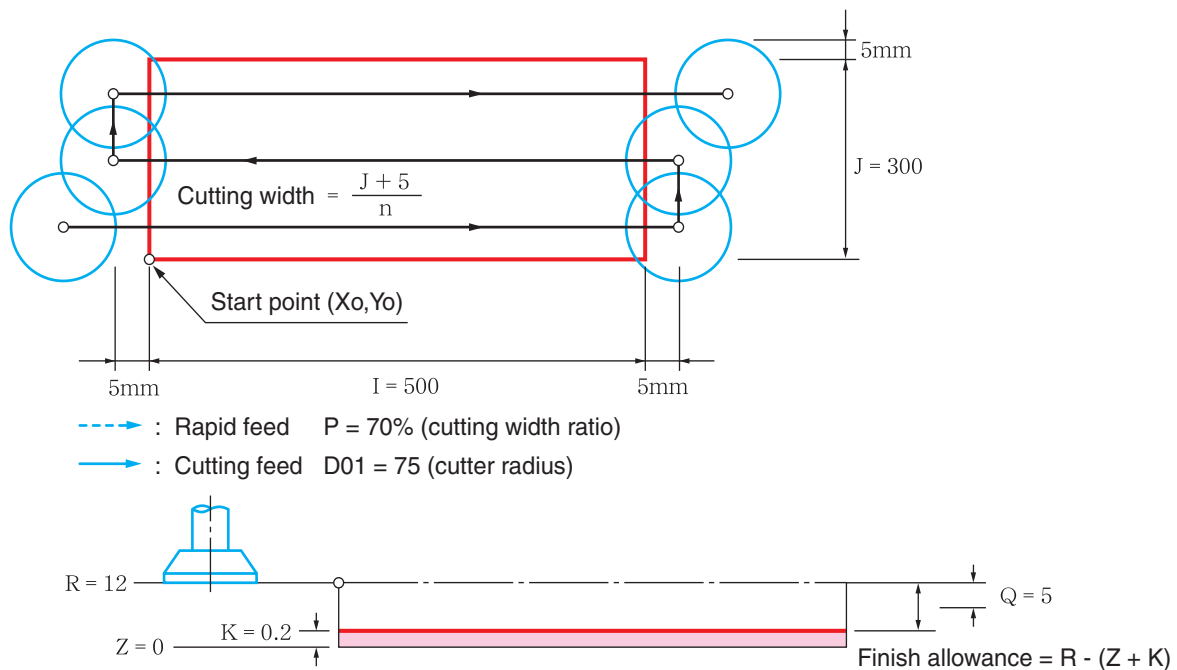
R : Rapid retraction level (rz)

The value set at the cutter radius compensation number

D : Cutter radius compensation number (nn)

D is doubled when depth of cut is calculated.

F : Feedrate

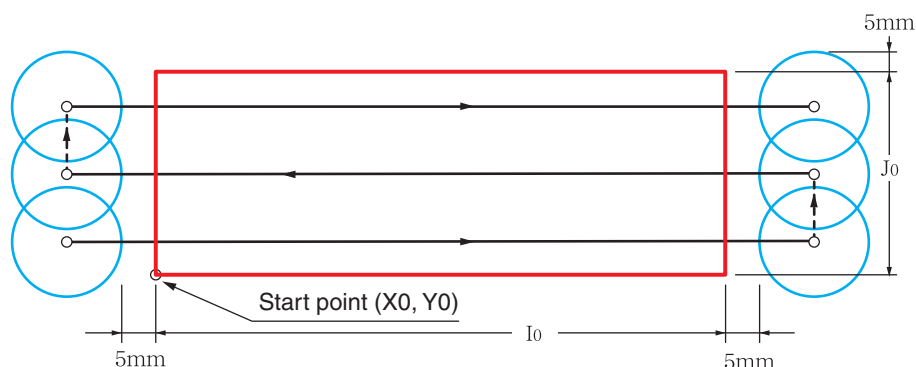
Tool-ON type (Tool remains on workpiece) (FMILR)

FMILR X0 Y0 Z0 I500 J300 K0.2 P70 Q5 R12 D01 F400

Cutter movements:

1. The reference point is established at (X_0, Y_0) .
 2. The cutter has a radius specified by D01.
 3. The rapid feed level is set by R12.
 4. The cutter is infed by the amount $Q = 5$ mm (0.20 in.).
 5. At this level, area machining is carried out with a cutting width ratio of $(J + 5)/n$ at a feedrate of F400.
 6. After each cycle of surface machining, the cutter is infed by the amount $Q = 5$ mm (0.20 in.).
Machining is repeated until the level where finish allowance, $K = 0.2$ mm (0.008 in.), is left on the finish surface level.
Details of axis movements will be described on later pages.
- * : Stock (finish allowance) = $R - (Z + K)$

ME33018R1001100130002

Tool-OFF type (Tool moves off of workpiece) (FMILF)

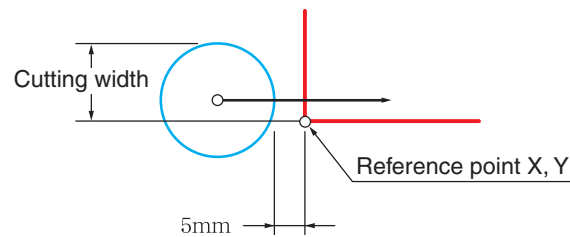
ME33018R1001100130003

FMILF X0 Y0 Z I0 J0 K P Q R D F

In the FMILF mode, although the Z-axis moves in the same way as in the FMILR mode, on the machining surface, the cutter is fed to a point where the cutter is disengaged from the workpiece before it is positioned to the next tool path.

Positioning of a cutter

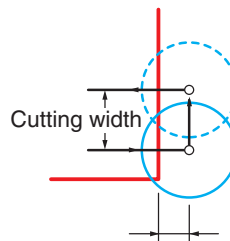
(1) First positioning



ME33018R1001100130004

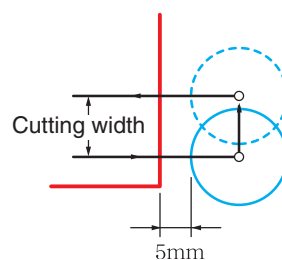
- In the narrower direction of a workpiece, the cutter is positioned so that the specified width of cutting* is engaged on the workpiece.
Width of cutting = (Shorter side + 5 mm / n)
- In the wider direction of a workpiece, the cutter is positioned so that its periphery is 5 mm (0.20 in.) away from the workpiece.

(2) Cutting path in the longitudinal direction



ME33018R1001100130005

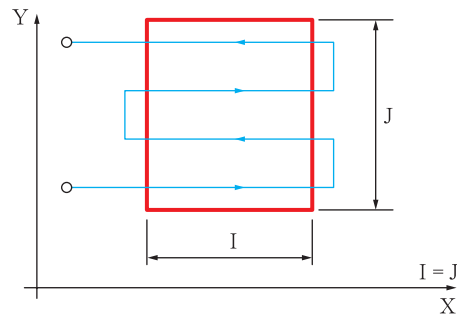
- FMILR (Workpiece-ON type)
Machining continues until the center of the cutter moves 5 mm (0.20 in.) away from the workpiece.
- FMILF (Workpiece-OFF type)
Machining continues until the outside diameter of the cutter moves 5 mm (0.20 in.) away from the workpiece.
- I = J



ME33018R1001100130006

For both the FMILR and the FMILF function, cutting is performed in the X direction when I equals J.

- (3) Cutting path along the shorter sides of a workpiece (from the reference point)



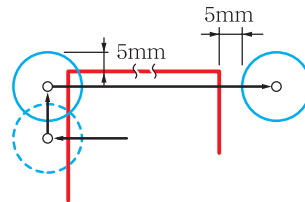
$$\frac{\text{Shorter side} + 5\text{mm}}{n} \times i - \text{Cutter radius compensation value}$$

(i = 1, 2, 3, ...)

(n = number of cuts)

ME33018R1001100130007

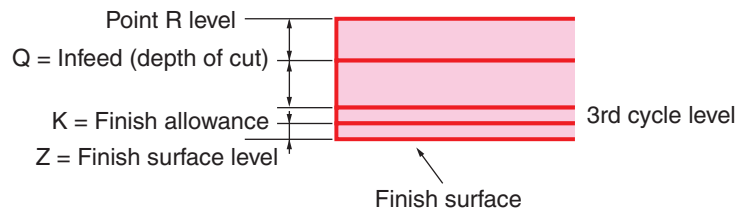
- (4) Final cutting path



ME33018R1001100130008

For both the FMILR and the FMILF, the cutter is positioned so that the periphery of the cutter projects 5 mm (0.20 in.) from the workpiece. Machining continues until the cutter periphery is 5 mm (0.20 in.) away from the workpiece.

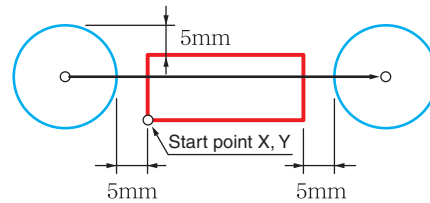
- (5) Depth of cut in the Z-axis direction



ME33018R1001100130009

- Starting at the specified point R level, face milling operation continues by repeating infeed (specified depth of cut (Q)) and the face milling cycle at each level until the level of “finish surface level (Z) + finish allowance (K)” is reached.
- If the depth of the cut (Q) is greater than the stock to be removed (R - (Z + K)), the cutting is performed in one infeed.

(6) Tool path for the workpiece with width smaller than cutting width



ME33018R1001100130010

- Positioning point
 - Along the shorter side of the workpiece, positioning is performed so that the cutter periphery projects 5 mm (0.20 in.) from the workpiece.
 - Along the longer side of the workpiece, positioning is performed so that the cutter periphery is 5 mm (0.20 in.) away from the workpiece.
- End-of-machining point
Machining continues until the cutter periphery reaches a point 5 mm (0.20 in.) away from the workpiece.

6. Pocket Milling (PMIL, PMILR)

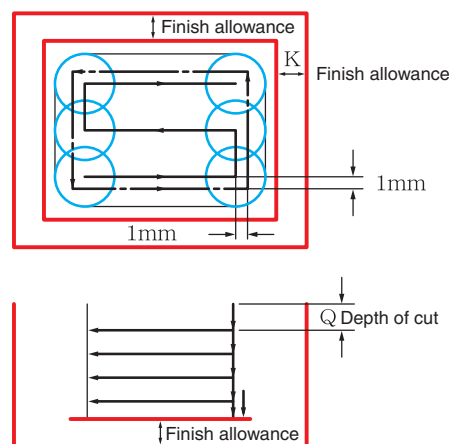
The pocket milling function is classified into two types: zigzag (PMIL) and spiral (PMILR). These two types of functions are described below.

6-1. Zigzag Pattern Pocket Milling Function (PMIL)

[Function]

The zigzag pattern pocket milling function uses the specified coordinate values as a reference point and cyclically machines the rectangular pocket range specified by the X- and Y-axis lengths (I and J) at a certain depth of cut (Q) until the final finish allowance (K) remains on the finish surface level (Z). In the pocket milling operation, finish allowance (K) is also left on the side faces of the pocket in the X- and Y-axis directions.

- Operation



ME33018R1001100150001

- The cutter is infed by the specified depth of cut, Q.
- The cutter moves inside the rectangular pocket range specified with X and Y in a zigzag pattern.
- Steps 1) and 2) are repeated until only the finish allowance remains on the finish surface in the Z-direction. Then, a rectangle 1 mm (0.04 in.) larger than the rectangle machined in step 2) is machined.

[Programming format]

PMIL X__ Y__ Z__ I__ J__ K__ P__ Q__ R__ D__ F__ FA =__ FB =__

- X: X coordinate value of start point
If omitted, the X coordinate value of the current point is regarded as that of the start point.
- Y: Y coordinate value of start point
If omitted, the Y coordinate value of the current point is regarded as that of the reference point.
- Z: Z coordinate value of finish surface
In the G90 mode: Height from the zero point of the selected coordinate system to the finish surface level (bottom face of the pocket)
In the G91 mode: Distance from the point R level to the finish surface level (bottom face of the pocket)
- I: Length of the pocket to be machined along the X-axis
Length referenced to the start point

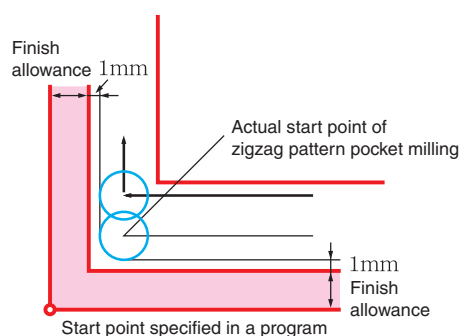
- J: Length of the pocket to be machined along the Y-axis
Length referenced to the start point
- K: Finish allowance (valid in all directions, X, Y, and Z)
If omitted, "0" is regarded.
- P: Cutting width expressed as a percentage (%)
Ratio, in percentage terms, of the cutting width to the cutter diameter. Although the ratio is expressed as a percentage, the percent symbol (%) must not be specified.
If omitted, "P70" (70%) is assumed to apply.
- Q: Depth of cut
If omitted, the depth of cut is determined so that the cycle axis reaches the level "finish surface level + finish allowance" in a single cut.
- R: Z coordinate value of the level to which positioning is performed in rapid feed or rapid return
- D: Cutter radius compensation number
- F: Cutting feedrate
This feedrate is used for the zigzag pattern machining and the final peripheral machining.
- FA: Feedrate after point R level
After each cycle of zigzag pattern machining, the cutter returns to the point R level once and then moves to the point 1 mm away from the previously machined level. This feedrate is used for the cycle axis movement to this point from the point R level. If omitted, "FA = 4 × F" is assumed to apply.
- FB: Feedrate for Z-axis infeed
If omitted, "FB = F / 4" is assumed to apply.
Note that both FA and FB are only valid in the specified block.

Machining Sequence

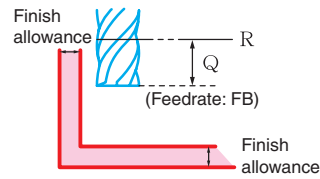
Before starting the PMIL operation, the function checks if the programmed operation is possible based on the programmed pocket shape and the specified cutter diameter. An alarm occurs if the following is not satisfied.

Shorter side - $(1\text{mm} + \text{Finish allowance} + \text{Cutter radius}) \times 2 > 5\text{mm}$

- (1) In the X-Y plane, the cutter is positioned at the start point. Taking the start point specified by X and Y in the program as the reference point, the system positions the cutter inward of the rectangle considering the finish allowance (K) and residual finish amount for X- and Y-axis directions. That is, the cutter is positioned inward by the amount of the finish allowance (K) adding 1 mm for the specified value of X and Y. Note that the residual finish amount is taken into consideration to determine the start point so that residual portion can be finally removed shown in the following figure. This amount is fixed at 1 mm (0.04 in.).

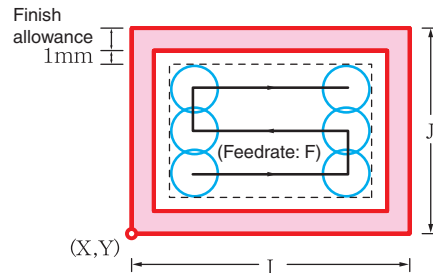


- (2) The infeed axis Z is positioned to the point R level at a rapid feedrate.



ME33018R1001100150003

- (3) Starting at the point R level, the Z-axis is fed by the specified depth of cut, Q, at the feedrate specified by FB.



ME33018R1001100150004

- (4) The inside of the rectangle is cyclically machined in a zigzag pattern at the feedrate specified by F.

Note that the cutting width is different from the specified value as indicated below:

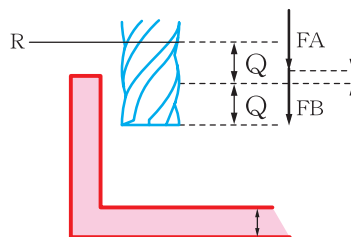
$$\text{Actual cutting width} = \frac{\text{Shorter side} - 2 (K + \text{Cutter radius compensation amount} + 1\text{mm})}{n - 1}$$

n: Number of cuts; obtained by rounding up the decimal fraction of the value calculated using the following formula.

$$n = \left(\frac{\text{Shorter side} - 2 (K + \text{Cutter radius compensation amount} + 1\text{mm})}{\text{Cutter radius compensation amount} \times 2 \times \frac{P}{100} + 1} + 1 \right)$$

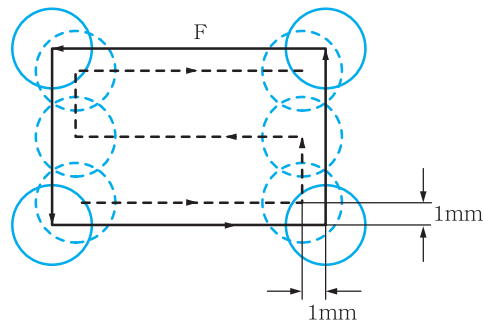
ME33018R1001100150005

- (5) The cutter returns to the initial positioning point (X, Y, R) at a rapid feedrate. It is then positioned from the point R level to a point 1 mm (0.04 in.) above the surface level machined in the previous machining cycle at a feedrate of FA. Then, the cutter is infeed by the amount "Q + 1 mm (0.04 in.)" at a feedrate of FB. The next machining cycle is performed in a zigzag pattern at a feedrate of F.



ME33018R1001100150006

- (6) Step (5) above is repeated until the final finish allowance remains on the finish surface. Finally, the cutter machines a rectangular pocket 1 mm (0.04 in.) wider than the machined pocket. In the final cycle, the feedrate specified by F is used.



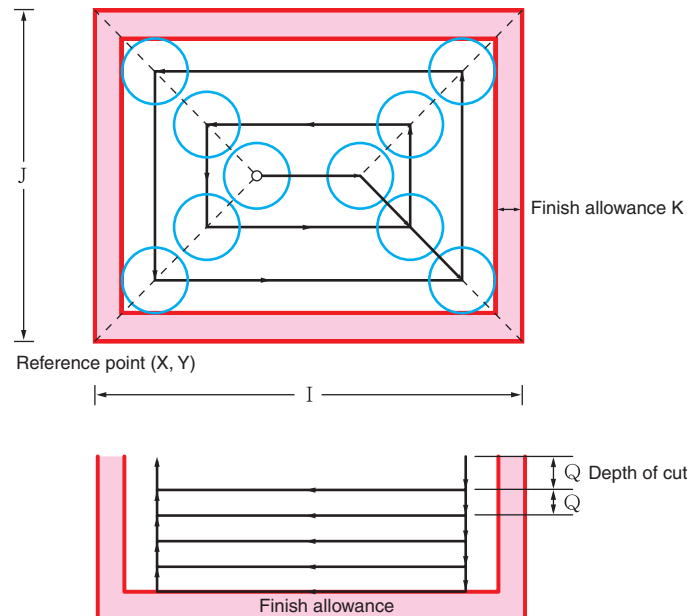
ME33018R1001100150007

- (7) After completion of the cycle, the axes are positioned at a point 5 mm (0.20 in.) away from the workpiece in each axis direction.

6-2. Spiral Pattern Pocket Milling Function (PMILR)

[Function]

The spiral pattern pocket milling function uses the specified coordinate values as a reference point and cyclically machines the rectangular pocket range specified by the X- and Y-axis lengths (I and J) at a certain depth of cut (Q) until the final finish allowance (K) remains on the finish surface level (Z). In the pocket milling operation, a finish allowance (K) is also left on the side faces of the pocket in the X- and Y-axis directions.



ME33018R1001100160001

- Operation
 - a. The cutter is infed by the specified depth of cut, Q.
 - b. The cutter moves inside the specified rectangular pocket range in a spiral pattern. However, leave the finish allowance K.
 - c. Steps a. and b. above are repeated until only the finish allowance K remains on the finish surface in the Z-direction.

[Programming format]

PMILR X__Y__Z__I__J__K__P__Q__R__D__F__FA=__FB=__

- X: X coordinate value of start point
If omitted, the X coordinate value of the current point is regarded as that of the start point.
- Y: Y coordinate value of start point
If omitted, the Y coordinate value of the current point is regarded as that of the reference point.
- Z: Z coordinate value of finish surface
In the G90 mode: Height from the zero point of the selected coordinate system to the finish surface level (bottom face of the pocket)
In the G91 mode: Distance from the point R level to the finish surface level (bottom face of the pocket)
- I: Length of the pocket to be machined along the X-axis
Length referenced to the start point
- J: Length of the pocket to be machined along the Y-axis
Length referenced to the start point

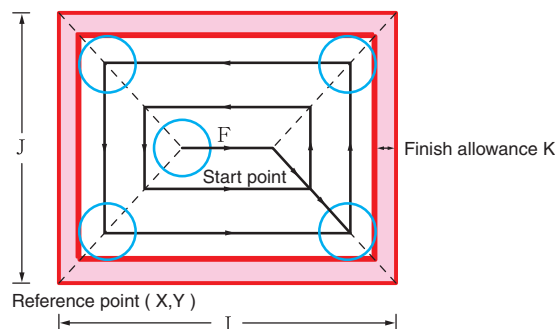
- K: Finish allowance (valid in all directions, X, Y, and Z)
If omitted, "0" is regarded.
- P: Cutting width expressed in percent (%)
Ratio, in percentage terms, of the cutting width to the cutter diameter. Although the ratio is expressed as a percentage, the percent symbol (%) must not be specified.
If omitted, "P70" (70%) is assumed to apply.
- Q: Depth of cut
If omitted, the depth of cut is determined so that the cycle axis reaches the level "finish surface level + finish allowance" in a single cut.
- R: Z coordinate value of the level to which positioning is made in rapid feed or rapid return
- D: Cutter radius compensation number
- F: Cutting feedrate
This feedrate is used for the zigzag pattern machining and the final peripheral machining.
- FA: Feedrate after point R level
After each cycle of zigzag pattern machining, the cutter returns to the point R level once and then moves to a point 1 mm away from the previously machined level. This feedrate is used for the cycle axis movement to this point from the point R level.
If omitted, "FA = 4 × F" is regarded.
- FB: Feedrate for Z-axis infeed
If omitted, "FB = F / 4" is regarded.
Note that both FA and FB are only valid in the specified block.

Machining Sequence

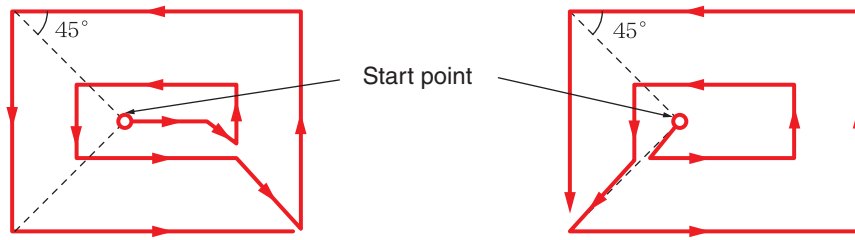
Before starting the PMILR operation, the function checks if the programmed operation is possible based on the programmed pocket shape and the specified cutter diameter. An alarm occurs if the following is not satisfied.

Shorter side - (Finish allowance + Cutter radius) × 2 > 5mm

- (1) In the X-Y plane, the cutter is positioned to the start point.



Taking the start point specified in the program as the reference point, the system determines a rectangle from the lengths specified by I and J. In this rectangle, the system defines another rectangle by leaving finish allowance K on four sides, then it determines the start point based on the cutting width. The cutter is positioned at that start point.

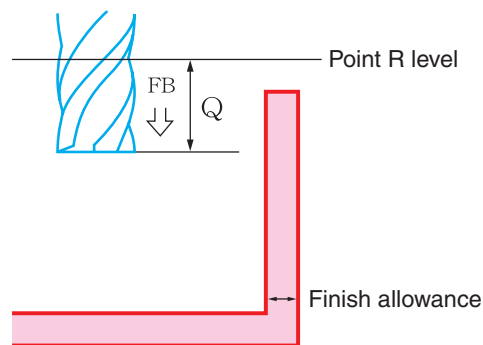


(If machined in an odd number of spiral cycles)

(If machined in an even number of spiral cycles)

ME33018R1001100160003

- (2) The infeed axis Z is positioned to the point R level at a rapid feedrate.



ME33018R1001100160004

- (3) Starting at the point R level, the Z-axis is fed by the specified depth of cut, Q, at the feedrate specified by FB.
- (4) The inside of the rectangle is cyclically machined in a spiral pattern at the feedrate specified by F.

Note that the cutting width is different from the specified value as indicated below:

$$\text{Actual cutting width} = \frac{\text{Shorter side} - 2 (K + \text{Cutter radius compensation amount})}{n - 1}$$

n: Number of cuts; obtained by rounding up the decimal fraction of the value calculated using the following formula.

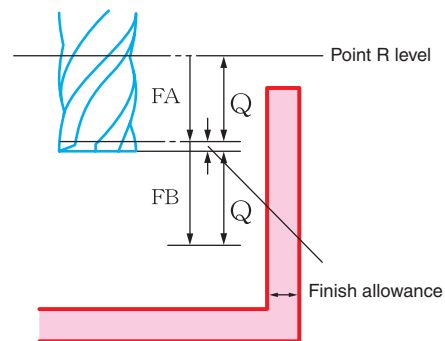
$$n = \left(\frac{\text{Shorter side} - 2 (K + \text{Cutter radius compensation amount})}{\text{Cutter radius compensation amount} \times 2 \times \frac{P}{100}} + 1 \right)$$

ME33018R1001100160005

The diagonal cutting feedrate F toward the corner can be overridden by the override setting at OVERRIDE IN PMLR CORNER CUTTING of the NC optional parameter (AREA MACHINING).

SECTION 9 AREA MACHINING FUNCTIONS

- (5) The cutter returns to the initial positioning point (X, Y, Z) at a rapid feedrate. It is then positioned from the point R level to a point 1 mm (0.04 in.) above the surface level machined in the previous machining cycle at a feedrate of FA. Then, the cutter is infed by the amount “Q + 1 mm (0.04 in.) at a feedrate of FB”. The next machining cycle is performed in a spiral pattern at a feedrate of F.



ME33018R1001100160006

- (6) Step (5) above is repeated until the final finish allowance remains on the finish surface.
- (7) After completion of the cycle, the axes are positioned at a point 5 mm (0.20 in.) away from the workpiece in each axis direction.

7. Round Milling Functions (RMILO, RMILI)

[Function]

The round milling function uses the specified coordinate values as a reference point and cyclically machines the rectangle specified by the X- and Y-axis lengths (I and J), which has stock Q to be removed at the level finish allowance (K) above the finish surface level (Z).

There are two types of round milling functions as indicated below:

- RMILO in which the external side of the defined rectangle is machined
- RMILI in which the internal side of the defined rectangle is machined

[Programming format]

RMILO } $X \pm x \ Y \pm y \ Z \pm z \ I \pm dx \ J \pm dy \ _{kfl} P\% \ _{Qdp} R \pm rz \ _{Dnn} F _ \ FA _ _$
 RMILI }

X : X coordinate value (x) of reference point

If omitted, the X coordinate value of the current point is regarded as that of the reference point.

Y : Y coordinate value (y) of reference point

If omitted, the Y coordinate value of the current point is regarded as that of the reference point.

Z : Position of finish surface (z)

In the G90 mode: Height from the programming zero to the finish surface level

In the G91 mode: Distance from the point R level to the finish surface level

I : Length of the rectangle to be cut along the X-axis (dx)

Length referenced to the reference point (x)

J : Length of the rectangle to be cut along the Y-axis (dy)

Length referenced to the reference point (y)

K : Finish allowance (fl)

If omitted, "fl = 0" is regarded.

P : Cutting width expressed in percent (%)

Ratio, in percentage terms, of the cutting width to the cutter diameter. Although the ratio is expressed as a percentage, the percent symbol (%) must not be specified.

If omitted, "P70" (70%) is assumed to apply.

As will be explained later, the command value is slightly different from the actual cutting width.

Q : Stock to be removed (dp)

If omitted, the cutter reaches the surface "finish surface position + finish allowance (K)" in a single cut.

Number of cuts: The number of cuts repeated to reach the level indicated above is calculated as indicated below.

$$n = \text{Fup} \left(\frac{Q - K}{\text{Cutter radius compensation amount} \times 2 \times \frac{p}{100}} \right) \quad * : \text{Fup indicates the processing to round up decimal fractions.}$$

R : Rapid retraction level (rz)

D : Cutter radius compensation number (nn)

F : Feedrate

Feedrate used for machining the external or internal circumference of the defined rectangle

FA : Feedrate

Feedrate used when infeeding the Z-axis from the point R level to the finish surface (+ finish allowance K).

If omitted, "FA = 4 x F" is assumed to apply.

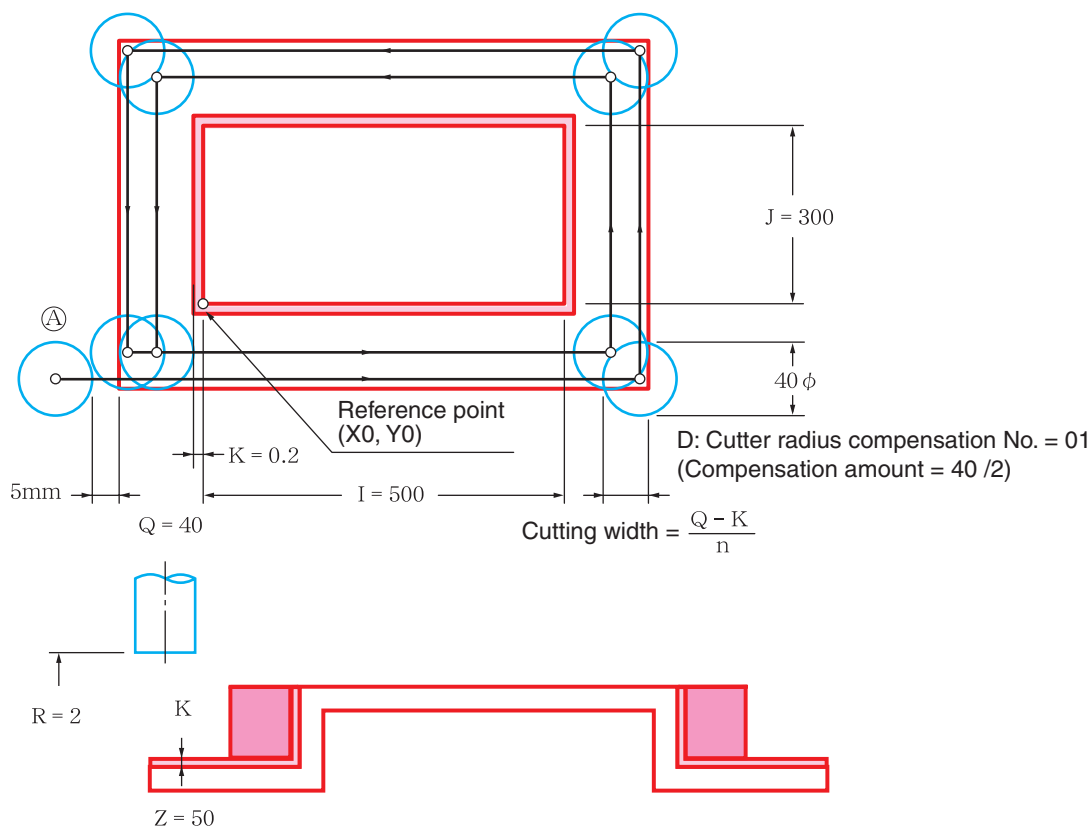
ME33018R1001100170001

RMILO - External Cutting

- The first positioning point (A) is the point where the cutter periphery is 5 mm (0.20 in.) away in the longitudinal direction, and in the crosswise direction, the cutter is infed by the specified cutting width from the edge of the blank workpiece.

- Before starting the RMILO operation, the function checks the relationship between the finish allowance and the stock to be removed. An alarm occurs if the following is not satisfied.

$$Q \geq K$$



ME33016R1001100170002

RMILO X0 Y0 Z-50 I500 J300 K0.2 P70 Q40 R2 D01 F400 FA800

How the RMILO operation is executed by the commands above is described below.

The reference point is taken as X0, Y0.

The rectangle is specified with I and J, 500 mm in the direction of X-axis and 300 mm in the Y-axis direction.

A cutter with a specified diameter of D01 is used.

Rapid feed positioning level is R (2 mm (0.078 in.)).

Machining allowance is Q (40 mm (1.57 in.)).

Cycle machining level is "finish level (Z)" + "finishing allowance (K)".

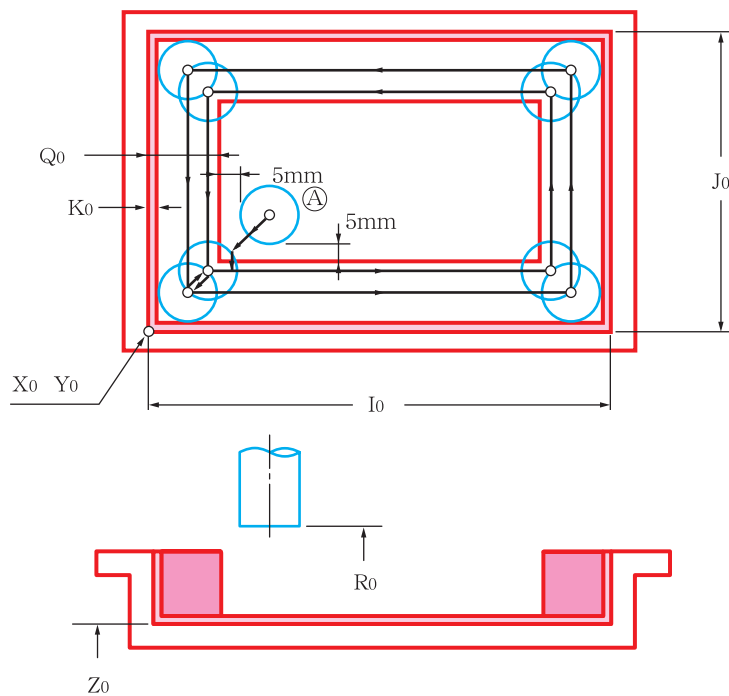
For this example: -50 mm + 0.2 mm.

Cutter is fed by the amount " $\left(\frac{Q - K}{n}\right)$ " from the edge of the blank workpiece and cycle machining is performed at feedrate F.

Cycle machining is repeated until the specified finish allowance remains on the surface.

RMILI - Internal Cutting

- The first positioning point (A) is the point where the cutter periphery is 5 mm (0.20 in.) away from the edge of the blank workpiece.
- Before starting the RMILI operation, the function checks the relationship between the defined shape and the specified values. An alarm occurs if the following is not satisfied.
 - $(\text{Cutter radius compensation amount} + Q + 5\text{mm}) \times 2 < \text{Shorter side}$
Alarm B 2315 Area machine: area command occurs unless this inequality is satisfied.
 - $Q \geq K$
Alarm B 2319 Area machining: large finish allowance occurs unless this inequality is satisfied.



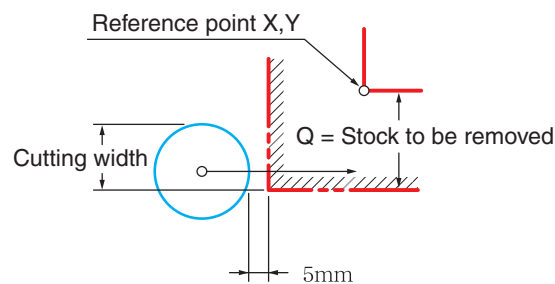
ME33018R1001100170003

RMILI X0 Y0 Z0 J0 K0 PQ0 R0 D F FA

Positioning in Round Milling Cycle

- (1) First positioning
Positioning is executed at the following position.

- External cutting (RMILO):

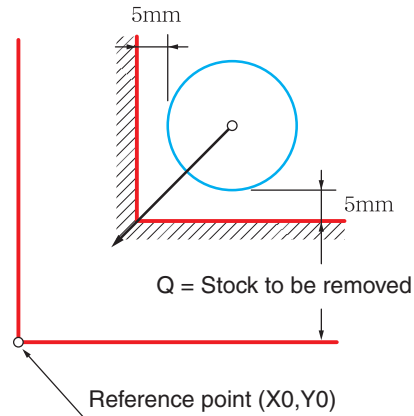


ME33018R1001100170004

Along the shorter side direction, the cutter engages the workpiece by the cutting width (cutter diameter \times P).

Along the longer side direction, the cutter periphery is located 5 mm (0.20 in.) away from the workpiece edge.

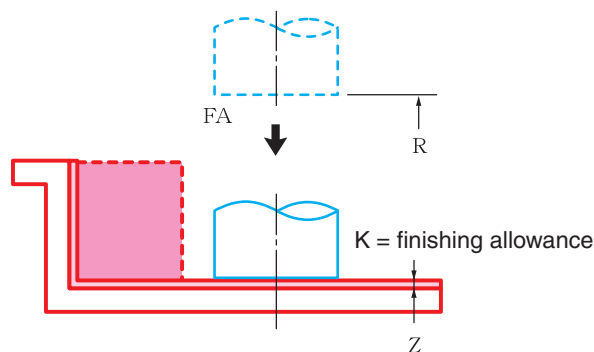
- Internal cutting (RMIL):



ME33018R1001100170005

Both in the X- and Y-axis directions, the cutter is positioned to a point where the cutter periphery is 5 mm (0.20 in.) away from the workpiece.

- (2) Positioning in the Z-axis direction

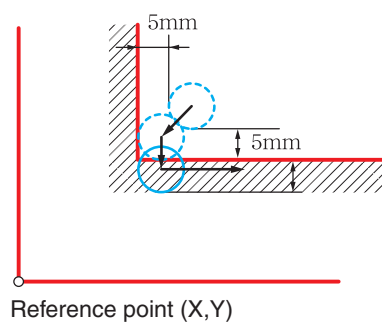


ME33018R1001100170006

The cutter is fed from the point R level to the level of the finish surface level (Z) plus the finishing allowance (K) at a feedrate specified by FA.

In the Z-axis direction, machining to the specified level (Z + K) is executed in one infeeding operation.

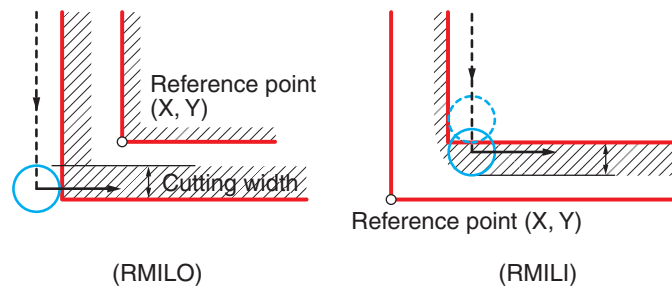
- (3) First infeed to milling surface



ME33018R1001100170007

- External cutting (RMILO):
Since the cutter is positioned, in the first positioning, at a point where it is engaged with the workpiece by the specified cutting width, the round milling cycle can be started from the first positioning point.
- Internal cutting (RMILI):
The cutter is moved toward the reference point so that its periphery contacts the workpiece (5 mm (0.20in.) both in the X- and Y-axis directions) from the first positioning point. The cutter is then infed by the cutting width along the shorter side direction. The round milling cycle then starts from that point.

(4) Second and succeeding infeds to milling surface

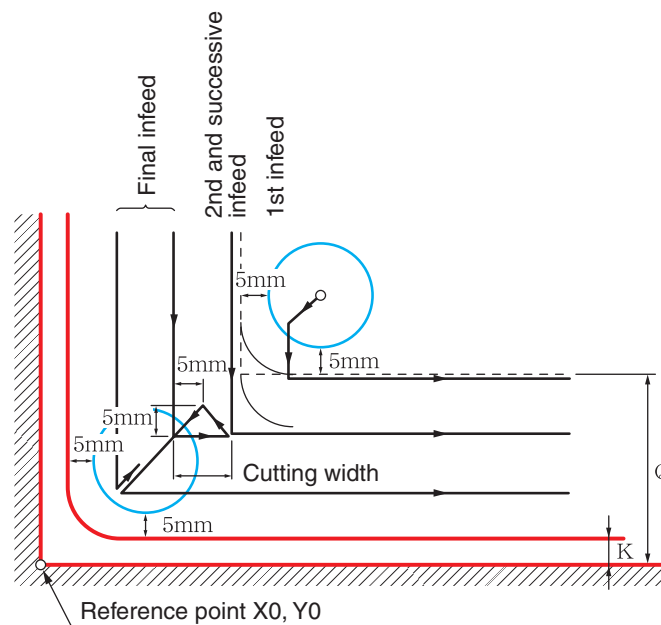


ME33018R1001100170008

(5) Final infeed

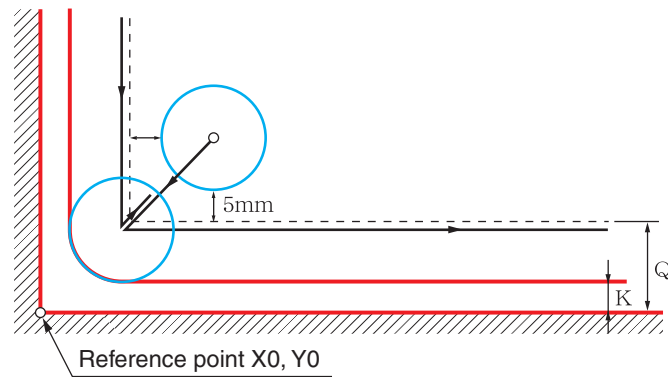
In the case of the RMILI function only, the final infeed is executed in the manner shown below.

- Machining with more than one cut



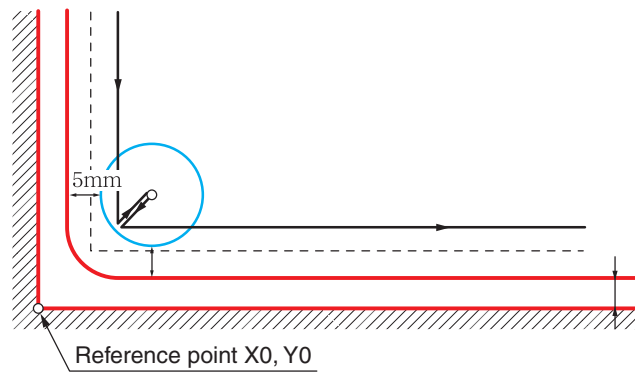
ME33018R1001100170009

- Machining with a single cut



ME33018R1001100170010

- Without Q (stock) command



ME33018R1001100170011

(6) Retraction from workpiece

In the RMILI (internal cutting) mode, the cutter retracts inward from the workpiece as it is in contact with the workpiece at the end of the cycle. The retraction amount is 5 mm (0.20 in.) along both axes.

In the RMILO (external cutting) mode, the cutter does not retract.

SECTION 10 SUBPROGRAM FUNCTIONS

1. Overview

Programming sometimes uses similar patterns repeatedly or uses patterns already programmed for other operations. The subprogram function allows such patterns which are used repeatedly to be stored as subprograms so that they can be called when necessary. Therefore, the subprogram function not only simplifies programming, but also it allows quick and accurate programming.

1-1. Calling a Subprogram

- A subprogram can be called not only from the main program but also from another subprogram, and up to 16-subprogram nesting is possible.
- Three subprogram call modes are available as indicated below. Note that a subprogram call command must be specified independently. Accordingly, if a subprogram call command is specified with axis move commands in the same block, the subprogram call command is disregarded. For details, refer to the following sub sections.
 - a. Simple call (CALL)
 - b. Call after axis move (MODIN, MODOUT)
 - c. G/M code macros
 - Simple calls (G111 - G120, M201 - M210)
 - Call after axis move (G100 - G110)
 - Maker macro calls (for simple calls only)
- Subprograms to be called may be grouped into three types:

Subprogram	File Device	File Extension	Usable Subprogram Name
User subprogram (System subprogram)	MD1:	.SUB	O * * *
	MD1:	.SSB	└── Other than O
Maker subprogram	System memory	.MSB	O O * * *

ME33018R1001200020001

- When a subprogram call command is executed, the specified subprogram is searched for among the subprogram files displayed in response to the execution of the directory command key in the EDIT/AUX mode. How the specified subprogram is searched for differs depending on the type of subprogram; whether it is a user subprogram or a maker subprogram. However, if there is a subprogram immediately following a main program, then the search is made from that subprogram.
 - a. User subprograms

The specified subprogram is first searched for in the file where the program select command is specified. If the subprogram is not found in this file, it is searched for in all files whose device name is MD1: with extension .SSB.

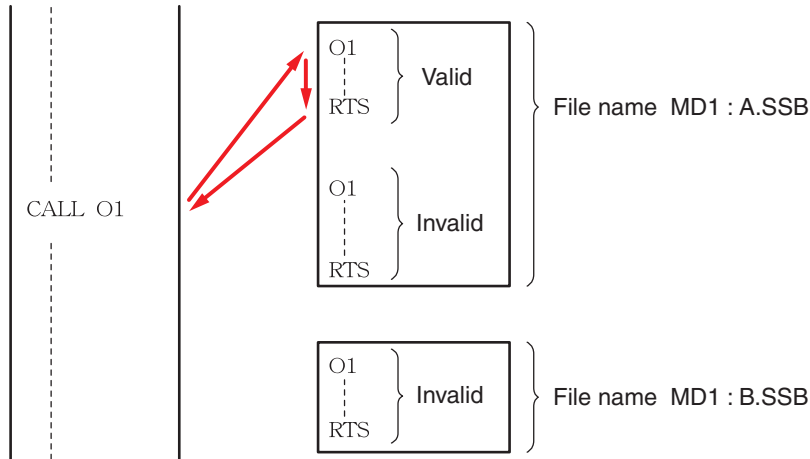
b. **Maker subprogram**

The specified subprogram is searched for in all the files in the system memory: with an extension of MSB.

- If a subprogram file contains more than one subprogram of the same name, then only the one found first is valid.

Example 1:

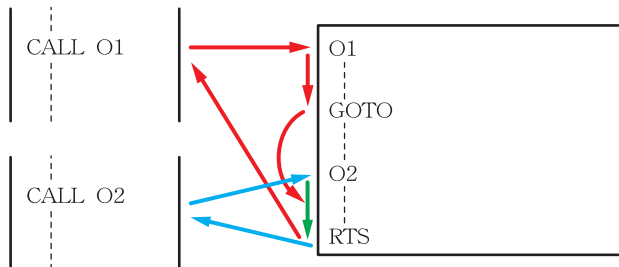
Program select command is specified without a subprogram file name.



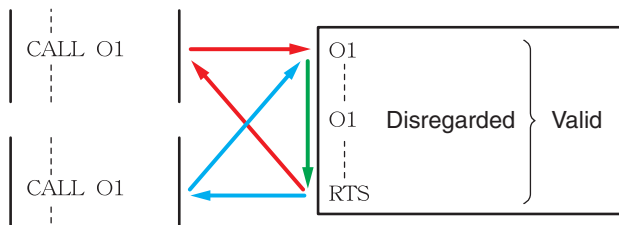
ME33018R1001200020002

Example 2:

If the located program has a program name in it.



* If a program name differing from the program called is present, use a GOTO statement to skip over it.

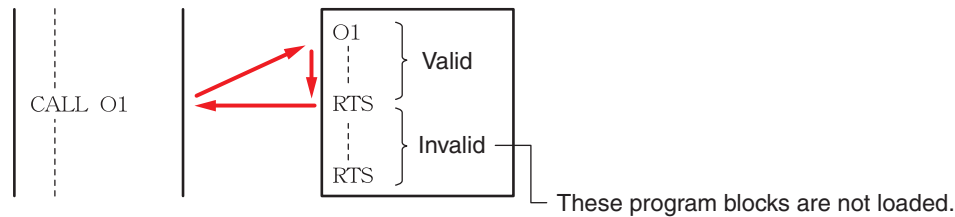


* If the same program name is used more than once, only the first program name is valid.

ME33018R1001200020003

Example 3:

If the located program has RTS at more than one place



ME33018R1001200020004

[Other]

- The allowable maximum number of subprograms that can be used or called for one program is 126.
- In the block which contains a subprogram call command, only a program name, "/" (block skip) and/or a sequence name, may be specified before the program call command. If other commands are written before it, an alarm will occur. Commands entered after the program name, specified following the program call command and up to the end of the block code in that block, are disregarded.

```
N10 X100 Y200 F300 CALL O1
      └──────────┘
      These cause an alarm
```

ME33018R1001200020005

```
N10 CALL O1 X100 Y200 F300
              └──────────┘
              Disregarded
```

ME33018R1001200020006

ME33018R0601200020006

- In the operation method A
The schedule program, main program, sub program, and library programs can be operated until their total capacity reaches the maximum operation buffer capacity.
- In the operation method B
The schedule program, library program, and subprograms can be operated until their total capacity reaches the maximum operation buffer capacity.
When the option code S is specified with the PSELECT command, an alarm will be caused by issuance of a subprogram call command (CALL; call after axis move; G code macro). However, the subprograms registered in the library can run even when the option code S is specified.
- A subprogram call command may or may not be executed in the MDI mode depending on the type of subprogram call command. An alarm occurs if a call after axis move command (MODIN, MODOUT, G100 to G110) is executed in the MDI mode. A simple call command may be executed if it is specified with the name of subprogram set by the program select command or with the library-registered subprogram name.

- Comments can not be used between a CALL statement and program name.
Example) CALL (**) OTEST

2. Simple Call (CALL)

[Function]

The simple call function executes the specified subprogram when the CALL command is specified.

[Programming format]

CALL O__ Q__ Variable-setting (variable=expression, variable=expression, ...)

O : Program name to be called

Q : Number of repetitions (Max. 9999)

If Q is not specified, "1" is assumed to apply.

ME33018R1001200030001

Variable-setting

- Set the data and variables to be passed to the called subprogram.
- In a variable-setting expression, enter a variable used in the called subprogram at the left side. The variable to be entered may be a local, common, system, or input/output variable.
- At the right side, enter the variable of the calling subprogram or data.

[Example program]

Example 1:

In the following case, the numerical values of LB, LC, and LD are:

LB = 10, LC = 10, LD = 20

O1

N1 LA = 10

N2 CALL O2 LA = 20 LB = LA

N3 LC = LA

M02

O2

N1 LD=LA

___ : Local variable of O1

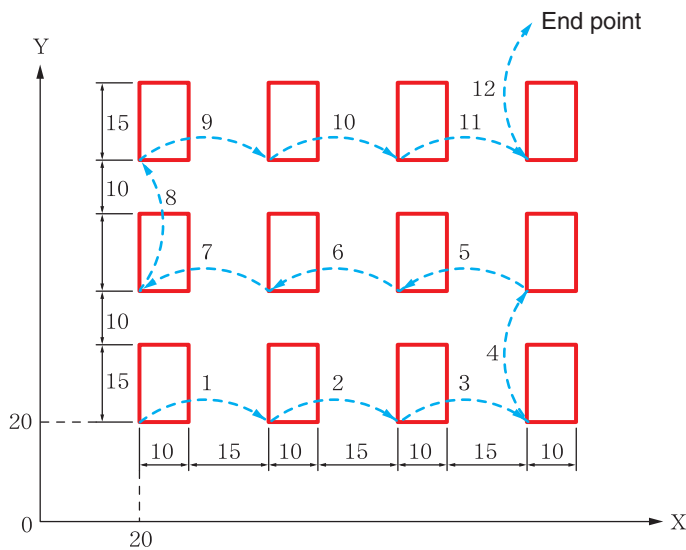
RTS

___ : Local variable of O2

- * : When the number of subprogram repetitions, specified by a Q word, is two or more, the present arguments are not reset and are executed as they are.

ME33018R1001200030002

Example 2:



ME33018R1001200030003

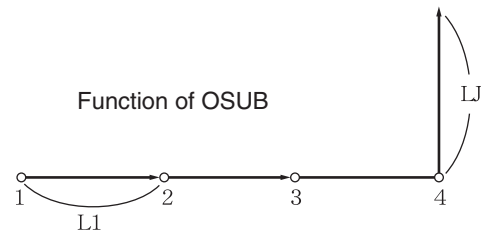
• Main program:

```
O1
N1 G90 G00 X20 Y20
N2 CALL OSUB Q3 LX=10 LI=25 LP=4
& LY=15 LJ=25 LZ=50
M02
```

ME33018R1001200030004

• Subprogram (Positioning):

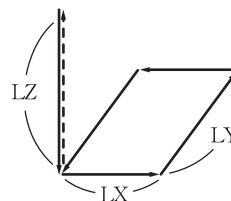
```
OSUB
N11 LC = LP
N12 CALL OSQR LX = LX LY = LY LZ = LZ
N13 LC = LC - 1
N14 IF [LC LE 0] N17
N15 G91 G00 X = LI ..... Positioning in the X-axis direction
N16 GOTO N12
N17 G91 G00 Y = LJ ..... Positioning in the Y-axis direction
N18 LI = -LI ..... Direction reversed on the X-axis
RTS
```



ME33018R1001200030005

• Subprogram (Cutting):

```
OSQR
N21 G91 G01 Z = -LZ
N22 X = LX
N23 Y = LY
N24 X = -LX
N25 Y = -LY
N26 G00 Z = LZ
RTS
```



ME33018R1001200030006

Programmers must record the following:

- Program name: OSUB
- Number of repetitions: No. of elements in the Y-axis direction
- Variables to be passed
 - LX: Cutting distance of a pattern (X-axis direction)
 - LY: Cutting distance of a pattern (Y-axis direction)
 - LZ: Depth of cut
 - LI: Distance to the next pattern (X-axis direction)
 - LJ: Distance to the next pattern (Y-axis direction)
 - LP: No. of elements in the X-axis direction

3. Subprogram Call after Axis Movement (MODIN, MODOUT)

[Function]

A call after axis move command establishes the call after axis move mode. In this mode, the specified subprogram is executed at each execution of an axis move command.

When the call after axis move command is specified with an axis move command in the same block, the axis move command is disregarded although the call after axis move mode is established.

[Programming Format]

Call after axis move :

MODIN O__ Q__ Variable-setting (variable = expression, variable = expression, ...)

ME33018R1001200040001

O: Program name

Q: Number of repetitions (Max. 9999)

If Q is not specified, "1" is assumed to apply.

Variable-setting:

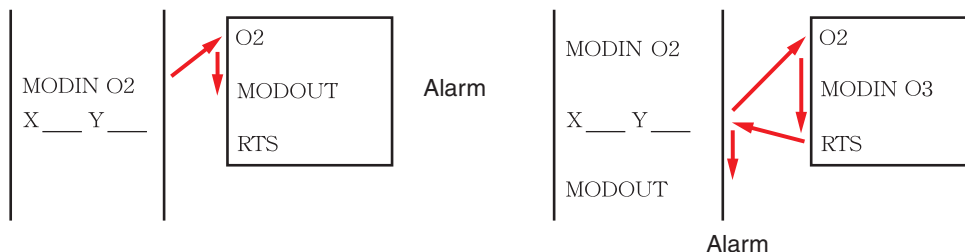
For details, refer to "Simple Call". The variable setting is executed each time the specified subprogram is called after the execution of an axis move command. However, if the subprogram is repeatedly executed according to the setting for Q, the variables are not reset in the succeeding executions.

Canceling the call after axis move mode: MODOUT

The MODOUT command cancels the MODIN command specified last, but this does not apply to MODIN commands that have already been canceled.

A MODOUT command must always be specified in the program containing the corresponding MODIN command.

An alarm occurs in the following cases:



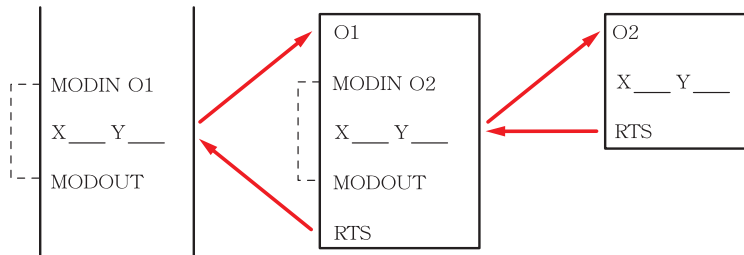
ME33018R1001200040002

[Details]

Nesting subprogram call in the call after axis move mode

It is possible to call subprograms in as many as eight levels without canceling a call after axis move command. This is called "nesting": how the call after axis move command is executed is explained below.

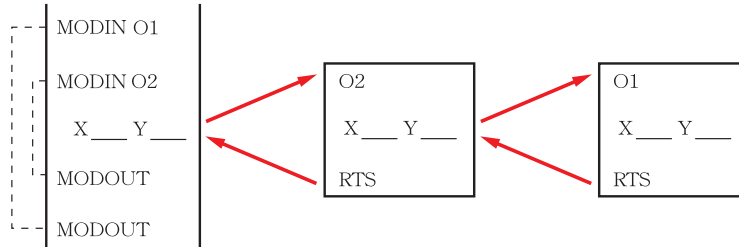
- (1) When a MODIN command is specified in the subprogram which is called by the execution of another MODIN command specified in a main program, the subprograms are executed in the manner shown below.



ME33018R1001200040003

- (2) After the specification of a MODIN command, if another MODIN command is specified in the same program, or if another MODIN command is specified in a subprogram which is called by the execution of a subprogram call command other than the first MODIN command, the subprograms are executed in the manner shown below.

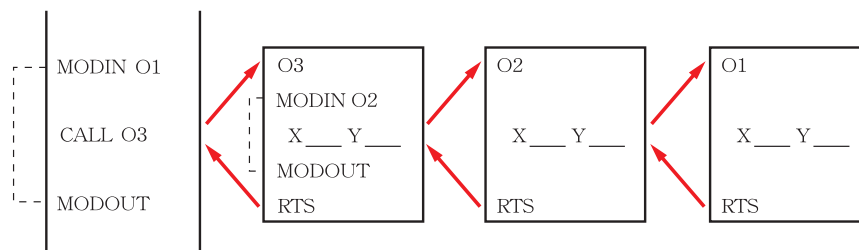
- A MODIN command is followed by another in the same program.



ME33018R1001200040004

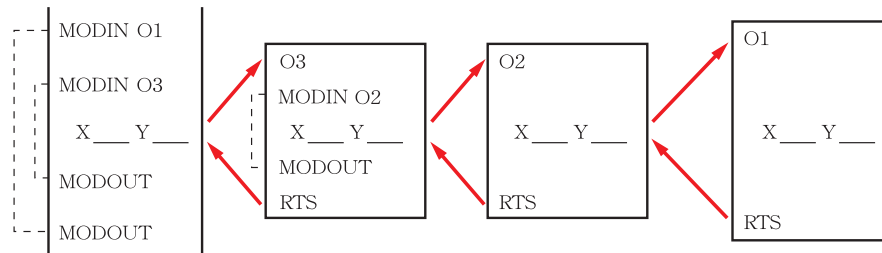
- After the specification of a MODIN command, another MODIN command is specified in the subprogram (O3) which is called by a subprogram call command other than the first MODIN command.

- a. Subprogram O3 is called by a CALL command.



ME33018R1001200040005

b. Subprogram O3 is called by other MODIN command.



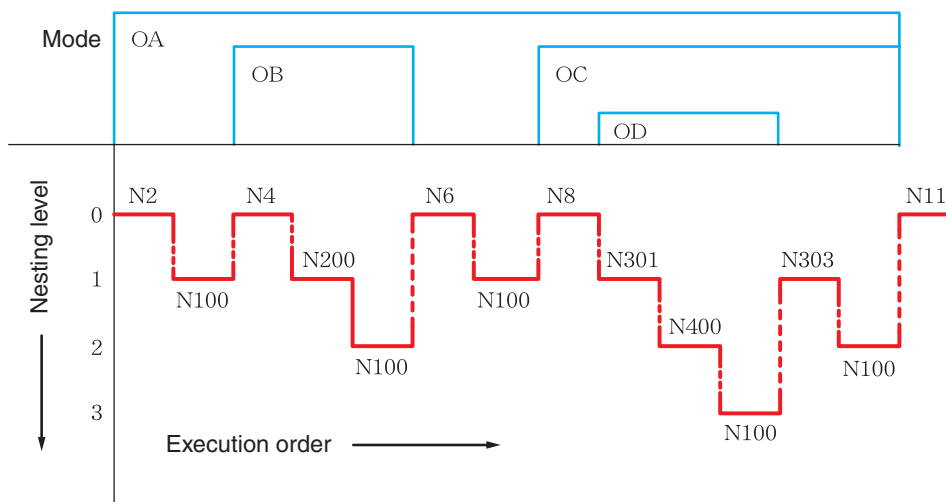
ME33018R1001200040006

In item (2) above, the relationship between O3 and O2 is the case explained in item (1) and that between O1 and O2 is the case explained in item (2).

Example:

Execution order of nested subprogram calls and the nesting level

Main program		Subprogram	
O1		OA	OD
N1	MODIN OA	N100 X__Y__	N400 X__Y__
N2	X__Y__	RTS	RTS
N3	MODIN OB		
N4	X__Y__	OB	
N5	MODOUT	N200 X__Y__	
N6	X__Y__	RTS	
N7	MODIN OC		
N8	X__Y__	OC	
N9	MODOUT	N300 MODIN OD	
N10	MODOUT	N301 X__Y__	
N11	X__Y__	N302 MODOUT	
M02		N303 X__Y__	
		RTS	



ME33018R1001200040007

SECTION 10 SUBPROGRAM FUNCTIONS

- When setting variables to be specified following a MODIN command, if a local variable is used at the right side of the setting, the following point must be taken into consideration when the subprogram is called by axis move commands specified in a program other than the one that contains the corresponding MODIN command.
 O2 is called by the execution of axis move commands in N3, then O1 is called by the execution of axis move commands in N10 of O2. Although variables are set at the time O1 is called, an alarm occurs since LA is not defined in O2.

Example:

```

N1  MODIN O1 LB = LA
N2  MODIN O2
N3  X__Y__
N4  :
      
```

→ O2
→ N10 X__Y__ → O1
→ N11 : LA undefined error

The O2 block must be specified as follows:

```
N2 MODIN O2 LA = LA
```

The same applies to the CALL command.

```

N1  MODIN O1 LB = LA
N2  CALL O3 LB = LA
N4  :
      
```

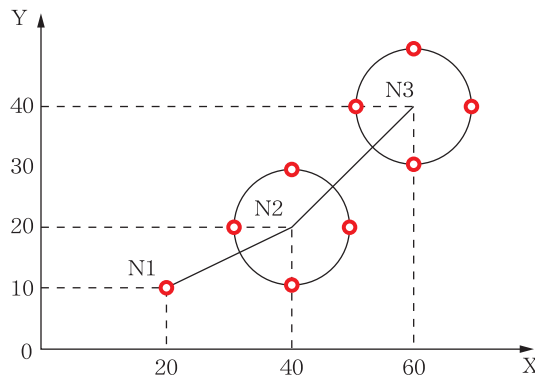
→ O3
→ N20 X__Y__ → O1
→ N21 :
→ RTS

ME33018R1001200040008

- Difference between the Subprogram Call after Axis Move Function and Fixed Cycles
 - When a drilling cycle is executed using the MODIN command, drilling is not carried out at points N2 and N3.

```

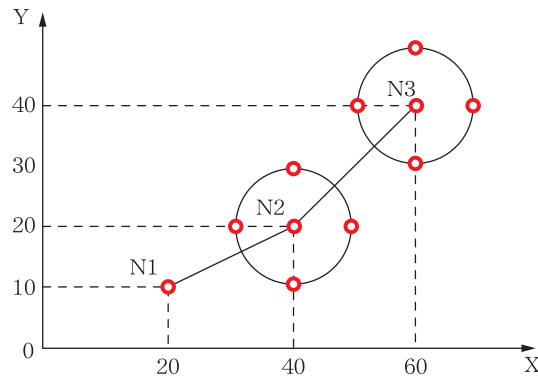
MODIN OFXC
N1 G90 G00 X20 Y10
MODIN OCYC
N2 G90 G00 X40 Y20
N3 G90 G00 X60 Y40
MODOUT
MODOUT
      
```



ME33018R1001200040009

- b. When a drilling cycle is executed using the fixed cycle function, drilling is carried out at points N2 and N3.

```
G81 .....
N1 G90 X20 Y10
MODIN OCYC
N2 G90 X40 Y20
N3 G90 X60 Y40
MODOUT
G80
```



ME33018R1001200040010

- c. It is possible to skip drilling at the N2 and N3 blocks by specifying NCYL. In the example above, subprograms OCYC and OFXC are as indicated below:

```
OCYC          OFXC
N10 G91 G00 X10 G81 Z__ R__ F__
N11 X - 10 Y10 G80
N12 X - 10 Y - 10 RTS
N13 X10 Y - 10
RTS
```

ME33018R1001200040011

- It is possible to program a fixed cycle within a subprogram called by the MODIN command. Example:

Main Program		Subprogram
:	} A fixed cycle command cannot be specified within these blocks.	OA
MODIN OA		:
X__Y__		G81
		:
MODOUT		G80
:		:
		RTS

ME33018R1001200040012

If a fixed cycle is specified in a subprogram, the fixed cycle cancel G code (G80) must be specified in the same subprogram before the RTS command is specified.

4. G and M Code Macro Functions

G Code Macro Function

[Function]

A subprogram may be called using a G code instead of the CALL or MODIN/MODOUT command. The feature to call a subprogram using a G code is called the G code macro function.

With a G code macro, since address characters are used for setting variables, it will help programmers familiar with conventional NC programming to use subprograms easily.

[Programming format]

G__ Variables-setting (<Address> <Expression>, <Address> <Expression>,)

ME33018R1001200050001

G100 : Same as MODOUT; variable setting is not necessary.
 G101 - G110 : Same as MODIN O__
 G111 - G120 : Same as CALL O__
 Only subprograms in the user subprogram file may be called.

- Program names that correspond to G101 - G120 are set using parameters. Note that program names OO000 - OO999 cannot be used.

G300 : Same as MODOUT; variable setting is not necessary.
 G301 - G349 : Same as MODIN O__
 G350 - G399 : Same as CALL O__
 Only subprograms in the make subprogram file may be called.

- For G300 - G399, subprograms to be called are determined by the system and a user cannot assign a program name for these G codes.

Variable-settings: : <Address> <Expression>, <Address> <Expression> ...
 Here address characters excluding G, M, N and O and extended address characters may be used.
 The command value is checked when a variable is actually specified.
 For referencing/updating a variable, a local variable name (address character is preceded by "P") is used.
 If a command is not specified, referencing that command does not cause an undefined error, but the command is handled as "EMPTY (undefined)".

M Code Macro Function

[Function]

A subprogram may be called using an M code instead of a CALL command. This feature is called the M code macro function.

The M code macro function does not set variables. The function must be specified in a block without other commands.

[Programming format]

M__

M201 - M210: Same as CALL O__
 Only subprograms in the user subprogram file may be called.

- Program names that correspond to M201 - M210 are set using parameters. Note that program names OO000 - OO999 cannot be used.

Common Items

- An alarm will occur if the parameter setting does not include the program name that corresponds to the specified G or M code macro, or if that name is not defined by the system. An alarm will also occur if that program is not included in the subprogram file. In this case, the alarm occurs at the time the program select command is executed. However, if the specified program is not selected by the program select command, because, for example, the program name is altered by the parameter setting after the execution of the program select command, then the alarm occurs when the G or M code macro command is executed.
- For “G” and “M” used as the G or M code macro command, values cannot be passed using a variable.
- For “G” and “M”, only numerical values are loaded when the program select command is executed and other data, if specified, will result in an alarm.

Example:

```
G = 111 } These are not loaded and an alarm occurs
G = VC1 } (G code error) when they are executed.
G112     A program named G112 is loaded.
```

ME33018R1001200050002

[Example program]
Line at angle
[Programming format]

```
G111 X__Y__I__J__K__
X, Y : Reference point (Absolute)
I    : Interval
J    : Angle
K    : Number of points
```

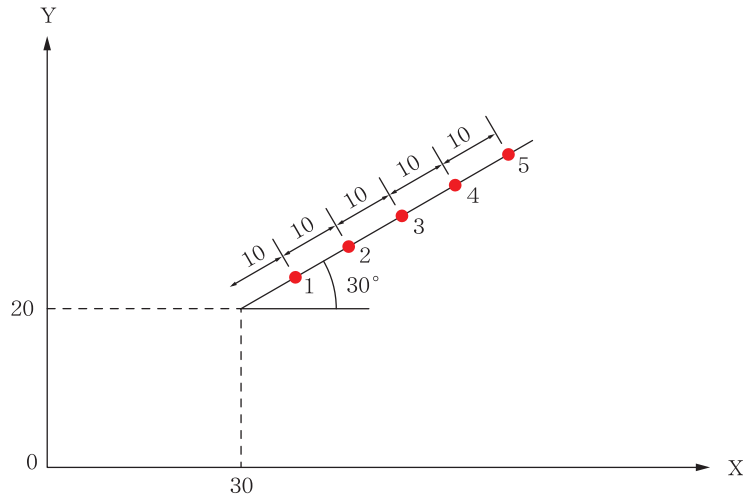
ME33018R1001200050003

This example may be expressed by the following subprogram, which assumes that OLAA is set for G111 by parameter setting.

```
OLAA
N1  LEN = 0
N2  PK = PK - I
N3  IF [ PK LT 0] N7
N4  LEN = LEN + PI
N5  G90 G00 X = LEN * COS [ PJ ] + PX
&      Y = LEN * SIN [ PJ ] + PY
N6  GOTO N2
N7  RTS
```

ME33018R1001200050004

The command "G111X30Y20I10J30K5" gives the result shown below:



ME33018R1001200050005

5. Program Call Function Using Variables

5-1. Outline

This function consists of the following two functions.

Items which are not described here are the same as those described in the conventional sub-program function. Therefore, read this manual in conjunction with the "Sub-program Function" in the Programming Manual.

- Program call function by variables
This function makes it possible to call a program by designation of an expression as well as a program number. This function, however, cannot be fully used in S operation method, remote buffer operation, or DNC-DT operation.
- Program registration function
The programs to be used in the main program are previously written in the part program, and are registered upon selection of the program. This function cannot be used in the S operation method, remote buffer operation, or DNC-DT operation.

5-2. Program Call function by Variables

This function allows you to call a program by designating a program number using an expression.

5-2-1. Application Range

There are five program call commands, as given below:

- a. CALL
- b. MOD IN
- c. G code macros (G101 to G120/G300 to G399)
- d. M code macros (M201 to M210)
- e. Maker macro call (PCIR, TAPR, etc.)

This function becomes effective by designation of the following two commands:

- a. CALL
- b. MOD IN

In this manual, the description of "program call" refers only to the program call by the above two commands.

This function becomes effective with all the operation methods. In the S operation, remote buffer operation, or DNC-DT operation, however, programs to be called must be previously registered in the library, and "PN" command has no meaning as a program number.

This function also becomes effective even in the library program.

5-2-2. Program Command Format

CALL O= [Expression] Q___ [Argument setting] (PN = ___)
 MODIN O= [Expression] Q___ [Argument setting] (PN = ___)

The designation must be made in this order, excluding "PN".
 "PN" may be designated in or before the argument setting.

o: Name (number) of program to be called

- "=" may be added and be handled like an "expression".
 An "expression" is formed with "o" followed by "=" (a space is allowed between "o" and "="). An "expression" is regarded as such even if a value is designated after "=" (Example: o = 100).
- In an expression, a program name cannot be designated. Only a program number is allowed.
- In an expression, a program number is rounded to the nearest integer (with fractions after decimal point rounded off) and is handled as a 4-digit number. "0" is deemed to be 0000. The range must be between 0000 and 9999.
- In an expression, a sub-program is not automatically registered upon selection of a program. Therefore, designate the number of a program to be used with "PN" command or the "PREG" command (described later), or previously register the program number the library. In this case, note that the "PN" command can only be used with the operation method A or B.
- In [Expression], no space should be used.
- A space is allowed between "0" and "=", or "=" and [Expression].

Q: Number of repetitions (within 9999 times)

When Q command is omitted, the frequency Q1 is automatically set.

Argument: variable = expression, variable = expression,

Data or variables to be transferred between programs are set. These settings may be omitted.

PN: Program number to be used

- The "PN" command may be omitted.
 A space is allowed between "PN" and "=", or "=" and [number].
- The "PN" command is used to designate the program number to be used, and not used for the operation.
- In the called program, a value called by "PN" can be referred to.
- The PN command in the CALL or MODIN command block is automatically registered upon selection of the program for the operation method A or B. (This is also carried out even when a program name is not designated by an "expression".)
 Aside from the PN command, there is a later-mentioned method for program registration.
- Only values (integers without sign) can be designated for "PN" command. Variables or arithmetic expressions cannot be used. An error will occur when a command value is beyond the range between 0 and 9999.
- The "PN" command is handled as a 4-digit number. If PN = 1, register O0001.
- "PN" is a local variable. "PN" can be used in the called program.

- If "PN" has already been used for another purpose, set "1" at the NC optional parameter (bit) No. 69, bit 4.
This "PN"-related function can be made invalid.
- This function only becomes effective with operation method A or B.

5-3. Program Registration Function

Subprograms to be used in the main program are previously written in the part program, and these subprograms are registered upon selection of the main program.

5-3-1. Application Range

This function only becomes effective with the operation method A or B.
It becomes effective in the library program, too.

When an operation method other than A or B is selected, this function does not work and its commands are ignored.

5-3-2. Program Command Format

PREG O____, O____,

Comma "," can be substituted by space or "/". A program name and others (except the string starting with 0 or :) are registered. No error or alarm occurs even if no program name is designated after PREG.

- A program name is handled as a character string. Therefore O1 and O0001 are different programs.
- Both the program number and program name can be designated.
- An expression cannot be used for a program number.
- PREG must be at the head of a block (sequence name may be omitted).
- G0, X or Y are not executed even if one is designated in the PREG command block.
- A program, called by the program designated by PREG, is also registered.

SECTION 11 USER TASK

1. User Task 1

User task 1 was developed to allow users to use the high speed processing function by themselves. User task 1 consists of the following three functions:

- Branch function
- Variable function
- Math function

1-1. Branch Function

- The branch function controls the execution order of the sequences within programs. It has a GOTO statement, which causes an unconditional jump, and an IF statement for a conditional jump.
- An alarm will occur if the destination of a jump does not belong to the program where a branch command is specified. When there is more than one sequence name as the destination of a jump, the jump takes place to the sequence name closer to the beginning of the program.
- In operation method A, a branch command searches for the destination of the jump from the beginning of a program. Accordingly, the closer the destination is to the end of the program, the longer the time it will take to execute a branch command. In contrast, the location of the destination has no influence on the branch command execution time when operation method B is used.
- A block which includes a branch command cannot accommodate anything other than the program name, "/" (block skip) and sequence name.

```
N10 X100 Y200 F300 GOTO N20
      └──────────┘
      Alarm
```

ME33018R1001300020001

- The commands programmed after the destination of the jump up to the end of the block code in that block are all disregarded.

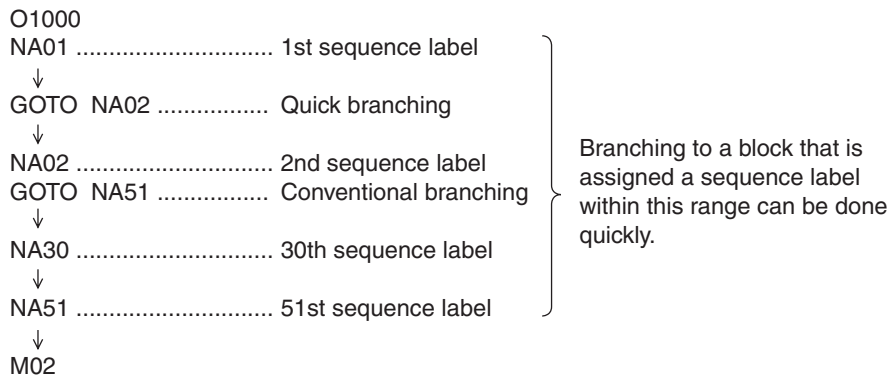
```
N10 GOTO N20 X100 Y200 F300
              └──────────┘
              Disregarded
```

ME33018R1001300020002

- An alarm occurs if a branch command is executed in the MDI mode.

- If operation method A is selected by the parameter setting, branching can be done quickly by designating sequence labels as the destination of a branch command. However, this quick branching is possible only up to 30 sequence labels from the beginning of a program. For sequence labels after these 30 sequence labels, the destination of a branch command is searched for from the beginning of a program, as is done conventionally.

Example:



* : Sequence label: A sequence name which contains alphabetic characters.

ME33018R1001300020003

- If operation method B is selected by parameter setting, the restrictions indicated below apply to the main program.
 - a. When the S option is specified in the program selection command, the execution of a branch command results in an alarm.
 - b. The sequence name of the destination of a jump must be a label, which is an alphanumeric string. A sequence number, which is a numeric string, will result in an “undefined” error.
 - c. Up to 30 sequence labels may be used, including ones not specified as the destination of a jump. If the number of sequence labels used exceeds this limit, an alarm in program selection occurs. It is recommended to use sequence labels only for blocks which will be the destination of a jump.

(1) GOTO Statement

[Function]

This is a branch function which causes a jump unconditionally.

[Programming format]

GOTO □N__

ME33018R1001300020004

N: Specify a sequence name or a sequence label of jump destination.

A GOTO statement and a sequence name (sequence label) must be delimited by a space.

(2) IF Statement

[Function]

This is a branch function which causes a conditional jump. If the condition is satisfied, the execution sequence jumps to the specified destination. If the condition is not satisfied, the execution sequence moves to the next sequence.

[Programming format]

IF <qualification> N____
 IF <qualification> GOTO □N____

ME33018R1001300020005

There are six types of qualifications available as indicated below.

Operator	Meaning	Example of IF Statement	Contents	Rule
LT	Less Than, <	IF [VC1 LT 5] N100	Jump to N100 when VC1 is less than 5.	Provide a space on either side of the operator.
LE	Less than or Equal to, ≤	IF [VC1 LE 5] N100	Jump to N100 when VC1 is less than or equal to 5.	
EQ	Equal to, =	IF [VC1 EQ 5] N100	Jump to N100 when VC1 is equal to 5.	
NE	Not Equal to, ≠	IF [VC1 NE 5] N100	Jump to N100 when VC1 is not equal to 5.	
GT	Greater Than, >	IF [VC1 GT 5] N100	Jump to N100 when VC1 is greater than 5.	
GE	Greater than or Equal to, ≥	IF [VC1 GE 5] N100	Jump to N100 when VC1 is greater than or equal to 5.	

1-2. Variable Function

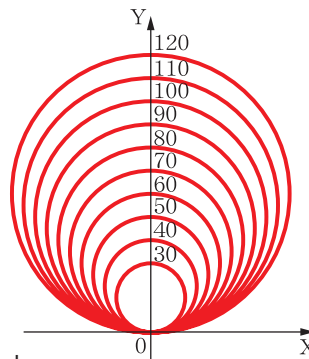
[Function]

The variable function allows the use of variables in the data section of an expression such as $X = VC1$ instead of directly specifying a numerical value such as $X100$. This gives programs more flexibility and versatility, since assigning numeric values to variables permits the same program to be used for machining similar types of workpieces.

Example:

Using the following program will facilitate machining the geometry shown below.

```
N1 G90 G00 X0 Y0
N2 G02 J = VC1
N3 VC1 = VC1 + VC2
N4 VC3 = VC3 - 1
N5 IF [VC3 GT 0] N2
N6 M02
```



Before starting the operation, set the variables using parameters as follows:

VC1 = 60 Radius of the first circle to be machined
 VC2 = -5 Incremental radius reducing value (Pitch of the radius)
 VC3 = 10 Number of circles to be machined

ME33018R1001300030001

[Details]

- Variables may be assigned to addresses (with an exception of O and N) and extended addresses. An equal symbol "=", blank or HT should be used as a delimiter to be placed between an address and a variable.

When assigning a variable to a G code or an extended address, always use the equal symbol "=".

Example: $G = VC1$ (If $VC1 = 1$, is interpreted as $G1$.)

- If a value outside the allowable limits of an individual address is assigned, an alarm occurs. For addresses that require the use of an integer, a decimal fraction is rounded off if a real number is used.

- EMPTY

The value of an undefined variable is represented by "EMPTY". Particularly, the local variables beginning with "P" are assigned "EMPTY" if a default is set. Other local variables not beginning with "P" are not assigned initial values unless they are defined.

- How "EMPTY" is interpreted:

- a. When a variable is assigned to an address
Assigning an undefined variable is equivalent to omitting the address. The use of an undefined variable in the right member causes an alarm.

VC1 = EMPTY	VC1 = 0
G90 X100 Y = VC1	G90 X100 Y = VC1
↓	↓
G90 X100	G90 X100 Y0

ME33018R1001300030002

- b. When an undefined variable is used in the operational expression
“EMPTY” is interpreted as 0, except when no operational symbols are used.

VC1 = EMPTY	VC1 = 0
VC2 = VC1	VC2 = VC1
↓	↓
VC2 = EMPTY	VC2 = 0
VC2 = +VC1	VC2 = +VC1
↓	↓
VC2 = 0	VC2 = 0
VC2 = VC1 * VC1	VC2 = VC1 * VC1
↓	↓
VC2 = 0	VC2 = 0

ME33018R1001300030003

- c. When an undefined variable is used in the qualification
“EMPTY” is different from 0 only when EQ or NE is used.

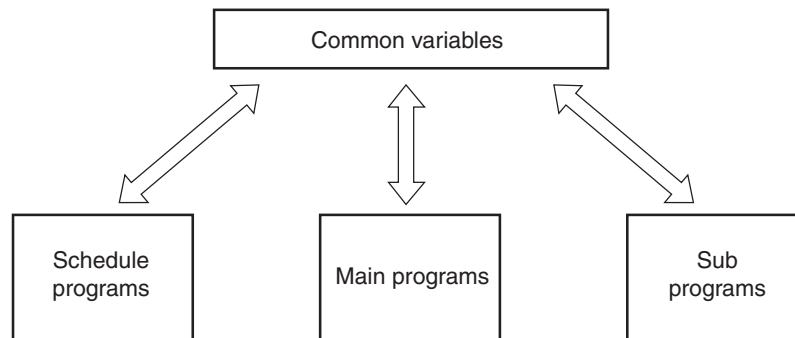
VC1 = EMPTY	VC1 = 0
N1 IF [VC1 EQ EMPTY] N10	N1 IF [VC1 EQ EMPTY] N10
N2 ↓	N2 ↓
Branching to N10	To the next N2
N1 IF [VC1 NE 0] N10	N1 IF [VC1 NE 0] N10
N2 ↓	N2 ↓
Branching to N10	To the next N2
N1 IF [VC1 GE EMPTY] N10	N1 IF [VC1 GE EMPTY] N10
N2 ↓	N2 ↓
Branching to N10	Branching to N10
N1 IF [VC1 GE 0] N10	N1 IF [VC1 GE 0] N10
N2 ↓	N2 ↓
To the next N2	To the next N2

ME33018R1001300030004

- **Array Variables**
An array is a set of data having the same elements. The array name should be immediately followed by a subscript enclosed by [] to represent a specific element.
Variables that permit the use of an array
- An array may not be used for local variables.
 - a. Common variables VC[n]
 - b. Some system variables Example: VZOFX [n], etc. (user task 2)
 - c. I/O variables VDIN[n] or VDOUT[n] (user task 2)
 - *: "n" is a subscript
 - Assume that an arithmetic expression is used. The subscript used in the arithmetic expression is called a subscript expression. If an array variable is used in the subscript expression, an alarm occurs.

1-2-1. Common Variables

Variables which are used in common for schedule programs, main programs and subprograms are referred to as common variables, and they may be referenced or updated in any of these programs.



ME33018R1001300040001

[Programming format]

Type 1 :

V	C	Numeric
---	---	---------

Numeric : 1 to 200 (standard)
1 to 1000 (optional)

Type 2 :

V	C	[Expression]
---	---	---	------------	---

Result of the expression : 1 to 200 (standard)
1 to 1000 (optional)

ME33018R1001300040002

[Details]

- In the standard specification there are 200 common variables from VC1 to VC200; optionally, this number of common variables can be expanded to 1000, from VC1 to VC1000. VC001 and VC01 are interpreted as VC1.
- Common variables can be set using parameters.
- With type 1, specify the common variable number directly. With type 2, specify the common variable number by expressions according to the array format.
Example of type 2 designation:
VC[1] Interpreted as VC1
VC[VC1 + 1] Interpreted as VC11, where VC1 = 10
- If the result of the expression is outside the range 1 to 200 with the standard specification, or 1 to 1000 with the optional specification, an alarm occurs.

- Backup range can be specified with the following parameters.
<NC optional parameter (word).>

NO.	Description	Setting range	Default
74	First common variable number to be backed up (Zero (0) is regarded as one (1).)	0 to 10000	0
75	Last common variable number to be backed up (Zero (0) is regarded as the maximum common variable.)	0 to 10000	0

All the common variables are to be backed up if "First = Last = 0" is specified. (Default)

No common variables are to be backed up if the first number specified for backup is larger than the last number. (First > Last)

Common variables that are not backed up will be cleared (EMPTY) when the system is reset.

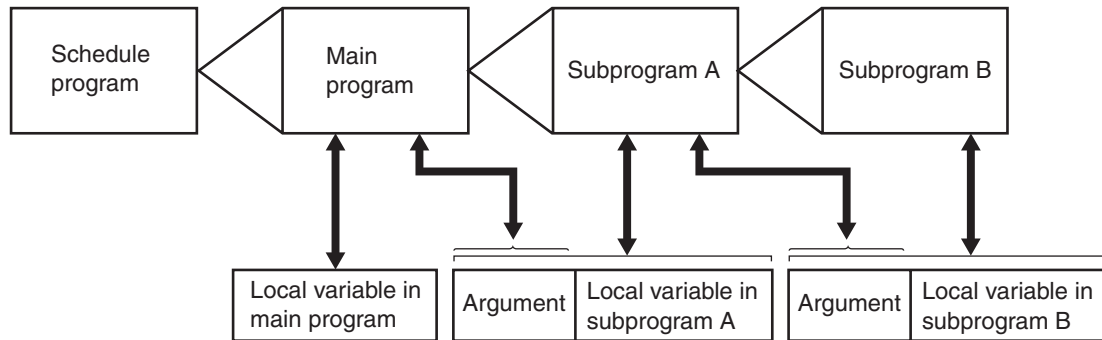
- Whether to clear (EMPTY) the backed up variables upon resetting can be set with the following parameter.
<NC optional parameter (bit).>

NO.	Bit	Description	<input checked="" type="checkbox"/>	<input type="checkbox"/>
72	4	Clear (EMPTY) the non-backed up variables in the area preceding the backed up area.	Clear	Do not clear (Default)
77	4	Clear (EMPTY) the non-backed up variables in the area following the backed up area.	Clear	Do not clear (Default)

1-2-2. Local Variables

Local variables may be used in a main program or a subprogram. They are valid only for a particular program and may be set, referenced, or updated only in this particular program. Therefore, it is not permissible to reference or update a local variable set in a certain program from other programs.

When the variable is set by an argument calling a subprogram, the left part local variable, which is a variable of a called subprogram, may be set by the calling subprogram and referenced or updated by the called subprogram. This variable can be used to pass arguments.



ME33018R1001300050001

[Format]

Type 1 :

Alphabet	Alphabet	Alphanumeric (1 or 2 characters)
----------	----------	----------------------------------

}

Alphabet other than O, N, V and P

Type 2 :

P	Alphabet	Alphanumeric (1 or 2 characters)
---	----------	----------------------------------

ME33018R1001300050002

[Details]

- A local variable may be set by defining less than 5 characters except for reserved words to the left of the "=" symbol. The local variables for the subroutine are erased after the subprogram has been executed as many times as specified. A total of 255 local variables may be set, including types 1 and 2. All of the local variables are erased by power ON/OFF, NC reset, etc.
- Local variables are set or updated in the same manner regardless of the local variable type, type 1 or type 2. When referenced (defined at the right part of "="), however, the processing differs between these two types of local variables if the local variable to be referenced has not been set. Although an alarm occurs with type 1 local variables, a local variable is set with "EMPTY" defined as a value when type 2 local variable is used.

- Addresses specified to assign arguments of G code macro instructions are set using a variable name with "P" at its start and are regarded as a local variable for type 2.

Example:

Specifying "G111 X100 Y200 P5;"

Set local variables as PX = 100, PY = 200, and PP = 5. If special processing is required for the omitted address, program the required processing in the subprogram that is called by a G code macro.

```

N1 IF [PI NE EMPTY] N2
    Special processing to be executed if address I is omitted
N2 IF [PJ NE EMPTY] N3
    Special processing to be executed if address J is omitted
N3 .....
  
```

} Subprogram called by
a G code macro

ME33018R1001300050003

1-3. Math Functions

Math functions allow to set arithmetic expressions (+, -, *, /) directly when specifying variables and address characters (X, Y, Z, I, K, ...).

[Programming format]

Address character or variable = Expression

Expression

Operator	Math Name	Example
+	Positive sign	+1234
-	Negative sign	-1234
+	Addition	X = 12.3+VC1
-	Subtraction	X = 12.3-VC1
*	Multiplication	X = VC1*10
/	Division	X = VC1/10

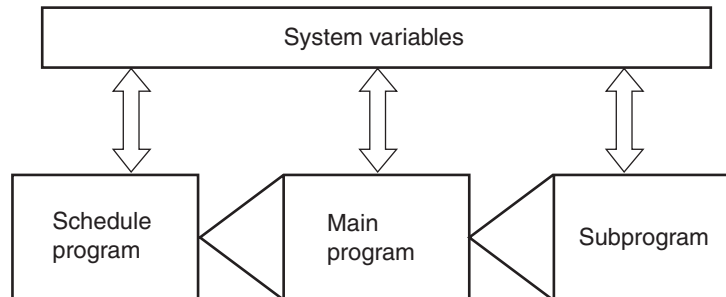
Example:

Conventional programming X135

Programming using math function X = 100 + XP2 XP2 = 35

1-4. System Variables

Variables which are determined by the system are referred to as system variables, and they may be referenced or updated in schedule programs, main programs and subprograms. A system variable is referenced or updated after the execution of the sequence that immediately precedes the referencing or updating sequence.



ME33018R1001300070001

[Programming format]

Type 1 :

V	Alphabet	Alphanumeric (Within 6 characters)
---	----------	------------------------------------

Type 2 :

V	Alphabet	Alphanumeric (Within 6 characters)	[Expression]
---	----------	------------------------------------	---	------------	---

The second alphabetic character, that follows V, must not be "C".

ME33018R1001300070002

[Details]

- The type of each system variable is determined and if the type used for specifying a specific parameter differs from the predetermined type, an alarm occurs.
- Some system variables may be set in the zero set, tool data and/or parameter mode.
- The system variables are classified into the following three types:
 - a. Read and write system variables
 - b. Read and write system variables requiring care in writing
 - c. Read only system variables
Details of these system variables are described below.

1-4-1. Read/Write System Variables

(1) Zero Offset

VZOF* [expression]

*: Axis name X - Z, U - W, A - C

Expression: Work coordinate system number

Allowable range: 1 to number of work coordinate system sets

The zero offset values for the work coordinate system indicated by the expression can be read and written. For this operation, the unit system is as set by NC optional parameter (INPUT UNIT SYSTEM). The axis is determined by the designated axis name.

- Example 1:

Writing X-axis zero offset value at No. 10

VZOFX[10] = 20

- μm unit system
Zero offset value at No. 10 = 20 μm
- mm unit system
Zero offset value at No. 10 = 20 mm

- Example 2:

Reading X-axis zero offset value at No. 10

VC1 = VZOFX[10] (Zero offset value at No. 10 is 20 mm)

- μm unit system
VC1 = 20000
- mm unit system
VC1 = 20
For details, refer to "General Rule for Conversion between Inches and Millimeters".
(Note: "inch system" refers to the English measurement system.)

(2) Tool Length Offset Value

VTOFH[expression]

Expression: Tool length offset number

Allowable range: 1 to number of tool data sets

The tool length offset values for the tool length offset number indicated by the expression can be read and written. For this operation, the unit system is as set by NC optional parameter (INPUT UNIT SYSTEM).

- Example 1:

Writing tool length offset value at No. 10

VTOFH[10] = 20

- μm unit system
Tool length offset value at No. 10 = 20 μm
- mm unit system
Tool length offset value at No. 10 = 20 mm

- Example 2:

Reading tool length offset value at No. 10

VC1 = VTOFH[10] (Tool length offset value at No. 10 is 20 mm)

- μm unit system
VC1 = 20000

- mm unit system
VC1 = 20
For details, refer to “General Rule for Conversion between Inches and Millimeters”.
(Note: “inch system” refers to the English measurement system.)

(3) Cutter Radius Compensation Values

VTOFD[expression]

Expression: Cutter radius compensation number

Allowable range: 1 to number of tool data sets

The cutter radius compensation values for the cutter radius compensation number indicated by the expression can be read and written. For this operation, the unit system is as set by NC optional parameter (INPUT UNIT SYSTEM).

- Example 1:
Writing cutter radius compensation value at No. 10
VTOFD[10] = 20
 - μm unit system
Cutter radius compensation value at No. 10 = 20 μm
 - mm unit system
Cutter radius compensation value at No. 10 = 20 mm
- Example 2:
Reading cutter radius compensation value at No. 10
VC1 = VTOFD[10] (Cutter radius compensation value at No. 10 is 20 mm)
 - μm unit system
VC1 = 20000
 - mm unit system
VC1 = 20
For details, refer to “General Rule for Conversion between Inches and Millimeters”.
(Note: “inch system” refers to the English measurement system.)

(4) Positive Programmable Travel Limit

VPPL*

*: Axis name X to Z, U to W, A to C

The programmable travel limit in the positive direction for the axis indicated by the axis name can be read and written. This sets the data for user parameter “P PROG LIMIT WRK” that is accessible in the parameter setting mode. For this operation, the unit system is as set by NC optional parameter (INPUT UNIT SYSTEM). The value to be set is determined in the work coordinate system currently selected.

[Supplement]

It is not permissible to set the positive programmable travel limit beyond the position set as the travel end limit in the positive direction

- Example 1:
Writing programmable travel end limit (+) of the X-axis
VPPLX = 500
 - μm unit system
Programmable travel end limit (+) = 500 μm

- mm unit system
Programmable travel end limit (+) = 500 mm
- Example 2:
Reading programmable travel end limit (+) of X-axis
VC1 = VPPLX (Programmable travel end limit (+) is 500 mm)
 - μm unit system
VC1 = 500000
 - mm unit system
VC1 = 500
For details, refer to “General Rule for Conversion between Inches and Millimeters”.
(Note: “inch system” refers to the English measurement system.)

[Supplement]

Although the programmable travel end limit (+) is set in the work coordinate system, the travel end limit (+) is set in the machine coordinate system.

(5) Negative Programmable Travel Limit

VNPL*

*: Axis name X to Z, U to W, A to C

The programmable travel limit in the negative direction for the axis indicated by the axis name can be read and written. This sets the data for user parameter “N PROG LIMIT WRK” that is accessible in the parameter setting mode. For this operation, the unit system is as set by NC optional parameter (INPUT UNIT SYSTEM). The value to be set is determined in the work coordinate system currently selected.

[Supplement]

It is not permissible to set the negative programmable travel limit beyond the position set as the travel end limit in the negative direction.

- Example 1:
Writing programmable travel end limit (–) of the X-axis
VNPLX = 500
 - μm unit system
Programmable travel end limit (–) = 500 μm
 - mm unit system
Programmable travel end limit (–) = 500 mm
- Example 2:
Reading programmable travel end limit (–) of X-axis
VC1 = VNPLX
If programmable travel end limit (–) is 500 mm,
 - μm unit system
VC1 = 500000
 - mm unit system
VC1 = 500
For details, refer to “General Rule for Conversion between Inches and Millimeters”.
(Note: “inch system” refers to the English measurement system.)

[Supplement]

Although the programmable travel end limit (–) is set in the work coordinate system, the travel end limit (–) is set in the machine coordinate system.

(6) Printer Control**VPCNT**

Setting range: Binary, 8 bits (1 byte); 0 – 255

This is used with a print statement.

To change a page, for example, set the “change page” code and output it to the printer along with the print statement.

If this system variable is printed using the print statement, the value set for VPCNT is output. Nothing is output when RS232C is not selected by the printer control variable VPRT.

When optional parameter (bit) No. 8, bit 4 (8-bit JIS) is OFF, data of 0 to 127 (\$00 - \$7F) is output with a parity bit (bit 7).

When the “PRINT VPCNT” command is executed, the line feed command is output after the output of the printer control code. Therefore, to output only the printer control code, execute the “SPRINT VPCNT” command.

- Example:

To output only “code 31 (1F in hexadecimal)” to the printer without line feed, enter as indicated below.

VPCNT=31

SPRINT VPCNT

(7) Automation Specification Judgment Result 1**VOK1**

Setting range: Binary, 8 bits (1 byte); 0 - 6, 10 - 16

This is used with a print statement.

It is convenient to use this system variable to print the result of gauging.

The relationship between the setting value for VOK1 and the print output is indicated below. In any case, the output comprises three characters.

PRINT VOK1

Set value	Print Contents & Display
VOK1 = 0 or 10	[] (three spaces)
= 1 or 11	[OK]
= 2 or 12	[+OK]
= 3 or 13	[+NG]
= 4 or 14	[-NG]
= 5 or 15	[NG]
= 6 or 16	[] (three spaces)
= EMPTY	[] (three spaces)

ME33018R1001300080001

With the setting of “10” to “16”, the output is in enlarged characters.

On the display screen, the result is displayed in characters of the same size whether the setting is 0 – 6 or 10 – 16.

When the setting calls for output in enlarged characters, the following is output.

Epson specification	\$0E
Okuma PP-5000 specification	\$1F
In other specifications	No special results

(8) Automation Specification Judgment Result 2

VOK2

Setting range: Binary, 8 bits (1 byte); 0 – 255

This is used with a print statement.

It is convenient to use this system variable to print the total result of gauging.

The relationship between the setting value for VOK2 and the print output is indicated below. In any case, the output consists of eight characters.

PRINT VOK2

Set value	Print Contents & Display	
VOK2	= 1	[TOTAL OK]
	= 2	[TOTAL NG]

ME33018R1001300080002

[Supplement]

The setting range is 0 – 255 if the setting is made only for VOK2. For use in combination with the PRINT statement, the setting must be “1” or “2”; setting another value causes an alarm.

(9) Gauging Number

VNUM

Setting range: 0 – 9999

This is used with a print statement.

It is convenient to use this system variable to print the gauging point number.

The relationship between the setting value for VNUM and the print output is indicated below. In any case, the output consists of seven characters.

PRINT VNUM

Set value	Print Contents & Display	
VNUM	= 0	[NO. 0]
	= 1	[NO. 1]
	= 10	[NO. 10]
	= 100	[NO. 100]
	= 1000	[NO. 1000]
	= EMPTY	[NO. 0]

ME33018R1001300080003

(10) Printer Control

VINTG

Setting range: 0 – ±99999.999

This is used with a print statement.

The data output in response to the print statement are all treated as a floating point variable; in the mm unit system, the data is displayed up to the third place right of the decimal point and in the inch unit system, the data is displayed up to the fourth place right of the decimal point. This system variable can be used conveniently to display the data in integer form.

The relationship between the setting value for VINTG and the print output is indicated below. In any case, the output consists of twelve characters.

PRINT VINTG

	Set value	Print Contents & Display
VINTG	= - 99999999	[- 99999999]
	= 0	[0]
	= 99999999	[99999999]
XX	= 0	

ME33018R1001300080004

When PRINT XX is executed, the displayed data will be [0.000] in mm unit system.

(11) Printer Control

VPRT

Setting range: Binary, 8 bits (1 byte); 0 – 255

Specify the output destination and output unit for outputting the character string set with the print statement.

Output Destination	Unit of Variable	
	Minimum Unit System	Standard Unit System
No output	VPRT = 0	VPRT = 4
RS232C only	VPRT = 1	VPRT = 5
Display screen only	VPRT = 2	VPRT = 6
RS232C and display screen	VPRT = 3	VPRT = 7

The display screen indicates the personal screen in the operation mode.

Output to the RS232C is valid only when the setting for NC optional parameter (bit) No. 2, bit 4 is ON.

The setting is cleared to "0" when the power is turned ON. However, the set state is not influenced by the resetting operation.

Concerning the "minimum unit system" and the "standard unit system" for output, the system selection is valid only for floating point variables, and the same data is output in different formats according to the selected unit system. The number of characters is "12" in either case. Some data which can be displayed with the minimum unit system selection cannot be displayed in the standard unit system since the number of digits exceeds eight. In this case "+OVERFLOW" or "-OVERFLOW" is displayed.

- Example:

VC1 = 12.34

PRINT VC1

	Set value	Display
VPRT	= 0 - 3	"mm" unit system [0.012]
	=	"inch" unit system [0.0012]
	= 4 - 7	"mm" unit system [12.340]
	=	"inch" unit system [12.3400]

ME33018R1001300080005

(12) Tool Length/Breakage Switching Flag

VFST

Setting range: Binary, 8 bits (1 byte); 0 – 255

The basic operation mode for automatic tool length offset and automatic tool breakage detection can be designated.

The relationship between each bit and the operation modes is indicated in the table below.

Bit No.	Operation Mode
Bit 7	1:The tool length referenced to the spindle nose is used as the tool length offset value. 0:The length of individual tools relative to the standard tool used for setting the work coordinate system is used as the tool length offset value.
Bit 6	To be "0".
Bit 5	To be "0".
Bit 4	To be "0".
Bit 3	1:The X-axis does not move in positioning to the touch sensor position. Set "1" when the touch sensor position is set independently of the X-axis.
Bit 2	1:Automatic tool length offset / tool breakage detection is consecutively executed in the Z-axis direction and then in the Y-axis direction.
Bit 1	1:Automatic tool length offset / tool breakage detection is executed in the Y-axis direction.
Bit 0	1:Automatic tool length offset is executed. 0:Automatic tool breakage detection is executed.

(13) Graphic Color Specification

VGCLR

The color for color graphic display can be designated.

Color designation:

VGCLR = n

n:Integer (0 to 9)

This system variable is used to designate the color for tool path display.

n = 0:Cancel of designated color

n = 1:Blue

n = 2:Green

n = 3:Light blue

n = 4:Red

n = 5:Purple

n = 6:Yellow

n = 7:White

n = 8:Black

n = 9:Not displayed

(14) Synchronized Tapping Torque Monitor Parameter No.

VTMNO

Setting range: 1 – 5

The tapping torque monitor parameter number during synchronized tapping can be read/written. Since VTMNO is backed up, the previously used value remains valid if no new value is specified.

- Example 1:

Writing the synchronized tapping torque monitor parameter No. 3

VTMNO = 3

- Example 2:
Reading the synchronized tapping torque monitor parameter No. 3
VC1 = VTMNO

(15) Spindle Overload Monitor Parameter No.

VSLNO

Setting range: 1 – 5

For the spindle overload monitor function, the spindle overload monitor parameter number can be read/written by indicating the parameter number. Since VSLNO is backed up, the previously used value remains valid if no new value is specified.

- Example 1:
Writing the spindle overload monitor parameter No. 3
VSLNO = 3
- Example 2:
Reading the spindle overload monitor parameter No. 3
VC1 = VSLNO

(16) F1-digit Parameter Feedrate

VPF1F[expression]

Expression: Specified number for F1-digit parameter feed

Allowable value: 1 – 9

The feedrate for the specified F-1 digit parameter feed number can be read/written.

- Example 1:
Writing feedrate for F-1 digit parameter feed No. 3.
VPF1F[3] = 20
 - μm unit system
Feedrate of 20 μm is set for parameter No. 3.
 - mm unit system
Feedrate of 20 mm is set for parameter No. 3.
- Example 2:
Reading the feedrate set for F-1 digit parameter feed No. 3.
Assume that feedrate set for F-1 digit parameter feed No. 3 is 20 mm.
VC1 = VPF1F[3]
 - μm unit system
VC1 = 20000
 - mm unit system
VC1 = 20

(17) F1-digit Parameter Maximum Value

VPF1M[expression]

Expression: Specified number for F1-digit parameter feed

Allowable value: 1 – 9

The maximum feedrate for the specified F-1 digit parameter feed number can be read/written.

- Example 1:
Writing the maximum feedrate for F-1 digit parameter feed No. 3.
VPF1M[3] = 20
 - μm unit system
The maximum feedrate of 20 μm is set for parameter No. 3.

- mm unit system
The maximum feedrate of 20 μm is set for parameter No. 3.

- Example 2:
Reading the maximum feedrate set for F-1 digit parameter feed No. 3.
Assume that the maximum feedrate set for F-1 digit parameter feed number 3 is 20 mm.
VC1 = VPF1F[3]

- μm unit system
VC1 = 20000
- mm unit system
VC1 = 20

(18) F1-digit Parameter Accel./Decel. Data

VPF1C

Setting range: 0 – 4000000

The acceleration/deceleration data of the F1-digit parameter can be read/written.

The acceleration/deceleration data is usually set for NC optional parameter (long word) No. 22.

The data is read or written in the minimum unit system.

- Example 1:
Writing the acceleration/deceleration data of the F1-digit parameter
VPF1C = 20
- Example 2:
Reading the acceleration/deceleration data of the F1-digit parameter
VC1 = VPF1C
Assume that the acceleration/deceleration data set for this parameter is 20 μm .
VC1 = 20 μm

(19) Tool Management Data

VTLD* [expression]

*: 1 to 8

Expression: Tool offset number

Allowable range: 1 to number of tool data sets

Reading/writing of the tool management data with the expression can be indicated. The objective to be read or written is designated by “*”.

The following explains how the data is handled:

- a. VTLD1: Tool group number
- b. VTLD2: Tool life management mode

Mode	Criterion for Tool Life Judgment	Replacement to Spare Tool when Tool Life is Expired
1	Tool life is judged on the basis of the accumulated cutting time in which the cutting tool is fed at a cutting feedrate.	When the T command calling the tool life expired tool is specified next
2		Not used
3		Tool is not replaced by spare tool and tool life expired tool continues to be used
4	Tool life is judged on the basis of count data such as total machined number of holes.	When the T command calling the tool life expired tool is specified next
5		Not used
6		Tool is not replaced by spare tool and tool life expired tool continues to be used
0	Tool life judgment is not conducted.	-

- c. VTLD3: OK/NG flag
 - Bit 0 to 4: Used for other functions (Never attempt to change)
 - 5: NG1
 - 6: NG2
 - 7: NG3
- d. VTLD4: Tool life flag
 - Bit 0: Tool life
 - 1: Tool wear
 - 2: Overload
 - 3: Torque
 - 4:
 - 5: Breakage
 - 6:
 - 7: User
- e. VTLD5: Second tool offset number
- f. VTLD6: Third tool offset number
- g. VTLD7: Tool life management setting value
 - h1: 2-byte data

- When tool life is judged on the basis of accumulated cutting time (tool life mode: 1 to 3):
 $0 \leq h1 \leq 32767$ (unit: min.)
- When tool life is judged on the basis of count data (tool life mode: 4 to 6):
 $0 \leq h1 \leq 32767$
- When tool life management is not executed (tool life mode: 0):
No restriction

h. VTLD8: Remaining tool life value
h2: 2-byte data

- When tool life is judged on the basis of accumulated cutting time (tool life mode: 1 to 3):
 $-32768 \leq h2 \leq 32767$ (unit: min.)
- When tool life is judged on the basis of count data (tool life mode: 4 to 6):
 $-32768 \leq h2 \leq 32767$
Always $h3 = 0$
- When tool life management is not executed (tool life mode: 0):
No restriction

(20) MOP Tool Number

VMPT

Setting range: 0 – 9 (integer)

This system variable sets the classification number of the MOP-TOOL tool data.

When using the same tool for rough machining and finishing, the selection of MOP-TOOL tool data can be changed by setting different numbers for VMPT.

- Example:
Enter VMPT = 0 in the NC program for rough machining and VMPT = 1 for finishing with the following tool number set as the tool data for MOP-TOOL.
Tool number in MOP-TOOL data = 50-0 Tool data for rough machining
Tool number in MOP-TOOL data = 50-0 Tool data for finishing

(21) MOP Control 1

VMPC1

Expression: Tool offset number

Allowable value: 1 to Number of tool data sets

It is possible to designate sampling of no-load data (for collective method data).

Sampling of no-load data is valid only when VMCP2 = 1 and the MOP-TOOL operation mode is "monitoring".

- Bit 7
1: No-load data sampling ON
0: No-load data sampling OFF
- Bit 0 to bit 3
Designate the load data number for which no-load data sampling is to be executed.

- Example:
 - VMPC1 = #80H.....Sampling of data under no-load condition for load data No. 1
 - VMPC1 = #81H.....Sampling of data under no-load condition for load data No. 2
 - VMPC1 = #82H.....Sampling of data under no-load condition for load data No. 3
 - :
 - VMPC1 = #8FH.....Sampling of data under no-load condition for load data No. 16
 - VMPC1 = #00H.....No-load data sampling OFF

(22) MOP Control 2

VMPC2

Designates the ON/OFF status of MOP-TOOL.

0: MOP OFF

1: MOP ON

(23) MOP Control 3

VMPC3

The overload monitoring function can be turned ON/OFF.

- Bit 7
 - 1: Load data No. 1 overload monitoring ON
 - 0: Load data No. 1 overload monitoring OFF
- Bit 6
 - 1: Load data No. 2 overload monitoring ON
 - 0: Load data No. 2 overload monitoring OFF
- Bit 5
 - 1: Load data No. 3 overload monitoring ON
 - 0: Load data No. 3 overload monitoring OFF
- Bit 4
 - 1: Load data No. 4 overload monitoring ON
 - 0: Load data No. 4 overload monitoring OFF
- Bit 3
 - 1: Load data No. 5 overload monitoring ON
 - 0: Load data No. 5 overload monitoring OFF
- Bit 2
 - 1: Load data No. 6 overload monitoring ON
 - 0: Load data No. 6 overload monitoring OFF
- Bit 1
 - 1: Load data No. 7 overload monitoring ON
 - 0: Load data No. 7 overload monitoring OFF
- Bit 0
 - 1: Load data No. 8 overload monitoring ON
 - 0: Load data No. 8 overload monitoring OFF

(24) MOP Control 4

VMPC4

The air-cut reduction function can be turned ON/OFF.

- Bit 7
 - 1: Air cut reduction for load data No. 1 ON
 - 0: Air cut reduction for load data No. 1 OFF

- Bit 6
1: Air cut reduction for load data No. 2 ON
0: Air cut reduction for load data No. 2 OFF
- Bit 5
1: Air cut reduction for load data No. 3 ON
0: Air cut reduction for load data No. 3 OFF
- Bit 4
1: Air cut reduction for load data No. 4 ON
0: Air cut reduction for load data No. 4 OFF
- Bit 3
1: Air cut reduction for load data No. 5 ON
0: Air cut reduction for load data No. 5 OFF
- Bit 2
1: Air cut reduction for load data No. 6 ON
0: Air cut reduction for load data No. 6 OFF
- Bit 1
1: Air cut reduction for load data No. 7 ON
0: Air cut reduction for load data No. 7 OFF
- Bit 0
1: Air cut reduction for load data No. 8 ON
0: Air cut reduction for load data No. 8 OFF

(25) MOP Control 5

VMPC5

Used to turn the adaptive control function ON/OFF.

- Bit 7
1: Adaptive control for load data No. 1 ON
0: Adaptive control for load data No. 1 OFF
- Bit 6
1: Adaptive control for load data No. 2 ON
0: Adaptive control for load data No. 2 OFF
- Bit 5
1: Adaptive control for load data No. 3 ON
0: Adaptive control for load data No. 3 OFF
- Bit 4
1: Adaptive control for load data No. 4 ON
0: Adaptive control for load data No. 4 OFF
- Bit 3
1: Adaptive control for load data No. 5 ON
0: Adaptive control for load data No. 5 OFF
- Bit 2
1: Adaptive control for load data No. 6 ON
0: Adaptive control for load data No. 6 OFF

- Bit 1
 - 1: Adaptive control for load data No. 7 ON
 - 0: Adaptive control for load data No. 7 OFF
- Bit 0
 - 1: Adaptive control for load data No. 8 ON
 - 0: Adaptive control for load data No. 8 OFF

1-4-2. Read/Write System Variables Requiring Special Attention in Writing

CAUTION

The system variables ((1) to (9)) explained below have critical influence over machine operations. After they have been written, the machined surface will deteriorate, or unexpected interference will result, if the written values are re-set to the original values. Therefore, do not attempt to write them unless absolutely necessary.

(1) Positive Travel End Limit Value

VPSL*

*: Axis name X to Z, U to W, A to C

The travel limit in the positive direction for the axis indicated by the axis name can be read/written. This sets the data for user parameter "P PROG LIMIT MC" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM). The value to be set is determined in the machine coordinate system.

[Supplement]

The travel end limit (+) is set as a value in the machine coordinate system, but if this system variable is re-written, this value, converted to the value in the work coordinate system, is simultaneously set as the programmable travel end limit (+).

- Example 1:
 - Reading X-axis travel end limit (+)
 - Assume that:
 - Travel end limit (+) is 500 mm (19.69 in.) in the work coordinate system, and
 - Work zero of the currently selected work coordinate system No. 2 is X = 150 mm (7.09 in.)
- VC1 = VPSLX..... Reading travel end limit (+) in the machine coordinate system
- VC2 = VACOD..... Reading current coordinate system number
- VC3 = VC1 – VZOFX [VC2]..... Conversion of read travel end limit (+) into the value in the work coordinate system
- μm unit system
 - VC1 = 650000 VC2 = 2 VC3 = 500000
 - mm unit system
 - VC1 = 650 VC2 = 2 VC3 = 500

(2) Negative Travel End Limit Value

VNSL*

*: Axis name X to Z, U to W, A to C

The travel limit in the negative direction for the axis indicated by the axis name can be read/written. This sets the data for user parameter "N PROG LIMIT MC" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM). The value to be set is determined in the machine coordinate system.

[Supplement]

The travel end limit (–) is set as a value in the machine coordinate system, but if this system variable is re-written, this value, converted to the value in the work coordinate system, is simultaneously set as the programmable travel end limit (–).

- Example 1:

Reading X-axis travel end limit (–)

Assume that:

Travel end limit (–) is –500 mm (19.69 in.) in the work coordinate system, and

Work zero of the currently selected work coordinate system No. 2 is X = 150 mm (7.09 in.)

VC1 = VNSLX..... Reading travel end limit (–) in the machine coordinate system

VC2 = VACOD..... Reading current coordinate system number

VC3 = VC1 – VZOFX [VC2]..... Conversion of read travel end limit (–) into the value in the work coordinate system

- μm unit system

VC1 = –650000 VC2 = 2 VC3 = –500000

- mm unit system

VC1 = –650 VC2 = 2 VC3 = –500

(3) Backlash Compensation Value

VBLC*

*: Axis name X to Z, U to W, A to C

The backlash compensation value of the axis indicated by the axis name can be read and written. This sets the data for user parameter "BACKLASH" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM).

- Example 1:

Reading X-axis backlash compensation value

Assume that backlash compensation value of X-axis is 0.05 mm (0.002 in.).

VC1 = VBLCX

- μm unit system

VC1 = 50

- mm unit system

VC1 = 0.05

(4) In-position Width

VINP*

*: Axis name X to Z, U to W, A to C

The in-position width value of the axis indicated by the axis name can be read and written. This sets the data for system parameter "IN POSITION" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM).

- Example:
Reading X-axis in-position width
Assume that the in-position width of the X-axis is 0.003 mm.
VC1 = VINPX

- μm unit system
VC1 = 3

- mm unit system
VC1 = 0.003

(5) In-position Width for Home Position
VHPI*

*: Axis name X to Z, U to W, A to C

The in-position width for home position of the axis indicated by the axis name can be read and written. This sets the data for system parameter "IN POSITION (H)" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM).

- Example:
Reading the in-position width for home position of X-axis
Assume that in-position width for X-axis home position is 0.020 mm.
VC1 = VHPIX

- μm unit system
VC1 = 20

- mm unit system
VC1 = 0.02

(6) Origin of Machine Coordinate System
VMOF*

*: Axis name X to Z, U to W, A to C

The zero point of the machine coordinate system on the axis indicated by the axis name can be read and written. This sets the data for system parameter "ZERO OFFSET (MACHINE)" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM).

- Example:
Reading X-axis origin of machine coordinate system
Assume that the X-axis zero point of the machine coordinate system is 2675.632 mm.
VC1 = VMOFX

- μm unit system
VC1 = 2675632

- mm unit system
VC1 = 2675.632

(7) Home Position Location

VHPP* [expression]

*: Axis name X to Z, U to W, A to C

Expression: Home position number

Allowable value: 1 – 32

The home position location can be read and written by indicating the home position number with the expression and also by indicating the axis name. The home position is referenced to the machine zero position. This sets the data for system parameter "HOME POSITION 1 – 32" that is accessible in the parameter setting mode. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM).

- Example:

Reading No. 3 home position location of X-axis

Assume that No. 3 home position location (X-axis) is at 457.987 mm.

VC1 = VHPPX[3]

- μm unit system
VC1 = 457987
- mm unit system
VC1 = 457.987

(8) Sensor Contact Value

VSAP*

*: Axis name X to Z, U to W, A to C

The contact point of the sensor or the touch probe after the execution of the maker subprogram (MSB) for the automatic gauging cycle can be read and written. The contact position coordinate value of the axis designated by the axis name is read or written. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM). The value is referenced from the zero point of the position encoder.

- Example:

Reading sensor contact point value on the X-axis

Assume that:

- Sensor contact point value from the zero position of the position encoder: 3500 mm
- Origin of machine coordinate system: 2000 mm
- Work coordinate system currently selected: No. 2
- Zero point (X) in the work coordinate system: 700 mm

VC1 = VSAPX..... Reading sensor contact point, referenced from the zero point of the position encoder

VC2 = VC1 – VMOFX..... Read value is converted into the value in the machine coordinate system

VC3 = VACOD..... Reading the number of the present work coordinate system

VC4 = VC2 – VZOFX[VC3]..... Value converted into the value in the work coordinate system

- μm unit system
VC1 = 3500000 VC2 = 1500000 VC3 = 2 VC4 = 800000
- mm unit system
VC1 = 3500 VC2 = 1500 VC3 = 2 VC4 = 800

(9) Active Tool Number

VTLCN

The tool number of the tool presently set in the spindle can be read and written.

- Example:
Reading the active tool number
VC1 = VTLCN

[Supplement]

Read only for the machine with ATC.

(10) Next Tool Number

VTLNN

The tool number of the next tool can be read and written.

- Example:
Reading the next tool number
VC1 = VTLNN

[Supplement]

Read only for the machine with ATC.

1-4-3. Read Only System Variables

(1) Calculated Value

VRCO*

*: Axis name X to Z, U to W, A to C

The calculated value (CON) of the axis designated by the axis name can be read. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM). The value is referenced from the zero point of the position encoder.

- Example:
Reading the calculated X-axis position
Assume that:
 - Calculated value: 3750 mm
 - Zero in the machine coordinate system: 2500 mm
 - Work coordinate system: No. 2
 - Work zero (X): 800 mm

VC1 = VRCOX.....	Reading the calculated value read (referenced to the zero point of position encoder)
VC2 = VC1 – VMOFX.....	Read value is converted into the value in the machine coordinate system
VC3 = VACOD.....	Reading the number of the present work coordinate system
VC4 = VC2 – VZOFX[VC3]	The value is converted into the value in the work coordinate system

- μm unit system
VC1 = 3750000 VC2 = 1250000 VC3 = 2 VC4 = 450000

- mm unit system
VC1 = 3750 VC2 = 1250 VC3 = 2 VC4 = 450

(2) Actual Position Data

VAPA*

*: Axis name X to Z, U to W, A to C

The actual value (APA) of the axis designated by the axis name can be read. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM). The value is referenced from the zero point of the position encoder.

- Example:
Reading the actual X-axis position
Assume that:
 - Actual value: 3750 mm
 - Zero in the machine coordinate system: 2500 mm
 - Work coordinate system: No. 2
 - Work zero (X): 800 mm

VC1 = VAPAX.....	Reading the actual value of the machine (referenced to the zero position of the position encoder)
VC2 = VC1 – VMOFX.....	Read value is converted into the value in the machine coordinate system
VC3 = VACOD.....	Reading the number of the present work coordinate system
VC4 = VC2 – VZOFX [VC3].....	The value is converted into the value in the work coordinate system

- μm unit system
VC1 = 3750000 VC2 = 1250000 VC3 = 2 VC4 = 450000
- mm unit system
VC1 = 3750 VC2 = 1250 VC3 = 2 VC4 = 450

(3) Active Work Coordinate System Number

VACOD

The work coordinate system number of the work coordinate system presently selected can be read.

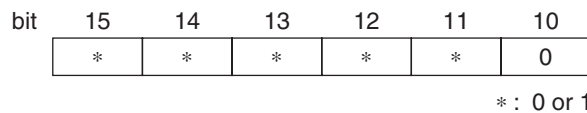
- Example:
Present work coordinate system number 2
VC1 = VRCOX
VC2 = VC1 – VMOFX
VC3 = VACOD
VC4 = VC2 – VZOFX [VC3]
VC3 = 2.

(4) Active Tool Number

VATOL

The tool management number (tool kind + tool number) of the tool presently set in the spindle can be read. The data is two-byte data; the upper six bits show the tool kind and the lower ten bits represent the tool number.

Tool kind (Some tool kinds cannot be set depending on the machine specifications.)



bit 15

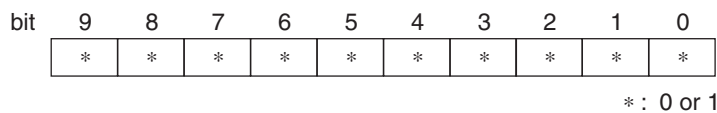
- 0: Normal tool
- 1: Large-diameter tool (L)

bit 14 to bit 11

- 0: Normal tool
- 1: Heavy tool (M)
- 2: Planer tool (P)
- 3: Planer heavy tool (PM)
- 4: Attachment tool (A)
- 5: Attachment heavy tool (AM)
- 6: U-axis tool (U)

ME33018R1001300100001

Tool number



ME33018R1001300100002

- Example 1:
Reading the management number of the active tool (Normal tool with the tool number 10)
VC1 = VATOL
VC1 = 10 (#0000000000001010)
- Example 2:
Reading the management number of the active tool (Large-diameter tool with the tool number 10)
VC1 = VATOL
VC1 = 32778 (#1000000000001010)
- Example 3:
Reading the active tool number (Heavy tool with the tool number 1)
VC1 = VATOL
VC2 = VATOL AND #03FFH
VC1 = 2049 (#0000100000000001)
VC2 = 1 (#0000000000000001)

(5) Next Tool Number

VNTOL

The tool management number (tool kind + tool number) of the tool to be used next can be read. The data is two-byte data; the upper six bits show the tool kind and the lower ten bits represent the tool number.

Tool kind (Some tool kinds cannot be set depending on the machine specifications.)

bit	15	14	13	12	11	10
	*	*	*	*	*	0

* : 0 or 1

bit 15

0: Normal tool

1: Large-diameter tool (L)

bit 14 to bit 11

0: Normal tool

1: Heavy tool (M)

2: Planer tool (P)

3: Planer heavy tool (PM)

4: Attachment tool (A)

5: Attachment heavy tool (AM)

6: U-axis tool (U)

ME33018R1001300100003

Tool number

bit	9	8	7	6	5	4	3	2	1	0
	*	*	*	*	*	*	*	*	*	*

* : 0 or 1

ME33018R1001300100004

- Example 1:
Reading the management number of the next tool (Normal tool with the tool number 10)
VC1 = VNTOL
VC1 = 10
- Example 2:
Reading the management number of the next tool (Large-diameter tool with the tool number 10)
VC1 = VNTOL
VC1 = 32778
- Example 3:
Reading only the next tool number (Heavy tool with the tool number 1)
VC1 = VNTOL
VC2 = VNTOL AND #03FFH
VC1 = 2049
VC2 = 1

- (6) Number of Coordinate Systems and Tool Data Sets (NC specification code No. 2)

VSPCO

The 1-byte specification code data which indicates the number of coordinate systems and tool data sets can be read. The relationship between the bit data and the specifications is indicated in the following table.

Bit No.	Specifications
Bit 7	100 sets of tool data
Bit 6	300 sets of tool data
Bit 5	200 sets of tool data
Bit 4	50 sets of coordinate systems
Bit 3	20 sets of coordinate systems
Bit 2	
Bit 1	
Bit 0	Large-volume program storage function

- (7) Specification Code for Maker Subprogram (NC Specification Code No. 24)

VSPSB

The 1-byte specification code data which indicates the maker subprogram specifications can be read. The relationship between the bit data and the specifications is indicated in the following table.

Bit No.	Specifications
Bit 7	Maker subprogram creating subprogram, standard tool 150 mm
Bit 6	FM-type touch probe subprogram
Bit 5	Automatic zero offset subprogram
Bit 4	Dimension check subprogram
Bit 3	Optical touch probe subprogram
Bit 2	Automatic tool breakage detection subprogram
Bit 1	Automatic cutter radius compensation subprogram
Bit 0	Automatic tool length offset subprogram

- (8) Machine Lock

VMLOK

It is possible to read whether or not the NC is presently in the machine lock status.

- Example 1:
When the NC is in the machine lock status
VC1 = VMLOK
VC1 = 128.
- Example 2:
When the NC is not in the machine lock status
VC1 = VMLOK
VC1 = 0.

- (9) Feedrate Clamp Value

VFMDX

The feedrate clamp value (the data set for NC optional parameter (long word) No. 10) can be read.

- Example:
Reading the feedrate clamp value
VC1 = VFMDX

(10) Program Unit System

VINCH

The unit system (set for NC optional parameter (INPUT UNIT SYSTEM), or NC optional parameter (bit) No. 3, bit 0 to bit 7 and No. 4, bit 0) used for the program which is being executed can be read.

- Example:

If the setting unit for "LENGTH UNIT SYSTEM" of NC optional parameter (INPUT UNIT SYSTEM) is "inch" and the length is "1", set as follows.

VC1=VINCH

VC1=3

The following shows the setting items on the NC Optional Parameter (INPUT UNIT SYSTEM) screen and the setting items of NC optional parameter (bit).

- NC Optional Parameter (INPUT UNIT SYSTEM) screen

PARAMETER OSP-P200. MIN A-MTD

2005/02/07 13:25:40

NC OPTIONAL PARAMETER

INPUT UNIT SYSTEM

	PART PROGRAM UNIT	DATA SET UNIT	Co	1
LENGTH UNIT SYSTEM	mm inch	mm inch	X	52.123
REAL NUMBER	YES NO	YES NO	Y	19.447
LENGTH	1 (mm)	1 (mm)	Z	10.000
ANGLE	1 (deg)	1 (deg)	(DISTANCE)	
FEDRATE	1 (mm/min)		X	0.000
FEDRATE	1 (mm/rev)		Y	0.000
DWELL TIME	1 (sec)		Z	0.000

ROUND, FIX, FUP ARE CONVERT INTEGER DECIMAL POINT

NC OPTIONAL PARAMETER BIT NO. 3 BIT0

PRE (NO.) 4

MENU [*] ITEM ↑ ITEM ↓ DISPLAY CHANGE

- NC optional parameter (bit) No. 3, bit 0 to bit 7 and No. 4, bit 0

Parameter No.	Bit No.	Contents	Check Mark	Without Check Mark
3	0	Unit for distance is either "mm" or "inches".(*2)	inch	mm
	1	Units are "1 mm", "1 inch", "1 degree", or "1 sec."	Same as indicated in the left	Conforms to the settings of bits 2 to 5 and bit 7 of No. 3 and bit 0 of No. 4.
	2	Unit for distance is either "0.01 mm", or "0.001 mm".	0.01 mm	0.001 mm
	3	Unit for feedrate is either "0.1 mm/min, 0.01 in/min", or "1 mm/min, 0.1 in/min".	0.1 mm/min, 0.01 in/min	1 mm/min, 0.1 in/min
	4	Unit for feedrate is either "0.001 mm/rev, 0.0001 in/rev", or "0.01 mm/rev, 0.001 in/rev".	0.001 mm/rev, 0.0001 in/rev	0.01 mm/rev, 0.001 in/rev
	5	Unit for time is either 0.01 sec or 0.1 sec.	0.01 sec	0.1 sec
	6	For decimal point data, the decimal point position indicates 1 mm, 1 inch, 1 deg or 1 sec.	Same as indicated to the left	Conform to the setting of bits 1 to 5 and bit 7.
4	7	Unit for time is either 0.001 sec or 0.1 sec. (*1)	0.001 sec	0.1 sec
	0	Unit for angle is either 0.001 degree or 0.0001 degree.	0.0001 degree	0.001 degree

(*1) If the setting for bit 5 is "1", unit of time is always 0.01 sec.

(*2) The setting for bit 0 is valid only when inch / mm switchable specification is selected.

* Unit system read and write operations area enabled by setting the data of the NC optional parameter (bit) No. 34, bit 4.

(11) Sequence Restart Flag

VRSTT

The flag that is turned on when the restart search command (RS) is executed in the automatic mode and turned off after the designated sequence is located can be read.

Setting range: Binary 8 bits (1 byte)

- Example 1:
During sequence restart
VC1 =VRSTT
VC1 = 128
- Example 2:
Not in sequence restart
VC1 =VRSTT
VC1 = 0

(12) Operating Time Counter

VDTIM[α , β]

α = 1: Power ON time 2: NC operating time 3: Spindle rotating time
 4: Cutting time 5: External input time

β = 1: Count value 2: Set value

ME33018R1001300100006

The time counted by counters and their set values are read.

- Example:
The sequence jumps to N010 when the cutting time reaches 10 hours.

```
N001 IF [VDTIM [4, 1] EQ 10] N010
N002
:
:
N010
```

ME33018R1001300100007

(13) Work Counter

VWRKC[α , β]

α = 1: Work counter A 2: Work counter B
 3: Work counter C 4: Work counter D

β = 1: Count value 2: Set value

ME33018R1001300100008

The values counted by counters, and their set values, are read.

- Example:
The sequence jumps to N010 when the count value at work counter A reaches 5.

```
N001 IF [VWRKC [1, 1] EQ 5] N010
N002
:
:
N010
```

ME33018R1001300100009

(14) G Code

VGCOD[expression]

Expression: Group number of the G code

Allowable range: 1 – 96

The mode of the present G code groups can be read.

The value to be read is the numerical value of a G code. However, “254” is read for G00.

Group	G Code	Group	G Code
1	G0 G1 G2 G3 G60	12	G90 G91
2	G4	13	G94 G95
3	G10 G11	14	G61 G64
4	G15 G16	15	G20 G21
5	G17 G18 G19	16	G30
6	G22 G23	17	G31
7	G40 G41 G42	18	G09
8	G43 G44	19	G62
9	G50 G51	20	G92
10	G53 G54 G55 G56 G57 G58 G59	21	G71
		32	G101-G120
11	G73 G74 G76 G80 G81 G82 G83 G84 G85 G86 G87 G89 G274 G284	66	G186 G187
		78	G174 G175

- Example 1:
In the G00 mode
VC1 = VGCOD[1]
VC1 = 254.
- Example 2:
In the G1 mode in incremental mode
VC1 = VGCOD[1]
VC2 = VGCOD[12]
VC1 = 1, VC2 = 91.

(15) S Code

VSCOD

The command value of the present spindle speed command (S) can be read as programmed.

The setting of the spindle speed override dial is ignored and the programmed value is read as programmed.

- Example:
S1200
VC1 = VSCOD
VC1 = 1200

(16) F Code

VFCOD

The command value of the present feedrate (F) can be read.

Setting of the feedrate override dial is ignored and the unit of the data to be read is 0.1 mm/min (for feed per minute mode) or 1 μ m/rev (for feed per revolution mode).

Note that conversion to the English system (inches) is not carried out.

- Example 1:
Reading the feedrate in units of mm/min to variable VFCOD/10 when the programming unit system "mm" and G94 mode is active.
 $VC1 = VFCOD/10$
- Example 2:
Reading the feedrate in units of inch/rev to variable VC1 when the programming unit system "inch" and G95 mode is active.
 $VC1 = VFCOD/[1000 \times 25.4]$
Note that VFCOD reads only 4-digit F commands and will not be changed by F1-digit commands.

(17) D Code

VDCOD

The present cutter radius compensation number can be read.

- Example:
Assume that the present cutter radius compensation number is "5".
 $VC1 = VDCOD$
 $VC1 = 5.$

(18) H Code

VHCOD

The present tool length offset number can be read.

- Example:
Assume that the present tool length offset number is "5".
 $VC1 = VHCOD$
 $VC1 = 5.$

(19) Mirror Image

VMRI

The present status of the mirror image function can be read. The data is of one-byte length and each axis corresponds to each bit.

Mirror image ON: 1

Mirror image OFF: 0

bit	7	6	5	4	3	2	1	0
	*	*	*	*	*	*	*	*
			6th-axis	5th-axis	4th-axis	Z-axis	Y-axis	X-axis

* : 0 or 1

ME33018R1001300100010

- Example:
Procedure for checking whether or not the mirror image function is active for the X and Z-axes.
 $VC1 = 1 + 4 \text{ X-axis mask (2) + Z-axis mask (2)}$
 $VC2 = VMRI \text{ AND } VC1$
 $\text{IF } [VC2 \text{ EQ } 0] \text{ N1}$

(20) Internal Clock

VPMNT: A “minute counter” which is cleared hourly can be read.

Allowable range: 0 – 59 (unit: minute).

VRMNT: A 4-byte free-running minute counter can be read.

VPHUR: An “hour counter” which is cleared daily can be read.

Allowable range: 0 – 23 (unit: hour)

VRHUR: A 4-byte free-running hour counter can be read.

VQDAT: A 2-byte free-running date counter can be read.

The free-running counter is cleared by turning on/off the power supply.

- Example:

To turn on the 60 minute timer

The example below assumes that the duration after turning on the power supply is much smaller than two minutes.

VC1 = VRMNT

NA1 VC2 = VRMNT - VC1

IF [VC2 GE 60] NA2

GOTO NA1

(21) Manual Intervention Shift Amount

VMSF*

*: Axis name X to Z, U to W, A to C

The manual intervention shift amount of the axis indicated by the axis name can be read. For this operation, the unit system is as set at NC optional parameter (INPUT UNIT SYSTEM).

- Example:

Reading manual intervention shift amount

Assume that manual intervention shift amount (X) is 150 mm (5.91 in.).

VC1 = VMSFX

- μm unit

VC1 = 150000

- mm unit

VC1 = 150

(22) Torque Monitor During Synchronized Tapping

VTMDT[expression]

Expression: Synchronized tapping torque monitor parameter No.

Setting range: 1 to 5

The set value for the synchronized tapping torque monitor parameter No. designated by the expression can be read.

- Example:

Reading the value of synchronized tapping torque monitor parameter No. 3

(Assume that the set value for synchronized tapping torque monitor parameter No. 3 is 100 kgf·m)

VC1 = VTMDT[3]

VC1 = 100.

(23) Spindle Overload Monitor Parameter

VSLDT[a,b]

a: Spindle overload monitor parameter No.

Setting range: 1 to 5

b: Spindle overload monitor parameter type

Setting range: 1 to 3

1...Maximum load value (%) for spindle overload monitor parameter

2...Continuously overloaded time (sec.) for spindle overload parameter

3...Monitor mode (alarm level) for spindle overload parameter

It is possible to read the set value for the spindle overload monitor (simple load monitor) parameter number designated by "a".

- Example:

Reading the value for the continuously overloaded time of spindle overload parameter No. 3

(Assume that the continuously overloaded time of spindle overload parameter No. 3 is 10 sec.)

VC1 = VSLDT [3, 2]

VC1 is 10 sec.

(24) Number of Tools for ATC

VSPTN

The number of tool pots in the ATC magazine can be read.

- Example:

Reading the number of tool pots in the ATC

(Assume that the number of tool pots in the ATC magazine is "50".)

VC1 = VSPTN

VC1 = 50.

(25) PPC Pallet Number (for the PPC specification)

VPLNO

The pallet number of the pallet presently mounted on the machine table can be read.

When machining identical workpieces using several pallets, system variable VPLNO is used to assign different work coordinate systems for the individual pallets if the zero offset values differ slightly among the pallets to be used.

- Example:

Assume that #1000 workpieces are set on the pallets No. 1 and No. 2, and that the work coordinate systems used for these pallets are as indicated below:

Work coordinate system No. 11 for pallet No. 1

Work coordinate system No. 12 for pallet No. 2

O1000

N001 IF [VPLNO EQ 1] N010.....N010 To N010 for pallet No. 1

IF [VPLNO EQ 2] N020 N020 To N020 for pallet No. 2

MSG(PALLET NO NG)

GOTO NEND Displays message "PALLET NO NG",
does not execute cutting cycle for this
pallet, and processes the next pallet.

N010 G15 H11 GOTO N100 Selects H11 coordinate system for pallet No. 1

N020 G15 H12 GOTO N100 Selects H12 coordinate system for pallet No. 2

N100 T1 Part program for workpiece #1000

:

:

NEND M02

(26) PPC Parameters (for the PPC specification)

VPPCP PPC parameter (word) data
 VPLDT[expression] PPC parameter (bit) data

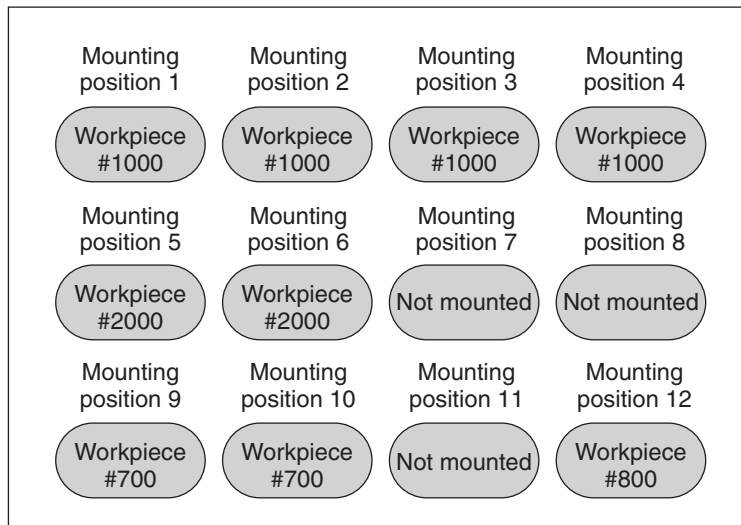
When multiple number of workpieces is set on a pallet with PPC set ON, this system variable is used to indicate the positions where the workpieces are set. The parameter must be set from the PPC panel beforehand.

VPPCP is used to read the parameters comprehensively.

VPLDT is used to read the data of the position indicated by the designated bit.

Up to 12 positions (12 bits) can be used to designate the workpiece mounting position (PPC parameter) on a single pallet.

- Example:



ME33018R1001300100012

Assume that the workpieces are mounted on a pallet as shown above.

System variables VPLDT[1] through VPLDT[12] and VPPCP are set as below when individual programs are executed. These settings can be referenced from part programs.

	O1000 executed	O2000 executed	O0700 executed	O0800 executed
VPLDT[1]	1	0	0	0
VPLDT[2]	1	0	0	0
VPLDT[3]	1	0	0	0
VPLDT[4]	1	0	0	0
VPLDT[5]	0	1	0	0
VPLDT[6]	0	1	0	0
VPLDT[7]	0	0	0	0
VPLDT[8]	0	0	0	0
VPLDT[9]	0	0	1	0
VPLDT[10]	0	0	1	0
VPLDT[11]	0	0	0	0
VPLDT[12]	0	0	0	1
VPPCP	# 00FH	# 030H	# 300H	# 800H

Example program

```

N001 IF [VPLDT[1] EQ 0] N002
      G15 H1
      CALL OSUB
N001 IF [VPLDT[2] EQ 0] N003
      G15 H2

```

ME33018R1001300100013

(27) Automatic Crossrail Positioning M Code

VECWM

The M code number corresponding to the present crossrail position where the crossrail was positioned using an automatic crossrail positioning M code can be read.

5 crossrail positioning level specification 81 – 85

10 crossrail positioning level specification 190 – 199

- Example:

To check the crossrail position using the commands at the head of a program

```

O100
IF [VECWM NE 85] NALM ..... Checking the crossrail position
  :                               } Machining with the crossrail
  :                               } positioned at M85 level
  :
  :
GOTO NEND
NALM VDOOUT[992] = 10 ..... Alarm processing
                               (If the level is other than M85)

NEND M02

```

ME33018R1001300100014

(28) Active Attachment Number

VATNT

The number of the presently mounted attachment can be read.

The read number is replaced with the number of attachments (≥ 1).

- Example:

When the active attachment number is 301

VC1 = VATNT

VC1 = 1.

- (29) The value of Year/Month/Day/Hour/Minute/Second
VTIME [Formula]

Formula	Data	Value range
1	Year	1980 -
2	Month	1 - 12
3	Day	1 - 31
4	Hour	0 - 23
5	Minute	0 - 59
6	Second	0 - 59

When VTIME [1] is executed, the value of "Year" is read and the values of "Month/Day/Hour/Minute/Second" are stored. When VTIME [formula] (formula = 2 to 6) is executed, the stored values can be read.

If VTIME [formula] (formula = 2 to 6) is executed instead of executing VTIME [1], the values stored in past will be read.

- Example: If VC1 = VTIME [1] is executed for a case "2003 (year)/4 (month)/9 (day)/10 (hour)/18 (minute)/54 (second)", the value is returned as VC1 =2003.
VC2 = VTIME [2]
VC3 = VTIME [3]
VC4 = VTIME [4]
VC5 = VTIME [5]
VC6 = VTIME [6]
Each value becomes as follows: VC2 = 4, VC3 = 9, VC4 = 10, VC5 = 18 and VC6 = 54.

- (30) Current Position of Work Coordinate System

There are two types of system parameter for work coordinate system: VWKA* and VWKB*.

VWKA*

It can read current position of work coordinate system with offset amount of current tool length, attachment rotation, and rotary spindle head. The mark * refers to an axis name.

VWKB*

It can read current position of work coordinate system without offset amount of current tool length, attachment rotation, and rotary spindle head. The mark * refers to an axis name.

- Example:
VC1=VWKAX
VC2=VWKBY

This is independent from the local coordinate systems used in the functions including coordinate conversion function (Parallel/rotary shift of coordinate system (G11/G10); copy (COPY; COPYE). This also is independent from the converted coordinate value during slope machining (not the coordinate value on the slope but the actual work coordinate position selected). Manually shifted amount is out of the scope. The value of system variable becomes zero (0) when the position of work coordinate system of the removed axis is read in.

(31) Tool Number (Pot Designation)

VPTNO [expression]

Expression: Pot number

Setting range: 1 to max number of pots

Tool number stored in the pot specified in the expression can be read.

- Example 1: Reading tool number in the pot number 3 (assuming tool number in the pot number 3 is 10)
VC1= VPTNO [3]
VC1= 10
- Example 2: Reading tool number in the pot number 5 (assuming no tool is in the pot number 5)
VC1= VPTNO [5]
VC1= 0

(32) Pot Number (Tool Designation)

VTPNO [expression]

Expression: Tool number

Tolerance: 1 to max number of tool

Pot number storing the tool number specified in the expression can be read.

- Example 1: Reading pot number storing the tool number 10 (assuming pot number storing the tool number 10 is 3)
VC1= VTPNO [10]
VC1= 3
- Example 2: Reading pot number storing the tool number 12 (assuming tool number 12 is not in any pot)
VC1= VTPNO [12]
VC1= 0

(33) M codes

VMCD9

The M code commanded in the preceding block can be read.

- When the preceding M code command is "M230," the value of "VMCD9" is "230."

(34) Sequence numbers

VSEQN

The sequence number in the preceding block can be read as a value.

Since a sequence number is read as a value, both N1 and N0001 will be read as 1.

Any sequence name (consisting of alphanumerical characters) will be read as 0.

- When the preceding sequence number is "N123," the value of "VSEQN" is "123."
- When the preceding sequence number is "N0123," the value of "VSEQN" is 123.

(35) Program numbers

VPRGN

Any current program number can be read as a value.

Since a program number is read as a value, both O1 and O0001 will be read as 1.

Any program name (consisting of alphanumerical characters) will be read as -1.

- When the current program number is "O123," the value of "VPRGN" is "123."
- When the current program number is "O0123," the value of "VPRGN" is 123.

- (36) Number of days
VQDAT
VQDAT is to read days on the assumption that January 1, 1980 is one (1).
- (37) Number of seconds
VRTIM
VRTIM is to read seconds on the assumption that "0:00:00 am" is zero (0).
- (38) Current position of machine coordinate system
VMCA*
Values for "Encoder coordinate system command (VRCO*) - Machine coordinate system zero point (VMOF*) - Virtual origin the multi-turn axis" can be read.
Input an axis name in the position of asterisk (*).
- (39) Current position including shift amount.
VMSC*
Input an axis name in the position of asterisk (*).
- (40) Skip value of slope coordinate system
VSCC*
The skip value of the slope coordinate system can be read.
The value will be a skip value on the workpiece coordinate system not during conversion into three-dimensional coordinates.
This variable is applicable only to the X-, Y-, and Z-axes, and the value for additional axes is always "0."
Input an axis name in the position of asterisk (*).
- (41) Block end position
VACP*
The current position of the specified coordinate system currently selected can be read.
This value corresponds to mirror images and slope.
Input an axis name in the position of asterisk (*).
- (42) After next tool number
VNNTL
The after next tool number can be read.

1-4-4. General Rule for Automatic Conversion between Inches and Millimeters

Automatic conversion follows the settings at the NC optional parameter (INPUT UNIT SYSTEM).

- (1) NC Optional Parameter (INPUT UNIT SYSTEM), part program unit system "LENGTH UNIT SYSTEM"

- Example:

VC1 = VTOFH[1] ("LENGTH UNIT SYSTEM" = "inch")

Tool length offset value set at tool offset number 1 is set at VC1 in inches.

- (2) NC Optional Parameter (INPUT UNIT SYSTEM), part program unit system "LENGTH"

- If the setting at LENGTH UNIT SYSTEM is "mm" and that at LENGTH is "1 (mm)", the unit of length is "mm".

- Example 1:

VC1 = VTOFH[1] (LENGTH = "1" (mm))

VC1 = 2.5 when VTOFH[1] is 2.5 mm (0.098 in.).

- Example 2:

VTOFH[1] = VC1 (LENGTH = "1" (mm))

VTOFH[1] = 5.5 mm (0.22 in.) when VC1 is 5.5.

- If the setting at LENGTH is "0.01 (mm)", the unit of length is "0.01 mm (1/100 mm)".

- Example 1:

VTOFH[1] = 100 (LENGTH = "0.01"(mm))

Then, VTOFH[1] = 1.0 mm

- Example 2:

VC1 = VTOFH[1] (LENGTH = "0.01"(mm))

VC1 = 520 when VTOFH[1] is 5.2 mm (0.20 in.)

- (3) NC Optional Parameter (INPUT UNIT SYSTEM), part program unit system "REAL NUMBER"

If YES is set at REAL NUMBER, system variable values are always treated in millimeter or inch disregarding of whether or not a decimal point is used.

If NO is set at REAL NUMBER, values are automatically converted according to the setting at LENGTH.

- Example 1:

VTOFH[1] = 4 (REAL NUMBER = YES)

Then, VTOFH[1] = 4 mm (0.16 in.)

- Example 2:

VTOFH[1] = 4.0 (REAL NUMBER = YES)

Then, VTOFH[1] = 4 mm (0.16 in.)

- Example 3:

VC1 = VTOFH[1] (REAL NUMBER = YES)

VC1 = 5.2 when VTOFH[1] = 5.2 mm (0.20 in.)

- (4) How numerical values are interpreted according to the setting for NC optional parameter (INPUT UNIT SYSTEM) is summarized in the table below.

Variables (local variables, common variables, system variables) in the right member of the expression are handled in the same manner as the decimal point data.

No.	Setting	Example
1.	"LENGTH UNIT SYSTEM" = mm "LENGTH" = 0.001 (mm) "REAL NUMBER" = NO	<ul style="list-style-type: none"> • Assume VTOFH[1] = 4.2 mm If VC1 = VTOFH[1], VC1 = 4200 • Assume VTOFH[1] = 5 (or 5.0) VTOFH[1] = 0.005 mm
2.	"LENGTH UNIT SYSTEM" = inch "LENGTH" = 0.0001 (inch) "REAL NUMBER" = NO	<ul style="list-style-type: none"> • Assume VTOFH[1] = 1.5 in. If VC1 = VTOFH[1] VC1 = 15000 • Assume VTOFH[1] = 1 (or 1.0) VTOFH[1] = 0.0001 in.
3.	"LENGTH UNIT SYSTEM" = mm "LENGTH" = 1 (mm) "REAL NUMBER" = NO	<ul style="list-style-type: none"> • Assume VTOFH[1] = 4.2 mm If VC1 = VTOFH[1] VC1 = 4.2 • Assume VTOFH[1] = 2 (or 2.0) VTOFH[1] = 2 mm
4.	"LENGTH UNIT SYSTEM" = inch "LENGTH" = 0.001 (inch) "REAL NUMBER" = NO	<ul style="list-style-type: none"> • Assume VTOFH[1] = 1.5 in. If VC1 = VTOFH[1] VC1 = 1.5 • Assume VTOFH[1] = 2 (or 2.0) VTOFH[1] = 2 in.
5.	"LENGTH UNIT SYSTEM" = mm "LENGTH" = 0.01 (mm) "REAL NUMBER" = NO	<ul style="list-style-type: none"> • Assume VTOFH[1] = 4.2 mm If VC1 = VTOFH[1] VC1 = 420 • Assume VTOFH[1] = 5 (or 5.0) VTOFH[1] = 0.05 mm
6.	"LENGTH UNIT SYSTEM" = mm "LENGTH" = 0.001 (mm) "REAL NUMBER" = YES	Same as 3.
7.	"LENGTH UNIT SYSTEM" = inch "LENGTH" = 0.0001 (inch) "REAL NUMBER" = YES	Same as 4.
8.	"LENGTH UNIT SYSTEM" = mm "LENGTH" = 1 (mm) "REAL NUMBER" = YES	Same as 3.
9.	"LENGTH UNIT SYSTEM" = inch "LENGTH" = 1 (inch) "REAL NUMBER" = YES	Same as 4.
10.	"LENGTH UNIT SYSTEM" = mm "LENGTH" = 0.01 (mm) "REAL NUMBER" = YES	<ul style="list-style-type: none"> • Assume VTOFH[1] = 4.2 mm If VC1 = VTOFH[1] VC1 = 4.20 • Assume VTOFH[1] = 5 (or 5.0) VTOFH[1] = 5 mm

1-4-5. Supplements

- Specifying a read only system variable at the left side will cause an alarm.
- Setting of EMPTY for system variables will cause value "0" to be set.
- System variables can be read and written even in the machine lock mode.
- Do not use system variables while in the cutter radius compensation mode.

Cutter radius compensation is executed based on the point data of the three points - actual position, programmed target point, and the next programmed target point. Therefore, the next block after the one to be executed is read in advance. The variable function is executed when the block of commands is read, i.e., before the execution of positioning to the programmed target point. However, this is not always applicable because the buffer reading may be canceled by the parameter setting.

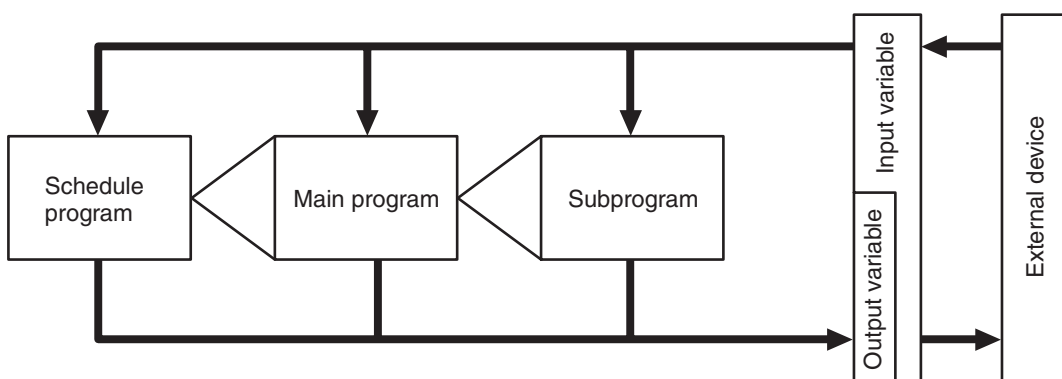
2. User Task 2

User task 2 allows the use of system variables, and logical and function operations, in addition to the functions available with user task 1. Selection of the I/O variable function is also possible.

2-1. I/O Variables

The I/O variables can reference or update an I/O signal external to the system, while the system variables are used as internal data.

- I/O variables are also determined by the system and may be referenced (input variable) or updated (output variable) in a schedule program, main program, or subprogram.
- The use of a control program made by using the I/O variables and the logical and functional operations makes it possible to control external devices.



ME33018R1001300140001

[Format]

Input variable: VDIN[expression]

Input variable number

Output variable: VDOUT[expression]

Output variable number

[Details]

- The system variables function (timer control, user alarm generation), which is effective for using the I/O variables, is also added.
- The I/O variable is referenced or updated after the previous sequence is executed.
- Updating (defined left of the equal sign "=") the input variable or referencing (defined right of the equal sign) the output variable results in an alarm.

2-1-1. Input Variables (VDIN)

Input Variable No. n	Data Contents	Input Device																
1 to 16	Reading the bit data of 16 input points 1 : ON 0 : OFF bit <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> </table> <div style="text-align: center;"> ↑ The number (n) is corresponded to "bit". ↓ </div> n = 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	I/O unit input
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
17	Reading the 1 byte data; n = 9 to 16 corresponds to bits 0 through 7	I/O unit input																
18	Reading the 1 byte data; n = 1 to 8 corresponds to bits 0 through 1	I/O unit input																
20	Reading the 1 word data; n = 1 to 16 corresponds to bits 0 through 15	I/O unit input																
1000	Free running time counter; 0.001 sec.																	
1001	Free running time counter; 1 sec.																	
1002	Free running time counter; 1 min.																	
1003	Free running time counter; 1 hour																	
1004	Free running time counter; 1 day																	

ME33018R1001300150001

*: Data at 1000 to 1004 is cleared to zero (0) when the power supply is turned on; it is not cleared by the NC reset operation.

2-1-2. Output Variable (VDOUT)

Output Variable No. n	Data Contents	Output Device																
1 to 16	<p>1 to 16: Reading the bit data of the input signals of 16 points</p> <p>1 : ON 0 : OFF</p> <p>bit <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td> </tr> </table></p> <p style="text-align: center;">↕ The number (n) is corresponded to "bit".</p> <p>n = 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1</p>	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	I/O unit output
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0			
17	Outputs the 1 byte data; n = 9 to 16 corresponds to bits 0 through 7	I/O unit output																
18	Outputs the 1 byte data; n = 1 to 8 corresponds to bits 0 through 7	I/O unit output																
20	Outputs the 1 word data; n = 1 to 16 corresponds to bits 0 through 15	I/O unit output																
990 to 993	<p>Used to generate the following alarm:</p> <p>Alarm No.</p> <p>990 : Alarm D 4209 991 : Alarm C 3220 992 : Alarm B 2395</p> <p>993 : Alarm A 1244</p> <p>Displays user alarm codes in the following format:</p> <p>VDOUT[n] = XXXX</p> <p>Here, "n" represents an output variable number for alarm level (A to D).</p> <p>"XXXX" represents user reserve code.</p> <p>Here, XXXX is a decimal number of 0 through 9999, with leading zeros suppressed.</p> <p>Example : VDOUT[992] = 2</p> <p style="padding-left: 40px;">represents 2395 Alarm B User reserve code 2</p>																	
998	<p>Cancels Alarm D4209 raised by the output variable number 990.</p> <p>Display of alarm D will be cleared.</p>																	

2-1-3. Alarm Message

User designated sub messages for user defined alarms can be displayed on the screen. Sub message designations can be set at system variable VUACM.

VUACM

[Format]

VUACM[n]

n: Subscript expression in the range from 1 to 16.

VUACM[1] = 'character-string (within 16 characters)' Describe a character-string between apostrophes.

[Details]

- Once a sub message is set at VUACM, it is displayed at the alarm display line of the screen when a user designated alarm occurs.
- VUACM is cleared by the NC reset operation. If character data is set at VUACM assigned a halfway subscript number while VUACM is cleared, no corresponding display will be available. Just after resetting the NC, VUACM [3] = 'ABC' Subscript number must begin with "1".
- The allowable subscript range is from "1" to "16" and an alarm occurs if a subscript outside this range is specified.
- Characters exceeding 16 characters are ignored.
- Data at the end: The last data set signals the end of all data.
VUACM[1] = 'ABCDEFGG'
VUACM[3] = '123'

With the setting indicated above, the display will be "AB123".

VUACM[1] = '^character-string'

The symbol "^" (circumflex) at the beginning of the character-string between apostrophes converts upper-case characters into lower-case characters.

```
VUACM [1] = '^ABCD'
           ↓
           abcd
```

ME33018R1001300170001

- To insert an apostrophe within a character-string, place two apostrophes in succession.

```
VUACM [1] = ' '123' ' AB'
           ↓
           123' AB
```

ME33018R1001300170002

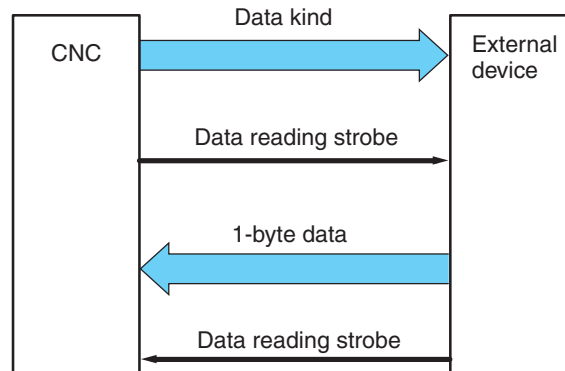
- The symbol "^" and one of the two apostrophes placed in succession are not counted as a character.

2-1-4. Supplements

- VDIN variables can be designed only in the right part of an operation command. If they are specified in the left part, an alarm occurs.
- VDOUT variables can be designated only in the left part of an operation command. If they are specified in the right part, an alarm will occur.
Therefore, if reference to a previous output condition is required, output it using another type of variable, such as a local variable or common variable, then access that variable.
Numerical values right of the decimal point are rounded.
- If a value greater than the allowable size of individual variables is used in an output variable, an alarm will occur.
Bit: 0 or 1
Byte: 0 to 255
Word: 0 to 65535
- EMPTY is regarded as zero (0).
- Input/output operations are executed with the input/output variables even during the machine lock mode.
- Input and output status is not influenced by NC reset operations. Output signals are cleared by turning the power off and then back on again.
- Output variables are output to the external interface within 25.6 msec. after the execution of the command.
This means that two output signals may be output simultaneously if they are designated consecutively.
VDOUT[1] = 1
VDOUT[2] = 1
- Do not use system variables while in the cutter radius compensation mode.
Cutter radius compensation is executed based on the point data of the three points - actual position, programmed target point, and the next programmed target point. Therefore, the next block after the one to be executed is read in advance. The variable function is executed when the block of commands is read, i.e., before the execution of positioning to the programmed target point. However, this is not always applicable because the buffer reading may be canceled by the parameter setting.

2-1-5. Application Example of Input/Output Variables

Assume that the information concerning the kind of data is output from the CNC to an external device and the corresponding one byte data is input to the CNC from the external device. This input and output process will require the procedure shown below:



ME33018R1001300190001

The procedure above can be executed by writing sub programs.

Data kind	VDOUT[17]
Data reading strobe (CNC to external device)	VDOUT[1]
One byte data from an external device	VDIN[17]
Data reading strobe (External device to CNC)	VDIN[1]

Main program	Subprogram
⋮	⋮
⋮	⋮
⋮	⋮
VC1 = 3	O100
CALL O100	VDOUT [17] = VC1
VC3 = VC2	G4 F1 Timer: 1 sec.
⋮	VDOUT [1] = 1
⋮	G4 F5 Timer: 5 sec.
⋮	IF [VDIN[1] EQ 0] NALM.....(*)
⋮	VC2 = VDIN[17]
⋮	VDOUT [1] = 0
⋮	GOTO NEND
⋮	NALM VDOUT[992] = 2
⋮	NEND RTS

(*): This block of commands generates an alarm if the data reading strobe from an external device is not turned on within 5 seconds.

ME33018R1001300190002

2-2. Math Functions

Various types of operations using variables are possible. These functions can be programmed in the same way as general calculations.

[Programming format]

Address character, Variables = Expression

The math function of user task 2 supports logical and functional operations in addition to the math functions available with user task 1.

For the math functions of user task 1, refer to "Math Functions".

2-2-1. Logical Operations

Operation	Math Name	Operation Example	VDOUT [17]
OR	Logical OR	$VDOUT[17] = VDIN[17] \ _ \text{OR} \ _ \ VDIN[18]$	01111110
AND	Logical AND	$VDOUT[17] = VDIN[17] \ _ \text{AND} \ _ \ VDIN[18]$	00001000
EOR	Exclusive OR	$VDOUT[17] = VDIN[17] \ _ \text{EOR} \ _ \ VDIN[18]$	01110110
NOT	Negation	$VDOUT[17] = \text{NOT} \ _ \ VDIN[17]$	10010101

* 1: The examples above are for input and output variables.

2: Place a space (indicated by $_$ in the table above) before and after the logical operation symbols.

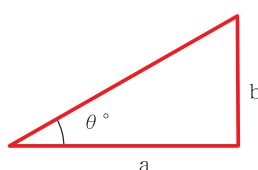
3: The values in the VDOUT column indicate those of VDOUT[17] for VDIN[17] = 01101010 and VDIN[18] = 00011100.

ME33018R1001300210001

2-2-2. Functions

Operation	Math Name	Operation Example	VC1	Remark
SIN	Sine	VC1 = SIN[30]	0.5	
COS	Cosine	VC1 = COS[VC2]	0.5	
TAN	Tangent	VC1 = TAN[45]	1	
ASIN	Arcsine	VC1 = ASIN[0.5]	30	
ACOS	Arccosine	VC1 = ACOS[0.5]	60	
ATAN	Arc tangent (1)	VC1 = ATAN[1]	45	Value range: -90° to 90° (*4)
ATAN2	Arc tangent (2)	VC2 = ATAN2[1, [-SQRT[3]]]	150	(*5)
SQRT	Square root	VC1 = SQRT[VC2 + 4]	8	
ABS	Absolute value	VC1 = ABS[20 - VC2]	40	
BIN	Decimal to binary conversion	VC1 = BIN[VDIN[17]]	(*6)	4 bytes
BCD	Binary to decimal conversion	VDOOUT[17] = BCD[VC1]	(*7)	4 bytes
ROUND	Integer implementation (rounding)	VC1 = ROUND[27.6348]	28	
FIX	Integer implementation (truncation)	VC1 = FIX[27.6348]	27	
FUP	Integer implementation (raising)	VC1 = FUP[27.6348]	28	
DROUND	Unit integer implementation (rounding)	VC1 = DROUND[13.26462]	13.265	(*8)
DFIX	Unit integer implementation (truncation)	VC1 = DFIX[13.26462]	13.264	(*8)
DFUP	Unit integer implementation (raising)	VC1 = DFUP[13.26462]	13.265	(*8)
MOD	Remainder	VC1 = MOD[VC2,7]	4	$60/7 = 8...4$

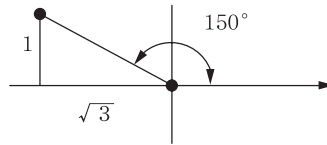
- (*1) The variables and numbers after the function operation symbols must be enclosed by '[' and ']'. These are also used for specifying the priority of operation execution order.
- (*2) When two elements are specified within the brackets, they must be separated with a comma.
- (*3) The VC1 column indicates the value obtained from the operation to its left, when VC2 equals 60.
- (*4) Arc tangent (1) (ATAN)
 $\theta = \text{ATAN}[b/a]$
 Arc tangent (2) (ATAN2)
 $\theta = \text{ATAN2}[b,a]$



- (*5) The value of $\text{ATAN2}[b,a]$ represents the angle of the point defined by the coordinate values (a,b). Its range is from -180° to 180° .

Example:

$\text{VC2} = \text{ATAN2}[1, [-\text{START}[3]]]$



- (*6) If the value of $\text{VDIN}[17]$ is "01011001" (BCD), the result of operation is $\text{VC1} = 59$.
- (*7) If the value of VC1 is 59, the result is $\text{VDOUT}[17] = 01011001$.
- (*8) In this example the setting unit is mm mm.
If inches are selected for the setting unit, values will be truncated, rounded up or rounded off to the fourth place to the right of the decimal point.

SECTION 12 SCHEDULE PROGRAMS

1. Overview

Schedule programs permit different types of workpieces to be machined continuously without any operator intervention by using a pallet changer, or other automated loading and unloading units.

- A schedule program specifies the execution order of several main programs.
- A schedule program is a set of the following five blocks.
If other blocks are specified, an alarm will occur.
The program must end with the END block.
 - a. PSELECT block Selects and executes a main program.
 - b. GOTO block Branches unconditionally.
 - c. IF block Branches conditionally.
 - d. VSET block Sets variables.
 - e. END block Ends a schedule program.
- These commands must be specified at the start of, or immediately after, the sequence name.
- Comments specified between '(' and ')' and continuous lines identified by '\$' are valid.
- The size of schedule programs differs according to the selected operation method, normal tape (method A) or large-volume tape (method B), as follows.
 - a. For method A, the total tape length for the main, sub, and library programs is up to the maximum size of operation buffer area which is selected by the specification.
 - b. For method B, the tape length of the schedule program must be less than about 5 meters.

For schedule program operations, such as program selection, refer to Schedule Program Function, SECTION 5 Automatic Mode Operation in II OPERATION of Operation Manual.

The blocks specified in a schedule program are described below.

2. PSELECT Block

[Function]

A PSELECT block selects and executes main programs for a workpiece to be machined.

- This function searches a specified main program file for a specified main program to be selected as a machining program. This function also searches a specified subprogram file, or system subprogram file, and manufacturer subprogram file for the required subprograms and selects them automatically.
- After the completion of selection, program execution starts if the single block function is “off”, or program execution does not start immediately but is suspended until the cycle start signal is given if the single block function is “on”. The selected programs are executed repeatedly as specified.
- If the system is not in the automatic mode, the selection and execution of a main program are delayed until the system is placed in the automatic mode.

[Programming Format]

The commands must be specified in the following order:

PSELECT [fm], [pm], [fs] [:OP], [n] (CR) or (LF)

Commands enclosed by [] may be omitted. Note that a comma “,” may also be omitted if the items that follow are all omitted.

(1) fm: Main program file name

[3 characters] : [Within 16 characters] [. 3 characters]
 Device name File name Extension

* : Entries enclosed by [] may be omitted.

ME33018R1001400020001

- If a device name, a file name, and/or an extension is omitted, entry of “MD1”, “A”, and “MIN”, respectively, are assumed to apply. If all entries for “fm” are omitted, “MD1:A.MIN” is assumed to apply.
- An alarm will occur if “*” or “?” is used in a main program file name.
- An alarm will occur if the specified file does not exist.

(2) pm: Main program name

O
 {
 Within 5 characters

ME33018R1001400020002

- If an entry of “pm” is omitted, it is considered that the program name of the first program in the file set by “fm” is set.
- An alarm will occur if the specified program does not exist in the selected main program file, fm.

- An alarm will occur if M02 or M30, indicating the end of the program, is not specified in the specified main program.

(3) fs: Subprogram file name

[3 characters] : [Within 16 characters] [. 3 characters]
 Device name File name Extension

* : Entries enclosed by [] may be omitted.

ME33018R1001400020003

- Entry of "fs" may be omitted when:
 - a. No subprogram call command is specified in a main program.
 - b. The subprogram called from a main program or subprogram exists in MD1:*.SSB (system subprogram) or in MD1:*.MSB (manufacturer subprogram).
 - c. Required subprograms other than SSB and MSB are contained in the main program file. If fs is specified, the device name and extension may be omitted. The defaults for the device name and extension are "MD1" and "SUB", respectively. Therefore, if everything is omitted, it is assumed that no file has been specified.
- An alarm will occur if the total number of subprograms used exceeds 126.
- An alarm will occur if "RTS" which means the end of a subprogram is not specified.
- An alarm will occur if the required subprogram name in the range OO000 to OO999 does not exist in the manufacturer subprogram file.
- An alarm will occur if the required subprogram name outside the range OO000 to OO999 does not exist in the "fs" specified file or the system subprogram file.
- An alarm will occur if the file specified by "fs" does not exist.

(4) n: Repetition count

Q	=	Expression
---	---	------------

Q : Number of repetitions (specified by address)

The setting range is from 1 to 9999 and "1" is assumed to apply if entry of "n" is omitted.

An alarm will occur if a number outside the range 1 to 9999 is specified.

= : Symbol "□" may be used instead of "=". "=" may be omitted if it is directly followed by a numeric value.

ME33018R1001400020004

- (5) OP: Option specifications
- a. Specification of S option

`; S`

ME33018R1001400020005

This is the command not to search for subprograms.

An S option significantly reduces the time needed to execute the PSELECT command. This option is effective only for main programs and if the subprogram function or the branch function is used, an alarm occurs.

The S option is valid only for operation method B and invalid for method A.

For the difference between a normal tape (method A) and a large capacity tape (method B), with and without the S option specification, refer to Table "Operation Comparison between Normal Storage Capacity Memory and Large Storage Capacity Memory" given later.

- b. Specification of A option

`; A`

ME33018R1001400020006

The specification of an A option always selects the program running method A independently of the program running method selected in the MAIN PROGRAM SELECT (MEMORY MODE) screen.

- c. Specification of B option

`; B`

ME33018R1001400020007

The specification of a B option always selects the program running method B independently of the program running method selected in the MAIN PROGRAM SELECT (MEMORY MODE) screen.

- Program requirements in each program running method

Item		Program of normal size	Large program		
Program running method		Method A	Method B	Method S	
Program size limit	Main program	Total program size is 2 MB.	2GB		
	Sub program		—		
	Library program		Total program size is about 1.8 MB.	Total program size is about 1.8 MB.	
	Schedule program				
Sub program function		Usable	Usable	Unusable (alarm)	
Branch function		Usable	Usable	Unusable (alarm)	
Destination of a jump specified in branch command	Main program	Sequence label or sequence number	Sequence label or sequence number	—	
	Sub program				
	Library program				
	Schedule program				
Main program sequence label limit		Unlimited	Unlimited	Unlimited	
Program selection time		*1	*1	Completed immediately	

*1. Time varies with the selected program size.

3. Branch Block

The branching function of the schedule program, which is identical to SECTION 11, 1-1. "Branch Functions", falls into GOTO and IF blocks, which provide unconditional branching and conditional branching, respectively.

(1) GOTO Block

[Function]

The GOTO block unconditionally changes program sequences. The destination of a jump is specified using a sequence name immediately after the GOTO command.

[Programming format]

Commands must be specified in the following order:

GOTO □ N _____

Specifies the destination of a jump

(2) IF Block

[Function]

The IF block conditionally changes program sequences. If the condition is 'true', the sequence branches to the destination of a jump. If the condition is 'false', it proceeds to the next sequence.

[Programming format]

Commands must be specified in the following order:

IF [Expression] □ [Comparison operator] □ [Expression]] N _____
 Specifies the destination of a jump

The comparison operators include LT (<), LE (\leq), EQ (=), NE (\neq), GE (\geq), GT (>).
 For details, refer to the table in Branch Function in SECTION 11.

ME33018R1001400030002

4. Variables Setting Block

[Function]

'VSET' command is used to set variables in a schedule program.

[Programming format]

Commands must be specified in the following order:

VSET □ [Variable] = [Expression] □ [Variable] = [Expression]

ME33018R1001400040001

Variable in the left part: Specify a common, system, or output variable. If other variable is specified in the left part, an alarm will occur.

Right part: An alarm will occur if an output variable is specified.

5. Schedule Program End Block

[Function]

At the end of a schedule program, an "END" block must always be specified. All blocks specified following the "END" block are invalid.

[Programming format]

END

SECTION 13 OTHER FUNCTIONS

1. Table Index Specification

For the additional axis index specification, 5 index specification and 1 index specification are available.

The following explanation assumes that the B-axis is installed as the fourth axis.

1-1. 5-Degree Index Commands

[Programming Format]

B△△△ or B△△△.

△△△ : Commands can be specified in units of 5 degrees (decimal fractions are rounded off) and the programmable range is from 0 to 360 degrees.

Note that the unit system for this command does not follow the unit system selected for programming.

ME33018R1001500020001

[Details]

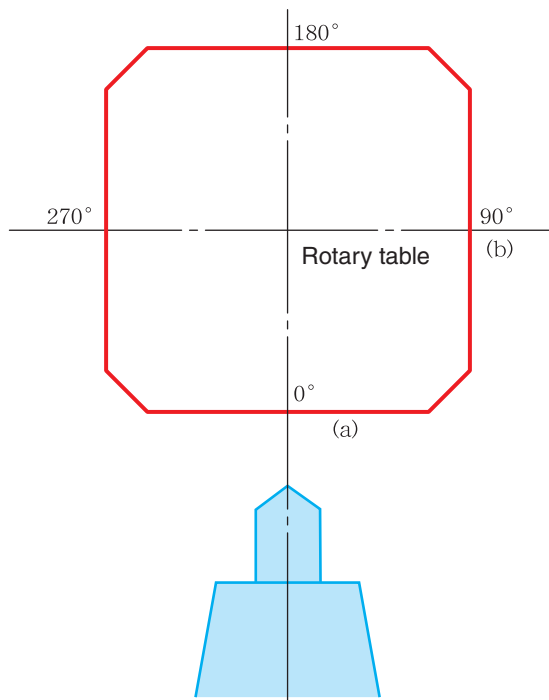
- Commands are executed as absolute even in the incremental mode.
- The direction of rotary table rotation is specified by M15 and M16.
M15: Forward rotation (CW rotation)
M16: Reverse rotation (CCW rotation)

Example:

In the illustration below, indexing to face (b) and then indexing back to face (a) is specified by the series of commands indicated below.

M15
B90
M16
B0

- In the G01 mode, a B command should be programmed in a block not containing other axis movement commands. In this case, the B command is executed at a rapid feedrate (G00 mode). In the G00 mode, it can be programmed with other axis movement commands in the same block.



Example:
Moving the B-axis in a fixed cycle mode

```
N....G81 G56 X...Y...R...Z...F...H...M52
N....B60 X50 M15
N....X0 M52
N....B90
N....X0 M52
```

ME33018R1001500020002

1-2. 1-Degree Index Commands

[Programming Format]

B△△△△ or B△△△△.

△△△△ : Commands can be specified in units of 1 degrees (decimal fractions are rounded off) and the programmable range is from 0 to 360 degrees.

Note that the unit system for this command does not follow the unit system selected for programming.

ME33018R1001500030001

[Details]

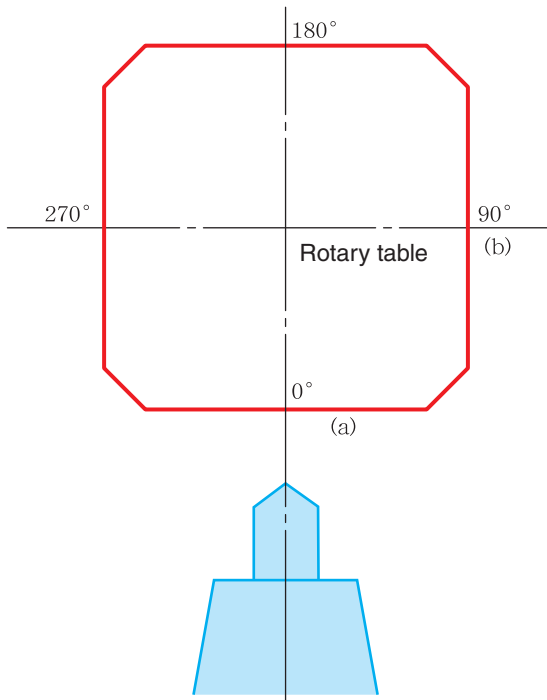
- Commands are executed as absolute even in the incremental mode.
- The direction of rotary table rotation is specified by M15 and M16.
M15: Forward rotation (CW rotation)
M16: Reverse rotation (CCW rotation)

Example:

In the illustration below, indexing face (b) and then indexing back to face (a) is specified by the series of commands indicated below.

```
M15
B90
M16
B0
```

- In the G01 mode, a B command should be programmed in a block not containing other axis movement commands. In this case, the B command is executed at a rapid feedrate (G00 mode). In the G00 mode, it can be programmed with other axis movement commands in the same block.



Example:
Moving the B-axis in a fixed cycle mode

```
N....G81 G56 X...Y...R...Z...F...H...M52
N....B60 X50 M15
N....X0 M52
N....B90
N....X0 M52
```

1-3. 0.001 Degree Commands (Optional)

With the 0.001° command specification, selection is possible whether the axis is treated as a rotary axis which allows the designation for operation within the range up to 360 degrees, or treated similarly to a linear axis (multi-turn axis) for which the operation range is not restricted.

1-3-1. Rotary Axis Command

[Programming Format]

B△△△.△△△ or B△△△△△△

ME33018R1001500050001

0.001° (0.0001°) unit: $-360.000 \leq B \leq 360.000$ ($-360.0000 \leq B \leq 360.0000$)

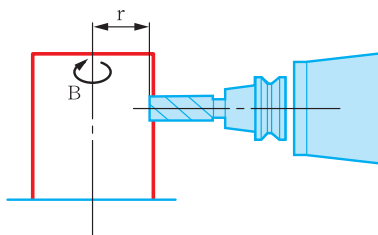
The unit system conforms to the setting at LENGTH and ANGLE of NC optional parameter (INPUT UNIT SYSTEM).

Programming is possible in units of 0.001°, 0.0001°, or 1°.

[Details]

- The direction of rotary table rotation is specified by M15 and M16.
M15: Forward rotation (CW rotation)
M16: Reverse rotation (CCW rotation)
In the incremental mode, direction of table rotation is determined as indicated below.
Positive command value: Forward rotation (CW rotation)
Negative command value: Reverse rotation (CCW rotation)
- As with linear axes (X, Y, and Z), cutting feed in the G01 mode and also linear interpolation designation with the combination of a linear axis are possible.
In this operation, however, the following must be taken into consideration for determining feedrate.
In a block where only a B-axis command is specified, F100 means "100 deg/min". However, if a B-axis command is specified with other axis commands in the same block, F100 is interpreted as "100 mm/min" or "100 inch/min".

- Example 1:

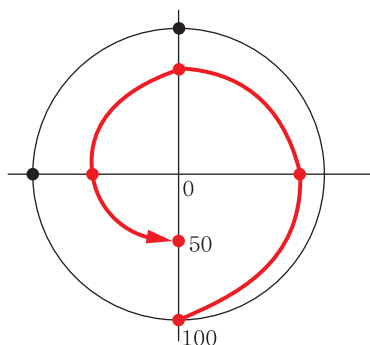


Assuming "r" in the figure at the left to be 100 mm, the B-axis rotates at a rate of 100 deg/min when "G91 G01 B100 F100" are programmed.

ME33018R1001500050002

The actual cutting speed (surface speed on a workpiece at a cutting point) is:
 $100 \times 2 \times 3.14 \times (100/360) = 174 \text{ mm/min}$

- Example 2:
If "G91 G01 B360 Z-50 F100" is specified.



The axis movement distance is calculated as:

$$\sqrt{(360)^2 + (-50)^2} = 363.456 \text{ mm}$$

The feedrate is 100 mm/min.

The time required for the axes to move the programmed distance is:
 $363.456/100 = 3.6 \text{ min}$

ME33018R1001500050003

1-3-2. Multi-turn Command

[Programming Format]

B△△△.△△△ or B△△△△△△

ME33018R1001500060001

0.001° (0.0001°) unit: $-9999.999 \leq B \leq 9999.999$ ($-9999.9999 \leq B \leq 9999.9999$)

The unit system conforms to the setting at LENGTH and ANGLE of NC optional parameter (INPUT UNIT SYSTEM).

Programming is possible in units of 0.001°, 0.0001°, or 1°.

[Details]

- The direction of rotation is determined according to the relative position of the target point with respect to the actual position regardless of the dimensioning mode (absolute or incremental). When the target point is located in the direction in which coordinate values increase with respect to the actual position, the axis rotates in the forward direction. If it is in the direction in which coordinate values decrease, the axis rotates in the reverse direction. Designation of M15/M16 is ignored.
- Travel limits (P/N) are fixed.
Travel limit (P): 9720.000 (9720.0000)
Travel limit (N): -9720.000 (-9720.0000)
- When the NC is reset, whether the workpiece coordinate value of the current table can be set to the value of the range of 0 to 359.999 degrees by using the item "Set the current workpiece coordinate value of multi-turn axis within 360° when resetting the NC" in the rotary axis parameter.

2. Angular Commands

[Function]

An angular command allows a target point to be defined by a coordinate value of one axis in the specified plane and the angle a line makes with the horizontal axis.

[Programming Format]

AG = $\triangle\triangle\triangle$

$\triangle\triangle\triangle$: Angle (unit: degree)

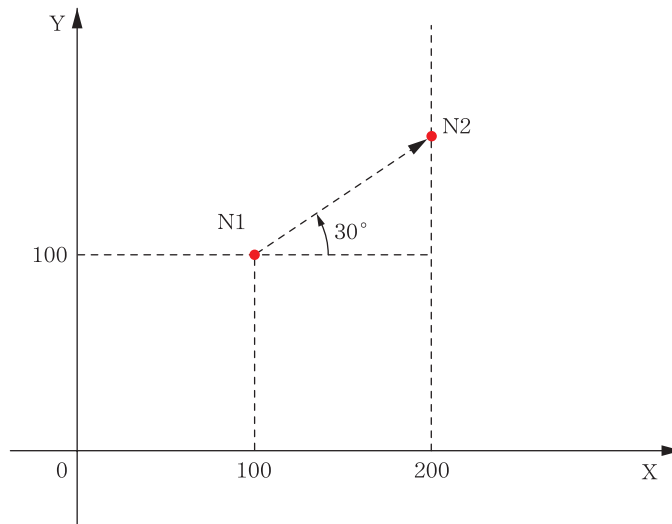
ME33018R1001500070001

The unit of an angle command can be changed by the setting at LENGTH and ANGLE of NC optional parameter (INPUT UNIT SYSTEM).

It is possible to specify a negative value.

- In the XY plane

N1 G00 X100 Y100 } This is equivalent to the commands
N2 G01 X200 AG = 30 } "N2 G01 X200 Y157.735".



ME33018R1001500070002

[Supplement]

If an angular command is used to define a point, do not specify both axes of the specified plane. Specify only one axis.

3. Manual Shift Amount Cancel Command

[Function]

The manual shift amount cancel command cancels the total distance moved in manual intervention during automatic operation by a command in the program without using switches on the operation panel.

The manual shift amount cancel function updates the coordinate values without actually moving the axes.

[Programming Format]

MITCAN

There are following two programming patterns. Note that a block skip command may be specified.

- Without a sequence number (label)
MITCAN
- With a sequence number (label)
N***MITCAN

Operation

(1) When the manual shift amount cancel command (MITCAN) is executed, the manual shift amount is canceled and the actual position data of the machine are taken as the calculated value. In this processing, no actual axis movement takes place.

- Example:

- Before canceling manual shift amount

$$\begin{array}{rcccl} 100.0 & + & 500.0 & = & 600.0 \\ \text{(Manual shift amount)} & & \text{(Calculated value)} & & \text{(Actual position data)} \end{array}$$

- After canceling manual shift amount

$$\begin{array}{rcccl} 0.0 & + & 600.0 & = & 600.0 \\ \text{(Manual shift amount)} & & \text{(Calculated value)} & & \text{(Actual position data)} \end{array}$$

ME33018R1001500080001

(2) If an axis movement command is specified in the incremental mode (G91) immediately after canceling the manual shift amount, it is executed as an incremental value with respect to the previously specified command.

- Example 1: Assume that manual shift amount is X = 50, Y = 50, and Z = 0.

```
N100 G90 G0 X400 Y300 Z0
N101 MITCAN
N102 G91 X20
N103 Y10
```

ME33018R1001500080002

- Axis movements

```
N100 / N101 (X, Y, Z) = (450, 350, 0)
N102 (X, Y, Z) = (420, 350, 0)
N103 (X, Y, Z) = (420, 310, 0)
```

ME33018R1001500080003

- Operations

- N100 Positioning is performed to $(X, Y, Z) = (450, 350, 0)$ in the work coordinate system.
Command value and calculated value are both $(X, Y, Z) = (400, 300, 0)$.
- N101 Calculated value is updated to $(X, Y, Z) = (450, 350, 0)$.
- N102 Positioning is performed to $(X, Y, Z) = (420, 350, 0)$ in the work coordinate system.
Command value is $(X, Y, Z) = (420, 300, 0)$.
Calculated value is $(X, Y, Z) = (420, 350, 0)$.
- N103 Positioning is performed to $(X, Y, Z) = (420, 310, 0)$ in the work coordinate system.
Command value is $(X, Y, Z) = (420, 310, 0)$.
Calculated value is $(X, Y, Z) = (420, 310, 0)$.

ME33018R1001500080004

Positioning is performed to the position where the manual shift amount is added to the calculated value. That is, the axes move from the previous calculated position according to the specified command with the manual shift amount added to the command value.

- Example 2: Assume that

Manual shift amount is $X = 50, Y = 50, \text{ and } Z = 0$.

Tool change position is $X = 700, Y = 0, Z = 0$

```
N100 G90 G0 X400 Y300 Z0
N101 M06
N102 MITCAN
N103 G91 X20 Y10
```

ME33018R1001500080005

- Axis movements

```
N100 (X, Y, Z) = (450, 350, 0)
N101 / N102 (X, Y, Z) = (700, 0, 0)
N103 (X, Y, Z) = (420, 310, 0)
```

ME33018R1001500080006

- Operations

- N100 Positioning is performed to $(X, Y, Z) = (450, 350, 0)$ in the work coordinate system.
Command value and calculated value are both $(X, Y, Z) = (400, 300, 0)$.
- N101 Positioning is performed to the tool change position $(X, Y, Z) = (700, 0, 0)$.
Command value is $(X, Y, Z) = (400, 300, 0)$.
Calculated value is $(X, Y, Z) = (650, -50, 0)$.
- N102 Calculated value is updated to $(X, Y, Z) = (700, 0, 0)$.
- N103 Positioning is performed to $(X, Y, Z) = (420, 310, 0)$ in the work coordinate system.
Command value is $(X, Y, Z) = (420, 310, 0)$.
Calculated value is $(X, Y, Z) = (420, 310, 0)$.

ME33018R1001500080007

[Details]

- An alarm occurs if the manual shift amount cancel command (MITCAN) is executed in the cutter radius compensation mode or the 3D offset mode.
- Before executing sequence re-start, the manual shift amount must be canceled. Note that manual shift amount cannot be canceled if the MITCAN command is specified in the block in sequence restart operation or the block for sequence restart.
- Only the sequence number (sequence label) and the block skip command can be specified before the MITCAN command.
- The MITCAN command must be specified without other commands in the same block.
Example:

```
N100 MITCAN X100 Y0
```

These commands are disregarded.

4. Print Format Function

[Function]

When executing print statement, the print format can be specified by this function.

Printing by print commands with format specification (PRINTF and SPRINTF), print according to the format specified the system variable VFMT [α , β] for print format specification. The following system variables, which are fixed by PRINT and SPRINT, are not the exception when printed by PRINTF and SPRINTF. Do not use print commands with format specification when performing the fixed format print for the following variables.

VRBDG

VINTG

VNUM

VOK

VOK2

VPCNT

(1) Print format system variable VFMT

This is a variable to command a print format of the data when issuing a print command with format specification (PRINTF, SPRINTF). This variable is used only in the PRINTF statement or SPRINTF statement.

VFMT [α , β]

α : The number of digits before decimal point (0 to 8)

β : The number of digits after decimal point (0 to 8)

The value must be $8 \geq \alpha + \beta$.

If the decimal point is included in α , β , digits after the decimal point are truncated and the value is handled as an integer.

If the value is larger than displayed number of digits, replace it with the maximum value as large as possible.

When resetting the NC and turning on the power, it is set as VFMT [5, 3].

VFMT [0, 0] is the same format as PRINT statement.

- Example) When VC1 = -12.35
If VFMT [1, 1], the value is "-9.9".

(2) Printer control system variable VPRT

Effectiveness of the leading space suppression function by a print command with BIT3 format command

Delete spaces preceding numbers.

0 ... Leading space suppression function is effective Example) X-12.345

1 ... Leading space suppression function is ineffective Example) X_-12.345

- Example) When VC1 = 1234.5678, VC2 = -1.3
PRINTF "AAA =" VFMT [5,3] VC1 "__BBB=" VFMT [2.2] "_END"

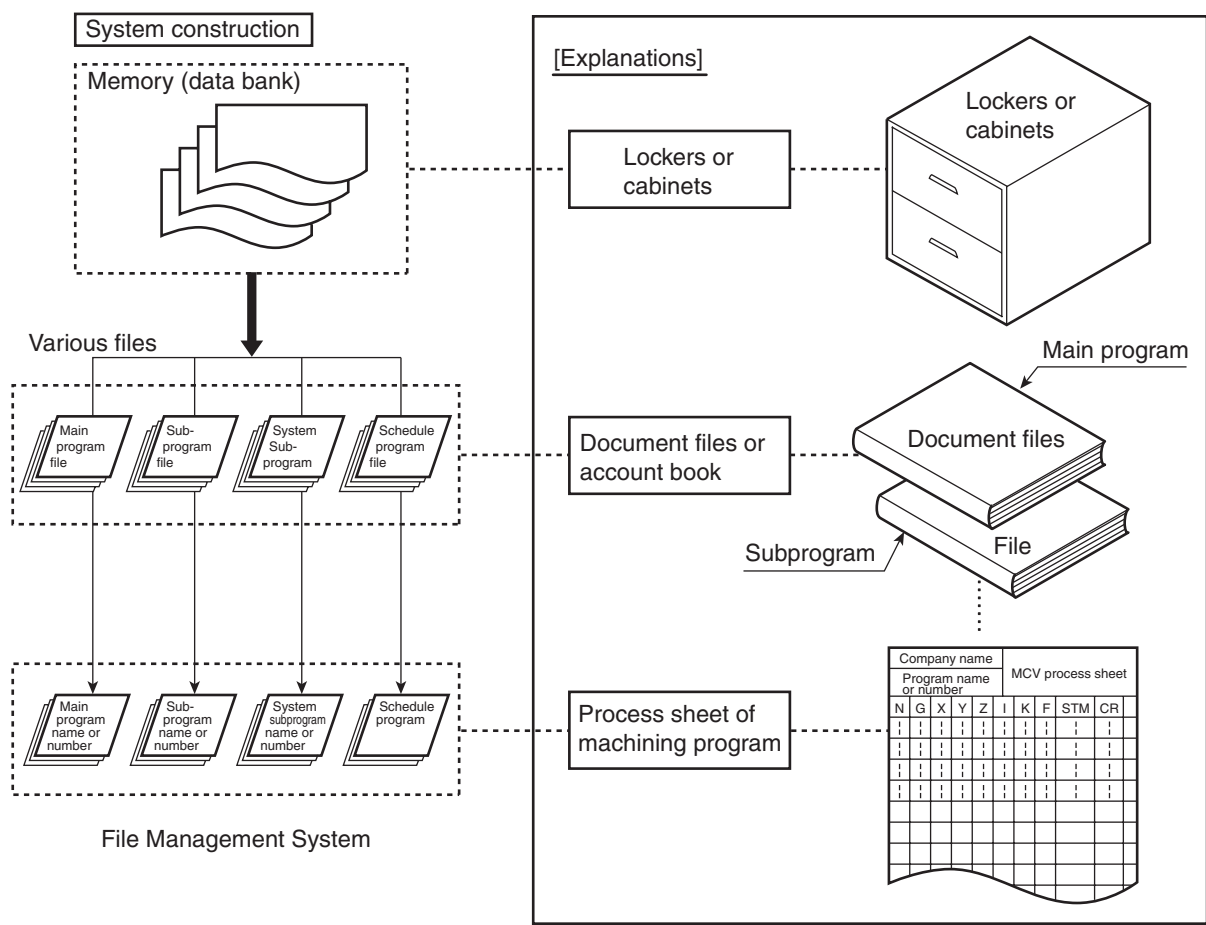
If BIT3 of VPRT is "1" AAA=1234.568__BBB=-1.30_END

If BIT3 of VPRT is "0" AAA=_1234.568__BBB=_-1.30_END

SECTION 14 FILE MANAGEMENT

1. Files

- (1) Programs are executed after they have been stored in the NC memory.
- (2) The memory has a storage space of 2 GB and can store a number of programs at the same time.
- (3) To facilitate the handling of stored programs, each is stored in respective files with different names. This is similar to keeping documents in files in lockers or cabinets. The following figure shows the image of file management system.



ME33018R1001600010001

- (4) The file management system is composed of the following three kinds:
 - Memory (data bank)
This corresponds to a locker or cabinet where data is stored in units of files.
 - Various files
Each file consists of documents or account books.
 - Program name or number
Program numbers or names correspond to individual documents. (process sheet)

2. Various Files

Files may be equivalent to document files or account books, and each file for the same workpiece type is assigned a name (file name), which consists of the main file name and an extension.

A file name should consist of up to 16 alphanumeric characters including a minus code, beginning with an alphabetical character. An extension should consist of 3 characters beginning with an alphabetical character, separated from a main file name by a period(.)between them.

Types of files:

- Main program file: [Main file name]. MIN
- Subprogram file: [Main file name]. SUB
- System subprogram file: [Main file name]. SSB
- Maker subprogram file: [Main file name]. MSB
- Library program file: [Main file name]. LIB
- Schedule program file: [Main file name]. SDF

(1) Main program file

This is a file which basically files main programs.

The main program file can be used to register subprograms which are called from the main program filed in it.

Example:

(2) Subprogram file

This is a file of subprograms.

When a machining cycle is to be carried out with the programs in this file, it is necessary to specify the subprogram name in selecting programs.

(3) System subprogram file

Sub programs, when they are called and used, must always be specified by program selection.

When using a subprogram in the system sub program, however, it is not necessary to specify the subprogram by program selection

(4) Maker subprogram file

This file is handled in the same manner as a system subprogram file. In this file, subprograms made by the machine tool builder for such automating are filed.

(5) Library program file

Generally, for the use of a subprogram in the MDI mode operation entering the desired subprogram number such as CALL O100, the main program which called this subprogram must be selected in advance.

By filing the subprograms which are often called in the MDI mode operation in the library program file, they can be easily called.

(6) Schedule program file

Schedule program automatically executes machining of different kinds of workpieces successively using the pallet changer or other automation devices without the help of setup change by the operator. In this schedule program, you can specify the execution order of several main programs.

SECTION 15 APPENDIX

1. G Code Table (Including Optional Functions)

G Code	G Group	Functions
G00 ***	1	Positioning
G01 ***		Linear interpolation
G02		Circular interpolation - Helical cutting (CW)
G03		Circular interpolation - Helical Cutting (CCW)
G04 **	2	Dwell
G09 **	18	Exact stop
G10 *	3	Cancel of G11
G11		Parallel and rotational shift of coordinate system
G14	64	Designation of axis name
G15	4	Selection of work coordinate system (Modal)
G16 **		Selection of work coordinate system (One-shot)
G17 ***	5	Plane selection: XY
G18 ***		Plane selection: ZX
G19 ***		Plane selection: YZ
G20 **	15	Inch input confirmation
G21 **		Metric input confirmation
G22 ***	6	Programmable stroke limit ON
G23 ***		Programmable stroke limit cancel
G30 **	16	Positioning to home position
G31 **	17	Skip function
G33	1	Unfixed cycle for thread cutting
G37	22	Cancel of G38
G38		Air cut reduction mode ON
G39 **	49	I, J, K commands and G39 ignored in the corner circular interpolation (with vector)
G40 *	7	Cutter radius compensation (G41, G42) cancel
G41		Cutter radius compensation, left
G42		Cutter radius compensation, right
G43 *	8	3D offset (G44) cancel
G44		3D offset ON
G50 *	9	Enlargement and reduction of geometry cancel
G51		Enlargement and reduction of geometry "ON"
G53 ***	10	Tool length offset cancel
G54		Tool length offset, X-axis
G55		Tool length offset, Y-axis
G56 ***		Tool length offset, Z-axis
G57		Tool length offset, 4th-axis
G58		Tool length offset, 5th-axis
G59	Tool length offset, 6th-axis	
G60	1	One-directional positioning

G Code	G Group	Functions
G61	14	Exact stop mode ON
G62	19	Programmable mirror image mode
G64 *	14	Cutting mode ON
G68	24	Slope coordinate OFF
G69		Slope coordinate ON
G71	21	Designation of return level for M53
G72	23	Designation of pattern reference point (start position) for the coordinate calculation function
G73	11	Fixed cycle, High speed drilling cycle
G74		Fixed cycle, Reverse tapping cycle
G75		Fixed cycle; Step & pick cycle
G76		Fixed cycle, Fine boring
G79		Fixed cycle; Variable pitch cycle
G80 *		Fixed cycle, Modal cancel
G81		Fixed cycle; Spot boring
G82		Fixed cycle; Counter boring
G83		Fixed cycle, Deep hole drilling cycle
G84		Fixed cycle, Tapping cycle
G85		Fixed cycle, Boring cycle
G86		Fixed cycle, Boring cycle
G87		Fixed cycle, Back boring cycle
G88		Fixed cycle; Multi-step boring cycle
G89	Fixed cycle, Boring cycle	
G90 ***	12	Absolute dimensioning mode
G91 ***		Incremental dimensioning mode
G92	20	Setting of work coordinate system
G93	13	Inverse time feed
G94 ***		Feed per minute mode
G95 ***		Feed per revolution mode
G96	20	Local coordinate system setting
G97 **	74	Coordinate recalculation function
G100	32	Cancel of macro MODIN mode selected by G101 to G110
G101		G code macro MODIN type
G102		
G103		
G104		
G105		
G106		
G107		
G108		
G109		
G110		
G111		G code macro CALL type
G112		

G Code	G Group	Functions
G113 G114 G115 G116 G117 G118 G119 G120	32	G code macro CALL type
G130 * G131	92	High-speed contouring control OFF High-speed contouring control ON
G133 * G134	89	Constant peripheral speed control OFF Constant peripheral speed control ON
G137 * G138	61	Contour machining mode OFF Contour machining mode ON
G139 * G140	90	G00 special linear interpolation OFF G00 special linear interpolation ON
G141 * G142	91	Tool grooving OFF Tool grooving ON
G143	66	Rotary axis position command (under rotary axis control by Super hi-NC)
G145 G146	93	Pattern reference point return mode Pattern end point return mode
G147	1	Machine coordinate system position command
G148 * G149	91	Turning cut function OFF Turning cut function ON
G150 G151 * G152 G153	89	Auto acceleration/deceleration ON Auto acceleration/deceleration cancel Auto acceleration Auto deceleration
G154 * G155	91	Spindle path control OFF Spindle path control ON
G159	50	In-position check disabling command
G164 * G165 G166	95	Continuous circular movement stop command in turning cut Continuous circular movement (CW) start command in turning cut Continuous circular movement (CCW) start command in turning cut
G167 * G168	91	Rotary table and spindle control OFF Rotary table and spindle control ON
G169 G170 * G171	90	Tool length offset at cutting edge ON Tool length offset at cutting edge and in axial direction cancel Tool length offset in axial direction ON
G174 * G175	78	Cylindrical side cutting "OFF" Cylindrical side cutting "ON"
G176 * G177	72	F, S codes ignore cancel F, S codes ignore

G Code	G Group	Functions
G178 *	1	Fixed thread cutting cycle in the 1st axis direction on a plane
G179		Fixed thread cutting cycle in the 2nd axis direction on a plane
G180 *	65	Attachment rotation offset cancel
G181		Attachment rotation offset; Frontward
G182		Attachment rotation offset; Leftward
G183		Attachment rotation offset; Backward
G184		Attachment rotation offset; Rightward
G185		Attachment rotation offset; Downward
G186 *	66	Tolerance control mode cancel
G187		Tolerance control mode "ON"
G188 *	70	2nd tool length offset cancel
G189		2nd tool length offset ON
G192 *	68	Fixed cycle; Upper surface detection cancel
G193		Fixed cycle; Upper surface detection ON
G194 *	69	Fixed cycle; Tool breakage detection cancel
G195		Fixed cycle; Tool breakage detection ON
G260 *	81	Spindle deflection compensation OFF
G261		Spindle deflection compensation ON
G266 *	83	Limit switch checked
G267		Limit switch not checked
G272	11	Fixed cycle; Deep hole tapping and reverse tapping
G273		Fixed cycle; High-speed deep hole tapping and reverse tapping
G274		Fixed cycle; Synchronized tapping, reverse tapping
G281		Fixed cycle; Small diameter and deep hole drilling
G282		Fixed cycle; Deep hole synchronized tapping
G283		Fixed cycle; High-speed deep hole synchronized tapping
G284		Fixed cycle; Synchronized tapping
G296 *	61	G297 cancel
G297		S command in C-axis mode (S command is regarded as the spindle position.)
G300	33	G301 to G349 macro MODIN cancel (NC. P. B No. 34 bit 3:1)
G336		I-MAP-A function; Fixed circle cutting cycle (CW)
G337		I-MAP-A function; Fixed circle cutting cycle (CCW)
G501 to G580	77	G code macro changeable between MODIN and CALL

* : Has already been set when power supply is turned on.

** : Valid only in the specified block.

*** : The initial state can be set with the relevant parameter.

2. Table of Mnemonic Codes (Including Optional Functions)

Mnemonic Code	Group	Functions
EIN DIN RTI	73	Program interruption function; Valid Program interruption function; Invalid Program interruption function; End code
READ WRITE	63	READ/WRITE GET/PUT function; Reading from an external device READ/WRITE GET/PUT function; Writing onto an external device
FOPENA FOPENB FWRITC CLOSE	88	READ/WRITE GET/PUT function; File A open READ/WRITE GET/PUT function; File B open READ/WRITE GET/PUT function; Writing onto a file READ/WRITE GET/PUT function; Closing a file
NURBS	1	NURBS curve interpolation
NOINC	40	G90 effective in one block (absolute) command (used only for the system)
MITCAN	26	Manual shift cancel
CALIN CALOUT	27	Block call for each sub program ON Block call for each sub program cancel
TRDL TRDT	71	Compound fixed cycle; Thread cutting in 1st axis direction on plane Compound fixed cycle; Thread cutting in 2nd axis direction on plane
OMIT RSTRT	29	Coordinate calculation function; Omit Coordinate calculation function; Restart
LAA ARC GRDX GRDY DGRDX DGRDY SQRX SQRY BHC	30	Coordinate calculation function; Line at angle Coordinate calculation function; Arc Coordinate calculation function; Grid X Coordinate calculation function; Grid Y Coordinate calculation function; Double grid X Coordinate calculation function; Double grid Y Coordinate calculation function; Square X Coordinate calculation function; Square Y Coordinate calculation function; Bolt hole circle
GRON GROF	56	Graphic tool path; High-speed drawing ON Graphic tool path; High-speed drawing OFF
GRSK	57	Graphic tool path; Skipping tool path (not displayed)
GRER GRST	58	Graphic tool path; Deleting tool path Graphic tool path; Tool path data setting
GRCI GRCV	59	Graphic tool path; Graphic drawing coordinate system invalid (work coordinate) Graphic tool path; Graphic drawing coordinate system valid (specified coordinate)
RTMCR	38	End of macro processing (generated in the system)
NOEX	34	Specifies the sequence that is only read and not executed.

Mnemonic Code	Group	Functions
CALL RTS MODIN MODOUT	27	Subprogram, Simple call Subprogram, Simple call End code Subprogram, Call after positioning mode "ON" Subprogram, Call after positioning mode "OFF"
GOTO IF	28	Branch command, Unconditional jump Branch command, Conditional jump
DEF DELETE SAVE	62	Animation function UGC (user graphic command); Declaration of registration Animation function UGC (user graphic command); Statement for deletion Animation function UGC (user graphic command); Backup of bank
PMIL PMILR FMILR FMILF RMILO RMILI	31	Area machining, Pocket milling (Zigzag) Area machining, Pocket milling (Spiral) Area machining, Face milling (Rough) Area machining, Face milling (Finish) Area machining, Round milling (Out) Area machining, Round milling (In)
COPY COPYE	39	Parallel/Rotary shift of the local coordinate system ON (initial setting) Parallel/Rotary shift of the local coordinate system OFF (incremental amount)
CLEAR DRAW	58	Animation function; Deletion of the graphic Animation function; Blank drawing
MOPLD	60	MOP-TOOL function; Request to read tool data file
MSG NMSG	35	Message display Return to the previous display
PRINT LPRINT SPRINT	41	Print function Print function (Designation of double-byte characters) Print function (No life feed)
CHFC CHFR	76	Arbitrary-angle chamfering function (Chamfering) Arbitrary-angle chamfering function (Rounding)
MHPS	61	Selection of a robot/loader program
RQNFL	60	Request to select next program from multi-volume files (used only for the system)
NCYL	36	Fixed cycle; No cycle axis movement
RTMDI	37	MDI end code (used only for the system)
PREG	60	Registration of sub programs
PRINTF SPRINTF	41	Formatted print function Formatted print function (No line feed)
TPRF	85	Fixed cycle; High speed profiling (scroll machining function)

Mnemonic Code	Group	Functions
PRMDO	33	I-MAP-B function; Solid shape machining (convex pyramid)
PRMDI		I-MAP-B function; Solid shape machining (concave pyramid)
CONEO		I-MAP-B function; Solid shape machining (convex cone)
CAMCV		I-MAP-B function; Solid shape machining (cam curves interpolation)
PGEND		I-MAP-B function; Ridge machining (end setting)
PCIR		I-MAP-A function; Area machining cycle (Circular pocket milling)
RCIRO		I-MAP-A function; Area machining cycle (Round milling: Outside)
RCIRI		I-MAP-A function; Area machining cycle (Round milling: Inside)
RSQRO		I-MAP-A function; Area machining cycle (Round milling for corner R: Outside)
RSQRI		I-MAP-A function; Area machining cycle (Round milling for corner R: Inside)
RSQCO		I-MAP-A function; Area machining cycle (Round milling for corner chamfering: Outside)
TAPR		I-MAP-B function; Ridge machining (linear)
PRSOL		I-MAP-B function; Ridge machining (convex arc)
HORN		I-MAP-B function; Ridge machining (concave arc)
BOUNS		I-MAP-B function; Ridge machining (straight line tangential to a circle upper side)
CONEI		I-MAP-B function; Solid shape machining (concave cone)
SPHRO		I-MAP-B function; Solid shape machining (convex hemisphere)
SPHRI		I-MAP-B function; Solid shape machining (concave hemisphere)
LCONO		I-MAP-B function; Solid shape machining (convex horizontal cone)
LCONI		I-MAP-B function; Solid shape machining (concave horizontal cone)
RLDPN	I-MAP-B function; Solid shape machining (3D slope interpolation)	
TLFON	43	Tool life management ON
TLFOFF		Tool life management OFF
TLFR	44	Tool life management; Transferring remaining times to all the tool numbers
TLOK	44	Tool life management; Changing NG to OK
TLCO	46	Tool life management; Count number command
DA	47	Tool life management; 1st cutter radius compensation number
DB		Tool life management; 2nd cutter radius compensation number
DC		Tool life management; 3rd cutter radius compensation number
HA	48	Tool life management; 1st tool offset number
HB		Tool life management; 2nd tool offset number
HC		Tool life management; 3rd tool offset number
TLID	86	Tool life management; Counting tools used
TLDC	87	Tool life management; Clearing count of tools used

3. M Code Table

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
M00	1	Program stop	After	One shot	Spindle and coolant stop (Can be selected by parameter setting)
01		Optional stop	After	One shot	
02	18	End of program	After	One shot	NC reset
03	2	Work spindle start (CW)	At the same time	Modal	Rotates the work spindle counterclockwise when viewed from the workpiece. Rotates the work spindle clockwise when viewed from the workpiece.
04		Work spindle start (CCW)	At the same time	Modal	
05		Spindle stop	After	Modal	
06	3	Vertical spindle tool change	After	One shot	
07	8	Oil mist coolant ON	At the same time	Modal	
08	10	Coolant pump ON	At the same time	Modal	
09	24	Coolant system OFF (M07, 08, 12, 50, 51, 59 OFF)	After	Modal	
10	30	A-axis clamp	After	Modal	
11		A-axis unclamp	After	Modal	
12	22	Chip air blow ON	At the same time	Modal	
15	5	4th-axis rotary index table CW	At the same time	Modal	
16		4th-axis rotary index table CCW	At the same time	Modal	
17		Indexing head index CCW	At the same time	One shot	Command effective for only M73-M76
19	2	Spindle orientation (forward)	After	Modal	
20	31	B-axis clamp	After	Modal	
21		B-axis unclamp	After	Modal	
22	32	Y-axis clamp	After	Modal	
23		Y-axis unclamp	After	Modal	
24	33	Z-axis clamp	After	Modal	
25		Z-axis unclamp	After	Modal	
26	35	C-axis clamp	After	Modal	
27		C-axis unclamp	After	Modal	
30	18	End of tape	After	One shot	
32		Splash guard door close	At the same time	Modal	
33		Splash guard door open	At the same time	Modal	

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
40	11	High/middle-high/middle-low/low range	At the same time	Modal	Spindle gears are automatically determined by spindle speed command.
41		High/middle-high/middle-low range	At the same time	Modal	
42		High/middle-high range	At the same time	Modal	
43		High range	At the same time	Modal	
44		AAC (F) 1 Next attachment clear	At the same time	One shot	F: Floor type T: Table type
45		AAC (F) 1 Preparation for attachment change preparation	At the same time	One shot	
46		AAC (F) 1 No next attachment	At the same time	One shot	
47		AAC (F) 1 No next attachment	At the same time	One shot	
48		AAC (T) Next attachment clear	At the same time	One shot	
49		AAC (T) Preparation for attachment change preparation	At the same time	One shot	
50	23	Through-the-tool coolant, low pressure ON	At the same time	Modal	
51		Through-the-tool coolant, high pressure ON	At the same time	Modal	
52	12	Return level in fixed cycle, Upper limit	At the same time	One shot	
53	13	Return level in fixed cycle, Specified level	At the same time	Modal	
54		Return level in fixed cycle, Point R level	At the same time	Modal	
57	34	W-axis clamp	After	Modal	
58		W-axis unclamp	After	Modal	
59	25	Chip air blow ON	At the same time	Modal	
60	4	Pallet change command	After	One shot	
62		Vertical spindle tool change preparation	After	Modal	
63		No next tool for ATC	At the same time	One shot	
64		Next tool return cycle	At the same time	One shot	
65		ATC preparation	At the same time	One shot	
66		Continuous tool change between the vertical and horizontal spindles (same tool)	At the same time	One shot	
67		Continuous tool change between the vertical and horizontal spindles (different tool)	At the same time	One shot	
68		Vertical spindle tool clamp	After	Modal	
69		Vertical spindle tool unclamp	After	Modal	
70	3	Manual tool change	After	One shot	

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
71		Manual attachment tool change	After	One shot	
72		Horizontal spindle tool change preparation	After	Modal	
73	15	Swivel head, front position	After	One shot	
74		Swivel head, left position	After	One shot	
75		Swivel head, rear position	After	One shot	
76		Swivel head, right position	After	One shot	
77		Horizontal spindle tool change	After	One shot	
78		Horizontal spindle tool clamp	After	Modal	
79		Horizontal spindle tool unclamp	After	Modal	
81		Automatic W-axis positioning 1	After	One shot	
82		Automatic W-axis positioning 2	After	One shot	
83		Automatic W-axis positioning 3	After	One shot	
84		Automatic W-axis positioning 4	After	One shot	
85		Automatic W-axis positioning 5	After	One shot	
87		Oil mist/air blow ON	At the same time	One shot	Valid for M90, 91 and 98
88		Dust collector ON	At the same time	Modal	
89		Dust collector OFF	At the same time	Modal	
90		Vertical spindle oil mist cycle mode ON	At the same time	Modal	
91		Tap-drill hole chip air blow cycle mode ON	At the same time	Modal	Turned OFF by M90
93		Thru-spindle coolant at medium pressure/2nd coolant ON	At the same time	Modal	
94	3	Attachment indexing by forward rotation	At the same time	One shot	
95		Attachment indexing by reverse rotation	At the same time	One shot	
98		Horizontal spindle oil mist cycle mode ON	At the same time	Modal	
101		Pallet 1 selection	After	One shot	
102		Pallet 2 selection	After	One shot	
103		Pallet 3 selection	After	One shot	
104		Pallet 4 selection	After	One shot	
105		Pallet 5 selection	After	One shot	
106		Pallet 6 selection	After	One shot	
107		Pallet 7 selection	After	One shot	
108		Pallet 8 selection	After	One shot	
109		Pallet 9 selection	After	One shot	
110		Pallet 10 selection	After	One shot	
111		Pallet 11 selection	After	One shot	
112		Pallet 12 selection	After	One shot	

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
115 116	6	5th-axis rotary table CW 5th-axis rotary table CCW	At the same time At the same time	Modal Modal	
118 119	2	Spindle orientation (reverse) Spindle orientation (forward/ reverse)	After After	Modal Modal	
120		Work shower ON	At the same time	Modal	
121		Attachment air blow ON/Tool nose air blow ON	At the same time	Modal	
127		Touch probe ON	At the same time	Modal	
130 131	9	Spindle rotation condition for cutting feed OFF Spindle rotation condition for cutting feed ON	At the same time At the same time	Modal Modal	Cutting feed (G01, G02, G03) is enabled if the spindle is rotating. Cutting feed (G01, G02, G03) is disabled if the spindle is not rotating.
132 133	37	Single block invalid Single block valid	At the same time At the same time	Modal Modal	
134 135	16	Spindle speed override invalid Spindle speed override valid	At the same time At the same time	Modal Modal	
136 137	17	Feedrate override invalid Feedrate override valid	At the same time At the same time	Modal Modal	
138 139	35	Dry run invalid Dry run valid	At the same time At the same time	Modal Modal	
140 141	39	Slide hold invalid Slide hold valid	At the same time At the same time	Modal Modal	
142 143		Spindle overload detection invalid Spindle overload detection valid	At the same time At the same time	Modal Modal	
144 145	19	Touch sensor advance Touch sensor retract	After After	Modal Modal	
146 147 148 149		Work air blow OFF Work air blow ON Work clamp (fixture 1) Work unclamp (fixture 1)	After At the same time At the same time At the same time	Modal Modal Modal Modal	
150 151 152 153		Coolant group specification, Vertical spindle Coolant group specification, Horizontal spindle Coolant group specification, 3rd group Coolant group specification, 4th group	At the same time At the same time At the same time At the same time	Modal Modal Modal Modal	

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
154 155	20	Sensor air blow OFF Sensor air blow ON	After At the same time	Modal Modal	
157 158 159		AAC (2 st.), No next tool AAC (2 st.), Next tool clear AAC (2 st.), Preparation for the next tool	After After After	One shot One shot One shot	
160 161		PPC pallet loading PPC pallet unloading	After After	One shot One shot	
163 165		Long tool, No next tool Long tool, Preparation for tool change	At the same time At the same time	One shot One shot	Specification of the safety cover for the next tool
166		ATC active tool return mode specification	At the same time	One shot	
170 171 172 173		AAC (F), Attachment change AAC (T), Attachment change Long tool, Tool change command AAC (2 st.), Attachment change command	After After After After	One shot One shot One shot One shot	F: Floor type T: Table type Column type
176		Dust collection mode, Air blow	At the same time	One shot	
177		Angular attachment, Tool change	After	One shot	AT-ATC
178 179		Dust collection mode, ON Dust collection mode, OFF	At the same time At the same time	Modal Modal	
181 182 183 184 185 186 187 188		External M signal External M signal External M signal External M signal External M signal External M signal External M signal External M signal	At the same time At the same time At the same time At the same time At the same time At the same time At the same time At the same time	One shot One shot One shot One shot One shot One shot One shot One shot	When "1" is set for machine user parameter (bit) No. 3, bit 1, an external M signal is output after the completion of axis movement.
190 191 192 193 194 195 196 197 198 199		Automatic W-axis positioning 1 Automatic W-axis positioning 2 Automatic W-axis positioning 3 Automatic W-axis positioning 4 Automatic W-axis positioning 5 Automatic W-axis positioning 6 Automatic W-axis positioning 7 Automatic W-axis positioning 8 Automatic W-axis positioning 9 Automatic W-axis positioning 10	After After After After After After After After After After	One shot One shot One shot One shot One shot One shot One shot One shot One shot One shot	For automatic W-axis positioning 10-level specification

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220	26	M code macro	At the same time	One shot	
230 231	42	Tool length offset direction; Used as it is Tool length offset direction; Used after reversing the direction	At the same time At the same time	Modal Modal	
232 233	41	3D tool length offset direction; Used as it is 3D tool length offset direction; Reversed and used	At the same time At the same time	Modal Modal	
234 235 236 237	51	Synchronized tapping gear selection: Low, Middle low, Middle high, High Synchronized tapping gear selection: Middle low, Middle high, High Synchronized tapping gear selection: Middle high, High Synchronized tapping gear selection: Middle high, High	At the same time At the same time At the same time At the same time	One shot One shot One shot One shot	
238 239	40	Software override invalid Software override valid	At the same time At the same time	Modal Modal	
278 279		Chip conveyor OFF Chip conveyor ON	At the same time At the same time	Modal Modal	
280		Work counter count up	After	One shot	

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
281		Work clamp (fixture 2)	At the same time	Modal	
282		Work unclamp (fixture 2)	At the same time	Modal	
287		Work clamp (fixture 3)	At the same time	Modal	
288		Work unclamp (fixture 3)	At the same time	Modal	
289		Pallet identification	At the same time	One shot	
292	45	Chamfering OFF	At the same time	Modal	
293		Chamfering ON	At the same time	Modal	
294		Fixed compound cycle; Cutting pattern 1	At the same time		
295	44	Fixed compound cycle; Cutting pattern 2	At the same time		
296		Fixed compound cycle; Cutting pattern 3	At the same time		
297	55	Thermal deformation amount transfer command	At the same time	One shot	
300		Ignoring Spindle rotation answer	At the same time	One shot	
301	46	Ignoring M code answer other than spindle rotation	At the same time	One shot	
302		M301 answer confirmation	At the same time	One shot	
306		Indexing B-axis attachment	At the same time	One shot	
311		Robot request 1	At the same time	One shot	
312	4	Robot request 2	At the same time	One shot	
313		Robot request 3	At the same time	One shot	
314		Robot request 4	At the same time	One shot	
315		U-axis connection OFF	After	Modal	
316		U-axis connection ON	After	Modal	
317		U-axis air blow OFF	At the same time	Modal	
318		U-axis air blow ON	At the same time	Modal	
319	53	Operation time transfer command (DNC-C)	At the same time	One shot	
320		Tailstock advance	At the same time	Modal	
321		Tailstock retract	At the same time	Modal	
326	52	Synchronized tapping monitor OFF	At the same time	Modal	
327		Synchronized tapping monitor ON	At the same time	Modal	
328		Spindle no-tool interlock valid	At the same time	Modal	
329		Spindle no-tool interlock invalid	At the same time	Modal	
331	57	6th rotary axis CW	At the same time	Modal	
332		6th rotary axis CCW	At the same time	Modal	
339		Thru-spindle air blow ON	At the same time	Modal	

M Code	Group	Function	Execution Timing (In Reference to Axis Movement Command)	Modal/ One shot	Remarks
340		Work seating monitor ON	At the same time	Modal	
341		Work seating monitor OFF	At the same time	Modal	
342		Work seating confirmation air ON	At the same time	Modal	
343		Work seating confirmation air OFF	At the same time	Modal	
346	59	B-axis rotation interlock valid	At the same time	Modal	
347		B-axis rotation interlock invalid	At the same time	Modal	
351		AT retraction to MG upper part (MCV-A II simplified 5-face MC)	At the same time	One shot	
352		Adaptor retraction to MG upper part (MCV-A II simplified 5-face MC)	At the same time	One shot	
354		Chip flusher OFF	At the same time	Modal	
355		Chip flusher ON	At the same time	Modal	
356		MG pot swing to tool change position	At the same time	One shot	
368		Mist collector OFF	At the same time	Modal	
369		Mist collector ON	At the same time	Modal	
374		Non-contact sensor; Laser OFF	At the same time	Modal	
375		Non-contact sensor; Laser ON	At the same time	Modal	
378		Coolant flow rate detection; Monitor mode ON	At the same time	Modal	
379		Coolant flow rate detection; Monitor mode OFF	At the same time	Modal	
385		Work clamp at low pressure (fixture 1)	At the same time	Modal	

The commanded state of the following M codes may be displayed in the M code field (BLOCK).

- M03, 04, 05, 19
- M06, 77
- M07
- M10, 11
- M15, 16
- M17
- M20, 21
- M22, 23
- M24, 25
- M40 - 43
- M53, 54
- M57, 58
- M60
- M73 - 76
- M115, 116
- M130, 131
- M132, 133
- M134, 135
- M136, 137
- M138, 139
- M140, 141
- M08
- M12
- M26, 27
- M30
- M50, 51
- M52
- M59
- M144, 145
- M154, 155
- M201 - 210
- M254

Note: In the M code column, modal state of up to 26 M codes is displayed.

4. Table of Reserved Local Variable Words

ABS	DRAW	HA	NOEX	RS
AG	DROUND	HB	NOINC	RSQCO
AND	DS	HC	NOT	RSQRI
ARC	EIN	HORN	NP	RSQRO
ATAN	EMPTY	HS	OMIT	RSTRT
ATAN2	EOR	IF	OR	RT
BCD	EQ	KA	PCIR	RTI
BHC	FA	KB	PMIL	RTMCR
BOUNS	FC	LCONI	PN	RTMDI
CALL	FI	LCONO	PREG	SAVE
CAMCV	FIX	LE	PRINT	SC
CHFC	FMILF	LMV	PRMDI	SIN
CHFR	FMILR	LMW	PRMDO	SPHRI
CLEAR	FOPENA	LPRINT	PRSOL	SPHRO
CLOSE	FOPENB	LT	PUT	SPRINT
CONEI	FUP	MITCAN	QC	SQRT
CONEO	FWRITC	MOBOR	QD	SQRX
COPY	GE	MOD	RCORO	SQRY
COPYE	GET	MODIN	RCIRO	SS
COS	GOTO	MODOUT	READ	TAN
DA	GRCI	MODRL	RGEND	TAPR
DB	GRCV	MODSF	RH	TLCO
DC	GRDX	MOEND	RI	TLFOFF
DEF	GRDY	MOPLD	RLDPN	TLFON
DELETE	GRER	MOREM	RM	TLFR
DFIX	GROF	MOTAP	RMILI	TLOK
DFUP	GRON	MSG	RMILO	TN
DGRDX	GRSK	NCYL	ROUND	WP
DGRDY	GRST	NE	RP	WRITE
DIN	GT	NMSG	RQNFL	

5. Table of System Variables

System Variable	Format	Setting Range	Subscript	Read / Write	Inch / mm Conversion
Zero offset data	VZOF* [expression]	0 to ±99999.999	Work coordinate system number	R / W	Automatically converted
Tool length offset data	VTOFH [expression]	0 to ±999.999	Tool length offset number	R / W	Automatically converted
Cutter diameter compensation data	VTOFD [expression]	0 to ±999.999	Cutter radius compensation number	R / W	Automatically converted
Travel end value (positive direction) (User parameter)	VPSL*	0 to ±99999.999		R / W	Automatically converted
Travel end value (negative direction) (User parameter)	VNSL*	0 to ±99999.999		R / W	Automatically converted
Backlash compensation data	VBLC*	0 to 1.000		R / W	Automatically converted
In-position width	VINP*	0 to 1.000		R / W	Automatically converted
In-position width for home position	VHPI*	0 to 1.000		R / W	Automatically converted
Zero point of machine coordinate system	VMOF*	0 to ±99999.999		R / W	Automatically converted
Programmable travel end (positive direction) (User parameter)	VPPL*	0 to ±99999.999		R / W	Automatically converted
Programmable travel end (negative direction) (User parameter)	VNPL*	0 to ±99999.999		R / W	Automatically converted
Position of home position	VHPP* [expression]	0 to ±99999.999	1 to 32	R / W	Automatically converted
Sensor contact point data	VSAP*	0 to ±99999.999		R / W	Automatically converted
For communications for automating equipment	VNCOM [expression]	Binary 8 bits (1 byte)	1 to 4	R / W	Not changed
Calculated value	VRCO*			Read only	Automatically converted
Actual position data	VAPA*			Read only	Automatically converted
Active coordinate system number	VACOD			Read only	Not changed
Actual tool number	VATOL			Read only	Not changed
Next tool number	VNTOL			Read only	Not changed
Number of coordinate systems, tool data sets (NC specification code 2)	VSPCO			Read only	Not changed

System Variable	Format	Setting Range	Subscript	Read / Write	Inch / mm Conversion
Specification code for subprograms (NC specification code No. 24)	VSPSB			Read only	Not changed
Machine lock	VMLOK			Read only	Not changed
Printer control	VPCNT	Binary 8 bits (1 byte)		R / W	Not changed
Automating specification judgment result 1	VOK1	Binary 8 bits (1 byte)		R / W	Not changed
Automating specification judgment result 2	VOK2	Binary 8 bits (1 byte)		R / W	Not changed
Gauging number	VNUM	0 to 9999		R / W	Not changed
Printer control	VINTG	0 to ± 99999.999		R / W	Not changed
Printer control	VPRT	Binary 8 bits (1 byte)		R / W	Not changed
Actual tool number	VTLCN	0 to 65535		R / W	Not changed
Next tool number	VTLNN	0 to 65535		R / W	Not changed
Tool length / tool breakage detection selection flag	VFST	Binary 8 bits (1 byte)		R / W	Not changed
Feedrate clamp value (NC optional parameter (long word) No. 10)	VFDMX	0 to 20000		Read only	Not changed
Graphic color designation	VGCLR	0 to 9		R / W	Not changed
Program unit system	VINCH	Binary 8 bits (1 byte)		R / W	Not changed
Sequence return flag	VRSTT	Binary 8 bits (1 byte)		Read only	Not changed
Hour meter (run hour)	VDTIM [a, b]		a: 1 to 5 b: 1 to 2	Read only	Not changed
Work counter	VWRK [a, b]		a: 1 to 4 b: 1 to 2	Read only	Not changed
Synchronized tapping torque monitoring parameter number	VTMNO	1 to 5		R / W	Not changed
Synchronized tapping monitoring parameter	VTMDT [expression]		1 to 5	Read only	Not changed
Spindle overload monitoring parameter number	VSLNO	1 to 5		R / W	Not changed
Spindle override monitoring parameter	VSLDT [a, b]		a: 1 to 5	Read only	Not changed 207
			b: 1 to 3	R / W	Automatically converted
F1-digit parameter set feedrate	VPF1F [expression]	0 to 4000.000	1 to 9	R / W	Automatically converted
F1-digit max. parameter setting value	VPF1M [expression]	0 to 4000.000	1 to 9	R / W	Not changed
F1-digit parameter acceleration/ deceleration data	VPF1C	0 to 4000000		Read only	Not changed

System Variable	Format	Setting Range	Subscript	Read / Write	Inch / mm Conversion
Executing G code	VGCOD [expression]		1 to 96	Read only	Not changed
Executing M code	VMCOD [expression]		1 to 64	Read only	Not changed
Executing S code	VSCOD			Read only	Not changed
Executing F code	VFCOD			Read only	Not changed
Executing D code	VDCOD			Read only	Not changed
Executing H code	VHCOD			Read only	Not changed
Mirror image status	VMRI			Read only	Not changed
Minute counter for hours	VPMNT			Read only	Not changed
Minute counter	VRMNT			Read only	Not changed
Hour counter for date	VPHUR			Read only	Not changed
Hour counter	VRHUR			Read only	Not changed
Day counter	VQDAT			Read only	Not changed
Manual shift amount	VMSF*			Read only	Automatically converted
Number of ATC tools	VSPTN			Read only	Not changed
PPC parameter	VPPCP			Read only	Not changed
PPC pallet number	VPLNO			Read only	Not changed
PPC parameter bit data	VPLDT [expression]		1 to 12	Read only	Not changed
Crossrail automatic positioning M code	VECWM			Read only	Not changed
Actual attachment number	VATNT			Read only	Not changed
Tool management data Tool group number	VTLD1 [expression]	0 to 255	Tool management number	R / W	Not changed
Tool management data Tool life management mode	VTLD2 [expression]	0 to 9	Tool management number	R / W	Not changed
Tool management data OK / NG flag	VTLD3 [expression]	0 to 255	Tool management number	R / W	Not changed

System Variable	Format	Setting Range	Subscript	Read / Write	Inch / mm Conversion
Tool management data Tool life flag	VTLD4 [expression]	0 to 255	Tool management number	R / W	Not changed
Tool management data Second tool offset number	VTLD5 [expression]	0 to 320	Tool management number	R / W	Not changed
Tool management data Third tool offset number	VTLD6 [expression]	0 to 320	Tool management number	R / W	Not changed
Tool management data Set tool life data	VTLD7 [expression]	0 to 32767	Tool management number	R / W	Not changed
Tool management data Tool life count data	VTLD8 [expression]	-32768 to 32767	Tool management number	R / W	Not changed
MOP tool number	VMPT	0 - 9		R / W	Not changed
MOP control 1	VMPC1	Binary 8 bits (1 byte)		R / W	Not changed
MOP control 2	VMPC2	Binary 8 bits (1 byte)		R / W	Not changed
MOP control 3	VMPC3	Binary 8 bits (1 byte)		R / W	Not changed
MOP control 4	VMPC4	Binary 8 bits (1 byte)		R / W	Not changed
MOP control 5	VMPC5	Binary 8 bits (1 byte)		R / W	Not changed
I / O data (input)	VIRD [expression]		Identifier	Read only	Not changed
I / O data (output)	VORD [expression]		Identifier	Read only	Not changed
Pallet type in the front of machine	VPLTK	0: None 1: L (A) 2: R (B)		Read only	Not changed
Optional parameter bit	VOPRB	0 to 255	1 to 64	R / W	Not changed
Program branch 1	VPBR1			Read only	Not changed
Program branch 2	VPBR2			Read only	Not changed
Date, time	VTIME [expression]		1 to 6	Read only	Not changed
Tool No.	VPTNO		Pot No.	Read only	Not changed
Pot No.	VTPNO		Tool No.	Read only	Not changed
Attachment spindle; Overload monitor parameter	VSLNB	1 to 5		R / W	Not changed
Attachment spindle; Overload monitor parameter	VSLDB [a, b]		a = 1 to 5 b = 1 to 3	Read only	Not changed

System Variable	Format	Setting Range	Subscript	Read / Write	Inch / mm Conversion
Attachment spindle; Synchronized tapping torque monitor parameter No.	VTMNB	1 to 5		R / W	Not changed
Attachment spindle; Synchronized tapping torque monitor parameter	VTMDB [expression]	1 to 127	1 to 5	Read only	Not changed
BC-axis attachment vertical angle	VUNVB	0 to 3600		Read only	Not changed
BC-axis attachment horizontal angle	VUNVC	0 to 3600		Read only	Not changed
Tool length wear amount	VTWOH	0 to ± 999.999	Tool length offset No.	R / W	Automatically converted
Cutter radius wear amount	VTWOD	0 to ± 999.999	Cutter radius compensation No.	R / W	Automatically converted
Work coordinate system position A	VWKA*			Read only	Automatically converted
Work coordinate system position B	VWKB*			Read only	Automatically converted
Tool length shape compensation	VTFGH	0 to ± 999.999	Tool length offset No.	R / W	Automatically converted
Cutter radius shape compensation amount	VTFGD	0 to ± 999.999	Cutter radius compensation No.	R / W	Automatically converted

*: Represents an axis name such as X, Y and Z

LIST OF PUBLICATIONS

Publication No.	Date	Edition
5228-E	April 2005	1st
5228-E-R1	April 2006	2nd
5228-E-R2	February 2007	3rd
5228-E-R3	April 2007	4th
5228-E-R4	August 2007	5th
5228-E-R5	January 2008	6th
5228-E-R6	June 2008	7th
5228-E-R7	August 2008	8th
5228-E-R8	April 2009	9th
5228-E-R9	October 2010	10th

This manual may be at variance with the actual product due to specification or design changes.

Please also note that specifications are subject to change without notice.

If you require clarification or further explanation of any point in this manual, please contact your OKUMA representative.