Manix CNC Machine

NC-310 User's Manual

MANIX

Table of Contents

Chapter 1 NC310 (Mini NC lathe)

1.1 Characteristics

The NC310 product is a mini precision NC the manufactured to process small precision metal parts.

The main spindle is penetrated so that the cross-section and tip can be processable and products in the long axis can be processable using the center in the tailstock spindle.

A variety of products can be processable using a turret tool post (optional) for complex processed products.

It has also a specialized cover to prevent scatter, noise, and safety accident of coolant.

It can improve space utilization, economics, and productivity, and enables fast response with less cost during maintenance in the future than other general universal NC equipment.
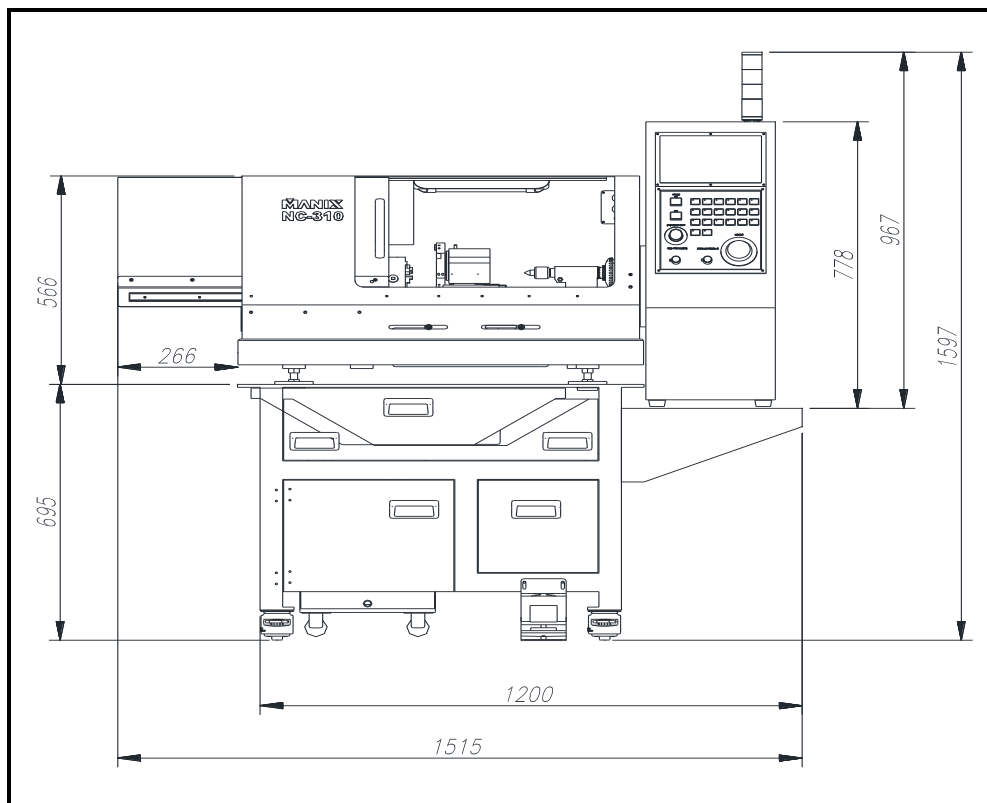
Now, you can produce precise products reliably which were only available for high skill technologies.

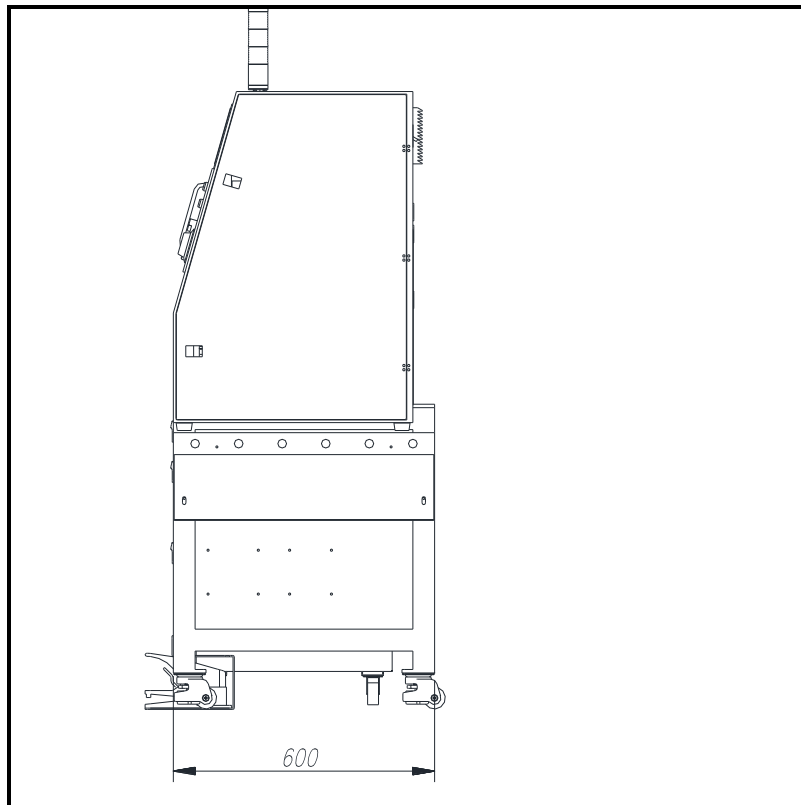It can be utilized for not only industrial but also educational purposes.

We will guarantee a fast supply of customized products through consultation with customers.
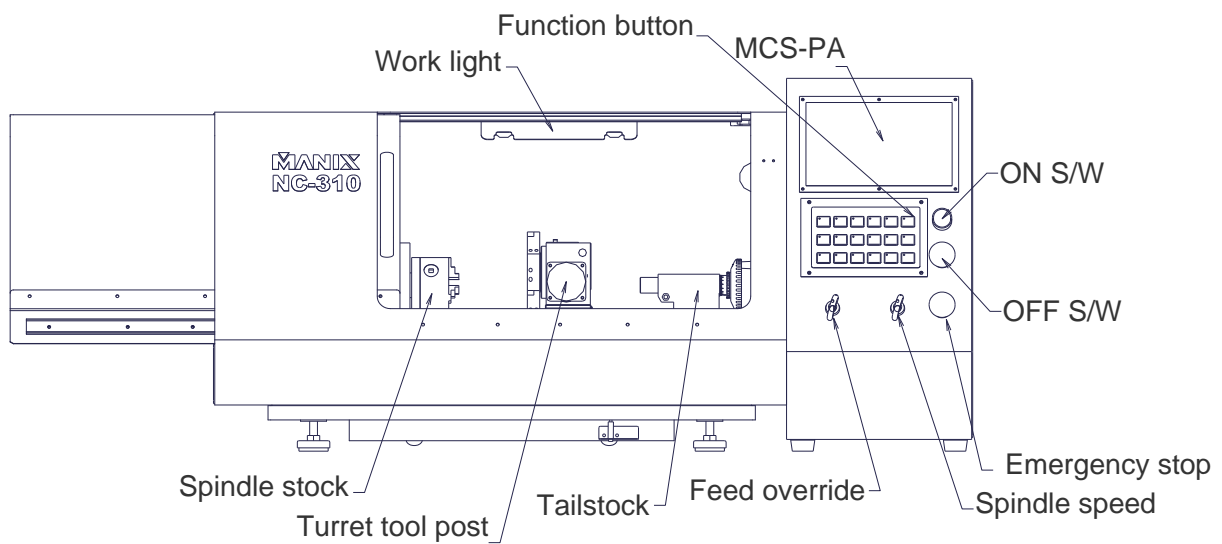
1.2 Machine dimension drawings

1.2.1 Front view

1.2.2 Side view

1.2.3 Names of the parts

1.2.4 Basic components and option

1) Basic components
- 3-inch self-centering chuck
- Live center
- 4-way tool post
- Tool post

2) Optional components
- Pneumatic chuck
- Collet chuck
- Turret tool post
- Bite
- Coolant
- Oil mist

1.2.5 Specification of the main body

1) Specifications
- Swing over bed: 80mm in case of reverse jaw
- Swing over carriage: MAX 70mm
- Distance between swindle and tailstock: 360mm
- Maximum hole through: 16mm
- Maximum turning diameter: 200mm

2) Feed
- X-axis feed rate: 55mm
- Z-axis feed rate: 250mm

3) Main spindle
- Main spindle speed: Max 300rpm
- Spindle speed command: 0–3000rpm
- Main spindle taper type: MT#2

4) Tool post
- Tool post: 4-way
- No of tools: standard 4 EA (optional □ x3EA, o x 3EA)
- Dimension of each bite: □ 8mm
- Boring bite shank diameter: 12mm
- Tool exchange time: 4 sec

5) Feed rate
- Rapid feed rate: 3000mm/min

6) Tailstock
- Feed rate of the tailstock spindle: 45mm
- Diameter of the tailstock spindle: 30mm
- Tailstock taper type: MT#2
- Feed rate of the tailstock: 250mm

7) Main spindle and feed motor
- Main spindle: 750W servo motor
- Feed motor: 100W servo motor

8) Feed screw
- X and Z axes cutting ball screw

9) Power Supply
- Single phase 220V, 10A

10) Machine dimension
- W x D x H: 1515 X 600 x 1600(H)
- Weight: abt 220Kg

1.2.6 Electricity and specification of NC device

1) Control axis
- No. of control axes: two axes (X and Z)
- No. of simultaneous control axes: two axes
  (positioning, linear interpolation, circular interpolation)

2) Feed
- Minimum setup unit: 0.001mm
- Minimum feed unit: 0.001mm
- Decimal point input

3) Interpolation
- Positioning: G00
- Linear interpolation: G01
- Circular interpolation: G02, G03

4) Feed
- Cutting feed rate: G94 (mm/min)
- Dwell: G04
- Handle feed: manual pulse generator (X1, X10, X100)
- Automatic acceleration deceleration: at rapid feed and cutting feed (mm/min)
- Rapid and cutting speed override: 150%

5) Program save
- NC program: Max 68 EA, 255 Byte
- Program: deletion, edition, and change

6) Operation and indication
- Operation panel indication: 70mm x 30mm
- Display function: current position, absolute position, relative position, remaining position
- MDI function

7) Input and output
- RS-232C
- Dedicated computer I/O software (MSW-MCS)

8) S/T/M function
- Main spindle speed: M code
- Tool function: T code
- Auxiliary function: M code

9) Tool compensation
- Tool position off-set: T function

10) Coordinate system
- Manual return to reference position
- Automatic return to reference position

11) Other functions
- Single step function
- Optional stop function
- Backlash compensation function

12) Safety function
- Emergency stop
- Over-travel
- Alarm description saving function

Transfer software installation method

This is enclosed with the product. The transfer software in the CD is installed. (MSW_MCS.Exe)
Please follow the instruction in the installation screen.

RS232C cable is connected from NC310 to the PC
- Basically, PC should be equipped with RS232C port for the connection.
  If a conversion adapter from USB to RS232C is used, it can be a source of trouble.
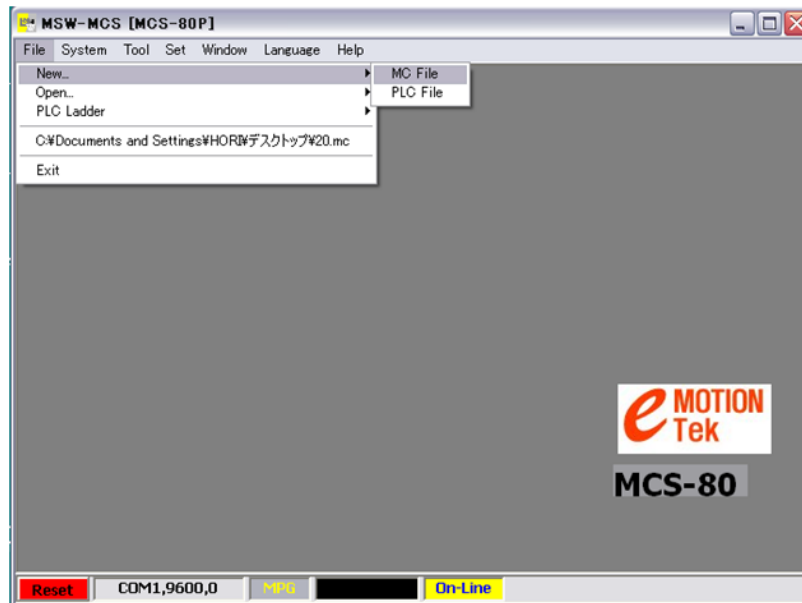
Please turn on the power of NC310.

Double click MSW-MCS MMI to start.
Once the communication is successful, the following screen is displayed.
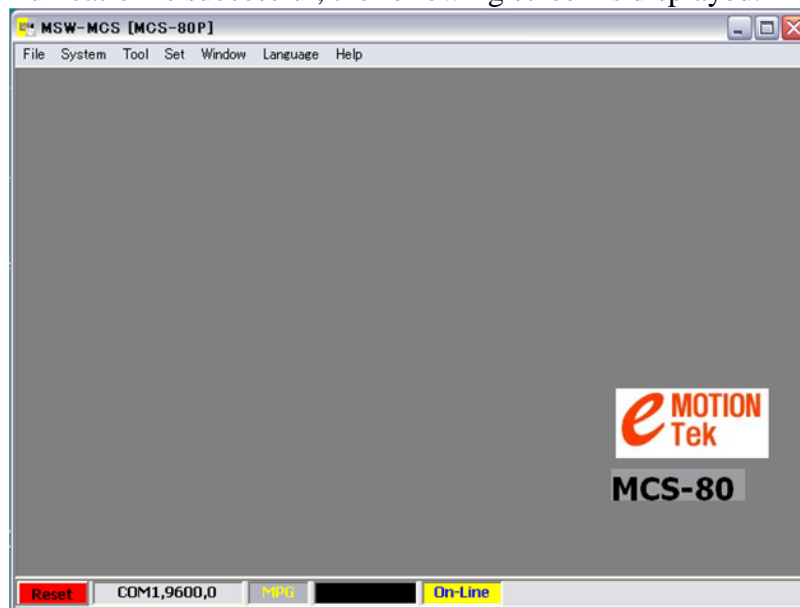
① Push the File--New--MC File.

② Create G code.
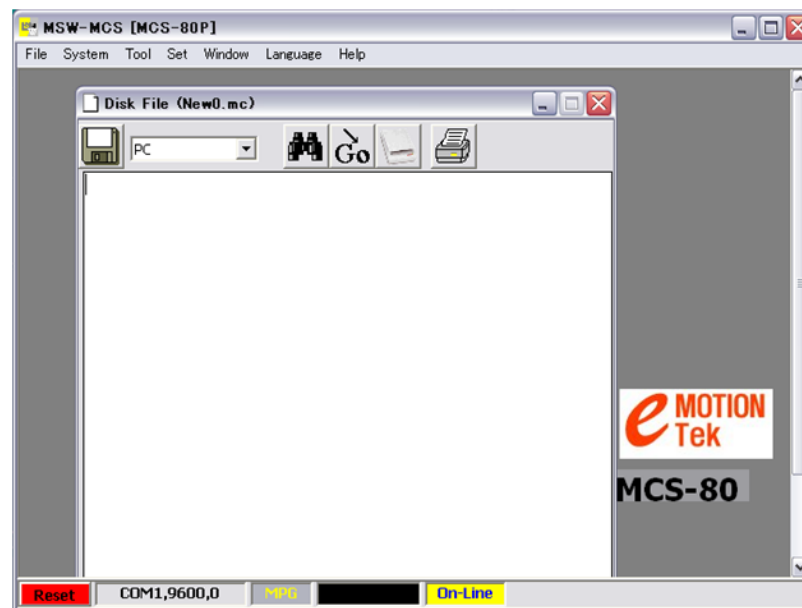
③ The following items are changed from PC to MAC.
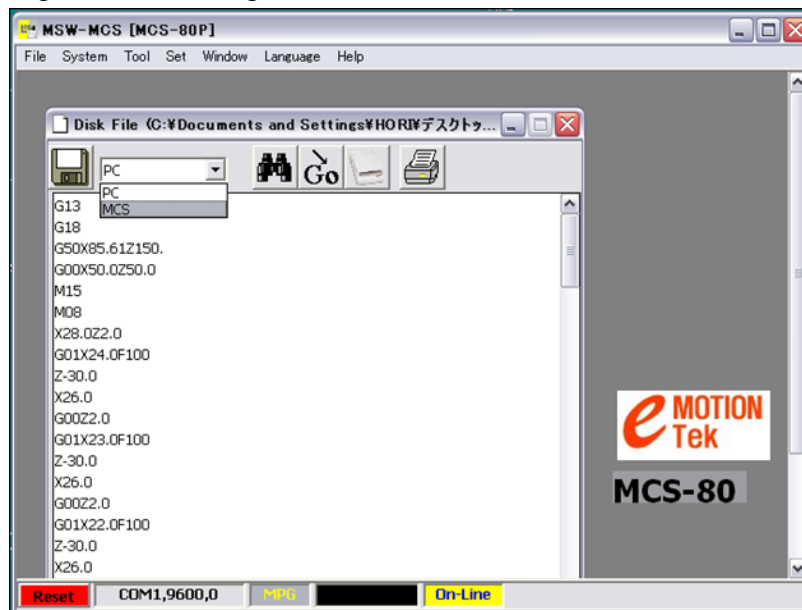
④ File NO.（1–71）are inputted and saved.
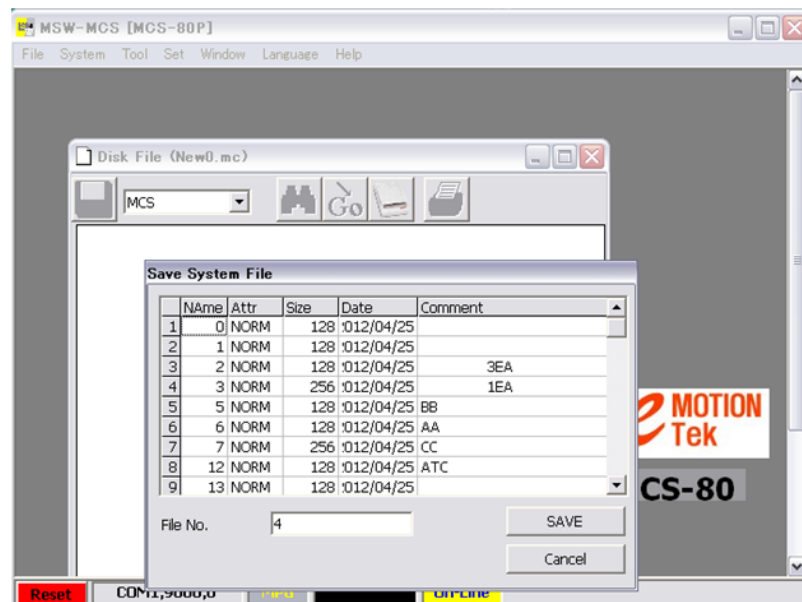
⑤ Now new program is transferred to NC310.

① Push File--Open--MCFile.
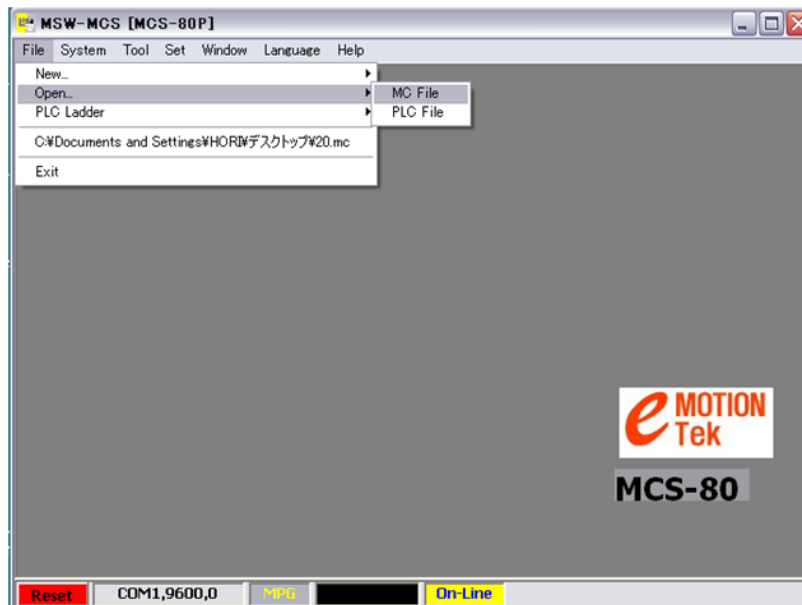
② Open the data folder and specify MC data
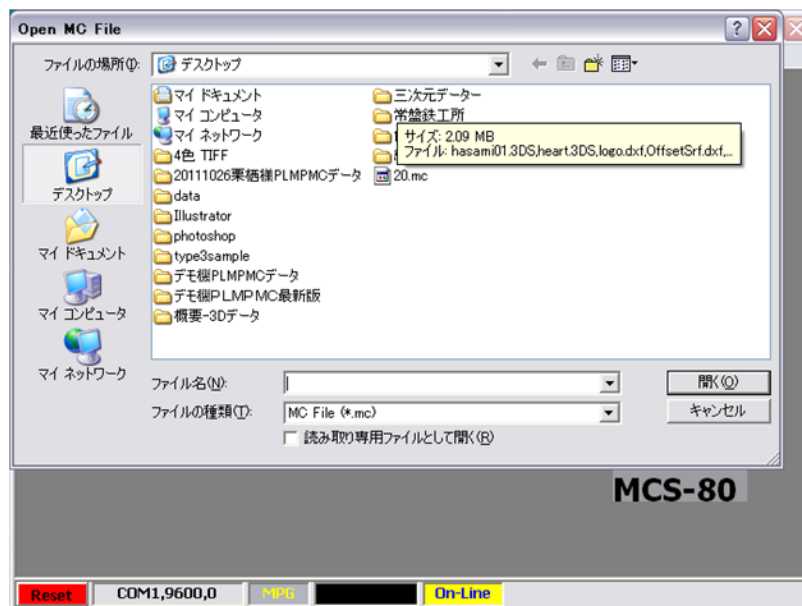
③ While the program is displayed, the following items are changed to MCS.
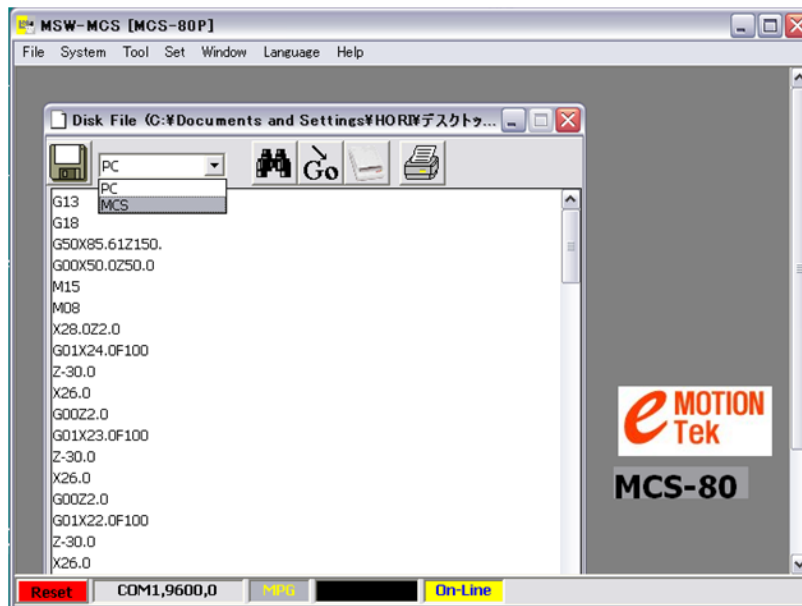
④ File NO.（1–71）are inputted and saved.
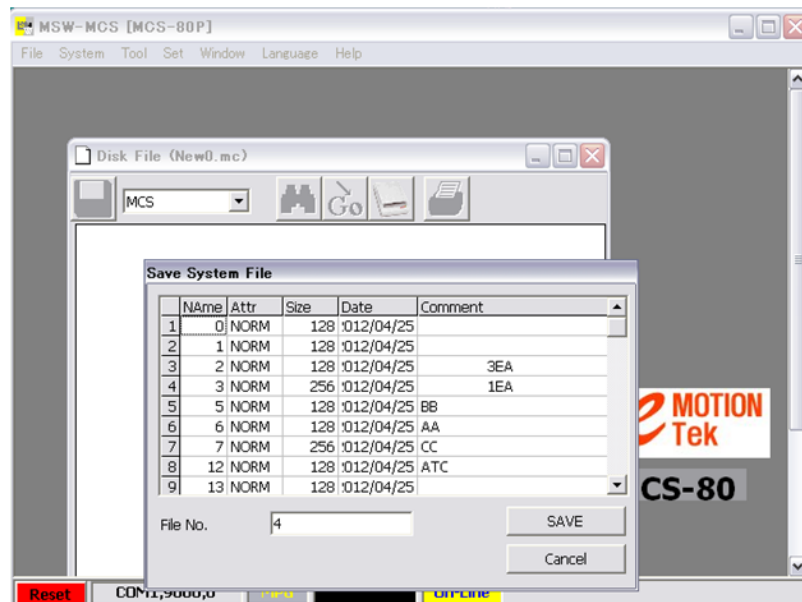
⑤ Now, transfer to NC310 is complete.

① Push System--Open--MC Program
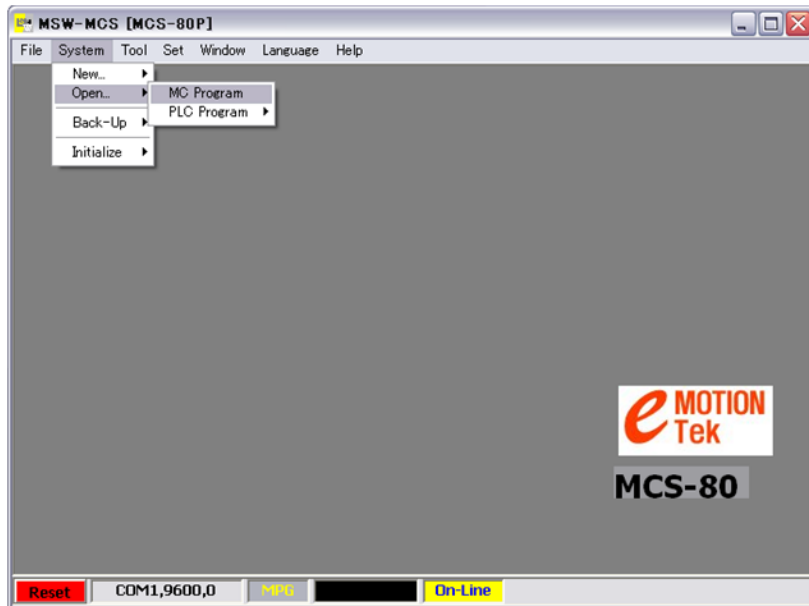
② Once the program list of the controller is displayed, specify the program and open the program.
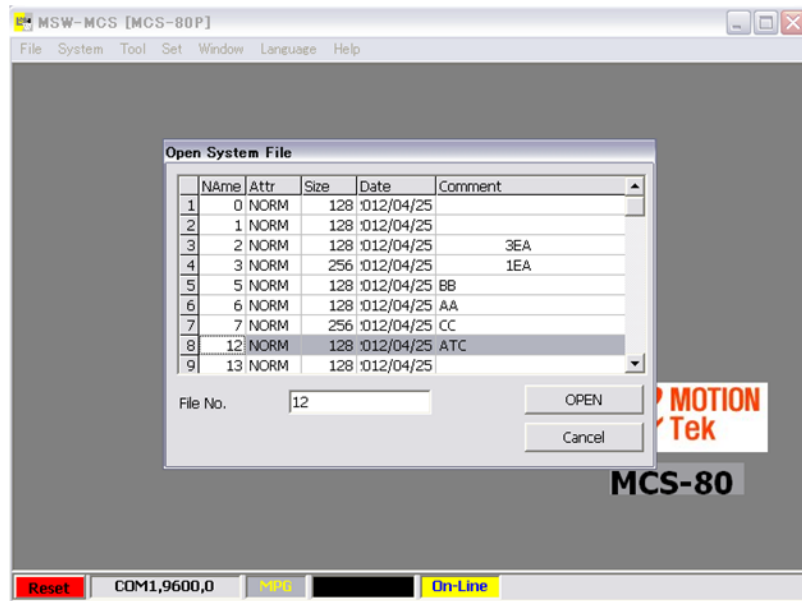
③ The following items are changed to PC and specify the folder to be saved and save them.
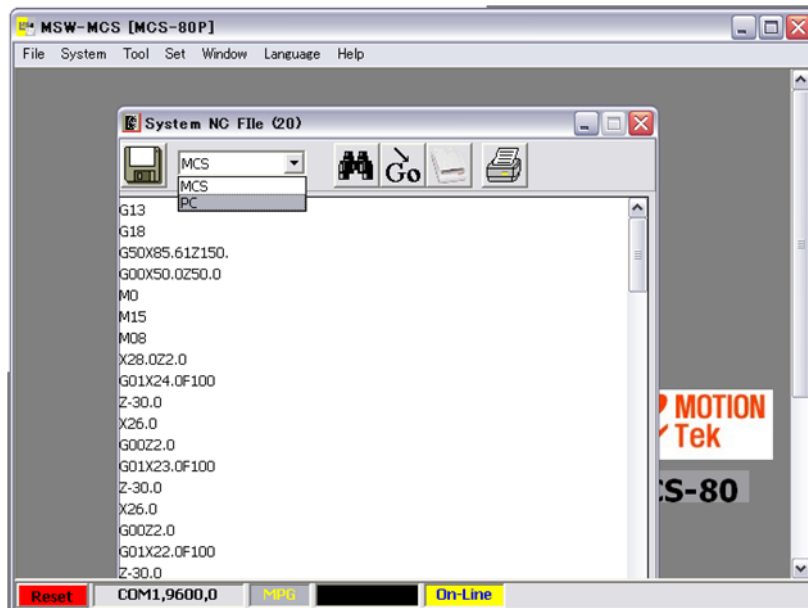
How to clear alarm

There is a limit switch in the + and - directions in the X and Y axes.
If the limit of each axis is over, it moves only in a single direction.

① Move to the opposite direction to the alarm generated direction.

② Push the ESC or RESET button.

③ Alarm is clear.

Other alarms are also clear using the above method.

How to create a new program

① Set to EDIT mode.

② Push F2 button.

③ Push 1 button. (Select 1.MC)

④ Push F4.(Select NEW)

⑤ Put program No. (From 01 to 71)

⑥ Program input

⑦ Push ESC.

⑧ When asked Update?(ENT/ESC), push ENTER to save.

How to change a process program

① Set to EDIT mode.

② Push F2 button.

③ Push 4 button.（Select RnPG）

④ Using the arrow key, set the cursor to the process program No. to be called and push the ENTER key.

⑤ The specified program is loaded.

How to change a program

① Set to EDIT mode.

② Push F2 button.

③ Push 1. (select 1.MC）

④ Using the arrow key, set the cursor to the program No. to be changed and push the ENTER key.

⑤ Modify the program.

⑥ Once the program modification is complete, press the ESC button.

⑦ When asked Update?(ENT/ESC), push ENTER to save.

Chapter 2 Programming

Program Configuration and Memory Capacity

Program	Program Storage Type	Category	Remark
MC Program	Registering maximal 68 MC programs		Total Capacity: 255K bytes
PLC Program	Interrupt PLC Program: Maximal 1000 steps		
	Scan PLC Program: Maximal 1000 steps		

In order to use data, the data are memorized in the flash memory that is the non-volatile element.

* MC Program Commands

* Overview

The motion control is concerned with moving a machine from an arbitrary position to the desired position at the designated speed or controlling the input and output. The control axis is operated variously depending on several control environments. The commands used for the MC program controlling the motion are called the program commands.

The MC program is executed in the AUTO/MDI mode after it is prepared and saved.

Each command and function used in the program is summarized as follows.

N: Statement number
G: G code
M: M code
X,Y,Z,U,V,W,A, and B: Coordinate order address
XX,YY,ZZ,UU,VV,WW,AA,BB, and CC: Increment coordinate order
S and SX: Spindle speed order
D: For designating the dwell time
F: For designating the movement time
FX,FY,FZ,FU,FV,FW,FA,FB, and FC: The rapid movement speed of each axis
L: For the variable L (variable)
Q: For the Q variable (contact point)
P: The point number, the program number, and the dwell time
IF, GO TO, WHILE DO, and END: Refer to the explanation of functions
The Other Alphabets: Special codes
    (The special codes mean the codes made for the functions our company develops and supplies
     to enable the functions of the various exclusive machines produced by machine manufacturers.

    Contact our company if you want the details on special codes or need the special functions.)

* Fundamentals of Programming

* Position Movement

Linear Movement I: G01 Y···;
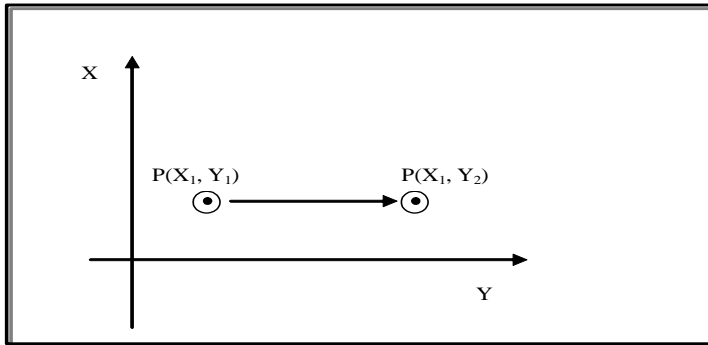
Linear Movement II (Linear Interpolation): G01 X... Y....;
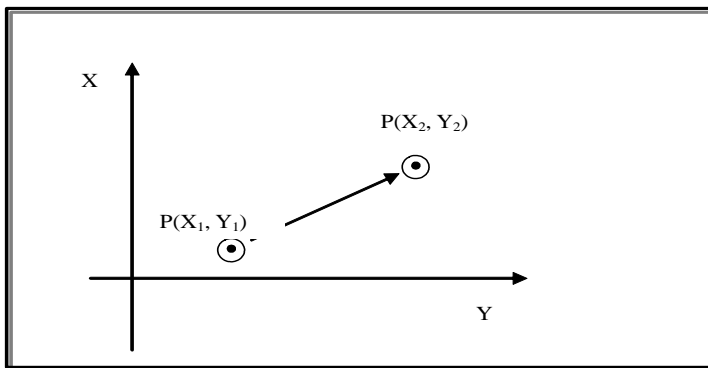
Arc Movement (Arc Interpolation):   G02(X... Y....R....; or   G03 X... Y....R....;
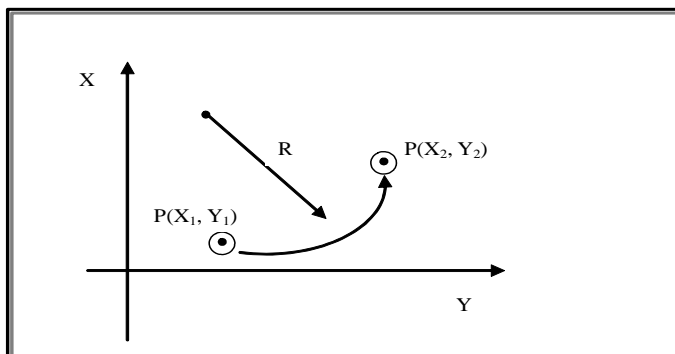
* Movement Speed (F: mm/min)

In case of the movement from an arbitrary position to a desired position, the movement speed can be ordered in the program with the F command.

For example, state F2.0 in the program to move it at the 2mm/min speed.

If there is no other statement, the unit is mm/min.

Relevant Parameters: P33 (maximal interpolation speed/clamp), P34 (standard interpolation speed), and P42 (acceleration or deceleration during interpolation)

* Program Configuration

The MC program consists of the various commands for motion control. The set of the commands executed in the program is called a block.

* Standard Program Configuration

```
                        ┌─────────────┐
                        │  BLOCK #1   │
                        ├─────────────┤
                        │  BLOCK #2   │
        ┌─────────┐     ├─────────────┤          The execution order of the co
        │ Program │     │  BLOCK #3   │          ntroller
        └─────────┘     ├─────────────┤
                        │  BLOCK #4   │
                        └─────────────┘
                        ┌─────────────┐
                        │  BLOCK #n   │
                        └─────────────┘
```

### * Block Configuration

```
   N OOOO      G00    X OO . O   Z O O . O   M O O    EOB
```

Sequence Number/ Preparation Function/ Position Data/ Secondary   Function/   Block End
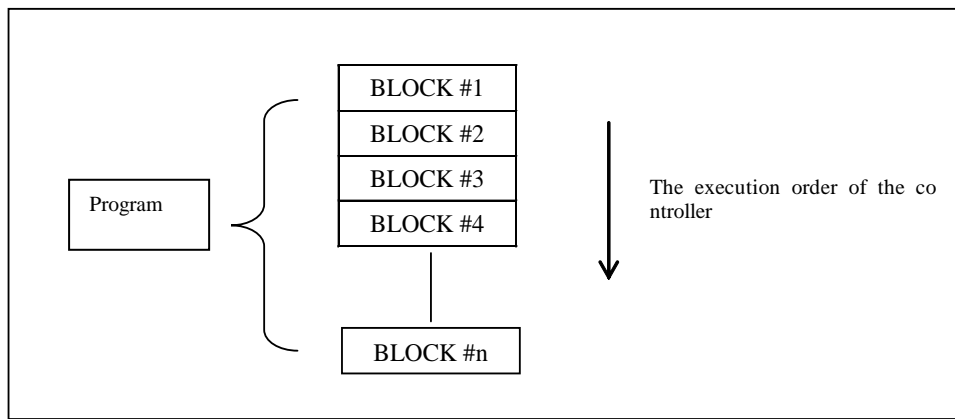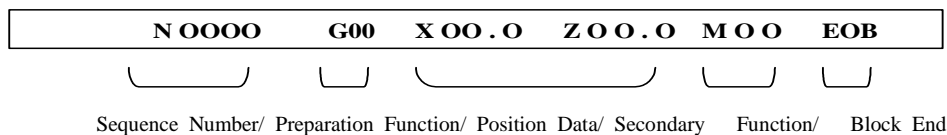
A block contains the sequence number (optional) representing the MC operation order in the front and the EOB representing the block end in the rear.

However, the integer position data are a desk electronic calculator type. Example) 15 is recognized as 15mm or 15000um.

Note) Processing the Comments

/*: The line whose letters in the left are declared with /* is processed as a comment.

2.1 Command Function (G Function)

The number following the address G defines the meaning of the block command.

There are two types in the G code as follows.

- One Shot G Code: The G code has the meaning only in the block ordered.

- Modal G Code: The G code is effective until another G code in the same group appears.

  Example) G01 and G00 are modal G codes.

    G01 X___;
    Y___;   } G01 is effective in the section.
    X___;
    G00 Y___;

Table 2.1 Standard G Code List

Group	Code	Function
1	G00 *	Position determination
	G01 *	Linear interpolation
	G02	CW arc interpolation
	G03	CCW arc interpolation
0	G04	Dwell
3	G17 **	Determination of the XY plane during the arc interpolation
	G18	Determination of the ZX plane during the arc interpolation
	G19	Determination of the YZ plane during the arc interpolation
0	G28	Return to the starting point
	G50	Setting of the absolute coordinate system
	G51	Machine coordinate system setting #1
	G65	Subprogram calling
2	G90 *	Absolute order
	G91 *	Incremental order

Note)
- The G code with the symbol of ** represents that it becomes the G code state when power is supplied.

   The G code with the symbol of * represents that the G code state is determined with the parameter after power is supplied or after it is reset. (It is done with parameters P22.0 and P22.1.)

- The G code in the group 0 is not modal so that only the ordered block is effective.

- If the G code that is not in the G code list is ordered, an alarm is sounded.

- Several G codes of different groups can be ordered for the same block.

However, the integer position data are a desk electronic calculator type. Example) 15 is recognized as 15mm or 15000um.

Table 2.2 The Expanded G Code List

Group	Code		Function
01	G05		Arc interpolation (intermediate coordinate order)
	G06		Circle interpolation (intermediate coordinate order)
	G07		
	G08		
00	G09		Inposition check function
	G10		The function of pre-executing the block being next to the XYZ axis (on the basis of the ratio)
	G11		The function of pre-executing the block being next to the XYZ axis (on the basis of the distance)
	G12		Setting (V9.29.1 or lower) or releasing (V9.29.1 or higher) the acceleration or deceleration non-execution mo
	G13		Releasing (V9.29.1 or lower) or setting (V9.29.1 or higher) the acceleration or deceleration non-execution mod
	G14		Movement after the output (on) of the set contact point
	G15		Output (on) of the set contact point after movement
	G16		Output (off) of the set contact point after the movement after the output (on) of the set contact point
	G20		Dwell with the encoder (simple)
	G21		Dwell with the external encoder (accumulated)
	G22		The function of the dwell based on an external pulse (accumulated/high speed check)
	G24		Movement after the output (off) of the set contact point
	G25		Output (off) of the set contact point after movement
	G26		Output (on) of the set contact point after the movement after the output (off) of the set contact point
	G27		
	G29		Rapid movement on the basis of the machine coordinate system
	G30		Return to the 2nd starting point
01	G31		Skip movement
	G32		
	G33		
	G34		
00	G35		The function of pre-executing the block being next to the previous axis (on the basis of the ratio)
	G36		The function of pre-executing the block being next to the previous axis (on the basis of the distance)
00	G37		
00	G38		The synchronized operation function based on the external pulse of a designated axis
00	G39		Release of the synchronized operation based on the external pulse of a designated axis
04	G40		The release of the normal control mode and the setting of the normal axis coordinate
	G41		Setting of the normal control mode/setting of the normal axis (left)/designation of the speed during the normal-axis single movement
	G42		Setting of the normal control mode/setting of the normal axis (right)/designation of the speed during the normal-axis single movement
	G43		Designation of the axis excluded from the speed calculation during interpolation
	G44		
	G45		
	G46		
	G47		
	G48		
	G49		
00	G52		Machine coordinate system setting #2
	G53		
	G54		
	G55		
	G56		
	G57		
	G58		
	G59		

	G60		
	G61		

	G62		
	G63		
	G64		
	G65		
	G66		Subroutine call
	G67		
	G68		
	G69		
00	G70		Palletizing movement
	G71		Palletizing counter clear
	G72		
	G73		
	G74		
	G75		
	G76		
	G77		
	G78		
	G79		
00	G80		Waiting until the corresponding contact point is 'ON'(checking the X.M contact point)
	G81		Waiting until the corresponding contact point is 'OFF'(checking the X.M contact point)
	G82		Waiting until the corresponding contact point is 'ON'- High speed check (checking the X contact point)
	G83		Waiting until the corresponding contact point is 'OFF'- High speed check (checking the X contact point)
	G84		
	G85		
	G86		
	G87		
	G88		
	G89		
	G90		
	G91		
	G92		
	G93		
	G94		
00	G95		The function of the speed order based on the long axis
	G96		
	G97		
	G98		
	G99		

M Code	M 00		Program stop
	M 02,M30		PROGRAM END
	M 98		Subprogram call
	M 99		Subprogram call end
Position Order	X,Y,Z,U,V,W, A,B,C		Absolute position order (in case of G90 declaration) Incremental position order (in case of G91 declaration)
	XX,YY,ZZ,UU,VV, WW,AA,BB,CC		Incremental position order

Spindle Speed	S,SX	The spindle and the spindle #1 order
Speed Order	F, FX,FY··· FV , FF	Movement speed order (mm/Min)
Dwell Time	D	Designation of the dwell time
Repeat Statement	WHILE DO END	
Variable	L , LT	L Variable (32bit float variable) function (L0~L7999)
	Q , QT	Q Variable (Contact Point/Word Data) Function: Processing with the bit unit
The Other Commands	N	Designation of a statement number
	P(M98/G65)	Designation of a subprogram call number
	P	The position point coordinate, the stop time, and the program number
	IF,IFT	The conditional branch command and the conditional calculation
	GOTO,GOTO1	Branch command
	/*	Comment processing command
	<= ,>=, =, < , >, <>	Comparison command
	+ , -, *, /, A , O , ^, !,%	Calculation command
	= , .EQ	Equal to
	<=, .LE	Less than or equal to
	<, .LT	Less than
	>=, .GE	Greater than or equal to
	>, .GT	Greater than
	<>, .NE	Not equal to
	SIN,COS,TAN, ASIN,ACOS,ATAN	Trigonometric function
	ABS	Absolute value
	SQRT	√

1) Movement Function
(1) Position Determination (G00)

The position determination is ordered with G00. The table is moved very fast to the target order positi on of the work coordinate system in case of the absolute order (G90) and to the incremental position being away from the current position by the order value in case of the incremental order (G91).

　　G00 IP__ or G00 P__

　　　IP__; The absolute and incremental combination order can be given in the absolute order mode of the arbitrary axis addresses such as X_ Y_ Z_ U_ V_ W_ A_ B_ C_ and th e coordinate values (the incremental movement in the absolute mode) such as XX_ YY_ ZZ_ UU_ VV_ WW_ AA_ BB_ CC_.
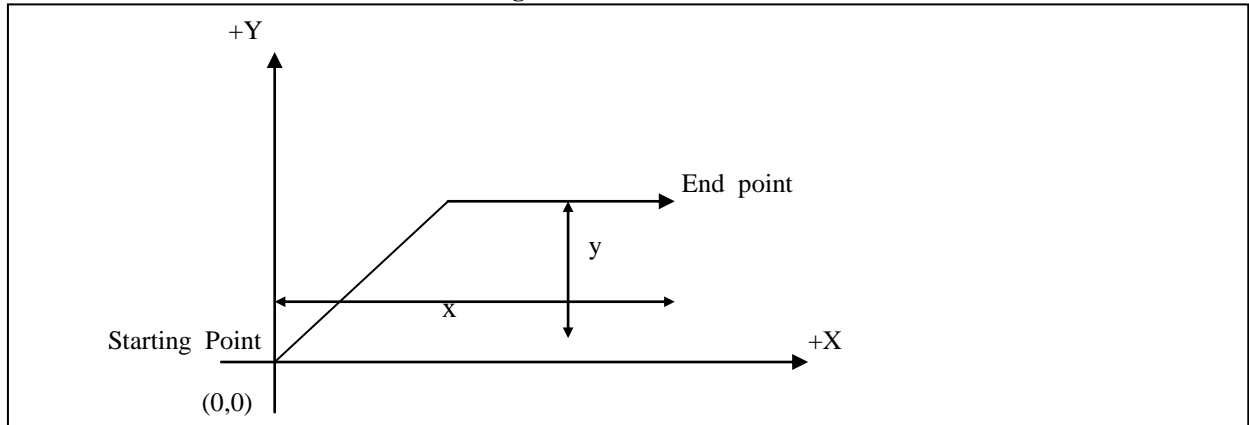　　　P__; If the address of the L variable is ordered, the setting value of the corresponding L variab le reflecting the number of set axes from the ordered address is considered as the coordin ate value of each axis.

The table movement path is independent of each axis. The position is determined very fast.
The table movement path is not usually linear.

Examples of Usage
1) 　G90　　　　　　　　　　　　-- The absolute movement mode 　G00 X10.0 Y20.0 Z30　　--- If the axis address and coordinate value set in the parameter are inputted directly 　　　　　　　　　　　　　　(Moved to the absolute position with 10.0 mm at the X axis, 20.00mm at the Y a xis, and 30.0mm at the Z axis) 　G00 X10.0 YY20.0 Z30　　--- If the absolute and incremental positions are used in a mixed way 　　　　　　　　　　　　　　(Moved to the absolute position with 10.0 mm at the X axis, 40.00mm at the Y a xis, and 30.0mm at the Z axis) 　G91　　　　　　　　　　　　--- The incremental mode 　G00 X10.0 Y20.0 Z30　　--- If the axis address and coordinate value set in the parameter are inputted directly 　　　　　　　　　　　　　　(Moved incrementally to the position with 10.0 mm at the X axis, 20.00mm at the Y axis, and 30.0mm at the Z axis)
2) G00 XX10.0 Y10.0　　---　The X-axis incremental and Y-axis absolute movement order (in case of the G90 m ode) -> 　　　　　　　　　　Same as G91 G00 X10.0 and G90 G00 Y10.0
3) G00 YL0;　　　　--- If the L Variable is used directly instead of a coordinate value 　　　　　　　　It moves to the value registered in ʻ0ʼ of the L Variable . Namely, if the L0 v alue is ʻ12345ʼ , the Y axis moves to 12.345.
4) G00 ZLL0;　　　　-- If the L Variable is used indirectly instead of a coordinate value 　　　　　　　　　It moves to the position of the L Variable with the value registered in ʻ0ʼ of the L Variable . 　　　　　Namely, if the L0 value is ʻ12ʼ , it moves to the value registered in L12.
5) G00 P0;　　　　-- If the previous axis is ordered directly with the start number of the L variable (must incl ude the G code) 　　　　　　　The values registered in ʻ0ʼ , ʻ1ʼ , and ʻ2ʼ of the L Variable become the coordina tes of the X, Y, and Z axes sequentially.
6) G00 PL0　　　　-- If the previous axis is ordered directly with the start number of the L variable (must inclu de the G code) 　　　　　　　Each axis moves to the position of the L variable from the value registered in ʻ0ʼ of the L variable. 　　　　　　　Namely, if the L0 value is ʻ12ʼ , the values registered in L12, L13, and L14 becom e the coordinates of the X, Y, and Z axes sequentially.
7) G00 XP0 FX1000　　--- Only the X axis of the P0 point is moved.

1) The Speed Change and Order Methods during the Fast Movement (G00)

Figure 2.1

A. In case of the G00 order, the fast movement speed is set in the **parameter (P32)** independently of each axis.

In the mode of the position determination with GOO, it is accelerated with the speed determined at the start of the block. In the end of the block, the following block is executed after the deceleration and stop.

B. FF__

The movement speed of each axis in the program can be designated as the variable with the above order.

If the address of the L Variable is ordered, the setting value of the corresponding L Variable reflecting the number of set axes from the ordered address is set to the rapid movement speed of each axis. (If four axes are used, four units are needed from the start L variable.)
Note) It cannot go beyond the "maximal interpolation speed" parameter.

C. FV__

The movement speed of each axis in the program can be ordered with the percent (0~100%) of the value set in the **parameter (P32)** by using the above order. (Examples: FV0: 0 %, FV10: 10%, FV100: 100%)

D. FX__ FY__... ; The speed of each axis can be ordered separately.

Examples of Usage
1) G90   G00 X1.0 Y2.0 ZZ3.0 - It moves at each axis speed set in the parameter (P32).

2)
   G00 X10.0 FF0   - The values registered in '0' , '1' , and '2' of the L variable becomes the speeds of the X, Y, and Z axes. Each axis speed set once is effective until it is reset.
   G00 YY10.0 FFL0   - The speed order using the L variable

3) G00 Z10.0 FV50   - It moves at the speed being equivalent to 50% of each axis speed set in the parameter (P32).
                      Each axis speed set once is effective until it is reset.
   G00 ZZ10.0 FVL50 - It moves at the speed of L50.

4) G00 X10 FX10   - The X axis moves at the speed of 10mm/min.   G00 X10 FXL10   - The X axis order at the speed of L10

(2) Linear Interpolation (G01)

G01 IP__F__;

Linear interpolation can be done with the order.

IP is the designation of the travel. Depending on the states of G90 and G91, the absolute or incremental order can be given.

For example, it moves linearly at the speed ordered as F to the order position of the selected work coordinate system in case of an absolute order and to the incremental position being away by the order value from the current position in case of an incremental order.

The movement speed ordered as F is effective until it is reset so that it doesn't have to give an order every time. (Modal)

Note) 1. If the value ending with '1' is set in the maximal interpolation speed parameter (P33), it is automatically calculated to have the resultant speed not to go beyond the rapid movement speed parameter (P32) of each axis.

Examples of Usage (It is the same as the G00 order type.)
1) G90   G01 X10.0 Y20.0 Z30 F1000 – If the axis address and coordinate value set in the parameter are inputted directly
2) G01 X0 YL0 ZLL1 FL100   -- It is the same as the example of G00. It moves at the speed of L 100.   '0', the value (Unit: um) registered in '0' of the L variable , and the value of the L variable to which the value registered in '1' of the L variable points become the coordinates of X, Y, and Z axes sequentially.
3) G01 XX10.0   --- The X axis incremental movement order (in case of the G90 mode)
4) G01 P0   -- If the previous axis is ordered directly by using the L variable start number (must include the G code)   The value registered in '0' of the L Variable , '1' , and '2' become the coordinates of X, Y, and Z axes sequentially.

5) G01 PL0   -- If the previous axis is ordered indirectly with the L variable start number (must include the G code)
   Each axis moves to the position of the L Variable from the value registered in '0' of the L Variable .
   Namely, if the L0 value is '12' , the value registered in L12, the value registered in L13, and the value registered in L14 become the coordinates of the X, Y, and Z axes sequentially.

6) G01 XP0   --- Only the X axis of the P0 point moves.

Figure 2.2
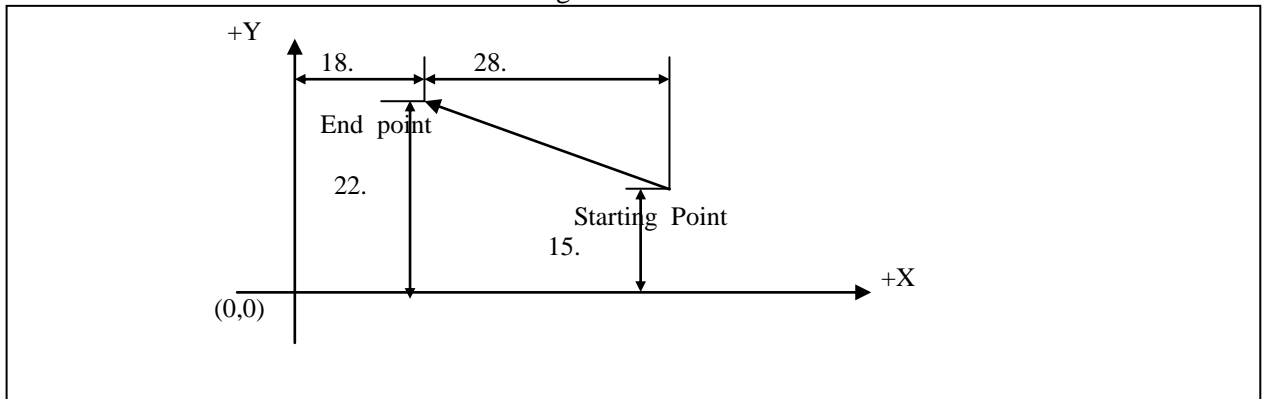
G90 G01 X18. Y22. F200 or G91 G01 X-28. Y7. F200

The speed designated with F is the speed moving the table linearly. (It doesn᾽t go beyond **P33**.)

If the speed is not designated with F once, the speed of the **parameter (P34)** is effective.

☞ **As for the additional speed functions, refer to Chapter 2 Programming 2) Speed and Dwell Functions (2) Movement Speed.**

☞ **The speed in the direction of each axis during interpolation is as follows.**

G01 Xa Yb Ff; In case of an order ----- Xa , Yb: The incremental travel of each axis

● The speed in the X axis direction $Fx = a/L * f$
● The speed in the Y axis direction $Fy = b/L * f$
● $L = SQRT(a*a + b*b)$

(3) Arc Interpolation (G02,G03)

The table can be moved in the arc direction according to the following order.
Be sure to check and select a plane. A standard plane can be selected with parameters.

The arc of the XY plane

G17 $\left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\}$ X__Y__ $\left\{ \begin{matrix} R__ \\ I__J__ \end{matrix} \right\}$ F__

The arc of the ZX plane

G18 $\left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\}$ X__Z__ $\left\{ \begin{matrix} R__ \\ I__K__ \end{matrix} \right\}$ F__

The arc of the YZ plane

G19 $\left\{ \begin{matrix} G02 \\ G03 \end{matrix} \right\}$ Y__Z_ $\left\{ \begin{matrix} R__ \\ J__K_ \end{matrix} \right\}$ F__

X: X axis or its parallel axis (parameter (P04))
Y: Y axis or its parallel axis (parameter (P04))
Z: Z axis or its parallel axis (parameter (P04))

DATA			Command	Meaning
1	Plane designation		G17	Designation of the arc of the XY plane
			G18	Designation of the arc of the ZX plane
			G19	Designation of the arc of the YZ plane
2	Rotation direction		G02	Clockwise (CW) arc interpolation
			G03	Counterclockwise (CCW) arc interpolation
3	Endpoint	G90 Mode	2 axes out of X, Y, and Z	The position of the endpoint in the work coordination system
		G91 Mode	2 axes out of X, Y, and Z	Distance from the starting point to the endpoint
4	Distance from the starting point		2 axes out of I, J, and K	Distance from the starting point to the arc center (increment)
	Radius of the arc		R	Radius of the arc
5	Movement speed		F	Speed of the movement along the arc

The clockwise or counterclockwise direction means the forward or backward direction of the Z axis (Y or X axis) for the XY plane (ZX or YZ plane) of the right hand rectangular coordinate system

Figure 2.3
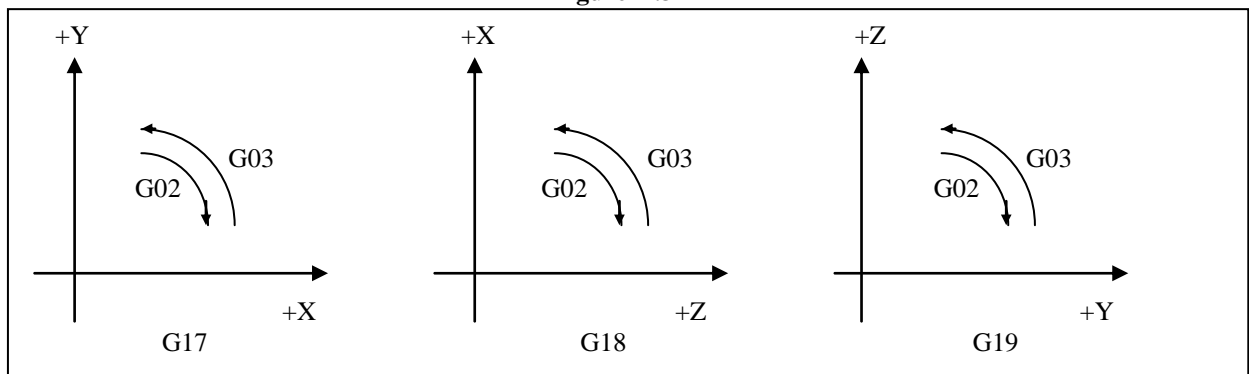
The endpoint of an arc is ordered as the Address X, Y, or Z and is represented as the absolute order value and the incremental order value depending on G90 or G91. In case of the incremental order value,

the distance from the starting point of the arc to the endpoint of the arc is ordered.

The arc center is ordered as Address I, J, and K corresponding to the X, Y, and Z axes. However, **the numerical value placed after I, J, and K** is the vector viewing the center from the starting point of the arc and is always ordered with <u>the incremental value</u> regardless of G90 and G91.

Figure 2.4
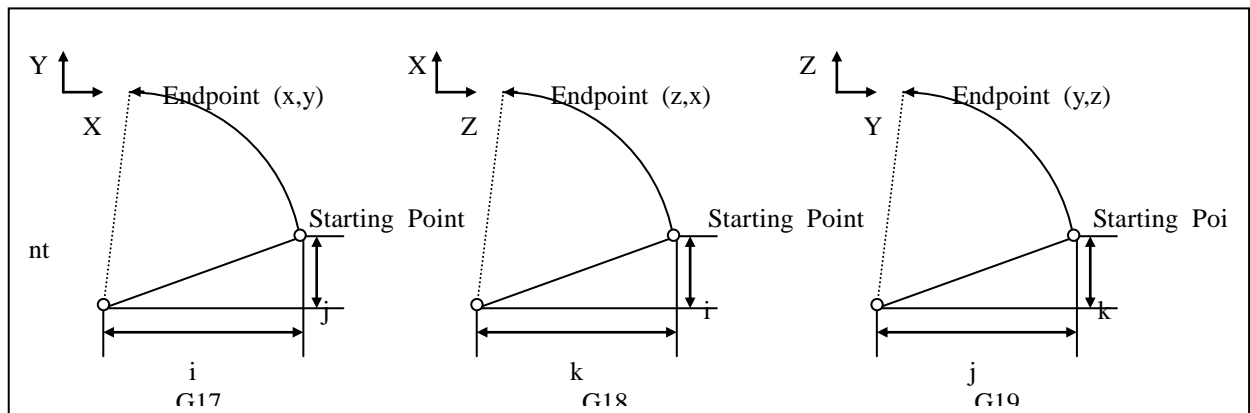
Examples	
G00 X0. Y0. Z0;	
G17;	-- The XY plane is selected in case of the arc interpolation.
G02 X100 Y100 Z50 R100 F1000;	-- It moves from the starting point (0,0) to the endpoint (100, 100) with the '100' radius in the CW direction (clockwise) through the arc (the quarter circle) interpolation. The Z axis moves from '0' to ' 50.000'.
G00 X0. Y0. Z0;	
G03 X100 Y100 I0 J100 F1000;	-- It moves from the starting point (0,0) to the endpoint (100,100) with the '100' radius in the CCW direction (counterclockwise) through the arc (the quarter circle) interpolation.
G17 U0;	-- The UY plane is selected in case of the arc interpolation. (The X axis is set to the parallel axis of the U axis in the parameter.)
G00 U0. Y0. Z0;	
G02 U100 Y100 R100 F1000;	-- It moves from the starting point (0,0) to the endpoint (100,100) with the '100' radius in the CW direction (clockwise) through the arc (the quarter circle) interpolation.
G00 U0. Y0. Z0;	
G03 U0 Y0 I100. J0 F1000;	-- It moves from the starting point (0,0) to the endpoint (100,100) with the '100' radius in the CCW direction (counterclockwise) through the arc (the quarter circle) interpolation.

I, J, and K are marked depending on the direction. The center can be ordered with the radius R instead of I, J, and K.

The order is as follows.

$$\left.\begin{matrix} \textbf{G02} \\ \textbf{G03} \end{matrix}\right\} \quad \textbf{X\_\_ Z\_\_ R\_\_;}$$

When the arc with 180 or more degrees out of the two arcs with 180 or less degrees and 180 or more degrees is ordered, the radius is ordered with the negative (-) value.

Figure 2.5
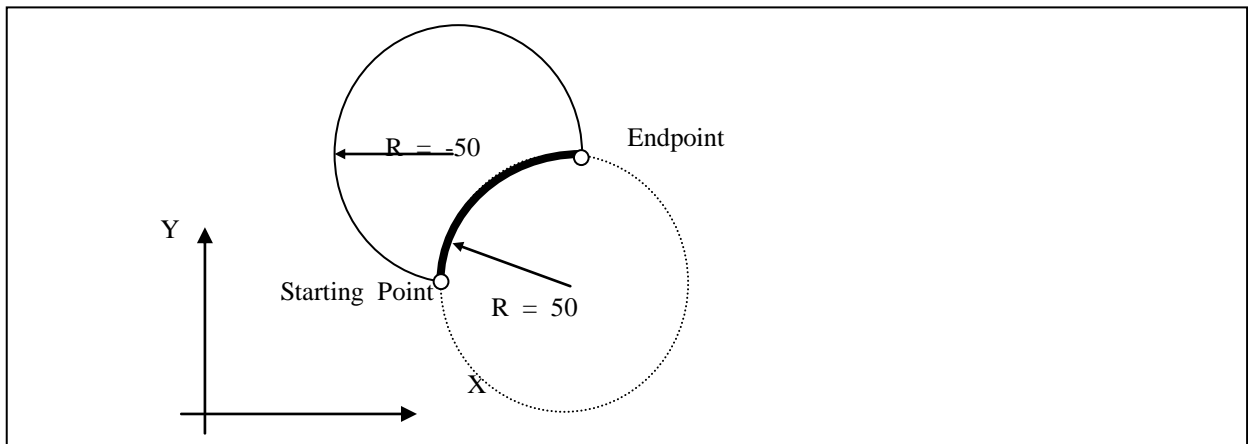
The arc interpolation movement speed becomes the movement speed designated with the F code. The arc interpolation speed (the tangent direction speed of the arc) is controlled to have it become the designated movement speed.

I0, J0, or K0 can be omitted.

The 360-degree arc order equates the starting point and the endpoint. The arc center is order by using I, J, and K.

  Note) (If there are no X,Y, and Z, it is considered that the endpoint is the same as the starting point.)

  G02 I__;

If R is used, it becomes the arc with 0 degree.

  G02 R__; (No movement)

Note) 1. The axis except for the axis making a designated plane is the linear movement axis.

   2. Even though the U axis being parallel to the X axis is ordered when the XY plane is ordered, the alarm is given.

   3. If I, J, K, and R are ordered at the same time, the designation with R has the priority so that I, J, and K are disregarded.

   4. If the controller is 80P (pulse output), enter "12345" in P115.

   5. The R order is modal.

(4) Interpolation of the Arc (G05) or Circle (G06) with the Intermediate Point Ordered

The table can be moved in the direction of the arc or circle passing **the intermediate point** by using the following order.

The coordinate ordered first becomes the intermediate point. The following coordinate value becomes the target value.

The arc of the XY plane
G17 G05 X__Y__ X__Y__Z__F__
G17 G05 P__P__F__

The arc of the ZX plane
G18 G05 Z__X__ Z__X__Y__F__
G18 G05 P__P__F__

The arc of the YZ plane
G19 G05 Y__Z__ Y__Z__X__F__
G19 G05 P__P__F__

Examples
G00 X0 Y0 Z0
G17   -- The XY plane is selected in case of the arc interpolation.
G05 X20 Y70 X100 Y100 Z50 F1000 -- The order for the interpolation of the arc moving from the starting point (0,0) through the intermediate point (20,70,0) to the endpoint (100,100,50) and for the Z axis linear interpolation
G00 X0 Y0 Z0
G19   -- The YZ plane is selected in case of the arc interpolation.
G05 Y20 Z70 Y100 Z100 X50 F1000 -- The interpolation of the arc moving from the starting point (0,0,0) through the intermediate point (0,20,70) to the endpoint (50,100,100)
The X axis moves from '0' to '50.000'. (The arc interpolation plane axis should be ordered first. The other axes should be ordered later on.)
G00 X0 Y0 Z0
G17   -- The XY plane is selected in case of the arc interpolation.
G05 P0 P5 F1000   -- The interpolation of the arc moving from the starting point (0,0,0) through the intermediate point (L0,L1,L2) to the endpoint (L5,L6,L7)
As for the intermediate point, the value from '0' of the L Variable becomes the coordinate of each axis. As for the endpoint, the value from '5' of the L Variable becomes the coordinate of each axis.
G00 X0 Y0 Z0
G17   -- The XY plane is selected in case of the arc interpolation (can be omitted in case of the prior declaration).
G05 PL0 PL1 F1000   -- The interpolation of the arc moving from the starting point (0,0,0) through the intermediate point (L10,L11,L12) to the endpoint (L20,L21,L22) when the value of '0' of the L Variable is '10' and the value of '1' of the L Variable is '20'
As for the intermediate point, the value from '10' of the L Variable becomes the coordinate of each axis. As for the endpoint, the value from '20' of the L Variable becomes the coordinate of each axis.

Note) 1. If the controller is 80P, enter "12345" in P115 [Inposition].

(5) The Next Block Pre-Execution Mode (G10,G11)

G10 X__   -> The ratio order (0%~99%)

If the remaining movement quantity of the block being moved is below the ordered ratio, the next block is overlapped and moved.

'0'   means the release of the pre-execution mode.

G11 X__   -> The distance order

If the remaining movement quantity of the block being moved is below the ordered distance, the next block is overlapped and moved.

'0'   means the release of the pre-execution mode.

Note) 1. The above function is effective in G01, G02, G03, and G05 and is ineffective in G00.
2. In case of 80P, enter "12345" in P115 [Inposition].
3. If G10 and G11 are declared with G12 and G13, G10 and G11 have the priority.

Examples
G00 X0.0 Y0.0 Z0.0
G10 X10   -- The pre-execution mode is set. (Ratio: 10%)
G01 X200 F1000   -- The next block is executed from the 180 position that is 10% before the reaching.
G01 Y100   -- The next block is executed from the position of 90 that is 10% before the reaching.
G01 X0   -- The next block is executed from the position of 20 that is 10% before the reaching.
G01 Y0   -- The pre-execution mode is released in the next block so that it is executed with the standard mode.
G10 X0   -- The pre-execution mode is released.
G11 X15   -- The pre-execution mode is set. (Distance: 15mm)
G01 X200 F1000; -- The next block is executed from the 185 position that is 15mm before the reaching.
G01 Y100   -- The next block is executed from the 85 position that is 15mm before the reaching.
G01 X0   -- The next block is executed from the 15 position that is 15mm before the reaching.
G01 Y0   -- The pre-execution mode is released in the next block so that it is executed with the standard mode.
G11 X0;   -- The pre-execution mode is released.

(6) The Acceleration and Deceleration Non-Execution Mode (G12,G13)

<div style="border:1px solid">

– **Note) The V9.29.1 or lower version is applied.**

</div>

G12 ->It is set to the mode omitting the acceleration and deceleration between blocks. (The continuous movement mode)

G12 X__ ->After it is stopped for the time set after the movement is finished, the next block is executed.

G13 ->It returns to the standard mode. (The movement mode after the stop)

Examples
G00 X0.0 Y0.0 Z0.0
G12   The acceleration and deceleration non-execution mode is set. (Stop Time: 0msec)
G01 X200 F1000   The next block is executed without the deceleration of the X axis.
G01 Y100   The block is started without the acceleration of the Y axis. The next block is executed without deceleration. Acceleration and deceleration are omitted so that the movement of the Y axis is possible without the stop of the X axis. However, if the movement speed is high, it may strain the motor.
G01 G12 X5   The next block is executed if 5msec passes after the movement is finished without deceleration.
G01 X0   The block is started without the acceleration of the X axis. The next block is executed if 5msec passes after the movement is finished without deceleration.
G01 Y0   The acceleration and deceleration mode is released in the next block. Hence, it stops with deceleration as in the standard mode. The movement is finished.
G13;   It returns to the standard mode.

<div style="border:1px solid">

– **Note) The V9.29.1 or higher version is applied.**

</div>

- The movement mode after the stop in case of the continuous interpolation (G12)

G12; The command setting the acceleration and deceleration between blocks and moving it after the stop between blocks

G12 A_ D_; It set the interpolation acceleration and deceleration. The execution mode of the next block is set after the complete stop.
       A_ sets the acceleration time. D_ sets the deceleration time.
The acceleration and deceleration time is set up to or from P33 (the maximal interpolation speed). The time unit is ms.

Examples
G00 X0 Y0 Z0
G12 A100 D100   The movement mode is set after the change or stop of acceleration and deceleration.
G01 X200 F1000   The changed acceleration and deceleration are applied for movement.   The next command is executed after the complete stop.
G01 Y100   The changed acceleration and deceleration are applied for movement.   The next command is executed after the complete stop.

- The continuous movement mode (G13)

G13; The command setting the acceleration and deceleration between blocks and executing it without stopping the movement between blocks

G13 A_ D_; The interpolation acceleration and deceleration are set. The continuous movement mode between blocks is set.

A_ sets the acceleration time. D_ sets the deceleration time. The parameter is effective in case of omission. The acceleration and deceleration time is set up to or from P33 (the maximal interpolation speed). The time unit is ms.

X__; The next block is executed after it is stopped by the time set after the complete movement. It is '0' in case of omission.

Examples
G00 X0 Y0 Z0 G12 A100 D100   It changes the acceleration and deceleration and sets the continuous movement mode. G01 X200 F1000   The next block is executed without the stop of the interpolation movement. G01 Y100   The interpolation movement is started in continuance with the previous block. The next block is executed without the stop. G13   It sets the movement mode after the stop.

Note) 1. If G10 and G11 are declared with G12 and G13, G10 and G11 have the priority.

(7) Skip Movement (G31)

Setting the signal with G31 makes the following block executed after the stop with deceleration if the corresponding signal is sensed while G01 ordered next is being executed.

G31 X__(M__)   -> The signal is set. (X: Input contact, M: Auxiliary contact)
G01 IP__ F__

Examples
G00 X0. Y0. Z0; G31 X0.3:   The skip movement is set. G01 X200 F1000:   If the input X0.3 is sensed as 'ON' while the X axis is moving from '0' to '200', the next block is executed after the stop with deceleration. If the input X0.3 is not sensed as 'ON', the next block is executed after it is moved to the ordered '200'. G01 Y100;

Note) In case of 80P, enter "12345" in P115 [Inposition].

(8) Normal Control (G40,G41,G42)

G41 X__(Y,Z,U,V,W) F_-> Adjusting the left of the direction of progress and designating the speed in case of the single movement of the normal axis

G42 X__(Y,Z,U,V,W) F_-> Adjusting the right of the direction of progress and designating the speed in case of the single movement of the normal axis

G40 X__(Y,Z,U,V,W) -> Releasing the normal control and resetting the coordinate of the normal axis

The axis set with the normal control is rotate automatically to be vertical to the direction of progress when another axis moves through interpolation.

Note) 1. Be sure to select a plane.
　　　 2. The number being next to the axis address is ineffective.

Examples
G17　　　　　　　An XY plane is selected in case of the normal control.
G00 X0 Y0 Z0 U0
G41 U0　　　　　It is set to the left adjustment normal control U axis.
G01 X100 F1000 It is the order moving the X axis from　'0'　to　'100' . The X axis moves after the U axis moves to　'-90'　first before the X axis moves with the left adjustment normal control.
G01 Y100.0　　　It is the order moving the Y axis from　'0'　to　'100' . The X axis moves after the U axis moves to　'0'　first before the Y axis moves with the left adjustment normal control.
G01 X0　　　　　It is the order moving the X axis from　'100'　to　'0' . The X axis moves after the U axis moves to　'90'　first before the X axis moves with the left adjustment normal control.
G01 Y0　　　　　It is the order moving the Y axis from　'100'　to　'0' . The Y axis moves after the U axis moves to　'180'　first before the Y axis moves with the left adjustment normal control.
G40:　　　　　　The normal control is released.

* The rotation unit of the corresponding axis is set in the 4th item of　"P114 reservation parameter 5" .
: It is reset to the following unit in case of the G40 declaration.
　　0 : 360,000　　　　2 : 3,600
　　1 :　36,000　　　　3 :　　360

Designating the axis excluding the interpolation speed (G43)

G43 X__(Y,Z,U,V,W) -> Designating the axis excluding the interpolation speed

(9) Palletizing Movement (G70,G71)

G70 L__ L__;

The L variable designated first + 0: The number of the 1st axis pallet points

The L variable designated first + 1: The number of the 2nd axis pallet points

The L variable designated first + 2: The number of the 3rd axis pallet points

The L variable designated first + 3: The 1st axis pallet start point

The L variable designated first + 4: The 2nd axis pallet start point

The L variable designated first + 5: The 3rd axis pallet start point

The L variable designated first + 6: Setting the output contact after the pallet completion (0 : Y0.0 ~ 55 : Y3.7)

The L variable designated first + 8: Setting the starting point of the output contact after the pallet completion

(0: Not used, 1: After the completion of all layers, 2: After the completion of each layer)

The L variable designated second + 0 ~ 2: The 1st position value

The L variable designated second + 3 ~ 5: The 2nd position value

The L variable designated second + 6 ~ 8: The 3rd position value

The L variable designated second + 9 ~ 11: The 4th position value

It moves through the linear interpolation.

G71 L__; The pallet start point clear

If the L variable designated first in G70, the pallet start point of each axis is reset to '1'.

L10	4 (The number of the 1st axis pallet points)	L20	0 (The 1st position value of the X axis)
L11	3 (The number of the 2nd axis pallet points)	L21	0 (The 1st position value of the Y axis)
L12	1 (The number of the 3rd axis pallet points)	L22	0 (The 1st position value of the Z axis)
L13	1 (The 1st axis pallet start point)	L23	300000 (The 2nd position value of the X axis)
L14	1 (The 2nd axis pallet start point)	L24	0 (The 2nd position value of the Y axis)
L15	1 (The 3rd axis pallet start point)	L25	0 (The 2nd position value of the Z axis)
L16	2 (Setting the output contact (Y0.2) after the pallet completion)	L26	0 (The 3rd position value of the X axis)
L17	0	L27	150000 (The 3rd position value of the Y axis)
L18	1 (Setting the output of the contact point (Y0.2) after the completion of all layers)	L28	0 (The 3rd position value of the Z axis)
		L29	0 (The 4th position value of the X axis)
		L30	0 (The 4th position value of the Y axis)
		L31	0 (The 4th position value of the Z axis)

Examples
G00 X0. Y0. Z0   G71 L10   The pallet start point is reset. Namely, L13, L14, and L15 are reset to '1' .   G70 L10 L20 F1000   It moves to the pallet 1 point (0,0,0).   G00 X-100 Y-100   G70 L10 L20 F1000   It moves to the pallet 2 point (100,0,0).   G00 X-100 Y-100   G70 L10 L20 F1000   It moves to the pallet 3 point (200,0,0).   G00 X-100 Y-100

Examples
G00 X0. Y0. Z0   G71 L10   The pallet start point is reset. Namely, L13, L14, and L15 are reset to '1' .   N1   G70 L10 L20 F1000   It moves to the pallet (1~12) point.   G00 X-100 Y-100   IF Q200.0 = 0 N1   If the pallet work is not done, the pallet work continues to be done after the ju   mp to N1. The Q Variable 200.0 means the PLC contact point region N0.0.   <PLC Program>   LOAD Y0.2   OUT M0.0
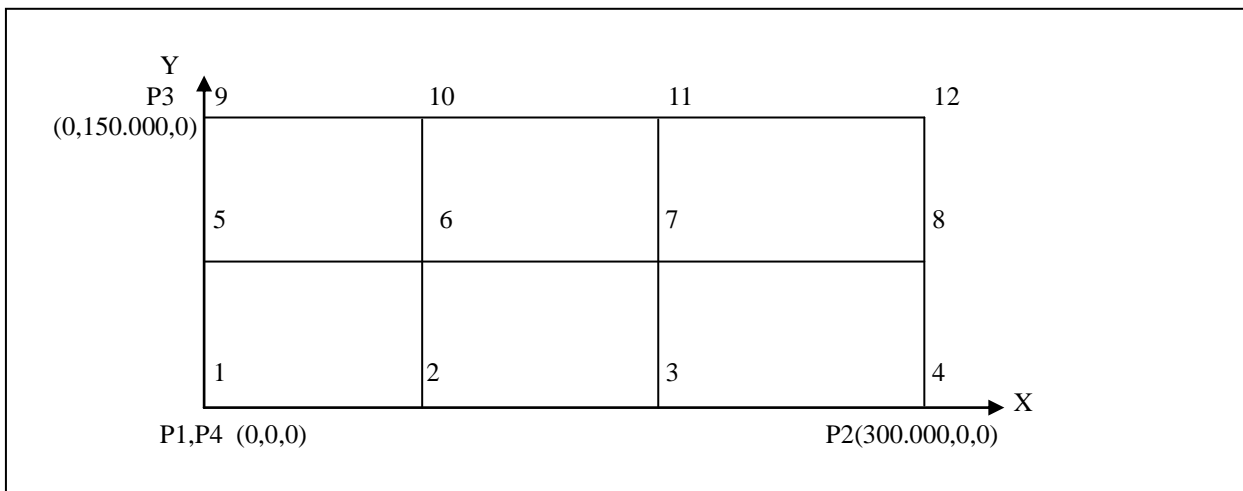
(10) Movement Including the Output Control (G14,G15,G16,G24,G25,G26)

G14(G15,G16,G24,G25,G26): It is the command controlling the output set while the movement command of the block following an order is being executed.

G14(G15,G16,G24,G25,G26): It is effective only in the block following an order.

The output contact is set in the 2nd item (the 1st contact point) and 3rd item (the 2nd contact point) of the "P114 reservation parameter 5".

Code	Function
G14	It moves after the 1st and 2nd contact points are on.
G15	The 1st and 2nd contact points are off after movement.
G16	The 1st and 2nd contact points are off after it moves after the 1st and 2nd contact points are on.
G24	It moves after the 1st and 2nd contact points are off.
G25	The 1st and 2nd contact points are on after movement.
G26	The 1st and 2nd contact points are on after it moves after the 1st and 2nd contact points are off.
	Address and Function
A	The 1st contact point is on or off after it moves the distance ordered when the 1st contact point is controlled before movement.
C	The 2nd contact point is on or off after it moves the distance ordered when the 2nd contact point is controlled before movement.
B	The 1st contact point is on or off after it moves the distance ordered when the 1st contact point is controlled after movement.
D	The 2nd contact point is on or off after it moves the distance ordered when the 2nd contact point is controlled after movement.
P	In case of controlling the 1st and 2nd contact points before movement, it moves if the ordered time passes after the 1st and 2nd contact points are on or off.

10.1) G14 A__ C__; (Unit: μm)

Without A and C: It moves after outputting (on) the 1st and 2nd contact points set when the next block is moved.

A: Outputting (on) the 1st contact point after moving the set distance

C: Outputting (on) the 2nd contact point after moving the set distance

10.2) G14 P__; (Unit: msec)

It moves if the set time passes after the 1st and 2nd contact points are outputted (on).

10.3) G15 B__ D__; (Unit: μm)

Without B and D: The 1st and 2nd contact points set after the completion of the movement of the next block are outputted (off).

B: The 1st contact point is outputted (off) with the movement of the set distance left.

D: The 2nd contact point is outputted (off) with the movement of the set distance left.

10.4) G16 A__ B__ C__ D__; (Unit: μm)

Without A, B, C, and D: After the output (on) of the 1st and 2nd contact points set when the next block is moved, the 1st and 2nd contact points set after the completion of movement are outputted (off).

A: The 1st contact point is outputted (on) after the movement of the set distance.

C: The 2nd contact point is outputted (on) after the movement of the set distance.

B: The 1st contact point is outputted (off) with the movement of the set distance left.

D: The 2nd contact point is outputted (off) with the movement of the set distance left.

10.5) G16 P__; (Unit: msec) It moves if the set time passes after the 1st and 2nd contact points are outputted (on).

10.6) G24 A__ C__; (Unit: μm)

Without A and C: It moves after outputting (off) the 1st and 2nd contact points set when the next block is moved.

A: Outputting (off) the 1st contact point after moving the set distance

C: Outputting (on) the 2nd contact point after moving the set distance

10.7) G24 P__; (Unit: msec) It moves if the set time passes after the 1st and 2nd contact points are outputted (off).

10.8) G25 B__ D__; (Unit: μm)

Without B and D: The 1st and 2nd contact points set after the completion of the movement of the next block are outputted (on).

B: The 1st contact point is outputted (on) with the movement of the set distance left.

D: The 2nd contact point is outputted (on) with the movement of the set distance left.

10.9) G26 A__ B__ C__ D__; (Unit: μm)

Without A, B, C, and D: After the output (off) of the 1st and 2nd contact points set when the next block is moved, the 1st and 2nd contact points set after the completion of movement are outputted (on).

A: The 1st contact point is outputted (off) after the movement of the set distance.

C: The 2nd contact point is outputted (off) after the movement of the set distance.

B: The 1st contact point is outputted (on) with the movement of the set distance left.

D: The 2nd contact point is outputted (on) with the movement of the set distance left.

10.10) G26 P__; (Unit: msec) It moves if the set time passes after the 1st and 2nd contact points are outputted (off).

A: The 1st contact point is outputted (on) after the movement of the set distance.

C: The 2nd contact point is outputted (on) after the movement of the set distance.

B: The 1st contact point is outputted (off) with the movement of the set distance left.

D: The 2nd contact point is outputted (off) with the movement of the set distance left.

10.5) G16 P__; (Unit: msec) It moves if the set time passes after the 1st and 2nd contact points are outputted (on).

10.6) G24 A__ C__; (Unit: μm)

Without A and C: It moves after outputting (off) the 1st and 2nd contact points set when the next block is moved.

A: Outputting (off) the 1st contact point after moving the set distance

C: Outputting (on) the 2nd contact point after moving the set distance

10.7) G24 P__; (Unit: msec) It moves if the set time passes after the 1st and 2nd contact points are outputted (off).

10.8) G25 B__ D__; (Unit: μm)

Without B and D: The 1st and 2nd contact points set after the completion of the movement of the next block are outputted (on).

B: The 1st contact point is outputted (on) with the movement of the set distance left.

D: The 2nd contact point is outputted (on) with the movement of the set distance left.

10.9) G26 A__ B__ C__ D__; (Unit: μm)

Without A, B, C, and D: After the output (off) of the 1st and 2nd contact points set when the next block is moved, the 1st and 2nd contact points set after the completion of movement are outputted (on).

A: The 1st contact point is outputted (off) after the movement of the set distance.

C: The 2nd contact point is outputted (off) after the movement of the set distance.

B: The 1st contact point is outputted (on) with the movement of the set distance left.

D: The 2nd contact point is outputted (on) with the movement of the set distance left.

10.10) G26 P__; (Unit: msec) It moves if the set time passes after the 1st and 2nd contact points are outputted (off).

2) Speed (F) and Dwell Functions

(1) Rapid Movement Speed

A position is determined with the position determination order (**G00**) at the rapid movement speed.

As the rapid movement speed is set in the parameter (P32) for each axis, the order doesn't have to be given with the program.

Rapid Override: 0, 25, 50, and 100% overriding can be done with the switch of the machine operation panel on the basis of the rapid movement speed. (**Refer to P23.2.**)

Note) Refer to how to change or order the speed (FF ,FV , FX/Y/Z···) in case of the movement at a rapid speed (G00).

(2) Interpolation Movement Speed (F)

The speed at which the motor is moved through the linear interpolation (**G01**) and the arc interpolation (**G02,G03**) is ordered with the F value. (The standard unit is mm/min in case of a linear axis.)

① **The Constant Speed in the Tangent Direction**

The interpolation movement is controlled to have the speed in the tangent direction always become the ordered speed.
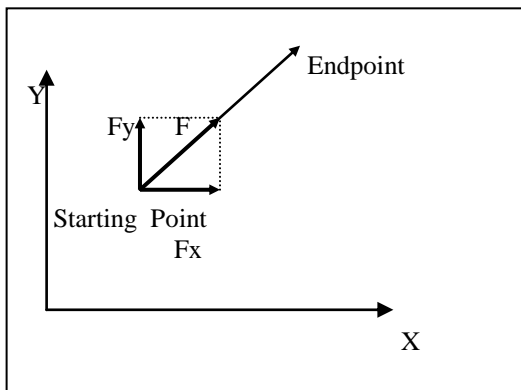
Figure 2.6	**Figure 2.7**
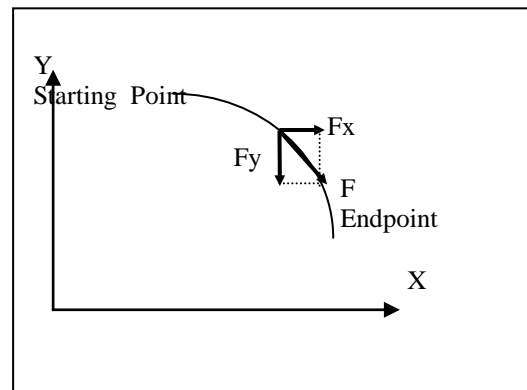
```
F  : The speed in the tangent direction
Fx : The speed component in the X axis direction
Fy : The speed component in the Y axis direction
```

The speed in the direction of each axis in case of interpolation is as follows.

In case of the G01 Xa Yb Ff order ----- Xa, Yb: The incremental travel of each axis

- The speed component in the X axis direction $Fx = a/L * f$
- The speed component in the Y axis direction $Fy = b/L * f$
- $L = SQRT(a*a + b*b)$

The Clamp of the Interpolation Movement Speed

1. **The interpolation movement speed limit** can be set in the parameter (P33).

2. Even though the actual interpolation movement speed (the movement speed with 100% overriding) is beyond the upper limit, the movement speed is clamped to have the interpolation movement speed limit not go beyond **the parameter (P33)**. The clamp value is set with the unit of mm/min.

3. **If the parameter (P33) is set to the value ending with 'I', it is automatically calculated to have the ordered resultant speed not go beyond the rapid movement speed (P32) of each axis.**

② **Input of the Movement Speed Setting**

1. If the interpolation movement speed doesn't have to be changed during processing, the interpolation movement speed can be designated in the parameter (P34).

   Hence, the interpolation movement speed (F code) doesn't have to be ordered in the MC program. (Modal)

2. If the numerical value except for 0 is set in the parameter, the interpolation movement speed during the power supply is initially set to the value set in the **parameter (P34)**.

   If 0 is set, it means that the interpolation movement speed is not set in the parameter (P34). Hence, the interpolation movement speed during the power supply is not set initially.

   However, even though the interpolation movement speed is designated in **the parameter (P34)**, the interpolation movement speed order in the MC program is effective.

   Moreover, if another interpolation movement speed is ordered, the value is preserved before the power is turned off.

Namely, if the interpolation movement speed is ordered in **the parameter (P34)**, the F code is not cleared with the reset.

(3) Override

① **Movement Speed Override (G01/02/03 ···Interpolation Feed Override)**

It can be overridden with the switch of the machine operation panel. (0% ~ 150%)

- **Refer to the explanation of the parameter P23.1 / contact point Q1.8 ~ Q1.F (the contact point OV1.OV2,OV4, OV8).**

② **Rapid Movement Override (G00 Rapid Override)**

0, 25, 50, and 100% overriding can be done with the switch of the machine operation panel depending on the rapid movement speed.

- **Refer to the explanation of the contact point Q1.D(R02),Q1.C(R01) / Q1.E(RT) contact point, and the parameter P23.2/P35.**

(4) Automatic Acceleration and Deceleration

It is accelerated and decelerated automatically at the start and end of movement so that it can be started and stopped smoothly.

Moreover, when the movement speed is changed, it is also accelerated and decelerated automatically so that the speed can be changed smoothly.

Hence, acceleration and deceleration doesn't have to be considered specially in the program.

The number of acceleration and deceleration corrections can be designated in the acceleration and deceleration parameters (P41 and P42) for rapid movement or interpolation movement.

Figure 2.8
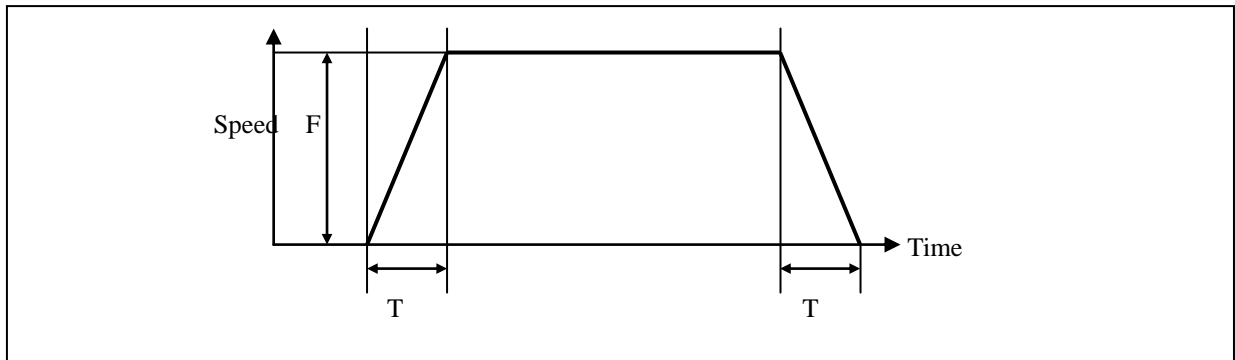
(5) Dwell (G04)

The G04 order can delay it by the dwell time until the motion of the next block is reached.

The motion of the next block is delayed by the ordered time.

Type:

G04 X__ or **D__**   The dwell time is ordered in the unit of sec. (0.000~9999.999 sec)

G04 P__   The dwell time is ordered in the unit of msec. (0~9,999,999 msec)

Note) If the L variable is used, it is processed in the unit of msec regardless of address.

(6) Dwell (G20, The Simple Dwell with the External Encoder)

The G20 order can delay the motion of the next block until it is inputted by the number of ordered **external encoder pulses (MPG port)**.

Type:

G01 IP__:
G20 X100: ---- It waits until the external encoder 100 pulse is inputted after the front block is completed

G01 IP__:
G20 X200: ---- It waits until the external encoder 200 pulse is inputted after the front block is completed

Examples
G00 X0. Y0. Z0;
G20 X100;　　　　It waits until 100 external encoder pulses are inputted after the execution of the front block (0,0,0) is finished.
G01 X100 Y100;
G20 X100;　　　　It waits until 100 external encoder pulses are inputted after the execution of the front block (100,100,0) is finished. Namely, the external encoder pulse inputted during the execution of the front block is disregarded.
G01 X200 Y200;

(7) Dwell (G21, The Accumulated Dwell with the External Encoder)

The G21 order can delay the motion of the next block until the external encoder is inputted by the number of ordered **external encoder pulses (MPG port)**.

Type:

G01 IP__;
G21 X100; ---- It waits until the accumulated external encoder becomes 100 after the program is started.

G01 IP__;
G21 X200; ---- It waits until the accumulated external encoder becomes 200 after the program is started.

Examples
G00 X0. Y0. Z0;　　The accumulated external encoder becomes ʻ0ʼ if the program is started.
G21 X100;　　　　It waits until 100 accumulated external encoder pulses are inputted after the program is started.
G01 X100 Y100;　　The external encoder pulse continues to be accumulated.
G21 X200;　　　　It waits until 200 accumulated external encoder pulses are inputted after the program is started.
G01 X200 Y200

3) Additional MCS Commands

- **The orders of the additional G codes are in the following table.**

Code	Description
G09	Inposition check
G10	The next block pre-execution based on the XYZ axes (on the basis of the ratio)
G11	The next block pre-execution based on the XYZ axes (on the basis of the distance)
G35	The pre-execution of the block being next to the previous axis (on the basis of the ratio)
G36	The pre-execution of the block being next to the previous axis (on the basis of the distance)
G22	Dwell based on an external pulse
G38	The synchronized operation based on the external pulse of a designated axis
G39	Releasing the synchronized operation based on the external pulse of a designated axis
G66	Subroutine call
G80	Waiting until the corresponding contact point is 'ON'
G81	Waiting until the corresponding contact point is 'OFF'
G82	Waiting until the corresponding contact point is 'ON'- High speed check
G83	Waiting until the corresponding contact point is 'OFF'- High speed check
G95	The speed order based on a long axis

(1) G09 [Inposition Check]

Details) The sum of the movement errors of all the controlled axes should be within the set distance to proceed to the next block.

Example) G09 X1.5

--If the sum of the movement errors of all the controlled axes is within 1.5mm, the next block is executed. If it is greater than 1.5mm, the waiting process continues until it is within the range.

Note) The function is effective in MCS-80P only if an external encoder is mounted.

(2) G10 [The Next Block Pre-Execution Based on the XYZ Axes (on the Basis of the Ratio)]

Details) If the residual movement quantity of the block being moved is left below the ordered ratio, the next block is moved in an overlapped way.

If there is no movement of X,Y and Z axes in the next block order, the pre-execution movement is not executed.

If the set value is '0', it means the release of the pre-execution mode.

Note) 1. However, if "12345" is entered in P115 when the controller is 80P (pulse output), it is effective.
   2. The function is effective in G01, G02, G03, and G05 and ineffective in G00.

Type) G10 D_ or G10 X_

Exmaples [G10 X_ ; G10 D_]	
G00 X0 Y0 Z0 A 100	
G10 X10; or G10 D10;	The pre-execution mode is set (Ratio: 10%).
G01 X200 F1000;	The next block is executed from the position 180 that is 10% before it is reached.
G01 Y100 ;	The next block is executed from the position 90 that is 10% before it is reached.
G01 X0;	There is no order for the X, Y and Z axes in the next block so that it is operated in the standard mode.
G01 A100;	
G01 X100;	The next block is executed from the position 90 that is 10% before it is reached.
G01 Y200	
G10 X0; or G10 D0;	Pre-execution is released in the next block so that it is operated in the standard mode. The pre-execution mode is released.

In the above example, the difference between G10 X10 and G10 D10 generates the difference in the movement path overlapping the next block.

G10 X10

G10 D10

Pre-Execution
Path
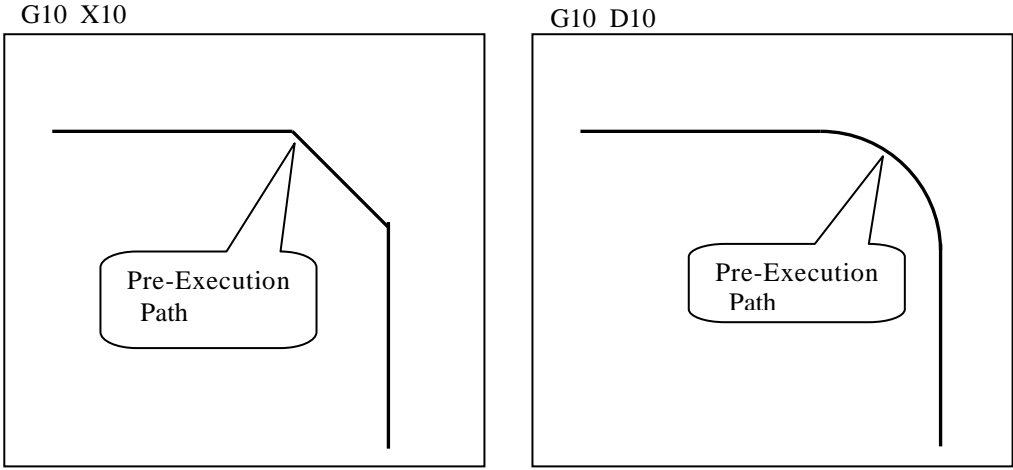
Pre-Execution
Path

(3) G11 [The Next Block Pre-Execution Based on the XYZ Axes (on the Basis of the Distance)]

Details) If the residual movement quantity of the block being moved is left below the ordered distance, the next block is moved in an overlapped way.

If there is no movement of XYZ axes in the next block order, the pre-execution movement is not executed.

If the set value is 'ㅇ', it means the release of the pre-execution mode.

Note) 1. The function is effective in G01, G02, G03, and G05 and ineffective in G00.

2. However, if "12345" is entered in P115 when the controller is 80P type, it is effective.

Type) G11 D_ or G11 X_

Examples [G11 X_ ; G11 D_]	
G00 X0 Y0 Z0 A100 G11 X15; or G11 D15; G01 X200 F1000; G01 Y100; G01 X0; G01 A100; G01 X100; G01 Y200 G11 X0 or G11 D0	The pre-execution mode is set (Distance: 15mm). The next block is executed from the position 185 that is 15mm before it is reached. The next block is executed from the position 85 that is 15mm before it is reached. The next block is executed from the position 185 that is 15mm before it is reached. The next block is executed from the position 85 that is 15mm before it is reached. Pre-execution is released in the next block so that it is operated in the standard mode. The pre-execution mode is released.

In the above example, the difference between G11 X15 and G11 D15 generates difference in the movement path overlapping the next block.

G11 X10

Pre-Execution Path

G11 D10
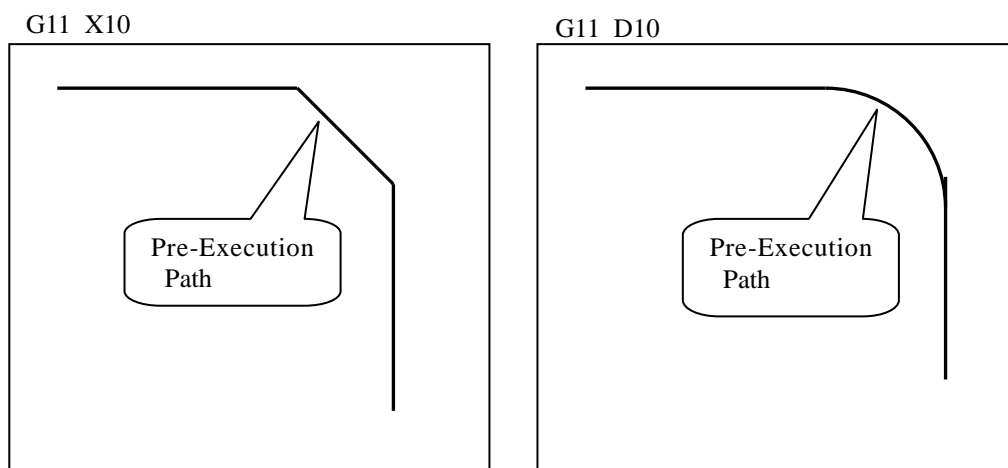
Pre-Execution Path

(4) G35 [Pre-Execution of the Block Being Next to the Previous Axis (on the Basis of the Ratio)]

Details) If the residual movement quantity of the block being moved is left below the ordered ratio, the next block is moved in an overlapped way.

If the set value is '0', it means the release of the pre-execution mode.

Note) 1. The function is effective in G01, G02, G03, and G05 and ineffective in G00.
2. However, if "12345" is entered in P115 when the controller is 80P (pulse output), it is effective.

Type) G35 D_ or G35 X_

Examples [G35 X_ ; G35 D_]	
G00 X0 W0 V0 A100 G35 X10; or G35 D10; G01 X200.0 F1000; G01 W100; G01 X0; G01 A100; G01 X100; G01 V200; G35 X0; or G35 D0;	The pre-execution mode is set (Ratio: 10%). The next block is executed from the position 180 that is 10% before it is reached. The next block is executed from the position 90 that is 10% before it is reached. The next block is executed from the position 20 that is 10% before it is reached. The next block is executed from the position 90 that is 10% before it is reached. The next block is executed from the position 90 that is 10% before it is reached. Pre-execution is released in the next block so that it is operated in the standard mode. The pre-execution mode is released.

In the above example, the difference between G35 X10 and G11 D10 generates difference in the movement path overlapping the next block.

G35 X10

G35 D10

Pre-Execution Path
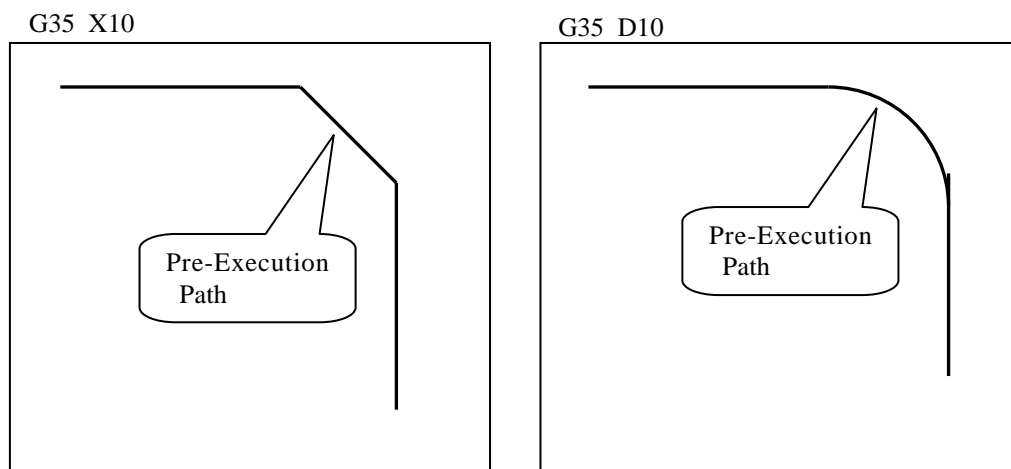
Pre-Execution Path

(5) G36 [Pre-Execution of the Block Being Next to the Previous Axis (on the Basis of the Distance)]

Details) If the residual movement quantity of the block being moved is left below the ordered distance, the next block is moved in an overlapped way.

If the set value is '0', it means the release of the pre-execution mode.

.

Note) 1. The function is effective in G01, G02, G03, and G05 and ineffective in G00.
2. However, if "12345" is entered in P115 when the controller is 80P (pulse output), it is effective.

Type) G36 D_ or G36 X_

Examples [G36 X_ ; G36 D_]	
G00 X0 W0 V0 A100 G36 X15; or G36 D15.	The pre-execution mode is set (Distance: 15mm).
G01 X200 F1000; G01 W100; G01 X0; G01 A100; G01 X100; G01 V200	The next block is executed from the position 185 that is 15mm before it is reached. The next block is executed from the position 85 that is 15mm before it is reached. The next block is executed from the position 15 that is 15mm before it is reached. The next block is executed from the position 85 that is 15mm before it is reached. The next block is executed from the position 85 that is 15mm before it is reached. Pre-execution is released in the next block so that it is operated in the standard mode.
G36 X0 or G36 D0	The pre-execution mode is released.

In the above example, the difference between G36 X10 and G36 D10 generates difference in the movement path overlapping the next block.

G36 X10

G36 D10

Pre-Execution Path

Pre-Execution Path
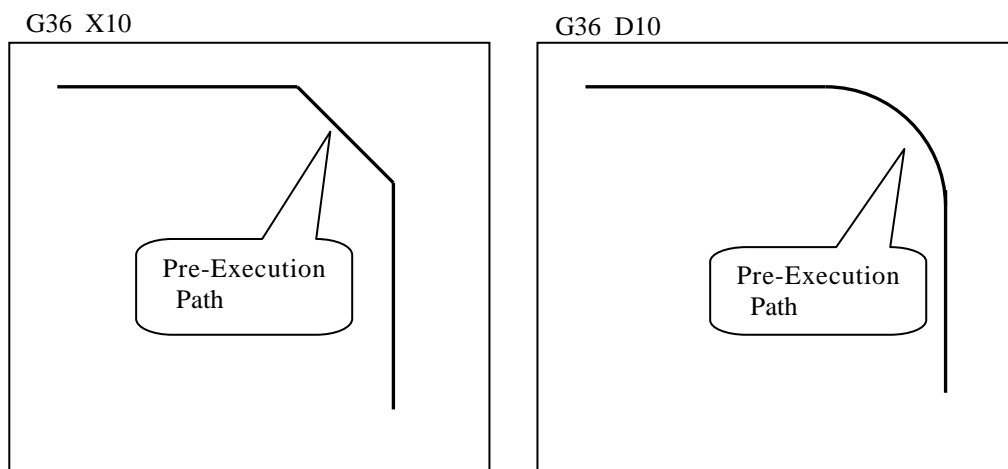
(6) G22 [The Dwell Function Based on the External Pulse / Accumulation / High Speed]

Details) The dwell operation is executed to handle the pulse heat inputted in the designated port.

Type) G22 A[Port] P[Pulse];

Port	0: MPG port, 1 ~ 8: Axis encoder port axis 1 ~ axis 8
Pulse	The input pulse to be stopped

Example 1) G22 A0 P100;

The dwell operation is executed until 100 pulses are inputted from the MPG port.

Example 2) G22 A4 P50;

The dwell operation is executed until 50 pulses are inputted from the axis 4 encoder port.

(7) G38 [The Synchronized Operation Based on the External Pulse of a Designated Axis] / G39 [Release of the Synchronized Operation]

Details) G38 executes the synchronized operation to handle the pulse heat inputted in the designated port.

Type) **G38 [Sync Axis] A[Port] D[Distance/Rev] P[Pulse/Rev];**

Sync Axis	The movement axis for synchronized operation is set.
Port	0: MPG port, 1 ~ 8: Axis encoder port axis 1 ~ axis 8
Distance/Rev	The movement distance of the synchronization axis for each rotation of the external pulse generator
Pulse/Rev	The number of pulses for each rotation of the external pulse generator

Example 1) G38 X0 A0 D100. P1000;

If 1000 pulses are inputted from the MPG port, the X axis starts the 100mm synchronized operation. Namely, the 0,1mm synchronized movement for each pulse is executed. The order is effective until the synchronized operation is released by G39.

Example 2) G38 Y0 A4 D100. P4000;

If 4,000 pulses are inputted from the axis 4 encoder port, the Y axis starts the 100mm synchronized operation.

Namely, the 0.025mm synchronized movement for each pulse is executed.

The order is effective until the synchronized operation is released by G39.

Note) 1. If the controller is 80P, enter "12345" in P115 [Inposition] of the axis for synchronized axis.

(8) G66 [Subroutine Call]

Details) It returns after it jumps to the ordered number N for the execution until M02 is met.

Type) **G66 Nxx;** It is branched to the designated Nxx.
   ...

    Nxx;
    ...
    M02; It returns to the called G66 if M02 is met after the branch.

Note) Write the subroutine in the lower part of the call G66 as in the example if possible.

Example)

```
        1:
        2:
        3:
  G66 N10
      4:
        5:
        6:
  M02

  N10
        7:
        8:
  G66 N20
      9:
       10:
  M02

  N20
       11:
       12:
  M02
```
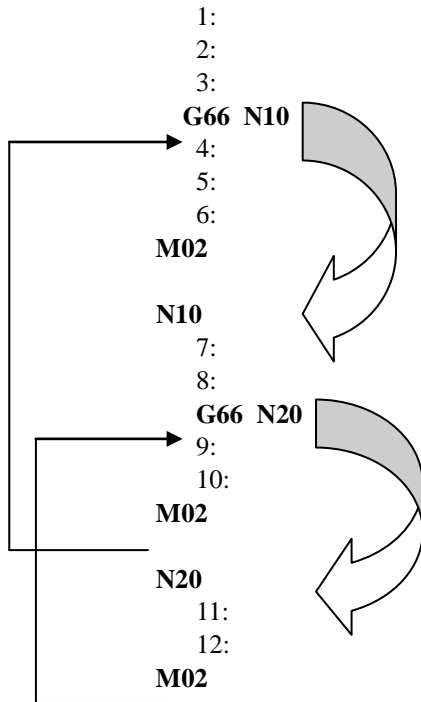
If it is ordered as above, the execution order is as follows.
  1 → 2 → 3 → 7 → 8 → 11 → 12 → 9 → 10 → 4 → 5 → 6

(9) G80 [Waiting Until the Corresponding Contact Point Is 'ON']

Details) It waits until the corresponding contact point is 'ON'.

Example) **G80 X0.1 or G80 M0.0**

It waits until the input contact X0.1 or M0.0 is 'ON'.

Note) The orderable contact points are X (input) and M (secondary).

(10) G81 [Waiting until the Corresponding Contact Point Is 'OFF']

Details) It waits until the corresponding contact point is 'OFF'.

Example) **G81 X0.1 or G80 M0.0**

It waits until the input contact X0.1 or M0.0 is 'OFF'.

Note) The orderable contact points are X (input) and M (auxiliary).

(11) G82 [Waiting until the Corresponding Contact Point is "ON' – High Speed Check]

Details) It waits until the corresponding contact point is 'ON'.

Example) **G82 X0.1**

It waits with the check at a high speed until the input contact X0.1 is 'ON'.

Note) The orderable contact point is X (input).

(12) G83 [Waiting until the Corresponding Contact Point is 'OFF' – High Speed Check]

Details) It waits until the corresponding contact point is 'OFF'.

Example) **G83 X0.1**

It waits with the check at a high speed until the input contact X0.1 is 'OFF'.

Note) The orderable contact point is X (input).

(13) G95 [The Speed Order Based on the Long Axis]

Details) It is the command changing the speed order type on the G01 command.

The movement speed of the axis with the long movement distance is applied as the speed order value. Another movement axis is calculated automatically on the basis of the movement speed of the long movement axis. Moreover, the speed order can be changed to "RPM".

Type 1) **G95 X1** ----   The speed order based on the long axis is applied in the unit of mm/min.

Type 2) **G95 X2** ---   The speed order based on the long axis is applied in the unit of RPM.

Type 3) **G95 X0**   --- The standard speed order mode is applied.

Example 1)

```
G00 X0 Y0 Z0
G95 X1                    --- The speed order based on the long axis is set in the unit of mm/min.
G01 X100. Y10. Z50 F1000  --- The movement distance of the X axis is 100.0mm that is the longest.
                              Hence, the speed of the X axis is set and executed with 1000 mm/M.
                              The speed of the Y axis is set and executed with 100 mm/M. The speed
                              of the Z axis is set and executed with 50mm/M.


G01 X90 Y210 Z0           --- The movement distance of the Y axis is 200.0mm that is the longest.
                              Hence, the speed of the X axis is set and executed with 5 mm/M. The
                              speed of the Y axis is set and executed with 1000 mm/M. The speed of
                              the Z axis is set and executed with 250mm/M.
G95 X0                    --- The standard speed order mode is set.
```

Example 2)

```
G00 X0 Y0 Z0
G95 X2                    --- The speed order based on the long axis is set in the unit of RPM.
G01 X100. Y10. Z50 F1000  --- The movement distance of the X axis is 100.0mm that is the longest.
                              Hence, the speed of the X axis is set and executed with 1000 RPM.
                              The speed of the Y axis is set and executed with 100 RPM. The
                              speed of the Z axis is set and executed with 50RPM.
G01 X90 Y210 Z0           --- The movement distance of the Y axis is 200.0mm that is the longest.
                              Hence, the speed of the X axis is set and executed with 5 RPM. The
                              speed of the Y axis is set and executed with 1000 RPM. The speed
                              of the Z axis is set and executed with 250RPM.
G95 X0                    --- The standard speed order mode is set.
```

2.2 Regarding the Coordinate System

1) Origin

The reference point is the fixed point of a machine. It is the point to which the table can be moved easily.

The position relationship between the reference point and the origin of a machine are set in the parameter (P11).

Figure 2.9

α,β: Parameter (P11)
γ,ζ: The offset quantity of the work coordinate system
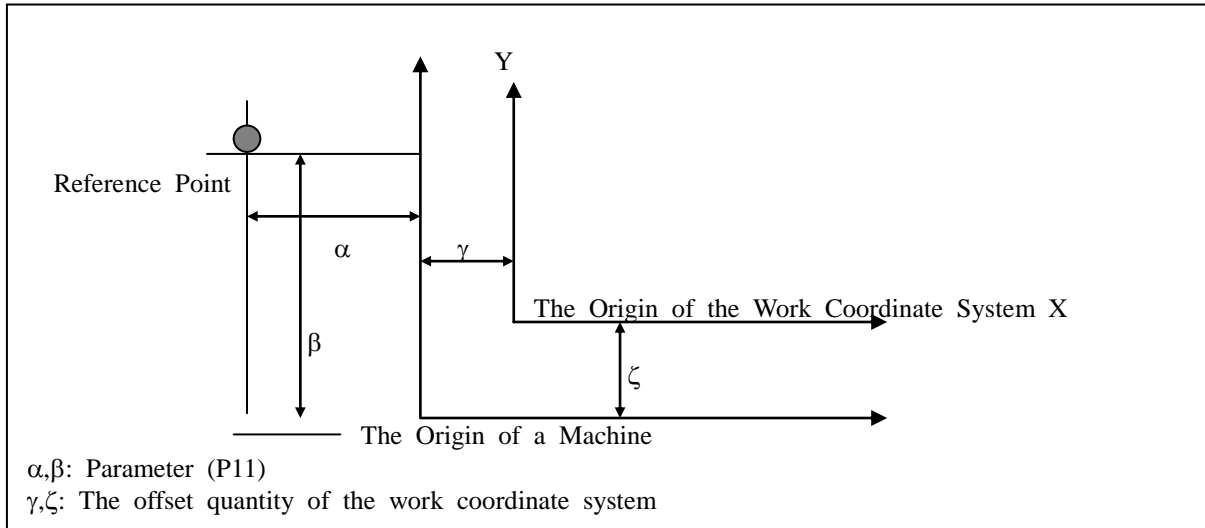
(1) The Automatic Return of the Reference Point (G28)

Type: G28 IP__

The axis ordered by the above order can be automatically returned to the reference point.

However, if it is not returned to the origin, the origin return action executed in the original mode is made.

IP__ is the order of the movement to the intermediate point to return to the reference point. It is ordered with an absolute value or an incremental value.

As for the motion of the G28 block, the ordered axis determines the position at the intermediate point at a rapid speed and the position from the intermediate point to the reference point at a rapid speed (**Figure 2.10**)
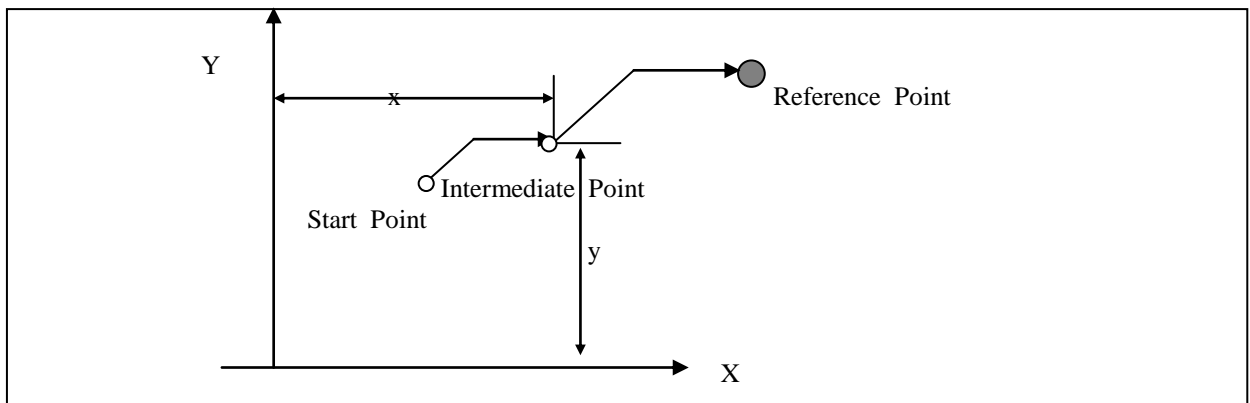
(2) The Rapid Movement Based on the Machine Coordinate System (G29)

Type: G29 IP__

The axis ordered by the above order is rapidly moved to the position based on the machine coordinate system.

(3) The Rapid Movement of the 2ⁿᵈ Origin (G30)

Type: G30 IP__

The axis ordered by the above order is rapidly moved to the position saved in P821.

2) Coordinate System

The position which the table has to reach is notified to the controller. The controller moves the table to the position.

The reached position is ordered with the coordinate value depending on the coordinate system.

There are two types of coordinate systems such as the machine coordinate system and the work coordinate system.

The reached position is ordered with the coordinate value of the selected coordinate system if needed.

The coordinate value is displayed depending on the axis component of the program.

If the program axes are three axes such as X, Y, and Z, they are displayed as X__Y__Z__.

The number of program axes, the number of the components of the coordinate value, is different depending on a machine.

(1) Setting the Machine Coordinate System

The point becoming the basis in a machine is called the Origin of the machine. The origin of the machine is usually determined depending on a machine. The coordinate system with the origin of the machine becoming the origin of the coordinate system is the machine coordinate system.

The table is not necessarily moved to the origin of the machine.

The origin of the machine is sometimes in the position to which the table cannot be moved.

The machine coordinate system is established by returning to the reference point after power supply.

The machine coordinate system established once doesn't change with the reset, setting of the work coordinate system, and another manipulation if power is not turned off.

The stored stroke limit (P12 and P13) setting the operation range of a machine is set as the machine coordinate value of the machine coordinate system.

① **Setting the Machine Coordinate System**

The machine coordinate system is the unique coordinate system of a machine. If the return to the origin is achieved manually, the coordinate system is set with the coordinate value with the Origin set in the parameter (P11).

② **Resetting the Machine Coordinate System**

The machine coordinate system can be reset with G51.

Type: **G51 IP__**

With the command, the machine order system (namely, the unique coordinate system) is reset with the ordered value.

③ **Resetting the Machine Coordinate System (Recommended)** - **Recommended to Use It in 80A**

The machine coordinate system can be reset with G52.

 Type: G52 IP__

With the command, the machine order system (namely, the unique coordinate system) is reset with the ordered value.

Note) In case of 80A, the position of G51 changes <u>by the servo offset</u>. G52 can prevent it.

Figure 2.11
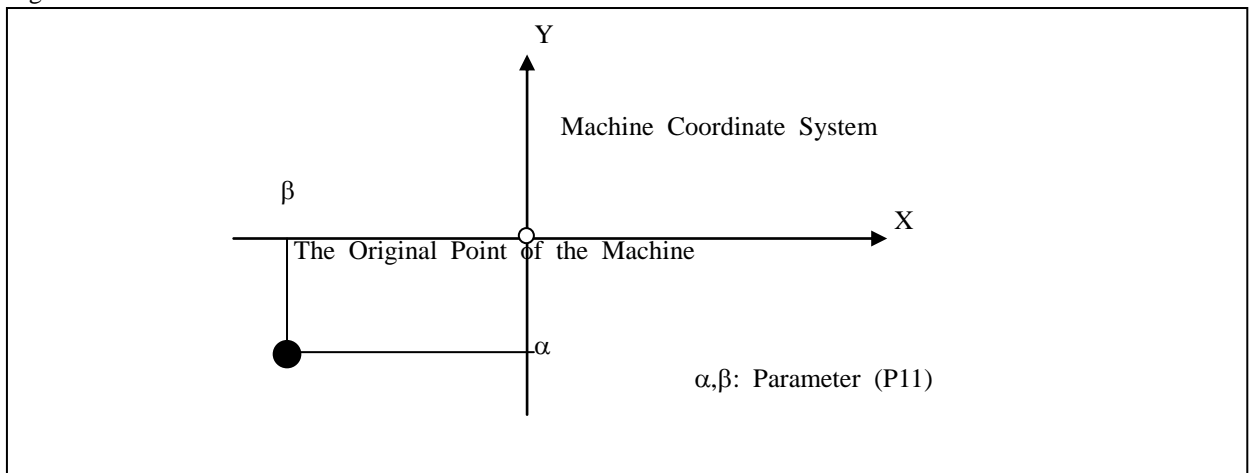
(2) Setting the Work Coordinate System

The coordinate system used for table movement is the work coordinate system.

The work coordinate system can be set with G50.

Type: G50 IP__

With the order, the work coordinate system in which the table point, the table position, becomes the (I P) of the coordinate system is established.

If the coordinate system is established, a further absolute order becomes the position the work coordinate system follows.

If it returns to the origin with the manual return to the origin, the offset of the origin set as G50 is cancelled so that the work coordinate system is set.

Figure 2.12 (G50 X700. Y500.; If ordered (resetting the absolute coordinate system))
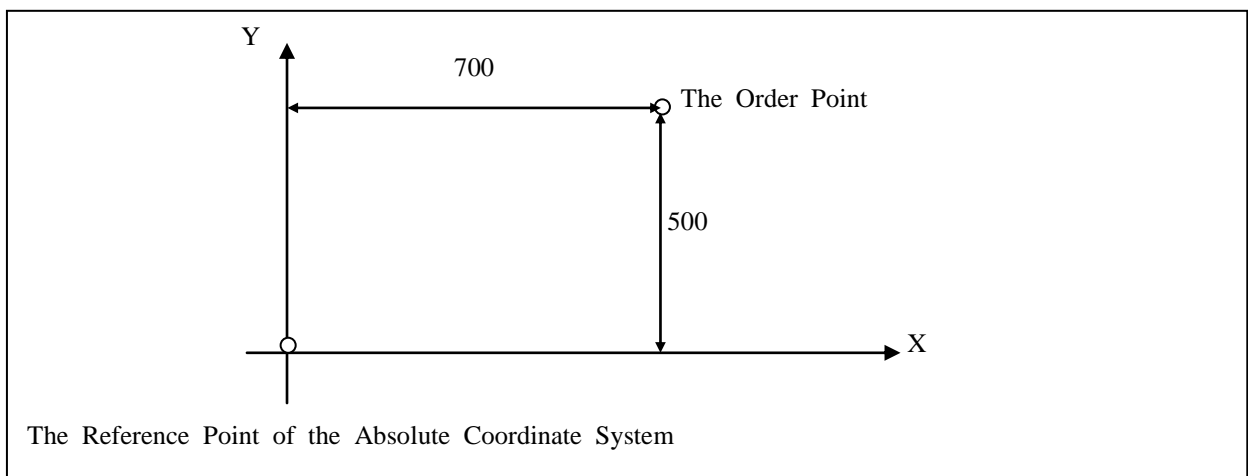
Example	Absolute Coordinate	Machine Coordinate	
G00 X0 Y0;	(0.0,0.0)	(0.0,0.0)	
G01 X100 Y-100;	(100.0,-100.0)	(100.0,-100.0)	
	(0.0,0.0)	(100.0,-100.0)	The offset (100,-100) occurs by G50.
G50 X0 Y0; G01 X50 Y50;	(50.0,50.0)	(150.0,-50.0)	The difference between the absolute coordinate and the machine coordinate is maintained by the offset.
G29 X50 Y50;	(-50.0,150.0)	(50.0,50.0)	The machine coordinate order with the offset excluded by G29
G01 X50 Y50;	(50.0,50.0)	(150.0,-50.0)	The difference between the absolute coordinate and the machine coordinate is maintained by the offset.
G51 X0 Y0;	(0.0,0.0)	(0.0,0.0)	The coordinate system without the offset is reset by G51.

(3) Selecting a Plane (G17, G18, G19)

The arc interpolation plane is selected with the G code.

G17------→ XY plane
G18------→ ZX plane
G19------→ YZ plane

X is the X axis or its parallel axis. Y is the Y axis or its parallel axis. Z is the Z axis or its parallel axis. An axis identified out of several parallel axes by referring to the axis address of the block with G17, G18, and G19 ordered.

Example)   When X and U, Y and V, and Z and W are parallel axes

G17 X__Y__ → XY plane
G17 U__Y__ → UY plane
G18 X__W__ → XW plane
G18 U__W__ → UW plane
G19 Y__Z__ → YZ plane
G19 V__Z__ → VZ plane

A plane doesn't change in the block with G17, G18, and G19 not ordered.

Ex.) G18 X__Z__ → ZX plane
     G01 X__Y__ → A plane doesn't change.(ZX plane)

If the axis address is omitted in the block with G17, G18 and G19 ordered, it is considered that the axis addresses of the standard three axes are omitted.

Ex.) G17   → XY plane
      G17 X__   → XY plane
      G17 Y__   → XY plane
      G17 U__   → UY plane
      G17 V__   → XV plane
      G18   → ZX plane
      G18 W__   → WX plane

 - Which one of X, Y, and Z is parallel to the additional axis can be determined in the parameter (P04).

 - The movement order is irrelevant to the selection of a plane.

   For example, in case of the order of G17 Z__;, Z is the axis that doesn't exist in the XY plane. Only the XY plane is selected. Z moves the Z axis regardless of the plane.

 - G17(XY) is selected during the power input with reset.

3) The Coordinate System and the Dimension

(1) The Absolute Order and the Incremental Order: (G90,G91)

There are two methods to order the travel of each axis. They are the absolute order and the incremental order.

The absolute order is the method used to program the coordinate value of the position of the movement endpoint of each axis.

The incremental order is the method used to directly program the travel of each axis. The absolute or incremental order is ordered by G90 and G91.

G90: Absolute order

G91: Incremental order

Figure 2.13
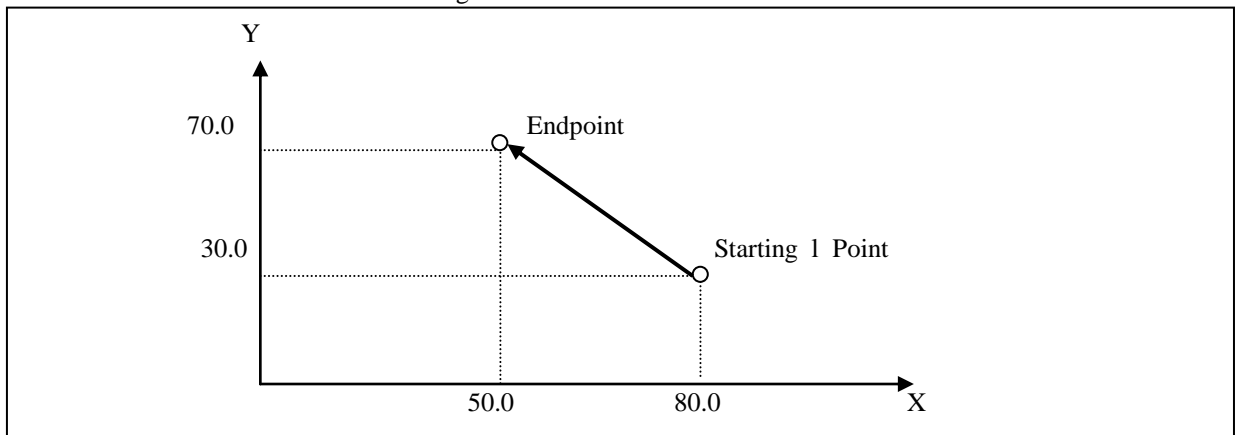
In the above figure, if it is programmed with the incremental order, it becomes G91 X-30.0 Y40.0;. In case of the absolute order, it becomes G90 X50.0 Y70.0;.

Note) When the axis movement order is used in the abosolute order mode (G90) by mixing the absolute order with the incremental order and the corresponding axis address is declared twice, the corresponding axis moves incrementally.

Ex.) If "G00 X100.0 **YY**100.0 " is executed, the X axis moves to the absolute position 100mm, and the Y axis moves to the incremental position 100mm.

(2) Dimension Input

In the control device, a numerical value can be inputted by using a decimal point. A decimal point is used for the unit of distance, time, and speed. However, there is a restriction depending on an address. The unit of a decimal point is mm or sec.

Ex.)　　G00 Z15.0　　　→ Z15mm
　　　　　G04 D1.0　　　　→ Dwell for a second
　　　　　G00 X1000　　　→ 1000mm
　　　　　G00 X1000.　　　→ 1000mm
　　　　　G00 X0.500　　　-> 0.5 mm or 500um
　　　　　G00 XL100 (if 200 is inputted in L100) → 0.2 mm or 200um

2.3 Auxiliary Function (M Function)

It is the command controlling the PLC contact point in the MC program.
A movement order and an M function order can be started at the same time.

**(The standard I/O signal MC-> MF, M02 and M11~M28 contact points of PLC /
PLC->Refer to MC contact point QQ.7,Q1.7.)**

1) Auxiliary Function (M Function)

If values are ordered consecutively to the address M, the MF (M Function) signal is outputted.

The signal is used for the on/off control of a machine.

Only one M code is effective in one block. If two or more M codes are ordered, only the M code de clared later is effective.

If an M code is ordered while the MC program is being executed, the corresponding PLC contact poi nt (**MC->PLC** contact point I2.8~2.F,I2.7) becomes '1'. It waits until the Q0.7 contact point (FIN) becomes '1' again after the execution is stopped for a moment.

At this time, if the Q0.7 contact point (FIN) is set from '1'->'0'in the PLC program, the MC progra m continues to execute the next block. (However, it is recommended to set the contact point Q1.7 (FINE) to '1'.)

MC Program		The State of the PLC Contact Point		PLC Program
. . .	1. If the M code is order during execution the cor responding conta ct point is turne d 'ON' and is st opped momentari ly			.. ..
M15		I2.C=1 I2.A =1 I2.8=1 I2.7=1	2. If PLC prog ram sets Q0.7 to '1' I 2.7 be come '0'	
	3. If Q0.7 bec omes '0', the next block is e xecuted.	I2.C=1 I2.A =1 I2.8=1 I2.7=0		LOAD ...... SET Q0.7 ..
. G01 X100. . .		or		LOAD ..... RST Q0.7 ..
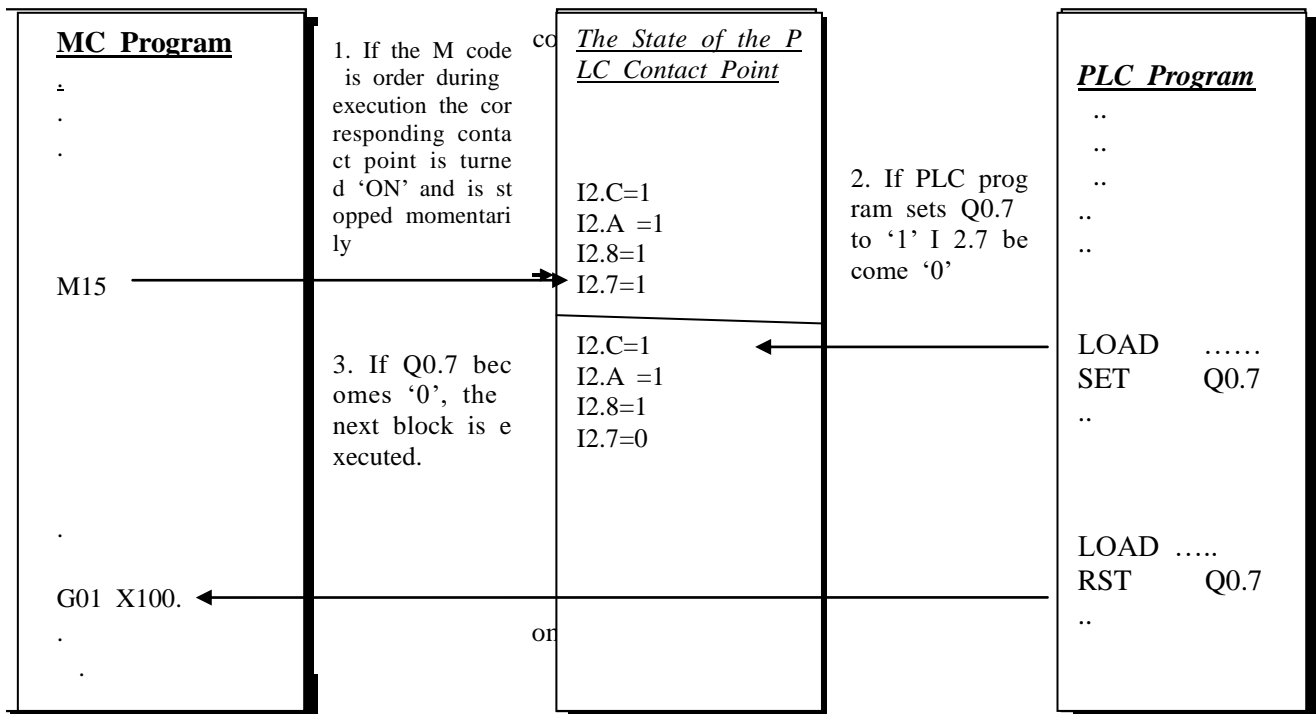
[Example of the M Code Usage]

Note) M02 and M30 is outputted (MF signal) to the machine and should be processed wit h MFIN. (The standard I/O signal MC->Refer to the contact points **MF,M02, and M11 ~M28** of PLC.)

Special M codes are as follows.

- **M02 (END)**: END OF Program (The FIN processing is needed.)

  The end of the main program is shown. The automatic operation is stopped. The contact point M02 is outputted.
  (Refer to the standard I/O signal MC->Refer to the contact point M02.)

- **M30 (END)**: END OF Program **(same as the function of M02)**

- **M98:** Subprogram call (FIN processing is unnecessary.)

  Ex.) M98 P10 L5;
  It is the order of consecutively calling the subprogram number 10 five times.
  Calling a subprogram in another subprogram is the same as calling a subprogram in the main program.

- **M99**: END OF SUB Program (FIN processing is unnecessary.)
  - The end of subprogram is shown. If M99 is executed, it returns to the main program.
  - If it is used in the main program, the main program is executed repetitively.

Note) M98 and M99 don't generate the code and strobe signals (MF signals).
(The standard I/O signal MC->Refer to the contact point MF of PLC.)

2.4 Main Program and Subprogram

1) Main Program

Programs are classified into a main program and subprograms. The controller can be operated by the main program. However, a subprogram can be also operated with the subprogram call in the main program and with the return to the main program in the subprogram.

Note) M02 and M30 is outputted (MF signal) to the machine. MFIN processing should be done. (The standard I/O signal MC->Refer to the contact points MF, M02, and M11~M28 of PLC.)

2) Subprogram

The fixed sequence or repetitive pattern in a program can be registered in the memory as a subprogram in advance to simplify the program. A subprogram can call a separate subprogram. Namely, the subprogram called by a main program is a single call. Maximal octuple calls are possible.

```
   MAIN(01)           SUB (10)           SUB(20)            SUB(30)
   ┌─────────┐        ┌─────────┐        ┌─────────┐        ┌─────────┐
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │ M98 P10;│        │ M98 P20;│        │ M98 P30;│        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │    :    │        │    :    │        │    :    │        │    :    │
   │ M02;    │        │ M99;    │        │ M99;    │        │ M99;    │
   └─────────┘        └─────────┘        └─────────┘        └─────────┘

                       Single            Double             Triple   ------ Octuple
```
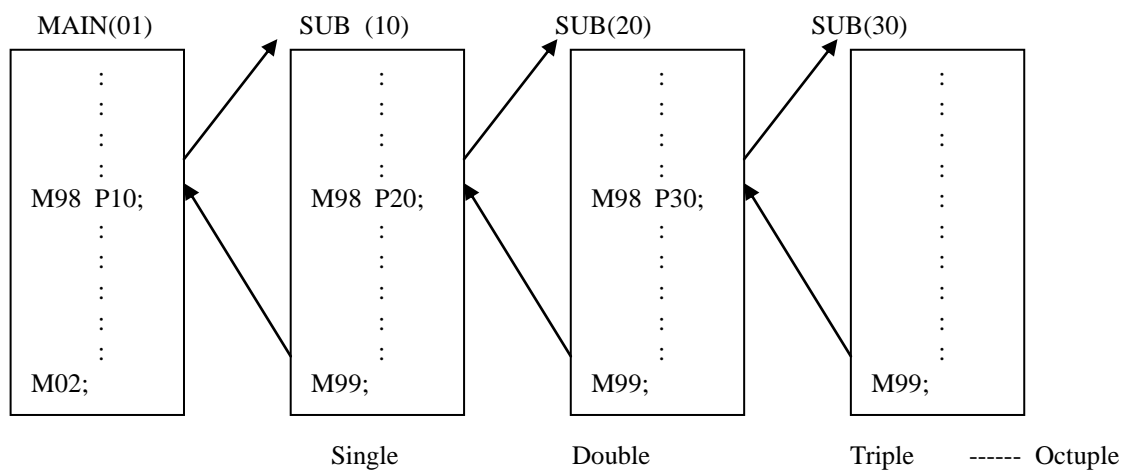
A subprogram can continue to be called repetitively 9999 times with a call order.

(1) Preparing the Subprogram

A subprogram is prepared with the following type.

```
        ;
        ;
        ;
   M99;
```

M99 is ordered at the end of the subprogram.

(2) Executing the Subprogram (M98 and G65)

A subprogram is called for execution by a main program or a similar subprogram.

A subprogram is called as follows.

- **Executing a subprogram with the M code**

    Type: M98 P__ L__;

    P__: Subprogram number

    L__: If the number of repetitive calls is omitted, it becomes once.

    Ex.) M98 P1002 L5;

    It is the order of the five consecutive calls of the subprogram number 1002.

    Calling a subprogram in another subprogram is the same as calling a subprogram in a main program.

    Note) 1. M98 and M99 signals are not outputted to MF (the standard I/O signal MC->PLC).

    2. If there is no subprogram number designated as the address P, an alarm is given.

    3. The M98 P__; M99; block doesn't perform the single block stop.

- **Executing the subprogram with the G code**

    Type: G65 P__ L__;

    P __: Subprogram number

    L __: If the number of repetitive calls is omitted, it becomes once.

    The operation is the same as M98.

2.5 Variables and Conditional Statements

1) Operator: It is the operator that can be used for the arithmetic operation of variables or numbers.

	Operator	Details		Operator	Details
1	+	Add	10	COS( )	Cos
2	-	Subtract	11	TAN( )	Tan
3	*	Multiply	12	ASIN( )	Arc Sin
4	/	Divide (floating value)	13	ACOS( )	Arc Cos
5	!	Quotient of division (integer)	14	ATAN( )	Arc Tan
6	%	Remainder of division (integer)	15	ABS( )	Absolute value
7	O	Bit OR (**Note**)	16	SQRT( )	$\sqrt{}$
8	A	Bit AND (**Note**)	17	(,)	Parenthesis
9	SIN( )	Sin	18	=	Assignment

(**Note**) **1.** If 'O' or 'A' is to be used, **be sure to insert space before and after it**.

2) L Variable (32-Bit Floating Variable) / Range: 0 ~ 9999 (1000 Units in Total)

Standard : L0 ~ L7999 (User Area) : Read/write possible
L8000 ~ L8999 (System Area) : Only read
L9000 ~ L9999 (System Area) : PLC area mapping

Operators Used: +, -, *, /, = , (,), A,O,!,%,SIN, COS, TAN, ASIN, ACOS, ATAN, ABS, SQRT

1. L00 ~ L7999: It is stored and saved in the controller memory.(non-volatile)
2. A user can use it arbitrarily as a variable for various kinds of arithmetic operation and control, a position variable (0.001mm/0.001Deg), speed (mm/Min), or count.
3. The numerical operation of numbers and integers is possible. The arithmetic operation of **hexadecimal values** is possible.
    However, in case of calculating a hexadecimal value, '**0**' should be inserted **in front of a hexadecimal value** with the '**H**' **in the end** to be recognized as a hexadecimal value.
    Ex.) L0 = L0 + 01AD2H
4. The arithmetic integer guarantee value is **-9,999,999 ~ 9,999,999** during the **integer** calculation.

Examples (Using the Coordinate Value Based on Arithmetic Operation)
L100 = 100 * SIN(30);   --- The L Variable is used for arithmetic operation. 50 is inputted in the L Variable 100.
G01 X100 YL100;   --- The calculated L Variable is ordered as the Y axis coordinate.

Examples (Conditional Branch)
L0=0;   -- 0 is inputted in the L Variable 0.
N1
G00 X0 Y0;
G01 X100. Y100.
L0=L0+1;   -- 1 is inputted in the L Variable 0 according to arithmetic operation.
IF L0 .LT 11 N1;   -- If the value of L0 is less than the number 11, it jumps to N1. Otherwise, it performs the next block. Namely, it is the example of executing the loop 10 times in the above program.

Examples (Position Data)
G00 X0 Y0 Z0
G00 XL10 YL11 ZL12 FL13
G01 P0;   -- The order is given with the coordinate of each axis from the L variable '0'. L0 100000, L1: -50000, L2:   If it is 10000, the order coordinate of X,Y,Z is (100., -50.,10.).

Note) In case of executing the above command, it might night be synchronized with the previous motion command.

Ex.) G00 X100.0
L100=1234
Before the X axis reaches 100mm during execution, 1234 can be inputted in the L variable 100. if the execution is desired after the completion of a motion command, input the G04 command in front of the order L or use the expanded L Variable order function below the lower part.

- Additional application command (A23.15.1/ P23.25.1 version or higher)

a. The expanded L Variable order command: LT

LT****: As for an order, the program line is executed after the motion command read first before the command.
Ex.) LT100=1234 is the same as **G04 X0.0**.
                  L100=1234

b. Reading a Double Word:
- It is the function of reading the data in the designated address and the next address as the WORD(16 Bits)/INT(16 Bits) data and process them as the 32-bit (float) data.
Ex.) When L100 = DL9005 is executed

	High WORD(16BIT)	Low WORD(16BIT)
L100 <=	L9006	L9005

c. Q Variable Calculation
- The Q word data can be calculated too.

d. LLOD: Duplication of the L variable (However, it is effective in the version 23_*7_1 or higher.)
LLOD   A** B**** C*** ;**
A: The target address of the L variable to be copied /B: The start address of the L variable to be copied /C: The number of units to be copied
Ex.) LLOD   A10 B20 C4   (L20 -> L10   Copying four variables)

L10	0		L20	10		L10	10		L20	10
L11	1		L21	11	After execution->	L11	11		L21	11
L12	2		L22	12		L12	12		L22	12
L13	3		L23	13		L13	13		L23	13

e. LCLR: L variable initialization (However, it is effective in the version 23_*7_1 or higher.)
   LCLR   A** D**** C*****
      A: The target address of the L variable to be inputted /D: The value to be initialized
         /C: The number of the units to be initialized
   Ex.) LCLR   A10 D100 C4

L10	0				After execution->	L10	100		
L11	1					L11	100		
L12	2					L12	100		
L13	3					L13	100		

Note) Use the check and setting in the parameter related to the usage of the L variable .

   1. Regarding Special Variables
      1. P802.0   -> Special GOTO function (L7400 ~ L7999 used)
      2. P800.0   -> Point variable offset
      3. P800.1   -> Tool variable offset

   2. Variable Initialization and M Contact Point Backup
      4. P800.5   -> Initial L Variable initialization function
      5. P800.7   -> M contact point backup function

   3. Using the L Variable Block Parameter Function
      Expanded Parameter: Refer to P809.0 ~ P809.4.

   4. Operating the L Variable with the Contact Point
      Q50 ~ Q59 (Refer to Chapter 9 Standard I/O Signals.)

*) Expanded System L Variable #1 (9000 ~ 9999): 1,000 units

Note) Version MCSA – V22.35.1(80A)/V22.45.1(80P) or higher is applied.

MC Program L variable		PLC	Remark	User	
L9000 ~ L9099		I00 ~ I99	MC -> PLC	READ Only	
L9100 ~ L9199		Q00 ~ Q99	PLC-> MC	READ / WRITE	
L9200 ~ L9299		M00 ~ M99	Auxiliary contact	READ / WRITE	
L9300 ~ L9305		X00 ~ X05	X area (H/W input)	READ Only	
L9306 ~ L9311		Y00 ~ Y05	Y area (H/W output)	READ / WRITE	
L9312 ~ L9321		F00 ~ F09	F area (Flag)	READ Only	
L9322 ~ L9323		T00 ~ T01	T area (Timer)	READ Only	
L9324 ~ L9325		C00 ~ C01	C area (Counter)	READ Only	
L9400 ~ L9783			D area (16-bit user integer)	READ / WRITE	
L9784 ~ L9815			T area (Timer set value)	READ / WRITE	
L9816 ~ L9847	⇔		T area (The present value of the timer)	READ / WRITE	
L9848 ~ L9879		D0 ~ D511	C area (The set value of the counter)	READ / WRITE	
L9880 ~ L9911			C area (The present value of the counter)	READ / WRITE	
MC Program L variable		**Parameter**	**Remark**	**User**	
Note) The changed value is memorized after power off or on only if the flash storage is executed with the **PLC command (FWR)** after writing a value. If the above flash storage is not executed, the changed value cannot be memorized after power off or on.					
L9912 ~ L9919		P12.0 ~P12.7	P12 SW of each axis +LIMIT	READ / WRITE	
L9920 ~ L9927		P13.0 ~ P13.7	P13 SW of each axis-LIMIT	READ / WRITE	
L9928 ~ L9935		P32.0 ~ P32.7	P32 Rapid movement speed	READ / WRITE	
L9936 ~ L9943		P35.0 ~ P35.7	P35 Maximal movement speed	READ / WRITE	
L9944 ~ L9951		P36.0 ~ P36.7	P36 JOG speed	READ / WRITE	
L9952 ~ L9959		P37.0 ~ P37.7	P37 Origin return speed #1	READ / WRITE	
L9960 ~ L9968		P38.0 ~ P38.7	P38 Origin return speed #2	READ / WRITE	
L9969		P33	P33 Maximum interpolation speed	READ / WRITE	
L9970		P34	P34 Interpolation movement speed	READ / WRITE	

*) Expanded System L Variable #2 (8000 ~ 8999): 1,000 units － prohibiting a user from using the writing function

Note) Version MCSA - V22.35.1(80A)/V22.45.1(80P) or higher is applied.

L Variable	Details	Remark
L8000 ~ L8189	Non-volatile system L Variable	
L8190 ~ L8999	Volatile system L Variable (8190~8499/8500~826/..)	

L Variable	Specific Details	Remark
L8000	Processing quantity count	
L8001	Operation time	
L8002	Total power input time	
L8003		
L8004	Previous L variable block parameter number	
L8005		
L8006		
L8007		
L8008		
L8009		
L8010	Tool frequency	
L8180 ~ L8189	Reserved (regarding alarm)	
8190~8193	AD IN value (only MCSA-80A)	

8496~8499	Reserved (L Variable block parameter function)	
8500~8507	The present coordinate of each axis	

3) Q Variable (Contact Point): Range 0.0~ 911.F

Type: Q000.0 ~ Q911.F
Operators Used: A (bit AND), O(bit OR), ^,+,-,=, (,)

It is possible to calculate a bit/integer word (16 bits) and a **hexadecimal value**.
However, in case of displaying a hexadecimal value, '**0**' should be inserted **in front of a n umerical value** with the '**H**' **in the end** to be recognized as a hexadecimal value.
Ex.) Q200 = Q201 A 01AD2H
Q201.0=1

Q000.0 ~ **Q**099.F: It is the contact point for which only the Read function is possible and the Write fu nction is impossible in the MC program. The area **I00.0 ~ I99.F** is read.

Q100.0 ~ **Q1**99.F: It is the contact point outputted to PLC in the MC program. It is the same as the **Q 00.0 ~ Q99.F** area.

Q200.0 ~ **Q**299.F: It is the contact point connecting the MC program with PLC. The registration or stat e is read in the **M00.0 ~ M99.F** area.

Q300.0 ~ **Q**305.F: Only the Read function is possible in the MC program. It is a contact point. It is th e same as the H/W input area **X00.0 ~ X05.F**.

Q306.0 ~ **Q**311.F: It is the I/O contact point in the MC program. It is registered in the H/W output ar ea **Y00.0 ~ Y05.F.**

Note) **The Y output control method** not using Q306.0 ~ Q311.F but **using the PLC program**
Ex.) Applying the PLC/motion program by mapping M0.0 with Y0.0
[**PLC Program**] [**Motion Program**]   -> If '1' is inputted in Q200.0(PLC의 M0.0)
the PLC program outputs '1' for Y0.0.
LOAD M0.0   Q200.0=1
OUT Y0.0

- Refer to the "**9.3 System Memory Map**" for the detailed functions of **other Q areas**.
Note) Version MCSA - V22.35.1(80A)/V22.45.1(80P) or higher is applied.

MC Program		PLC	Remark	User	
Q000.0 ~ Q099.F		I00.0 ~ I99.F	MC -> PLC	READ Only	
Q100.0 ~ Q199.F		Q00.0 ~ Q99.F	PLC-> MC	READ / WRITE	
Q200.0 ~ Q299.F		M00.0 ~ M99.F	Auxiliary contact	READ / WRITE	
Q300.0 ~ Q305.F		X00.0 ~ X05.F	X area (H/W input)	READ Only	
Q306.0 ~ Q311.F		Y00.0 ~ Y05.F	Y area (H/W output)	READ / WRITE	
Q312.0 ~ Q321.F		F00.0 ~ F09.F	F area (Flag)	READ Only	
Q322.0 ~ Q323.F		T00.0 ~ T01.F	T area (Timer)	READ Only	
Q324.0 ~ Q325.F		C00.0 ~ C01.F	C area (Counter)	READ Only	
Q400.0 ~ Q783.F	⇔	D0 ~ D511	D area (16-bit user integer)	READ / WRITE	
Q784.O ~ Q815.F			T area (Timer set value)	READ / WRITE	
Q816.O ~ Q847.F			T area (The present value o f the timer)	READ / WRITE	

Q848.O ~ Q879.F			C area (Counter set value)	READ / WRITE	
Q880.O ~ Q911.F			C area (The present value of the counter)	READ / WRITE	

Examples
Q200.0=1   -- It makes M0.0 become 'ON'. N1 IF Q210.0=1 N2   -- If M10.0 is 'ON', it jumps to N2. Otherwise, the next block is executed. G4 X0 GOTO N1   -- If M10.0 is not 'ON', it jumps to N1. It waits until M10.0 becomes 'ON'. N2 G00 X0 Y0;

Note) In case of executing the above command, it might night be synchronized with the previous motion command.

Ex.) G00 X100.0
Q200.1=1

- In case of execution, 1 can be in the Q Variable 200.1 before the X axis reaches 100mm. At this time, if execution is desired after the motion command is completed, input the G04 command in front of the command Q or use the expanded Q Variable order function below the lower part.

* The Expanded Q Variable Order: QT

QT****: As for an order, the program line is executed after the motion command read first is executed before the command.
Ex.) QT200.1=1 is the same as G04 X0.0.
Q200.1=1

Note) Relevant Parameters
1. P800.7   -> The contact point M backup function

4) Conditional Branch Commands (IF, IFT, GOTO , GOTO1, N)

(1) IF, IFT – They are the commands deciding and branching the conditions of the L Variable or Q.

Conditional Formulas Used: In case of **L variable** condition formula: .EQ(=), .LE(<=), .LT(<), .GE (>=), .GT(>), .NE(<>)

In case of a **Q variable** condition formula: =, <>

Comparison/Decision Command		Explanation
= , .EQ	=	Equal to
.LE	<=	Less than or equal to
.LT	<	Less than
.GE	>=	Greater than or equal to
.GT	>	Greater than
.NE	<>	Not equal to

Ex.) 1. IF L5 .LT 123 N100;

Explanation – If L5 is less than 123, the N number 100 is searched from the start of the program executed. If it exists, it jumps to N100 for execution. Otherwise, an alarm is given.
If L5 is greater than or equal to L5, the next block is executed.

2. IF Q200.0 =1 N100;

Explanation – If Q200.0(M0.0) is equal to 1, the N number 100 is searched from the start of the execution program. If it exists, it jumps to N100 for execution. Otherwise, an alarm is given.
If Q200.0(M0.0) is not equal to 1, the next block is executed.

Note) In case of executing the above command, it might night be synchronized with the previous motion command.
Ex.) G00 X100.0
IF L100=1 N100
- The IF branch command can be executed before the X axis reaches 100mm during execution. As for the execution after the completion of a motion command, input the G04 command in front of the IF command or use the expanded IF command order function below the lower part.

* Expanded IF Command Order: IFT

IFT ****: As for an order, the program line is executed after the motion command read first is executed before the command.
Ex.) IFT L100=1 N100 is the same as **G04 X0.0**.
IF L100=1 N100

(2) GOTO , GOTO1

Type: GOTO N___
The N number ___ is searched from the start of the execution program to the end. If it exists, it jumps to ___ for execution. Otherwise, an alarm is given.

Type: GOTO1 N___
The N___ is searched from the present line to previous lines. If the N___ exists, it jumps to ___ for execution. Otherwise, an alarm is given.

5) WHILE Statement (Conditional Repetitive Execution)

Type: WHILE <Conditional Formula> DOm (m=1,2,3···)

       ~

       ENDm (m=1,2,3···)

If <Conditional Formula> is true, the block after Dom to the block before ENDm is executed repetitively.

If <Conditional Formula> is false, the block after ENDm is executed.

WHILE <Conditional Formula> is the same as the IF command and can be omitted. If omitted, DOm to ENDm is repeated infinitely.

WHILE <Conditional Formula> DOm and ENDm are recognized in pairs. The identification number m enables the mutual identification.

Ex.) L120=0

   WHILE L120 .LT 100 **DO1**   -> After the program between WHILE and END1 is executed 100 times, the block after END1 is executed.

```
      ~
   L120=L102+1
```

   END1

   ~

☞ When the program to be repeated is prepared, **be sure to observe** the following matters.

1. DOm must be ordered first. ENDm must be ordered later.

```
...
 END 1   (Error)
...
 DO1
...
```

2. DOm and ENDm must be one-to-one correspondent in the same program

```
...
 DO1
...
 DO1    (Error)
...
 END1
```

3. The same identification numbers can be used several times.

```
 DO1
 ...
 END1
 ...
 DO1            (Possible)
 ...
 END1
```

4. The multiplicity of DO is triple at most.

```
..
DO1
..
DO2
..
DO3
..
END3
..
END2
..
END1
```

5. The DO range **cannot** be crossed.

```
..
DO1
..
DO2
..
END1    (Error)
..
END2
```

6. A branch can be used outside the DO range.

```
..
DO1
..
GOTO  N10
..
END1
..
N10
```

7. A branch **cannot** be used to the inside of the DO range.

```
..
GOTO  N10
..
DO1
..
N10      (Error)
..
END1
..
```

8. A subprogram or a subroutine can be called within the DO range.

```
..
DO1
..
G65  …
..
M98  …
..
END1
..
```

9. If there is no servo movement command or G04 between DOm and ENDm, insert the G04 X0.0 command between DOm and ENDm. Otherwise, communication becomes offline.

2.6 Main Axis Order (S(Standard #1) / SX(Expanded #2) Code)

2.6.1 S__ , SX_-> Main Axis RPM Order

The coupler is always on so that the driver is not sometimes driven. In this case, wiring is done with the interface method 2.

(In case of the connection with the servo driver, the line driver type wiring is done.)

2.6.1 SS__ , SXS__-> The position movement order of the main axis (However, it is valid when the EnCoder wiring is done in the corresponding port.)
If it comes to the corresponding position on the basis of the C phase, it stops suddenly.
Give the low speed RPM order to the block before the usage of the SS_ command is used.
%) The above command is valid only in MCS-80P (pulse output type).
Relevant Contact Points: I2.6 I2.5
2.6.3 SA__ , SXA__ -> Setting the acceleration time of the main axis (Unit: 1ms, Rpm/Sec)
The larger a value, the faster a response.

2.6.5 SD__ ,SXD__ -> Setting the deceleration time of the main axis (Unit: 1ms ,Rpm/Sec)
The larger a value, the faster a response.

Note) 1. Relevant Parameters Relevant to **S**: P51/52/53 , P801.0/1
Contact Points: Q32.8 / Q03.B~Q03.F / Q31.0~Q31.F / I66

2. Parameters Relevant to **SX**: P804.0/1/2 , P801.2/3
Contact Points: Q32.9 / Q32.B~Q32.F / Q33.0~Q33.F / I67

2.7 T Code Order (T Code)
Give an order with a single block.
-> The tool offset (tool adjustment / 10 sets) function is executed.
Note) 1. Only Version V23.10.1(80A) / V23.20.1(80P) or higher is supported.
2. The relevant parameter P800.AXIS1 (L Variable correspondence start number)

T00 -> Tool Offset Adjustment Cancellation
T01 -> T01 Coordinate Adjustment
..
T09 -> T09 Coordinate Adjustment
T10 -> T10 Coordinate Adjustment

2.8 Examples of Program Usage

2.8.1 Example of Usage #1 - Company H Tool Tip Pick & Place Equipment Application Program

```
/* START
N1
FVL302
L8=1
G90
G83 X3.0
G83 X3.1
G00 Z0
/* CLAMP CHECK
IF Q200.2=1 N2

L8=18
G01 ZL101 FL103
G01 XL100 FL103

/* PLATE EXIST
/*G80 X4.2
L8=17
G80 M0.4

/*GIRPER OPEN
G66 N100

L8=18
/*SKIP MOVE FOR PLATE LOADING
G31 X3.F
G01 Z560.0 FL300

/*GIRPER CLOSE
G66 N200
N2

/* UNLOADING POSITION
L8=18
G01 ZL111 FL113
G01 XL110 FL113

L8=11
/* START FROM PICK UP
/*G80 X4.8

N10
IF Q213.D=1 N998
G04 X0.2
IF Q204.8=1 N11
G04 X0.2
GOTO N10
N11
```

```
/* SKIP FOR PLATE
G31 X3.F
G01 Z95.0 FL301
/* CLAMP TO PICK UP
G82 X3.0
L8=21
/* FIN CLAM FROM PICK UP
G80 X4.9
G83 X3.0
/*GIRPER OPEN
G66 N100

L8=18
/* BACK MOVE
G00 Z0
/* END TO PICK UP
G82 X3.1
G04 X1.0
IF Q213.D=1 N998
GOTO N1
N998
G83 M10.E
G83 M13.D
G00 Z0
L8=0
M02

N100
/*GIRPER OPEN
L8=19
G04 X0.0
G83 X2.3
G82 X2.4
/* GRIPER OPEN CHECK
G80 X3.E
G81 X3.D
G04 X0.5
M02

N200
L8=20
/*GIRPER CLOSE
G04 X0.0
G82 X2.3
G83 X2.4
/* GRIPER CLOSE CHECK
G81 X3.E
G80 X3.D
G04 X0.5
M02
```

2.8.2 Example of Usage #2 – Company D Wire Cutting/Coating Stripping and Compression Equipment Application Program

```
Q246.6=0
Q245.0=0
Q246.4=0
Q246.5=0
Q245.1=1
/*  DEL
L132=L100-1
L133=L110-1
L111=0
L125=0
L131=0
L503=10000
Q221.F=0
Q208.A=0
/*Q217.F=0
Q208.B=0
G04  X0.0
Q213.1=0
Q206.5=0
G04  X0.01
Q237.1=0
/*Q209.0=1
Q211.4=0
Q242.7=0
Q209.2=0
Q208.E=0
Q208.F=0
Q210.5=0
Q208.0=0
Q208.1=0
Q208.6=0
Q208.7=0
Q230.0=0
Q230.1=0
Q230.2=0
Q230.3=0
Q219.0=0
Q231.2=0
Q231.3=0
Q231.8=0
Q231.A=0
Q231.B=0
Q231.C=0
Q233.0=0
Q233.1=0
Q233.6=0
Q234.2=0
Q234.3=0
Q234.4=0
```

```
Q234.5=0
Q234.6=0
Q235.0=0
L205=L6
L202=L22+L28
G65  P43
/*SEAL
L44=L160+L145
L502=-17000
L501=L160/2
L135=13000
L124=L135
L134=L135
L6=L7
/*HALF  TALPI
IF  L40  .GT  3  N102
GOTO  N103
N102
L6=0-L31
N103
/*13MM
IF  L3  .GT  L135  N101
/*SEAL
IF  Q213.0=1    N101
G04  X0.0
GOTO  N100
N101
/*13MM
Q235.0=1
L124=0
L6=0
/*HALF  TALPI
IF  L40  .GT  3  N112
GOTO  N113
N112
L6=0-L31
L117=0-L31
N113
N100
/*  INIT
Q211.0=0
FV10
/*MANUAL
IF  Q211.9=1  N40
/*ZRN
Q206.4=0
G04  X0.1
G65  P36
FF71
Q208.B=0
Q208.9=1
```

```
G28  Z0  U0  W0
G28  V0
G00  UL126  ZL127
G28  Y0
G51  U0  X0  Z0
G80  X4.6
/*SEAL  INIT
IF  Q213.0=0  N30
G65  P14
N30
/*VEL
G65  P31
G04  X0.0
Q237.1=1
G00  XL0  FF71
G51  X0
G80  M10.0
Q237.1=0
ZL1
G80  M10.0
G04  X0.1
G00  Z0  FF71
G80  M10.0
/*  SEAL
IF  Q213.0=0  N21
G65  P13
G04  X0.3
AL501
Q208.3=0
G04X0.2
Q208.2=0
G04  X0.2
Q208.3=1
G04  X0.3
G65  P49
G04  X0.3
Q208.3=0
G04  X0.1
Q208.2=1
G04  X0.3
Q208.3=1
G04  X0.3
N21
L2=L3+L34-L124
L121=L2
G00  XXL2  FX10000
FF71
/*COMMON
IF  Q235.0=1  N120
G65  P30
G80  M10.0
```

```
G00  XXL6  FF81
L121=L121+L6
GOTO  N121
N120
G65  P40
N121
G80  M10.0
G00  Z0  FF71
/*  FRONT  PRESS
IF  L40  .GT  1  N10
G80  M10.0
Q233.0=1
Q233.1=1
YL8
G80  M10.0
L121=L9
XL9  FX10000
X-18
G80  M10.0
/*  PRESS
G04  X0.0
Q208.0=1
Q208.E=1
/*  FRONT
G81  M8.E
G04  X0.2
Q230.0=1
G04  X0.1
Q230.0=0
G81  M34.0
Q208.0=0
Q208.6=0
L146=L146+1
N10
Y0
G80  M10.0
FVL80
FF71
L12=L11+L33-L121
L16=L15+L35-L134
Q242.A=0
/*  Work  START
G65  P46
Q245.1=1
G65  P11
Q217.8=0
Q213.1=1
Q217.0=0
Q211.4=0
Q213.9=0
Q237.1=0
```

```
L111=0
Q217.F=0
Q211.7=0
Q233.6=0
Q234.2=0
Q234.4=0
Q234.6=0
Q242.7=0
M02

/*MANUAL
N40
Q217.8=1
G65  P20
M02
```

Chapter 3 PC MMI

3.1 MSW-MCS (for PC)

The MSW-MCS program is the program for Windows provided by Emotiontek to operate the MCS remotely in PC through a communication port (COM).
The program can edit the MCS operation programs and operate each operation mode of the MCS.
Moreover, the operation parameters of the MCS can be set. The internal variable (L variable) and the PLC contact points can be searched and set.

1) Installing the MSW-MCS

The MSW-MCS program can be downloaded from the Emotiontek site (www.emotiontek.com).
If the installation program is executed in the PC (for MCS operation) after downloading the installation program, the MSW-MCS program is installed.

Downloading the Installation Program

- Connect to the Emotiontek site (www.emotiontek.com).
- Connect to "Information" >> "Motion Controller" >> "MCS" >> "MMI".
- The installation program being suitable to the operation environment of the PC in which the MSW-MCS is to be installed is downloaded.

MSW-MCS Motion Operating Environment

- 32-bit MS Windows OS that is MS Windows 98SE or higher
- The PC with a communication port (COM) installed

Executing the Installation Program

- The PC in which the MSW-MCS is to be installed should satisfy the requirements of "The MSW-MCS Motion Operating Environment" explained above.
- The compressed installation program is decompressed.
- The MSW-MCS installation program is executed.
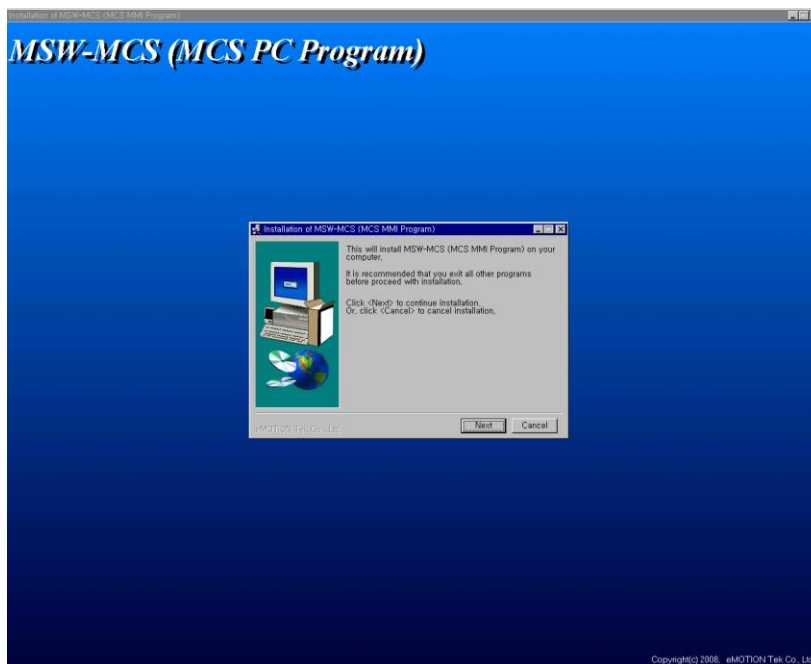
- If the installation is executed according to the instructions of the dialog box appearing while the install ation program is being executed, the program installation is completed.
- Reboot the PC after the execution of the installation program is completed.
- The MSW-MCS icon is created in the desktop. Double-clicking the icon has the MSW-MCS program e xecuted.

(Note) The MSW-MCS installation program is the program prepared to be executed in a 32-bit Windows OS.
A problem may occur if it is installed in a 64-bit Windows OS.

(Note) In order to reinstall the MSW-MCS program, install it after deleting the MSW-MCS program insta lled before.

2) Connecting the MSW-MCS

The MSW-NCS program is the program used for remotely operating the MCS and communicates with the MCS through the PC communication port (COM).

(1) Communication Cable Wiring Diagram

The cable wiring diagram for the communication with the MCS is as follows.

```
        PC                                      MCS-80
     RS-232C                                    RS-232C

   2  RXD ○────────\    /──────────○ 2  RXD
                    \  /
   3  TXD ○──────────\/───────────○ 3  TXD
                     /\
                    /  \
   5  SD  ○────────────────────────○ 5  SD
```
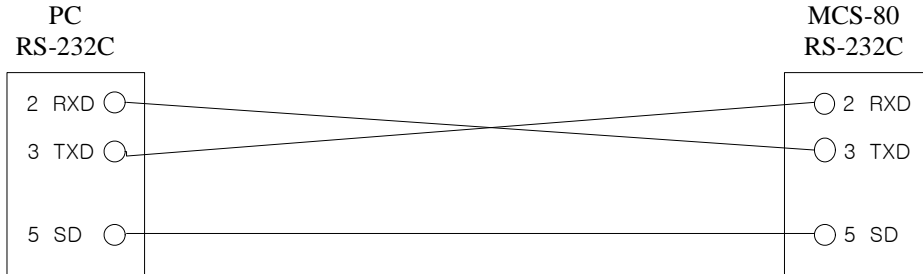
Note) Wiring into the pin other than pins 2, 3, and 5 may impact the PC port or the MCS port. Don't wire into the other pins.

(2) Connecting the Communication Cable

The PC communication port (COM) is connected to the MCS RS232 port to execute the MSW-MCS program. If there is no error in the communication setting and wiring between the PC and the MCS, the MSW-MCS program is operated normally.

If there is no communication response from the MCS due to a communication error, it is notified to the user that a communication problem occurred between the MSW-MCS program and the MCS.
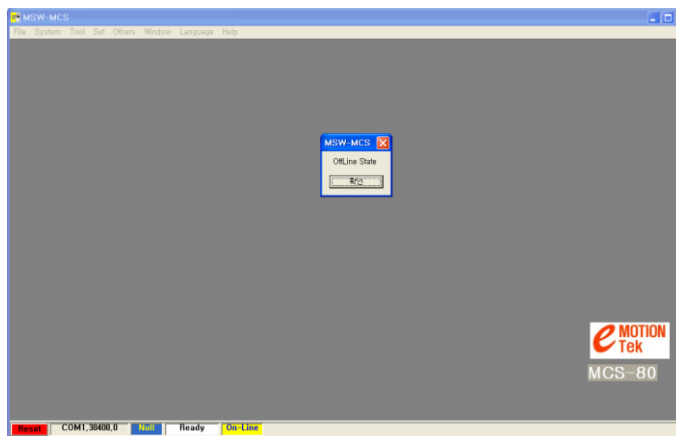
The communication failure message box appears as above. In this case, execute it again after removing the cause occurring communication errors. The causes that may generate communication errors are as follows.

- Power is not supplied to the MCS.
- The communication cable is disconnected or wired incorrectly.
- PC communication port setting error

If there is an error in the PC communication port setting, reset the communication port with the following method.

If the program is installed first, the PC 'COM1' port is set as the communication port.

The example of changing a communication port to the 'COM2' port is as follows.

- The communication bar set currently can be checked in the status bar in the lower part of the MSW-MCS main form.

`COM1,9600,8,n,1`

- The Communication Port Setting window is selected through the MSW-MCS menu bar.
  In the Main menu, the "Setting" >> "External Communication" >> "Data Communication" menu is selected.

```
JOB Manager

Basic Parameter
Extended Parameter
String Parameter

PLC Parameter

External COM        ▶   COM
```
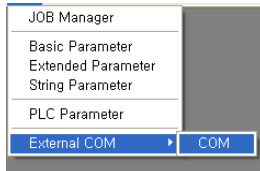
- "Port No." is selected as the "COM2" port in the Communication Port Setting window.

```
Set COM Port                    ×
Port No. COM2         ▼
Waiting Value
1000        ▼   msec

   OK            Cancel
```
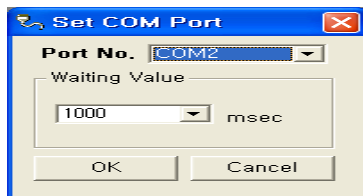
(Note) Do not change the other communication specifications. It may cause an error.

The OK! button is selected.

- It can be checked that the communication port was changed to the 'COM2' port in the status bar.

`COM2,9600,8,n,1`

The communication with the MCS is performed through the changed communication port.

If communication is performed normally, it shows the normal execution in the status bar of the MSW-MCS.

```
MSW-MCS [Unknown]                              _ □ ×
File  System  Tool  Set  Others  Window  Language  On-Line  Help




                                              OTION
                                              k
                          Display of Communication Status   80
                          Normal: On-Line
                          Abnormal: Off-Line

Reset  COM1,38400,0  Null  Ready  Off-Line
```
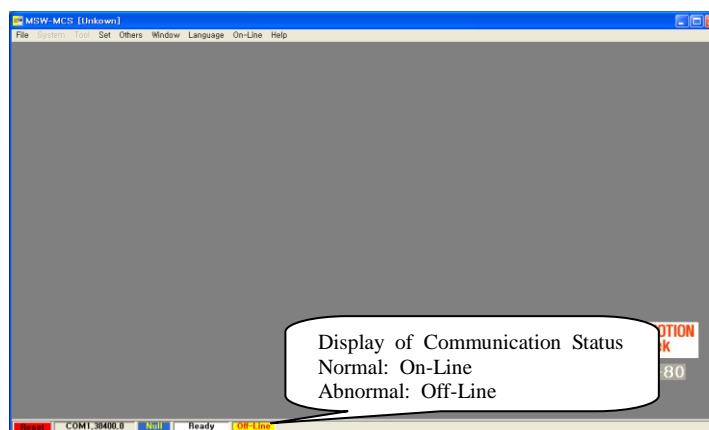
3) Configuring the MSW-MCS

The MSW-MCS program consists of the title bar, menu bar, status bar, and execution area like the basic form organization of a general Windows program.
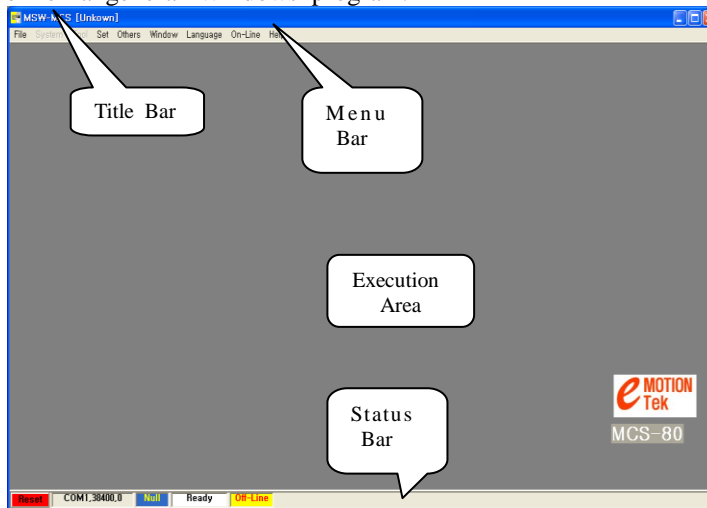
Title Bar: A program name and the type of the connected MCS are displayed.
Menu Bar: It is organized with the menus executing the panels operating the MCS.
Status Bar: The communication port specification and the operating and communication status of the MCS can be checked. The reset button is also included.
Execution Area: It is the area where various panels are executed.

(1) Title Bar

The title bar located in the upper part of the main form consists of the program icon, program name, and Windows system button and displays the type of the connected MCS.
The MCS types to be displayed are as follows.

Controller Type	Explanation
MCS-80A	It is the MCS series and the analog output type [speed control type] controller.
MCS-80P	It is the MCS series and the pulse output type [position control type] controller.
MCSA-80A	It is the MCSA series and the analog output type [speed control type] controller.
MCSA-80P	It is the MCSA series and the pulse output type [position control type] controller.
Unknown	Unknown

(2) Status Bar

The status bar located in the lower part of the main form displays the setting value of the communication port, the communication status, and the operation mode and operating status of the MCS. Moreover, the reset button is located so that the MCS reset can be executed.

- **Reset** : The reset button is selected to execute the MCS reset.

- `COM1,9600,8,n,1` : The communication port setting value used by the MSW-MCS is displayed. In case of a communication failure, the area blinks.

- `AUTO` : The current operation mode of the MCS is displayed. The real-time search is done in the MCS.

Displayed Value	Explanation
AUTO	Automatic program operation mode
JOG	Manual jog operation mode
STEP	Manual step operation mode
ORIGIN	Manual return-to-origin operation mode
MPG	Manual MPG operation mode
MDI	Automatic MDI operation mode

- `Ready` : The motion status of the MCS is displayed. If the MCS is in the alarm status, the alarm number is displayed. Refer to "Chapter 8 Alarm" for further details on the alarm number.

Displayed Value	Explanation
READY	Normal waiting status
RUN	Automatic operation status
USER	User message output and user message number output
WARNING	Warning status and warning number output
ERROR	Alarm status and alarm number output

- `On-Line` : The communication status of the MSW-MCS is displayed. In case of a communication failure (off-line), the area blinks.

(3) Menu Bar

File   System   Tool Setting   More Functions   Window Help

The operation parameter and program of the MCS can be managed in the MSW-MCS menu bar. The panels operating the MCS are executed. The menu is classified for each function. There are submenus below each main menu.

- `File` Menu: The editing panel managing the files (MC and PLC files) saved in the PC is executed.
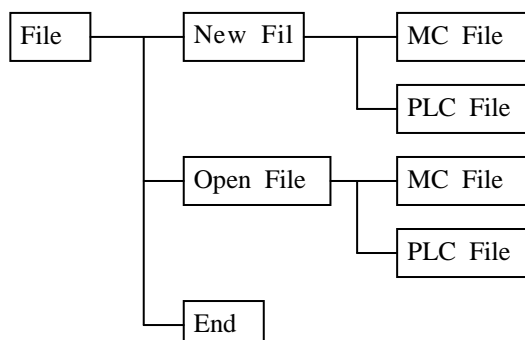
- System Menu: The editing panel managing the MC and PLC programs saved in the MCS and the management panel used for system backup and initialization are executed.

```
System ─── New Progra ─── MC Program
                      └── PLC Program ─── SCAN Program
                                      └── Interrupt Program
       ─── Open Program ─── MC Program
                        └── PLC Program ─── SCAN Program
                                        └── Interrupt Program
       ─── Back-Up ─── Save
                   └── Restore
       ─── Initialize ─── L Variable
                      └── Parameter
```
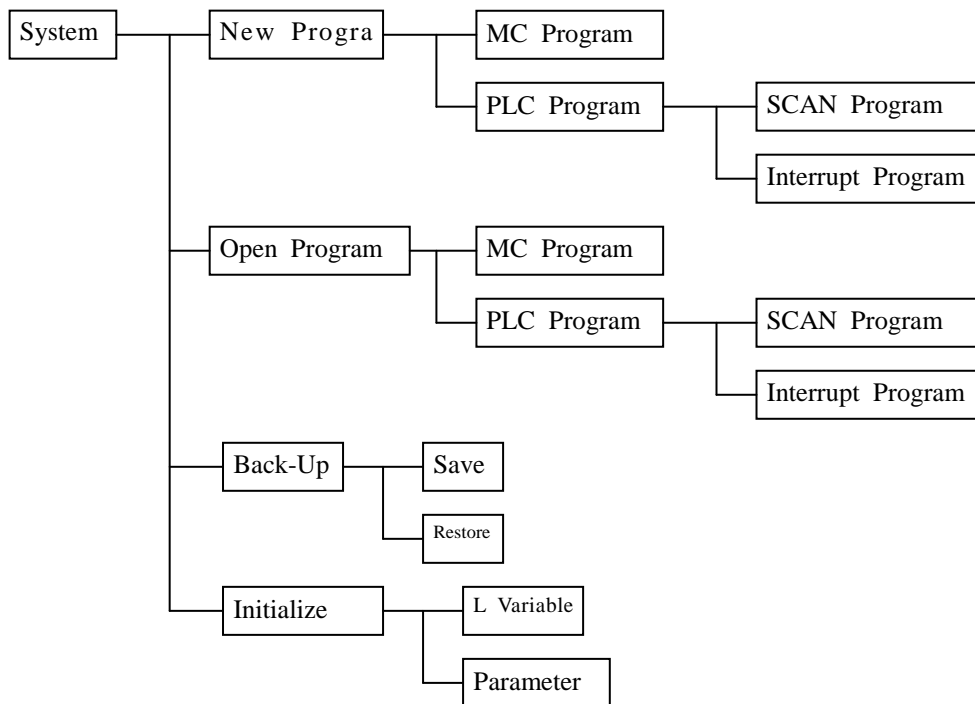
- Tool Menu: The system panel and the operation panel for viewing all coordinates, program finder, L variable finder, and so on are executed.

```
Tool ─── System Panel
     ─── View All Coordinates
     ─── Program Finder
     ─── L Variable Finder 1,2,3
     ─── Point Teaching
     ─── Axis IO Finder
     ─── PLC Info. Finder 1,2,3
     ─── Terminal
```
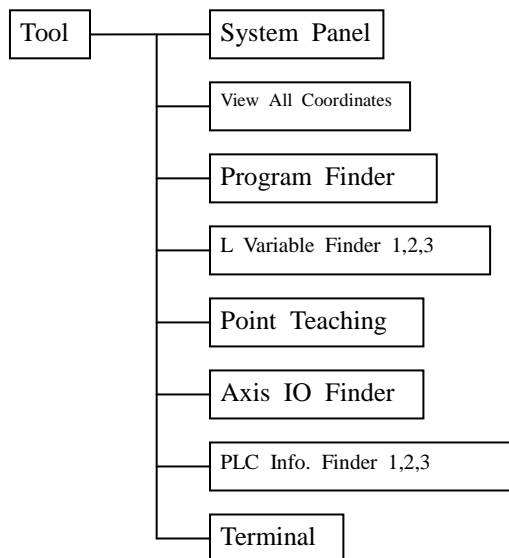
- **Setting** Menu: The parameter panels setting the parameter, PLC parameter, communication variable, and so on are executed.

```
Setting ┬─── Standard  Parameter

        ├─── Expanded  Parameter

        ├─── String  Parameter

        ├─── PLC  Parameter

        └─── External  Communication ──── Data  Communication
```
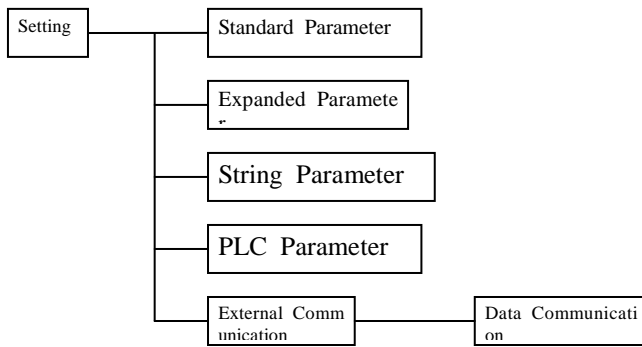
- **More Functions** Menu: The operation panels with the alarm history, analog input finder, and so on added are executed.

```
More Functions ┬─── Alarm  History

               ├─── Spindle  Finder

               └─── Analog  Input  Finder
```
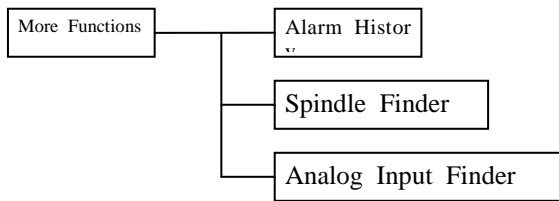
- **Window** Menu: The list of the window currently executed is displayed and sorted.

- **Help** Menu: The MSW-MCS program and the MCS information are displayed.

4) Setting the Parameters

In order to operate the MCS, The characteristics (movement distance for each rotation, number of pulses for each rotation, maximal movement speed, etc.) values of the equipment with the MCS mounted should be set in the MCS.

These equipment characteristics values and the standard values needed for MCS operation are known generically as parameters.

Refer to "Chapter 5 Parameters" for the detailed explanation on parameters.

It explains how to set the parameters of the MCS by using the MSW-MCS.

(1) Standard Parameter

If the "Standard Parameter" menu below the "Setting" menu is selected, the Standard Parameter setting window is executed.
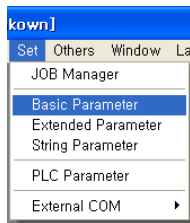
The standard parameters saved in the MCS or a PC file can be imported or saved by using the Standard Parameter setting window. If the window is executed by selecting the "Standard Parameter" menu, the window is organized by importing the standard parameters saved in the MCS.
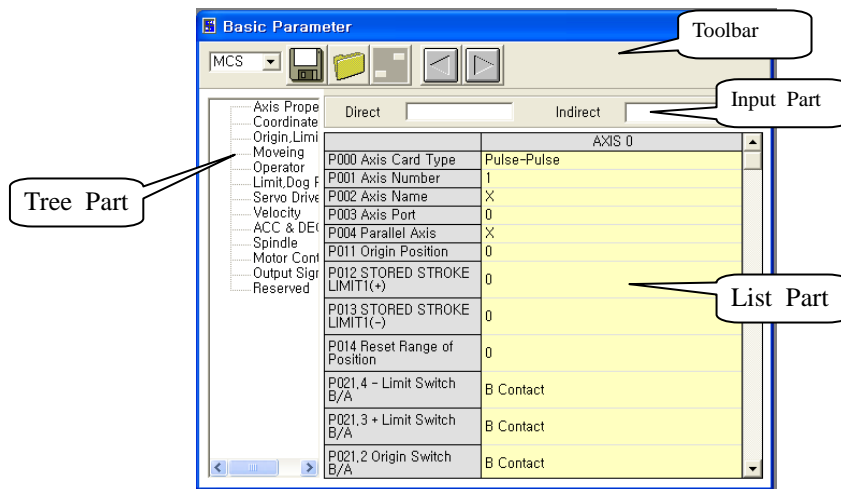
The Standard Parameter setting window consists of the toolbar, the input part, the list part, and the tree part.

The parameter values can be checked through the list part. The parameter value is changed through the input part.

The tree part shows the parameters in the tree type of the names classified for each function. If a name is selected in the tree part, the parameter with the selected name is shown.

A parameter can be read or saved through the toolbar.

① Changing the Parameters

- The item to be changed is selected in the list part.
- The input part is changed suitably to the selected parameter input type.

In case of the

direct input

parameter for which a number should be inputted, it is possible to input it in the direct input 직접입력 [99999999] part.

In case of the parameter that should be selected from the list, the details of the indirect input indirect   A Contact_point are organized suitably to the parameter and can be selected.

- A parameter is set through the input part.

In case of the direct input type, press the 'Enter' key after inputting a numeric value in the in put box.

In case of the indirect input type, select an input value through the selection list.

Saving the Parameters

A changed parameter can be saved in the MCS or PC.

Saving a parameter in the MCS

- Select the storage location with the `MCS ▼` box.
- If the 🖫 button is selected, the message box confirming the storage appears.
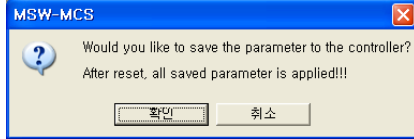
- If the `Confirmation` button is selected, the parameter is saved in the MCS.
  (Note) If a parameter is saved in the MCS, the value is not reflected directly. If the MCS reset is executed, the changed parameter is applied.

(Note) Specific parameters are reflected when power is supplied to MCS again.

Saving a parameter in the PC

- Select the storage location with the `PC ▼` box.
- If the 🖫 button is selected, the message box confirming the storage appears.
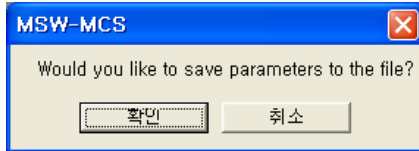
- If the `Confirmation` button is selected, the dialog box with which a storage file can be selected appears.
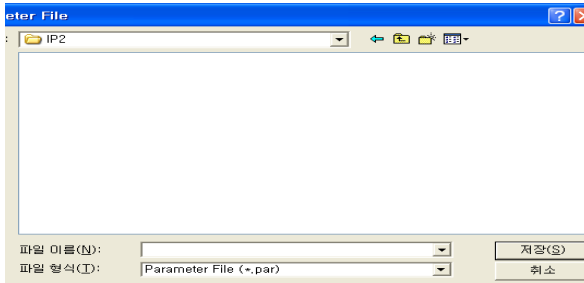
- If the `Save` button is selected after file selection, the parameter is saved in the file.

② **Reading the Parameters**

A parameter can be read from the MCS or PC.

Reading the parameters from the MCS

- Select the reading location with the `MCS ▼` box.
- If the 🖿 button is selected, the message box confirming reading appears.
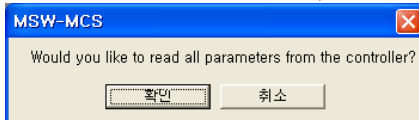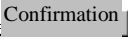
- If the `Confirmation` button is selected, the parameter is read from the MCS, and the displayed value in the list part is renewed.

Reading the Parameters from the PC

- Select the reading location with the `PC ▾` box.
- If the ☐ button is selected, the message box confirming reading appears.
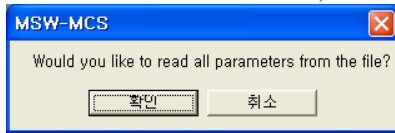
- If the `Confirmation` button is selected, the dialog box with which the file to be read can be selected appears.
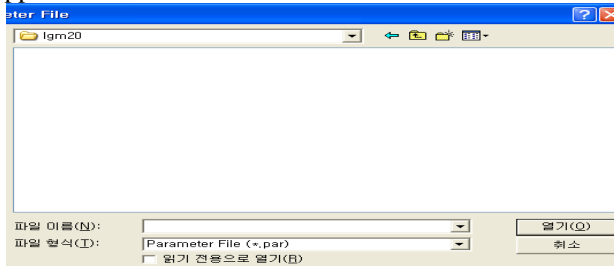
- If the `Open` button is selected after file selection, the parameter is read from the file, and the displayed value in the list part is renewed.

The ◁▷ button is used to decrease or increase the number of axes displayed in the list part.

(2) Expanded Parameters

If the "Expanded Parameter" menu below the "Setting" menu is selected, the Expanded Parameter setting window is executed.
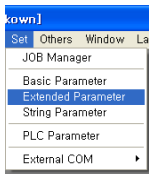
The expanded parameters saved in the MCS or a PC file can be imported or saved by using the Expanded Parameter setting window. If the window is executed by selecting the "Expanded Parameter" menu, the window is displayed by importing the expanded parameters saved in the MCS.
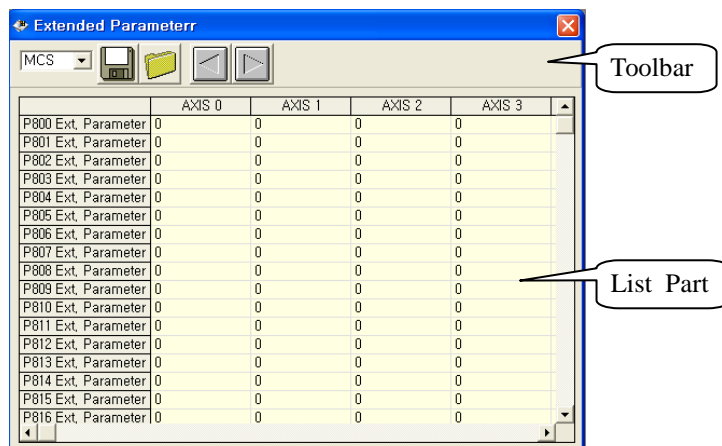
The Expanded Parameter setting window consists of the toolbar and the list part. A parameter value is changed through the list part. A parameter can be read or saved through the toolbar.

Changing the Parameters

- The parameter value to be changed in the list is double-clicked.
- The input box [   ] appears for a selected parameter.
- The parameter value to be changed is inputted in the input box.
- Press the 'Enter' key after inputting a numeric value in the input box.

① **Saving and Reading the Parameters**

The expanded parameter that is changed can be saved in the MCS or PC. The method is the same as the method of saving and reading the standard parameter. Refer to the method of saving and reading the standard parameters.

(Note) If an expanded parameter is saved in the MCS, the value is not reflected directly. If the MCS reset is executed, the changed parameter is applied.

The [◁▷] button is used to decrease or increase the number of axes displayed in the list part.

5) Operating the MCS

The MCS is operated by using the concept of the operation mode. The operation mode is classified into the automatic operation mode operated by the automatic movement command (Chapter 2 Programming) and the manual operation mode operated with the manual manipulation. Furthermore, the automatic and manual operation modes are classified into various operation modes as shown in the following table.

Automatic Operation Mode	Manual Operation Mode
AUTO mode MDI mode	JOG mode STEP mode ORIGIN mode MPG mode

Each operation mode is selected and operated through the system panel executed through the "System Panel" menu below the "Tool" menu.
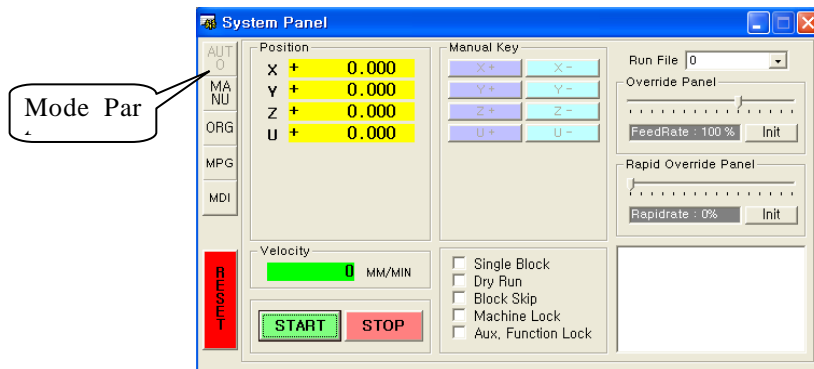
The operation mode is selected through the mode part of the system panel. The button of the current operation mode of the MCS is displayed as being deactivated.

(1) ORIGIN Mode

The ORIGIN mode is the manual operation mode used to execute the return to the origin. In order to operate the operation mode, the parameters needed for the operation of the ORIGIN mode should be set first. The parameters related to the ORIGIN mode are as follows.

P21.1 : Existence of origin
P21.2 : Origin direction +/-
P111 : How to return to the origin
P21.4, P21.3, P21.2 : - Limit Switch B/A, + Limit Switch B/A, and Origin Switch B/A
P24, P25 : Limit Choice and Dog Choice
P11 : Return-to-origin coordinate value
P36, P110, P37, P38 : JOG speed, return-to-origin speed, return-to-origin speed 1, and return-to-origin speed 2
P44 : Acceleration and deceleration of the return to the origin

Refer to "Chapter 5 Parameters" for the detailed explanation on the above parameters.

If the [The Return-to-Origin] button of the mode part is selected, the MCS operation mode is changed to the ORIGIN mode. If the MCS is changed to the ORIGIN [The Return-to-Origin] button is displayed as being deactivated.
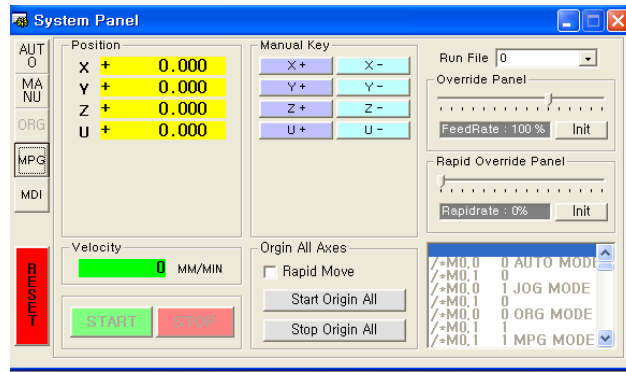
[ORIGIN Mode]

① The Return-to-Origin of Each Axis

- The manual operation key of the axis performing the return to the origin is selected.
  Select the manual operation key that is the same as the parameter direction set in **P21.2**.
  Select the [X -] button if the - direction is set. Select the [X +] button if the + direction is set.
- Continue to select the manual operation button while the return to the origin is being executed. If the selection is released, the return-to-origin motion is cancelled.
- If the return to the origin is completed, the axis movement is stopped. The coordinate value is displayed as the value set in **P11**.

② The Return-to-Origin of All Axes

- I{n order to return all axes to the origin, sele[The return to the origin of all axes is started.] button.
- All axes start the return-to-origin movement to the direction set in **P21.2.**
- The axis movement is stopped in the sequence with the return-to-origin completed. As for the coordinate value of each axis, the value set in **P11** is displayed.
- The return-to-origin motion of all axes is to be cancelled, sele[The return to the origin of all axes is stopped.] button.

③ Auxiliary Function

- [□ Rapid Move] : If the checkbox is selected, the movement-to-DOG speed is changed to the rapid movement speed.

(2) JOG Mode

The JOG mode is the manual movement operation mode used while the manual movement order is being given.
If the [Manual] button is selected of the mode part, the system panel is changed suitably to the manual operation mode. If ...t[JOG mode] button is selected, the MCS operation mode is changed to the JOG mode.
If the MCS is changed to the JOG mode [Manual] button is displayed as being deactivated.
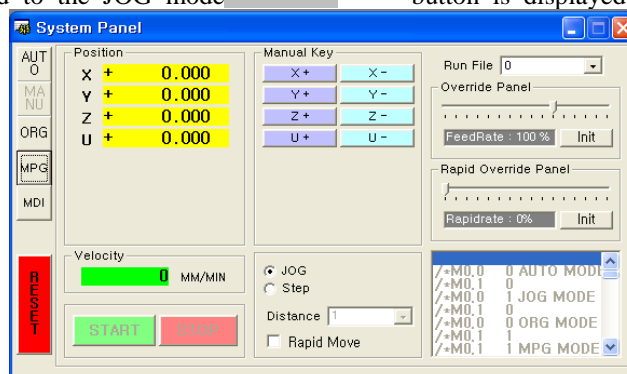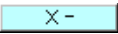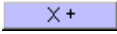
[JOG Mode]

① JOG Operation
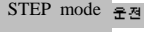
- Select the manual axis operation key for JOG operation.
  Select the [X -] button if the JOG operation is performed in the - direction. Select the [X +] button if the JOG operation is performed in the + direction.
- The axis movement is performed while the manual operation key is being selected.
  The axis is moved at the speed set in the **P36** JOG speed parameter.
- The axis movement is stopped if the selection of the manual operation key is released.

② Auxiliary Function

- [□ Rapid Move] : If the checkbox is selected, the JOG movement speed is changed to the rapid movement speed.

(3) STEP Mode

The STEP mode is the manual operation mode for moving a certain distance.
If the [Manual] button of the mode part is selected, the system panel is changed suitably to the manual operation mode. If [STEP mode 운전] button is selected, the MCS mode is changed to the STEP mode.
If the MCS is changed to the STEP mode, the [Manual] button is displayed as being deactivated.
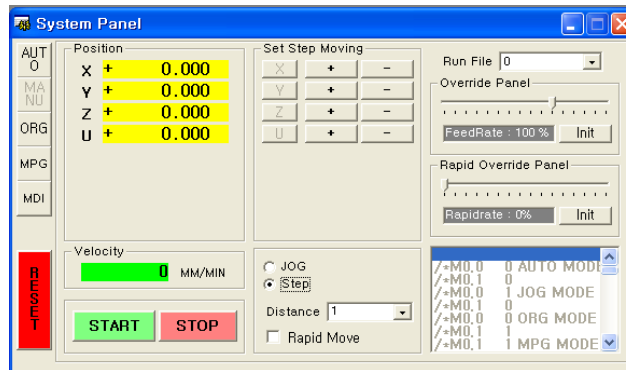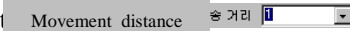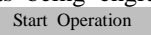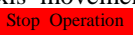
[STEP Mode]

① STEP Operation
- The STEP operation movement distance is set.
  The movement distance is set through t[Movement distance 송 거리 1 ▼] input box. The movement distance can be set in the list. A user can input the movement distance directly. The unit of the movement distance is '㎛' .
- The STEP operation axis and direction are selected.
  In order to perform the operation in the - direction, select the [-] button. In order to perform the STEP operation in the + direction, select the [+] button. A selected direction button is displayed as being engraved. Several axes can be selected at the same time.
- If the [Start Operation] button is selected, the STEP movement of a certain distance in the selected axis direction is performed.
  The axis movement is performed at the speed set in the **P36** JOG speed parameter.
- If the [Stop Operation] button is selected during movement, the movement can be paused. I[Start Operation] [운전 시작] button is selected again, the axis movement is resumed.

② Auxiliary Function

- [□ Rapid Move] : If the checkbox is selected, the STEP movement speed is changed to the rapid movement speed.

(4) MPG Mode

The MPG mode is the manual operation mode for axis movement in proportion with the number of the pulse columns inputted through the MPG port of the MCS.
If the MPG button of the mode part is selected, the MCS operation mode is changed to the MPG mode.
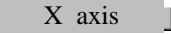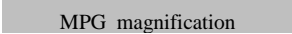
If the MCS is changed to the MPG mode, the MPG button is displayed as being deactivated.
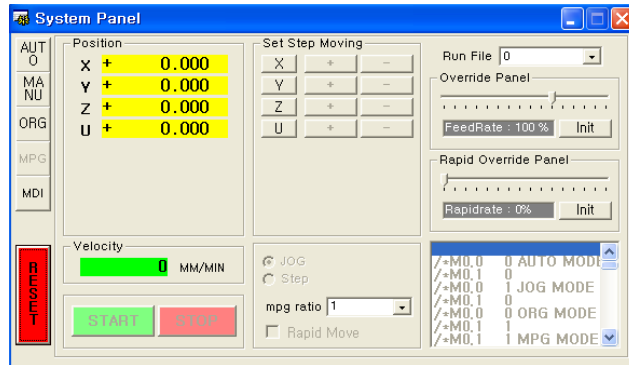
[MPG Mode]

Wire the MPG (Manual Pulse Generator) into the MPG port of the MCS. Refer to "Chapter 6 Connection" for the wiring method.

① MPG Operation

- Select the MPG operation axis.
  Select the operation axis by using the [X axis] buttons. The selected axis is displayed as being engraved.
  Only one axis can be selected.
- The MPG movement magnification is selected.
  The MPG movement magnification is selected through [MPG magnification] selection box.

- The axis is moved through the MPG rotation.

(5) AUTO Mode

The AUTO mode is the automatic operation mode to operate the MCS with the pattern prepared in the MC program.
If the [AUTO] button of the mode part is selected, the MCS operation mode is changed to the AUTO mode.
If the MCS is changed to the AUTO mode, the 자동 button is displayed as being deactivated.
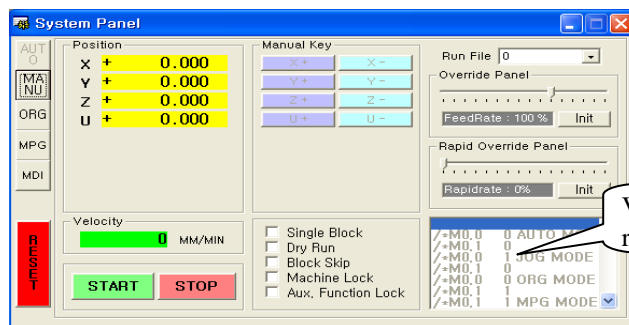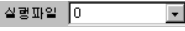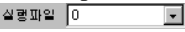
[AUTO Mode]

① Selecting the MC Program
- The MC program to be operated   [Executed File]

[Executed File]

in the AUTO mode is selected by using the ![실행파일 선택박스](selection box) selection box. If the operation status is "READY", it can be selected. If the operation status is "RUN", the box displays the name of the MC program being executed.

- The selected MC program can be checked through the View-Execution-Program part located in the lower right corner of the system panel.

② AUTO Operation

- If the `Start Operation` button is selected, the automatic operation is performed with the pattern of a selected MC program.
- The AUTO operation execution status is displayed through the View-Execution-Program part and the execution file selection box.
  The number of an executed file is displayed through the execution file selection box.
  The details of an executed file are shown in the View-Execution-Program part. The current execution row is displayed with a blue bar.
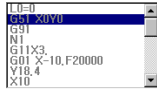
- If the `Stop Operation` button is selected during the AUTO operation, the AUTO operation can be paused. `Start Operation` button is selected again, the AUTO operation is resumed continuously from the suspension point.
- If the `Reset` button is selected during the AUTO operation, the AUTO operation can be cancelled.
  If the Reset button is selected, the AUTO operation being executed is cancelled completely.
  The Reset button is located in the lower right corner of the system panel or in the status bar of the main form.

③ Auxiliary Function

- `Single Block` : It is the function of performing the AUTO operation in the unit of the block of the MC program.
  If the Single Block operation and the `Start Operation` button is selected, the MCS is paused after executing one block inside the execution program.
  If the `Start Operation` button is selected again, the next block is executed.
- `Dry Run` : It is the function of performing the AUTO operation at the movement speed with the AUTO operation interpolation movement speed set in the "**P31** Dry Run Speed".
- `Block Skip` : It is the function of skipping during execution the block that has the Block Skip command, '/', inside the MC program.
- `Machine Lock` : It is the function of not outputting the movement signal to the external motor and executing the AUTO operation. The coordinate value displayed in the system panel is changed, but the actual movement is not performed.
- `Aux. Function Lock` : It is the function of skipping during execution the secondary command (M Code: The command related to the PLC motion) inside the execution program.

(6) MDI Mode

The MDI (Manual Direct Input) mode is the AUTO operation mode to operate the MCS with the pattern programmed similarly to the AUTO mode.
In the MDI mode, the operation pattern is inputted through the program input box of the system panel.
If the `MDI` button of the mode part is selected, the MCS operation mode is changed to the MDI mode.
If the MCS is changed to the MDI mode, the `MDI` button is displayed as being deactivated.
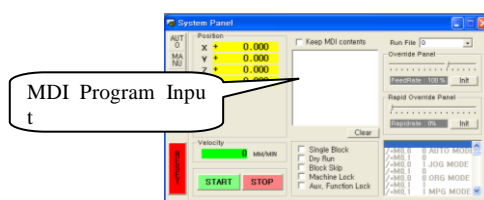
MDI Program Input

[MDI Mode]

① MDI Program Input

- Before the MDI operation is executed, a program is prepared through the MDI program input box. A program can be prepared within 600 characters.
- If the `Clear` button is selected, all the details of the MDI program input box are deleted.

② MDI Operation

- If the `Start Operation` button is selected after a program is prepared in the MDI program input box, the AUTO operation is executed with the prepared pattern.
- The details of the program prepared in the MDI program input box after the MDI operation is completed are deleted automatically. In order not to perform the automatic deletion function, the program is not deleted after the completion of the MDI operation if the `☑ Keep MDI contents` selection box is `Stop Operation`
- If the `운전 정지` button is selected during MDI operation, the MDI operation can be paused. If the `Start Operation 작` button is selected again, the MDI operation is resumed continuously from the suspension point.
- The `Reset` button can be selected during MDI operation to cancel the MDI operation. If the Reset button is selected, the MDI operation being executed is cancelled completely. The Reset button is located in the lower left corner of the system panel or in the status bar of the main form.

(7) The Other Functions of the System Panel

① Setting the Override
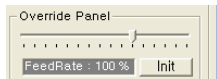
The Override function is the function of changing the manual operation speed and the automatic operation interpolation speed in percentage. If 100% is selected, it is moved at the ordered movement speed. Since the movement speed is '0' if 0% is selected, the movement is stopped.

Movement Speed = Manual Operation Speed or Automatic Operation Interpolation Speed X Override (%)

The Override function is the function applied when the "**P23.1** OV ENB" parameter is 'Enable'. The function can be applied in the system panel when the "**P23.0** OP CH" parameter is 'Enable'.

If the `Initial Value` button is selected, the Override is changed to the '100%' value.

② Setting the Rapid Override
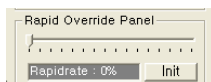
The Rapid Override function changes the rapid movement speed in percentage. If 100% is selected, it is moved at the ordered rapid movement speed. Since the rapid movement speed is '0' if 0% is selected, the movement is stopped.

Rapid Movement Speed = Ordered Rapid Movement Speed X Rapid Override (%)

The Rapid Override function is applied when the "**P23.2** ROV ENB" parameter is 'Enable'. The function can be applied in the system panel when the "**P23.0** OP CH" parameter is 'Enable'.

If the ⬚ Initial Value ⬚ button is selected, The Rapid Override is changed to the value of ‘100%’.

③ Displaying the Location and the Speed

The current location (absolute location) and the current speed are displayed.
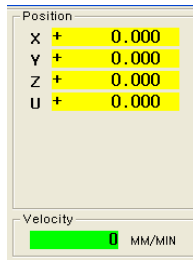
(8) Releasing the Alarm

An alarm may be given during the test operation or normal motion of the MCS.
If an alarm is given, the automatic operation of the MCS is stopped. The red "ERR" LED of the front panel of the MCS is turned on.

If the MCS is connected to the MSW-MCS program when an alarm is given, the following alarm window is executed.
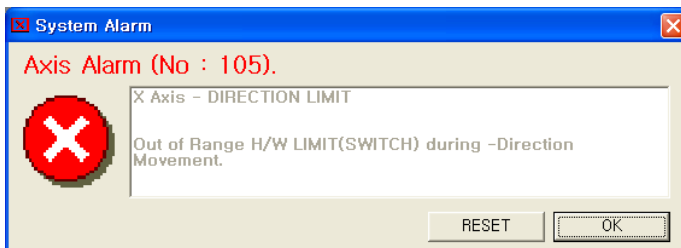
Moreover, an alarm number is displayed in the area displaying the MCS motion status of the main form status bar.

If an alarm is given, remove the cause of an alarm and reset the MCS.

The Reset button is located in the status bar of the main form, the alarm window, and the system panel.

6) Editing the Program

MCS operation means that the motion of the equipment with the MCS mounted is programmed for storage in the MCS and that the MCS operates the equipment suitably to the details of the saved program.
The types of the programs to be prepared for MCS operation include the PLC program and the MC program.
The PLC program controls the total equipment operation and is operated from the time when power is inputted. The MC program is operated from the time when the MCS operation mode is selected as the AUTO mode and the operation start order is given.

The method of operating the editor for program preparation and the program debugging panel are explained.

A program can be prepared by using the "File" and "System" menus of the main form.
Two menus support the same program editing function. However, as for the difference between two menus, the file saved in the PC is read to start the editing function in the "File" menu. The program saved in the MCS is read to start the editing function in the "System" menu.

In this section, the explanation is given on the basis of the motion through the "System" menu, and the difference of two menus is stated.

(1) MC Program

The MC program is prepared for the automatic operation of the MCS. Maximal 72 units can be saved in the MCS. A name is assigned with a number. An MC program name can be selected from '0' to '250'.
Refer to "Chapter 2 Programming" for the detailed commands of the MC program.
How to use the program panel is explained.

① Preparing a New Program
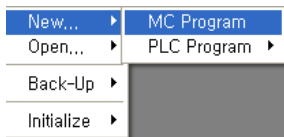
If the "MC Program" menu in the lower part of the "New Program" menu below the "System" menu is selected, the editing window for editing a new MC program is executed.
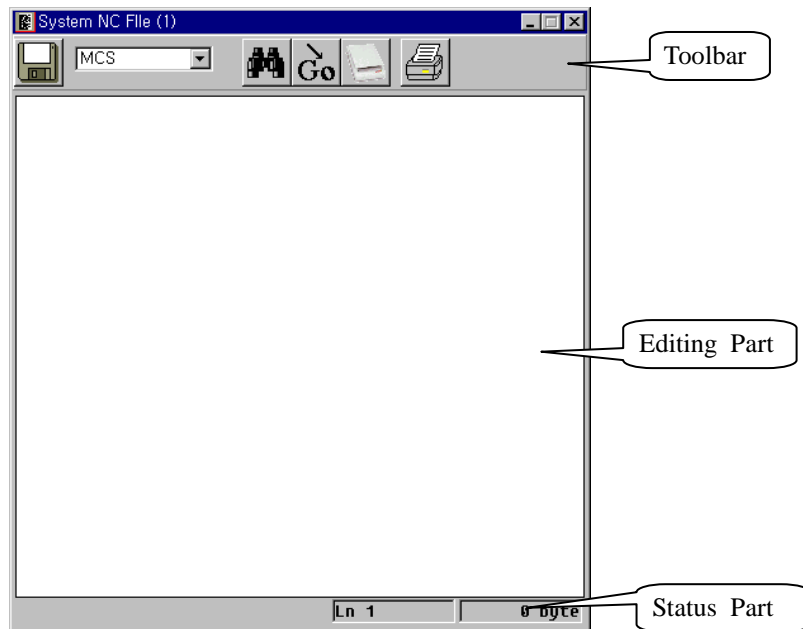
The editing window consists of the toolbar, the editing part, and the status part. If a new program is prepared, an empty editing part is displayed. A new MC program can be prepared through the editing part.

② **Opening the Program**
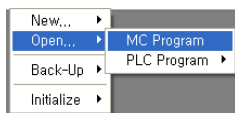
If the "MC Program" menu in the lower part of the "Open Program" menu below the "System" menu is selected, the system dialog window showing the MC program list registered in the MCS is executed.
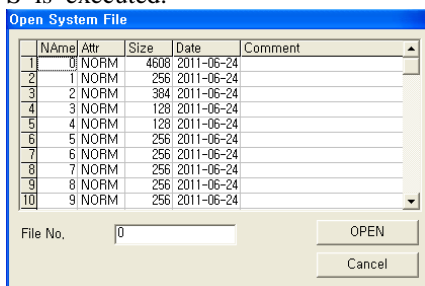
If the Open File button is selected after the program to be edited is selected from the list, the editing window reading the selected program from the system and showing it in the editing part is executed.

```
System NC FIle (0)

[MCS ▼]

L0=0
G51 X0Y0
G91
N1
G11X3,
G01 X-10,F20000
Y18,4
X10
G11X0
G4 X0,2
L0=L0+1
GOTO N1


Ln 14          100 byte
```
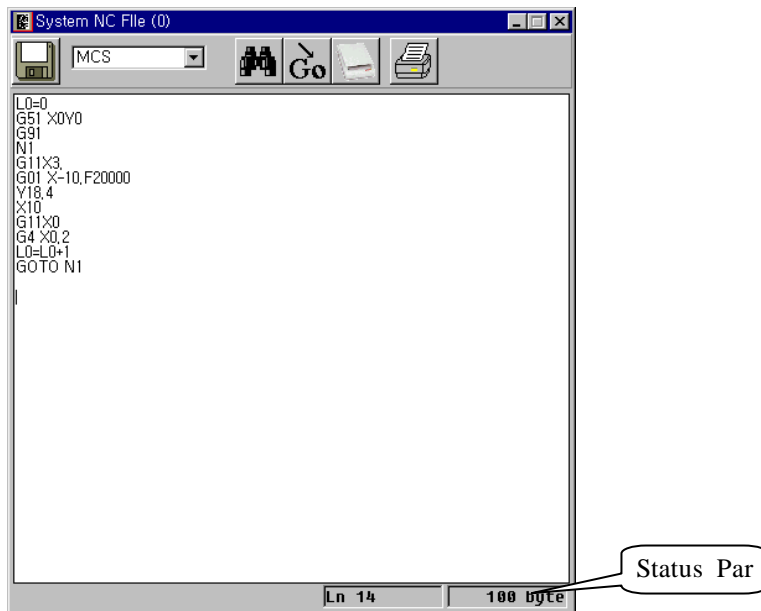
Status Par

The line number where the current cursor is located and the program size are displayed in the status part. The program can be modified through the editing part.

*** If Open Program is executed through the "File" menu, the file dialog window in which a file can be selected from PD appears instead of the system dialog box. If a file is selected, the editing window appears.
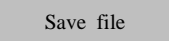
③ **Saving the Program**

The modified program is saved in the system.

- The target MCS box storing a program is selected.
- The 🖫 button is selected.
- The system dialog box in which an MC program name can be selected is executed.
  The name of the MC program to be saved is inputted in the file name input box.
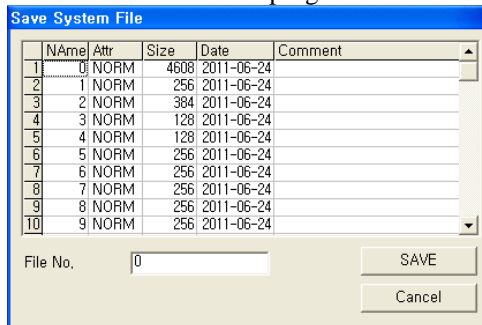
- If the **Save file** button is selected, the details of the modified MC program are saved in the system.

*** In order to save the modified MC program in a PC file, the storage target is selected with the PC box.
*** When it is saved in a PC file, the file dialog window is executed instead of the system dialog.

④ **Finding a String**

A special string can be found from the MC program being edited.

- If the 🔍 button is selected, the string finding and changing window where a string can be inputted is executed.
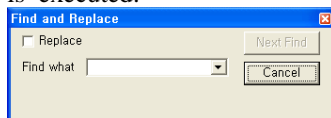
- The string to be found is inputted through ▥, the input box for the details to be found.
- If the **Find Next** 다음 찾기 button is selected, the cursor is located in the position searched by finding the string to be found in the editing window.
- If the string to be found is not searched, the message window notifying that such a string doesn't exist is displayed.
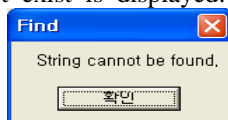
⑤ **Changing a String**

A specific string can be changed to another string in the MC program being edited.

- If the 🔍 button is selected, the string finding and changing window where a string can be inputted is executed.
- If the **Changing** checkbox is selected in the string finding and changing window, the window shape is changed to the change window.
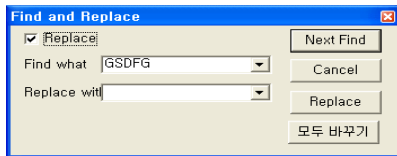
- The string to be found is inputted through [N11] which is the input box for the details to be found. The string to be changed is inputted through [N12], the input box for the details to be found.
- If the [Changing] button is selected, the string is modified to the changed string by searching the string to be found, and the cursor is located in the position.
- If the [Change All] button is selected, all the strings to be found are searched in the editing window, and all the strings found are modified to the changed strings.

⑥ **Printing the Program**

The details of the editing window can be outputted to a printer. In order to use the function, the printer should be connected to the PC in which the MSW-MCS is being executed.

- If the 🖨 button is selected, the print dialog window is executed.
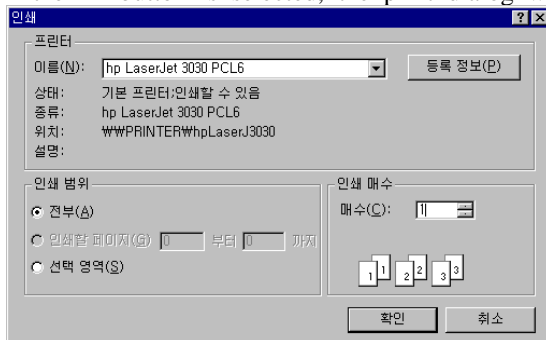
- If the [Confirmation] button is selected, the details of the editing window are outputted to a printer.

The [-] button gives the order executing the panel used for debugging the MC program details being edited.

(2) PLC Program

The PLC program is prepared for the overall operation of the equipment with the MCS installed and is classified into the interrupt PLC program and the SCAN PLC program.
The interrupt program is operated as the main PLC program of the MCS. The SCAN PLC program is operated as the secondary PLC program.
If only one PLC program is prepared, prepared it as the interrupt PLC program.

Refer to "Chapter 4 PLC Program" for the detailed commands of the PLC program.
How to use the program tool is explained from the example of the editing of the interrupt PLC program. The difference in editing the SCAN PLC program is explained.

① Preparing a New Program
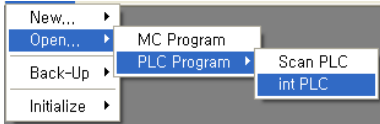
If the "Interrupt Program" menu below the "PLC Program" menu below the "New Program" menu below the "System" menu is selected, the editing window editing a new interrupt PLC program is executed.
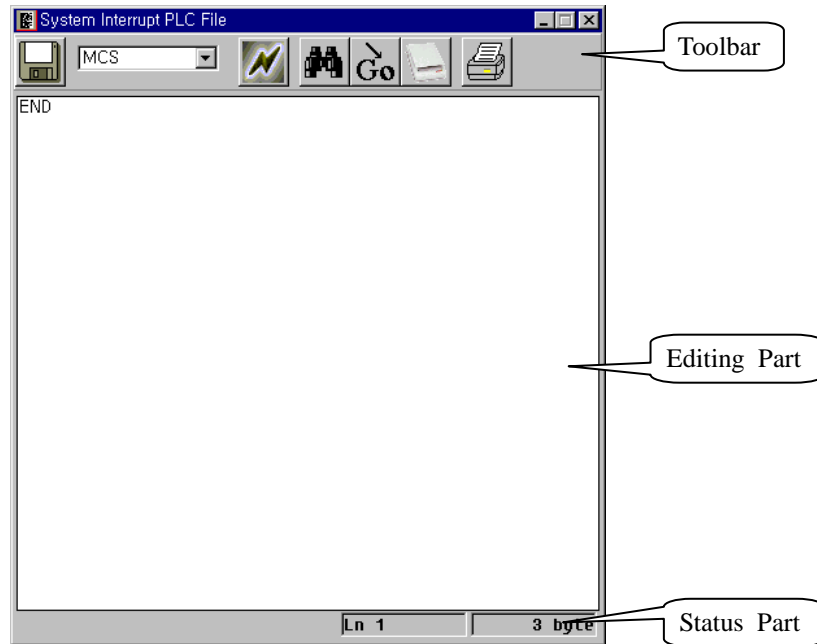
The editing window consists of the toolbar, the editing part, and the status part. If a new program is prepared, the empty editing part is displayed. A new interrupt PLC program can be prepared through the editing part.

If an interrupt PLC program is saved in the MCS, the current program is deleted, and the message checking whether a new program is prepared is displayed.
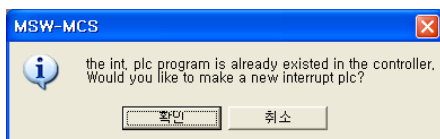
*** In order to prepare a new SCAN PLC program, "Scan Program" below the "PLC Program" below "New Program" below the "System" menu is selected.

② **Opening a Program**
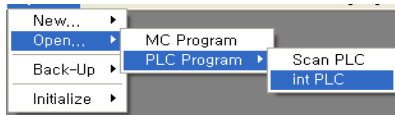
If "Interrupt Program" below the "PLC Program" menu below the "Open Program" menu below the "System" menu is selected, the interrupt program registered in the MCS is read to execute the editing window shown in the editing part.
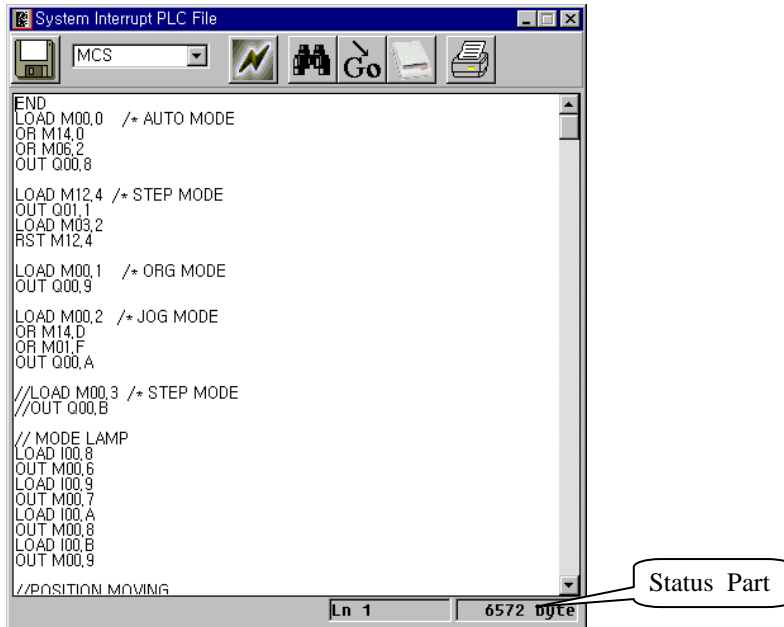
The line number with the cursor located currently and the program size are displayed in the status part. A program can be modified through the editing part.

*** In order to open the SCAN PLC program of the MCS, "Scan Program" below the "PLCS Program" menu below the "Open Program" menu below the "System" menu is selected.

*** If Open PLC Program is selected through the "File" menu, the file dialog window in which a file can be selected in the PC appears. If a file is selected, the editing window appears.

③ **Saving a Program**

A modified PLC program is saved in the system.

- A program storage target is selected with the MCS box.
- If the 🖫 button is selected, the message box notifying the MCS operation method appears.
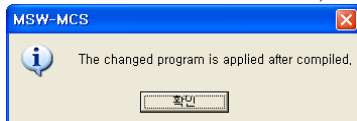
- If the Confirmation button is pressed, the PLC program is saved in the MCS.

(Note) A saved PLC program is not reflected to the MCS operation immediately. The PLC Compile command should be given after storage to compile the saved PLC program and reflect it to the MCS operation.

*** In order to save the modified PLC program in a PC file, the storage target is selected with the `PC ▾` box.

*** When it is saved in a PC file, the file dialog window for file selection is executed.

④ **Compiling a Program**

Compiling a PLC program is to analyze the PLC program saved in the MCS into a machine language file, create a machine language file, and reflect it to the following MCS operation.

- The program storage target is selected with the `MCS ▾` box.
  If the program storage target is different, the Compile order button is not displayed.

- If the ◰ button is selected, the order of compiling the PLC program is sent to the MCS. If the PLC compilation is completed in the MCS, the completion message is displayed.

System Information
System Informationr(No : 2)
PLC COMPILE IS COMPLETED.

PLC Compile is completed with NO ERROR.

OK
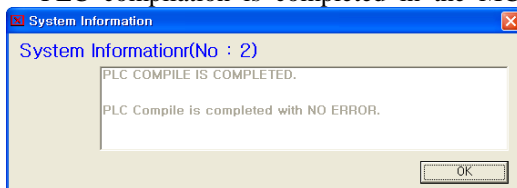
- If a syntax error is detected from the PLC program during compilation, the error message with the detailed information on the error stated is displayed. Recompile it after correction.

(Note) PLC Compile cannot be executed during the MCS AUTO operation.

- If the scan PLC program is compiled, the MCS PLC operation is stopped. If the interrupt PLC program is compiled, the MCS PLC operation is started. Even though the scan PLC program is compiled as only the scan PLC program is changed, compile the interrupt PLC program to start the MCS PLC operation.

[The Interrupt PLC Program Change Manipulation Sequence]

Modify the Interrupt PLC ▷ Save the Interrupt PLC ▷ Compile the Interrupt PLC

[Scan PLC Program Change Manipulation Sequence]

Modify the Scan PLC ▷ Save the Scan PLC ▷ Compile the Scan PLC ▷ Compile the Interrupt PLC
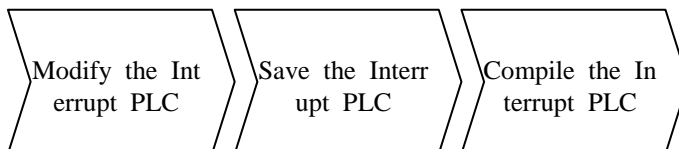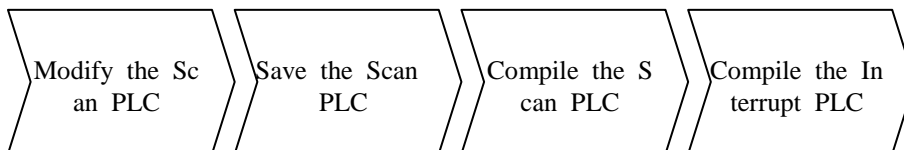
Refer to "MC Program" for the string finding and changing button, ⚏.

Refer to "MC Program" for the program print button, ⊟.

The ⊟ button is the order button executing the panel to debug the PLC program details being edited. The details are explained in "Program Monitoring" of the section.

(3) Monitoring a Program

The values of the MCS contact points or variables used during the preparation and test of the MC and PLC programs need to be found in the real time.
In order to check the used MCS contact points and variables in the real time, the cross reference function is supported in the MSW-MCS. The cross reference function makes one list of all used contact points and variables and search them in the real time after searching the details of the MC or PLC program.
The corresponding value can be set compulsorily in the list prepared.
As for the cross reference function, if the ⊡ button of the editing window is selected, the function is executed.

① Searching All Contact Points and Variables

It is the function of making and monitoring the list of all the MCS contact points or variables used in the MC and PLC programs.
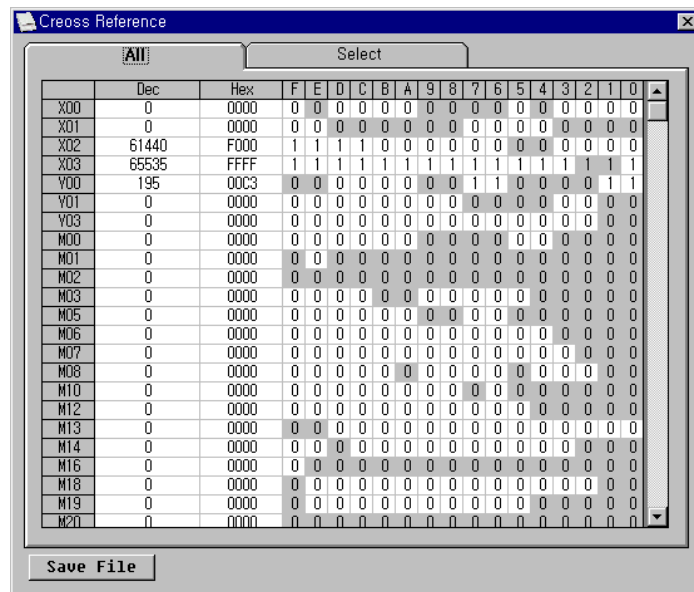
- If the ⊡ button is selected, the Cross Reference window is executed.

- If the [All] tab is selected, the current value is searched in the real time through the list in which all contact points and variables can be searched. The current value is shown as the decimal number, the hexadecimal number, and the contact point. The contact point used in the program is displayed in a dark color.

- A value can be set compulsorily through the list. The compulsory setting method is to double-click the displayed area of the value to be changed. The input box appears in the location of the double-clicked value, The compulsory setting function is executed through the input box. If the decimal number (Dec) area is double-clicked, it can be set compulsorily in the unit of a decimal number. If the hexadecimal number (Hex) area is double-clicked, it can be set compulsorily in the unit of a hexadecimal number. Moreover, if the contact point (Bit) area is selected, it can be set compulsorily in the unit of a contact point.

② Searching a Selected Contact Point or Variable

It is the function of monitoring the 18 data selected from the list of all the MCS contact points or variables used in the MC and PLC programs.
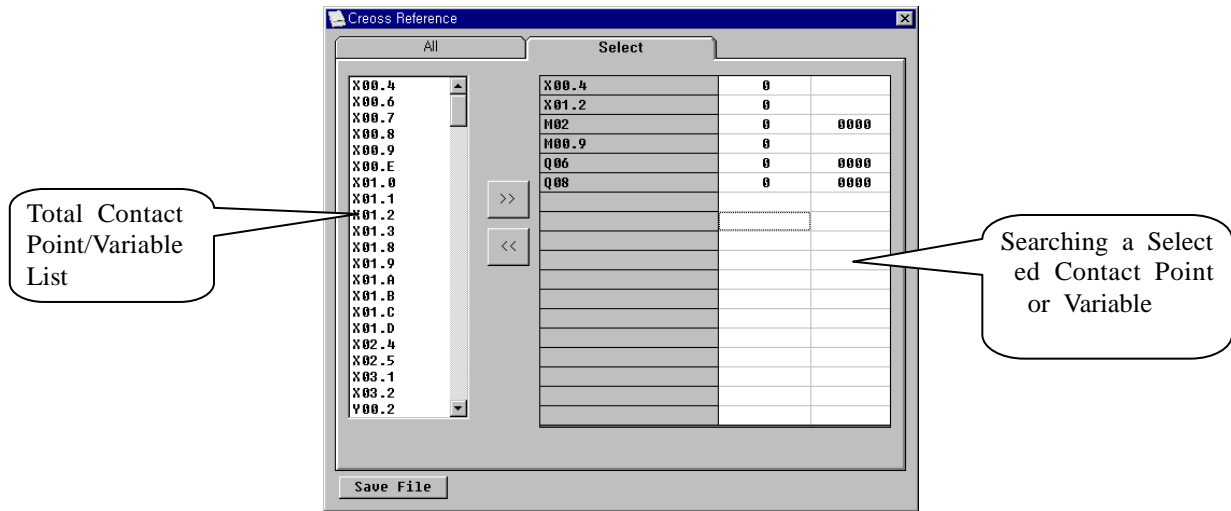
- If the ⬛ button is selected, the Cross Reference window is executed.
- If the [Select] tab is selected, the window is changed to enable the search for a selected contact point or variable.
  The window consists of the total contact point and L variable list part in which all the contact points and variables used in the MC and PLC programs are arranged and the selected contact point and variable search part in which a selected contact point or variable can be monitored.
- The ⬛ button is utilized to register the contact point or variable of the total contact point and L variable list in the selected contact point and variable search part. Maximal 18 contact points or variables can be registered.
- The ⬛ button is utilized to delete the contact point or variable registered in the selected contact point and variable search part.
- A value can be set compulsorily through the selected contact point and variable search part. As for the compulsory setting method, the displayed area of the variable to be changed is double-clicked out of the variables registered for search. The input box appears in the location of the double-clicked value. The compulsory setting function is executed through the input box.

③ Output in the Total Contact Point and Variable File

The list of all the MCS contact points and variables used in the MC and PLC programs can be saved in a file.

- The [Save File] button is selected.
- The dialog box from which the file to be saved can be selected appears.
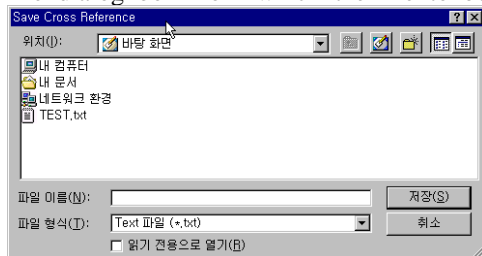
- If the [Save(S)] button is selected after file selection, the list of used contact points and variables is saved in the file.

- The details of a created file can be checked through edit-program(ex. Notepad..).
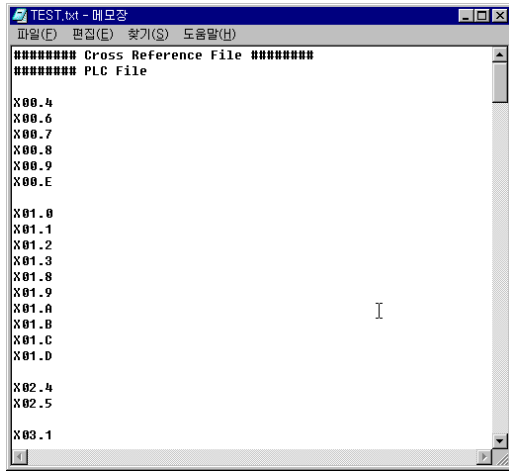
(4) Program Explorer

The function of manipulating the MC program registered in the MCS through the program explorer in a file unit can be executed.

If the "Program Explorer" menu below the "Tool" menu is selected, the program explorer window is executed.
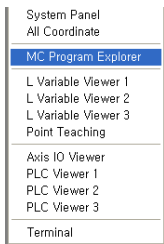
The program explorer window consists of the MC program list and command parts.
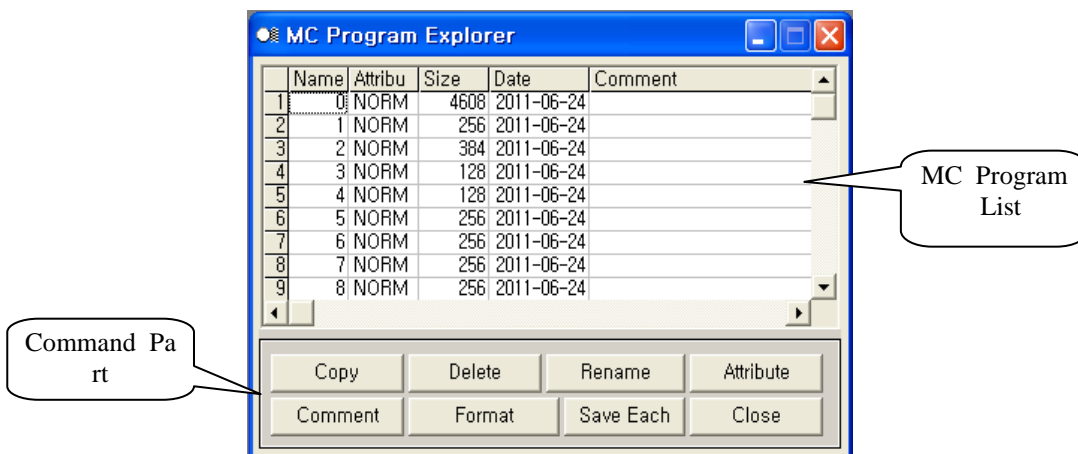
The information of all the MC programs registered in the MCS is displayed in the MC program list part.
The command part consists of the command buttons with which the MC program can be manipulated.

① Copying an MC Program

It is the function of copying a registered MC program to the MC program with another name. An

MC program is copied as follows.

- Select the program to be copied from the MC program list.
- If the [Copy] button is selected, the command part is changed to the following check part.

Would you copy the selected file(0) to another file?

Target File Nam [0]   Execute   Cancel
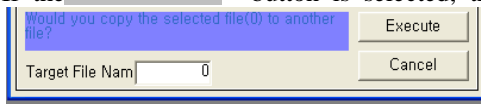
- The name of the program to be copied is inputted in the file name input text box of the check part.
- If the [Execute] button of the check part is selected, the MC program is copied, and the check part is changed to the command part again.
- If copying the MC program is to be cancelled, select the [Cancel] button. Then, the execution is cancelled, and the check part is changed to the command part again.

② **Deleting an MC Program**

It is the function of deleting the MC program registered in the MCS.
The MC program deletion method is as follows.

- Select the program to be deleted from the MC program list.
- If the [Delete] button is selected, the command part is changed to the following check part.
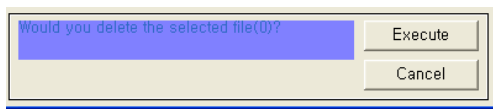
Would you delete the selected file(0)?   Execute   Cancel

- If the [Execute] button of the check part is selected, the MC program is copied, and the check part is changed to the command part again.
- In order to cancel deleting the MC program, select the [Cancel] button. Then, the execution is cancelled, and the check part is changed to the command part again.

③ **Changing a Program Name**

It is the function of changing the name of the MC program registered in the MCS to another name.
The MC program file name can be changed as follows.

- Select the MC program name to be changed from the MC program list.
- If the [Changing a File Name] button is selected, the command part is changed to the check part as follows.

Would you rename the selected file(5)?   Execute

Target File Nam [5]   Cancel
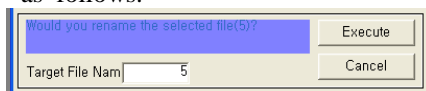
- Input the program name to be changed in the program name input text box of the check part.
- If the [Execute Command] button of the check part is selected, the name change is executed, and the check part is changed to the command part again.
- In order to cancel the name change, select [Cancel Execution] button. Then, the execution is cancelled, and the check part is changed to the command part again.

④ **Setting the Program Properties**

The MC program properties include "Norm" and "Lock". It is the function of setting the properties.
Two MC program properties can be set. The properties are changed from "Norm" to "Lock" and from "Lock" to "Norm".

If the MC program property is set to "Lock", the MC program cannot be deleted or modified. The MC program properties can be changed as follows.

- The MC program whose property is to be changed is selected from the MC program list.
- If the [Attribute] button is selected, the order part is changed to the following check part.
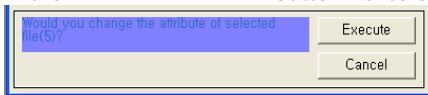
- If the [Execute] button of the check part is selected, the property change is executed, and the check part is changed to the command part again.
- In order to cancel the property change, select [Cancel] button. Then, the execution is cancelled, and the check part is changed to the command part again.

⑤ Inputting a Program Comment

A comment is the statement explaining an MC program and is prepared in English within 30 characters.
A comment can be inputted in the MC program registered in the MCS by using the function,
A comment can be inputted as follows.

- Select from the MC program list the MC program in which a comment is to be inputted.
- If the [Comment] button is selected, the command part is changed to the check part as follows.
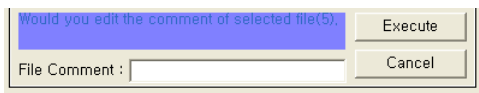
- The comment to be changed is inputted in the program comment input text box of the check part.

- If the [Execute] button of the check part is selected, the comment change is executed, and the check part is changed to the command part again.
- In order to cancel the comment change, select [Cancel] button. Then, the execution is cancelled, and the check part is changed to the command part again.

⑥ Initializing the Program Area

Initializing the program area is the function of initializing the whole area where all the MC and PLC programs within the MCS are saved. If the function is executed, all programs are deleted so that it should be executed carefully.
Initializing the program area is as follows.

- If the [Format] button is selected, the message box for checking the command execution appears. If the function execution button is selected, the command part is changed to the check part as follows.
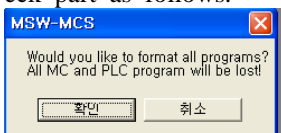
- If the [Execute] button of the check part is selected, initializing the program area is executed, and the check part is changed to the command part again.
- In order to cancel initializing the program area, select the [Cancel] button. Then, the execution is cancelled, and the check part is changed to the command part again.

⑦ Saving a Program Individually in a File

It executes the function of individually selecting a registered MC program and save it individually in a PC file.

An MC program can be saved in an individual PC file as follows.

- If the ⸢Save Individually⸥ button is selected, the window supporting the function of individual saving a program in a PC file is executed.
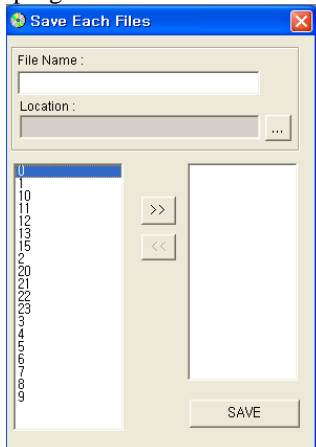
- The list of the MC programs registered in the MCS is displayed in the lower left box. If the ⸢>>⸥ button is selected after selecting the MC program to be saved, the program displayed in the lower left box is moved to the lower right box for display.
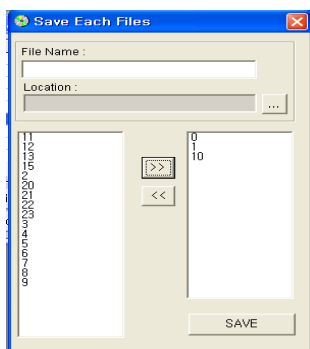
- If the ⸢...⸥ button is selected, the dialog window with which a file to be saved can be selected is executed. The name and path of a selected file are displayed through the window.
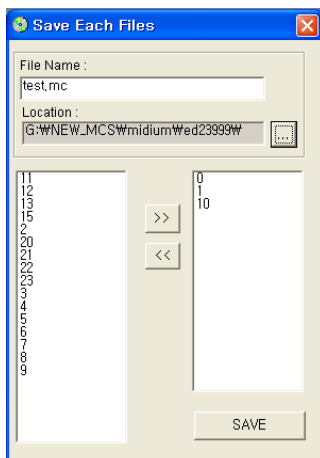
- If the ⸢SAVE⸥ button is selected, a selected file is created in a file. If the three programs such as MC program 0, 1, and 2 are selected and test.mc is selected as the file name, the three files such as test(0).mc, test(1).mc, and test(2).mc are created.

⑧ Ending the Program Explorer

The ⬜ Ending the Explorer button is the function of ending the program explorer window.

7) System Management

The system tool used for managing the MCS is explained. Most of the system tools are located below the "Tool" menu and the "More Functions" menu of the main menu.

(1) Viewing All Coordinates

The Viewing-All-Coordinates panel displays the coordinate values (absolute value, mechanical coordinate, remaining distance, and movement error) of all axes. If the "Viewing All Coordinates" menu below the "Tool" menu is selected, the panel is executed
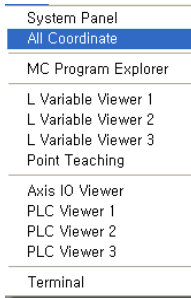
- An absolute coordinate is the standard coordinate system used as the basis to prepare an MC program.
  A coordinate is reset by using the G40 (the coordinate system setting), the location is displayed on the basis of a new coordinate system.
  A coordinate on the system panel is displayed on the basis of the absolute coordinate system.
- A mechanical coordinate displays a location on the basis of the reference point of a machine.
- A remaining distance displays the distance to be moved during axis movement.
- A movement error is displays the error between the actual location calculated with the positional signal inputted from the encoder and the absolute location.
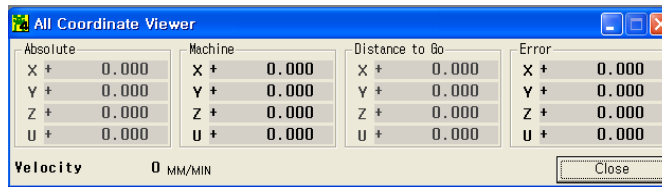
(2) L Variable Explorer-1,2,3

The value of the L variable used in the MC program can be searched and set compulsorily by using the L variable explorer.

The MCS operates 10,000 L variables. 8,000 L variables (L0 ~ L7999) are operated as real values and maintain the saved values even though power is turned off. The remaining 2,000 L variables (L8000 ~ L9999) are operated as integer values and are used for special purposes.

L variable	Type	Purpose
L0 | L7999	Real Variable	- As a user variable - A non-volatile variable - Possible compulsory setting
L8000 | L9999	Integer Variable	- As a system variable - A non-volatile variable - Impossible compulsory setting

If "L variable Explorer-1", "L variable Explorer-2 ", or "L variable Explorer-3" below the "Tool" menu is selected, the L variable explorer is executed.
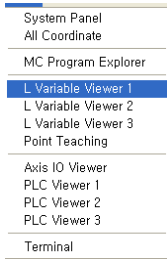
The L variable explorer consists of the View-Variables part and the toolbar.
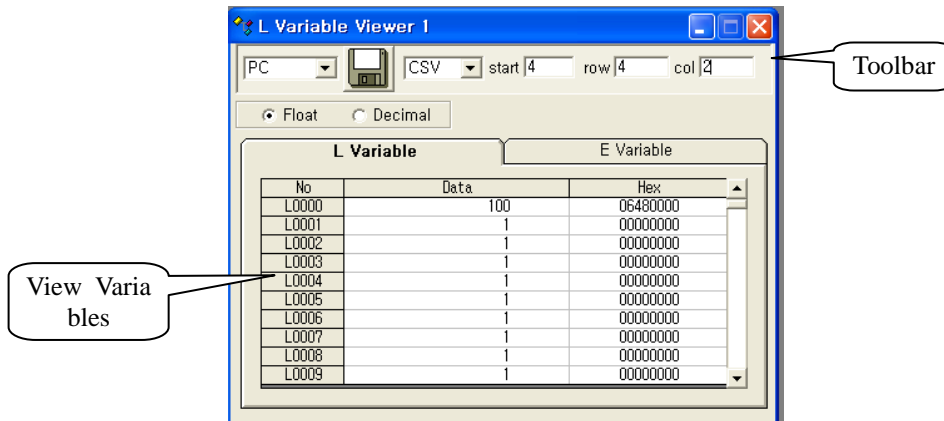
The View-Variables part is the area where an L variable (MCS: 0~9999) can be searched and set compulsorily.

① **Searching a L variable**

- If the [Float] radio button is selected to search a L variable as a real number type, the data of the L variable are displayed in the View-Variables area on the basis of the value of a real number. The real number type HEX value is displayed in the HEX value area too.

- If the [Decimal] radio button is selected to search a L variable as an integer number type, the data of the L variable are displayed in the View-Variables area on the basis of the value of a real number. The real number type HEX value is displayed in the HEX value area too.

(Note) In case of the search for L8000 or higher, do it in an integer type.

② **Setting a L variable Compulsorily**

- Only the L variable (L0 ~ L7999) for a user can be set compulsorily in a real number type.

- The area of the value of the L variable to be set compulsorily is double-clicked.

- The input box appears in the location where a selected L variable is displayed. Pressing the 'Enter' key after inputting a setting value enables the compulsory setting.

③ **Saving a L variable in a File**

It is the function of saving the L variable of the MCS in a file for management. A file can be saved in an LVA or a CSV type. The LVA type is the file type exclusively for Emotiontek. The CSB type is the file type that can be used in Excel.

* Saving in an LVA File Type

- If the 'LVA' type is selected from the [LVA] file type selection box, the toolbar is organized as follows.

[PC ▼] [💾] [LVA ▼]   Start   4   End   4

- 'PC' is selected from the [PC ▼] storage location selection box.

- The variable to be saved is selected through [Start   4 End   4] input box. In order to save 100 L variables from L0 to L99, '0' is inputted in the start input box, and '99' is inputted in the end input box. In order to save 600 L variables from L100 to L799, '100' is inputted in the start input box, and '799' is inputted in the end input box.
  If all L variables are saved in a file, it takes a lot of time.

- If the [💾] storage button is selected, the message box checking whether variables L are to be saved in a file is executed.
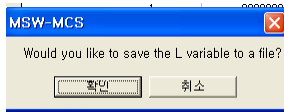
- If the [Confirmation] button is selected, the dialog box where a file can be selected is executed.
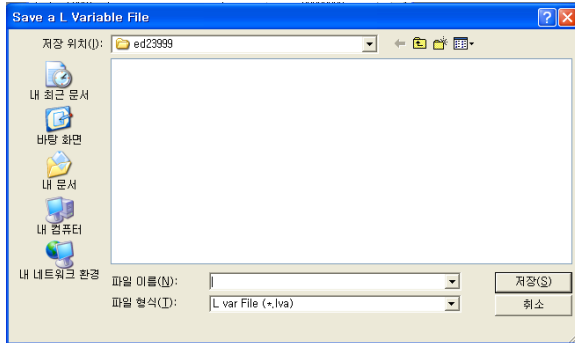
- If the [Save (S)] button is selected after file selection, the L variable is saved in the LVA file.

* Saving in a CSV File Type

- If the 'CSV' type is selected from the [CSV ▼] file type selection box, the toolbar is organized as follows.

      Start 0   Column40   Row 4
[PC ▼] [💾] [CSV ▼] 시작[0]   열[40]   행[4]

- 'PC' is selected from the [PC ▼] storage location selection box.

- The variable to be saved is selected through [Start 0   Column40   Row 4] input box.
  A user can use it when a L variable is to be managed in the point data type of the robot controller and when the usage in Excel is desired.
  The point data start from L100. One point data consists of four L variables. If 50 point data are to be saved, '100' is inputted in the start input box with '50' in the column input box and with '4' in the row input box.
  The point data start from L10. One point data consists of nine L variables. If 70 point data are to be saved, '10' is inputted in the start input box with '70' in the column input box and with '9' in the row input box.

- If the ⊞ storage button is selected, the message box checking whether L variables are to be saved in a file is executed.
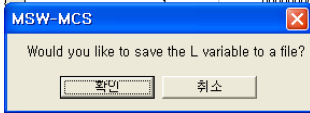
- If the [Confirmation] button is selected, the dialog box where a file can be selected is executed.
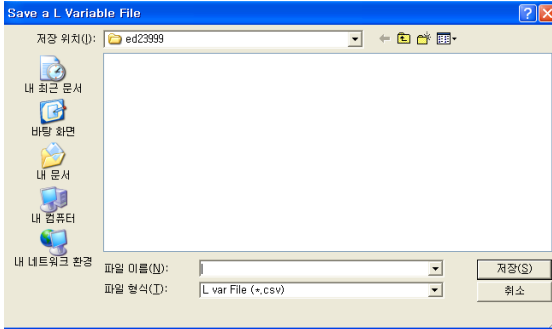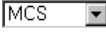
- If the [Save (S)] button is selected after file selection, the L variable is saved in the CSV file.

④ Recovering a L variable from a File

It is the function of recovering the L variable of the MCS with the value of the saved L variable.

* Recovering a L variable from an LVA File

- Select the ‘LVA’ type from the [LVA ▼] file type selection box.

- Select ‘MCS” from the [MCS ▼] storage location selection box.

- If the ⊞ storage button is selected, the message box checking whether the L variable of the MCS is to be recovered with the L variable file is executed.
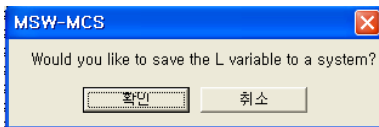
- If the [Confirmation] button is selected, the dialog box where an LVA file can be selected is executed.
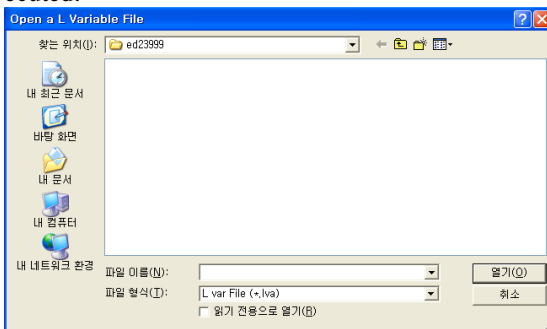
- If the [Open (O)] button is selected after file selection, the L variable is recovered with the LVA file.

* Recovering a L variable from an CSV File

- Select the ‘CSV’ type from the [CSV ▼] file type selection box.

- Select 'MCS' from the [MCS ▼] storage location selection box.

- If the 🖫 storage button is selected, the message box checking whether the L variable of the MCS is to be recovered with the L variable file is executed.
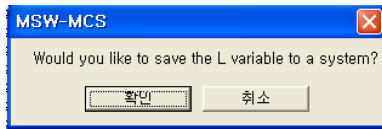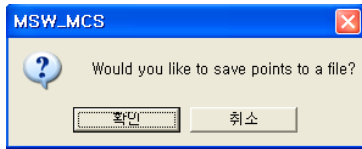

- If the [Confirmation] button is selected, the dialog box where an CSV file can be selected is exe cuted.
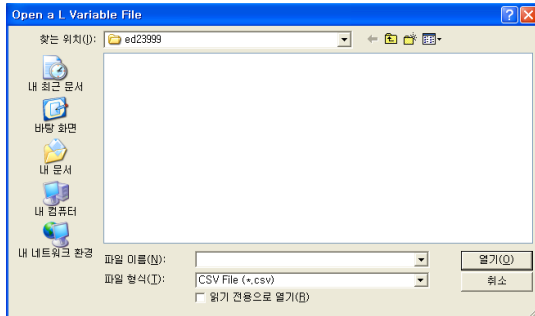
- If the [Open (O)] button is selected after file selection, the L variable is recovered with the CSV file.

(3) Point Teaching

The robot controller user uses the point data with the movement location saved. One point data consist of the location values of several axes. In the MCS, the L variable can replace the point data. The L variable is not the group data like the point data so that it is difficult for a robot controller user to us e the L variable.
In order to solve the difficulties, the tool managing the L variable like the point data similarly to the robot controller is the point teaching panel.

If the "Point Teaching" menu below the "Tool" menu is selected, the point teaching panel is exec uted.
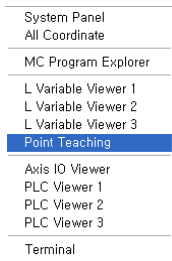
The data edited through the point teaching panel are the L variable.
The point teaching panel consists of the toolbar, point data list, data setting method, and View Locatio n part.

Toolbar

Point Data List

Data Setting Method

View Location

① **Organizing the Point Teaching Panel**

If the point teaching panel is executed, the panel is executed with the empty list with the point data list part not organized. The point data list part is organized by using the toolbar function. The organization methods include the type of reading the L variable from the MCS for organization and the type of reading a file (*.RTP) for organization.
Moreover, there is the type of organizing it with the point data list part organization value.

* Organizing the Panel with the Point Data List Part Organization Value

- Select the number of the axes of the point data from 4 Axes. Selection of four axes is explained as an example.

- Set the L variable from which the point data to be read start in L 0, the input box of L 0 ~ L#### Number 100 Set the number of point data Number 100 , the input box.

- If the Initial button is selected, the message box checking whether the point data list part is organized with the value set in the point data list part is executed.
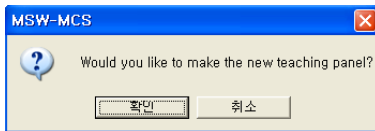
- If the Confirmation button is selected, the new point data list is organized.
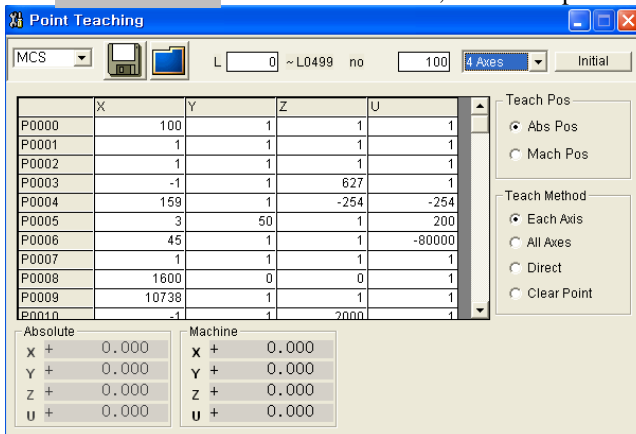
* Reading the L variable from the MCS and Organizing the Panel

- Select 'MCS' from MCS, the box for selecting the location to be read.

- Select the number of point data axes from 4 Axes. Selection of four axes is explained as an example.

- Set the L variable from which the point data to be read start in L 0, the input box of L 0 ~ L#### Number 100 Set the number of point data Number 100 0 , the input box.

- If the open button, is selected, the message box checking whether point data are read from the MCS is selected.
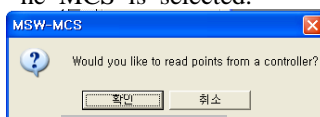
- If the Confirmation button is selected, the L variable is read from the MCS to organize the point data list.
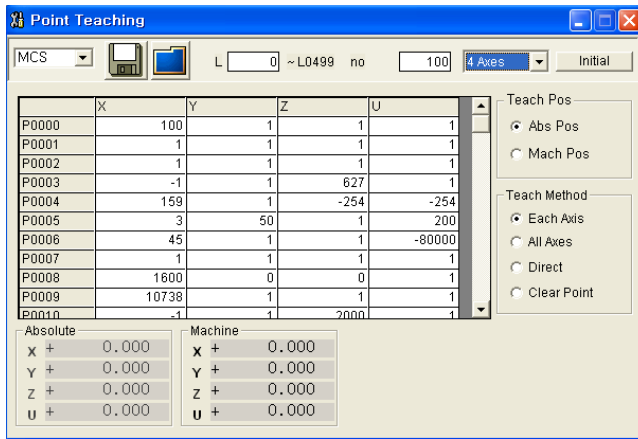
* Organizing the panel from a file

- It is the method of organizing the panel by using the file saved with the point teaching panel.

- 'PC' is selected from [PC ▼], the box for selecting a location to be read.

- If the ▣ open button is selected, the message box checking whether point data are read from a file is executed.
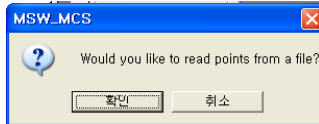
- If the [Confirmation] button is selected, the dialog box from which a file can be selected is executed.
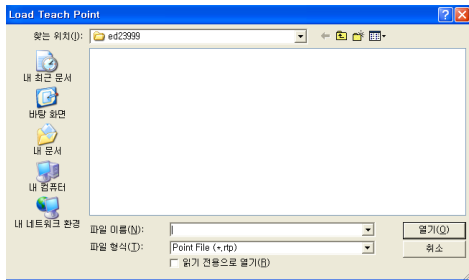
- If the [Open(O)] button is selected after RTP file selection, the point data list is organized with the point teaching panel organization value and the point data value.
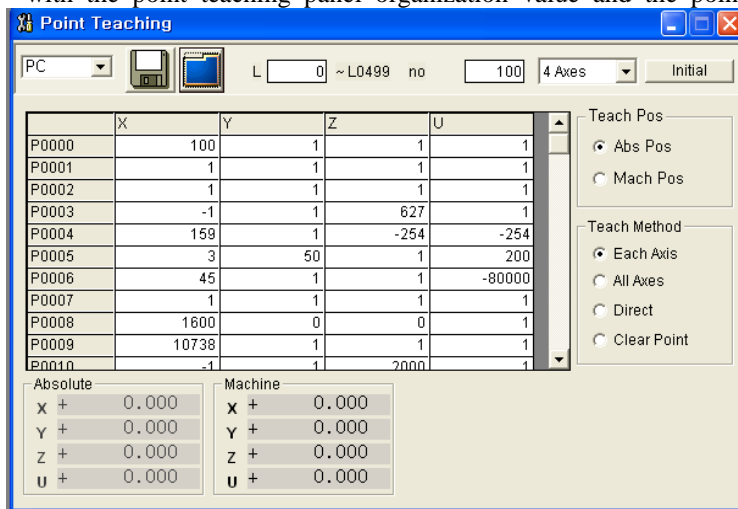
② **Editing the Point Teaching Data**

The point data editing methods include the direct editing type and the indirect editing type. The direct editing type is the method of inputting it through the data input box. The indirect editing type is the method of selecting and inputting an absolute or mechanical coordinate value. There are four editing methods. An editing method can be selected through the teaching method selection area.

Teach Method
- Each Axis
- All Axes
- Direct
- Clear Point

* **Indirectly Inputting the Data of Each Axis of the Pont**

- Select the `Each Axis` radio button in the teaching method.
- Select the `Abs Pos` button to input an absolute coordinate. Select `Mach Pos.`
  button to input a mechanical coordinate.
- If the axis of the point data to be modified is double-clicked, the current absolute or mechanical coordinate of the selected axis is inputted.

* **Indirectly Inputting the Data in All the Axes of the Point**

- Select the `All Axes` radio button in the teaching method.
- Select the `Abs Pos` button to input an absolute coordinate. Select t `Mach Pos.`
  button to input a mechanical coordinate.
- If the point data to be modified are double-clicked, the current absolute or mechanical coordinate is inputted in the all the axes of the point data.

* **Directly Inputting the Data of Each Axis of the Point**

- Select the `Direct` radio button in the teaching method.
- Double-clicking the axis of the point data to be modified makes the `260000` input box appear.
- Pressing the 'Enter' key after inputting the data in the input box makes the point data changed.

* **Initializing the Data of All the Axes of the Point**

- It is the function of initializing the data of all the axes of the point to '0' .
- Select the `Clear Point` radio button in the teaching method.
- If the point data to be modified are double-clicked, the data of all the axes of the point data are initialized to '0' .

③ **Saving the Point Data**

The edited point data can be saved in the MCS or a file.

* **Saving in the MCS**

- Select 'MCS' from `MCS`, the storage location selection box.

- If the 🖫 storage button is selected, the message box checking whether the point data are saved in the MCS is executed.
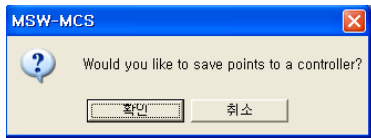
- If the [Confirmation] button is selected, the point data are saved in the MCS.

* Saving in a File

- Select 'PC' from [PC ▼], the storage location selection box.

- If the 🖫 storage button is selected, the message box checking whether the point data are saved in a file is executed.

- If the [Confirmation] button is selected, the dialog box where the RTP file to be saved is selected is executed.
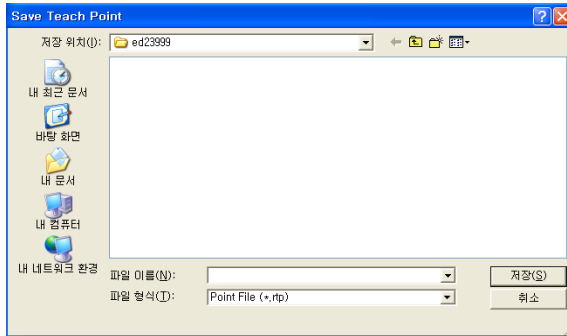
- If the [Save(S)] button is selected after RTP file selection, the point data are saved in the file.

(4) Axis I/O Explorer

Through the axis I/O explorer, the I/O contact point related to the axis can be searched, and the value of a contact point can be set compulsorily.
If the "Axis I/O Explorer" below the "Tool" menu is selected, the axis I/O explorer panel is executed.
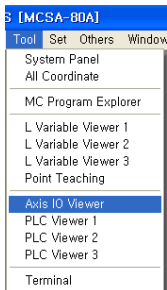
The axis I/O explorer is organized as follows. All the contact points related to each axis can be searched.
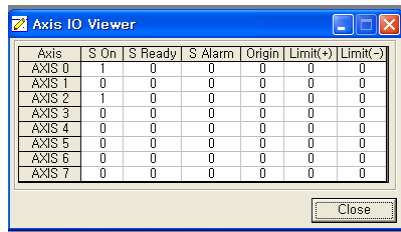
The Servo ON contact point out of the contact points related to an axis can be set compulsorily.

* Compulsorily Setting a Servo ON Contact Point

- The Servo ON contact point value to be set in the variable area is double-clicked.

- The view area of the corresponding contact point becomes a text box so that it is converted into the input-enabled area. A value (0 or 1) can be set through the box.

  If the 'Enter' key is selected after value input, the value is set.

(5) PLC Information Explorer-1,2,3

A PLC contact point can be searched or set compulsorily through the PLC information explorer.
If the "PLC Information Explorer-1", "PLC Information Explorer-2, or "PLC Information Explorer-3" menu below the "Tool" menu is selected, the PLC information explorer panel is selected.
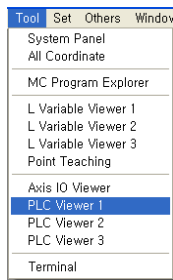
The PLC information explorer can execute three windows at the same time so that the contact point in another area can be searched or set in the same window.
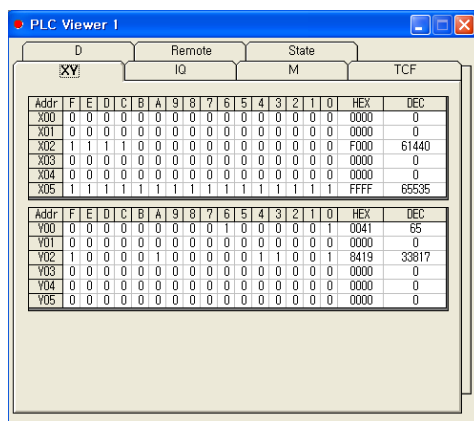
The PLC information explorer window is organized as in the above figure. It is divided into the 7 variable view areas (XY area, IQ area, M area, timer/counter/F, D area, remote area, and execution status).

Refer to "Chapter 9 Standard I/O Signal" for the meaning of each area.

① XY Area

In the XY area, a PLC I/O contact point can be searched or set compulsorily [XY Area | Y영역] tab of the PLC information explorer is selected, the function is executed.

In the XY area, a bit, HEX, or DEC value can be searched.

* Bit Setting

- The [0] bit value to be set is double-clicked in the bit search area.
- The bit view area of the corresponding contact point becomes a text box so that it is converted in to the input-enabled area. A bit value (0 or 1) can be set through the box.
- Selecting the 'Enter' key after inputting a bit value sets the bit value.
- The bit setting is possible only in the Y area.

* HEX Setting

- The [0000] HEX value to be set is double-clicked in the HEX search area.
- The HEX view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A HEX value (0~FFFF) can be set through the box.
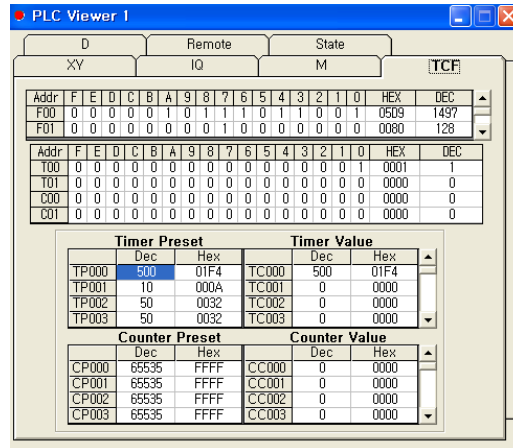- Selecting the 'Enter' key after inputting a HEX value sets the HEX value.
- The HEX setting is possible only in the Y area.

* DEC Setting

- The [0] DEC value to be set is double-clicked in the DEC search area.
- The DEC view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A DEC value (0~65535)can be set through the box.
- Selecting the 'Enter' key after inputting a DEC value sets the DEC value.
- The DEC setting is possible only in the Y area.

① IQ Area

In the IQ area, the value of the area of the communication between the MC inside the MCS and the PLC is searched, or a PLC I/O contact point can be searched or set [IQ Area] y. If the [IQ 영역] tab of the PLC information explorer is selected, the function is executed.

In the IQ area, a bit, HEX, or DEC value can be searched.

* Bit Setting

- The [0] bit value to be set is double-clicked in the bit search area.
- The bit view area of the corresponding contact point becomes a text box so that it is converted into the input-enabled area. A bit value (0 or 1) can be set through the box.
- Selecting the 'Enter' key after inputting a bit value sets the bit value.
- The bit setting is possible only in the Q area.

* HEX Setting

- The [0000] HEX value to be set is double-clicked in the HEX search area.
- The HEX view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A HEX value (0~FFFF) can be set through the box.
- Selecting the 'Enter' key after inputting a HEX value sets the HEX value.
- The HEX setting is possible only in the Q area.

* DEC Setting

- The [0] DEC value to be set is double-clicked in the DEC search area.
- The DEC view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A DEC value (0~65535) can be set through the box.
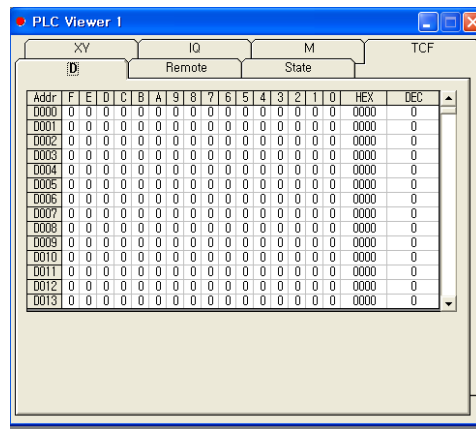- Selecting the 'Enter' key after inputting a DEC value sets the DEC value.
- The DEC setting is possible only in the Q area.

② **M Area**

In the M area, the memory in variable area of the PLC is searched or can be set compulsorily. If t M Area tab of the PLC information explorer is selected, the function is executed.

In the M area, a bit, HEX, or DEC value can be searched.

* **Bit Setting**

- The [0] bit value to be set is double-clicked in the bit search area.
- The bit view area of the corresponding contact point becomes a text box so that it is converted into the input-enabled area. A bit value (0 or 1) can be set through the box.
- Selecting the 'Enter' key after inputting a bit value sets the bit value.

* **HEX Setting**

- The [0000] HEX value to be set is double-clicked in the HEX search area.
- The HEX view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A HEX value (0~FFFF) can be set through the box.
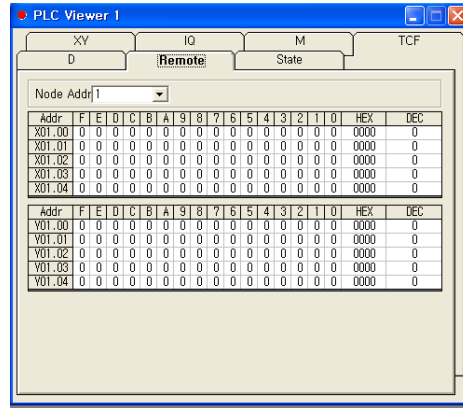- Selecting the 'Enter' key after inputting a HEX value sets the HEX value.

* **DEC Setting**

- The [0] DEC value to be set is double-clicked in the DEC search area.
- The DEC view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A DEC value (0~65535) can be set through the box.
- Selecting the 'Enter' key after inputting a DEC value sets the DEC value.

③ Timer, Counter, and F Area

In the timer, counter, and F area, all the contact points and variables used in the timer and the counter out of the PLC variables and the value of the F contact point showing the internal status are searched or can be set compulsorily. If the | Timer,Counter,F | tab of the PLC information explorer is selected, the function is executed.

The timer, counter, and F area is divided into the area where a bit, HEX, or DEC value can be searched and the area where only a DEC or HEX value can be searched.

* Bit Setting

- The bit setting is possible only in the F, T, and C contact point areas.
- The | 0 | bit value to be set is double-clicked in the bit search area.
- The bit view area of the corresponding contact point becomes a text box so that it is converted into the input-enabled area. A bit value (0 or 1) can be set through the box.
- Selecting the 'Enter' key after inputting a bit value sets the bit value.

* HEX Setting

- The | 0000 | HEX value to be set is double-clicked in the HEX search area.
- The HEX view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A HEX value (0~FFFF) can be set through the box.
- Selecting the 'Enter' key after inputting a HEX value sets the HEX value.
- The HEX input is possible in all areas.

* DEC Setting

- The | 0 | DEC value to be set is double-clicked in the DEC search area.
- The DEC view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A DEC value (0~65535) can be set through the box.
- Selecting the 'Enter' key after inputting a DEC value sets the DEC value.

④ D Area

In the D area, the value in the internal variable area of the PLC is searched or can be set compuls
D Area [D 영역] tab of the PLC information explorer is selected, the function is executed.

In the D area, a bit, HEX, or DEC value can be searched.

* **Bit Setting**

- The [0] bit value to be set is double-clicked in the bit search area.
- The bit view area of the corresponding contact point becomes a text box so that it is converted in to the input-enabled area. A bit value (0 or 1) can be set through the box.
- Selecting the 'Enter' key after inputting a bit value sets the bit value.

* **HEX Setting**

- The [0000] HEX value to be set is double-clicked in the HEX search area.
- The HEX view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A HEX value (0~FFFF) can be set through the box.
- Selecting the 'Enter' key after inputting a HEX value sets the HEX value.

* **DEC Setting**

- The [0] DEC value to be set is double-clicked in the DEC search area.
- The DEC view area of the corresponding word becomes a text box so that it is converted into the input-enabled area. A DEC value (0~65535) can be set through the box.
- Selecting the 'Enter' key after inputting a DEC value sets the DEC value.

⑤ Remote Area

In the remote area, the contact point information of the corresponding remote I/O is searched or can
 be set compulsorily when the remote I/O is connected through the RS422/485 port of th[Remote Area]
the [리모트 정보] tab of the PLC information explorer is selected, the function is executed.
The function is reserved to expand the MCS function and is not used yet.

In the remote area, a bit, HEX, or DEC value can be searched.

The code (1~63) of the remote I/O to be searched can be selected by us[Node Number   1 ▾] sele
ction box.

* Bit Setting

- The [0] bit value to be set is double-clicked in the bit search area.
- The bit view area of the corresponding contact point becomes a text box so that it is converted in
 to the input-enabled area. A bit value (0 or 1) can be set through the box.
- Selecting the 'Enter' key after inputting a bit value sets the bit value.

* HEX Setting
- The [0000] HEX value to be set is double-clicked in the HEX search area.
- The HEX view area of the corresponding word becomes a text box so that it is converted into the
  input-enabled area. A HEX value (0~FFFF) can be set through the box.
- Selecting the 'Enter' key after inputting a HEX value sets the HEX value.

* DEC Setting

- The [0] DEC value to be set is double-clicked in the DEC search area.
- The DEC view area of the corresponding word becomes a text box so that it is converted into the
  input-enabled area. A DEC value (0~65535) can be set through the box.
- Selecting the 'Enter' key after inputting a DEC value sets the DEC value.

⑥ **Execution Status**

It is the function of searching or setting the execution status of the interrupt PLC and the scan PLC. Execution Statu 랙 상태 tab is selected, the function is executed.
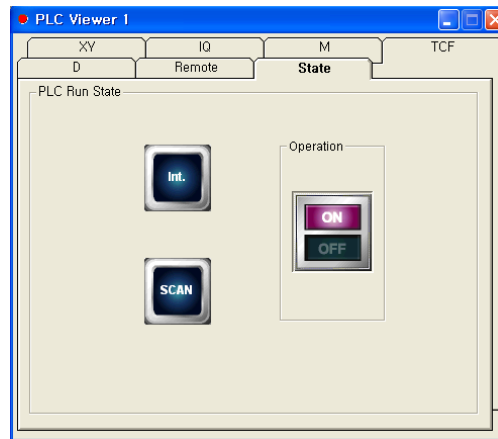
- If the interrupt and scan lamps are in green color, it mans that the PLC is being executed. If they are in red color, the PLC is in the stop status.

- If the toggle switch is selected, the execution of the scan PLC can be stopped compulsorily or can be executed.

(6) Terminal

It is the function of executing the terminal panel with the type of the console window in which the contact point and variable inside the MCS can be searched or set.

If the "Terminal" menu below the "Tool" menu is selected, the Terminal panel is executed.
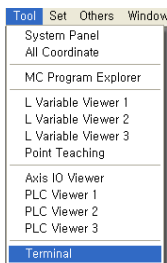
The Terminal panel has the type of the console in which a command can be inputted.
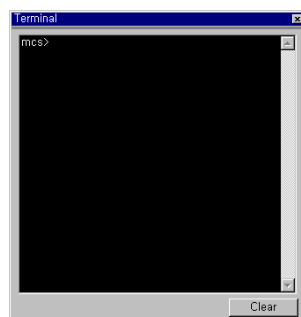
If a user inputs a command through the console window and presses the 'Enter' key, the inputted command is executed with the result displayed in the next line.

① **Command**

The values of the contact points and variables used in the MC and PLC programs are checked, or the commands set compulsorily are explained.

	Operation	Command	
Screen Clear		CLS<ENTER> Select the [Clear] button.	
L variable	Variable Check	L[No]?<ENTER>	Yes: L100? <ENTER>
	Variable Setting	L[No]=[Data]<ENETR>	Yes: L100=1100 <ENTER>
Variable Q	Word Check	QQ[No]?<ENTER>	Yes: QQ200? <ENTER>
	Bit Check	QQ[No].[Bit]?<ENTER>	Yes: QQ200.5? <ENTER>
	Word Setting	QQ[No]=[Data]<ENTER>	Yes: QQ210=0x0300 <ENTER>
	Bit Setting	QQ[No].[Bit]=[Bit Data]<ENETR>	Yes: QQ210.3=1 <ENTER>
Contact X	Word Check	X[No]?<ENTER>	Yes: X0? <ENTER>
	Bit Check	X[No].[Bit]?<ENTER>	Yes: X0.A? <ENTER>
Contact Point Y	Word Check	Y[No]?<ENTER>	Yes: Y0? <ENTER>
	Bit Check	Y[No].[Bit]?<ENTER>	Yes: Y0.8? <ENTER>
	Word Setting	Y[No]=[Data]<ENTER>	Yes: Y0=0x1111 <ENTER>
	Bit Setting	Y[No].[Bit]=[Bit Data]<ENETR>	Yes: Y1.7=0 <ENTER>
Contact Point I	Word Check	I[No]?<ENTER>	Yes: I30? <ENTER>
	Bit Check	I[No].[Bit]?<ENTER>	Yes: I0.A? <ENTER>
Contact Point Q	Word Check	Q[No]?<ENTER>	Yes: Q8? <ENTER>
	Bit Check	Q[No].[Bit]?<ENTER>	Yes: Q0.1? <ENTER>
	Word Setting	Q[No]=[Data]<ENTER>	Yes: Q0=0x0400 <ENTER>
	Bit Setting	Q[No].[Bit]=[Bit Data]<ENETR>	Yes: Q0.1=1 <ENTER>
Contact Point M	Word Check	M[No]?<ENTER>	Yes: M23? <ENTER>
	Bit Check	M[No].[Bit]?<ENTER>	Yes: M20.F? <ENTER>
	Word Setting	M[No]=[Data]<ENTER>	Yes: M10=0xFF0F <ENTER>
	Bit Setting	M[No].[Bit]=[Bit Data]<ENETR>	Yes: M20.1=0 <ENTER>
Contact Point F	Word Check	F[No]?<ENTER>	Yes: F0? <ENTER>
	Bit Check	F[No].[Bit]?<ENTER>	Yes: F0.E? <ENTER>
Contact Point T	Word Check	T[No]?<ENTER>	Yes: T0? <ENTER>
	Bit Check	T[No].[Bit]?<ENTER>	Yes: T1.5? <ENTER>
	Word Setting	T[No]=[Data]<ENTER>	Yes: T0=0x0010 <ENTER>
	Bit Setting	T[No].[Bit]=[Bit Data]<ENETR>	Yes: T0.4=1 <ENTER>
Contact Point C	Word Check	C[No]?<ENTER>	Yes: C0? <ENTER>
	Bit Check	C[No].[Bit]?<ENTER>	Yes: C1.E? <ENTER>
	Word Setting	C[No]=[Data]<ENTER>	Yes: C0=0x0210 <ENTER>
	Bit Setting	C[No].[Bit]=[Bit Data]<ENETR>	Yes: C0.C=1 <ENTER>
Contact Point D	Word Check	D[No]?<ENTER>	Yes: D100? <ENTER>
	Bit Check	D[No].[Bit]?<ENTER>	Yes: D20.1? <ENTER>
	Word Setting	D[No]=[Data]<ENTER>	Yes: D30=0x40 <ENTER>
	Bit Setting	D[No].[Bit]=[Bit Data]<ENETR>	Yes: D33.1=1 <ENTER>

[No]: The number of a variable or a contact point is inputted.
[Bit]: The number of a contact point bit is inputted. '0' ~ 'F'

[Data]: An integer type data value is inputted. As for the input type of a hexadecimal integer, '0 x' is added to the front of the number, or 'H is added to the end of the number.

[Bit Data]: '0' or '1'

Maximum 10 recently ordered commands are buffered. A buffered command can be called with the <Up> or <Down> key.
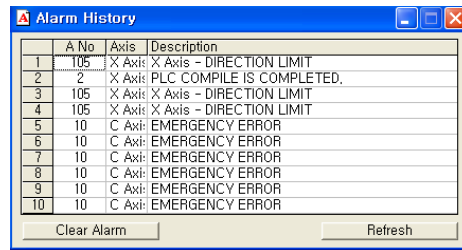
(7) Alarm History

The details of the alarms occurred in the MCS are saved in the MCS. The details of the 10 alarms occurred recently are saved in the non-volatile memory so that they are also maintained during power-off.

If the "Alarm History" menu below the "More Functions" menu, the Alarm History panel is executed.
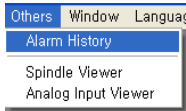
The panel displaying maximum 10 recent alarms is as follows.

- All the alarm history information saved in the MCS is deleted with the button.

- The alarm history information is imported again from the MCS with the button.

(8) Spindle Explorer

The MCS can operate maximum two spindles. The spindle explorer is the function of checking the movement speed of the axis operated by a spindle.
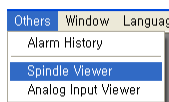
If the "Spindle Explorer" menu below the "More Functions" menu is selected, the Spindle Explorer panel is executed.
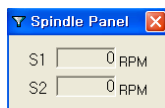
(9) Analog Input Explorer

The MCS can operated maximum four analog input ports. The analog input explorer is the function of checking the value of the signal inputted into an analog port.
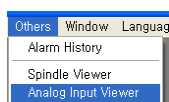
If the "Analog Input Explorer" menu below the "More Functions" menu is selected, the analog input explorer panel is executed.
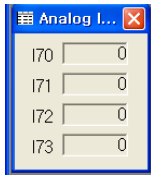
(10) System Version Check

If it is connected to the MCS remotely, the MCS OS version and the MCS type can be checked.

MSW-MCS Information
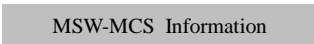
If the "MSW-MCS Information" menu below the "Help" menu is selected, the MSW-MCS information window is executed.
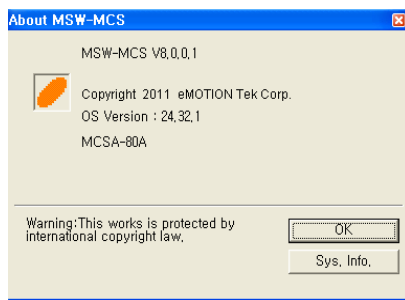
The MSW-MCS program information and the OS version and type of the MCS remotely connected can be checked.

- MSW-MCS V5.0A   : It is the MSW-MCS program version.

- OS Version   : MCS OS Version

- MCSA-80A   : MCS type

Ex.) Explanation on the OS Version

OS Type
1 Standard OS
2 Special OS

MCS Type
1: Standard Analog Type
2: Standard Pulse Type
3: Premium Analog Type
4: Premium Pulse Type

OS Version
22.6
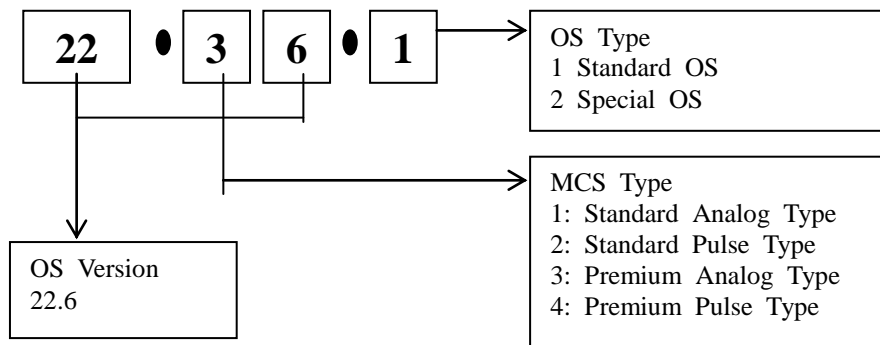
8) Special Parameter

The MCS can be used in connection with various upper devices. The MCS can be operated in connection with the PLC, the general purpose manipulator (touch panel, etc.), and the exclusive manipulator (MCS-PA).
The PLC parameter is set for the use in connection with various upper devices.

Moreover, the user message on equipment operation can be executed. A user message is set through a string parameter.

The special parameters are located below the "Setting" menu of the main menu.

(1) PLC Parameter

When the MCS is operated by communicating with the PLC of another company or the general- purpose touch panel, it communicates with the PLC of another company or the general-purpose touch panel through the RS422 port of the MCS. Furthermore, it can be connected to the exclusive panel (MCS-PA) through the RS422 port.

The PLC parameter panel is the function of setting the communication protocol and characteristics used for the communication with the PLC of another company or the general-purpose touch panel. The communication types are slave and master.

Furthermore, the watch-dog time of the interrupt and scan PLC programs of the MCS can be set.

If the "PLC Parameter" menu below the "Setting" menu is selected, the PLC parameter window is executed.
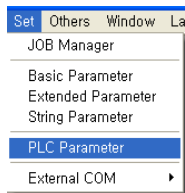
The PLC parameter window consists of the "Standard & Communication" tab in which the PLC watch-dog time and the RS422 communication port parameter can be set and the "Master Table " tab in which the master communication list can be set.
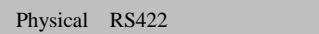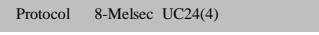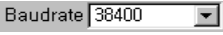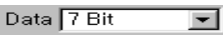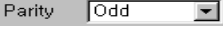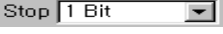
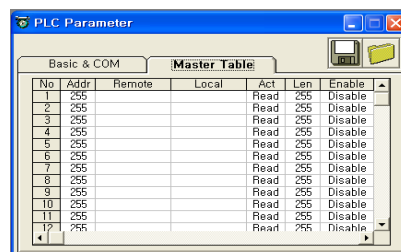① **Setting the PLC Standard and Communication Parameters**
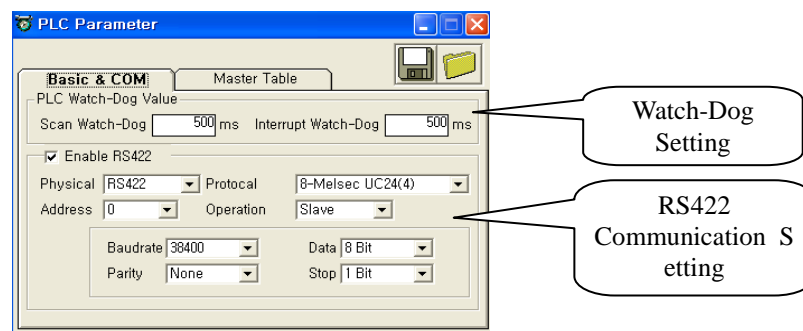
The ⬜ Basic & COM tab is selected to set the PLC standard and communication parameters.

The "Standard & Communication" tab consists of the part of setting the watch-dog time of the interrupt and scan PLCs and the part of setting the RS422 communication parameter.

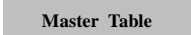- `Scan Watch-Dog [    6000] ms`: The watch-dog time of the scan PLC program is set. The 500ms or longer value can be set.

- `Interrupt Watch-Dog` `3000` ms : The watch-dog time of the interrupt PLC program is set. The 500ms or longer value can be set.

- `☑ Enable RS422` : Setting of the communication through the upper device (PLC, etc.) or the panel and the RS422 port

- `Physical   RS422` : The type of the physical connection to the upper device (PLC, etc.) or the panel is selected.
  'RS422' and 'RS485' can be selected.

- `Protocol   8-Melsec UC24(4)` : The protocol to be used for the communication with the upper device (PLC, etc.) or the panel is selected.
  'MelSec-C24', 'MelSec-UC24', or 'Master-K/Glofa' type can be selected.
  The 'EmotionTek-1', 'EmotionTek-2', and 'EmotionTek-3' types can be selected to communicate with the exclusive panel.

- `Address  0` : The MCS communication code is set when the MCS is operated as the communication slave.

- `Operation  Slave` : The MCS communication operation type ('Master' or 'Slave') is selected. If 'Master' is selected, set the details of the "Master Table" tab.

- `Baudrate 38400 ▼` : The communication speed is set. It is possible to set the maximum 128K communication speed.

- `Data 7 Bit ▼` : The number of communication data bits is set.

- `Parity Odd ▼` : The parity bit is set to verify a communication error.

- `Stop 1 Bit ▼` : The number of the stop bits of communication data is set.

② **Setting the Master Table**

The "Master Table" tab is the function possibly used when the "Communication Operation Type" of the "Standard & Communication" tab is selected as 'Master' and is organized as shown in the above window.

The `Master Table` tab is selected to set the master table.

- Code: The communication code of a slave device for communication is set.

- Target Contact Point: The contact point of the slave for communication is inputted by using a string. The input type of a target contact point is changed depending on the "Communication Protocol" type of the "Standard & Communication" tab.
  Refer to the supplementary manual for the further details on the setting of the master table.

- Internal Contact Point: The internal contact point of the MCS for communication is inputted by using a string.
    Refer to the supplementary manual for the further details on the setting of the master table.

- Operation: 'Read' or 'Write' is set.
    'Read' is the operation of importing the details of the slave target contact point to the internal contact point of the MCS.
    'Write' is the operation of copying the details of the internal contact point of the MCS to the slave target contact point.

- Length: The length of the contact point word performing the "Operation" item is set.

- Application: It is set whether a set operation is performed. If the item is set to 'Enable', the set details are added to the master communication.
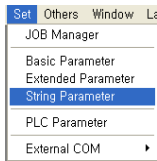
Double-clicking it makes the input window or list appear so that a value can be inputted.
Maximum 16 lists can be registered.

③ **Saving and Reading a Parameter**

- 🖫 Button: A set parameter is saved in the MCS.
- 📂 Button: A parameter value is read from the MCS.

(2) String Parameter

If the "String Parameter" menu below the "Setting" menu is selected, the string parameter window is executed.

```
Set  Others  Window  La
  JOB Manager
  Basic Parameter
  Extended Parameter
  String Parameter
  PLC Parameter
  External COM        ▶
```

If a special situation occurs due to the registration of a user message when the panel used exclusively for the MCS (Teach-Pendant, MCS-PA, etc.) is used, the user message can be displayed in the exclusive panel. The user message is registered in the string parameter window.
A string parameter consists of the 40 parameters from P900 to P939.
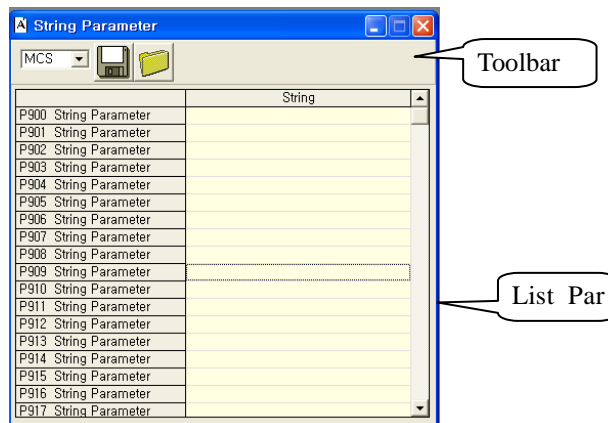
Toolbar
List Par

The string parameter window consists of the toolbar and the list part. A parameter value is changed through the list part. A parameter can be read or saved through the toolbar.

① **Modifying a Parameter**

    - The list item to be modified is double-clicked.
    - The `STRING 1` input box appears in the location of a selected item.
    - A string parameter is inputted through the input part.
    - Press the 'Enter' key after inputting a string in the input box.

② **Saving and Reading a Parameter**

The modified string parameter can be saved in the MCS or PC.
The detailed method is the same as the method of saving and reading a standard parameter.

Refer to the method of saving and reading a standard parameter.

③ **Editing a Parameter Name**

String parameter names are displayed in the "P800 String Parameter" type after the MSW-MCS is installed. The standard name can be changed to the name desired by a user.

The "StrParaTitle.nam" file is in the directory with the MSW-MCS installed.

If the MSW-MCS is restarted after the details of the file are modified, it can be checked that the p arameter name is changed for display.

9) System Backup, Recovery, and Initialization

It is the function of performing the backup or recovery of the parameters and variables of the MCS and of initializing them as the release value.
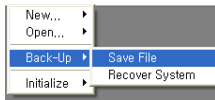It is located below the "System" menu of the main menu.
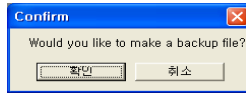
(1) System Backup

The user data (**All programs, parameters, interrupt and scan PLC programs, and 422 setting data**) of the MCS can be contained in a back-up file for storage. (**Except for a L variable**)

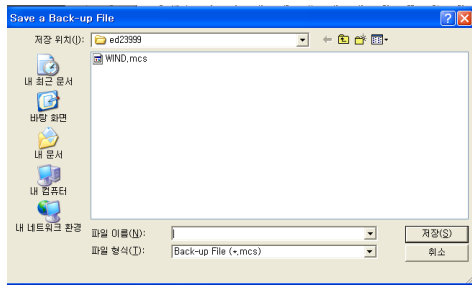(Note) Back up a L variable with the L variable explorer.

The "Save" menu below the "Back-Up" menu below the "System" menu can be selected to execute the system backup function.
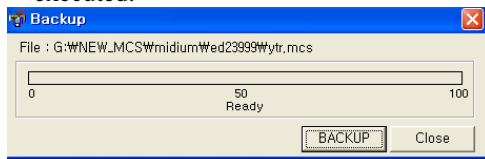

- If the "Save" menu is selected, the message box checking the system backup is executed.

- If the [Confirmation] button is selected, the dialog box where the file for system backup is selected is executed.

- If the [Save (S)] button is selected after the file for backup is selected, the system backup window is executed.

- If the [BACKUP] button is selected, the MCS backup file is created.

(2) System Recovery

The user data (**All programs, parameters, interrupt and scan PLC programs, and 422 setting data**) of the MCS can be recovered with the system backup file. (**Except for a L variable**)

(Note) Recover a L variable with the L variable explorer.

The "Recover" menu below the "Back-Up" menu below the "System" menu can be selected to execute the system recovery function.

- If the "Recover" menu is selected, the message box checking the system recovery is executed.

- If the [Confirmation] button is selected, the dialog box where the file for system backup is selected is executed.

- If the [Open (O)] button is selected after the file for recovery is selected, the system backup window is executed.
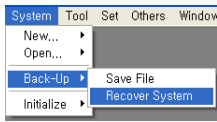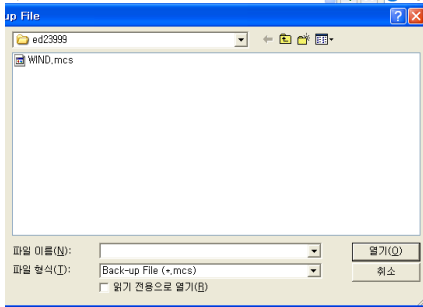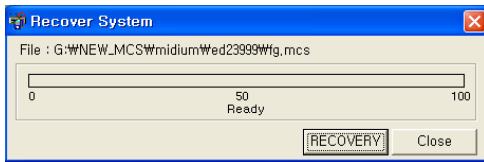

- If the [Recovery] button is selected, the MCS is recovered with the selected backup file.

(Note) Supply the power of the MCS again after the completion of recovery.

(Note) Compile PLC programs after the completion of recovery.

(3) Initializing a L variable
Initialize the user L variables (L0 ~ L7999) of the MCS to '0'.

The "L variable" menu below the "Initialize" menu below the "System" menu can be selected to execute the L variable initialization function.
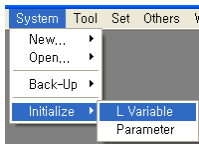
- If the "L variable" menu is selected, the password check window checking the initialization of the L variable is executed.
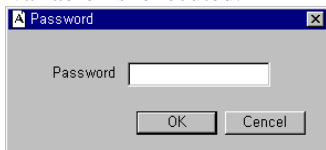
- If the [OK] button is selected after the password "1234" is inputted, the L variable initialization window is executed.
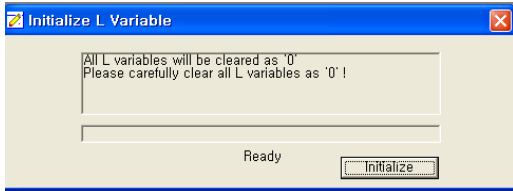
- If the [Initialize] button is selected, the message box checking the execution of the L variable initialization again is executed.
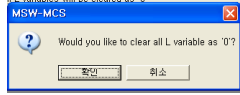
- If the [Confirmation] button is selected, the L variable is initialized.

(Note) Be careful that in case of the execution of the L variable initialization, the current data are deleted.

(4) Initializing a Parameter

The standard and expanded parameters of the MCS are initialized to the status of the release from the factory.
The "Parameter" menu below the "Initialize" menu below the "System" menu can be selected to execute the parameter initialization function.
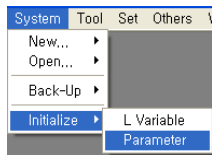
- If the "" Parameter" menu is selected, the password check window checking the parameter initialization is executed.
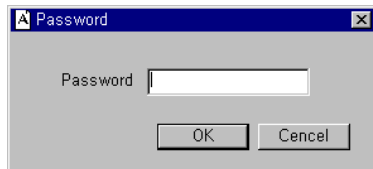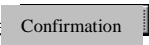
- If the [OK] button is selected after the password "1234" is inputted, the parameter initialization window is executed.
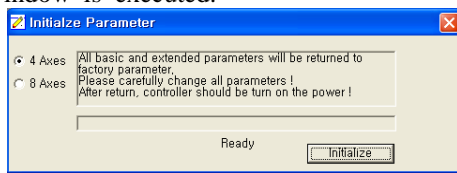
- Select the [4 Axes] radio button to initialize the MCS parameter for 4 axes. Select the [8 Axes] radio button to initialize the MCS parameter for 8 axes.

- If the [Initialize] button is selected, the message box checking one more execution of parameter initialization is executed.
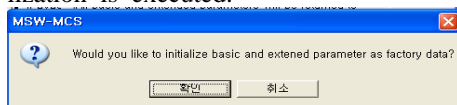
- If the Confirmation button is selected, the standard and expanded parameters are initialized.

(Note) Be careful that the execution of parameter initialization delete the current standard and expanded parameters.

Chapter 4 Alarm

Alarm List

Category	No.	Message	Details
Information	2	The PLC program is compiled normally.	
	3	The compilation is impossible since the MC program is being executed.	It is impossible to compile the PLC program since the MC program is being executed in the automatic mode.
	10	EMERGENCY ERROR	
Axis	100	SERVO NOT READY	The ready signal was not inputted from the servo of the corresponding axis.
	101	SERVO ALARM	The alarm signal occurred in the servo of the corresponding axis.
	102	+ DIRECTION S./W LIMIT	The corresponding axis entered the restricted area of the stored stroke limit during the movement in the + direction.
	103	- DIRECTION S/W LIMIT	The corresponding axis entered the restricted area of the stored stroke limit during the movement in the - direction.
	104	+ DIRECTION H/W LIMIT	The limit was sensed with the + direction limit switch of the corresponding axis.
	105	- DIRECTION H/W LIMIT	The limit was sensed with the - direction limit switch of the corresponding axis.
	106	FOLLOWING ERROR	The difference between the order location of the corresponding axis and the actual location is beyond the range of the parameter (P71: the limit value of the positional deviation).
MC	200	INVALID Character	The command or character causing a syntax error was found during program execution.
	201	Unavailable G Code	The G code not allowed to use was used.
	202	M Code beyond the Range	The M code not allowed to use was used.
	203	No Selected Program	The program to be executed in the AUTO operation mode is not selected, or the selected program doesn't exist.
	204	MISMATCH CIRCLE CENTER	If the arc center is ordered by I, J, and K in the arc order, there is a difference between the radius in the initial point and the radius in the endpoint.
	205	Very Small R Value	If the radius by R is ordered in an arc order, the ordered radius is small so that arc interpolation cannot be performed.
	206	No Number/Label for Jump	When an allocated sequence number is found by the GOTO and IF statements, the number N cannot be found.
	207	Formula Exceeding the Multiplicity	In the macro formula, the multiplicity by (,) is more than quintuple.
	208	Incorrect Number of Parentheses	The number of (,) in the MACRO formula is incorrect.
	209	The character next to = is not allowed to use.	The character not allowed to use next to = in the MACRO formula was ordered.
	210	The character next to (is not allowed to use.	The character not allowed to use next to (in the MACRO formula was ordered.
	211	The character next to) is not allowed to use.	The character not allowed to use next to) in the MACRO formula was ordered.
	212	The character next to an operator is not allowed to use.	The character not allowed to use next to the operator such as +,-,*,/, and so on in the MACRO formula was ordered.

	213	The character next to a constant is not allowed to use.	The character not allowed to use next to a constant such as a number in the MACRO formula was ordered.

Category	No.	Message	Details
MC	214	The character next to a variable is not allowed to use.	The character not allowed to use next to the variable L or Q in the MACRO formula was ordered.
	215	Variable Number beyond the Range	The number of the variable L or Q is beyond the range of the order number in the MACRO formula.
	216	'=' Omitted	The order of = was omitted in the MACRO formula.
	217	No Called Program	In case of a subprogram call, the called program doesn't exist.
	218	SubProgram-End Unfound	In case of a subprogram call, SubProgram-End cannot be found in the called program.
	219	Extremely Long Dwell Time	The value of a DWELL (G04) order is beyond the range.
	220	Beyond the Program Call Multiplicity	The subprogram call multiplicity is more than octuple.
	221	Beyond the Range of an Input Value	The data exceeding the allowed number of places were inputted during the input of numerical data.
	222	Division by 0 was ordered.	The command of division by 0 was ordered in the MACRO formula.
	223	The number after DO and END is beyond the range.	The number after DO and END in the WHILE statement was ordered as the number that is not one of 1 to 10.
	224	No Correspondence between DO and END	DO doesn't correspond with END in the WHILE statement.
	225	The WHILE Statement is beyond the allowed multiplicity.	The multiplicity of the WHILE statement is decuple or more.
	226	Unavailable Program-End	Program-End were not ordered in the program being executed.
	227	The MoveCommand is over S/W Limit	The movement to the restricted area of the stored stroke limit was ordered in the program.
	228	Beyond the Range of an P Value.	
	229	Beyond the Range of an Velocity Value.	
	230	The Point-NO Unfound.	
	231	'.' Omitted	The order of . was omitted in the MACRO formula.
	232	Point Number beyond the Range	
	233		
	234		
	235		
	236		
	237		
	238		
	239		

| Category	No.	Message	Details
PLC	300	The PLC program is compiled normally.	
	301	An error was found during the compilation of the PLC program.	
	302	The capacity of the PLC program exceeded the range.	
	303	END was not ordered in the PLC program.	
	304	It is beyond the contact point and data designation area in the PLC program.	
	305	The contact point and data designation number is used incorrectly in the PLC program.	
	306	The PLC program has an error in the contact point and data designation character.	
	307	It is the command not allowed to use in the PLC program.	
	308	There is a syntax error before and after the command in the PLC program.	
	309	It is the command that cannot be used in the program designated in the PLC program.	
	310	The command for the contact point LOAD is needed in the PLC program.	
	311	The maximum allowed number of the contact point LOAD is exceeded in the PLC program.	
	312	The command (- OR LOAD, AND LOAD) for BLOCK END is needed in the PLC program.	
	313	The MCS command is not used in the PLC program.	
	314	The maximum allowed number of the MCS commands is exceeded in the PLC program.	
	315	The maximum allowed number of the PULSE commands is exceeded in the PLC program.	
	316	The MPUSH command is not used in the PLC program.	
	317	The maximum allowed number of the MPUSH commands is exceeded in the PLC program.	
	318	There is an error before and after the MLOAD command in the PLC program.	
	319	There is an error before and after the MPOP command in the PLC program.	
	320	The names of JME or LABL to designate the branch are used repetitively in the PLC program.	
	321	The branch designation number (0~99) used as the name of JME or LABL for branch designation is beyond the range in the PLC program.	

Cate gory	No.	Message	Details
PLC	322	The JME or LABL name for branch designation is needed in the PLC program.	
	323	The maximum allowed number of the JMP, JPC, and JPN commands for branch is exceeded in the PLC program.	
	324	The JPM, JPC, and JPN commands are beyond the range of the branch designation number (0~99) for branching in the PLC program.	
	325	There are errors in the usage (location) of the JMP, JPC, and JPN commands for branching and the JME or LABL name for branch designation in the PLC program.	
	326	As the execution time of the interrupt program exceeds the set execution time during the execution of the PLC program so that a watch-dog error occurs.	
	327	As the execution time of the scan program exceeds the set execution time during the execution of the PLC program so that a watch-dog error occurs.	
	328		
	329		
	330		
	331		
	332		
	333		
	334		
	335		
	336		
	337		
	338		
	339		

Chapter 5 Additional function

1. Tool Function

The four numbers following address "T"are command numbers used to specify a tool number that selects a tool at the machine side and tool compensation number that selects tool compensation data.

Single T code can be used in a single block. For more details, please refer to the user manual of the manufacturer of the machine. The numbers following T code specify the tool number selection and tool compensation number.

T 00 00 ⟶ Tool compensation number
⟶ Tool selection number
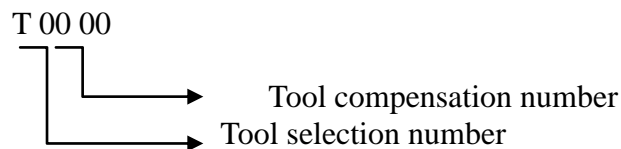
1. Tool selection number: Please refer to the user manual of the machine manufacturer for tool selection.
2. Tool compensation number: Compensation that corresponds to the specified compensation number is added to the compensation value of the selected tool.

[Note] 1. **Once the command is given to change the tool selection number, the calculation operation of the tool is initiated immediately at the turret type lathe. Thus, please be careful of the interference of the tool. The tool exchange command should be executed after moving to the place where no interference of the tool is applied.**

2. **Tool compensation No. 00 means CANCEL of the tool compensation.**

1.1 Compensation memory

[1] The memory of various data (shape, wear, NOSE_R) for tool compensation is called compensation memory, which can be checked and modified using the tool compensation screen. To select the tool compensation screen, press $\mathbb{OFS}$ key in any mode.

[2] The tool compensation numbers consist of 99 shape compensation sets and 99 wear compensation sets and the data are stored in EEPROM. That is, if a single compensation number is selected, wear and shape compensation values are specified simultaneously to make up a tool compensation amount.
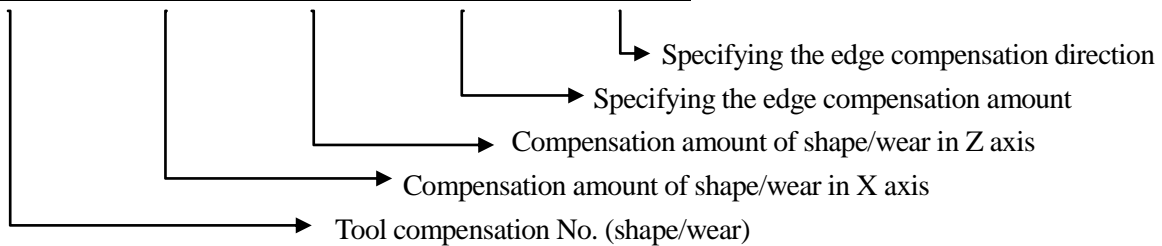
[Tool compensation amount = Shape compensation amount + Wear compensation amount]

[Structure of shape compensation memory] total 99 sets

NO.	X	Z	R	T
G01	0	0	0	0
G02	0	0	0	0
:	:	:	:	:
:	:	:	:	:
:	:	:	:	:
G99	0	0	0	0

[Structure of wear compensation memory] total 99 sets

NO.	X	Z	R	T
W01	0	0	0	0
W02	0	0	0	0
:	:	:	:	:
:	:	:	:	:
:	:	:	:	:
W99	0	0	0	0

→ Specifying the edge compensation direction

→ Specifying the edge compensation amount

→ Compensation amount of shape/wear in Z axis

→ Compensation amount of shape/wear in X axis

→ Tool compensation No. (shape/wear)

[Note] The data for compensation should be entered to the CNC unit prior to automatic operation after program edit, and operation should start after checking the value if the number is already inputted. Please refer to Tool screen of machine operation in Chapter 2 for the tool data input method.

1.2 Overview of the tool compensation operation

[1] As mentioned in the above, when a tool selected at T#### is moved, the compensation amount that corresponds to the tool compensation No. is added to the command value algebraically followed by the tool movements toward the added point. That is, once T code is specified for preferred block of tool position correction, T code is interpreted and the tool selection signal (BCD code) is transmitted. Then, compensation operation is initiated according to the compensation amount that corresponds to the specified compensation No.

[2] T code is MODAL, and it is valid until new T code is specified. If the auxiliary function lock (AFL) switch is ON, tool commands are invalid. If only a compensation amount is preferred to be changed, command of T code where only compensation number is changed is given.

[3] If no traverse command (G00, G01, G02, and G03) is supplied inside the block of T code command, only the selected tool is exchanged and the tool compensation amount is saved internally. Then, if traverse command is supplied in the next block, the saved compensation amount is added and the tool is moved to the added position. The feed rate during compensation follows the speed set in the traverse command.

[e.g.] T0302:   Tool No. 3 is selected and compensation amount that
                                    corresponds to compensation No. 2 is saved internally.
      G01 X...Z...F... :   The compensation amount that corresponds to the above saved
                                    compensation No. 2 is added.
      T0316:   The compensation No. is changed from   02 to 16, and the
                                    compensation amount is also changed.

[4] If T code and traverse command are specified in the same block, the tool is moved toward the goal value calculated after compensation. Thus, in the following example, TAPER angle is moved to the modified shape as much as the difference in compensation amount between T0202 and T0206.

[e.g.]   T0202;

            G01 X...Z...F...;

            G01 U+..W-..F...T0216;

[Note] 1. T code is modal, and it is valid as long as the compensation amount is not canceled. (It is not canceled even if the reset key is pressed.)
   2. To cancel the compensation amount, select compensation No. 0. [e.g.] T0100;
   3. The cancellation command of compensation amount must be issued prior to PROGRAM END(M02) command.
   4. The cancellation command of compensation amount should also be given in principle even in the case of command of automatic return to reference position (G28).
   5. The cancellation command of compensation amount must be given even in the case of command of reference position return check (G27). Once the command of G27 is issued with tool compensation, the compensation amount is added to the coordinate command, which may be a cause of the reference position return check alarm.

1.3 Example of tool compensation during program

[1] How to program without using G30 and G50

1. After soft jaw is processed, a work piece is clamped.
2. A mode is set to return to reference position, and each axis is returned to reference position manually.
3. JOG or MPG is selected to approach each axis to the work piece.
   (distance between each axis and cross-section of work piece: around 10mm for each axis)
4. Command of S500 M03 is issued in MDI mode and press the CYCLE START switch.
5. A mode is set to MPG and the JOG switch in the Z axis is pressed to select magnification (select one of X1, X10, or X100) and a tool is moved to a position of "ZO" to be processed. Then, the X-axis is selected to process the cross-section.
6. If **OFS** out of data input keys is selected, the shape compensation screen is displayed (if the wear compensation screen is selected, press the "shape compensation" from the menu key.) Place the pointer to the No. to be compensated, and "Z0" is entered. Then "ENT" is pressed to enter the machine coordinate value in the Z-axis with regard to Z0 value as Z compensation value. (here, Z-axis should not be moved.)
7. The surface of the outer diameter of workpiece is processed using MPG (arbitrary diameter), and the Z-axis is moved after stopping the main spindle to facilitate convenient measurement of outer diameter. Then, the outer diameter of the workpiece is measured using a measurement device.
8. If the measurement value of the outer diameter is φ50.3, **ENT** is pressed after entering I50.3 to the shape compensation in the tool compensation screen. Then, machine coordinate value in the X-axis with respect to X50.3 is entered as the X compensation value.
9. The above method is the same to other tool.
   [e.g.] Example of program at GANG TOOL TYPE lathe
   N0010 G00 X100. Z100. T0001 M08;
   N0020 G50 S2000 M03;
   N0030 G96 S150;
   N0040 G00 X35. Z2. ;
   N0050 G92 X30. Z-25. F0.2;
   N0060 G94 X0 Z0;
   N0070 G00 Z50. ;
   N0080 M01;
   N0090 G97 S1000;
   N0100 G00 X0 Z2. T0002;
   N0110 G01 Z-5. F0.15;
   N0120 G00 Z5. M09;
   N0130 X100. Z100. T0001;
   N140 M02;

[2] Example of set up of the second reference position in the program using G30 and G50

1. A mode is set to return to reference position, and each axis is returned to reference position manually.
2. JOG or MPG is selected to approach each nr to the work piece.
   (distance between each axis and cross-section of work piece: around 10mm for each axis)
3. Command of S500 M03 is issued in MDI mode and press the CYCLE START switch.
4. A mode is set to MPG and the JOG switch in the Z axis is pressed to select magnification (select one of X1, X10, or X100) and a tool is moved to a position of "ZO" to be processed. Then, the X-axis is selected to process the cross-section.
5. Select MDI to perform G50 Z0.

6. The surface of the outer diameter of workpiece is processed using MPG (arbitrary diameter),and the Z-axis is moved after stopping the main spindle to facilitate convenient measurement of outer diameter. Then, the outer diameter of the workpiece is measured using a measurement device.

7. After processing the outer diameter of the workpiece, select MDI if the measurement value is φ50.3. G50 X50.3 is entered, and **ENT** is pressed followed by pressing the CYCLE START key to set X50.3.

8. If the second reference position is X100. and Z100., G00 X100. Z100. is entered in the MDI mode. **ENT** is pressed followed by pressing CYCLE START to move to the second reference position of X100. and Z100.

9. **MON** Selection → **Parameter** selection → P000 input → **ENT** are performed in sequence to search parameter P000.

10. If the cursor is placed in the preferred N No. and push **INS**, then machine coordinate value X and Z values are automatically inputted. (Refer to Parameter)

[Note] 1. The shape and wear compensation values of the reference holder are ZERO(0) for each of X and Z.
2. Once the command of G30 and G50 (coordinate system setup) is issued, the reference of shape compensation is always the shape compensation value of the reference holder.

2. Linear interpolation to specify an angle

 G01 Z(W)....A....F....;
 G01 X(U)....A....F....;

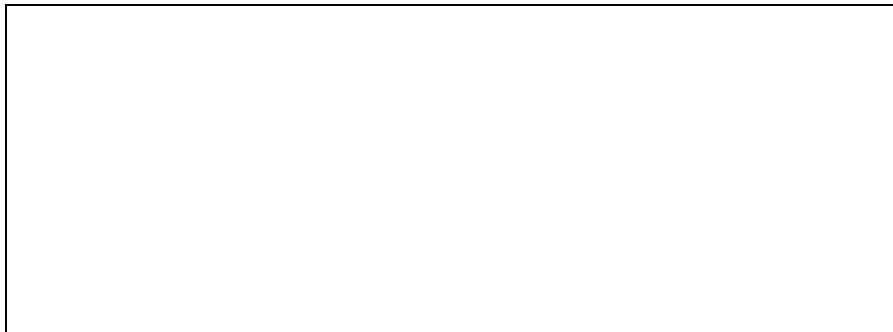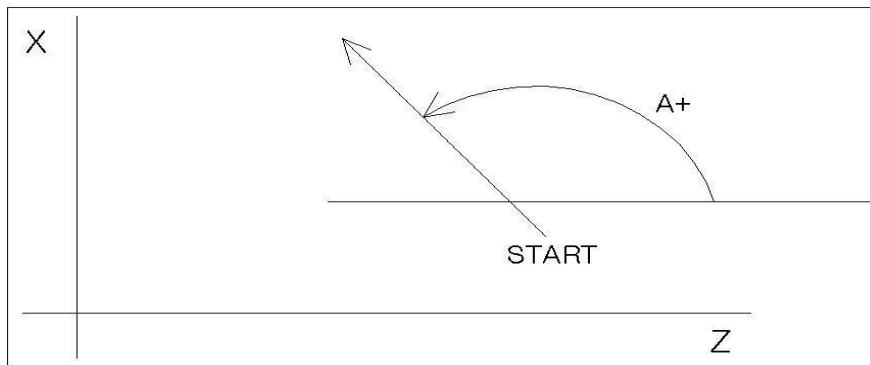It is a method of linear interpolation by issuing a command of target position angle from the current position.
Here, **angle (A)** refer to an angle made by the straight line from a certain point to the end point in the Z-axis.

 1. Direction to specify an angle

A='-'	Clockwise direction
A='+'	Anti-clockwise direction

 2. Angle value range: -360.000 to 360.000
 3. Unit: angle (degree)

[Note] The angle command is ignored in the following cases.
  1. Two commands of X and Z are issued in a single block simultaneously.
  2. No intersection is made by the given angle.

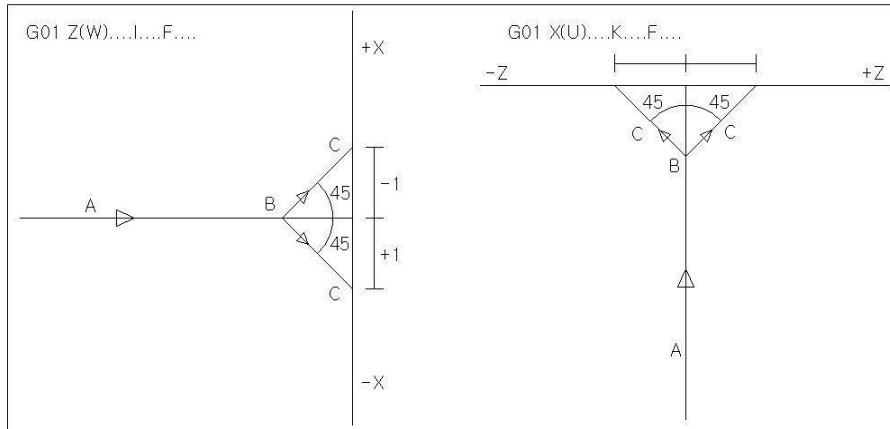## 3. Chamfering and CORNER R

Chamfering and **CNRNER R jobs** can be simply performed using G01 between two blocks angled at the right angle.

[1] How to command chamfering

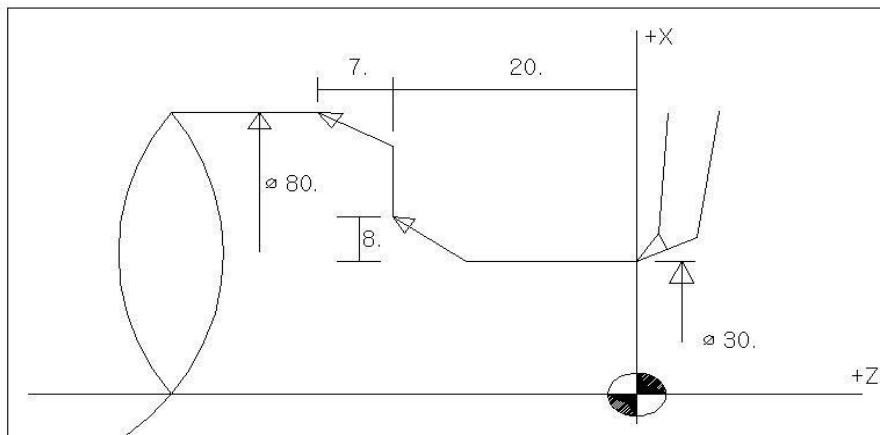**G01 Z(W)....I....;**
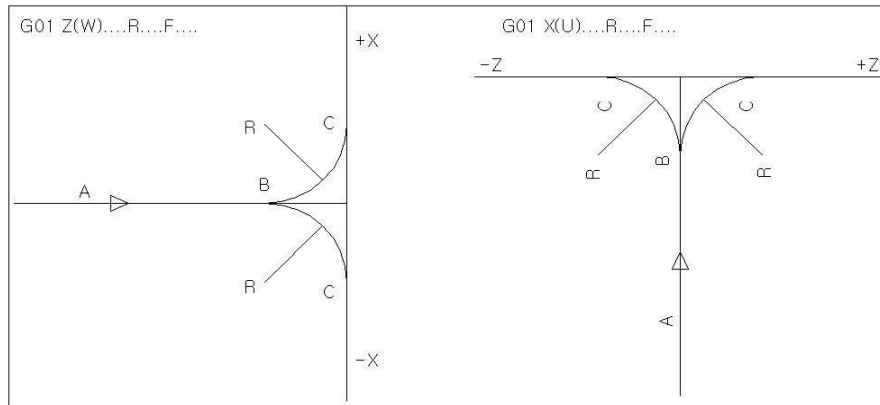**G01 X(U)....K....;**

[e.g.]
G00 X30. Z0.;
G01 Z-20. I8. F30;
(G01) X80. K-7.;

[2] How to command CORNER R

G01 Z(W)....R....;
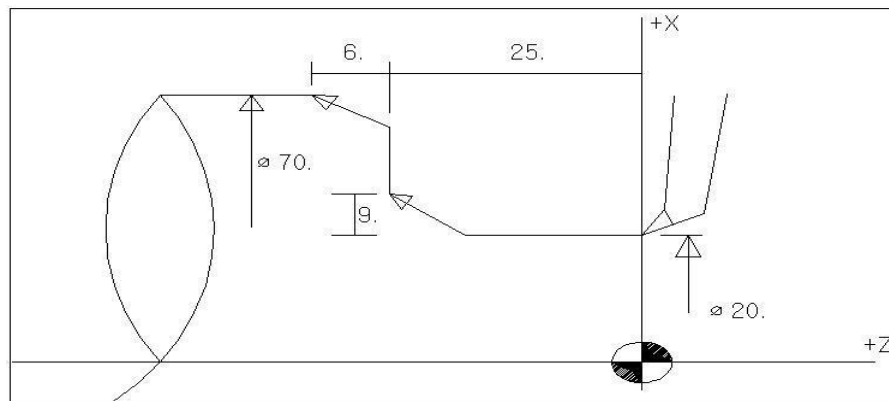**G01 X(U)....R....;**

[e.g.]
G00 X20. ZO.;
G01 Z-25. I9. F30;
(G01) X70. K-6.;

[Note] 1. **Chamfering and CORNER R should only be used in the following case: the movement of G01 command is on one of of X or Z axis, and the next block is on one of Z or X axis angled at the right angle with the previous block movement.**
2. **Chamfering and CORNER R cannot be used in screw cutting block.**
3. **The stop position of a single block is C position where chamfering or CORNER R is performed.**
4. **The movement of chamfering or CORNER R command block or the next block should be larger than that of chamfering or CORNER R.**
5. **It is also valid even in R compensation.**
6. **I and K values refer to radius command.**