Changes for the Better

MITSUBISHI ELECTRIC

MITSUBISHI CNC

700/70 Series

PLC Programming Manual

IB-1500036(ENG)-F

Introduction

These specifications are the programming manual used when creating the sequence program with the PLC development software, or Mitsubishi Electric Co.'s integrated FA software MELSOFT series (GX Developer).
The PLC (Programmable Logic Controller) instructions are largely categorized into the basic instructions, function instructions and exclusive instructions. There are many types of instructions. The instructions can be used according to the purpose and application such as the PLC support function used when supporting the user PLCs.
In addition to the explanation of instructions and functions, the environment to develop the user PLC using GX Developer, especially the usage unique to MITSUBISHI CNC, is described. Explanations on the built-in PLC edit function (onboard PLC edit function) operations are also given.

Details described in this manual

⚠ CAUTION

⚠ An effort has been made to describe special handling of this machine, but items that are not described must be interpreted as "not possible".

⚠ Some screens and functions may differ or some functions may not be usable depending on the NC version.

General precautions

Refer to each manual for details on the MITSUBISHI CNC Series PLC, and for details on the various tools in this manual.
The explanations and screens for the various tools in this manual may differ slightly according to the tool version. Refer to the respective manual for details.

[MELSEC Series Software Package Manual]

GX Developer Version 8 Operating Manual (Startup Section)	GXDEV8-0-IN-E	13JU40	SH-080372E
GX Developer Version 8 Operating Manual	GXDEV8-0-E	13JU41	SH-080373E
GX Converter Version 1 Operating Manual	SW0D5-CNVW (OPE)-E	13J949	IB-080004E

(Caution)

- The version numbers are current as of the editing of this manual, but may be updated in the future.
- GX Developer Version 8 (Model SW8D5C-GPPW) is the new name of the old "Windows Version GPP Function Software package" (common name GPPW).

Precautions for Safety

Always read the specifications issued by the machine tool builder, this manual, related manuals and attached documents before installation, operation, programming, maintenance or inspection to ensure correct use.
Understand this numerical controller, safety items and cautions before using the unit.
This manual ranks the safety precautions into "DANGER", "WARNING" and "CAUTION".

⚠ **DANGER**	When there is a great risk that the user could be subject to fatalities or serious injuries if handling is mistaken.
⚠ **WARNING**	When the user could be subject to fatalities or serious injuries if handling is mistaken.
⚠ **CAUTION**	When the user could be subject to injuries or when physical damage could occur if handling is mistaken.

Note that even items ranked as " ⚠ **CAUTION**", may lead to major results depending on the situation. In any case, important information that must always be observed is described.

⚠ DANGER

Not applicable in this manual.

⚠ WARNING

Not applicable in this manual.

⚠ CAUTION

1. Items related to product and manual

- ⚠ For items described as "Restrictions" or "Usable State" in this manual, the instruction manual issued by the machine tool builder takes precedence over this manual.
- ⚠ An effort has been made to describe special handling of this machine, but items that are not described must be interpreted as "not possible".
- ⚠ This manual is written on the assumption that all option functions are added. Refer to the specifications issued by the machine tool builder before starting use.
- ⚠ Refer to the Instruction Manual issued by each machine tool builder for details on each machine tool.
- ⚠ Some screens and functions may differ or some functions may not be usable depending on the NC version.

2. Items related to start up and maintenance

- ⚠ Read this manual carefully and confirm the safety enough before executing the operation of the program change, forced output, RUN, STOP, etc. during operation. Operation mistakes may cause damage of the machine and accidents.

(To be continued to the next page.)

⚠ CAUTION

3. Items related to program development

⚠ Always observe the cautions before development to develop a program.

⚠ If the data transferred does not follow the file name rule, the CNC will mistake it for another data, resulting in unexpected operation, e.g. PLC program erasure.

⚠ Do not read a sequence program on which a conversion error occurred into the GX Developer. The file may include unexpected contents to result an illegal operation.

⚠ When an error occurred at GX Developer On-line function, the error message may not explain exactly the state in the CNC side.
Always refer to the error list.

Contents

I OUTLINE

II PROGRAMMING EXPLANATION

III PERIPHERAL DEVELOPMENT ENVIRONMENT

IV Explanation of built-in editing function

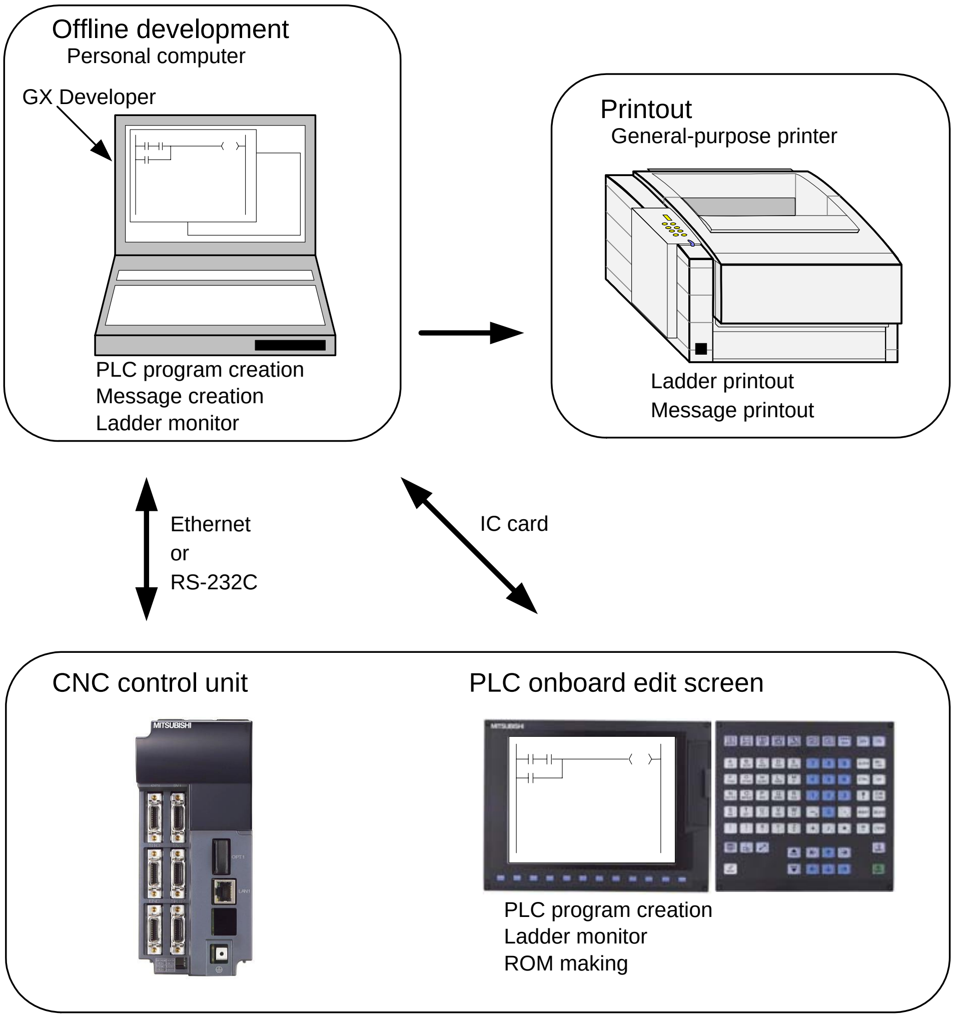
V APPENDIX

I OUTLINE

1. System Configuration

1.1 System Configuration for PLC Development

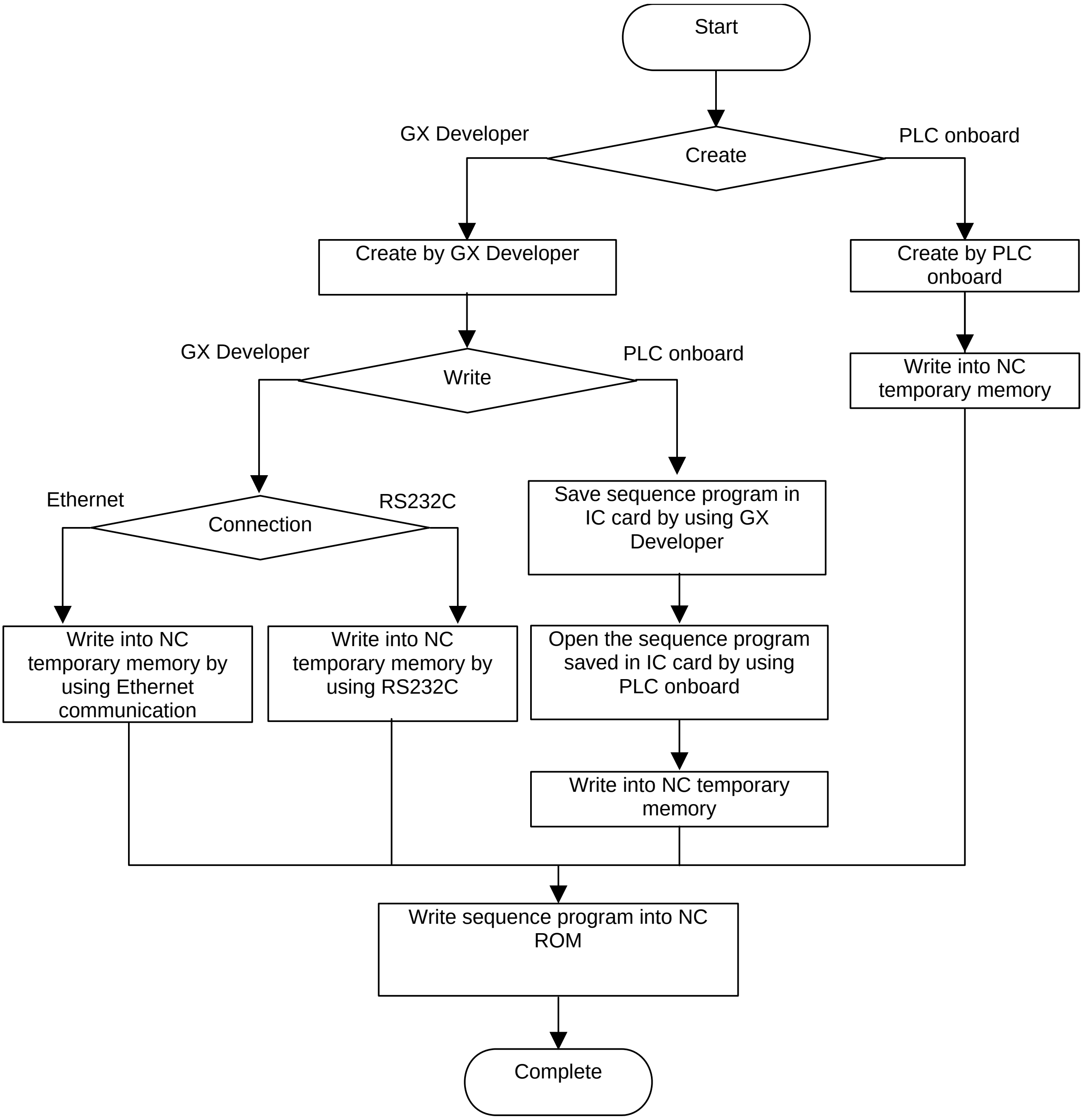
The general configuration of the development environment is shown below.
Most of the development work is carried out with "GX Developer", which runs on a personal computer. GX Developer and the CNC control unit are connected with Ethernet or an RS-232C cable at this time.
On the CNC unit PLC onboard edit screen, it is possible to use the data saved with GX Developer or develop PLC programs, as well. Note that some functions may be limited. (Print output, Japanese input, etc.)



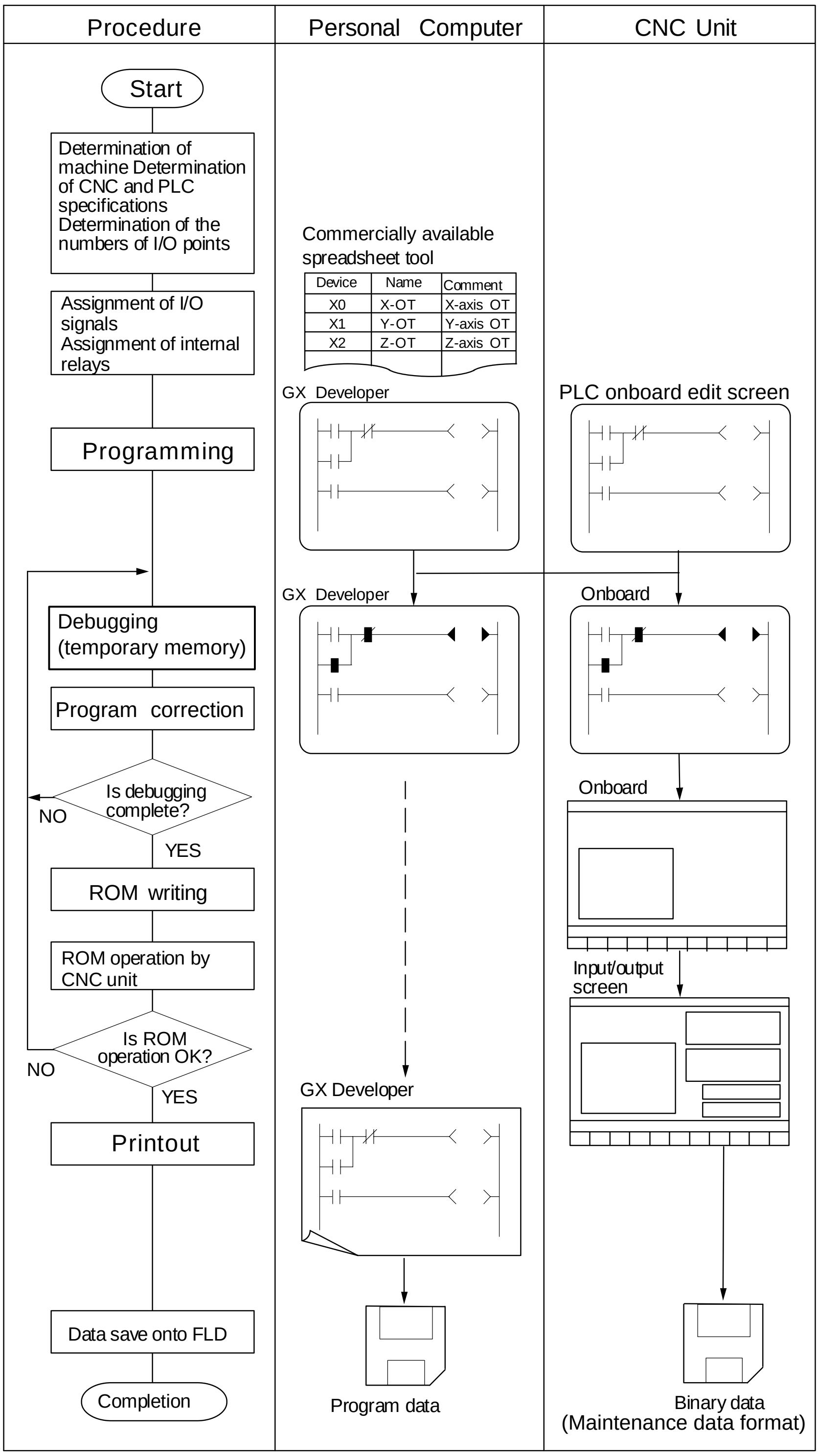
General configuration of development environment

1.2 User PLC (Ladder) Development Procedure

User PLC programs can be created and input by means of either GX Developer installed in the external PC or PLC onboard edit screen.
Procedures for creating/inputting PLC programs with either method are shown below.



Next, procedures for creating sequence programs are shown below.



The data created with the commercially available spreadsheet tool can be used as ladder comment data.

Use GX Developer for programming.
After completion, download the data through RS-232C.
A new program can also be created by using the CNC onboard function.

Perform monitoring/correction with GX Developer's online function or onboard function.

Perform ROM making operations with F-ROM write screen.

Output binary data with maintenance data format using input/output screen.

Printout to a commercial printer connected with the personal computer from GX Developer.

Program data:
Saved using GX Developer

Binary data:
Saved using input/output screen

II PROGRAMMING EXPLANATION

1. Outline

This programming manual is used when creating a sequence program for this CNC using the MELSEC PLC development software package (GX Developer).
The PLC (Programmable Logic Controller) instructions are largely categorized into the basic instructions, function instructions and exclusive instructions. There are many types of instructions. The instructions can be used according to the purpose and application such as the PLC support function used when supporting the user PLCs.

2. PLC Processing Program

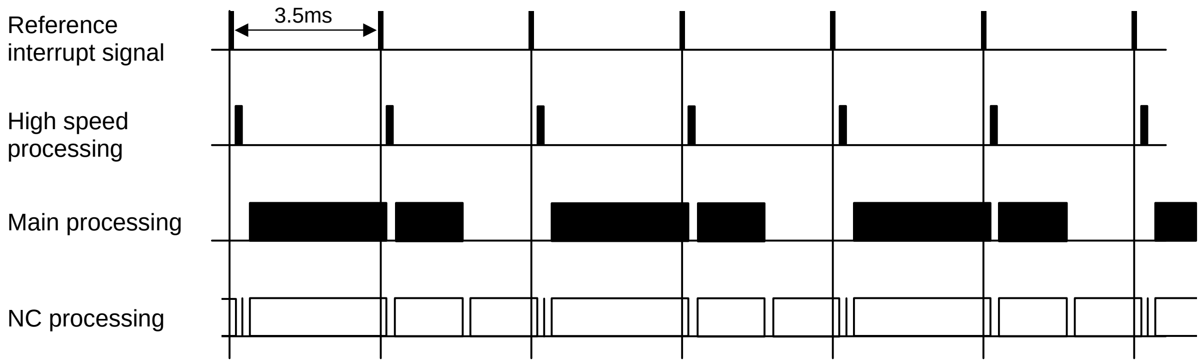
2.1 PLC Processing Program Level and Operation

The details of the user PLC processing level and the time chart are shown below.

PLC processing level

Program name	Description (frequency, level, etc.)
Initial processing program	This program starts only once at power ON. When this program operates, machine input and operation board input are not read.
High-speed processing program	This program starts periodically at each standard interrupt signal. This program has the highest level as a program that starts periodically. It is used in signal processing where high-speed processing is required. The steps for high-speed processing program should be up to 1000 steps with basic instructions. **(Application example)** Position count control of turret and ATC magazine **(Note)** The standard interrupt signal cycle differs according to each model, and must be confirmed separately.
Main processing program	This program runs constantly except during the high-speed process program. When the user PLC one-scan process is completed, the next scan process starts at the next reference interrupt signal cycle.

When reference interrupt signal is 3.5ms



PLC processing program operation timing chart

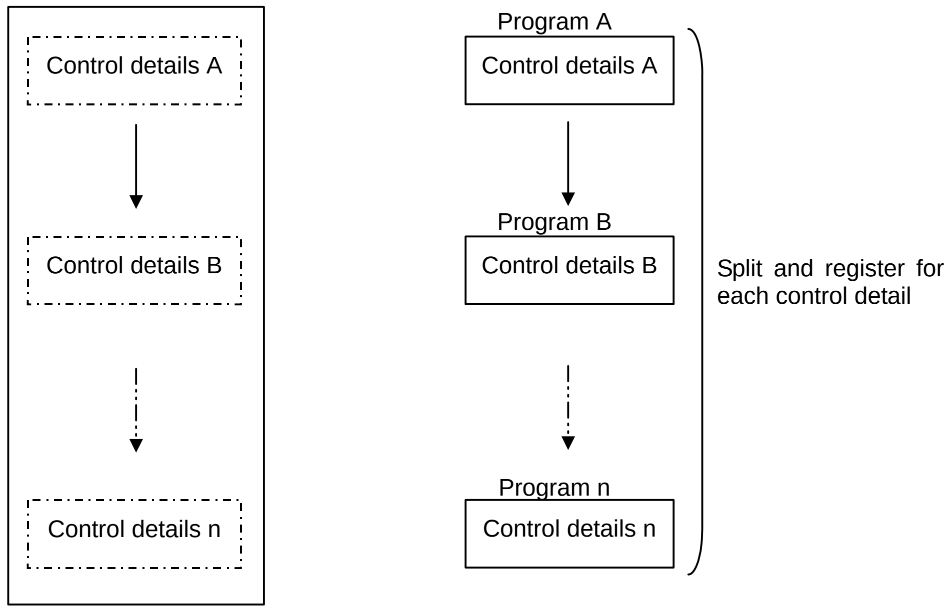
2.2 Outline of PLC Processing Program (Two Program Method)

The MITSUBISHI CNC 700 series program execution control methods include the conventional method which controls with one program, and the method that splits the program into multiple sections for each control unit.
When splitting into multiple programs, the order for executing the split programs can be designated on the setting screen. This is called the multi-programming function.

- Method controlling with one program (conventional method) : Independent program method
- Method splitting control into multiple programs : Multi-program method

Control with one program (Independent control method)

Control by splitting into multiple programs (Multi-program method)



2.3 Independent Program Method

This method lays importance on compatibility with the conventional models.
One sequence program can be stored. The execution type and head of processing are designated with reserved labels.
The execution type and execution order cannot be designated on the setting screen.

- Initialization process (reserved label P4003) : This starts up only once when the power is turned ON.
- High-speed process (reserved label P4001) : This starts up at the standard interrupt cycle.
- Main process (reserved label P4002) : This starts up constantly except during the high-speed process.

2.4 Multi-program Method

Several sequence programs can be registered in the CNC and sequentially executed.
By using this function, the sequence program can be split into each process and developed.
With the multi-program method, the execution type and execution order are designated on the GX Developer setting screen, and the parameter files are sent to the NC.
The execution type and the head of the process cannot be designated with reserved labels.

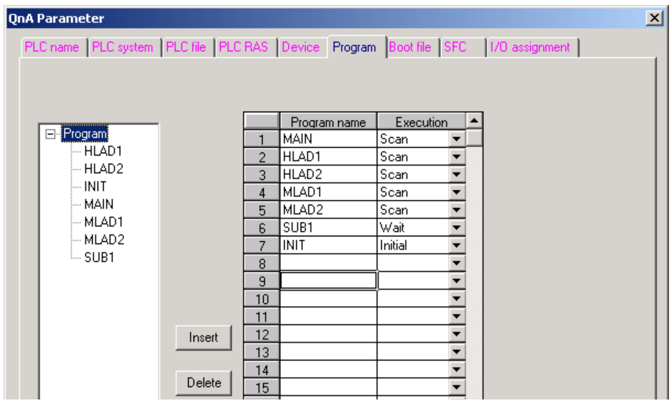
2.4.1 Number and Types of Registerable Programs

Up to 20 sequence programs can be registered. Only one execution type can be set in one program. The following five types of execution types can be used.

- "Initial" (Initialization process) : This starts up only once when the power is turned ON.
- "Scan" (High-speed process) : This starts up at the standard interrupt cycle.
- "Scan" (Main process) : This starts up constantly except during the high-speed process.
- "Standby" (Standby process) : This is called from the high-speed process or main process.
- "Low speed" : This execution type is not used.

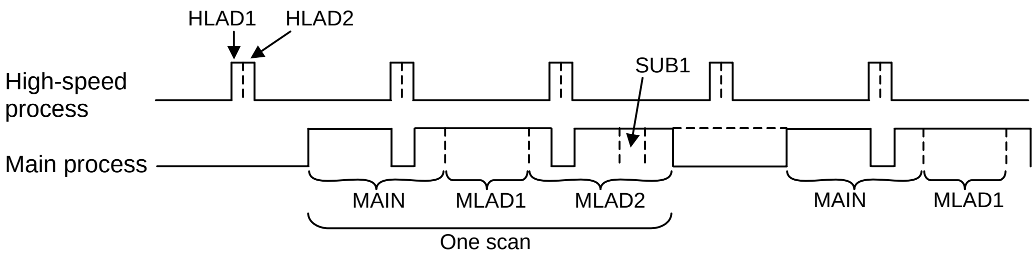
2.4.2 Program Execution Order

Several programs are executed in a predetermined order. They are not executed simultaneously. The order is determined with the development tool (GX Developer or onboard) setting screen. The programs are executed from the smallest number in the same execution type. An example of the setting screen for GX Developer is shown below.



The execution order when seven sequence programs are registered in the CNC, as shown in the above setting screen, is indicated below.

Program name	Execution type	Execution order	Remarks
INIT	Initialization sequence program	1	Starts up only once when the power is turned ON.
HLAD1	High-speed process execution program Execution type is set as "Scan"	1	"Scan type" for which program name starts with "H"
HLAD2		2	
MAIN	Main process sequence program Execution type is set as "Scan"	1	"Scan type" for which program name does not start with "H"
MLAD1		2	
MLAD2		3	
SUB1	Standby sequence program	1	Here, subroutine that is called from MLAD2 with CALL instruction is stored

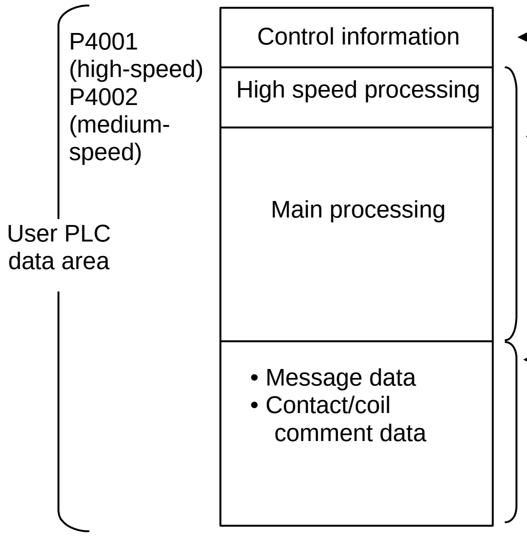


[Caution] If the process jumps to END (P4005) in the sequence program, the process will jump to the end of each process (high-speed, main) instead of the end of the program.

2.5 User Memory Area Configuration and Size

The user memory area approximate configuration and size are shown below. The configuration and size differ according to the program method.

2.5.1 Independent Program Method



(1) Internal information table of User PLC
(The table is automatically generated.)

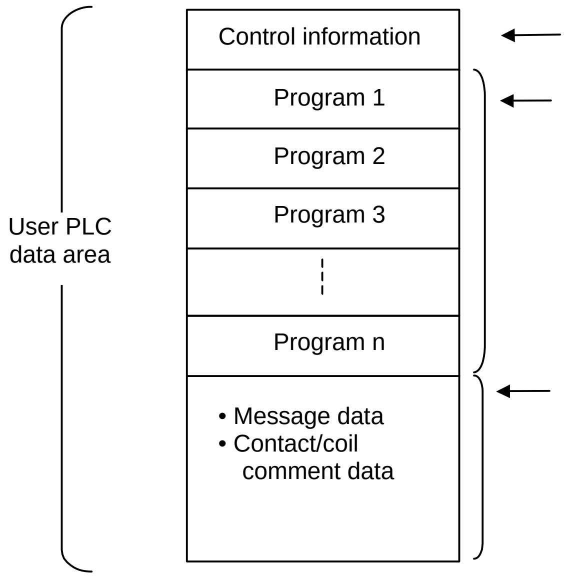
(2) Sequence program storage area
This is not required for programs other than the main process.
The initialization, high-speed and main process program order is arbitrary.

Total	700 Series	42000 steps
	70 Series type A	32000 steps
	type B	20000 steps

(3) Data storage area (Other than sequence program)
• Alarm messages
• Operator messages
• PLC switches
• Load meter (Each can be stored in eight languages)
• Contact/coil comment data, etc.

Total 256 Kbyte

2.5.2 Multi-program Method



(1) Internal information table of User PLC
(The table is automatically generated.)

(2) Sequence program storage area
The initialization, high-speed, main and standby processes can be split into multiple programs for each control unit and stored.
The program storage order is arbitrary.
There must be at least one main process.

Total	700 Series	42000 steps
	70 Series type A	32000 steps
	type B	20000 steps

(3) Data storage area (Other than sequence program)
• Alarm messages
• Operator messages
• PLC switches
• Load meter (Each can be stored in eight languages)
• Contact/coil comment data, etc.

Total 256 Kbyte

2.6 Storing PLC Processing Program and Execution Mode

The user memory area storage method and the PLC processing program execution method are explained. User memory area is stored in the internal flash ROM (internal F-ROM) and a sequence program is executed according to the following path.

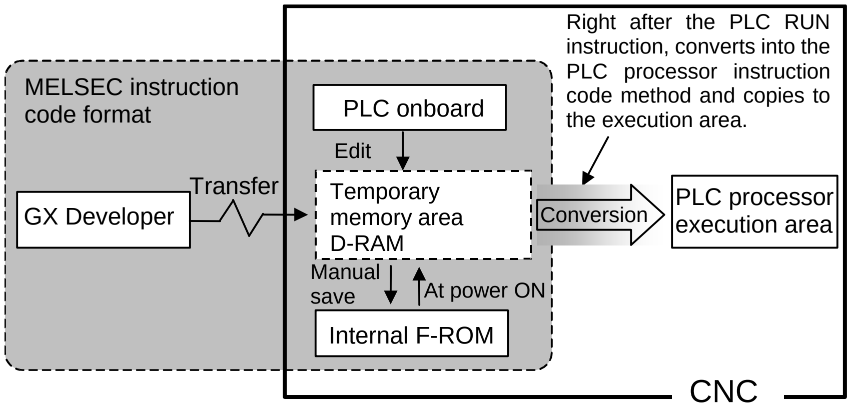
2.6.1 Path from Storage to Execution

(1) During PLC development

Sequence program data transferred from development environment such as GX Developer or PLC onboard is stored in the volatile RAM (hereinafter, D-RAM) for the temporary memory. The sequence program is transferred to the PLC processor execution area before PLC execution, and is then executed. The D-RAM in the temporary memory is not held when the power is turned OFF. If the data needs to be held even after the power is turned OFF, it must be stored in the internal F-ROM.

(2) At power ON

The data is transferred from the internal F-ROM to the PLC processor execution area via the temporary memory D-RAM, and is then executed.



2.6.2 Conversion of Instruction Code at Execution

In the internal F-ROM/temporary memory area shown on the left in the figure above, a sequence program is stored in the instruction code format that is compatible with the MELSEC sequencer. During execution, however, a sequence program is analyzed to optimize the references and/or converted into the PLC processing processor instruction code for the CNC. Thus, the length (number of steps) of an instruction for each instruction changes before and after the conversion. Refer to "6.2 Instruction List" for details on the number of steps during storage and execution for each instruction.

2.6.3 How to Confirm the Number of Steps at Storage/Execution

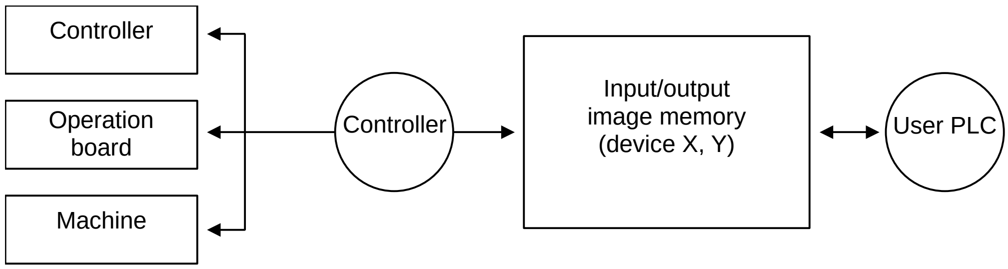
The number of steps under the PLC development environment (GX Developer, PLC onboard edit function) is usually all displayed as the number of steps at "storage".
The number of steps at execution can be checked with some dedicated methods. Refer to "III PERIPHERAL DEVELOPMENT ENVIRONMENT 5.2.4 (2) How to confirm the size of execution area" or "IV EXPLANATION OF BUILT-IN EDITING FUNCTION 13.2 (9) EXECUTE STEP" for details.

3. Input/Output Signals

3.1 Input/Output Signal Types and Processing

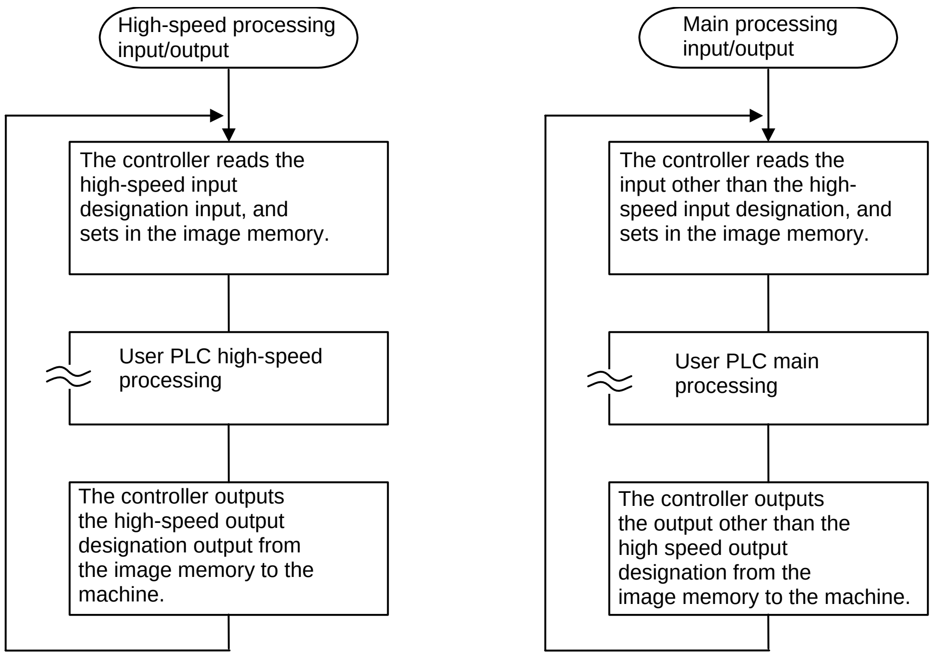
The input/output signals handled in user PLC are as follows:
(1) Input/output from/to controller
(2) Input/output from/to operation board (Note 1)
(3) Input/output from/to machine

The user PLC does not directly input or output these signals from or to hardware or controller; it inputs or outputs the signals from or to input/output image memory. For the reading and writing with the hardware or controller, the controller will perform the input/output according to the level of the main process or high-speed process.



(Note 1) The operation board here refers to when the remote I/O is installed on the communication terminal.

Concept of input/output processing



Input/output processing conforming to program level

The table below shows whether or not high-speed input/output can be performed.

Whether or not high-speed input/output can be performed

	High-speed input specification	High-speed output specification
Input signal from control unit	×	×
Output signal to control unit	×	×
Input signal from machine	○ (2-byte units)	×
Output signal to machine	×	○ (2-byte units)
Input signal from operation board	×	×
Output signal to operation board	×	×
Input signal from MELSEC when connected to MELSEC	×	×
Output signal to MELSEC when connected to MELSEC	×	×

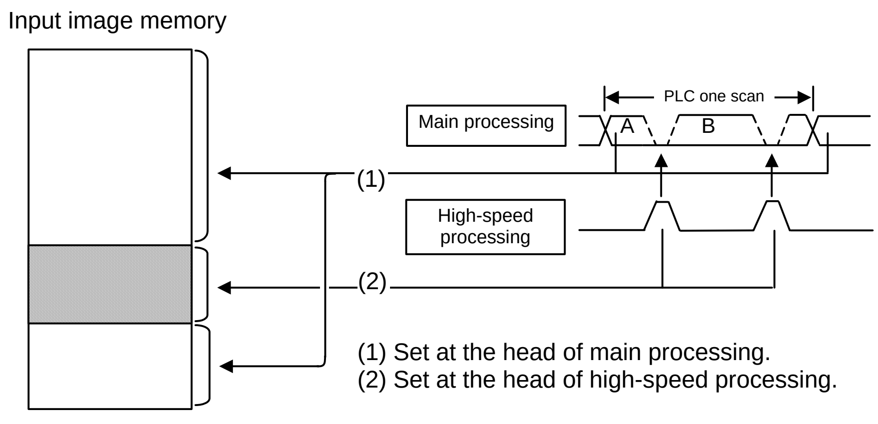
○ : Possible × : Not possible

The operation board here refers to when the remote I/O is installed on the communication terminal.

3.2 Handling of Input Signals Designated for High-speed Input

The input/output signals used in user PLC are input/output for each program level as shown in the figure below. In high-speed processing, input/output signal for which high-speed input or output designation (parameter) is made is input or output each time the high-speed processing program runs. In main processing, signals other than the high-speed input/output designation are input/output.

When high-speed input designation signal is used in main processing, the input signal may change within one scan because high-speed processing whose level is higher than main processing interrupts. Input signal which must not change within one scan should be saved in temporary memory (M), etc., at the head of main processing and the temporary memory should be used in the main program, for example.



The hatched area is high-speed input designation part. Whenever the high-speed processing program runs, data is reset in the hatched area. Thus, the signal in the hatched area may change in main processing (A) and (B) because the high-speed process interrupts between (A) and (B) and re-reads the input signal in the hatched area.

3.3 High-speed Input/Output Designation Method

High-speed input/output is designated by setting the corresponding bit of the bit selection parameter as shown below.

(1) High-speed input designation

	7	6	5	4	3	2	1	0	Bit
Bit selection parameter #6457	X70 ~ X7F	X60 ~ X6F	X50 ~ X5F	X40 ~ X4F	X30 ~ X3F	X20 ~ X2F	X10 ~ X1F	X00 ~ X0F	These bits correspond to the low-order byte (bits 0 to 7) of file register R7828
#6458	XF0 ~ XFF	XE0 ~ XEF	XD0 ~ XDF	XC0 ~ XCF	XB0 ~ XBF	XA0 ~ XAF	X90 ~ X9F	X80 ~ X8F	These bits correspond to the high-order byte (bits 8 to F) of file register R7828

If there are multiple remote I/O connection channels, designate RIO2 with the same configuration as RIO1 above.

RIO2 : Designate X100 to X1FF with bit selection parameters #6459 and #6460

RIO3 : Designate X200 to X2FF with bit selection parameters #6465 and #6466

(2) High-speed output designation

	7	6	5	4	3	2	1	0	Bit
Bit selection parameter #6461	Y70 ~ Y7F	Y60 ~ Y6F	Y50 ~ Y5F	Y40 ~ Y4F	Y30 ~ Y3F	Y20 ~ Y2F	Y10 ~ Y1F	Y00 ~ Y0F	These bits correspond to the low-order byte (bits 0 to 7) of file register R7830
#6462	YF0 ~ YFF	YE0 ~ YEF	YD0 ~ YDF	YC0 ~ YCF	YB0 ~ YBF	YA0 ~ YAF	Y90 ~ Y9F	Y80 ~ Y8F	These bits correspond to the high-order byte (bits 8 to F) of file register R7830

If there are multiple remote I/O connection channels, designate RIO2 with the same configuration as RIO1 above.

RIO2 : Designate Y100 to Y1FF with bit selection parameters #6463 and #6464

RIO3 : Designate Y200 to Y2FF with bit selection parameters #6473 and #6474

- As listed above, one bit corresponds to two bytes (16 points).
- Input or output in which 1 is set in the table is not performed at the main processing program level.
- Although the number of bits set to 1 is not limited, set only necessary ones from viewpoint of overhead.
- High-speed input/output designation corresponds to the bit selection parameter and can be set in the parameter. However, it is recommended to set in a sequence program to prevent a parameter setting error, etc.

(Example) —[MOV H3 R7828]— ..... To designate X00 to X0F, X10 to X1F (bit 0 and 1 for H3)

4. Parameters

4.1 PLC Constants

The parameters that can be used in user PLC include PLC constants set in the data type. The PLC constants include the basic area and the extended area.

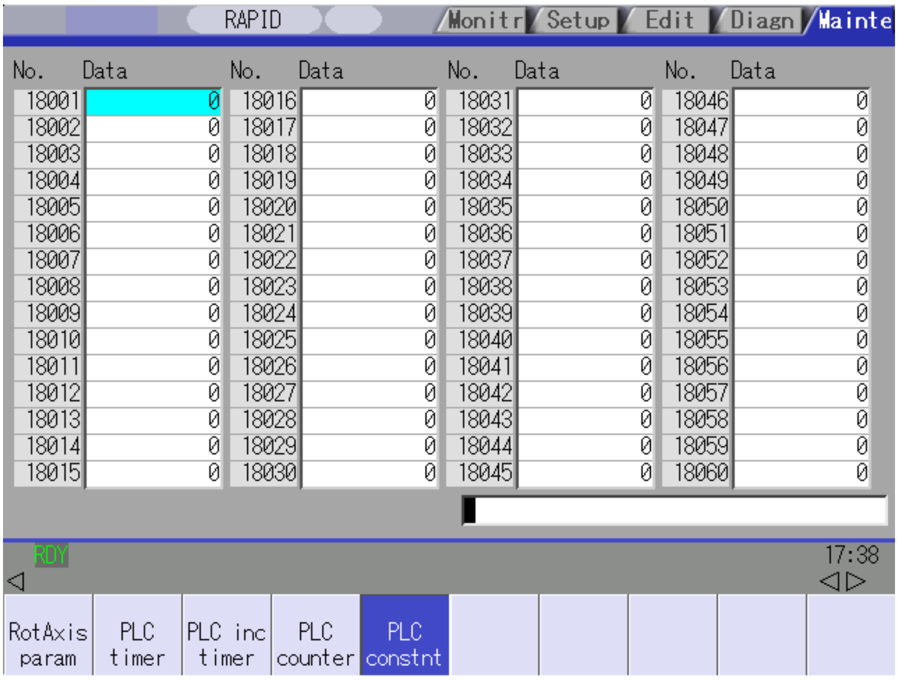
(1) Basic area

Set up data is stored in a file register and is backed up. In contrast, if data is stored in the file register corresponding to PLC constant by using sequence program MOV instruction, etc., it is backed up. However, display remains unchanged. Display another screen once and then select the screen again. 150 PLC constants are set (the setting range is ±8 digits). (Signed 4-byte binary data)

PLC constant No. and R register correspondence table

Item (# No.)		Corresponding register	Details	Setting range
PLC constant #1 (#18001)	LOW side	R7500	Data type parameters which can be used in user PLC	-99999999 to 99999999 (Signed 8-digit integer)
	HIGH side	R7501		
PLC constant #2 (#18002)	LOW side	R7502		
	HIGH side	R7503		
PLC constant #3 (#18003)	LOW side	R7504		
	HIGH side	R7505		
…		…		
PLC constant #148 (#18148)	LOW side	R7794		
	HIGH side	R7795		
PLC constant #149 (#18149)	LOW side	R7796		
	HIGH side	R7797		
PLC constant #150 (#18150)	LOW side	R7798		
	HIGH side	R7799		

PLC constant setting and display screen



(2) Extended area

Up to 750 PLC constants can be secured by using the user backup area (R8300 to R9799) as the extended area. The extended area start register and number can be set with the parameters. A total of 900 PLC constants can be set with 150 (#18001 to #18150) in the basic area and up to 750 (#18151 to #18900) in the extended area.
Set up data is stored in a file register and is backed up. In contrast, if data is stored in the file register corresponding to PLC constant by using sequence program MOV instruction, etc., it is backed up. However, display remains unchanged. Display another screen once and then select the screen again.
750 PLC constants are set (the setting range is ±8 digits). (Signed 4-byte binary data)

PLC constant No. and R register correspondence table

Item (# No.)		Corresponding register	Details	Setting range
PLC constant #151 (#18151)	LOW side	R8300 to R9799 The area for the number determined with parameter #1326 is continuously secured.	Data type parameters which can be used in user PLC	-99999999 to 99999999 (Signed 8-digit integer)
	HIGH side			
PLC constant #152 (#18152)	LOW side			
	HIGH side			
PLC constant #153 (#18153)	LOW side			
	HIGH side			
…				
PLC constant #898 (#18898)	LOW side			
	HIGH side			
PLC constant #899 (#18899)	LOW side			
	HIGH side			
PLC constant #900 (#18900)	LOW side			
	HIGH side			

The extended area quantity is set with basic common parameter #1326.

# No.	Item	Details	Setting range
1326	PLC Const Ext. Number	• Set number of PLC constant extension points. • This is valid after the power is turned OFF and ON.	0 to 750

4.2 Bit Selection Parameters

The parameters that can be used in user PLC include bit selection parameters set in the bit type.
Set up data is stored in a file register and is backed up.
When using bit operation in a sequence program, use a word device bit-designation format.
If data is stored in the file register corresponding to bit selection by using the MOV instruction etc., it is backed up. However, display remains unchanged. Once display another screen and again select screen.
The corresponding between the bit selection parameters and file registers is listed below. The setting and display screens are also shown.

Bit selection parameter (# No.)		Corresponding register	Details	Setting range
#1	(#6401)	R7800-Low side	Use bit selection parameters #6401 to #6448 freely.	8 bits
#2	(#6402)	R7800-High side		
#3	(#6403)	R7801-L		
#4	(#6404)	R7801-H		
…		…		
#45	(#6445)	R7822-L		
#46	(#6446)	R7822-H		
#47	(#6447)	R7823-L		
#48	(#6448)	R7823-H		
#49	(#6449)	R7824-L	Bit selection parameter #6449 to #6496 are PLC operation parameters used by the machine tool builder and MITSUBISHI. The contents are fixed.	
#50	(#6450)	R7824-H		
#51	(#6451)	R7825-L		
#52	(#6452)	R7825-H		
…		…		
#93	(#6493)	R7846-L		
#94	(#6494)	R7846-H		
#95	(#6495)	R7847-L		
#96	(#6496)	R7847-H		
#97	(#6497)	R7848-L	Use bit selection parameters #6497 to #6596 freely.	
#98	(#6498)	R7848-H		
#99	(#6499)	R7849-L		
#100	(#6500)	R7849-H		
#101	(#6501)	R7850-L		
#102	(#6502)	R7850-H		
#103	(#6503)	R7851-L		
#104	(#6504)	R7851-H		
#105	(#6505)	R7852-L		
#106	(#6506)	R7852-H		
…		…		
#187	(#6587)	R7893-L		
#188	(#6588)	R7893-H		
#189	(#6589)	R7894-L		
#190	(#6590)	R7894-H		
#191	(#6591)	R7895-L		
#192	(#6592)	R7895-H		
#193	(#6593)	R7896-L		
#194	(#6594)	R7896-H		
#195	(#6595)	R7897-L		
#196	(#6596)	R7897-H		

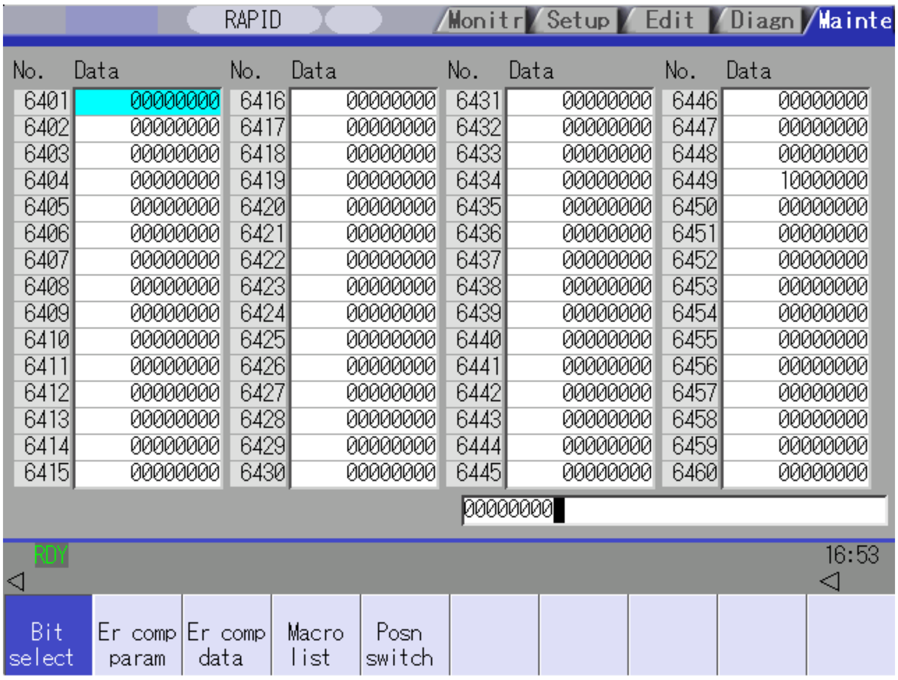
Contents of bit selection parameters #6449 to #6496

	Symbol name	7	6	5	4	3	2	1	0
0	#6449 R7824 L	Control unit thermal alarm on	Setting and display unit thermal mgmt on ■	-		Counter C retention	Integrated timer ST retention	PLC counter program on	PLC timer program on
1	#6450 R7824 H	-	External alarm message display ■	Alarm/ operator change ■	Full screen display of message ■	-	Operator message on	1: R method 0: F method	Alarm message on
2	#6451 R7825 L	-	-	Serial GPP communi-cation on			Onboard editing not possible ■	Onboard simple operation mode on	Onboard on ■
3	#6452 R7825 H	-	Branch destination label check valid		Serial handy terminal comm. on	-	-	Extended PLC instruction mode valid	-
4	#6453 R7826 L	Integrated timer ST Variable/fixed Number of points setting					Message language change code		
5	#6454 R7826 H	Counter C Variable/fixed Number of points setting				Timer T Variable/fixed Number of points setting			
6	#6455 R7827 L	-	-	-	-	-	-	-	-
7	#6456 R7827 H	-	-	-	-	-	-	-	-
8	#6457 R7828 L	High-speed input specification 1							
9	#6458 R7828 H								
A	#6459 R7829 L	High-speed input specification 2							
B	#6460 R7829 H								
C	#6461 R7830 L	High-speed output specification 1							
D	#6462 R7830 H								
E	#6463 R7831 L	High-speed output specification 2							
F	#6464 R7831 H								

	Symbol name	7	6	5	4	3	2	1	0
0	#6465 R7832 L	High-speed input specification 3							
1	#6466 R7832 H								
2	#6467 R7833 L	High-speed input specification 4							
3	#6468 R7833 H								
4	#6469 R7834 L	-	-	-	-	-	-	-	-
5	#6470 R7834 H	-	-	-	-	-	-	-	-
6	#6471 R7835 L	-	-	-	-	-	-	-	-
7	#6472 R7835 H	-	-	-	-	-	-	-	-
8	#6473 R7836 L	High-speed output specification 3							
9	#6474 R7836 H								
A	#6475 R7837 L	High-speed output specification 4							
B	#6476 R7837 H								
C	#6477 R7838 L	-	-	-	-	-	-	-	-
D	#6478 R7838 H	-	-	-	-	-	-	-	-
E	#6479 R7839 L	-	-	-	-	-	-	-	-
F	#6480 R7839 H	-	-	-	-	-	-	-	-

(Note 1) Be sure to set the bits indicated - and blanks to 0.
(Note 2) Parameters #6481 to #6496 are reserved for debugging by MITSUBISHI.
(Note3) Functions marked with ■ may not be available for some machine types.

Bit selection screen



4.3 Other Parameters

4.3.1 PLC Startup Condition Switchover

Parameter "#11004 PLCautorun enable" allows PLC to startup at NC startup even if no setting display unit is used.
For safety, use this function only for the machine with no NC screen displayed by HMI.

(1) Basic common parameter

# No.	Item		Details	Setting range	Standard value
11004 (PR)	PLCautorun enable	PLC automatic startup valid	Switch starting condition of the PLC. 0: Start PLC after NC screen startup 1: Start PLC at NC startup	0, 1	0

(2) Precautions

Parameter "#11004 PLCautorun enable" is the parameter prepared on the assumption that the setting and display unit is not used.
For the machine with NC screen displayed, to ensure your safety, always set "#11004 PLCautorun enable" to "0" and start PLC after NC screen startup.
When PLC automatic startup is validated without confirming the pre-operation status on the NC screen, unexpected incident may occur.

5. Explanation of Devices

5.1 Devices and Device No.

The devices are address symbols to identify signals handled in PLC. The device Nos. are serial Nos. assigned to the devices. The device Nos. of devices X, Y, SB, B, SW, W and H are represented in hexadecimal notation. The device numbers of other devices are represented in decimal notation.

5.2 List of Devices

Device	Device Range		Units	Details	Remarks
X	X0 to X1FFF	8192 points	1-bit	Input signals to the PLC. Machine input, etc.	
Y	Y0 to Y1FFF	8192 points	1-bit	Output signals from the PLC. Machine output, etc.	
M	M0 to M10239	10240 points	1-bit	Temporary memory	
L	L0 to L511	512 points	1-bit	Latch relay (Backup memory)	
F	F0 to F1023	1024 points	1-bit	Temporary memory. Alarm message interface	
SB	SB to SB1FF	512 points	1-bit	Special relay for link	
B	B0 to B1FFF	8192 points	1-bit	Link relay	
SM	SM0 to SM1023	1024 points	1-bit	Special relay	
V	V0 to V255	256 points	1-bit	Edge relay	
SW	SW0 to SW1FF	512 points	16-bit	Special register for link	
SD	SD0 to SD1023	1024 points	16-bit	Special register	
T	T0 to T703	704 points	1-bit/16-bit	Timer (Fixed/variable boundary is set with parameters)	*1
ST	ST0 to ST63	64 points	1-bit/16-bit	Incremented timer (100ms unit)	
C	C0 to C255	256 points	1-bit/16-bit	Counter (Fixed/variable boundary is set with parameters)	
D	D0 to D2047	2048 points	16-bit/32-bit	Data register. Register for calculation	
R	R0 to R13311	13312 points	16-bit/32-bit	File register. CNC word I/F	
W	W0 to W1FFF	8192 points	16-bit/32-bit	Link register	
Z	Z0 to Z1	2 points	16-bit	Address index	
N	N0 to N7	8 points		Master controller nesting level	
P	P0 to P2047 P4000 to P4005	2048 points		Conditional jump, subroutine call label	*2
K	K-32768 to K32767			Decimal constant for 16-bit instruction	
	K-2147483648 to K2147483647			Decimal constant for 32-bit instruction	
H	H0 to HFFFF			Hexadecimal constant for 16-bit instruction	
	H0 to HFFFFFFFF			Hexadecimal constant for 32-bit instruction	

*1 : The 10ms timer and 100ms timer are differentiated with instructions. (Refer to 5.3.4 Timer T)
*2 : The P device has two types of pointers, local and common. The number of points given above is the total number of points.

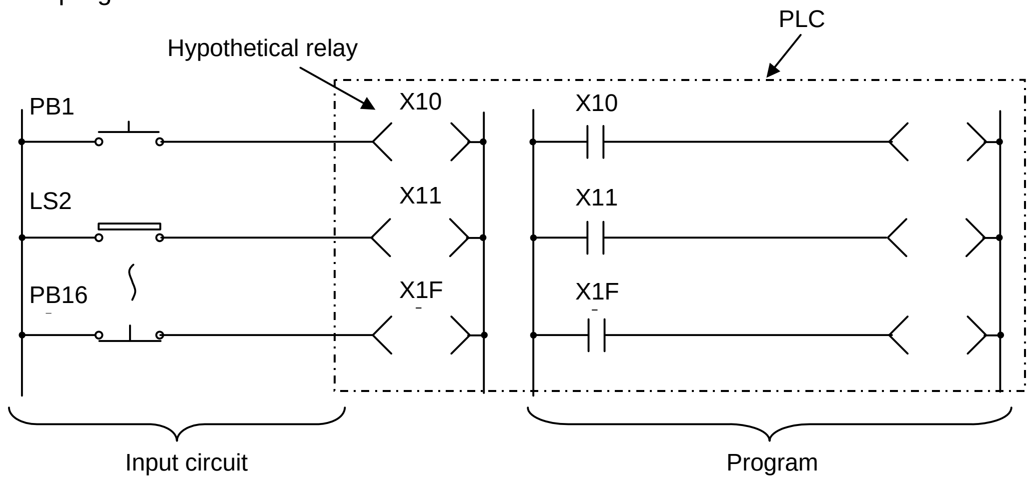
5.3 Detailed Explanation of Devices

5.3.1 Input/Output X, Y

Input/output X and Y are windows for executing communication with the PLC and external device or controller.

Input X

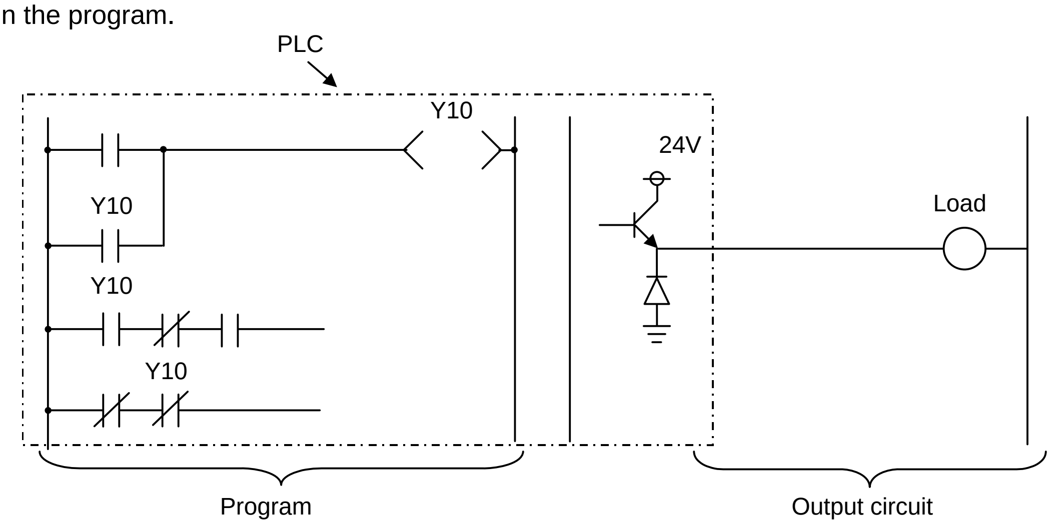
(a) This issued commands or data from an external device such as a push-button, changeover switch, limit switch or digital switch to the PLC.
(b) Assuming that there is a hypothetical relay Xn built-in the PLC per input point, the program uses the "A" contact and "B" contact of that Xn.
(c) There is no limit to the number of "A" contacts and "B" contacts of the input Xn that can be used in the program.



(d) The input No. is expressed with a hexadecimal.

Output Y

(a) This outputs the results of the program control to the solenoid, magnetic switch, signal lamp or digital indicator, etc.
(b) The output (Y) can be retrieved with the equivalent of one "A" contact.
(c) There is no limit to the number of "A" contacts and "B" contacts of the output Yn that can be used in the program.



(d) The output No. is expressed with a hexadecimal.

5.3.2 Internal Relays M and F, Latch Relay L

The internal relay and latch relay are auxiliary relays in the PLC that cannot directly output to an external source.

Internal relay M

(a) The relay is cleared when the power is turned OFF.
(b) There is no limit to the number of "A" contacts and "B" contacts of the internal relays that can be used in the program.
(c) The internal relay No. is expressed with a decimal.

Internal relay F

Internal relay F is an interface for the alarm message display.
Use the bit selection parameter to determine whether to use this relay for the alarm message interface. The target will be all F0 to F1023. This internal relay can be used in the same manner as the internal relay M when not used as the alarm message interface.

Latch relay L

(a) The original state is held even when the power is turned OFF.
(b) There is no limit to the number of "A" contacts and "B" contacts of the latch relay that can be used in the program.
(c) The latch No. is expressed with a decimal.

5.3.3 Special Relay for Link (SB), Special Register for Link (SW)

Special relay for link (SB)

(a) This interacts between various kinds of network cards and PLC programs.

(b) ON/OFF control is applied due to various factors occurred at the time of data link. By monitoring the special relay for link, abnormal state of data link can be detected.

Special register for link (SW)

(a) This interacts between various kinds of network cards and PLC programs.

(b) Information at the time of data link is stored. By monitoring the special register for link, abnormal area and the cause can be examined.

5.3.4 Link Relay B, Link Register W

(1) Link relay B is the bit type device that performs data link with various link functions. Unused area can be used as the primary memory, etc.

(2) Link register W is the word type device that performs data link with various link functions. Unused area can be used as the primary memory, etc.

Link relay B, link register W

(a) This relay is cleared when the power is turned OFF.

(b) There is no limit in the number that can be used in the program.

(c) The relay and register No. are expressed with a hexadecimal.

5.3.5 Special Relay SM, Special Register SD

(1) Special relay is the relay whose application is fixed. (i.e. Carry flag of operation result, display request signal to the setting display device, etc.) Do not use the currently unused area from SM0 to SM1023 as the primary memory.

(2) Special register SD is the data register whose application is fixed. (i.e. 1-second counter) Do not use the currently unused area from SD0 to SD1023 as the primary memory.

Link relay B, link register W

(a) This relay is cleared when the power is turned OFF.

(b) There is no limit in the number that can be used in the program.

(c) The relay and register No. are expressed with a decimal.

(3) Some of the main relays and registers whose applications are fixed are listed below.
Refer to "Appendix 3 List of Special Relays and Special Registers" for details.

Device	Name	Details
SM0	PLC error	Turns ON at PLC error occurrence; resets when changed from STOP to RUN.
SM12	Carry flag	Used with various machine types
SM400	Always ON	Always ON
SM401	Always OFF	Always OFF
SM402	After RUN, turned ON by only 1 scan.	(For medium-speed ladder)
SM403	After RUN, turned OFF by only 1 scan.	
SM404	After RUN, turned ON by only 1 scan.	(For high-speed ladder)
SM405	After RUN, turned OFF by only 1 scan.	
SM410	0.1-second clock	• ON/OFF is repeated every specified amount of time divided by 2. • Operation is continued even during STOP • Starts from OFF when starting up
SM411	0.2-second clock	
SM412	1-second clock	
SM413	2-second clock	
SM414	2n-second clock	ON/OFF is repeated according to the second specified with SD414.

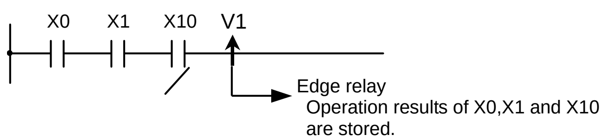
Device	Name	Details
SD0	PLC error No.	Error code when a PLC error occurs.
SD412	1-second clock	Number of counts in 1sec unit
SD414	2n-second clock set	Used for the 2n-second clock setting
SD420	Scan counter (Medium-speed ladder)	Number of counts per 1 scan • After RUN, +1 is added every 1 scan.
SD430	Scan counter (High-speed ladder)	

5.3.6 Edge relay V

Edge relay V

(a) This stores the operation result (ON/OFF information) from the head of ladder block.

(b) This can be used only at contacts. This cannot be used as a coil.



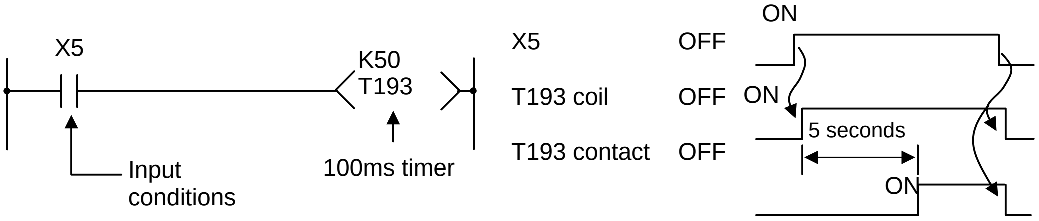
(c) The relay No. is expressed with a decimal.

5.3.7 Timer T

(4) The 100ms timer and 10ms timer are available for this count-up type timer.
The 100ms timer and 10ms timer are differentiated by the instructions used. Refer to the following explanation on basic instructions for details.

100ms Timer T

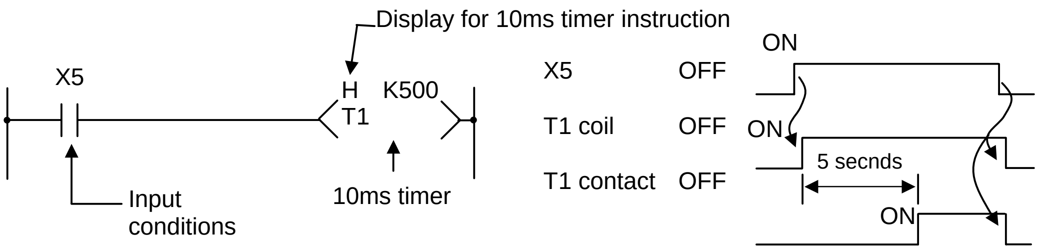
(a) When the input conditions are set, the count starts. When the set value is counted, that timer contact will turn ON.
(b) If the input conditions are turned OFF, the 100ms timer count value will be set to 0, and the contact will turn OFF.



(c) The value is set with a decimal, and can be designated from 1 to 32767 (0.1 to 3276.7 s).
The data register D or file register R data can be used as the setting value.

10ms Timer T

(a) When the input conditions are set, the count starts. When the set value is counted, that timer contact will turn ON.
(b) If the input conditions are turned OFF, the 10ms timer count value will be set to 0, and the contact will turn OFF.



(c) The value is set with a decimal, and can be designated from 1 to 32767 (0.01 to 327.67 s).
The data register D or file register R data can be used as the setting value.

(2) With the device T, the contact/coil is handled as bit device, and the current value is handled as word device. In the function instructions described after, the word device T indicates the current value even if there is no description about it.

(3) Setting the timer setting value from the setting and display unit
The timer T setting value can be set with the following two methods.

- Method to validate the setting value (Kn) programmed with the sequence program (Fixed timer)
- Method to validate the setting value set from the setting and display unit (Variable timer)
  (Note that even when this method is used, the setting value (Kn) must be programmed in the sequence program. In this case, the Kn value will be ignored during the operation. When a data register D is used for the setting value, the contents of the data register D will be the setting value regardless of the parameter.)

(a) Methods for setting the number of fixed timer and variable timer points
The ratio of the fixed timer and variable timer in all of the timer T points can be set with the bit selection parameter.
The boundary of the two setting methods is set using 100 points of the timer as one unit. This setting is validated when the PLC is restarted.

<table>
<tr><th colspan="2">Variable timer</th><th colspan="4">Bit selection (#6454)</th><th rowspan="2">Remarks</th></tr>
<tr><th>Number of points</th><th>Range</th><th>Bit 3</th><th>Bit 2</th><th>Bit 1</th><th>Bit 0</th></tr>
<tr><td>0</td><td></td><td>0</td><td>0</td><td>0</td><td>0</td><td>Use all points as fixed timer</td></tr>
<tr><td>100</td><td>(0 to 99)</td><td>0</td><td>0</td><td>0</td><td>1</td><td rowspan="6">Use range other than that shown on left as fixed timer</td></tr>
<tr><td>200</td><td>(0 to 199)</td><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>300</td><td>(0 to 299)</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>400</td><td>(0 to 399)</td><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>500</td><td>(0 to 499)</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>600</td><td>(0 to 599)</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>All points</td><td>(0 to 703)</td><td>0</td><td>1</td><td>1</td><td>1</td><td>Use all points as variable timer</td></tr>
</table>

(b) Variable timer validity setting bit on program side
A bit selection parameter is provided as a switch for the variable timer to invalidate all of the setting values set from the setting and display unit and validate the setting values in the sequence program. This setting is valid when the PLC is restarted. (This bit is valid also for the integrated timer.)

```
                     7  6  5  4  3  2  1  0
# (6449) Data      ( 0  0  0  0  0  0  1  1 )
     ↑                                  |  |
Use No. 6449                            |  └── 0: PLC timer screen setting valid
                                        |      1: PLC timer program valid
                                        └───── 0: PLC counter screen setting valid
                                               1: PLC counter program valid
```

(c) Methods for setting the setting value from the setting and display unit
The timer and counter setting value can be set from the parameter setting screen. Steps 1 to 4 also apply for the counter.

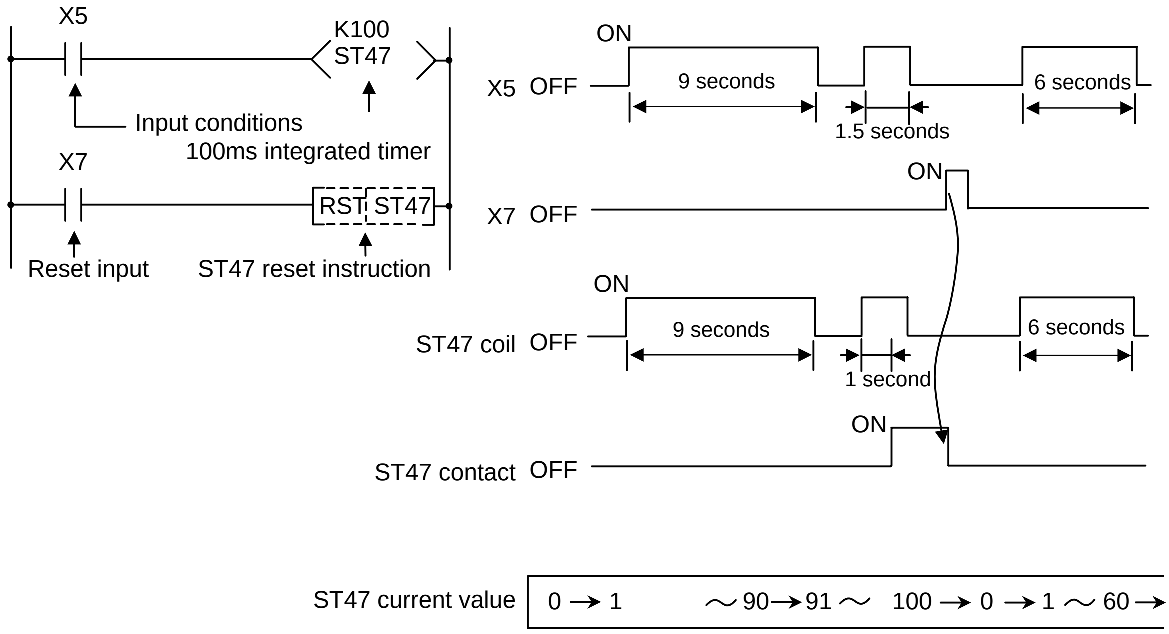
1) The set value is validated when the PLC is restarted.
2) On the setting screen, only the number of variable timer points set with (a) are valid. The (a) setting is immediately reflected on the display of the settable range.
3) The setting screen values are not affected even if the (b) "variable timer validity setting bit on program side" setting is changed.
4) It is possible to judge whether the setting value is valid within the current storage ladder (whether that timer is used with the setting value (Kn) in the ladder).
5) The timer type (10ms, 100ms) can be judged on the setting screen.

5.3.8 Integrated Timer ST

(1) The 100ms integrated timer is available for this count-up type timer.

100ms Integrated timer ST

(a) When the input conditions are set, the count starts. When the set value is counted, that timer contact will turn ON.
(b) Even the input conditions are turned OFF, the 100ms integrated timer current value (count value) will be held, and the contact state will not change.
(c) The 100ms integrated timer count value will be set to 0 and the contact will turn OFF when the RST instruction is executed.



(d) The value is set with a decimal, and can be designated from 1 to 32767 (0.1 to 3267.7 s). The data register D or file register R data can be used as the setting value.
(e) When the bit selection parameter is set, the 100ms integrated timer current value (count value) will be held even when the power is turned OFF.

(2) Handling the device ST types
With the device ST, the contact/coil is handled as bit device, and the current value is handled as word device. In the function instructions described after, the word device T indicates the current value even if there is no description about it.

(3) Setting the timer setting value from the setting and display unit
The ratio of the variable and fixed can be set with the bit selection parameter in the same manner as timer T.

Variable integrated timer		Bit selection (#6453)			Remarks
Number of points	Range	Bit 7	Bit 6	Bit 5	
0		0	0	0	Use all points as fixed integrated timer
20	(0 to 19)	0	0	1	Use range other than that shown on left as fixed integrated counter
40	(0 to 39)	0	1	0	Use range other than that shown on left as fixed integrated counter
All points	(0 to 63)	0	1	1	Use all points as variable integrated timer

In the same manner as timer T, a bit selection parameter is provided as a switch for the variable integrated timer to invalidate all of the setting values set from the setting and display unit and validate the setting values in the sequence program. (This bit is used for both the timer T and integrated timer ST.)

5.3.9 Counter C

(1) The counter counts up and detects the rising edge of the input conditions. Thus, the count will not take place when the input conditions are ON.

Counter C

(a) The value is set with a decimal, and can be designated from 1 to 32767. The data register D or file register R data can be used as the setting value.
(b) The counter count value will not be cleared even if the input conditions turn OFF. The counter count value must be cleared with the RST instruction.
(c) When the bit selection parameter is set, the counter current value (count value) will be held even when the power is turned OFF. Note that some cannot be held depending on the version of CNC.

(2) With the device C, the contact/coil is handled as bit device, and the current value (counter value) is handled as word device. In the function instructions described after, the word device C indicates the current value (counter value) even if there is no description about it.

(3) The counter C setting value can be set with the following two methods.
(a) Method to validate the setting value (Kn) programmed with the sequence program (Fixed counter)
(b) Method to validate the setting value set from the setting and display unit (Variable counter)
(Note that even when this method is used, the setting value (Kn) must be programmed in the sequence program. In this case, the Kn value will be ignored during the operation. When a data register D is used for the setting value, the contents of the data register D will be the setting value regardless of the parameter.)

The ratio of the fixed counter and variable counter in all of the counter C points can be set with the bit selection parameter.

Variable counter		Bit selection (#6454)				Remarks
Number of points	Range	Bit 7	Bit 6	Bit 5	Bit 4	
0		0	0	0	0	Use all points as fixed counter
40	(0 to 39)	0	0	0	1	Use range other than that shown on left as fixed counter
80	(0 to 79)	0	0	1	0	
120	(0 to 119)	0	0	1	1	
160	(0 to 159)	0	1	0	0	
200	(0 to 199)	0	1	0	1	
240	(0 to 239)	0	1	1	0	
All points	(0 to 255)	0	1	1	1	Use all points as variable counter

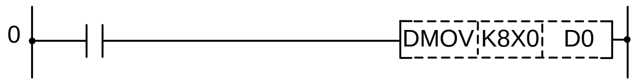
The bit selection parameter is set using 40 counter points as one unit.
A bit selection parameter is provided as a switch for the variable counter to invalidate all of the setting values set from the setting and display unit and validate the setting values in the sequence program.
(Refer to the explanation on the timer.)

5.3.10 Data Register D

(1) The data register is the memory that stores the data in the PLC.
(2) The data register has a 1-point 16-bit configuration, and can be read and written in 16-bit units.
To handle 32-bit data, two points must be used. The data register No. designated with the 32-bit instruction will be the low-order 16-bit, and the designated data register No. +1 will be the high-order 16-bit.

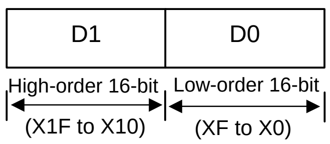
Circuit example

(Example) Use of the DMOV instruction is shown below.



The X0 to 1F data is stored in D0, 1.

Data storage



(3) The data that is stored once in the sequence program is held until other data is stored.
(4) The data stored in the data register is cleared when the power is turned OFF.
(5) Values that can be stored:

Decimal	-32768 to 32767	For 16-bit instruction (Using Dn)
Hexadecimal	0 to FFFF	
Decimal	-2147483648 to 2147483647	For 32-bit instruction (Using Dn+1, Dn)
Hexadecimal	0 to FFFFFFFF	

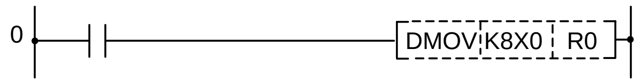
(6) Data registers D0 to D2047 are all user release data registers.

5.3.11 File Register R

(1) As with the data registers, the file registers are memories used to store data. However, there are some that have fixed applications, and those that are released.
(2) The file register has a 1-point 16-bit configuration, and can be read and written in 16-bit units.
To handle 32-bit data, two points must be used. The file register No. designated with the 32-bit instruction will be the low-order 16-bit, and the designated file register No. +1 will be the high-order 16-bit.

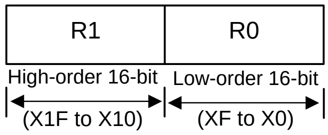
Circuit example

(Example) Use of the DMOV instruction is shown below.



The X0 to 1F data is stored in R0, 1.

Data storage

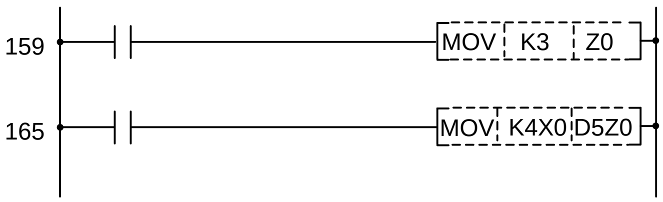


(3) The data that is stored once in the sequence program is held until other data is stored.
(4) With the file registers, the following registers are the user release.
R8300 to R9799, R9800 to R9899
The following registers of the registers above are not cleared when the power is turned OFF.
R8300 to R9799
The other file registers have fixed applications such as interface of the PLC and CNC, parameter interface, etc., so use according to the application.
(5) Values that can be stored:

Decimal	-32768 to 32767	For 16-bit instruction (Using Rn)
Hexadecimal	0 to FFFF	
Decimal	-2147483648 to 2147483647	For 32-bit instruction (Using Rn+1, Rn)
Hexadecimal	0 to FFFFFFFF	

5.3.12 Index register Z

(1) The index register is used as ornaments for the device (T, ST, C, D, R, W, SW, SD).



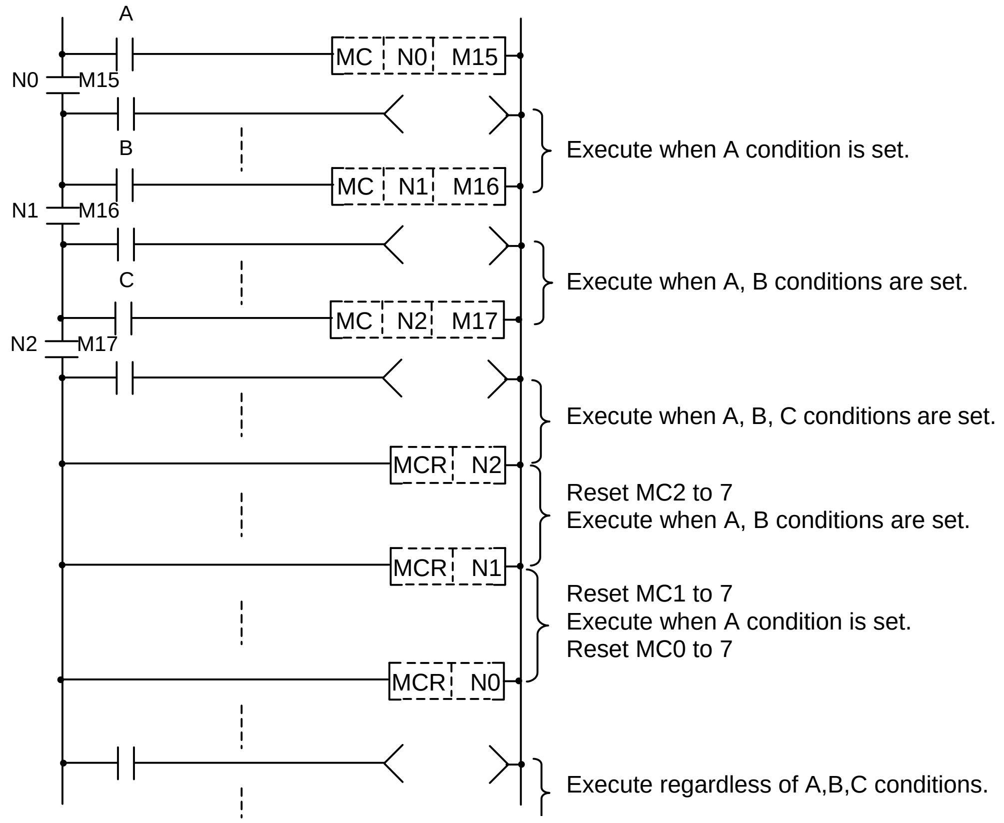
D5Z0 indicates D(5+Z0)=D8

(2) The index register has a 1-point 16-bit configuration, and can be read and written in 16-bit units.
(3) The data stored in the index register is cleared when the power is turned OFF.
(4) Values that can be stored:

Decimal	-32768 to 32767
Hexadecimal	0 to FFFF

5.3.13 Nesting N

(1) This indicates the master control nesting structure.
(2) The master control nesting N is used in order from smallest No.



(a) The conditions for each master control to turn ON are as follow.

$\boxed{\text{MC N0 M15}}$.......... ON when condition A is ON

$\boxed{\text{MC N0 M16}}$.......... ON when conditions A, B are ON

$\boxed{\text{MC N0 M17}}$.......... ON when conditions A, B, C are ON

(b) The timer and counter when the master control is OFF is as follows.

· 100ms timer, 10ms timer	: The count value is set to 0.
· 100ms integrated timer	: The current count value is retained.
· Counter	: The current counter value is retained.
· OUT instruction	: All turn OFF.

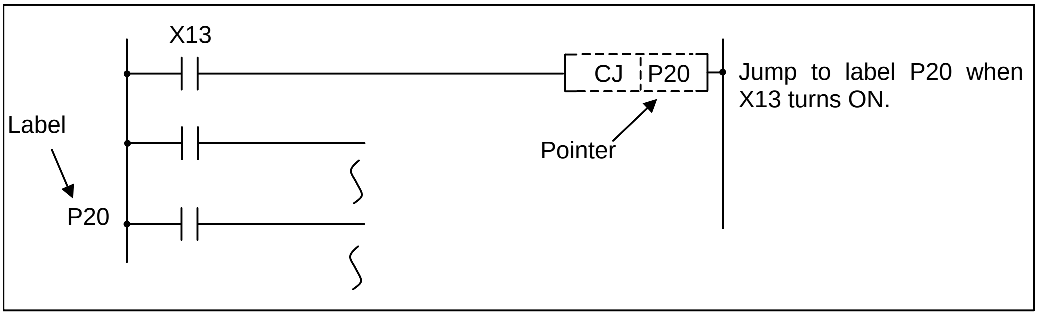
5.3.14 Pointer P

(1) What is a pointer?

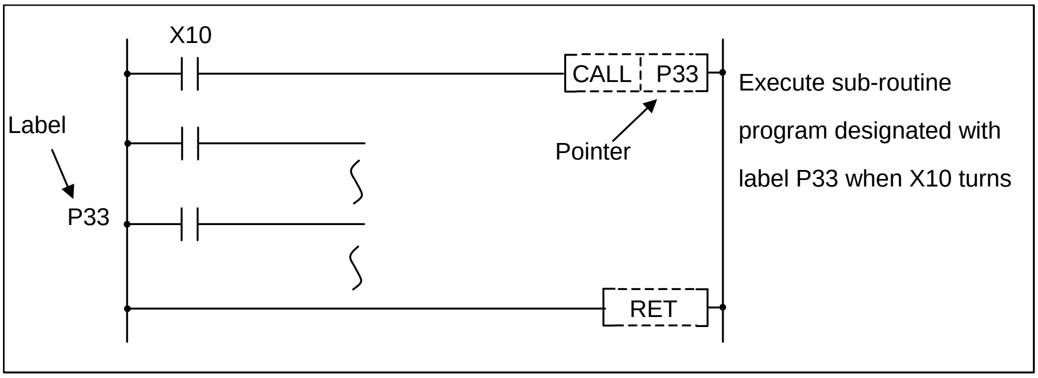
A pointer is a device used with branch instructions. A total of 2048 points is used in all executed programs.
The reserved pointers use the 4000 addresses separately.

(2) Pointer applications

(a) Jump instruction (CJ, JMP) jump destination designation and label (Designation of jump destination head)



(b) Subroutine call instruction (CALL) call destination and label (Designation of subroutine program head)



(3) Types of pointers

The details of the pointers differ according to the program method.

(a) Independent program method
The following two types of pointers are used.
- General pointer : Pointer which can jump or call with a jump instruction or subroutine call instruction
- Reserved pointer : Pointer with fixed application, such as a start label

(b) Multi-program method
The following three types of pointers are used.
- Local pointer : Pointer used independently in each program
- Common pointer : Pointer which can be called with subroutine call instruction from all programs being executed
- Reserved pointer : Pointer with fixed application, such as an END label

5.3.14.1 General Pointers

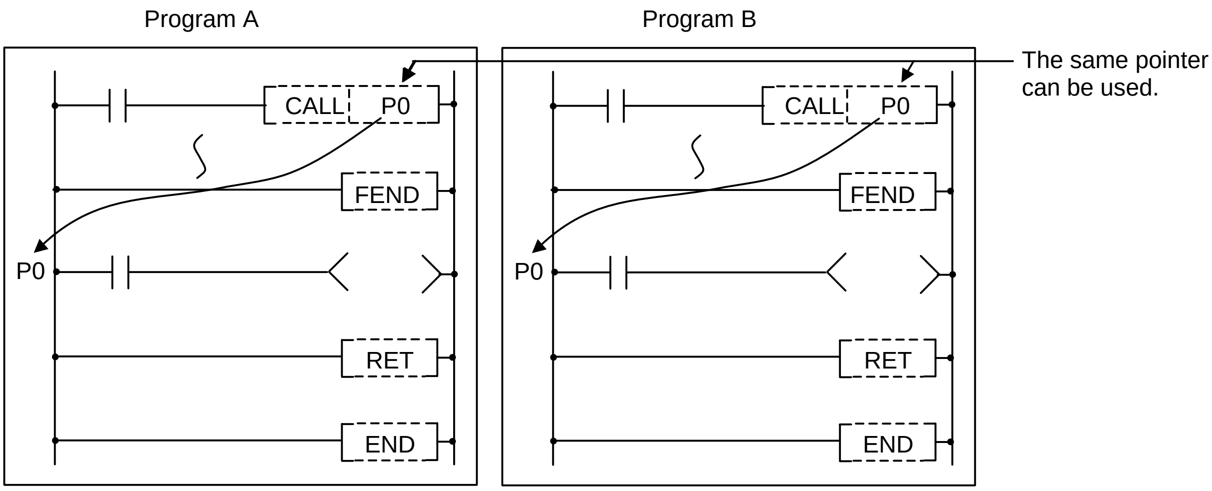
General pointers are pointer which can be used only with the independent program method, which lays importance on compatibility with conventional models.
The general pointer can be used with the jump instructions and subroutine call instructions. The same pointer No. cannot be used.

5.3.14.2 Local Pointers

Local pointers are pointers that can be used only with the multi-program method.

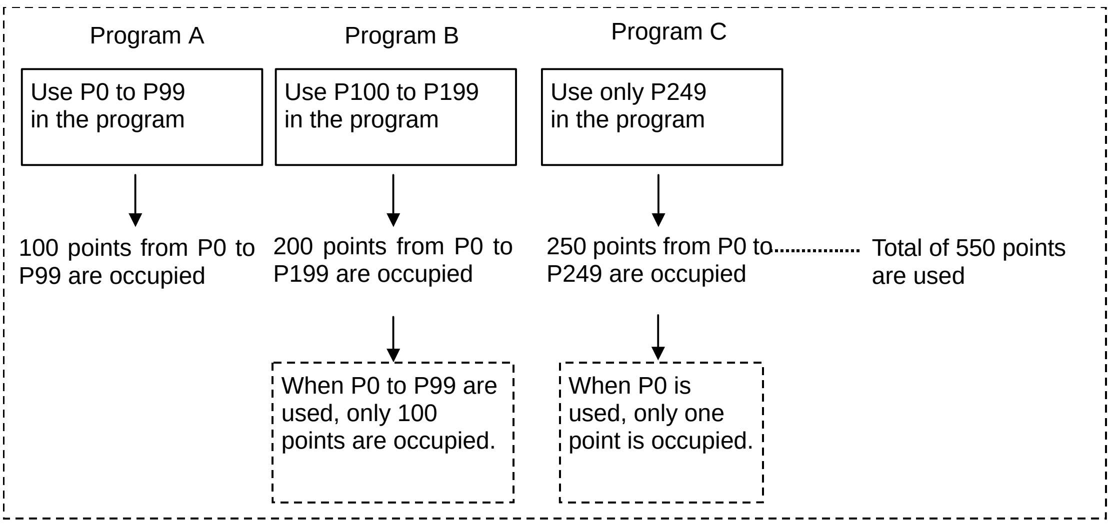
(1) What is a local pointer?

(a) A local pointer is a pointer that can be used independently with each program stored in the CNC controller. The local pointer can be used with the jump instructions and subroutine call instructions.
(b) The same pointer No. can be used in each program.
The pointers from P0 to the common pointer usage range setting value (explained later) can be used.



(2) Concept of number of local pointer points

The local pointers split and use the local pointer area (arbitrarily settable with user settings) in all programs. Up to the maximum No. of local pointers in use can be used in each program. When using the local pointers in multiple programs, start use from P0.
An error will occur if the total of local pointers used in each program exceeds the setting number.



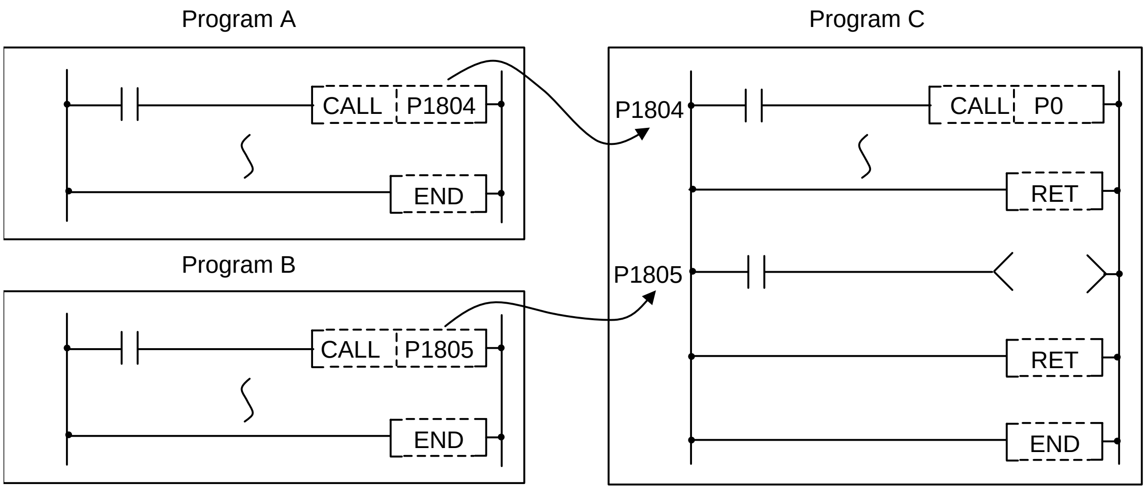
5.3.14.3 Common Pointers

Common pointers are pointers that can be used only with the multi-program method.

(1) What is a common pointer?

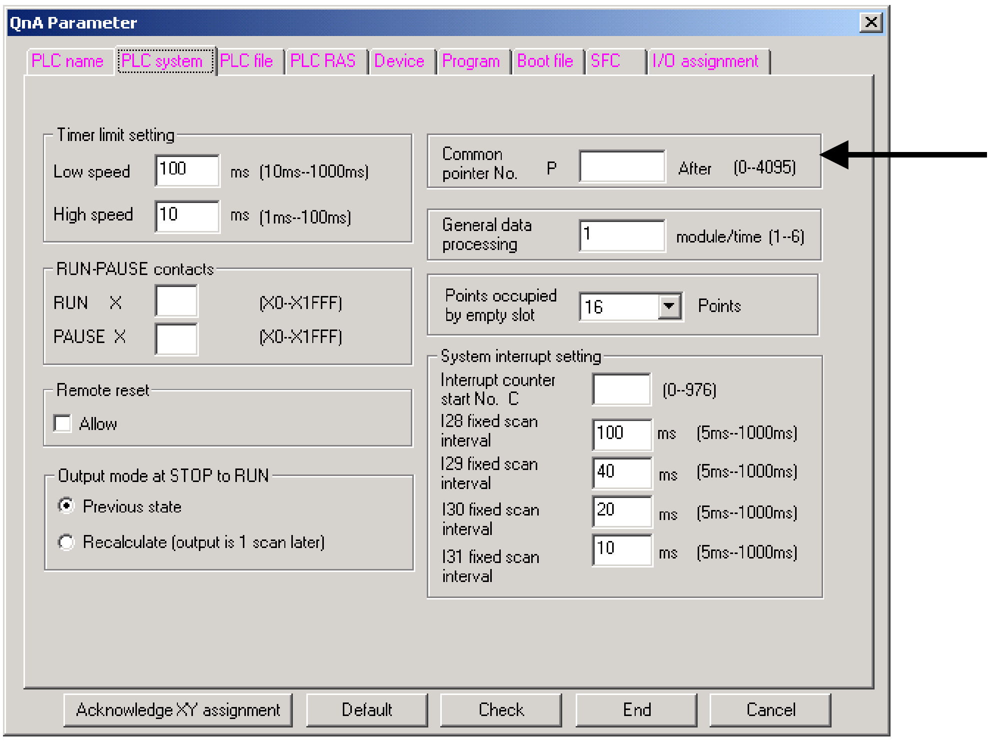
(a) A common pointer is a pointer that calls the subroutine program from all programs executed with the CNC controller.
The common pointer can be used only with the subroutine call instruction, and cannot be used with the jump instruction.

(b) The same pointer No. cannot be used as a label.



(2) Common pointer usage range

The common pointer usage range can be set with the GX Developer parameter settings. The range following the set No. is the common pointers. The range that can be set as the head No. of the common pointer is P0 to P2047. The default value is set to 1800, and P1800 to P2047 can be used as common pointers.



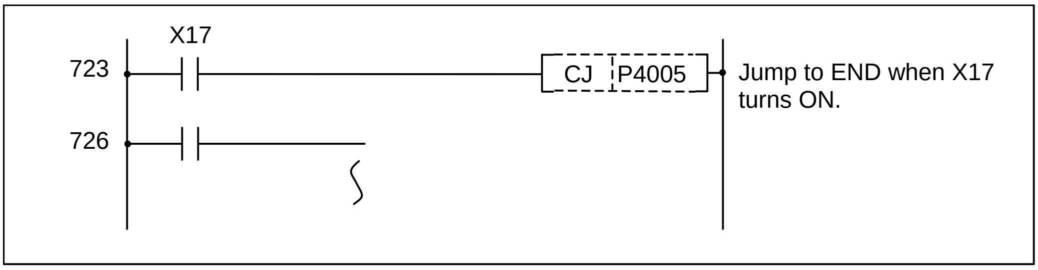
5.3.14.4 Reserved Pointers

Reserved pointers are pointers with fixed applications.

(1) Independent program method

P4001 (high-speed) : Start label for PLC high-speed processing program.
P4002 (medium-speed) : Start label for PLC main (ladder) processing program.
P4005 (END) : Label indicating END.

P4005 (END) can be used as a device for the CJ instruction, etc., but cannot be used as a label.
In addition, it cannot be used for a CALL instruction device.



[CAUTION]

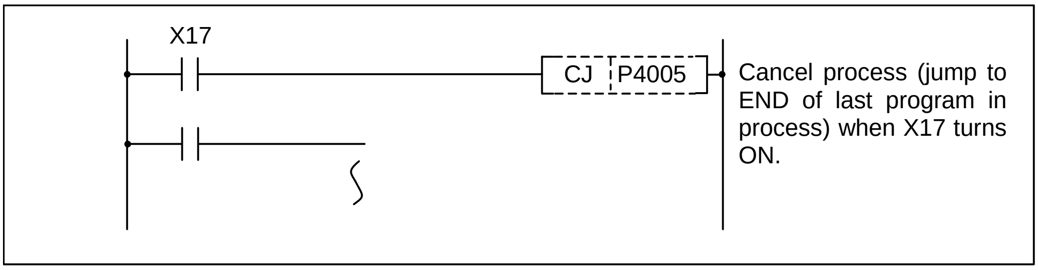
1. Do not omit the P4002 (medium-speed) label even when using only the PLC main processing program.
2. Do not use P4001 (high-speed) or P4002 (medium-speed) as a CJ instruction or CALL instruction device.
3. Do not program to jump to P** in the PLC high-speed processing program from the PLC main processing program.
4. P** used as a CJ instruction or CALL instruction device must be programmed so that it is in the same program file as the label instruction.

The PLC will not run properly if even one of Cautions 1 to 4 is not observed.

(2) Multi-program method

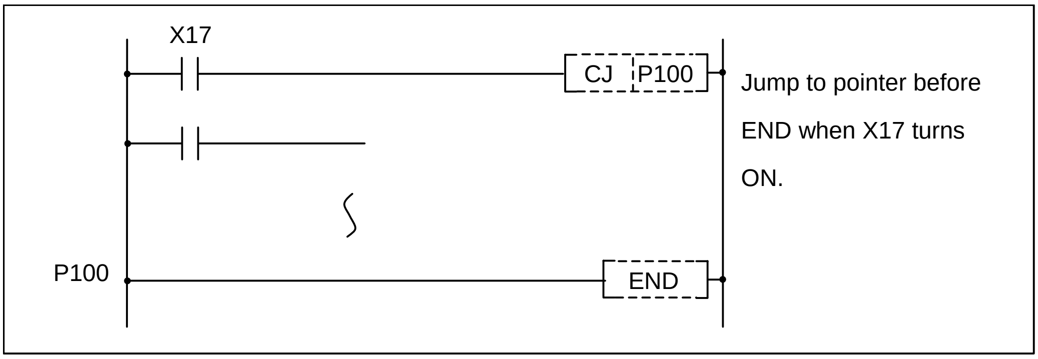
(a) Label indicating END (P4005)
P4005 is used as the CJ instruction jump destination, and cannot be used as a normal label. It also cannot be used as the CALL instruction call destination.
If CJ P4005 is executed when multiple PLC programs are registered with the multi-programming function, the process will jump to the end of all PLC programs (in other words, the scan process is canceled).

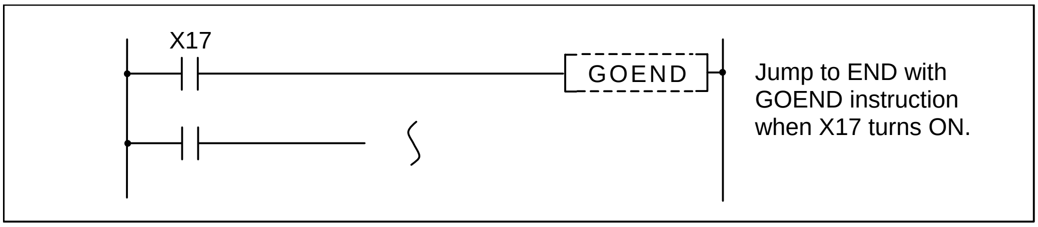
To jump to end of all processes



The following two methods of jumping to the end of each program are available.
• Sets the local pointer right before the END instruction and jumps to that position.
• Executes GOEND instruction. (Usable only with the extended instruction mode)

To jump to the end of the program





[Cautions for pointers in multi-program method]

1. Each process (initialization, high-speed, main) is executed from the head of the program executed at the start of each process.
   Thus, the reserved labels for starting, used with the conventional models, cannot be used.
2. The common pointers can be used only with subroutine call instructions, and cannot be used with jump instructions.
3. The labels using the CJ instruction, JMP instruction or CALL instruction as a device must be programmed so that one of the following exists.
   • If the pointer is a local pointer, it must exist in the same program file as the used instruction.
   • If the pointer is a common pointer, it must exist in one of the registered program files.

The PLC will not run properly if even one of Cautions 1 to 3 is not observed.

5.3.15 Decimal Constant K

(1) The decimal constant can be used in the following ways.
   (a) Timer counter setting value : Designate in the range of 1 to 32767.
   (b) Pointer No. : 0 to 159
   (c) Bit device digit designation : 1 to 8
   (d) Basic instruction, function instruction, exclusive instruction value setting
      • 16-bit instruction : -32768 to 32767
      • 32-bit instruction : -2147483648 to 2147483647

(2) The decimal constant is stored by binary value in the PLC.

5.3.16 Hexadecimal Constant H

(1) The hexadecimal constant is used to designate the basic instruction, function instruction and exclusive instruction values.
      • 16-bit instruction : 0 to FFFF
      • 32-bit instruction : 0 to FFFFFFFF

6. Explanation of Instructions

6.1 Compatible Instructions and Extended Instructions

The following two PLC instruction modes are available with this CNC. Characteristics and setting methods for these instructions are explained here.

- Compatible PLC instruction mode (Usable model: 700 Series, 70 Series type A/type B)
- Extended PLC instruction mode (Usable model: 700 Series, 70 Series type A)

(1) Outline and differences of each mode

"Compatible PLC instruction mode" is set when instructions must be compatible with those of the conventional machine type. In this mode, only the PLC instruction specification which is conventionally compatible with can be used. If the extended PLC instruction is used, an execution error occurs at the time of input or edit.

"Extended PLC instruction mode" is set when the extended instruction specification is used. "Extended PLC instruction mode" includes the specification of "Compatible PLC instruction mode". (Note that operations may differ for some instructions. Details are explained later.)
Specifications for each mode are given below.

	Conventional machine type	Compatible PLC instruction mode	Extended PLC instruction mode
Number of basic instructions	22 instructions	←	37 instructions
Number of function instructions	71 instructions	←	198 instructions
Usable device	15 devices	22 devices	←
Device designation range of instruction argument	–	←	Extended

(2) Setting method of PLC instruction mode

PLC instruction mode is set by bit selection parameter #6452.

(a) Bit selection parameter

# No.	Bit	Item	Details	Setting range	Standard value
6452	Bit 1	PLC instruction extension valid	The condition of the usable instruction for the built-in PLC can be switched. 0: Operated in the compatible PLC instruction mode 1: Operated in the extended PLC instruction mode	0, 1	0

(b) Notes

- This parameter will be valid when the power is turned OFF and ON again.
- If expanding the PLC instruction is disabled during the use of expansion PLC instruction, an error occurs at PLC RUN.
- Even if expanding the PLC instruction is enabled in 70 Series type B, the instruction is ignored and operated in the compatible PLC instruction mode.

(3) Notes

The following instructions have different operations even if they are the same instructions, depending on each instruction mode. For these instructions, it is highly recommended that the instructions should be replaced with ones that are usable in both modes.

Instruction	Compatible PLC instruction mode	Extended PLC instruction mode	Instruction replacement method
LD<= AND<= OR<=	Operated as bit test instruction. (Alternative instruction for LDBIT,ANDBIT,ORBIT)	Operated as comparison operation instruction. (LD<=,AND<=,OR<= instruction)	"Replaceable instructions" given in the instruction specification details for "LDBIT" indicated in "6.2.12 Special Instructions for Old Machine Type Compatible"
LD<> AND<> OR<>	Operated as bit test instruction. (Alternative instruction for LDBII,ANDBII,ORBII)	Operated as comparison operation instruction. (LD<>,AND<>,OR<> instruction)	
ANDP	Alternative instruction for DEFR (pulse in respect to the operation result) instruction	Operated as leading edge pulse series connection instruction. (ANDP instruction)	Refer to "Appendix 1.3.1 Alternative Circuits Resulted from the Ban on DEFR Instruction"

6.2 Instruction Tables

6.2.1 How to Read Instruction Table

The instruction tables have been made according to the following format.

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
+ (BIN)	16-bit	+	[+ | S | D]	(D)+(S)→(D) (BIN)		■	3	3	86
		+P	[+P | S | D]			■	3	7	86
		+	[+ | S1 | S2 | D]	(S1)+(S2)→(D) (BIN)			4	4	88
		+P	[+P | S1 | S2 | D]			■	4	8	88

(1) (2) (3) (4) (5) (6) (7) (8) (9) (10)

(1) • • • Classifies instructions according to their application

(2) • • • Indicates the processing unit of instructions.

(3) • • • Indicates the instruction symbol used to enter the instruction in a program
Instruction code is built around the 16-bit instruction, with the following notations used to mark 32-bit instructions, instructions executed only at the leading edge of OFF to ON, real number instructions, and character string instructions.

- 32-bit instruction • • • The letter "D" is added to the first line of the instruction

  (Example)

  + → D+

  16-bit instruction → 32-bit instruction

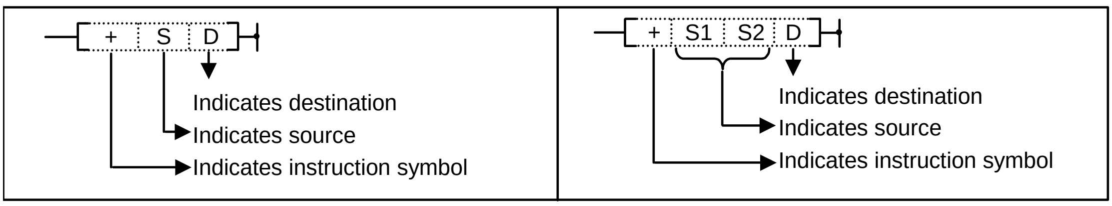
- Instructions executed only at the leading edge of OFF to ON • • • The letter "P" is appended to the end of the instruction

  (Example)

  + → +P

  Instructions executed when ON → Instructions executed only at the leading edge of OFF to ON

(4) • • • Shows symbol drawing on the ladder

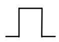


Destination……….Indicates where data will be sent following operation
Source………………Stores data prior to operation

(5) • • • Indicates the type of processing that is performed by individual instructions

(D) + (S) ⟶ (D) Indicates 16 bits	(D+1,D) + (S+1,S) ⟶ (D+1,D) 16 bits 16 bits Indicates 32 bits D+1: Upper 16 bits D: Lower 16 bits

(6) • • • The details of conditions for the execution of individual instructions are as follows.

Symbol	Execution Condition
No symbol recorded	Instruction executed under normal circumstances, with no regard to the ON/OFF status of conditions prior to the instruction. If the precondition is OFF, the instruction will conduct OFF processing.
	Executed during ON; instruction is executed only while the precondition is ON. If the precondition is OFF, the instruction is not executed and no processing is conducted.
	Executed once at ON; instruction executed only at leading edge when precondition goes from OFF to ON. Following execution, instruction will not be executed and no processing conducted even if condition remains ON.
	Executed once at OFF; instruction executed only at trailing edge when precondition goes from ON to OFF. Following execution, instruction will not be executed and no processing conducted even if condition remains OFF.

(7) • • • "■" mark indicates that the instruction is an extended instruction.
An extended instruction operates in "Extended PLC instruction mode". When an extended instruction is used in "Compatible PLC instruction mode", an error occurs at input, edit or execution.

(8) • • • Indicates the number of steps when storing each instruction.
This is the number of steps that is consumed when each instruction is stored in F-ROM. Refer to "2.6 Storing PLC Processing Program and Execution Mode " for details.

(9) • • • Indicates the number of steps when executing each instruction.
This is the number of steps that is consumed in the PLC processor execution area when each instruction is executed. The number of steps may be different from that of when stored in F-ROM. Refer to "2.6 Storing PLC Processing Program and Execution Mode " for details.

(10) • • • Indicates the page numbers where the individual instructions are discussed.

6.2.2 Basic Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Basic instruction	Bit	LD		Start of logic operation (A contact operation start)			1/2 *1	1/2 *2	58
		LDI		Start of logic negation operation (B contact operation start)			1/2 *1	1/2 *2	58
		AND		Logical AND (A contact serial connection)			1/2 *1	1/2 *2	58
		ANI		Logical AND negation (B contact serial connection)			1/2 *1	1/2 *2	58
		OR		Logical OR (A contact parallel connection)			1/2 *1	1/2 *2	58
		ORI		Logical OR negation (B contact parallel connection)			1/2 *1	1/2 *2	58
		ANB		AND between logical blocks (Serial connection between blocks)			1	1	60
		ORB		OR between logical blocks (Parallel connection between blocks)			1	1	60
		LDP		Starts leading edge pulse operation		■	1/2 *1	4	62
		LDF		Starts trailing edge pulse operation		■	1/2 *1	4	62
		ANDP		Leading edge pulse series connection		■	1/2 *1	4	62
		ANDF		Trailing edge pulse series connection		■	1/2 *1	4	62
		ORP		Leading edge pulse parallel connection		■	1/2 *1	4	62
		ORF		Trailing edge pulse parallel connection		■	1/2 *1	4	62
		INV		Inversion of operation result		■	1	3	64
		MEP		Conversion of operation result to leading edge pulse		■	1	3	65
		MEF		Conversion of operation result to trailing edge pulse		■	1	3	65
		EGP	Vn	Conversion of operation result to leading edge pulse (Stored at Vn)		■	1	3	66
		EGF	Vn	Conversion of operation result to trailing edge pulse (Stored at Vn)		■	1	3	66
		OUT		Device output			1/2 *1	1/2 *2	67
		OUT T/C		100ms timer/counter output			4	3	69,71
		OUT H		10ms timer output			4	3	69

(To be continued on the next page)

*1: Argument will be 2 steps at F device.
*2: Argument will be 1 step at bit device, 2 steps at word device.

Basic instructions (continued)

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Basic instruction	Bit	SET	[SET D]	Device set			1/2 *1	1/2 *2	73
		RST	[RST D]	Device reset			1/2 *1	1/2 *2	75
		RST T/C	[RST D]	Timer/counter reset			4	2	75
		MC	[MC n D]	Master control start			2	2	77
		MCR	[MCR n]	Master control release			1	1	77
		PLS	[PLS D]	Generate one cycle worth of pulses at rising edge of input signal			2	2	79
		PLF	[PLF D]	Generate one cycle worth of pulses at falling edge of input signal			2	2	79
		FF	[FF D]	Reversal of device output		■	2	5	81
		SFT	[SFT D]	Device 1-bit shift			2	1/2 *2	82
		SFTP	[SFTP D]			■	2	6	82
		MPS	MPS	Registration of operation result			1	1	84
		MRD	MRD	Read of operation results registered in MPS			1	1	84
		MPP	MPP	Reading and resetting of operation results registered in MPS			1	1	84
		NOP		Ignored (For program deletion or space)			1	1	86
		NOPLF	[NOPLF]	Ignored (To change pages during printouts)		■	1	1	86
		PAGE	[PAGE n]	Ignored (Subsequent programs will be controlled from step 0 of page n)		■	1	1	86

*1: 2 steps when the argument is F device.
*2: 1 step when the argument is bit device; 2 steps when the argument is word device.

6.2.3 Comparison Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps Storage	No. of steps Execution	See for Description
=	16-bit	LD=	[= S1 S2]	Continuity state when (S1) = (S2) Non-continuity state when (S1) ≠ (S2)			3	3	90
		AND=	[= S1 S2]				3	3	90
		OR=	[= S1 S2]				3	3	90
	32-bit	LDD=	[D= S1 S2]	Continuity state when (S1+1,S1)=(S2+1,S2) Non-continuity state when (S1+1,S1)≠(S2+1,S2)			3/4 *1	3/4 *1	92
		ANDD=	[D= S1 S2]				3/4 *1	3/4 *1	92
		ORD=	[D= S1 S2]				3/4 *1	3/4 *1	92
≠	16-bit	LD<>	[<> S1 S2]	Continuity state when (S1) ≠ (S2) Non-continuity state when (S1) = (S2)		■	3	3	90
		AND<>	[<> S1 S2]			■	3	3	90
		OR<>	[<> S1 S2]			■	3	3	90
	32-bit	LDD<>	[D<> S1 S2]	Continuity state when (S1+1,S1)≠(S2+1,S2) Non-continuity state when (S1+1,S1)=(S2+1,S2)		■	3/4 *1	3/4 *1	92
		ANDD<>	[D<> S1 S2]			■	3/4 *1	3/4 *1	92
		ORD<>	[D<> S1 S2]			■	3/4 *1	3/4 *1	92
>	16-bit	LD>	[> S1 S2]	Continuity state when (S1)>(S2) Non-continuity state when (S1)≦(S2)			3	3	90
		AND>	[> S1 S2]				3	3	90
		OR>	[> S1 S2]				3	3	90
	32-bit	LDD>	[D> S1 S2]	Continuity state when (S1+1,S1)>(S2+1,S2) Non-continuity state when (S1+1,S1)≦(S2+1,S2)			3/4 *1	3/4 *1	92
		ANDD>	[D> S1 S2]				3/4 *1	3/4 *1	92
		ORD>	[D> S1 S2]				3/4 *1	3/4 *1	92
≧	16-bit	LD>=	[>= S1 S2]	Continuity state when (S1) ≧ (S2) Non-continuity state when (S1) < (S2)		■	3	3	90
		AND>=	[>= S1 S2]			■	3	3	90
		OR>=	[>= S1 S2]			■	3	3	90
	32-bit	LDD>=	[D>= S1 S2]	Continuity state when (S1+1,S1)≧(S2+1,S2) Non-continuity state when (S1+1,S1)<(S2+1,S2)		■	3/4 *1	3/4 *1	92
		ANDD>=	[D>= S1 S2]			■	3/4 *1	3/4 *1	92
		ORD>=	[D>= S1 S2]			■	3/4 *1	3/4 *1	92

(To be continued on the next page)

*1: 1 step is added when either S1 or S2 is a constant number.

Comparison instructions (continued)

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
<	16-bit	LD<	[< S1 S2]	Continuity state when (S1) < (S2) Non-continuity state when (S1) ≧ (S2)			3	3	90
		AND<	[< S1 S2]				3	3	90
		OR<	[< S1 S2]				3	3	90
	32-bit	LDD<	[D< S1 S2]	Continuity state when (S1+1,S1)<(S2+1,S2) Non-continuity state when (S1+1,S1)≧(S2+1,S2)			3/4 *1	3/4 *1	92
		ANDD<	[D< S1 S2]				3/4 *1	3/4 *1	92
		ORD<	[D< S1 S2]				3/4 *1	3/4 *1	92
≦	16-bit	LD<=	[<= S1 S2]	Continuity state when (S1) ≦ (S2) Non-continuity state when (S1) > (S2)		■	3	3	90
		AND<=	[<= S1 S2]			■	3	3	90
		OR<=	[<= S1 S2]			■	3	3	90
	32-bit	LDD<=	[D<= S1 S2]	Continuity state when (S1+1,S1)≦(S2+1,S2) Non-continuity state when (S1+1,S1)>(S2+1,S2)		■	3/4 *1	3/4 *1	92
		ANDD<=	[D<= S1 S2]			■	3/4 *1	3/4 *1	92
		ORD<=	[D<= S1 S2]			■	3/4 *1	3/4 *1	92

*1: 1 step is added when either S1 or S2 is a constant number.

6.2.4 Arithmetic Operation Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
+ (BIN)	16-bit	+	[+ S D]	(D)+(S)→(D) (BIN)		■	3	3	96
		+P	[+P S D]			■	3	7	96
		+	[+ S1 S2 D]	(S1)+(S2)→(D) (BIN)			4	4	94
		+P	[+P S1 S2 D]			■	4	8	94
	32-bit	D+	[D+ S D]	(D+1,D)+(S+1,S)→(D+1,D) (BIN)		■	3/4 *1	3/4 *1	100
		D+P	[D+P S D]			■	3/4 *1	7/8 *1	100
		D+	[D+ S1 S2 D]	(S1+1,S1)+(S2+1,S2)→(D+1,D) (BIN)			4/5 *2	4/5 *2	98
		D+P	[D+P S1 S2 D]			■	4/5 *2	8/9 *2	98

(To be continued on the next page)

*1: 1 step is added when S is a constant number.
*2: 1 step is added when either S1 or S2 is a constant number.

Arithmetic operation instructions (continued)

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
- (BIN)	16-bit	-	[- S D]	(D)−(S)→(D) (BIN)		■	3	3	96
		-P	[-P S D]			■	3	7	96
		-	[- S1 S2 D]	(S1)−(S2)→(D) (BIN)			4	4	94
		-P	[-P S1 S2 D]			■	4	8	94
	32-bit	D-	[D- S D]	(D+1,D)−(S+1,S)→(D+1,D) (BIN)		■	3/4 *1	3/4 *1	100
		D-P	[D-P S D]			■	3/4 *1	7/8 *1	100
		D-	[D- S1 S2 D]	(S1+1,S1)−(S2+1,S2)→(D+1,D) (BIN)			4/5 *2	4/5 *2	98
		D-P	[D-P S1 S2 D]			■	4/5 *2	8/9 *2	98
* (BIN)	16-bit	*	[* S1 S2 D]	(S1)× (S2)→(D+1,D) (BIN)			4	4	102
		*P	[*P S1 S2 D]			■	4	8	102
	32-bit	D*	[D* S1 S2 D]	(S1+1,S1) × (S2+1,S2) → (D+3,D+2,D+1,D) (BIN)			4/5 *2	4/5 *2	104
		D*P	[D*P S1 S2 D]			■	4/5 *2	8/9 *2	104
/ (BIN)	16-bit	/	[/ S1 S2 D]	(S1) ÷ (S2) → Quotient(D),Remainder(D+1) (BIN)			4	4	102
		/P	[/P S1 S2 D]			■	4	8	102
	32-bit	D/	[D/ S1 S2 D]	(S1+1,S1) ÷ (S2+1,S2) → Quotient(D+1,D), Remainder(D+3,D+2) (BIN)			4/5 *2	4/5 *2	104
		D/P	[D/P S1 S2 D]			■	4/5 *2	8/9 *2	104
Four arithmetic operations (BCD)	16-bit	B+	[B+ S1 S2 D]	(S1)+(S2)→(D) (BCD)		■	4	5	106
		B+P	[B+P S1 S2 D]			■	4	9	106
		B-	[B- S1 S2 D]	(S1)−(S2)→(D) (BCD)		■	4	5	106
		B-P	[B-P S1 S2 D]			■	4	9	106
		B*	[B* S1 S2 D]	(S1) × (S2)→(D+1,D) (BCD)		■	4	5	108
		B*P	[B*P S1 S2 D]			■	4	9	108
		B/	[B/ S1 S2 D]	(S1)÷ (S2) → Quotient(D),Remainder(D+1) (BCD)		■	4	5	108
		B/P	[B/P S1 S2 D]			■	4	9	108

(To be continued on the next page)

*1: 1 step is added when S is a constant number.
*2: 1 step is added when either S1 or S2 is a constant number.

Arithmetic operation instructions (continued)

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
+1	16-bit	INC	[INC D]	(D)+1→(D)			2	2	110
		INCP	[INCP D]			■	2	6	110
	32-bit	DINC	[DINC D]	(D+1,D)+1→(D+1,D)			2	2	112
		DINCP	[DINCP D]			■	2	6	112
-1	16-bit	DEC	[DEC D]	(D)−1→(D)			2	2	110
		DECP	[DECP D]			■	2	6	110
	32-bit	DDEC	[DDEC D]	(D+1,D)−1→(D+1,D)			2	2	112
		DDECP	[DDECP D]			■	2	6	112
Complement of 2	16-bit	NEG	[NEG D]	• (D) → (D) BIN data			2	2	114
		NEGP	[NEGP D]			■	2	6	114
	32-bit	DNEG	[DNEG D]	• (D+1, D) → (D+1, D) BIN data		■	2	2	114
		DNEGP	[DNEGP D]			■	2	6	114

*1: 1 step is added when S is a constant number.
*2: 1 step is added when either S1 or S2 is a constant number.

6.2.5 BCD<->BIN Conversion Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
BCD	16-bit	BCD	[BCD S D]	• (S) $\xrightarrow{\text{BCD conversion}}$ (D) BIN (0 to 9999)			3	3	116
		BCDP	[BCDP S D]			■	3	7	116
	32-bit	DBCD	[DBCD S D]	• (S+1, S) $\xrightarrow{\text{BCD conversion}}$ (D+1, D) BIN (0 to 99999999)			3	3	116
		DBCDP	[DBCDP S D]			■	3	7	116
BIN	16-bit	BIN	[BIN S D]	• (S) $\xrightarrow{\text{BIN conversion}}$ (D) BCD (0 to 9999)			3	3	118
		BINP	[BINP S D]			■	3	7	118
	32-bit	DBIN	[DBIN S D]	• (S+1, S) $\xrightarrow{\text{BIN conversion}}$ (D+1, D) BCD (0 to 99999999)			3	3	118
		DBINP	[DBINP S D]			■	3	7	118

6.2.6 Data Transmission Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Transmission	16-bit	MOV	[MOV S D]	· (S) → (D)			3	3	120
		MOVP	[MOVP S D]			■	3	7	120
	32-bit	DMOV	[DMOV S D]	· (S+1,S) → (D+1,D)			3/4 *1	3/4 *1	120
		DMOVP	[DMOVP S D]			■	3/4 *1	7/8 *1	120
	16-bit	CML	[CML S D]	· $\overline{(S)}$ → (D)		■	3	3	122
		CMLP	[CMLP S D]			■	3	7	122
	32-bit	DCML	[DCML S D]	· $\overline{(S+1,S)}$ → (D+1,D)		■	3/4 *1	3/4 *1	122
		DCMLP	[DCMLP S D]			■	3/4 *1	7/8 *1	122
Conversion	16-bit	XCH	[XCH D1 D2]	· (D1) ↔ (D2)			3	3	124
		XCHP	[XCHP D1 D2]			■	3	7	124
	32-bit	DXCH	[DXCH D1 D2]	· (D1+1,D1) ↔ (D2+1,D2)			3	3	124
		DXCHP	[DXCH D1 D2]			■	3	7	124
Batch transmission	16-bit	BMOV	[BMOV S D n]	(S) → (D) n			4	4	126
		BMOVP	[BMOVP S D n]			■	4	8	126
Batch transmission of same data	16-bit	FMOV	[FMOV S D n]	(S) → (D) n			4	4	128
		FMOVP	[FMOVP S D n]			■	4	8	128
Timer transmission		S.TMOV	[S.TMOV S D]	Transfer of timer and counter setting value			6	3	130

*1: 1 step is added when S is a constant number.

6.2.7 Program Branch Instruction

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Jump	-	CJ	CJ P**	Jump to Pn upon establishment of input condition			2	2	131
		JMP	JMP P**	Jump to Pn unconditionally		■	2	2	131
		GOEND	GOEND	Jump to END instruction upon establishment of input condition		■	1	2	134
Program end	-	FEND	FEND	End process during sequence program			1	1	133
		END	END	End sequence program			1	1	
Subroutine call	-	CALL	CALL P**	Execute P** sub-routine program after input conditions are met			2	3	135
	-	CALLP	CALLP P**			■	2	7	135
Return	-	RET	RET	Return to main program from subroutine program			1	1	135
Repetition	-	FOR	FOR n	Execute the interval between FOR and NEXT for n times.		■	2	3	137
	-	NEXT	NEXT			■	1	3	137
	-	BREAK	BREAK D P**	Forcibly end the execution of the interval between FOR and NEXT, and jump to the pointer Pn.		■	3	4	139
	-	BREAKP	BREAKP D P**			■	3	8	139

6.2.8 Logical Operation Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Logical AND	16-bit	WAND	WAND S D	· $(D) \wedge (S) \rightarrow (D)$		■	3	3	142
		WANDP	WAND S D			■	3	7	142
		WAND	WAND S1 S2 D	· $(S1) \wedge (S2) \rightarrow (D)$			4	4	141
		WANDP	WANDP S1 S2 D			■	4	8	141
	32-bit	DAND	DAND S D	· $(D+1,D) \wedge (S+1,S) \rightarrow (D+1,D)$			3/4 *1	3/4 *1	142
		DANDP	DAND S1 S2 D			■	3/4 *1	7/8 *1	142
		DAND	DAND S1 S2 D	· $(S1+1,S1) \wedge (S2+1,S2) \rightarrow (D+1,D)$		■	4/5 *2	4/5 *2	141
		DANDP	DANDP S1 S2 D			■	4/5 *2	8/9 *2	141

(To be continued on the next page)

*1: 1 step is added when S is a constant number.
*2: 1step is added when either S1 or S2 is a constant number.

Logical operation instructions (continued)

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Logical OR	16-bit	WOR	[WOR S D]	· (D)∨(S)→(D)		■	3	3	145
		WORP	[WORP S D]			■	3	7	145
		WOR	[WOR S1 S2 D]	· (S1)∨(S2)→(D)			4	4	144
		WORP	[WORP S1 S2 D]			■	4	8	144
	32-bit	DOR	[DOR S D]	· (D+1,D)∨(S+1,S)→(D+1,D)			3/4 *1	3/4 *1	145
		DORP	[WNXR S1 S2 D]			■	3/4 *1	7/8 *1	145
		DOR	[DOR S1 S2 D]	· (S1+1,S1)∨(S2+1,S2)→(D+1,D)		■	4/5 *2	4/5 *2	144
		DORP	[DORP S1 S2 D]			■	4/5 *2	8/9 *2	144
Exclusive OR	16-bit	WXOR	[WXOR S D]	· (D)⊻(S)→(D)		■	3	3	148
		WXORP	[WXORP S D]			■	3	7	148
		WXOR	[WXOR S1 S2 D]	· (S1)⊻(S2)→(D)			4	4	147
		WXORP	[WXORP S1 S2 D]			■	4	8	147
	32-bit	DXOR	[DXOR S D]	· (D+1,D)⊻(S+1,S)→(D+1,D)			3/4 *1	3/4 *1	148
		DXORP	[DXORP S D]			■	3/4 *1	7/8 *1	148
		DXOR	[DXOR S1 S2 D]	· (S1+1,S1)⊻(S2+1,S2)→(D+1,D)		■	4/5 *2	4/5 *2	147
		DXORP	[DXORP S1 S2 D]			■	4/5 *2	8/9 *2	147
Non exclusive logical sum	16-bit	WXNR	[WXNR S D]	· $\overline{(D)⊻(S)}$→(D)		■	3	3	151
		WXNRP	[WXNR S D]			■	3	7	151
		WXNR	[WXNR S1 S2 D]	· $\overline{(S1)⊻(S2)}$→(D)		■	4	4	150
		WXNRP	[WXNRP S1 S2 D]			■	4	8	150
	32-bit	DXNR	[DXNR S D]	· $\overline{(D+1,D)⊻(S+1,S)}$→(D+1,D)		■	3/4 *1	3/4 *1	151
		DXNRP	[DXNRP S D]			■	3/4 *1	7/8 *1	151
		DXNR	[DXNR S1 S2 D]	· $\overline{(S1+1,S1)⊻(S2+1,S2)}$→(D+1,D)		■	4/5 *2	4/5 *2	150
		DXNRP	[DXNRP S1 S2 D]			■	4/5 *2	8/9 *2	150

*1: 1 step is added when S is a constant number.
*2: 1 step is added when either S1 or S2 is a constant number.

6.2.9 Rotation Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps: Storage	No. of steps: Execution	See for Description
Right rotation	16-bit	ROR	ROR D n	b15 (D) b0 SM12 Rotate n bits right.			3	3/4	153
		RORP	RORP D n			■	3	7/8	153
		RCR	RCR D n	b15 (D) b0 SM12 Rotate n bits right.			3	3/4	153
		RCRP	RCRP D n			■	3	7/8	153
	32-bit	DROR	DROR D n	(D+1) b31 ~ b16 (D) b15 ~ b0 SM12 Rotate n bits right.			3	3/4	157
		DRORP	DRORP D n			■	3	7/8	157
		DRCR	DRCR D n	(D+1) b31 ~ b16 (D) b15 ~ b0 SM12 Rotate n bits right.			3	3/4	157
		DRCRP	DRCRP D n			■	3	7/8	157
Left rotation	16-bit	ROL	ROL D n	SM12 b15 (D) b0 Rotate n bits left.			3	3/4	155
		ROLP	ROLP D n			■	3	7/8	155
		RCL	RCL D n	SM12 b15 (D) b0 Rotate n bits left.			3	3/4	155
		RCLP	RCLP D n			■	3	7/8	155
	32-bit	DROL	DROL D n	SM12 (D+1) b31 ~ b16 (D) b15 ~ b0 Rotate n bits left.			3	3/4	159
		DROLP	DROLP D n			■	3	7/8	159
		DRCL	DRCL D n	SM12 (D+1) b31 ~ b16 (D) b15 ~ b0 Rotate n bits left.			3	3/4	159
		DRCLP	DRCLP D n			■	3	7/8	159
Right shift	16-bit	SFR	SFR D n	b15 bn b0 b15 b0 SM12 0 ~ 0			3	3/4	161
		SFRP	SFRP D n			■	3	7/8	161
	Device unit	DSFR	DSFR D n	n (D) 0			3	3/4	163
		DSFRP	DSFRP D n			■	3	7/8	163
Left shift	16-bit	SFL	SFL D n	b15 bn b0 SM12 b15 b0 0~0			3	3/4	161
		SFLP	SFLP D n			■	3	7/8	161
	Device unit	DSFL	DSFL D n	n (D) 0			3	3/4	163
		DSFLP	DSFLP D n			■	3	7/8	163

6.2.10 Data Processing Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps Storage	No. of steps Execution	See for Description
Search	16-bit	SER	SER S1 S2 D	(S1) (S2) n (D): Match No. (D+1): Number of matches			5	6	165
		SERP	SERP S1 S2 D			■	5	10	165
	32-bit	DSER	DSER S1 S2 D	32 bit (S1) (S2) n (D): Match No. (D+1): Number of matches		■	5	6	165
		DSERP	DSERP S1 S2 D			■	5	10	165
Number of bits set to "1"	16-bit	SUM	SUM S D	(S) b15 b0 (D): Number of "1"s			3	3	167
		SUMP	SUMP S D			■	3	7	167
	32-bit	DSUM	DSUM S D	(S+1) (S) (D): Number of "1"s		■	3	3	167
		DSUMP	DSUMP S D			■	3	7	167
Decode	16-bit	SEG	SEG S D	b3 to bO (S) (D) 7SEG			3	3	169
		SEGP	SEGP S D			■	3	7	169
	2^n bit	DECO	DECO S D n	8→256 decode (S) Decode (D) n 2^n bit			4	4	171
		DECOP	DECOP S D n			■	4	8	171
Encode	2^n bit	ENCO	ENCO S D n	256→8 encode (S) 2 bit Encode (D) n		■	4	4	173
		ENCOP	ENCOP S D n			■	4	8	173
Average value	16-bit	S.AVE	S.AVE S D n	16-bit data average value $\frac{1}{n}\sum_{i=1}^{n}$ (S+i) → (D)		■	7	4	175

6.2.11 Other Function Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps		See for Description
							Storage	Execution	
Carry flag set	—	S.STC	S.STC	Carry flag contact (SM12) is turned ON.			4	1	176
Carry flag reset	—	S.CLC	S.CLC	Carry flag contact (SM12) is turned OFF.			4	1	176

6.2.12 Special Instructions for Old Machine Type Compatible

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps Storage	No. of steps Execution	See for Description
BIT	1-bit	LDBIT	[<= S1 n]	Bit test (A contact operation start handling) (Note 1)			3	2	177
		ANDBIT	[<= S1 n]	Bit test (A contact series connection handling) (Note 1)			3	2	177
		ORBIT	[<= S1 n]	Bit test (A contact parallel connection handling) (Note 1)			3	2	177
		LDBII	[<> S1 n]	Bit test (B contact operation start handling) (Note 1)			3	2	179
		ANDBII	[<> S1 n]	Bit test (B contact series connection handling) (Note 1)			3	2	179
		ORBII	[<> S1 n]	Bit test (B contact parallel connection handling) (Note 1)			3	2	179

(Note) These instructions can be used with the compatible instruction mode as they are compatible with old machine types but will not be available in the future.

6.2.13 Exclusive Instructions

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps Storage	No. of steps Execution	See for Description
ATC	—	S.ATC	[S.ATC Kn Rn Rm Mn]	K1: Tool No. search			8	5	191
				K2: Tool No. AND operation search					192
				K3: Tool change					193
				K4: Arbitrary position tool change					194
				K5: Forward run of pointer					195
				K6: Reverse run of pointer					195
				K7: Forward run of tool table					196
				K8: Reverse run of tool table					196
				K9: Tool table read					197
				K10: Tool table write					198
				K11: Automatic write of tool table					199
ROT	—	S.ROT	[S.ROT Kn Rn Rm Mn]	K1: Rotary body index			8	5	204
				K3: Ring counter					207

6.3 Data Designation Method

The following three types of data can be used in each instruction.

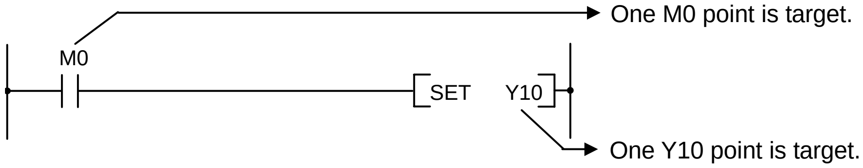
(1) Bit data
(2) Word (16-bit) data
(3) Double-word (32-bit) data

6.3.1 Bit Data

Bit data is data which handles contacts and coils, etc., in 1-bit units.
"Bit devices" and "bit-designated word devices" can be used as bit data.

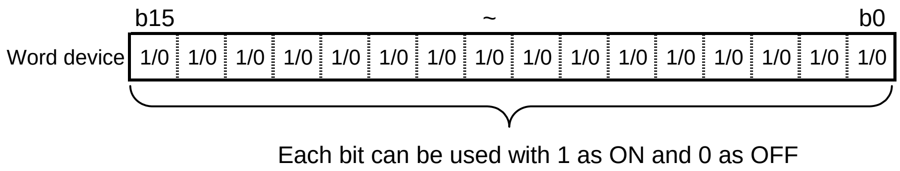
(1) When using bit devices

The bit device is designated with a 1-point unit.

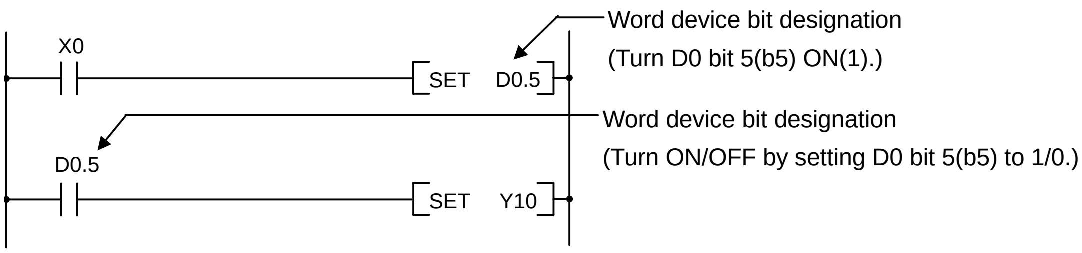


(2) When using word devices

With the word device, when the bit No. is designated, the bit for the designated bit No. can be used as bit data.



The word device bit is designated as "word device","bit number".
(The bit No. is designated with a hexadecimal.)
For example, D0 bit 5 (b5) is designated as D0.5, and D0 bit 10 (b10) is designated as D0.A.
Note that the bit designated cannot be used for the timer (T), integrated timer (ST) or counter (C).



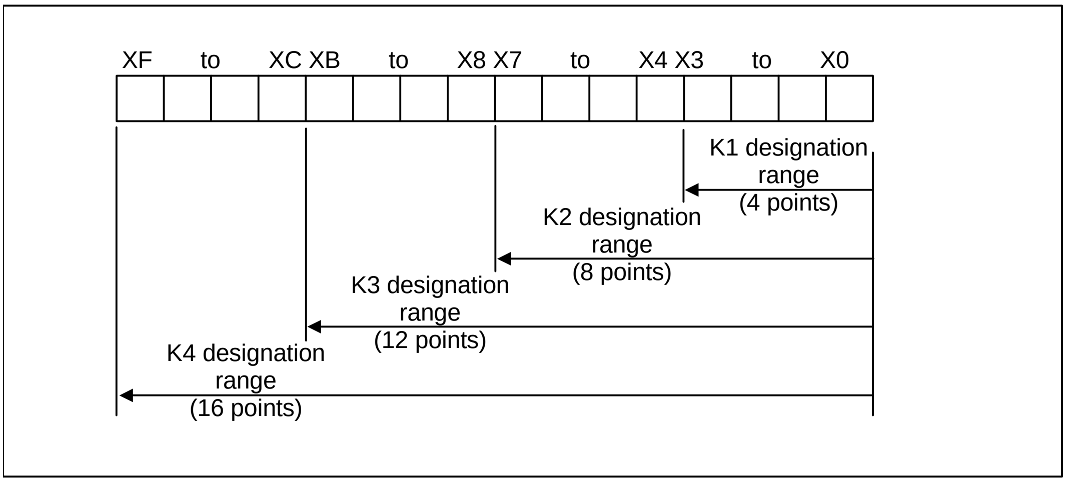
6.3.2 Word (16-bit) Data

Word data is a 16-bit numerical value data used with the basic instructions and applied instructions.
Word devices and digit-designated bit devices can be used with the word data.

(1) When using bit devices

Bit devices can be used as word devices by designating digits.
The bit data digit is designated as "number of digits""bit device head No.".
The digits can be designated between K1 and K4 with a 4-point (4-bit) unit.
The target numbers of points when X0 is designated as the digit are shown below.

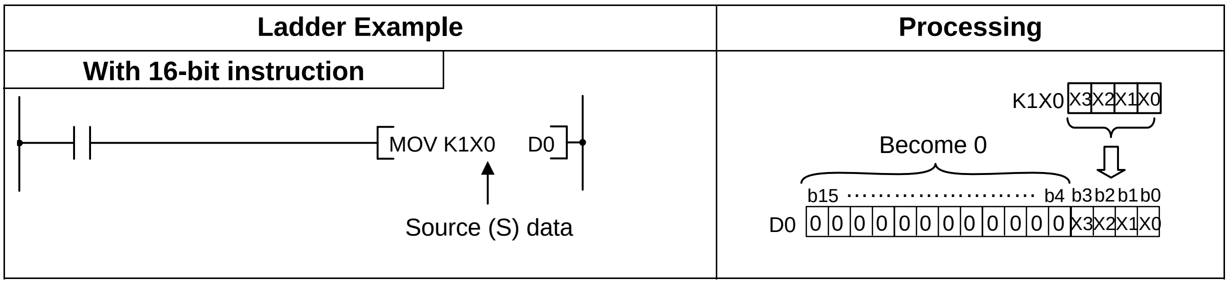
- K1X0 --- 4 points X0 to X3 are the target
- K2X0 --- 8 points X0 to X7 are the target
- K3X0 --- 12 points X0 to XB are the target
- K4X0 --- 16 points X0 to XF are the target



List of Numeric Values that Can Be Dealt with as Source Data for Digit Designation at Source (S) Side

Number of Digits Designated	With 16-Bit Instruction
K1 (4 points)	0 to 15
K2 (8 points)	0 to 255
K3 (12 points)	0 to 4095
K4 (16 points)	-32768 to 32767

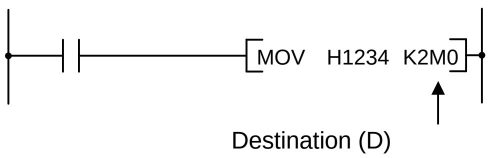
If the source side is a digit-designated bit device, and the destination is a word device, the word device on the destination side will be 0 after the digit-designated bits on the source side.

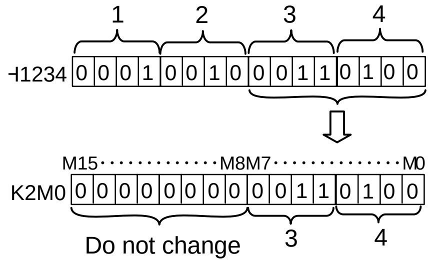


When a digit is designated on the destination (D) side, the No. of points designated by the digit will be the target of the destination side.

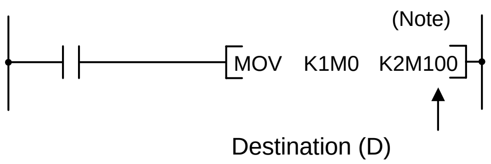
Ladder Example	Processing

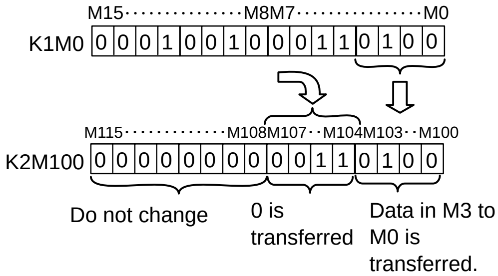
When source data (S) is a numerical value



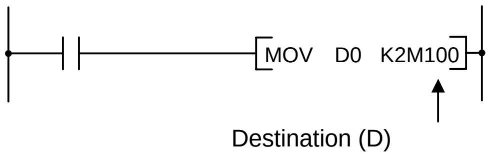


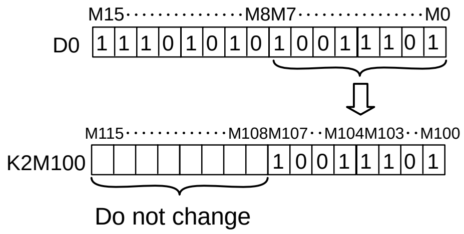
When source data (S) is a bit device





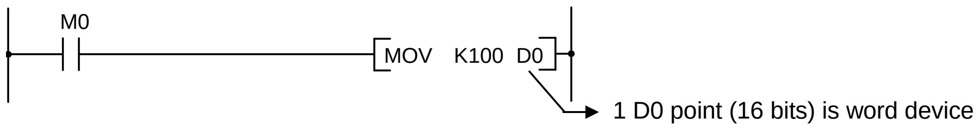
When source data (S) is a word device





(2) When using word devices

Word devices are designated in 1-point (16 bits) units.



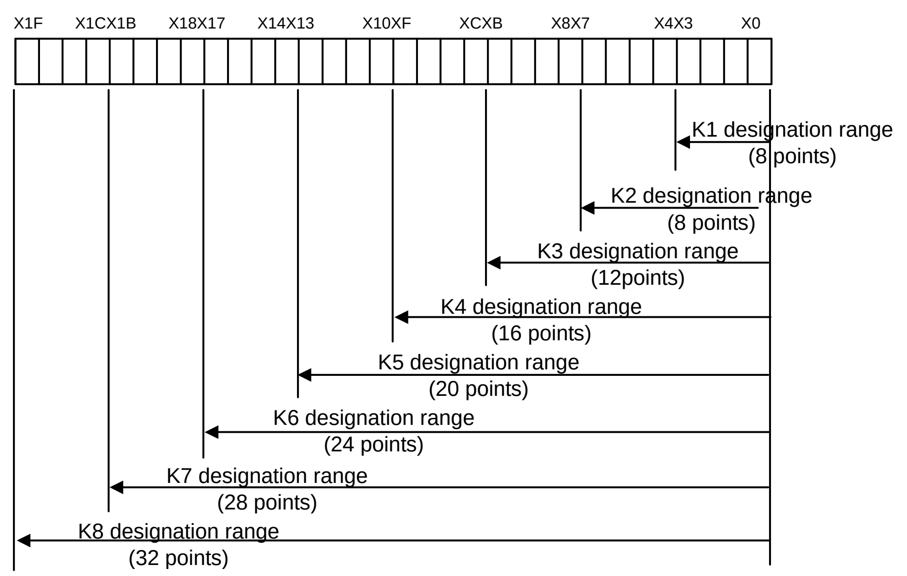
6.3.3 Using Double Word Data (32 bits)

Double word data is 32-bit numerical data used by basic instructions and application instructions.
Word devices and bit devices designated by digit designation can be used as double word data.

(1) When using bit devices

Digit designation can be used to enable a bit device to deal with double word data.
Digit designation of bit devices is done by designating "Number of digits" and "Initial number of bit device".
Digit designation of bit devices can be done in 4-point (4-bit) units, and designation can be made for K1 to K8.
For example, if X0 is designated for digit designation, the following points would be designated:

- K1X0 ….. The 4 points X0 through X3 are designated
- K2X0 ….. The 8 points X0 through X7 are designated
- K3X0 ….. The 12 points X0 through XB are designated
- K4X0 ….. The 16 points X0 through XF are designated
- K5X0 ….. The 20 points X0 through X13 are designated
- K6X0 ….. The 24 points X0 through X17 are designated
- K7X0 ….. The 28 points X0 through X1B are designated
- K8X0 ….. The 32 points X0 through X1F are designated

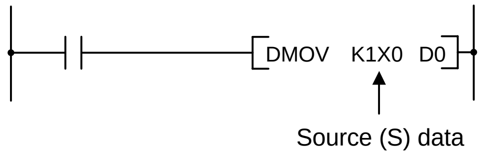


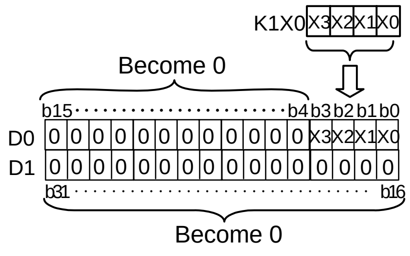
List of Numeric Values that Can Be Dealt with as Source Data for Digit Designation at Source (S) Side

Number of Digits Designated	With 32-bit Instructions	Number of Digits	With 32-bit Instructions
K1 (4 points)	0 to 15	K5 (20 points)	0 to 1048575
K2 (8 points)	0 to 255	K6 (24 points)	0 to 16772165
K3 (12 points)	0 to 4095	K7 (28 points)	0 to 268435455
K4 (16 points)	0 to 65535	K8 (32 points)	-2147483648 to 2147483647

In cases where the source is a bit device designated by digit designation, and the destination is a word device, the word device for the destination becomes 0 following the bit designated by digit designation at the source.

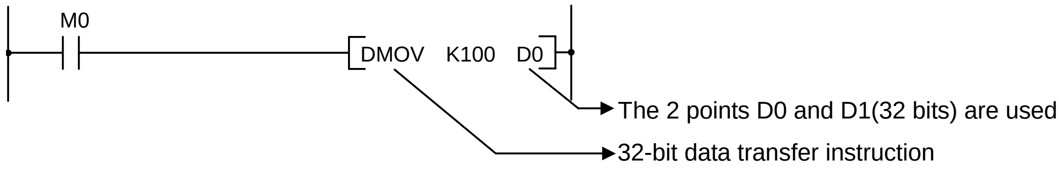
Ladder Example	Processing
With 32-bit Instructions	





(2) When using word devices

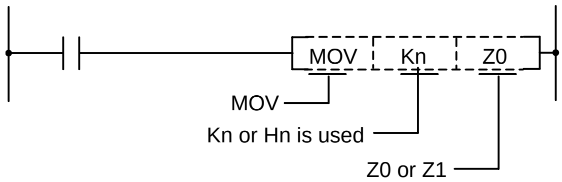
A word device designates devices used by the lower 16 bits of data.
A 32-bit instruction uses (designation device number) and (designation device number + 1).



6.4 Index Qualification

Index qualification is an indirect setting made by using an index register.
When an index qualification is used in a sequence program, the device to be used will become the device number designated directly plus the contents of the index register.
For example, if D0Z2 has been designated, and the content of Z2 is 3, D(2+3), or D5, will become the designated device.

(1) The index (Z0,Z1) can be set in the range of –32768 to 32767 with a sign added.
(2) The index qualification is used only for the MOV instruction. (DMOV cannot be used.)
(3) The usable instruction format is as shown below.
(a) Transmission of data to Z0, Z1.



(b) Possible device combination of MOV instruction with index qualification

	S (Source)	D (Destination)	Program example
MOV	Constant Kn or Hn	(Word device)•Z **Example:** D0Z0, R500Z1	MOV K100 D0Z0
	Word device **Example:** D0,R1900	(Word device)•Z **Example:** D0Z0, R500Z1	MOV D0 D100Z1
	(Word device)•Z **Example:** D0Z0	(Word device)•Z **Example:** D1Z0, D0Z1	MOV D0Z0 D20Z0
	(Word device)•Z **Example:** D0Z0	Bit device digit designation **Example:** K2Y20	MOV D0Z0 K2M10
	Bit device digit designation **Example:** K2M00	(Word device)•Z **Example:** D0Z0, R1900Z1	MOV K2M10 D0Z0

(Note 1) Word device indicates T, C, D, R, W, SW, and SD.

[Note] The range of the devices will not be checked if index qualification is applied to the devices during sequence program execution. Thus, keep in mind that the index register contents exceeds the device range at qualification, unexpected type of device are referred or renewed.

6.5 Operation Error

Operation error occurs in the following cases during execution of basic instruction and function instruction.

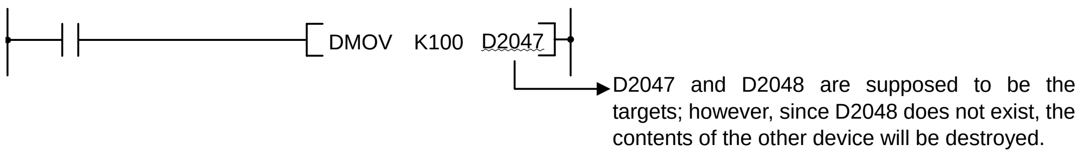
• In the case where an error described in each instruction's explanation page has occurred:

(1) Device range check

The range check for the devices to be used in basic instruction and function instruction is as shown below.

(a) If instruction handles a fixed length of device (MOV, DMOV, etc.), device range check will not be performed. In case that the relevant device range has exceeded, the data is written into another device.

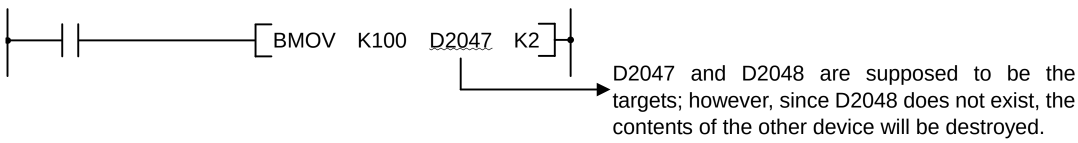
In such a case indicated below, error will not occur even if D2047 has been exceeded.



Device range check will not be performed when index qualification is carried out.

(b) If instruction handles a variable length of device (BMOV, FMOV, etc. that would specify the number of transfers), device range check will not be performed. In case that the relevant device range has exceeded, the data is written into another device.

In such a case indicated below, error will not occur even if D2047 has been exceeded.



(2) Device data check

Device's data check used for the basic instruction and function instruction is as shown below.

(a) BIN data

• Error does not occur even if the operation result is overflowed or underflowed. At this time, carry flag does not turn ON either.

(b) BCD data

• Checking as to whether each digit is BCD value (0/ to 9) is performed.
If the digit is other than 0 to 9 (A to F), an operation error will result.

• Error does not occur even if the operation result is overflowed or underflowed. At this time, carry flag does not turn ON either.

6.6 Execution Condition of Instruction

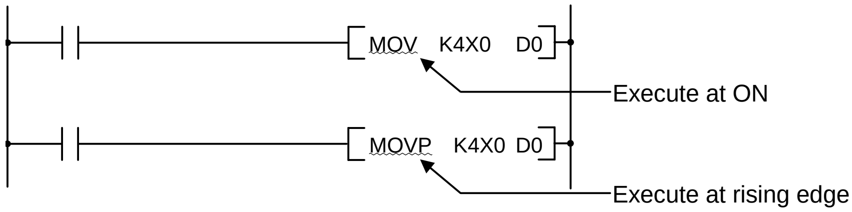
There are the following 4 kinds of execution conditions for basic instruction and function instruction.

- Execute all the time … Instruction executed regardless of device's ON/OFF
  Example: LD X0 , OUT Y10
- Execute at ON … Instruction executed when input condition is turned ON
  Example: MOV instruction, CJ instruction
- Execute at rising edge … Instruction executed only at the rising edge (OFF to ON) of input condition
  Example: PLS instruction, MOVP instruction
- Execute at falling edge … Instruction executed only at the falling edge (ON to OFF) of input condition
  Example: PLF instruction

With coil equivalent basic instruction and function instruction, if "execution at ON" and "execution at rising edge" are both possible with the same instruction, add "P" at the end of instruction to differentiate the execution condition.

- Instruction when executed at ON Instruction name
- Instruction when executed at rising edge Instruction name + P

With MOV instruction, execution at ON and execution at rising edge are specified as shown below.



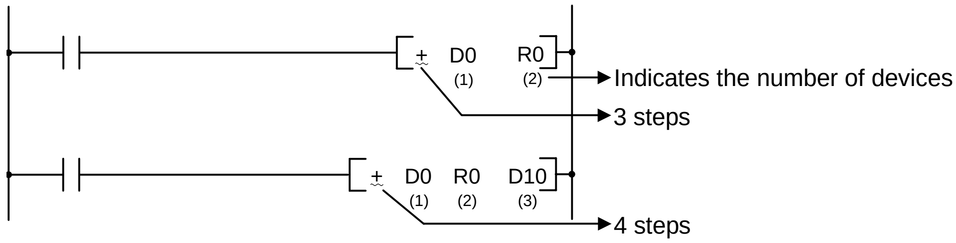
6.7 Counting Step Number

There are the following 2 step numbers. Refer to "2.6 Storing PLC Processing Program and Execution Mode " for details on the sequence program storage and execution.

(1) Number of steps during storage

This is the number of steps to be consumed when each instruction is stored in F-ROM. This is stored in the instruction code format which is compatible with the MELSEC sequencer.

Basic number of steps for basic instruction and function instruction is (number of specified devices + 1). For example, if "+ instruction", the number of steps is as shown below.



Devices where number of steps increases

Condition	Added Steps	Example
Specified device is a 32 bit constant	1	DMOV K123 D0
Instruction is a character string (S.xxx)	When the number of characters (S.xxx's "xxx") in the character string is: Even number : Number of characters /2 + 1 Odd number : (Number of characters+1)/2 + 1	S.AVE D882 D0 K7

(2) Number of steps during execution

This is the number of steps to be consumed when each instruction is executed. When executed, a sequence program is analyzed, optimizing the references and converting into the instruction code for the PLC processing processor which is unique to the CNC. Thus, the length of each instruction (number of step) is varied before and after the conversion.

The number of steps at storage and execution for each instruction is shown in "6.2 Instruction Tables".

Instruction list

Class	Process unit	Instruction sign	Symbol	Process details	Execution Condition	Ext. inst.	No. of steps Storage	No. of steps Execution	See for Description
+ (BIN)	16-bit	+	[+ S D]	(D)+(S)→(D) (BIN)		●	3	3	86
		+P	[+P S D]			●	3	7	86

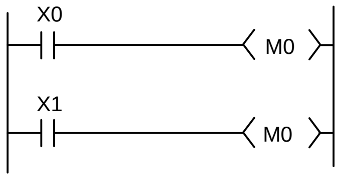
(a) (b)

6.8 Operations when the OUT, SET/RST, or PLS/PLF Instruction of the Same Device is Used

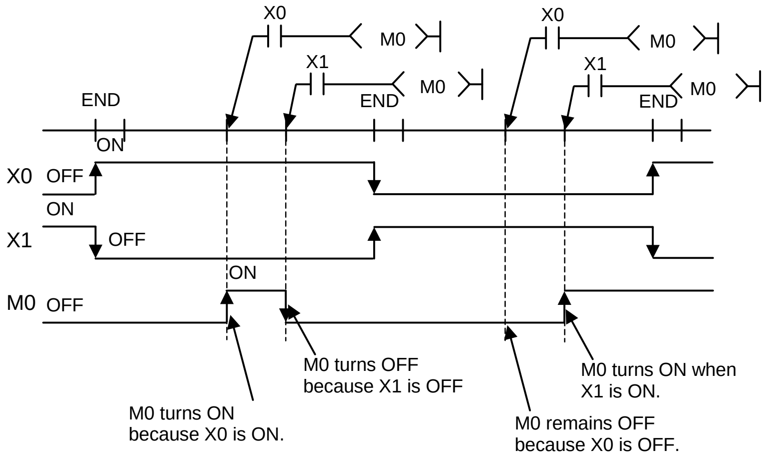
Operations when multiple OUT instructions, SET/RST instructions or PLS/PLF instructions using the same device are executed in one scan are explained.

(1) When the OUT instruction of the same device is used:
Do not execute the OUT instruction of the same device multiple times within one scan.
When the OUT instruction of the same device is executed multiple times in one scan, a specified device is turned ON/OFF at the time of each execution of OUT instruction, depending on the operation result up to the OUT instruction.
Because ON/OFF of the specified device is determined at the time of each execution of OUT instruction, ON/OFF may be repeated within one scan.
In the ladder example below, the same internal relay (M0) is turned ON/OFF with the input X0 and X1.

[Ladder]



[Timing chart]



In the case of refresh type CPU unit, when output (Y) is specified with OUT instruction, ON/OFF status of the OUT instruction executed at the end of 1 scan is output.

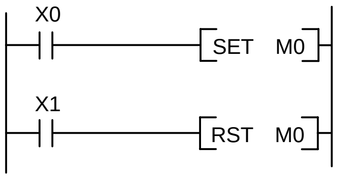
(2) When the SET/RST instruction of the same device is used:

(a) The SET instruction turns a specified device ON when the SET command is ON and it does not operate when the SET command is OFF. Therefore, when the SET instructions of the same device are executed multiple times within one scan, and if one or more SET commands are ON, the specified device will be turned ON.

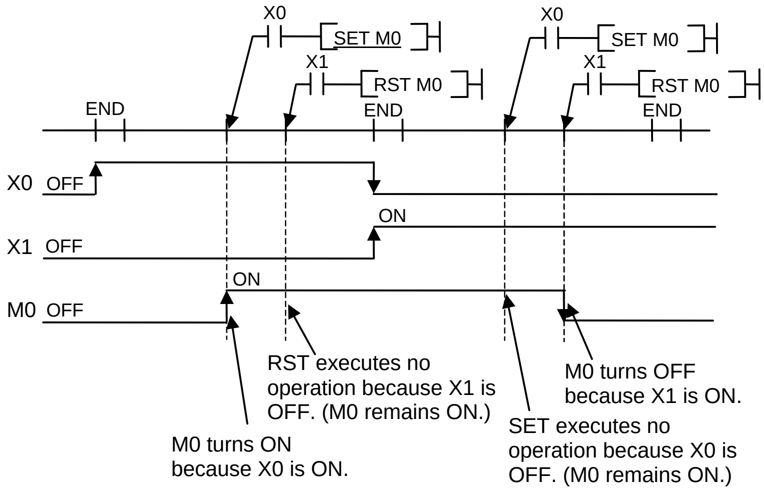
(b) The RST instruction turns a specified device OFF when the RST command is ON and it does not operate when the RST command is OFF. Therefore, when the RST instructions of the same device are executed multiple times within one scan, and if one or more RST commands are ON, the specified device will be turned OFF.

(c) When both the SET instruction and RST instruction of the same device exist within one scan, the SET instruction turns a specified device ON when the SET command is ON, and the RST instruction turns a specified device OFF when the RST command is ON.
When both the SET command and RST command are OFF, the specified device's ON/OFF status does not change.

[Ladder]



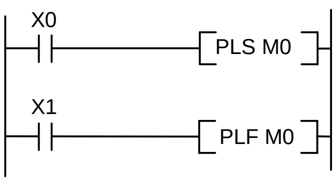
[Timing chart]



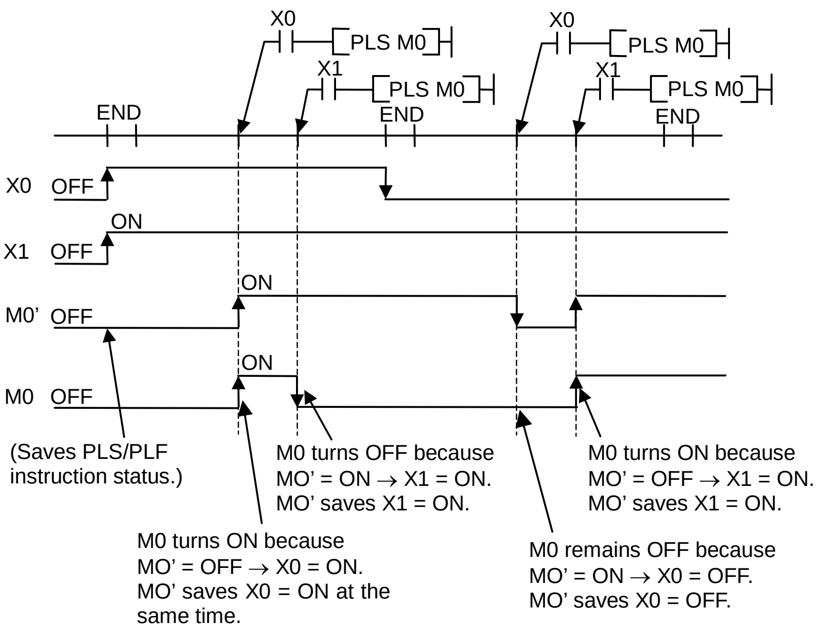
(3) When the PLS/PLF instruction of the same device is used:

The PLS instruction turns a specified device ON when the PLS command changes from OFF to ON, and it turns a specified device OFF when the PLS command changes in a way other than "from OFF to ON" (which means from OFF to OFF, ON to ON, or ON to OFF). Also, the PLF instruction turns a specified device ON when the PLF command changes from ON to OFF, and it turns a specified device OFF when the PLF command changes in a way other than "from ON to OFF" (which means from OFF to OFF, OFF to ON, or ON to ON). Note that, however, the PLS/PLF command's previous scan status is stored in the area that is unique to the device specified with the PLS/PLF instruction. Thus, when the PLS/PLF instruction of the same device is executed multiple times in one scan, the PLS/PLF command status saved with the first PLS/PLF instruction is used as the second PLS/PLF instruction's PLS/PLF command's previous scan status. In this way, when the PLS/PLF instruction is executed multiple times in one scan, the operation may not be carried out in a way as expected by the PLS/PLF instruction.

[Ladder]



[Timing chart]



6.9 How to Read Instruction Tables

The basic instructions and function instructions are explained as follows.

Instruction symbol is displayed

Outline of instruction is displayed

○D+, D+P, D-, D-P ……BIN32-bit addition and subtraction (Storage destination device independent type)

Compatible instruction mode	Usable instruction: **D+, D-**																						
Setting data	**Usable device**																					**Digit designation**	**Index**
	Bit device										**Word device**								**Constant**		**Pointer**		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1											○	○	○	○	○	○		○				○	
S2											○	○	○	○	○	○		○	○	○			
D											○	○	○	○	○	○		○					

Extended instruction mode	- - Blank - - (Usable for any instructions)																						
Setting data	**Usable device**																					**Digit designation**	**Index**
	Bit device										**Word device**								**Constant**		**Pointer**		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

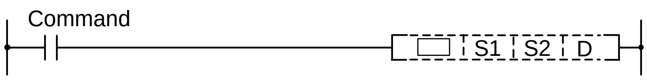
"T" is representing T, ST, and C.

A circle is placed for the device that can be used with the D+/D- instruction. A triangle is placed when there is a restriction in use.

A circle is placed when digit designation is possible for the bit device.

A circle is placed for the instruction where index (Z0 to Z1) can be used.

A list of usable instructions and devices for each instruction mode is provided here. "Compatible instruction mode" is the PLC instruction mode having instruction's compatibility with the conventional machine type. "Extended instruction mode" is the instruction mode in which the kinds of instructions and usable devices are extended. Refer to "6.1 Compatible Instructions and Extended Instructions" for details.

Instruction symbol	Execution condition	▭ indicates D+ / D-
D+, D-		
D+P, D-P		

Execution condition and ladder display of D+ / D- instructions are shown below.

Execution condition	Always executes	Executes at ON	Executes once at ON	Executes once at OFF
Signs in the explanation page	No sign			

Setting data	Details	Data type
S1	Augend/minuend data or head No. of the device where augend/minuend data is stored.	BIN 32 bits
S2	Addend/subtrahend data or head No. of the device where addend/subtrahend data is stored.	
D	Head No. of the device to store addition/subtraction result.	

Explanation of the setting data and data type for each instruction is provided.

The functions first, then execution conditions, then program examples are described on the following pages.

7. Basic Instructions

Basic instructions include instructions to describe relay control ladders, etc. They are divided into the following categories.

Instruction	Meaning
Contact instruction	Operation start, series connection, parallel connection
Connection instruction	Ladder block connection, creation of pulses from operation results, store/read operation results
Output instruction	Bit device output, pulse output, output reversal
Shift instruction	Bit device shift
Master control instruction	Master control
Termination instruction	Program termination
Other instructions	Instructions which do not fall into the above categories, such as no operation.

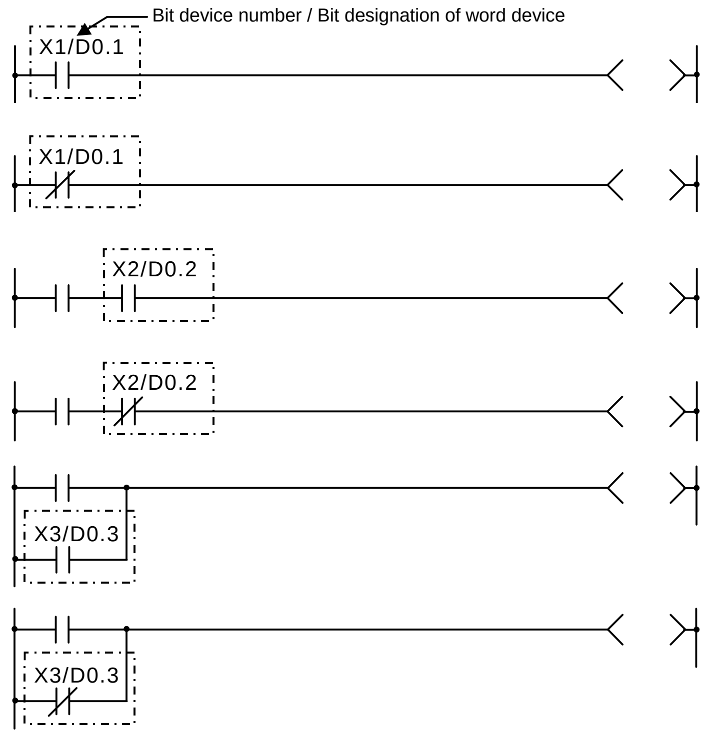
○ LD, LDI (Operation start) AND, ANI (Series connection) OR, ORI (Parallel connection)

Compatible/Extended instruction mode	

Set data	Usable devices																					Digit designation	Index
	Bit Devices									Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○	○	○				○	○	○	○		○					

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

Instruction symbol	Execution condition
LD	
LDI	
AND	
ANI	
OR	
ORI	



Set Data

Set Data	Meaning	Data Type
S	Devices used as connections	Bit

Functions

LD,LDI (1) LD is the A contact operation start instruction, and LDI is the B contact operation start instruction. They read ON/OFF information from the designated device (if a word device bit has been designated, this becomes the 1/0 status of the designated bit), and use that as an operation result.

AND,ANI (1) AND is the A contact series connection instruction, and ANI is the B contact series connection instruction. They read the ON/OFF data of the designated bit device (if a bit designation has been made for a word device, the 1/0 status of the designated bit is read), perform an AND operation on that data and the operation result to that point, and take this value as the operation result.

OR,ORI (1) OR is the A contact single parallel connection instruction, and ORI is the B contact single parallel connection instruction. They read ON/OFF information from the designated device (if a word device bit has been designated, this becomes the 1/0 status of the designated bit), and perform an OR operation with the operation results to that point, and use the resulting value as the operation result.

REMARK
Word device bit designations are made in hexadecimal. Bit b11 of D0 would be D0.B.

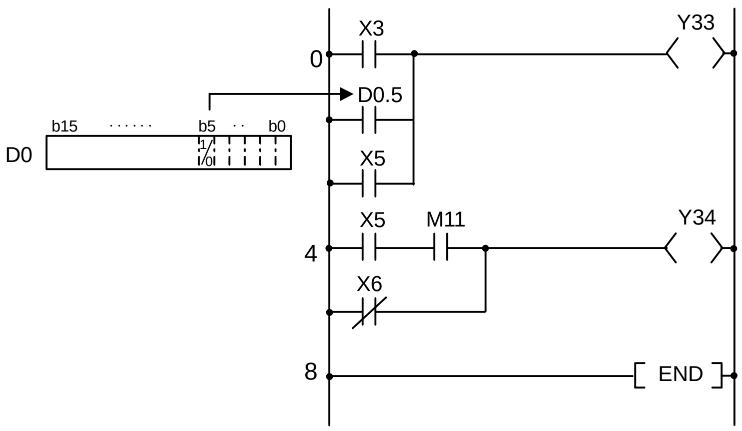
Operation Errors

(1) There are no operation errors with LD,LDI,AND,ANI,OR,or ORI instructions.

Program Example

(1) A program using LD, AND,OR, and ORI instructions.

[Ladder Mode]

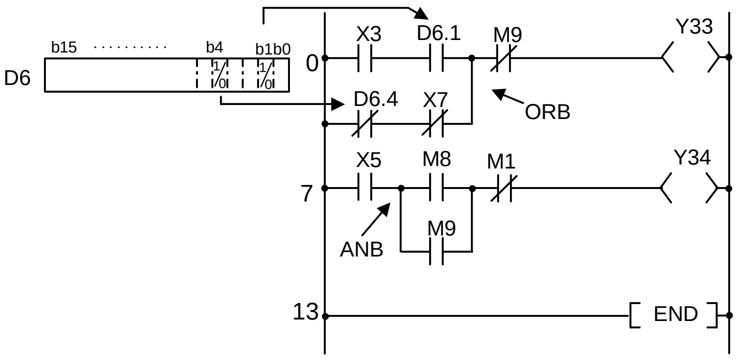


[List Mode]

Steps	Inst.	Device
0	LD	X3
1	OR	D0.5
2	OR	X5
3	OUT	Y33
4	LD	X5
5	AND	M11
6	ORI	X6
7	OUT	Y34
8	END	

(2) A program linking contact points established through the use of ANB and ORB instructions.

[Ladder Mode]

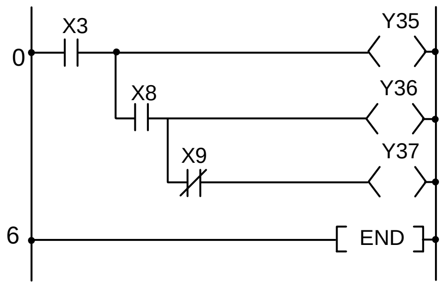


[List Mode]

Steps	Inst.	Device
0	LD	X3
1	AND	D6.1
2	LDI	D6.4
3	ANI	X7
4	ORB	
5	ANI	M9
6	OUT	Y33
7	LD	X5
8	LD	M8
9	OR	M9
10	ANB	
11	ANI	M11
12	OUT	Y34
13	END	

(3) A parallel program with OUT instruction

[Ladder Mode]



[List Mode]

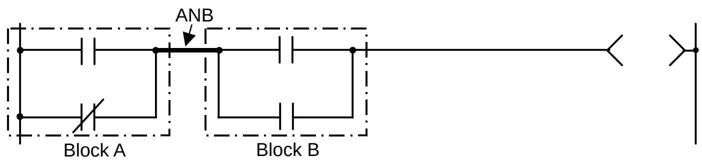
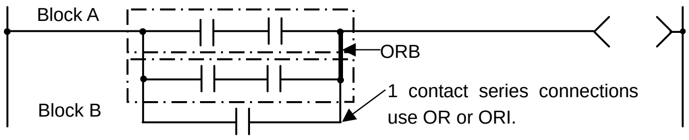
Steps	Inst.	Device
0	LD	X5
1	OUT	X35
2	AND	X8
3	OUT	Y36
4	ANI	X9
5	OUT	Y37
6	END	

○ ANB,ORB … Ladder block series connections and parallel connections

Compatible/Extended instruction mode	

Set data	Usable devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

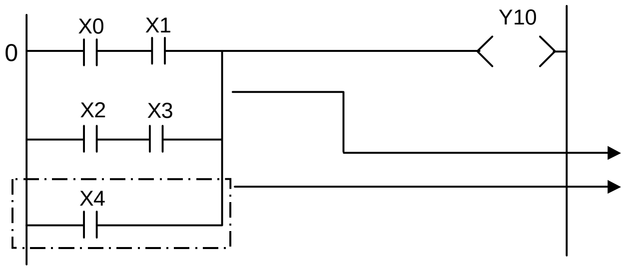
Instruction symbol	Execution condition	
ANB		
ORI		

Functions

ANB
(1) Performs an AND operation on block A and block B, and takes the resulting value as the operation result.
(2) The symbol for ANB is not the contact symbol, but rather is the connection symbol.
(3) When programming in the list mode, up to 7 instructions of ANB and ORB combined (8 blocks) can be used consecutively.

ORB
(1) Conducts an OR operation on block A and block B, and takes the resulting value as the operation result.
(2) ORB is used to perform parallel connections for ladder blocks with two or more contacts. For ladder blocks with only one contact, use OR or ORI; there is no need for ORB in such cases.

[Ladder Mode]



[List Mode]

```
0  LD   X0
1  AND  X1
2  LD   X2
3  AND  X3
4  ORB
5  OR   X4
6  OUT  Y10
```

(3) The ORB symbol is not the contact symbol, but rather is the connection symbol.
(4) When programming in the list mode, up to 7 instructions of ANB and ORB combined (8 blocks) can be used consecutively.

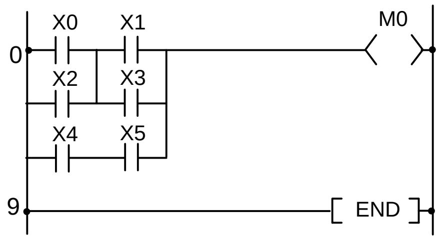
Operation Errors

(1) There are no operation errors with ANB or ORB instructions.

Program Example

(1) A program using ANB and ORB instructions

[Ladder Mode]



[List Mode]

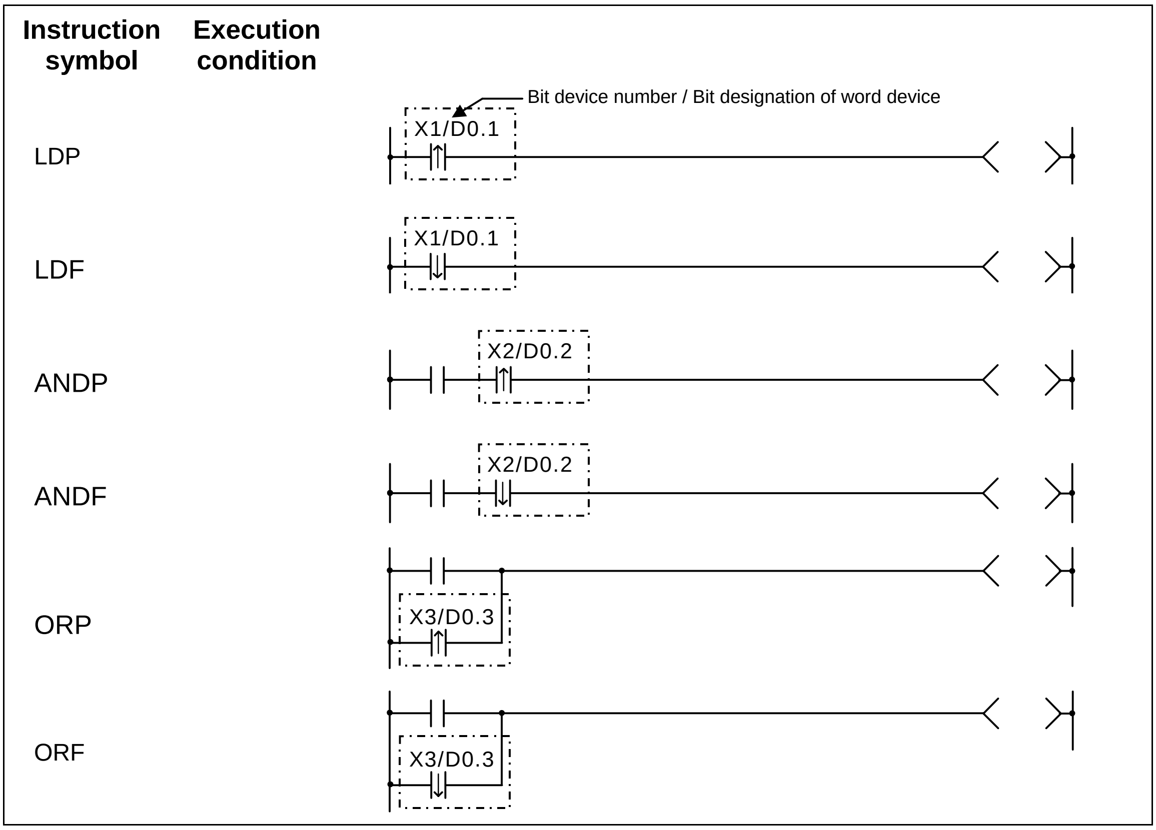
Steps	Inst.	Device
0	LD	X0
1	OR	X2
2	LD	X1
3	OR	X3
4	ANB	
5	LD	X4
6	AND	X5
7	ORB	
8	OUT	M0
9	END	

○ LDP,LDF,ANDP,ANDF,ORP,ORF … Pulse operation start, pulse series connection, pulse parallel connection

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable devices																				
	Bit Devices										Word Devices								Constant		Pointer	Digit desig-nation	Index
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○	○	○				○	○	○	○		○					



Set Data

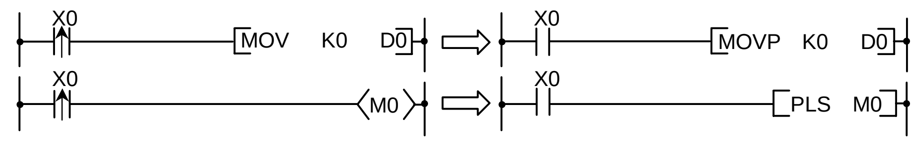
Set Data	Meaning	Data Type
S	Devices used as contacts	Bit

Functions

LDP,LDF

(1) LDP is the leading edge pulse operation start instruction, and is ON only at the leading edge of the designated bit device (when it goes from OFF to ON).
If a word device has been designated, it is ON only when the designated bit changes from 0 to 1.
In cases where there is only an LDP instruction, it acts identically to instructions for the creation of a pulse that are executed during ON (□ P).

A ladder using LDP instruction　　　　A ladder not using an LDP instruction



(2) LDF is the trailing edge pulse operation start instruction, and is ON only at the trailing edge of the designated bit device (when it goes from ON to OFF). If a word device has been designated, it is ON only when the designated bit changes from 1 to 0.

ANDP,ANDF

(1) ANDP is a leading edge pulse series connection instruction, and ANDF is a trailing edge pulse series connection instruction. They perform an AND operation with the operation result to that point, and take the resulting value as the operation result.
The ON/OFF data used by ANDP and ANDF are indicated in the table below:

Devices Designated by ANDP		ANDP State	Devices Designated by ANDF		ANDF State
Bit Device	Word Device Bit Designation		Bit Device	Word Device Bit Designation	
OFF→ON	0→1	ON	OFF→ON	0→1	OFF
OFF	0	OFF	OFF	0	
ON	1		ON	1	
ON→OFF	1→0		ON→OFF	1→0	ON

ORP,ORF

(1) ORP is a leading edge pulse parallel connection instruction, and ORF is a trailing edge pulse parallel connection instruction. They perform an OR operation with the operation result to that point and take the resulting value as the operation result.

Devices Designated by ORP		ORP State	Devices Designated by ORF		ORF State
Bit Device	Word Device Bit Designation		Bit Device	Word Device Bit Designation	
OFF→ON	0→1	ON	OFF→ON	0→1	OFF
OFF	0	OFF	OFF	0	
ON	1		ON	1	
ON→OFF	1→0		ON→OFF	1→0	ON

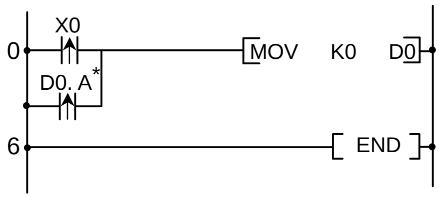
Operation Errors

(1) There are no operation errors with LDP, LDF, ANDP, ANDF, ORP, or ORF instructions.

Program Example

(1) The following program executes the MOV instruction at input X0, or at the leading edge of b10 (bit 10) of data register D0.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LDP	X0
2	ORP	D0. A
4	MOV	K0
		D0
6	END	

REMARK

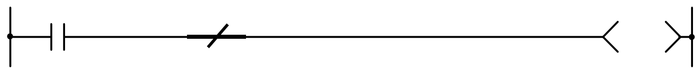
1)*: Word device bit designations are performed in hexadecimal.
Bit b10 of D0 would be D0.A.

○ INV … Operation results inversion

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																			Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

Instruction symbol	Execution condition
INV	

Functions

Inverts the operation result immediately prior to the INV instruction.

Operation Result Immediately Prior to the INV Instruction.	Operation Result Following the Execution of the INV Instruction.
OFF	ON
ON	OFF

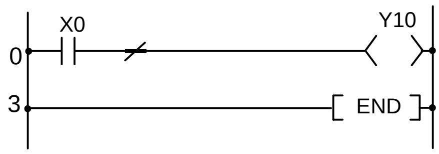
Operation Errors

(1) There are no operation errors associated with the INV instruction.

Program Example

(1) A program which inverts the X0 ON/OFF data, and outputs from Y10.

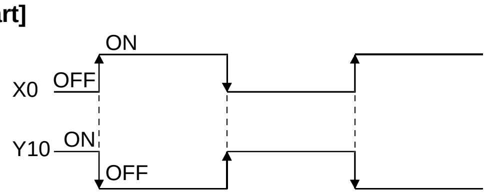
[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0
1	INV	
2	OUT	Y10
3	END	

[Timing Chart]



POINT
(1) The INV instruction operates based on the results of calculation made until the INV instruction is given. Accordingly, use it in the same position as that of the AND instruction. The INV instruction cannot be used at the LD and OR positions.

○ MEP,MEF … Operation result pulse conversion

Compatible instruction mode	Not available

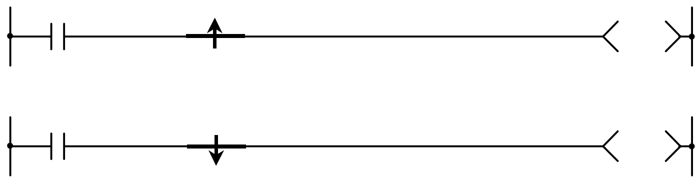
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

Instruction symbol **Execution condition**

MEP

MEF



Functions

MEP (1) If operation results up to MEP instruction are leading edge (from OFF to ON), goes ON (continuity state).
If operation results up to MEP instruction are anything other than leading edge, goes OFF (non-continuity state).

(2) Use of the MEP instruction simplifies pulse conversion processing when multiple contacts are connected in series.

MEF (1) If operation results up to MEF instruction are trailing edge (from ON to OFF), goes ON (continuity state).
If operation results up to MEF instruction are anything other than trailing edge, goes OFF (non-continuity state).

(2) Use of the MEF instruction simplifies pulse conversion processing when multiple contacts are connected in series.

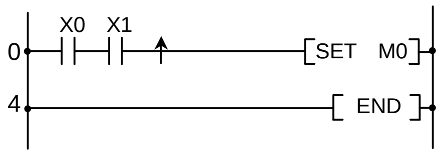
Operation Errors

(1) There are no operation errors associated with the MEP or MEF instructions.

Program Example

(1) A program which performs pulse conversion on the operation results of X0 and X1.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0
1	AND	X1
2	MEP	
3	SET	M0
4	END	

POINT

(1) Because the MEP and MEF instructions operate with the operation results immediately prior to the MEP and MEF instructions, the AND instruction should be used at the same position.
The MEP and MEF instructions cannot be used at the LD or OR position.

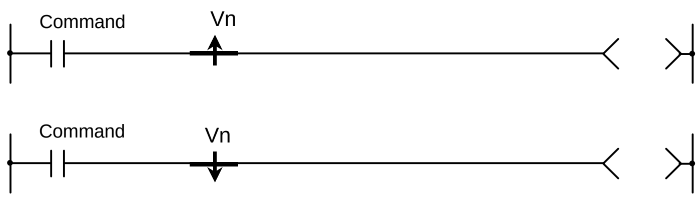
○ EGP,EGF … Pulse conversion of edge relay operation results

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
										○													

Instruction symbol	Execution condition
EGP	
EGF	



Set Data

Set Data	Meaning	Data Type
Vn	Edge relay No. where operation results are stored	Bit

Functions

EGP (1) Operation results up to the EGP instruction are stored in memory by the edge relay (V).

(2) Goes ON (continuity state) at the leading edge (OFF to ON) of the operation result up to the EGP instruction.
If the operation result up to the EGP instruction is other than a leading edge (i.e., from ON to ON, ON to OFF, or OFF to OFF), it goes OFF (non-continuity state).

(3) The EGP instruction can be used like an AND instruction.

EGF (1) Operation results up to the EGF instruction are stored in memory by the edge relay (V).

(2) Goes ON (continuity state) at the trailing edge (from ON to OFF) of the operation result up to the EGF instruction.
If the operation result up to the EGF instruction is other than a trailing edge (i.e., from OFF to ON, ON to ON, or OFF to OFF), it goes OFF (non-continuity state).

(3) The EGF instruction can be used like an AND instruction.

Operation Errors

(1) There are no operation errors associated with the EGP or EGF instructions.

POINT

(1) Because the EGP and EGF instructions operate with the operation results immediately prior to the EGP and EGF instructions, the AND instruction should be used at the same position.
The EGP and EGF instructions cannot be used at the LD or OR position.

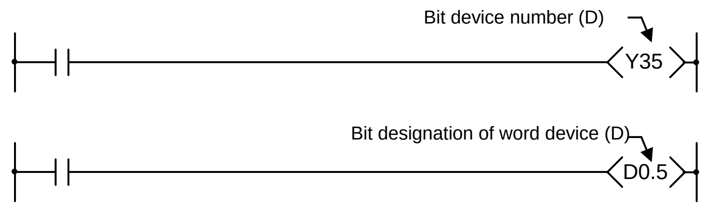
○ OUT(Y,M,L,F,B,SB,SM) … Out instructions (Y,M,L,F,B,SB,SM)

Compatible instruction mode

Set Data	Usable Devices																									Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D		○	○	○	○	○	○		○				○	○	○	○		○					

Extended instruction mode

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○				○	○	○	○		○					

Instruction symbol	Execution condition
OUT	

Set Data

Set Data	Meaning	Data Type
D	Number of device to be turned ON and OFF	Bit

Functions

(1) Operation results up to the OUT instruction are output to the designated device.

Operation results	When Using Bit Devices			When Bit Designation has been Made for Word Device
	Coil	Contact		Bit Designated
		A Contact	B Contact	
OFF	ON	Non-continuity	Continuity	0
ON	ON	Continuity	Non-continuity	1

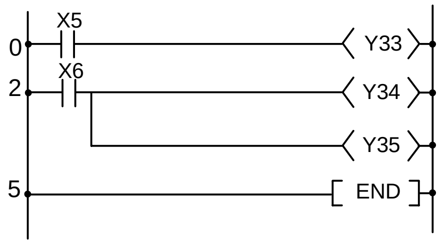
Operation Errors

(1) There are no operation errors associated with the OUT instruction.

Program Example

(1) When bit device is in use

[Ladder Mode]

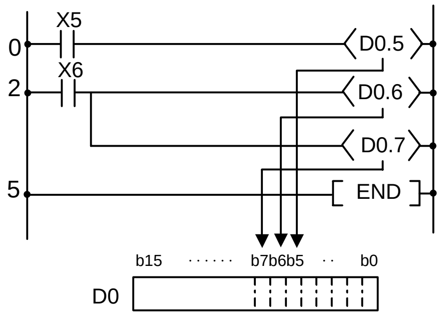


[List Mode]

Steps	Inst.	Device
0	LD	X5
1	OUT	Y33
2	LD	X6
3	OUT	Y34
4	OUT	Y35
5	END	

(2) When bit designation has been made for word device

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X5
1	OUT	D0.5
2	LD	X6
3	OUT	D0.6
4	OUT	D0.7
5	END	

○ OUT(T,ST), OUTH(T) … 100ms, 10ms timer output

Compatible/Extended instruction mode	

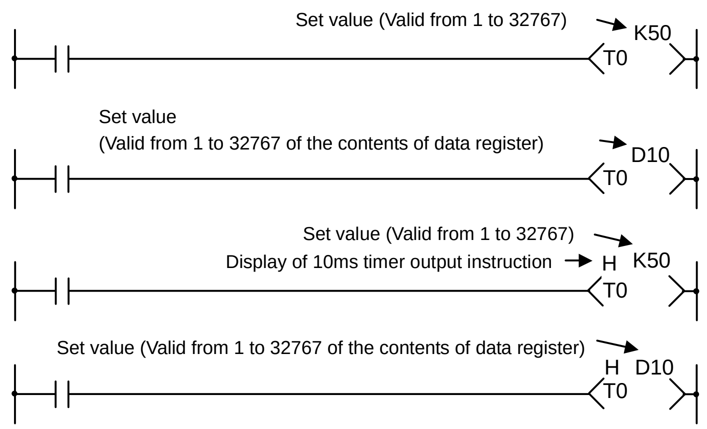
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○												
Set value													○	○	○	○		○	○				

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

Instruction symbol **Execution condition**

OUT T ☐
(100ms timer)
OUT ST ☐
(100ms retentive timer)

OUTH T ☐
(10ms timer)



Set Data

Set Data	Meaning	Data Type
D	Timer number	Bit
Set value	Value set for timer	BIN 16 bits

REMARK
(1) Timer values can be set only as a decimal constant (K). Hexadecimal constants (H) cannot be used for timer settings.
(2) The retentive timer (ST) cannot be used for the 10ms timer.

Functions

(1) When the operation results up to the OUT instruction are ON, the timer coil goes ON and the timer counts up to the value that has been set; when the time up state (total numeric value is equal to or greater than the setting value), the contact responds as follows:

A contact	Continuity
B contact	Non-continuity

(2) The following will apply if the calculation result up to OUT instruction changes from ON to OFF.

Type of Timer	Timer Coil	Present Value of Timer	Prior to Time Up		After Time Up	
			A Contact	B Contact	A Contact	B Contact
100ms timer	OFF	0	Non-continuity	Continuity	Continuity	Non-continuity
100ms retentive timer	OFF	Maintains the present value	Non-continuity	Continuity	Continuity	Non-continuity
10ms timer	OFF	0	Non-continuity	Continuity	Continuity	Non-continuity

(3) The contact status of retentive timer after time-up will not be changed until the RST instruction is executed.

(4) A negative number (-32768 to -1) cannot be set as the setting value for the timer.
If a negative value is set for the word device, operation is carried out taking the value as a positive value with no signs.

(5) When 0 is set for the set value, time will be up instantly.

(6) In cases where the OUT instruction is not executed while the OUT instruction is ON due to the JMP instruction, etc., no present value update or contact ON/OFF operation is conducted. Also, if the same OUT instruction is conducted two or more times during the same scan, the present value will be updated for the number of times executed.

POINT

(1) Setting the timer setting value using the setting display device.
The method for setting the value of timer T and retentive timer ST includes the following two ways.
- A method in which the setting value (Kn) programmed by a sequence program is validated. (Fixed timer)
- A method in which the setting value set with the setting display device is validated. (Variable timer)
Refer to "5.3 Detailed Explanation of Devices" for details on variable timer.

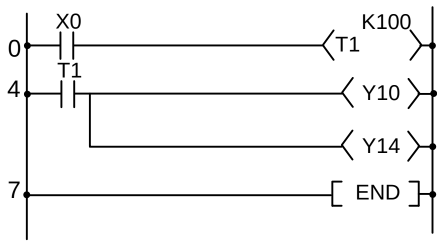
Operation Errors

(1) There are no operation errors associated with the OUT T ☐ or OUTH T ☐ instruction.

Program Example

(1) The following program turns Y10 and Y14 ON 10 seconds after X0 has gone ON.

[Ladder Mode]

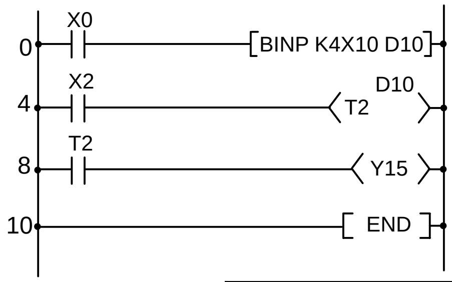


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	OUT	T1
		K100
4	LD	T1
5	OUT	Y10
6	OUT	Y14
7	END	

(2) The following program uses the BCD data at X10 to X1F as the timer's set value.

[Ladder Mode]



Converts BCD data at X10 to X1F to BIN and stores at D10.

When X2 goes ON, the data stored at D10 is calculated as the set value.

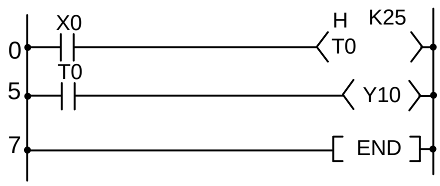
Y15 goes ON when T2 counts up.

[List Mode]

Steps	Inst.	Device
0	LD	X0
1	BINP	K4X10
		D10
4	LD	X2
5	OUT	T2
		D10
8	LD	T2
9	OUT	Y15
10	END	

(3) The following program turns Y10 ON 250ms after X0 has gone ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0
1	OUTH	T0
		K25
5	LD	T0
6	OUT	Y10
7	END	

○ OUT(C) … Counters

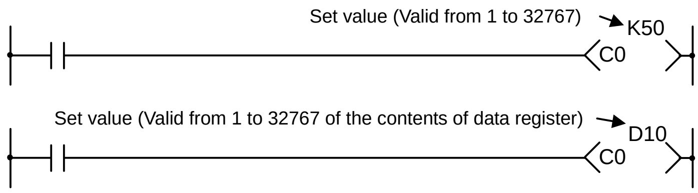
Compatible/Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D												○											
Set value													○	○	○	○		○	○				

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

Instruction symbol **Execution condition**

OUT C □



Set Data

Set Data	Meaning	Data Type
D	Counter No.	Bit
Set value	Counter set value	BIN 16 bits

REMARK

(1) Only decimal constant (K) can be used for the counter setting value.
Hexadecimal constant (H) cannot be used for the counter setting value.

Function

(1) When the operation results up to the OUT instruction change from OFF to ON, 1 is added to the present value (count value) and the count up status (present value = set value), and the contacts respond as follows:

A contact	Continuity
B contact	Non-continuity

(2) Not counted if the operation result is remained ON. (There is no need to perform pulse conversion on count input.)

(3) After "present value≧set value" has been realized, the contact state will not be changed until RST instruction is executed, but the present value is further added by +1.
In this case, the present value is added by +1 up to 65535, and then counted up again by +1 from 0.

(4) A negative number (-32768 to -1) cannot be set as the setting value. If a negative value is set for the word device, operation is carried out taking the value as a positive value with no signs. If the set value is 0, the processing is identical to that of when 1 is set.

POINT

(1) Setting the timer setting value using the setting display device.
The method for setting the value of timer T and retentive timer ST includes the following two ways.
• A method in which the setting value (Kn) programmed by a sequence program is validated. (Fixed timer)
• A method in which the setting value set with the setting display device is validated. (Variable timer)
Refer to "5.3 Detailed Explanation of Devices" for details on variable timer.

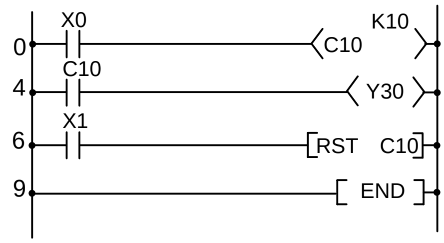
Operation Errors

(1) There are no operation errors associated with the OUT C ☐ instruction.

Program Example

(1) The following program turns Y30 ON after X0 has gone ON 10 times, and resets the counter when X1 goes ON.

[Ladder Mode]

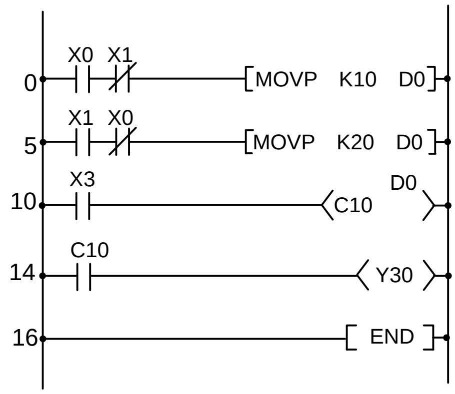


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	OUT	C10
		K10
4	LD	C10
5	OUT	Y30
6	LD	X1
7	RST	C10
9	END	

(2) The following program sets the value for C10 at 10 when X0 goes ON, and at 20 when X1 goes ON.

[Ladder Mode]



Stores 10 at D0 when X0 goes ON

Stores 20 at D0 when X1 goes ON

C10 takes data stored at D0 as set value, and counts

Y30 goes ON when C10 reaches count up state

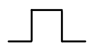
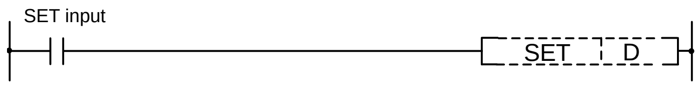
[List Mode]

Steps	Inst.	Device
0	LD	X0
1	ANI	X1
2	MOVP	K10
		D0
5	LD	X1
6	ANI	X0
7	MOVP	K20
		D0
10	LD	X3
11	OUT	C10
		D0
14	LD	C10
15	OUT	Y30
16	END	

○ SET … Setting devices (ON)

Compatible instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D		○	○	○	○	○	○		○				○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○				○	○	○	○		○					

Instruction symbol	Execution condition	
SET		

Set Data

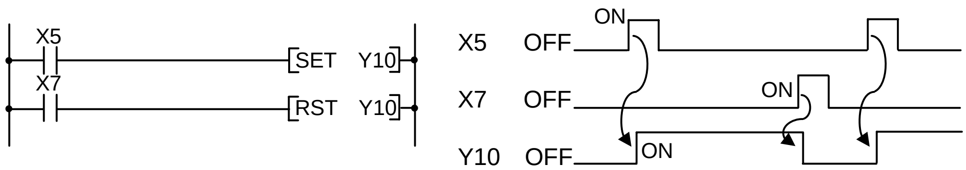
Set Data	Meaning	Data Type
D	Bit device number to be set (ON)	Bit

Functions

(1) When SET input is ON, the designated devices respond as follows:

Device	Device Status
Bit device	Coils and contacts turned ON
Word device	Designation bit set at 1

(2) Devices turned ON will stay ON even if SET input goes to OFF. Devices turned ON by the SET instruction can be turned OFF by the RST instruction.



(3) Device status does not change when SET input is OFF.

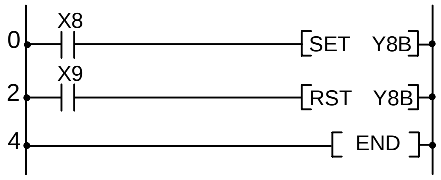
Operation Errors

(1) There are no operation errors associated with the SET instruction.

Program Example

(1) When X8 is turned ON, Y8B is set (turned ON); when X9 is turned ON, Y8B is reset (turned OFF).

[Ladder Mode]

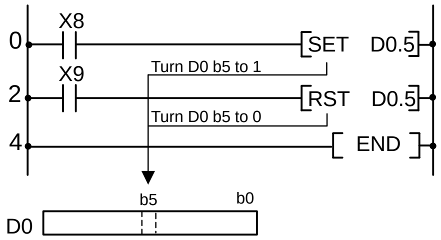


[List Mode]

Steps	Inst.	Device
0	LD	X8
1	SET	Y8B
2	LD	X9
3	RST	Y8B
4	END	

(2) When X8 is turned ON, D0 bit5 (b5) is turned to 1; when X9 is turned ON, D0 bit 5 (b5) is turned to 0.

[Ladder Mode]



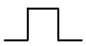
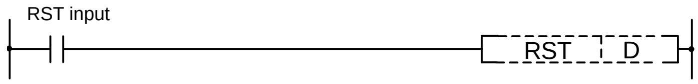
[List Mode]

Steps	Inst.	Device
0	LD	X8
1	SET	D0.5
2	LD	X9
3	RST	D0.5
4	END	

○ RST … Resetting devices

Compatible instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D		○	○	○	○	○	○		○				○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○				○	○	○	○		○					

Instruction symbol	Execution condition	
RST		

Set Data

Set Data	Meaning	Data Type
D	Device No. to be reset	Bit

Functions

(1) Designated devices respond as follows when RST input is turned ON:

Device	Device Status
Bit device	Turns coils and contacts OFF
Timers and counters	Sets the present value to 0, and turns coils and contacts OFF
Word device	Sets value of designated bit to 0

(2) Device status does not change when RST input goes OFF

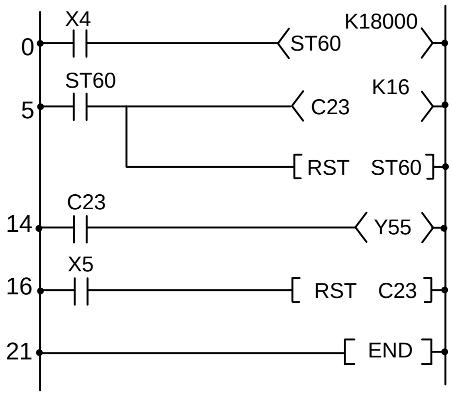
Operation Errors

(1) There are no operation errors associated with the RST instruction.

Program Example

(1) Reset of 100ms retentive timer and counter is executed.

[Ladder Mode]



When ST60 is set for the retentive timer, ST60 is turned ON when X4's ON time reaches 30 minutes.

The number of times when ST60 has turned ON is counted.

When ST60 is turned ON, ST60 is reset.

When C23 counts up, Y55 is turned ON.

When X5 is turned ON, C23 is reset.

[List Mode]

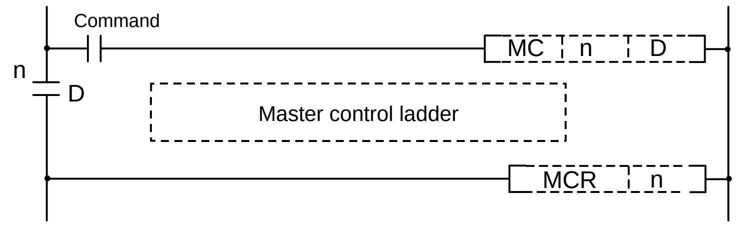
Steps	Inst.	Device
0	LD	X4
1	OUT	ST60
		K18000
5	LD	ST60
6	OUT	C23
		K16
10	RST	ST60
14	LD	C23
15	OUT	Y55
16	LD	X5
17	RST	C23
21	END	

○ MC,MCR … Setting and resetting the master control

Compatible instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Level		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	N		
n																					○		
D		○	○	○	○	○	○		○														

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Level		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	N		
n																					○		
D	○	○	○	○	○	○	○		○				○	○	○	○		○					

Instruction symbol	Execution condition
MC	
MCR	

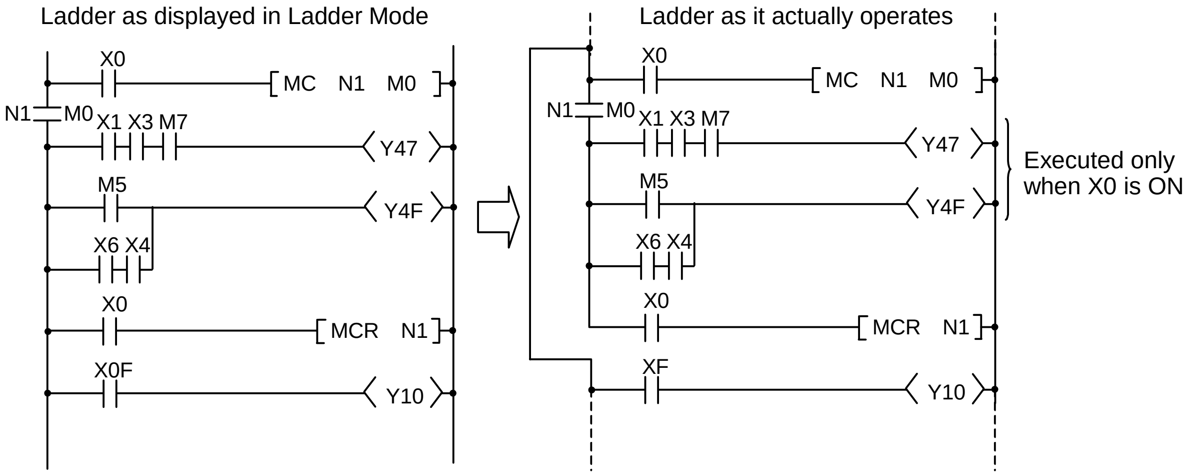


Set Data

Set Data	Meaning	Data Type
n	Nesting (N0 to N7)	Nesting
D	Number of device to turn ON	Bit

Functions

The master control instruction is used to enable the creation of highly efficient ladder switching sequence programs, through the opening and closing of a common bus for ladders.



A ladder using the master control would look as shown below:

MC

(1) If the ON/OFF command of the MC instruction is ON when master control is commenced, the operation result between the MC instruction and MCR instruction will be exactly as the instruction (ladder) shows.

If the MC ON/OFF instruction is OFF, the operation result between MC and MCR instructions will be as follows:

Device	Device Status
100ms, 10ms timer	Count value goes to 0
100ms cumulative timer counter	Current count value is kept as it is.
Devices in OUT instruction	All turned OFF
Devices in the following instructions: SET/RST, SFT function	Maintain current status

(2) Nesting can be used up to 8 times (N0 to 7). When using nesting, nests should be inserted from the lower to higher nesting number (N) with the MC instruction, and from the higher to the lower order with the MCR instruction.

(3) Regardless of the MC instruction's ON/OFF state, scan between MC instruction and MCR instruction can be executed.

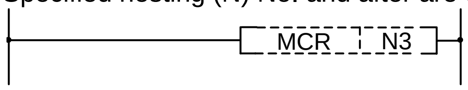
(4) MC instruction can be used as many times as you wish within one scan by changing devices in the destination D.

(5) When MC instruction is ON, coil of the device specified in the destination turns ON.

MCR

(1) This is the instruction for recovery from the master control, and indicates the end of the master control range of operation.

(2) Specified nesting (N) No. and after are cleared.



N3 to N7 master control
is cleared

Operation Errors

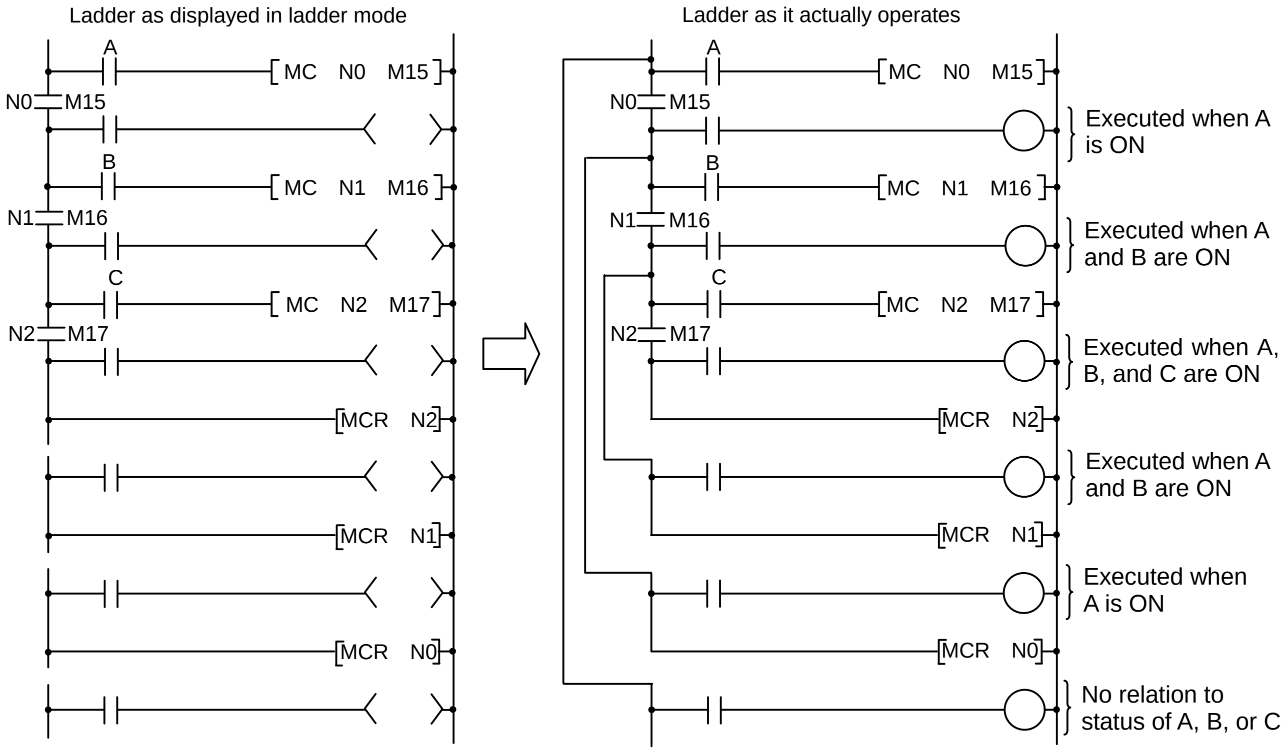
(1) There are no operation errors associated with the MC or MCR instructions.

Program Example

The master control instruction can be used in nesting. The different master control regions are distinguished by nesting (N). Nesting can be used from N0 through N7.

The use of nesting enables the creation of ladders which successively limit the execution condition of the program.

A ladder using nesting would appear as shown below:



○ PLS,PLF … Pulse (1 scan ON)

Compatible instruction mode

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D		○	○	○	○	○	○		○														

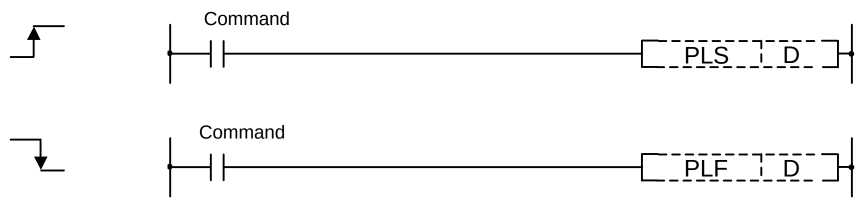
Extended instruction mode

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○														

Instruction symbol | **Execution condition**

PLS

PLF



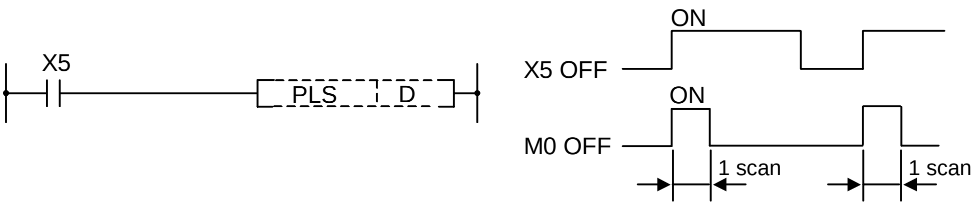
Set Data

Set Data	Meaning	Data Type
D	Pulse conversion device	Bit

Functions

PLS

(1) When PLS instruction is turned OFF to ON, turn specified device ON for 1 scan; otherwise (ON→ON,ON→OFF,OFF→ON), turn OFF.

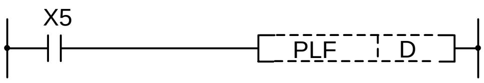


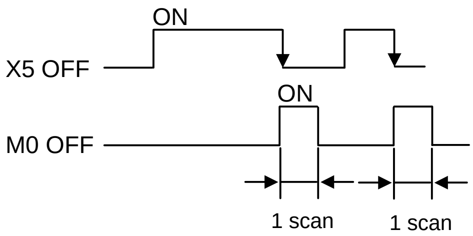
(2) If the RUN key switch is changed from RUN to STOP after the execution of the PLS instruction, the PLS instruction will not be executed again even if the switch is set back to RUN. PLS instruction will be executed if the PLS instruction has been ON when the power was turned ON.

(3) When a latch relay (L) is specified for the PLS instruction, switching power OFF with the latch relay (L) ON and then switching it ON again executes the specified device's 1scan ON.

PLF

(1) When PLF instruction is changed from ON to OFF, the designated device is turned 1 scan ON. For the other cases (OFF→OFF,OFF→ON,ON→ON), the designated device is turned OFF.





(2) Turn the sequence program RUN switch to STOP after PLF instruction. Even if switched to RUN again, PLF instruction will not be executed.

POINT

Note that the device designated by D may be ON more than one scan if the PLS or PLF instruction is jumped by the CJ instruction or if the subroutine program where the PLS/PLF instruction had been executed was not called by the CALL instruction.

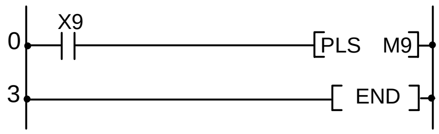
Operation Errors

(1) There are no operation errors associated with the PLS or PLF instructions.

Program Example

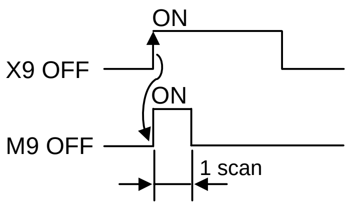
(1) The following program executes the PLS instruction when X9 goes ON.

[Ladder Mode]



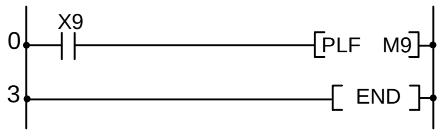
[List Mode]

Steps	Inst.	Device
0	LD	X9
1	PLS	M9
3	END	



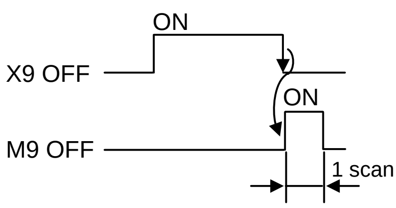
(2) The following program executes the PLF instruction when X9 goes OFF.

[Ladder Mode]



[List Mode]

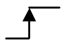
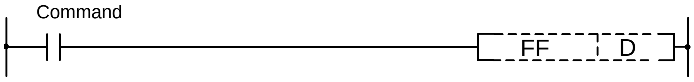
Steps	Inst.	Device
0	LD	X9
1	PLF	M9
3	END	



○ FF … Reversing the operation result

Compatible instruction mode	Not available

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○				○	○	○	○		○					

Instruction symbol	Execution condition	
FF		

Set Data

Set Data	Meaning	Data Type
D	Device number to invert	Bit

Function

(1) The status of the device designated by (D) is inverted when the inversion command goes from OFF to ON.

Device	Device Status	
	Prior to FF execution	After FF execution
Bit device	OFF	ON
	ON	OFF
Bit designation of word device	0	1
	1	0

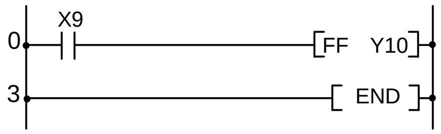
Operation Errors

(1) There are no operation errors associated with the FF instruction.

Program Example

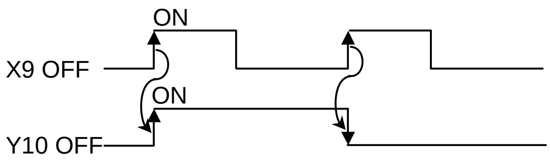
(1) The following program inverts the output of Y10 when X9 goes ON.

[Ladder Mode]



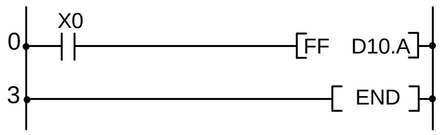
[List Mode]

Steps	Inst.	Device
0	LD	X9
1	FF	Y10
3	END	



(2) The following program reverses b10 (bit 10) of D10 when X0 goes ON

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0
1	FF	D10.A
3	END	

○ SFT,SFTP … Device shift

Compatible instruction mode	Usable instruction: **SFT**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D		○	○	○	○	○	○		○														

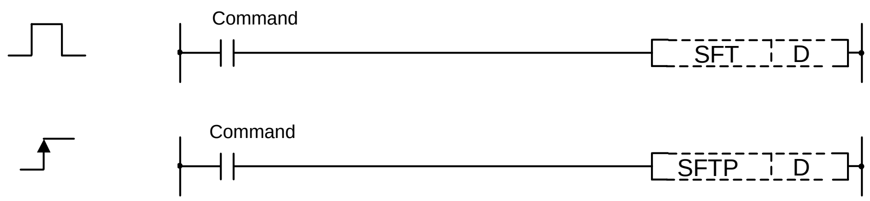
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○				○	○	○	○		○					

Instruction symbol **Execution condition**

SFT

SFTP



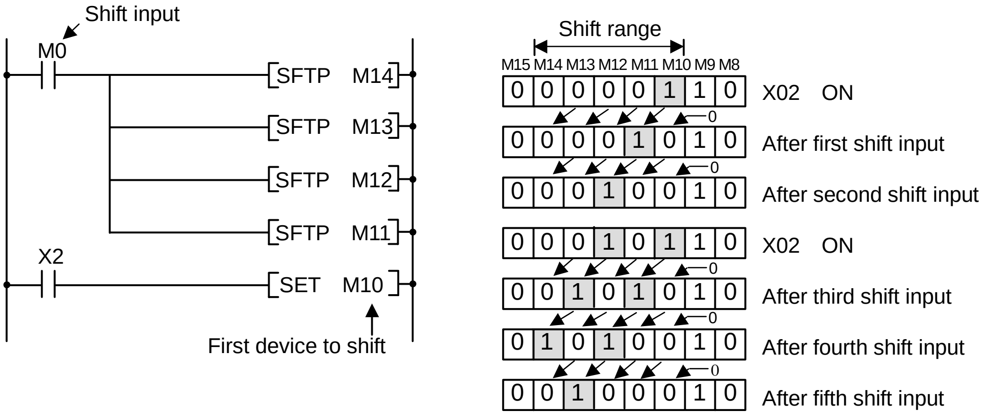
Set Data

Set Data	Meaning	Data Type
D	Number of device to shift	Bit

Functions

(1) When bit device is used

(a) Shifts to a device designated by (D) the ON/OFF status of the device immediately prior to the one designated, and turns the prior device OFF.



*At M8 to 15, "1" indicates ON and "0" indicates OFF.

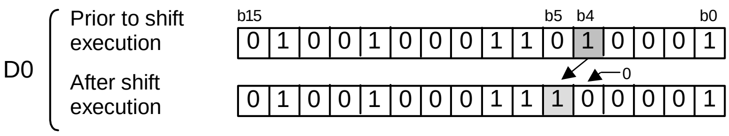
For example, if M11 has been designated by the SFT instruction, when the SFT instruction is executed, it will shift the ON/OFF status of M10 to M11, and turn M10 OFF.

(b) Turn the first device to be shifted ON with the SET instruction.

(c) When the SFT and SFTP are to be used consecutively, the program starts from the device with the larger number.

(2) When word device bit designation is used

(a) Shifts to a bit in the device designated by (D) the 1/0 status of the bit immediately prior to the one designated, and turns the prior bit to 0.
For example, if D0.5 (bit 5 [b5] of D0) has been designated by the SFT instruction, when the SFT instruction is executed, it will shift the 1/0 status of b4 of D0 to b5, and turn b4 to 0.



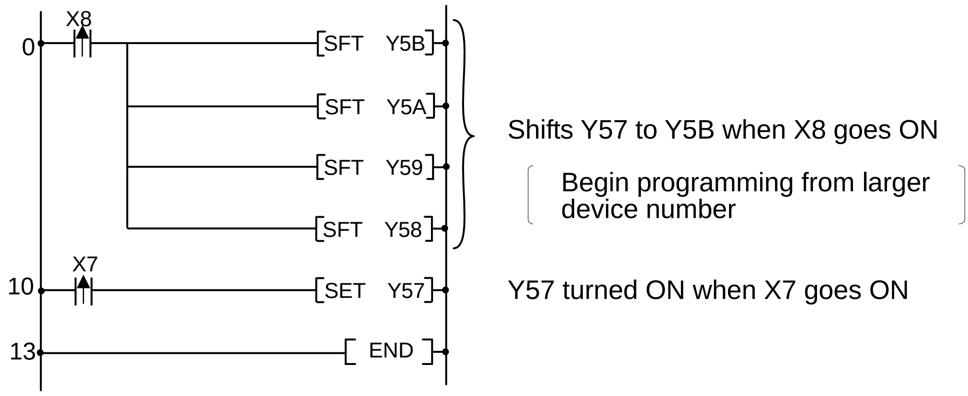
Operation Errors

(1) There are no operation errors associated with the SFT(P) instruction.

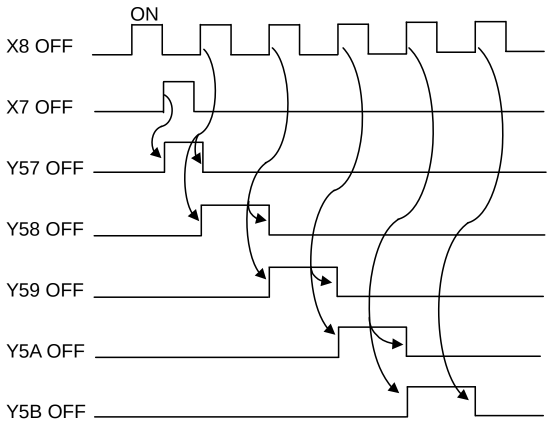
Program Example

(1) The following program shifts Y57 to Y5B when X8 goes ON.

[Ladder Mode]



[Timing Chart]



[List Mode]

Steps	Inst.	Device
0	LDP	X8
2	SFT	Y5B
4	SFT	Y5A
6	SFT	Y59
8	SFT	Y58
10	LDP	X7
12	SET	Y57
13	END	

○ MPS,MRD,MPP … Store, read and clear of operation result

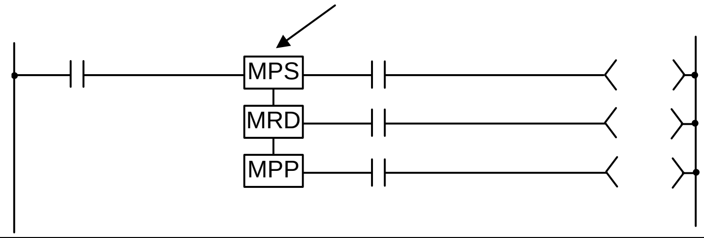
Compatible/Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

Instruction symbol	Execution condition

MPS, MRD, and MPP are not displayed as a part of the ladder display.



Function

MPS

(1) Stores in memory the operation result (ON or OFF) immediately prior to the MPS instruction.

(2) Up to 8 MPS instructions can be used successively. However, if an MPP instruction is used in the middle of process, the number of uses calculated for the MPS instruction will be decremented by one.

MRD

(1) Reads the operation result stored for the MPS instruction, and uses that result to perform the operation in the next step.

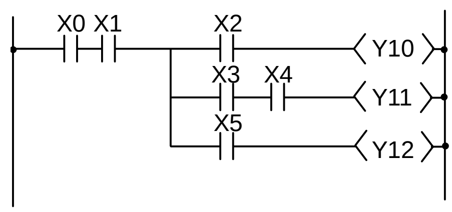
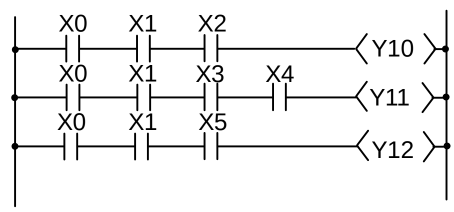
MPP

(1) Reads the operation result stored for the MPS instruction, and uses that result to perform the operation in the next step.

(2) Clears the operation results stored by the MPS instruction.

POINT

(1) The following shows ladders both using and not using the MPS, MRD, and MPP instructions.

Ladder using the MPS, MRD, and MPP instructions.	Ladder not using MPS, MRD, and MPP instructions.
	

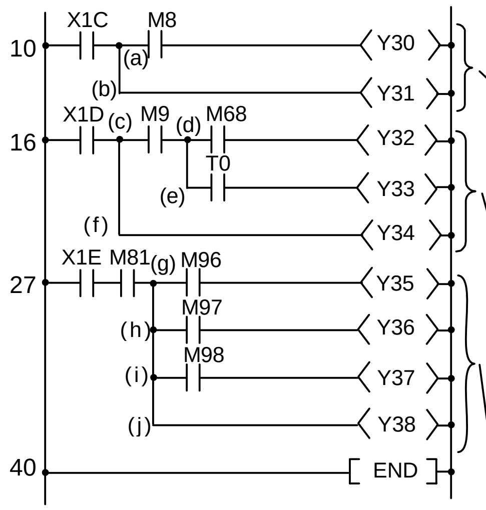
Operation Errors

(1) There are no errors associated with the MPS, MRD, or MPP instructions.

Program Example

(1) A program using the MPS, MRD, and MPP instructions.

[Ladder Mode]



[List Mode]

	Steps	Inst.	Device
	10	LD	X1C
(a)	11	MPS	
	12	AND	M8
	13	OUT	Y30
(b)	14	MPP	
	15	OUT	Y31
	16	LD	X1D
(c)	17	MPS	
	18	AND	M9
(d)	19	MPS	
	20	AND	M68
	21	OUT	Y32
(e)	22	MPP	
	23	AND	T0
	24	OUT	Y33
(f)	25	MPP	
	26	OUT	Y34
	27	LD	X1E
	28	AND	M81
(g)	29	MPS	
	30	AND	M96
	31	OUT	Y35
(h)	32	MRD	
	33	AND	M97
	34	OUT	Y36
(i)	35	MRD	
	36	AND	M98
	37	OUT	Y37
(j)	38	MPP	
	39	OUT	Y38
	40	END	

○ NOP,NOPLF,PAGE n … No operation

Compatible instruction mode	Usable instruction: **NOP**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
n																			○	○			

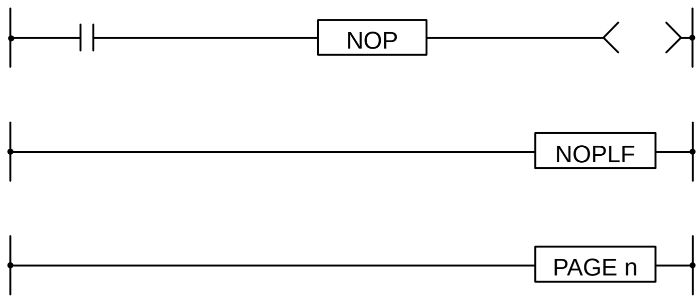
Instruction symbol	Execution condition

*NOP is not displayed in ladder display.

NOP

NOPLF

PAGE n



Functions

NOP

(1) This is a no operation instruction that has no impact on any operations up to this point.

(2) The NOP instruction is used in the following cases:
  (a) To insert space for sequence program debugging.
  (b) To delete an instruction without having to change the number of steps. (Replace the instruction with NOP)
  (c) To temporarily delete an instruction.

NOPLF

(1) This is a no operation instruction that has no impact on any operations up to that point.

(2) The NOPLF instruction is used when printing from a peripheral device to force a page change at any desired location.
  (a) When printing ladders
    • A page break will be inserted between ladder blocks with the presence of the NOPLF instruction.
    • The ladder cannot be displayed correctly if an NOPLF instruction is inserted in the midst of a ladder block.
      Do not insert an NOPLF instruction in the midst of a ladder block.
  (b) When printing instruction lists
    • The page will be changed after the printing of the NOPLF instruction.

(3) See the Operating Manual for the peripheral device in use for more information regarding printouts from peripheral devices.

PAGE n (1) This is a no operation instruction that has no impact on any operations up to that point.

(2) Programs after PAGEn instruction are controlled as 0 step and after of the specified n-th page. (Peripheral device display, printers, etc.)

(3) If there is no PAGEn instruction, processing begins from page 0.

Operation Errors

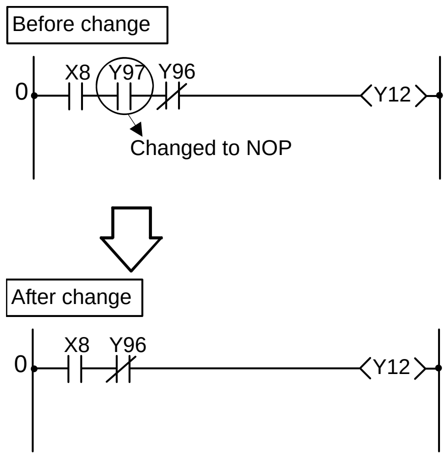
(1) There are no errors associated with the NOP, NOPLF, or PAGE instructions.

Program Example

NOP

(1) Contact closed…Deletes AND or ANI instruction

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X8
1	AND	Y97
2	ANI	X96
3	OUT	Y12
4	END	

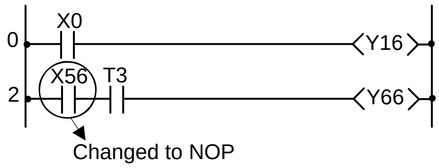
[List Mode]

Steps	Inst.	Device
0	LD	X8
1	NOP	
2	ANI	X96
3	OUT	Y12
4	END	

(2) Contact closed…LD, LDI changed to NOP
(Note carefully that changing the LD and LDI instructions to NOP completely changes the nature of ladder.)

[Ladder Mode]

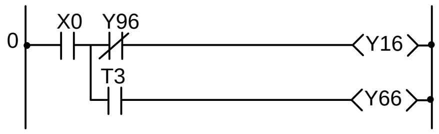
Before change



[List Mode]

Steps	Inst.	Device
0	LD	X0
1	OUT	Y16
2	LD	X56
3	AND	T3
4	OUT	Y66
5	END	

After change

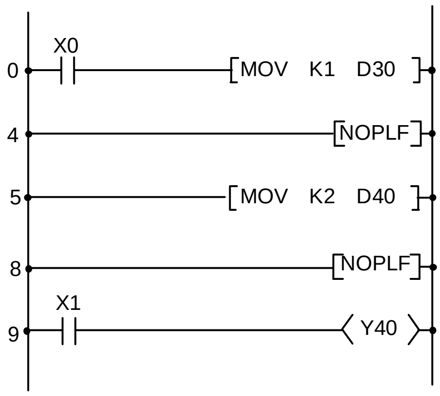


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	OUT	Y16
2	NOP	
3	ANI	T3
4	OUT	Y66
5	END	

NOPLF

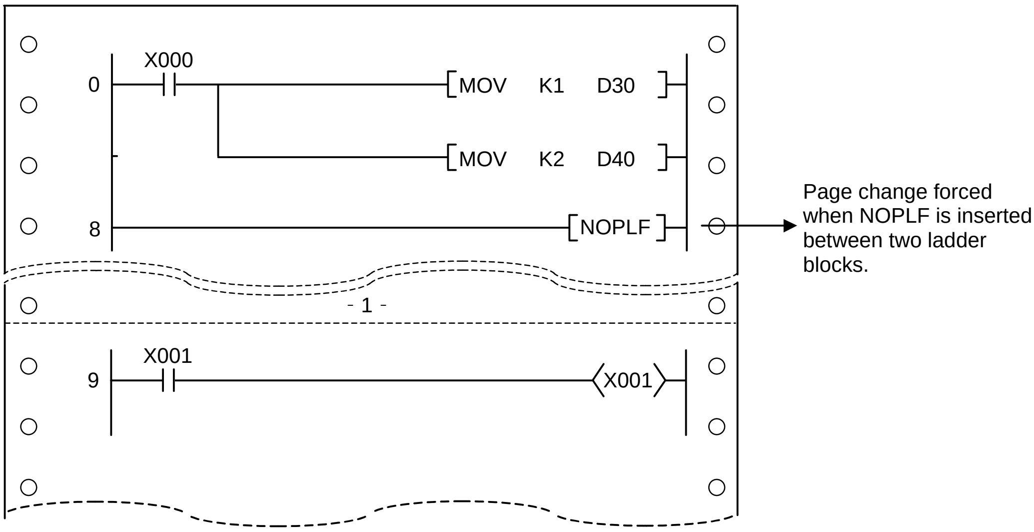
[Ladder Mode]



[List Mode]

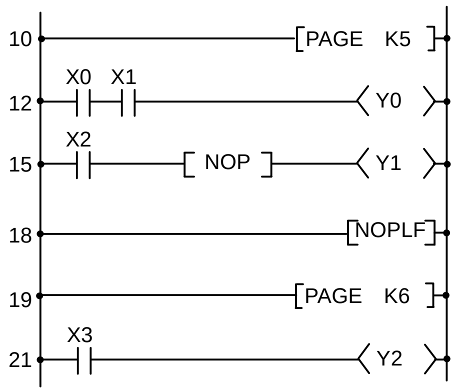
Steps	Inst.	Device
0	LD	X0
1	MOV	K1
		D30
4	NOPLF	
5	MOV	K2
		D40
8	NOPLF	
9	LD	X1
10	OUT	Y40
11	END	

• Printing the ladder will result in the following:



PAGE n

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
10	PAGE	K5
12	LD	X0
13	AND	X1
14	OUT	Y0
15	LD	X2
16	NOP	
17	OUT	Y1
18	NOPLF	
19	PAGE	K6
21	LD	X3
22	OUT	Y2
23		

8. Function Instructions

The function instruction includes the following types.

Type of Instructions	Meaning
Comparison operation instruction	Compare data to data
Arithmetic operation instruction	Adds, subtracts, multiplies, divides, increments, or decrements data with other data
Data conversion instruction	Coverts data types
Data transfer instruction	Transmits designated data
Program branch instruction	Program jumps
Logical operation instructions	Logical operations such as logical sum, logical product, etc.
Rotation instruction	Rotation/shift of designated data
Data processing instructions	Data searches, data processing such as decoding and encoding
Other instructions	Instructions which do not fall into the above categories
Special instructions for old machine type compatible	Instructions used to achieve the compatibility with sequence programs in the old machine type. (Can be used in the compatible instruction mode only.)

○ =, <>, >, <=, <, >= … 16-bit data comparisons

Compatible instruction mode	Usable instruction **=, >, <**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			

△ : S1 and S2 cannot be specified as constant at the same time.

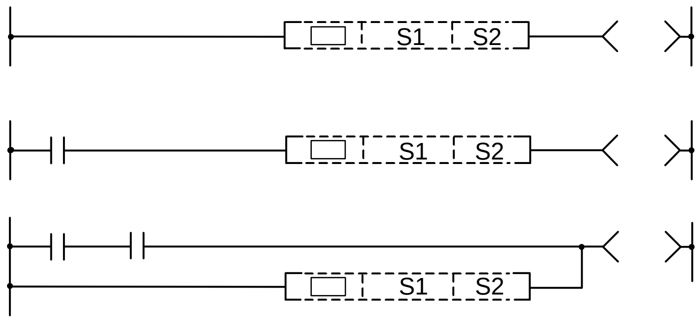
[Instruction Symbol] [Execution Condition]

□ indicates the signs =, <>, >, <=, <, >=

LD □

AND □

OR □



Set Data

Set Data	Meaning	Data Type
S1	Comparative data, or device No. where comparative data is stored	BIN 16 bits
S2		

Functions

(1) Treats BIN 16-bit data from device designated by (S1) and BIN 16-bit data from device designated by (S2) as an A contact, and performs comparison operation.

(2) The results of the comparison operations for the individual instructions are as follows:

Instruction Symbol in □	Condition	Comparison Operation Result	Instruction Symbol in □	Condition	Comparison Operation Result
=	S1＝S2	Continuity	=	S1≠S2	Non-continuity
<>	S1≠S2		<>	S1＝S2	
>	S1＞S2		>	S1≦S2	
<=	S1≦S2		<=	S1＞S2	
<	S1＜S2		<	S1≧S2	
>=	S1≧S2		>=	S1＜S2	

(3) In cases where hexadecimal constants have been designated by (S1) and (S2), or when a numerical value (8 to F) where the highest bit (b15) will be 1 has been designated, the value will be read as a negative BIN value number for purposes of the comparison.

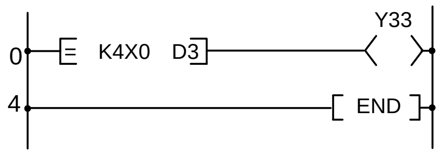
Operation Errors

(1) There are no operation errors associated with the =, <>, >, <=, <, or >= instructions.

Program Example

(1) The following program compares the data at X0 to XF with the data at D3, and turns Y33 ON if the data is identical.

[Ladder Mode]

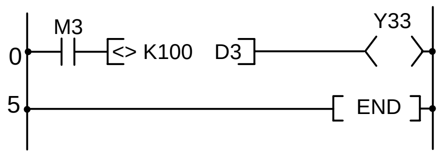


[List Mode]

Steps	Inst.	Device
0	LD=	K4X0
		D3
3	OUT	Y33
4	END	

(2) The following program compares BIN value K100 to the data at D3, and establishes continuity if the data in D3 is something other than 100.

[Ladder Mode]

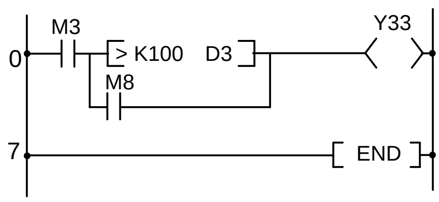


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND<>	K100
		D3
4	OUT	Y33
5	END	

(3) The following program compares BIN value K100 to the data at D3, and establishes continuity if the D3 data is less than 100.

[Ladder Mode]

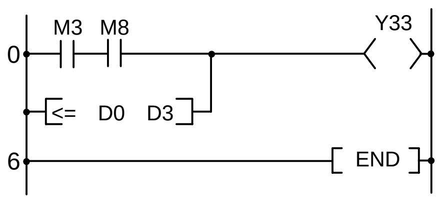


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	LD>	K100
		D3
4	OR	M8
5	ANB	
6	OUT	Y33
7	END	

(4) The following program compares the data in D0 and D3, and if the data in D0 is equal to or less than the data in D3, establishes continuity.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND	M8
2	OR<=	D0
		D3
5	OUT	Y33
6	END	

○ D=,D<>,D>,D<=,D<,D>= … 32-bit data comparison

Compatible instruction mode	Usable instruction: **D=, D>, D<**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			

△ : S1 and S2 cannot be specified as constant at the same time.

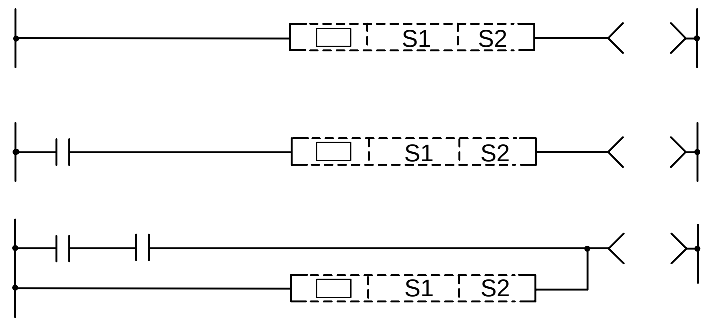
[Instruction Symbol] [Execution Condition]

☐ indicates the signs D=, D<>, D>, D<=, D<, D>=

LD ☐

AND ☐

OR ☐



Set Data

Set Data	Meaning	Data Type
S1	Comparative data, or device No. where comparative data is stored	BIN 32 bits
S2		

Functions

(1) Treats BIN 32-bit data from device designated by (S1) and BIN 32-bit data from device designated by (S2) as an A contact, and performs comparison operation.

(2) The results of the comparison operations for the individual instructions are as follows:

Instruction Symbol in ☐	Condition	Comparison Operation Result	Instruction Symbol in ☐	Condition	Comparison Operation Result
D=	S1＝S2	Continuity	D=	S1≠S2	Non-continuity
D<>	S1≠S2		D<>	S1＝S2	
D>	S1＞S2		D>	S1≦S2	
D<=	S1≦S2		D<=	S1＞S2	
D<	S1＜S2		D<	S1≧S2	
D>=	S1≧S2		D>=	S1＜S2	

(3) In cases where hexadecimal constants have been designated by (S1) and (S2), or when a numerical value (8 to F) where the highest bit (b31) will be 1 has been designated, the value will be read as a negative BIN value number for the purpose of the comparison.

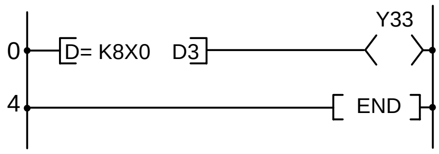
Operation Errors

(1) There are no operation errors associated with the D=, D<>, D>, D<=, D<, or D>= instructions.

Program Example

(1) The following program compares the data at X0 to XF with the data at D3 and D4, and turns Y33 ON if the data is identical.

[Ladder Mode]

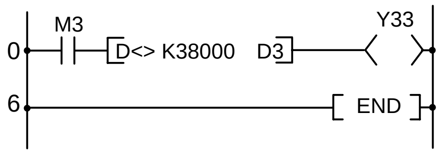


[List Mode]

Steps	Inst.	Device
0	LDD=	K8X0
		D3
3	OUT	Y33
4	END	

(2) The following program compares BIN value K38000 to the data at D3 and D4, and establishes continuity if the data in D3 and D4 is something other than 38000.

[Ladder Mode]

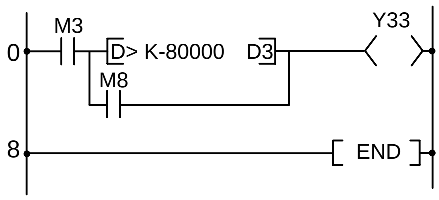


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	ANDD<>	K38000
		D3
5	OUT	Y33
6	END	

(3) The following program compares BIN value K-80000 to the data at D3 and D4, and establishes continuity if the data in D3 and D4 is less than –80000.

[Ladder Mode]

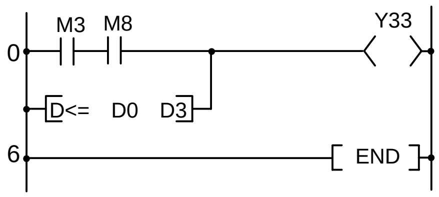


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	LDD>	K-80000
		D3
5	OR	M8
6	ANB	
7	OUT	Y33
8	END	

(4) The following program compares the data in D0 and D1 with the data in D3 and D4, and establishes continuity if the data in D0 and D1 is equal to or less than the data in D3 and D4.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND	M8
2	ORD<=	D0
		D3
5	OUT	Y33
6	END	

○ +, +P, -, -P … BIN 16-bit addition and subtraction operations
(Device at storage destination: Independent type)

Compatible instruction mode	Usable instruction: **+, -**																				
Set Data	Usable Devices																			Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1											○	○	○	○	○	○		○	△	△			
S2											○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

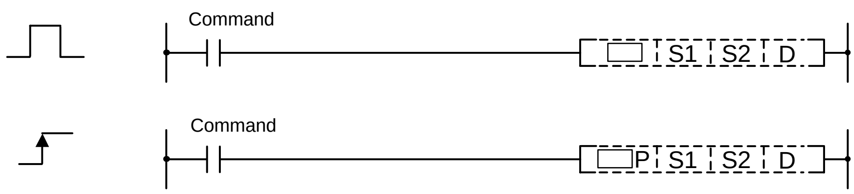
△ : S1 and S2 cannot be specified as constant at the same time.

[Instruction Symbol] [Execution Condition]

▭ indicates the signs + / -

+, -

+P, -P



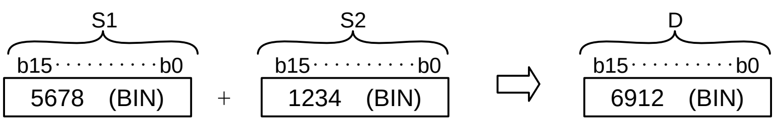
Set Data

Set Data	Meaning	Data Type
S1	Data to be added to or subtracted from, or the first number of the device storing such data	BIN 16 bits
S2	Addition or subtraction data, or first number of device storing addition or subtraction data	
D	First number of device storing addition or subtraction data	

Functions

+

(1) Adds 16-bit BIN data designated by (S1) to 16-bit BIN data designated by (S2) and stores at the device designated by (D).



(2) Values for S1, S2, and D can be designated between –32768 and 32767 (BIN 16 bits).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b15).

b15	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result:
The carry flag in this case does not go ON.
- K32767+K2 → K-32767 ---- A negative value is generated if b15 is 1.
(H7FFF) (H0002) (H8001)
- K-32768+K-2 → K32766 ---- A positive value is generated if b15 is 0.
(H8000) (HFFFE) (H7FFE)

-

(1) Subtracts 16-bit BIN data designated by (S1) from 16-bit BIN data designated by (S2) and stores the result of the subtraction at the device designated by (D).

S1		S2		D
b15 ·········· b0		b15 ·········· b0		b15 ·········· b0
5678 (BIN)	−	1234 (BIN)	⇨	4444 (BIN)

(2) Values for S1, S2, and D can be designated between –32768 and 32767 (BIN 16 bits).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b15).

b15	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result:
The carry flag in this case does not go ON.
- K-32768-K2 → K32766 ---- A positive value is generated if b15 is 0.
(H8000) (H0002) (H7FFE)
- K32767-K-2 → K-32767 ---- A negative value is generated if b15 is 1.
(H7FFF) (HFFFE) (H8001)

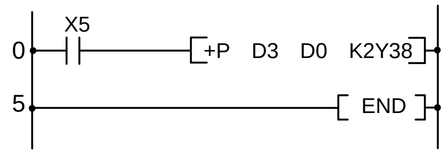
Operation Errors

(1) There are no operation errors associated with the +(P) or -(P) instructions.

Program Example

(1) The following program adds the contents of D3 and the contents of D0 when X5 goes ON, and outputs result to Y38 through 3F.

[Ladder Mode]

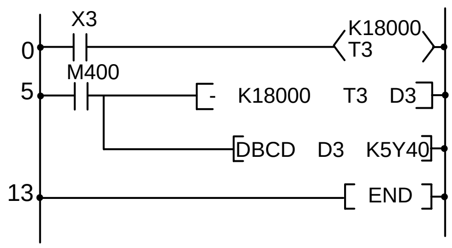


[List Mode]

Steps	Inst.	Device
0	LD	X5
1	+P	D3
		D0
		K2Y38
5	END	

(2) The following program outputs the difference between the set value for timer T3 and its present value to Y40 to 53 by BCD.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X3
1	OUT	T3
		K18000
5	LD	M400
6	-	K18000
		T3
		D3
10	DBCD	D3
		K5Y40
13	END	

○ +, +P, -, -P … BIN 16-bit addition and subtraction operations (Device at storage destination: Shared type)

Compatible instruction mode	Not available

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

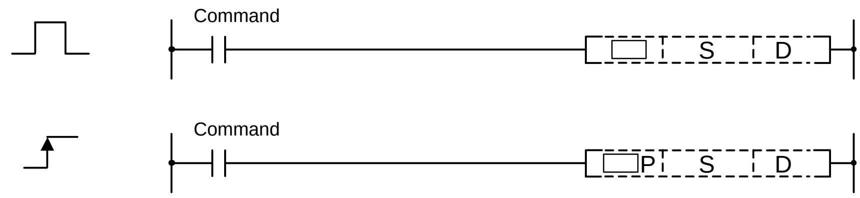
This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

[Instruction Symbol] [Execution Condition]

☐ indicates the signs + / -

+, -

+P, -P



Set Data

Set Data	Meaning	Data Type
S	Addition or subtraction data, or first number of device storing addition or subtraction data	BIN 16 bits
D	First number of device storing data to be added to or subtracted from (Addition or subtraction data is stored in this device.)	

Functions

+

(1) Adds 16 bit BIN data designated by (D) to 16 bit BIN data designated by (S), and stores the result of the addition at the device designated by (D).

D: b15··········b0 5678 (BIN) + S: b15··········b0 1234 (BIN) ⇒ D: b15··········b0 6912 (BIN)

(2) The value for (S) and (D) can be designated at between –32768 and 32767 (BIN 16 bits).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b15).

b15	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result. The carry flag in this case does not go ON.

- K32767+K2 → K-32767 ---- A negative value is generated if b15 is 1.
  (H7FFF) (H0002) (H8001)
- K-32768+K-2 → K32766 ---- A positive value is generated if b15 is 0.
  (H8000) (HFFFE) (7FFE)

-

(1) Subtracts 16-bit BIN data designated by (D) from 16-bit BIN data designated by (S) and stores the result of the subtraction at the device designated by (D).

D		S		D
b15 · · · · · · · · · · b0		b15 · · · · · · · · · · b0		b15 · · · · · · · · · · b0
5678 (BIN)	–	1234 (BIN)	⇨	4444 (BIN)

(2) The values for (S) and (D) can be designated at between –32768 and 32767 (BIN 16 bits).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b15).

b15	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result. The carry flag in this case does not go ON.
- K-32768-K2 → K32766 ---- A positive value is generated if b15 is 0.
  (H8000) (H0002) (H7FFE)
- K32767-K-2 → K-32767 ---- A negative value is generated if b15 is 1.
  (H7FFF) (HFFFE) (H8001)

Operation Errors

(1) There are no operation errors associated with the +(P) or -(P) instructions.

○ **D+, D+P, D-, D-P … BIN 32-bit addition and subtraction operations (Device at storage destination: Independent type)**

Compatible instruction mode	Usable instruction: **D+, D-**

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1											○	○	○	○	○	○		○	△	△			
S2											○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

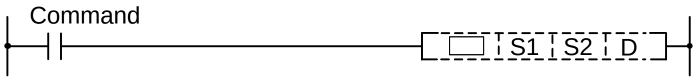
Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

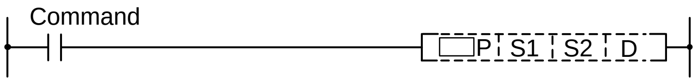
△ : S1 and S2 cannot be specified as constant at the same time.

[Instruction Symbol] [Execution Condition] ☐ indicates the signs D+ / D-

D+, D-



D+P, D-P



Set Data

Set Data	Meaning	Data Type
S1	Data to be added to or subtracted from, or the first number of the device storing such data	BIN 32 bits
S2	Addition or subtraction data, or number of device storing addition or subtraction data	
D	First number of device storing addition or subtraction data	

Functions

D+

(1) Adds 32-bit BIN data designated by (S1) to 32-bit BIN data designated by (S2), and stores the result of the addition at the device designated by (D).

S1+1 / S1 (b31 · · · b16 b15 · · · b0)		S2+1 / S2 (b31 · · · b16 b15 · · · b0)		D+1 / D (b31 · · · b16 b15 · · · b0)
567890 (BIN)	+	123456 (BIN)	⇨	691346 (BIN)

(2) The values for (S1), (S2) and (D) can be designated at between –2147483648 and 2147483647 (BIN 32 bits).

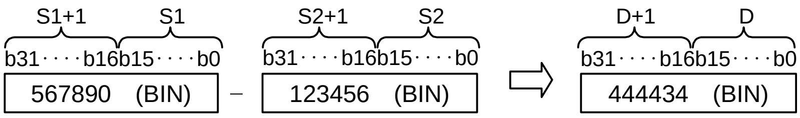
(3) The judgment of whether data is positive or negative is made by the most significant bit (b31).

b31	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result. The carry flag in this case does not go ON.
- K2147483647+K2 → K-2147483647 ---- A negative value is generated if b31 is 1.
  (H7FFFFFFF) (H0002) (H80000001)
- K-2147483648+K-2 → K2147483646 ---- A positive value is generated if b31 is 0.
  (H80000000) (HFFFE) (H7FFFFFFE)

D-

(1) Subtracts 32-bit BIN data designated by (S1) from 32-bit BIN data designated by (S2), and stores the result of the subtraction at the device designated by (D).



(2) The values for (S1), (S2) and (D) can be designated at between –2147483648 and 2147483647 (BIN 32 bits).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b31).

b31	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result. The carry flag in this case does not go ON.
- K-2147483648-K2 → K2147483646 ---- A positive value is generated if b31 is 0.
  (H80000000) (H0002) (H7FFFFFFE)
- K2147483647-K-2 → K-2147483647 ---- A negative value is generated if b31 is 1.
  (H7FFFFFFF) (HFFFE) (H80000001)

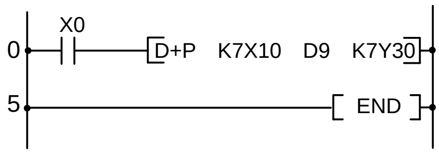
Operation Errors

(1) There are no operation errors associated with the D+(P) or D-(P) instructions.

Program Example

(1) The following program adds 28-bit data from X10 to X2B to the data at D9 and D10 when X0 goes ON, and outputs the result of the operation to Y30 to Y4B.

[Ladder Mode]

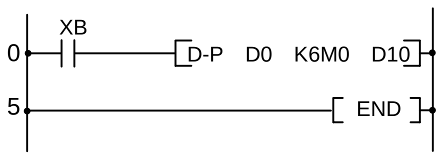


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	D+P	K7X10
		D9
		K7Y30
5	END	

(2) The following program subtracts the data from M0 to M23 from the data at D0 and D1 when XB goes ON, and stores the result at D10 and D11.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0B
1	D-P	D0
		K6M0
		D10
5	END	

○ D+, D+P, D-, D-P … BIN 32-bit addition and subtraction operations (Device at storage destination: Shared type)

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

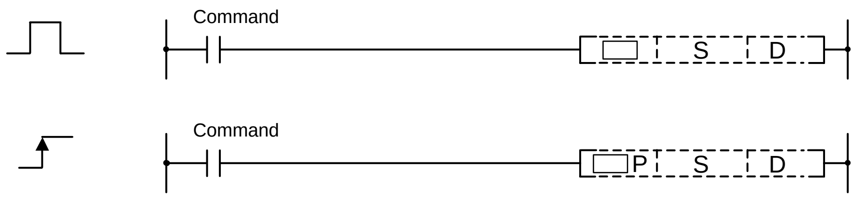
This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

[Instruction Symbol] [Execution Condition]

▭ indicates the signs D+ / D-

D+, D-

D+P, D-P



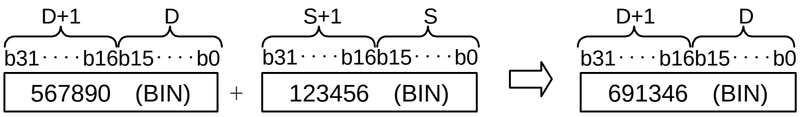
Set Data

Set Data	Meaning	Data Type
S	Head No. of the addition/subtraction data or device in which addition or subtraction data is stored.	BIN 32 bits
D	Head No. of the device in which augend/dividend data is stored. (Addition result is stored in this device.)	

Functions

D+

(1) Adds 32-bit BIN data designated by (D) to 32-bit BIN data designated by (S) and stores at the device designated by (D).



(2) The values for (S) and (D) can be designated at between –2147483648 and 2147483647 (BIN 32 bits).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b31).

b31	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result.
The carry flag in this case does not go ON.
- K2147483647+K2 → K-2147483647 ---- A negative value is generated if b31 is 1.
(H7FFFFFFF) (H0002) (H80000001)
- K-2147483648+K-2 → K2147483646 ---- A positive value is generated if b31 is 0.
(H80000000) (HFFFE) (H7FFFFFFE)

D-

(1) Subtracts 32-bit data designated by (D) from 32-bit data designated by (S) and stores the result of the subtraction at the device designated by (D).

D+1 / D: b31 · · · · b16 b15 · · · · b0		S+1 / S: b31 · · · · b16 b15 · · · · b0		D+1 / D: b31 · · · · b16 b15 · · · · b0
567890 (BIN)	–	123456 (BIN)	⇨	444434 (BIN)

(2) The values for (S) and (D) can be designated at between –2147483648 and 2147483647 (BIN 32 bit).

(3) The judgment of whether data is positive or negative is made by the most significant bit (b31).

b31	Judgment (+/-)
0	Positive
1	Negative

(4) The following will happen when an underflow or overflow is generated in an operation result. The carry flag in this case does not go ON.
- K-2147483648-K2 → K2147483646 ---- A positive value is generated if b31 is 0.
  (H80000000) (H0002) (H7FFFFFFE)
- K2147483647-K-2 → K-2147483647 ---- A negative value is generated if b31 is 1.
  (H7FFFFFFF) (HFFFE) (H80000001)

Operation Errors

(1) There are no operation errors associated with the D+(P) or D-(P) instructions.

○ ***, *P, /, /P … BIN 16-bit multiplication and division operations**

Compatible instruction mode	Usable instruction: ***, /**

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1											○	○	○	○	○	○		○	△	△			
S2											○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

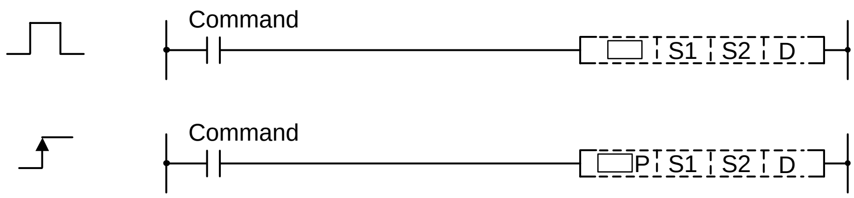
△ : S1 and S2 cannot be specified as constant at the same time.

[Instruction Symbol] [Execution Condition]

▭ indicates the signs * / /

*, /

*P, /P



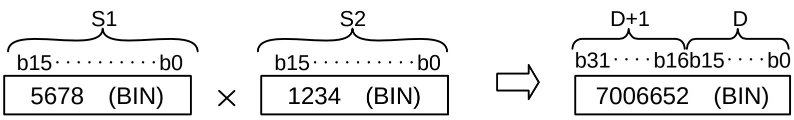
Set Data

Set Data	Meaning	Data Type
S1	Data to be added to or subtracted from, or the first number of the device storing such data	BIN 16 bits
S2	Addition or subtraction data, or first number of device storing addition or subtraction data	
D	First number of device storing addition or subtraction data	BIN 32 bits

Functions

*

(1) Multiplies BIN 16-bit data designated by (S1) and BIN 16-bit data designated by (S2), and stores the multiplication result in the device designated by (D).



(2) If (D) is a bit device, designation is made from the lower bits.
Example K1 --- Lower 4 bits (b0 to 3)
K4 --- Lower 16 bits (b0 to 15)
K8 --- 32 bits (b0 to 31)

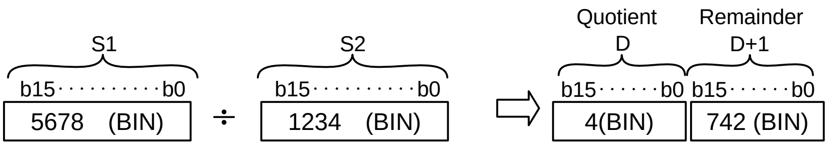
(3) The values for (S1) and (S2) can be designated at between –32768 and 32767 (BIN 16 bits).

(4) Judgments whether (S1),(S2), and (D) are positive or negative are made on the basis of the most significant bit (b15 for (S1) and (S2), and b31 for (D)).

b15/b31	Judgment (+/-)
0	Positive
1	Negative

/

(1) Divides BIN 16-bit data designated by (S1) and BIN 16-bit data designated by (S2), and stores the division result in the device designated by (D).



(2) If a word device has been used, the result of the division operation is stored as 32 bits, and both the quotient and remainder are stored.
Quotient ---- Stored at the lower 16 bits
Remainder --- Stored at the higher 16 bits

(3) The values for (S1) and (S2) can be designated at between –32768 and 32767 (BIN 16 bits).

(4) Judgment whether values for (S1), (S2), and (D) are positive or negative is made on the basis of the most significant bit (b15 for (S1) and (S2), and b31 for (D)).

b15/b31	Judgment (+/-)
0	Positive
1	Negative

(5) When divisor S2 is 0, no operation is carried out.

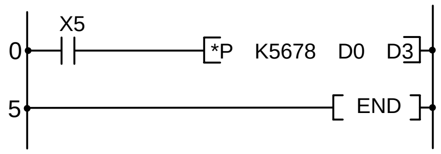
Operation Errors

(1) There are no operation errors associated with the *, *P, /, or /P instructions.

Program Example

(1) When X5 is turned ON, "5678" of BIN and multiplication result of D0 are stored in D3 and D4.

[Ladder Mode]

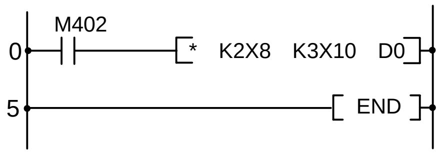


[List Mode]

Steps	Inst.	Device
0	LD	X5
1	*P	K5678
		D0
		D3
5	END	

(2) Multiplication result of BIN data of X8 to XF and BIN data of X10 to X1B is output to D0 and D1.

[Ladder Mode]

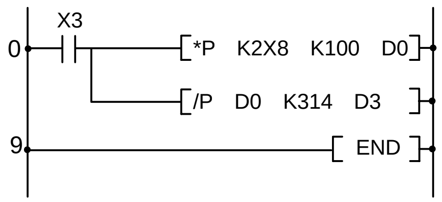


[List Mode]

Steps	Inst.	Device
0	LD	M402
1	*	K2X8
		K3X10
		D0
5	END	

(3) When X3 is turned ON, the data of X8 to XF is divided by 3.14 and the result is output to D3 and D4.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X3
1	*P	K2X8
		K100
		D0
5	/P	D0
		K314
		D3
9	END	

○ D*, D*P, D/, D/P … BIN 32-bit multiplication and division operations

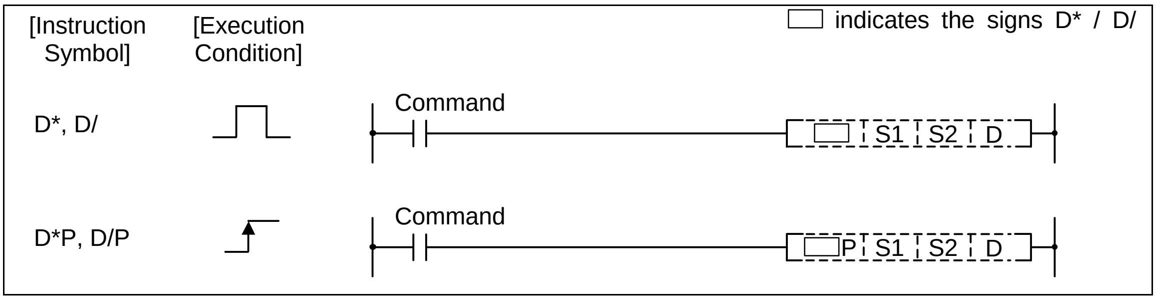
Compatible instruction mode	Usable instruction: **D*, D/**

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1											○	○	○	○	○	○		○	△	△			
S2											○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

△ : S1 and S2 cannot be specified as constant at the same time.



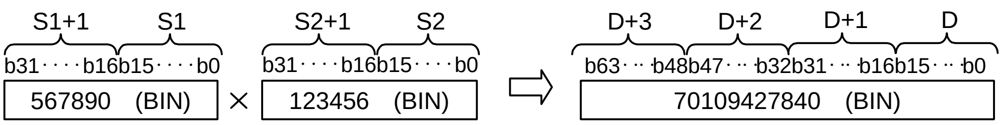
Set Data

Set Data	Meaning	Data Type
S1	Data to be added to or subtracted from, or the first number of the device storing such data	BIN 32 bits
S2	Addition or subtraction data, or first number of device storing addition or subtraction data	
D	First number of device storing addition or subtraction data	

Functions

D*

(1) Multiplies BIN 32-bit data designated by (S1) and BIN 32-bit data designated by (S2), and stores the multiplication result in the device designated by (D).



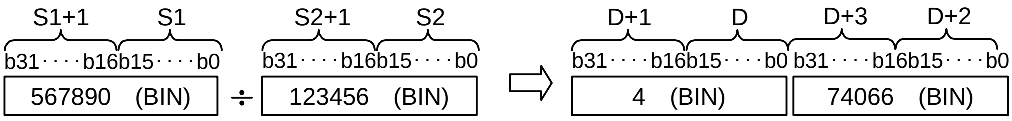
(2) The values for (S1) and (S2) can be designated at between –2147483648 and 2147483647 (BIN 32 bits).

(3) Judgment whether values for (S1), (S2), and (D) are positive or negative are made on the basis of the most significant bit (b31 for (S1) and (S2), and b63 for (D)).

b31/b63	Judgment (+/-)
0	Positive
1	Negative

D/

(1) Divides BIN 32-bit data designated by (S1) and BIN 32-bit data designated by (S2), and stores the division result in the device designated by (D).



(2) If a word device has been used, the result of the division operation is stored as 64 bits, and both the quotient and remainder are stored.
Quotient ---- Stored at the lower 32 bits
Remainder --- Stored at the upper 32 bits

(3) The values for (S1) and (S2) can be designated at between –2147483648 and 2147483647 (BIN 32 bits).

(4) Judgment whether values for (S1), (S2), (D), and (D)+2 are positive or negative is made on the basis of the most significant bit (b31).
(A sign is used with both the quotient and the remainder)

b31	Judgment (+/-)
0	Positive
1	Negative

(5) When divisor S2 is 0, no operation is carried out.

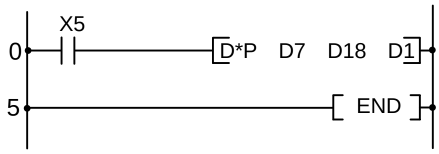
Operation Errors

(1) There are no operation errors associated with the D*, D*P, D/, or D/P instructions.

Program Example

(1) When X5 is turned ON, multiplication result of BIN data of D7 and D8 and BIN data of D18 and D19 is stored in D1 to D4.

[Ladder Mode]

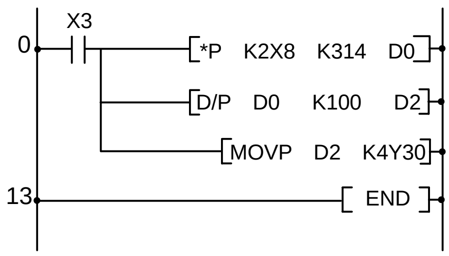


[List Mode]

Steps	Inst.	Device
0	LD	X5
1	D*P	D7
		D18
		D1
5	END	

(2) When X3 is turned ON, the data of X8 to XF is multiplied by 3.14 and the result is output to Y30 to Y3F.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X3
1	*P	K2X8
		K314
		D0
5	D/P	D0
		K100
		D2
10	MOVP	D2
		K4Y30
13	END	

○ B+, B+P, B-, B-P … BCD 4-digit addition and subtraction operations

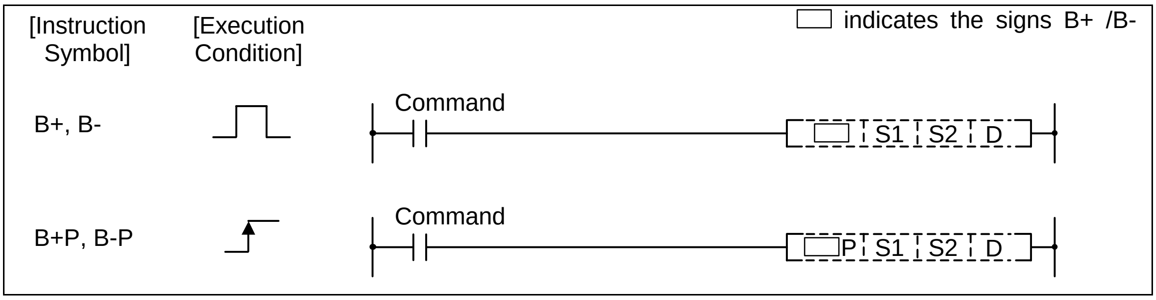
Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

△ : S1 and S2 cannot be specified as constant at the same time.



Set Data

Set Data	Meaning	Data Type
S1	Data to be added to or subtracted from, or the first number of the device storing such data	BCD 4-digit
S2	Addition or subtraction data, or the head No. of device storing addition or subtraction data	
D	First number of device storing addition or subtraction data	

Functions

B+

(1) Adds the BCD 4-digit data designated by "S1" and the BCD 4-digit data designated by "S2", and stores the result of the addition at the device designated by "D".

S1: 5 6 7 8 + S2: 1 2 3 4 ⇒ D: 6 9 1 2

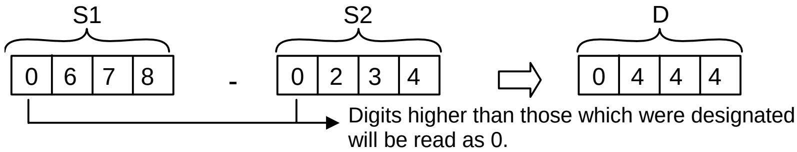
(2) The values for "S1", "S2", and "D" can be specified between 0 and 9999 (BCD 4-digit).

(3) If the result of the addition operation exceeds 9999, the higher bits are ignored. The carry flag in this case does not go ON.

6 4 3 2 + 3 5 8 3 ⇒ 0 0 1 5

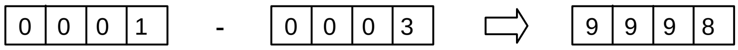
B-

(1) Subtracts the BCD 4-digit data designated by "S1" and the BCD 4-digit data designated by "S2", and stores the result of the subtraction at the device designated by "D".



(2) The values for "S1", "S2", and "D" can be specified between 0 and 9999 (BCD 4-digit).

(3) The following will result if an underflow is generated by the subtraction operation:
The carry flag in this case does not go ON.



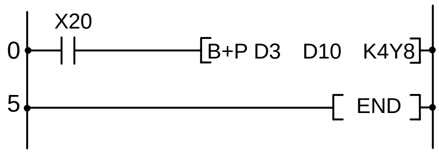
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
• The "S1", "S2" or "D" BCD data is outside the 0 to 9999 range. (Error code: 80)

Program Example

(1) The following program adds the D3 BCD data and the Z1 BCD data when X20 goes ON, and outputs the result to Y8 to Y17.

[Ladder Mode]

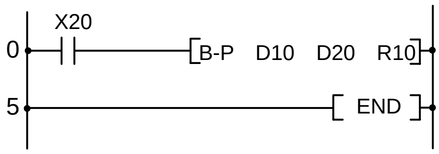


[List Mode]

Steps	Inst.	Device
0	LD	X20
1	B+P	D3
		D10
		K4Y8
5	END	

(2) The following program subtracts the BCD data at D20 from the BCD data at D10 when X20 goes ON, and stores the result at R10.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X20
1	B-P	D10
		D20
		R10
5	END	

○ B*, B*P, B/, B/P … BCD 4-digit multiplication and division operations

Compatible instruction mode	Not available

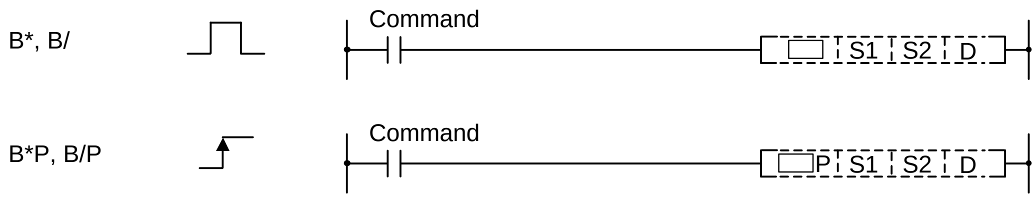
Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
D											○	○	○	○	○	○		○					

This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.
△ : S1 and S2 cannot be specified as constant at the same time.

[Instruction Symbol] [Execution Condition]

☐ indicates the signs B* /B/



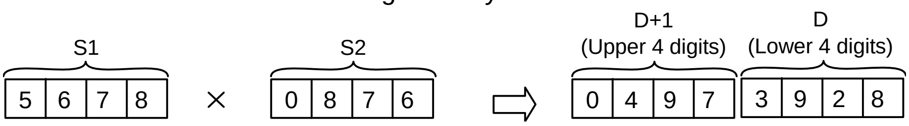
Set Data

Set Data	Meaning	Data Type
S1	Data to be added to or subtracted from, or the first number of the device storing such data	BCD 4-digit
S2	Addition or subtraction data, or the head No. of device storing addition or subtraction data	BCD 4-digit
D	First number of device storing addition or subtraction data	BCD 8-digit

Functions

B*

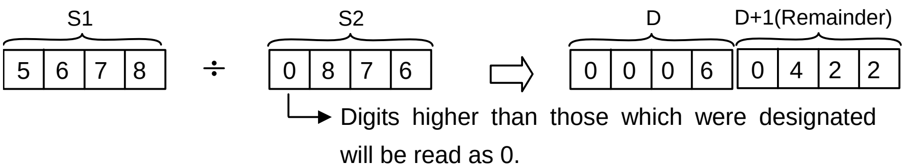
(1) Multiplies BCD 4-digit data designated by "S1" and BCD 4-digit data designated by "S2", and stores the result in the device designated by "D".



(2) Values for "S1" and "S2" can be specified between 0 and 9999 (BCD 4 digits).

B/

(1) Divides BCD 4-digit data designated by "S1" and BCD 4-digit data designated by "S2", and stores the result in the device designated by "D".



(2) 0 to 9999 (BCD 4-digit) for S1, and 1 to 9999 (BCD 4-digit) for S2 can be specified.

(3) Division result (quotient and remainder) is stored by using 32 bits.
Quotient (BCD 4 digits) .... Stored at the lower 16 bits
Remainder (BCD 4 digits) .... Stored at the upper 16 bits

(4) When divisor S2 is 0, no operation is carried out.

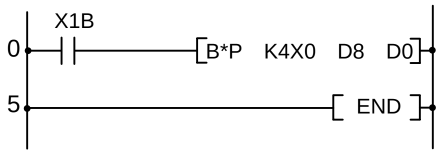
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
• The BCD data of "S1","S2" or "D" is outside the 0 to 9999 range. (Error code: 80)

Program Example

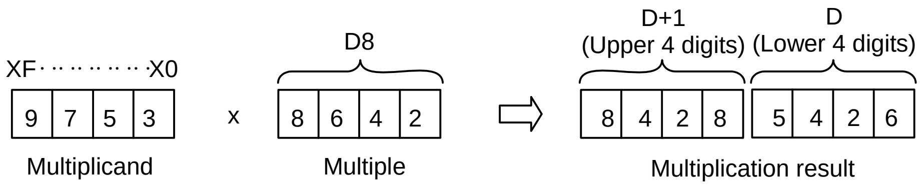
(1) The following program multiplies the BCD data at X0 to XF and the BCD data at D8 when X1B goes ON, and stores the result at D0 and D1.

[Ladder Mode]



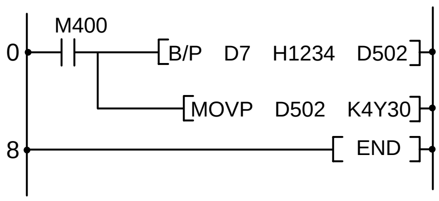
[List Mode]

Steps	Inst.	Device
0	LD	X1B
1	B*P	K4X0
		D8
		D0
5	END	



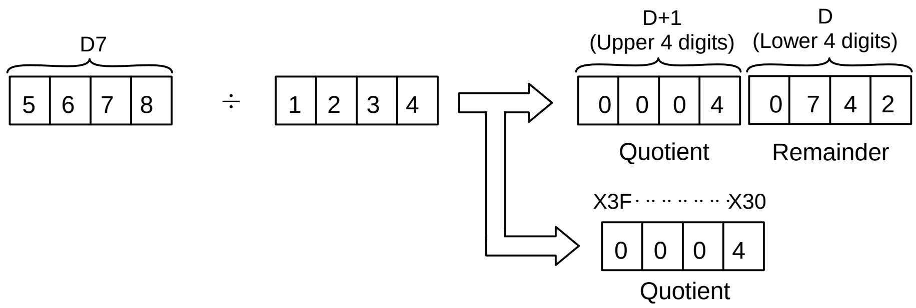
(2) The following program divides the BCD data D7 by the BCD data 1234, stores the result at D502 and D503, and at the same time outputs the quotient to Y30 to Y3F.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M400
1	B/P	D7
		H1234
		D502
5	MOVP	D502
		K4Y30
8	END	



○ INC,INCP,DEC,DECP … Incrementing and decrementing 16-bit BIN data

Compatible instruction mode	Usable instruction: **INC, DEC**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					

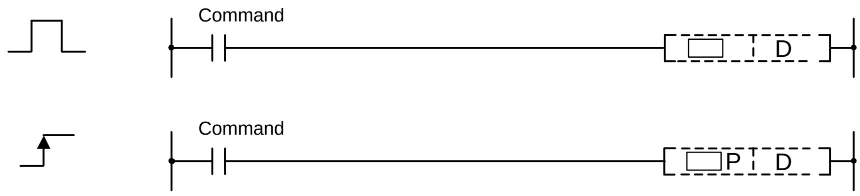
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	

[Instruction Symbol] [Execution Condition] ▭ indicates the signs INC/DEC

INC, DEC

INCP, DECP



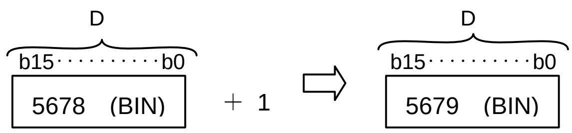
Set Data

Set Data	Meaning	Data Type
D	Head number of device conducting INC (add 1) or DEC (subtract 1) operation	BIN 16 bits

Functions

INC

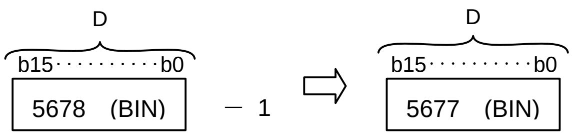
(1) Adds 1 to device designated by "D" (16-bit data).



(2) If the contents of the device designated by "D" were 32767, and the INC or INCP instruction were executed on that device, the value -32768 would be stored in the device designated by "D".

DEC

(1) Subtracts 1 from device designated by "D" (16-bit data).



(2) If the contents of the device designated by "D" were 0, and the DEC or DECP instruction were executed on that device, the value -1 would be stored in the device designated by "D".

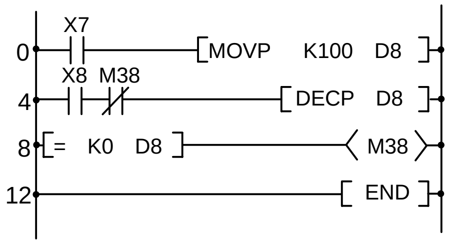
Operation Errors

(1) There are no operation errors associated with the INC(P) or DEC(P) instructions.

Program Example

(1) The following is a down counter program.

[Ladder Mode]



Transfers the value of 100 to D8 when X7 is ON

When M38 is OFF, X8 goes from OFF to ON, and 1 is decremented from D8.

At D8=0, M38 goes ON.

[List Mode]

Steps	Inst.	Device
0	LD	X7
1	MOVP	K100
		D8
4	LD	X8
5	ANI	M38
6	DECP	D8
8	LD=	K0
		D8
11	OUT	M38
12	END	

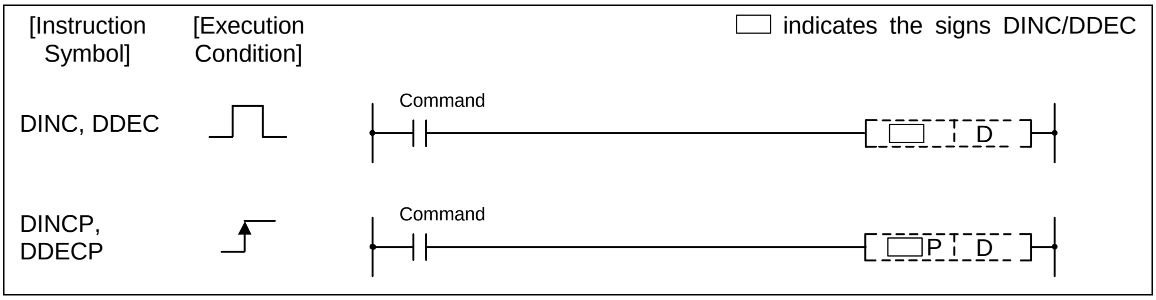
○ DINC,DINCP,DDEC,DDECP … Incrementing and decrementing 32-bit BIN data

Compatible instruction mode	Usable instruction: **DINC, DDEC**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	



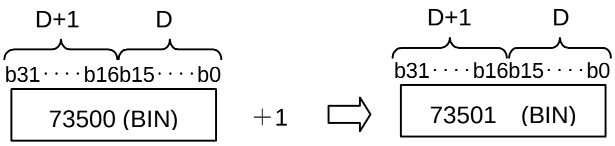
Set Data

Set Data	Meaning	Data Type
D	Head number of device what will execute the DINC (+1) or DDEC (-1) operation	BIN 32 bits

Functions

DINC

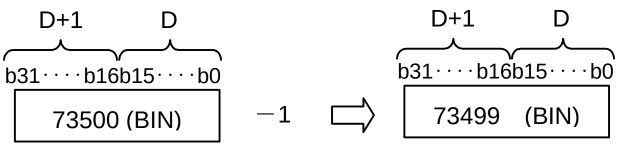
(1) Adds 1 to the device designated by "D" (32-bit data).



(2) If the contents of the device designated by "D" are 2147483647, and the DINC or DINCP instruction is executed, the value -2147483648 will be stored at the device designated by "D".

DDEC

(1) Subtracts 1 from the device designated by "D" (32-bit data).



(2) If the contents of the device designated by "D" are 0, and the DDEC or DDECP instruction is executed, the value -1 will be stored at the device designated by "D".

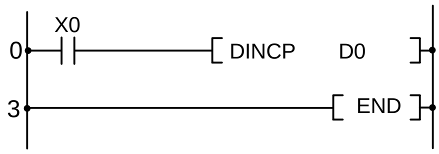
Operation Errors

(1) There are no operation errors associated with the DINC(P) or DDEC(P) instruction.

Program Example

(1) The following program adds 1 to the data at D0 and D1 when X0 is ON.

[Ladder Mode]

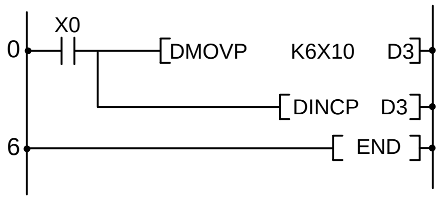


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	DINCP	D0
3	END	

(2) The following program adds 1 to the data set at X10 to X27 when X0 goes ON, and stores the result at D3 and D4.

[Ladder Mode]

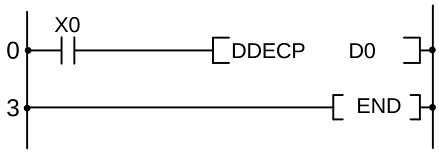


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	DMOVP	K6X10
		D3
4	DINCP	D3
6	END	

(3) The following program subtracts 1 from the data at D0 and D1 when X0 goes ON.

[Ladder Mode]

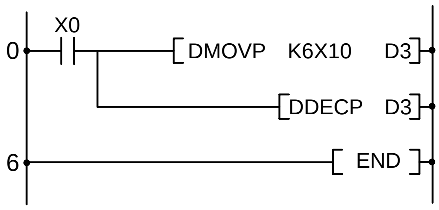


[List Mode]

Steps	Inst.	Device
0	LD	X0
1	DDECP	D0
3	END	

(4) The following program subtracts 1 from the data set at X10 to X27 when X0 goes ON, and stores the result at D3 and D4.

[Ladder Mode]



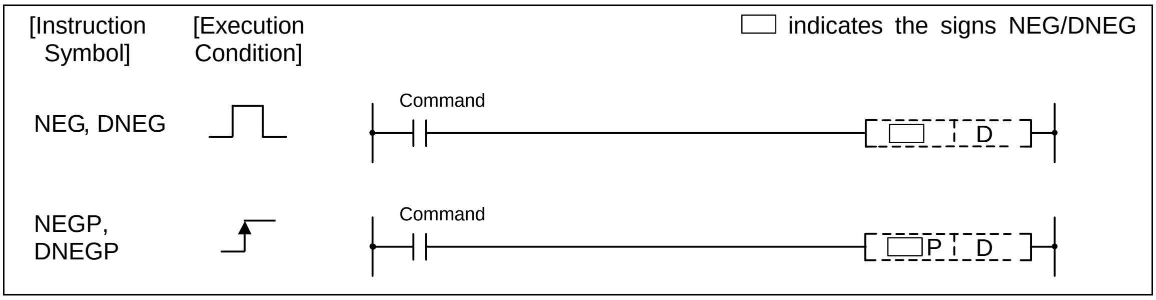
[List Mode]

Steps	Inst.	Device
0	LD	X0
1	DMOVP	K6X10
		D3
4	DDECP	D3
6	END	

○ NEG,NEGP,DNEG,DNEGP … Complement of 2 of BIN 16- and 32-bit data (sign reversal)

Compatible instruction mode	Usable instruction: **NEG**																						
Set Data	Usable Devices																					Digit desig- nation	Index
	Bit Devices										Word Devices								Consta nt		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit desig- nation	Index
	Bit Devices										Word Devices								Consta nt		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○				○	○	○	○		○				○	



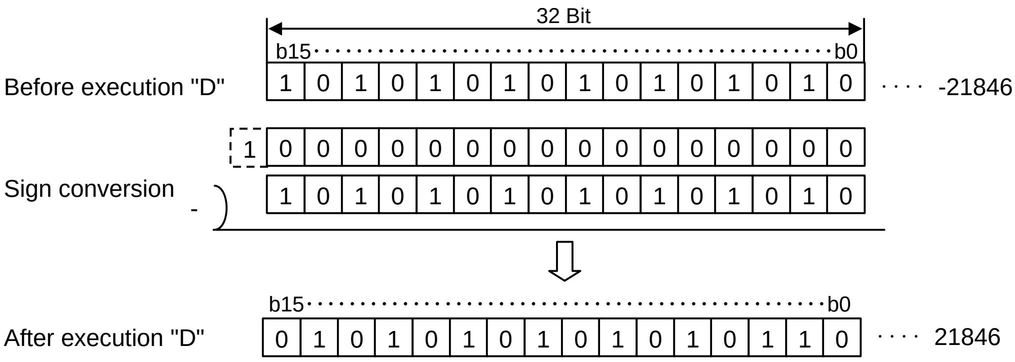
Set Data

Set Data	Meaning	Data Type
D	Head number of the device where data for the complement of 2 operation is stored.	BIN 16/32 bits

Functions

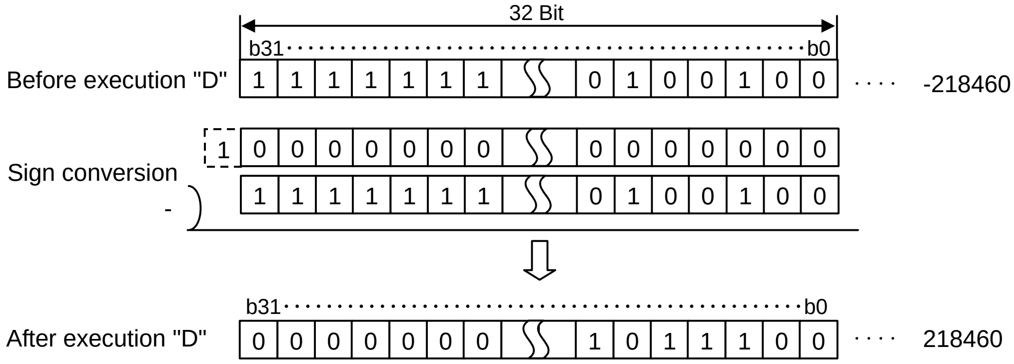
NEG

(1) Reverses the sign of the 16-bit device designated by "D" and stores at the device designated by "D".



(2) Used when reversing positive and negative signs.

DNEG (1) Reverses the sign of the 32-bit device designated by "D" and stores at the device designated by "D".



(2) Used when reversing positive and negative signs.

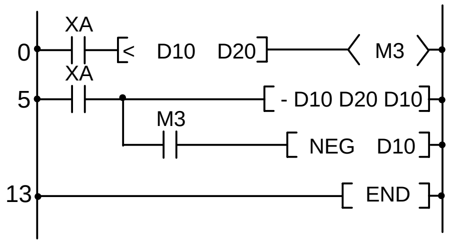
Operation Errors

(1) There are no operation errors associated with the NEG(P) or DNEG(P) instructions.

Program Example

(1) The following program calculates a total for the data at D10 through D20 when XA goes ON, and seeks an absolute value if the result is negative.

[Ladder Mode]



M3 goes ON when D10 is smaller than D20.

Subtracts D20 from D10.

Seeks an absolute value (complement of 2) when M3 is ON.

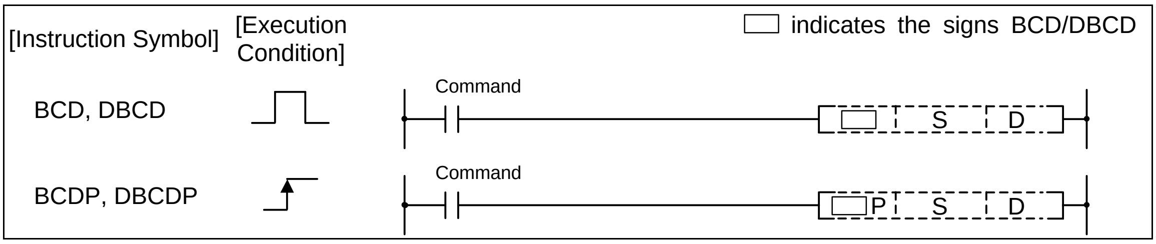
[List Mode]

Steps	Inst.	Device
0	LD	XA
1	AND<	D10
		D20
4	OUT	M3
5	LD	XA
6	-	D10
		D20
		D10
10	AND	M3
11	NEG	D10
13	END	

○ BCD,BCDP,DBCD,DBCDP … Conversion from BIN data to 4-digit and 8-digit BCD

Compatible instruction mode	Usable instruction: **BCD, DBCD**																						
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D											○	○	○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

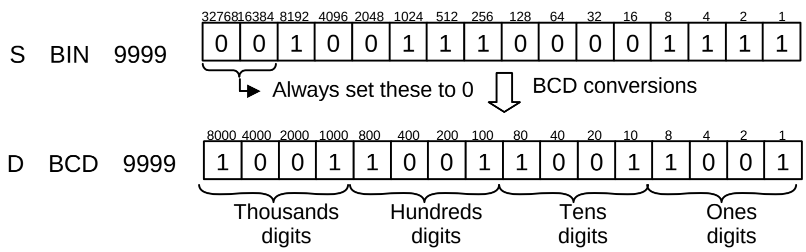


Set Data

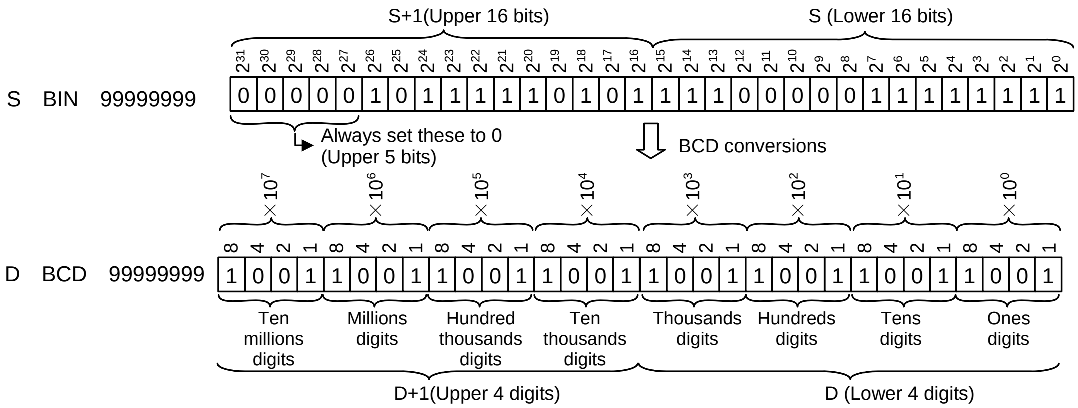
Set Data	Meaning	Data Type
S	Head number of the device where BIN data is stored	BIN 16/32 bits
D	Head number of the device that will store BCD data	BCD 4/8 digits

Functions

BCD (1) Converts BIN data (0 to 9999) at the device designated by "D" to BCD data, and stores it at the device designated by "D".



DBCD (1) Converts BIN data (0 to 99999999) at the device designated by "S" to BCD data, and transfers it at the device designated by "D".

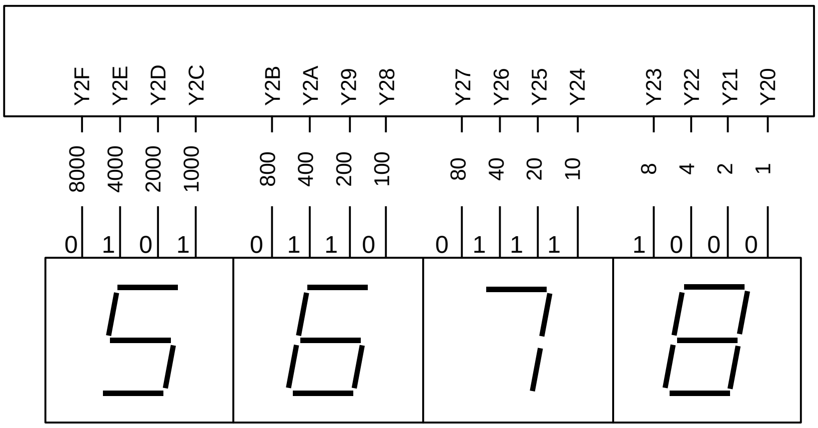


Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- The data at S was not in the 0 to 9999 range when the BCD instruction was issued. (Error code: 80)
- The data at "S"+1 and "S" was not in the 0 to 99999999 range when the DBCD instruction was issued. (Error code: 80)

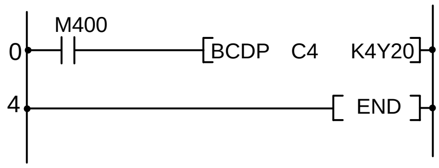
Program Example

(1) The following program outputs the present value of C4 from Y20 to Y2F to the BCD display device.



7-element display unit

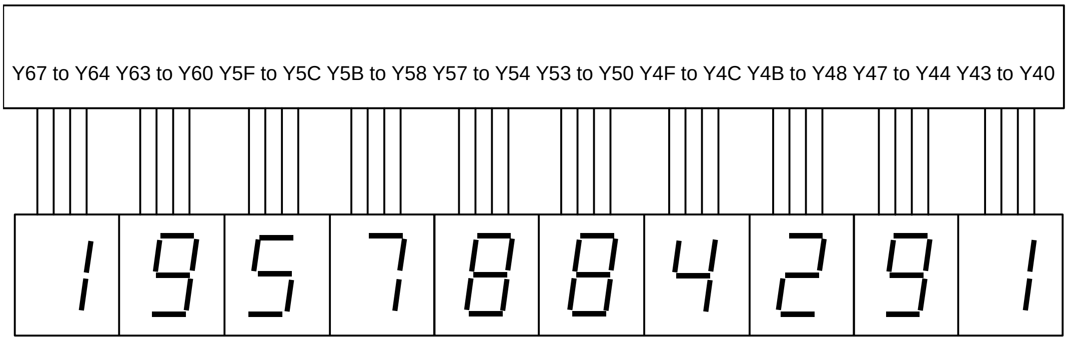
[Ladder Mode]



[List Mode]

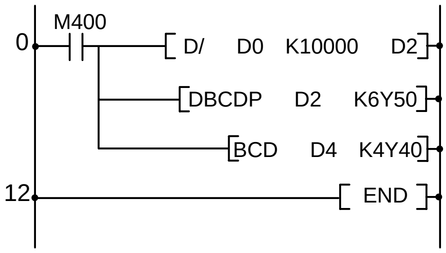
Steps	Inst.	Device
0	LD	M400
1	BCDP	C4
		K4Y20
4	END	

(2) The following program outputs bit data from D0 to D1 to Y40 to Y67.



7-element display unit

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M400
1	D/	D0
		K10000
		D2
6	DBCDP	D2
		K6Y50
9	BCD	D4
		K4Y40
12	END	

○ BIN,BINP,DBIN,DBINP … Conversion from BCD 4-digit and 8-digit data to BIN data

Compatible instruction mode	Usable instruction: **BIN, DBIN**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D											○	○	○	○	○	○		○					

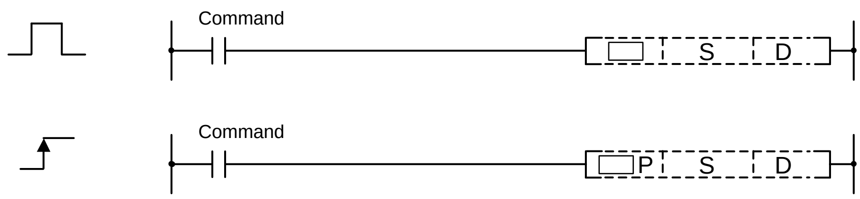
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol] [Execution Condition] ▭ indicates the signs BIN/DBIN

BIN, DBIN

BINP, DBINP



Set Data

Set Data	Meaning	Data Type
S	Head number of device storing BCD data	BCD 4/8 digit
D	Head number of device that will store BIN data	BIN 16/32 bits

Functions

BIN

(1) Converts BCD data (0 to 9999) at device designated by "D" to BIN data, and stores at the device designated by "D".

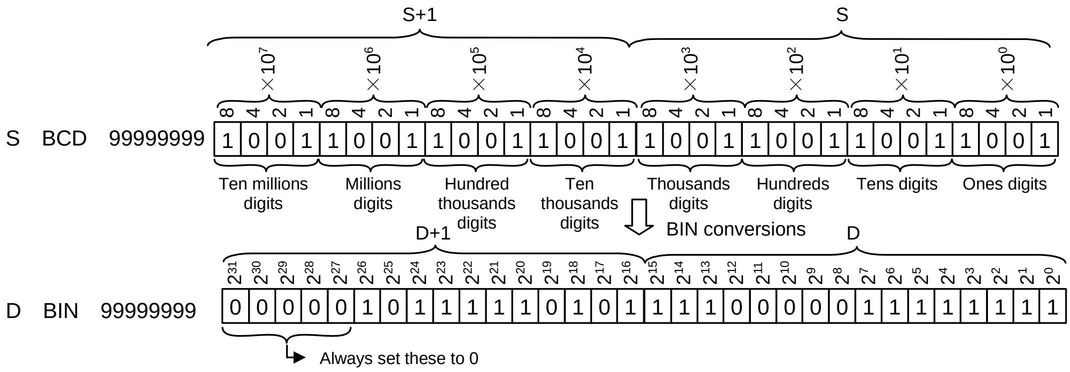
	8000	4000	2000	1000	800	400	200	100	80	40	20	10	8	4	2	1
S BCD 9999	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1
	Thousands digits				Hundreds digits				Tens digits				Ones digits			

⇩ BIN conversion

	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
D BIN 9999	0	0	1	0	0	1	1	1	0	0	0	0	1	1	1	1

Always set these to 0

DBIN (1) Converts BCD data (0 to 99999999) at device designated by "S" to BIN data, and stores at the device designated by "D".

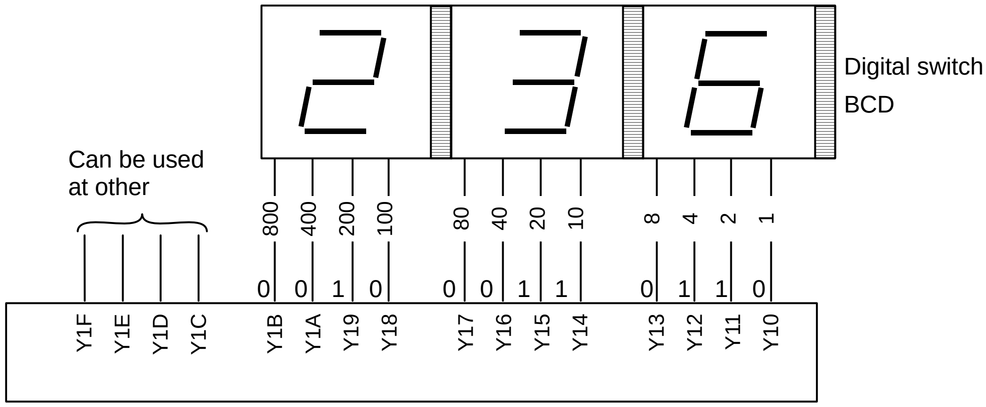


Operation Errors

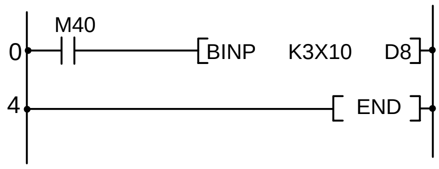
(1) In the following cases, an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- When values other than 0 to 9 are designated to any digits of "S". (Error code: 81)

Program Example

(1) The following program converts the BCD data at X10 to X1B to BIN when X8 is ON, and stores it at D8.



[Ladder Mode]

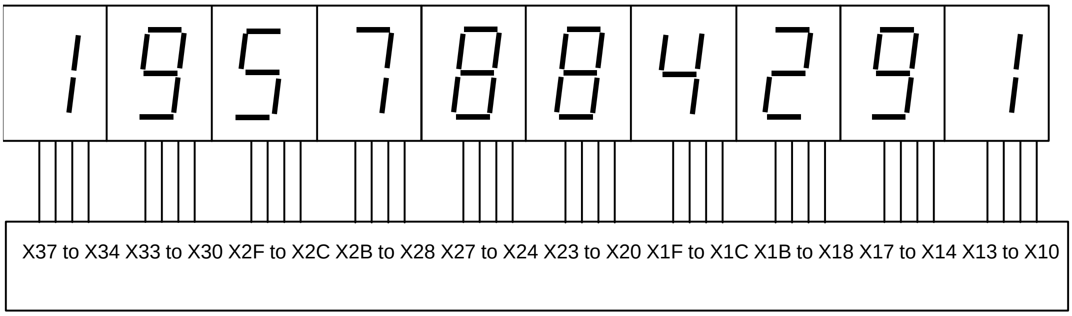


[List Mode]

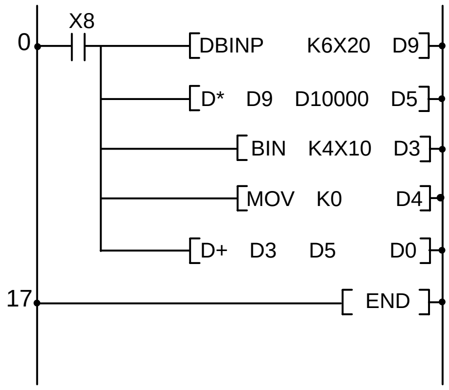
Steps	Inst.	Device
0	LD	M400
1	BINP	K3X10
		D8
4	END	

(2) The following program converts the BCD data at X10 to X37 to BIN when X8 is ON, and stores it at D0 and D1.

BCD digital switch



[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X8
1	DBINP	K6X20
		D9
4	D*	D9
		D10000
		D5
8	BIN	K4X10
		D3
11	MOV	K0
		D4
13	D+	D3
		D5
		D0
17	END	

○ MOV,MOVP,DMOV,DMOVP … 16-bit and 32-bit data transfers

Compatible instruction mode	Usable instruction: **MOV, DMOV**

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○	△1	○	○	○		○	△3
D		○	○	○	○	○	○		○		○	○	○	○	○	○	△2	○					

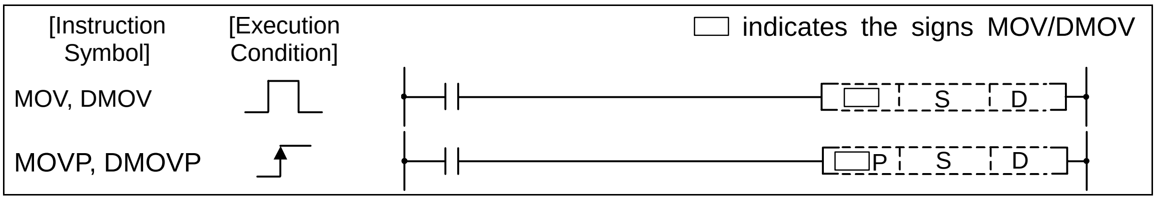
△1: Z cannot be used independently at the source side unless index qualification is given to the word device.

△2: Transferring to Z from bit device is not possible.

Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	△3
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			

△3: Index qualification is possible only with MOV(P)0 and is not possible with DMOV(P).



Set Data

Set Data	Meaning	Data Type
S	Transfer data, or number of device storing transfer data	BIN 16/32 bits
D	Number of device to store transferred data	

Functions

MOV (1) Transfers the 16-bit data from the device designated by "S" to the device designated by "D".

Prior to transfer "S" (b15 ··· b0)

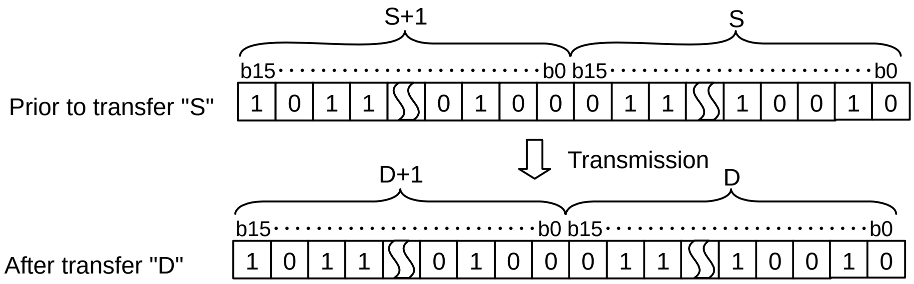
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

⇩ Transmission

After transfer "D" (b15 ··· b0)

1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0

DMOV (1) Transfers the 32-bit data from the device designated by "S" to the device designated by "D".



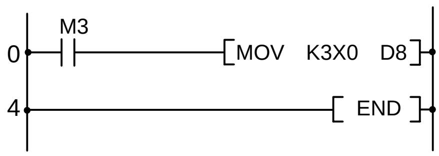
Operation Errors

(1) There are no operation errors associated with the MOV(P) or DMOV(P) instructions.

Program Example

(1) The following program stores input data from X0 to XB at D8.

[Ladder Mode]

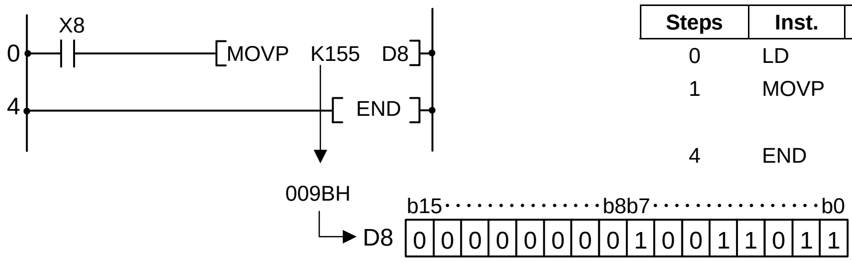


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	MOV	K3X0
		D8
4	END	

(2) The following program stores the constant K155 at D8 when X8 goes ON.

[Ladder Mode]

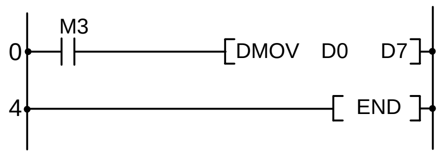


[List Mode]

Steps	Inst.	Device
0	LD	X8
1	MOVP	K155
		D8
4	END	

(3) The following program stores the data from D0 and D1 at D7 and D8.

[Ladder Mode]

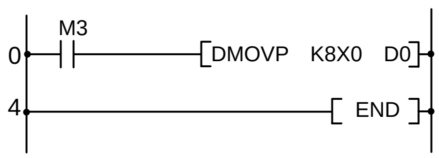


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	DMOV	D0
		D7
4	END	

(4) The following program stores the data from X0 to X1F at D0 and D1.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M3
1	DMOVP	K8X0
		D0
4	END	

○ CML,CMLP,DCML,DCMLP … 16-bit and 32-bit negation transfers

Compatible instruction mode	Not available

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

Set Data

Set Data	Meaning	Data Type
S	Data to be inverted, or number of device storing this data	BIN 16/32 bits
D	Number of device that will store results of inversion	

Functions

CML

(1) Inverts 16-bit data designated by S bit by bit, and transfers the result to the device designated by "D".

	b15															b0
Before execution "S"	1	0	1	1	0	1	0	0	0	1	1	1	0	0	1	0
⇩ Inversion																
After execution "D"	0	1	0	0	1	0	1	1	1	0	0	0	1	1	0	1

DCML

(1) Inverts 32-bit data designated by S bit by bit, and transfers the result to the device designated by "D".

	S+1 b15								S+1 b0	S b15								S b0
Before execution "S"	1	0	1	1	～	0	1	0	0	0	1	1	～	1	0	0	1	0
⇩ Inversion																		
	D+1 b15								D+1 b0	D b15								D b0
After execution "D"	0	1	0	0	～	1	0	1	1	1	0	0	～	0	1	1	0	1

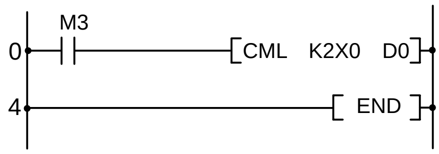
Operation Errors

(1) There are no operation errors associated with the CML(P) or DCML(P) instructions.

Program Example

(1) The following program inverts the data from X0 to X7, and transfers result to D0.

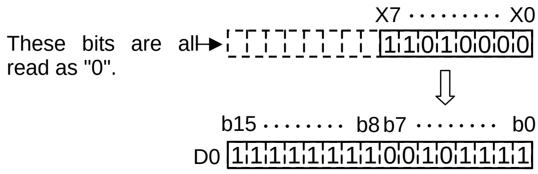
[Ladder Mode]



[List Mode]

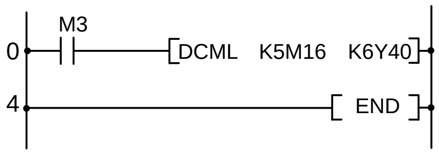
Steps	Inst.	Device
0	LD	M3
1	CML	K2X0
		D0
4	END	

When the number of bits at "S" is less than the number of bits at "D"



(2) The following program inverts the data at M16 to M35, and transfers the result to Y40 to Y53.

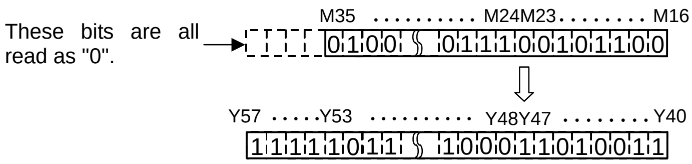
[Ladder Mode]



[List Mode]

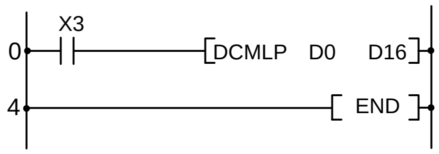
Steps	Inst.	Device
0	LD	M3
1	DCML	K5M16
		K5Y40
4	END	

When the number of bits at "S" is less than the number of bits at "D"



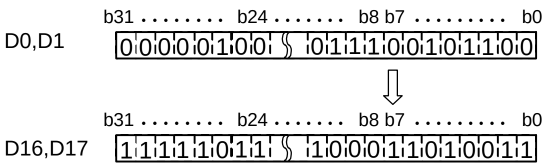
(3) The following program inverts the data at D0 and D1 when X3 is ON, and stores the result at D16 and D17.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X3
1	DCMLP	D0
		D16
4	END	



○ XCH,XCHP,DXCH,DXCHP … 16-bit and 32-bit data exchanges

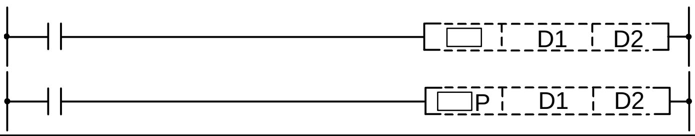
Compatible instruction mode	Usable Devices: **XCH, DXCH**																						
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D1		○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D2											○	○	○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol] [Execution Condition] ▭ indicates the signs XCH/DXCH

XCH, DXCH

XCHP, DXCHP



Set Data

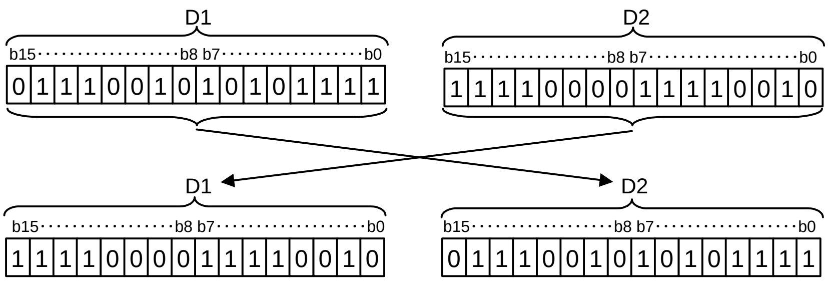
Set Data	Meaning	Data Type
D1	Head number of device storing data to be exchanged	BIN 16/32 bits
D2		

Functions

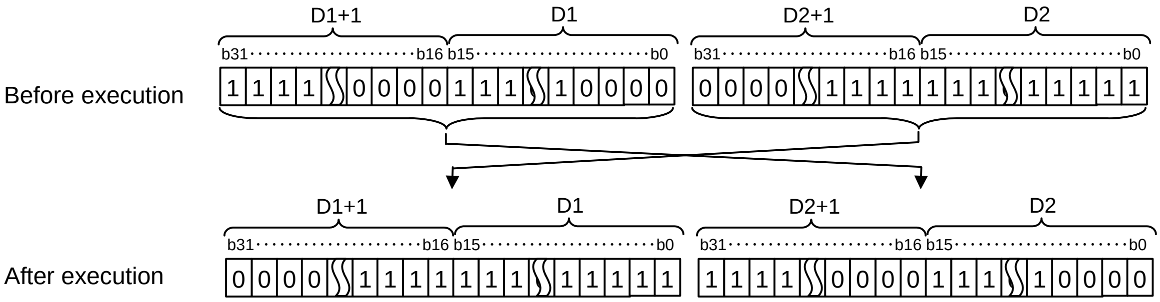
XCH (1) Conducts 16-bit data exchange between "D1" and "D2".

Before execution

After execution



DXCH (1) Conducts 32-bit data exchange between "D1"+1, "D1" and "D2"+1, "D2".



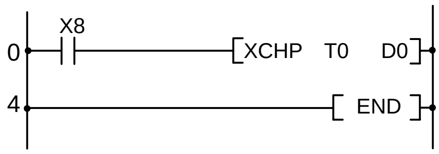
Operation Errors

(1) There are no operation errors associated with the XCH(P) or DXCH(P) instructions.

Program Example

(1) The following program exchanges the present value of T0 with the contents of D0 when X8 goes ON.

[Ladder Mode]

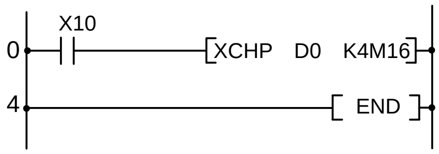


[List Mode]

Steps	Inst.	Device
0	LD	X8
1	XCHP	T0
		D0
4	END	

(2) The following program exchanges the contents of D0 with the data from M16 to M31 when X10 goes ON.

[Ladder Mode]

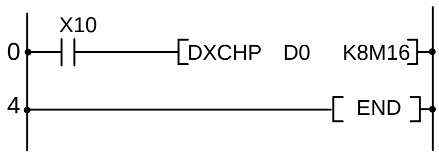


[List Mode]

Steps	Inst.	Device
0	LD	X10
1	XCHP	D0
		K4M16
4	END	

(3) The following program exchanges the contents of D0 and D1 with the data at M16 to M47 when X10 goes ON.

[Ladder Mode]

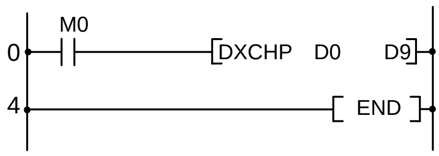


[List Mode]

Steps	Inst.	Device
0	LD	X10
1	DXCHP	D0
		K8M16
4	END	

(4) The following program exchanges the contents of D0 and D1 with those of D9 and D10 when M0 goes ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M0
1	DXCHP	D0
		D9
4	END	

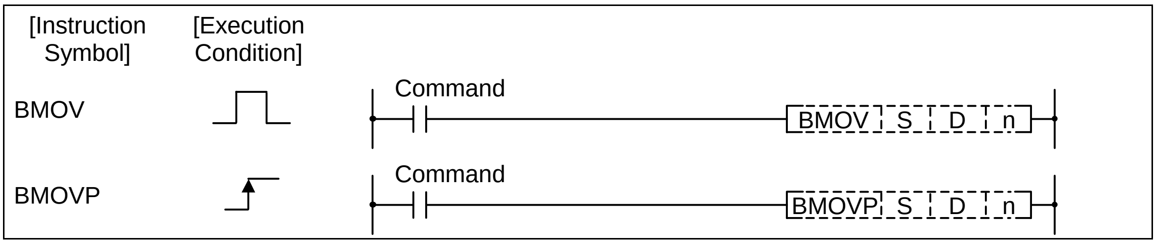
○ BMOV,BMOVP … 16-bit data block transfers

Compatible instruction mode	Usable Devices: **BMOV**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					
n																			○	○			

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					
n																			○	○			

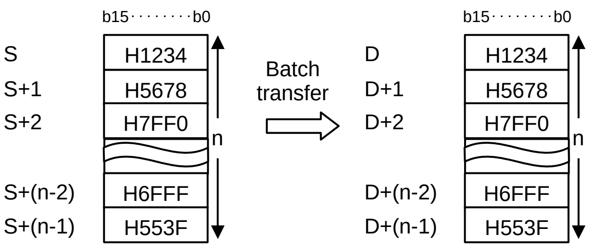


Set Data

Set Data	Meaning	Data Type
S	Head number of device storing data to transfer	BIN 16 bits
D	Head number of destination device	
n	Number of transfers	

Functions

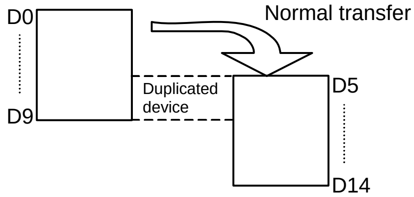
(1) Batch transfers "n" points of 16-bit data starting from the device designated by "S" to the area of "n" points starting from the device designated by "D".



(2) When transfer source and transfer destination are duplicated, the following operations are expected.

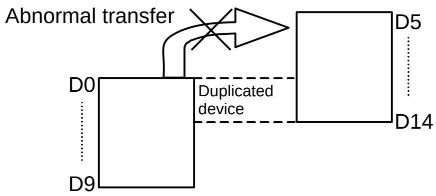
(a) Transferring to a smaller device No. results in normal operation.

(Example) When BMOV D0 D5 K10 are executed



(b) Transferring to a larger device No. results in abnormal operation.

(Example) When BMOV D5 D0 K10 are executed



Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.

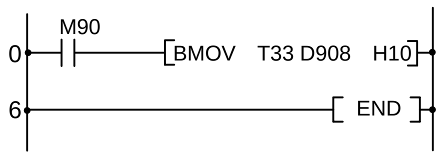
• In the case where a part of device area between S/D and n-th device does not exist. (Error code: 82)

POINT
Note that if the number of devices in the area between S/D and n-th device exceeds the relevant device, error will not occur. (In this case, transferred to the other device, as well.)

Program Example

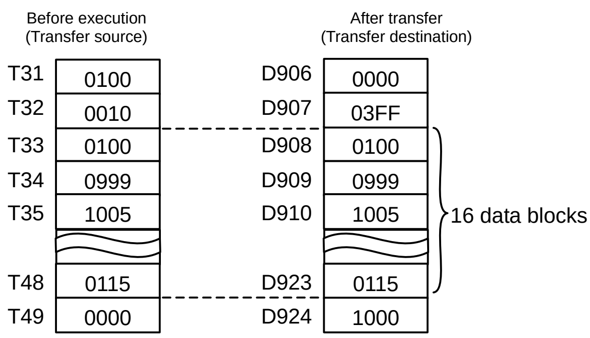
(1) The current value of T33 to T48 is transferred to D908 to D923.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M90
1	BMOV	T33
		D908
		H10
6	END	



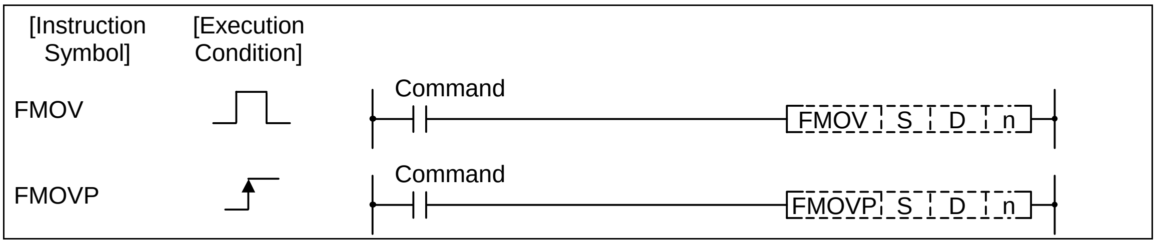
○ FMOV,FMOVP … 16-bit identical data batch transfer

Compatible instruction mode	Usable Devices: **FMOV**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○	○	○			
D											○	○	○	○	○	○		○					
n																			○	○			

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○	○	○			
D											○	○	○	○	○	○		○					
n																			○	○			

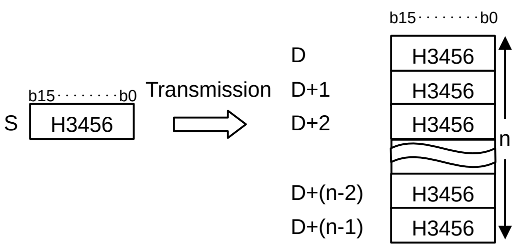


Set Data

Set Data	Meaning	Data Type
S	Data to transfer, or head number of device storing data to transfer	BIN 16 bits
D	Head number of destination device	
n	Number of transfers	

Functions

(1) Transfers 16-bit data from device designated by "S" to location n-points from device designated by "D".



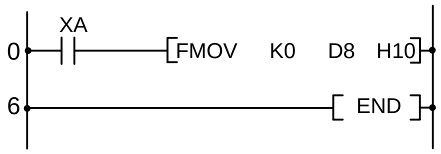
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- In the case where a part of device area between D and n-th device does not exist.
  (Error code: 82)

Program Example

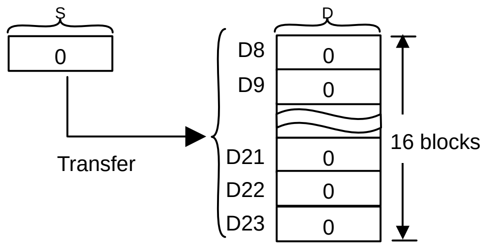
(1) When XA is turned ON, D8 to D23 is reset (cleared).

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XA
1	FMOV	K0
		D8
		H10
6	END	



○ S.TMOV … Transfer of timer and counter setting value

Compatible/Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○											
D													○	○	○	○		○					

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

[Instruction Symbol]	[Execution Condition]	
S.TMOV		

Set Data

Set Data	Meaning	Data Type
S	No. of timer and counter device to which setting value is transferred.	BIN 16 bits
D	Device No. of the transfer destination	

Functions

The setting value of timer and counter device specified with S is transferred to the device specified with D.
Note that, however, the actual setting value is transferred only if specified with constant.
If the setting value is specified with word device, normal transfer will not be carried out.

Timer setting	Setting value designation method with the timer output instruction				Setting value to be transferred by TMOV	
Fixed timer setting	Constant designation	OUT	Tx	Kn	○	Constant "n"
	Word device designation	OUT	Tx	Dn	×	Constant 0(zero)
Variable timer setting	Constant designation	OUT	Tx	Kn	○	Setting value set with the setting display device
	Word device designation	OUT	Tx	Dn	×	Setting value set with the setting display device

[Note]
When this instruction is monitored, the current value is displayed on the timer and counter device specified with S. If the timer and counter device are used with any function instructions other than above, everything indicates the current value.

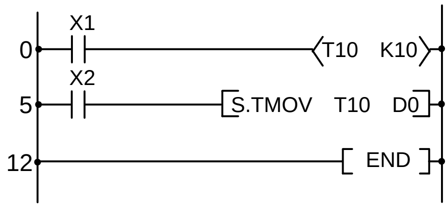
Operation Errors

(1) There are no operation errors associated with the S.TMOV instruction.

Program Example

(1) The setting value of T10 is transferred to D0.

[Ladder Mode]



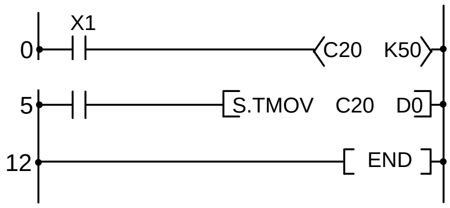
In this example, D0 equals to 10.

[List Mode]

Steps	Inst.	Device
0	LD	X1
1	OUT	T10
		K10
5	LD	X2
6	S.TMOV	T10
		D0
12	END	

(2) The setting value of C20 is transferred to D0.
Condition: Variable timer is set for the counter C20. This is the case where 100 is set for C20, using the setting display device.

[Ladder Mode]



In this example, D0 equals to 100.

[List Mode]

Steps	Inst.	Device
0	LD	X1
1	OUT	C20
		K50
5	LD	X2
6	S.TMOV	C20
		D0
12	END	

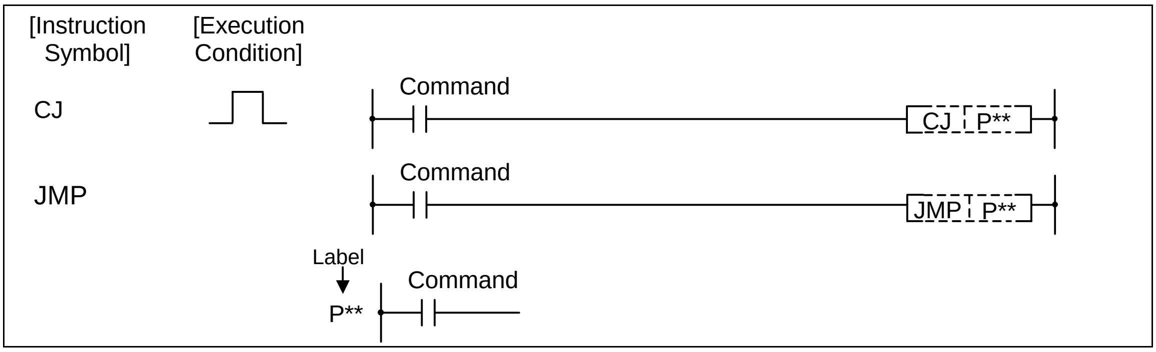
○ CJ,JMP ……Conditional jump

Compatible instruction mode	Usable instruction: **CJ**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
P																					○		

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
P																					○		



Set Data

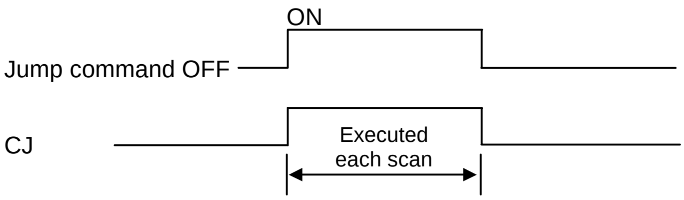
Set Data	Meaning	Data Type
P	Pointer number of jump destination	Device name

Functions

CJ

(1) Executes program of designated pointer number within the same program file when jump command is ON.

(2) Executes next step in program when jump command is OFF.



JMP

(1) A program of specified pointer No. in the same program file is executed unconditionally.

POINTS

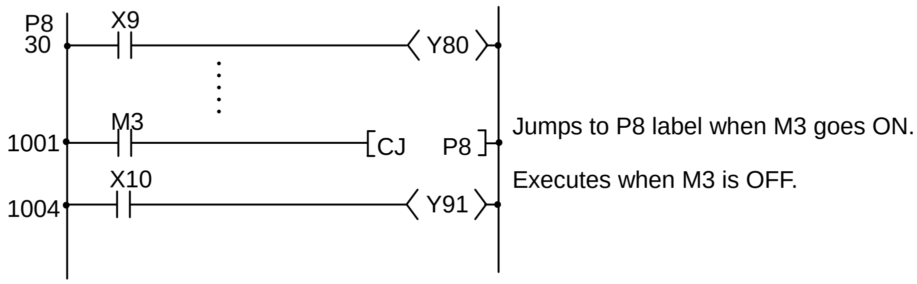
Note the following points when using the jump instruction.

(1) Jump instructions can be used only for pointer numbers within the same program file.

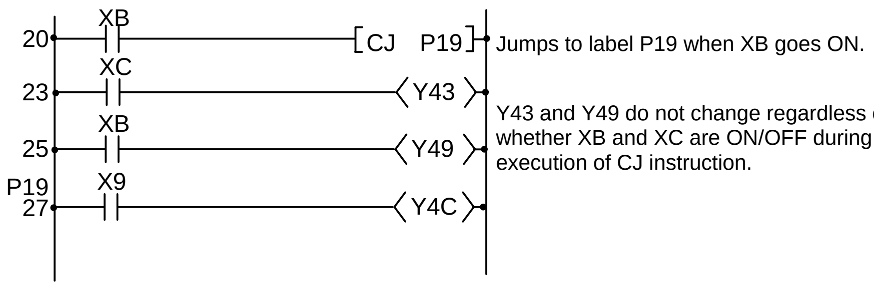
(2) After the timer coil has gone ON, accurate measurements cannot be made if there is an attempt to jump the timer of a coil that has been turned ON using the CJ or JMP instructions.

(3) Scan time is shortened if the CJ or JMP instruction is used to force a jump to the rear.

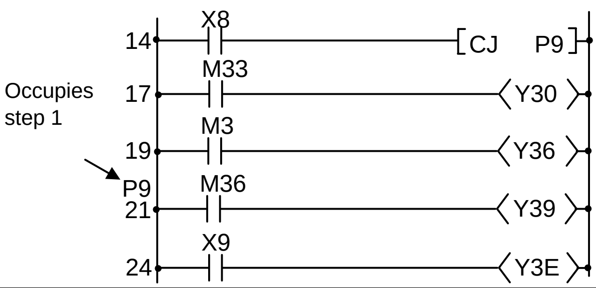
(4) The CJ and JMP instructions can be used to jump to a step prior to the step currently being executed.



(5) The device to which a jump has been made with CJ or JMP does not change.



(6) The lable (P**) occupies step 1.



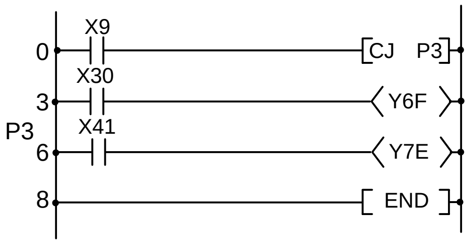
Operation Errors

(1) In the following cases an operation is returned, the error flag (SM0) goes ON, and the error code is stored at SD0.
- A pointer number which is not in use as a label in the same program has been designated. (Error code: 20 or 85)
- A common pointer in the other program has been designated. (Error code: 20 or 85)

Program Example

(1) The following program jumps to P3 when X9 goes ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X9
1	CJ	P3
3	LD	X30
4	OUT	Y6F
5		P3
6	LD	X41
7	OUT	Y7E
8	END	

○ FEND … Program termination

Compatible/Extended instruction mode	

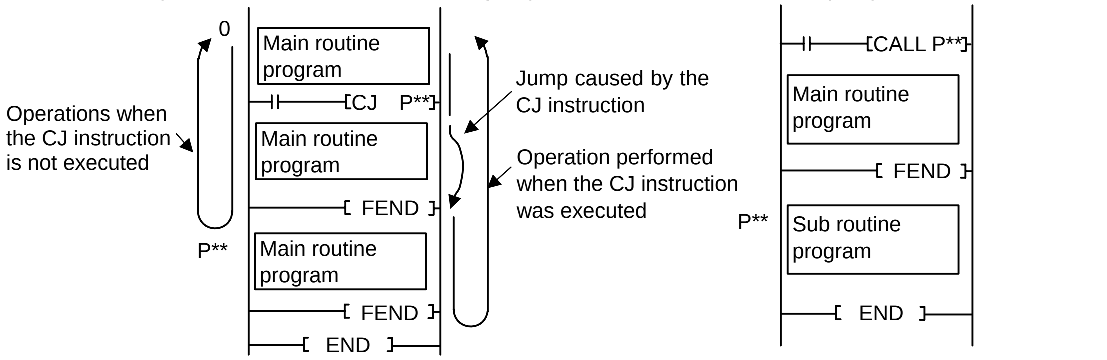
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

[Instruction Symbol]	[Execution Condition]	
FEND		[FEND]

Functions

(1) FEND instruction is used when branching a sequence program operation by CJ instruction, etc., or when dividing between the main routine program and the sub routine program.



(a) When using the CJ instruction

(b) When there is a subroutine program

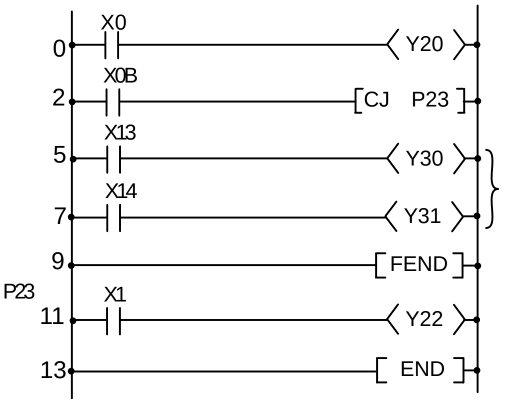
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- A FEND instruction is executed after the execution of the CALL instruction, and before the execution of the RET instruction. (Error code: 26)

Program Example

(1) The following program uses the CJ instruction.

[Ladder Mode]



When XB is ON, jumps to label P23; from P23, executes the next step

Executed when XB is OFF

Indicates the termination of the sequence program when XB is OFF

[List Mode]

Steps	Inst.	Device
0	LD	X0
	:	
	:	
9	FEND	
10		P23
11	LD	X1
12	OUT	Y22
13	END	

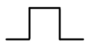
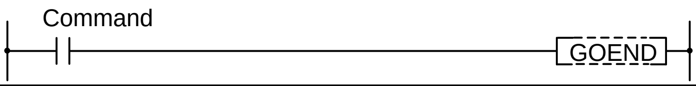
○ GOEND … Jump to END

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

[Instruction Symbol]	[Execution Condition]	
GOEND		

Functions

(1) Jumps to FEND or END instruction in the same program file.

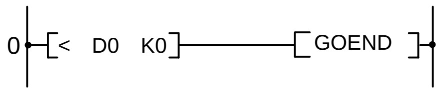
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- A GOEND instruction has been executed after the execution of the CALL instruction, and prior to the execution of the RET instruction. (Error code: 26)

Program Example

(1) The following program jumps to the END instruction if D0 holds a negative number.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD<	D0
		K0
3	GOEND	

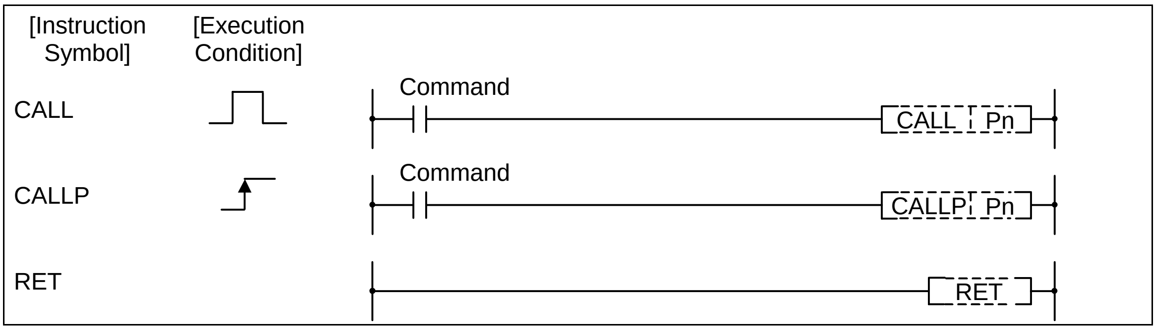
○ CALL,CALLP,RET … Sub-routine program calls and return from sub-routine programs

Compatible instruction mode	Usable instruction: **CALL, RET**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
P																					○		

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
P																					○		



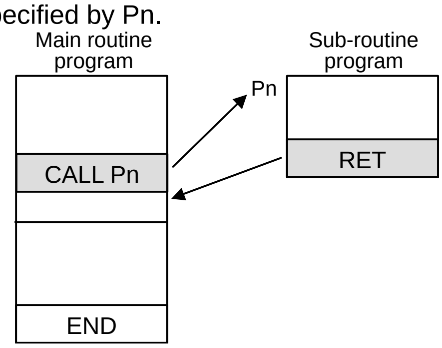
Set Data

Set Data	Meaning	Data Type
Pn	First pointer number of sub-routine program	Device name

Functions

CALL

(1) When the CALL (P) instruction is executed, executes the sub-routine program of the program specified by Pn.



(2) CALL or CALLP instructions can be nested up to 8 deep.

POINT

There are the following two pointer numbers to be set by the CALL(P) instruction.
Refer to "5.3.14 Pointer P" of "5.3 Detailed Explanation of Devices" for details.

- Local pointer
- Common pointer

RET (1) Indicates end of sub-routine program

(2) When the RET instruction is executed, returns to the step following the CALL(P) instruction which called the sub-routine program.

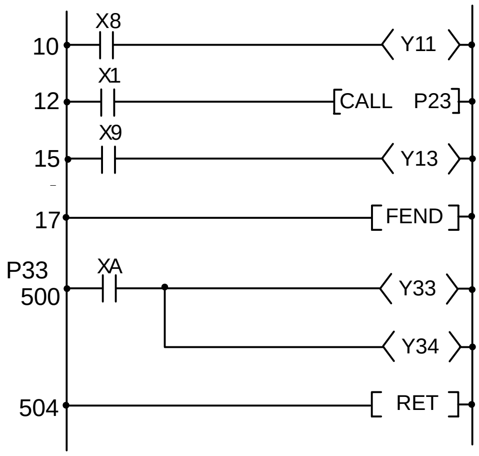
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- Following the execution of the CALL(P) instruction, an END or FEND instruction is executed before the execution of the RET instruction. (Error code: 26)
- An RET instruction is executed prior to the execution of the CALL (P) instruction. (Error code: 26)
- When stuck area has exceeded due to CALL(P) instruction's nesting, etc. (Error code: 86)

Program Example

(1) The following program executes a sub-routine program when X1 turns ON from OFF.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
10	LD	X8
11	OUT	Y11
12	LD	X1
13	CALL	P33
15	LD	X9
16	OUT	Y13
17	FEND	
18		
:		
500		P33
501	LD	XA
502	OUT	Y33
503	OUT	Y34
504	RET	
505		

○ FOR,NEXT … FOR to NEXT instruction loop

Compatible instruction mode	Not available

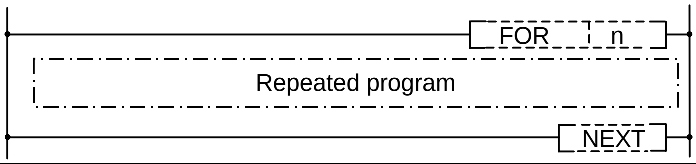
Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	

This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

[Instruction Symbol] [Execution Condition]

FOR

NEXT



Set Data

Set Data	Meaning	Data Type
n	Number of repetitions of the FOR to NEXT loop (from 1 to 32767)	BIN 16 bits

Functions

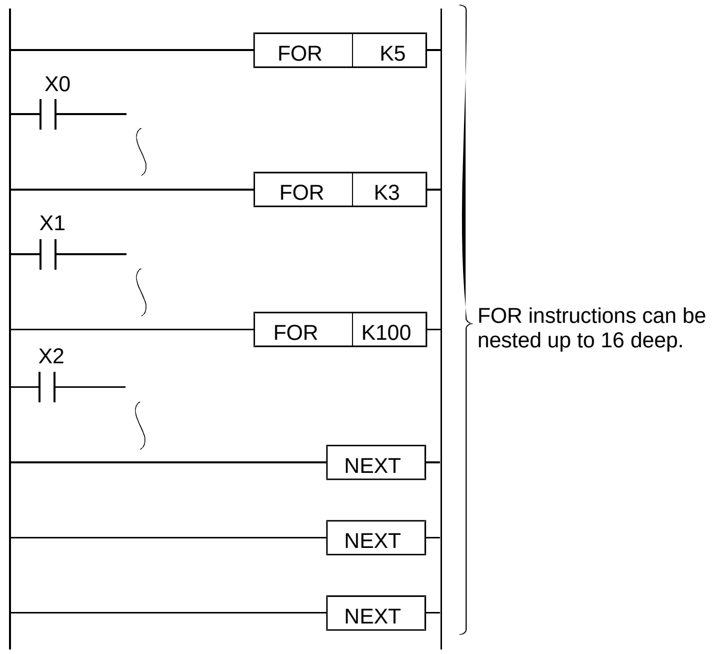
(1) When the processing in the FOR to NEXT loop is executed n-times without conditions, the step following the NEXT instruction will be executed.

(2) The value of n can be designated at between 1 and 32767. If it is designated at a value of from -32768 to 0, it will be executed as though n=1.

(3) If you do not desire to execute the processing called for within the FOR to NEXT loop, use the CJ instruction to jump. Setting the repetition times to "0" would not allow you to skip the process between FOR and NEXT instructions.

(4) To force an end to the repetitious execution of the FOR to NEXT loop during the execution of the loop, insert a BREAK instruction. Premature termination with CJ instruction, etc. will result in an operation error.

(5) FOR instructions can be nested up to 16 deep.



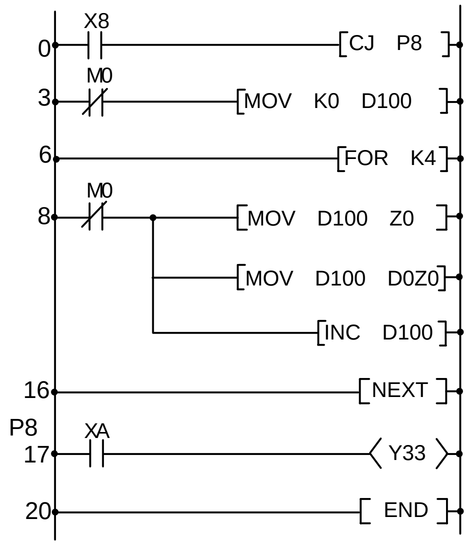
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- An END (FEND) instruction is executed after the execution of a FOR instruction, but before the execution of a NEXT instruction. (Error code: 31)
- A NEXT instruction is executed prior to the execution of a FOR instruction. (Error code: 31)
- The 17th FOR instruction is encountered when FOR instructions have been nested. (Error code: 30)

Program Example

(1) The following program executes the FOR to NEXT loop when X8 is OFF, and does not execute it when X8 is ON.

[Ladder Mode]



[List Mode]

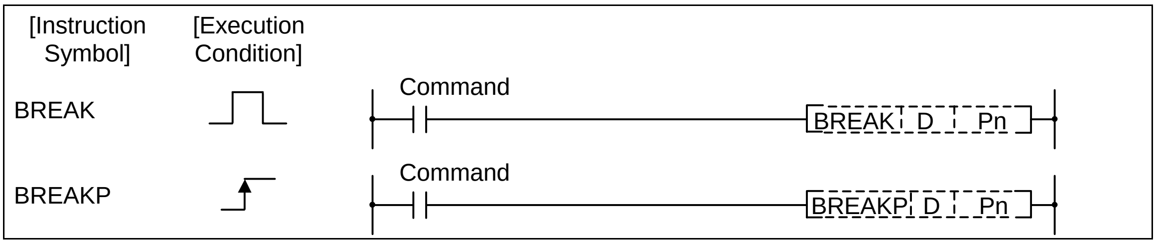
Steps	Inst.	Device
0	LD	X8
1	CJ	P8
3	LDI	M0
4	MOV	K0
		D100
6	FOR	K4
8	LDI	M0
9	MOV	D100
		Z0
12	MOV	D100
		D0Z0
15	INC	Z3
16	NEXT	
17		P8
18	LD	XA
19	OUT	Y33
20	END	

○ BREAK,BREAKP … Forced end of FOR to NEXT instruction loop

Compatible instruction mode	Not available

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
P																					○		

This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

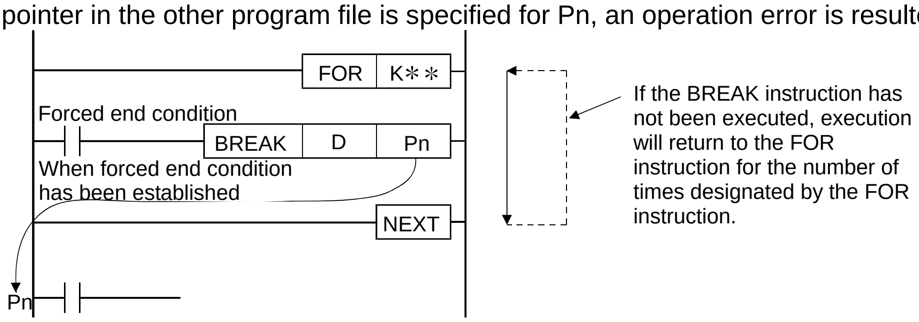


Set Data

Set Data	Meaning	Data Type
D	Number of device that will store the number of repetitions remaining	BIN 16 bits
Pn	Number of branch destination pointer when the end of processing repetitions.	Device name (pointer)

Functions

(1) Forces the end of repetition processing based on the FOR to NEXT instruction loop, and shifts to the execution of the pointer designated by Pn.
Only the pointer within the same program file can be specified for Pn.
If a pointer in the other program file is specified for Pn, an operation error is resulted.



(2) The number of repetitions remaining at the point that the FOR to NEXT loop was brought to a forced end is stored at "D".
However, the number of times when the BREAK instruction was executed is also included in the number of repetitions remaining.

(3) The BREAK instruction can be used only during the execution of a FOR to NEXT instruction loop.

(4) The BREAK instruction can be used only when there is only one level of nesting.
If an end is forced when there are multiple nesting levels, execute the same number of BREAK instructions as there are nesting levels.

Operation Errors

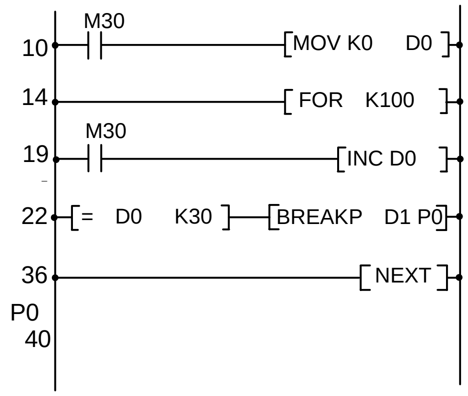
(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.

- The BREAK instruction is used in a case other than with the FOR to NEXT instruction loop. (Error code: 32)
- The jump destination for the pointer designated by Pn does not exist. (Error code: 85)
- The pointer of another program file is designated for Pn. (Error code: 85)

Program Example

(1) The following program forces the FOR to NEXT loop to end when the value of D2 reaches 30 (when the FOR to NEXT loop has been executed 30 times).

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
10	LD	M30
11	MOV	K0
		D0
14	FOR	K100
19	LD	M30
20	INC	D0
22	LD=	D0
		K30
25	BREAKP	D1
		P0
36	NEXT	
40		P0

REMARK

(1) The value 71 is stored at D1 as the remaining number of repetitions when the BREAK instruction is executed.

○ WAND,WANDP,DAND,DANDP … Logical products with 16-bit and 32-bit data (Device at storage destination: Independent type)

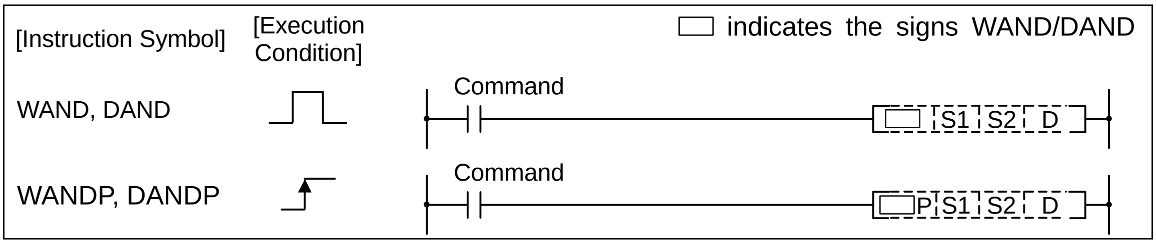
Compatible instruction mode	Usable instruction: **WAND**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

△: S1 and S2 cannot be specified as constant at the same time.



Set Data

Set Data	Meaning	Data Type
S1	Data from which logical product will be determined, or number of devices storing such data	BIN 16/32 bits
S2		
D	Number of devices where logical product operation results will be stored	

Functions

WAND

(1) A logical product operation is conducted for each bit of the 16-bit data of the device designated at "S1" and the 16-bit data of the device designated at "S2", and the results are stored in the device designated at "D".

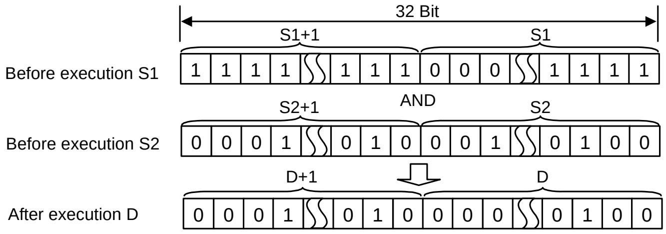
16 Bit

Before execution S1	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1
	AND															
Before execution S2	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
After execution D	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0

(2) In the case of bit devices, digits above the number designated are processed as 0 in the operation.

DAND

(1) Conducts a logical product operation on each bit of the 32-bit data for the device designated by "S1" and the 32-bit data for the device designated by "S2", and stores the results at the device designated by "D".



(2) In the case of bit devices, digits other than the number designated are processed as 0 in the operation.

○ WAND,WANDP,DAND,DANDP … Logical products with 16-bit and 32-bit data (Device at storage destination: Shared type)

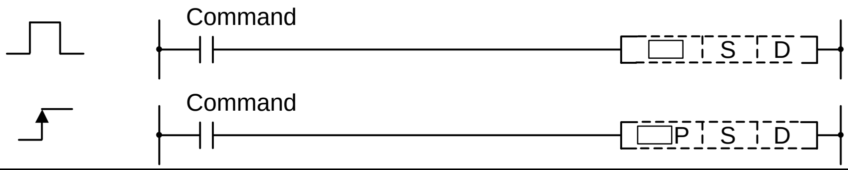
Compatible instruction mode	Usable instruction: **DAND**																					
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D											○	○	○	○	○	○		○					

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol] [Execution Condition] ☐ indicates the signs WAND/DAND

WAND, DAND

WANDP, DANDP



Set Data

Set Data	Meaning	Data Type
S	Data from which logical product will be determined, or number of devices storing such data	BIN 16/32 bits
D		

Functions

WAND

(1) A logical product operation is conducted for each bit of the 16-bit data of the device designated at "D" and the 16-bit data of the device designated at "S", and the results are stored in the device designated at "D".

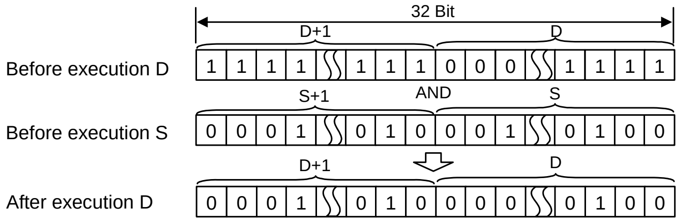
16 Bit

Before execution D	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1
AND																
Before execution S	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
After execution D	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0

(2) In the case of bit devices, digits other than the number designated are processed as 0 in the operation.

DAND

(1) Conducts a logical product operation on each bit of the 32-bit data for the device designated by "D" and the 32-bit data for the device designated by "S", and stores the results at the device designated by "D".



(2) In the case of bit devices, digits other than the number designated are processed as 0 in the operation.

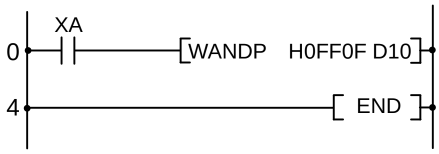
Operation Errors

(1) There are no operation errors associated with the WAND(P) or DAND(P) instructions.

Program Example

(1) The following program masks the digit in the 10s place of the 4-digit BCD value at D10 (second digit from the end) to 0 when XA is ON.

[Ladder Mode]



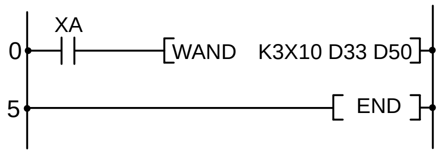
[List Mode]

Steps	Inst.	Device
0	LD	X0A
1	WANDP	H0FF0F
3		D10
4	END	

D10	BCD1234	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
		AND															
	HFF0F	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1
D10	BCD1204	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0

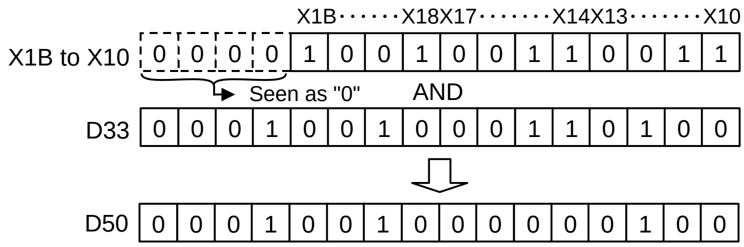
(2) The following program performs a logical product operation on the data from X10 to X1B and the data at D33 when XA is ON, and outputs the results to D50.

[Ladder Mode]



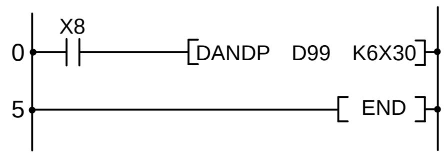
[List Mode]

Steps	Inst.	Device
0	LD	XA
1	WAND	K3X10
		D33
		D50
5	END	



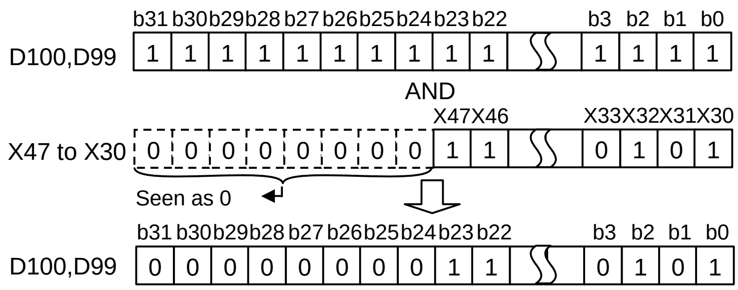
(3) The following program performs a logical product operation on the data at D99 and D100, and the 24-bit data between X30 and X47 when X8 is ON, and stores the results at D99 and D100.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X8
1	DANDP	D99
		K6X30
5	END	

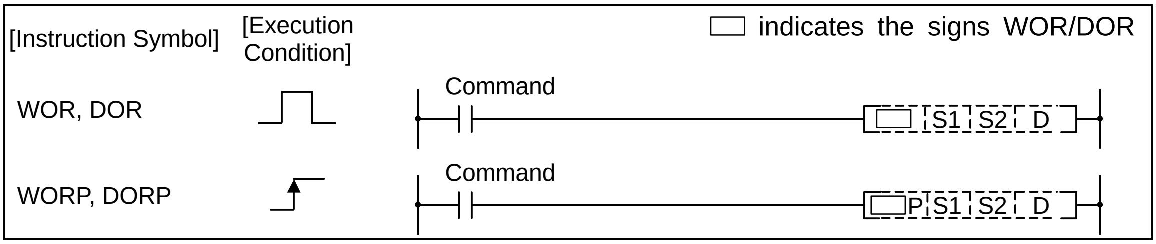


○ WOR,WORP,DOR,DORP … Logical sums of 16-bit and 32-bit data (Device at storage destination: Independent type)

Compatible instruction mode	Usable instruction: **WOR**																						
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

△: S1 and S2 cannot be specified as constant at the same time.



Set Data

Set Data	Meaning	Data Type
S1	The data on which a logical sum operation will be performed, or the number of the devices storing this data	BIN 16/32 bits
S2		
D	Number of devices that will store the results of the logical sum operation	

Functions

WOR

(1) Conducts a logical sum operation on each bit of the 16-bit data of the device designated by "S1" and the 16-bit data of the device designated by "S2", and stores the results at the device designated by "D".

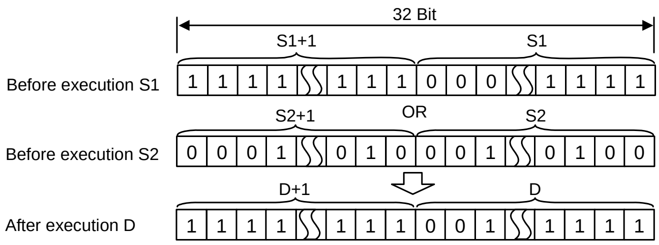
16 Bit

Before execution S1	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1
OR																
Before execution S2	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
After execution D	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1

(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

DOR

(1) Conducts a logical sum operation on each bit of the 32-bit data of the device designated by "S1" and the 32-bit data of the device designated by "S2", and stores the results at the device designated by "D".



(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

○ WOR,WORP,DOR,DORP … Logical sums of 16-bit and 32-bit data (Device at storage destination: Shared type)

Compatible instruction mode	Usable instruction: **DOR**																					
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D											○	○	○	○	○	○		○					

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol]	[Execution Condition]	□ indicates the signs WOR/DOR
WOR, DOR	⎍	Command ─┤├─ [□ S D]
WORP, DORP	↑	Command ─┤├─ [□P S D]

Set Data

Set Data	Meaning	Data Type
S	The data on which a logical sum operation will be performed, or the number of the devices storing this data	BIN 16/32 bits
D		

Functions

WOR

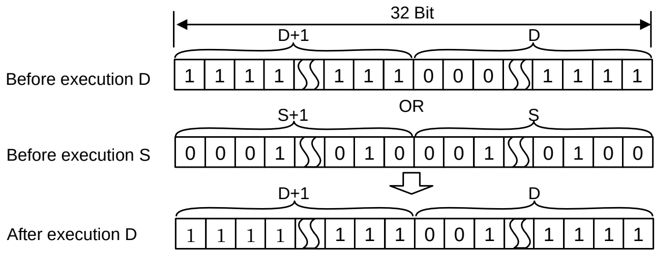
(1) Conducts a logical sum operation on each bit of the 16-bit data of the device designated by "D" and the 16-bit data of the device designated by "S", and stores the results at the device designated by "D".

16 Bit

Before execution D	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1
OR																
Before execution S	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
⇩																
After execution D	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1

(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

DOR (1) Conducts a logical sum operation on each bit of the 32-bit data of the device designated by "D" and the 32-bit data of the device designated by "S", and stores the results at the device designated by "D".



(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

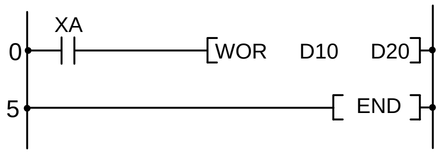
Operation Errors

(1) There are no operation errors associated with the WOR(P) or DOR(P) instructions.

Program Example

(1) The following program performs a logical sum operation on the data at D10 and D20 when XA goes ON, and stores the results at D10.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0A
1	WOR	D10
		D20
5	END	

D10	1	1	0	0	1	1	0	0	1	1	1	1	0	0	0	0

OR

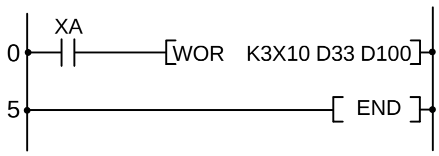
D20	0	0	0	0	1	0	1	1	1	0	0	1	0	0	0	1

⇩

D10	1	1	0	0	1	1	1	1	1	1	1	1	0	1	0	1

(2) The following program performs a logical sum operation on the data from X10 to X1B, and the data at D33, and outputs the result to D100 when XA is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XA
1	WOR	K3X10
		D33
		D100
5	END	

X1B······X18X17·······X14X13·······X10

X1B to X10	0	0	0	0	1	0	0	1	0	0	1	1	0	0	1	1

Seen as 0 OR

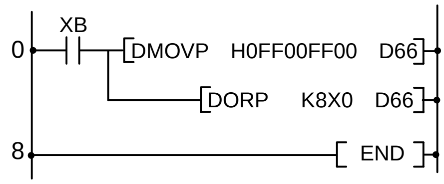
D33	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0

⇩

D100	0	0	0	1	1	0	1	1	0	0	1	1	0	1	1	1

(3) The following program performs a logical sum operation on the 32-bit data from X0 to X1F, and on the hexadecimal value FF00FF00H when XB goes ON, and stores the results at R66 and R67.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X8
1	DMOVP	H0FF00FF00
		D66
5	DORP	K8X0
		D66
8	END	

○ WXOR,WXORP,DXOR,DXORP … 16-bit and 32-bit exclusive OR operations (Device at storage destination: Independent type)

Compatible instruction mode	Usable instruction: **WXOR**

Set Data	Bit Devices										Word Devices								Constant		Pointer	Digit designation	Index
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D											○	○	○	○	○	○		○					

Extended instruction mode	

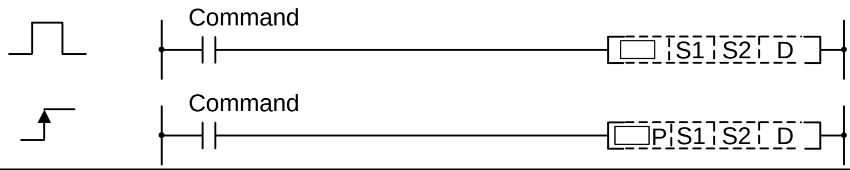
Set Data	Bit Devices										Word Devices								Constant		Pointer	Digit designation	Index
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

△: S1 and S2 cannot be specified as constant at the same time.

[Instruction Symbol] [Execution Condition] ▭ indicates the signs WXOR/DXOR

WXOR, DXOR

WXORP, DXORP



Set Data

Set Data	Meaning	Data Type
S1	Data on which exclusive OR operation will be performed, or number of devices storing such data	BIN 16/32 bits
S2		
D	Number of devices storing data to be EXCLUSIVE ORed	

Functions

WXOR

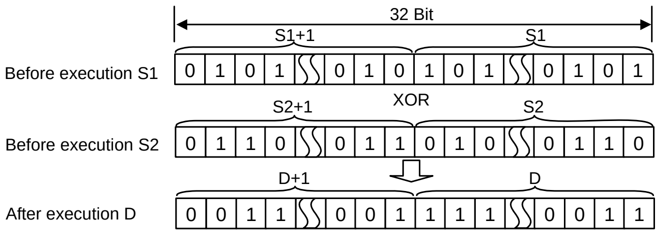
(1) Conducts an exclusive OR operation on each bit of the 16-bit data of the device designated by "S1" and the 16-bit data of the device designated by "S2", and stores the results at the device designated by "D".

16 Bit

Before execution S1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
XOR																
Before execution S2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
After execution D	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

DXOR (1) Conducts an exclusive OR operation on each bit of the 32-bit data designated by "S1" and the 32-bit data designated by "S2", and stores the results at the device designated by "D".



(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

○ WXOR,WXORP,DXOR,DXORP … 16-bit and 32-bit exclusive OR operations (Device at storage destination: Shared type)

Compatible instruction mode	Usable instruction: **DXOR**

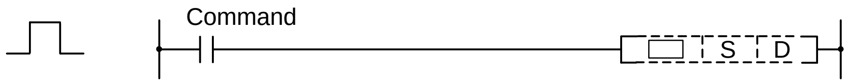
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D											○	○	○	○	○	○		○					

Extended instruction mode	

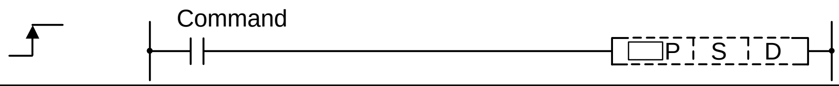
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol] [Execution Condition] ▭ indicates the signs WXOR/DXOR

WXOR, DXOR



WXORP, DXORP



Set Data

Set Data	Meaning	Data Type
S	Data on which exclusive OR operation will be performed, or number of devices storing such data	BIN 16/32 bits
D		

Functions

WXOR

(1) Conducts an exclusive OR operation on each bit of the 16-bit data of the device designated by "D" and the 16-bit data of the device designated by "S", and stores the results at the device designated by "D".

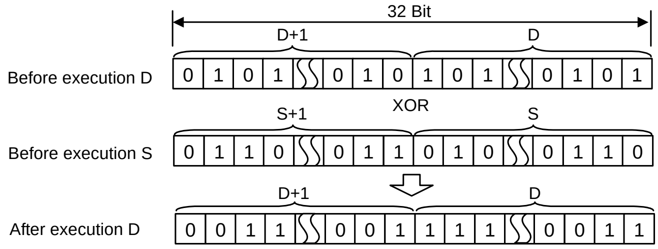
16 Bit

Before execution D	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
	XOR															
Before execution S	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
After execution D	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

DXOR

(1) Conducts an exclusive OR operation on each bit of the 32-bit data designated by "D" and the 32-bit data designated by "S", and stores the results at the device designated by "D".



(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

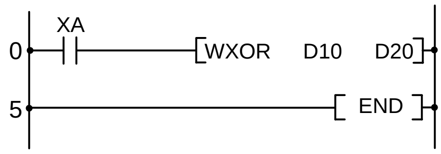
Operation Errors

(1) There are no operation errors associated with the WXOR(P) or DXOR(P) instructions.

Program Example

(1) The following program performs an exclusive OR operation on the data at D10 and D20 when XA is ON, and stores the result at D10.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0A
1	WXOR	D10
		D20
5	END	

D10	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

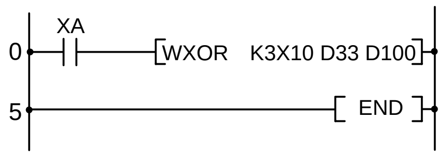
XOR

D20	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	1

D10	0	1	1	0	1	1	0	0	0	1	1	0	1	1	0	0

(2) The following program conducts an exclusive OR operation on the data from X10 to X1B and the data at D33 when XA is ON, and outputs the result to D100.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XA
1	WXOR	K3X10
		D33
		D100
5	END	

X1B······X18X17·······X14X13·······X10

X1B to X10	0	0	0	0	1	0	0	1	0	0	1	1	0	0	1	1

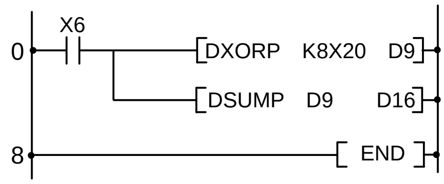
Seen as "0" XOR

D33	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0

D100	0	0	0	1	1	0	1	1	0	0	0	0	0	1	1	1

(3) The following program compares the bit pattern of the 32-bit data from X20 to X3F with the bit pattern of the data at D9 and D10 when X6 is ON, and stores the number of differing bits at D16.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X6
1	DXORP	K8X20
		D9
5	DSUMP	D9
		D16
8	END	

○ WXNR,WXNRP,DXNR,DXNRP … 16-bit and 32-bit data non-exclusive logical sum operations (Device at storage destination: Independent type)

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△		○	
S2	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	△	△			
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

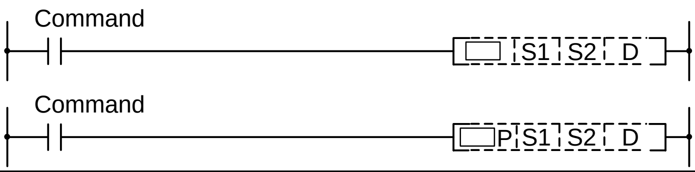
This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.

△: S1 and S2 cannot be specified as a constant at the same time.

[Instruction Symbol] [Execution Condition] ▭ indicates the signs WXNR/DXNR

WXNR, DXNR

WXNRP, DXNRP



Set Data

Set Data	Meaning	Data Type
S1	Data on which non-exclusive logical sum operation will be performed, or number of devices where such data is being stored	BIN 16/32 bits
S2		
D	Number of devices that will store results of the non-exclusive logical sum operation	

Functions

WXNR

(1) Conducts a non-exclusive logical sum operation on each bit of the 16-bit data of the device designated by "S1" and the 16-bit data of the device designated by "S2", and stores the results at the device designated by "D".

16 Bit

Before execution S1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

XNR

Before execution S2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1

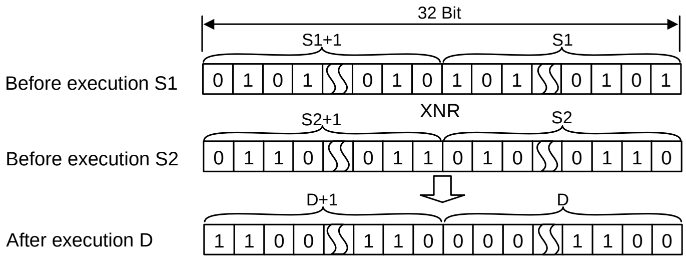
⇩

After execution D	1	0	1	0	0	1	0	1	1	0	1	0	0	1	0	1

(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

DXNR

(1) Conducts a non-exclusive logical sum operation on each bit of the 32-bit data of the device designated by "S1" and the 32-bit data of the device designated by "S2", and stores the results at the device designated by "D".



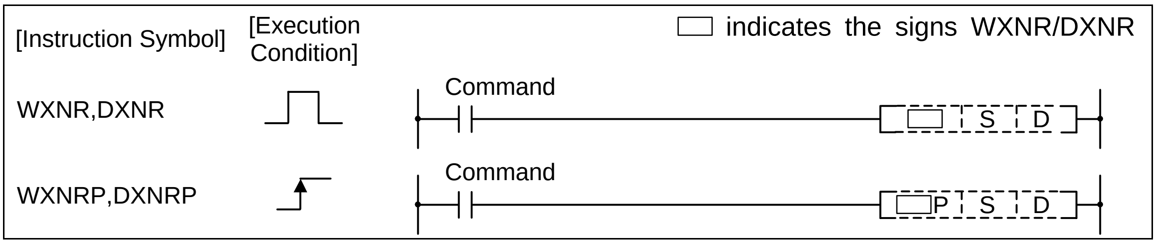
(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

○ WXNR,WXNRP,DXNR,DXNRP … 16-bit and 32-bit data non-exclusive logical sum operations (Device at storage destination: Shared type)

Compatible instruction mode	Not available

Extended instruction mode																						
Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

This cannot be used with the compatible instruction mode. Refer to "6.2 Instruction Tables" for correspondence.



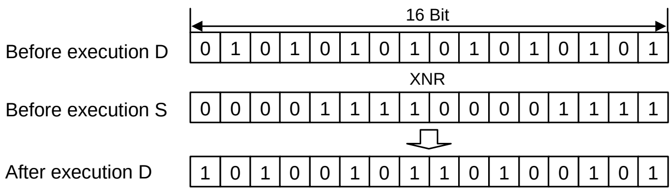
Set Data

Set Data	Meaning	Data Type
S	Data on which non-exclusive logical sum operation will be performed, or number of devices where such data is being stored	BIN 16/32 bits
D		

Functions

WXNR

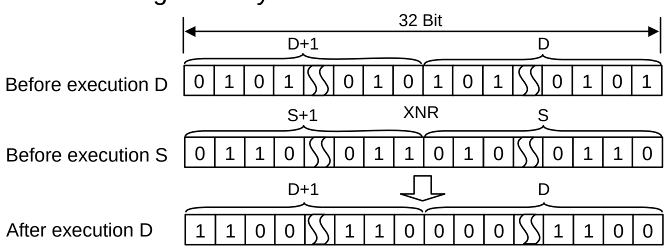
(1) Conducts a non-exclusive logical sum operation on each bit of the 16-bit data of the device designated by "D" and the 16-bit data of the device designated by "S", and stores the results at the device designated by "D".



(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

DXNR

(1) Conducts a non-exclusive logical sum operation on each bit of the 32-bit data of the device designated by "D" and the 32-bit data of the device designated by "S", and stores the results at the device designated by "D".



(2) For bit device, the numbers of the digits other than the designated digit are processed as 0 in the operation.

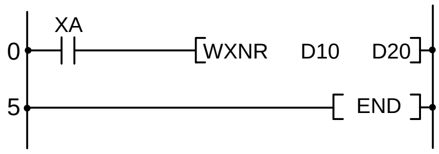
Operation Errors

(1) There are no operation errors associated with the WXNR(P) or DXNR(P) instructions.

Program Example

(1) The following program performs a non-exclusive OR operation on the data at D10 and D20 when XA is ON, and stores the result at D10.

[Ladder Mode]



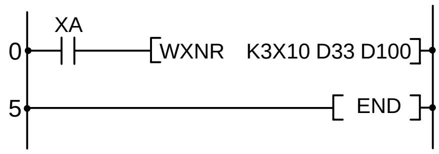
[List Mode]

Steps	Inst.	Device
0	LD	X0A
1	WXNR	D10
		D20
5	END	

D10	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
XNR																
D20	0	0	1	1	1	0	0	1	0	0	1	1	1	0	0	1
D10	1	0	0	1	0	0	1	1	1	0	0	1	0	0	1	1

(2) The following program conducts a non-exclusive OR operation on the data from X10 to X1B and the data at D33 when XA is ON, and outputs the result to D100.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XA
1	WXNR	K3X10
		D33
		D100
5	END	

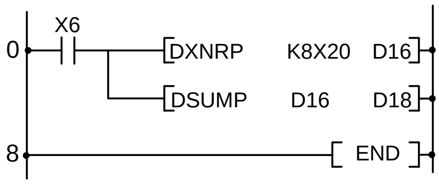
X1B······X18X17·······X14X13·······X10

X1B to X10	0	0	0	0	1	0	0	1	0	0	1	1	0	0	1	1
D33	0	0	0	1	0	0	1	0	0	0	1	1	0	1	0	0
D100	1	1	1	0	0	1	0	0	1	1	1	1	1	0	0	0

Seen as "0" XNR

(3) The following program compares the bit patterns of the 32-bit data located from X20 to X3F with the bit patterns of the data at D16 and D17 when X6 is ON, and stores the same number of bits at D18.

[Ladder Mode]



[List Mode]

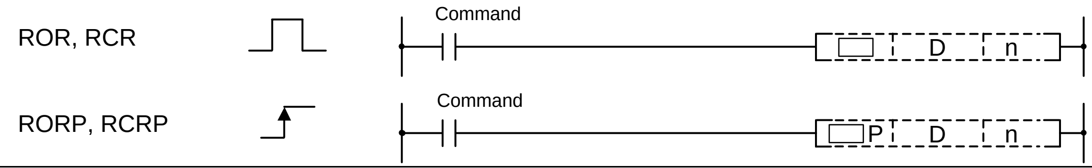
Steps	Inst.	Device
0	LD	X6
1	DXNRP	K8X20
		D16
5	DSUMP	D16
		D18
8	END	

○ ROR,RORP,RCR,RCRP … Right rotation of 16-bit data

Compatible instruction mode	Usable instruction: **ROR, RCR**																						
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n																			○	○			

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	

[Instruction Symbol] [Execution Condition] ▭ indicates the signs ROR/RCR



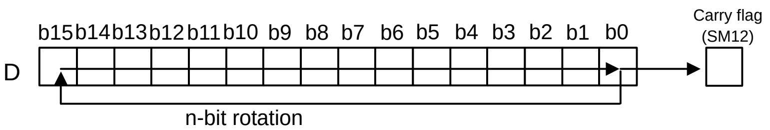
Set Data

Set Data	Meaning	Data Type
D	Initial number of devices to perform rotation	BIN 16 bits
n	Number of rotations (0 to 15)	

Functions

ROR

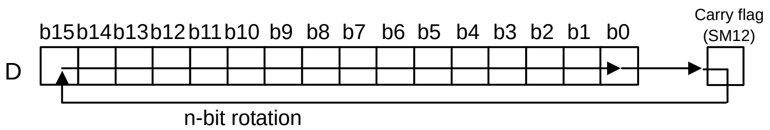
(1) Rotates 16-bit data of the device designated by "D", not including the carry flag, n-bits to the right. The carry flag is ON or OFF depending on the status prior to the execution of the ROR instruction.



(2) Specify any of 0 to 15 as n. If the value specified as n is 16 or greater, operations will be as follows:
- When n is 16, the value becomes the one when 16-bit rotation was executed.
- When n is 17 or above, the value of D becomes indefinite.

RCR

(1) Rotates 16-bit data of the device designated by "D", including carry flag, n bits to the right. The carry flag is ON or OFF depending on the status prior to the execution of the ROR instruction.



(2) Specify any of 0 to 15 as n. If the value specified as n is 16 or greater, operations will be as follows:
- When n is 16, the value becomes the one when 16-bit rotation was executed.
- When n is 17 or above, the value of D becomes indefinite.

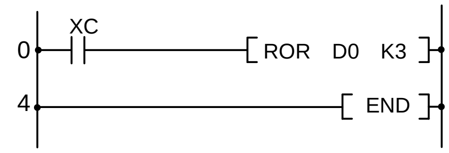
Operation Errors

(1) There are no operation errors associated with the ROR(P) or RCR(P) instructions.

Program Example

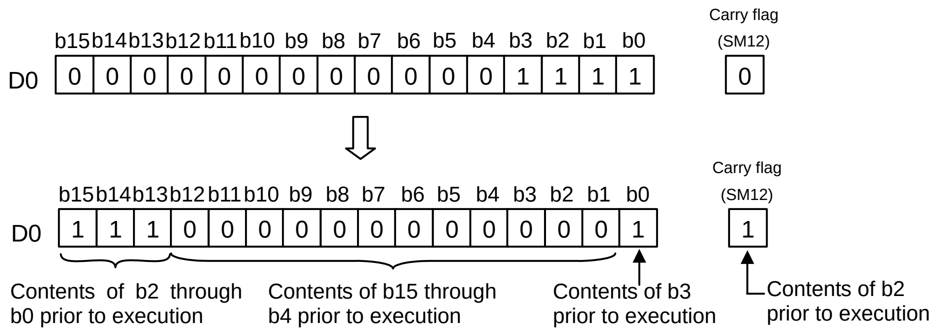
(1) The following program rotates the contents of D0, though not including the carry flag, 3 bits to the right when XC is ON.

[Ladder Mode]



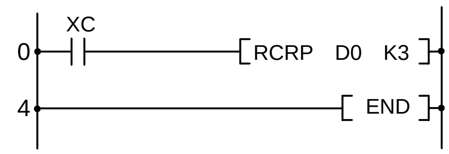
[List Mode]

Steps	Inst.	Device
0	LD	XC
1	ROR	D10
		K3
4	END	



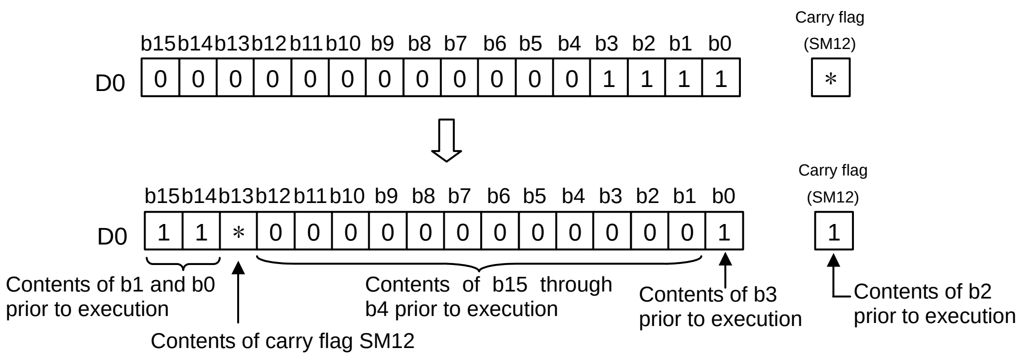
(2) The following program rotates the contents of D0, including the carry flag, 3 bits to the right when XC is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XC
1	RCRP	D10
		K3
4	END	



* The carry flag goes ON or OFF depending on its status prior to the execution of RCR.

○ ROL,ROLP,RCL,RCLP … Left rotation of 16-bit data

Compatible instruction mode	Usable instruction: **ROL, RCL**

Set Data	Bit Devices										Word Devices								Constant		Pointer	Digit designation	Index
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n																			○	○			

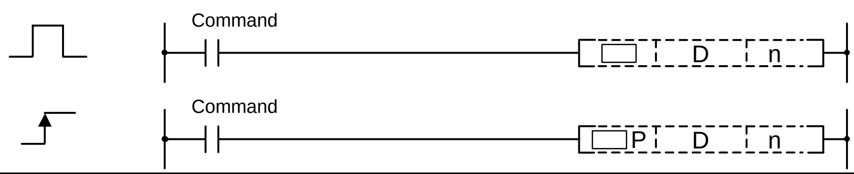
Extended instruction mode	

Set Data	Bit Devices										Word Devices								Constant		Pointer	Digit designation	Index
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	

[Instruction Symbol] [Execution Condition] ☐ indicates the signs ROL/RCL

ROL, RCL

ROLP, RCLP



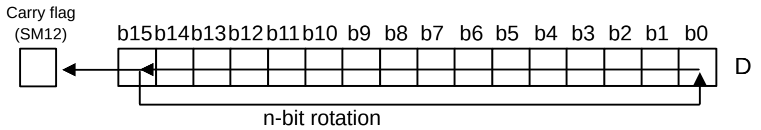
Set Data

Set Data	Meaning	Data Type
D	Initial number of devices to perform rotation	BIN 16 bits
n	Number of rotations (0 to 15)	

Functions

ROL

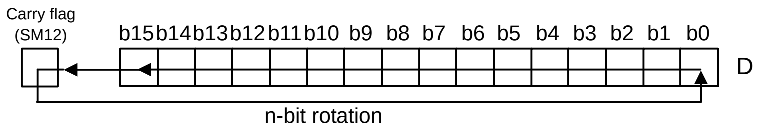
(1) Rotates the 16-bit data of the device designated at "D", not including the carry flag, n-bits to the left. The carry flag goes ON or OFF depending on its status prior to the execution of ROL instruction.



(2) Specify any of 0 to 15 as n. If the value specified as n is 16 or greater, operations will be as follows:
- When n is 16, the value becomes the one when 16-bit rotation was executed.
- When n is 17 or above, the value of D becomes indefinite.

RCL

(1) Rotates the 16-bit data of the device designated by D , including the carry flag, n-bits to the left. The carry flag goes ON or OFF depending on its status prior to the execution of RCL instruction.



(2) Specify any of 0 to 15 as n. If the value specified as n is 16 or greater, operations will be as follows:
- When n is 16, the value becomes the one when 16-bit rotation was executed.
- When n is 17 or above, the value of D becomes indefinite.

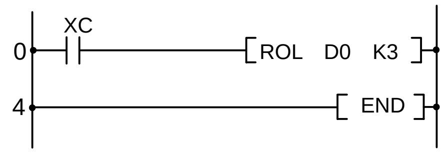
Operation Errors

(1) There are no operation errors associated with the ROL(P) or RCL(P) instructions.

Program Example

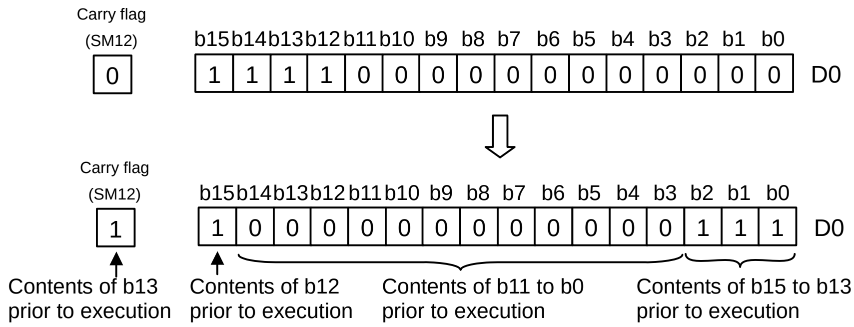
(1) The following program rotates the contents of D0, not including the carry flag, 3 bits to the left when XC is ON.

[Ladder Mode]



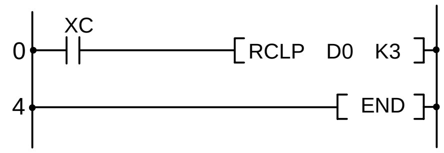
[List Mode]

Steps	Inst.	Device
0	LD	XC
1	ROL	D0
		K3
4	END	



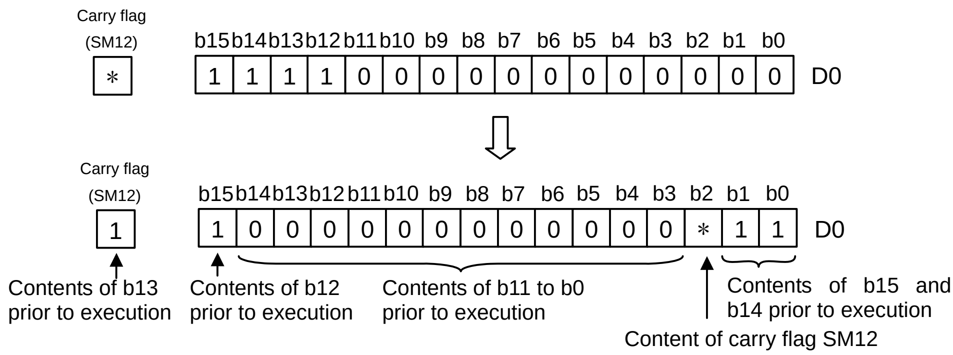
(2) The following program rotates the contents of D0, including the carry flag, 3 bits to the left when XC is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XC
1	RCLP	D0
		K3
4	END	



*The carry flag goes ON or OFF depending on its status prior to the execution of RCL.

○ DROR,DRORP,DRCR,DRCRP … Right rotation of 32-bit data

Compatible instruction mode	Usable instruction: **DROR, DRCR**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n																			○	○			

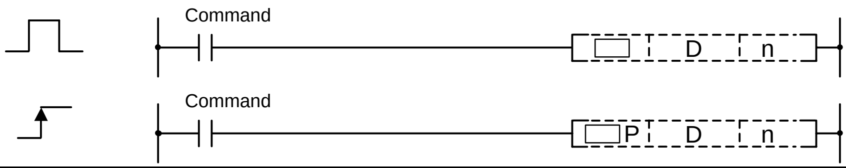
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	

[Instruction Symbol] [Execution Condition] ☐ indicates the signs DROR/DRCR

DROR, DRCR

DRORP, DRCRP



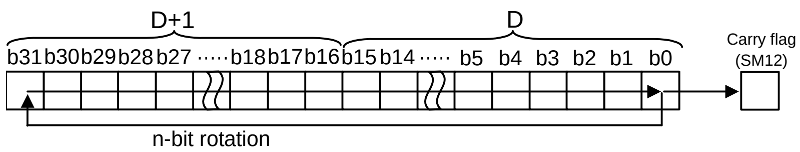
Set Data

Set Data	Meaning	Data Type
D	First device number of devices to perform rotation	BIN 32 bits
n	Number of rotations (0 to 31) 2 words (32 bits) are used for a word device.	BIN 32 bits

Functions

DROR

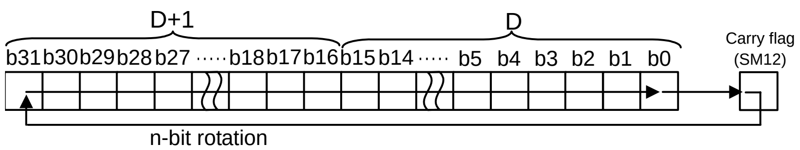
(1) The 32-bit data of the device designated at "D", not including the carry flag, is rotated n-bits to the right.
The carry flag goes ON or OFF depending on its status prior to the execution of the DROR instruction.



(2) Specify any of 0 to 31 as n. If the value specified as n is 32 or greater, the values of D+1 and D become indefinite.

DRCR

(1) Rotates 32-bit data, including carry flag, at device designated by D n bits to the right.
The carry flag goes ON or OFF depending on its status prior to the execution of the DRCR instruction.



(2) Specify any of 0 to 31 as n. If the value specified as n is 32 or greater, the values of D+1 and D become indefinite.

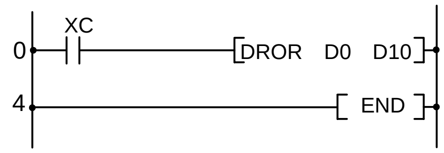
Operation Errors

(1) There are no errors associated with DROR(P) or DRCR(P) instructions.

Program Example

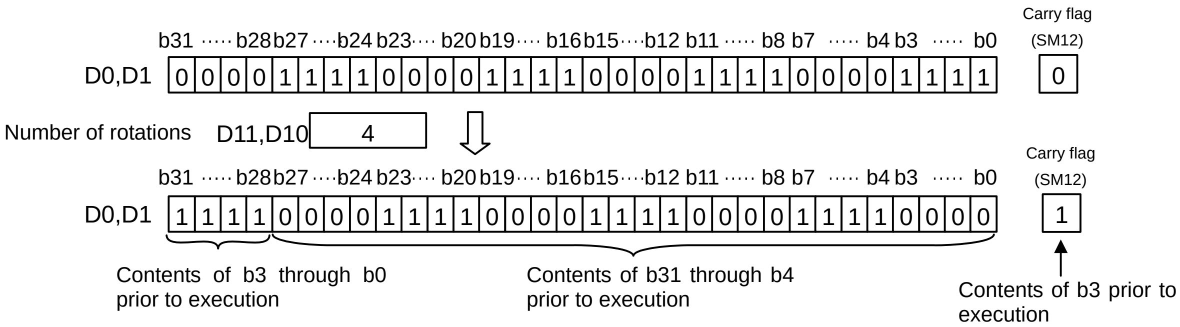
(1) The following program rotates the contents of D0 and D1, not including the carry flag, 4 bits to the right when XC is ON.

[Ladder Mode]



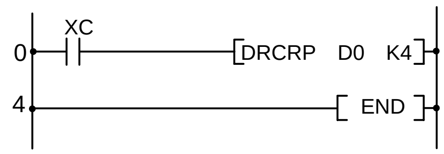
[List Mode]

Steps	Inst.	Device
0	LD	XC
1	DROR	D0
		D10
4	END	



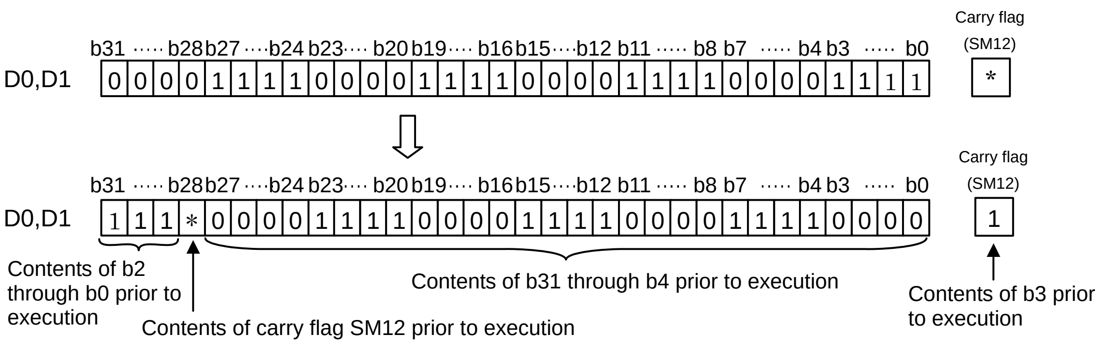
(2) The following program rotates the contents of D0 and D1, including the carry flag, 4 bits to the right when XC is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XC
1	DRCRP	D10
		K4
4	END	



* The carry flag goes ON or OFF depending on its status prior to the execution of the DRCR instruction.

○ DROL,DROLP,DRCL,DRCLP … Left rotation of 32-bit data

Compatible instruction mode — Usable instruction: **DROL, DRCL**

Set Data	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P	Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
D											○	○	○	○	○	○		○					
n																			○	○			

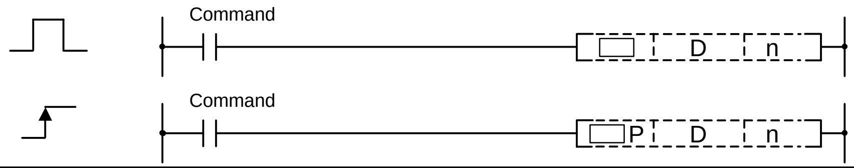
Extended instruction mode

Set Data	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P	Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
D											○	○	○	○	○	○		○					
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	

[Instruction Symbol] [Execution Condition] ▭ indicates the signs DROL/DRCL

DROL, DRCL

DROLP, DRCLP



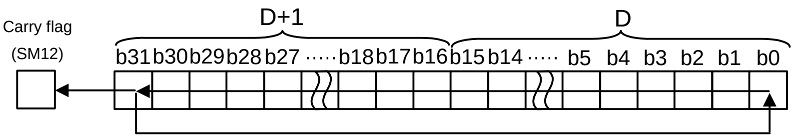
Set Data

Set Data	Meaning	Data Type
D	First device number of devices to perform rotation	BIN 32 bits
n	Number of rotations (0 to 31) 2 words (32 bits) are used for a word device.	BIN 32 bits

Functions

DROL

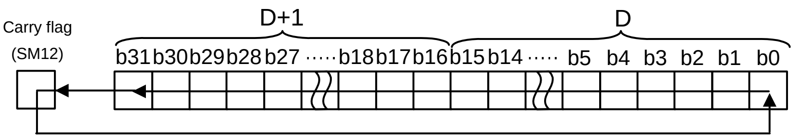
(1) The 32-bit data of the device designated at "D", not including the carry flag, is rotated n-bits to the left. The carry flag goes ON or OFF depending on its status prior to the execution of the DROL instruction.



(2) Specify any of 0 to 31 as n. If the value specified as n is 32 or greater, the values of D+1 and D become indefinite.

DRCL

(1) Rotates 32-bit data, including carry flag, at device designated by D n bits to the left. The carry flag goes ON or OFF depending on its status prior to the execution of the DRCL instruction.



(2) Specify any of 0 to 31 as n. If the value specified as n is 32 or greater, the values of D+1 and D become indefinite.

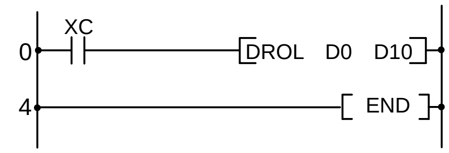
Operation Errors

(1) There are no operation errors associated with the DROL(P) or DRCL(P) instructions.

Program Example

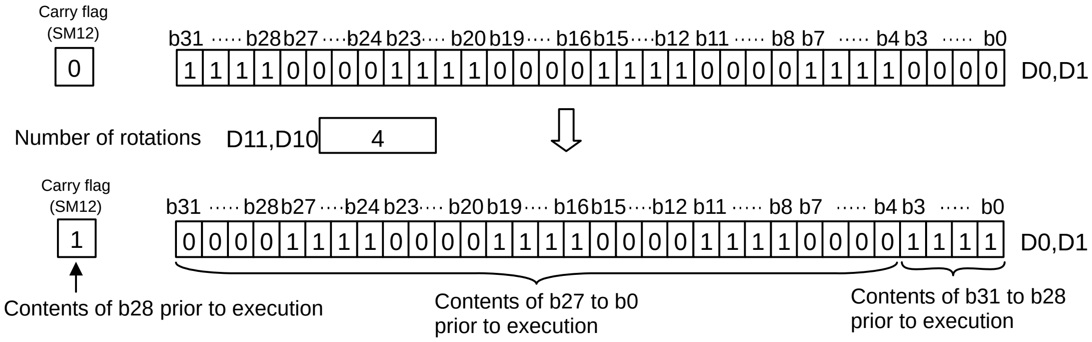
(1) The following program rotates the contents of D0 and D1, not including the carry flag, 4 bits to the left when XC is ON.

[Ladder Mode]



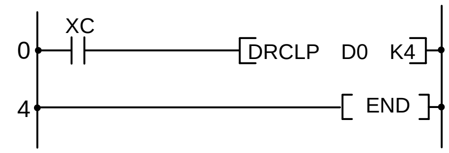
[List Mode]

Steps	Inst.	Device
0	LD	XC
1	DROL	D0
		D10
4	END	



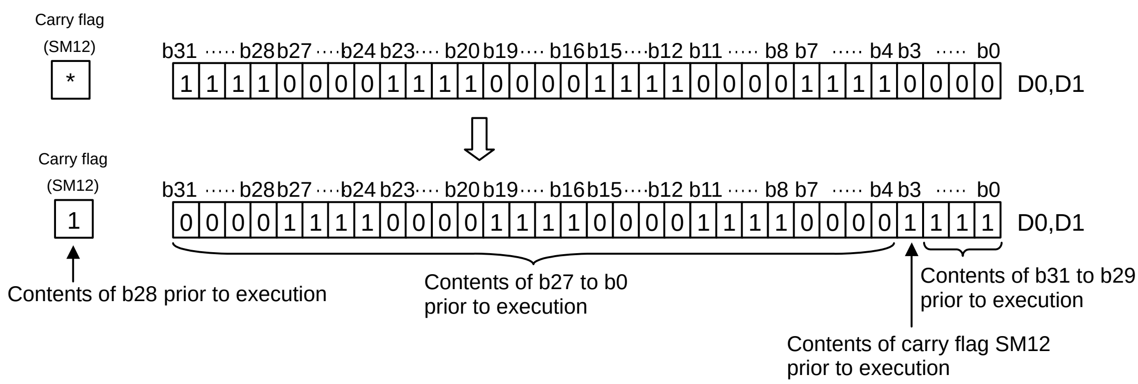
(2) The following program rotates the contents of D0 and D1, including the carry flag, 4 bits to the left when XC is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XC
1	DRCLP	D10
		K4
4	END	



* The carry flag goes ON or OFF depending on its status prior to the execution of the DRCL instruction.

○ SFR,SFRP,SFL,SFLP … n-bit shift to right or left of 16-bit data

Compatible instruction mode	Usable instruction: **SFR, SFL**

Set Data	Bit Devices X	Y	M	L	F	B	SB	T	SM	V	Word Devices T	C	D	R	W	SW	Z	SD	Constant K	H	Pointer P	Digit designation	Index
D											○	○	○	○	○	○		○					
n																			○	○			

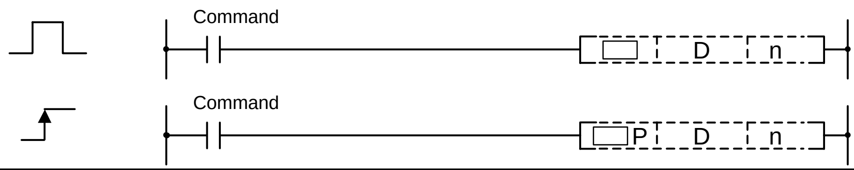
Extended instruction mode	

Set Data	Bit Devices X	Y	M	L	F	B	SB	T	SM	V	Word Devices T	C	D	R	W	SW	Z	SD	Constant K	H	Pointer P	Digit designation	Index
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○			

[Instruction Symbol] [Execution Condition] ▭ indicates the signs SFR/SFL

SFR, SFL

SFLP, SFLP

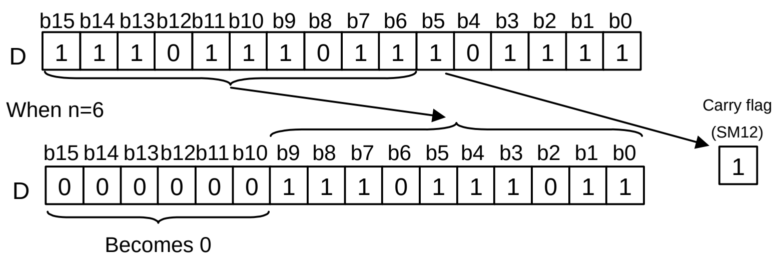


Set Data

Set Data	Meaning	Data Type
D	First device number of devices where shift data is being stored	BIN 16 bits
n	Number of shifts (0 to 15)	

Functions

SFR (1) Causes a shift to the right by n bits of the 16-bit data from the device designated at "D".

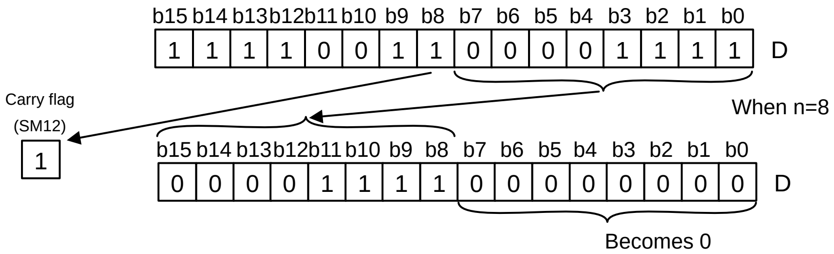


(2) A total of n bits from the upper level become 0.

(3) A shift by the data of the specified digit is conducted when a bit device has been designated at "D".
[See Program Example (1)]

(4) Specify any of 0 to 15 as n. If the value specified as n is 16 or greater, operations will be as follows:
- When n is 16, the value becomes the one when 16-bit shifting was executed.
- When n is 17, the value of D becomes indefinite.

SFL (1) Shifts 16-bit data at device designated by "D" n-bits to the left.



(2) The bits starting at n-bits from lowest bit become 0.

(3) A shift by the data of the specified digit is conducted when a bit device has been designated at "D".
[See Program Example (1)]

(4) Specify any of 0 to 15 as n. If the value specified as n is 16 or greater, operations will be as follows:
- When n is 16, the value becomes the one when 16-bit shifting was executed.
- When n is 17, the value of D becomes indefinite.

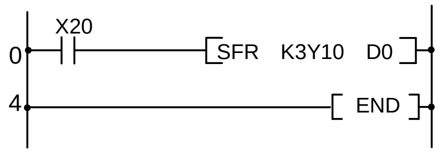
Operation Errors

(1) There are no operation errors associated with the SFR(P) or SFL(P) instructions.

Program Example

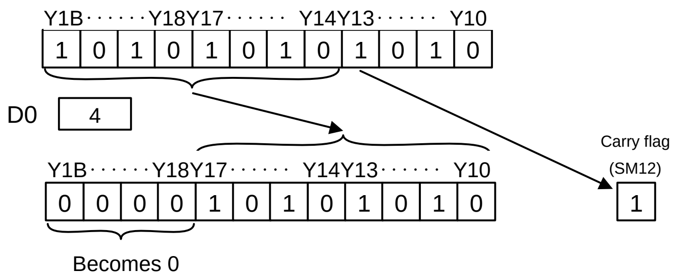
(1) The following program shifts the contents of Y10 to Y1B to the right by the number of bits designated by D0 when X20 goes ON.

[Ladder Mode]



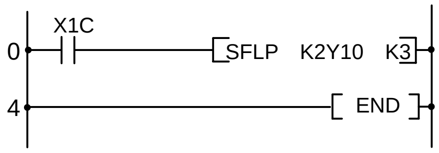
[List Mode]

Steps	Inst.	Device
0	LD	X20
1	SFR	K3Y10
		D0
4	END	



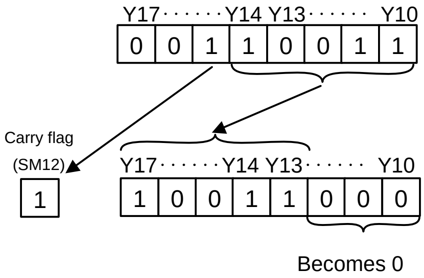
(2) The following program shifts the contents of X10 to X17 3 bits to the left when X1C is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XC
1	SFLP	K2Y10
		K3
4	END	



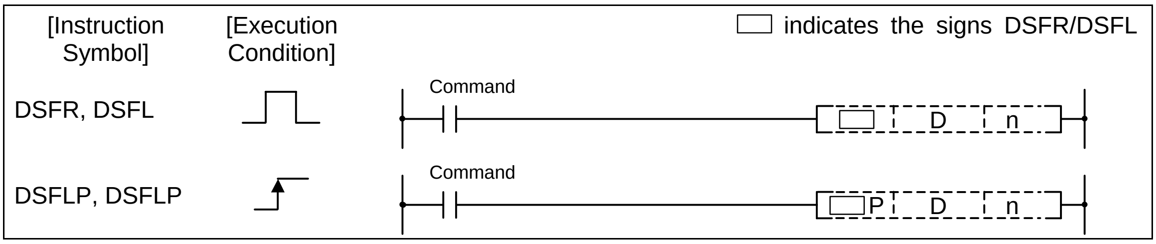
○ DSFR,DSFRP,DSFL,DSFLP … Right/Left shift of word device in batch

Compatible instruction mode	Usable instruction: **DSFR, DSFL**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n																			○	○			

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
D											○	○	○	○	○	○		○					
n																			○	○			

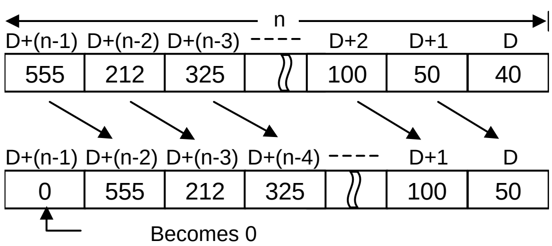


Set Data

Set Data	Meaning	Data Type
D	First device number of devices to shift	BIN 16 bits
n	Number of devices where shift will be conducted	

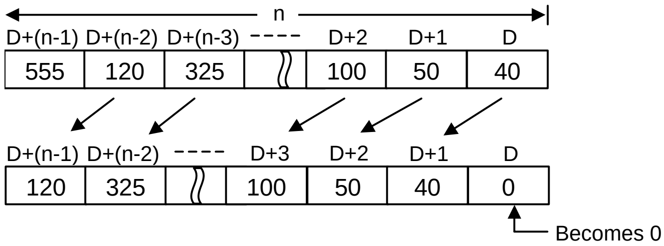
Functions

DSFR (1) Shifts data n points from device designated by "D" 1 word to the right.



(2) Device designated by "D"+(n-1) becomes 0.

DSFL (1) Shifts data n points from device designated by "D" 1 word to the left.



(2) Device designated by "D" becomes 0.

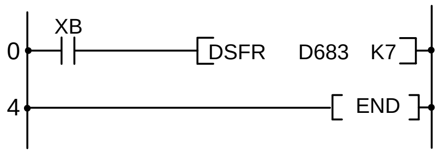
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- A part of range of the device n-points from the "D" device exceeds the relevant device. (Error code: 82)

Program Example

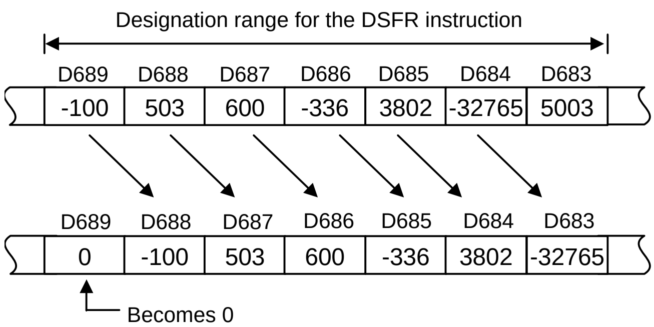
(1) The following program shifts the contents of D683 to D689 to the right when XB is ON.

[Ladder Mode]



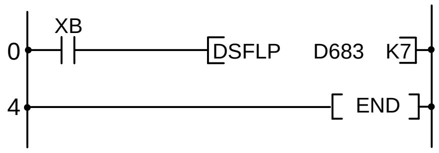
[List Mode]

Steps	Inst.	Device
0	LD	XB
1	DSFR	D683
		K7
4	END	



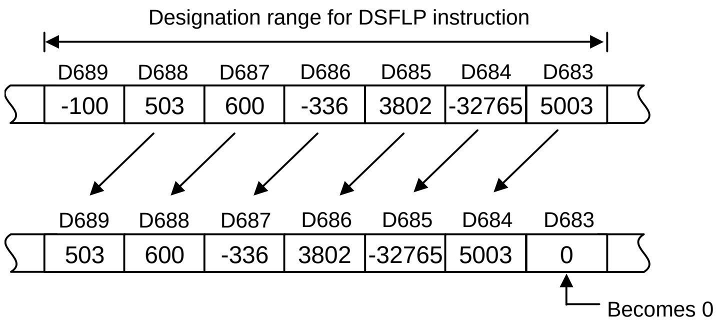
(2) The following program shifts the contents of D683 to D689 to the left when XB is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XB
1	DSFLP	D683
		K7
4	END	



○ SER,SERP,DSER,DSERP … 16-bit and 32-bit data searches

Compatible instruction mode	Usable instruction: **SER**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1											○	○	○	○	○	○		○					
S2											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					
n																			○	○			

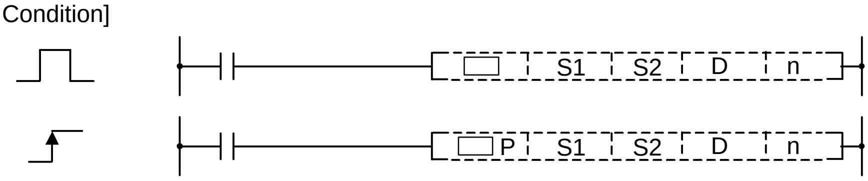
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S1	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
S2											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					
n	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○	○	○		○	

[Instruction Symbol] [Execution Condition] ☐ indicates the signs SER/DSER

SER, DSER

SERP, DSERP

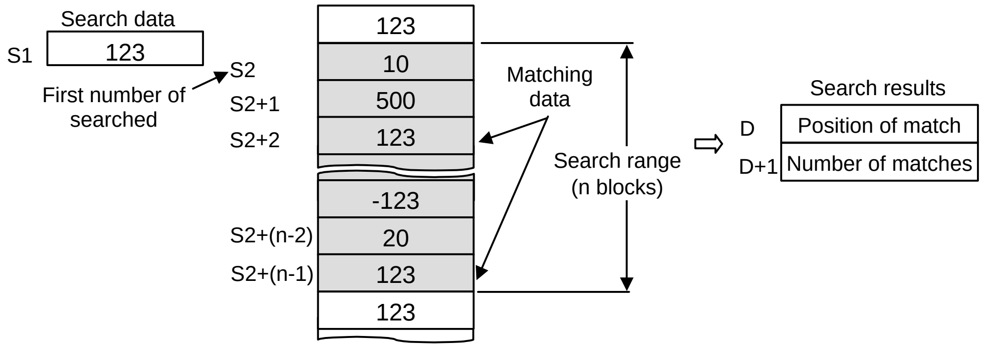


Set Data

Set Data	Meaning	Data Type
S1	First device number of the devices where search data is being stored	Word
S2	First number of the device where data which is an object of search is being stored.	
D	First device number of devices which will store search results	
n	Number of searches	

Functions

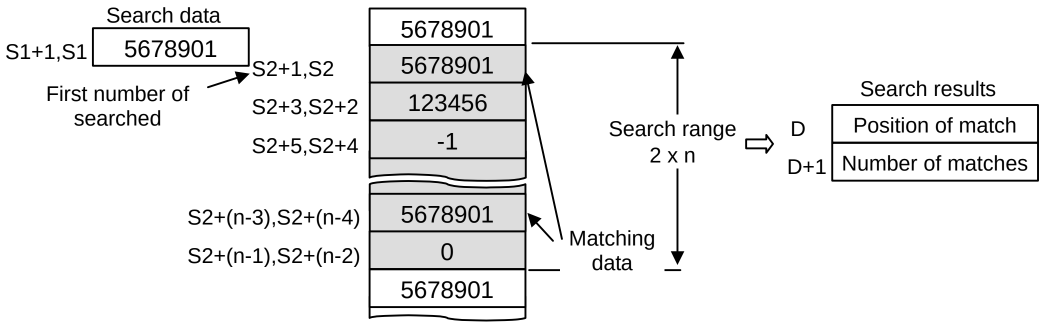
SER (1) Takes the 16-bit data of the device designated at "S1" as the entry code to search for n number of blocks from the 16-bit data from the device designated at "S2".
The number of matches with the entry code is stored at the device designated by "D"+1, and the relative value of the number of points that the device where the first match was found is from S2 is stored at the device designated by "D".



(2) No processing is conducted if n is 0 or a negative value.

(3) If no matches are found in the search, the devices designated at "D" and "D" +1 become "0".

DSER (1) Takes the 32-bit data of the device designated at "S1+1", "S1" as the entry code to search for n number of blocks (for 2 x n points) in 32-bit unit from the device designated at "S2"
The number of matches with the entry code is stored at the device designated by "D"+1, and the relative value of the number of points that the device where the first match was found is from S2 is stored at the device designated by "D".



(2) No processing is conducted if n is 0 or a negative value.

(3) If no matches are found in the search, the devices designated at "D" and "D" +1 become "0".

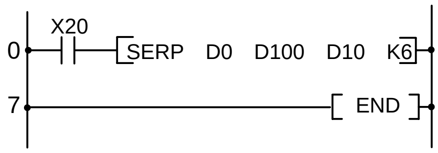
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- When the area of n number of blocks from the device designated at "S2" does not exist. (Error code: 82)

Program Example

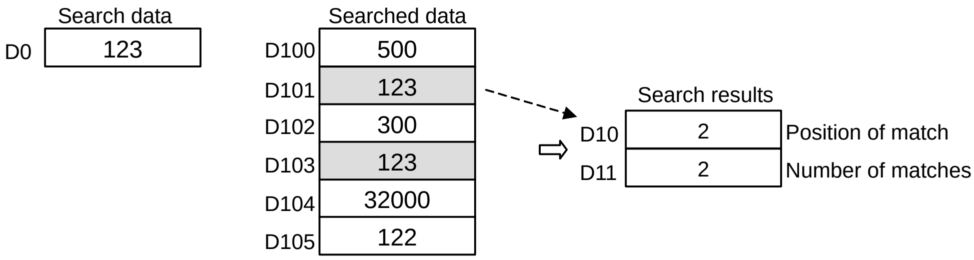
(1) The following program searches D100 to D105 for the contents of D0 when X20 is ON, and stores the search results at D10 and D11.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X20
1	SERP	D0
		D100
		D10
		K6
7	END	

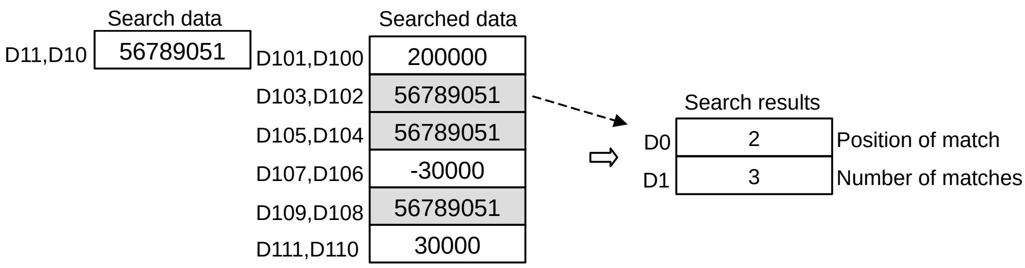


(2) The following program searches D100 to D111 for the contents of D11 and D10 when X20 is ON, and stores the search results at D0 and D1.

[Ladder Mode]

[List Mode]

Steps	Inst.	Device
0	LD	X20
1	DSERP	D10
		D100
		D0
		K6
7	END	



○ SUM,SUMP,DSUM,DSUMP … 16-bit and 32-bit data bit check

Compatible instruction mode	Usable instruction: **SUM**

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					

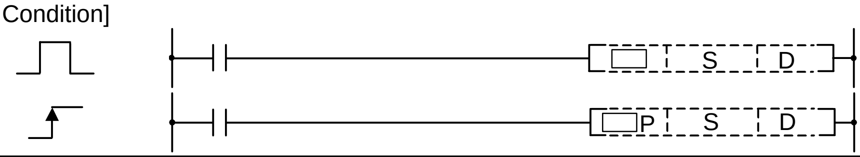
Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol] [Execution Condition] ▭ indicates the signs SUM/DSUM

SUM, DSUM

SUMP, DSUMP



Set Data

Set Data	Meaning	Data Type
S	First device number of devices that will count total number of bits that are at 1	BIN 16/32 bits
D	First device number of devices that will store total number of bits	

Functions

SUM

(1) From the 16-bit data in the device designated by "S", stores the total number of bits that are at 1, in the device designated by "D".

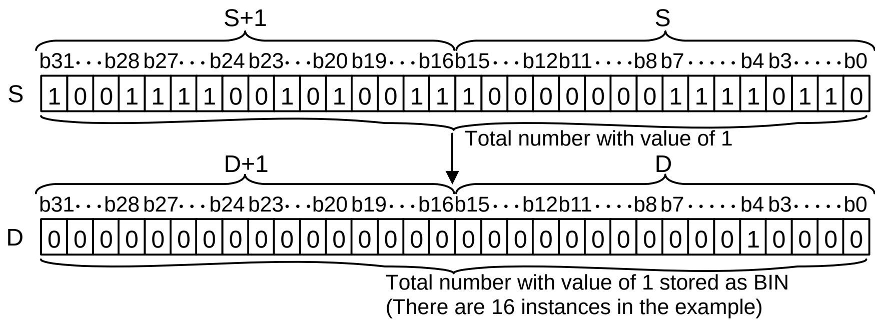
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
S	1	1	0	0	1	0	1	1	0	0	1	1	0	0	0	1

↓ Total number with value of 1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Total number with value of 1 stored as BIN
(There are 8 instances in the example)

DSUM (1) From the 32-bit data in the device designated by "S", stores the total number of bits that are at 1, in the device designated by "D".



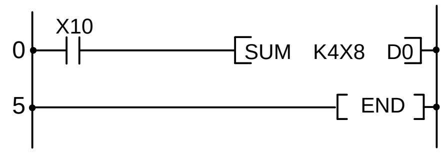
Operation Errors

(1) There are no operation errors associated with the SUM(P) or DSUM(P) instructions.

Program Example

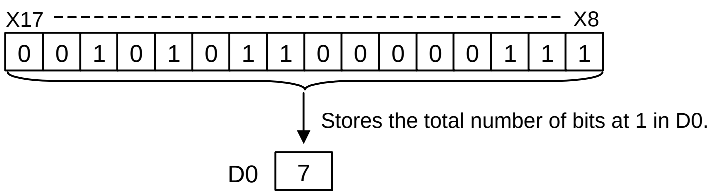
(1) The following program stores the number of bits from X8 to X17 which are ON when X10 is ON at D0.

[Ladder Mode]



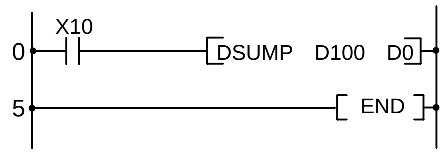
[List Mode]

Steps	Inst.	Device
0	LD	X10
1	SUM	K4X8
		D0
5	END	



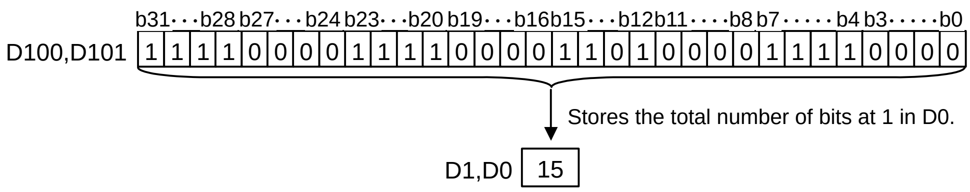
(2) The following program stores the number of bits from D100 and D101 which are ON when X10 is ON at D0.

[Ladder Mode]



[List Mode]

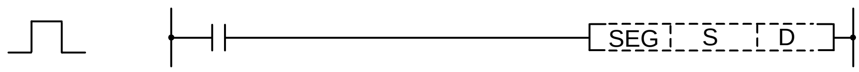
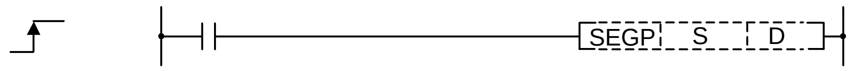
Steps	Inst.	Device
0	LD	X10
1	DSUMP	D100
		D0
5	END	



○ SEG,SEGP … 7-segment decode

Compatible instruction mode	Usable instruction: **SEG**																				
Set Data	Usable Devices																			Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					

Extended instruction mode																							
Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					

[Instruction Symbol]	[Execution Condition]	
SEG		
SEGP		

Set Data

Set Data	Meaning	Data Type
S	Device number of devices where decoded data is being stored	BIN 16 bits
D	First device number of devices where decoding results are stored	

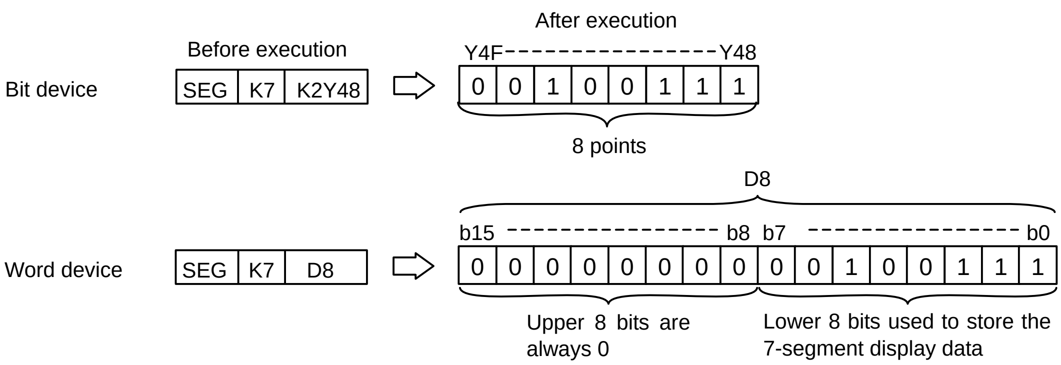
Functions

SEG

(1) Decodes the data from 0 to F designated by the lower 4 bits of S to 7-segment display data, and stores at D.

(2) If "D" is a bit device, indicates the first number of the devices storing the 7-segment display data; if it is a word device, indicates the number of the device that is storing the data.

(3) Storage is done as follows for bit devices and word devices:

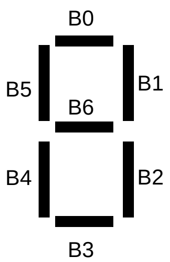


(4) Refer to the next page for the 7-segment display.

Operation Errors

(1) There are no operation errors associated with the SEG(P) instruction.

7-segment decode display

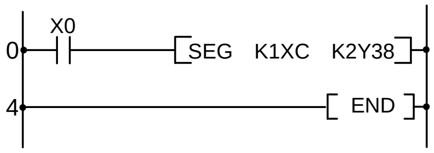
S		Configuration of 7 Segments	D								Display Data
Hex.	Bit Pattern		B7	B6	B5	B4	B3	B2	B1	B0	
0	0000		0	0	1	1	1	1	1	1	
1	0001		0	0	0	0	0	1	1	0	
2	0010		0	1	0	1	1	0	1	1	
3	0011		0	1	0	0	1	1	1	1	
4	0100		0	1	1	0	0	1	1	0	
5	0101		0	1	1	0	1	1	0	1	
6	0110		0	1	1	1	1	1	0	1	
7	0111		0	0	1	0	0	1	1	1	
8	1000		0	1	1	1	1	1	1	1	
9	1001		0	1	1	0	0	1	1	1	
A	1010		0	1	1	1	0	1	1	1	
B	1011		0	1	1	1	1	1	0	0	
C	1100		0	0	1	1	1	0	0	1	
D	1101		0	1	0	1	1	1	1	0	
E	1110		0	1	1	1	1	0	0	1	
F	1111		0	1	1	1	0	0	0	1	

First number of bit device
Lowest bit of word device

Program Example

(1) The following program converts the data from XC to XF when X0 is ON to 7-segment display data and outputs it to Y38 to Y3F.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X0
1	SEG	K1XC
		K2Y38
4	END	

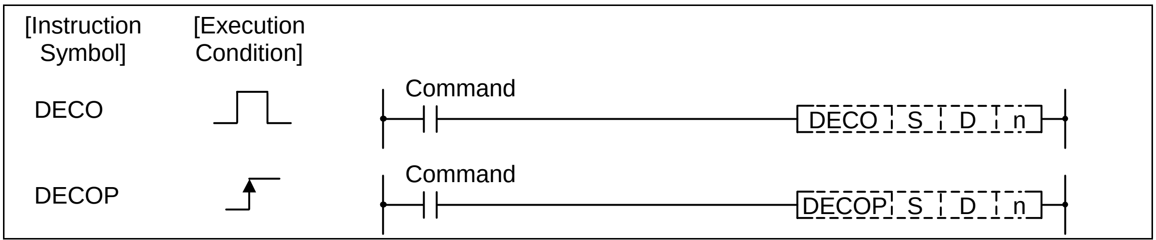
○ DECO,DECOP … Decoding from 8 to 256 bits

Compatible instruction mode	Usable instruction: **DECO**

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					
n																			○	○			

Extended instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○					
n																			○	○			



Set Data

Set Data	Meaning	Data Type
S	Number of device where decoded data is stored	BIN 16 bits
D	First device number of devices where decoding results are stored	Device name
n	Valid bit length (1 to 8)	BIN 16 bits

Functions

DECO

(1) Decodes the lower n-bits of the device designated by "S", and stores the results at a location 2n-bits from the device designated by "D".

(2) The value of n can be designated between 1 and 8.

(3) No processing is conducted if n=0, and there are no changes in data for the device designated at "D". When n is 9 or above, the device data specified by D becomes indefinite.

(4) The bit devices specified by D are treated as 1 bit, and the word devices specified by D are treated as 16 bits.

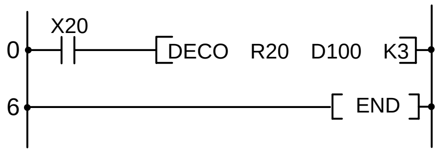
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- When the area of a part of 2n bit blocks from the device designated at D does not exist. (Error code: 82)

Program Example

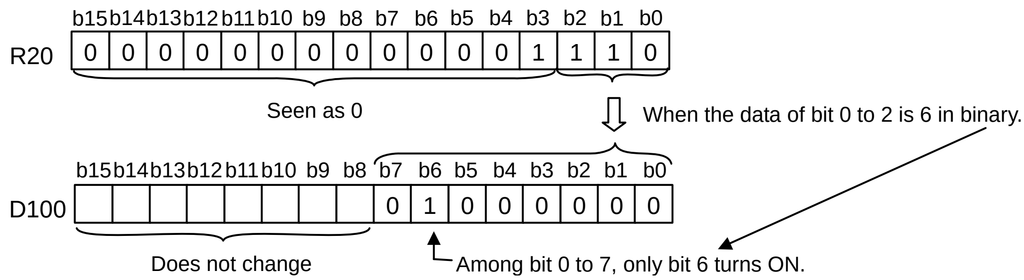
(1) The following program decodes 3 bits of R20 bit 0 to 3 when X20 turns ON and turns D100 corresponding bit to ON.

[Ladder Mode]



[List Mode]

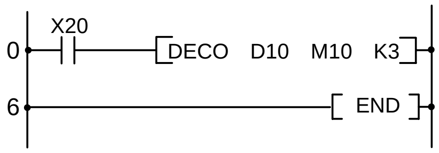
Steps	Inst.	Device
0	LD	X20
1	DECO	R20
		D200
		K3
6	END	



Note 1) When R20 b0 to b2 is 0, D100 bit 0 turns ON.
Note 2) D100 data remains the same even when X20 turns OFF.

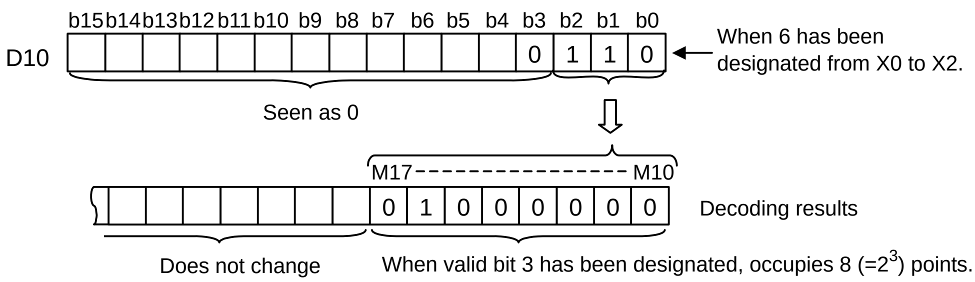
(2) The following program decodes the 3 bits from D10 bit 0 to 2 and stores the results at M10 when X20 is ON.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X20
1	DECO	D10
		M10
		K3
6	END	

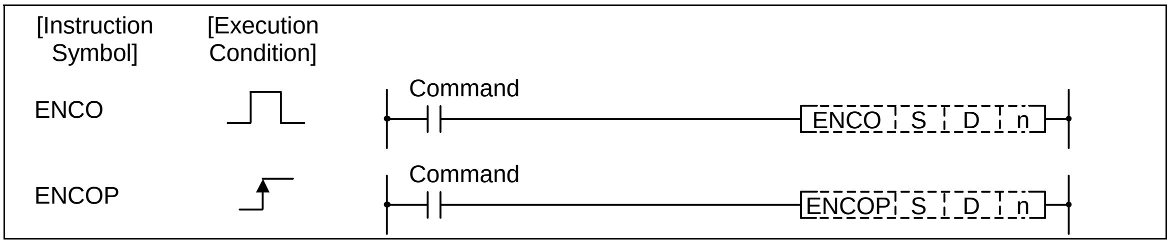


○ ENCO,ENCOP … Encoding from 256 to 8 bits

Compatible instruction mode	Not available

Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S	○	○	○	○	○	○	○		○		○	○	○	○	○	○		○				○	
D											○	○	○	○	○	○		○					
n																			○	○			



Set Data

Set Data	Meaning	Data Type
S	Number of devices where encoded data is being stored	BIN 16 bits
D	Number of device where encoding results will be stored	
n	Valid bit length (1 to 8)	

Functions

ENCO

(1) 2n bit of the device specified by S is encoded and the result is stored in the lower n bit from the device specified by D as encode data.

(2) The value of n can be designated between 1 and 8.

(3) When n=0, the device data specified by D does not change due to no operation. When n is 9 or above, the device data specified by D becomes indefinite.

(4) As for the devices designated at D, bit devices are treated as 1 bit, and word devices as 16 bits.

(5) If more than 1 bit is at 1, processing will be conducted at the upper bit location.

(6) When the 2n bit blocks data is all 0, "0" is stored in D.

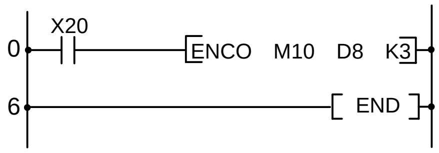
Operation Errors

(1) In the following cases an operation error occurs, the error flag (SM0) turns ON, and an error code is stored at SD0.
- When the area of a part of 2n bit blocks from the device designated at S does not exist. (Error code: 82)

Program Example

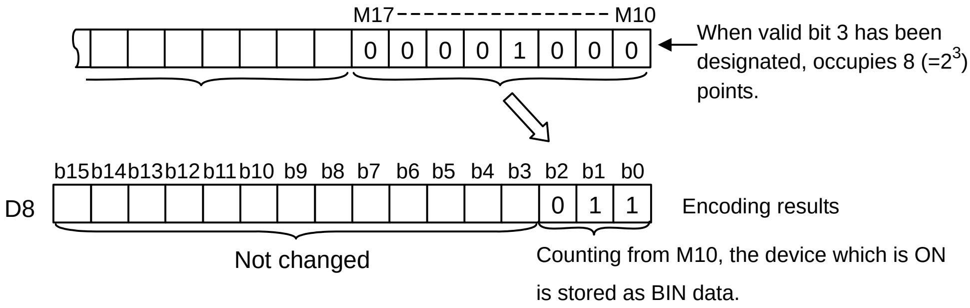
(1) The following program encodes the 3 bits from M10 when X20 is ON, and stores the results at D8.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	X20
1	ENCO	M10
		D8
		K3
6	END	

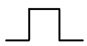


○ S.AVE … Average value calculation

Compatible/Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
D											○	○	○	○	○	○		○					
n																			○	○			

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

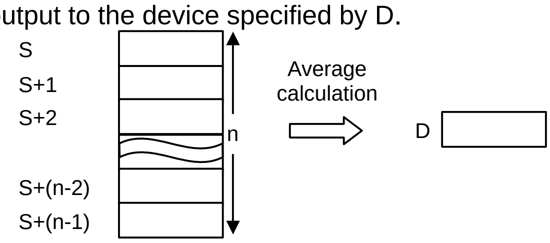
[Instruction Symbol]	[Execution Condition]	
S.AVE		

Set Data

Set Data	Meaning	Data Type
S	Head No. of the device whose average is calculated	BIN 16 bits
D	Device No. at the output destination	
n	Average value	

Functions

S.AVE (1) The average of n points of devices counting from the device specified by S is calculated and output to the device specified by D.



(2) The device specified by S is calculated with signs.

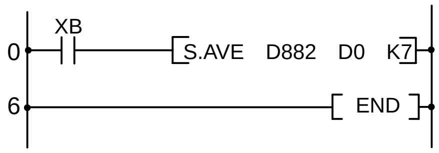
Operation Errors

(1) There are no operation errors associated with the S.AVE instruction.

Program Example

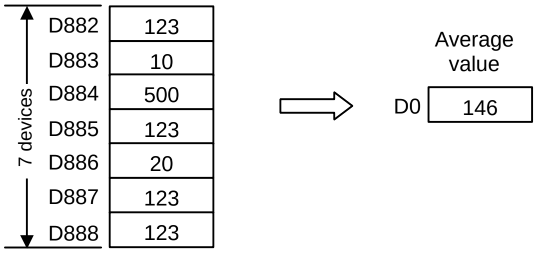
(1) In this program the average of D882 to D888 is calculated when XB turns ON, and the result is output to D0.

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	XB
1	S.AVE	D882
		D0
		K7
5	END	



Note) Digits after the decimal point is truncated.

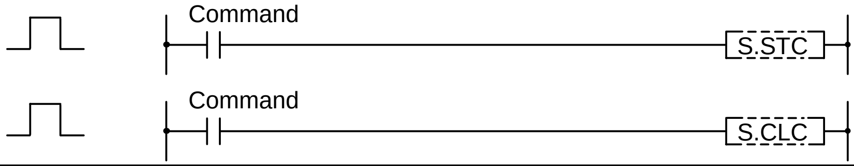
○ S.STC,S.CLC … Carry flag set/reset

Compatible/Extended instruction mode	

Set Data	Usable Devices																					Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		

The compatible instruction mode and extended instruction mode share the same specifications for this instruction.

[Instruction Symbol]	[Execution Condition]	
S.STC		
S.CLC		



Functions

S.STC (1) Set (turn ON) the carry flag contact (SM12).

S.CLC (1) Reset (turn OFF) the carry flag contact (SM12).

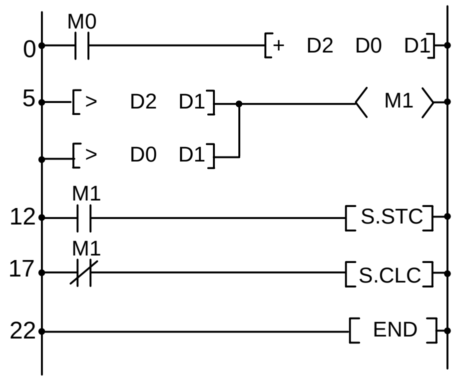
Operation Errors

(1) There are no operation errors associated with the S.STC or S.CLC instructions.

Program Example

(1) In this program, D0 data is added to D2 data when M0 turns ON. The carry flag (SM12) is turned ON if the result exceeds 32767, and turned OFF if 32767 or less.

[Ladder Mode]



Add D2 to D0, and store the result in D1. (D2+D0→D1)

Turn M1 ON when:
(Summand)>(Addition result) or
(Augend)>(Added result)

Turn ON the carry flag when M1 turns ON

Turn OFF the carry flag when M1 turns OFF

[List Mode]

Steps	Inst.	Device
0	LD	M0
1	+	D2
		D0
		D1
5	LD>	D2
		D1
8	OR>	D0
		D1
11	OUT	M1
12	LD	M1
13	S.STC	
17	LDI	M1
18	S.CLC	
22	END	

○ LDBIT,ANDBIT,ORBIT … Bit test with "A" contact handling

Compatible instruction mode																								
Set Data	Usable Devices																						Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer			
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P			
S											○	○	○	○	○	○		○						
n																			○	○				

The compatible instruction mode is available for this instruction; however since this mode will be discontinued in the future, replacing with the instruction having the same function is strongly recommended.

Extended instruction mode	Not available

⚠ CAUTION

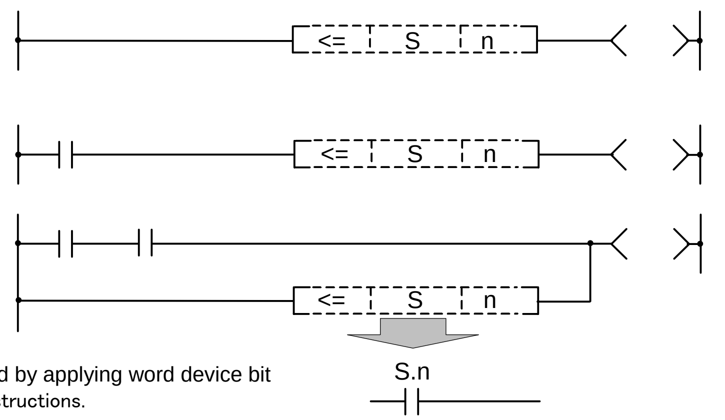
Note that when a sequence program using these instructions is executed in the extended instruction mode, unexpected operation would occur because the instructions will be processed as ones having different operations.

[Instruction Symbol] [Execution Condition]

LDBIT

ANDBIT

ORBIT



Above instructions can be replaced by applying word device bit designation to LD, AND, and OR instructions.

Set Data

Set Data	Meaning	Data Type
S	Device No. to which bit test is performed	BIN 16 bits
n	Bit to which bit test is performed	

Functions

(1) Bit test for 16-bit device is executed with "A" contact handling.

(2) Results of bit test are as shown below.

Condition	Result
Tested bit = 1	Continuity
Tested bit = 0	Non-continuity

Replaceable instructions

LDBIT, ANDBIT, and ORBIT instructions can be replaced by executing a word device bit designation using LD, AND, OR instructions. (Refer to Program Example)

Instruction	Example of replacement by word device bit designation
LDBIT	(Example: LD<= D10 K3 → LD D10.3)
ANDBIT	(Example: AND<= D10 K10 → AND D10.A)
ORBIT	(Example: OR<= D10 HF → OR D10.F)

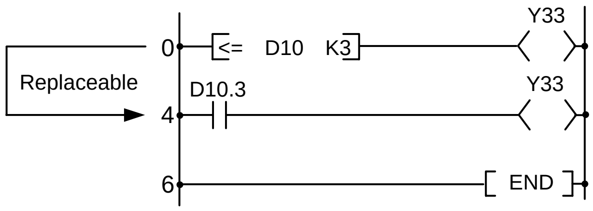
Operation Errors

(1) There are no operation errors associated with the LDBIT, ANDBIT, or ORBIT instructions.

Program Example

(1) Program to test D10 bit 3
(The 0th step and 4th step are the ladders having the same operations.)

[Ladder Mode]

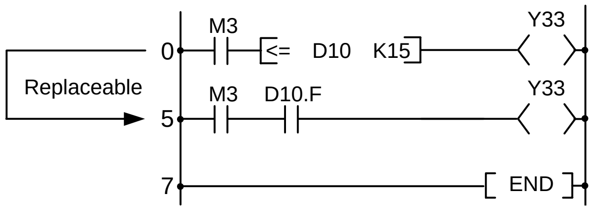


[List Mode]

Steps	Inst.	Device
0	LD<=	D10
		K3
3	OUT	Y33
4	LD	D10.3
5	OUT	Y33
6	END	

(2) Program to test D10 bit 15
(The 0th step and 5th step are the ladders having the same operations.)

[Ladder Mode]



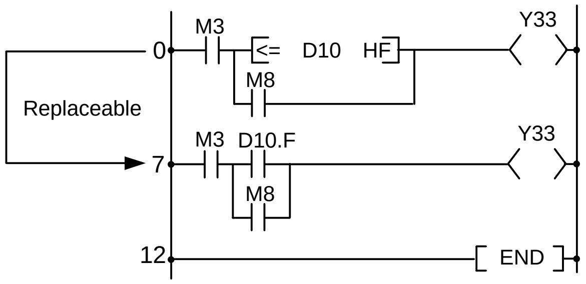
[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND<=	D10
		K15
4	OUT	Y33
5	LD	M3
6	AND	D10.F
7	OUT	Y33
8	END	

(3) Program to tes D10 bit 15

(The 0th step and 7th step are the ladders having the same operations.)

[Ladder Mode]



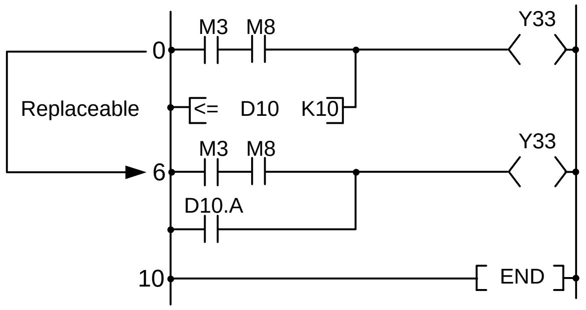
[List Mode]

Steps	Inst.	Device
0	LD	M3
1	LD<=	D10
		HF
4	OR	M8
5	ANB	
6	OUT	Y33
7	LD	M3
8	LD	D10.F
9	OR	M8
10	ANB	
11	OUT	Y33
12	END	

(4) Program to test D10 bit 10

(The 0th step and 6th step are the ladders having the same operations.)

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND	M8
2	OR<=	D10
		K10
5	OUT	Y33
6	LD	M3
7	AND	M8
8	OR	D10.A
9	OUT	Y33
10	END	

○ LDBII,ANDBII,ORBII … Bit test with "B" contact handling

Compatible instruction mode	

Set Data	Usable Devices																				Digit designation	Index
	Bit Devices										Word Devices								Constant		Pointer		
	X	Y	M	L	F	B	SB	T	SM	V	T	C	D	R	W	SW	Z	SD	K	H	P		
S											○	○	○	○	○	○		○					
n																			○	○			

The compatible instruction mode is available for this instruction; however since this mode will be discontinued in the future, replacing with the instruction having the same function is strongly recommended.

Extended instruction mode	Not available

⚠ CAUTION

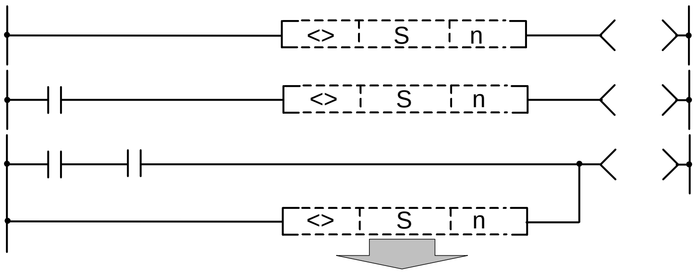
Note that when a sequence program using these instructions is executed in the extended instruction mode, unexpected operation would occur because the instructions will be processed as ones having different operations.

[Instruction Symbol] [Execution Condition]

LDBII

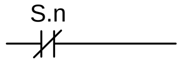
ANDBII

ORBII



Above instructions can be replaced by applying word device bit designation to LDI, ANI, ORI instructions.

S.n



Set Data

Set Data	Meaning	Data Type
S	Device No. to which bit test is performed	BIN 16 bits
n	Bit to which bit test is performed	

Functions

(1) Bit test for 16-bit device is executed with "B" contact handling.

(2) Results of bit test are as shown below.

Condition	Result
Tested bit = 0	Continuity
Tested bit = 1	Non-continuity

Replaceable instructions

LDBII, ANDBII, and ORBII instructions can be replaced by executing a word device bit designation using LDI, ANI, and ORI instructions. (Refer to Program Example)

Instruction	Example of replacement by word device bit designation
LDBII	(Example: LD<> D10 K3 → LDI D10.3)
ANDBII	(Example: AND<> D10 K10 → ANI D10.A)
ORBII	(Example: OR<> D10 HF → ORI D10.F)

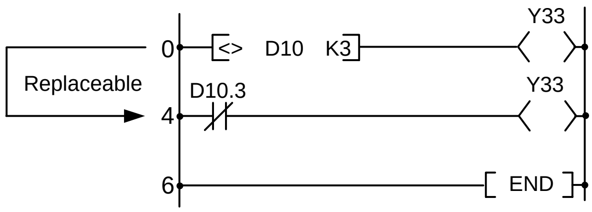
Operation Errors

(1) There are no operation errors associated with the LDBII, ANDBII, or ORBII instructions.

Program Example

(1) Program to test D10 3
(The 0th step and 4th step are the ladders having the same operations.)

[Ladder Mode]

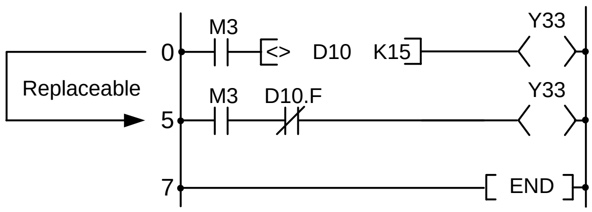


[List Mode]

Steps	Inst.	Device
0	LD<>	D10
		K3
3	OUT	Y33
4	LDI	D10.3
5	OUT	Y33
6	END	

(2) Program to test D10 bit 15
(The 0th step and 5th step are the ladders having the same operations.)

[Ladder Mode]

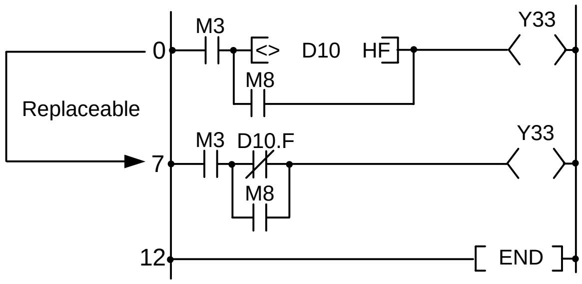


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND<>	D10
		K15
4	OUT	Y33
5	LD	M3
6	ANI	D10.F
7	OUT	Y33
8	END	

(3) Program to test D10 bit 15
(The 0th step and 7th step are the ladders having the same operations.)

[Ladder Mode]

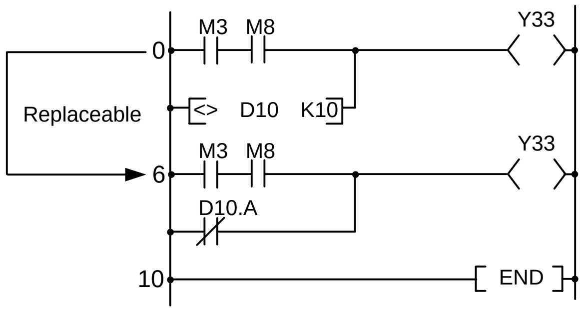


[List Mode]

Steps	Inst.	Device
0	LD	M3
1	LD<>	D10
		HF
4	OR	M8
5	ANB	
6	OUT	Y33
7	LD	M3
8	LDI	D10.F
9	OR	M8
10	ANB	
11	OUT	Y33
12	END	

(4) Program to test D10 bit 10
(The 0th step and 6th step are the ladders having the same operations.)

[Ladder Mode]



[List Mode]

Steps	Inst.	Device
0	LD	M3
1	AND	M8
2	OR<>	D10
		K10
5	OUT	Y33
6	LD	M3
7	AND	M8
8	ORI	D10.A
9	OUT	Y33
10	END	

9. Exclusive Instructions

PLC dedicated instruction is limited to its application for the processes which may be difficult to perform only with basic instruction or function instruction.

PLC dedicated instructions include:

(1) ATC dedicated instruction (ATC)

With this instruction, ATC, such as magazine index control, tool change by arm, etc. is performed.
ATC dedicated instructions include:

- Tool No. search
- Tool change
- Tool table forward/reverse run
- Pointer (which displays magazine index position) forward/reverse run
- Tool data read/write, etc.

(2) Rotary body control instruction (ROT)

With this instruction, the rotary body's target position and rotation direction are determined, as well as the function as a ring counter is realized.
This is used when calculating the rotation direction or number of index steps of the magazine and turret, etc. based on the output data figured with tool No. search of ATC dedicated instruction, or used when controlling the rotary body position.

9.1 ATC Exclusive Instruction

9.1.1 Outline of ATC Control

The ATC (Automatic Tool Change) can be controlled in the following two ways:

(1) Mechanical random control

With the information of magazine position from the machine, and T command, the control method determines the direction of magazine rotation, number of steps, etc. for index of the magazine, according to the given command.
Each tool and magazine tool pot (socket) have a one-on-one corresponding relation.
Usually, the "intermediate pot" that supports the transfer of the tool is provided between the spindle and the magazine.
This control is possible by not using ATC instruction, but ROT instruction only.

(2) Memory random control

With the information of magazine rotation, or magazine position from the machine, the control method controls tool No. stored in the memory. For index of the magazine, the direction of magazine rotation and number of steps, etc., are determined by the given T command and tool No. stored in the memory.
Each tool and magazine tool pot (socket) does not always have a one-on-one corresponding relation.
Usually, the "intermediate pot" is not provided.

9.1.2 ATC Operation

The motions related to ATC operation can be largely divided into the following four motions:

(1) Index of magazine .... (ATC-K1, K2, K5, K6, K7, K8)
(2) Tool change (arm, or the like is used) .... (ATC-K3, K4)
(3) Transfer of tool to intermediate pot or arm .... (Normal function instructions such as MOV, XCH are used.)
(4) Others .... (ATC-K9, K10, K11)

9.1.3 Explanation of Terminology

(1) Pointer

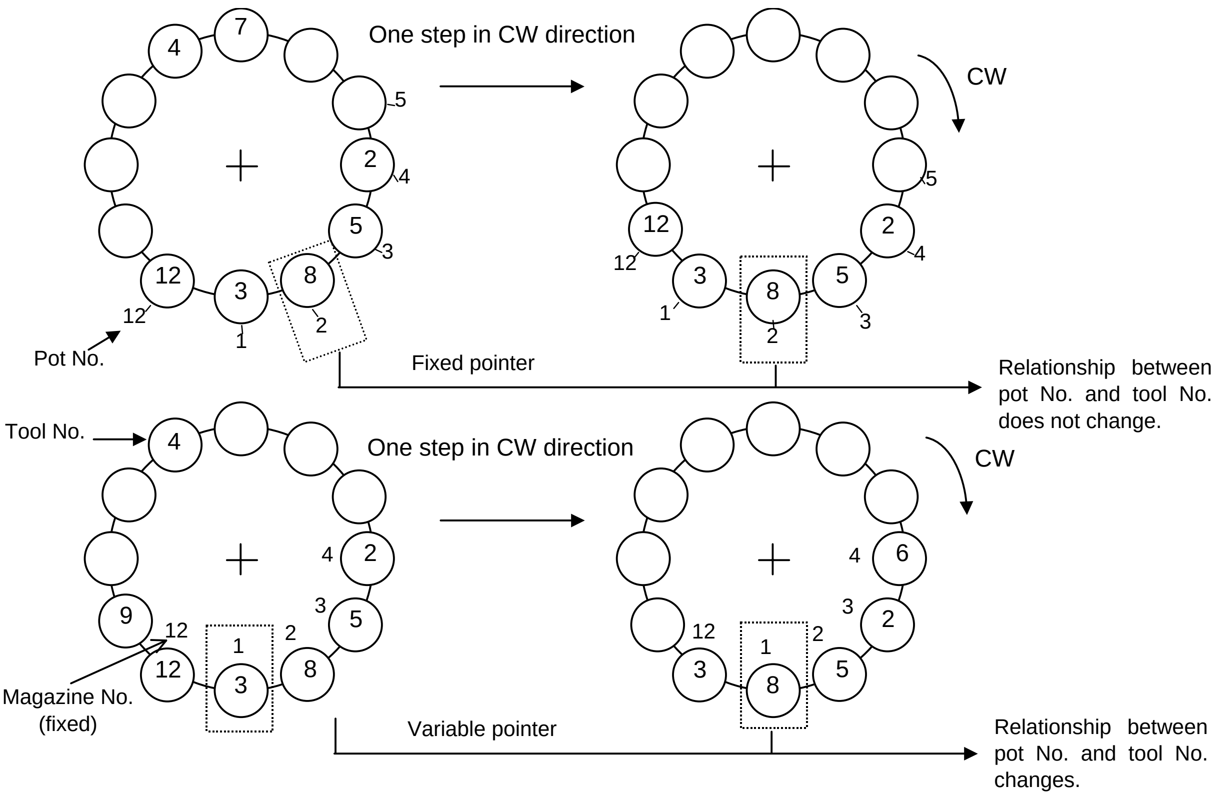
This points out the position where the magazine is indexed. When a tool table in which tool No. are previously recorded is used, the tool table does not rotate with rotation of the magazine and the pointer serves as "ring counter" for control of magazine position.

(2) Fixed pointer method

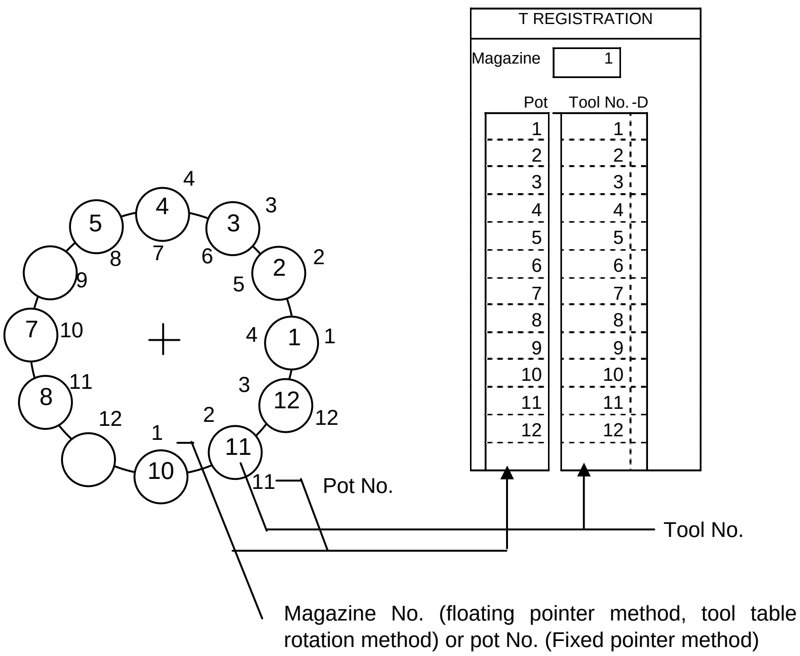
This is the type with tool pots numbered and the relationship between tool pot and tool No. is fixed if the magazine is rotated. When the tool table is rotated, fixed pointer does not functionally differ from variable pointer method.

(3) Variable pointer method

This is the type with numbered fixed position on magazine and the relationship between magazine No. and tool No. changes when the magazine rotates.



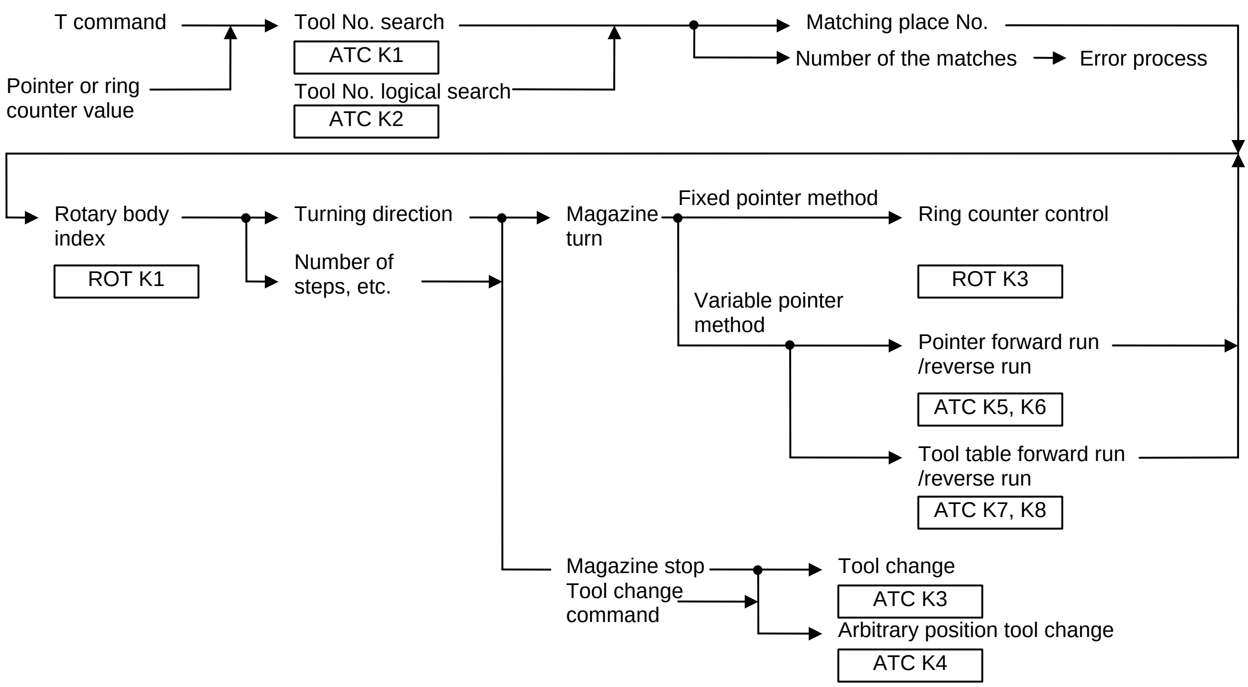
9.1.4 Relationship between Tool Registration Screen and Magazines



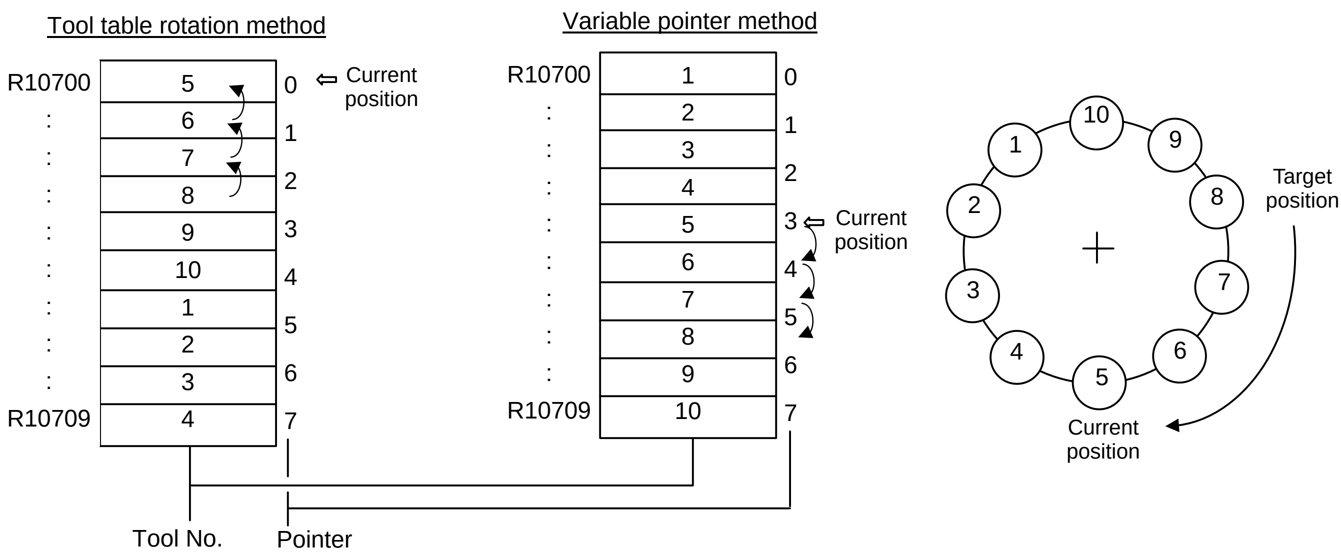
When the floating pointer method or tool table rotation method is selected on the tool registration screen, correspondence display between the magazines and tools changes each time the magazine rotates; when the fixed pointer method is selected, it does not change.

9.1.5 Use of ATC and ROT Instructions

The use order of the ATC and ROT instructions during the T command or tool change command is shown below:



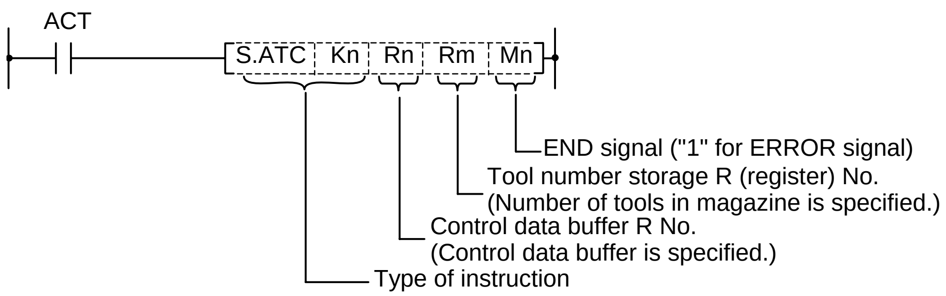
The relationship between the tool number search instruction and rotary body indexing instruction when the tool table rotation method or variable pointer method is used is explained below.

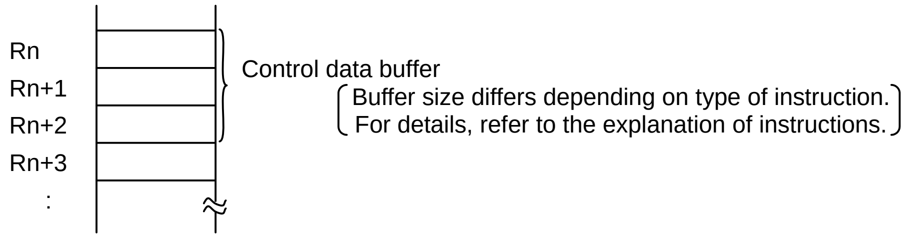


(1) When indexing tool number 8 in the situation given in the previous page
   (a) In the tool table rotation method, the tool number search instruction outputs 3.
   (b) In the variable pointer method, the tool number search instruction outputs 7.

(2) The tool number search instruction output result is used by the rotary body indexing instruction to find the rotation direction, the number of steps, etc.
   (a) In the tool table rotation system, rotation direction CW and number of steps 3 are found from the relationship between current value 0 (pointer 0) and tool number search output result 3.
   (b) In the variable pointer method, rotation direction CW and number of steps 3 are found from the relationship between current value 4 (pointer 4) and tool number search output result 7, as in (a) above.
   In the fixed pointer system, the pointer is fixed to 0 and the ring counter of 0 to n-1 (n is the number of magazines) separate from the pointer is controlled. The counter value is used as the current position.

9.1.6 Basic Format of ATC Exclusive Instruction





9.1.7 Instruction List

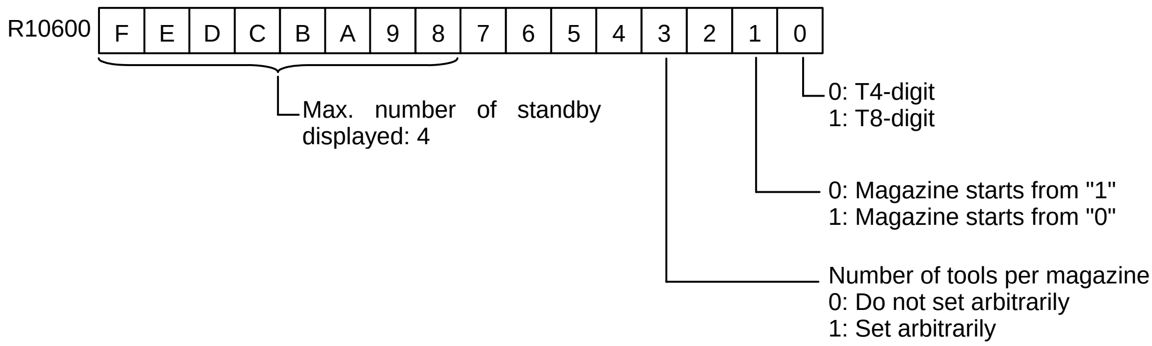
Instruction					Description
S.ATC	K1	Rn	Rm	Mn	Tool No. search
S.ATC	K2	Rn	Rm	Mn	Tool No. logical AND search
S.ATC	K3	Rn	Rm	Mn	Tool change
S.ATC	K4	Rn	Rm	Mn	Arbitrary position tool change
S.ATC	K5	Rn	Rm	Mn	Pointer forward run
S.ATC	K6	Rn	Rm	Mn	Pointer reverse run
S.ATC	K7	Rn	Rm	Mn	Tool table forward run
S.ATC	K8	Rn	Rm	Mn	Tool table reverse run
S.ATC	K9	Rn	Rm	Mn	Tool data read
S.ATC	K10	Rn	Rm	Mn	Tool data write
S.ATC	K11	Rn	Rm	Mn	Automatic tool data write

9.1.8 Control Data Buffer Contents

	Instruction	Rn	Rn+1	Rn+2
1	Tool No. search	R No. to store search data	R No. to which data output	-
2	Tool No. logical AND search	R No. to store search data	R No. to which data output	Logical AND data position R No.
3	Tool change (Ex.: Spindle↔ Indexposition)	R No. to specify the position of tool change	-	-
4	Arbitrary position tool change	R No. to specify the position of tool change	R No. to specify the tool to be changed	-
5	Pointer forward run	-	-	-
6	Pointer reverse run	-	-	-
7	Tool table forward run	-	-	-
8	Tool table reverse run	-	-	-
9	Tool data read	R No. for magazine position (to be read)	R No. to which data read	-
10	Tool data write	R No. for magazine position (to be written)	R No. to which data written	-
11	Automatic tool data write	R No. to store Initial data	-	-

9.1.9 File Register (R Register) Assignment and Parameters

(1) Control parameter contents



For details on the control parameters, refer to "Examples of Tool Registration Screen".

(2) Arbitrary setting of number of tools per magazine

This function allows the number of tools per magazine to be set freely.

(a) Do not set arbitrarily

The number of magazine rows is set to the maximum three rows, and the number of tools that can be registered is set to maximum 120 tools/magazine.
The magazine tool data assignment is fixed.

(b) To set arbitrarily

There is a maximum of five magazine rows, and the number of tools that can be registered is a total of 360 tools for all magazines.
Set the number of tools per magazine in the "number of magazine designation" register.
(If there are any magazines not being used, set the designation register to 0.)

"Number of magazine designation" register
No. 1 magazine ... R10610
No. 2 magazine ... R10611
No. 3 magazine ... R10612
No. 4 magazine ... R10613
No. 5 magazine ... R10614

The magazine tool data is assigned for the set tools in order from No. 1 magazine based on the set number of tools.

(3) File registers for ATC control

The file registers used with the ATC are as shown below.

	Corresponding file (R) register										Remarks (data type)
Magazine	No.1 magazine		No.2 magazine		No.3 magazine		No.4 magazine		No.5 magazine		
T4-digit/T8-digit specifications	T4-digit	T8-digit	T4-digit	T8-digit	T4-digit	T8-digit	T4-digit	T8-digit	T4-digit	T8-digit	
ATC control parameters	R10600	←	←	←	←	←	←	←	←	←	
Number of magazine designation	R10610	←	R10611	←	R10612	←	R10613	←	R10614	←	Binary
Pointer designation	R10615	←	R10616	←	R10617	←	R10618	←	R10619	←	Binary
Spindle tool	R10620	R10620 R10621	R10630	R10630 R10631	R10640	R10640 R10641	R10650	R10650 R10651	R10660	R10660 R10661	BCD
Standby 1 tool	R10621	R10622 R10623	R10631	R10632 R10633	R10641	R10642 R10643	R10651	R10652 R10653	R10661	R10662 R10663	BCD
Standby 2 tool	R10622	R10624 R10625	R10632	R10634 R10635	R10642	R10644 R10645	R10652	R10654 R10655	R10662	R10664 R10665	BCD
Standby 3 tool	R10623	R10626 R10627	R10633	R10636 R10637	R10643	R10646 R10647	R10653	R10656 R10657	R10663	R10666 R10667	BCD
Standby 4 tool	R10624	R10628 R10629	R10634	R10638 R10639	R10644	R10648 R10649	R10654	R10658 R10659	R10664	R10668 R10669	BCD
Spindle tool D	R10670	←	R10675	←	R10680	←	R10685	←	R10690	←	Binary
Standby 1 tool D	R10671	←	R10676	←	R10681	←	R10686	←	R10691	←	Binary
Standby 2 tool D	R10672	←	R10677	←	R10682	←	R10687	←	R10692	←	Binary
Standby 3 tool D	R10673	←	R10678	←	R10683	←	R10688	←	R10693	←	Binary
Standby 4 tool D	R10674	←	R10679	←	R10684	←	R10689	←	R10694	←	Binary
AUX data	R10604	←	←	←	←	←	←	←	←	←	Binary
Magazine tool data Magazine tool data (auxiliary D)	• When not setting arbitrarily The number of magazine rows is set to the maximum three rows, and the number of tools that can be registered is set to maximum 120 tools/magazine. The magazine tool data assignment is fixed. Refer to (a) on the next page for details. • When setting arbitrarily There is a maximum of five magazine rows, and the number of tools that can be registered is a total of 360 tools for all magazines. The assignment of the magazine tool data to each magazine varies between R10700 and R11779. Refer to the assignment examples in (b) on the next page for details on the assignment method.										

(a) Tool data assignment when not setting arbitrarily

There are up to three magazines, and up to 120 tools per magazine.
The tool data assignment is fixed between R10700 and R11779 as shown below.

		Corresponding file (R) register										Remarks (data type)
Magazine		**No.1 magazine**		**No.2 magazine**		**No.3 magazine**		**No.4 magazine**		**No.5 magazine**		
T4-digit/T8-digit specifications		**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	
Magazine tool data	Pot 1 (MG1)	R10700	R10700 R10701	R11060	R11060 R11061	R11420	R11420 R11421	-	-	-	-	BCD
	Pot 2 (MG2)	R10701	R10702 R10703	R11061	R11062 R11063	R11421	R11422 R11423	-	-	-	-	BCD
	Pot 3 (MG3)	R10702	R10704 R10705	R11062	R11064 R11065	R11422	R11424 R11425	-	-	-	-	BCD
	Pot 119 (MG119)	R10818	R10936 R10937	R11178	R11296 R11297	R11538	R11656 R11657	-	-	-	-	BCD
	Pot 120 (MG120)	R10819	R10938 R10939	R11179	R11298 R11299	R11539	R116568 R11659	-	-	-	-	BCD
Magazine tool data (auxiliary D)	Pot 1	R10940	←	R11300	←	R11660	←	-	-	-	-	Binary
	Pot 2	R10941	←	R11301	←	R11661	←	-	-	-	-	Binary
	Pot 3	R10942	←	R11302	←	R11662	←	-	-	-	-	Binary
	Pot 119	R11058	←	R11418	←	R11778	←	-	-	-	-	Binary
	Pot 120	R11059	←	R11419	←	R11779	←	-	-	-	-	Binary

(b) Example of tool data assignment when setting arbitrarily

There are up to five magazines, and a total of 360 tools for all magazines.
The tool data assignment varies between R10700 and R11779.
The data is assigned to the tools starting from the No. 1 magazine based on the set number of tools.

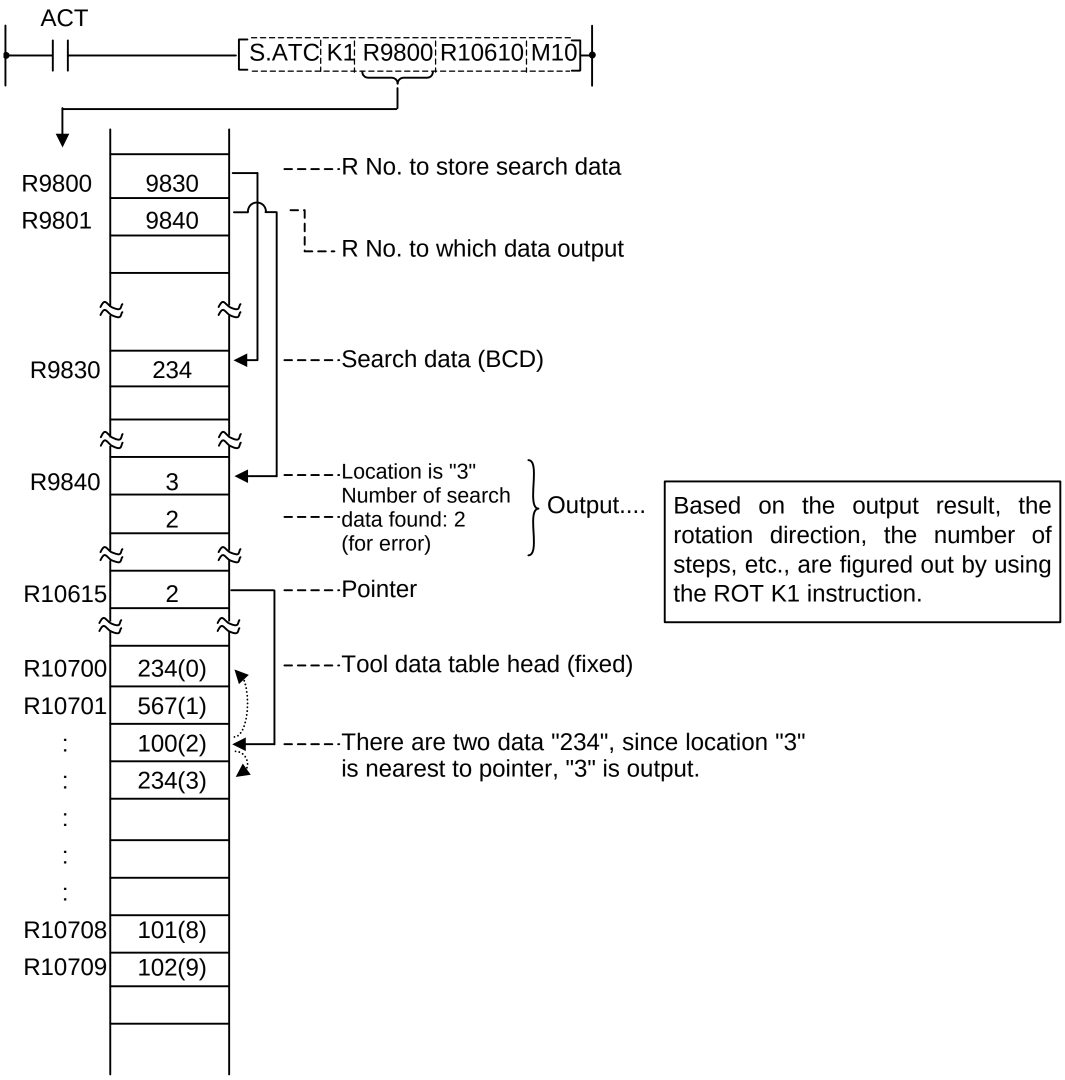
(Example) Number of magazines : 5 magazines
Number of tools
No. 1 magazine : 100 tools
No. 2 to No. 5 magazines : 50 tools each

		Corresponding file (R) register										
Magazine		**No.1 magazine**		**No.2 magazine**		**No.3 magazine**		**No.4 magazine**		**No.5 magazine**		**Remarks (data type)**
T4-digit/T8-digit specifications		**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	**T4-digit**	**T8-digit**	
Magazine tool data	Pot 1 (MG1)	R10700	R10700 R10701	R11000	R11000 R11001	R11150	R11150 R11151	R11300	R11300 R11301	R11450	R11450 R11451	BCD
	Pot 2 (MG2)	R10701	R10702 R10703	R11001	R11002 R11003	R11151	R11152 R11153	R11301	R11302 R11303	R11451	R11452 R11453	BCD
	Pot 3 (MG3)	R10702	R10704 R10705	R11002	R11004 R11005	R11152	R11154 R11155	R11302	R11304 R11305	R11452	R11454 R11455	BCD
	Pot 49 (MG49)	R10748	R10796 R10797	R11048	R11096 R11097	R11198	R11246 R11247	R11348	R11396 R11397	R11498	R11546 R11547	BCD
	Pot 50 (MG50)	R10749	R10798 R10799	R11049	R11098 R11099	R11199	R11248 R11249	R11349	R11398 R11399	R11499	R11548 R11549	BCD
	Pot 99 (MG99)	R10798	R10896 R10897	-	-	-	-	-	-	-	-	BCD
	Pot 100 (MG100)	R10799	R10898 R10899	-	-	-	-	-	-	-	-	BCD
Magazine tool data (auxiliary D)	Pot 1	R10900	←	R11100	←	R11250	←	R11400	←	R11550	←	Binary
	Pot 2	R10901	←	R11101	←	R11251	←	R11401	←	R11551	←	Binary
	Pot 3	R10902	←	R11102	←	R11252	←	R11402	←	R11552	←	Binary
	Pot 49 (MG49)	R10948	←	R11148	←	R11298	←	R11448	←	R11598	←	Binary
	Pot 50 (MG50)	R10949	←	R11149	←	R11299	←	R11449	←	R11599	←	Binary
	Pot 99 (MG99)	R10998	←	-	-	-	-	-	-	-	-	Binary
	Pot 120 (MG120)	R10999	←	-	-	-	-	-	-	-	-	Binary

9.1.10 Details of Each Instruction

(1) Tool No. search

This instruction is used to search for tool No. stored in the magazine (tool data table).
When the instruction tool No. is found, number of searched data and its location are output. If two or more tool No. are found, the location of tool No. nearest to the pointer is output.



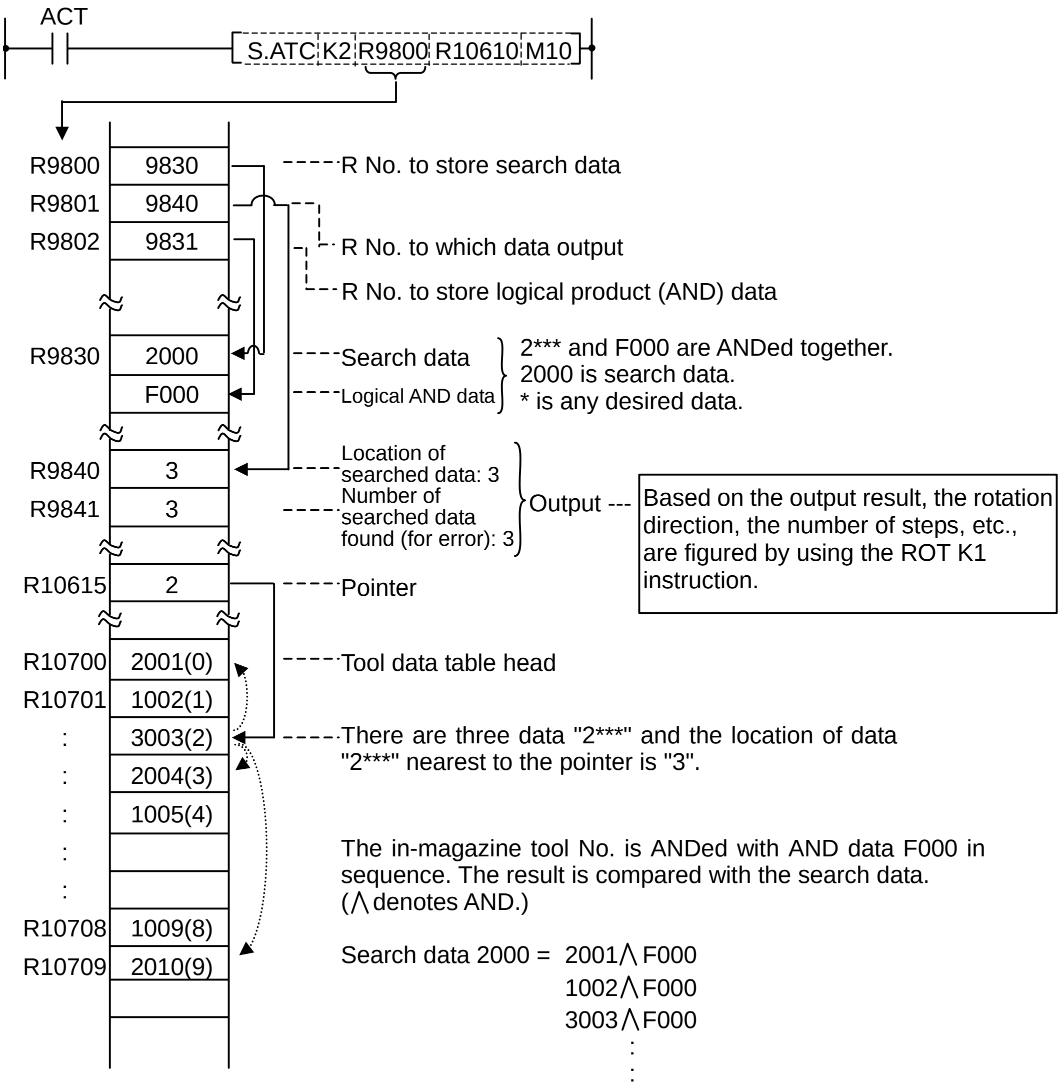
(Example for 10 magazines)

(Note 1) Pointer and location are counted up, like 0,1,2...9, in the tool data table, starting from the tool data table head.
(Note 2) When pointer is not used, R10615 should be set to "zero".

Example: [MOV K0 R10615]

(2) Tool No. logical product (AND) search

Tool number AND search is the same as the tool number search instruction (ATC K1) in function: search data and in-magazine tool number and AND data are ANDed together for a search.



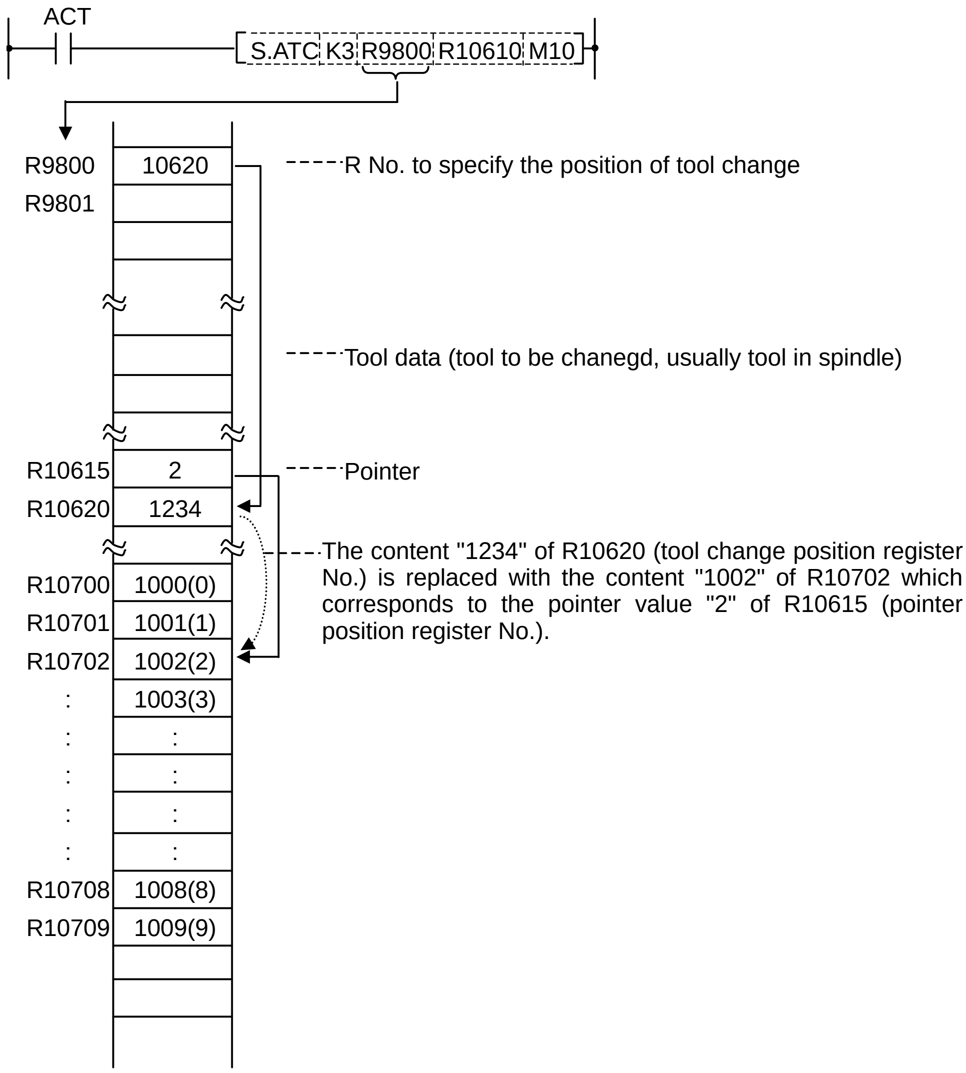
(Note 1) Pointer and location are counted up, like 0, 1, 2 .... 9, in the tool data table, starting from the tool data table head.

(Note 2) When pointer is not used, R10615 should be set to "zero".

(Example) MOV K0 R10615

(3) Tool change

When a spindle tool and a magazine index tool are exchanged by the ATC arm, etc., the contents in the memory (R register) must be updated correspondingly.

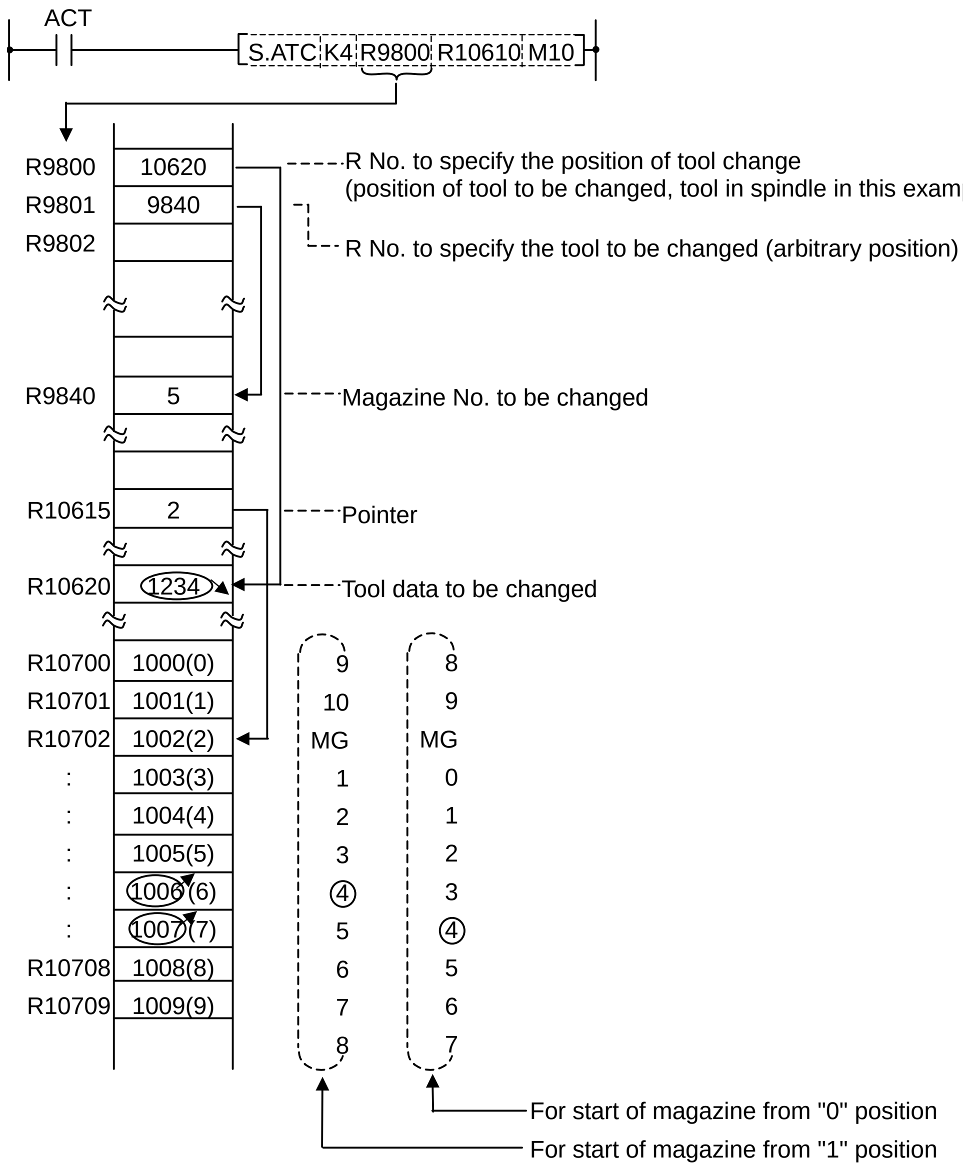


(Note) When pointer is not used, R10615 should be set to "zero".

(Example) MOV K0 R10615

(4) Arbitrary position tool change

In tool change, a spindle tool is usually exchanged with a magazine index tool. It may often occur, however, that tool change must be performed at a station other than the usual tool change position (tool change at spare tool change position, for example). This instruction is used in such cases.



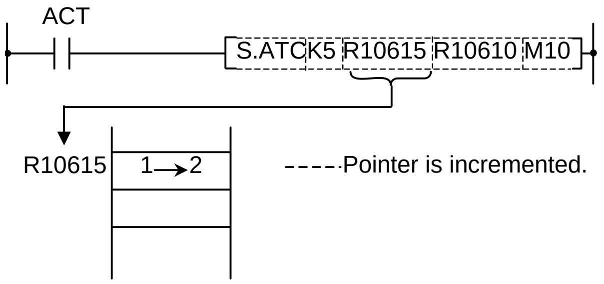
(Note 1) Tool change position differs depending on whether magazine No. starts with "0" or "1".
However, the substantial consequence does not differ.

(Note 2) When pointer is not used, R10615 should be set to "zero".

(Example) MOV K0 R10615

(5) Pointer forward run

In the ATC control with variable pointer, pointer count is controlled so that it coincides with the actually indexed magazine position when the magazine rotates in forward direction for index.

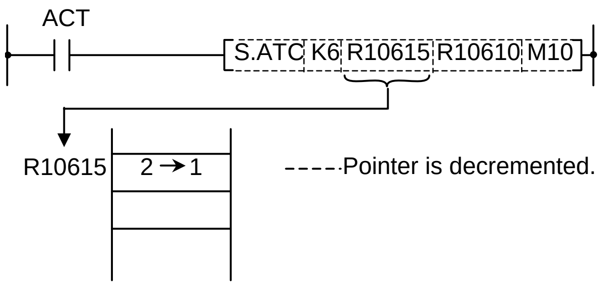


When a magazine with 10 tools is used, for example, the control sequence is as follows:
0, 1, 2, 3 .... 9, 0, 1, 2, ... 8, 9, 0, 1 ...

(Note 1) When this instruction is executed, the relationship between magazine No. and tool No., appearing on the tool entry display, changes accordingly.

(6) Pointer reverse run

In the ATC control with variable pointer, pointer count is controlled so that it coincides with actually indexed magazine position when the magazine rotates in reverse direction for index.

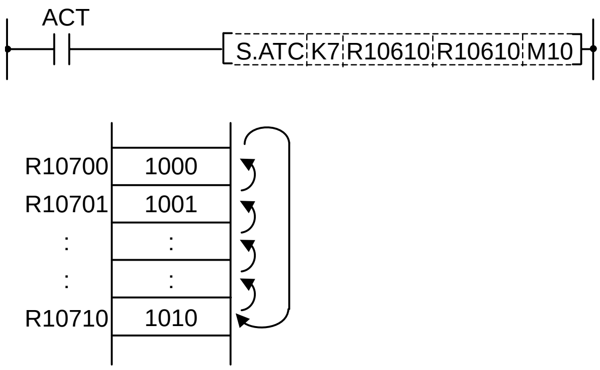


When a magazine with 10 tools is used, for example, the control sequence is as follows:
2, 1, 0, 9, 8 .... 2, 1, 0, 9, 8 .... 1, 0, 9, 8 ...

(Note 1) When this instruction is executed, the relationship between magazine No. and tool No., appearing on the tool entry display, changes accordingly.

(7) Tool table forward run

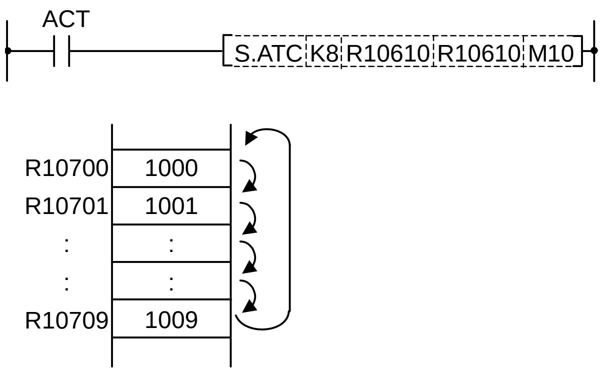
The tool table rotates in forward direction in accordance with the magazine rotation.



(Note 1) In this control mode, pointer always indicates "0" (tool table head).
(Note 2) When this instruction is executed, the relationship between magazine No. and tool No., appearing on the tool entry display changes accordingly.

(8) Tool table reverse run

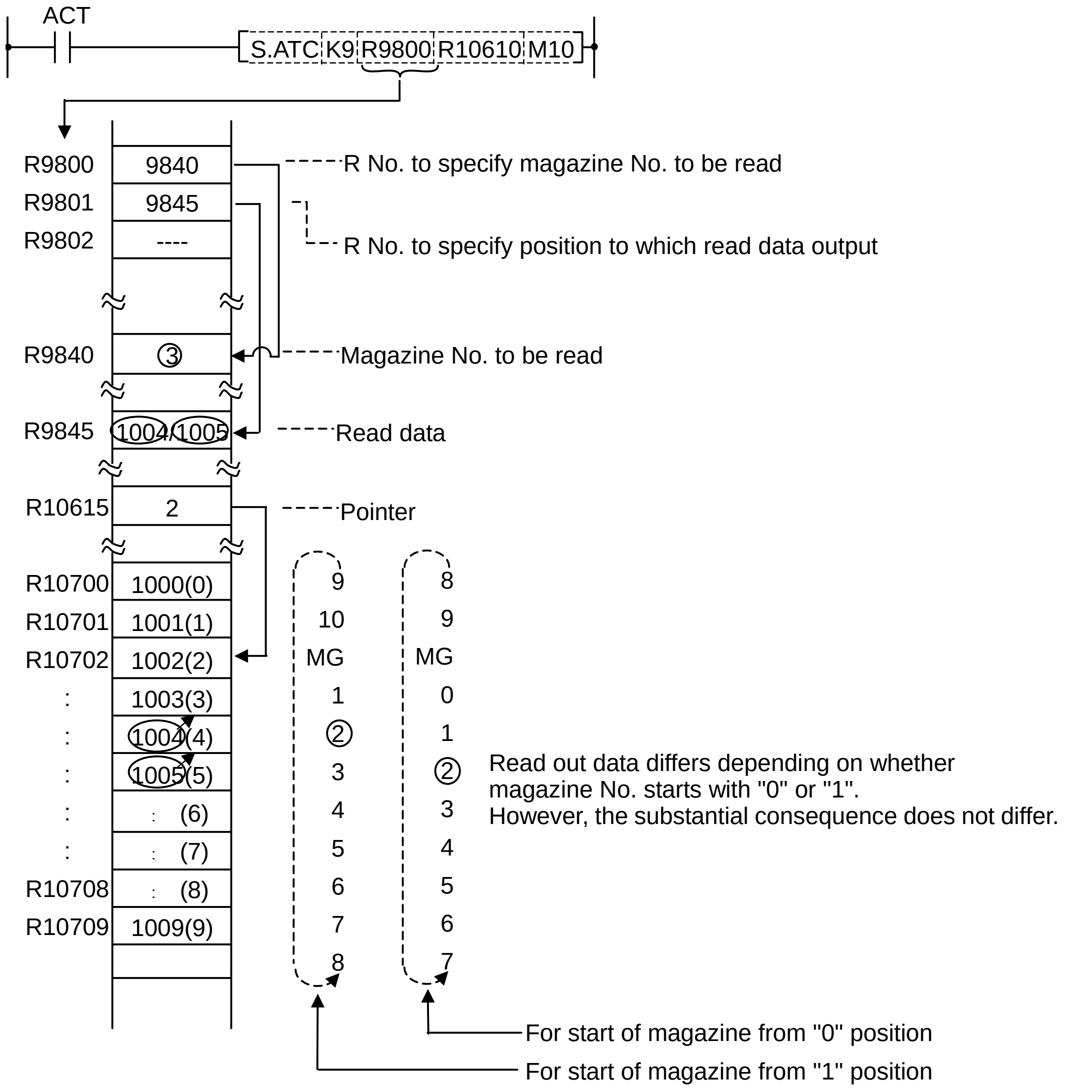
The tool table rotates in reverse direction in accordance with the magazine rotation.



(Note 1) In this control mode, pointer always indicates "0" (tool table head).
(Note 2) When this instruction is executed, the relationship between magazine No. and tool No., appearing on the tool entry display changes accordingly.

(9) Tool data read

This instruction is used to call a specific tool No. in the magazine.

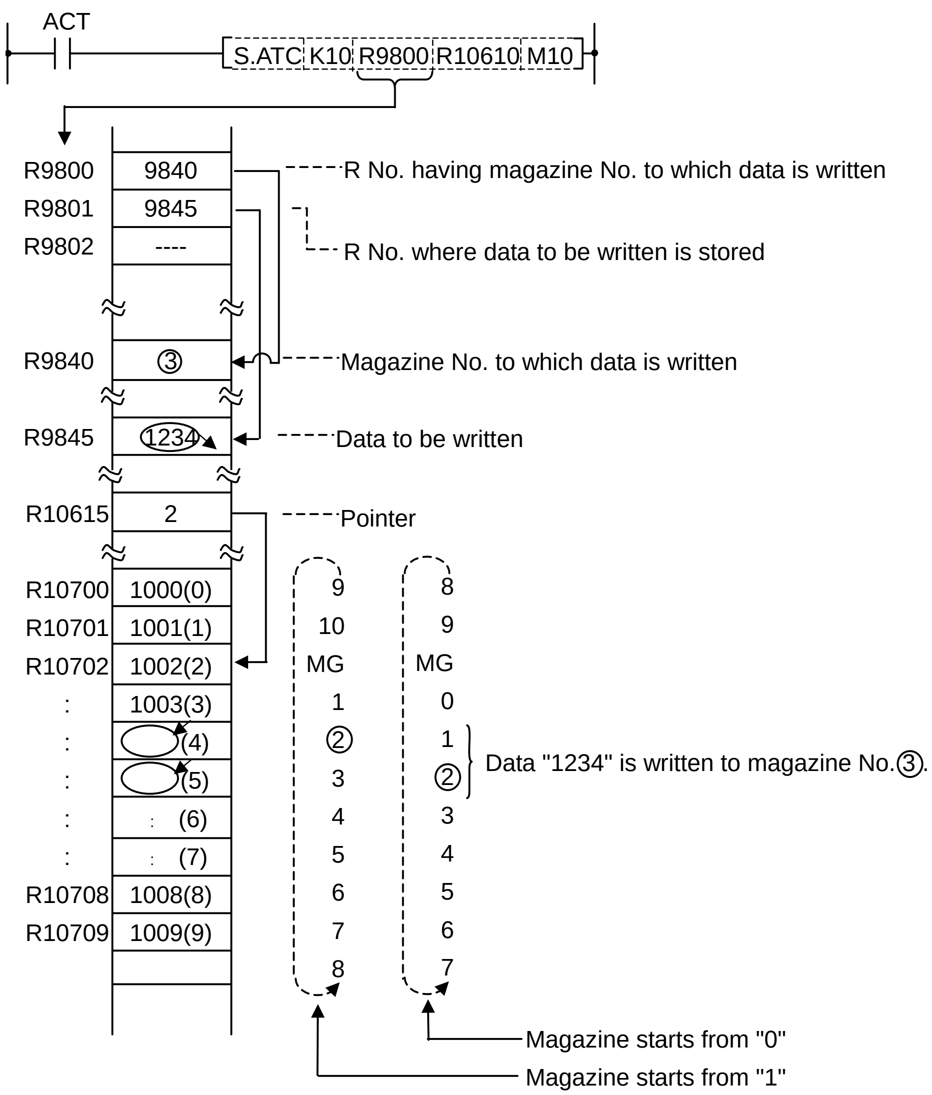


(Note) When pointer is not used, R10615 should be set to "zero".

(Example) MOV K0 R10615

(10) Tool data write

Instead of setting tool No. through the setting and display unit, the tool No. is entered to each magazine No. set through PLC program.

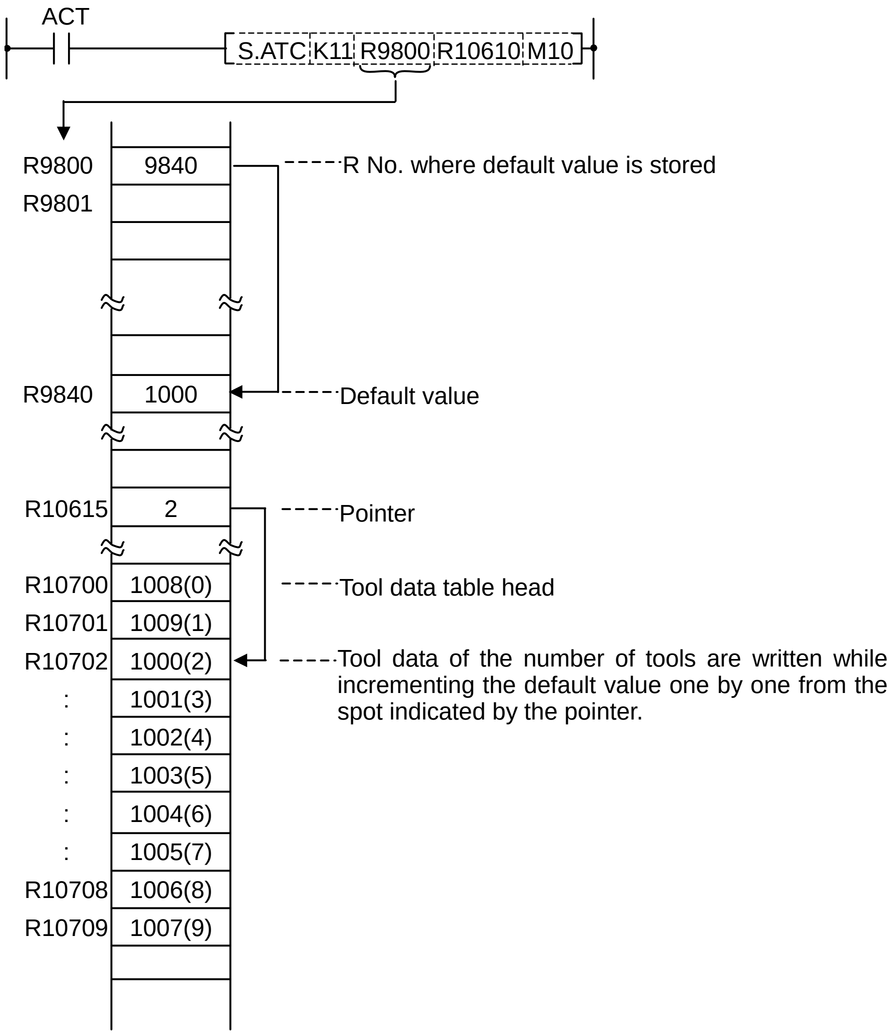


(Note) When pointer is not used, R10615 should be set to "zero".

(Example) MOV K0 R10615

(11) Automatic tool data write

All tool Nos. are written (entered) in batch. This instruction is used for initialization, etc.
The data are written one after another for each tool, starting from the default value.



(Note) When pointer is not used, R10615 should be set to "zero".

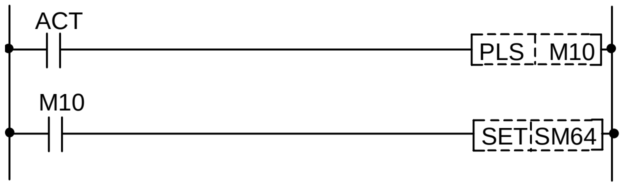
(Example) MOV K0 R10615

9.1.11 Precautions for Using ATC Exclusive Instructions

(1) When tool data is rewritten by ATC or other than ATC instruction, tool registration screen display is not updated. The following processing is required:

- Turn ON special relay SM64 by using the SET instruction.

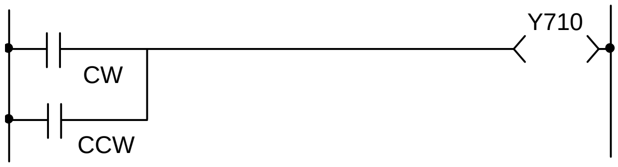
(Program example)



- SM64 processing is not required for ATC instructions ATC K5, K6 (forward run, reverse run of pointer), ATC K7, K8 (forward run, reverse run of tool table).
- SM64 is set through the use of the user PLC and reset by controller.

(2) Method of tool registration prohibiting during magazine rotation
If tool data is set on the tool registration screen during magazine rotation, data may be set in erroneous position. To prevent this error, a signal called "tool registration screen setting prohibited Y710" is provided.

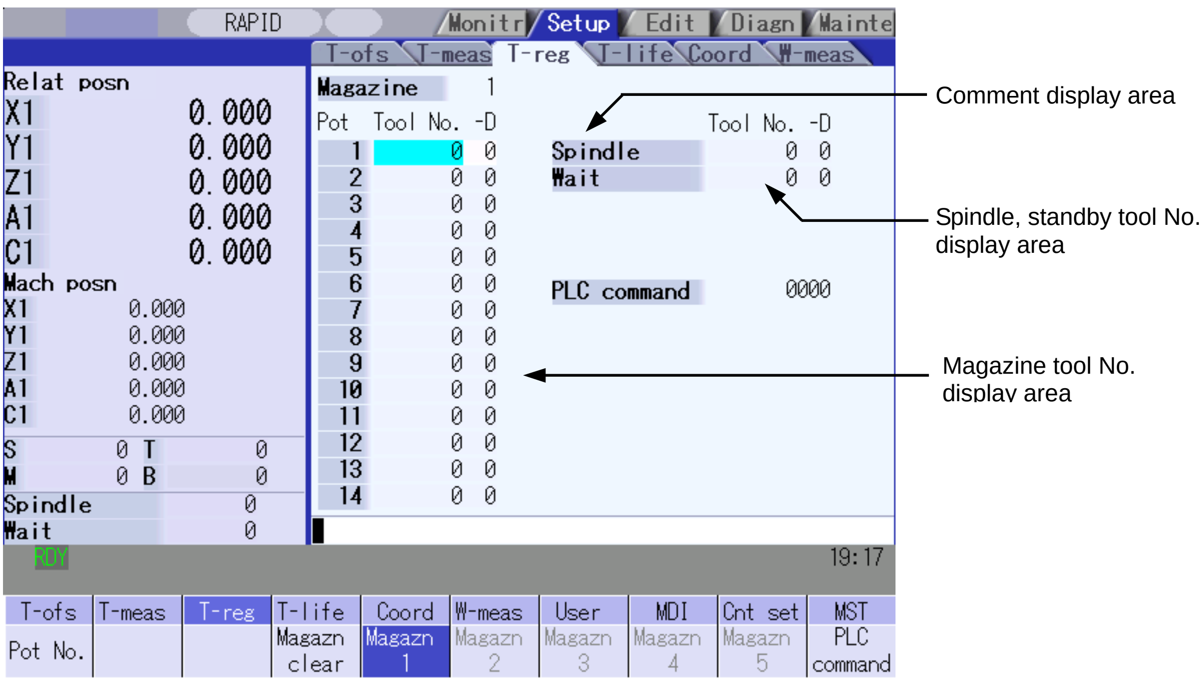
(Program example)



- Y710 turns ON during magazine rotation.
- Setting of AUX data (R10604) is valid while Y710 is being ON.

9.1.12 Examples of Tool Registration Screen

Tool registration screen examples are given below. For operation, refer to the Operation Manual.



(1) Comment display area

The display at the comment display area is created using the message creation function explained in the "III Peripheral Development Environment".

(2) Spindle tool, standby tool display area

The number of display items can be changed according to the control parameter value.

Control parameter (R10600)

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

00: Only spindle tool is displayed.
01: Spindle tool and standby 1 are displayed.
02: Spindle tool and standby 1 and 2 are displayed.
03: Spindle tool and standby 1 to 3 are displayed.
04: Spindle tool and standby 1 to 4 are displayed.
05 or more: No spindle tool or standby tool is displayed.
Hexadecimal expression

(3) Magazine tool No. display area

The number of displayed magazine tools and the magazine No. start value can be changed according to the number-of-magazine parameter and control parameter values.

(a) Number of magazines
Number-of-magazine parameter (R10610) · Setting range
0 to 120: When not arbitrary setting the tools
0 to 360: When arbitrary setting the tools

(Note) If 0 is set, the magazine tool is not displayed. However, the magazine No. and magazine tool number guide area is displayed.

(b) Magazine No. start value
Control parameter (R10600)

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

0: The magazine No. starts at 1.
1: The magazine No. starts at 0.

(Example) Magazine number display when the number of magazines is 12.

MG	TOOL-D	MG	TOOL-D
1		≀	
2		11	
≀		12	

The magazine No. starts at 1

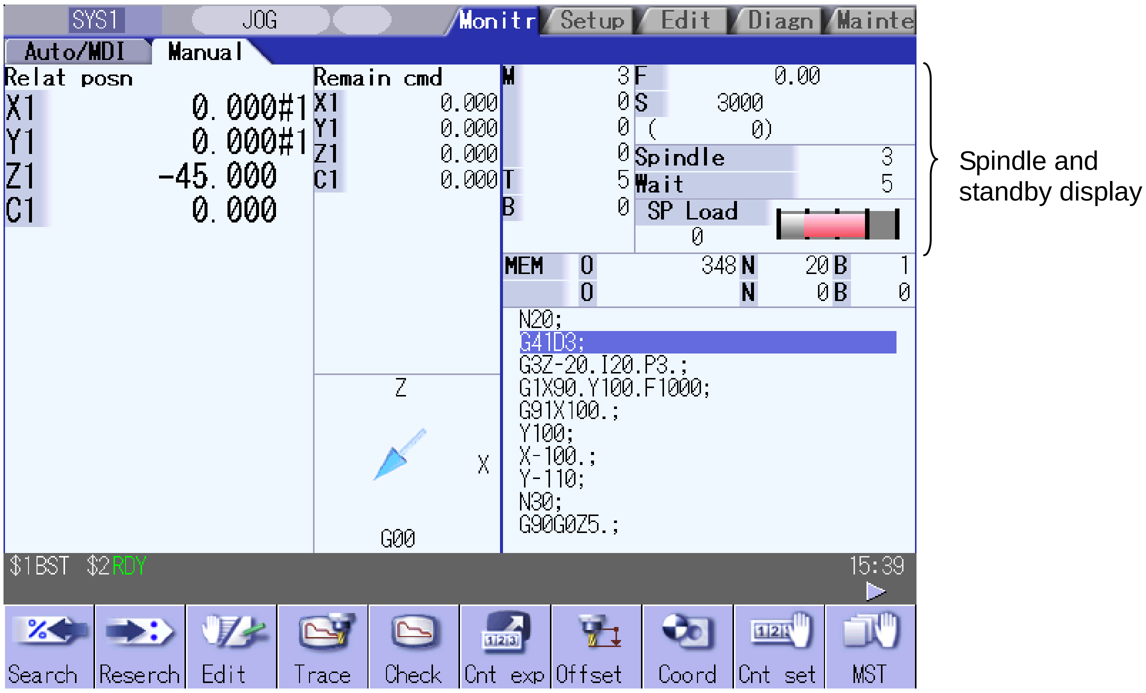
MG	TOOL-D	MG	TOOL-D
0		≀	
1		10	
≀		11	

The magazine No. starts at 0

9.1.13 Display of Spindle Tool and Standby Tool

The tool mounted on the spindle or the tool to be mounted next on the spindle (standby tool) and tool No. in the magazine are set and displayed on the tool registration screen. However, the spindle and standby tool Nos. can also be displayed on the position display screen and tool length measurement screen that are often used. With this, the changes in the magazine pot and spindle tool No. according to the tool selection command or tool change command can be confirmed.

(1) Position display screen



(2) Display tool selection parameter

A maximum of four standby tools can be displayed on the tool registration screen. The No. of the standby tool and the title to be displayed on the monitor screen and setup screen, etc., are selected.

Display tool selection parameter (R10603)

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Bits F to 8: Spare

Bits 5 to 0: Selection of display tool
- 000000 : Spindle, standby 1
- 000001 : Spindle, standby 2
- 000010 : Spindle, standby 3
- 000011 : Spindle, standby 4
- Others : Not display

Bit 7: Selection of display
- 1 : Display tool
- 0 : Do not display tool

9.2 ROT Instructions

With this instruction, the rotary body's target position and rotation direction are determined, as well as the function as a ring counter is realized.
This is used when calculating the rotation direction or number of index steps of the magazine and turret, etc. based on the output data figured with tool No. search of ATC dedicated instruction, or used when controlling the rotary body position.

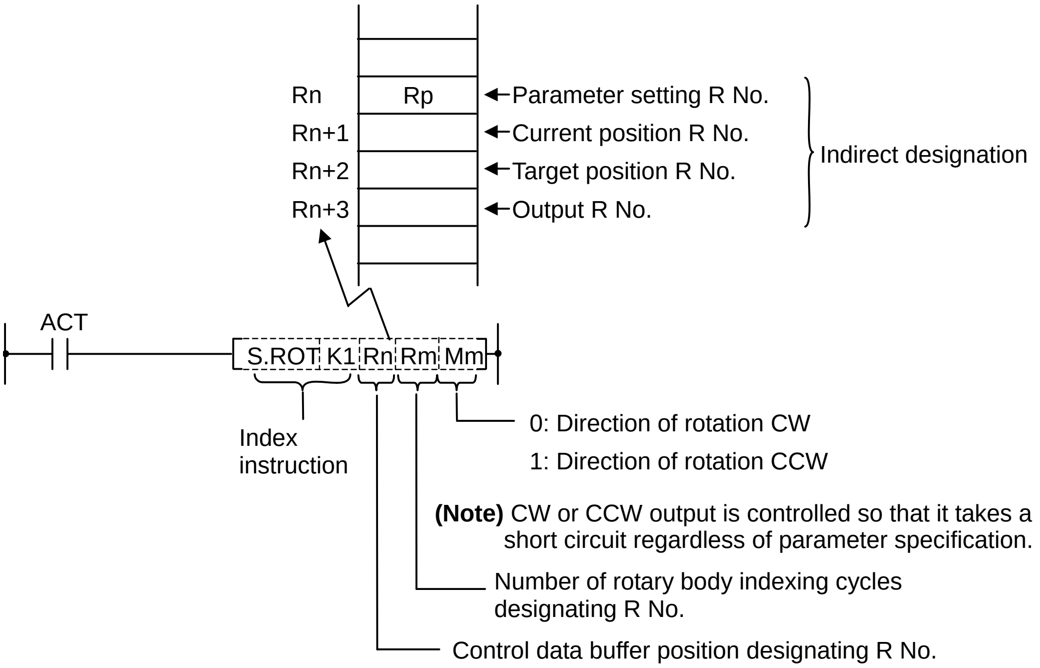
9.2.1 Instruction List

Instruction					Description
S.ROT	K1	Rn	Rm	Mn	Rotary body indexing
S.ROT	K3	Rn	Rm	Mn	Ring counter

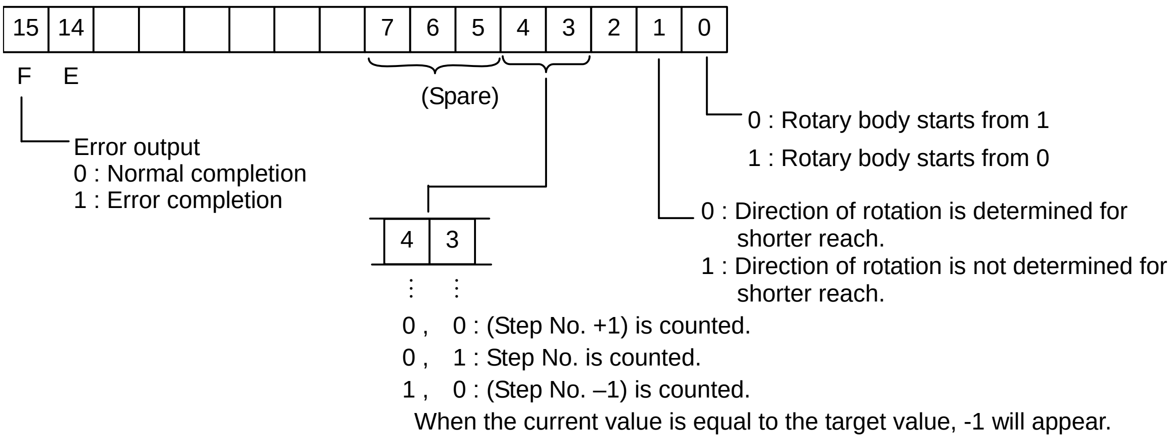
(Note 1) Rot instructions are programmed with "S.ROT".

(1) Rotary body indexing

Direction of rotation and number of steps of ATC magazine (or turret) are determined.



Rp (parameter) contents

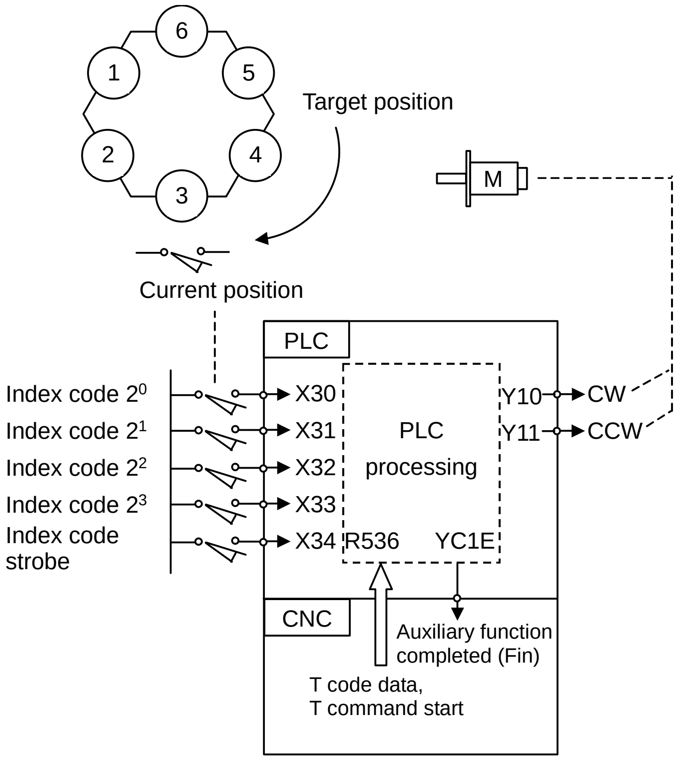


(Note 1) The Index instruction is executed after setting R Nos. to Rn to Rn+3 and writing data in the file registers (R) each corresponding to the R Nos. However, data setting to the parameter (Rp) is done once before execution of the index instruction; this is to prevent the error code from being cleared.

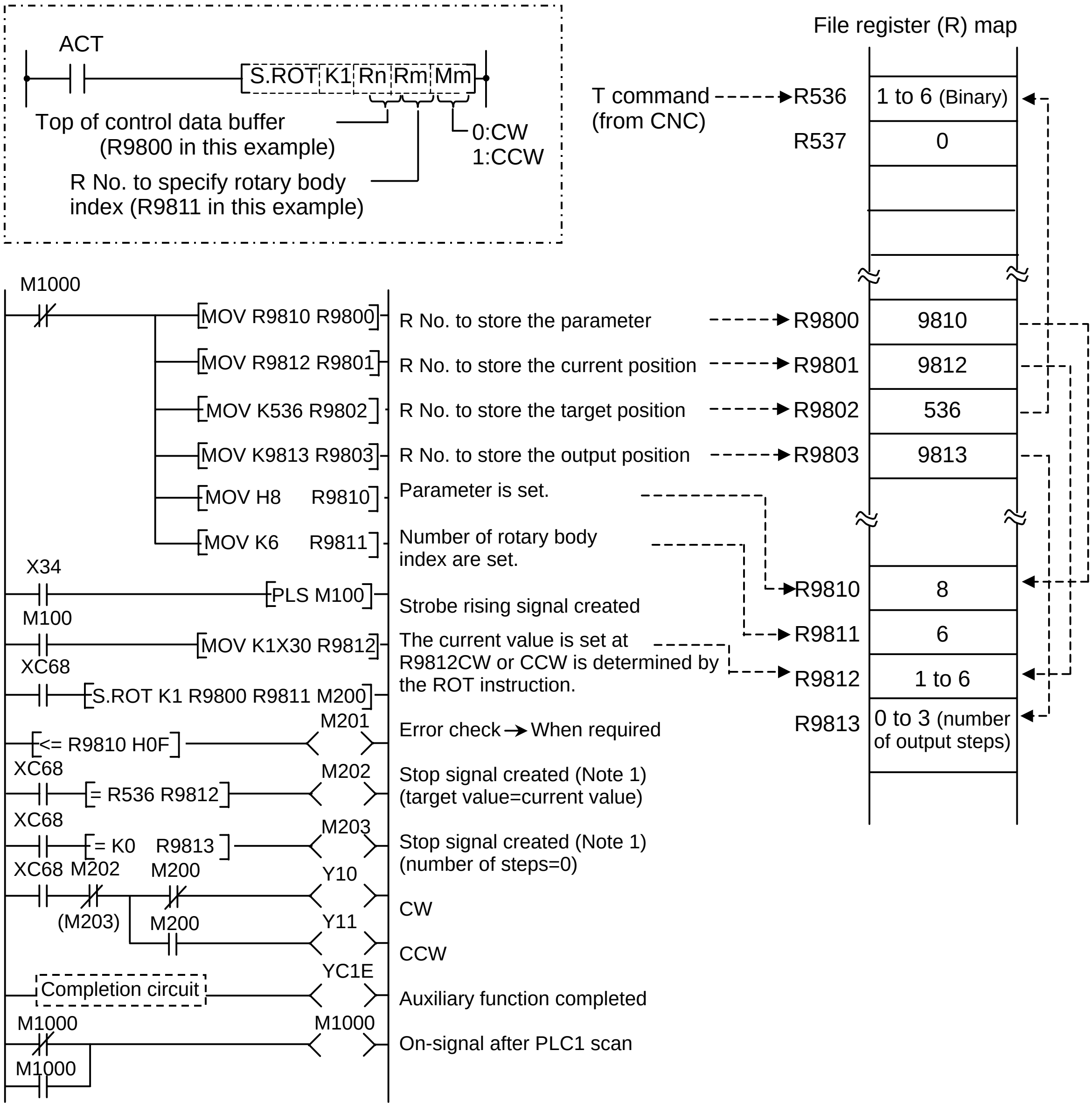
(Note 2) The error code stored in bit F of the parameter (Rp) is not cleared even if the index instruction activating signal (ACT) goes OFF.

(a) Example of rotary body index by ROT K1 instruction

Conditions: (i) The number of rotary body index is 6.
(ii) The target position is set with the T command. The T command is set with the parameters so that it outputs to the PLC with a binary. (Set base specification parameter Tbin to 1.)



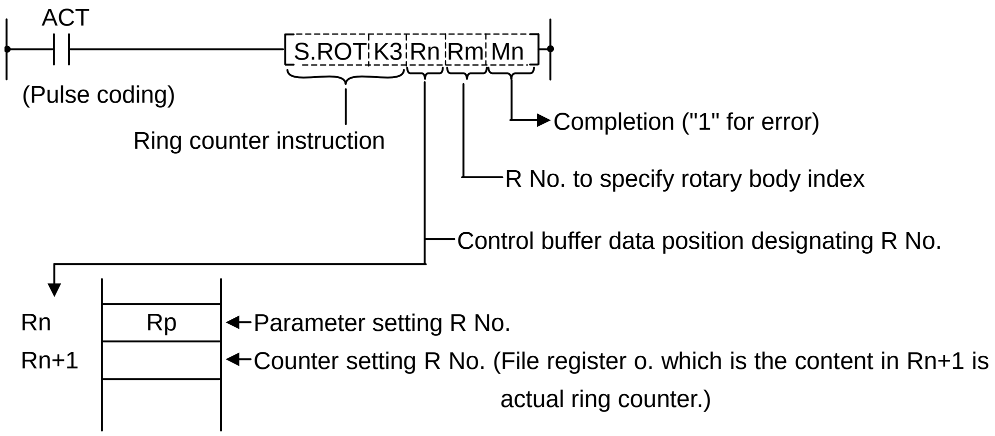
In the example of ladder circuit shown below, the rotation direction is determined by the T command and current position data given by the machine, and the rotary body is rotated in that direction until the target position reaches the current position. When indexing is completed, the auxiliary command completion signal is turned ON.



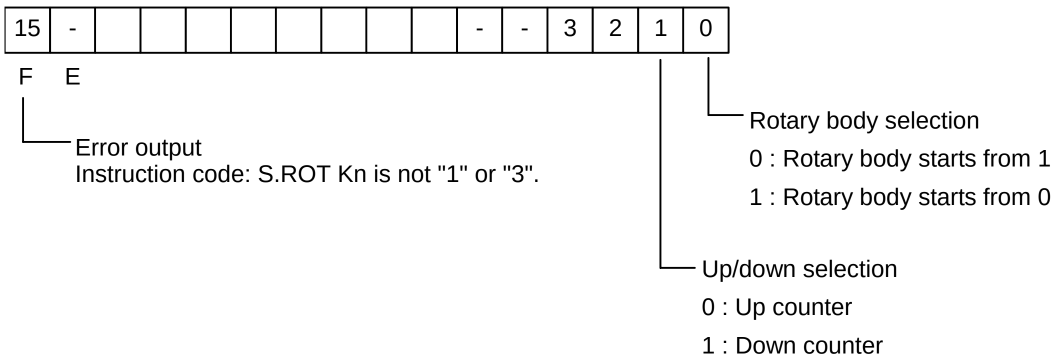
(Note 1) Either M202 or M203 can be used for a stop signal.
(Note 2) The devices (X, Y, and R) are used in this example for no special purpose. Use any device within the available range.
(Note 3) If a number from 1 to 6 has not been specified for current position data (R9812) before the ROT instruction is activated, an error will occur.
(Note 4) The control parameters (R9810) are specified as follows:
(1) Rotary body starts from 1
(2) Take a short cut.
(3) Calculate the number of steps.

(2) Ring counter (Up/down counter)

This instruction is used to control position of rotary body (or turret).



The ring counter is a binary counter; it is used as an up/down counter of "start from 0" or "start from 1" according to the parameter rotary body instruction.



(Note 1) The ring counter instruction is executed after setting R No. to Rn to Rn+1 and specifying data for the parameter.

(Note 2) The error completion (Mm) of the ring counter instruction and the error output in bit F of the parameter (Rp) are cleared when the activating signal (ACT) goes OFF. The activating signal (ACT) of the ring counter instruction is generally pulsed. This makes it hard for the interface diagnostic and ladder monitor programs to detect an error signal. For debugging, therefore, an error hold circuit is provided after the ring count instruction to ease error detection.

10. PLC Help Function

To help the user PLC, an exclusive interface is provided between the user PLC and controller.
The function and interface are explained below.

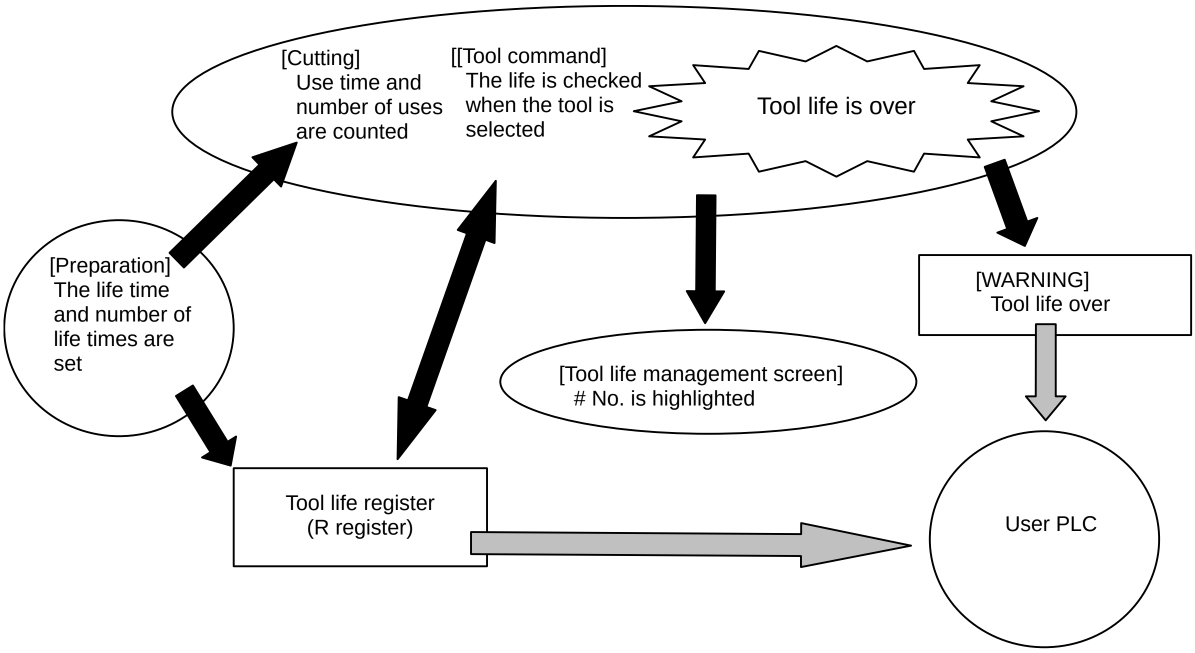
PLC help function examples:

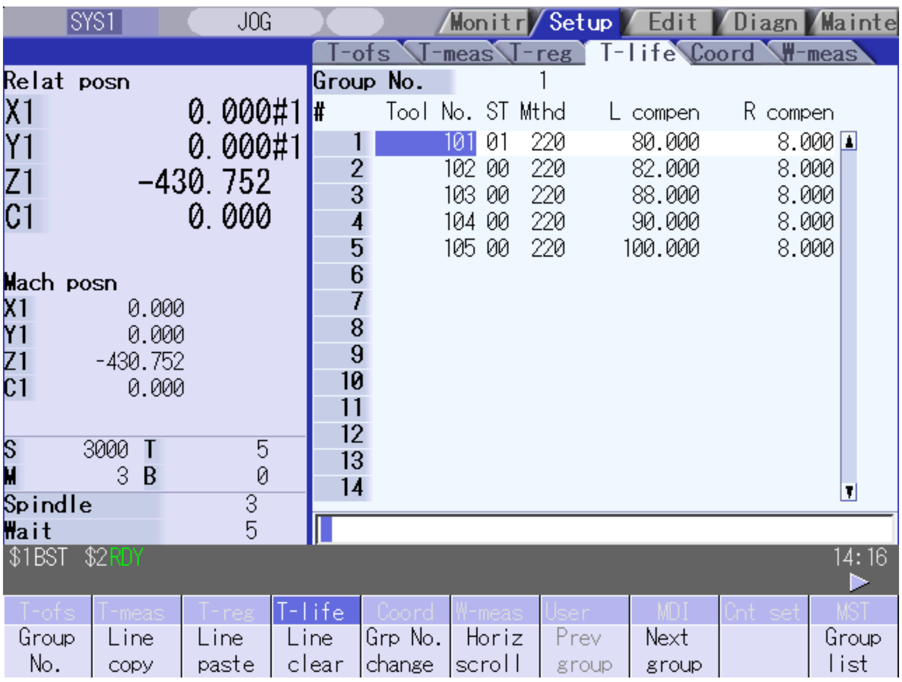
- Tool life management
- External search
- PLC axis control
- External machine coordinate system compensation
- Alarm message display
- Operator message display
- PLC switches
- Key operation by user PLC
- Load meter display
- User PLC version display

10.1 Tool Life Management (Machining Center System)

10.1.1 Outline of Tool Life Management Function

- The tool status is monitored by counting the tool usage time and number of uses.
- When the tool being used reaches its life, the tool life over signal is output, and the tool No. displayed on the LIFE MANAGEMENT DATA screen is highlighted.
- When the tool is commanded, an arbitrary tool is selected from the tool group.
  (Only for tool life management II.)





10.1.2 Tool Life Management Methods

The following two management methods are available.

(1) Tool life management I (When Base common parameter "#1096 T-Ltyp" is set to 1.)

The use time or use count of the spindle tool specified from user PLC (R12200, R12201) is integrated and the tool use state is monitored. Tool data corresponding to the spindle tool is also output. (R11824 to R11847)

(2) Tool life management II (When Base common parameter "#1096 T-Ltyp" is set to 2.)

A function to select a spare tool has been added to the tool management I. The spare tool is selected from the group by the spare tool selection processes executed by the NC when the tool is commanded, etc. The tool data for that spare tool is output.
Tool data corresponding to the spindle tool specified from user PLC (R12200, R12201) is output (R11824 to R11847) and tool compensation corresponding to the spindle tool is made.

10.1.3 Procedure when tool command is executed

(1) Tool life management I

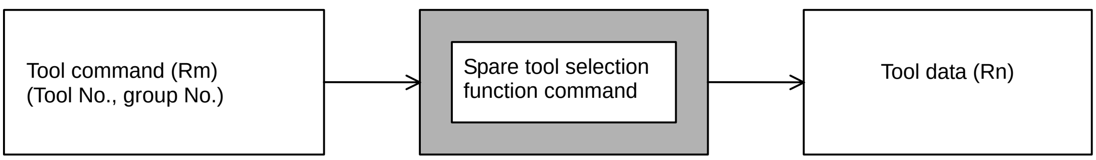
(a) When the tool command (T command) is commanded, the NC outputs the T code data (BCD) and the tool strobe signal.
The T code data (BCD) is binary converted and then used.
(b) The NC checks the tool command, and executes the tool selection process if there is a need for life management.
(c) The tool selection process outputs the tool data for the tool corresponding to the designated tool No.
(d) The user PLC decides whether or not the tool can be used according to the status in the output tool data, and selects command tool or performs alarm processing.

(2) Tool life management II

(a) When the tool command (T command) is commanded, the NC outputs the T code data (BCD) and the tool strobe signal.
The T code data (BCD) is binary converted and then used.
(b) The NC checks the tool command, and executes the spare tool selection process if there is a need for life management.
(c) The spare tool selection process selects the spare tool corresponding to the specified No. (group No., tool No.) and outputs the tool data of the spare tool.
(d) The user PLC decides whether or not the tool can be used according to the status in the output tool data, and selects command tool or performs alarm processing.

(Note) If -1 is set in the group No. in the output tool data, the tool data is invalid. At the time, the specified tool No. is output to the tool No. in the output tool data as it is.

<When tool command is executed>



In tool life management I, tool No. is only specified and spare tool is selected.

Tool is selected according to tool No. in tool data.

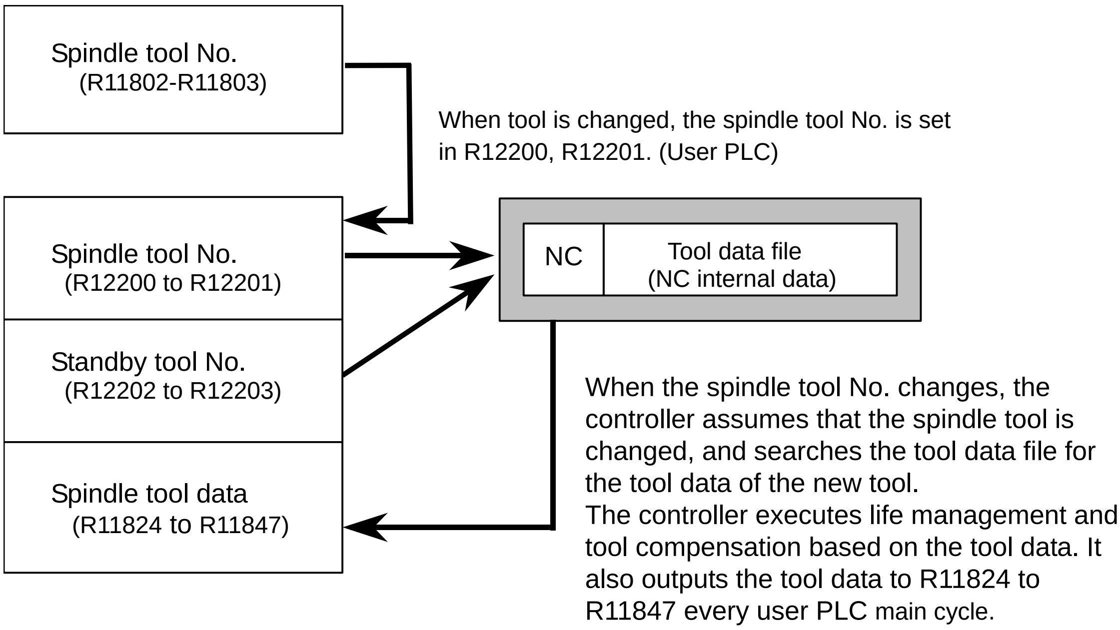
10.1.4 Procedure when Spindle Tool is Changed

(1) When spindle tool is changed during the spindle tool change command (M06), etc., the user PLC specifies the tool No. of the spindle tool (R12200 to R12201).
NC outputs the spindle tool data corresponding to the tool No. of the spindle tool every user PLC main cycle (R11824 to R11847).

(2) NC integrates the use time or use count of the spindle tool based on the spindle tool data in the tool data file.
In tool life management II, it also executes tool compensation corresponding to the spindle tool.

(Note) If -1 is set in the group No. in the output spindle tool data, the spindle tool data is invalid. At the time, the specified tool No. (R12200 to R12201) is output to the tool No. in the output spindle tool data as it is. NC does not integrate the usage time or usage count of the spindle tool or make tool compensation.

<When tool is changed>



10.1.5 Tool Life Management II Method

(1) Tool command method

One of the following two can be selected by using a parameter for command tool No. (Rm contents) input to the spare tool selection process in tool life management II:

(a) Group No. command method (When Base common parameter "#1104 T-Com2" is set to 0.)
The command tool No. (Rm contents) input to the spare tool selection process is handled as group No. Spare tool is selected among the tools corresponding to the group No. in tool data.

(a) Tool No. command method (When BASE SPEC parameter "#1104 T-Com2" is set to 1.)
The command tool No. (Rm contents) input to the spare tool selection process is handled as a tool No. The group No. containing the command tool No. is found and spare tool is selected among the group.

(2) Spare tool selection method

One of the following two can be selected by using a parameter for the spare tool selection method of the spare tool selection process in tool life management II:

(a) Selection in tool registration order (When Base common parameter "#1105 T-Sel2" is set to 0.)
This selects the tool from the tools in use in the same group, following the registration No. order.
If there are no "Tools in use", the tools are selected in order of "Tools not in use", "Normal life tools" and "Abnormal tools", following the registration No. order.

(b) Life equality selection (When Base common parameter "#1105 T-Sel2" is set to 1.)
This selects the tool with the maximum remaining life from the tools in use and not in use in the same group.
When several tools have the same remaining life, the tools are selected in order of registration No. If there are no "Tools in use" or "Tools not in use", the tools are selected in order of "Tools not in use", "Normal life tools" and "Abnormal tools", following the registration No. order.

10.1.6 Maximum Number of Registerable Tools

The maximum number of registerable tools is decided by the system's specifications.

10.1.7 Tool Data

The tool data is tool management data such as the group No., tool No., and tool status.

Tool data name	Explanation	Data range
Group No.	No. to manage tools of the same type (form and dimensions) in a group is set. The tools assigned the same group No. are assumed to be spare tools.	1 to 99999999
Tool No.	No. unique to each tool actually output during tool command execution	1 to 99999999
Tool data flag	Set the parameter for tool life management method, length compensation method, radius compensation method, etc.	bit 7 6 5 4 3 2 1 0 (1) (3) (2) (1) Tool life management method 0-2 (2) Tool length compensation method 0-2 (3) Tool radius compensation method 0-2
Tool status	The tool state is indicated.	0 to FF (H)
Auxiliary data	May differ according to the machine tool builder specifications.	0 to 65535
Tool life data	Life time or life count for each tool is set. (If 0 is set, infinity is assumed to be specified.)	0 to 4000 (minutes) 0 to 65000 (times)
Tool usage data	Usage time or usage count for each tool. (Refer to the following "Usage time, usage count" section for details on the count method.)	0 to 4000 (minutes) 0 to 65000 (times)
Tool length compensation data	The tool length compensation data is set with the format designated with the tool data flag.	Compensation No. 1 to 999 **(Note)** Direct offset amount ±99999.999 Addition offset amount ±99999.999
Tool radius compensation data	The tool radius compensation data is set with the format designated with the tool data flag.	Compensation No. 1 to 999 **(Note)** Direct offset amount ±99999.999 Addition offset amount ±99999.999

(Note) The data range for the tool offset Nos. is decided by the specifications of the number of tool offset sets.

Number of tool offset sets	Data range
40 sets	1 to 40
200 sets	1 to 200
400 sets	1 to 400
999 sets	1 to 999

(1) Tool data flag

Parameter	Details	Value	Explanation
Tool life management method	Usage time	0	Manages the cutting feed with the execution time.
	Mount count	1	Manages the number of times the tool becomes the spindle tool at tool change, etc.
	Work count	2	Manages with the number of times the cutting feed command is issued.
Tool length compensation method Tool radius compensation method	Compensation number method	0	Handles the compensation data in the tool data as the compensation No., and compensates by replacing the compensation No. commanded in the machining program with this value.
	Addition compensation method	1	Handles the compensation data in the tool data as the addition compensation amount. Compensates by adding the amount to the compensation amount indicated by the compensation No. commanded in the machining program.
	Direct compensation method	2	Handles the compensation data in the tool data as the direct compensation amount. Compensates by replacing the amount to the compensation amount indicated by the compensation No. commanded in the machining program with this value.

(2) Tool status

Details	Value	Explanation
Unused tool	0	Set to 0 when replacing the tool with a new tool.
Tool in use	1	This state is activated when cutting is actually started.
Normal life tool	2	This state is activated when the usage data exceeds the life data.
Error tool 1	3	This state is activated when the NC receives tool error 1 signal.
Error tool 2	4	This state is activated when the NC receives tool error 2 signal.

- Values 3 and 4 may differ according to the machine tool builder specifications.
- The unused tool and tool in use are usable spare tools.

(3) Tool life data, tool usage data

The setting range and unit differ according to the tool data flag's tool life management method.

Tool life management method	Setting range	Unit
0 : Usage time	0 to 4000	Minute
1 : Mount count	0 to 65000	Times
2 : Work count	0 to 65000	Times

(4) Tool length compensation data, tool radius compensation data

The tool corresponding to the spindle tool can be compensated with tool life management II.
The setting details and range differ according to the tool data flag's tool length compensation method and tool radius compensation method.

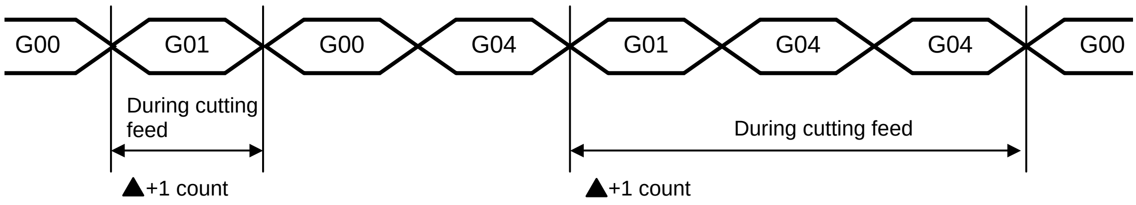
Tool compensation data	Setting details	Setting range
0 : Compensation number method	Compensation No.	1 to 999 **(Note)**
1 : Addition compensation method	Compensation amount	±99999.999
2 : Direct compensation method	Compensation amount	±99999.999

(Note) The data range for the tool offset Nos. is decided by the specifications of the number of tool offset sets.

Number of tool offset sets	Data range
40 sets	1 to 40
200 sets	1 to 200
400 sets	1 to 400
999 sets	1 to 999

10.1.8 Usage Time, Work Count

The usage data is counted with the life system (usage time, work count, mount count) set for each tool.
Tool life management is executed even when the operation mode is MDI.

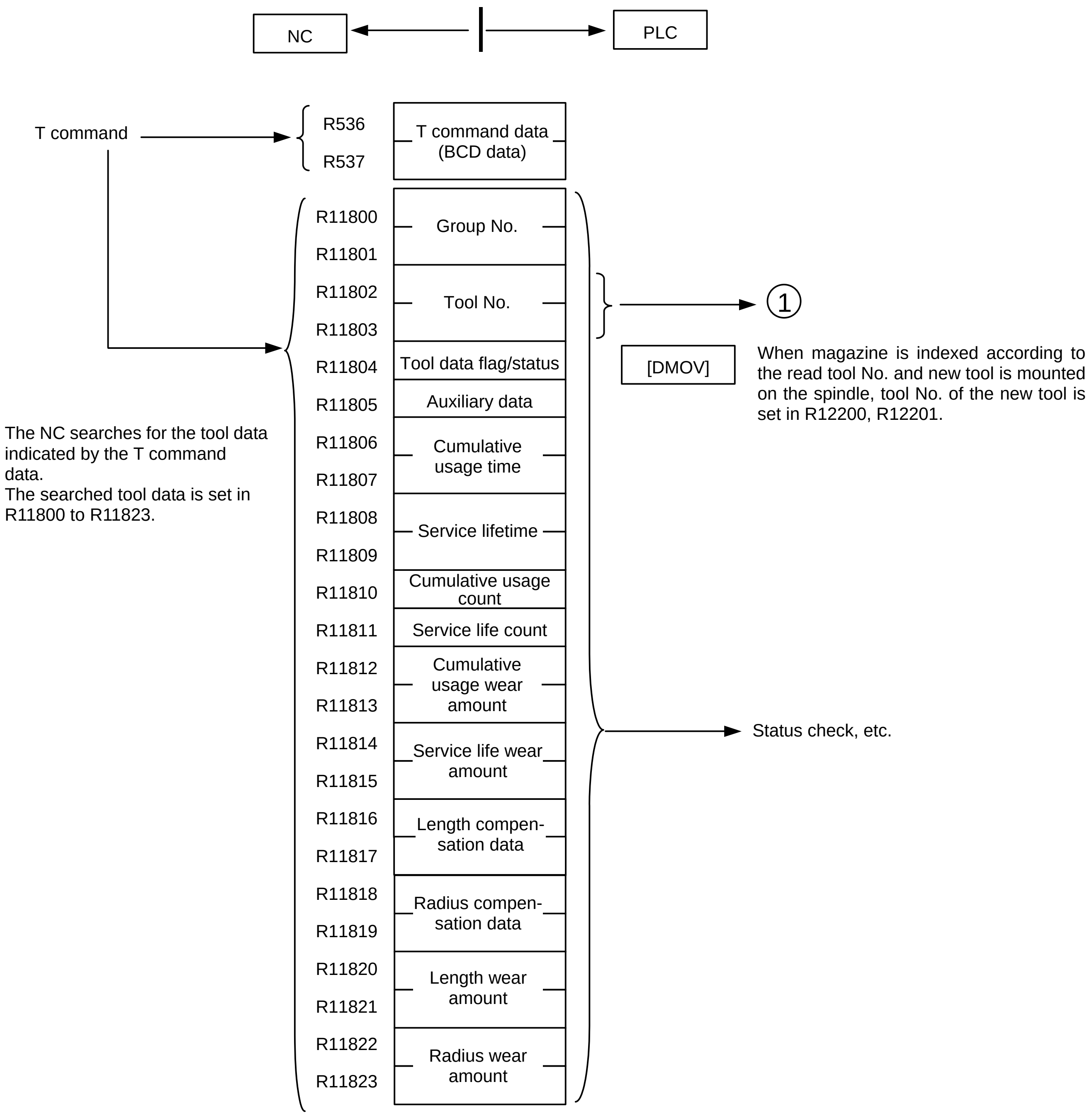


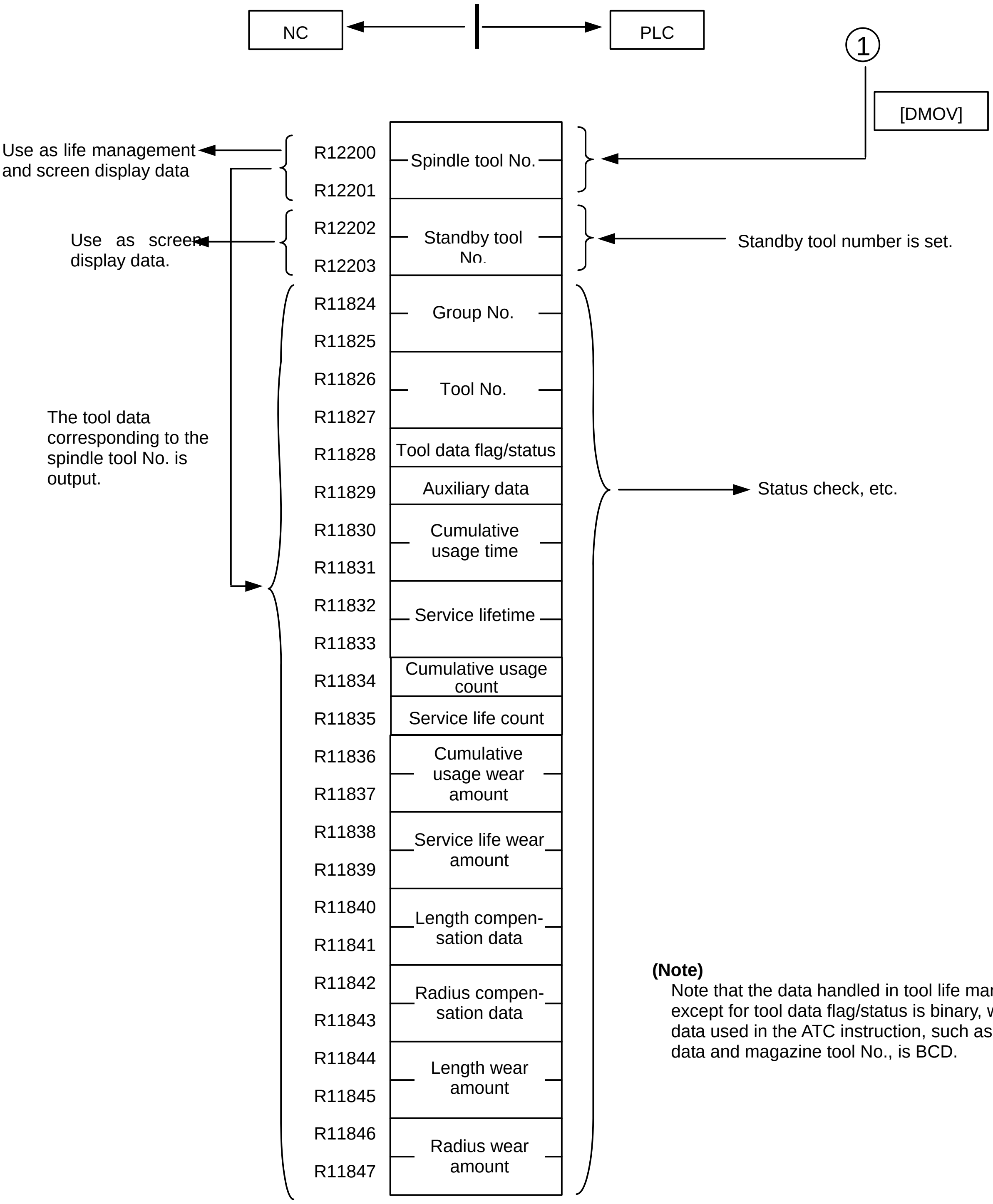
- The usage time is the cumulative time during operation (during cutting feed) in the group 1 modal.
- The cut count is the number of times the state was changed to the group 1 modal (G01, G02, G03, G33).
  Note that rapid traverse and cutting feed commands with no movement are not counted.
  If a command other than a rapid traverse command is issued between the cutting feed commands, the data will not be counted.
- The mount count is the number of times the tool became the spindle tool with tool change.
  If the group 1 modal is not activated even once after becoming the spindle tool, the mounting will not be counted.

The following cases are not counted even when tool life management is valid.

- When control parameter "tool management valid" is OFF
- When the usage data count valid signal is OFF
- When the life data setting value is 0
- When there are two or more tool statuses (normal life, abnormal tool 1, abnormal tool 2)
- During machine lock
- During miscellaneous function lock
- During dry run
- Single block
- During skip

10.1.9 Tool Data Flow (R Register)



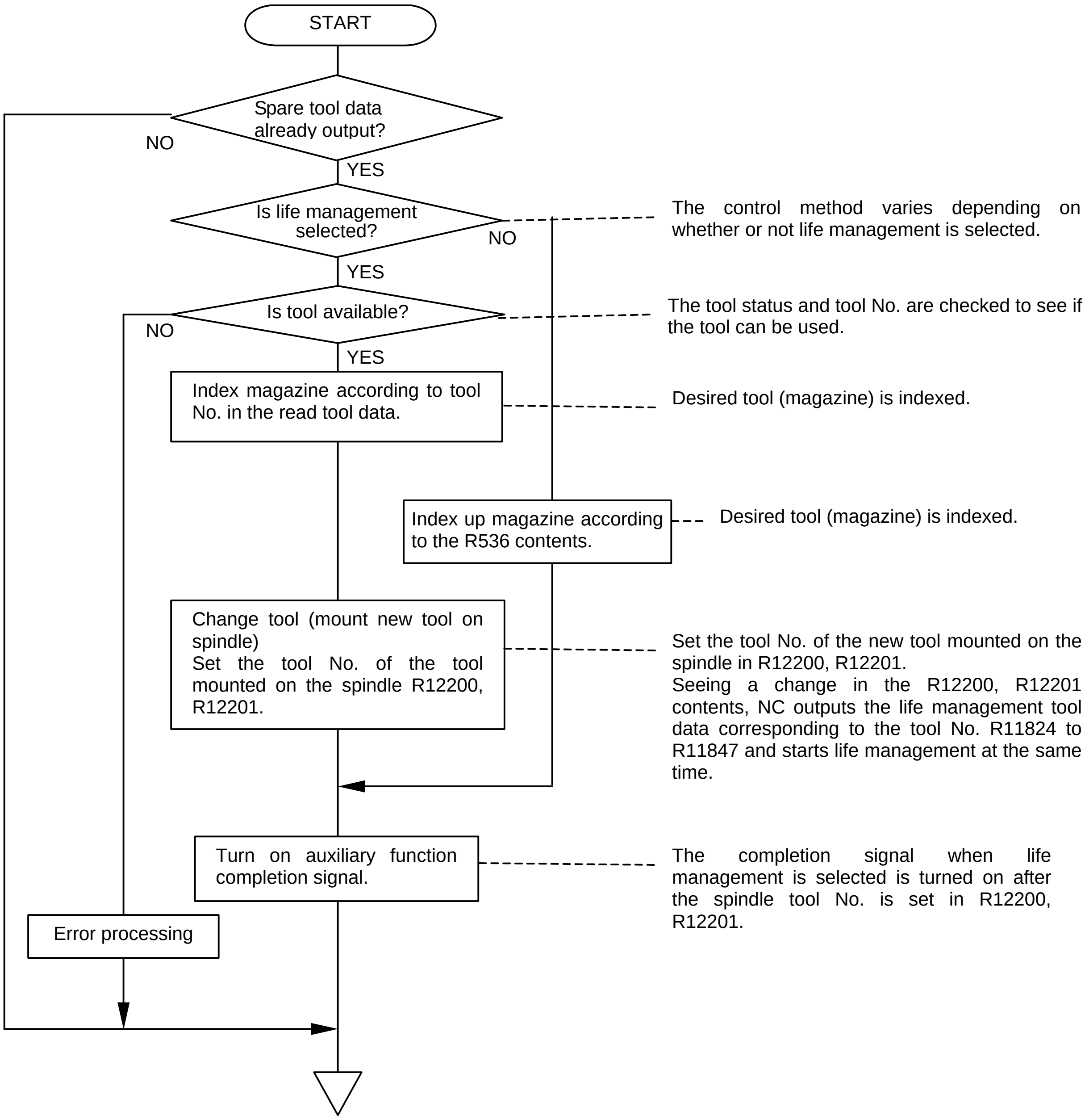


(Note)

Note that the data handled in tool life management except for tool data flag/status is binary, while the data used in the ATC instruction, such as search data and magazine tool No., is BCD.

10.1.10 User PLC processing

A PLC processing example when tool change is made by the T command is given below:



10.1.11 Interface with PLC

(1) NC → PLC

X (part system)

1st part system	2nd part system	Name	Details
XC2B	XD6B	In tool life management	Sent from the NC to the PLC when the tool life management function is selected (when user parameter is ON).
XC2E	XD6E	Tool life over	Sent from the NC to the PLC when the usage data count value exceeds the life data setting value.
XC2F	XD6F	Tool group life over	Sent from the NC to the PLC when all tools in the tool group exceed the life. (Valid only in the tool life management II.)
XC60	XDA0	M function strobe 1	Sent from the NC to the PLC when the miscellaneous function (M function) is executed.
XC68	XDA8	T function strobe 1	Sent from the NC to the PLC when output of the spare tool's data is completed.

(2) PLC → NC

Y (axis)

1st part system	2nd part system	Name	Details
Y8A0	Y8A8	Auto machine lock 1st axis	Tool life management is not executed while these signals are received.
Y8A1	Y8A9	Auto machine lock 2nd axis	
Y8A2	Y8AA	Auto machine lock 3rd axis	
Y8A3	Y8AB	Auto machine lock 4th axis	
Y8A4	Y8AC	Auto machine lock 5th axis	
Y8A5	Y8AD	Auto machine lock 6th axis	
Y8A6	Y8AE	Auto machine lock 7th axis	
Y8A7	Y8AF	Auto machine lock 8th axis	

Y (part system)

1st part system	2nd part system	Name	Details
YC12	YD52	Single block	Tool life management is not executed while this signal is received.
YC15	YD55	Dry run	Tool life management is not executed while this signal is received.
YC1E	YD5E	M function finish 1	Execution of the machining program waits until this signal is received.
YC5A	YD9A	Miscellaneous function lock	Execution of the machining program waits until this signal is received.
YC88	YDC8	Tool alarm 1	The tool status is changed to 3 when the NC receives this signal.
YC89	YDC9	Tool alarm 2	The tool status is changed to 4 when the NC receives this signal.
YC8A	YDCA	Data count valid	The tool usage data is not counted when this signal is not being received.
YC8B	YDCB	Tool life management input	Tool life management is executed when NC receives this signal, and the output during tool life management is output to PLC.

(3) R registers

R (part system)

1st part system	2nd part system	Name	Details
R504	R704	M code data 1	This No. is designated with the M command.
R505	R705		
R536	R736	T code data 1	This No. is designated with the T command.
R537	R737		
R567	R767	Group in tool life management	This is the No. of the group for which life management is active.
R628	R828	Tool life usage data	This is the usage time and work count of the tool for which life management is active.
R629	R829		
R630	R830	Number of registerable tool life control tools	This is the number of tools for which life management is active.
R2588	R2788	Tool life management data sort	This is the tool life data sort necessity flag.
R2590	R2790	Tool group number designation	This is the tool group No. commanded with the T command.
R2591	R2791		

R (ATC, life management): Tool life management data (NC → PLC) standby tool data

1st part system	2nd part system	Name	Details
R11800	R11850	T life mgmt Standby tool: Group No.	This is the standby tool's group No.
R11801	R11851		
R11802	R11852	Standby tool: Tool No.	This is the standby tool's tool No.
R11803	R11853		
R11804	R11854	Standby tool: Flag/Status	This is the standby tool's flag/status.
R11805	R11855	Standby tool: Auxiliary data	This is the standby tool's auxiliary data.
R11806	R11856	Standby tool: Cumulative usage time	This is the standby tool's usage time.
R11807	R11857		
R11808	R11858	Standby tool: Service lifetime	This is the standby tool's service lifetime.
R11809	R11859		
R11810	R11860	Standby tool: Cumulative usage count	This is the standby tool's work count time.
R11811	R11861	Standby tool: Service life count	This is the standby tool's service life count.
R11812	R11862	Standby tool: Cumulative usage wear amount	This is the standby tool's usage wear amount.
R11813	R11863		
R11814	R11864	Standby tool: Service life wear amount	This is the standby tool's service life wear amount.
R11815	R11865		
R11816	R11866	Standby tool: Length compensation amount	This is the spare tool's length compensation amount.
R11817	R11867		
R11818	R11868	Standby tool: Radius compensation amount	This is the standby tool's radius compensation amount.
R11819	R11869		
R11820	R11870	Standby tool: Length wear amount	This is the standby tool's length wear amount.
R11821	R11871		
R11822	R11872	Standby tool: Radius wear amount	This is the standby tool's radius wear amount.
R11823	R11873		

R (ATC, life management): Tool life management data (NC → PLC) Active spindle tool data

1st part system	2nd part system	Name	Details
R11824	R11874	T life mgmt Active tool: Group No.	This is the active tool's group No.
R11825	R11875		
R11826	R11876	Active tool: Tool No.	This is the active tool's tool No.
R11827	R11877		
R11828	R11878	Active tool: Flag/Status	This is the active tool's flag/status.
R11829	R11879	Active tool: Auxiliary data	This is the active tool's auxiliary data.
R11830	R11880	Active tool: Cumulative usage time	This is the active tool's usage time.
R11831	R11881		
R11832	R11882	Active tool: Service lifetime	This is the active tool's service lifetime.
R11833	R11883		
R11834	R11884	Active tool: Cumulative usage count	This is the active tool's work count time.
R11835	R11885	Active tool: Service life count	This is the active tool's service life count.
R11836	R11886	Active tool: Cumulative usage wear amount	This is the active tool's usage wear amount.
R11837	R11887		
R11838	R11888	Active tool: Service life wear amount	This is the active tool's service life wear amount.
R11839	R11889		
R11840	R11890	Active tool: Length compensation amount	This is the active tool's length compensation amount.
R11841	R11891		
R11842	R11892	Active tool: Radius compensation amount	This is the active tool's radius compensation amount.
R11843	R11893		
R11844	R11894	Active tool: Length wear amount	This is the active tool's length wear amount.
R11845	R11895		
R11846	R11896	Active tool: Radius wear amount	This is the active tool's radius wear amount.
R11847	R11897		

R (ATC, life management): Tool life management data (NC → PLC) Spindle/Standby tool Nos.

1st part system	2nd part system	Name	Details
R12200	R12210	T life mgmt Spindle tool No.	This is the active spindle tool No.
R12201	R12211		
R12202	R12212	T life mgmt Standby tool No.	This is the standby tool No.
R12203	R12213		

10.2 External Search

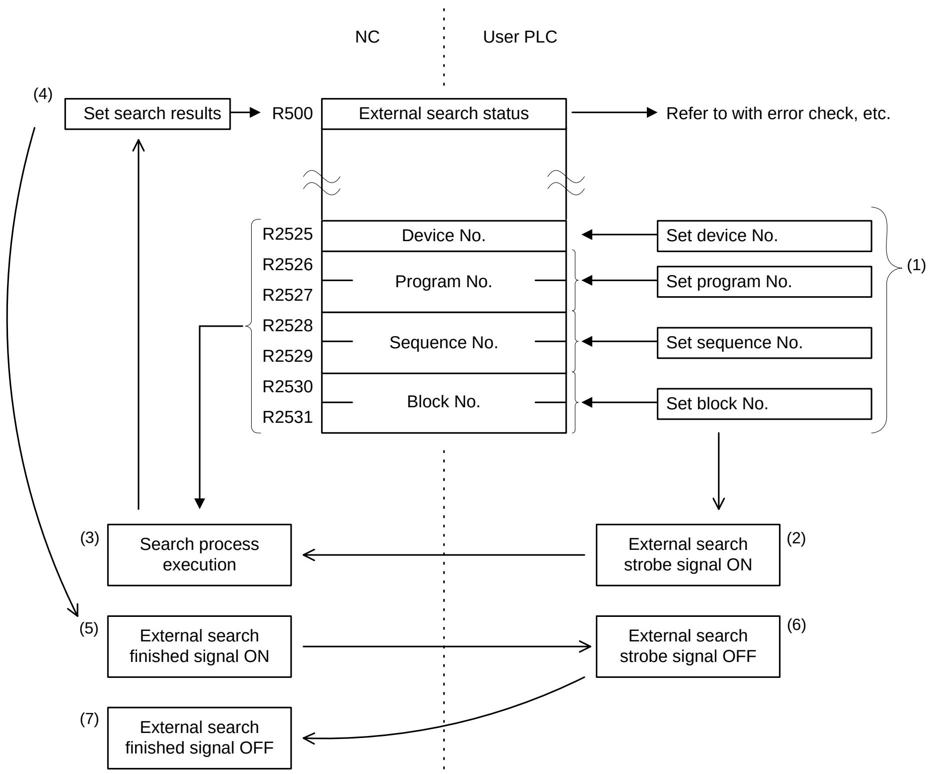
External search is a function that searches the machining program according to the machining program storage device, program No., sequence No. and block No. designated from the user PLC.

10.2.1 Detailed Explanation

The flow of the external search execution process is given below.

(1) The user PLC sets the device, program No., sequence No. and block No.
(2) The user PLC sets the external search strobe signal ON.
(3) The NC searches for the target machining program from the designated device, program No., sequence No. and block No.
(4) The NC sets the search results as the external search status.
(5) The NC turns the external search finished signal ON.
(6) The user PLC turns the external search strobe signal OFF.
(7) The NC turns the external search finished signal OFF.

Flow of external search process for 1st part system



10.2.2 PLC → NC Interface Signal

Signal name	Part system No.			
	1st part system	2nd part system	3rd part system	4th part system
External search device No.	R2525	R2725	R2925	R3125
External search program No.	R2526 R2527	R2726 R2727	R2926 R2927	R3126 R3127
External search sequence No.	R2528 R2529	R2728 R2729	R2928 R2929	R3128 R3129
External search block No.	R2530 R2531	R2730 R2731	R2930 R2931	R3130 R3131
External search strobe	YC1D	YD5D	YE9D	YFDD

(1) External search device No.

The device storing the machining program to be searched is designated with a No.
When a front CF card is selected in 70 Series, either "2" or "4" can be selected.

Device number	Device
0	Memory
1	HD (D drive)
2	IC card (Drive E)
3	Floppy disk (Drive A)
4	High-speed program server
5	Tape (RS232C)

(2) Program No.

Designate the No. of the machining program to be searched as a binary. 1 to 99999999 (8 digits)

(3) Sequence No.

Designate the sequence No. of the machining program to be searched as a binary. 1 to 99999 (5 digits)

(4) Block No.

Designate the block No. as a binary. 0 to 99999 (5 digits)

(5) External search strobe

The NC starts the external search at the rising edge of this signal.
The combinations of designation conditions and correspondence of the search blocks are shown below.
Under each condition, the search block + block with designated block No. are searched.

Condition		Search block
Program No.	Sequence No.	
Designated	Designated	Designated sequence No. for designated program
Designated	Not designated (= 0)	Head of designated program
Not designated (= 0)	Designated	Designated sequence No. in currently selected program
Not designated (= 0)	Not designated (= 0)	Error: 4 Refer to 10.2.5 External search status

10.2.3 NC → PLC Interface Signal

Signal name	Part system No.			
	1st part system	2nd part system	3rd part system	4th part system
External search finished	XC1D	XD5D	XE9D	XFDD
External search status	R500	R700	R900	R1100

(1) External search finished

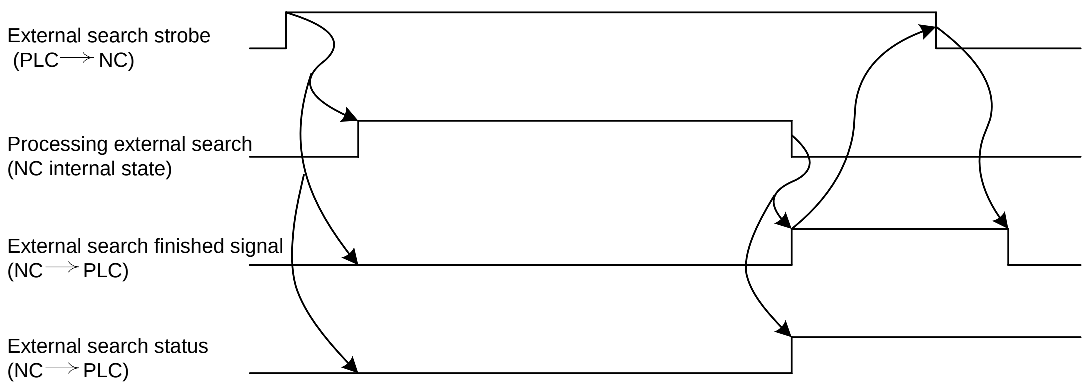
This turns ON when the external search is finished. This also turns ON when an error occurs.
This signal turns OFF when the "External search strobe signal" is turned OFF from the user PLC.

(2) External search status

The status at the end of the external search is output.
Refer to "10.2.5 External search status".

10.2.4 Timing Chart

The timing chart for the external search is shown below.



10.2.5 External Search Status

The correspondence of the external search status values and details output from the NC based on the external search is shown below.

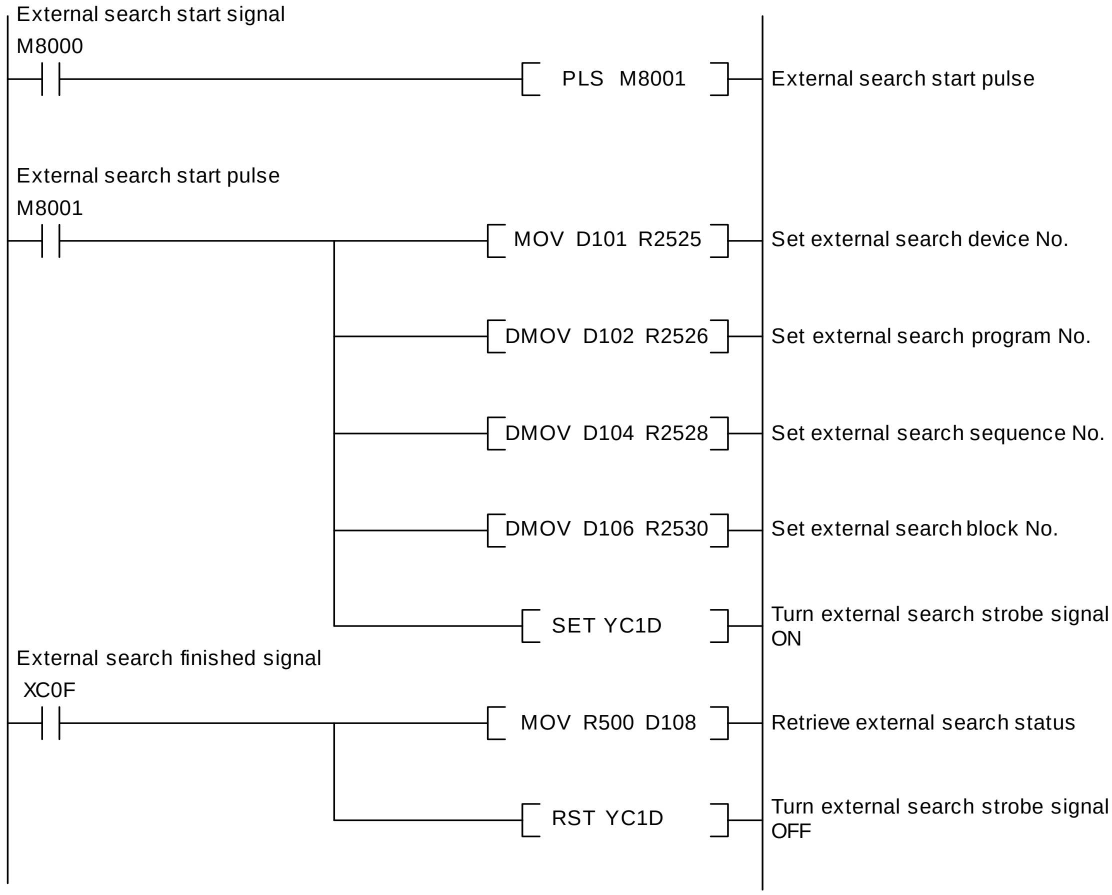
External search status value	Details	Remedy
0	Normally finished	-
1	Operation search is being carried out	Wait for other function's operation search to finish before searching.
2	Search was attempted while the program was running	Stop the program before searching
3	A device that does not exist or which is disabled was designated	Confirm the presence of the device, and that the device is within the specifications
4	The program file is not designated	Designate the program No. or sequence No.
5	The block with the designated program No., sequence No. or block No. was not found	-
6	The external search specifications are not available	Check the specifications

10.2.6 Precautions

Even if the external search strobe is already OFF when the NC finishes the external search, the external search finished signal will turn ON for one cycle of the user PLC.

10.2.7 Usage Example

An example of the external search ladder for 1st part system is shown below.



10.3 PLC Axis Control

This function allows an independent axis to be controlled with commands from the user PLC, separately from the NC control axis.

10.3.1 Specifications

Item	Details
Number of control axes	Max. :2 axes (700 Series), 6 axes (70 Series)
Simultaneous control axes	The PLC control axis is controlled independently of the NC control axis. Simultaneous start of multiple PLC axes is possible.
Command unit	Least command increment **(Note 1)** 0.001mm (0.0001 inch) 0.0001mm (0.00001 inch) 0.00001mm (0.000001 inch) 0.000001mm (0.0000001 inch)
Feedrate	0 to 1000000 mm/min (0 to 100000 inch/min) (The speed is fixed regardless of the unit system.)
Movement commands	Incremental value commands from the current position. Absolute value commands of the machine coordinate system. 0 to ±99999999 **(Note 1)**
Operation modes	Rapid traverse, cutting feed Jog feed (+), (-) Reference position return feed (+), (-) Handle feed
Backlash compensation	Provided
Stroke end	Not provided
Soft limit	Provided
Rotation axis commands	Provided Absolute value commands....... Rotation amount within one rotation. (Rotates the remainder divided by rotational axis division count.) Incremental commands............ Rotates the commanded rotation amount.
Inch/mm changeover	Not provided Command to match the feedback unit.
Position detector	Encoder (absolute position detection also possible)

(Note 1) The following units are used for the input/output data in PLC axis control according to the parameter "#1005 plcunit" setting.

Setting value	Unit
B	0.001 mm (0.0001 inch)
C	0.0001 mm (0.00001 inch)
D	0.00001 mm (0.000001 inch)
E	0.000001 mm (0.0000001 inch)

The screen display changes as follows according to the parameter "#1003 iunit" setting.

Setting value	Unit	Display
B	0.001 mm (0.0001 inch)	Displays up to three digits after the decimal point
C	0.0001 mm (0.00001 inch)	Displays up to four digits after the decimal point
D	0.00001 mm (0.000001 inch)	Displays up to five digits after the decimal point
E	0.000001 mm (0.0000001 inch)	Displays up to six digits after the decimal point

(Note 2) The unit system is split into the display (iunit) and control data (plcunit), so when confirming the effective value of the PLC axis control data on the screen, set the display unit (iunit) to the same unit as the control data unit (plcunit).

Other restrictions

(1) There is no mirror image, external deceleration or machine lock function.
(2) Rapid traverse override, cutting override and dry run control are not possible.
(3) Automatic operation start, automatic operation stop, reset and interlock NC controls are invalid for PLC control axes.
The same control can be realized using an interface dedicated for PLC control axes.
(4) There is no dedicated emergency stop. The emergency stop is valid in the same manner as the NC control axis.

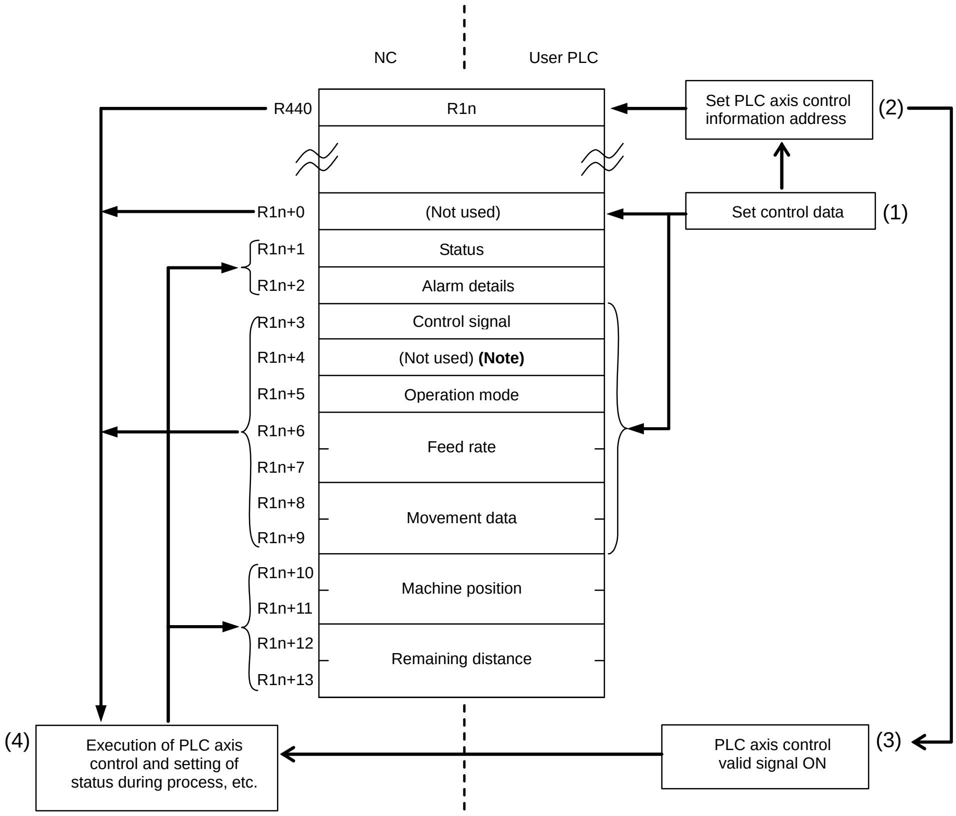
10.3.2 Detailed explanation

The flow of the process during PLC axis control execution is shown below.

(1) The user PLC sets the control data in the R register.
(2) The user PLC sets the PLC axis control information address in the R register.
(3) The user PLC turns the PLC axis control valid signal ON.
(4) The NC executes PLC axis control based on the control data.

During PLC axis control, the status, alarm details, machine position and remaining distance are set in the R register.

Flow of PLC axis control for 1st PLC axis in single mode



(Note) Refer to "(4) Axis specification" in "10.3.4 Details of PLC Axis Control Information Data" for the explanation of unused register No. R1n+4.

10.3.3 PLC Interface

The PLC and NC interface is carried out by turning ON the PLC axis control valid signal or the PLC axis control buffering mode valid signal after the PLC sets the control information data in the R register.
The operation is executed in single mode if the PLC axis control valid signal is ON, executed in buffering mode if the PLC axis control buffering mode valid signal is ON.
Single mode and buffering mode cannot be operated simultaneously. An alarm (command mode overlap) occurs when the valid signal of either mode is turned ON while the other mode is operating. Note that the alarm does not cancel the operation in the first active mode.

Refer to "10.3.8 Single Mode" and "10.3.9 Buffering Mode" for the details of each mode. For the details of alarms, refer to "(2) Alarm Details" in "10.3.4 Details of PLC Axis Control Information Data".

(1) PLC axis control valid signal

The PLC axis control process is executed in single mode with the control information data while the PLC axis control valid signal is ON.
The reset state is activated when the PLC axis control valid signal is turned OFF. In this case, the axis control is reset only once; when the ON signal is turned OFF.

Signal name	PLC axis No.					
	1st axis	2nd axis	3rd axis	4th axis	5th axis	6th axis
PLC axis control valid	Y770	Y771	Y772	Y773	Y774	Y775

(2) PLC axis control buffering mode valid signal

When the PLC axis control buffering mode valid signal is ON, the PLC axis control is executed upon the control information data in buffering mode.
If the PLC axis control buffering mode valid signal is turned OFF, the axis control is reset. In this case, the axis control is reset only once; when the ON signal is turned OFF.

Signal name	Common for all axes
PLC axis control buffering mode valid	Y723

(3) PLC axis control information address

The PLC axis control information address stores head R registers of the control information for each PLC axis.

Signal name	PLC axis No.					
	1st axis	2nd axis	3rd axis	4th axis	5th axis	6th axis
PLC axis control information address	R440	R441	R442	R443	R444	R445

(Note 1) The following R registers can be used.
R8300 to R9799 (Area backed up by battery)
R9800 to R9899 (Area not backed up by battery)
(Note 2) If the setting value of R register is out of range or odd number, the target PLC axis turns invalid.
(Note 3) In the following explanations, the R register Nos. stored in the PLC axis control information address are indicated as R1n for the 1st axis, R2n for the 2nd axis, R3n for the 3rd axis, and R4n for the 4th axis, R5n for the 5th axis, R6n for the 6th axis.

(4) PLC axis control buffering mode information address

The PLC axis control buffering mode information address stores head R registers of the buffering mode control information for each PLC axis. **(Note 1)**

Signal name	Common for all axes
PLC axis control buffering mode information address	R448

(Note 1) The following R registers can be used.
R8300 to R9799 (Area backed up by battery)
R9800 to R9899 (Area not backed up by battery)
(Note 2) If the setting value of R register is out of range or odd number, the target PLC axis turns invalid.
(Note 3) In the following explanations, the R register Nos. stored in the PLC axis control buffering mode information address are indicated with Rn. The PLC axis control information data arrangement follows the arrangement of "R1n" and others, which is referred in "(3) PLC axis control information address".

(5) PLC axis control information data

The control information data is stored in the R register value indicated by the PLC axis control information address and the subsequent register values before starting the PLC axis control process. The control information data arrangement is shown below.

Signal name	PLC axis No.					
	1st axis	2nd axis	3rd axis	4th axis	5th axis	6th axis
(Not used)	R1n + 0	R2n + 0	R3n + 0	R4n + 0	R5n + 0	R6n + 0
Status	R1n + 1	R2n + 1	R3n + 1	R4n + 1	R5n + 1	R6n + 1
Alarm details	R1n + 2	R2n + 2	R3n + 2	R4n + 2	R5n + 2	R6n + 2
Control signal	R1n + 3	R2n + 3	R3n + 3	R4n + 3	R5n + 3	R6n + 3
Axis specification **(Note)**	R1n + 4	R2n + 4	R3n + 4	R4n + 4	R5n + 4	R6n + 4
Operation mode	R1n + 5	R2n + 5	R3n + 5	R4n + 5	R5n + 5	R6n + 5
Feedrate	R1n + 6 R1n + 7	R2n + 6 R2n + 7	R3n + 6 R3n + 7	R4n + 6 R4n + 7	R5n + 6 R5n + 7	R6n + 6 R6n + 7
Movement data	R1n + 8 R1n + 9	R2n + 8 R2n + 9	R3n + 8 R3n + 9	R4n + 8 R4n + 9	R5n + 8 R5n + 9	R6n + 8 R6n + 9
Machine position	R1n + 10 R1n + 11	R2n + 10 R2n + 11	R3n + 10 R3n + 11	R4n + 10 R4n + 11	R5n + 10 R5n + 11	R6n + 10 R6n + 11
Remaining distance	R1n + 12 R1n + 13	R2n + 12 R2n + 13	R3n + 12 R3n + 13	R4n + 12 R4n + 13	R5n + 12 R5n + 13	R6n + 12 R6n + 13

(Note) Axis specification is valid only for buffering mode.

10.3.4 Details of PLC Axis Control Information Data

(1) Status

The status is set by the NC to indicate the execution status of this function instruction and the status of the axis being controlled.

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
R1n + 1 (1st axis)																

bit 0:	busy	Command processing	bit 8 :	oper	Option error
1:	den	Axis movement completed	9 :		
2:	move	Axis moving	A :		
3:	SA	Servo ready	B :		
4:	svon	Servo ON	C :		
5:	ZP	Reference position reached	D :		
6:	IMP	In in-position	E :	ALM2	Axis in control alarm
7:	WAIT	Axis movement wait	F :	ALM1	Control information data designation alarm

bit 0: busy Command processing

This turns ON when the command is being processed.
The next command is not received while this bit is ON.
The next command to be issued is received while this bit is OFF.

bit 1: den Axis movement completed

This bit turns ON when the initialization and commanded movement are completed.
This bit stays OFF during movement, even when an interlock is applied.
This bit turns ON at reset or servo OFF, or when PLC axis control valid is "0".

bit 2: move Axis moving

This bit turns ON when the machine is moving, and turns OFF when the machine is stopped.

bit 3: SA Servo ready

This bit turns ON when the servo is ready.
It turns OFF during emergency stops and servo alarms.

bit 4: svon Servo ON

This bit turns OFF when a servo OFF signal is output.
It also turns OFF during emergency stops and servo alarms.
Machine movement is possible when this signal is ON.

bit5: ZP Reference position reached

This bit turns ON when the reference position is reached after completion of a reference position return.
It turns OFF when the machine moves.

bit 6: IMP In in-position

This bit turns ON when the PLC axis is in the in-position state, and turns OFF when not in the in-position state.

bit7: WAIT Axis movement wait

This bit turns ON in the buffering mode when the axis movement of the previous block has been completed, and the machine is in a WAIT status. It turns OFF when the previous block movement is completed and the movement of the next block begins.

bit 8: oper Option error

This bit turns ON when an attempt is made to execute PLC axis control when there is no PLC axis control option.

bit E: ALM2 Axis in control alarm

This bit turns ON when an alarm occurs (such as a servo alarm) during execution of axis control. Axis control cannot be executed while this bit is ON.
After the cause of the alarm has been removed, turn the bit OFF by outputting a reset signal, setting PLC axis control valid signal to 0, or turning the power OFF then ON again.

(Note) When alarms occur during axis control, the same alarms appear in the screen as for NC control axes. Set the PLC 1st axis to "1", and the PLC 2nd axis to "2".

Example: When a servo alarm occurs for the PLC 1st axis

S03 Servo alarm 52 1
PLC axis

bit F: ALM:1 Control information data designation alarm

This bit turns ON when the designated details of the control information data are illegal. When an alarm occurs for the simultaneous operation of buffering mode and single mode, the mode designated later is not executed and the mode designated first continues processing.
PLC axis control is not executed when any other alarm occurs. Turn the bit OFF by correcting the data, outputting a reset signal, or setting PLC axis control valid signal to 0.

(Note) The status of the PLC axis, to which the PLC axis control valid signal is ON, is automatically updated.

(2) Alarm details

The alarm Nos. of status ALM1 and ALM2 are set.

	F 8	7 0
R1n + 2 (1st axis)	ALM1 Alarm No.	ALM2 Alarm No.

The details of each alarm No. are shown below.

ALM1 (Control information data designation alarm)

Alarm No.	Details
01	Control signal illegal (A signal other than a registered control signal has been commanded.)
02	Axis No. illegal
03	Operation mode illegal (0 to 6)
04	Movement data range exceeded -99999999 to +99999999
05	
06	
07	
10	Reference position return not complete (absolute value command not possible)
11	
12	Command modes overlap **(Note 1)**

ALM2 (Axis in control alarm)

Alarm No.	Details
0	Servo alarm (Alarm No. is displayed in the PLC axis monitor screen. Refer to the Drive Unit Maintenance Manual for details.)
1	Z-phase not passed
2	Soft limit (+)
3	Soft limit (-)

(Note 1) This alarm occurs before the simultaneous operation of single mode and buffering mode.
(Note 2) The alarm details of the PLC axis, to which the PLC axis control valid signal is ON, are automatically updated.

(3) Control Signals (PLC axis control information data)

Control signals such as start, interlock, reset, axis removal and axis removal 2 are designated for the PLC axis.

	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0
$R1_n$ + 3 (1st axis)																

bit 0: Start	bit 8 : Absolute value command
1: Interlock	9 :
2: Reset	A :
3: Servo OFF	B :
4: Axis removal	C :
5: Axis removal 2	D :
6:	E :
7:	F :

bit 0: Start

Starting begins at the rising edge (OFF -> ON) of the start signal, based on the control information data.
The axis does not move during interlock, servo OFF, axis removal and axis removal 2.
Movement starts after interlock, servo OFF, axis removal and axis removal 2 are canceled.
Start is invalid during resetting.

bit 1: Interlock

The moving PLC axis executes a deceleration stop when the interlock signal turns ON.
The stopped PLC axis will resume movement when the interlock signal turns OFF (is canceled).

bit 2: Reset

The PLC axis is reset when the reset signal turns ON.
Moving PLC axes will execute a deceleration stop.
Commands and controls are invalid during resetting.
If the reset signal turns ON during an alarm occurrence, the alarm will be cleared.

bit 3: Servo OFF

The PLC axis will execute a deceleration stop and its servo will turn OFF when the servo OFF signal turns ON.
Whether the PLC axis movement is compensated during servo OFF can be selected in the basic specification parameter "#1064 svof".
A servo ON status will result when the power is turned ON.

bit4: Axis removal

The axis will execute a deceleration stop, and a servo OFF status will result, when the axis removal signal turns ON.
A servo ON status will result and the stopped PLC axis will resume movement when the axis removal signal turns OFF (is canceled).
Axis removal is validated when either this signal or machining parameter and axis parameter "#8201 Axis Removal" is validated.
The reference position return will become incomplete when the axis is removed. Therefore, a dog-type reference position return must be completed again when starting with an absolute value command.

bit 5: Axis removal 2

The axis will execute a deceleration stop, and a servo OFF/ready OFF status will result, when the axis removal 2 signal turns ON.
A servo ON/ready ON status will result for the stopped PLC axis when the axis removal 2 signal turns OFF (is canceled).
A restart must be executed to start the movement again.
Position control cannot be carried out while the axis removal 2 signal is ON. However, position detection is possible so the position will not be lost.

bit 8: Absolute value command

Turn this bit ON when the movement data is commanded in absolute values.
When this bit is OFF, the commands will be processed as incremental value commands.

(4) Axis specification

PLC axis No. is designated.

R1n+4 (1st axis)	Axis specification

0: 1st axis
1: 2nd axis
2: 3rd axis
3: 4th axis
4: 5th axis
5: 6th axis

Only buffering mode allows the axis specification upon this data.
Single mode, which provides the PLC axis control valid signal for each axis, does not allow the axis specification upon this data.

(5) Operation Mode

The operation mode for the PLC axis is designated.
For example, in the handle mode, R1n+5=6 (DATA) is set.

$R1_n + 5$ (1st axis)	Operation mode

(Numeric value data)

0: Rapid traverse (G0)
1: Cutting feed (G1)
2: Jog feed (+)
3: Jog feed (-)
4: Reference position return (+)
5: Reference position return (-)
6: Handle feed

The axis movement will not be affected by changing the operation mode, even while the axis is moving.
The new operation mode is validated at the next start.

(6) Feedrate

When the operation mode is cutting feed or jog feed (operation mode, register = 1 to 3), the PLC axis feedrate is designated with a binary code.

$R1_n + 6$ $R1_n + 7$	Feedrate
(1st axis)	

Designation value : 1 to 1000000 mm/min. (0.1 to 100000 inch/min.)

(Note 1) The speed unit is fixed regardless of the unit system set with the command unit.
(Note 2) The feedrate designated in the parameters is used for the rapid traverse mode and reference position return mode.
(Note 3) The feedrate can be changed during axis movement. In that case, change using a direct feedrate data (R1n + 6, 7) is possible.

(7) Movement Data

When the operation mode is rapid traverse or cutting feed, the movement data is designated with a binary code.

$R1_n + 8$ $R1_n + 9$	Movement data
(1st axis)	

Designation value : 0 to ±99999999

(Note 1) Refer to the explanations in "10.3.1 Specifications" "#1003 iunit" and "#1005 plcunit" for details on the unit.
(Note 2) The movement data is classified as follows by the absolute value command flag (bit 8) of the command signal.
Absolute value command flag = 0: Incremental value from the current position
Absolute value command flag = 1: Absolute value of the machine coordinate system
(Note 3) If the movement amount is changed during axis movement, the new movement amount will be validated at the next start.

(8) Machine Position

The machine position output to the machine system is expressed. The machine position becomes the Rfp (reference position) when the reference position is reached.

$R1_n + 10$	Machine position (input unit)
$R1_n + 11$	
(1st axis)	

(Note 1) Refer to the explanations in "10.3.1 Specifications" "#1003 iunit" and "#1005 plcunit" for details on the unit.
(Note 2) The alarm details of the PLC axis, to which the PLC axis control valid signal is ON, are automatically updated.

(9) Remaining Distance

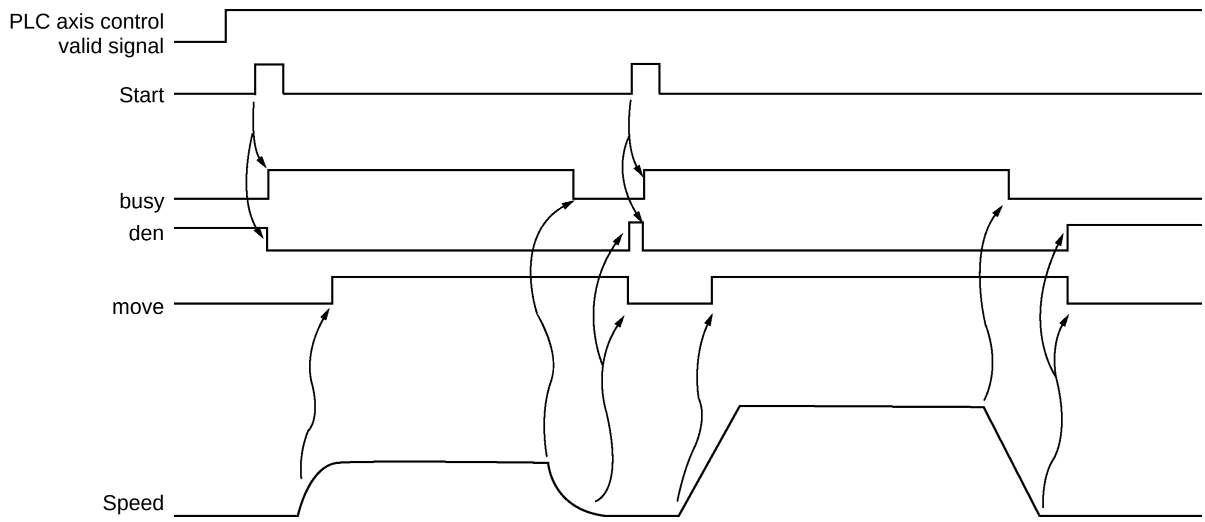
The remaining distance of the movement data output to the machine system is expressed.

$R1_n + 12$	Remaining distance (input unit)
$R1_n + 13$	
(1st axis)	

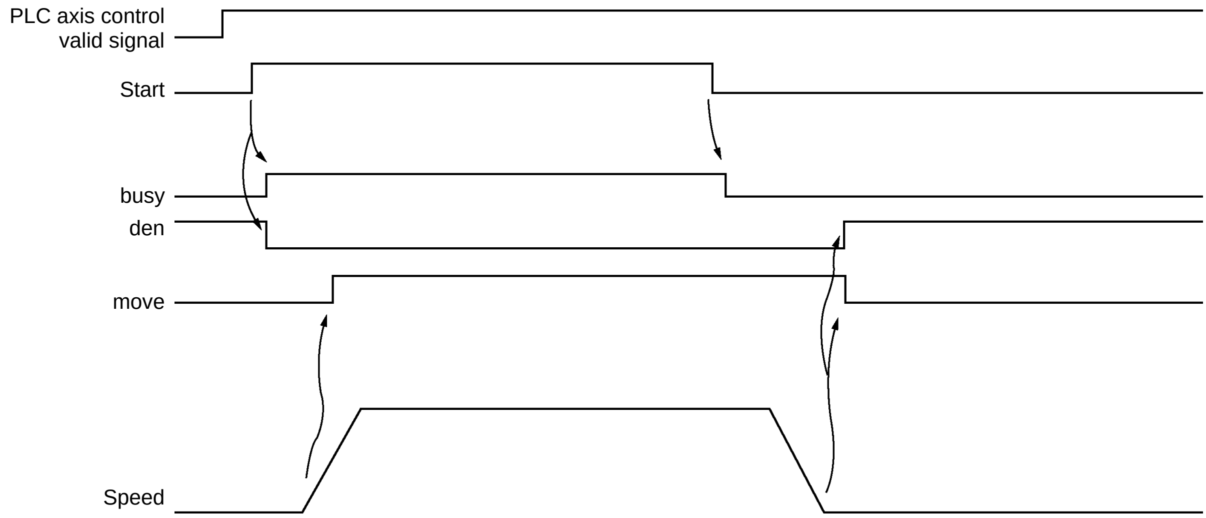
(Note 1) Refer to the explanations in "10.3.1 Specifications" "#1003 iunit" and "#1005 plcunit" for details on the unit.
(Note 2) The alarm details of the PLC axis, to which the PLC axis control valid signal is ON, are automatically updated.

10.3.5 Timing Chart

(1) For rapid traverse and cutting feed mode



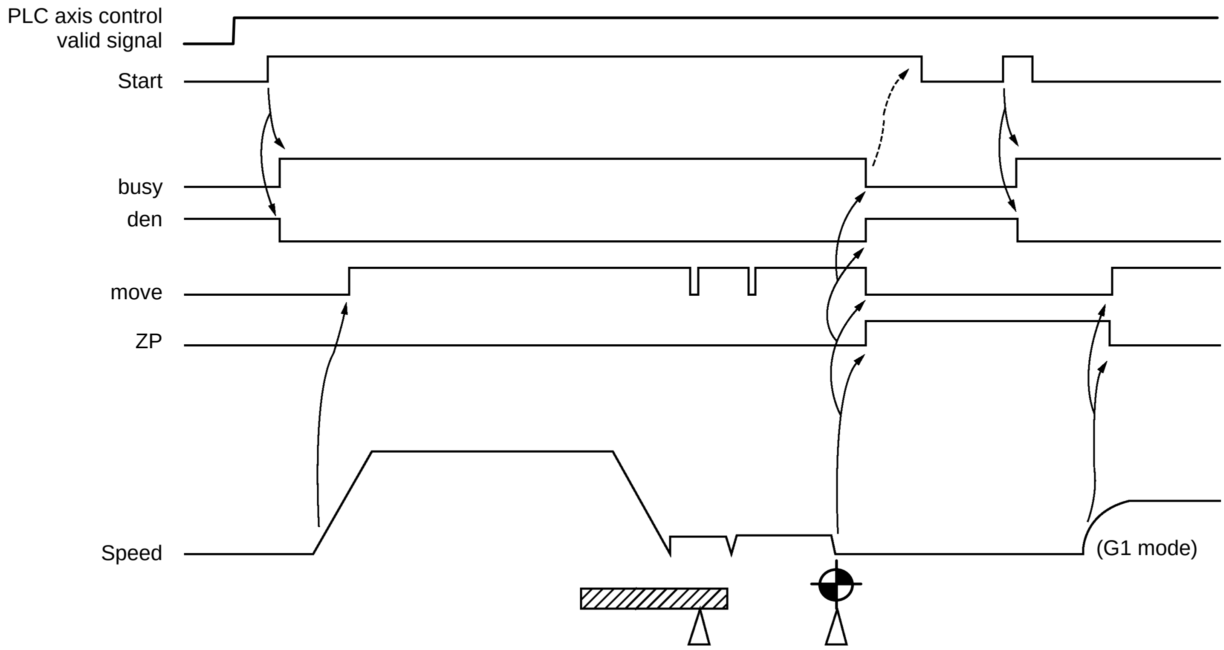
(2) For jog feed mode



(Note) The axis moves by jog feed only during start ON.

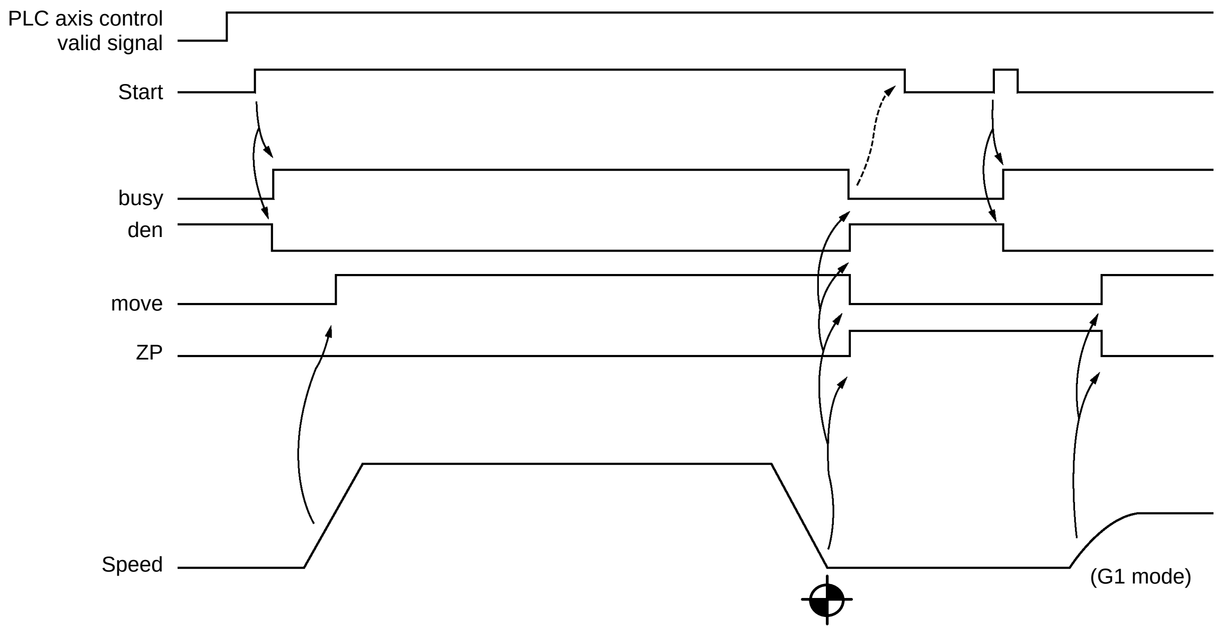
(3) For reference position return feed mode

(a) Dog-type reference position return

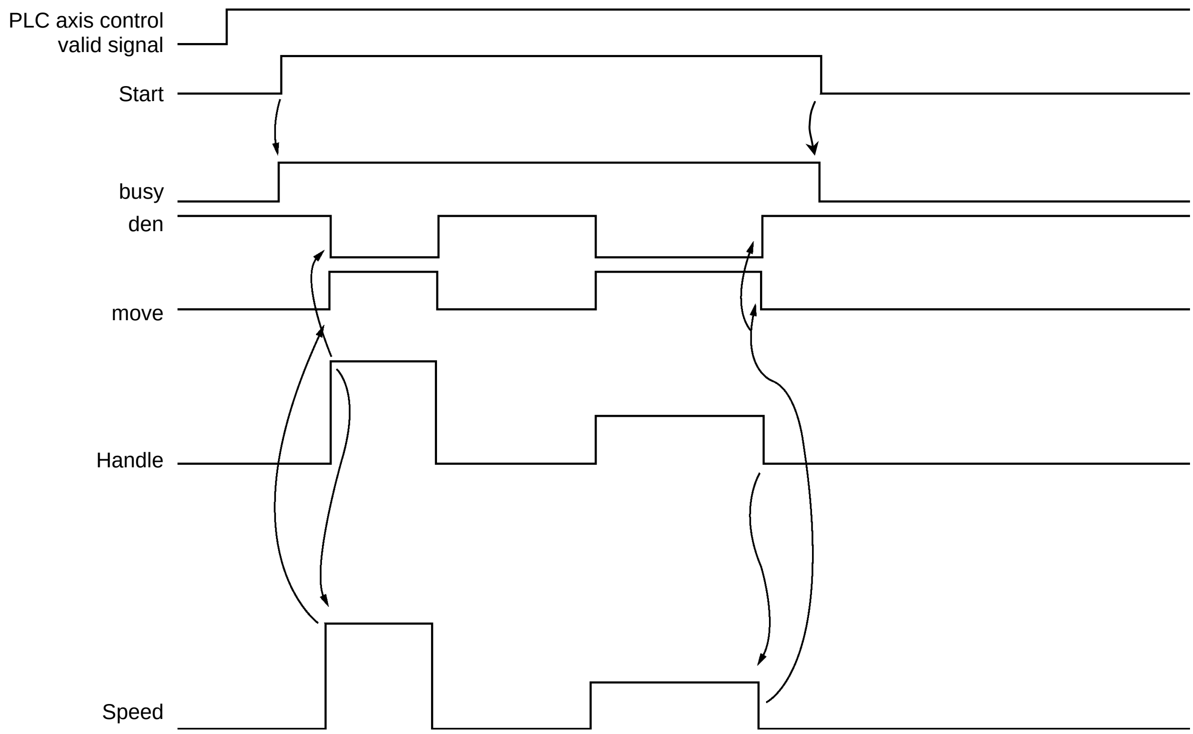


(Note 1) The axis moves by reference position return feed only during start ON. Turn the start OFF after confirming that the reference position has been reached.
(Note 2) The first reference position return after the power is turned ON is always dog-type. All returns after that are high-speed reference position returns.

(b) High-speed reference position return

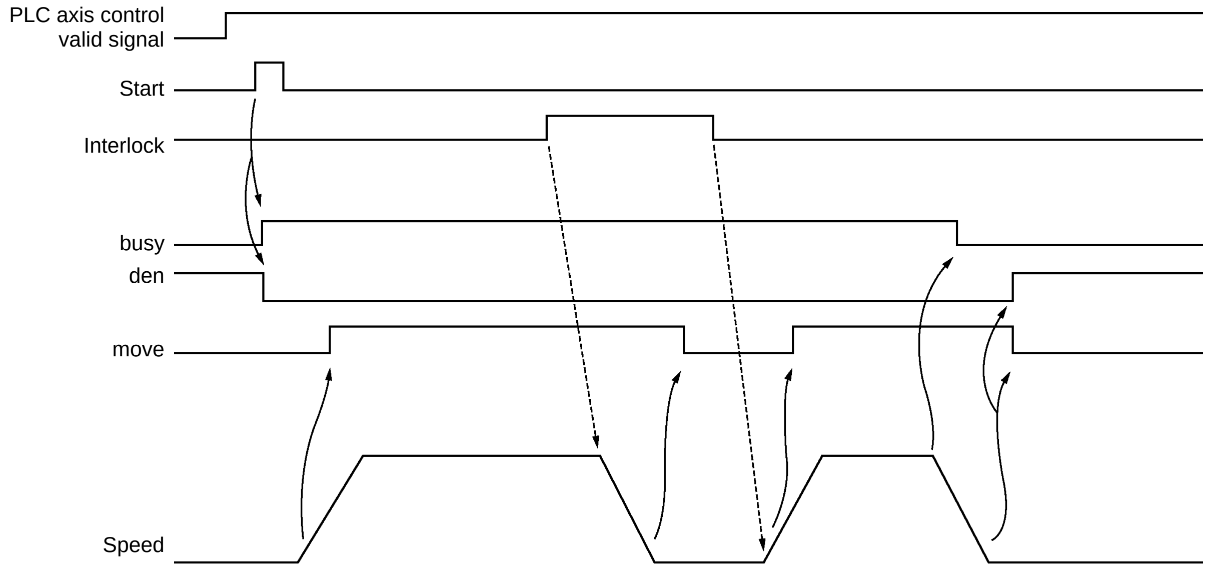


(4) For handle feed mode

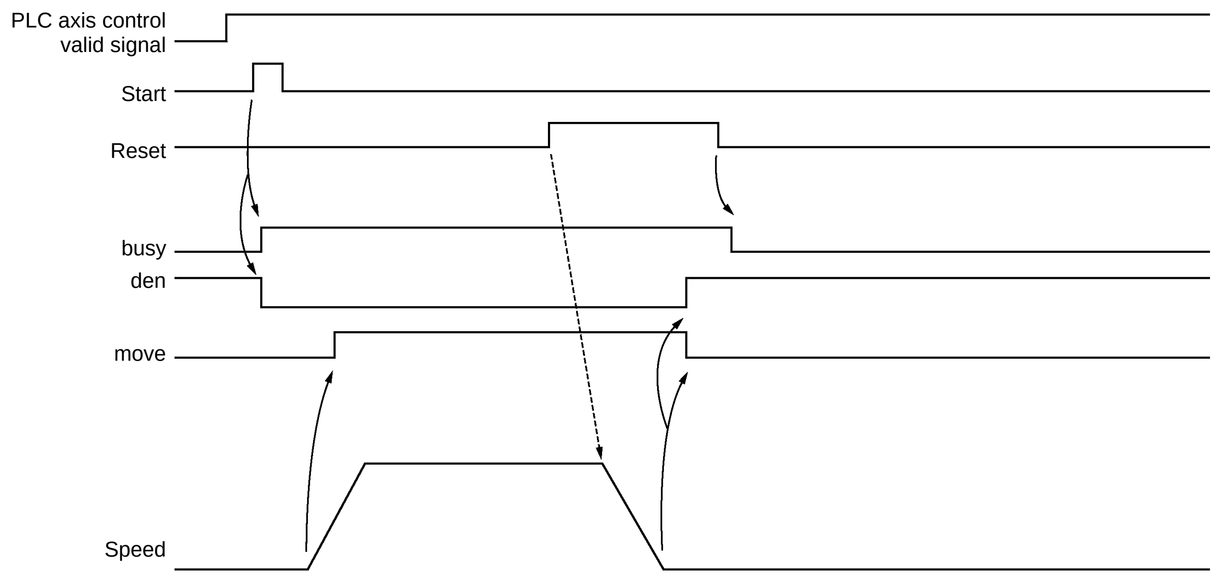


(Note) Handle feed is possible only during start ON.

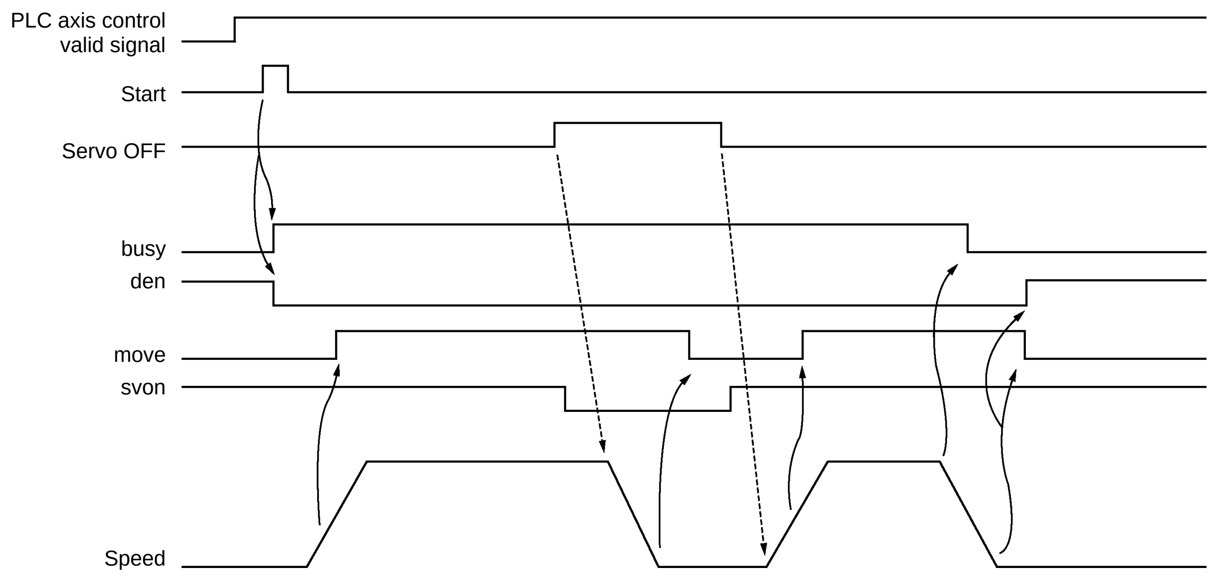
(5) When the interlock signal is ON (= 1)



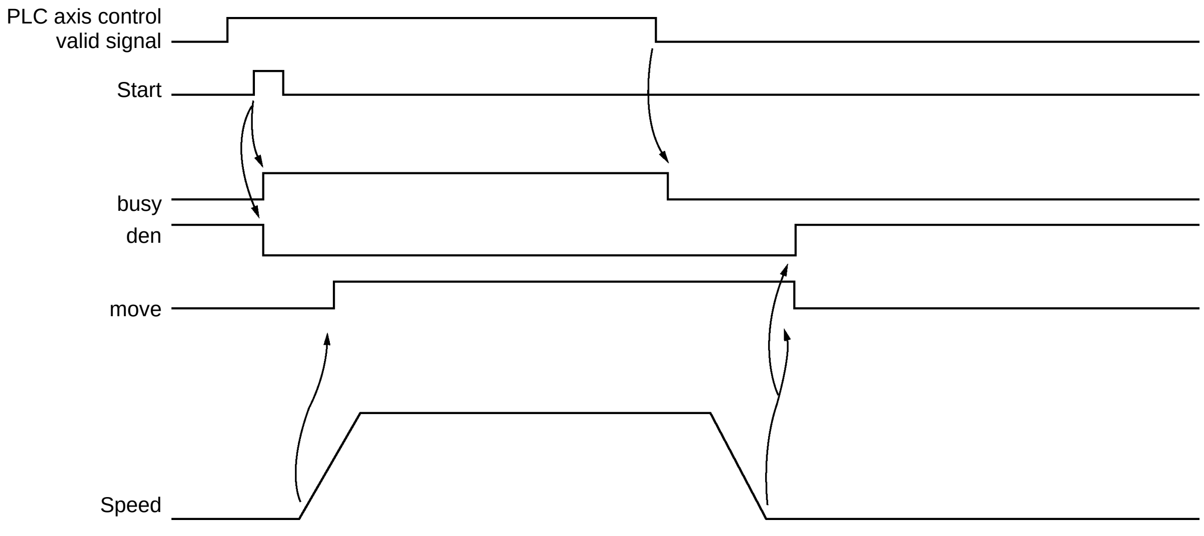
(6) When the reset signal is ON (= 1)



(7) When the servo OFF signal is ON (= 1)



(8) When the PLC axis control valid signal is OFF (= 0)



10.3.6 Reference Position Return Near Point Detection

Set the near point dog signal of the PLC axis reference position return for the following devices in the PLC.

Device No.		Signal name
Y718	*PCD1	PLC axis near point detect 1st axis
Y719	*PCD2	PLC axis near point detect 2nd axis
Y71A	*PCD3	PLC axis near point detect 3rd axis
Y71B	*PCD4	PLC axis near point detect 4th axis
Y71C	*PCD5	PLC axis near point detect 5th axis
Y71D	*PCD6	PLC axis near point detect 6th axis

(Note) The responsiveness when the dog signal is set in PLC middle-speed processing is worse than when set in PLC high-speed processing.

10.3.7 Handle Feed Axis Selection

The axis is designated for the following devices when handle feed is carried out with a PLC axis.

Device No.		Signal name
Y720	HS1P	PLC axis 1st handle valid
Y721	HS2P	PLC axis 2nd handle valid
Y722	HS3P	PLC axis 3rd handle valid

When Y720, Y721, Y722 are ON, each handle changes to PLC axis dedication.
YC40 to YC44, YC47, YC48 to YC4C, YC4F, YC50 to YC54 and YC57 used with the normal control device are used to select each handle axis.
PLC axes are counted as PLC such as first axis and second axis. Therefore, if you will operate the first handle in the first axis of PLC, turn ON Y720, YC40 to YC44 and YC47.

(Note) The handle feed magnification is also used for NC control axes.

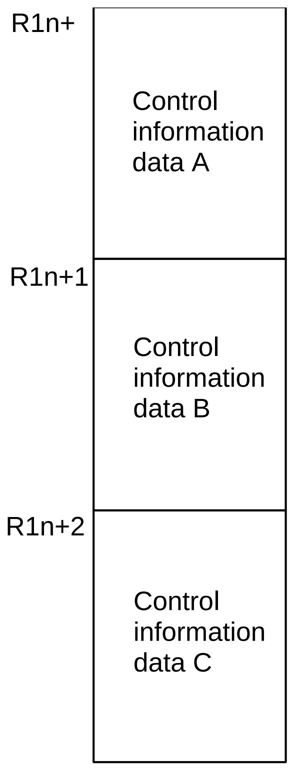
10.3.8 Single Mode

In the single mode, the command to PLC axis (control information data) is set in one block. Only one command is used for the PLC axis control.

10.3.9 Buffering Mode

In the buffering mode, the PLC axis command (control information data) is commanded to several blocks. This enables smooth changeover of commands.
Axis specifications can be set to each control information data. Up to three axes can be controlled in sequence. (Refer to "G1→ G1→ G0→ G1 (two axes)" in "(2)Timing Chart".)

(1) Control information data

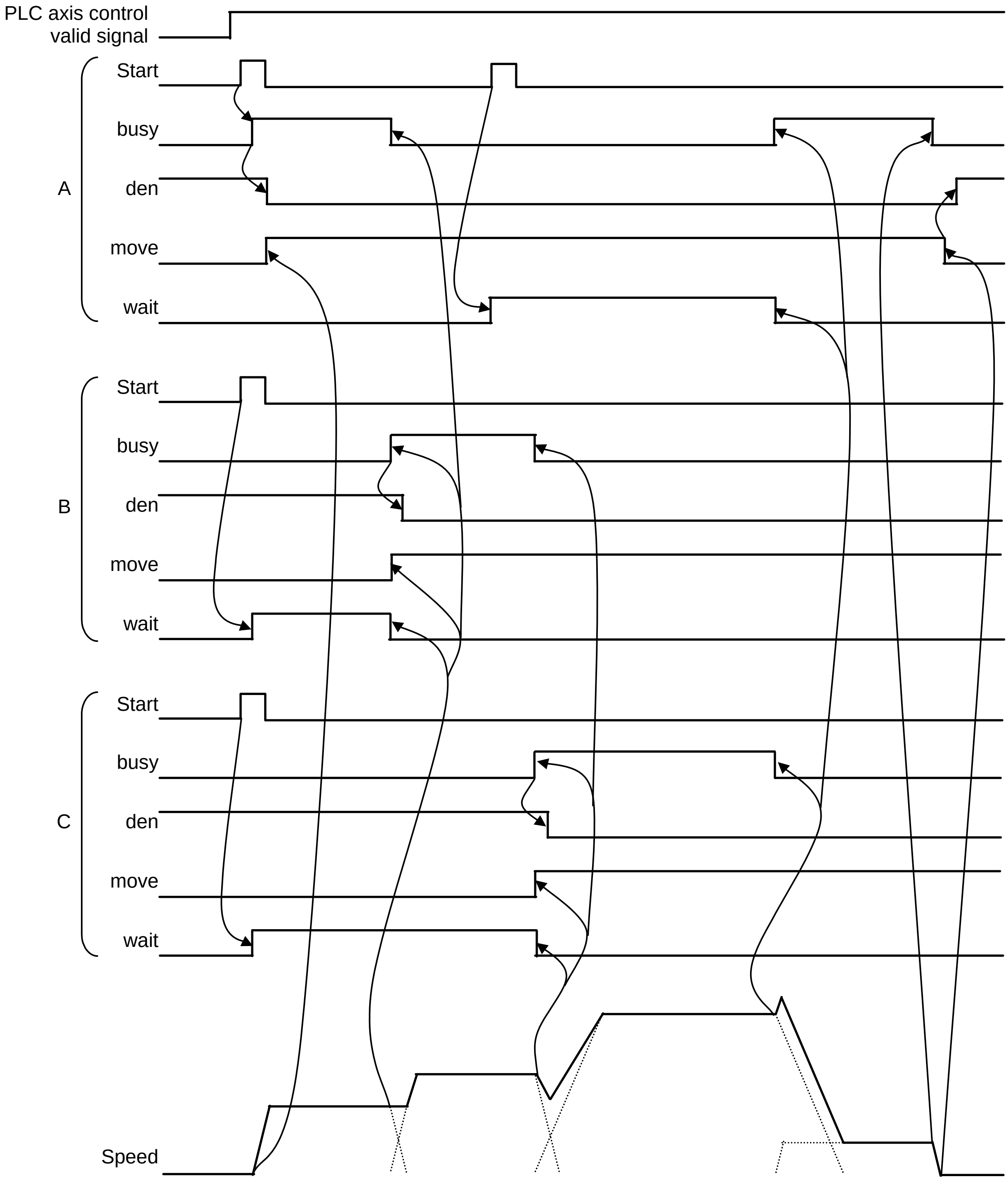


The control data for the buffering mode is set in three blocks.
The contents of each control information data have the same configuration as the normal control information data.
When executing, the axis is moved in the order that each control information is started.
When the movement is completed, the movement for the next block starts.
(If the start signals are turned ON simultaneously, the blocks are executed in the order of A, B and C.)

(Note) Only one set of the buffering mode can be commanded. If two or more sets are commanded simultaneously, the sets commanded later will cause an alarm.

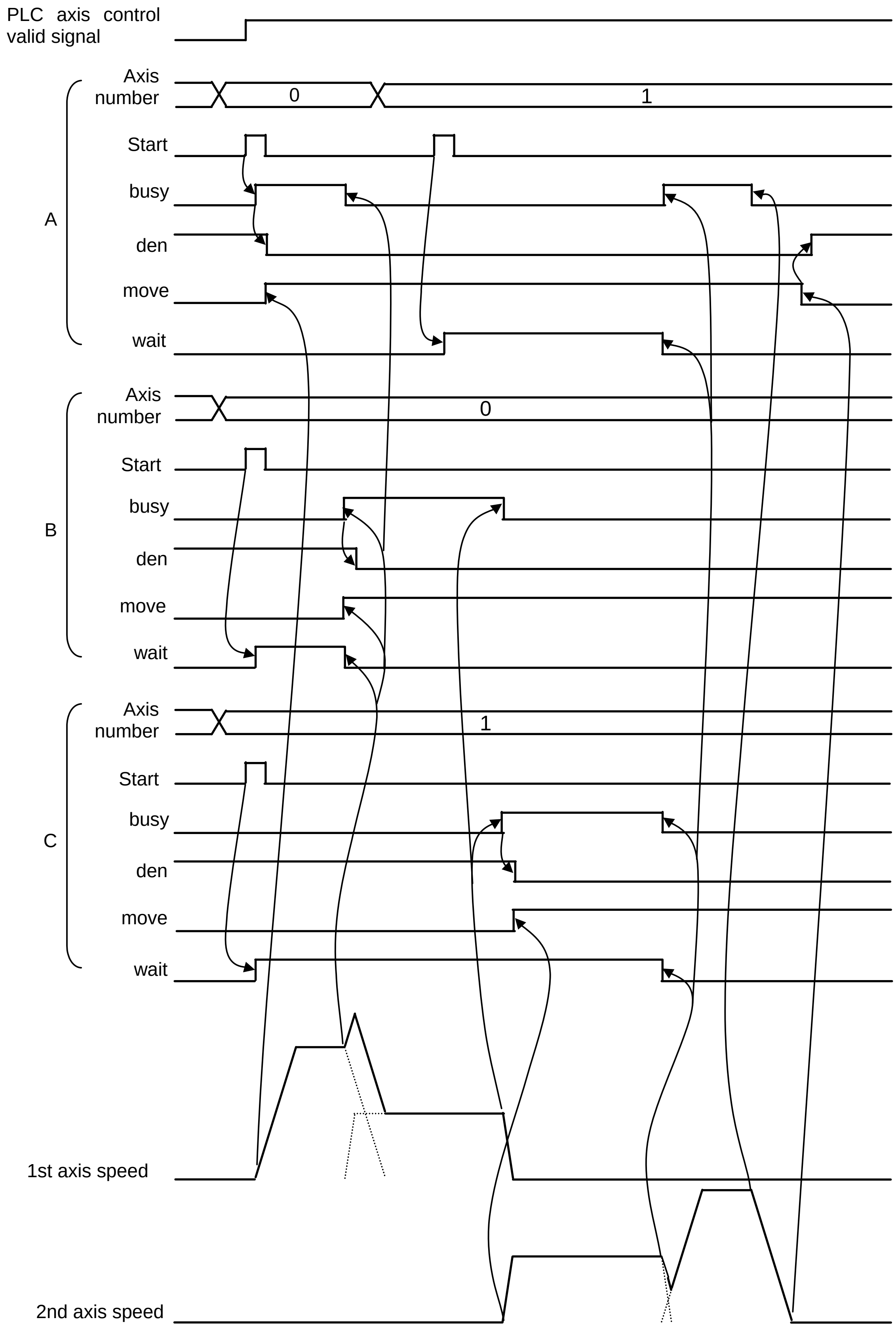
(2) Timing chart

G1→ G1→ G0→ G1 (same axis)



(Note) Change and start the data after the busy signal turns OFF.
Starting while the busy signal is ON will be ignored.

G1→ G1→ G0→ G1 (two axes)



10.3.10 PLC Axis Monitor

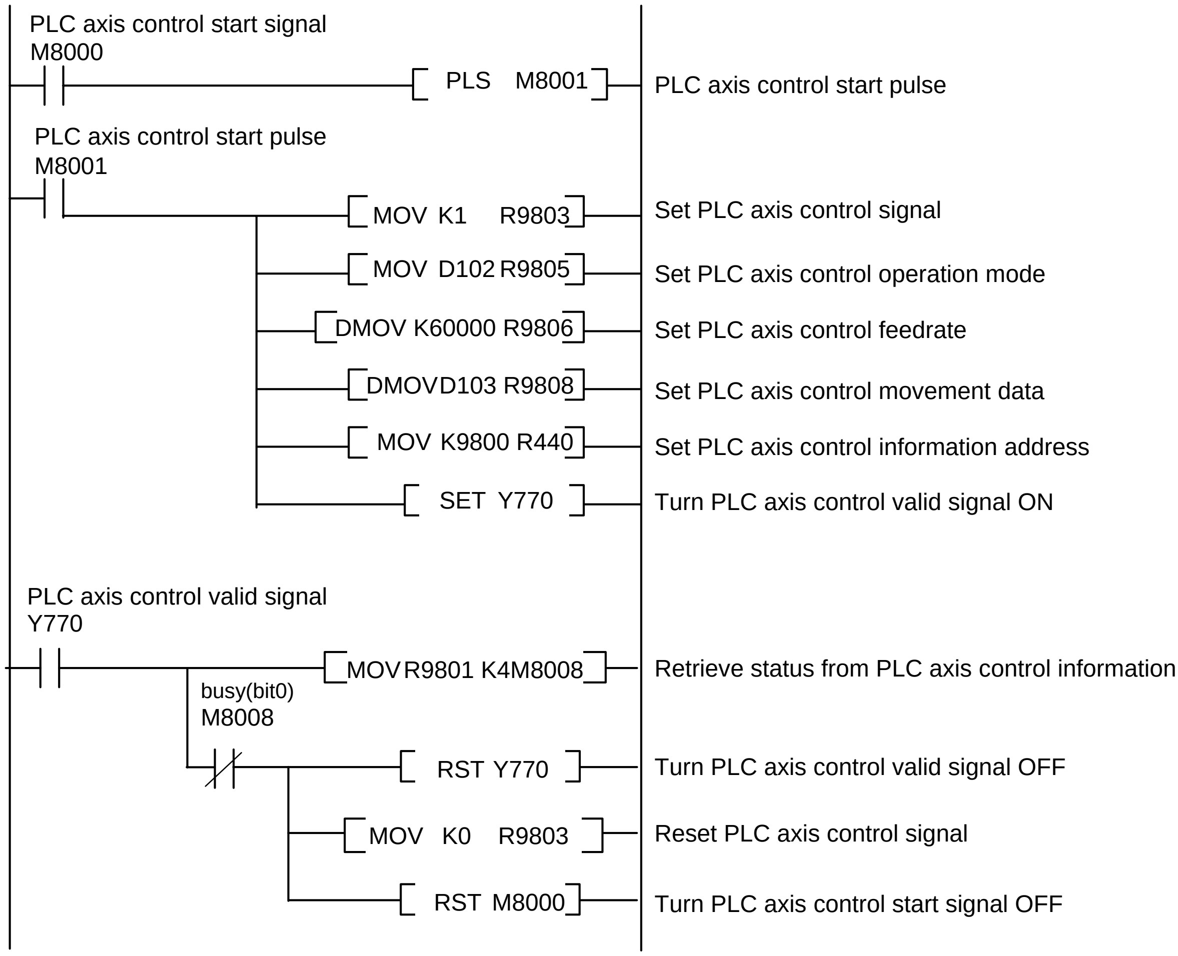
The PLC axis operation status can be confirmed by checking the PLC section on the servo monitor.

10.3.11 Absolute Position Detection

The absolute position of the PLC axis can be detected in the same manner as the NC control axis.
The initial setting operations are the same as the NC control axis. However, the following restrictions apply to the absolute position detection with the PLC axis.
(1) Automatic initialization is not possible.
(2) The verify function cannot be used.
(3) The near zero-point signal cannot be output.

10.3.12 Usage Example

The following shows an example of the PLC axis control ladder for the 1st PLC axis in single mode.



Control data	Setting value
Control signal	Start
Operation mode	D102 value
Feedrate	60 mm/min
Movement data	D103 value (Value multiplied by plcunit unit.)

10.4 External Machine Coordinate System Compensation

External machine coordinate system compensation is executed by setting compensation data (absolute amount) in the PLC file register (R) for each axis.
Thus, the compensation timing is when PLC rewrites file register (R) compensation data. Necessary condition, timing, etc., are set by user PLC.
The interface between user PLC and CNC is shown below.

File register	Details	File register	Details
R5700, 5701	$1 compensation data 1st axis	R5716, R5717	$2 compensation data 1st axis
R5702, 5703	$1 compensation data 2nd axis	R5718, R5719	$2 compensation data 2nd axis
R5704, 5705	$1 compensation data 3rd axis	R5720, R5721	-
R5706, 5707	$1 compensation data 4th axis	R5722, R5723	-
R5708, 5709	-	R5724, R5725	-
R5710, 5711	-	R5726, R5727	-
R5712, 5713	-	R5728, R5729	-
R5714, 5715	-	R5730, R5731	-

(Note) File resisters for $1 are used for models with no part systems.

Data in file registers (R5700 to R5731) is not backed up. If it must be backed up, use back-up file registers (R8300 to R9799).

(Note) The maximum delay to compensation is (one user PLC scan + 15ms). However, smoothing time constant and servo follow delay are not contained.

10.5 Alarm Message Display

The details of the alarms which occur during the sequence (user PLC) process can be displayed on the setting and display unit.
There are two types of alarm message, which can be selected with a parameter (described later).

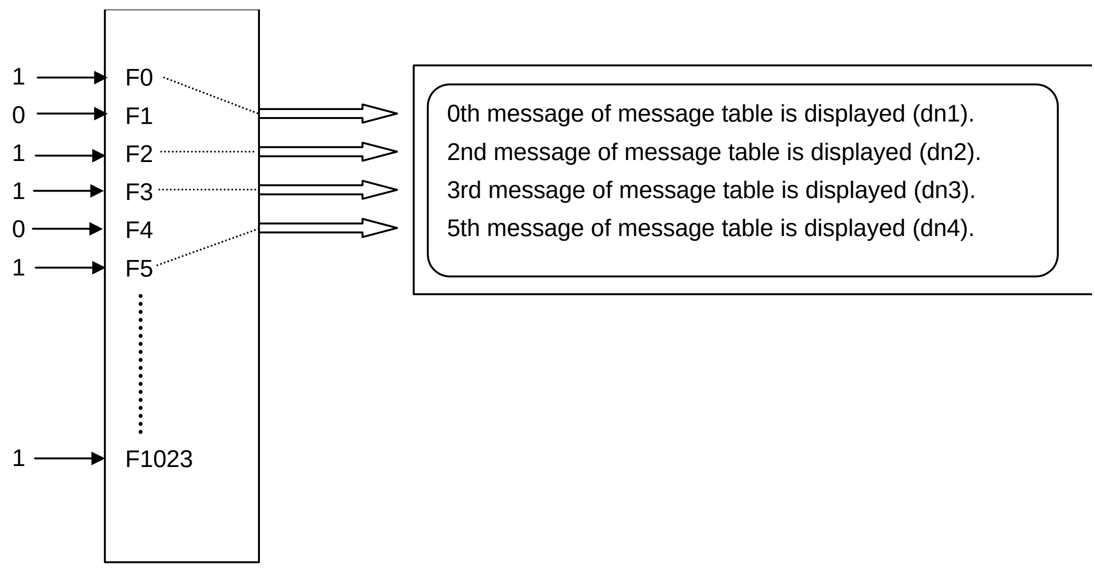
Format	Alarm message
Max. number of messages	1024 messages
Max. data length	46 bytes per message
Number of Display messages	4 messages
Interface	F type / R type (classification No. designated)
Available language	8 languages
Store method	User PLC attached data

10.5.1 Interface

The alarm message display interface is available in the two types: F type in which temporary memory F is used for message display request and R type in which file register (R) is used for message display request. Either type is selected by using a parameter.

(1) F type interface

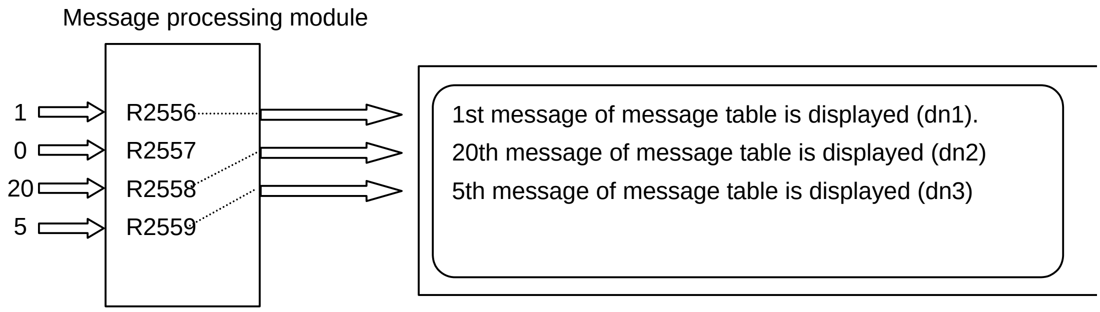
This interface applies to 1024 points of temporary memory F0 to F1023.
If temporary memory F is used as the alarm interface, do not use it for another purpose.



The highest priority is assigned to the F0 signal. The message corresponding to Fn set to 1 is fetched from the message table and displayed in order starting at F0. If no messages are prepared or Fm greater than the number of prepared messages is set to 1, the message "USER PLC ERROR m" is displayed.

(2) R type interface

This interface applies to file registers R2556 to R2559. The numeric value (binary) contained in each of the R registers indicates the position of the message to be displayed in the message table.
The message is cleared by setting the R register to 0.

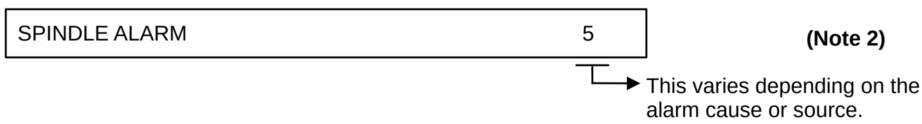


The messages are displayed starting at the message corresponding to R2556 from top to bottom.
Since message display is cleared by setting the R register to 0, No. 0 in the table message cannot be used in the R type.
If greater value than the number of prepared messages, m is set in the R register, the message "USER PLC ERROR m" is displayed.

(3) Alarm classification display (Only for Alarm message type)

Classification No. can be displayed following the message to be displayed regardless of the F or R type. (dn1 to dn4 in the figure of (1) and (2))
For example, one typical alarm message is prepared and classification No. can be used to indicate the alarm source or cause.

(Example) When spindle alarm occurs, the message "SPINDLE ALARM" is displayed and the alarm source or cause is indicated by the classification No.



For the classification No., the contents of each data register specified in alarm message preparation are displayed. Data register D0 cannot be specified.

(Note 2) The display of the classification No. by cause is updated when an alarm message display changes. It is not updated if only the contents (dn1 to dn4) of the specified data register (Dn1 to Dn4) change. If the contents of the specified data register are 0, no classification Nos. are displayed.

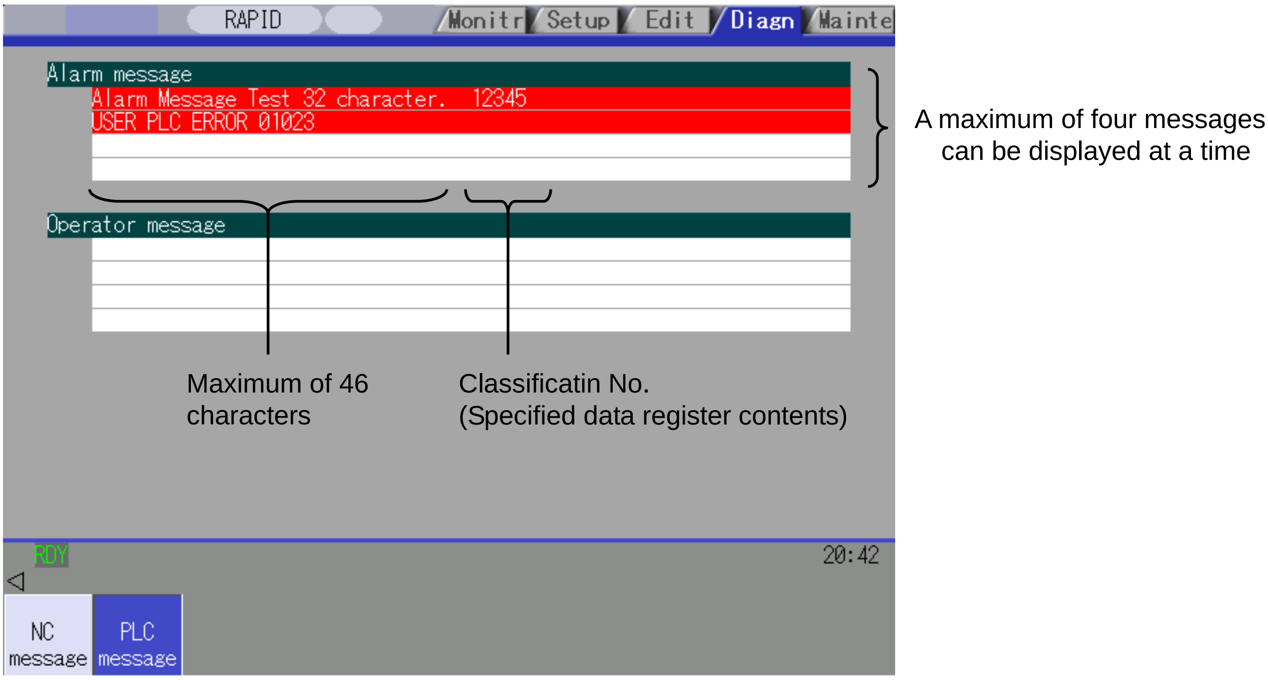
10.5.2 Screen Display

Screen display depends on the message type as described below.

(1) Alarm message type

Message length is up to 46 characters.
Alarm messages corresponding to four classification Nos. can be displayed.

Example of setting and display unit display



10.5.3 Message creation

(1) Alarm message type

Create messages by using PLC development software (GX Developer).
According to the description format, set the number of characters for one message and the number of messages to be prepared, then enter message data.
The maximum length of an alarm message is 46 characters.
A maximum of 1024 alarm messages can be prepared. For details, refer to "III PERIPHERAL DEVELOPMENT ENVIRONMENT".

10.5.4 Parameters

(1) PLC alarm message selection parameter

[Bit selection parameter screen]

7 6 5 4 3 2 1 0 ← Bit

(6450) Data (0 0 0 0 0 0 0 0)

Use No. 6450.

0: PLC alarm message display in user PLC

The operation is as the following depending on the bit state of the bit selection #6450.

Bit 6 = 0
The PLC alarm message in the user PLC is displayed as usual.

Bit 6 = 1
Do not set this value, which clears the display of PLC alarm message.

(2) Language selection parameter

There are two methods for the message data language selection. Method 2 (Method linked with language selection on the setting and display screen) is recommended although both methods are available.

(Method 1) Specify with 3 bits of bit selection parameter #6453 bit 0 to 2. (Language selection method using PLC alone)
(Method 2) Specify with display language selection parameter (Base specifications parameter #1043) (Method linked with language selection on the setting and display screen)

Each method has a different storage method. Refer to "III PERIPHERAL DEVELOPMENT ENVIRONMENT 4.4 File Name" for details.
The parameter specifications where method 1 (Language selection method using PLC alone) is applied is shown below.

[Bit selection parameter screen]

7 6 5 4 3 2 1 0 ← Bit

(6453) Data (0 0 0 0 0 0 0 0)

Use No. 6453.

Message language selection code

No.	Bit			Notes
	2	1	0	
#6453	0	0	0	The language 1 is displayed.
	0	0	1	The language 2 is displayed.
	0	1	0	The language 3 is displayed.
	0	1	1	The language 4 is displayed.
	1	0	0	The language 5 is displayed.
	1	0	1	The language 6 is displayed.
	1	1	0	The language 7 is displayed.
	1	1	1	The language 8 is displayed.

(3) F or R type selection parameter

Set the parameter on the bit selection screen of PLC parameter (setup para).

[Bit selection parameter screen]

```
                          7   6   5   4   3   2   1   0   <— Bit
# (6450)   Data   ( 0   0   0   0   0   0   *   1 )
```

Use No. 6450.

Bit 0:
0: Alarm message invalid.
1: Alarm message valid.

Bit 1:
0: F type interface
1: R type interface

[Reference] #6450 corresponds to the high-order byte of the file register R7824.

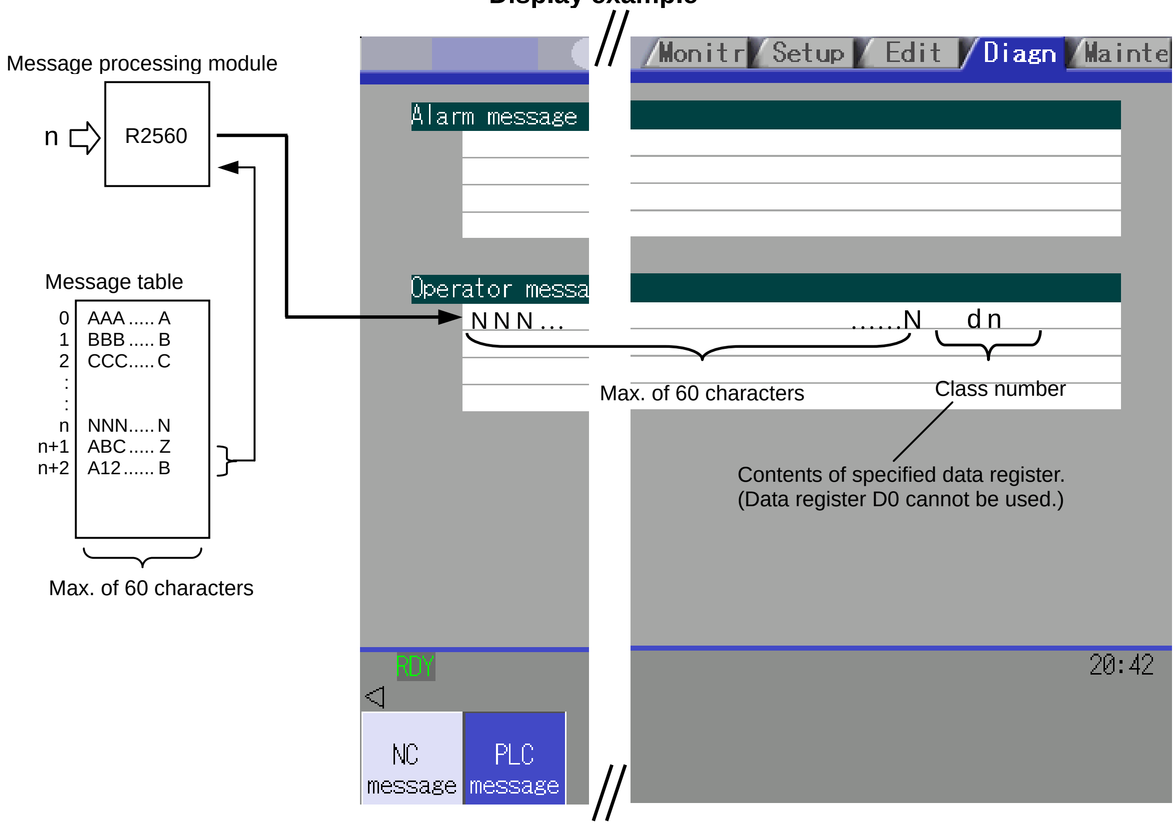
10.6 Operator Message Display

When a condition to inform the operator of a message occurs, an operator message can be displayed independently of an alarm message.
A maximum of 60 characters can be displayed for the operator message on the alarm diagnosis screen. One operator message can be displayed at a time.

10.6.1 Interface

An operator message is displayed by setting the No. of the operator message table to be displayed in file register R2560. It is cleared by setting R2560 to 0. Thus, No. 0 of the operator message table cannot be displayed.

Display example



As with alarm messages, the contents of the data register specified for the class No. display in operator message preparation are also displayed when creating operator message.

(Note 1) The class No. display is updated when the contents of file register R2560 change. It is not updated if only the contents of the specified data register (Dn) change.
To change the class No. display only, the contents of R2560 must be cleared to 0. If the contents of the specified data register are 0, no class Nos. are displayed.

10.6.2 Operator Message Preparation

Create messages by using PLC development software (GX Developer).
According to the description format, set the number of characters for one message and the number of messages to be prepared, then prepare message data.
The maximum length of an operator message is 60 characters. A maximum of 512 operator messages can be prepared.
However, the number of operator messages may be limited depending on the available memory capacity.
For details, refer to the section "III PERIPHERAL DEVELOPMENT ENVIRONMENT".

10.6.3 Operator Message Display Validity Parameter

Set the parameter on the bit selection screen of PLC parameter (setup para).

[Bit selection parameter screen]

```
                        7 6 5 4 3 2 1 0  ← bit
#(6450)   Data        ( 0 0 0 0 0 0 0 0 )
                                  └── 0 : Operator message display invalid.
  └── Use No. 6450                    1 : Operator message display valid.
```

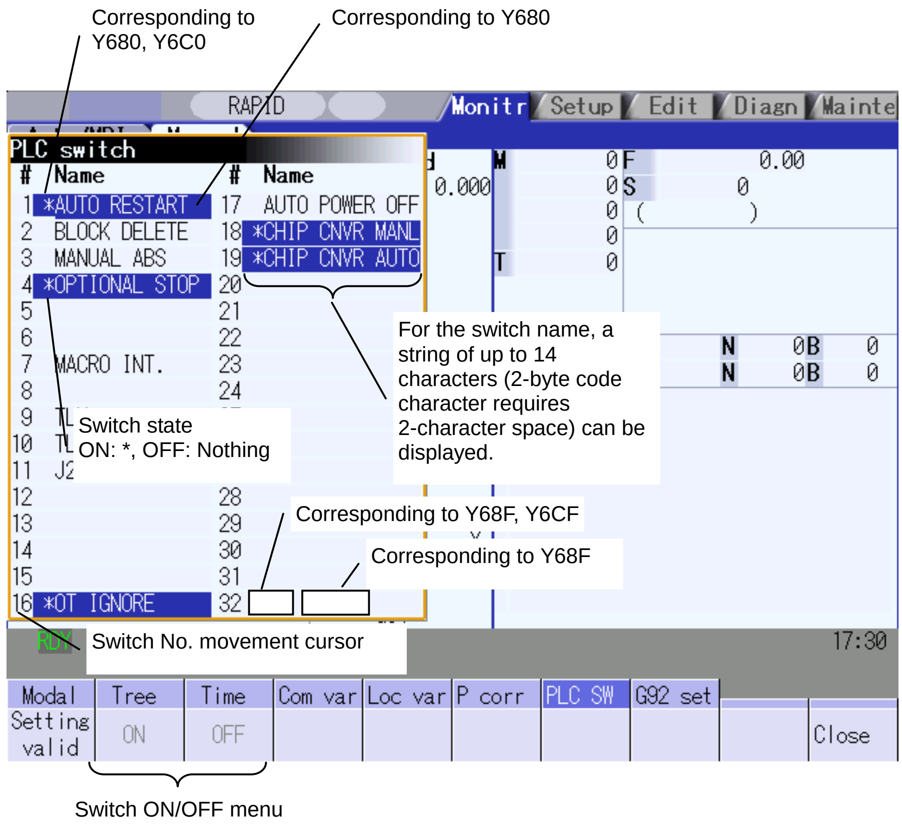
[Reference] #6450 corresponds to the high-order byte of file register R2924.

10.7 PLC Switches

Similar function to machine operation switches can be provided by using the controller setting and display unit. The number of switch points is 32. The switches can be turned ON and OFF from the PLC Switch screen or the user PLC. The switch names can be given as desired.

10.7.1 Explanation of Screen

The screen is explained below.



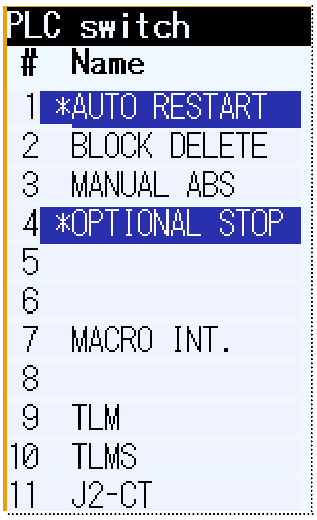
10.7.2 Explanation of Operation

To turn ON or OFF a switch, press the "setting valid" key, move to arbitrary No. with the cursor of #No., then press "ON" or "OFF" menu key.
Depending on the state of the switch, its input device X is turned ON or OFF and accordingly the switch mark indicates the ON or OFF state.

To turn block start interlock ON

Move to "#4" with the cursor keys

Press the switch ON menu key.

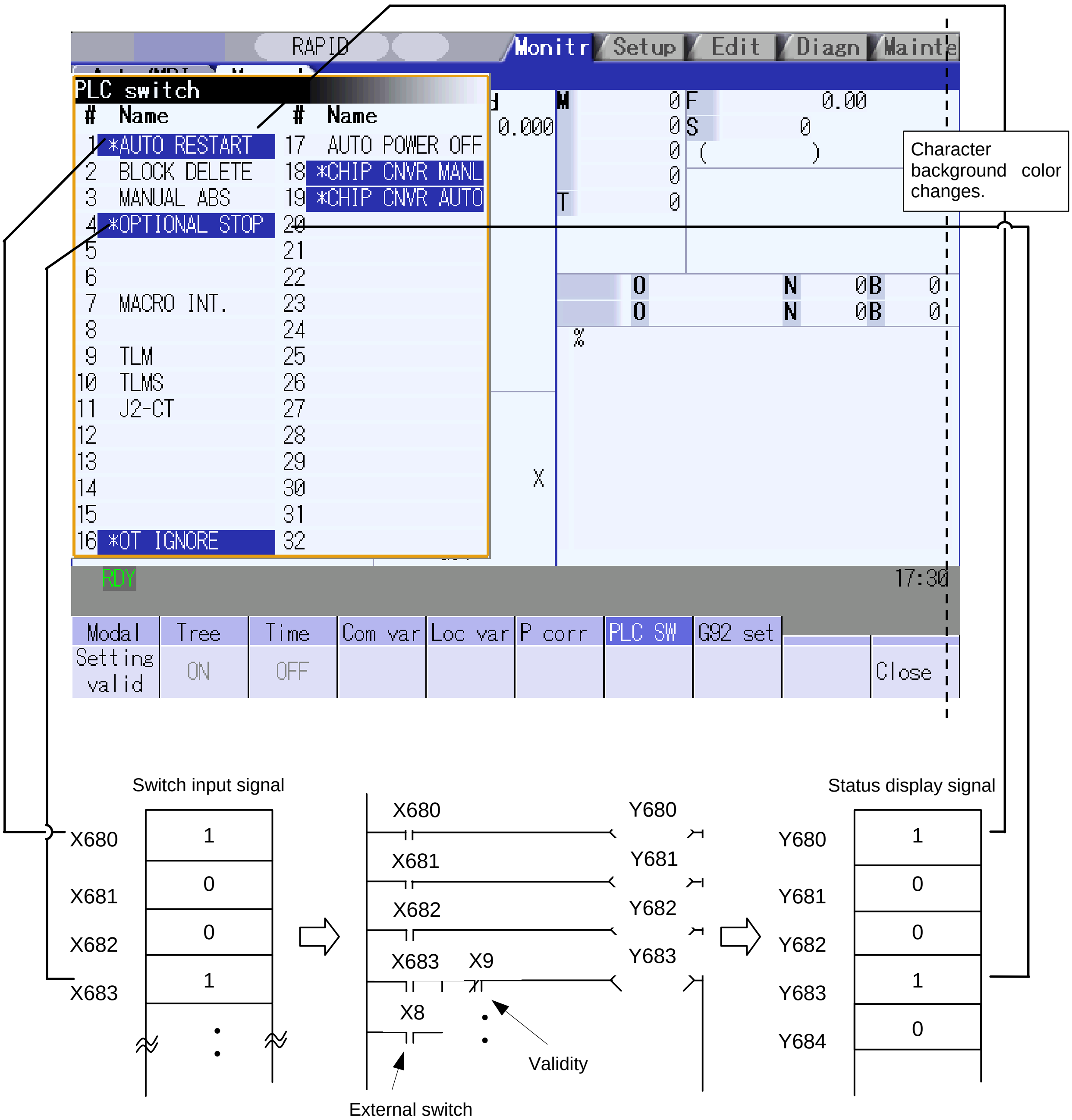


To display the switch validity state, etc., the switch name can be highlighted. To do this, turn ON or OFF output device Y corresponding to the switch name.
Reversing signal Y can reverse the switch ON/OFF states. When reversing signal Y is activated, the ON/OFF state of the corresponding switch and device X is reversed.
The corresponding table of the switch No., input device X, highlight output device Y, and reversing signal Y is listed below:

Switch No.	Corresponding device			Switch No.	Corresponding device		
	Input X	**Output Y**	**Reverse Y**		**Input X**	**Output Y**	**Reverse Y**
#1	X680	Y680	Y6C0	#17	X690	Y690	Y6D0
#2	X681	Y681	Y6C1	#18	X691	Y691	Y6D1
#3	X682	Y682	Y6C2	#19	X692	Y692	Y6D2
#4	X683	Y683	Y6C3	#20	X693	Y693	Y6D3
#5	X684	Y684	Y6C4	#21	X694	Y694	Y6D4
#6	X685	Y685	Y6C5	#22	X695	Y695	Y6D5
#7	X686	Y686	Y6C6	#23	X696	Y696	Y6D6
#8	X687	Y687	Y6C7	#24	X697	Y697	Y6D7
#9	X688	Y688	Y6C8	#25	X698	Y698	Y6D8
#10	X689	Y689	Y6C9	#26	X699	Y699	Y6D9
#11	X68A	Y68A	Y6CA	#27	X69A	Y69A	Y6DA
#12	X68B	Y68B	Y6CB	#28	X69B	Y69B	Y6DB
#13	X68C	Y68C	Y6CC	#29	X69C	Y69C	Y6DC
#14	X68D	Y68D	Y6CD	#30	X69D	Y69D	Y6DD
#15	X68E	Y68E	Y6CE	#31	X69E	Y69E	Y6DE
#16	X68F	Y68F	Y6CF	#32	X69F	Y69F	Y6DF

(Note 1) Input devices X hold the state even if power is turned OFF.

10.7.3 Signal Processing

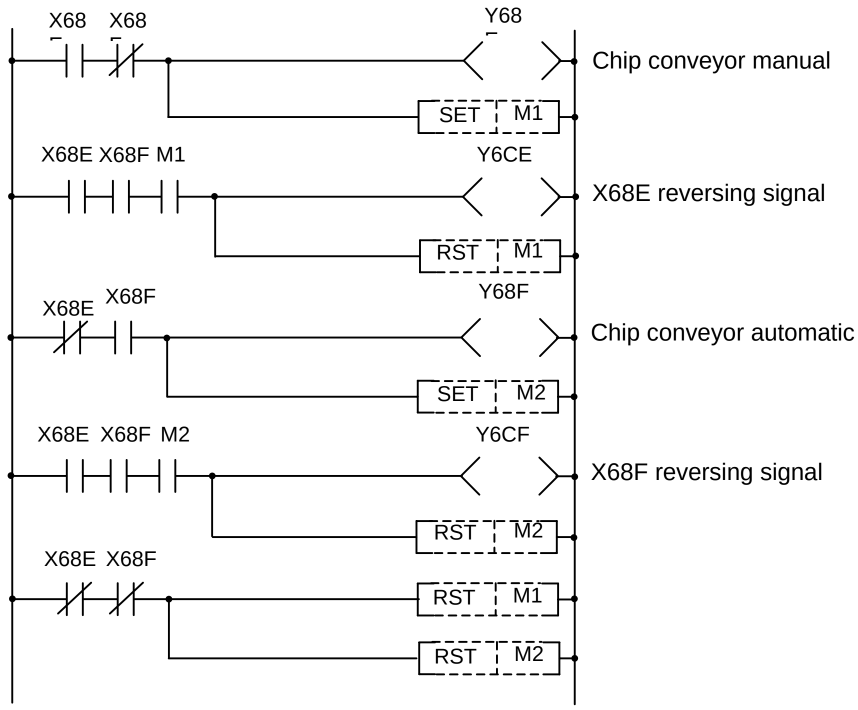


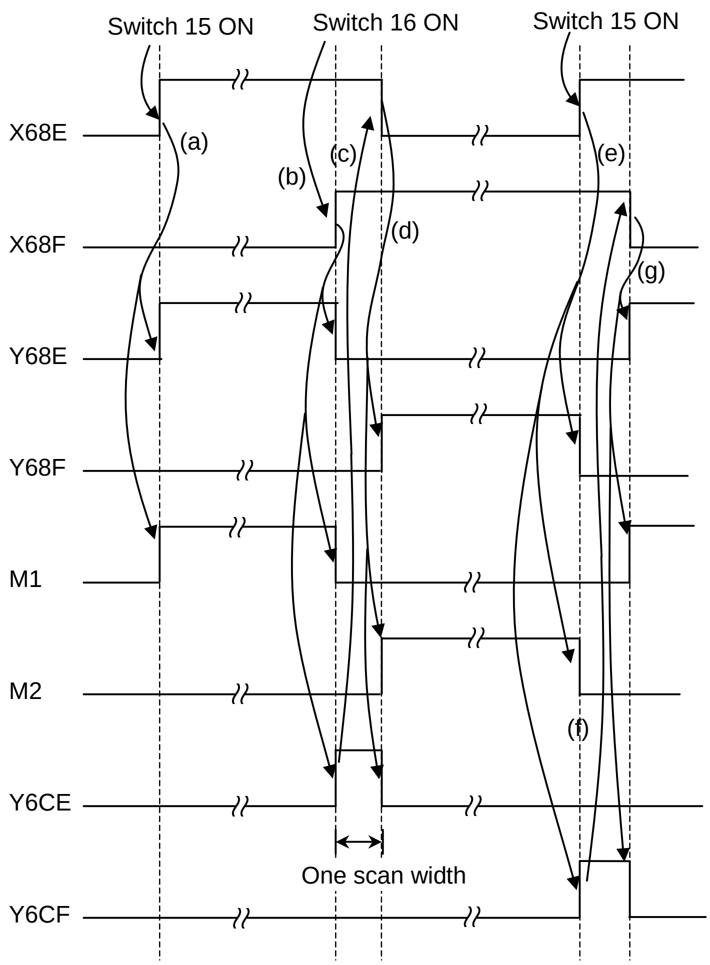
- When setting is done on the PLC switch screen, the input device X corresponding to the specified switch No. is turned ON or OFF to switch over the switch state.
- When reversing signal Y is turned ON from the user PLC, its corresponding input device X and the switch state are reversed. Reversing signal Y is reset immediately after the CNC reverses the input device X and the switch state. It is turned ON by one pulse (scan) only also in the user PLC. In either case, when output device Y is set to ON based on the input device X state, the corresponding switch name is highlighted.

The following shows an example of operation of reversing signal Y from the user PLC.

(1) Two-point switch

(Example) When two opposite switches, chip conveyer manual and chip conveyer automatic, are provided;





(a) Y68E, M1 turns ON when switch 15 (X68E) turns ON and switch 16 (X68F) turns OFF. [Default state]

(b) If switch 16 (X68F) is turned ON in state (a), Y68E turns OFF, Y6CE turns ON, and M1 turns OFF.

(c) Y6CE has turned ON, so X68E is reversed (OFF).

(d) Y6CE turns OFF, and Y68F and M2 turns ON from the X68E OFF and X68F ON state.

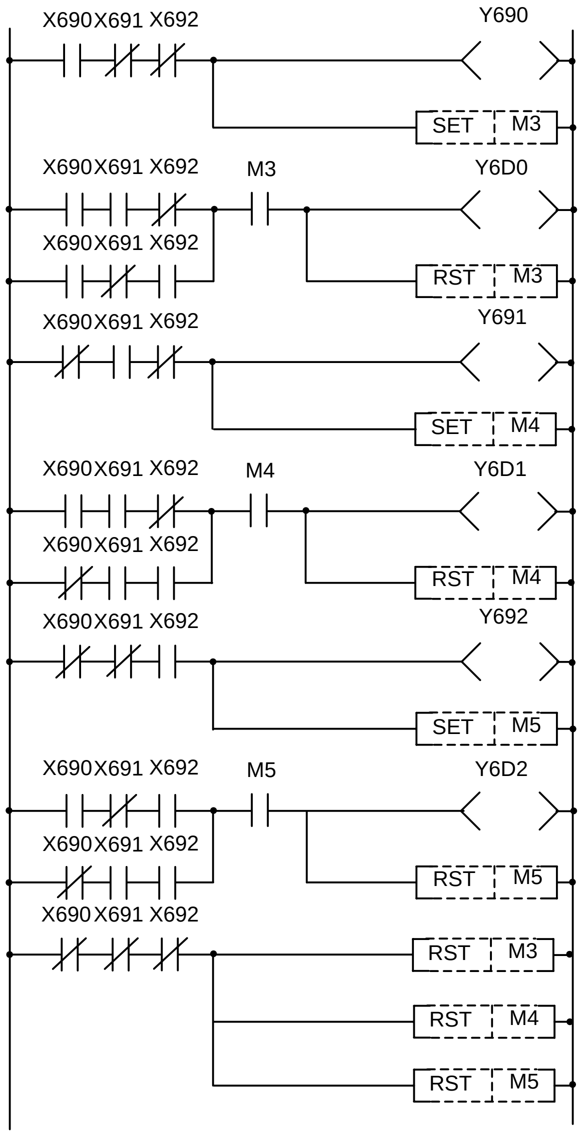
(e) If switch 15 (X68E) is turned ON in state (d), Y68F turns OFF, Y6CF turns ON, and M2 turns OFF.

(f) Y6CF has turned ON, so X68F is reversed (OFF).

(g) Y6CF turns OFF, and Y68E and M1 turns ON again from the X68F OFF and X68E ON state.

(2) Three-point switch

(Example) When three opposite switches 17, 18, and 19 are provided;



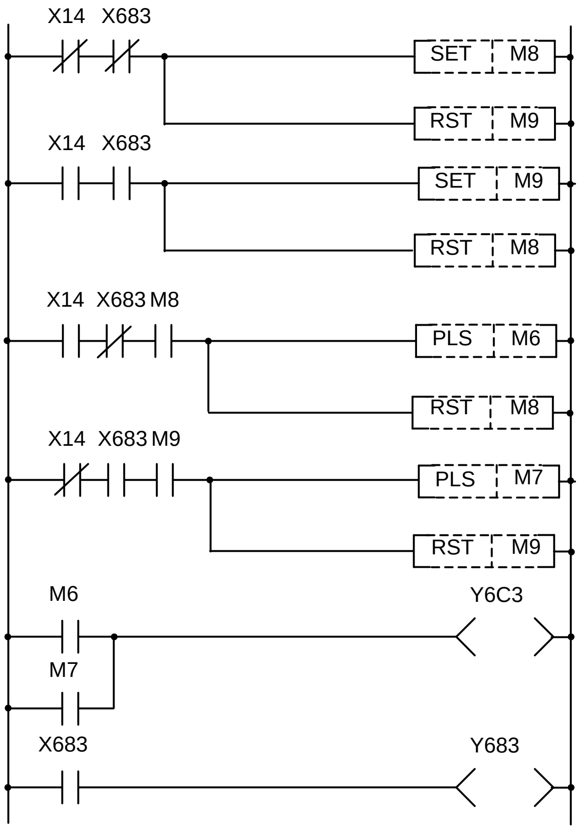
X690 turns OFF when Y6D0 turns ON.

X691 turns OFF when Y6D1 turns ON.

X692 turns OFF when Y6D2 turns ON.

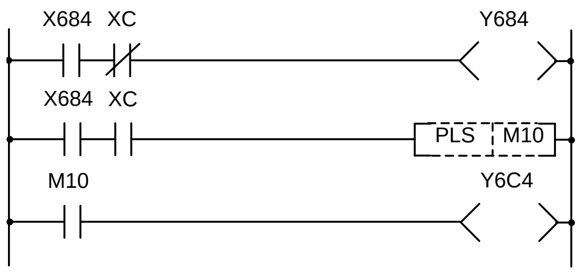
(3) External switch and PLC switch

(Example 1) When an external optional stop switch (X14) is provided;



Under sequence control in the above example, the switch marks on the PLC switch screen can be operated from both external and PLC switches.

(Example 2) When an external switch (XC) that inhibits a PLC switch handle interrupt is provided;



Under sequence control in the above example, when the external switch (XC) is ON, the PLC switch for a handle interrupt cannot be turned ON.

10.7.4 Switch name preparation

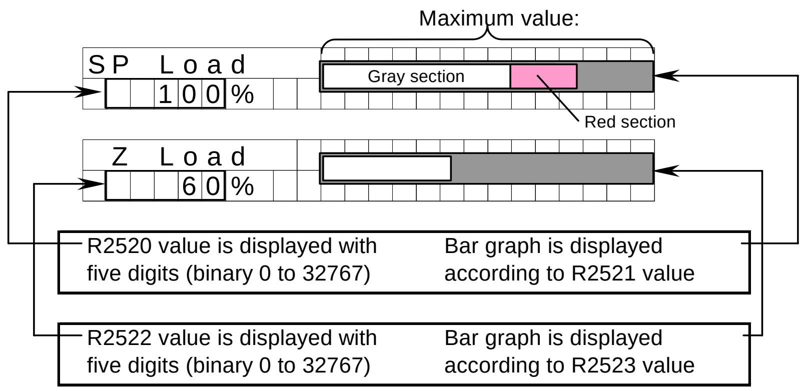
Prepare PLC switch names by using PLC development software (GX Developer).
According to the description format, set the number of characters for one switch name and the number of switch names to be prepared, then prepare switch name data. The maximum length of a switch name is 14 characters. A maximum of 32 switch names can be prepared.
For details, refer to the section "III PERIPHERAL DEVELOPMENT ENVIRONMENT".

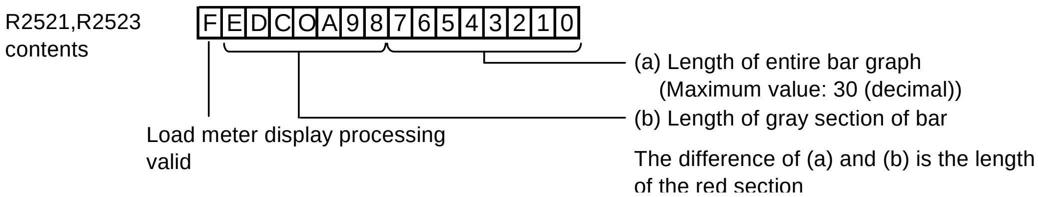
10.8 Load Meter Display

The load meter can be displayed by setting a value in the designated file register (R) with the ladder program. The spindle load, Z axis load, etc. characters and scale are created with comments in the PLC development software (GX Developer) message function.
For details, refer to the section "III PERIPHERAL DEVELOPMENT ENVIRONMENT".

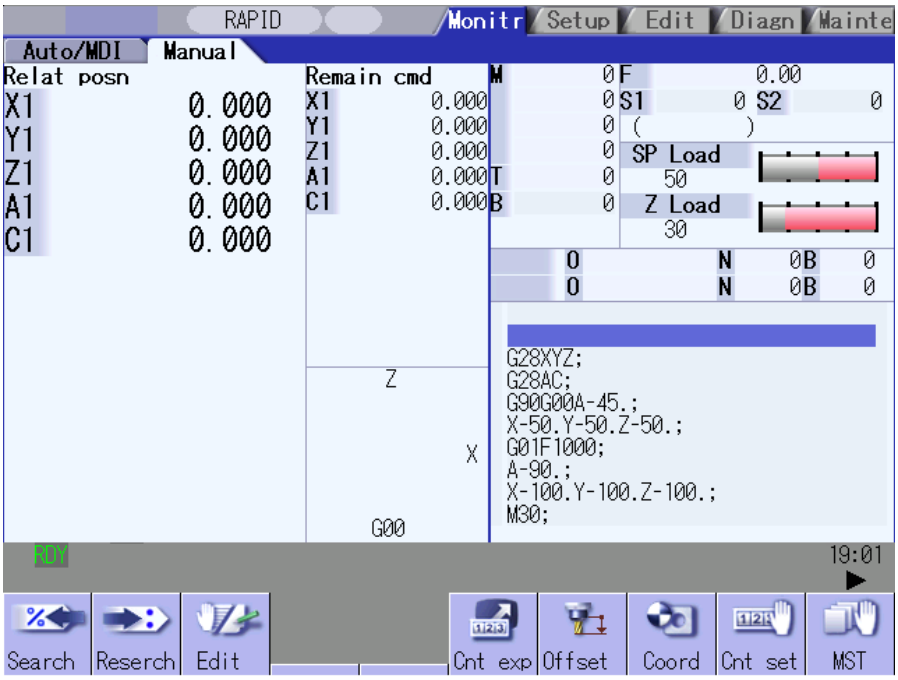
10.8.1 Interface

If Spindle/Standby is not displayed, two load meters are displayed using the four-line area for the spindle/standby and load meter. If Spindle/Standby is displayed, one load meter is displayed using the two-line area for the load meter.





Display example of setting and display unit



When the detail of R377 is "0" (detailed explanation is given later), the name of load meter is displayed by describing the following messages

Line	Message	Detail		
1	;M,2,0, spindle load $1	First	Name (Max. 10 characters)	For 1st part system
2	;M,2,0,		(Not used)	
3	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
4	;M,2,0,		(Not used)	
5	;M,2,0, Z-axis load $1	Second	Name (Max. 10 characters)	
6	;M,2,0,		(Not used)	
7	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
8	;M,2,0,		(Not used)	
9	;M,2,0, spindle load $2	First	Name (Max. 10 characters)	For 2nd part system
10	;M,2,0,		(Not used)	
11	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
12	;M,2,0,		(Not used)	
13	;M,2,0, Z-axis load $2	Second	Name (Max. 10 characters)	
14	;M,2,0,		(Not used)	
15	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
16	;M,2,0,		(Not used)	
17	;M,2,0, spindle load $3	First	Name (Max. 10 characters)	For 3rd part system
18	;M,2,0,		(Not used)	
19	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
20	;M,2,0,		(Not used)	
21	;M,2,0, Z-axis load $3	Second	Name (Max. 10 characters)	
22	;M,2,0,		(Not used)	
23	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
24	;M,2,0,		(Not used)	
25	;M,2,0, spindle load $4	First	Name (Max. 10 characters)	For 4th part system
26	;M,2,0,		(Not used)	
27	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
28	;M,2,0,		(Not used)	
29	;M,2,0, Z-axis load $4	Second	Name (Max. 10 characters)	
30	;M,2,0,		(Not used)	
31	;M,2,0,******%		Character string displayed on the right of numerical value (Max. 4 characters) (Note 1)	
32	;M,2,0,		(Not used)	

(Note 1) Setting of the character string displayed on the right of numerical value:
Of the data set as character string, only four characters, or 7th to 10th characters, are displayed on the screen.
The 1st to 6th characters will be ignored.
To have "%" displayed next to a numerical value, character string must be set as "******%". (This will be the same even if the 1st to 6th characters are the characters other than "*".)

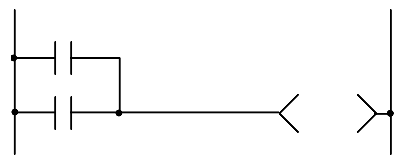
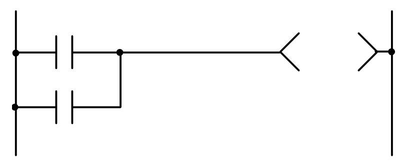
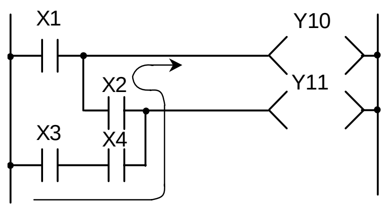
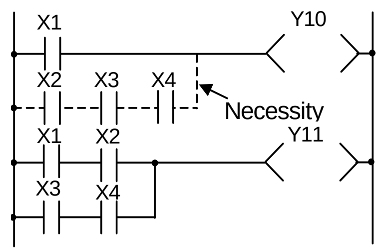
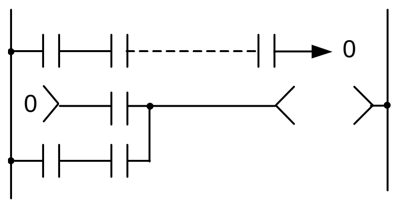
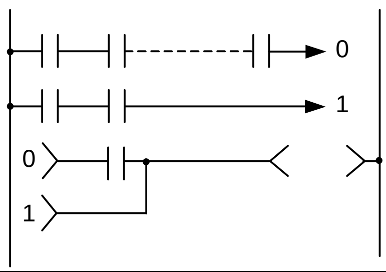
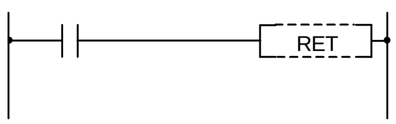
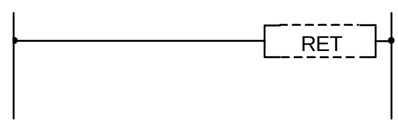
(Note 2) Designation of the name of load meter
Depending on the detail of R377 (0 to 10), the name of load meter can be changed.
In the following case, when "2" is set to R377, "spindle load 3" and "Z-axis load 3" will be displayed.

```
;M,2,0, spindle load 1
;M,2,0,
;M,2,0,******%
;M,2,0,
;M,2,0, Z-axis load 1
;M,2,0,
;M,2,0,******%
;M,2,0,
;M,2,1, spindle load 2
;M,2,1,
;M,2,1,******%
;M,2,1,
;M,2,1, Z-axis load 2
;M,2,1,
;M,2,1,******%
;M,2,1,
;M,2,2, spindle load 3
;M,2,2,
;M,2,2,******%
;M,2,2,
;M,2,2, Z-axis load 3
;M,2,2,
;M,2,2,******%
;M,2,2,
```

11. Appendix

11.1 Example of Faulty Circuit

Wrong configurations of circuits are shown below. Correct the circuitry, if any.

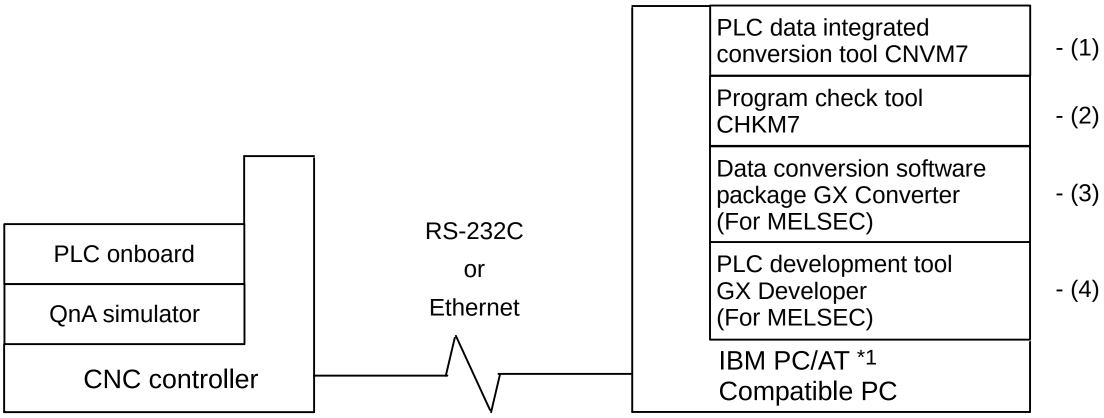
Faulty circuit producing errors →	Correct circuit
(1) Circuit containing OR 	
(2) Rounding circuit  Whether or not the Y10 condition includes X3, X4 and X2 is unknown.	
(3) Modification of loopback circuit  	
(4) Presence of a contact before RET, FEND, or MCR circuit  	

III PERIPHERAL DEVELOPMENT ENVIRONMENT

1. Outline

This CNC supports the user PLC development environment which uses the Mitsubishi integrated FA software MELSOFT Series (GX Developer), which is the PLC development tool for the Mitsubishi PLC MELSEC Series.
This manual explains user PLC development environment using GX Developer, mainly usage specific to MITSUBISHI CNC.

1.1 Software Configuration



*1: IBM PC/AT compatible personal computer with Windows OS

(1) CNVM7 (PLC data integrated conversion tool)

This tool converts programs or contact and coil comment data developed for older models into instructions and devices for arbitrary models, and outputs them with a ladder list format. By converting the data converted with this tool using GX Converter, the sequence programs or contact and coil comment data developed for older models can be used by the GX Developer for this CNC.

(2) CHKM7 (program check tool)

This tool checks whether the sequence program created with GX Developer is described in an instruction format complying with the MITSUBISHI CNC specifications. Similar checks are carried out when writing from the GX Developer or PLC onboard to the CNC controller.
This tool can be used to check the data before writing to the CNC controller. This is used to check the sequence programs before writing them to the CNC controller with the IC Card.

(3) GX Converter (data conversion software package)

The GX Converter is a tool that carries out file conversion of GX Developer data files and the following: GX Converter is an add-on tool of GX Developer and is started from the GX Developer menu.

- Ladder list files and comment text files output by the CLST6L
- Alarms and operator messages created by the text editor
- Data files of commercially available spreadsheet software, word processors and editors

This tool is a software package for various MELSEC support. GX Converter needs to be used with the versions following GX Developer Version 3 (SW3D5C-GPPW). Refer to the enclosed Operating Manual for function details.

(4) GX Developer (PLC development software package)

GX Developer is a programming software package (model name: SW8D5C-GPPW) designed for Mitsubishi Electric's MELSEC series programmable logic controllers. By performing operations similar to those of the MELSEC series, you can develop user PLC ladders for the MITSUBISHI CNC series. Note that some functions specific to the "MELSEC series" may not be unavailable.
For MITSUBISHI CNC series sequence program development, we recommend you to use GX Developer Version 4 (SW4D5C-GPPW) or later. For function details, refer to the Operating Manual supplied.

1.2 Operating Environment

The tools that make up the development environment can be used with the personal computer that satisfies the following operating environment of GX Developer.
The following are the specifications of the "GX Developer Version 8" operating environment. For the specifications of different versions, check them in the respective Operating Manuals (startup).

Item		Description
Computer unit		PC-9800 Series *1, or personal computer running Windows
Performance	CPU	Refer to the following "Performance required for basic software and personal computer"
	Required memory	
Hard disk area		150MB or more
Disk drive		CD-ROM disk drive 3.5 inch floppy disk drive *2
Display		800 × 600 pixel or more resolution
Communication interface		RS-232C port
Basic software *3		Microsoft Windows 95 operating system Microsoft Windows 98 operating system Microsoft Windows Millennium Edition operating system Microsoft WindowsNT Workstation 4.0 operating system Microsoft Windows 2000 Professional operating system Microsoft Windows XP Professional operating system Microsoft Windows XP Home Edition operating system

*1: The PC-9800 Series (excluding PC98-NX) is not compatible with Windows Me, Windows 2000 Professional, Windows XP Professional or Windows XP Home Edition.
*2: Required for GX Developer Version 6 or earlier (to provide protection by FLD).
*3: When the basic software of the language except for English, contact MITSUBISHI.

Performance required for basic software and personal computer

Basic software		Performance required for personal computer	
		CPU	Required memory
Windows 95 (Service Pack 1 or higher)		Pentium 133MHz or faster	32MB or more
Windows 98		Pentium 133MHz or faster	32MB or more
Windows Me		Pentium 150MHz or faster	32MB or more
WindowsNT Workstation 4.0 (Service Pack 3 or higher)		Pentium 133MHz or faster	32MB or more
Windows 2000 Professional		Pentium 133MHz or faster	64MB or more
Windows XP Professional	"XP compatibility mode" and "Fast User Switching" are not supported.	Pentium 300MHz or faster	128MB or more
Windows XP Home Edition		Pentium 300MHz or faster	128MB or more

2. GX Developer Functions Supported by MITSUBISHI CNC

The GX Developer functions explained here are those supported by this CNC in the "offline functions" operated with the GX Developer independently, and the "online functions" carried out in connection with the CNC controller.
Refer to the enclosed Operating Manual for function details.

2.1 Function Support Conditions (General Section)

The GX Developer outline functions supported by this CNC are listed below.
The ◎ mark indicates functions that can be used with this CNC. An × mark indicates that the function cannot be used because it is related to "MELSEC Series" characteristic functions. The function details during on-line are described in the next section.

List of general section functions (1) ◎ : Possible, △ : Limitedly possible, × : Not possible

Program type	Support	Remarks
Ladder	◎	
List	◎	
SFC	×	
MELSAP-L	×	
Function block	×	

Function	Menu	Sub menu	Support	Remarks
Project	New project		◎	
	Open project		◎	
	Close project		◎	
	Save		◎	
	Save as		◎	
	Delete project		◎	
	Verify		◎	
	Copy		◎	
	Edit Data	New	◎	
		Copy	◎	
		Delete	◎	
		Rename	◎	
	Change PLC type		△	Fixed Q4A
	Import file	Import from GPPQ format file	◎	
		Import from GPPA format file	×	
		Import from FXGP(WIN) format file	×	
		Import from FXGP(DOS) format file	×	
		Import from TEXT ,CSV format file	◎	
	Export file	Export to GPPQ format files	◎	
		Export to GPPA format files	×	
		Export to FXGP(WIN) format file	×	
		Export to FXGP(DOS) format file	×	
		Export to TEXT ,CSV format file	◎	

List of general section functions (2) ◎ : Possible, △ : Limitedly possible, ✕ : Not possible

Function	Menu	Sub menu	Support	Remarks
(Project)	Macro	Registration macros	◎	
		Macro utilize	◎	
		Delete macros	◎	
		Macro reference path	◎	
	Printer setup		◎	
	Print		◎	
	Start new GX Developer session		◎	
	Exit GX Developer		◎	
Edit	Undo		◎	
	Restore after ladder conversion		◎	
	Cut		◎	
	Copy		◎	
	Paste		◎	
	Insert line		◎	
	Delete line		◎	
	Insert row		◎	
	Delete row		◎	
	Insert NOP batch		◎	
	Delete NOP batch		◎	
	Draw line		◎	
	Delete line		◎	
	Change TC setting		◎	
	Read mode		◎	
	Write mode		◎	
	Ladder symbol	Open contact	◎	
		Close contact	◎	
		Open branch	◎	
		Close branch	◎	
		Coil	◎	
		Application instruction	◎	
		Vertical line	◎	
		Horizontal line	◎	
		Delete vertical line	◎	
		Delete horizontal line	◎	
		Rising pulse	△	When expanding PLC instruction mode
		Falling pulse	△	
		Rising pulse open branch	△	
		Falling pulse close branch	△	
		Invert operation results	△	
		Convert operation results to rising pulse	△	
		Convert operation results to falling pulse	△	
	Documentation	Comment	◎	
		Statement	◎	
		Note	◎	
		Statement/Note block edit	◎	

List of general section functions (3) ◎ : Possible, △ : Limitedly possible, × : Not possible

Function	Menu	Sub menu	Support	Remarks
Find/Replace	Find device		◎	
	Find instruction		◎	
	Find step no.		◎	
	Find character string		◎	
	Find contact or coil		◎	
	Replace device		◎	
	Replace instruction		◎	
	Change open/close contact		◎	
	Replace character string		◎	
	Change module start address		◎	
	Replace statement/note type		◎	
	Cross reference list		◎	
	List of used devices		◎	
Convert	Convert		◎	
	Convert (All programs being edited)		◎	
	Convert (Online change)		×	
View	Comment		◎	
	Statement		◎	
	Note		◎	
	Alias		◎	
	Macro instruction format display		◎	
	Comment format	4*8 characters	◎	
		3*5 characters	◎	
	Alias format display	Replace device name and display	◎	
		Arrange with device and display	◎	
	Toolbar		◎	
	Status bar		◎	
	Zoom	50%	◎	
		75%	◎	
		100%	◎	
		150%	◎	
		Specify	◎	
		Auto	◎	
	Project data list		◎	
	Instruction list		◎	
	Set the contact	9 contacts	◎	
		11 contacts	◎	
	Elapsed time		×	
Online	Refer to "List of on-line section functions"	Refer to "2.2 Function Support Conditions (Online Section)"		
Diagnostics	PLC diagnostics		×	
	MELSECNET(II)/10/H diagnostics		×	
	Ethernet diagnostics		×	
	CC-Link/CC-Link/LT		×	
	System monitor		×	
	Online module change		×	

List of general section functions (4) ◎ : Possible, △ : Limitedly possible, ✕ : Not possible

Function	Menu	Sub menu	Support	Remarks
Tools	Check program		◎	
	Merge data		◎	
	Check parameter		✕	
	Transfer ROM	Read	✕	
		Write	✕	
		Verify	✕	
		Write to file	✕	
	Delete unused comments		◎	
	Clear all parameters		✕	
	IC memory card	Read IC memory card	✕	
		Write IC memory card	✕	
		Read image data	✕	
		Write image data	✕	
	Start ladder logic test		✕	
	Set TEL data	Connection	✕	
		Disconnection	✕	
		TEL data	✕	
		AT command	✕	
		Call book	✕	
	Intelligent function utility	Utility list	✕	
	Customize keys		◎	
	Change display color		◎	
	Options		△	Limited partly
	Create start-up setting file		◎	
Window	Cascade		◎	
	Tile vertically		◎	
	Tile horizontally		◎	
	Arrange icons		◎	
	Close all windows		◎	
Help	CPU error		✕	
	Special relay/register		✕	
	Key operation list		◎	
	Product information		◎	
	Connect to MELFANSweb		◎	

2.2 Function Support Conditions (Online Section)

The GX Developer functions supported by this CNC are listed below.
The ◎ mark indicates functions that can be used currently with this CNC. An × mark indicates that the function cannot be used because it is related to "MELSEC Series" characteristic functions.

List of online section functions (1) ◎ : Possible, △ : Limitedly possible, × : Not possible

Menu	Sub menu	Detailed function	Support	Remarks
Transfer setup		PC side I/F	◎	
		PLC side I/F	△	Only for QnACPU
		Other station	×	
		Network route	×	
		Co-existence network route	×	
Read from PLC		Target memory	◎	
		Title	◎	
		File selection	◎	
		Device data	×	
		Program	×	
		Common	×	
		Local	×	
		Refresh view	◎	
		Free space volume	◎	
		Create title	×	
Write to PLC		Target memory	△	
		Title	◎	
		File selection	◎	
		Device data	×	
		Program	×	
		Common	×	
		Local	×	
		Free space volume	◎	
		Create title	×	
Verify with PLC		Target memory	◎	
		Title	◎	
		File selection	◎	
		Program	×	
		Refresh view	◎	
		Free space volume	◎	
		Create title	×	
Write to PLC (Flash ROM)	Write the program memory to ROM		×	
	Write to PLC (Flash ROM)		×	
Delete PLC data		Target memory	◎	
		Title	◎	
		File selection	◎	
		Refresh view	◎	
		Free space volume	◎	
		Create title	×	
Change PLC data attributes			×	
PLC user data	Read PLC user data		×	
	Write PLC user data		×	
	Delete PLC user data		×	

List of online section functions (2) ◎ : Possible, △ : Limitedly possible, × : Not possible

Menu	Sub menu	Detailed function	Support	Remarks
Monitor	Monitor mode/Start/Stop	ON/OFF state	◎	
		Scan time display	◎	
		CPU state display	◎	
	Monitor [Write mode]		×	
	Start monitor [All windows]		◎	
	Stop monitor [All windows]		◎	
	Change current value monitor [Decimal]		◎	
	Change current value monitor [Hexadecimal]		◎	
	Local device monitor		×	
	Device batch	Device	◎	
		Connect	◎	
		Coil	◎	
		Setting value	◎	
		Current value	◎	
		Monitor format : Bit & word	◎	
		Monitor format : Bit	◎	
		Monitor format : word	◎	
		Display : 16bit integer	◎	
		Display : 32bit integer	◎	
		Display : Real number	×	
		Display : ASCII character	×	
		Value : DEC	◎	
		Value : HEX	◎	
		T/C set value Reference program	◎	
		Device test	◎	
	Entry data monitor	Device	◎	
		ON/OFF/Current	◎	
		Setting value	◎	
		Connect	◎	
		Coil	◎	
		Display : 16bit integer	◎	
		Display : 32bit integer	◎	
		Display : Real number	×	
		Display : ASCII character	×	
		Value : DEC	◎	
		Value : HEX	◎	
		T/C setting value, Local label Reference program	◎	
		Device test	◎	
	Buffer memory batch		×	
	Monitor condition setup	Device	◎	
		Step No.	◎	
	Monitor stop condition setup	Device	◎	
		Step No.	◎	
	Program monitor list		×	
	Interrupt program monitor list		×	
	Scan time measurement		×	
	Entry ladder monitor		◎	
	Delete all entry ladder		◎	

List of online section functions (3) ◎ : Possible, △ : Limitedly possible, × : Not possible

Menu	Sub menu	Detailed function	Support	Remarks
Debug	Device test	FORCE ON	◎	
		FORCE OFF	◎	
		Toggle force	◎	
		Device	◎	
		Forced input output registration/ cancellation	×	
		Buffer memory	×	
	Debug		×	
	Skip execution		×	
	Partial execution		×	
	Step execution		×	
Trace	Sampling trace	Wizard setting/execution	◎	
		Individual setting/execution	◎	
		Trace data storage destination	×	Cannot select
		No. of traces	△	Setting range: 1 to 8192
		Trigger position	◎	
		Trace additional information	×	
		Trace point setup	△	Interval, step No. not possible
		Trigger point setup	△	STRA instruction, step No. not possible
		Device point setup	△	Limit to device type/No. of points
		Trace operation	◎	
		Trace status	◎	
		Trace result	◎	
		CSV file creation	◎	
		Trace settings file operation	◎	
		Trace settings PLC operation	◎	
		Data retention at power OFF	×	
Remote operation		PLC status	◎	
		RUN	◎	
		STOP	◎	
		PAUSE	×	
		Latch clear	×	
		STEP-RUN	×	
		Reset	×	
		Operation during RUN	×	
		Specify execution destination	×	

Menu	Sub menu	Detailed function	Support	Remarks
Keyword setup	Register		◎	
	Delete		◎	
	Disable		◎	
Clear PLC memory			×	
Format PLC memory		Target memory	△	
		Format Type	×	
Arrange PLC memory			×	
Set time		YY MM DD Hr. Min. Sec.	◎	
		Day of week	×	
		Specify execution destination	×	

3. Preparation

3.1 Installing the Tools

In this CNC's PLC development environment, it is assumed that the various tools are used with an IBM PC/AT compatible personal computer. Prepare each tool so that it is IBM PC/AT compatible personal computer.
Refer to the enclosed Operating Manual (Startup) and Operating Manual for the setup and start procedures of each tool.

3.2 Preparation for Serial (RS-232C) Communication

3.2.1 Connecting the Serial Cable

The serial port connected with the CNC controller differs depending on the model. Also, it may require a special branch cable. Refer to the connection manual of that model.

Between the IBM PC/AT compatible personal computer that uses GX Developer and the CNC controller, use an RS-232C serial cable equivalent to the one shown below in the RS-232C connection diagram.

[Note] The cables given in the connection diagrams of the GX Developer Operating Manual cannot be used.

NC side (25-pin D-SUB)			**Personal computer side (9-pin D-SUB)**	
Signal name	**Pin No.**	**Cable connection and signal direction**	**Pin No.**	**Signal name**
CD	8	- - - - - - - - - - -	1	DC
SD	2	→	2	RD
RD	3	←	3	SD
DR (DSR)	6	←	4	ER (DTR)
SG	7	↔	5	SG
ER (DTR)	20	→	6	DR (DSR)
CS (CTS)	5	←	7	RS (RTS)
RS (RTS)	4	→	8	CS (CTS)
	22	- - - - - - - - - - -	9	RI

* The above shows a general RS-CS method connection format.
* The pin Nos. of dotted lines are not used.

3.2.2 Setting the Connection Target

The connection target must be specified before performing online operations from GX Developer to the CNC controller.
Perform the following operation with GX Developer to start the setting screen.

[Online]→[Connection Setup]

Set the following items only. Leave the other items unchanged from the initial values.

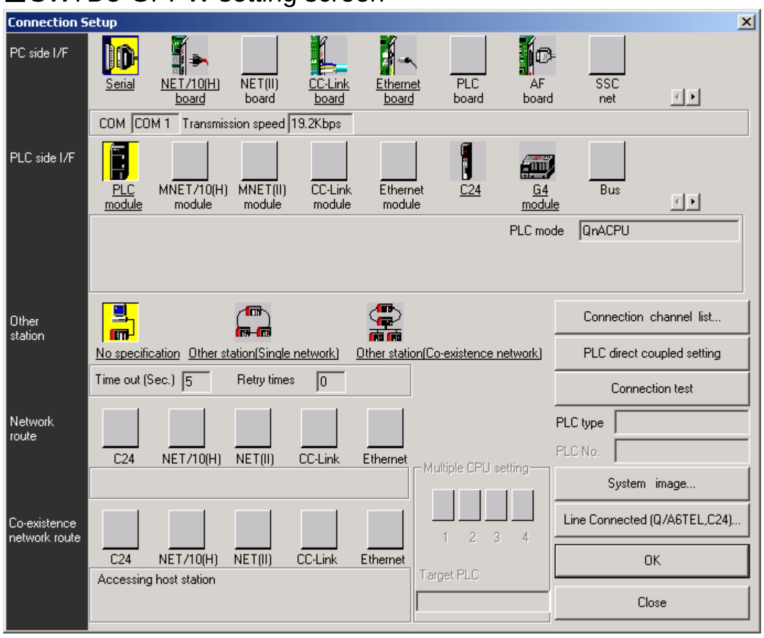
(1) Personal computer side

Interface : [Serial]
Serial port name: [COM1] or [COM2]
Baudrate : [19.2Kbps]

(2) PLC side

Interface : [CPU unit]

■SW7D5-GPPW setting screen



3.3 Preparation for Ethernet Communication

Procedures for preparing for Ethernet communication is as follows. Refer to each respective instruction manual for details.

(1) Confirm IP address of the CNC unit.
(2) Set IP address for the personal computer side.
(3) Connect the Ethernet cable.
(4) Set the connection target of GX Developer.

3.3.1 Confirming IP Address of the CNC Unit

Confirm the IP address set in the CNC unit. IP address is set in the parameters below.

Base common parameter	Item	Details	Setting example
#1926	Global IP address	IP address of the CNC unit looking from outside	192.168.200.1
#1927	Global Subnet mask	Subnet mask of #1926	255.255.255.0

Example given here is the case where "192.168.200.1" is set as the initial value after SRAM clear.

3.3.2 Setting IP Address for the Personal Computer Side

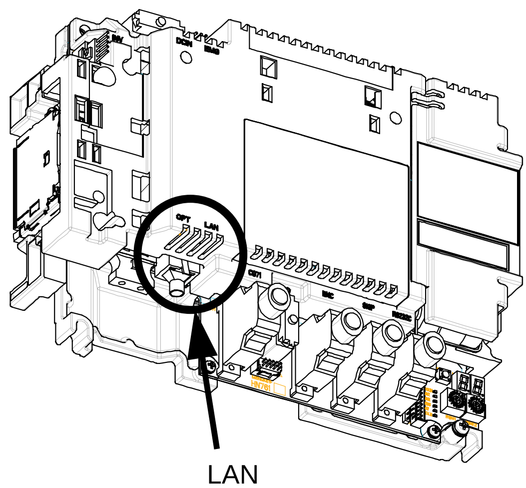
For the personal computer for which GX Developer is available, IP address has to be set within the same subnet mask as NC unit.
In the case of section 3.3.1, set the address other than "192.168.200.1", using from "192.168.200.0" to "192.168.200.255".
Note that when other CNC and devices are connected on the same network, be careful not to duplicate the address with other CNC and devices.

3.3.3 Connecting the Ethernet Cable

Connect the Ethernet cable to the connector LAN1 (or LAN) of the control unit below. If the connector is already connected, connect the cable to the HUB of the connection target, etc.

700 Series
Control unit
back side

70 Series
Control unit
back side



3.3.4 Setting the Connection Target

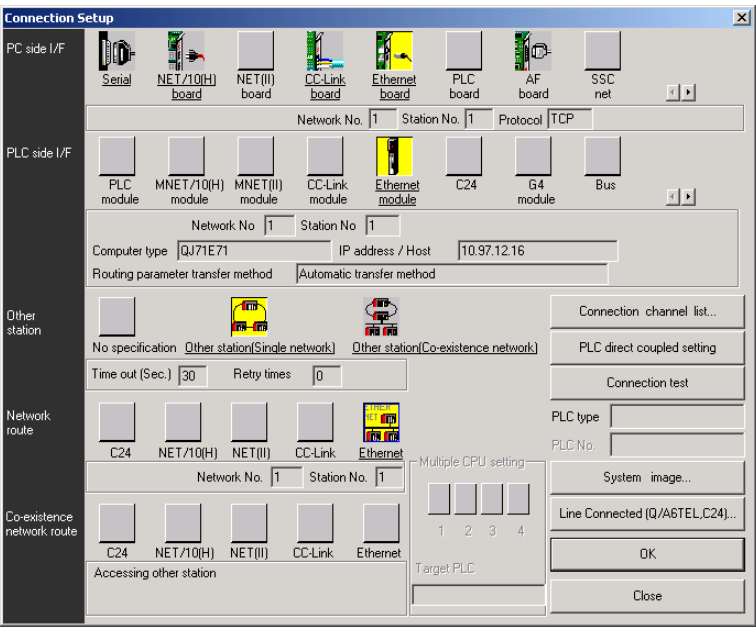
The connection target must be specified before performing online operations from GX Developer to the CNC controller.
The following two connection methods are supported when connecting with Ethernet. Use these accordingly.

- TCP protocol: Connection type used with typical networks. The communication amount is high so the speed is not as fast, but the reliability is high. (A resend request is made when there is a lapse in communication.)
- UDP protocol: Connection-less type suitable for closed networks such as small-scale LAN. The reliability is not as high, but the speed is fast. (A resend request is not made when there is a lapse in communication.)

Perform the following operation with GX Developer to start the setting screen.
[Online]→[Transfer Setup]
Set the following items in order. Leave the other items unchanged from the initial values.
Normally use connection setting example 1 (using TCP protocol) for setting.

Setting item \ Connection protocol		Connection setting example 1	Connection setting example 2	Connection setting example 3
		TCP	**UDP**	**UDP**
PC side I/F	Interface	Ethernet		
	Protocol	TCP	UDP	None
PLC side I/F	Interface	Ethernet unit		
	Unit type name	QJ71E71		AJ71QE71
	Station No.	1		
	IP address	Set CNC controller's IP address		
	Routing parameter conversion method	Automatic conversion method		
Other station	Interface	Other station (same network)		

■SW7D5-GPPW setting screen



4. Common Items

4.1 Precautions Before Development

Pay careful attention to the following items before developing sequence programs using the GX Developer.

⚠ CAUTION **Always observe the following precautions during work.**

(1) PLC Type Selection

The PLC type must be set when newly creating programs, etc. Select the following CPU type when requested to select the PLC type by the GX Developer. An error will occur during transfer of the sequence program to the CNC controller if another PLC type is selected.

⚠ CAUTION **Select "Q4A" for CPU type.**

(2) Device Setting

The number of device points must always be set when developing the sequence program for the CNC controller. The sequence program will not be transferred to the CNC controller correctly if it is developed with the setting values given later.

⚠ CAUTION **Always set the number of devices.**

(3) PLC Instructions

MELSEC-specific PLC instructions cannot be used in the sequence program development for the CNC controller. Only PLC instructions and formats in "II PROGRAMMING EXPLANATION" can be used.
The format, etc., are changed with some instructions. Refer to "Appendix 1.3 Instructions with Changed Designation Format".

⚠ CAUTION **MELSEC-specific PLC instructions cannot be used.**

(4) Saving the sequence program

The sequence program transferred from the GX Developer or PLC onboard to the CNC controller is stored in the temporary memory. The temporary memory is erased when the power is turned OFF. (This is because the sequence program stored in the internal flash ROM is validated when the power is turned ON again.)
Thus, if the sequence program is to be held even when the power is turned ON again, always write it to the internal flash ROM.

⚠ CAUTION **Sequence programs that are not saved in the internal flash ROM are not held when the power is turned OFF.**

4.2 NC-related Parameters

Each model has bit selection parameters related to GX Developer. If an appropriate value is not set in the parameters, an error will occur in communication with GX Developer.

(1) GX Developer serial communication enabled

			7	6	5	4	3	2	1	0
#	(6451)	Data	(0	0	1	0	0	0	0	0)

Use No. 6451.

0 : GPP communication disabled
1 : GPP communication enabled

[Reference] #6451 corresponds to the low side of the file register R7825.

Bit 5 = 0

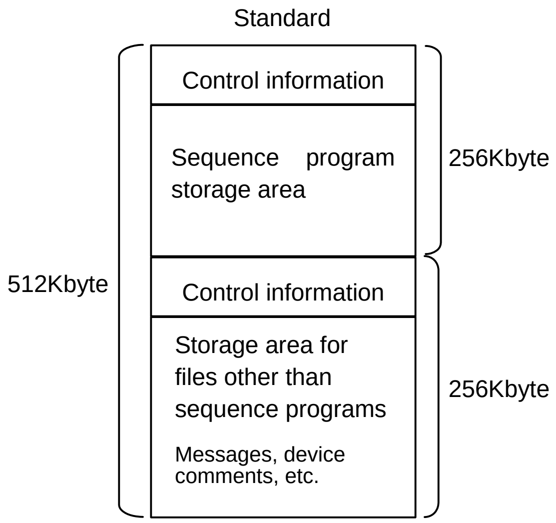
The serial port is not used for communication with GX Developer.
(When the serial port is used for another function)

Bit 5 = 1

The serial port is used for communication with GX Developer.

4.3 PLC Data Storage Areas

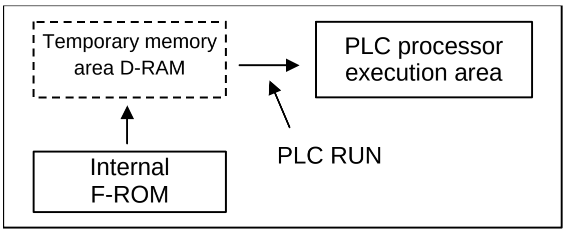
The M700 Series CNC stores the PLC data in the internal flash ROM (hereinafter, internal F-ROM).
The following shows the storage area structure.



The PLC data transferred from GX Developer or PLC onboard is executed with the following path.

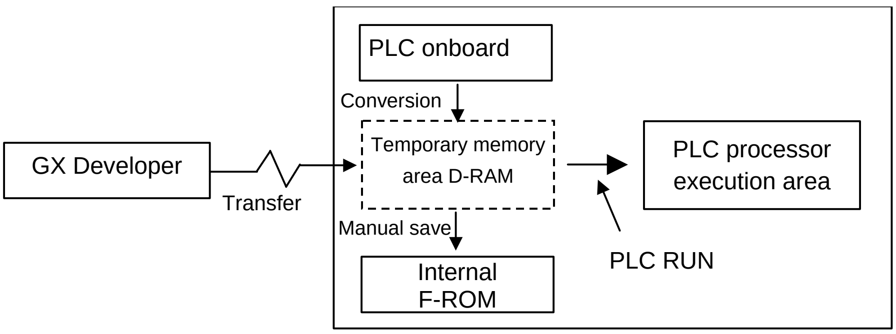
(1) At power ON

The data is transferred from the internal F-ROM to the PLC processor execution area via the temporary memory D-RAM, and is then executed.



(2) During PLC development

The PLC data transferred from the GX Developer or PLC onboard is stored in the volatile RAM (hereinafter, D-RAM) for the temporary memory. The sequence program is transferred to the PLC processor execution area before PLC execution, and is then executed.
The D-RAM in the temporary memory is not held when the power is turned OFF. If the data needs to be held even after the power is turned OFF, it must be stored in the internal F-ROM.



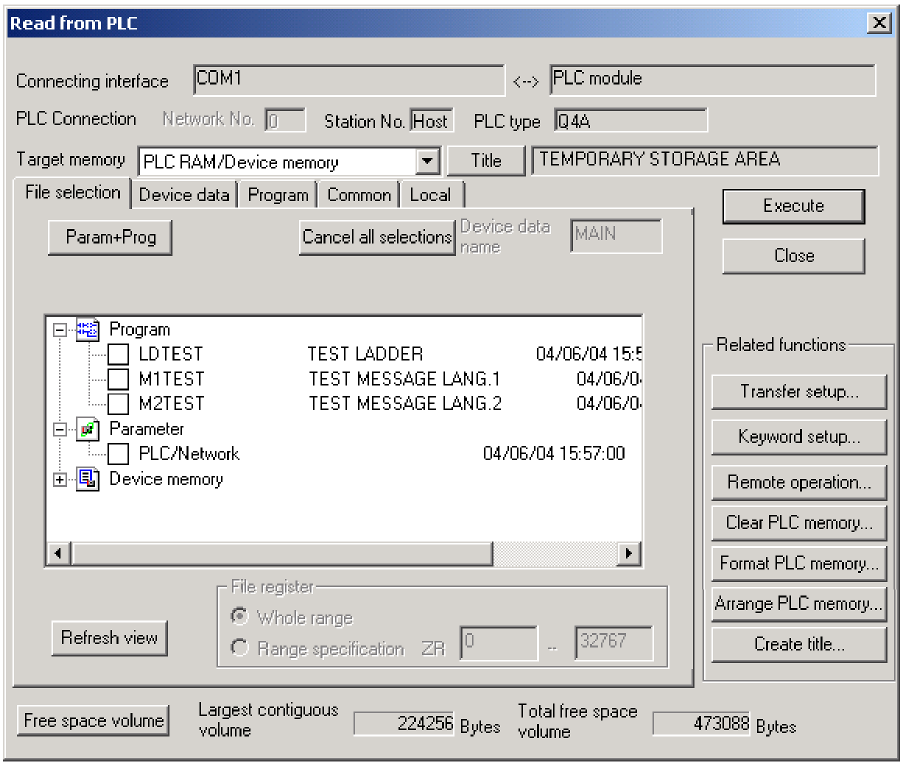
(3) Areas that can be selected on GX Developer

The following table indicates the storage areas that can be selected for the online functions of GX Developer.

Function	Selectable storage area	
	[Target memory]	Storage area (M700 Series)
Write to PLC, Delete PLC data, Format PLC memory	Internal RAM/device memory	Temporary memory D-RAM
Read from PLC, Verify PLC	Internal RAM/device memory IC memory card A (ROM)	Temporary memory D-RAM Internal F-ROM
Read from PLC (Checking the execution area size)	IC memory card A (RAM)	Execution area

(4) Display of storage area on GX Developer

Any of the storage areas that can be selected for the online functions of GX Developer can be specified as a [Target memory] item on the corresponding operation screen.
Also, pressing the [Title] button displays the comment of that storage area if PLC data exists. It is not displayed if the storage memory is not fitted or the data does not exist.



The free space in the selected area will appear when the "Free space volume" button is pressed.
"Largest continuous volume" and "Total free space volume" have the following meanings.

"Largest continuous volume": Free space in sequence program storage area

"Total free space volume" : Free space in sequence program storage area + other file storage areas

"Total free space volume" - "Largest continuous volume":
Free space in file storage areas other than sequence program storage area

The following table indicates the relationships between the [Target memory] items and storage areas.

Target memory	Meaning in M700 series	Title indication	Free area indication
Internal RAM/device memory	Temporary memory D-RAM	TEMPORARY STORAGE AREA	Largest continuous volume: Free space in sequence program storage area Total free space volume: Total free space
IC memory card A (RAM)	Execution area size confirmation	LAD.EXEC.SIZE AREA	Not used (insignificant value)
IC memory card A (RAM)	Internal F-ROM	FLASH ROM AREA	Largest continuous volume: Free space in sequence program storage area Total free space volume: Total free space
IC memory card B (RAM)	Not used (cannot be selected)	None	
IC memory card B (RAM)	Not used (cannot be selected)	None	

4.4 File Name

The PLC related data, such as the sequence programs and message data, are controlled and stored with the following categories in this CNC.
Therefore, they are also developed in the same categories.

⚠ CAUTION

⚠ If the data transferred does not follow the file name rule, the NC will mistake it for another data, resulting in unexpected operation, e.g. sequence program erasure.

4.4.1 File Name Rule for Sequence Program, Parameter, and Device Comment

File name rule (Sequence program, parameter, device comment)

y x x x x x x x . W x x

- Wxx: Extensions (expresses file classification): Automatically attached, depending on GX Developer or PLC onboard
- xxxxxxx: Arbitrary character string
- y: Reserved character string or arbitrary character

When storing data with GX Developer or PLC onboard, the data type is identified with the file name. Up to 8 one-byte alphanumeric characters (excluding the extension), hyphen (-) and underline (_) can be used to the file name. The extension is automatically attached and expresses classification of the file.
Note that the first character may be reserved for expressing the data type.

[Caution] Reserved file names
The file name is reserved on the NC side if the following head character is used.
- "H"

Do not use file name in combination with this head character.

List of sequence program, parameter, and device comment

	Data class	Data type	File name	Storage quantity	Remarks
1	Sequence program	High-speed process	H+[Arbitrary character string].WPG	Total 32 programs	Execution type (scan) *1
		Main process	[Arbitrary character string].WPG		Execution type (scan) *1
		Initialization process	[Arbitrary character string].WPG		Execution type (initialization) *1
		Standby process	[Arbitrary character string].WPG		Execution type (standby/low-speed) *1
2	Parameter	PLC parameters	PARAM.WPA (fixed)	1 parameter	
		Network parameters			
3	Device comment	Common comment	COMMENT.WCD (fixed)	Total 10 comments	Common for all sequence programs
		Comment for each program	[Arbitrary character string].WCD		For sequence programs having same name

"Arbitrary character string" means a character string consisted of up to 8 one-byte alphanumeric characters, hyphen (-) and underline (_).

*1: The execution type is the sequence program operation type designated with the parameters.

4.4.2 File Name Rule for Message Data

There are two methods for the message data language selection. Method 2 (Method linked with language selection on the setting and display screen) is recommended although both methods are available.

(Method 1) Specify with 3 bits of bit selection parameter #6453 bit 0 to 2. (Language selection method using PLC alone)

(Method 2) Specify with display language selection parameter. (Base specifications parameter #1043) (Method linked with language selection on the setting and display screen)

File name rule (Message data)

(Method 1) MNXxxxxx.WPG

- Arbitrary character string 6 characters (1st character has to be other than number)
- Number (1 digit)
- Reserved character

(Method 2) MNNxxxxx.WPG

- Arbitrary character string 5 characters
- Number (2 digits)
- Reserved character

Alphanumeric characters, hyphen (-) and underline (_) can be used for the arbitrary character string. The extension is automatically attached and expresses classification of the file.
Note that the first character may be reserved for expressing the data type.

Specify which method is valid with the name of the message file to be stored.
If the file with condition 1 and 2 exist together, the method 1 will be valid.

(Condition 1) Method 1 is valid when the first two characters of the file name is "M + 1-digit number".
(Condition 2) Method 2 is valid when the first three characters of the file name is "M + 2-digit number"

(1) Method 1

Language is specified with bit 3 of the bit selection parameter #6453 bit 0 to 2, and the No. corresponds to the No. used in message file name.

List of message file name (Method 1)

Data class	Bit selection parameter #6453 bit0-2	Data type	File name	Storage quantity	Remarks
Message (Method 1)	0	1st language	M1Xxxxxx.WPG	One for each language	It is not possible to store multiple files having the same language Nos. even if their names are different. A message confirming overwriting the same language No. is displayed.
	1	2nd language	M2Xxxxxx.WPG		
	2	3rd language	M3Xxxxxx.WPG		
	3	4th language	M4Xxxxxx.WPG		
	4	5th language	M5Xxxxxx.WPG		
	5	6th language	M6Xxxxxx.WPG		
	6	7th language	M7Xxxxxx.WPG		
	7	8th language	M8Xxxxxx.WPG		

* "Xxxxxx" part of a file name consists of up to 6 arbitrary alphanumerical characters including hyphen (-) and underline (_). Note that however, the third character "X" has to be other than numerical character. (To avoid confusion with method 2.)

(2) Method 2

Language is specified with the language parameter #1043 on the setting and display screen, and the No. corresponds to the 2-digit No. used in message file name.
When no corresponding message file for a certain language parameter is stored, an English language display file (M00xxxxx.WPG) is referred to as an alternative file. Thus, message data file for the English language display must be stored.

List of message file name (Method 2)

Data class	Language parameter (Base specifications parameter #1043)	File name	Storage quantity	Remarks
Message (Method 2)	0 (English) [mandatory]	M00xxxxx.WPG	One for each language Total 8 languages	When no corresponding file for a certain language parameter is stored, an English language display file (M00xxxxx.WPG), if stored, is referred to as an alternative file. It is not possible to store multiple files having the same language Nos. even if their names ("xxxxx" part) are different. A message confirming overwriting the same language No. is displayed.
	1 (Japanese)	M01xxxxx.WPG		
	11 (German)	M11xxxxx.WPG		
	12 (French)	M12xxxxx.WPG		
	13 (Italian)	M13xxxxx.WPG		
	14 (Spanish)	M14xxxxx.WPG		
	15 (Chinese -traditional-)	M15xxxxx.WPG		
	16 (Korean)	M16xxxxx.WPG		
	17 (Portuguese)	M17xxxxx.WPG		
	18 (Dutch)	M18xxxxx.WPG		
	19 (Swedish)	M19xxxxx.WPG		
	20 (Hungarian)	M20xxxxx.WPG		
	21 (Polish)	M21xxxxx.WPG		
	22 (Chinese -simplified-)	M22xxxxx.WPG		
	Other than above; up to 99	If the message for a corresponding No. exists, it is displayed. If not, an English file (M00xxxxx.WPG) is displayed.		
	Other than above; above 100	An English file (M00xxxxx.WPG) is displayed.		

(3) Precautions

- Even if the file is made with the conventional specifications method 1, when the third character of the file name is a numerical character, it may be identified as method 2 (language selection parameter.
  (Example) "M1720V02.WPG", "M750MESS.WPG", "M65S-MES.WPG", etc.

- Files having the same No. and different arbitrary names are identified as the same files and will be overwritten.
  (Example) "M1TEST.WPG" and "M1JAPAN.WPG", "M00ENG01.WPG" and "M00ENG02.WPG", etc.

- Files having the same arbitrary name ("xxxxx" part) and different method types are not identified as the same files and will not be overwritten. If method 1 and 2 exist together, method 1 will be valid.
  (Example) "M1TEST.WPG" and "M01TEST.WPG", "M1JAPAN.WPG" and "M01JAPAN.WPG", etc.

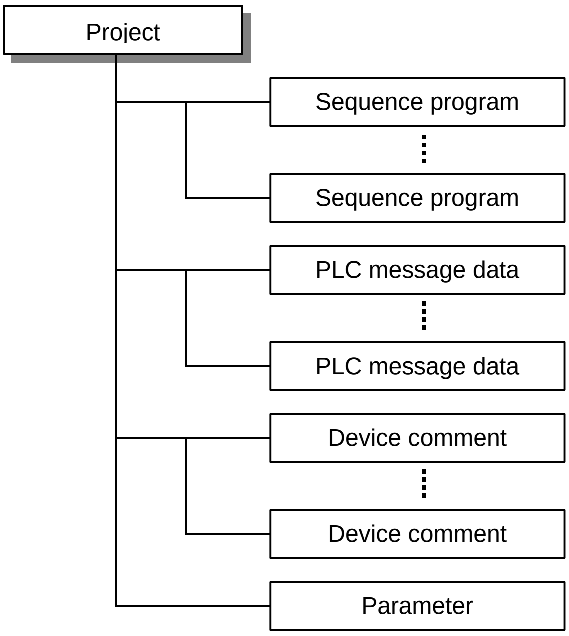
- Files that are not applied to condition 1 and 2 are not identified as a message file. They are identified as a sequence program (ladder), instead.
  (Example) "M0TEST.WPG", "M9MESS.WPG", "M0-1TEST.WPG", "M-01JPN.WPG", "MM00ENG.WPG", etc.

4.5 Creating a Project

GX Developer treats PLC related data such as sequence program and message data collectively as a "project".
In this section, the method of creating a project is explained.

4.5.1 Project

Project is a collection of sequence program, device comment, PLC message data and parameters. Usually, data is controlled by the project level per machine type or version.



Item	Details
Sequence program	Sequence program for MITSUBISHI CNC (User PLC)
PLC message data	This defines PLC messages such as alarm message and PLC switch.
Device comment	This is the comment for a sequence program device. There are "common comment" which is common for a project and "comment by program" which is particular for each program.
Parameter	This sets the device range and execution order of the sequence program.

(1) One project per GX Developer
Only one project can be edited with a GX Developer. Therefore, if two or more projects are to be edited, multiple GX Developers must be operated.

(2) Device comment

Comment type	Number to be created	Details
Common comment	1	Common device comment for all the programs existed in the project.
Comment by program	Same as the number of programs	Device comment set for each program. Always set the same name as that of program.

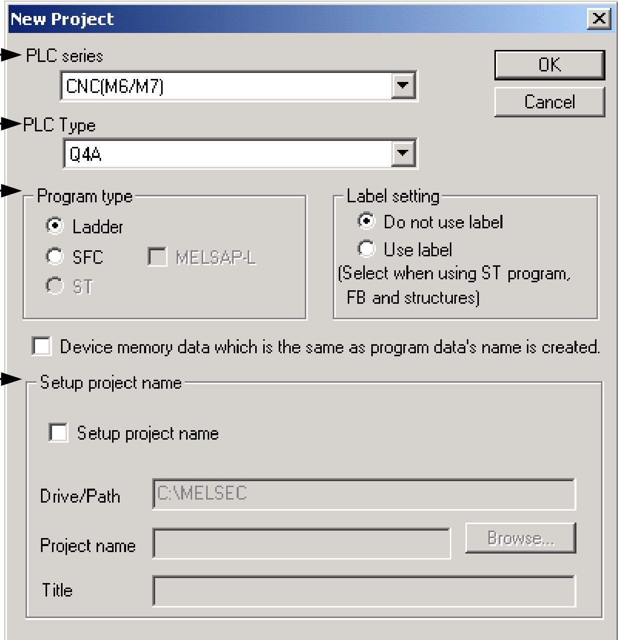
4.5.2 Operating Procedures

Perform the following operation from GX Developer to start the operation screen.

[Project] → [New Project]

Set the PLC series, PLC type and project name required to create a new project with the screen below.

(1)
(2)
(3)
(4)



(1) PLC series

Sets the series name of PLC (Programmable controller). Select "CNC(M6/M7)" or "QnACPU" here. Note that selecting "CNC(M6/M7)" or "QnACPU" does not make a difference.
"CNC(M6/M7)" can be selected for GX Developer ver8.23Z or later version.

(2) PLC type

Set the PLC type. Set "Q4A" here.

(3) Program type / Label setting

Default values are set for these items. Refer to the enclosed Operating Manual for details.

(4) Project name / Title

Set drive/path+project name (8 or more characters can be set) with up to 154 one-byte alphanumerical characters together. Title must be set with up to 32 one-byte characters.
Refer to the enclosed Operating Manual for details on the available characters.
When inputting/outputting built-in PLC editing function (PLC onboard function) and project data by turns, consider the following restrictions and set.

PLC onboard function	Alphanumerical display	Alphanumerical input	Japanese display	Japanese input	Number of characters
Project name	○	○	○	×	18 (abbreviated hereafter)
Title	×	○	×	×	32

4.6 Setting the Parameters

The GX Developer parameters must be set before the CNC controller sequence program can be developed with GX Developer. The required parameter settings are shown below. Refer to the "II PROGRAMMING EXPLANATION" for details on each setting item.

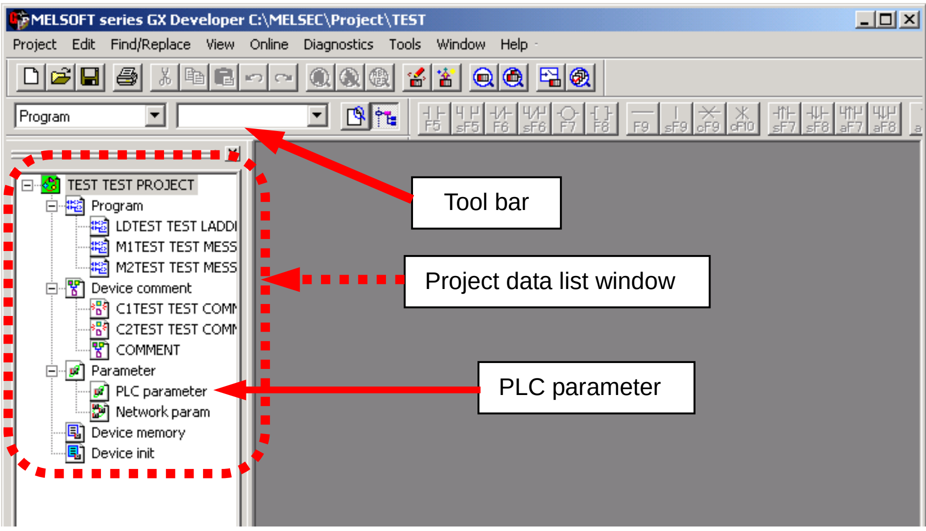
- Setting the number of device points
- Setting the number of common pointer points
- Setting the program execution order

4.6.1 Parameter Setting Screen

(1) Displaying the project data list and Parameter Setting screen

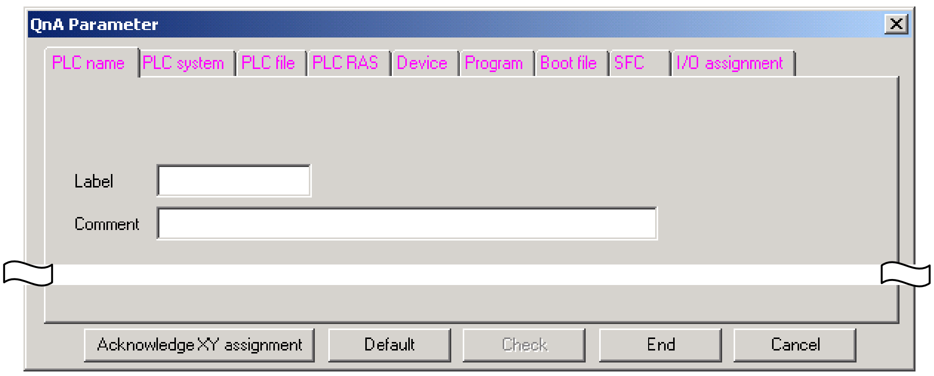
Open the "Project data list" window with the following operations. Double-click on "PLC parameter" to open the parameter setting screen. The same operation can be completed by designating the data type "Parameter" and data name "PLC parameter" from the tool bar.

Double-click on "PLC parameter" under [View] → [Project data list].



(2) Displaying the Parameter Setting screen

If the characters on the tab are "red", the default values are set.



4.6.2 Setting the Number of Device Points

After creating the area (project) for controlling the sequence program, the number of devices used in that project must be set first. This operation must be carried out each time a new project is created.

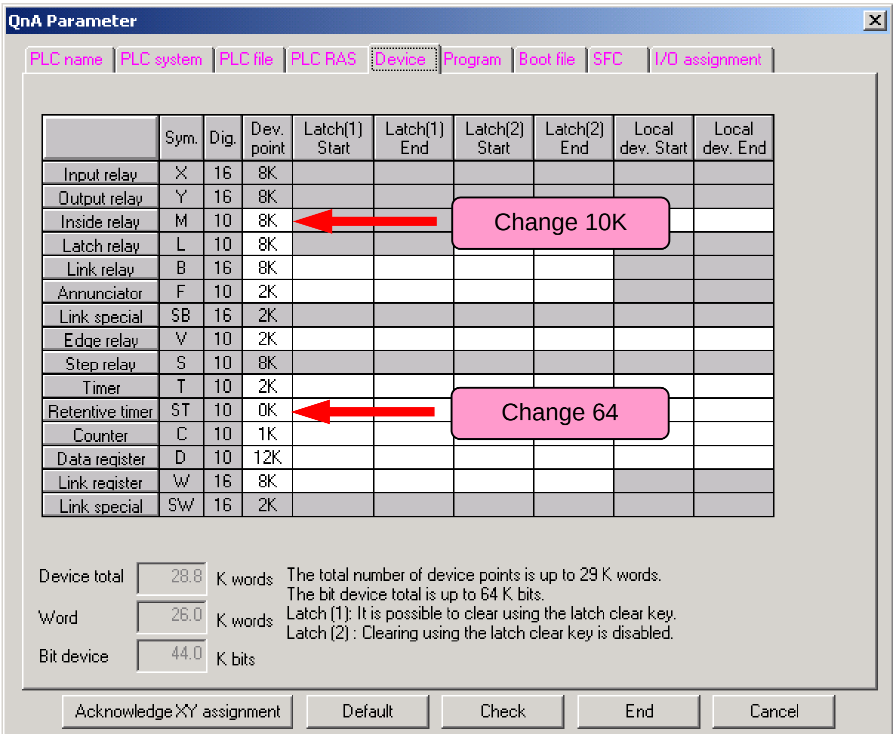
[Caution] An error will occur if the sequence program is downloaded to the CNC controller without completing this operation.

Select the [Device] tab

Set the value for the number of device points on the following screen, and then click on [End].

Internal relay M : Change from [8K] to [10K]

Retentive timer ST : Change from [0K] to [64] (**[NOTE]** K is not added.)

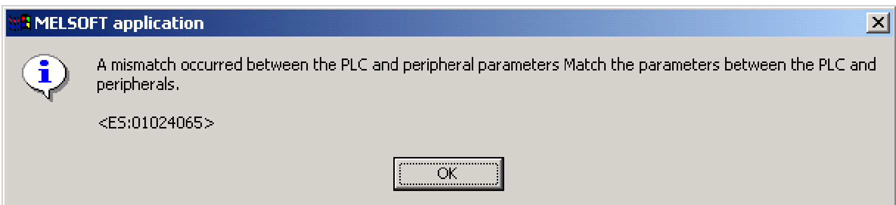


	Sym.	Dig.	Dev. point	Latch(1) Start	Latch(1) End	Latch(2) Start	Latch(2) End	Local dev. Start	Local dev. End
Input relay	X	16	8K						
Output relay	Y	16	8K						
Inside relay	M	10	8K						
Latch relay	L	10	8K						
Link relay	B	16	8K						
Annunciator	F	10	2K						
Link special	SB	16	2K						
Edge relay	V	10	2K						
Step relay	S	10	8K						
Timer	T	10	2K						
Retentive timer	ST	10	0K						
Counter	C	10	1K						
Data register	D	10	12K						
Link register	W	16	8K						
Link special	SW	16	2K						

[Note] If a non-designated value is set, an error will occur when downloading to the CNC controller.

(1) When device setting value is illegal

If there is a problem with the device setting value, the following error dialog will appear when writing the sequence programs to the CNC controller. Set the number of device points as explained above in this case.



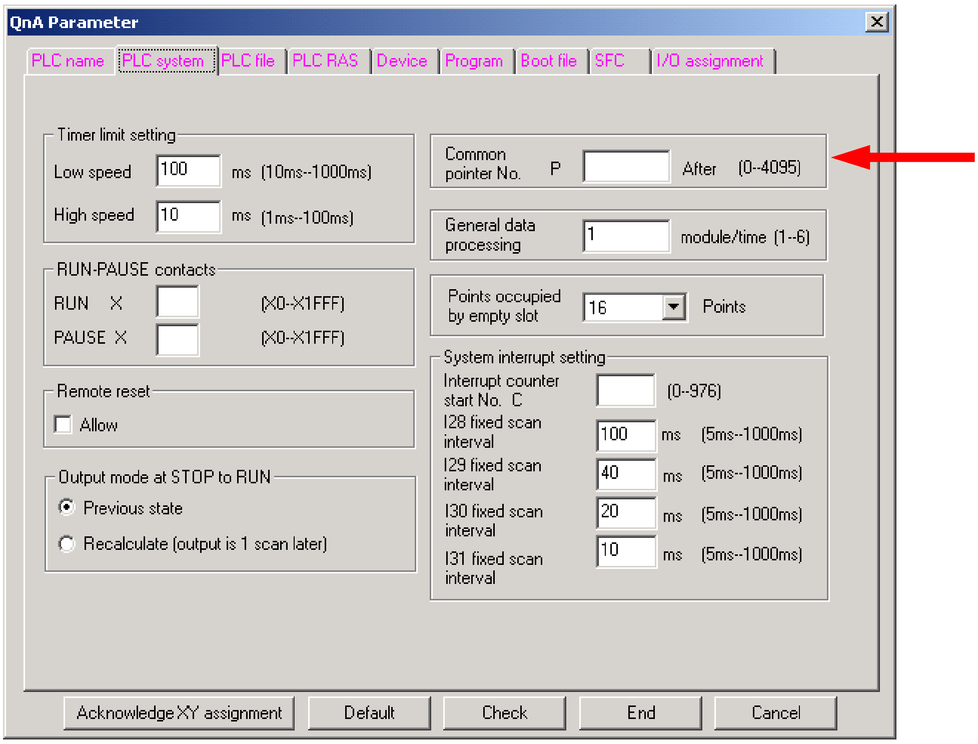
4.6.3 Setting the Number of Common Pointer Points

When creating the sequence programs with the multi-program method, the number of common pointers must be set for each sequence program. This operation must be carried out each time a new project is created and the parameter file is written to the CNC controller.

[Note] The default value will be used for the number of common pointer points if this operation is not completed.

Select the [PLC System] tab

Set the "Common pointer No." value on the following screen, and then click on [End].



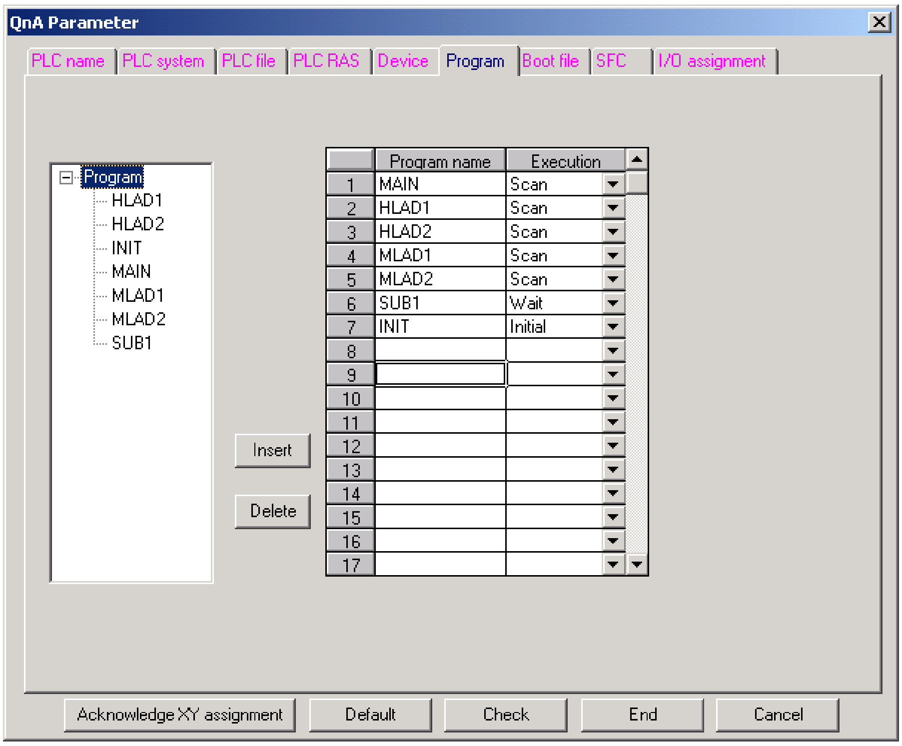
4.6.4 Setting the Program Execution Order

When creating the sequence programs with the multi-program method, the execution method and execution order of each sequence program must be set. This operation is carried out before the sequence program is executed with the CNC controller. The parameter file must be written to the CNC controller.

[Caution] If this operation is not completed when using the multi-program method, an error will occur when RUN is executed in the sequence program.

Select the [Program] tab

Select the sequence program name to be registered for execution from the program list on the left of the following screen, and then press the "Insert" button. Select the execution mode from the registration program list on the right side. After registering all sequence programs to be executed, click on [End].



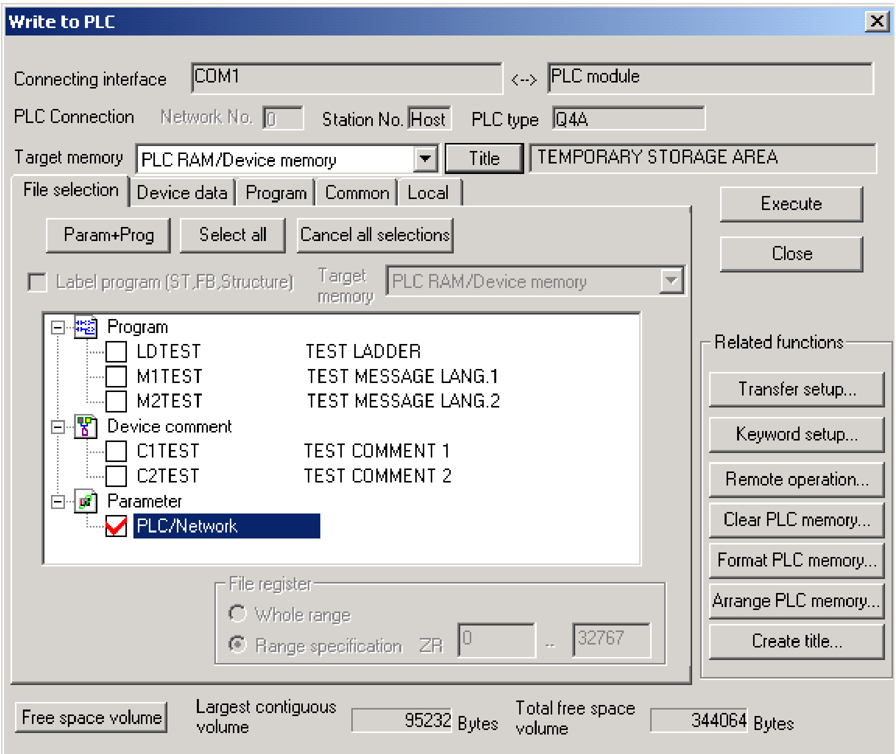
4.6.5 Writing and Reading Parameters to and from the CNC Controller

When creating the sequence programs with the multi-program method, the parameter file must be written to the CNC controller. The parameter file can also be read from the CNC controller and used with GX Developer.
The operation methods are the same as reading and writing the sequence programs.

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Write to PLC]

On the following screen, choose the parameter [PLC/Network] file to be written from the [File selection] tab and click [Execute].



[Note] As [Target memory], only [PLC RAM/Device memory] is valid.
Do not set the other tabs ([Device data], [Program], [Common], [Local]) than [File selection].

4.7 Starting/Stopping the PLC of the CNC Controller

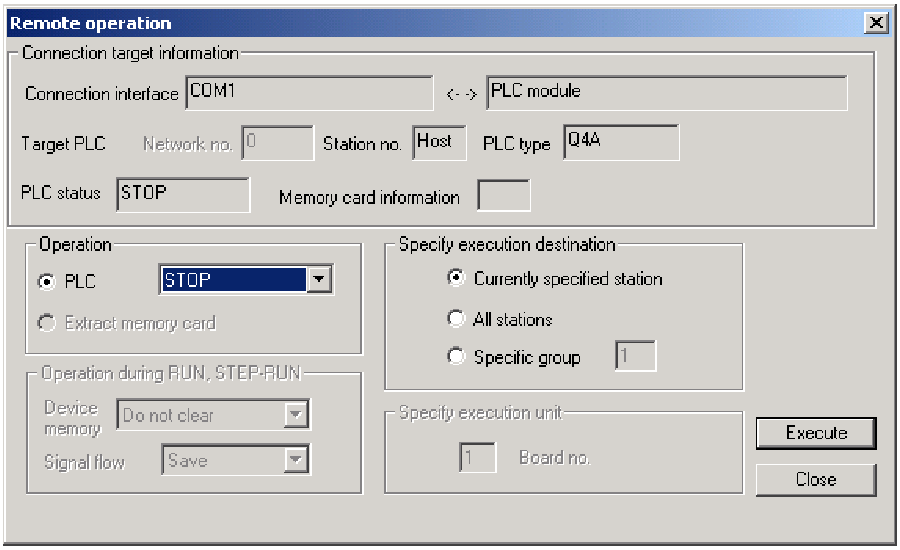
Before writing a sequence program, you must stop the PLC of the CNC controller.

4.7.1 Operation Procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Remote operation] or Alt + 6

On the following screen, set "STOP" or "RUN" in the [PLC] part under [Operation] and click [Execute]. The current status is displayed in [PLC status] under [Connection target information].



[Note] Operations other than RUN and STOP cannot be executed.

The operation is completed when the following dialog appears. Click [OK]. The status after completion appears in [PLC status] on the remote operation screen displayed behind. If the status does not change, check whether an alarm is displayed or not on the CNC controller side.

4.8 Keyword Registration

Keyword is used to protect the sequence programs stored in CNC. Read and overwrite operations by GX Developer or with the onboard PLC edit function are prohibited.

4.8.1 Data Protected by a Keyword

A keyword protects a whole range of data. The data to be protected includes all the PLC-related data stored in the flash ROM in CNC (files of sequence programs, device comments, PLC messages and parameters). Only the files with particular names can be free from the keyword protection.
The keyword is also set to the backup data that is output in the I/O operation: the backup data is kept protected by the keyword after copied to other CNCs.

4.8.2 Operations Prohibited by a Keyword

The following two types of conditions can be selected at the keyword registration.

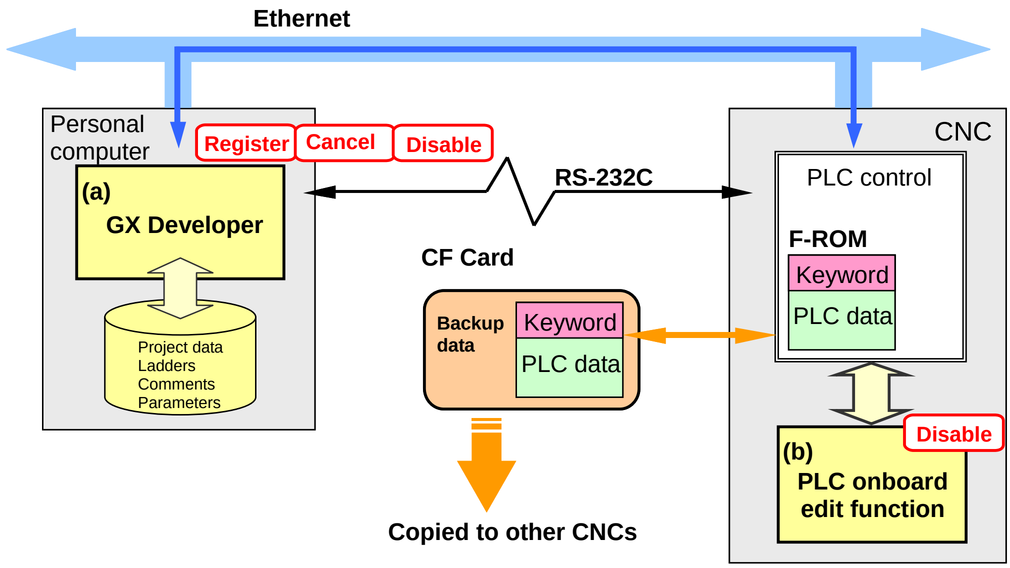
- Write protection: Writing and editing are restricted.
- Read/Write/Display protection: Writing, reading, verification and list display are restricted.

The operations, once prohibited at the keyword registration, cannot be executed by GX Developer or with the onboard PLC edit function until the keyword is disabled.

4.8.3 Structure of the Keyword Function

The keyword is encoded and stored in the PLC data of CNC. GX Developer does not store the keyword in its saved data after executing the keyword registration. The keyword is kept encoded when it is set to the backup data output in the I/O operation.
Registering a keyword means to write a keyword into the stored data. Canceling a keyword means to delete a keyword from the stored data. Disabling a keyword means to give the keyword information for verification to request an access to the stored data.

(a) GX Developer can register, cancel and disable a keyword.
(b) The PLC onboard edit function can only be used to disable a keyword.



4.8.4 File Names Excluded from the Target of Keyword Protection

Only the files with particular names as follows can be free from the keyword protection.

• The file names that have the character "U$" in the "arbitrary character string" that is defined by the file name rule.

The file name list is shown below.

Table 4.8.1 List of file names excluded from the target of keyword protection

<table>
<tr><th></th><th>Data class</th><th>Data type</th><th colspan="2">File name
: 8 + 3(extension name) characters</th><th>Storage quantity</th></tr>
<tr><td rowspan="4">1</td><td rowspan="4">Sequence program</td><td>High-speed process</td><td>H+[Arbitrary character string(7)].WPG</td><td>"U$" in the arbitrary character string sets off the keyword protection
Ex) "HxxxxxU$.WPG”</td><td rowspan="4">Total 32 comments</td></tr>
<tr><td>Main process</td><td rowspan="3">[Arbitrary character string(8)].WPG</td><td rowspan="3">"U$" in the arbitrary character string sets off the keyword protection
Ex) "U$xxxxxx.WPG"
Ex) "xxxU$xxx.WPG"</td></tr>
<tr><td>Initialization process</td></tr>
<tr><td>Standby process</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Parameter</td><td>PLC parameters</td><td colspan="2" rowspan="2">PARAM.WPA (fixed)</td><td rowspan="2">1 parameter</td></tr>
<tr><td>Network parameters</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">Device comment</td><td>Common comment</td><td colspan="2">COMMENT.WCD (fixed)</td><td rowspan="2">Total 10 comments</td></tr>
<tr><td>Comment for each program</td><td>[Arbitrary character string].WCD</td><td>"U$" in the arbitrary character string sets off the keyword protection
Ex) "U$xxxxxx.WCD"</td></tr>
<tr><td rowspan="2">4</td><td rowspan="2">PLC message</td><td>Language selection method using PLC alone</td><td>M1xxxxxx.WPG
:
M8xxxxxx.WPG</td><td>"U$" in the arbitrary character string sets off the keyword protection
Ex) "MnxxxxU$.WPG"</td><td rowspan="2">One for each language
Total 8 languages</td></tr>
<tr><td>Method linked with language selection on the screen)</td><td>M00xxxxx.WPG
:
M99xxxxx.WPG</td><td>"U$" in the arbitrary character string sets off the keyword protection
Ex) "MnnU$xxx.WPG"</td></tr>
</table>

"Arbitrary character string" means a character string consisted of up to 8 one-byte alphanumeric characters, hyphen (-) and underline (_).

4.8.5 Compatibility and Precautions when Using the System with No Keyword Support

The data before the keyword registration is compatible with the old system. The data after the keyword registration has the following limitations and precautions.

(1) The backup data after the keyword registration is not recognized as PLC data at the restoration to the old system. An error occurs when the data is restored.

(2) If the PLC data stored in F-ROM has originally been created in the old system, the data needs to be formatted before the first keyword registration.

(3) When a keyword has been registered, the data is secured in various ways. If the data is changed by any editor and the like, neither the registered keyword can be disabled nor is the data recognized as PLC data.

(4) If the registered keyword is forgotten, reformatting the data is the only way to recover the keyword.

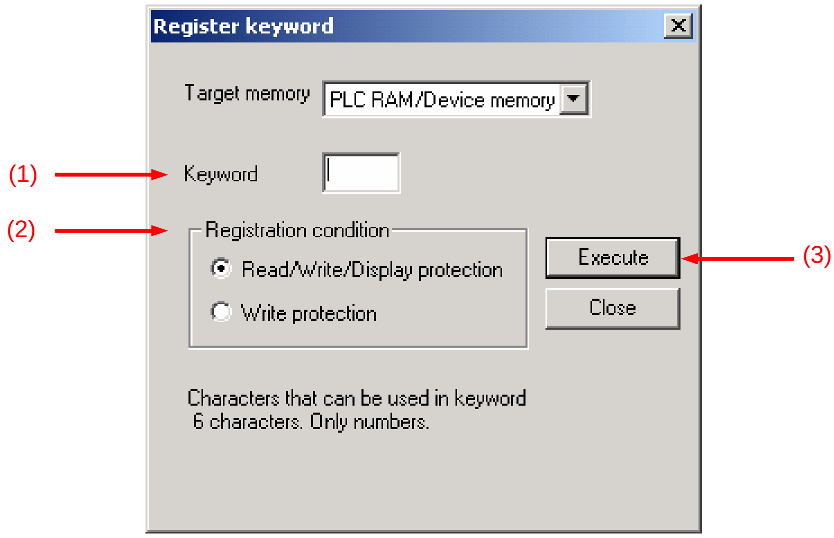
(5) The following ways are available to undo a disabled keyword.
- Turning the CNC power OFF
- Terminating the GX Developer that disabled the keyword.
- Terminating the PLC onboard edit function that was used to disable the keyword.
- Enabling the keyword with the same PLC onboard edit function as was used to disable it.

4.8.6 Registering a Keyword

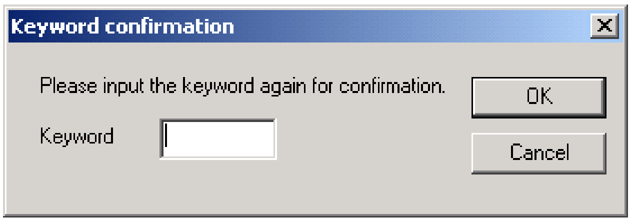
The following shows how to register or change a keyword. Open the operation screen according to the following procedure.

[Online] → [Keyword setup] → [Register]

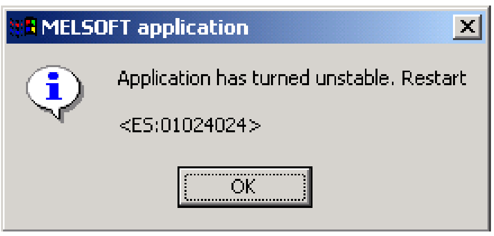
Set the "Keyword" and "Registration condition" and then click [Execute] in the screen as follows.



(1) Input a keyword (6 numeric characters).
(2) Select the functions restricted by the keyword.
   • Read/Write/Display protection: writing, reading, verification, deletion and list display are restricted.
   • Write protection: writing is restricted.
(3) When the [Execute] button is clicked, the following dialog box appears. Input the same keyword again.



[Note] Only "PLC RAM/Device memory" is available for "Target memory".
The keyword is directly written into CNC when the registration is executed. Further writing operation is not needed.
If the PLC data stored in F-ROM has originally been created in the old system, the following error message may appear. When the message appears, select [Online] - [Format PLC memory] to write the sequence programs again before registering a keyword.

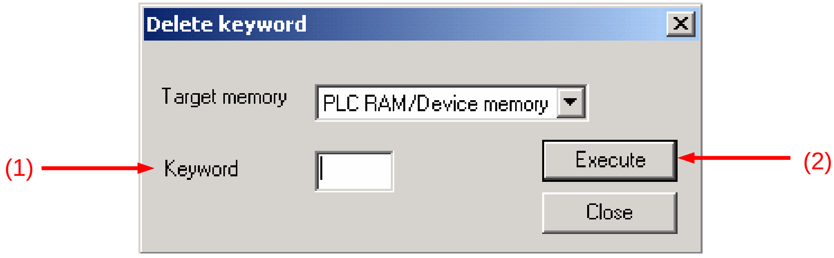


4.8.7 Canceling the Keyword

The following shows how to cancel (delete) the registered keyword. Open the operation screen according to the following procedure.

[Online] → [Keyword setup] → [Delete]

Set the "Keyword" and then click [Execute] in the following screen.



(1) Input the currently registered keyword.
(2) After inputting the keyword, click [Execute].

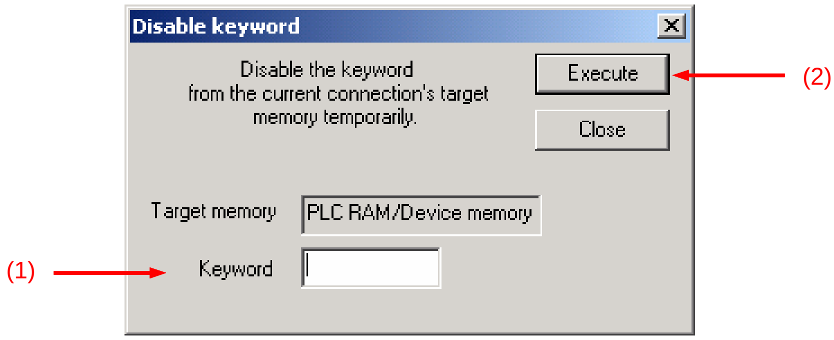
(Note) Only "PLC RAM/Device memory" is available for "Target memory".

4.8.8 Disabling the Keyword

The following shows how to release the lock imposed by a keyword in order to allow access to the CNC on which the keyword is registered. After a keyword has been released, there is free access to the CNC until GX Developer is terminated. Open the operation screen according to the following procedure.

[Online] → [Keyword setup] → [Disable]

Set the "Keyword" and then click [Execute] in the following screen.

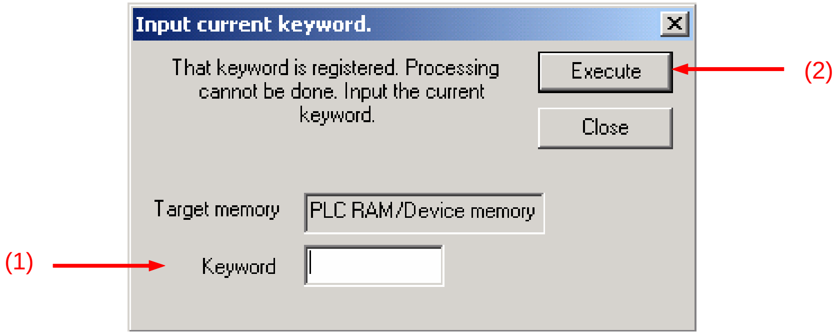


(1) Input the currently registered keyword.
(2) After inputting the keyword, click [Execute].

(Note) Only "PLC RAM/Device memory" is available for "Target memory".
Whether the keyword has been successfully disabled is not confirmed until the access is confirmed.
The following ways are available to cancel the "disabled" status of a keyword.
• Terminating GX Developer
• Closing the project that is currently opened in GX Developer

4.8.9 Disabling the Keyword as Required when Accessing to CNC

The following screen appears and requires disabling the keyword when any access is attempted to CNC on which the keyword was registered.



(1) Input the currently registered keyword.
(2) After inputting the keyword, click [Execute].

5. Sequence Program Development

The sequence program development procedures are explained in this section focusing on usage methods unique to MITSUBISHI CNC.

5.1 Development Procedures

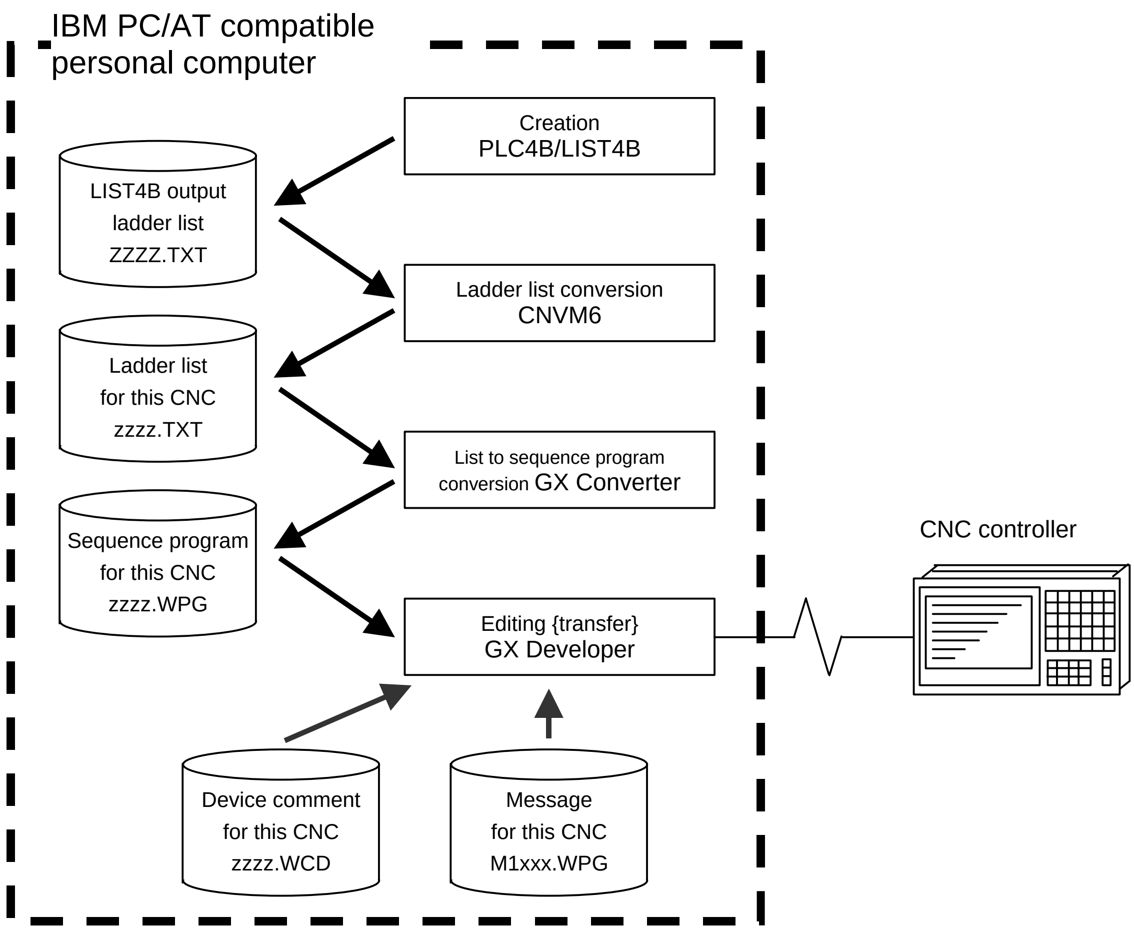
The following two methods can be used to develop the sequence program.

(1) Method to use sequence program developed with conventional MITSUBISHI CNC PLC programming tool (PLC4B).

(2) Method to newly create sequence program with GX Developer.

Refer to the enclosed GX Developer Operating Manual for the method to newly create sequence program.

5.1.1 Method for Using Sequence Program Developed with PLC4B



(1) Creation

The sequence program created for the old model is output in a list format.

(2) Conversion

The data is converted into a sequence program (list format) for this CNC using CNVM6 (PLC data integrated conversion tool). Using GX Converter (data conversion software package), the list format program is converted into the GX Developer data.

(3) Editing/transfer

The resultant program can be handled like a newly created sequence program.

5.2 Writing the Sequence Program to the CNC Controller

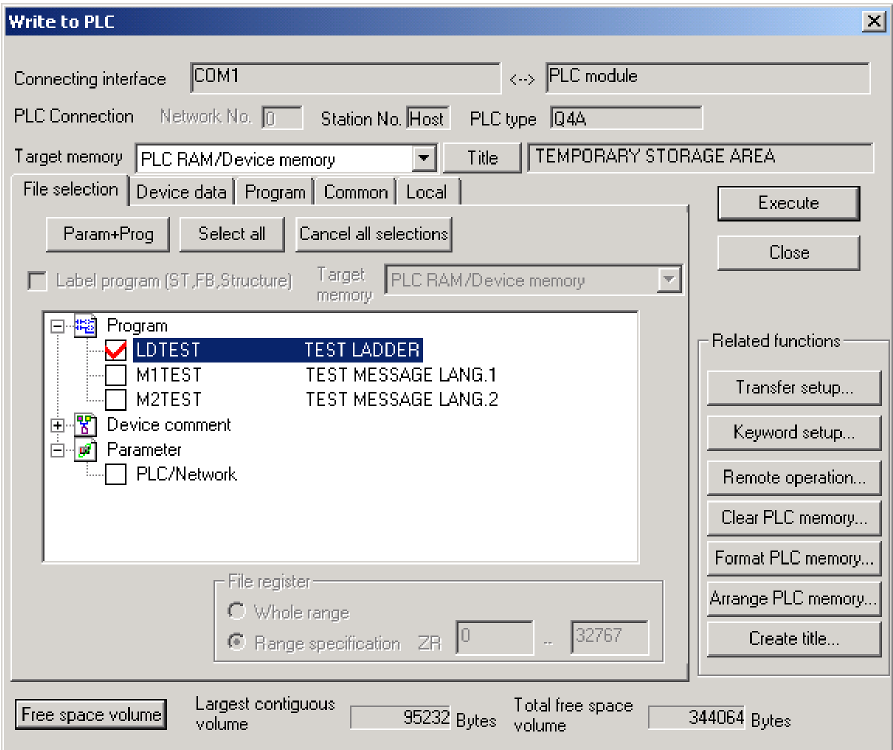
The following section explains how to write sequence programs from GX Developer to the CNC controller (especially the restrictions and CNC-specific operations.)

5.2.1 Operation Procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Write to PLC]

On the following screen, choose the sequence program file to be written from the [File selection] tab and click [Execute].
You can command RUN/STOP of the PLC using [Remote operation] under [Related functions].



[Note] As [Target memory], only [PLC RAM/Device memory] is valid.
Do not set the other tabs ([Device data], [Program], [Common], [Local]) than [File selection].

5.2.2 Writing Operation

As soon as a ladder is written from GX Developer to the CNC controller, the CNC controller converts it into the CNC-specific ladder machine code.
A conversion error occurs if any of the devices and instruction formats not supported by the CNC controller side is used.
The writing will not stop even if a conversion error occurs. The instruction causing an error is converted into a "NOP instruction" (no process instruction), and the sequence program is transferred up to the last step.

[Note] A ladder resulting in an error cannot be RUN for safety purposes.

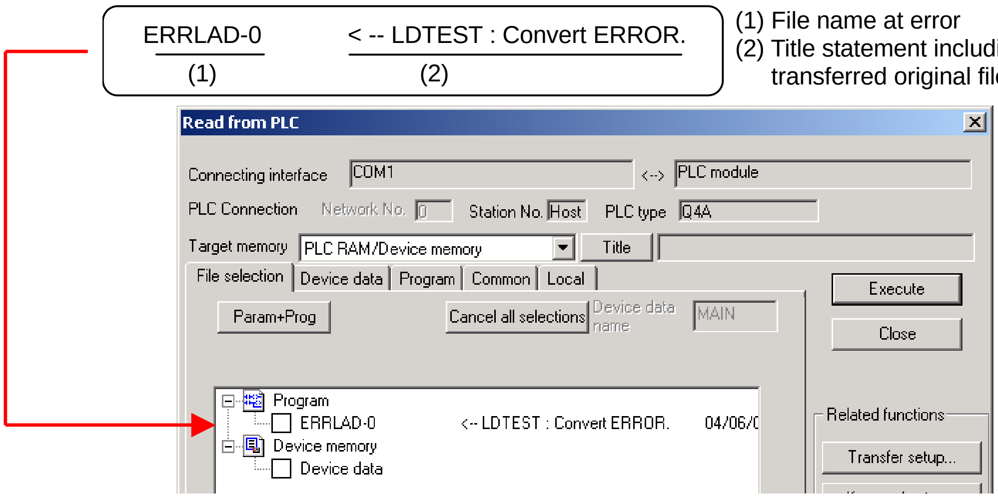
5.2.3 Operations and Check Items at Conversion Error

(1) Operations at conversion error

The following dialog appears on the GX Developer screen when a conversion error occurs.

If you execute RUN the PLC as-is, an alarm occurs on the CNC side and the PLC does not RUN.

If the ladder file resulting in a conversion error is selected with the [File Selection] tab on the [Read from PLC] screen, the file name and title will change and be displayed as shown below.
If this ladder file is read out to the GX Developer, it will be stored under the file name "ERRLAD-0".



(1) File name at error
(2) Title statement including transferred original file name

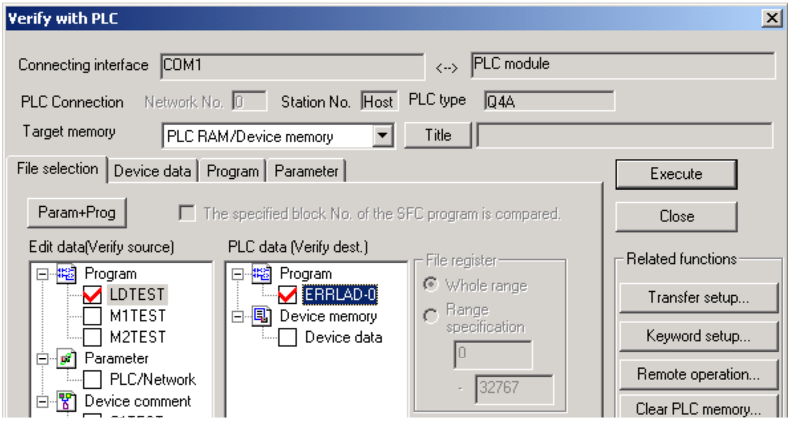
⚠ CAUTION

⚠ Do not read out a ladder file resulting in a conversion error to the GX Developer and use it. It may contain unexpected data, and result in incorrect operations.

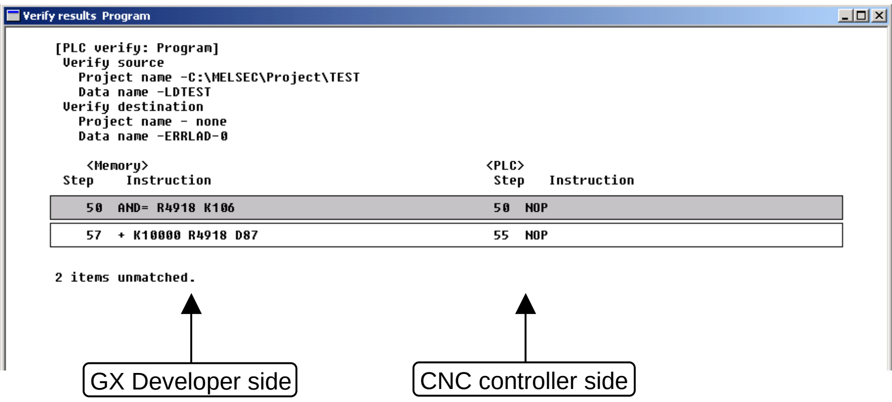
(2) How to confirm the conversion error step No.

The PLC verification function can be used to confirm the error step. For details of the PLC verification function, refer to "5.4 Verifying the Sequence Programs".

Verify source: Select the ladder file (source file) on the GX Developer side
Verify dest.: Select the file resulting in an error "ERRLAD-0" on the CNC controller side



When PLC verification is executed, the mismatching details will appear as shown in the following example. The NOP instruction section in the CNC controller side is the step with the conversion error. Double-click the mismatch to display and to edit the corresponding part of the GX Developer side.
<Memory> indicates the GX Developer side, and <PLC> the CNC controller side.



5.2.4 Operations and Check Items at the Other Errors

(1) Other errors

Some of the popular errors that would occur in writing sequence programs are given below.
Note that the description of the dialogue message is primarily provided for the MELSEC sequencer, and the precise description of the error may not be available for the M700 CNC. The last 4 digits of display No. are indicated in "Status".
For the other errors, refer to "8.1 List of Errors During GX Developer Online Operations".

Status	Message	Cause	Remedy
4005 (Note 1)	Writing of data which exceeds the capacity of the PLC was attempted. Execute again within the capacity of the PLC.	The maximum number of steps that can be executed with NC has been exceeded.	Check the size of execution area. (Refer to (2) of this section.) Reduce the number of steps for the sequence program to be executed according to that value.
4010	Cannot write because the PLC is executing a RUN command. Stop the PLC, then execute again.	The PLC of the NC is running.	After stopping the PLC of the NC, start execution again.
4021	The applicable drive is not ready. Check the applicable drive, then execute again.	The specified target memory does not exist or is not in a usable status.	Change the target memory.
4029	Insufficient file capacity. Execute again after deleting unnecessary files.	An attempt was made to write a file that exceeds the storage capacity.	Examine the file structure so that the data falls within the limited capacity.
4052	The file is write protected. Change the file attributes to enable writing to the file.	The specified target memory is a write-disabled device (F-ROM).	Specify "internal RAM" as the target memory.
4065	A mismatch occurred between the PLC and peripheral parameters Match the parameters between the PLC and peripherals.	There is a problem in the device setting value.	Set the number of device points. (Refer to section 4.6.)
4070	The program before correction differs from the registered program.	A ladder instruction outside the specification is included.	Perform verification to identify the instruction that is the cause of the problem. (Refer to section 5.2.3.)

(2) How to confirm the size of execution area

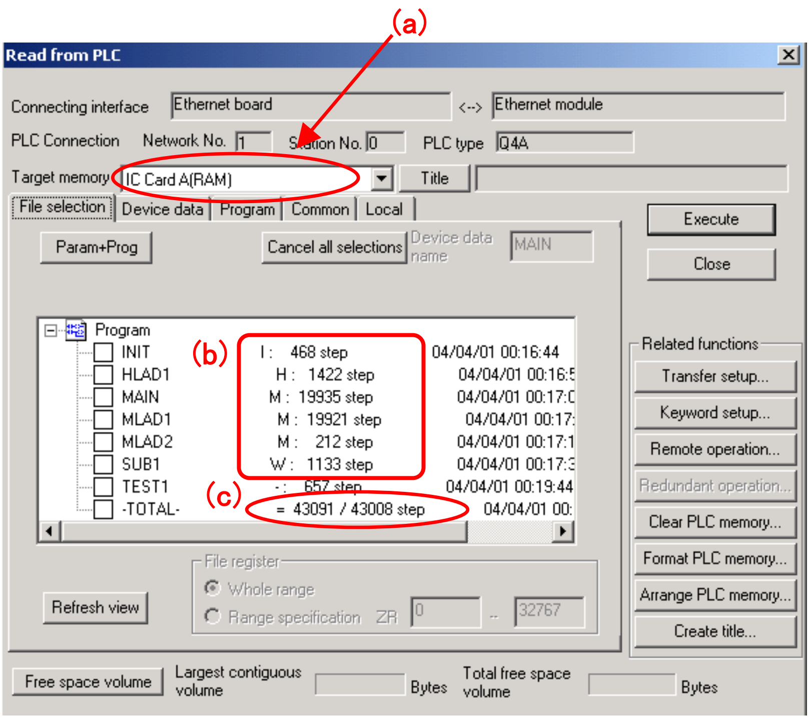
Confirm the size of execution area in the following manner when "error status (4005): Execution area size over" (See (Note 1) in the list of errors above.) occurs. Refer to "PROGRAMMING EXPLANATION 2.6 PLC Storing PLC Processing Program and Execution Mode" for details on execution area.

When [Target memory] "IC memory card A(RAM)" is selected with "Read from PLC" operation, the file name and title are changed and displayed as follows.

(a) indicates the settings of target memory. (Regular reading from PLC is executed with "internal RAM/device memory".)

(b) indicates the number of steps at execution when the title section is replaced.
An alphabet before the number of steps at execution represents the state of parameter designation.
I: Initial   H: High speed   M: Medium speed   W: Wait
-: No parameters or parameters not stored.

(c) indicates the total number of steps at execution. (Total of "initial", "high speed", "medium speed" and "wait to be executed.)
The denominator indicates the maximum size of the execution area.
When there is no parameter, the single program method is applied and "TOTAL" will not be displayed.



In the screen example above, the sequence program size must be adjusted so that the total number of steps at execution of (c) (43091) is smaller than the maximum size of the denominator's execution area (43008).

5.3 Reading the Sequence Program from the CNC Controller

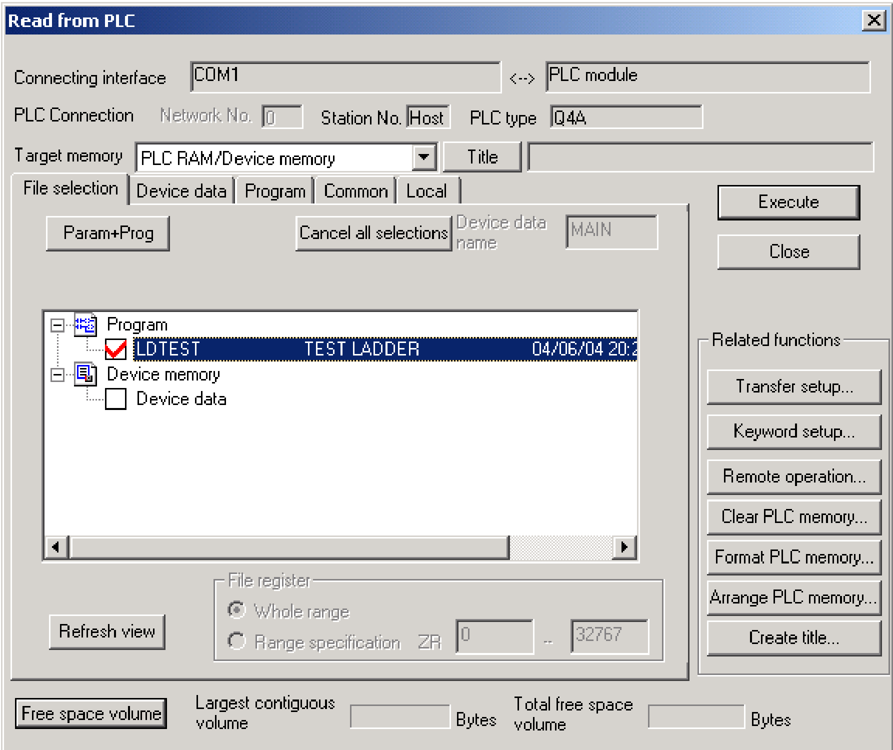
The following indicates how to read a sequence program from the CNC controller to GX Developer.

5.3.1 Operation Procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Read from PLC]

On the following screen, choose the sequence program file to be read from the [File selection] tab, and click [Execute].

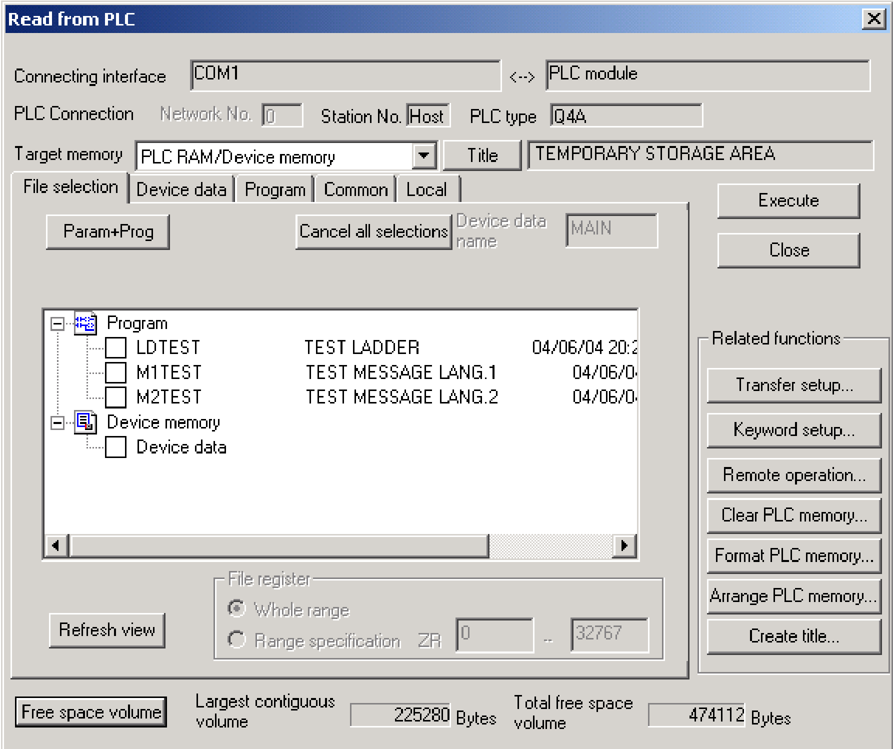


[Note] As [Target memory], the fitted memory is valid.
Do not set the other tabs ([Device data], [Program], [Common], [Local]) than [File selection].

If a sequence program file with the same name already exists in the GX Developer side, following dialogue will appear.

[Note] Choosing [Yes (Y)] in the dialogue will overwrite the GX Developer side sequence program file.
The file before overwriting will be erased.
Confirm the file enough before choosing [Yes (Y)].

The [Read from PLC] screen can also be used as a CNC controller side file listing function. Move the scroll bar of the [File selection] tab to the right to display the write date and size of each file. Click [Free space volume] to display the free area of the target memory.



5.4 Verifying the Sequence Programs

The following indicates how to verify sequence program between the CNC controller and GX Developer.

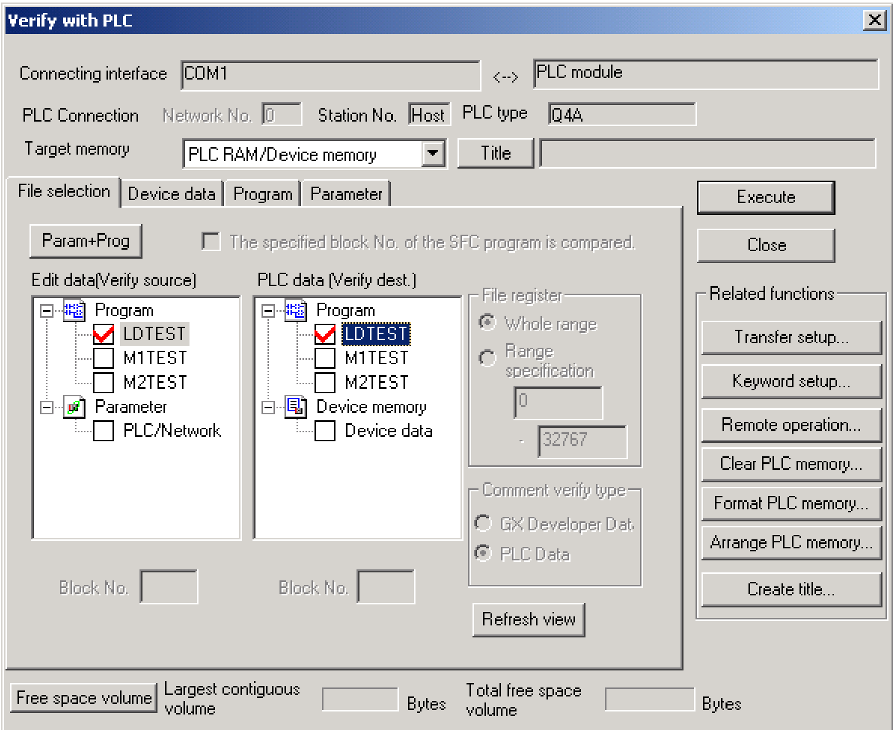
5.4.1 Operation Procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Verify with PLC]

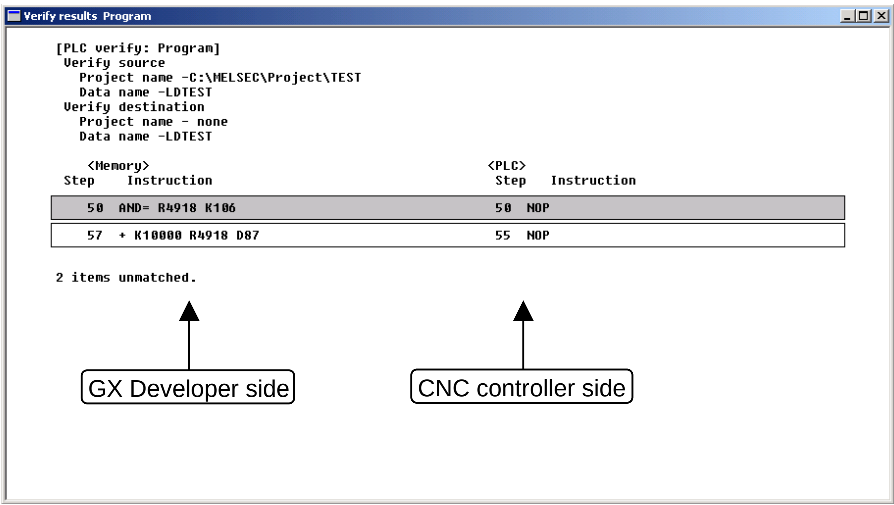
On the following screen, choose the sequence program files to be verified from the [File selection] tab, and click [Execute].

[Verify source] : GX Developer side, [Verify dest] : CNC side



[Note] As [Target memory], the fitted memory is valid.
Do not set the other tab ([Program], [Device data]) than [File selection].

If verification mismatches occur, the following mismatch screen appears. Double-click the mismatch to display the corresponding part of the GX Developer side file.



5.5 Using Sequence Programs from Older Models

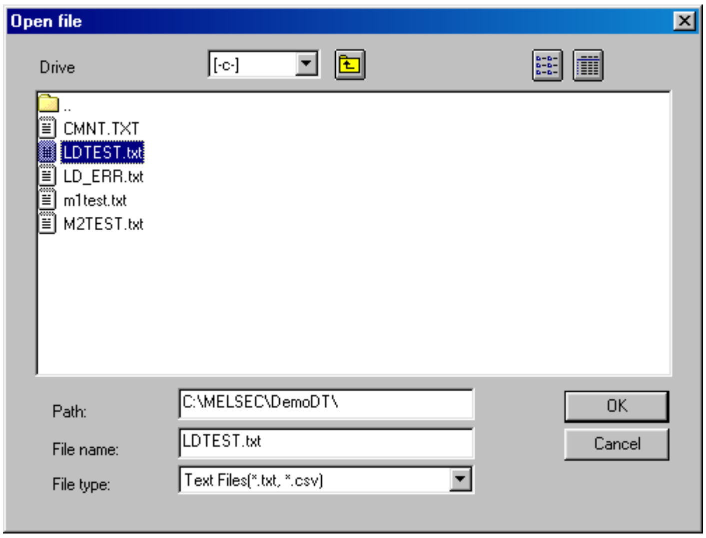
The sequence program list converted with the PLC data integrated conversion tool (CNVM7) is converted into GX Developer data with the following method. Use "GX Converter (data conversion software package)" for conversion. GX Converter can be started from the GX Developer menu.

5.5.1 Starting GX Converter and Specifying the File to be Converted

Perform the following operation from GX Developer to start GX Converter (read).

[Project] → [Import file] → [Import to TEXT, CSV format file]

On the following screen, choose the file to be converted (LDTEST.TXT) and click [OK].

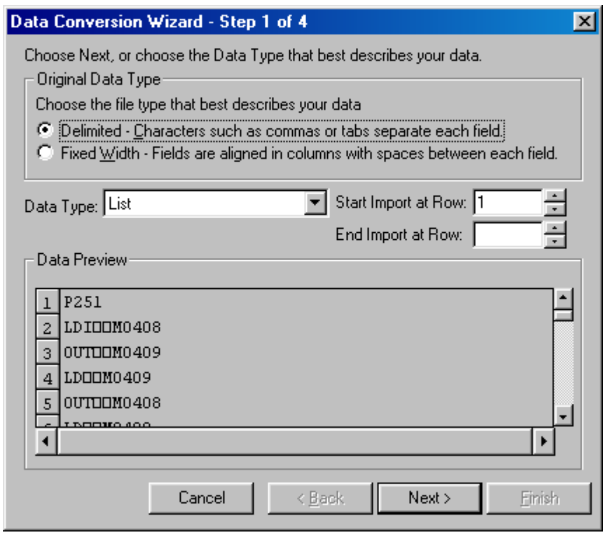


5.5.2 Conversion Format Setting

Set the conversion format on the following data conversion wizard screen.

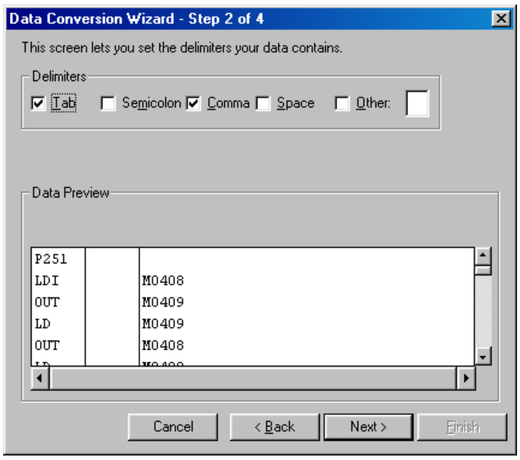
(1) Data conversion wizard 1/4

Choose [Original Data Type]-[Delimited] and [Data Type]-[List], and click [Next>].



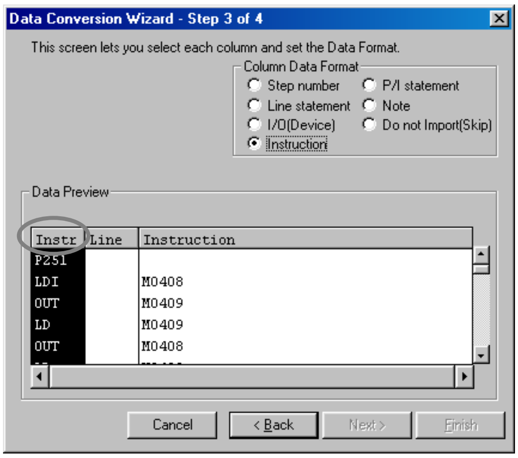
(2) Data conversion wizard 2/4

Choose [Delimiters]-[Tab] and click [Next>].



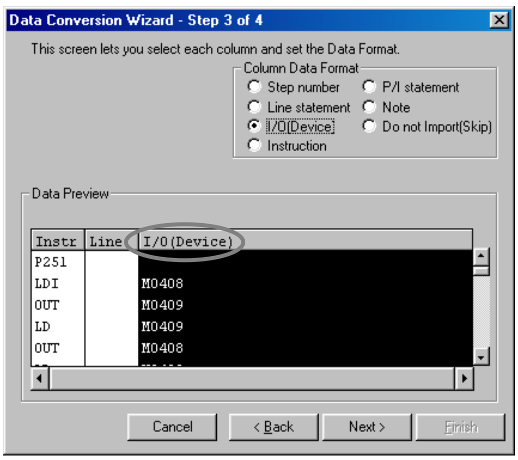
(3) Data conversion wizard 3/4

Choose to highlight the instruction column part in the [Data Preview] list and choose [Column Data Format]-[Instruction].



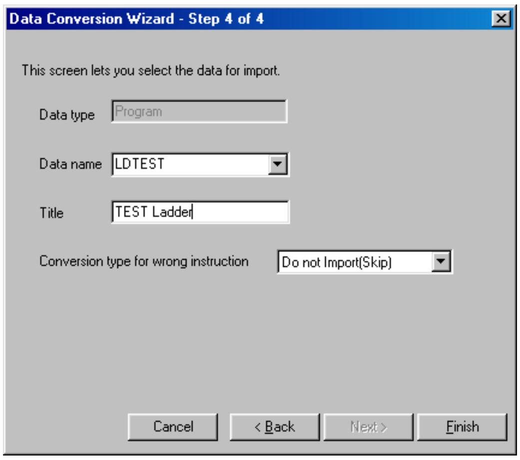
(4) Data conversion wizard 3/4

Further, choose to highlight the argument column part in the [Data Preview] list and choose [Column Data Format]-[I/O(Device)]. Click [Next>].



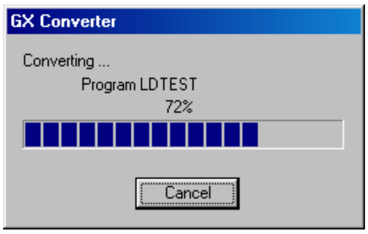
(5) Data conversion wizard 4/4

Set the program name used on GX Developer at [Data name] column and a sequence program annotation at [Title] column, and click [Finish].



(6) Completion

The setting is complete when the following completed dialog appears after the converting dialog. Click [OK].



5.6 Monitoring the Sequence Program

There are no MITSUBISHI CNC-specific operations to monitor a sequence program. Refer to the Operating Manual for the operation methods. For usable functions, refer to "2.2 Function Support Conditions (Online Section)". This section explains the operation procedure outline and precautions.

5.6.1 Operation Procedure

Perform the following operation from GX Developer to start monitoring.

(1) Display the sequence program to be monitored and move to the circuit part to be monitored.

(2) Perform the following operation to start monitoring.

[Online] → [Monitor] → [Monitor mode] or F3

(3) Perform the following operation to stop monitoring.

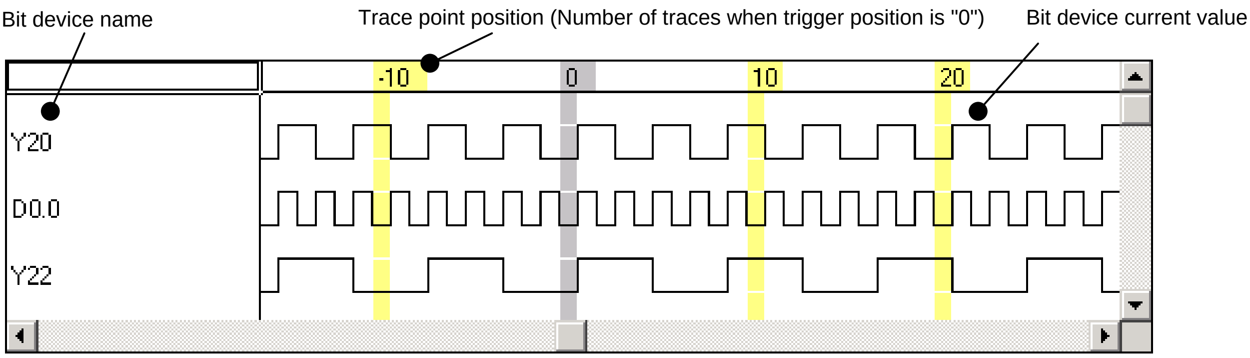
[Online] → [Monitor] → [Stop monitor] or Alt + F3

[Caution] If the sequence program being RUN with CNC controller differs from the sequence program displayed on GX Developer, monitoring will not result in an error but will appear to continue normally.
Confirm that the sequence program on the CNC controller side and GX Developer are the same before starting monitoring.

5.7 Executing Sampling Trace on Device

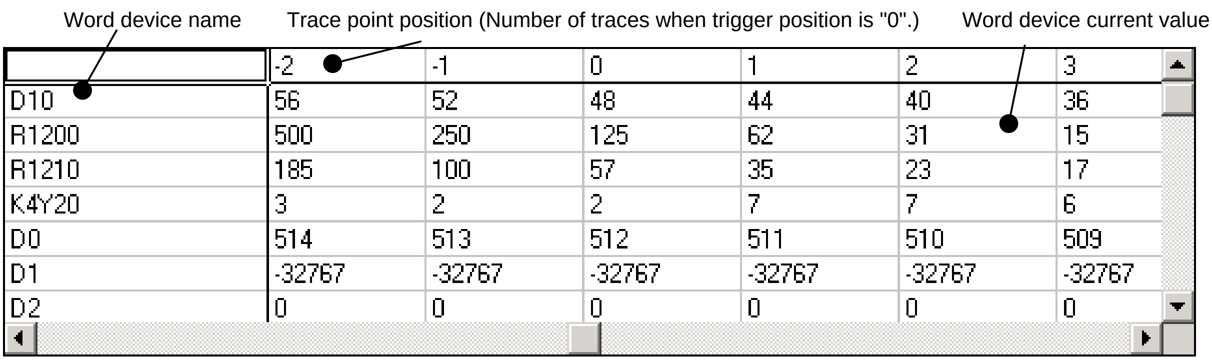
Status of various devices that are used for the CNC controller external signals and for the user ladder can be traced by using GX Developer. Trace result at an arbitrary point will be read out from CNC controller and displayed on a screen or output in a CSV file format.

► Example of trace result display for bit devices
Current value of bit device per sampling point is displayed.



Example of bit device trace result display

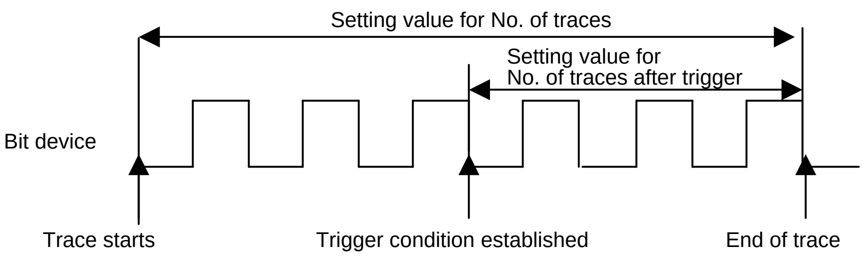
► Example of trace result display for word devices
Current value of word device per sampling point is displayed.



	-2	-1	0	1	2	3
D10	56	52	48	44	40	36
R1200	500	250	125	62	31	15
R1210	185	100	57	35	23	17
K4Y20	3	2	2	7	7	6
D0	514	513	512	511	510	509
D1	-32767	-32767	-32767	-32767	-32767	-32767
D2	0	0	0	0	0	0

Example of word device trace result display

No. of traces, trace condition, trigger condition, and No. of traces after trigger can be set as trace setting.
Trace is executed when the trace condition is set.
After the trigger condition is set, execute "No. of traces after trigger" only and end sampling. Sampling the status before and after the trigger is set is also possible.



Specifications chart (Establishment of trigger condition to the end of sampling)

5.7.1 Basic Operation

Use GX Developer for operation. Execute sampling trace function, following the procedures below.

(1) Carry out trace setting.

Setting item	Setting details
No. of traces	Set the number of traces to be saved. The latest data is saved for the number of traces. If exceeds the number of traces, oldest data is erased in order.
No. of traces after trigger	Set the number of traces taken place after trigger has been established until sampling is completed.
Trace point	Set the data trace timing. "Each scan" and "Detail" can be selected. "Detail" enables the trace execution condition setting.
Trigger point	Set the trigger timing. When trigger is established, status of trace device is taken in. "At the time of trigger operation from GX Developer" and "Detail" can be selected. "Detail" enables the trigger establishment condition setting.
Trace device	Set the device for sampling.

(2) Write the trace settings into CNC controller.

(3) Execute tracing. CNC controller starts tracing the device. Tracing execution status can be checked with GX Developer tracing status display or CNC special relay SM800 to SM805. Tracing will be continued until the following conditions are set.

(a) Tracing is completed upon establishment of trigger.

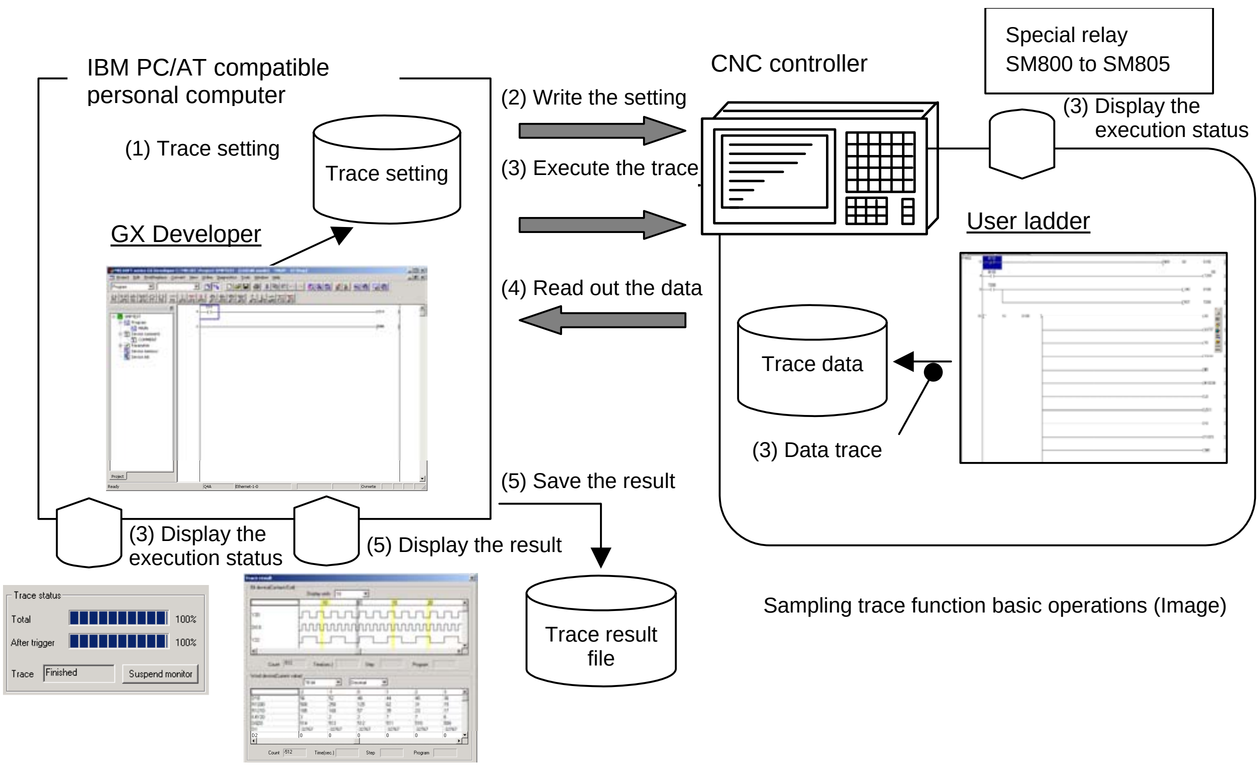
(b) Stop tracing from GX Developer.

(c) Turn the CNC controller power OFF.

(d) PLC turns to STOP mode.

(4) Read trace data out from CNC controller.

(5) Display the result and output in a file format, if necessary.



Sampling trace function basic operations (Image)

5.7.2 Basic Specifications

Basic specifications list

<table>
<tr><th>Item</th><th colspan="3">Specifications</th></tr>
<tr><td>No. of traces</td><td colspan="3">The number between 1 and 8192 can be set. Note that, however, the total trace data size has to be smaller than 60kbyte.
Refer to *1 for trace data size calculation.</td></tr>
<tr><td>No. of traces after trigger</td><td colspan="3">Sets the number smaller than the No. of traces.</td></tr>
<tr><td rowspan="3">Trace point
(Two types of setting – "Each scan" and "Detail" – are available.)</td><td>Each scan</td><td colspan="2">Traces per scan of the main process.</td></tr>
<tr><td rowspan="2">Detail
(Refer to *4 for precaution.)</td><td>Word device</td><td>Applicable device: Refer to *2.
Setting condition: Sets the value to be used for judgement. When the setting value becomes equal to the word device value, judgement will be effective and trigger is executed.</td></tr>
<tr><td>Bit device</td><td>Applicable device: Refer to *2.
Setting condition: Sets ↑ or ↓. When the setting condition is satisfied, trace is executed.</td></tr>
<tr><td rowspan="3">Trigger point
(Two types of setting – "At the time of trigger operation from GX Developer" and "Detail" are available.)</td><td>At the time of trigger operation from GX Developer</td><td colspan="2">Executes trigger by GX Developer operations.</td></tr>
<tr><td rowspan="2">Detail
(Refer to *5 for precaution.)</td><td>Word device</td><td>Applicable device: Refer to *2.
Setting condition: Sets the value to be used for judgement. When the setting value becomes equal to the word device value, judgement will be effective and trigger is executed.</td></tr>
<tr><td>Bit device</td><td>Applicable device: Refer to *2.
Setting condition: Sets ↑ or ↓. When the setting condition is satisfied, trace is executed.</td></tr>
<tr><td>Trace device</td><td colspan="3">50 points of word device, 50 points of bit device can be set. Note that, however, the total trace data size has to be smaller than 60kbyte.
Refer to *1 for trace data size calculation.
Refer to *3 for applicable devices.</td></tr>
</table>

*1 Calculation of trace data size

Set the number of traces and the trace devices so that the trace data size is 60kbyte (61440 byte) or smaller. Trace data size will be calculated as follows.

[Trace data size (byte)] = [Size required for one trace (byte)] x [Number of traces]

Size required for one trace is calculated from word device points and bit device points of the trace device and the size required for one trace of each device.

Size required for one trace of each device

Trace device type	Size (byte) required for one trace
Word device	2byte per 1 point
Bit device	2byte per 1 unit (1 unit = 16 points) 1 to 16 points → 1 unit → 2byte 17 to 32 points → 2 units → 4byte 33 to 48 points → 3 units → 6byte 49 to 50 points → 4 units → 8byte

Calculation example for trace data size is shown below.

Trace data size calculation example

No.	Bit device points	Word device points	Trace points	Trace data size	Sampling
1	16 points (1 unit)	2 points	8192 points	(1x2+2x2)x8192 = 49152 byte	Possible
2	8 points (1 unit)	8 points	1000 points	(1x2+8x2)x1000 = 18000 byte	Possible
3	50 points (4 units)	50 points	568 points	(4x2+50x2)x568 = 61344 byte	Possible
4	50 points (4 units)	50 points	569 points	(4x2+50x2)x569 = 61452 byte	Not possible

*2 Applicable devices with trace/trigger point details setting

Applicable device
Bit device: X, Y, M, L, F, SB, B, SM, T(contact), ST(contact), C(contact) Word device: T(current value), ST(current value), C(current value), D, R, SW, SD, W Following qualifications are available for the above devices. • Bit device digit designation • Word device bit designation

Process following an inapplicable device setting is as follows.
• If one or more applicable devices already exist, the setting of the inapplicable device will be ignored.
• If no applicable devices exist, error occurs when trace setting file is read upon trace execution.

*3 Applicable devices with device setting

Applicable device
Bit device: X, Y, M, L, F, SB, B, SM, T(contact), T(coil), ST(contact), ST(coil), C(contact), C(coil) Word device: T(current value), ST(current value), C(current value), D, R, SW, SD, W Following qualifications are available for the above devices. • Bit device digit designation • Word device bit designation

Process following an inapplicable device setting is as follows.
• If one or more applicable devices already exist, the setting of the inapplicable device will be ignored.
• If no applicable devices exist, error occurs when trace setting file is read upon trace execution.

*4 Precautions for trace point details setting

• When trace point is set only with "Step No.", the operation will be same as when trace point is set to "Each scan".
• When trace point is set with AND condition of "Step No." and "Device", "Step No." setting will be ignored and only "Device" setting will be effective.
• When only devices that are not corresponding with MITSUBISHI CNC are set for "Device", operation will be same as when trace point is set to "Each scan".

Operations at the time of trace point details setting are as shown in the table below.

List of operations at trace point details setting

Setting (○→ Yes ×→ No)		Operations
Device	Step No.	
○	○	Valid setting available in device → Device Valid setting not available in device → Each scan
○	×	Valid setting available in device → Device Valid setting not available in device → Each scan
×	○	Each scan
×	×	Setting disabled (Error is displayed on GX Developer)

*5 Precautions for trigger point details setting

- When trigger point is set only with "Step No.", the operation will be same as when trigger point is set to "At the time of trigger operation from GX Developer".
- When trigger point is set with AND condition of "Step No." and "Device", "Step No." setting will be ignored and only "Device" setting will be effective.
- When only devices that are not corresponding with MITSUBISHI CNC are set for "Device", operation will be same as when trigger point is set to "At the time of trigger operation from GX Developer".

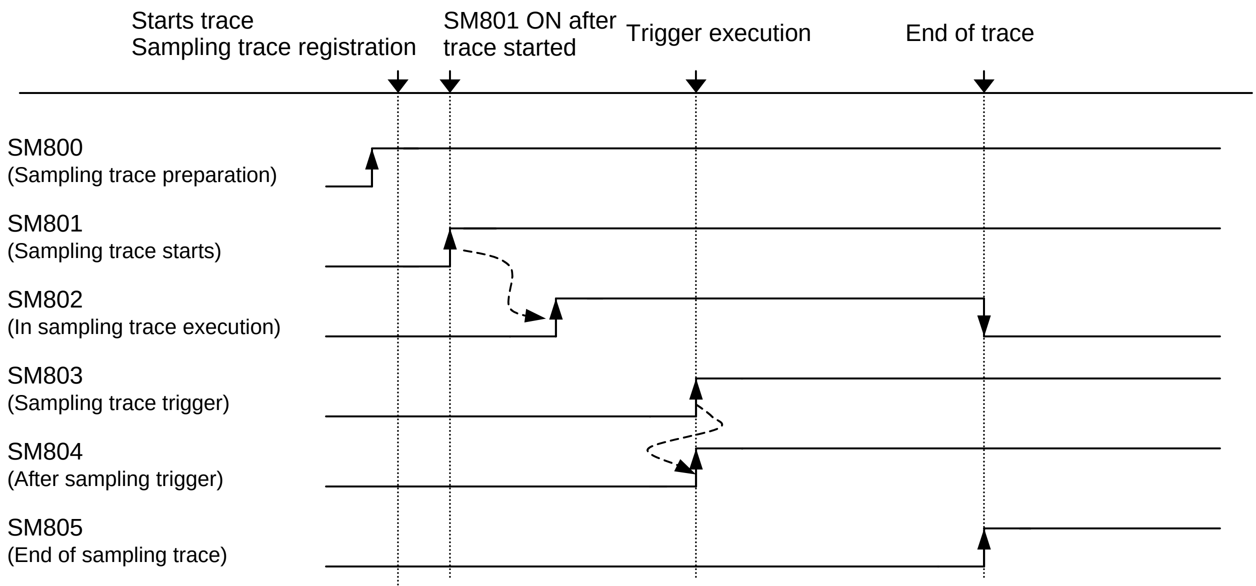
Operations at the time of trigger point details setting are as shown in the table below.

List of operations at trigger point details setting

Setting (○→ Yes ×→ No)		Operations
Device	**Step No.**	
○	○	Valid setting available in device → Device
		Valid setting not available in device → At the time of trigger operation from GX Developer
○	×	Valid setting available in device → Device
		Valid setting not available in device → At the time of trigger operation from GX Developer
×	○	At the time of trigger operation from GX Developer
×	×	Setting disabled (Error is displayed on GX Developer)

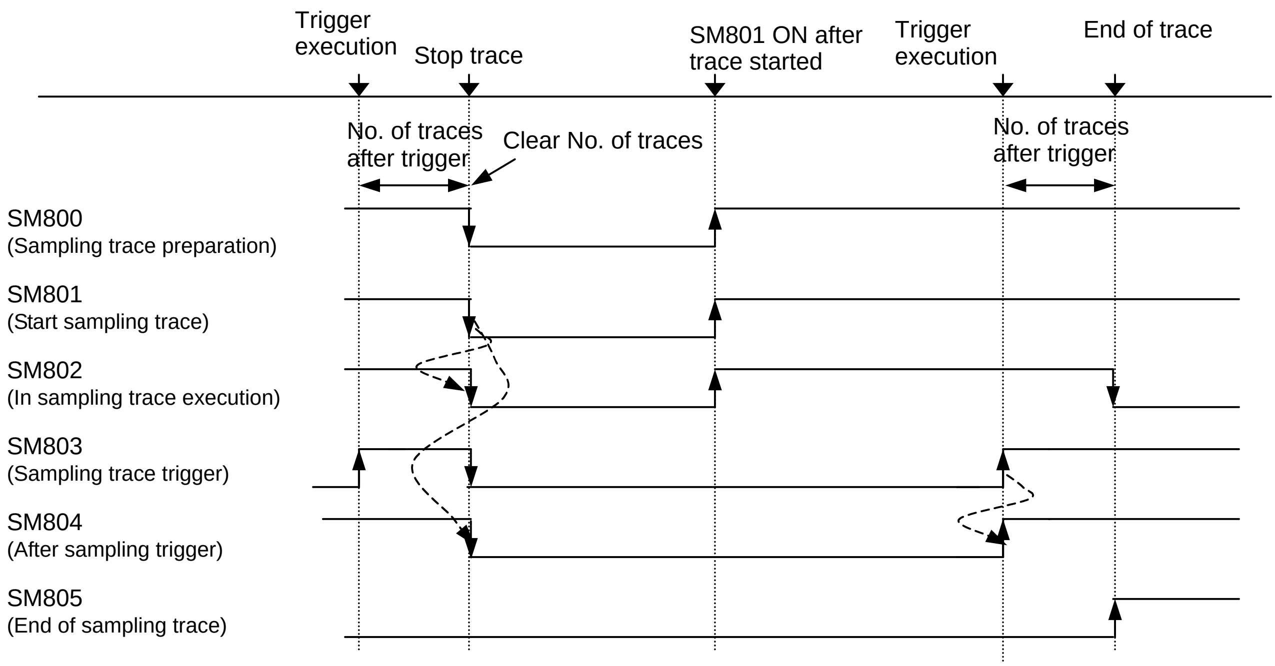
5.7.3 Status of Special Relay during Sampling Trace

Status during sampling trace function being executed can be checked by the status of SM device 800 to 805.



*SM800 automatically turns ON when sampling trace preparation is done.

Device status when trace execution operation is carried out.



Device status when trace stop operation is carried out

Specifications for SM800 to SM805 is as shown below.

Specifications list for the devices that are related with sampling trace function

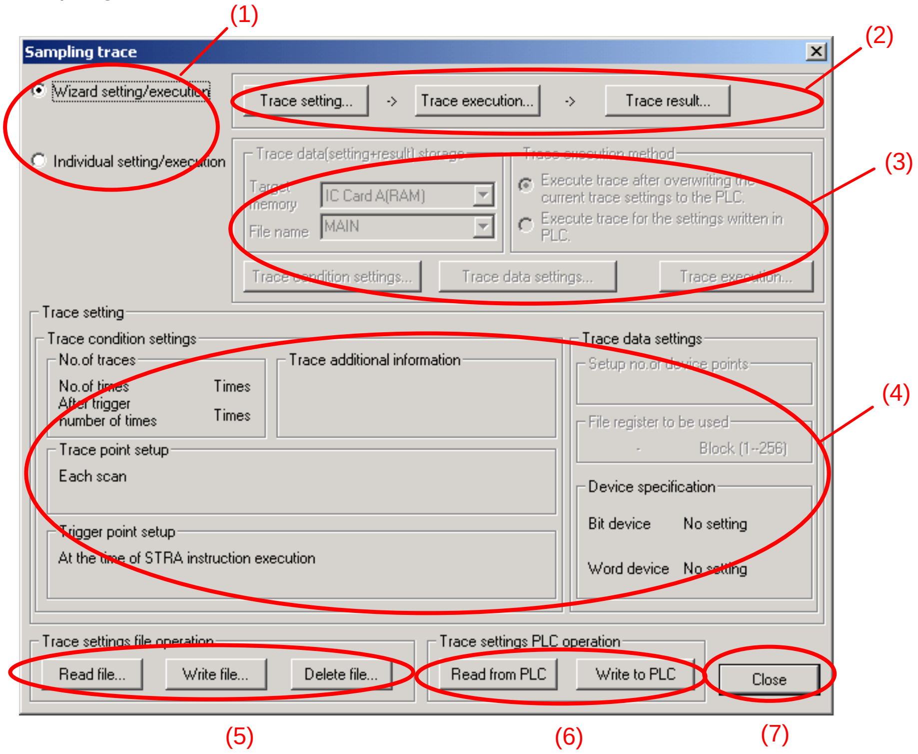
No.	Name	Details
SM800	Sampling trace preparation	OFF: Preparation not completed ON: Preparation completed
SM801	Starts sampling trace	OFF: Cancel ON: Start
SM802	In sampling trace execution	OFF: Cancel ON: Start
SM803	Sampling trace trigger	OFF→ON : Trigger execution
SM804	After sampling trace trigger	OFF: Not after trigger ON: After trigger
SM805	End of sampling trace	OFF: Not completed ON: Completed

5.7.4 Sampling Trace Operation Screen

In this section, outline of the operation procedures and precautions are explained, using the case where wizard setting/execution is applied.
Individual setting/execution is also available. For available functions, refer to "2.2 Function Support Conditions (Online Section)".
Refer to GX Developer Version 8 Operating Manual (SW8D5C-GPPW-*) for basic operations.

5.7.4.1 Sampling Trace Main Screen

Start the sampling trace main screen below by selecting [Online]→[Trace]→[Sampling trace]. All the operations for sampling trace functions are carried out on this main screen.



Sampling trace main screen

(1) Set the sampling trace execution method. Select either "wizard setting/execution" method or "Individual setting/execution" method. Necessary setting items are set in an order by using the wizard setting/execution method.

(2) This is the menu when the wizard setting/execution method is applied. Sampling trace execution method is valid when wizard setting/execution is applied. Click the button in the order of "Trace setting...", "Trace execution..." and "Trace result...".

(3) This is the menu when the individual setting/execution method is applied. Sampling trace execution method is valid when individual setting/execution is applied. Click the button in the order of "Trace condition setting", "Trace data setting" and "Trace execution". Setting details are same as when wizard setting/execution is applied.

(4) Display the details of setting for the trace currently valid.

(5) This is the menu for trace setting file operation. Saving of the currently valid trace settings in the local area is possible, as well as reading and deleting of the saved file.

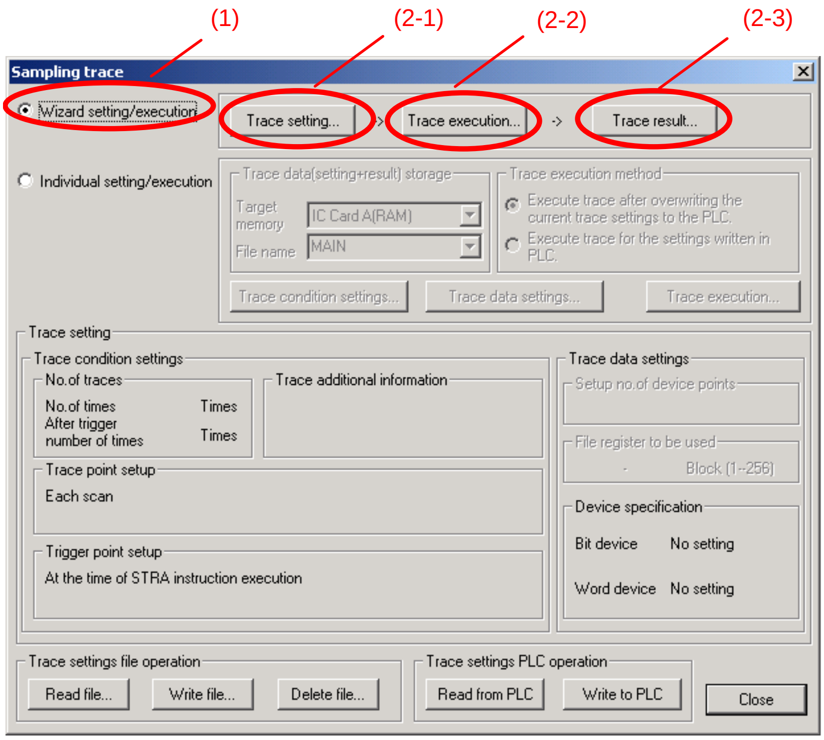
(6) This is the menu for trace setting PC operation. Currently valid trace settings can be written into CNC, and also the trace settings currently set in CNC can be read out.

(7) The [Close] button closes the "Sampling trace" screen. Tracing will continue even if the screen is closed during trace execution.

5.7.4.2 Wizard Setting/Execution Screen

Perform the following operation from GX Developer to start the sampling trace.
[Online]→[Trace]→[Sampling trace]

(1) Select wizard setting/execution with radio button.
(2) Click in the following order and operate according to the wizard.
(2-1) [Trace setting] : Refer to 5.7.4.3 for details.
(2-2) [Trace execution] : Refer to 5.7.4.4 for details.
(2-3) [Trace result] : Refer to 5.7.4.5 for details.

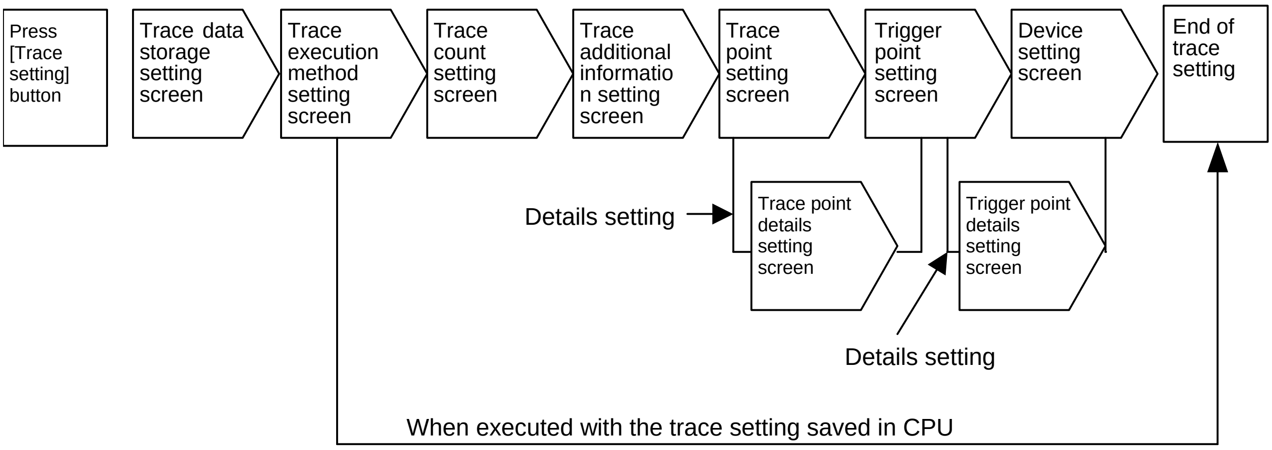


Explanation of wizard setting/execution

5.7.4.3 Trace Setting

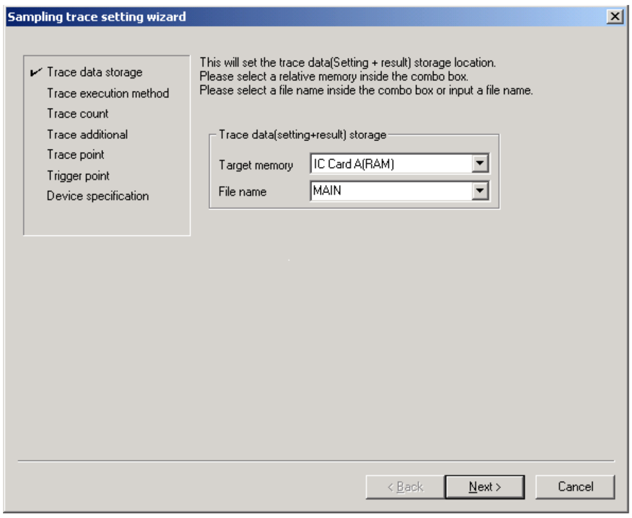
Start "Sampling trace setting wizard" screen by clicking [Trace setting].
Set the conditions and data required for sampling trace according to the wizard.

Screen transitions for trace setting is as shown below.



Trace setting screen transitions at wizard setting/execution

(1) Trace data (setting/result) storage setting screen
"Target memory" and "File name" cannot be set. Click [Next >] and proceed to the trace execution method setting screen.



At wizard setting/execution – Sampling trace setting wizard - Trace data (Setting + result) storage setting screen

[Note]

- Since "Target memory" is DRAM integrated in CNC controller, trace data (setting+result) will be deleted when the power is turned OFF.
- Trace data (setting+result) that can be saved on CNC controller is only one. Trace data result is overwritten every [Trace execution] execution.
- When file name is changed, the overwriting confirmation dialogue may not be dispalyed at PC write operation.
- When file name has been changed, trace result may not be displayed with [Trace result]. Read the data out from PLC, then click [Trace result] again.

(2) Trace execution method setting screen

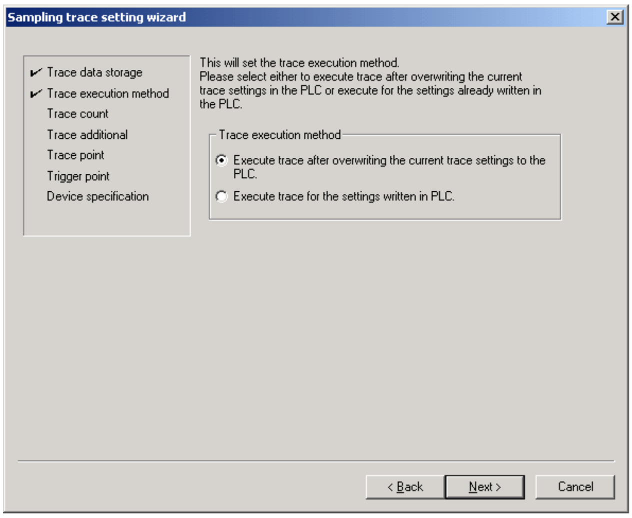
Select a trace execution method and click [Next >].

When "Execute trace after overwriting the current trace settings to the PLC" is selected:

Proceed to the trace count setting screen.

When "Execute trace for the settings written in PLC" is selected:

End "Sampling trace setting wizard" and return to the "Sampling trace" screen.

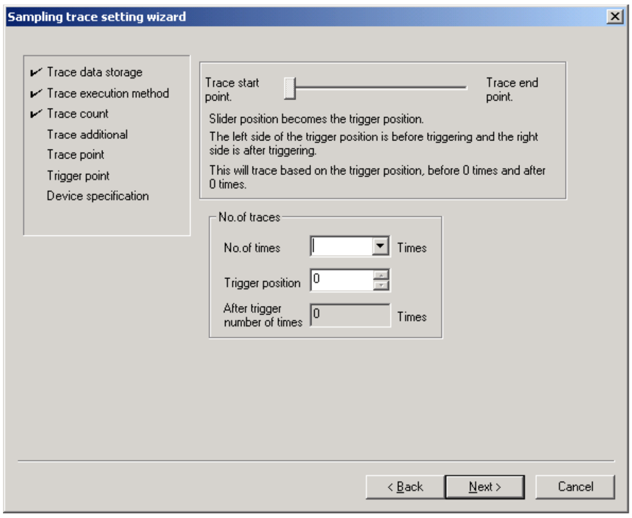


At wizard setting/execution – Sampling trace setting wizard – Trace execution method setting screen

(3) Trace count setting screen

Set the number of traces and trigger position, click [Next >] and proceed to the trace additional information setting screen.

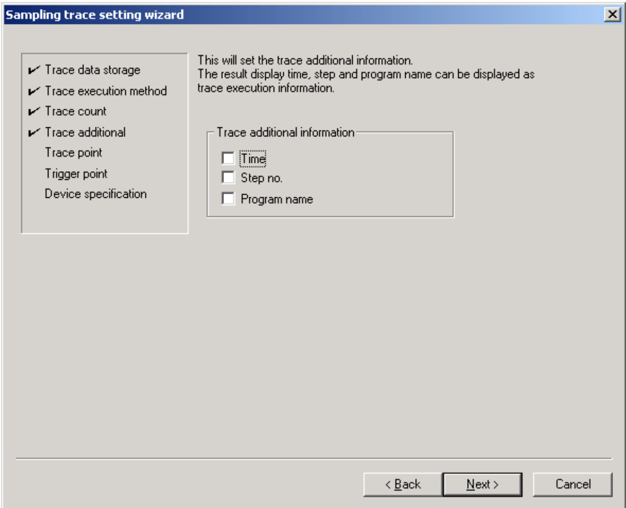
Refer to "5.7.2 Basic Specifications" for details on trace count that can be set.



At wizard setting/execution – Sampling trace setting wizard – No. of traces setting screen

(4) Trace additional information setting screen

Only "Time" can add information. Put a check mark at "Time" to display time when the trace result is shown. Click [Next >] and proceed to the trace point setting screen.

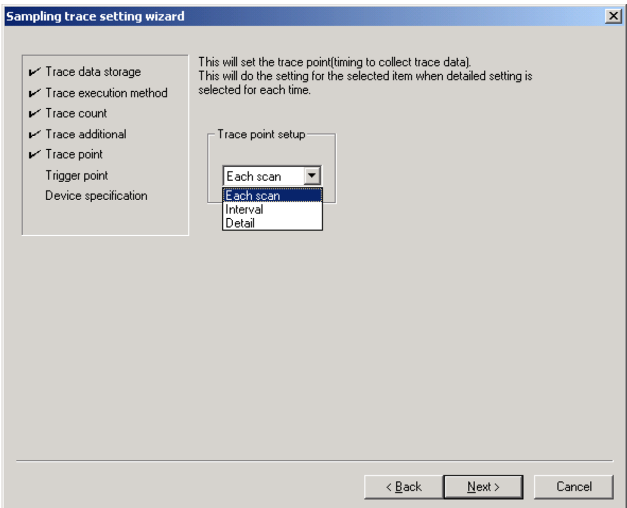


At wizard setting/execution – Sampling trace setting wizard – Trace additional information setting screen

(5) Trace point setting screen

Select "Each scan" or "Detail" and click [Next >].
When "Each scan" is selected, proceed to the trigger point setting screen.
When "Detail" is selected, proceed to the trace point details setting screen. Set "Device" at [Trace point setting].

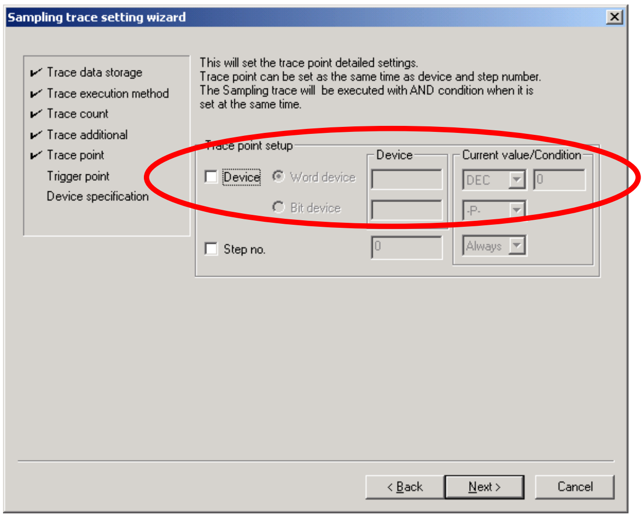
- "Each scan": Trace is executed after scanning the main process.
- "Detail": Trace is executed when the device set with trace point setting satisfies its condition after scanning the main process.



At wizard setting/execution – Sampling trace setting wizard – Trace point setting screen

[Note]

Only "Each scan" and "Detail" are valid for trace point setting.
When "Interval" is set, the movement will be the same as when "Each scan" is set.



At wizard setting/execution – Sampling trace setting wizard – Trace point setup – Details setting screen

[Note]

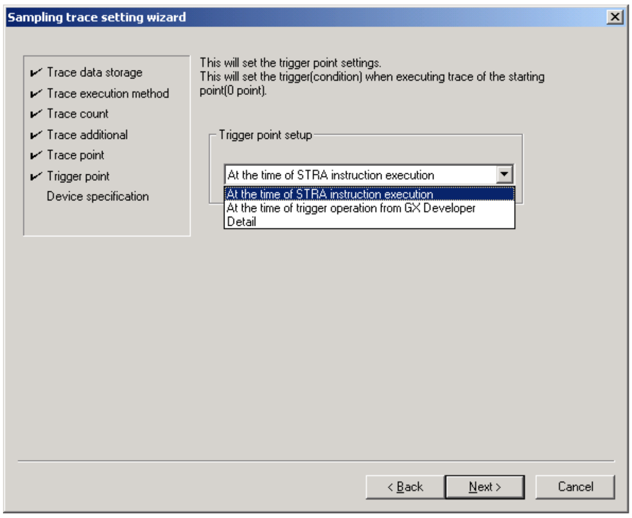
Only "Device" is valid for trace point setting.
Do not set "Step No.".
Refer to "5.7.2 Basic Specifications" for the devices that can be set.

(6) Trigger point setting screen

Select "At the time of trigger operation from GX Developer" or "Detail" and click [Next >].
When "At the time of trigger operation from GX Developer" is selected, proceed to the device setting screen.
When "Detail" is selected, set "Device" at [Trigger point setting].

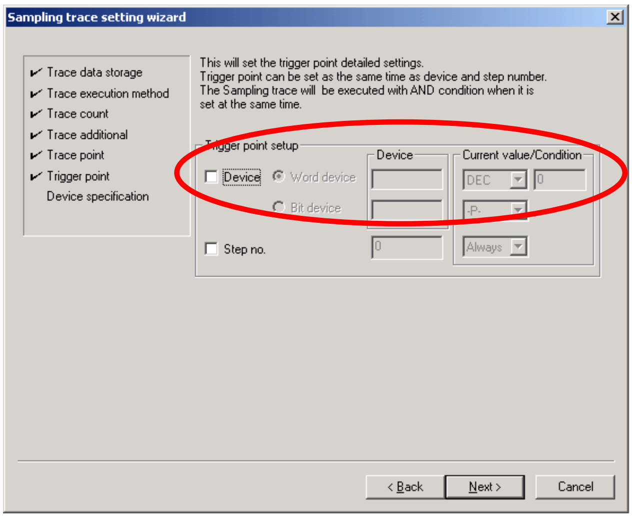
"At the time of trigger operation from GX Developer":
When executing trace, carry out "Execute trigger" from the "Sampling trace setting wizard execution" screen.
The data at the time of "Execute trigger" after scanning of the main process will be the start point (0 point).

"Detail":
Check trigger conditions after scanning of the main process. The data at the time of trigger condition establishment will be the start point (0 point). (Refer to [II. PROGRAMMING EXPLANATION 2.4.1 Number and Types of Registerable Programs] for explanation of 'Main process'.)



At wizard setting/execution – Sampling trace setting wizard – Trigger point setting screen

[Note]
Only "At the time of trigger operation from GX Developer" and "Detail" are valid for trigger point setting.
When "At the time of STRA instruction execution" is set, operation will be the same as when "At the time of trigger operation from GX Developer" is set.



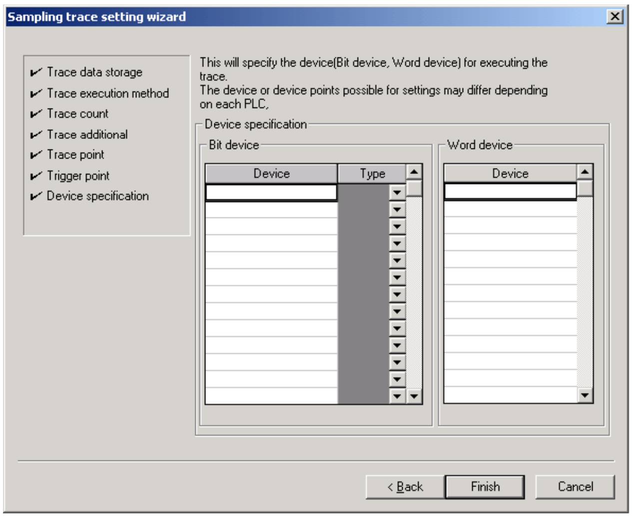
At wizard setting/execution – Sampling trace setting wizard – Trigger point setup – Details setting screen

[Note]

Only "Device" is available for trigger point "Details" setting.
Do not set "Step No.".
Refer to "5.7.2 Basic Specifications" for the devices that can be set.

(7) Device setting screen

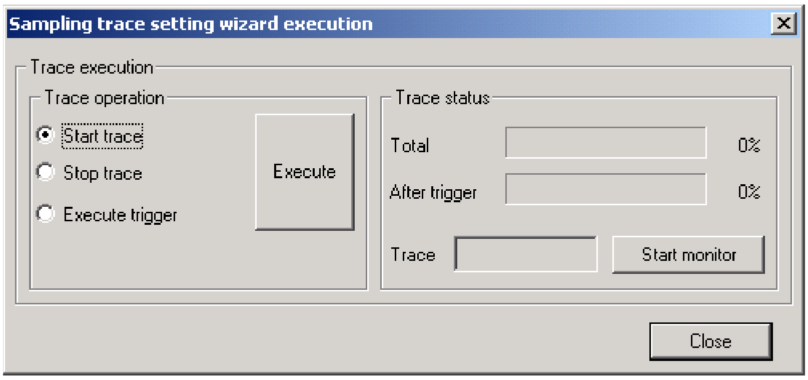
Set the devices of which trace is executed and click [Next >].
After setting the device, click [Finish] and end the trace setting.
Refer to "5.7.2 Basic Specifications" for the devices that can be set.



At wizard setting/execution – Sampling trace setting wizard – Trigger point setting – Device setting screen

5.7.4.4 Trace Execution

Start the "Sampling trace setting wizard execution" screen below by clicking [Trace execution].
Carry out "Start trace", "Stop trace" and "Execute trigger".



At wizard setting/execution – Trace execution screen

(1) Starting trace

Select "Start trace" under [Trace operation] and click [Execute]. Trace begins.
Tracing progress can be checked at [Trace status].

Status of each device under tracing can be check with [Trace result].
Display the current status by clicking [Trace result] after closing the trace execution screen by clicking [Close]. Tracing will continue even after [Close] is clicked.

Once [Close] is clicked and display the trace execution screen again, click [Trace execution]. Since [Trace status] is hidden at this point, click [Start monitor].

[Note]

Once trace is resumed, trace data up to the previous time will be deleted.
If resuming trace after once executing trace, end trace in the following manner.
• After trigger execution, execute trace after trigger.
• Execute "Stop trace" at [Trace operation].
• CNC power is turned OFF.
• PLC status is turned to "STOP".

(2) Trigger execution

Trigger can be executed at an arbitrary timing. Select "Execute trigger" and click [Execute].

(3) Stopping trace

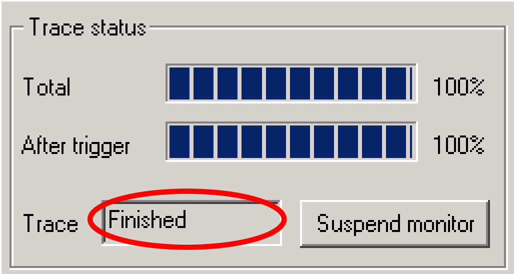
To stop tracing, select "Stop trace" and click [Execute].
To display the trace result before stop, click [Trace result] after pressing [Close].

[Note]

Once "Stop trace" is executed, trace cannot be resumed.
When the trace information before stop is required, save the data in CSV file with [Trace result].
If "Start trace" is executed before saving the data, the data before "Stop trace" will be deleted.

(4) End of trace

When trace after trigger has been completed after trigger execution, "Finished" is displayed on [Trace status], and then trace will be finished. Click [Close] upon completion of tracing and end the "Sampling trace setting wizard execution" screen.



At wizard setting/execution – Trace execution screen – Trace status display at trace "Finished"

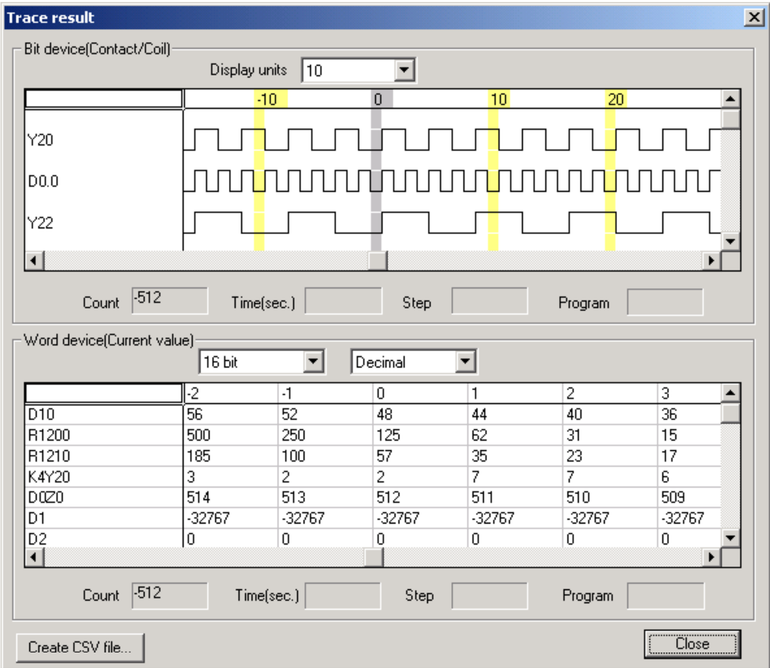
(5) Trace status display

When trace status is being monitored, the button displays [Suspend monitor]. To stop monitoring, click the button. When not monitoring, the button displays [Start monitor]. To start monitoring, click the button.

[Trace] within [Trace status] includes the following four display items.
• Executing: Trace is being executed.
• Suspend: Trace has been stopped.
• Execution failed: Trace is not executed.
• Finished: Trace has been completed.

5.7.4.5 Trace Result

Start the "Trace result" screen below by clicking [Trace result].
Trace result of bit device is chronologically shown in the upper section and word device in the lower section.
Trace result can be saved in the CSV file format. Trace result can be displayed even during trace execution.



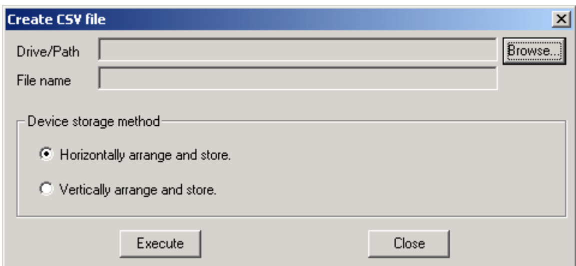
ON/OFF status of bit device is displayed per trace count

Value of word device is displayed per trace count

At wizard setting/execution – Trace result screen

"Count": Counts are displayed. (Trigger execution point is set as start point or 0 point)
"Time(sec.)": Time is displayed. (Head of trace data is set as standard or 0.000 sec.)
If trace interval exceeds 65 seconds, correct time will not be displayed.
"Step","Program": Not displayed.

[Create CSV file]
The following screen is displayed by clicking [Create CSV file].



At wizard setting/execution – Trace result screen – Create CSV file

Save the trace result data following the procedures (1) to (4) below.
(1) Click [Browse] and select the storage destination.
(2) Input the file name.
(3) Select [Device storage method].
(Refer to the Operating Manual for details on [Device storage method].)
(4) Click [Execute].

[Note]

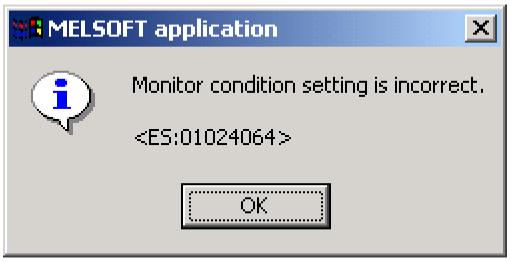
Trace result data is overwritten per each trace execution. Always save the necessary data in CSV file.

5.7.5 Operation at Error

Error will occur under the following conditions. The following dialog will be displayed on the GX Developer screen when parameter check during trace execution is carried out.

Error details

No.	Error definition	Remedies
1	No applicable device for sampling trace is set with device setting.	Set the applicable device for sampling trace with device setting.



Error dialogue

When sampling trace cannot be executed, check the following items.
• Check if CPU type is Q4ACPU.
• Check the trace execution status. (Trace execution command is not possible during tracing.)
• When trace execution method is set to "Execute trace for the settings written in PLC.", check if setting file is saved in CNC.
• Check if devices on the program and that set by trace match.

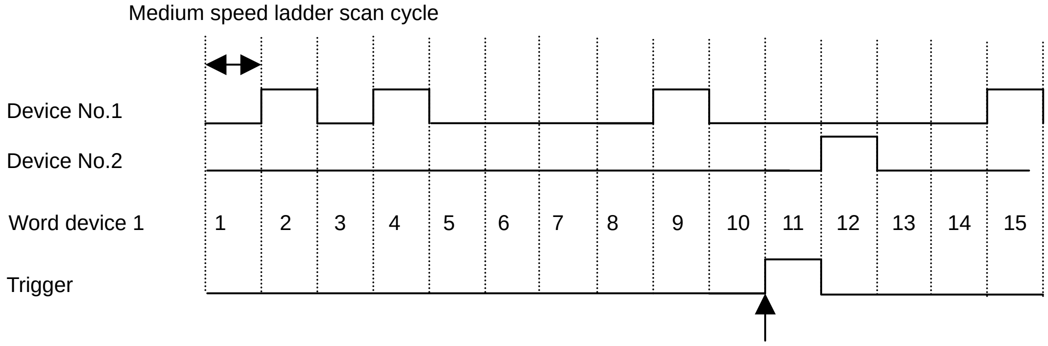
5.7.6 Operation Example

Example 1 Trigger judgment and trace result with device setting

[Setting details]

Trace setting details for example 1

Setting item	Setting value	Setting item	Setting value
Target memory	IC memory card A	Trigger point setting	■ Device Word device Word device 1 = 11 ■ Step 30
File name	MAIN		
Total number	10		
Number after trigger	4	Device setting	■ Bit device Device No.1 Device No.2 ■ Word device Word device 1
Trace additional information	Time, step No.		
Trace point setting	Interval: 10msec		



Trigger execution timing
(Scan when "word device 1=11" has been established.

Ignores step No. setting; only device setting will be valid for trigger point.

Device status when sampling trace is executed at example 1

[Trace result]

Trace result display for example 1

Count	-6	-5	-4	-3	-2	-1	0	1	2	3
Device No.1										
Device No.2										
Word device 1	5	6	7	8	9	10	11	11	12	13
Time	0	0.007	0.014	0.021	0.028	0.035	0.042	0.042	0.049	0.056
Step No	-	-	-	-	-	-	-	-	-	-
File name	-	-	-	-	-	-	-	-	-	-

← The scan in which trigger has been executed is regarded as count "0". Trace data is retrieved when trigger has been established. Thus, when trigger establishment and trace execution happen simultaneously, count "0" and count "1" will be the same data.

← "Interval" trace point setting is ignored. Execute trace per scan.

← Word device is shown as numerical value.

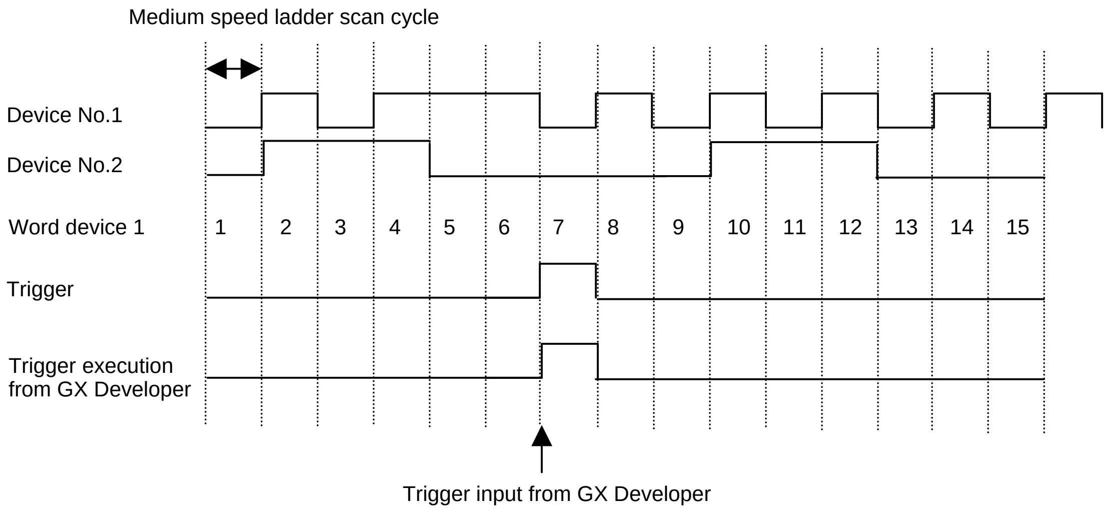
← Time is displayed as trace additional information.

← Nothing is displayed as trace additional information other than time.

Example 2 Trace judgment and trace result with device setting

[Setting details]
Trace setting details for example 2

Setting item	Setting value	Setting item	Setting value
Target memory	IC memory card A	Trigger point setting	■ At the time of STRA instruction execution
File name	MAIN		
Total number	6		
Number after trigger	4	Device setting	■ Bit device Device No.1 Device No.2 ■ Word device Word device 1
Trace additional information	None		
Trace point setting	■ Device Bit device Device No.1 ↑		



Ignores "At the time of STRA instruction execution" and "At the time of trigger operation from GX Developer" will be valid for trigger point.

Device status when sampling trace is executed at example 1

[Trace result]
Trace result display for example 2

Count	-2	-1	0	1	2	3
Device No.1						
Device No.2						
Word device 1	2	4	7	8	10	12

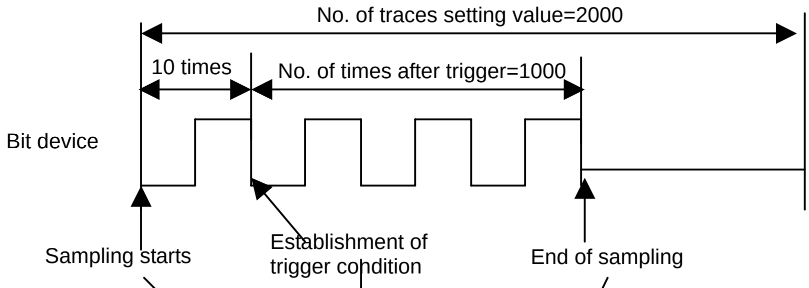
← The scan in which trigger has been executed is regarded as count "0".

← Trace is executed only for the scan where device No.1 turns ON from OFF.
Note that, however, the device data for the scan when trigger has been executed will be saved.

5.7.7 Precautions

(1) Sampling can be executed for the other stations on network or with serial communication connection. Note that, however, sampling is not possible from multiple areas at the same time.
(2) Since trace conditions and trace data that have been registered in CNC controller are saved in DRAM area, when the CNC power is turned OFF, the data will be deleted.
(3) Sampling trace is executed by connecting CNC controller and GX Developer.
(4) Trace data within CNC is deleted during trace execution. So, please be aware that the data up to previous time will be deleted.
(5) When trace is ended before reaching the set No. of traces, such as when trigger occurs as soon as trace is started, the shortened data will not be displayed.

<Example>
Trigger occurred at 10th trace when No. of traces is set to 2000 and No. of traces after trigger is set to 1000.



Trace result

Count	-10	•••	0	1	•••	1000
Device No.1						
Word device 1	9	10	11	12	13	14

(6) Descriptions about sampling trace in this manual are given under the assumption of GX Developer Ver.8 specifications.
Sampling trace setting wizard is not available for the version prior to GX Developer Ver.8.
Refer to the precautions for each setting item described in this manual and perform settings respectively.

(7) When the trace setting that is invalid with this CNC is performed, the setting is forcibly corrected to the valid one within CNC controller. Refer to "5.7.4 Sampling Trace Operation Screen" for trace settings that can be used.

(8) Status of SM800 to SM805 cannot be checked on CNC controller screen. Check the sampling trace status with GX Developer.

6. PLC Message Development

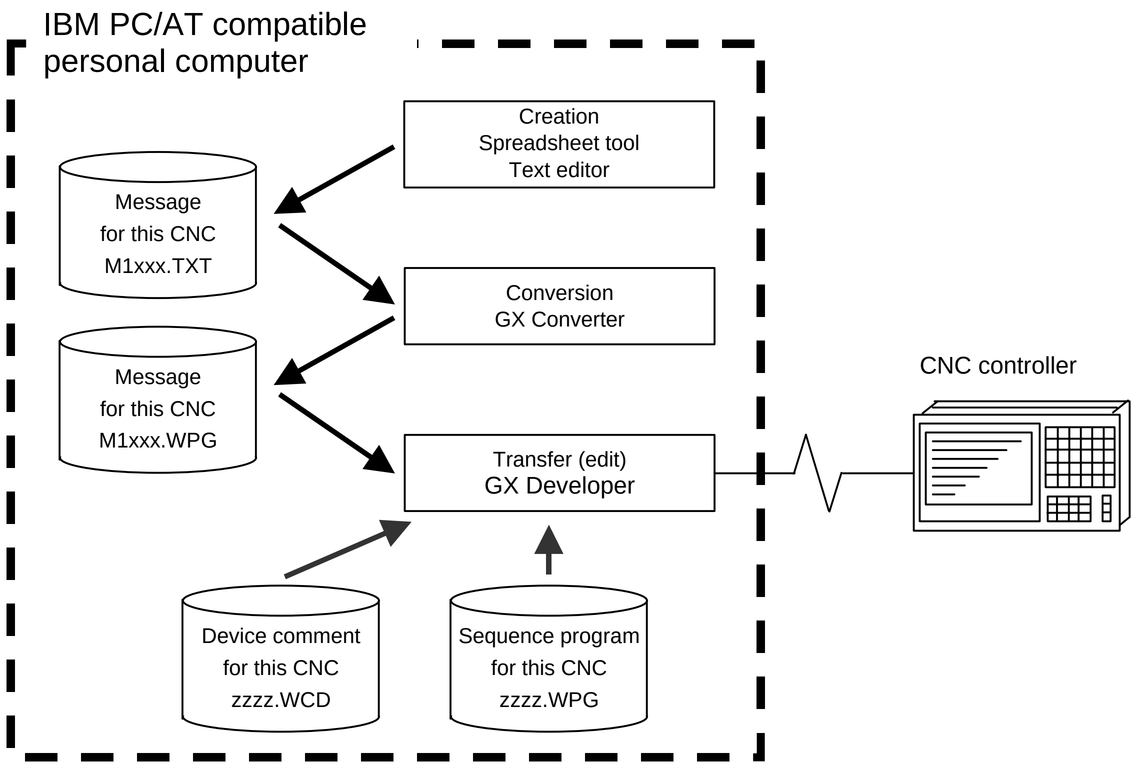
This chapter describes a procedure for developing PLC-related data such as alarm messages, operator messages, and PLC switches.

6.1 Development Procedure

There are the following two methods as a general development procedure of message data.
Refer to the Operating Manual for details on newly creating a PLC message.

(1) Making conversion into GX Developer data using a general text editor or spreadsheet tool and data conversion package.
(When there is a large volume of message data and you want to control them with a commercially available tool, for example)

(2) Entering messages directly from GX Developer
(When there is a small volume of message data or when addition or correction is to be made, for example)



6.1.1 Using a General Text Editor

(1) Creation

The message data is described using a general text editor. The description method and format will be described later.

(2) Conversion

The conversion from text data to GX Developer data is carried out using the "GX Converter (data conversion software package)".

(3) Transfer

With the GX Developer, the message data is handled as a sequence program interlinear comment, and can also be edited.
The message data is transferred to the CNC controller using the GX Developer, in the same manner as the sequence program.

6.1.2 Entering Messages Directly from GX Developer

(1) Creation

The message data is described directly from GX Developer. The message data is handled as a sequence program interlinear comment by GX Developer. The description method and format will be described later.

(2) Transfer

The message data is transferred from GX Developer to the CNC controller in the same manner as the sequence program.

6.2 Message Data Description Method

The message data can be described as text data by a general text editor and also by commercially available spreadsheet software in addition to the direct input with GX Developer.

6.2.1 Description Format

The message data is described using the following description format.
The description format cannot be abbreviated. Comma(,) and [CR] must be described, even the message character string is blank.

Message classification	Description format
Alarm message	;A, index No., data register No., message character string [CR]
Operator message	;O, index No., data register No., message character string [CR]
PLC switch	;P, switch No., message character string [CR]
Comment message	;M, device, device No., message character string [CR]

Message classification code	: A one-byte alphabetic character expressing each message classification
Index No.	: One-byte number (0 to No. of messages in the setting area - 1)
Switch No.	: One-byte number (0 to No. of messages in the setting area - 1)
Data register No.	: One-byte number
Device	: One-byte number (1 or 2)
Device No.	: One-byte number (0 to 10)
Message character string	: One-byte alphanumeric character, No. of characters in the setting area message length. Semicolons, commas, spaces and tabs can also be used. Note that the tab at the head of the message character string is ignored.
Semicolon(;)	: Message data identification code
Comma(,)	: Separator between each description (a comma only is used to leave a message character string blank)
[CR]	: Line feed code, (CR/LF) or (LF).
Back slash (\)	: Continue to next line. If the end of the description is a back slash, continue to head character on next line.

[Note] Setting area

The setting area (each message's message length and message quantity declaration) used with the old models is no longer required. The storage area in the CNC controller is always stored in the most efficient state.
Even if the old setting area remains in the data, it will be ignored and will not result in an error.

6.2.2 Description Method

The message data is described as text data by the following description format.

```
;#M700 Ladder ver1.0 ’04.03.03                --- Comment
NOPLF
 ; A,0,0,Emergency stop
;A,11,1,Spindle alarm
        ;
        ;
NOPLF                                         --- Message area
        ;                                         (Alarm messages)
        ;
NOPLF
        ;
        ;
NOPLF                                         --- Page break code
;0,1,9000,MELDAS 600 LADDER Ver1.0
;0,20,9000,BND-400W000-A0                     --- Message area
        ;                                         (Operator messages)
        ;
NOPLF
;P,1,Program restart
;P,2,Automatic power OFF                      --- Message area
        ;                                         (PLC switches)
        ;
NOPLF
;M,1,0,[ Spindle ]
;M,1,0,[ Standby 1 ]                          --- Message area
        ;                                         (Comment)
        ;
END                                           --- End code
```

(1) Comment

A line with a semicolon (;) followed by a sharp (#) at the head of the line are interpreted as comments.

; # comment character string [CR]

(2) Message area

Collect similar messages in a group and describe them. There is no description order in the respective messages, but the latter description is validated if there are descriptions with the same factors (index No., etc.).

(3) Page break code

A page break code is described at one or more places approx. every 15 lines in the setting area and message area. The message data may skip if there is no page break code.

(4) End code

An end code is described at the end of the description. Description after the end code are ignored.
An error will occur if there is no end code.

(5) Other descriptions

A description that does not have a semicolon (;) at the head will result in an error. A description with a format other than the above will also result in an error.

6.2.3 Precautions

No. of characters, quantity limitations, handling of information other than settings, handling of information other than format are described below.

(1) Message data maximum value

The maximum value of the described messages is shown below.

Message classification	Max. message length	Max. No. of messages
Alarm messages	46 byte	1024
Operator messages	60 byte	512
PLC switches	14 byte	32
Comments	60 byte	100

When maximum number of characters is exceeded: An error will not occur, but the excessive characters will not be displayed.
When maximum number of messages is exceeded: An error will occur when writing.

[Note] Two-byte data in the message character string is handled as two characters.

(2) When message cannot be described on one line

With GX Developer, up to 64 characters can be input on one line as an interlinear statement. However, information other than the message character string (such as message classification code, index No., data register No., etc.), so a 60-character message character string cannot be described on one line.
A 60-character description is possible by splitting the message into two lines using a back slash (\) at the end of the line.

```
;0,1,1000,Operator Message....0123456789012345678901234567890123456789
```

Section exceeding 64 characters, and which cannot be described

```
;0,1,1000,\
;Operator Message....0123456789012345678901234567890123456789
```

Description split into two lines using back slash

[Note] If the message is split into two lines, an asterisk (*) cannot be used as the head character of the second line. Use another character, such as a space, in this case.

6.3 Converting Data into GX Developer Format

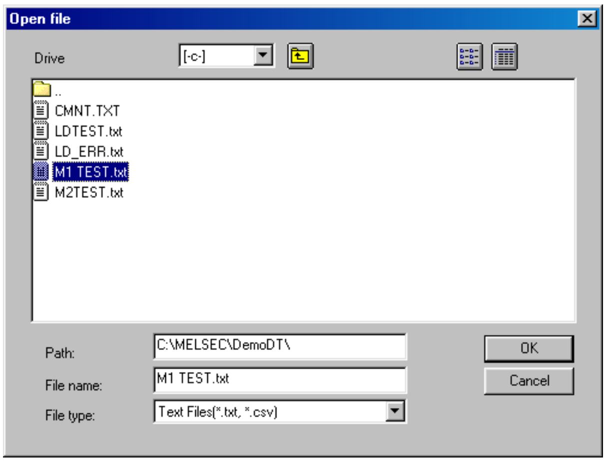
Convert the message data, which was described using a text editor or like, into GX Developer data in the following method. Use "GX Converter (data conversion software package)" for conversion. GX Converter can be started from the GX Developer menu.

6.3.1 Starting GX Converter and Specifying the File to be Converted

Perform the following operation from GX Developer to start GX Converter (read).

[Project] → [Import file] → [Import from TEXT ,CSV format file]

On the following screen, specify the file to be converted (M1TEST.TXT) and click [OK].

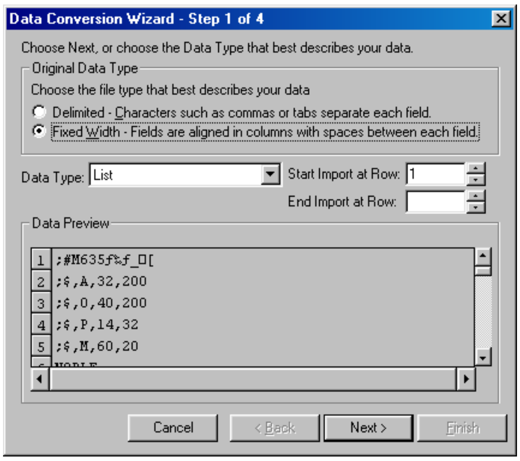


6.3.2 Conversion Format Setting

Set the conversion format on the following data conversion wizard screen.

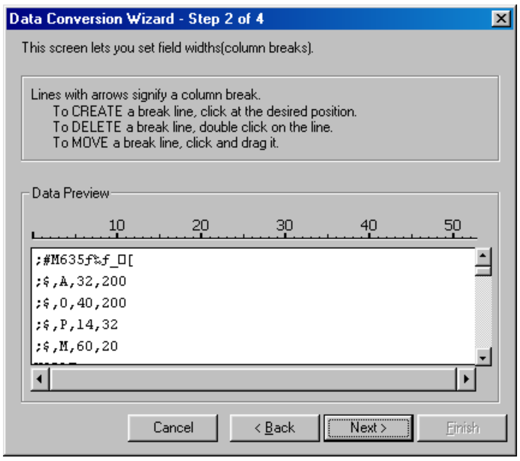
(1) Data conversion wizard 1/4

Choose [Original Data Type]-[Fixed Width] and [Data Type]-[List], and click [Next>].



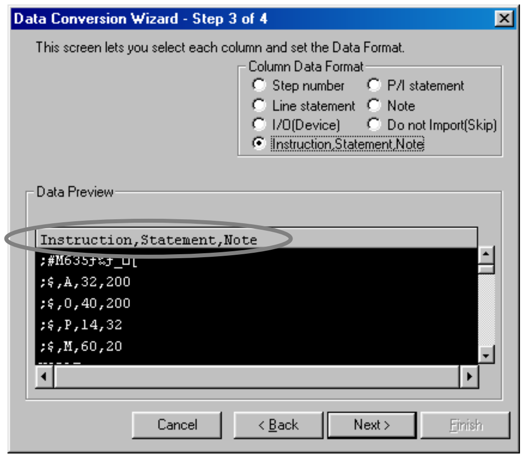
(2) Data conversion wizard 2/4

Just click [Next>].



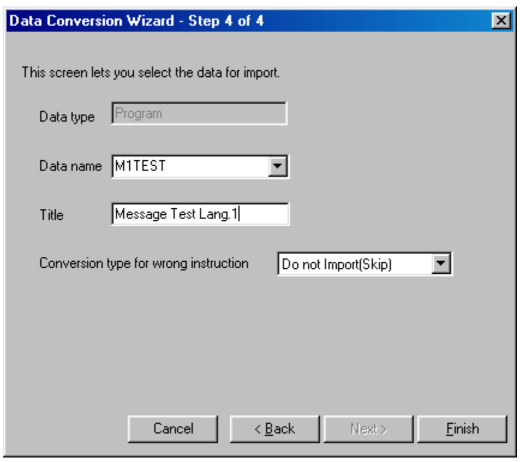
(3) Data conversion wizard 3/4

Choose to highlight the instruction column part in the [Data Preview] list and choose [Column Data Format]-[Instruction, Statement, Note]. Click [Next>].



(4) Data conversion wizard 4/4

Set the program name used on GX Developer in [Data name] and a data annotation in [Title], and click [Finish]. The setting is complete when the completed dialog appears. Click [OK].



6.4 Entering/Editing Data using GX Developer

The message data in GX Developer are handled as the "integrated type interlinear statements" of a sequence program. "Integrated type interlinear statements" are interlinear comments provided to assist the understanding of the sequence program, and those transferred to the controller together with the sequence program are called the "integrated type".
"Interlinear statements" can be displayed and edited using [Ladder] or [Instruction list].

6.4.1 Interlinear Statement Display using Circuit Display

(1) Display of project data list

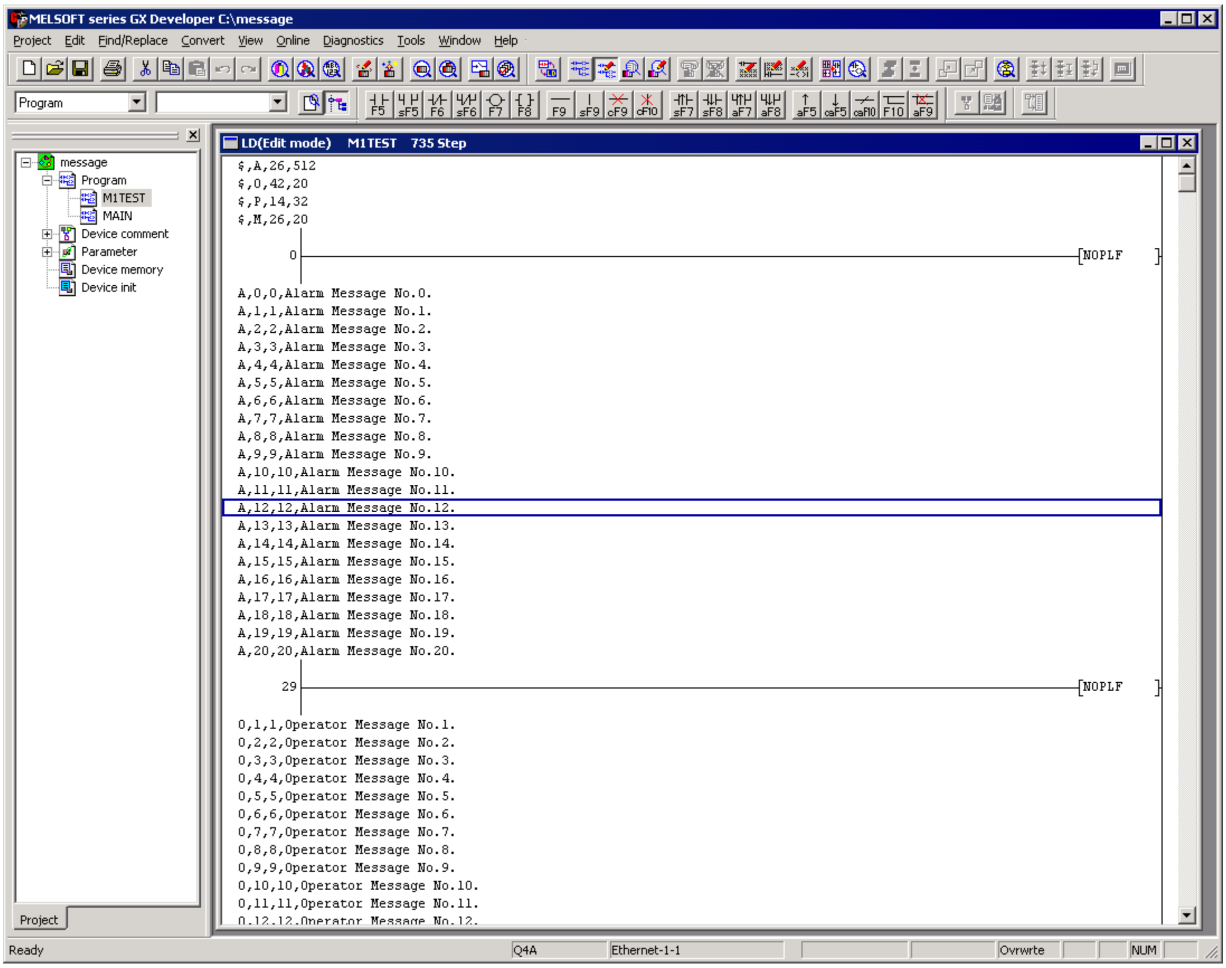
Perform the following operation to display the "Project data list" window and double-click the file name to display the edit screen. First, the normal ladder screen appears.

[View] → [Project data list], then double-click [File name you want to display].

(2) Display of message data

Perform the following operation to display the message data that are integrated type interlinear statements.

[View] → [Statement]



6.4.2 Interlinear Statement Display using List Display

(1) Display of project data list

Perform the following operation to display the "Project data list" window and double-click the file name to display the edit screen. First, the normal ladder screen appears.

[View] → [Project data list], then double-click [File name you want to display].

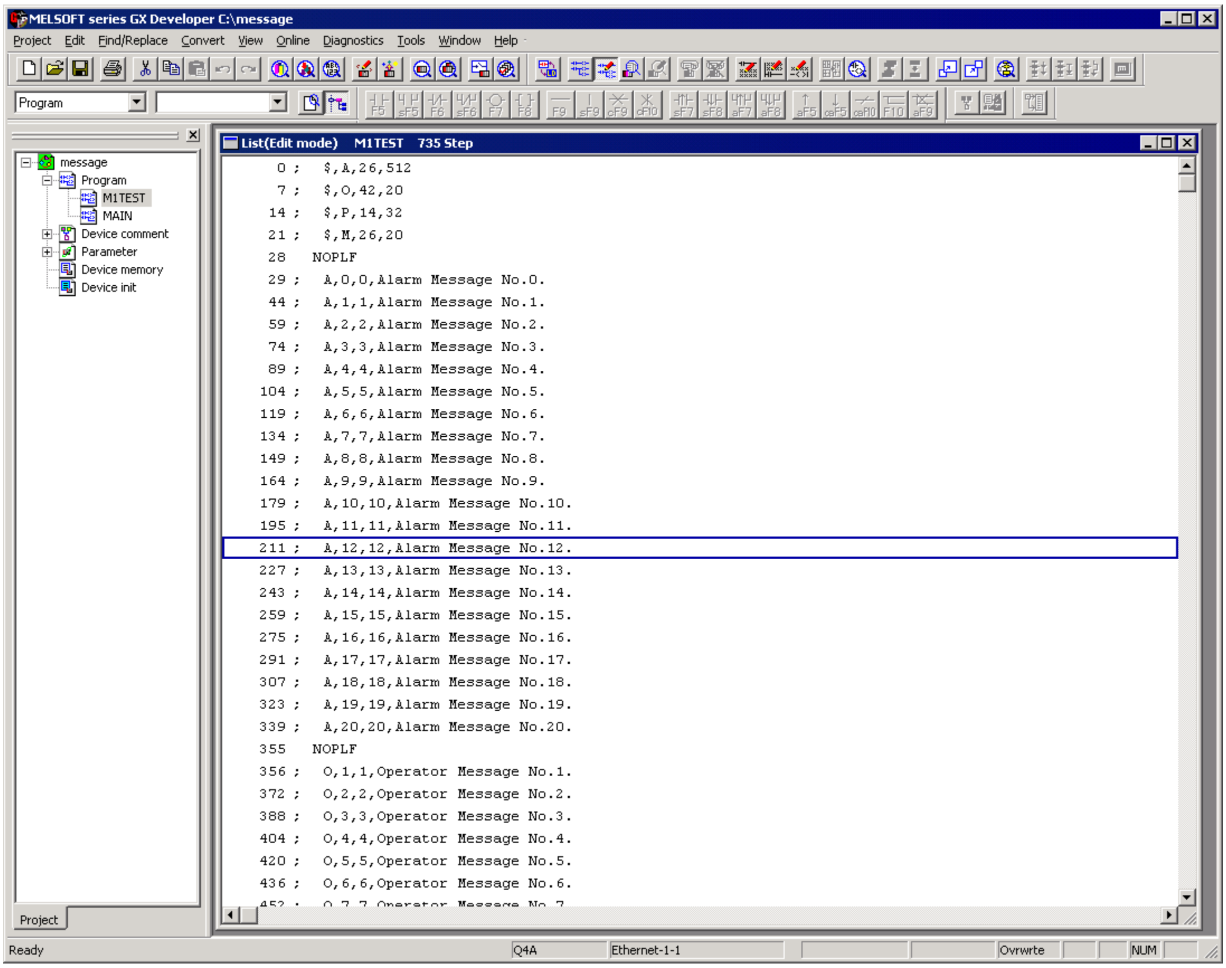
(2) Display of list data

Perform the following operation to display the list data. The list display also shows the message data that are integrated type interlinear statements.

[View] → [Instruction list]

Perform the following operation to return to the circuit display.

[View] → [Ladder]



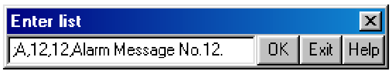
6.4.3 Editing of Integrated Type Interlinear Statements

(1) Circuit display

On the circuit display screen that shows the integrated type interlinear statements, double-clicking the interlinear statement you want to edit displays the following dialog. Perform editing operation on the dialog and click [OK] or press [Enter].

(2) List display

On the list display screen, double-clicking the interlinear statement you want to edit displays the following dialog. Perform editing operation on the dialog and click [OK] or press [Enter].

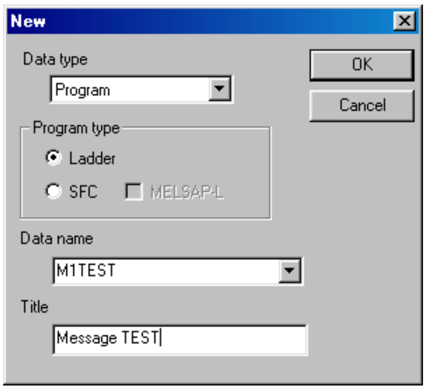


(3) Entering new message data

- **Displaying new edit screen**

  Perform the following operation to display the [New] dialog, and set the [Data name] and [Title]. After setting, click [OK].

  [Project] → [Edit Data] → [New]



- **Changing to list display mode**

  Perform the following operation to display the list data.

  [View] → [Instruction list]

- **Entering message data**

  Press "Enter" on the "END" line, enter data as in the section "(2) List display", and then press "Enter" on the next line and enter message data.

6.5 Writing to the CNC Controller

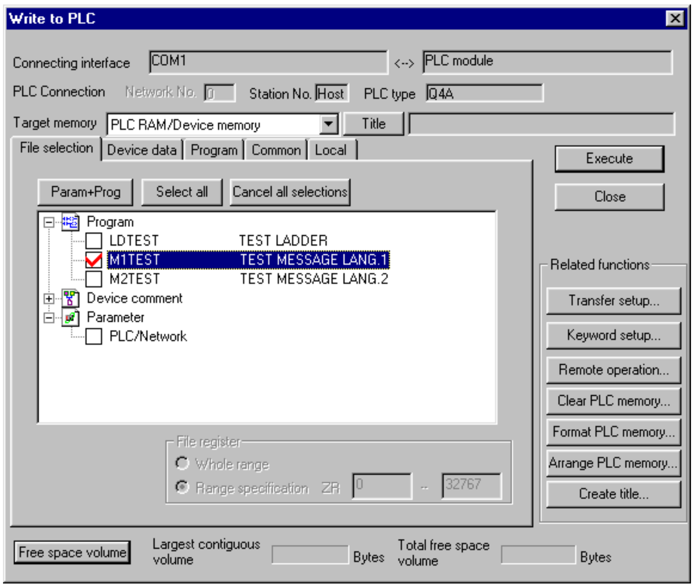
The following shows the method of transferring a message from the GX Developer to the CNC controller. The transfer method is the same as the sequence program transfer method. Sequence program and message data are distinguished by their file names only.

6.5.1 Operation Procedure

Perform the following operation to display the "Write to PLC" screen, and select the file to be written.

[Online] → [Write to PLC]

The following example transfers a message first language file "M1TEST.GPG".



[Note] Restrictions for writing message data
Only one message data file can be stored in the same language. If message data in the same language, but having a different file name, is overwritten, a dialog confirming overwrite will occur.

6.5.2 Writing Operation

When executing message write from GX developer, the data will be checked by the CNC controller.
A conversion error will occur if there is a description exceeding the specifications. An error mark will be added to the description section in which the error occurred, and the data will be transferred to the last step.

;A,0,Alarm Message No.0. ← Message occuring error
↓
;E,0,Alarm Message No.0. ← State with error mark added
↖ Error mark (Alphabetic character E)

[Note] The message data file in which an error occurred cannot be displayed.

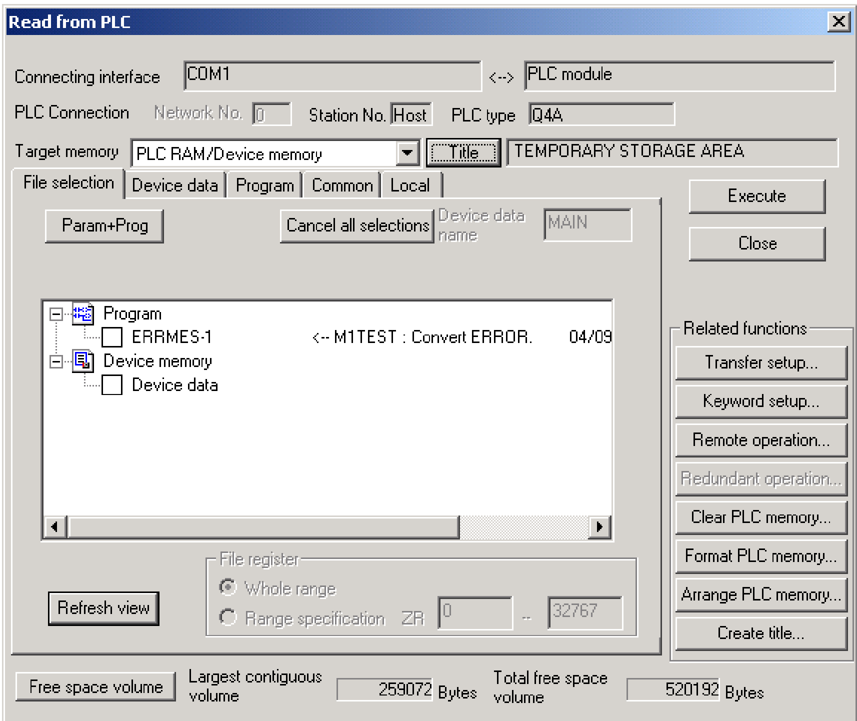
6.5.3 Operation at Write Error

The following dialog appears on the GX Developer screen when a conversion error occurs.

If the message file resulting in a conversion error is selected with the [File Selection] tab on the [Read from PLC] screen, the file name and title will change and be displayed as shown below.
If this message file is read out to the GX Developer, it will be stored under the file name "ERRMES-0".

ERRMES-0	<-- M1TEST : Convert ERROR.
(1)	(2)

(1) File name at error
(2) Title statement including transferred original file name



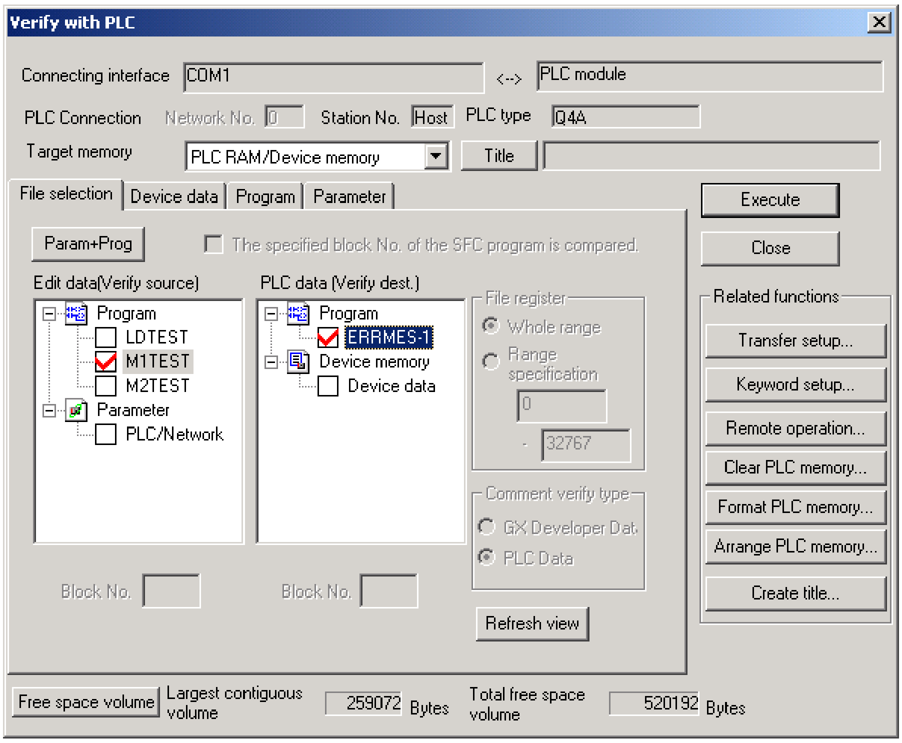
⚠ CAUTION

⚠ Do not read out a message file resulting in a conversion error to the GX Developer and use it. It may contain unexpected data, and result in incorrect operations.

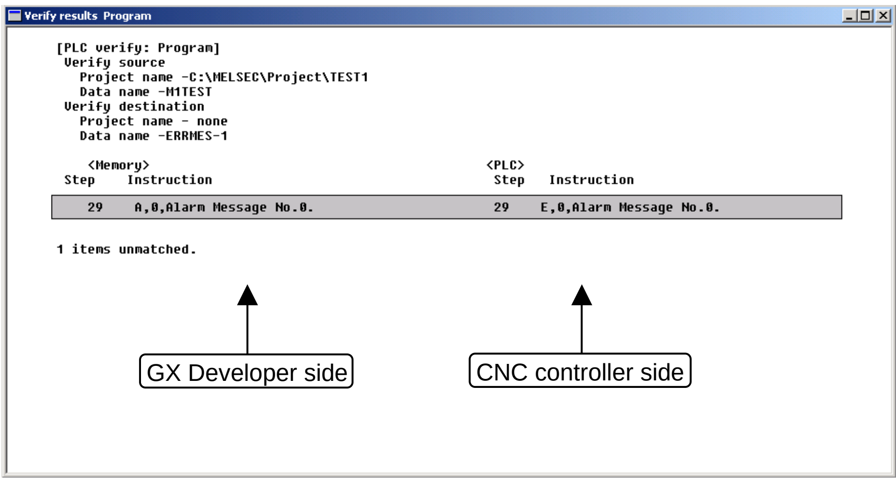
6.5.4 How to Confirm the Error Position

The error position can be confirmed with the PLC verification function. Refer to section "5.4 Verifying the Sequence Program" for details on the PLC verification function.

Verify source : Select the message file transferred to the GX Developer
Verify dest. : Select the message file resulting in an error "ERRMES-0" on the CNC controller



When PLC verification is executed, the mismatching details will appear as shown in the following example. The section with the error mark in the CNC controller side is the message description section with the check error. Double-click the mismatch to display and to edit the corresponding part of the GX Developer side. <Memory> indicates the GX Developer side, and <PLC> the CNC controller side.



6.6 Reading and Verifying from the CNC Controller

The following shows the method of reading and verifying a message from the CNC controller to the GX Developer. The method of reading and verifying is the same as that of sequence program. Sequence program and message data are distinguished by their file names only.

6.6.1 Menu Selection/Screen Operation

Refer to the following sections for operation methods.

For read : "5.3 Reading the Sequence Program from the CNC Controller"
For verification : "5.4 Verifying the Sequence Programs"

7. Device Comment Creation

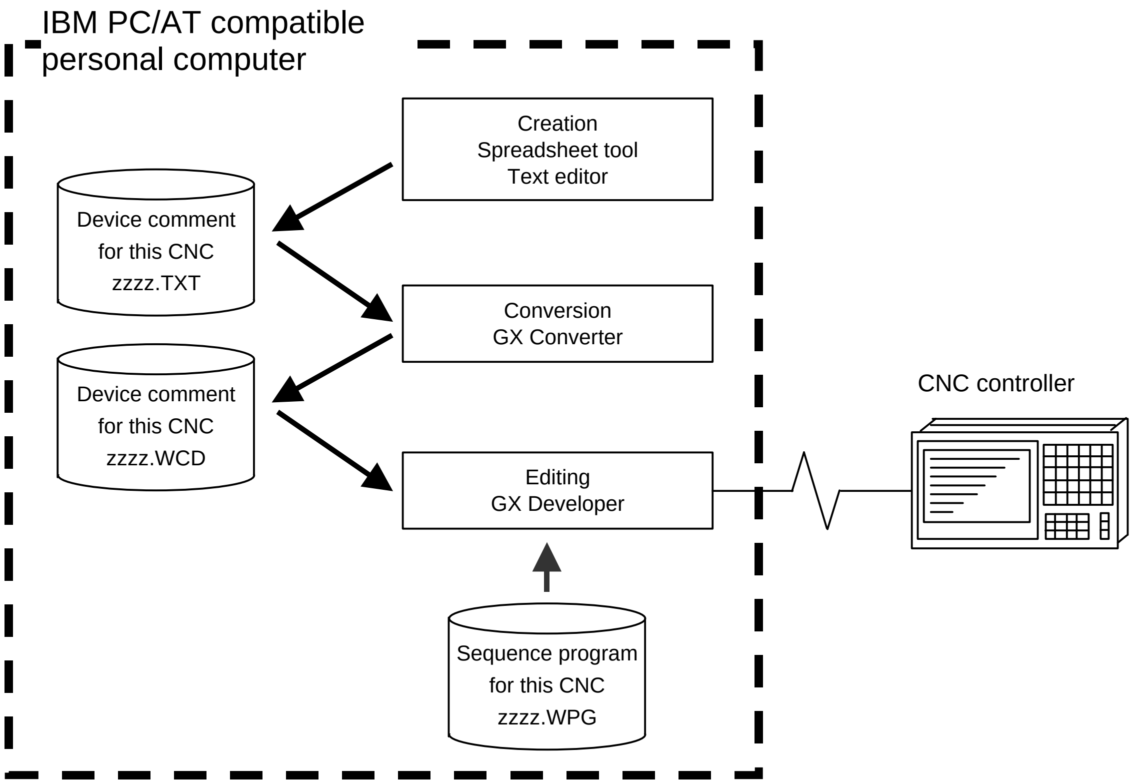
There are no MITSUBISHI CNC-specific operations for device comments. Therefore, refer to the Operating Manual for the development method. This section describes the device comment development procedure outline and the development method using a general-purpose tool.

7.1 Development Procedure

There are the following two methods as a general development procedure of device comments.

(1) Indirect entry

In this method, device comments are converted into GX Developer data using a general text editor or spreadsheet tool and data conversion package. Use this method when you want to divert the device comments of the old model or when a device comment volume is large and you want to control them with a commercially available tool, for example.



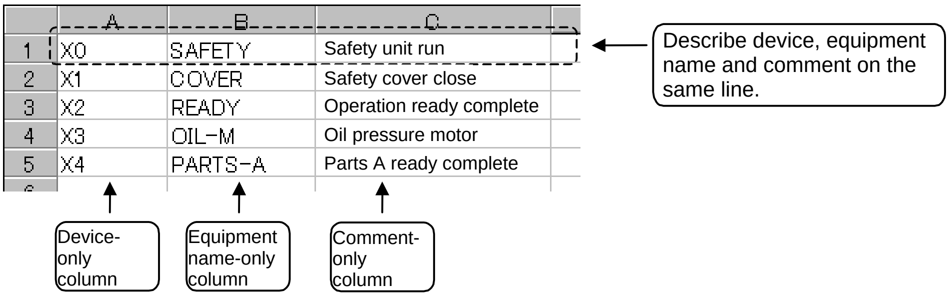
(2) Direct entry

In this method, device comments are entered directly from GX Developer. Use this method when a device comment volume is small or when addition or correction is to be made, for example.
There are the following three methods for direct entry from GX Developer. Refer to the Operating Manual for details.

- Creating comments on the device comment edit screen
- Creating device comments after circuit creation during sequence program creation
- Making addition/correction to device comments in the created sequence program

7.2 Description Method for Indirect Entry

The following explains the description method for creating device comments using a spreadsheet tool or like. The following example describes device comments using a spreadsheet tool.



Column data format	Explanation
Device	(1) Describe a device. • Conversion cannot be made if a device has not been described. Always describe a device. (2) A device is a required item. Describe it in one-byte code.
Comment	(1) Describe a comment. • It is not registered if the device part on the same row is blank or the device is illegal. (2) You can describe a comment of up to 32 characters.
Equipment name	(1) Describe an equipment name. • It is not registered if the device part on the same row is blank or the device is illegal. (2) You can describe an equipment name of up to 8 characters.

[Note] Describe data in any of the following combinations.
(1) Device, equipment name, comment
(2) Device, comment
(3) Device, equipment name

Save the above data in the CSV format. The following example shows the above data saved in the CSV format.

```
X0, SAFETY, Safety unit run
X1, COVER, Safety cover close
X2, READY, Operation ready complete
X3, OIL-M, Oil pressure motor
X4, PARTS-A, Parts A ready complete
```

7.3 Converting Comment Data into GX Developer Data

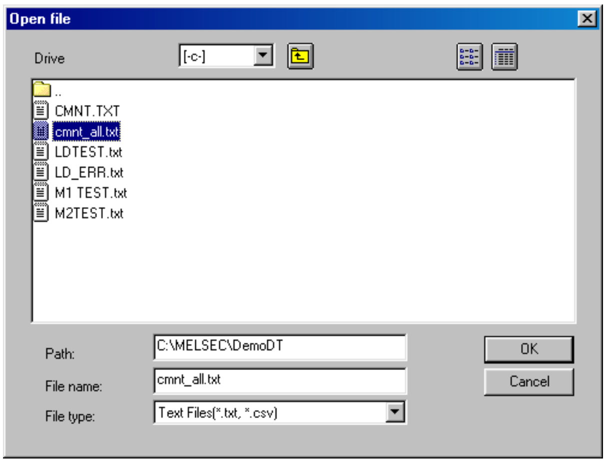
Convert the comment data (CSV format), which was created using a spreadsheet tool or like, into GX Developer data in the following method. Use "GX Converter (data conversion software package)" for conversion. GX Converter can be started from the GX Developer menu.

7.3.1 Starting GX Converter and Specifying the File to be Converted

Perform the following operation from GX Developer to start GX Converter (read).

[Project] → [Import file] → [Import from TEXT ,CSV format file]

On the following screen, specify the file to be converted (cmnt_all.txt) and click [OK].

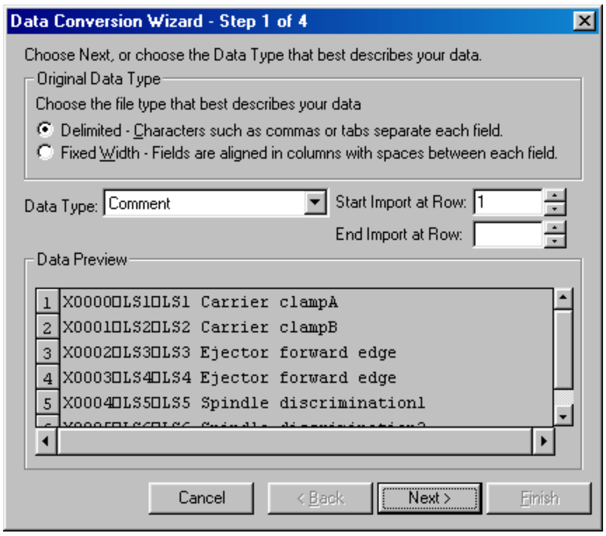


7.3.2 Conversion Format Setting

Set the conversion format on the following data conversion wizard screen.

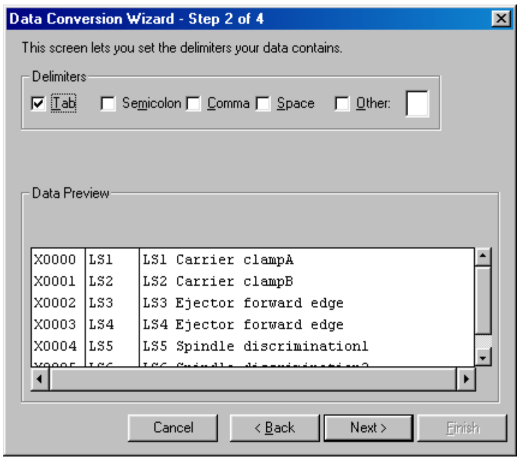
(1) Data conversion wizard 1/4

Choose [Original Data Type]-[Delimited] and [Data Type]-[Comment], and click [Next>].



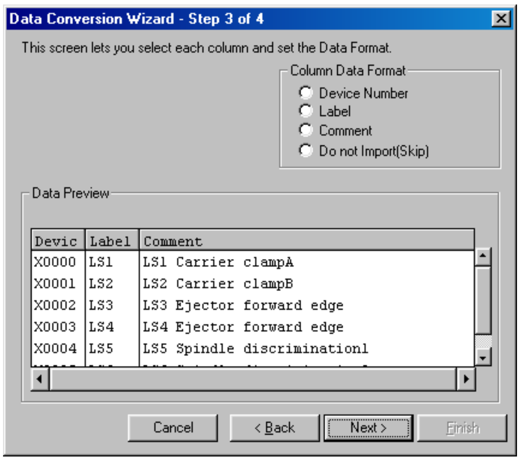
(2) Data conversion wizard 2/4

Choose [Delimiters]-[Tab] and click [Next>].



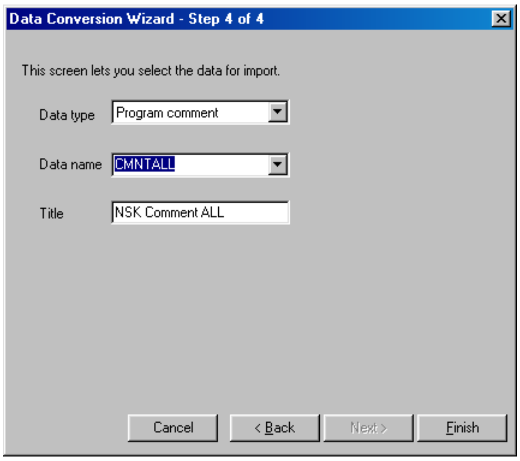
(3) Data conversion wizard 3/4

Make sure that the column parts in the [Data Preview] list are in order of [Device Number], [Label] and [Comment], and click [Next>].



(4) Data conversion wizard 4/4

Choose [Data type]-[Common comment] or [Program comment], set the comment file name used on GX Developer in [Data name] and a comment annotation in [Title], and click [Finish].

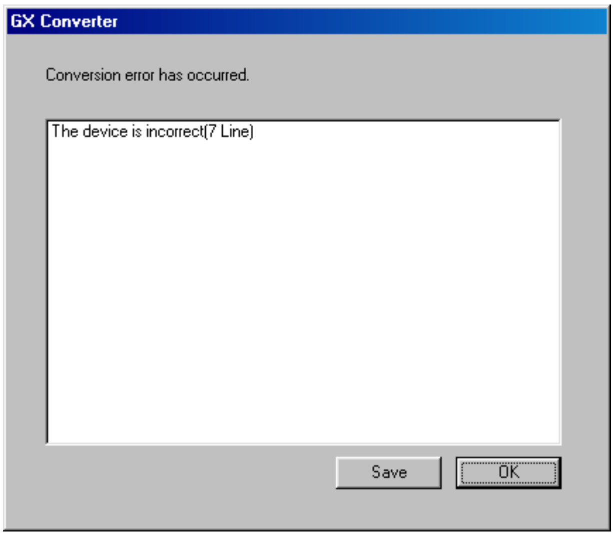


(5) Completion

The setting is complete when the following dialog appears. Click [OK].

(6) Error status

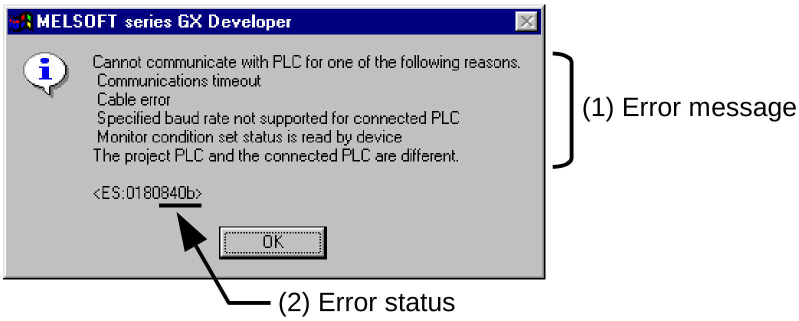
If an error occurred during conversion, its status and the line where it occurred are displayed.



8. Troubleshooting

8.1 List of Errors During GX Developer Online Operations

If an error has occurred in GX Developer, the following dialog appears. The error message and error status are displayed in the dialog.
Note that the display error messages are primarily for the MELSEC sequencer. Therefore, they are not necessarily accurate for M700 CNC.
Error status is shown in the last 4 digits of the number displayed.



⚠ CAUTION

⚠ When an error occurred at GX Developer On-line function, the error message may not explain exactly the state in the CNC side.
Always refer to the error list.

The following table indicates the causes and remedies of the errors that can occur during online operation with the CNC. For other errors, refer to the GX Developer Operating Manual.

Status	Message	Cause	Remedy
4002	The executed function is not supported. Please check the manual and other documentation.	An operation not supported by the specifications was attempted.	Check the operation procedures.
4005	Writing the data exceeding the size of sequencer was attempted. Execute again within the range of sequencer size.	Maximum number of steps that can be executed with NC has been exceeded.	Check the size of execution area. (Refer to the section 5.2.4.) Reduce the number of steps for the relevant sequence program to be executed with reference that value.
4010	Cannot write because the PLC is executing a RUN command. Stop the PLC, then execute again.	The PLC of the NC is running.	After stopping the PLC of the NC, start execution again.
4021	The applicable drive is not ready. Check the applicable drive, then execute again.	The specified target memory does not exist or is not in a usable status.	Change the target memory.
4024	Application has turned unstable. Restart	The format does not allow the keyword registration.	Reformat the PLC data storage area.
4029	Insufficient file capacity. Execute again after deleting unnecessary files.	An attempt was made to write a file that exceeds the storage capacity.	Examine the file structure so that the data falls within the limited capacity.
402b	The file cannot be accessed. Carry out formatting, then execute again.	An attempt was made to write the same type of file.	After deleting the same type of file from the NC side, start execution again.
4031	The specified device No. exceeds the permissible range. Specify a device No. that is within the range set in the parameter.	The access request given is outside the accessible device range.	Check the number range of each device.

Status	Message	Cause	Remedy
4052	The file is write protected. Change the file attributes to enable writing to the file.	The specified target memory is a write-disabled device (F-ROM).	Specify "internal RAM" as the target memory.
4065	A mismatch occurred between the PLC and peripheral parameters Match the parameters between the PLC and peripherals.	There is a problem in the device setting value.	Set the number of device points. (Refer to section 4.6.)
4070	The program before correction differs from the registered program.	A ladder instruction outside the specification is included.	Perform verification to identify the instruction that is the cause of the problem.
4080	Incorrect abnormal.	When executing "Read from PLC" or "Verify PLC" function: Data not included in the specifications was found in the designated file.	The sequence program or message data in the CNC controller may be damaged. Delete the corresponding file and start again, or initialize the PLC data storage area. If the problem cannot be resolved, contact the Mitsubishi System Department.
		When executing "Write to PLC": Multiple END instructions were found in the designated sequence program file.	Edit the sequence program in the list mode to delete END instructions except only one at the last line.
8008	An unusable port or IP address was specified. Execute again after checking the port/IP address setting.	The Ethernet setting on the GX Developer is incorrect.	Check the Ethernet setting, or check the cable connection.
8201	Cannot communicate with the PLC. Execute again after checking the connections with the PLC.	The serial communication cable is faulty. • Not connected • DTR signal off	Check the serial port setting and cable connection.
8301	Cannot communicate with the PLC. Execute again after checking the connections with the PLC.	There was no response from the NC connected with Ethernet.	Check the following: • CNC status • Cable connection • Ethernet address setting
840b	Cannot communicate with PLC for one of the following reasons. Communications timeout Cable error Specified baud rate not supported for connected PLC Monitor condition set status is read by device The project PLC and the connected PLC are different.	There is no response from the NC. • The CNC has not started properly. • The connection channel of the CNC side serial port is different. • The serial cable outside the specifications is used for signal connection. • An incorrect Ethernet address is set on the NC side	Check the following. • CNC side status • Cable connection • Bit selection: GPP communication valid • Set the Ethernet address

[Note] Read "PLC" in the message as "CNC controller".

8.2 Confirmation of PLC Alarms on CNC Controller Side

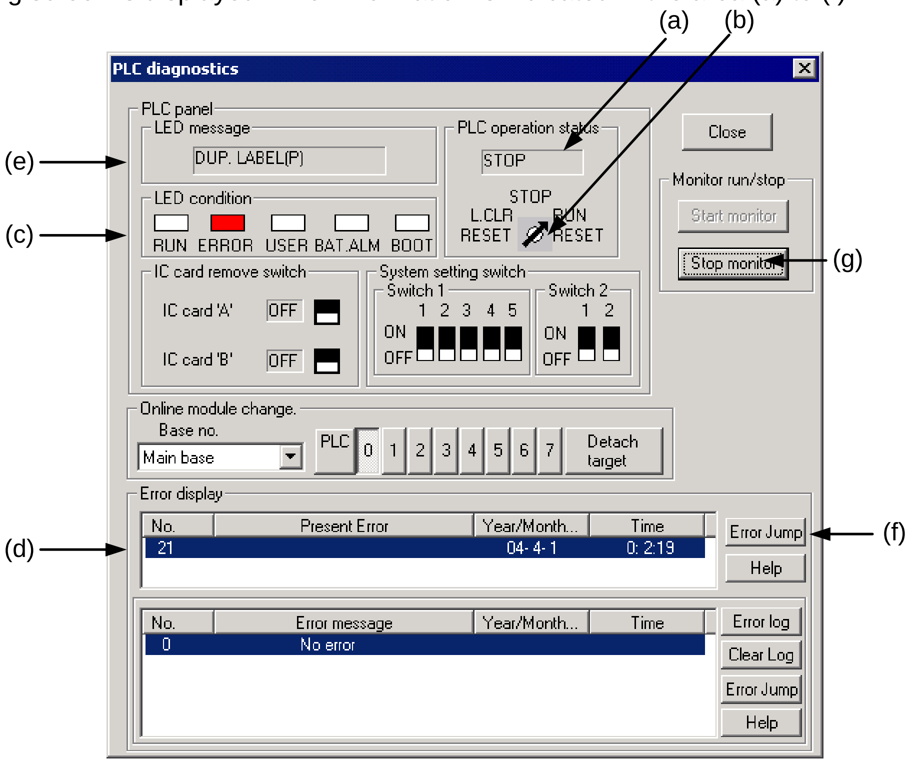
When an error occurs during starting or executing a sequence program, user PLC alarm status will be entered. There are three methods of confirming alarm details.

• Method by using the "ALARM" screen on the CNC controller setting display unit.
• Method by using the "PLC DIAGNOSIS" screen of the internal PLC edit function.
• Method by using the GX Developer "PLC diagnostics" window.

In the GX Developer "PLC diagnostics" window, the same kind of information as displayed in the "ALARM" screen of the CNC controller setting display unit is displayed in a simple manner. In addition, the ladder in which error has occurred is displayed, as well.

8.2.1 Operating Procedures

With the GX Developer, follow the procedures given below to startup the "PLC diagnostics" window.
[Diagnostics] → [PLC diagnostics]
The following screen is displayed. Error information is indicated in the area (a) to (f).



(a) PLC operation status display: RUN/STOP status of PLC is displayed.

(b) Rotary switch status display: The status of rotary switch "CS2" of the control unit is displayed.

(c) Error status display: "ERROR" section flickers in red when an error has occurred.

(d) Error information display: Error No. and error occurrence time are displayed.
- If there is no error, "No error" is displayed.
- Error No. is the upper 2 digits of the CNC side alarm sub status1.
- Further detail is available upon double-click. (Described later)

(e) Error message display: A brief message corresponding with the error No. is displayed.
- A character string with up to 16 alphabetical characters that shows the alarm details.

(f) Error jump button: Ladder in which error has occurred is displayed. (Described later)

(g) Error display, start monitor and stop monitor button: Monitoring error information is started/stopped.

[Note] All the buttons other than (a) to (g) and [CLOSE] do not have any significant meaning. Therefore, do not touch them.

8.2.2 Details of Each Display

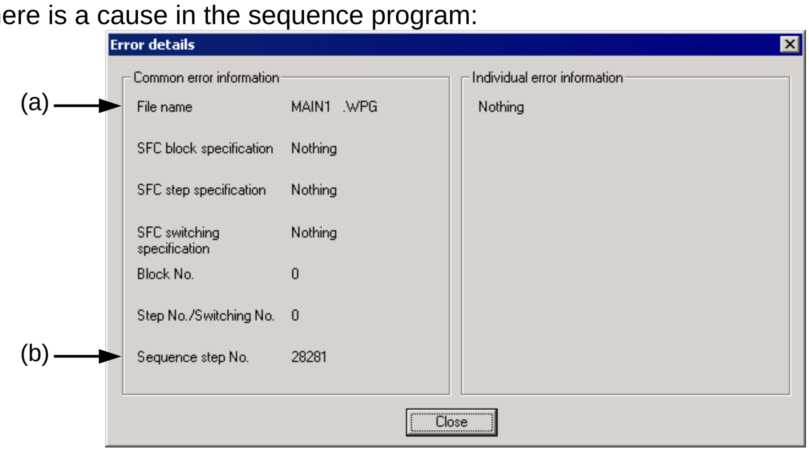
(1) Rotary switch status display (b)
The status of control unit rotary switch "CS2" is displayed.
"STOP" is displayed when "CS2" is set to "1", "RUN" is displayed for all the other settings.
"CS2" is usually set to "0". By setting to "1", PLC status can be compulsorily changed to STOP.

(2) Error status (c), error information (d), error message display (e)
The error occurred last after PLC RUN is displayed here. The display will not be changed until a new error has occurred.
Error display will be cleared only when RUN is executed from PLC STOP.

8.2.3 Detailed Error Information Display

Detailed error information can be displayed in the "PLC diagnostics window" following the operating procedures below.
Double-click "(d) error display list box" in the "PLC diagnostics window".
The "Error details" screen (see below) is displayed.

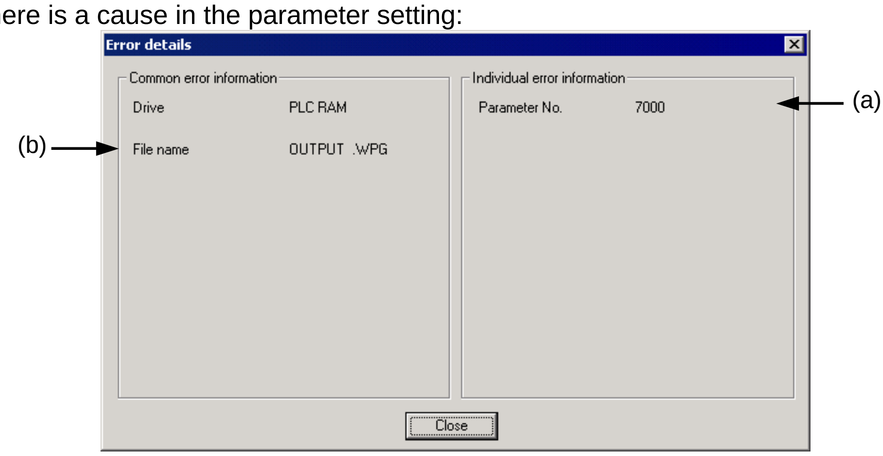
(1) When there is a cause in the sequence program:



(a) File name: The name of sequence program file in which error has occurred is displayed.

(b) Sequence step No.: The step No. for which an error occurred in (a) is displayed.

(2) When there is a cause in the parameter setting:



(a) Parameter No.: Illegal parameter No. is displayed.

1005: Common pointer No.

7000: Program setting

(b) File name: The name of parameter setting file relating with the error is displayed.

[Note] The displays other than (a) and (b) in (1) and (2) above have no significant meanings.
Also, depending on the error factors, (a) or (b) may not be displayed.

8.2.4 Display of the Error-generated Ladder

The ladder in which an error has occurred is displayed in the "PLC diagnostics" window. By following the operations below, the error-generated ladder which exists in the project currently opened with the GX Developer can be displayed in the edit mode.

Click "Error Jump" button (f) in the "PLC diagnostics" window.

The ladder of the file in which an error has occurred is displayed in the other window. Then the cursor moves to the step where the error has occurred.

The following conditions are required to accurately display the error-generated ladder section.

- The file name and sequence program No. must be displayed in the "Error details" screen.
  (In this case, no response or error dialog display will be given even if the button is clicked.)
- A project including the sequence program currently in operation in the CNC controller side must be opened.
  (Even when a project not in operation is opened, if the same program name already exists, a ladder different from the actual error-generated ladder is displayed. So, be careful.)
- Depending on the type of error, sequence step No. does not show the error-generated ladder accurately.

8.2.5 List of Corresponding PLC Alarms

The followings are the PLC alarms whose details and/or ladders in which an error has occurred are displayed in the "PLC diagnostics" window. Refer to "Appendix 4. List of PLC Alarms" for further detailed cause and remedy for each alarm.

NC side NC alarm display			GX Developer PC diagnosis display				Error details
Message	Sub status		Error code	Diagnosis display character string	File name	Step No.	
	1	2					
U10 Illegal PLC (Use PLC illegal) Program No. if the lower 16 bits are displayed as "xx" in NC alarm display sub status1. Step No. if "STP" in sub status 2.	0x0010	-	0	PROG. TIME OVER	-	-	Scan time error
	0x0040	-	0	PLCSEL ERR	-	-	Ladder selection parameter error
	0x04xx	STP	4	S/W INT. ERR	○	○	Software instruction interruption error
	0x20xx	STP	20	JUMP LABEL ERR	○	○	Label branching error
	0x21xx	STP	21	DUP. LABEL(P)	○	○	Label duplicate error
	0x22xx	-	22	LOCAL LABEL OVER	○	-	Local label over
	0x23xx	-	23	LABEL PARA. ERR	-	-	Global label boundary value error
	0x24xx	STP	24	RSV. LABEL ERR	○	○	Reserved label error
	0x25xx	-	25	PRG. PARA. ERR	○	-	Program setting error
	0x26xx	-	26	MISSING RET INS.	○	-	RET instruction error
	0x27xx	-	27	LAD. CODE ERR	○	-	Ladder code error
	0x28xx	-	28	MISSING LAD(M)	-	-	No main process ladder
	0x29xx	-	29	EXE. AREA OVER	○	-	Execution area over
	0x30xx	STP	30	FOR INS. OVER	○	○	FOR instruction nesting over
	0x31xx	STP	31	NEXT INS. ERR	△	△	NEXT instruction error
	0x32xx	STP	32	BREAK INS. ERR	○	○	BREAK instruction error
	0x400*	-	40	PLC SYSTEM DOWN	-	-	PLC system error
	0x80xx	STP	80	EXC.INT(BCD)	○	○	Exceptional interruption (BCD instruction error) has occurred.
	0x81xx	STP	81	EXC.INT(BIN)	○	○	Exceptional interruption (BIN instruction error) has occurred.
	0x82xx	STP	82	EXC.INT(D-BUS)	△	△	Exceptional interruption (bus error) has occurred.
	0x83xx		83	EXC.INT(INST.)	△	△	Exceptional interruption (unpopulated instruction error) has occurred.
	0x84xx		84	EXC.INT(I-FMT)	△	△	Exceptional interruption (instruction format error) has occurred.
	0x85xx		85	EXC.INT(I-BUS)	△	△	Exceptional interruption (instruction bus error) has occurred.
	0x86xx		86	EXC.INT(CALL)	△	△	Exceptional interruption (CALL/RET instruction error) has occurred.
	0x87xx		87	EXC.INT(MEM.)	△	△	Exceptional interruption (memory area error) has occurred.
	0x88xx		88	EXC.INT(ZERO)	△	△	Exceptional interruption (division-by-zero error) has occurred.
	0x89xx		89	EXC.INT(DUP.)	△	△	Exceptional interruption (Double exceptional generation error) has occurred.
	0x90xx		90	EXC.INT(HALT)	△	△	Exceptional interruption (HALT instruction halt) has occurred.

○: Correct information is displayed △: Information may not be accurate -: Not displayed

8.3 Initialization for PLC Data Storage Area

In the following cases, initialize the CNC controller's temporary memory area and start again.
The internal F-ROM is not initialized by this initialization operation. To hold the recovered state after the power is turned OFF, it must be stored in the internal F-ROM.

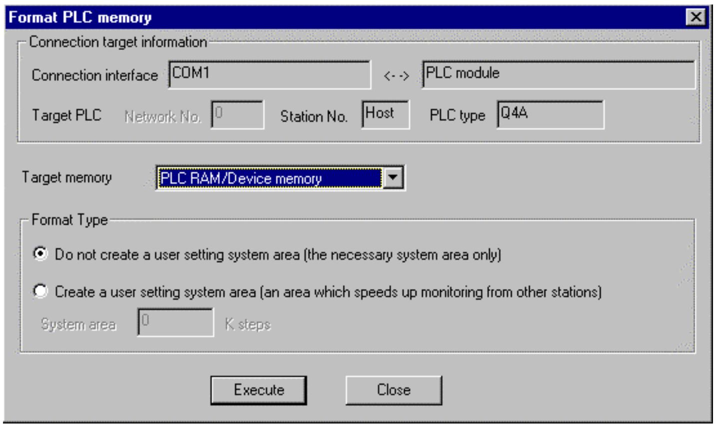
- If an error occurs while writing to the CNC controller
- If the state cannot be recovered even after remedying the error
- To delete the stored data at once

8.3.1 Operation Procedure

Perform the following operation from GX Developer to start the operation screen.

[Online] → [Format PLC memory]

On the following screen, click [Execute].



[Note] As [Target memory], only "PLC RAM/Device memory" is valid.
The setting is not necessary for [Format Type].

The setting is completed when the following dialog appears. Click [OK]. All data stored in the temporary memory have been deleted and initialized.

9. Procedures for Backing Up Data Such as Sequence Programs

This section explains the methods for backing up the developed sequence programs and PLC related data in a personal computer.

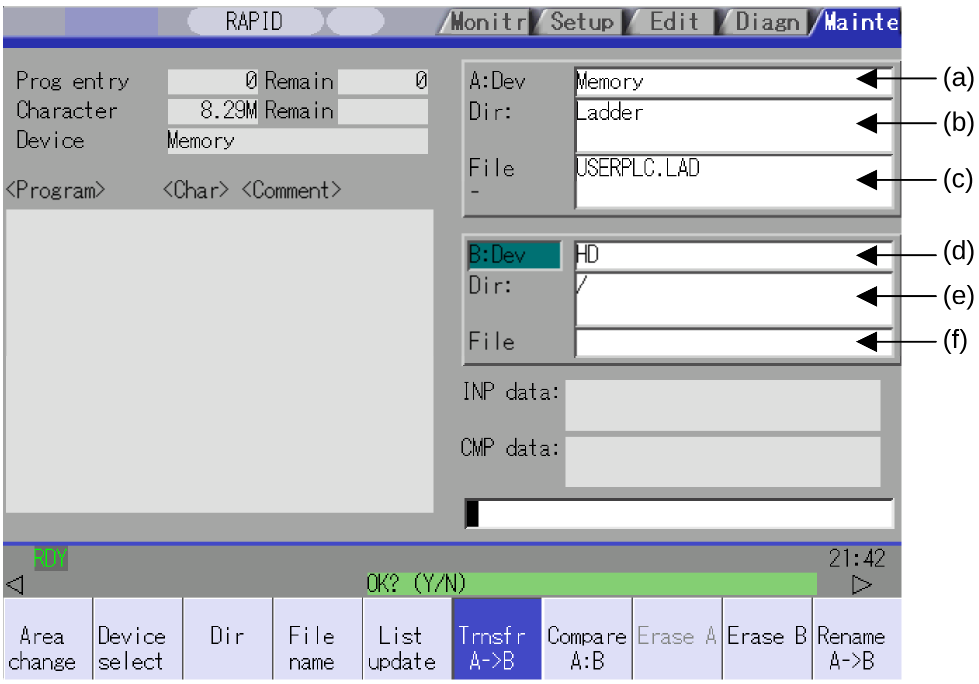
9.1 Backup Target Data

The following four types of data can be backed up.

	Related data classification	Remarks
1	Sequence program	Sequence program code
2	Parameter	Execution order setting information, etc.
3	Sequence program comment	GX Developer comment data
4	Messages in 1st to 8th languages	Message data such as alarm messages, operator messages and PLC switches in each language

9.2 Backup Procedures

The data is backed up from the CNC controller's input/output screen.
Designate the transmission source information (CNC controller side) and transmission target information (personal computer side) on the input/output screen. When transmission is started, the sequence programs, etc., stored in the RAM will be backed up in the personal computer.
The input/output screen and operation procedures are shown below.



Refer to the instruction manual for the respective model for details on the input/output screen.

(1) CNC controller settings

Set the "Device name", "Directory" and "File name" at "A: Device".

(a) Setting "Device name"
Select "Memory".

(b) Setting "Directory"
Input the character string "/LAD".

(c) Setting "File name"
"USERPLC.LAD" is automatically set when "Directory" is set.

(2) Personal computer settings (Setting "Device name", "Directory" and "File name")

Set the "Device name", "Directory" and "File name" at "B: Device".

(d) Setting "Device name"
Select "HD".

(e) Setting "Directory"
Input the character string "/".

(f) Setting "File name"
Set the name of the file to be stored. When omitted, "USERPLC.LAD" is assigned.

(3) Transmission

The backup process starts when the menu key "Send A→B" is pressed.

9.3 Restoring Backed Up Data

The data is restored following the backup operation in reverse (Set personal computer settings in "A: Device" and the CNC controller settings in "B: Device".)
The PLC must be STOP before restoration can be started.

IV EXPLANATION OF BUILT-IN EDITING FUNCTION

1. Outline

This manual explains the MITSUBISHI CNC 700/70 Series PLC onboard function. (Operations related to the PLC carried out with the CNC unit are collectively called as "onboard".)
Integration with the MELSEC Series PLC development tool (GX Developer) has been improved to enable reading and writing of data saved in each unit.
700 Series onboard includes "Standard operation mode" and "Simple operation mode".

Standard operation mode: All the onboard functions are available.
Simple operation mode: Limited to the functions mainly related to ladder monitoring, which are designed for routine maintenance operations.

Operation mode at the time of onboard startup can be switched between standard operation mode and simple operation mode by changing bit selection parameter settings. Switching modes is also possible after the onboard has been started.

1. Outline

The onboard functions are listed below.

List of functions

Function	Purpose of function	700		70
		Std	Sim	
■ Circuit monitoring				
Monitor start/stop	This starts or stops the monitor.	○	○	○
Device registration monitor	This monitors the circuit and the device registration simultaneously.	○	○	○
Circuit registration monitor	This monitors the circuit and the arbitrary registered circuit simultaneously.	○	○	○
Registered circuit all delete	This deletes all circuits registered with the circuit registration monitor.	○	○	○
Device test	This changes the device ON/OFF state, and changes the device value.	○	○	○
Monitor stop condition setting	This stops the monitor when the set device or step No. conditions are established.	○		
Current value monitor changeover (10/16)	This changes the circuit monitor device current value between the decimal and hexadecimal display.	○	○	○
■ Circuit editing				
Edit mode changeover	This edits the circuit.	○		
Line insert	This inserts a line at the cursor position.	○	○	○
Line delete	This deletes the line at the cursor position.	○	○	○
Copy & Paste	This copies and pastes the circuit in the designated range.	○	○	○
Statement edit	This edits the statements.	○	○	○
Note edit	This edits the notes.	○	○	○
Comment edit	This edits the comments.	○		
PLC message edit	This edits the PLC message.	○		○
Conversion	This converts the circuit.	○	○	○
Undo	This undoes the last edit operation.	○	○	○
■ Circuit search				
Simple search	This executes a simple search of contact, coil and device.	○	○	○
	This returns to the start.			○
Contact coil search	This searches the contact coils.	○		
Device search	This searches the devices.	○		
Instruction search	This searches the instructions.	○		
Step No. search	This searches the step Nos.	○	○	○
Character string search	This searches the character strings.	○		
AB contact change	This changes the circuit's contact between A and B.	○		
Device replace	This replaces the devices.	○		
TC setting value change	This changes the timer and counter setting values in a batch.	○		

Function	Purpose of function	700		70
		Std	Sim	
■ Circuit display				
Comment display	This sets whether to display a comment.	○	○	○
Program changeover	This changes the PLC program in circuit display. (Device comment tracking)			○
Data changeover	This changes the PLC data to be edited.	○	○	
Circuit display	This sets the circuit display size and the maximum number of contacts on one circuit line.	○		○
Zoom display	This switches the circuit display size.			
	Compact (64%) [Displays 11 contacts in 640x480 full screen] (70 Series: Reduction)			○
	Reduction (80%) [Displays 9 contacts in 640x480 full screen] (70 Series: Standard)	○	○	○
	Standard (100%) [Displays 9 contacts in 800x600 full screen] (70 Series: Expansion)	○	○	○
	Expansion (120%) [Displays 11 contacts in 1024x768 full screen]	○	○	
Zoom cursor	This enlarges the cursor display area at the Compact display (70 Series: Reduction).			○
Comment ON/OFF	This changes ON/OFF of the display for the contents set with "Comment display".	○	○	○
Comment line designation	This specifies the number of lines for the device comment display between 1 to 4.			○
Current value monitor line delete	This specifies whether to display the current value at monitoring.			○
■ Tool				
Contact coil usage list	This displays the device's step No. and usage state.	○		
Device usage list	This displays the usage list in a batch for each device type.	○		
Program check	This checks the sequence program (ladder).	○		
■ PLC data setting				
Add	This adds PLC data to the onboard editing area.	○		
Delete	This deletes PLC data from the onboard editing area.	○		
Name change	This changes the name of the PLC in the onboard editing area.	○		
Initial setting	This initializes the data in the onboard editing area.	○		
■ Device operation				
Device batch monitor	This monitors the devices in a batch.	○	○	○
Device registration monitor	This monitors registered devices on a dedicated screen.	○		
Sampling trace	This executes sampling trace.	○		
■ Parameters				
Program setting	This determines the sequence program (ladder) execution order.	○	○	○
Common pointer setting	This displays the common pointer head P No.	○		○

1. Outline

Function	Purpose of function	700		70
		Std	Sim	
■ NC file operation				
File list	This displays a list of the stored files. (execution step size is displayed)			○
Open	This displays the PLC data in the temporary memory at the onboard editing area.			
	Manually executed; the menu is selected when needed	○		○
	Automatically executed at power ON; no menu exists		○	○
Save	This saves the PLC data edited on the onboard in the temporary memory.			
	Manually executed; the menu is selected when needed (execution step size is displayed)	○		
	Automatically executed at conversion; no menu exists.	○	○	○
Verify	This verifies the PLC program in the onboard editing area with the PLC data in the temporary memory.	○		
ROM-Write	This saves the PLC data in the temporary memory to the ROM.	○	○	○
Delete	This deletes the PLC data from the temporary memory.	○		○
Format	This formats the temporary memory.	○		○
PLC RUN/STOP	This runs or stops the PLC.	○	○	○
PLC VERSION UP	This writes the ladders files in the external device into the NC's temporary memory and ROM.	○	○	
Disable keyword	This releases the keyword that has been set to the PLC data in NC.	○	○	○
■ External file operations				
Open project	This displays the PLC data in a project at the onboard.	○		○
Save project	This saves the PLC data edited on the onboard in a project.	○		○
Delete project	This deletes a project from the external device.	○		○
Verify project	This verifies the PLC data edited on the onboard with the PLC data in a project.	○		○
■ Environment setting				
Operation mode changeover	This changes modes between simple operation mode and standard operation mode.	○	○	
NC file operation setting	This sets the settings relating to the NC file operation.	○		
■ Diagnosis				
PLC diagnosis	This displays the error occurred during sequence program (ladder) execution.	○	○	○
■ Menu change				
Switch standard/simple menu key	This switches between standard and simple operation mode.	○	○	

Corresponding table for the files handled/not handled in simple operation mode, standard operation mode and 70 Series is shown below.

	Standard operation mode			Simple operation mode			70 Series		
	Handling	Display	Edit	Handling	Display	Edit	Handling	Display	Edit
Sequence program (Ladder)	○	○	○	○	○	○	○	○	○
PLC message data	○	○	○	×	×	×	○	○	○
Parameter	○	○ (Note2)	○ (Note2)	○	○ (Note2)	○ (Note2)	○	○ (Note2)	○ (Note2)
Device comment	○	○	○ (Note1)	○	○	×	○	○	×

(Note 1) Alphanumerical characters only
(Note 2) The available parameters are restricted to those for the program settings and the like.

2. Starting and Ending Onboard

2.1 Starting

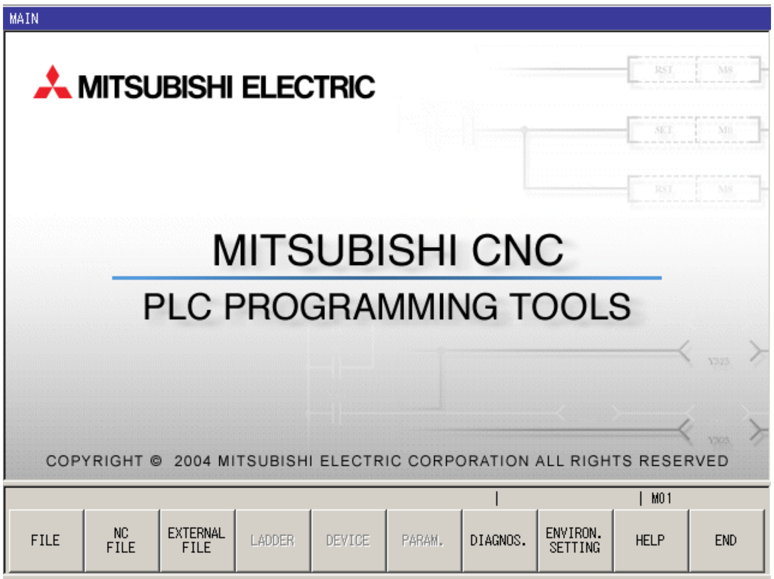
Onboard starts up when [F0] key is pressed on the NC.
In 700 Series, operation mode (Standard operation mode / Simple operation mode) at the time of startup can be determined by bit1 of the bit selection #6451.

```
                      7 6 5 4 3 2 1 0   <- Bit
#(6451)   Data ( 0 0 0 0 0 0 0 0 )
                                    |
                                    +---- 0: Startup with standard operation mode
Use #6451                                 1: Startup with simple operation mode
```

2.1.1 Startup with Standard Operation Mode

700		70
Standard	Simple	
○		

Initial screen



(1) State at Initial Startup

The state of the menu buttons on the main screen are as follows at the very initial startup of the onboard.

Menu button	State
FILE	This can be pressed at any time.
NC FILE	This can be pressed at any time.
EXTERNAL FILE	This can be pressed at any time.
LADDER	This cannot be pressed until the program data is opened in the onboard editing area.
DEVICE	This cannot be pressed until the program data is opened in the onboard editing area.
PARAM.	This cannot be pressed until the parameter data is opened in the onboard editing area.
DIAGNOS.	This can be pressed at any time.
ENVIRON. SETTING	This can be pressed at any time.
HELP	This can be pressed at any time.
END	This can be pressed at any time.

(2) PLC data previously opened on the onboard

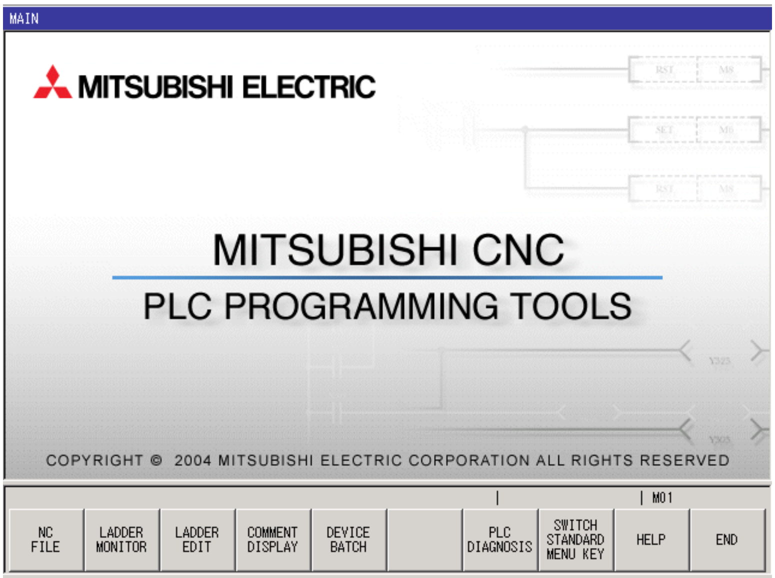
(a) PLC data on the onboard editing area
PLC data which was previously opened on the onboard will not be retained.
Open the program data from NC FILE or EXTERNAL FILE again.

(b) Connected NC
At the initial start-up, the onboard has to be connected to the same NC as the setting and display device screen side. Connected NC is displayed in the lower right corner of all the screens.

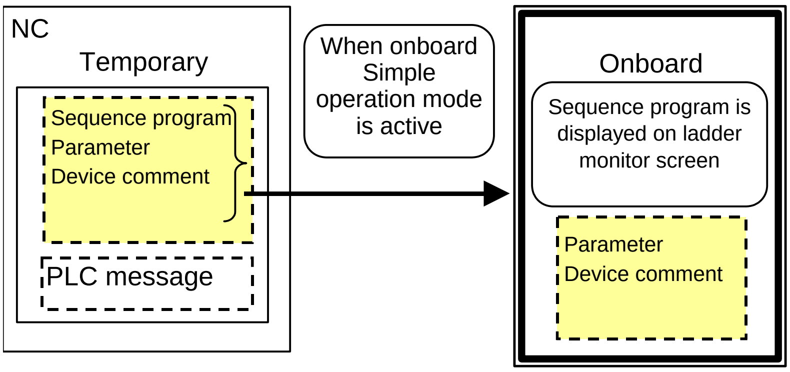
2.1.2 Startup with Simple Operation Mode

700		70
Standard	Simple	
	○	

Initial screen



When starting up the onboard, all the files except for PLC message data (sequence program, parameter, and device comment) will be read out from NC temporary memory and the sequence program will be automatically displayed on the ladder monitor screen.



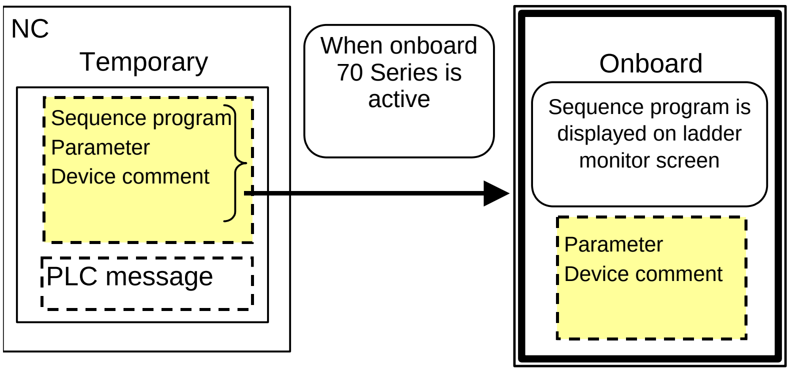
- When the number of stored sequence programs is one, that sequence program is displayed on the ladder monitor.
- When the number of stored sequence program is two or more, the upper level program determined by the parameter setting is displayed on the ladder monitor screen.
- When the device comment storage destination is specified with the standard operation mode environment setting menu key, device comment is read out from the specified storage area.

2.1.3 70 Series Startup

700		70
Standard	Simple	
		○

Initial screen

When starting up the onboard, all the files except for PLC message data (sequence program, parameter, and device comment) will be read out from NC temporary memory and the sequence program will be automatically displayed on the ladder monitor screen.



- When the number of stored sequence programs is one, that sequence program is displayed on the ladder monitor.
- When the number of stored sequence program is two or more, the upper level program determined by the parameter setting is displayed on the ladder monitor screen.
- If no sequence program can be read, the MAIN screen appears.

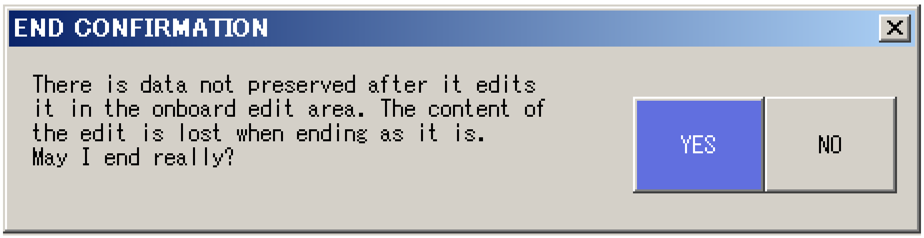
2.2 Ending

700		70
Standard	Simple	
○	○	○

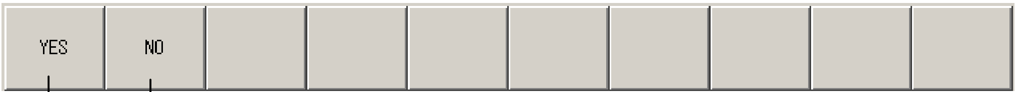
In 700 Series, onboard ends when the END menu key on the MAIN screen is pressed.
If there is any editing data which has not been saved in the temporary memory or external device when ending, the "END CONFIRMATION" popup screen will open.
In 70 Series, onboard ends when the CNC power is turned OFF.

■ When there is editing data which has not been saved

"END CONFIRMATION" popup screen (In 700 Series)



Menu corresponding to the popup screen



This cancels the ending process.

This ends the onboard. Any editing data which has not been saved will be lost.

(Note 1) If NC is shut down (power turned OFF) without executing ROM-Write (700 Series/ 70 Series)
The data in the onboard editing area and the data in the NC's temporary memory will be lost when the NC power is turned OFF. Always save this data on a ROM using ROM-Write.

(Note 2) If onboard is not ended with END menu (700 Series)
The data opened in the onboard editing area will be discarded.
If the program data is under the NC automatic update mode (LADDER screen's background color is white), the data up to the "converted" circuit will be saved in the NC's temporary data. (However, if it is not written to the ROM it will be lost when the NC power is turned OFF.)
If the program data is under the local editing mode (LADDER screen's background color is light blue), the data including the "converted" circuit will be lost. (The last "save" state will be retained.)

2.3 Switching from Simple Operation Mode to Standard Operation Mode

700		70
Standard	Simple	
	○	

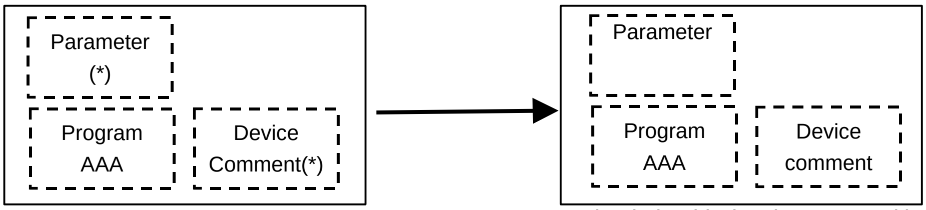
(1) Switching method

When "MAIN" → "SWITCH STANDARD MENU KEY" menu key is pressed, the following confirmation message will be displayed.

It switches to the standard operation menu key. Is it good?   YES/NO

When switching from simple operation mode to standard operation mode, PLC data in the onboard editing area will be held as it is.

Simple operation mode   Standard operation mode



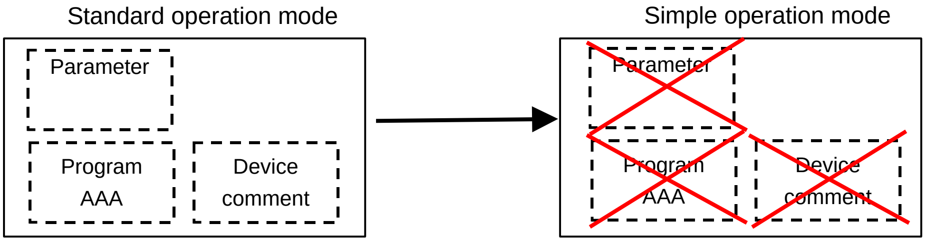
(*) Change not possible with simple operation mode

(Switch with the data opened.)

2.4 Switching from Standard Operation Mode to Simple Operation Mode

700		70
Standard	Simple	
○		

When switching from standard operation mode to simple operation mode, the state will be where the onboard is turned OFF and ON again. All the PLC data in the onboard editing area will be discarded and automatically reloaded from the NC temporary memory area.



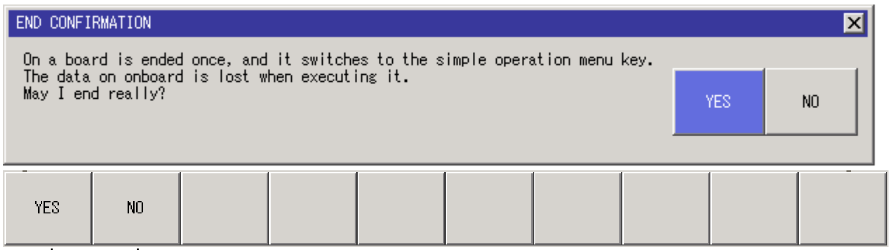
(State at the time of onboard startup)

(1) Switching method

When "SWITCH SIMPLE MENU KEY" menu key is pressed, the following "END CONFIRMATION" popup screen will be displayed.

"END CONFIRMATION" popup screen

Menus corresponding to popup screen



This cancels the ending process.

This ends the onboard. Restart the onboard with the simple operation mode menu key. Any editing data which has not been saved will be lost.

3. Screens

700		70
Standard	Simple	
○	○	VGA only

3.1 Screen Resolution

The setting and display unit's screen resolution differs according to the NC model. Onboard is compatible with the following two screen resolutions. This manual uses the VGA (640 x 480 pixel) screen display as an example.

(1) VGA (640 × 480 pixels)

(2) XGA (1024 × 768 pixels)

(Note 1) 70 Series onboard is compatible with VGA only.

700		70
Standard	Simple	
○	○	○

3.2 Types

The following three types of screens are displayed with the onboard.

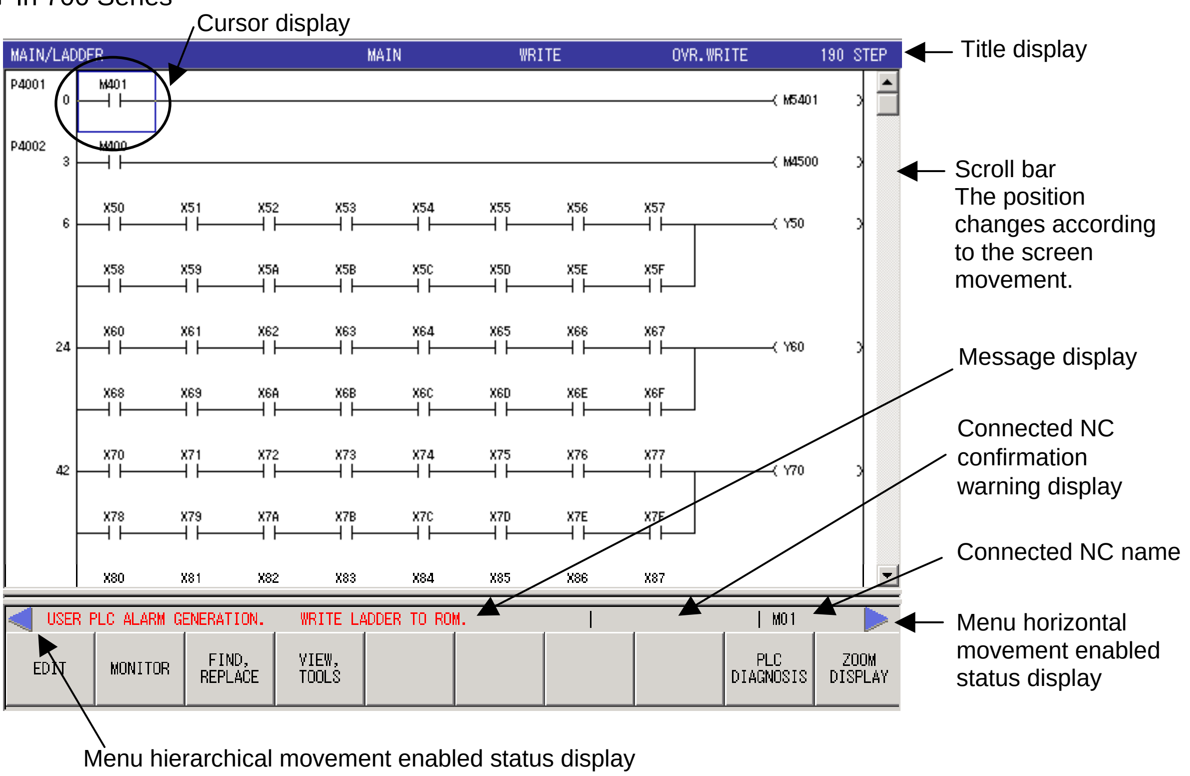
Full screen	This screen uses the full screen area.
Split screen	This screen splits the full screen into two areas and displays.
Popup screen	This screen is displayed over the full screen display or split screen. 70 Series have two display types; "window type" and "bar type".

700		70
Standard	Simple	
○	○	○

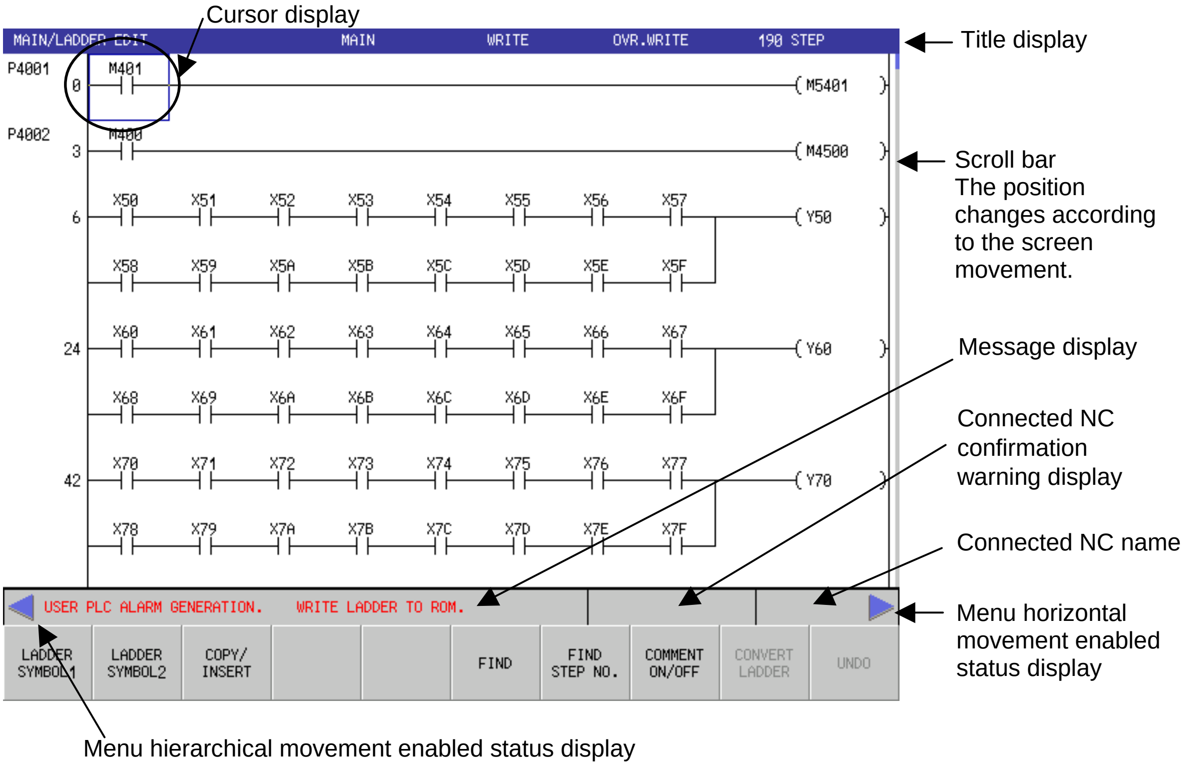
3.3 Full Screen Display

The full screen display ("LADDER" screen) is shown below.

■ In 700 Series



■ In 70 Series



(1) Title display

The currently displayed screen's menu hierarchy, program name, function mode (WRITE, MONITOR, etc.), editing mode (OVR.WRITE, INSERT), and total number of steps are displayed.
Refer to "3.9 Screen Title Display" for details.

(2) Scroll bar

This displays the position of the entire program.

(Note 1) In 700 Series, if the model is not provided with a pointing device such as a mouse or touch panel, the screen cannot be moved with the scroll bar. (Refer to (3) Moving the screen.)

(Note 2) 70 Series cannot move the screen with the scroll bar, even if any pointing device is provided.

(3) Screen movement

The screen can be moved with the following methods.

- Move the cursor with the arrow keys. (Move further at the top or bottom of the screen.)
- Move in page units with the page keys ⇧ and ⇩.

(Note) If there is an unconverted circuit on the "LADDER" screen, the movement range may be limited.

(4) Cursor display on the screen

The cursor displayed on the screen changes according to the function mode and writing mode state.

Function mode	Writing mode	Cursor	
		700 Series	70 Series
WRITE	OVR. WRITE	Blue Not filled in	Blue Not filled in
	INSERT	Purple Upper left corner is filled in	Purple Upper left corner is filled in
MONITOR START/STOP MONITOR		Blue Filled in	Red Not filled in

(5) Message display, progress display

Warning messages are displayed. ("WRITE RADDER TO ROM", etc.)
In 70 Series, a process progress (progress bar) is additionally displayed.

(6) Connected NC confirmation warning display

When the NC connected at the setting display unit screen side and the NC connected with the onboard differ, the warning is displayed. However, currently this display is not available.

(7) Connected NC name display

The number of the NC to which the onboard is currently connected is displayed.

Connected NC name display	Connected NC
M01	Connected with machine No. 1 NC
M02	Connected with machine No. 2 NC
:	:

(8) Menu hierarchical movement enabled status display, menu horizontal movement enabled status display

"Menu hierarchical movement enabled status display" is displayed when switching to the menu in the upward hierarchy is possible with $\boxed{\triangleleft}$ menu key. If this movement is not possible, this will not be displayed.

"Menu horizontal movement enabled status display" is displayed when switching menus within the same hierarchy or switching to the menu on the second page with $\boxed{\triangleright}$ menu key. If these movements are not possible, this will not be displayed.

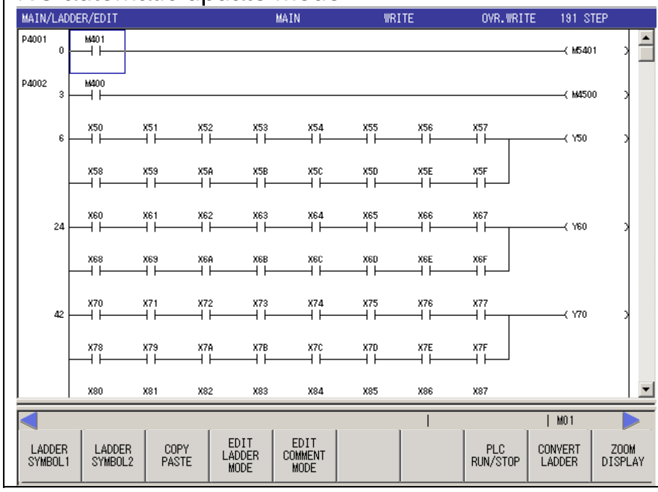
3.4 Color-coded Display of "LADDER" Screen

The "LADDER" screen (screen on which ladder circuit is displayed) is color-coded into two types and displayed.

700		70
Standard	Simple	
○	○	○

(1) NC automatic update mode

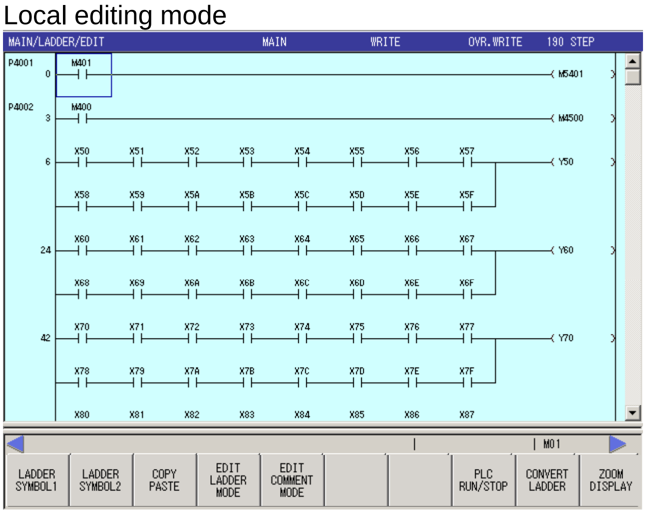
When the circuit is converted, the circuits of the program in the onboard editing area are converted, and the program with the same name in the NC temporary memory is rewritten.

Mode	Background color	Display conditions
NC automatic update mode 	White	• When displaying the circuits for a program for which OPEN , SAVE has been executed from the "NC FILE" menu.

(2) Local editing mode

700		70
Standard	Simple	
○		

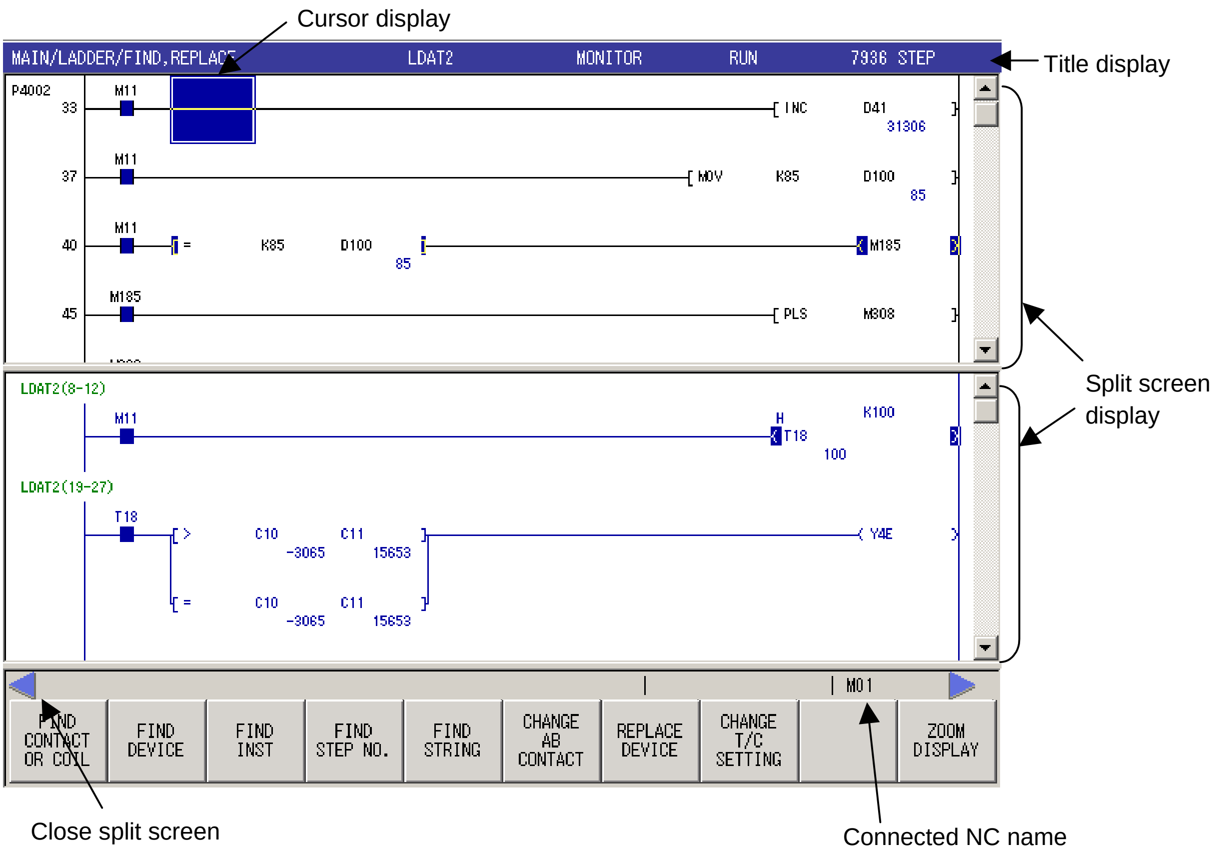
When the circuit is converted, only the program in the onboard editing area is rewritten.

Mode	Background color	Display conditions
Local editing mode 	Light blue	• When displaying the circuits for a program for which OPEN PROJECT , SAVE PROJECT has been executed from the "EXTERNAL FILE" menu. • When displaying the circuits for a program for which the data has been added or the data name has been changed with FILE on the "MAIN" menu. • When displaying the circuits for a program newly created with INITIAL SETTING from the "FILE" menu.

3.5 Split Display

700		70
Standard	Simple	
○	○	○

The split display ("LADDER MONITOR" screen) is shown below.
(The screen is an example of 700 Series)



(1) Title display

The currently displayed screen's menu hierarchy, program name, function mode (WRITE, MONITOR, etc.), editing mode (OVR.WRITE, INSERT), and total number of steps are displayed.
Refer to "3.9 Screen Title Display" for details.

(2) Cursor movement between split screens

Use "MOVEMENT ON SPLIT SCREEN" menu key to switch cursor between two split screens.
When the screen is not split, menu key is masked.
"#" key can also be used for switching cursor.

(3) Screen movement

The operation is the same as for the full screen display. Note that only the screen in which the cursor is displayed can be moved. In simple operation mode, when jumping to the other screen, split display will be cleared.

(4) Screen cursor display and message display

These displays are the same as the full screen.

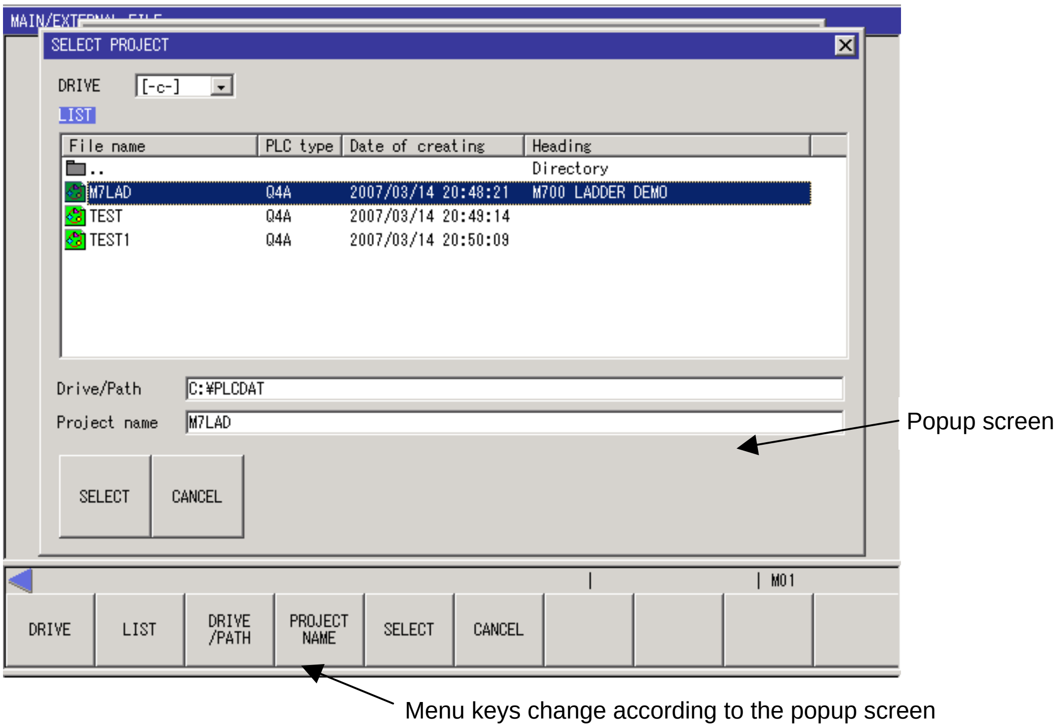
(5) Split screen menu key

When " ◁ " menu key is pressed, the split screen is closed.

3.6 Popup Screen

700		70
Standard	Simple	
○	○	○

The popup screen ("SELECT PROJECT" screen) is shown below.
(The screen is an example of 700 Series)



(1) Menus corresponding to popup screen

When a popup screen appears, menu keys corresponding to each item on the popup screen will appear.
The popup screen can be operated with these menu keys.
Refer to "3.11 Basic Screen Operations" for details.

(2) Closing the popup screen

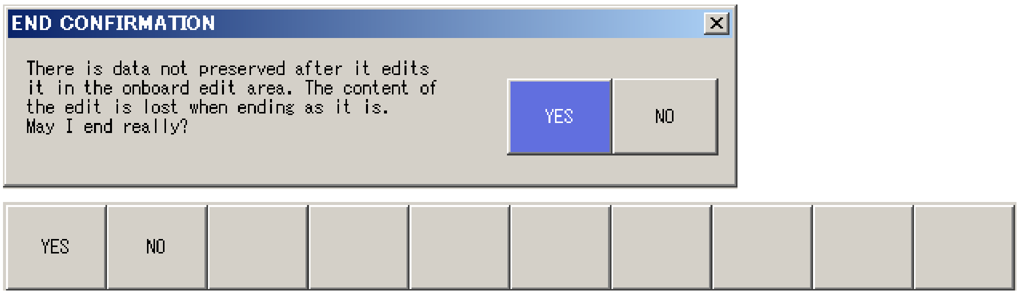
Press the ◁ key to close the popup screen.

3.7 Confirmation Popup Screen

700		70
Standard	Simple	
○	○	○

This popup screen opens to reconfirm the execution of operations or to confirm writing of data during PLC RUN.

(Example) "END CONFIRMATION" popup screen (The screen is an example of 700 Series)



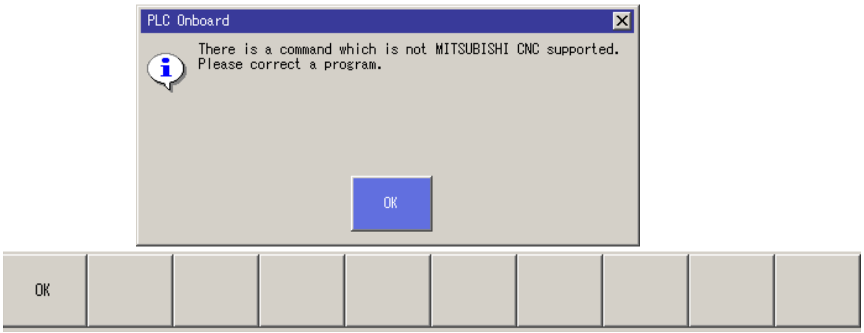
Menus corresponding to popup screen

3.8 Error Display Popup Screen

700		70
Standard	Simple	
○	○	○

This popup screen opens to reconfirm the execution of operations or to confirm writing of data during PLC RUN.

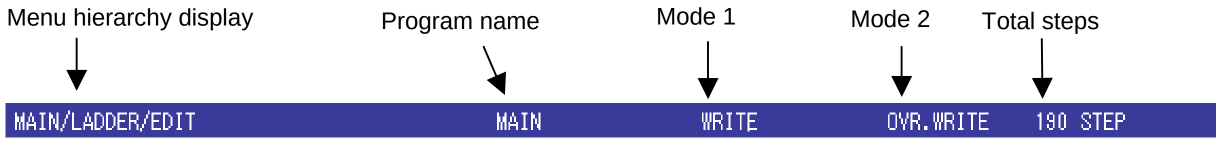
(Example) "ERROR" popup screen (The screen is an example of 700 Series)



Menus corresponding to popup screen

3.9 Screen Title Display

700		70
Standard	Simple	
○	○	○



(The screen is an example of 700 Series)

(1) Menu hierarchy display

The hierarchy of the currently displayed menu is displayed. The levels are delimited with "/".

(Example) For "EDIT" menu: "Main/circuit/edit"

(2) Program name

The name of the program currently targeted for editing and monitoring with onboard is displayed.

(3) Mode 1 display

The edit mode and monitor mode are indicated. The cursor display changes according to the mode status.

Mode 1	Status
WRITE	Indicates circuit editing.
MONITOR	Indicates monitoring is active.
STOP MONITOR	Indicates monitoring is stopped.

(4) Mode 2 display

OVR.WRITE/INSERT are indicated for the WRITE mode, and the PLC RUN/STOP are indicated during MONITOR and MONITOR STOP. The cursor display changes according to OVR.WRITE and INSERT.

Mode 2	Status
OVR. WRITE	The circuit can be edited with overwriting.
INSERT	The circuit can be edited with insertion.
RUN	The PLC is running.
STOP	The PLC is stopped.

(5) Total number of steps

The total number of steps in the program targeted for editing is displayed.

3.10 Menu Key Display

700		70
Standard	Simple	
○	○	○

The menu keys displayed at the bottom of the screen change according to the configuration given in "5.2 Menu Key". When a mouse or touch panel is used, pressing of the menu button can be confirmed by the instant denting and highlighting of the button. (The button is not highlighted in 70 Series.)
The menu keys, which allow the usage of INPUT key when entered, are enhanced (by the black border).

(Note) If a popup screen is displayed, the menu keys will change according to each item on the popup screen.

3.11 Basic Screen Operations

700		70
Standard	Simple	
○	○	○

The basic screen operations used commonly for each screen are explained in this section.

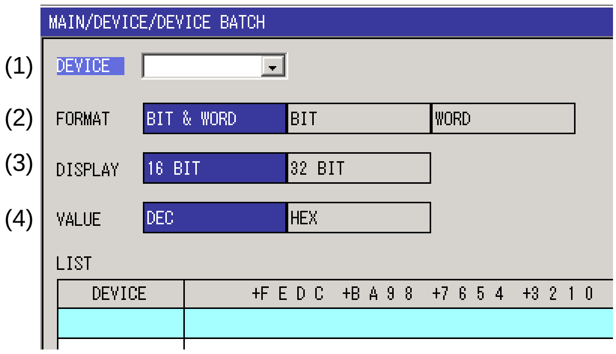
(1) Selection of items on screen

The input and selection items on the screen can be selected with the following two methods.

[Method 1] Selection of items with TAB keys (→|, |←)

When the TAB key is pressed, the active window will move in order of the items which must be input or selected. If the key is pressed when the active window is at the last item, it will return to the head item. The selected item is highlighted in light purple.

(Example) For "COMMENT DISPLAY" screen (The screen is an example of 700 Series)



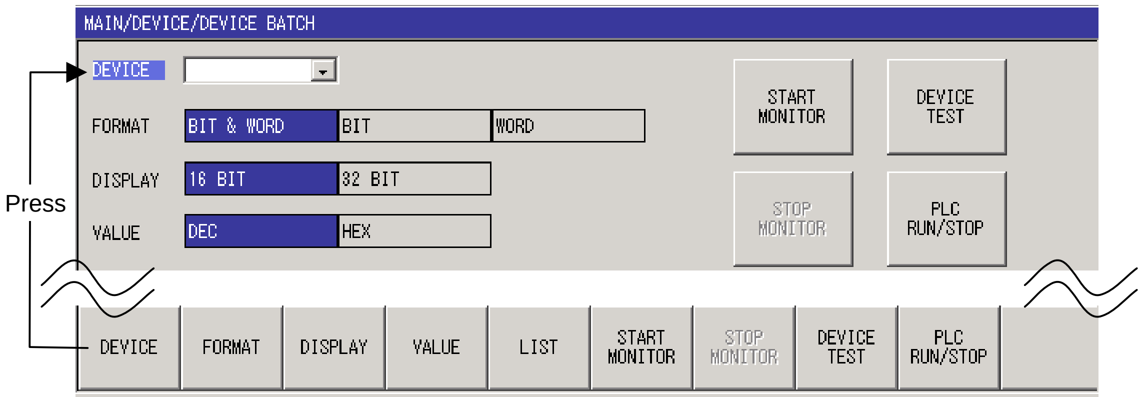
When →| is pressed, the selection items (light purple items) will move in the forward order 1 to 4.

When |← is pressed, the selection items (light purple items) will move in the reverse order 4 to 1.

[Method 2] Selection of items with menu keys (Direct selection of selection items)

When the menu button with the same name as the item displayed on the screen is pressed, the item can be selected. Once the button is pressed, the item name is displayed in "light purple' and the data can be set.

(Example) For "PROGRAM CHANGE" screen (The screen is an example of 700 Series)

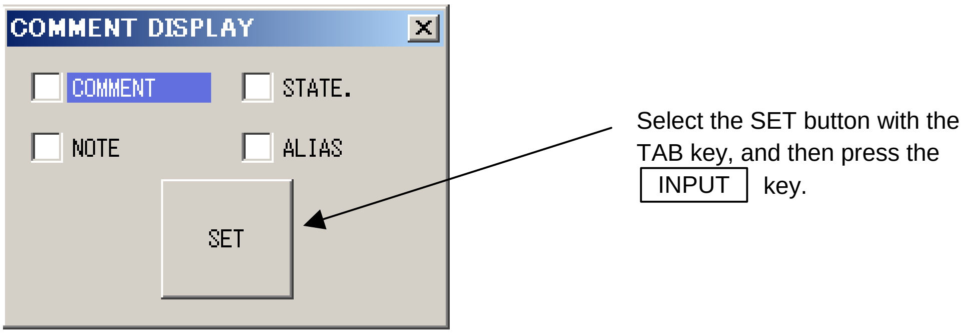


(2) Pressing buttons in screen (700 Series only)

The buttons on the screen can be operated with the following two methods.

[Method 1] Select button with TAB key and then press INPUT key.

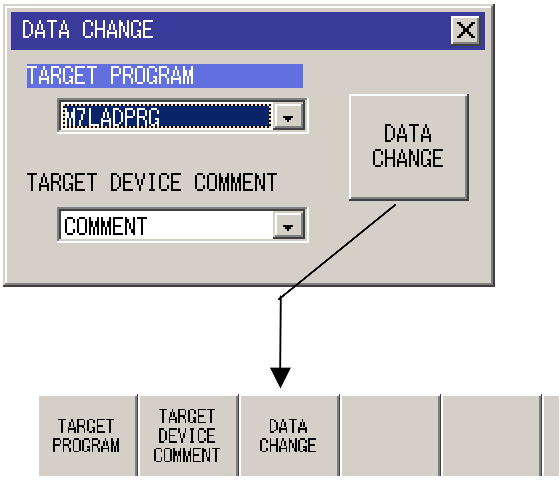
(Example) Operation using same menu as button on "COMMENT DISPLAY" screen.



(Supplement) If operations are possible with the touch panel, the buttons on the screen can be directly pressed.

[Method 2] Press the menu key with the same name as the button.

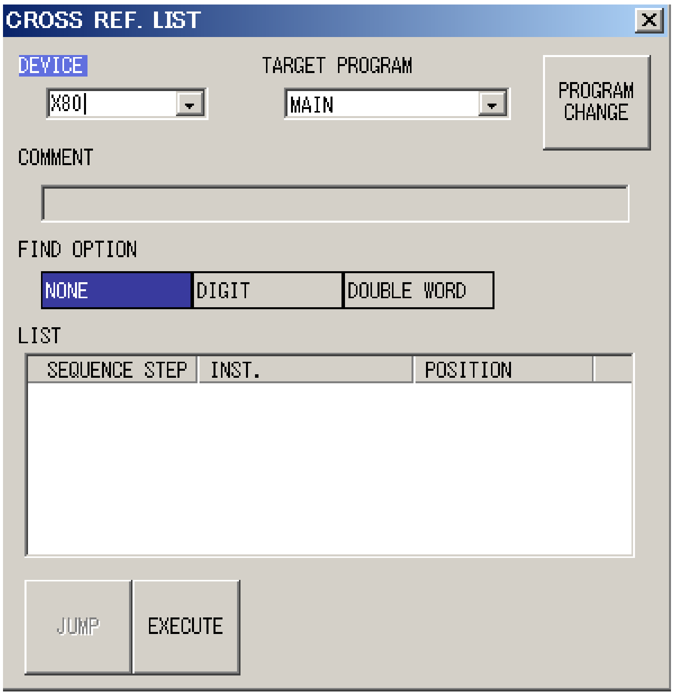
(Example) Operation using same menu as button on "PROGRAM CHANGE" screen.



(3) Setting characters (values)

Characters (values) can be set in the items with a white box. The input methods are explained below.
(a) Select the item to be input. Refer to (1) for details.
(b) Input the data.

(Example) To input X80 on the "CROSS REF. LIST" screen (The screen is an example of 700 Series)



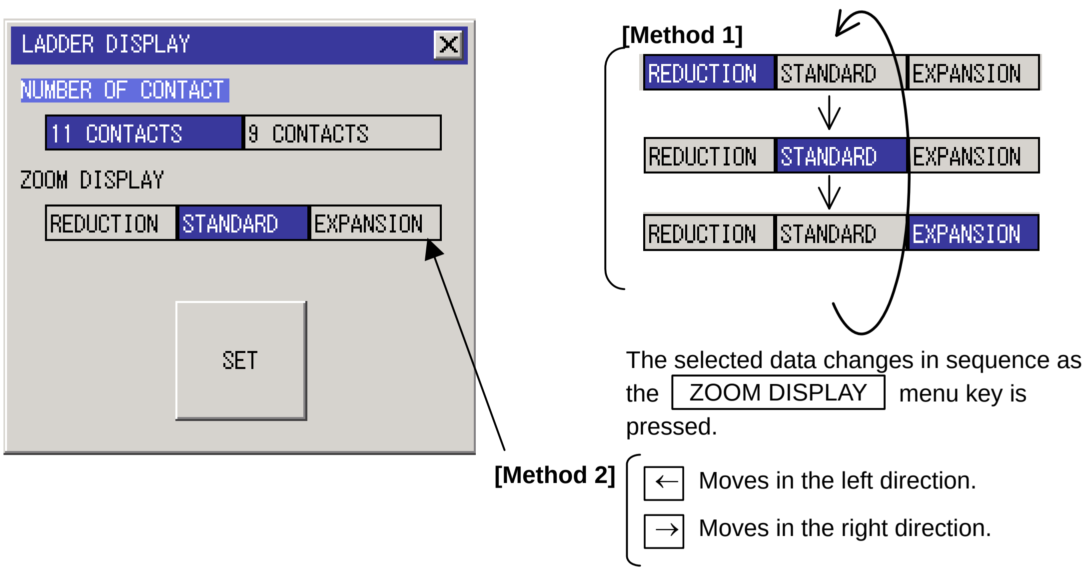
(4) Setting unique selection items

Only one data item is selected from two or more data items. The highlighted data is the currently selected item. There are two methods to change the selection.

[Method 1] When the menu key with the same name as the item is pressed, the selection data will move in the right direction. If the currently selected item is at the far right, the selection data will return to the head.

[Method 2] After selecting the item with the TAB key, change the data with the ← and → keys.

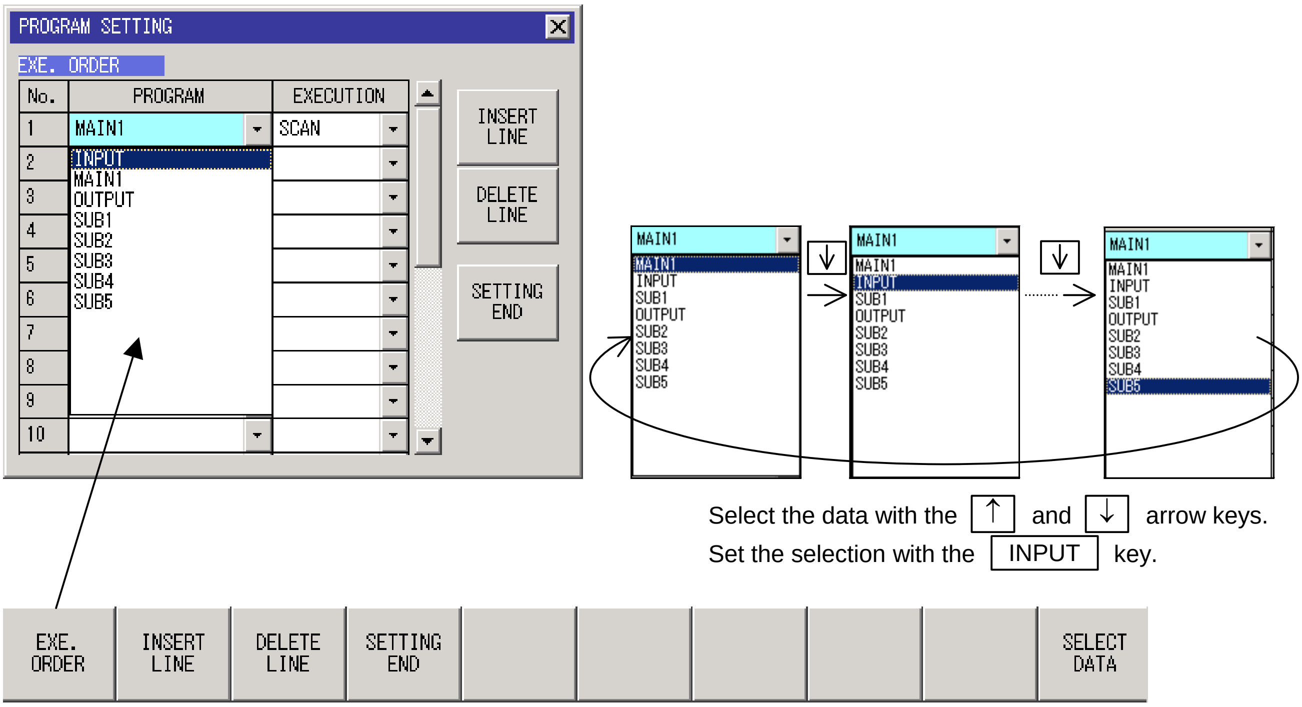
(Example) To select ZOOM DISPLAY on the "LADDER DISPLAY" screen (The screen is an example of 700 Series)



(5) Setting the list selection items

Data can be selected from a list. When the menu key with the same name as the item name is pressed, or when an item is selected with a TAB key, a list will appear. The data at the very top is selected (highlighted) first, and the selected data can be changed with the ↑ and ↓ arrow keys.
Press the INPUT key to set the selection.

(Example) When "EXE. ORDER" is selected on the "PROGRAM SETTING" screen (The screen is an example of 700 Series)



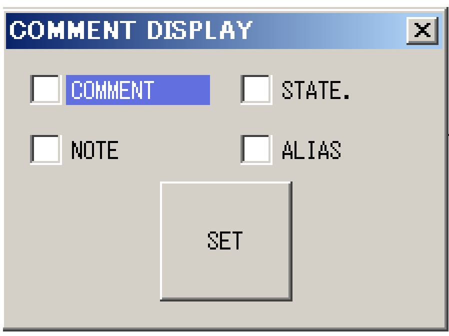
(6) Items with validity setting (700 Series only)

The validity of each item can be set with the following two methods.

[Method 1] When the menu key with the same name as the item is pressed, the status will alternate between valid (with check mark) and invalid (no check mark).

[Method 2] When the SP key is pressed after selecting the item with the TAB key, the validity status will alternate.

(Example) For "COMMENT DISPLAY" screen

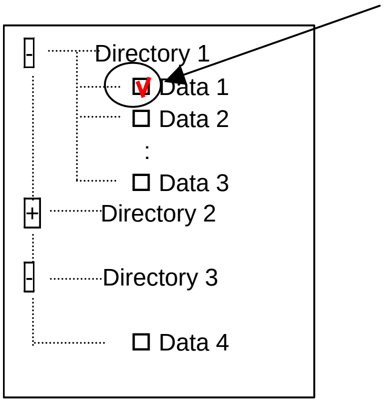


[Method 1]
Press COMMENT menu key.

[Method 2]
After selecting item with TAB key, press SP key.

(7) Selecting tree-structure data

Tree-structure data is configured of data and a directory. The data and directories are shown below.



Indicates valid data

<Moving the data and directory>

- The data will move by one item according to the direction of the pressed arrow key [↑], [↓], [←] or [→].

<Setting the data validity>

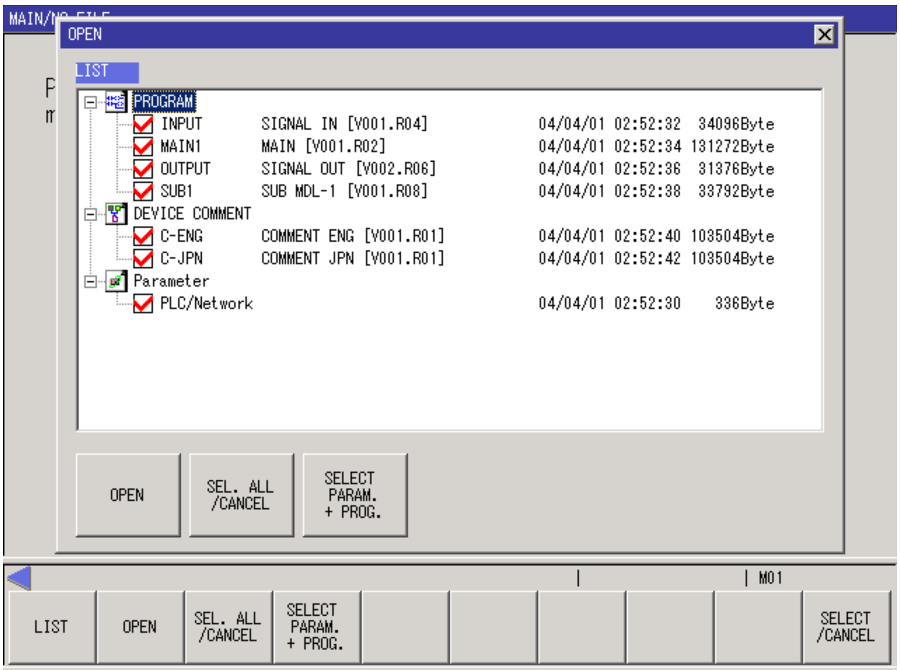
The following two methods of operations are available.

- Move the cursor to the data to be set, and press [SP]. (The validity will change with each press.)
- Move the cursor to the data to be set, and press the "SELECT/CANCEL" menu key. (The validity will change with each press.)

<Directory [+] and [-] >

- If the [+] key is pressed in the [→] state, the data in the directory will appear.
- If the [-] key is pressed in the [←] state, the data in the directory will be hidden.

(Example) NC file operation - "OPEN" screen (The screen is an example of 700 Series)



3.12 Language

700		70
Standard	Simple	
○	○	○

3.12.1 Screen Display Language

The language used for the buttons, items and error messages on the onboard can be changed with the NC's language parameter.
If the language parameter value is not for the valid language, the language will be handled as English characters.

3.12.2 Comment (Statement, Note, Comment, Device Name) Language

The fonts used to display comments (STATE, NOTE, COMMENT, ALIAS) can be changed with the language parameter setting. If the language parameter value is not for the valid language, the language will be handled as English characters.

4. PLC Data

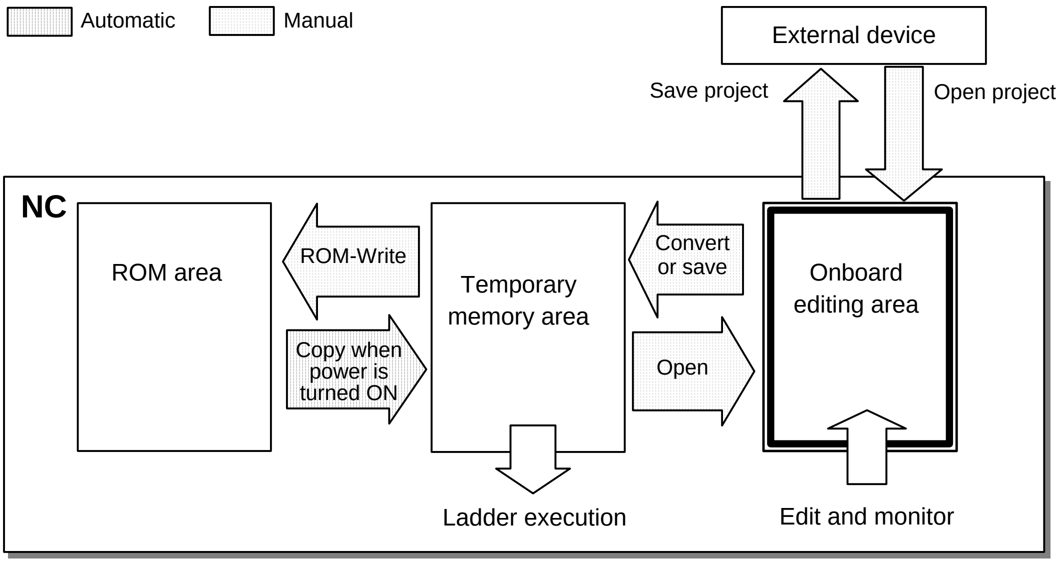
700		70
Standard	Simple	
○	○	○

4.1 PLC Data Storage Area

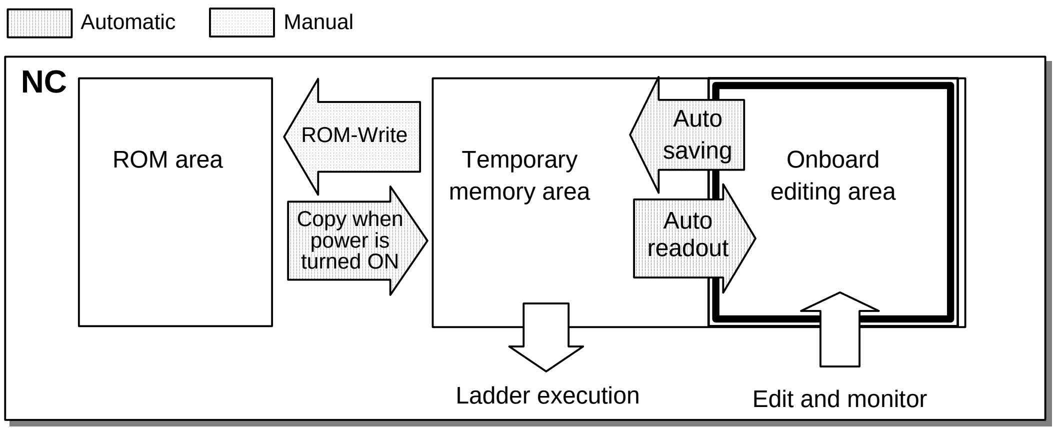
The PLC data is stored in the following areas of the NC.

Storage area	Application
Temporary memory area	This area stores the data when executing the PLC. It can be edited with onboard and GX Developer. The data in this area is lost when the NC power is turned OFF.
ROM area	This area stores the PLC data. The data in this area is not lost even when the NC power is turned OFF. **(Note)** The data in the temporary memory is lost when the NC power is turned OFF, so always save the data in the ROM area.
Onboard editing area	This area is used to edit and monitor the PLC data with the onboard. When editing or monitoring, open the PLC data in the temporary memory into the onboard editing area.
External device	PLC data can be read from the external device to the onboard editing area. The GX Developer project data can be read.

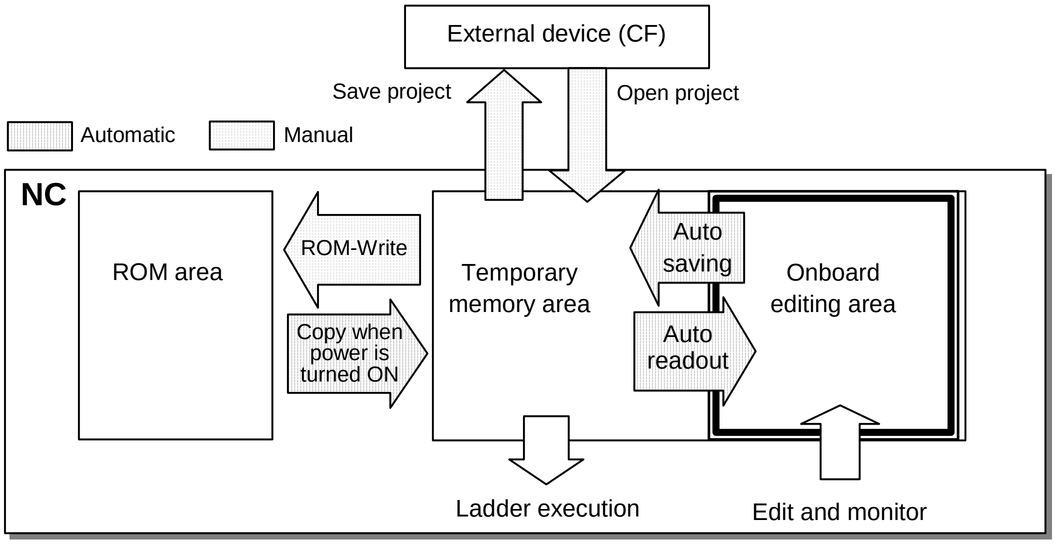
PLC data storage area configuration drawing - Standard operation mode -



PLC data storage area configuration drawing - Simple operation mode -



PLC data storage area configuration drawing - 70 Series -



4.2 Type of Data

Onboard can handle the same type of PLC data (programs, device comments, parameters) as GX Developer.

(Note 1) The following GX Developer data cannot be used with onboard.

- Parameter's network parameter data
- Device memory data
- Device default value data

List of data handled by onboard

Data type	Data name	Application
Program	The data name is configured of up to eight arbitrary one-byte uppercase alphanumeric characters *1 **(Note)** A reserved name is used only when storing the PLC message. * Refer to 2.	The user PLC, statements and notes can be stored. The PLC messages (alarm, operator, PLC switch, comment) can be stored with a reserved name.
Device comment	The data name is configured of up to eight arbitrary one-byte uppercase alphanumeric characters.	Comments (comment, device name) can be stored. **(Note 1)** With the onboard, the device name is only displayed and cannot be edited. **(Note 2)** Data name "COMMENT" is the reserved name for a common comment and cannot be changed. The common comment "COMMENT" will be displayed even when the other device comment data is designated if no comment is defined for the data.
Parameter	Fixed name "param"	The program setting parameters can be stored.

*1: Symbols and spaces cannot be used in the data name.
*2: When storing PLC message data, the following reserved names are used. Reserved name specifications differ depending on the language specification method. Refer to "III PERIPHERAL DEVELOPMENT ENVIRONMENT 4.4.2 File name rule for message data" for details.

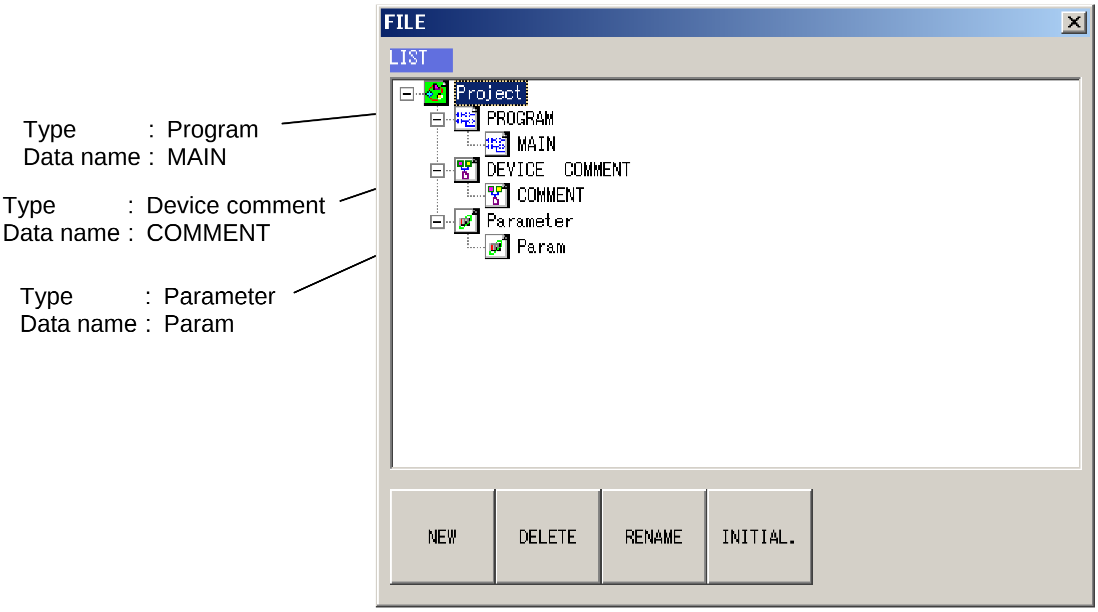
(Method 1)
Specify with 3 bits of bit selection parameter #6453 bit 0 to 2. (Language selection method using PLC alone)

Data type	Data name	Details of data
Program	M1xxxxxx	PLC message of the 1st language
	:	
	M7xxxxxx	PLC message of the 7th language
	M8xxxxxx	PLC message of the 8th language

(Method 2)
Specify with display language selection parameter. (Base specifications parameter #1043)
(Method linked with language selection on the setting and display screen)

Data type	Data name	Details of data
Program	M00xxxxx	Language para 0 (English) PLC message
	M01xxxxx	Language para 1 (Japanese) PLC message
	:	
	M22xxxxx	Language para 22 (Chinese -simplified-) PLC message

Example: PLC data on "FILE" screen (The screen is an example of 700 Series)



4.2.1 Program Data

The following details can be saved in the data created as program data.

Saved details	Outline
Program (ladder) circuit	Program (ladder) circuit is the data used to edit and monitor the user PLC with a circuit diagram consisting of -| |- -|/|- -( )- -[ ]-.
Statement	A statement is character string data added to each program (ladder) circuit block to make it easy to understand the flow of the entire program. Statements include integrated statements that can be stored in the NC, and peripheral statements that cannot be stored in the NC. (Refer to *1)
Note	A note is character string data added to each coil and function instruction in the program (ladder circuit) to make it easy to understand the flow of the entire program in the same manner as statements. Notes include integrated notes that can be stored in the NC, and peripheral notes that cannot be stored in the NC. (Refer to *1)

*1: Integrated statements and notes, peripheral statements and notes

Integrated type	The integrated types can be saved in the NC. Note that the program data takes up a large memory when stored in the NC.
Peripheral types	The peripheral types are deleted when saved in the NC, and cannot be saved. Thus, when using peripheral statements or notes, they must be controlled with GX Developer projects.

For PLC messages and special programs

Saved details	Outline
Alarm messages	These are PLC alarm message character strings displayed on the NC.
Operator messages	These are PLC operator message character strings displayed on the NC.
PLC switches	These are character strings for the PLC switch names.
Comment messages	These are PLC comment character strings displayed on the NC.

4.2.2 Device Comment Data

The following can be saved in data created as device comments.

Saved details	Outline
Comment	This is character string data added to each device. The program is easier to understand when meanings are assigned to the devices.
Device name	This character string is displayed instead of the device name, and makes it easier to see the devices. (Note that the device name cannot be edited with the onboard, so this is used only for display.)

4.2.3 Parameter Data

The following can be saved in data created as parameters.

Saved details	Outline
Program settings	The program (ladder) execution order can be defined when using multi-programs.

5. Explanation of Keys (Keys Related to Onboard)

700		70
Standard	Simple	
○	○	○

5.1 Basic Operation Keys

The keys used with onboard are shown below.

Key	Explanation
F0	This opens the onboard screen.
A to Z , 0 to 9	These are alphanumeric keys, etc., used to input arbitrary data.
DELETE	This key deletes the last character input before it is set. This key deletes one circuit during ladder circuit editing.
C • B	This key deletes the input character string before it is set (cancels the input).
INPUT	This key is used to set and select the input data.
INSERT	This key changes between the circuit overwrite and insert modes. In 70 Series, this key changes between the data overwrite and insert modes.
↑ ↓ ← →	These keys are used to move the cursor to up, down, left and right, and to select items.
→|	This key moves the items on the screen in the forward order. This key moves the instruction circuit in the forward order during ladder circuit editing.
|←	This key moves the items on the screen in the reverse order. This key moves the instruction circuit in the reverse order during ladder circuit editing.
#	This switches the cursor between screens when device registration monitor or circuit registration monitor is displayed in split screen.
EOB (;)	This switches ON and OFF of device comment display on the circuit screen.
◁ or ESC	This returns the menu of one level up in hierarchy. When a popup screen is displayed, this closes the popup screen.
▷	This changes the menu in the same hierarchy.
⇧	This moves the page forward in a page unit.
⇩	This moves the page backward in a page unit.
SP (Space)	This switches valid/invalid of the check box ☑ .
Ctrl+Z	This cancels the last edit operation and returns to the previous state with the edit ladder mode.
<,>	When <,> key does not exist on keyboard depending on the machine type, menu buttons on the "ENTER SYMBOL" screen, "FIND" screen, and "FIND INSTRUCTION" screen can be used to input.

*: Key allocations may differ depending on the machine type.

5.2 Menu Keys

5.2.1 Menu Keys in Standard Operation Mode and Simple Operation Mode

Menu keys in standard operation mode : Entire onboard function can be used.
Menu keys in simple operation mode : User-friendly configuration, limiting the functions strictly to maintenance purposes.

5.2.2 Menu Key Hierarchies and Movement

700		70
Standard	Simple	
○		

5.2.2.1 Menu keys in standard operation mode

The menu for standard operation mode is configured of the following four hierarchies.

Corresponding menu

Moves to downward hierarchy ▷

Moves to upward hierarchy ◁

Hierarchy 1	Hierarchy 2	Hierarchy 3	Hierarchy 4
MAIN	NC FILE		
	EXTERNAL FILE		
	LADDER	EDIT	LADDER SYMBOL 1
			LADDER SYMBOL 2
			COPY/INSERT
		MONITOR	
		FIND, REPLACE	
		VIEW, TOOLS	
	DEVICE	DEVICE BATCH	
		ENTRY DEVICE	
		SAMPLING TRACE	
	PARAM		

▷ Movement in same hierarchy ▷

700		70
Standard	Simple	
	○	

5.2.2.2 Menu keys in simple operation mode

The menu for simple operation mode is configured of the following three hierarchies.

Corresponding menu

Moves to downward hierarchy ▷

Moves to upward hierarchy ◁

Hierarchy 1	Hierarchy 2	Hierarchy 3
MAIN	NC FILE	
	LADDER MONITOR 1st page 2nd page	
	LADDER EDIT	LADDER SYMBOL 1
		LADDER SYMBOL 2
		COPY/INSERT

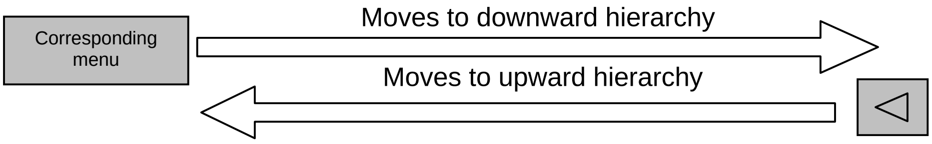
▷ Switch pages

▷ Movement within the same hierarchy

5.2.2.3 Menu Keys in 70 Series

700		70
Standard	Simple	
		○

The menu for 70 Series is configured of the following three hierarchies.



Hierarchy 1	Hierarchy 2	Hierarchy 3
MAIN	NC FILE	
	LADDER MONITOR 1st page 2nd page	
	LADDER EDIT 1st page 2nd page	LADDER SYMBOL 1 LADDER SYMBOL 2 COPY/INSERT

Switch pages

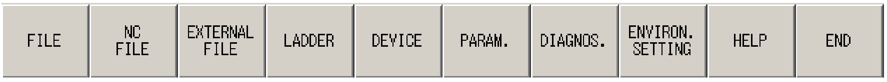
Movement within the same hierarchy

5.2.3 Details of Menu Keys

700		70
Standard	Simple	
○		

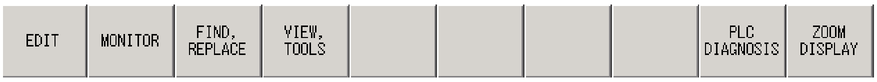
5.2.3.1 Menu keys in standard operation mode

(1) "MAIN" menu keys



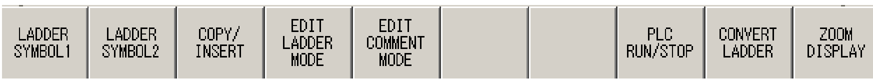
FILE	This opens the "FILE" popup screen.
NC FILE	This changes to the "NC FILE" menu key.
EXTERNAL FILE	This changes to the "EXTERNAL FILE" menu key.
LADDER	This changes to the "LADDER" menu key. When no program is opened in the onboard editing area, this button will be invalid (displayed in gray).
DEVICE	This changes to the "DEVICE" menu key.
PARAM.	This changes to the "PARAM." menu key.
DIAGNOS.	This changes to the "DIAGNOS." menu key.
ENVIRON. SETTING	This changes to the "ENVIRON. SETTING" menu key.
HELP	This opens the "HELP" screen.
END	This ends the onboard.

(2) "LADDER" menu keys



EDIT	This changes to the "EDIT" menu key.
MONITOR	This changes to the "MONITOR" menu key. (Note) Only the program in the NC's temporary memory can be monitored. Thus, the "monitor" menu for the circuit with light blue background turns gray and the circuit cannot be monitored.
FIND, REPLACE	This changes to the "FIND, REPLACE" menu key.
VIEW, TOOLS	This changes to the "VIEW, TOOLS" menu key.
PLC DIAGNOSIS	This opens the "PLC DIAGNOSIS" popup screen.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(2-1) "EDIT" menu keys

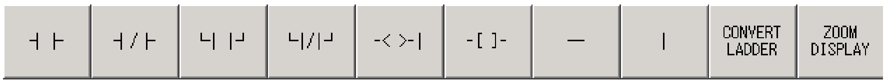


LADDER SYMBOL 1	This changes to the "LADDER SYMBOL 1" menu key.
LADDER SYMBOL 2	This changes to the "LADDER SYMBOL 2" menu key.
COPY/INSERT	This changes to the "COPY/INSERT" menu key.
EDIT LADDER MODE	This changes the mode to the EDIT LADDER mode.
EDIT COMMENT MODE	This changes the mode to the EDIT COMMENT mode.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
CONVERT LADDER	This converts the program (ladder) currently being edited. The converted ladder will be updated to the temporary memory in the NC automatic update mode.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(2-2) "COPY/INSERT" menu keys

MARK	The start point and end point are designated to select the circuit group.
COPY	This copies the circuit in the designated range.
PASTE	This pastes the copied circuit.
INSERT LINE	This inserts a line at the cursor position in the "LADDER" screen.
DELETE LINE	This deletes the line at the cursor position in the "LADDER" screen.
UNDO	This undoes the last edit operation.
DATA CHANGE	This opens the "DATA CHANGE" popup screen.
CONVERT LADDER	This converts the program (ladder) currently being edited. The converted ladder will be updated to the temporary memory in the NC automatic update mode.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(2-2-1) "LADDER SYMBOL 1", "LADDER SYMBOL 2" menu keys

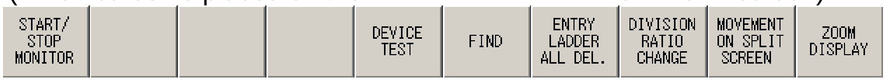


Ladder symbols other than the following	During edit mode This opens the "INPUT" popup screen. During monitor mode This opens the "FIND" popup screen.
- (Cross bar)	This writes a "cross bar" at the cursor position in the "LADDER" screen.
| (Vertical bar)	This writes a "vertical bar" at the cursor position in the "LADDER" screen.
| (Vertical bar) DELETE	This deletes the "vertical bar" at the cursor position in the "LADDER" screen.
CONVERT LADDER	This converts the program (ladder) currently being edited. The converted ladder will be updated to the temporary memory in the NC automatic update mode.
ZOOM DISPLAY	This expands/reduces the ladder display size.

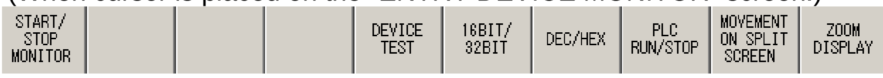
(2-3) "MONITOR" menu keys

(When cursor is placed on the "LADDER" screen)

(When cursor is placed on the "ENTRY LADDER MONITOR" screen)



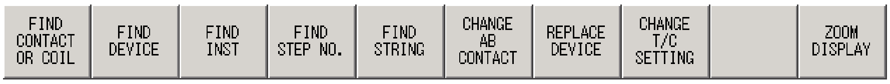
(When cursor is placed on the "ENTRY DEVICE MONITOR" screen.)



START/STOP MONITOR	This starts or stops the monitor.
ENTRY DEVICE	This splits the "ENTRY DEVICE" screen or cancels the split display.
ENTRY LADDER MONITOR	This splits the "ENTRY LADDER MONITOR" screen or cancels the split display.
REGISTER MONITOR	This registers the circuit on the cursor position in the entry ladder monitor.
DEVICE TEST	This opens the "DEVICE TEST" popup screen.
MONITOR STOP CONDITI.	This opens the "MONITOR STOP CONDITI." popup screen.
CHANGE MONITOR DEC/HEX	This changes displays of the current value for the device on the ladder monitor between decimal and hexadecimal.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
MOVEMENT ON SPLIT SCREEN	This moves the cursor between "LADDER" screen and "ENTRY DEVICE"/"ENTRY LADDER MONITOR" screen on the split screen.
ZOOM DISPLAY	This expands/reduces the ladder display size.
FIND	This opens the "FIND" popup screen.
ENTRY LADDER ALL DEL.	This deletes all the circuits registered in the entry ladder monitor.
DIVISION RATIO CHANGE	This changes the ratio of the split screen.
16 BIT/32 BIT	This displays when the cursor is on the "ENTRY DEVICE MONITOR" screen. This changes between word and W word of the word device registered in the ENTRY DEVICE MONITOR.

* "MONITOR" menu can be used only in the circuit (program) whose background color is white.
When the circuit's (program's) background color is light blue, the same hierarchial movement to the "MONITOR" menu will be skipped.

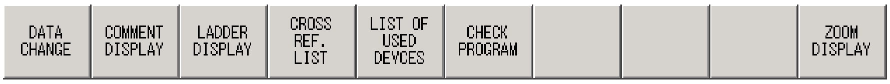
(2-4) "FIND, REPLACE" menu keys



FIND CONTACT OR COIL	This opens the "FIND CONTACT OR COIL" popup screen.
FIND DEVICE	This opens the "FIND DEVICE" popup screen.
FIND INST	This opens the "FIND INST" popup screen.
FIND STEP NO.	This opens the "FIND STEP NO." popup screen.
FIND STRING	This opens the "FIND STRING" popup screen.
CHANGE AB CONTACT	This opens the "CHANGE AB CONTACT" popup screen.
REPLACE DEVICE	This opens "REPLACE DEVICE" popup screen.
CHANGE T/C SETTING	This opens the "CHANGE T/C SETTING" popup screen.
ZOOM DISPLAY	This expands/reduces the ladder display size.

* "CHANGE AB CONTACT", "REPLACE DEVICE", and "CHANGE T/C SETTING" menus can be used only in the circuit (program) whose background color is light blue.

(2-5) "VIEW, TOOLS" menu keys

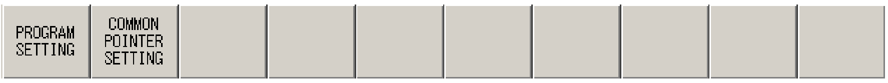


DATA CHANGE	This opens the "DATA CHANGE" popup screen.
COMMENT DISPLAY	This opens the "COMMENT DISPLAY" popup screen.
LADDER DISPLAY	This opens the "LADDER DISPLAY" popup screen.
CROSS REF. LIST	This opens the "CROSS REF. LIST" popup screen.
LIST OF USED DEVICES	This opens the "LIST OF USED DEVICES" popup screen.
CHECK PROGRAM	This opens the "CHECK PROGRAM" popup screen.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(3) "DEVICE" menu keys

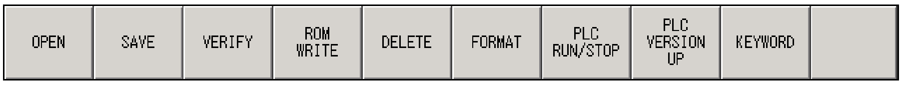
DEVICE BATCH	This opens the "DEVICE BATCH" screen.
ENTRY DEVICE	This opens the "ENTRY DEVICE" screen.
SAMPLING TRACE	This opens the "SAMPLING TRACE" screen.

(4) "PARAM." menu keys



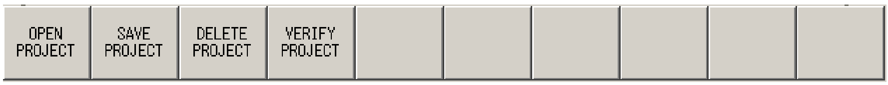
PROGRAM SETTING	This opens the "PROGRAM SETTING" popup screen.
COMMON POINTER SETTING	This opens the "COMMON POINTER SETTING" popup screen.

(5) "NC FILE" menu keys



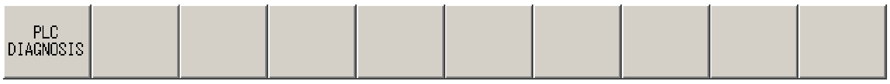
OPEN	This opens the "OPEN" popup screen.
SAVE	This opens the "SAVE" popup screen.
VERIFY	This opens the "VERIFY" popup screen.
ROM WRITE	This opens the "ROM WRITE" popup screen.
DELETE	This opens the "DELETE" popup screen.
FORMAT	This opens the "FORMAT" popup screen.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
PLC VERSION UP	This opens the "PLC VERSION UP" popup screen.
KEYWORD	This opens the "KEYWORD" screen.

(6) "EXTERNAL FILE" menu keys



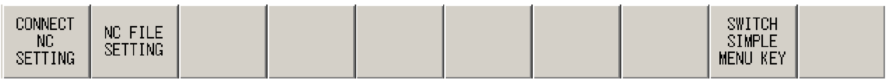
OPEN PROJECT	This opens the "OPEN PROJECT" popup screen.
SAVE PROJECT	This opens the "SAVE PROJECT" popup screen.
DELETE PROJECT	This opens the "DELETE PROJECT" popup screen.
VERIFY PROJECT	This opens the "VERIFY PROJECT" popup screen.

(7) "DIAGNOSIS" menu keys



PLC DIAGNOSIS	This opens the "PLC DIAGNOSIS" popup screen.

(8) "ENVIRON. SETTING" menu keys

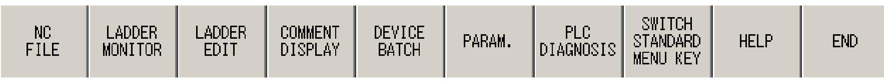


CONNECT NC SETTING	This opens the "CONNECT NC SETTING" popup screen.
NC FILE SETTING	This opens the "NC FILE SETTING" popup screen.
SWITCH SIMPLE MENU KEY	This changes the display to the simple menu key.

5.2.3.2 Menu keys in simple operation mode

700		70
Standard	Simple	
	○	

(1) "MAIN" menu keys



NC FILE	This changes to the "NC FILE" menu key.
LADDER MONITOR	This changes to the "LADDER MONITOR" menu key. (*1)
LADDER EDIT	This changes to the "LADDER EDIT" menu key. (*1)
COMMENT DISPLAY	This opens the "COMMENT DISPLAY" popup screen. (*1)
DEVICE BATCH	This opens the "DEVICE BATCH" screen. (*1)
PARAM.	This changes to the "PARAM." menu key.
PLC DIAGNOSIS	This opens the "PLC DIAGNOSIS" popup screen.
SWITCH STANDARD MENU KEY	This changes the display to the standard menu key.
HELP	This opens the "HELP" screen.
END	This ends the onboard.

*1: When no program is opened in the onboard editing area, this button will be invalid (displayed in gray).

(2) "NC FILE" menu keys

ROM WRITE	PLC VERSION UP	EXECUTE STEP					PLC RUN/STOP	KEYWORD	

ROM WRITE	This opens the "ROM WRITE" popup screen.
PLC VERSION UP	This opens the "PLC VERSION UP" popup screen.
EXECUTE STEP	This opens the "EXECUTE STEP" popup screen.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
KEYWORD	This opens the "KEYWORD" screen.

(3) "LADDER MONITOR" menu keys

(When cursor is placed on the "LADDER" screen.)

START/STOP MONITOR	ENTRY DEVICE	ENTRY LADDER MONITOR	REGISTER MONITOR	DEVICE TEST	FIND	FIND STEP NO.	COMMENT ON/OFF	MOVEMENT ON SPLIT SCREEN	ZOOM DISPLAY

					DATA CHANGE	CHANGE MONITOR DEC/HEX	PLC RUN/STOP	MOVEMENT ON SPLIT SCREEN	ZOOM DISPLAY

(When cursor is placed on the "ENTRY LADDER MONITOR" screen.)

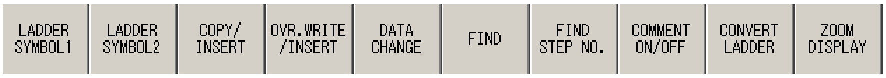
START/STOP MONITOR				DEVICE TEST	FIND	ENTRY LADDER ALL DEL.	DIVISION RATIO CHANGE	MOVEMENT ON SPLIT SCREEN	ZOOM DISPLAY

(When the cursor is placed on the "ENTRY DEVICE MONITOR" screen.)

START/STOP MONITOR				DEVICE TEST	16BIT/32BIT	DEC/HEX	PLC RUN/STOP	MOVEMENT ON SPLIT SCREEN	ZOOM DISPLAY

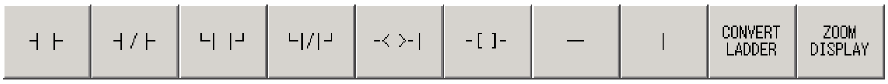
START/STOP MONITOR	This starts or stops the monitor.
ENTRY DEVICE	This splits the "ENTRY DEVICE" screen or cancels the split display.
ENTRY LADDER MONITOR	This splits the "ENTRY LADDER MONITOR" screen or cancels the split display.
REGISTER MONITOR	This registers the circuit on the cursor position in the entry ladder monitor.
DEVICE TEST	This opens the "DEVICE TEST" popup screen.
FIND	This opens the "FIND" popup screen.
FIND STEP NO.	This opens the "FIND STEP NO." popup screen.
COMMENT ON/OFF	This changes ON/OFF of the comment display.
MOVEMENT ON SPLIT SCREEN	This moves the cursor between "LADDER" screen and "ENTRY DEVICE"/"ENTRY LADDER MONITOR" screen on the split screen.
ZOOM DISPLAY	This expands/reduces the ladder display size.
DATA CHANGE	This opens the "DATA CHANGE" popup screen.
CHANGE MONITOR DEC/HEX	This changes displays of the current value for the device on the ladder monitor between decimal and hexadecimal.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
ENTRY LADDER ALL DEL.	This deletes all the circuits registered in the entry ladder monitor.
DIVISION RATIO CHANGE	This changes the ratio of the split screen.
16 BIT/32 BIT	This displays when the cursor is on the "ENTRY DEVICE MONITOR" screen. This changes between word and W word of the word device registered in the ENTRY DEVICE MONITOR.

(4) "LADDER EDIT" menu keys



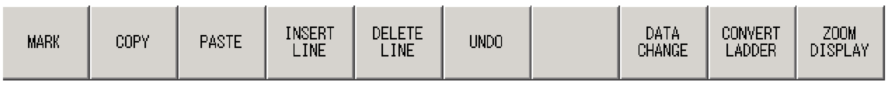
LADDER SYMBOL 1	This changes to the "LADDER SYMBOL 1" menu key.
LADDER SYMBOL 2	This changes to the "LADDER SYMBOL 2" menu key.
COPY/INSERT	This changes to the " COPY/INSERT " menu key.
OVR.WRITE/INSERT	This changes overwrite/insert.
DATA CHANGE	This opens the "DATA CHANGE" popup screen.
FIND	This opens the "FIND" popup screen.
FIND STEP NO.	This opens the "FIND STEP NO." popup screen.
COMMENT ON/OFF	This changes ON/OFF of the comment display.
CONVERT LADDER	This converts the program (ladder) currently being edited.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(4-1) "LADDER SYMBOL 1", "LADDER SYMBOL 2" menu keys



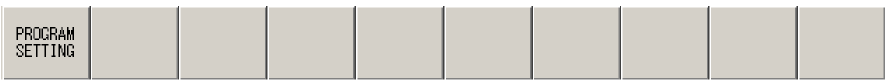
Ladder symbols other than the following	During edit mode This opens the "INPUT" popup screen. During monitor mode This opens the "FIND" popup screen.
- (Cross bar)	This writes a "cross bar" at the cursor position in the "LADDER" screen.
| (Vertical bar)	This writes a "vertical bar" at the cursor position in the "LADDER" screen.
| (Vertical bar) DELETE	This deletes the "vertical bar" at the cursor position in the "LADDER" screen.
CONVERT LADDER	This converts the program (ladder) currently being edited.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(4-2) "COPY/INSERT" menu keys



MARK	The start point and end point are designated to select the circuit group.
COPY	This copies the circuit in the designated range.
PASTE	This pastes the copied circuit.
INSERT LINE	This inserts a line at the cursor position in the "LADDER" screen.
DELETE LINE	This deletes the line at the cursor position in the "LADDER" screen.
UNDO	This undoes the last edit operation.
DATA CHANGE	This opens the "DATA CHANGE" popup screen.
CONVERT LADDER	This converts the program (ladder) currently being edited.
ZOOM DISPLAY	This expands/reduces the ladder display size.

(5) "PARAM." menu keys



PROGRAM SETTING	This opens the "PROGRAM SETTING" popup screen.

5.2.3.3 Menu Keys in 70 Series

700		70
Standard	Simple	
		○

(1) "MAIN" menu keys

NC FILE	EXT.FILE OPERATION	LADDER MONITOR	LADDER EDIT	DEVICE	PARAM.	PLC DIAGNOSIS		ENVIRON. SETTING	HELP

NC FILE	This changes to the "NC FILE" menu key.
EXT.FILE OPERATION	This changes to the "EXTERNAL FILE" menu key.
LADDER MONITOR	This changes to the "LADDER MONITOR" menu key. (*1)
LADDER EDIT	This changes to the "LADDER EDIT" menu key. (*1)
DEVICE	This opens the "DEVICE BATCH" screen. (*1)
PARAM.	This changes to the "PARAM." menu key. (*1)
PLC DIAGNOSIS	This opens the "PLC DIAGNOSIS" popup screen.
ENVIRON. SETTING	This changes to the "ENVIRON. SETTING" menu key.
HELP	This opens the "HELP" screen.

*1: When no program is opened in the onboard editing area, this button will be invalid (displayed in gray).

(2) "NC FILE" menu keys

LIST	OPEN		ROM WRITE		DELETE	FORMAT	PLC RUN/STOP	KEYWORD	

LIST	This opens the "FILE SIZE DISPLAY" screen.
OPEN	This opens the "OPEN" screen.
ROM WRITE	This opens the "ROM WRITE" screen.
DELETE	This opens the "DELETE" popup screen.
FORMAT	This opens the "FORMAT" popup screen.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
KEYWORD	This opens the "KEYWORD" screen.

(3) "EXTERNAL FILE" menu keys

EXT.->NC	NC->EXT.	VERIFY EXT.FILE	DELETE EXT. FILE				PLC RUN/STOP	KEYWORD	

EXT.->NC	This opens the "EXT.->NC" screen.
NC->EXT.	This opens the "NC->EXT." screen.
VERIFY EXT.FILE	This opens the "VERIFY" screen.
DELETE EXT.FILE	This opens the "DELETE" popup screen.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
KEYWORD	This opens the "KEYWORD" screen.

(4) "LADDER MONITOR" menu keys

(When cursor is on the "LADDER" screen.)

START/ STOP MONITOR	ENTRY DEVICE	ENTRY LADDER MONITOR	REGISTER MONITOR	DEVICE TEST	FIND	FIND STEP NO.	COMMENT ON/OFF	MOVEMENT ON SPLIT SCREEN	PROGRAM CHANGE

			ZOOM DISPLAY	ZOOM CURSOR		CHANGE MONITOR DEC/HEX	PLC RUN/STOP	MOVEMENT ON SPLIT SCREEN	PROGRAM CHANGE

(When cursor is on the "ENTRY LADDER MONITOR" screen.)

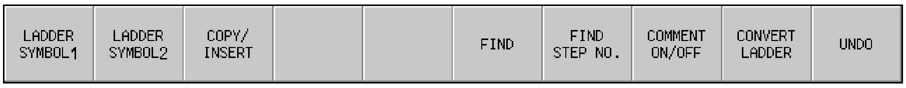
START/ STOP MONITOR			ZOOM DISPLAY	DEVICE TEST	FIND	ENTRY LADDER ALL DEL.	DIVISION RATIO CHANGE	MOVEMENT ON SPLIT SCREEN	

(When the cursor is on the "ENTRY DEVICE MONITOR" screen.)

START/ STOP MONITOR				DEVICE TEST	16BIT/ 32BIT	DEC/HEX	PLC RUN/STOP	MOVEMENT ON SPLIT SCREEN	

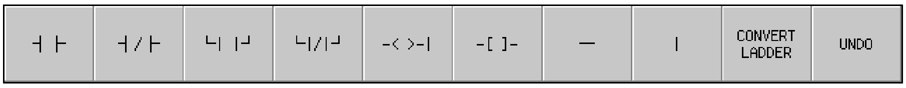
START/STOP MONITOR	This starts or stops the monitor.
ENTRY DEVICE	This splits the "ENTRY DEVICE" screen or cancels the split display.
ENTRY LADDER MONITOR	This splits the "ENTRY LADDER MONITOR" screen or cancels the split display.
REGISTER MONITOR	This registers the circuit on the cursor position in the entry ladder monitor.
DEVICE TEST	This opens the "DEVICE TEST" popup screen.
FIND	This opens the "FIND" popup screen.
FIND STEP NO.	This opens the "FIND STEP NO." popup screen.
COMMENT ON/OFF	This changes ON/OFF of the comment display.
MOVEMENT ON SPLIT SCREEN	This moves the cursor between "LADDER" screen and "ENTRY DEVICE"/"ENTRY LADDER MONITOR" screen on the split screen.
PROGRAM CHANGE	This changes the programs in the circuit display.
ZOOM DISPLAY	This expands/reduces the ladder display size.
ZOOM CURSOR	This enlarges the cursor display area at the compact display.
CHANGE MONITOR DEC/HEX	This changes displays of the current value for the device on the ladder monitor between decimal and hexadecimal.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
ENTRY LADDER ALL DEL.	This deletes all the circuits registered in the entry ladder monitor.
DIVISION RATIO CHANGE	This changes the ratio of the split screen.
16 BIT/32 BIT	This displays when the cursor is on the "ENTRY DEVICE MONITOR" screen. This changes between word and W word of the word device registered in the ENTRY DEVICE MONITOR.
DEC/HEX	This switches the decimal/hexadecimal display.

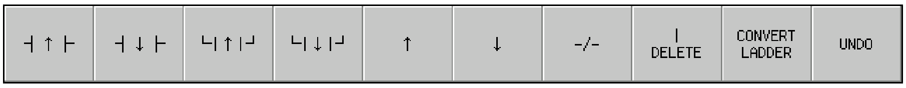
(5) "LADDER EDIT" menu keys



LADDER SYMBOL 1	This changes to the "LADDER SYMBOL 1" menu key.
LADDER SYMBOL 2	This changes to the "LADDER SYMBOL 2" menu key.
COPY/INSERT	This changes to the " COPY/INSERT " menu key.
FIND	This opens the "FIND" popup screen.
FIND STEP NO.	This opens the "FIND STEP NO." popup screen.
COMMENT ON/OFF	This changes ON/OFF of the comment display.
CONVERT LADDER	This converts the program (ladder) currently being edited.
UNDO	This undoes the last edit operation.
ZOOM DISPLAY	This expands/reduces the ladder display size.
ZOOM CURSOR	This enlarges the cursor display area at the compact display.
OVR.WRITE/INSERT	This changes overwrite/insert.
PLC RUN/STOP	This opens the "PLC RUN/STOP" popup screen.
CANCEL EDIT LADDER	This discards the unconverted circuit.
PROGRAM CHANGE	This changes the programs in the circuit display.

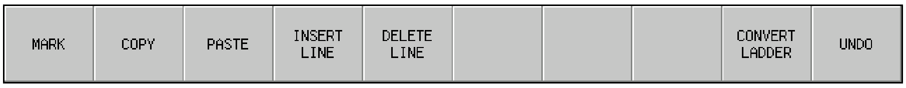
(5-1) "LADDER SYMBOL 1", "LADDER SYMBOL 2" menu keys





Ladder symbols other than the following	During edit mode This opens the "INPUT" popup screen. During monitor mode This opens the "FIND" popup screen.
- (Cross bar)	This writes a "cross bar" at the cursor position in the "LADDER" screen.
| (Vertical bar)	This writes a "vertical bar" at the cursor position in the "LADDER" screen.
| (Vertical bar) DELETE	This deletes the "vertical bar" at the cursor position in the "LADDER" screen.
CONVERT LADDER	This converts the program (ladder) currently being edited.
UNDO	This undoes the last edit operation.

(5-2) "COPY/INSERT" menu keys



MARK	The start point and end point are designated to select the circuit group.
COPY	This copies the circuit in the designated range.
PASTE	This pastes the copied circuit.
INSERT LINE	This inserts a line at the cursor position in the "LADDER" screen.
DELETE LINE	This deletes the line at the cursor position in the "LADDER" screen.
CONVERT LADDER	This converts the program (ladder) currently being edited.
UNDO	This undoes the last edit operation.

(6) "DEVICE" menu keys

DEVICE BATCH	This opens the "DEVICE BATCH" screen.

(7) "PARAM." menu keys

PROGRAM SETTING	This opens the "PROGRAM SETTING" screen.
COMMON POINTER SETTING	This opens the "COMMON POINTER SETTING" screen.

(8) "ENVIRON. SETTING" menu keys

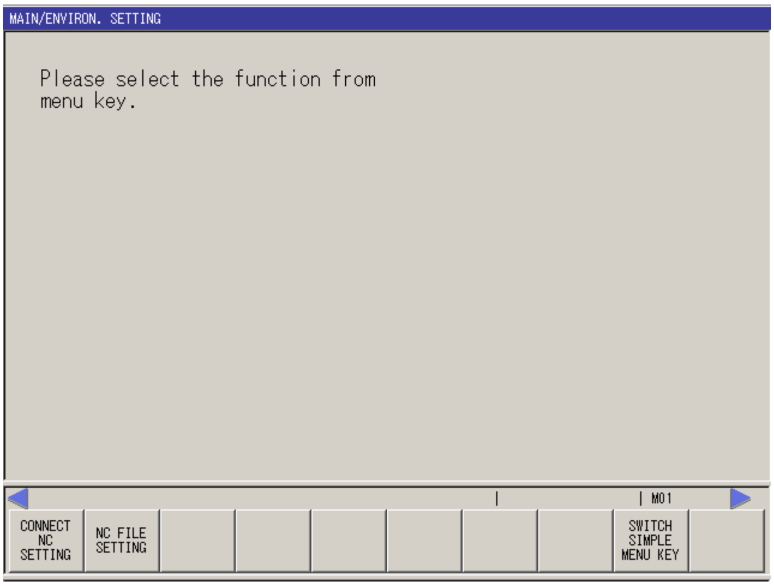
LADDER DISPLAY	This opens the "LADDER DISPLAY" screen.
COMMENT DISPLAY	This opens the "COMMENT DISPLAY" screen.

6. Environment Setting

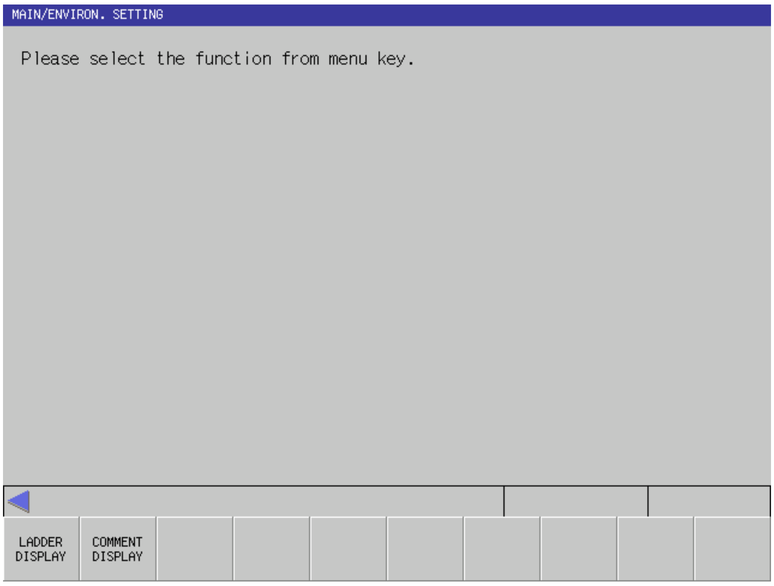
700		70
Standard	Simple	
○		○

Various settings required when the onboard is used are explained below.

■ In 700 Series



■ In 70 Series



(1) "ENVIRON. SETTING" screen display

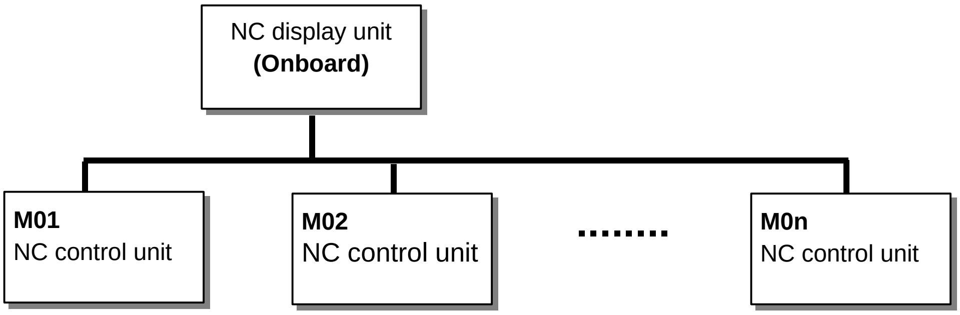
Select "MAIN" → ENVIRON. SETTING menu key. When selected, "ENVIRON. SETTING" screen will be displayed in full size.

When using 700 Series, refer to "6.1 Setting the Connected NC Control Unit", "6.2 NC File Operation Setting" and "6.3 Simple Operation Mode Menu Key Switchover".
When using 70 Series, refer to "6.4 Ladder Display Setting" and "6.5 Comment Display Setting".

6.1 Setting the Connected NC Control Unit

700		70
Standard	Simple	
○		

When NC display unit and NC control unit are connected in the proportion of one display unit to multiple NC control units, the NC unit connected with onboard can be selected. (At the initial start-up of onboard, connect to the same NC unit as HMI screen side.)
Connections of NC control unit can be changed on the "CONNECT NC SETTING" popup screen.
When a connected NC (control unit) is switched to another, the following information will be discarded.
(Note) All the PLC data in the onboard editing area will be discarded. (Program data having unconverted circuit will be discarded, as well.)

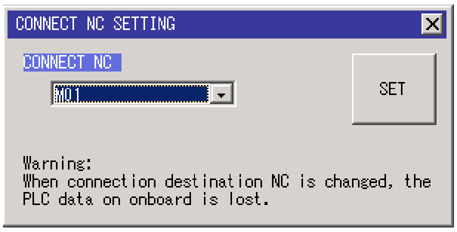


6.1.1 Arbitrary Switchover of Connected NC

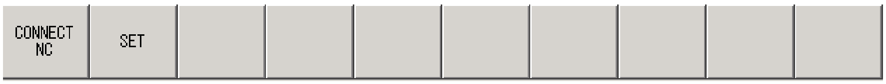
700		70
Standard	Simple	
○		

When multiple NC control units are connected, connection target can be switched freely.

"CONNECT NC SETTING" popup screen



Menus corresponding to popup screen



(1) "CONNECT NC SETTING" popup screen display

Select "MAIN" → ENVIRON. SETTING → CONNECT NC SETTING menu key.

When selected, the "CONNECT NC SETTING" popup screen will appear at the lower center of the screen.

* When the popup screen is displayed, the name of the NC unit currently connected is shown. ("M01" above)
* When connected NC is only one, "SET" menu button will be displayed in gray and disabled.

(2) Switchover of connected NC (control unit)

(a) Select CONNECT NC menu key and specify the name of the NC unit to be connected.

* NC unit name can be selected from the pull-down list.

(b) Select the SET menu key. When selected, "CONNECT NC SETTING" popup screen will be closed, and NC control unit connection switchover will be carried out.

(3) Closing the popup screen

Press the ◁ menu key. Connection switchover will not be carried out.

6.1.2 Information to be updated at connection switchover

When the NC control unit connection destination is changed, the followings on the onboard will be updated.

- According to the NC language parameter, language used for the screen menu display, button and error message will be changed.
- Font used for the comment (statement, note, comment, device name) display will be the one corresponding to the language parameter.
- Alarm on NC side (ROM-Write incomplete) is displayed. (Changes to the status of connected NC)
- Password setting state on the "maintenance" screen of HMI screen (Changes to the status of connected NC)
- PLC RUN/STOP status (Changes to the status of connected NC)
- Connected NC's warning display
- Connected NC's NC name ("M01", etc.)
- Onboard editing area will turn to the (no data) state where PLC data is cleared.

6.2 NC File Operation Setting

700		70
Standard	Simple	
○		

Settings related to NC file operation are explained below.

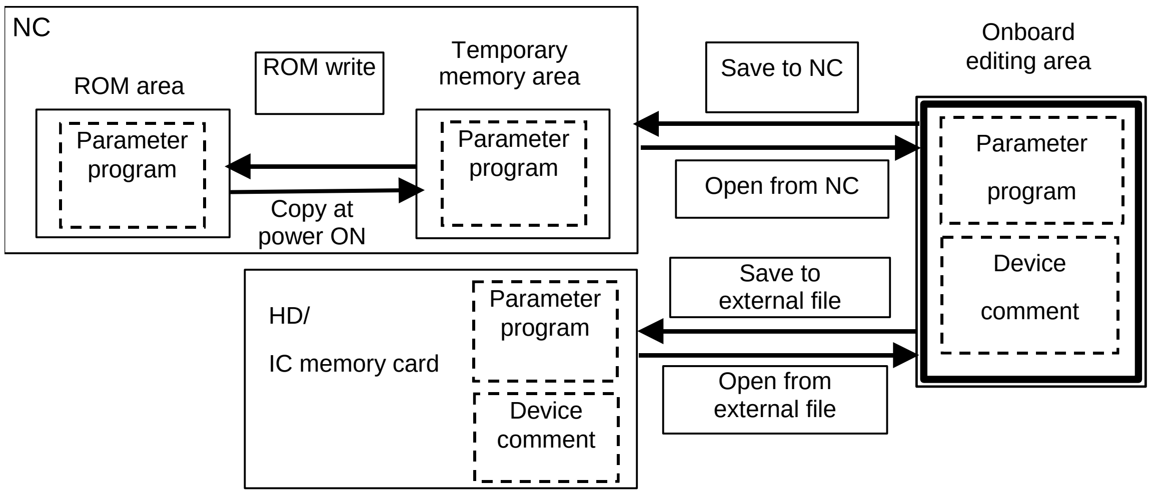
6.2.1 Setting the Storage Destination of Device Comment

When saving into NC temporary memory is not possible due to excessive device comment size, HD and IC memory card can be specified and used as a virtual NC area.

<Application example 1>

NC temporary memory serves as the device comment storage destination. (Normal application)

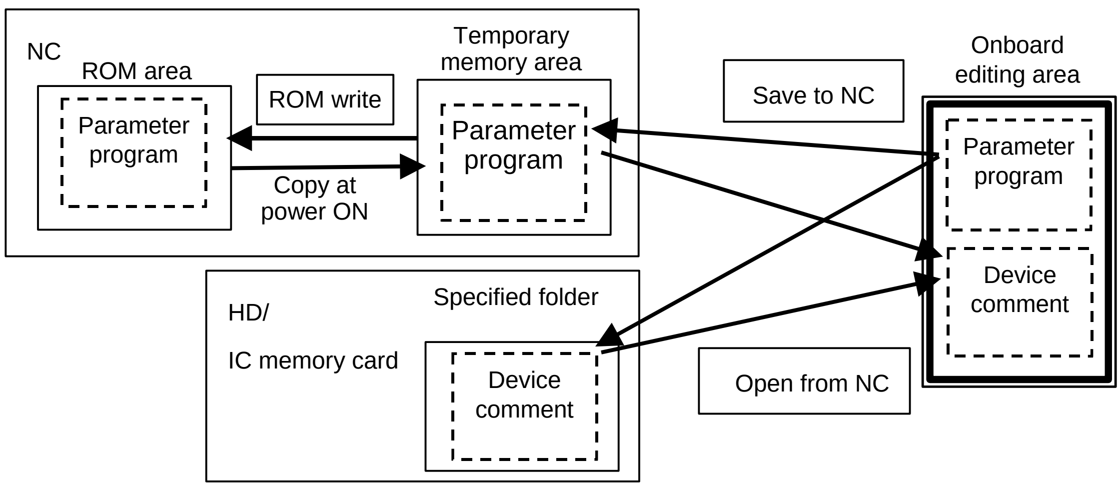
- With NC file operation, save the parameter and program into NC temporary memory and ROM area.
- For saving into HD and IC memory card, external file operation is applied.



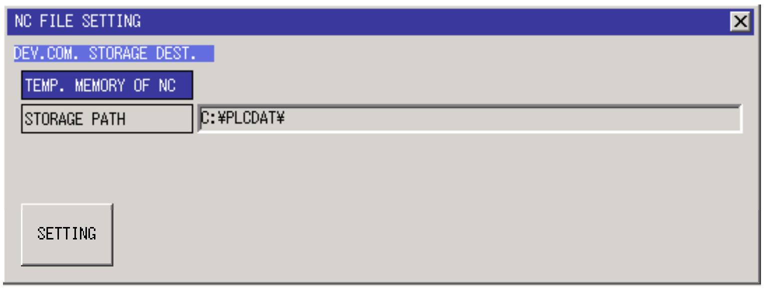
<Application example 2>

Device comment storage destination is specified.

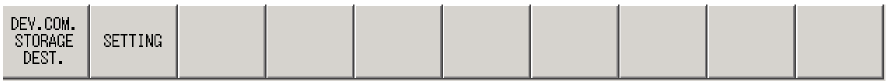
- When saving to the NC with NC file operation, parameter and program will be saved into NC temporary memory, and device comment will be saved into any specified folder.
- When opening from NC with NC file operation, parameter and program will be opened from NC temporary memory, and device comment will be opened from any specified folder.
- *When deleting NC data with NC file operation, delete the NC temporary memory for parameter and program, and the data in specified folder for device comment.
- *When formatting NC temporary memory, the device comment in specified folder will not be deleted.



"NC FILE SETTING" popup screen



Menus corresponding to popup screen



(1) "NC FILE SETTING" popup screen display

Select "MAIN" → "ENVIRON. SETTING" screen, then NC FILE SETTING menu key.

When selected, the "NC FILE SETTING" popup screen will appear at the lower center of the screen.

(2) Setting the device comment storage destination

(a) Select the DEV. COM. STORAGE DEST. menu key.

(b) Press the SET menu key. When pressed, the "NC FILE SETTING" popup screen will close.

* Device comment storage destination set once will be held even after terminating the onboard. When the onboard is started up again, the device comment storage destination is shown in the same state as it was set last.

6.3 Simple Operation Mode Menu Key Switchover

700		70
Standard	Simple	
○		

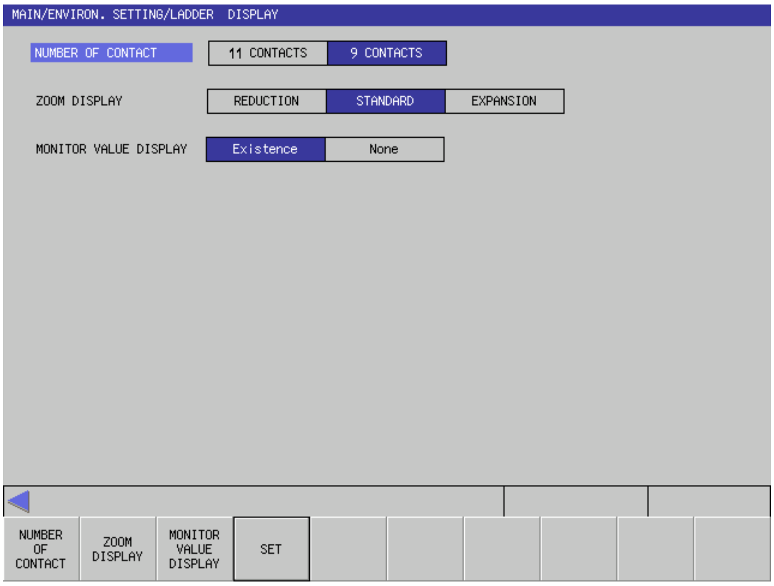
Change to the simple operation mode menu key.

Refer to "2.4 Switching from Standard Operation Mode to Simple Operation Mode" for details.

700		70
Standard	Simple	
		○

6.4 Ladder Display Setting

Ladder display settings are specified on this screen.



6.4.1 Maximum Number of Contacts

The maximum number of contacts is specified. 11 contacts display and 9 contacts display are available.

6.4.2 Zoom Display

The ladder display size is changed.

REDUCTION	Enables the ladder display with 11 contacts.
STANDARD	Enables the ladder display with 9 contacts.
EXPANSION	Cannot display the whole circuit.

For details, refer to "8.4.4 Setting the Circuit Display Scale".

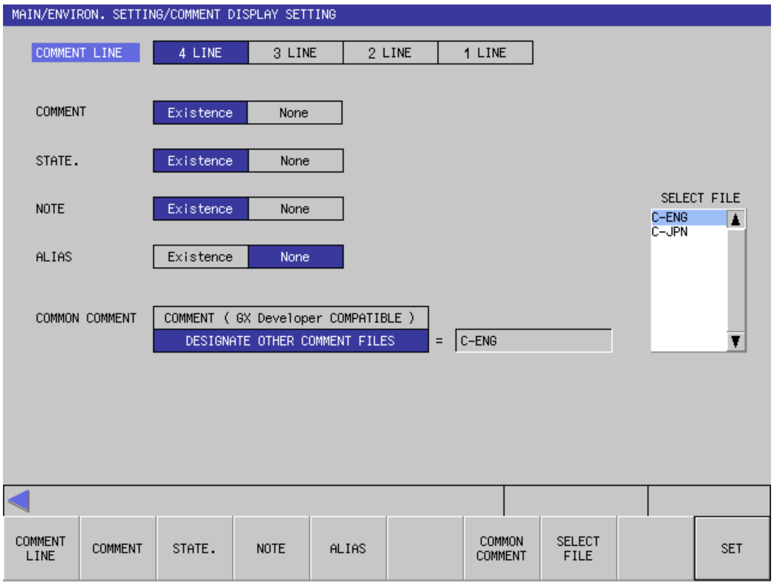
6.4.3 Current Monitor Value Display

The current monitor value display is set.
Setting "None" deletes the line for the current value display.
This helps to increase the density of the ladder display.

6.5 Comment Display Setting

700		70
Standard	Simple	
		○

Comment display settings are specified.



6.5.1 Comment Line

The number of lines is specified to display a device comment.
The characters out of the specified number of lines by this setting are not displayed.

6.5.2 Various Displays

The comment display, statement display, note display and device name display are set valid/invalid.
Refer to "8.4.2 Comment Display" for details.

6.5.3 Common Comment File

Settings for the common comment are specified. Two methods are available to specify a common comment file.

- COMMENT (GX Developer compatible) method
  This method is compatible with GX Developer. The "COMMENT" file, which is common for all the programs, and each comment file, which has the same name as each program file name, are used accordingly to the program displayed.
- Common comment file designation method
  A comment file is designated to be common for each program. This method is used to designate a comment file for each language.

(1) Setting the COMMENT (GX Developer compatible) method
   (a) Select "MAIN" → [ENVIRON. SETTING] → [COMMENT DISPLAY] → [COMMON COMMENT] menu key.
   (b) Select "COMMENT (GX Developer COMPATIBLE)" with the [COMMON COMMENT] menu or the cursor key.

(2) Setting the common comment file designation method
   (a) Select "MAIN" → [ENVIRON. SETTING] → [COMMENT DISPLAY] → [COMMON COMMENT] menu key.
   (b) Select "DESIGNATE OTHER COMMENT FILES" with the [COMMON COMMENT] menu or the cursor key.
   (c) Select the [SELECT FILE] menu. "SELECT FILE" field is focused.
   (d) Select any file with "↑" or "↓" key, and set the file by pressing the [SELECT FILE] menu or the [INPUT] key.

7. Basic Operations

700		70
Standard	Simple	
○		

7.1 Basic Operations 1 (Steps for Creating a Program for the First Time)

The steps (newly creating -> starting the ladder -> saving -> ending) for creating a program for the first time are given below.

Step	Operation Mode 1	Operation Menu button	
Start			
Newly create project data in the onboard	MAIN/	FILE	
		INITIAL	Parameters and the program MAIN are created
Create a circuit program	MAIN/	LADDER	Program is created
	MAIN/LADDER		
Save the program and parameters in the NC's temporary memory	MAIN/	NC FILE	
	MAIN/NC FILE	FORMAT	NC's temporary memory is formatted
		SAVE	
Run the PLC	MAIN/NC FILE	PLC RUN/STOP	Run the PLC
Read the program and parameters saved in the NC's temporary memory into the NC's ROM	MAIN/NC FILE	PLC RUN/STOP	Stop the PLC
		ROM WRITE	When ending, write the data in the temporary memory into the ROM so that the data is not lost even when the power is turned OFF
Save the created program in an external file	MAIN/	EXTERNAL FILE	
	MAIN/EXTERNAL FILE	SAVE PROJECT	
		EXTERNAL SAVE	
	PROJECT		
			Designate the drive and path for saving
End onboard	MAIN/	END	
Finished			

7.2 Basic Operations 2 (Creating, Monitoring and Testing Programs)

700		70
Standard	Simple	
○	○	

The steps for creating, monitoring and testing the program are given below.

<table>
<tr><th rowspan="2">Step</th><th colspan="3">Operation</th><th rowspan="2"></th></tr>
<tr><th>Mode 1</th><th colspan="2">Menu button</th></tr>
<tr><td>Start</td><td></td><td>Standard operation mode</td><td>Simple operation mode</td><td></td></tr>
<tr><td>Newly create project data on the onboard.</td><td></td><td></td><td></td><td>Refer to basic operation 1.</td></tr>
<tr><td>Create the ladder program.</td><td>MAIN/</td><td>LADDER</td><td>LADDER EDIT</td><td></td></tr>
<tr><td></td><td>MAIN/LADDER</td><td>EDIT</td><td></td><td>When no program exists in NC temporary memory, local editing mode (background color is light blue) is applied.</td></tr>
<tr><td></td><td>MAIN/LADDER/EDIT</td><td>EDIT LADDER MODE</td><td></td><td>Creates a program.
<Sample data>
M0
P4002|-|/|----[INC D0]-|</td></tr>
<tr><td></td><td></td><td>CONVERT LADDER</td><td>CONVERT LADDER</td><td>Edited circuit data is confirmed by conversion operation.</td></tr>
<tr><td>Save program and parameter in NC temporary memory.</td><td></td><td></td><td></td><td>Refer to basic operation 1.</td></tr>
<tr><td>Monitor the ladder.</td><td>MAIN/</td><td>LADDER</td><td>LADDER MONITOR</td><td></td></tr>
<tr><td></td><td>MAIN/LADDER</td><td>MONITOR</td><td></td><td>When a program exists in NC temporary memory, NC auto update mode (background color is white) is applied.</td></tr>
<tr><td></td><td>MAIN/LADDER/MONITOR</td><td>START/STOP MONITOR</td><td>START/STOP MONITOR</td><td>When running PLC with <Sample data>, value of D0 will be incremented.</td></tr>
<tr><td>Test the device.</td><td>MAIN/LADDER/MONITOR</td><td>DEVICE TEST</td><td>DEVICE TEST</td><td>Reverse ON/OFF of M0 forcibly.</td></tr>
<tr><td>Go to save operation</td><td></td><td></td><td></td><td></td></tr>
</table>

7.3 Basic Operations 3 (Correcting Programs Stored in NC)

700		70
Standard	Simple	
○		

The steps for correcting the programs stored in the NC's temporary memory are given below.

Step	Operation: Mode 1	Operation: Menu button	
Start			
Open the program and parameters in the temporary memory in the onboard	MAIN/	NC FILE	
	MAIN/NC FILE	OPEN	Open the data in the NC's temporary memory
Change the circuit program	MAIN/	LADDER	When there is a program in the NC's temporary memory, the local editing mode (white background) is activated
	MAIN/LADDER	EDIT	
	MAIN/LADDER/EDIT	PLC RUN/STOP	Stop the PLC to change the program
		EDIT LADDER MODE	Change the program
		CONVERT LADDER	When converted, the data in the NC's temporary memory is also changed
Go to monitor operations			

7.4 Basic Operations 4 (Creating Multiple Programs with Multi-program Method)

700		70
Standard	Simple	
○		

The steps for splitting the program and creating several programs are given below.

Step	Operation: Mode 1	Operation: Menu button	
Start			
Add the circuit program data in the onboard	MAIN/	FILE	
		DATA NEW	Create the 2nd to 4th circuit programs
Set the parameters	MAIN/	PARAM.	
	MAIN/PARAM.	PROGRAM SETTING	Set the circuit program execution type
Change the circuit program to be edited	MAIN/	LADDER	
	MAIN/LADDER	VIEW, TOOLS	
	MAIN/LADDER/VIEW, TOOLS	PROGRAM CHANGE	Create each program while changing the program to be edited.
Go to monitor operations			

<Sample data> (DATA NEW)

Data name	Header statement
MAIN	Machine A
INIT	Initial
LOW	Communication process
TAIKI	Error process

<Sample data> (PROGRAM SETTING)

Program name	Execution type
MAIN	Scan
INIT	Initial
LOW	Low-speed
TAIKI	Standby

7.5 Basic Operations 5 (Creating Device Comments)

700		70
Standard	Simple	
○		

The steps for creating a program device comment are given below.

Step	Operation Mode 1	Menu button	
Start			
Create the comment data in the onboard	MAIN/	FILE DATA NEW	Set a data type comment, and create the device comment MAIN
Create a circuit program			Refer to Basic operations 1
Set the device comment data to be displayed in the circuit program	MAIN/ MAIN/LADDER MAIN/LADDER/ VIEW, TOOLS	LADDER VIEW, TOOLS PROGRAM CHANGE	Set the device comment to be edited as MAIN
Create the comment data for the circuit program device Save the data in the NC's temporary memory	MAIN/LADDER MAIN/LADDER/EDIT	EDIT EDIT COMMENT MODE	Create a comment for the device at the cursor position in the circuit program
Change the comment display ON or OFF state	MAIN/LADDER MAIN/LADDER/VIEW, TOOLS	VIEW, TOOLS COMMENT DISPLAY	Change the comment display ON or OFF state
Go to save operations			

7.6 Basic Operations 6 (Upgrading the Program Version)

700		70
Standard	Simple	
○		

The steps for upgrading the program saved in the NC ROM to the version of the program in the IC card are given below.

Step	Operation Mode 1	Operation Menu button	
Start			
(Program design)			
Create and test the program with GX Developer or onboard			Set the file version, etc., in the header
Write the program file to be upgraded into the IC card			This operation cannot be completed with onboard. Use Windows Explorer, etc.
(Maintenance)			
Turn the NC power ON			The data saved in the NC ROM is read out to the NC's temporary memory
Upgrade the ROM contents to the version of the data in the IC card	MAIN/	NC FILE	
	MAIN/NC FILE	PLC VERSION UP	The programs in the NC ROM and IC card are both displayed The data can be upgraded to only one version
Go to monitor operations			

7.7 Basic Operations 7 (Loading Programs Created with GX Developer)

700		70
Standard	Simple	
○		

The steps for loading and adjusting ladders creating with GX Developer are given below.

Step	Operation: Mode 1	Operation: Menu button	
Start			
Create a program with GX Developer			
Syntax check tool			Correct instructions that cannot be used with NC
Save the program in IC card using GX developer			Write project data created with GX developer into IC card
Open the program in IC card with onboard	MAIN/	EXTERNAL FILE	
	MAIN/EXTERNAL FILE	OPEN PROJECT	Open IC card data
Save the program and parameters in NC's temporary memory			Refer to Basic operations 1
Go to monitor operations			

8. Circuit Operations

The PLC data read out onto the onboard editing area can be edited and monitored with circuit operations. The saving area and editing area are split, so before editing (including monitoring), open the PLC data in the temporary memory in the onboard editing area.

(Note) The circuit menu cannot be selected if even one program is not opened in the onboard editing area. (When moving in the same hierarchical menu, the operation will be skipped.)

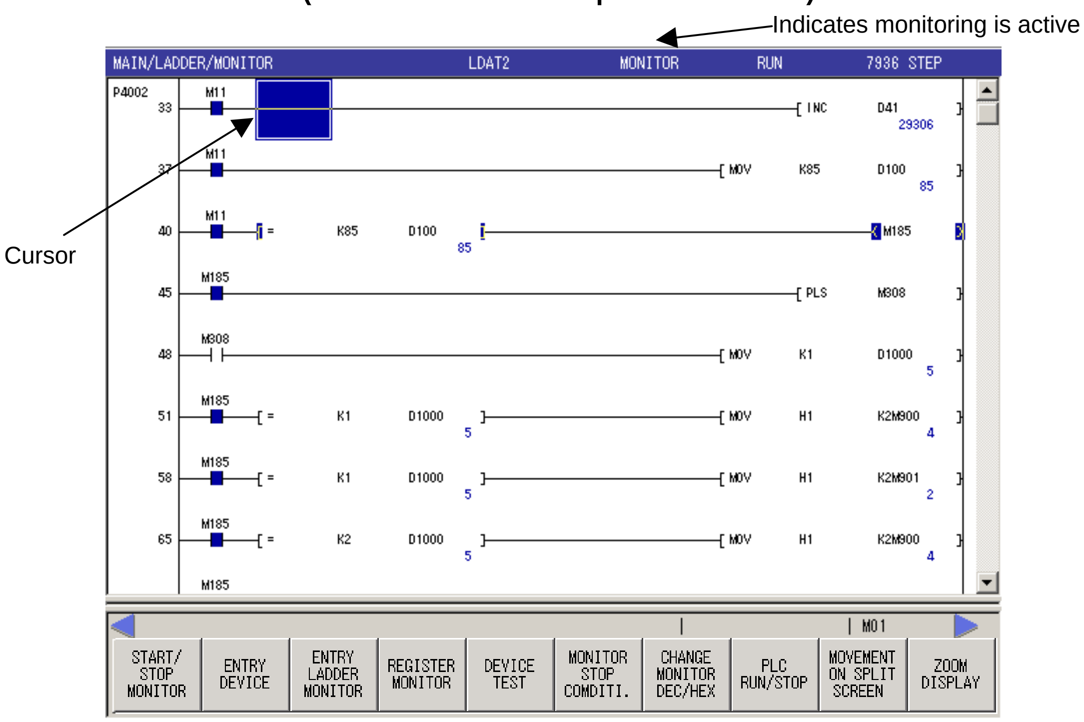
8.1 Monitoring a Program (Ladder)

700		70
Standard	Simple	
○	○	○

The continuity state of contacts and coils can be monitored while displaying the PLC circuits.

(Note 1) Open the PLC data in the temporary memory in the onboard editing area before starting monitoring.
(Note 2) Circuits (programs) in the local editing mode (light blue background) cannot be monitored. (The circuit menu "MONITOR" and "MONITOR" menu will all be displayed in gray and disabled. Movement in the same hierarchy to the "MONITOR" menu will also be skipped.) Monitor the data with a circuit (program) opened from the NC's temporary memory.

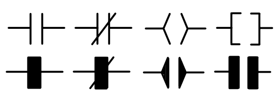
"LADDER" screen (The screen is an example of 700 Series)



(1) Circuit display

During monitoring, the contact and coil ON/OFF state and the device's current value are displayed. These will change according to the PLC operation sate.
The circuit ON/OFF state is shown in the following manner.

OFF
ON



*1
*1

*1: and can be used only for comparative instruction, which is equivalent to the contact and SET, RST, PLS, PLF, SFT and MC, which is equivalent to coils.

(2) Searching for circuits

Move the cursor to the circuit to be searched and press the [INPUT] key or the circuit symbol menu key. A popup screen for searching will open.
Refer to "8.3.1 Searching for Ladder (Simple search)" for details.

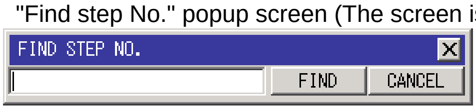
(The screen is an example of 700 Series)
When circuit symbol menu
(**Example :** [-| |-]) is pressed

When [INPUT] key is pressed.

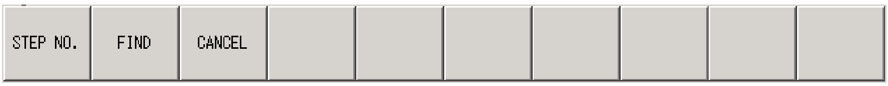
(3) Finding a step No.

When numerical key or [FIND STEP NO.] menu key is pressed, "Find step No." popup screen will open.
Refer to "8.3 Searching and Replacing" for details.

"Find step No." popup screen (The screen is an example of 700 Series)



Menus corresponding to popup screen



8.1.1 Restrictions

The following restrictions apply to the circuit display on the "LADDER DISPLAY" screen.

- One circuit block must be created with 24 or less lines. An error will occur if there are more than 24 lines.
- The maximum number of contacts in one circuit line can be changed with the "LADDER DISPLAY" setting.
- The number of characters displayed in the comment is shown below.

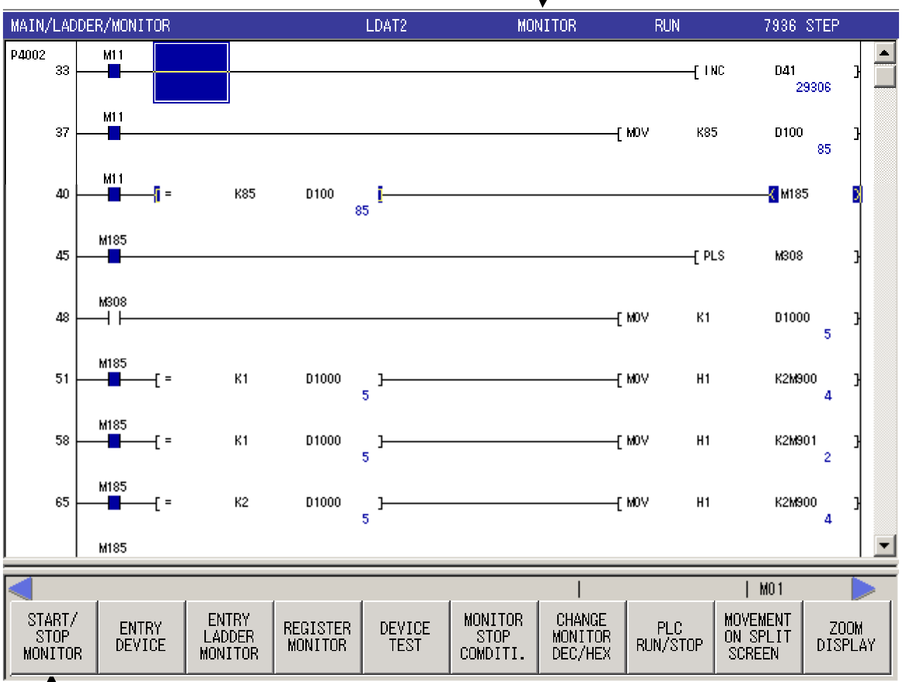
	Number of characters displayed on circuit screen
COMMENT	All characters are displayed with 8 characters on 4 lines
STATE	The set characters are all displayed
NOTE	
ALIAS	

8.1.2 Starting and Stopping Monitoring

700		70
Standard	Simple	
○	○	○

The monitor start and stop state can be changed.
(The screen is an example of 700 Series)

Function mode display



"START/STOP MONITOR" menu keys

(1) Setting the monitor start/stop state

Select
- Standard operation mode: "MAIN" → LADDER → MONITOR
- Simple operation mode: "MAIN" → LADDER MONITOR
- 70 Series: "MAIN" → LADDER MONITOR

; then, press the START/STOP MONITOR menu key to change the monitor start and stop state. The monitor status can be checked with the function mode display.

When MONITOR menu key is pressed in the ladder entry mode or comment input mode, the state will change to the monitor state upon changing the menu keys.

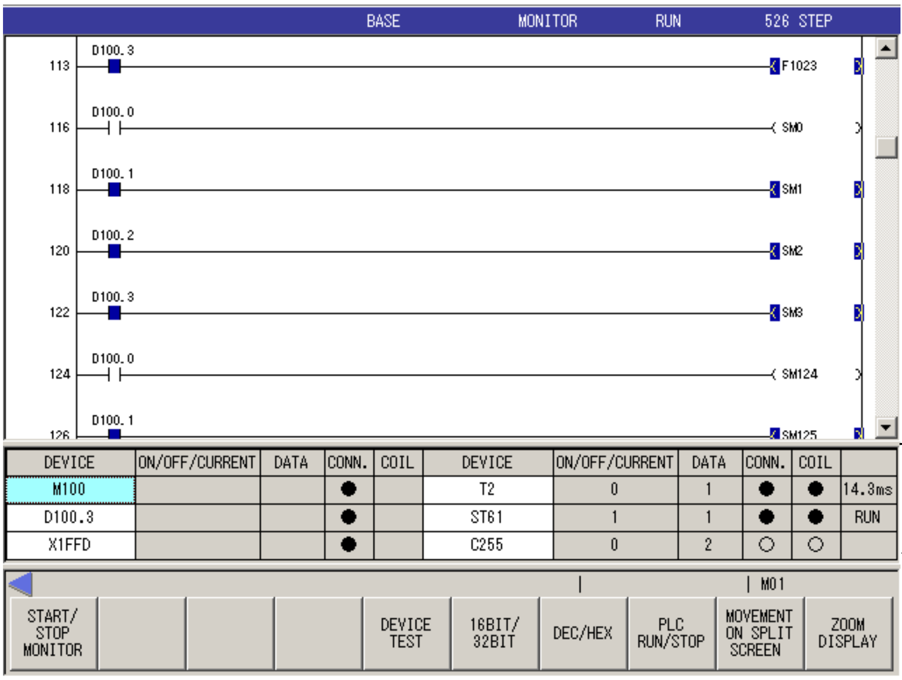
8.1.3 Device Registration Monitor (Split Screens)

700		70
Standard	Simple	
○	○	○

When [ENTRY DEVICE] is pressed, the screen is split, and the "ENTRY DEVICE MONITOR" screen opens. The circuit monitor can be displayed and the device registration can be monitored simultaneously.

(Note) When switching from the NC automatic update mode (white background color) circuit (program) to a local editing mode (light blue background) circuit (program), monitoring of the device registration on the split screen will end. Monitor the device registration for the circuit (program) opened from the NC's temporary memory.

(The screen is an example of 700 Series)



"LADDER" screen

"ENTRY DEVICE MONITOR" screen

(1) "ENTRY DEVICE MONITOR" screen display

Select
- Standard operation mode: "MAIN" → [LADDER] → [MONITOR]
- Simple operation mode: "MAIN" → [LADDER MONITOR]
- 70 Series: "MAIN" → [LADDER MONITOR]

; then, press the [ENTRY DEVICE] menu key to change the ON/OFF status of the "ENTRY DEVICE MONITOR" screen display.

When [◁] menu key is pressed on the "ENTRY DEVICE MONITOR" screen, "ENTRY DEVICE MONITOR" screen will be closed.

* Menus are different between "LADDER" screen (upper part) and "ENTRY DEVICE MONITOR" screen (lower part).

(2) Moving the cursor

Use the [MOVEMENT ON SPLIT SCREEN] menu key to move the cursor on the screen between "LADDER" screen and "ENTRY DEVICE MONITOR" screen. When the screen is not split, the menu key is masked. The [#] key can be used as well for this operation.

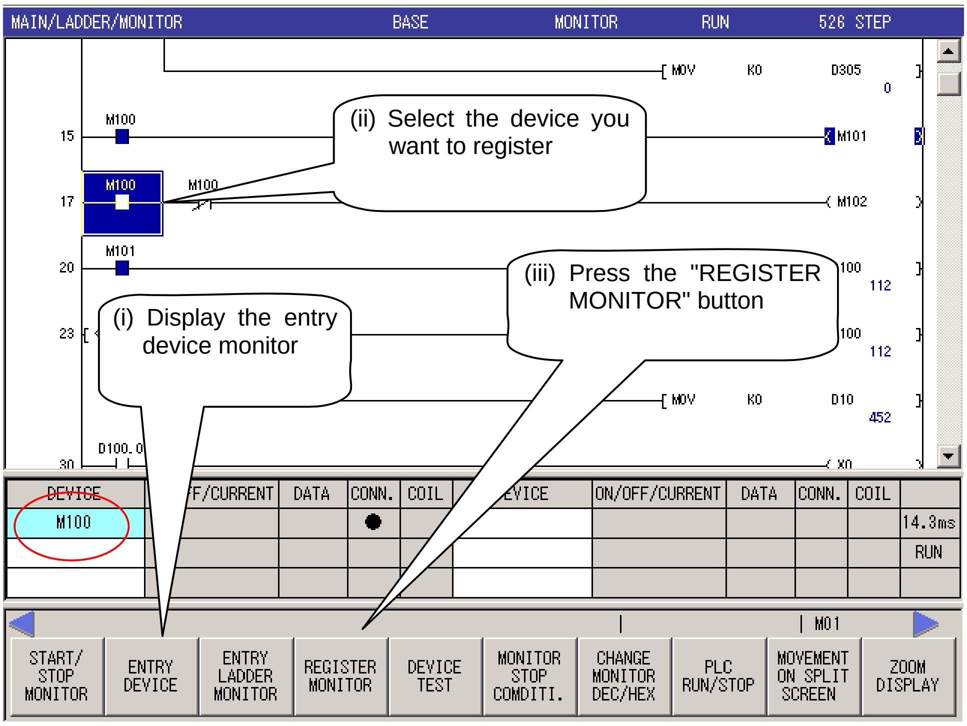
(3) Registering devices

Devices on the "ENTRY DEVICE MONITOR" can be registered in the following two methods.

- Register by using the "REGISTER MONITOR" button on the "LADDER" screen. (Registration is possible only when the "ENTRY DEVICE MONITOR" screen is displayed. When only the "LADDER" screen is displayed, ENTRY LADDER MONITOR is executed.)
- Register directly from the "ENTRY DEVICE MONITOR" screen.

(a) When registering by using the "REGISTER MONITOR" button on the "LADDER" screen.

(The screen is an example of 700 Series)



(i) Press the ENTRY DEVICE button and the "ENTRY DEVICE MONITOR" screen is split and displayed.
* When the "ENTRY DEVICE MONITOR" screen is not displayed, "ENTRY LADDER MONITOR" screen will be displayed. So, always display the "ENTRY DEVICE" screen.

(ii) Align the cursor with the circuit of the device to be registered from the "LADDER" screen (upper half of the screen).

(iii) Press the REGISTER MONITOR button.

(iv) Press the REGISTER MONITOR and the device at the cursor position on "LADDER" will be registered in the "ENTRY DEVICE MONITOR" screen (lower side of the split screen).
(A blank is searched from the row on the left side on the "DEVICE ENTRY" screen, and the device is registered in the first blank found.)

(b) When registering by using the REGISTER MONITOR button on the "LADDER" screen.

(i) Move the cursor to the row of "DEVICE" on the "ENTRY DEVICE" screen.

(ii) Switch to the input mode by using alphanumeric character or INPUT key.

(iii) Enter the device name and press the INPUT key.

(Note) Timer and counter's setting value display shows the program setting value to be edited.

(4) Deleting devices

(a) Move the cursor to the line where the device to delete exists on the "ENTRY DEVICE MONITOR" screen.

(b) Press the DELETE key.

(5) Switching between 16 bit and 32 bit for the device registered on the "ENTRY DEVICE MONITOR".

*Only word device can be changed.

(a) Move the cursor to the "ENTRY DEVICE MONITOR" side.

(b) Press the 16BIT/32BIT button. (For 32 bit, "(D)" is displayed next to the device name. For 16 bits, "(D)" is not displayed.)

(6) Switching between decimal and hexadecimal for the device registered on the "ENTRY DEVICE MONITOR".

(a) Move the cursor to the "ENTRY DEVICE MONITOR" side.

(b) Press the DEC/HEX button.

*Only word device can be changed. (Bit device will not be changed.)
*Current value display on the "LADDER" screen is not changed. (Change with the menus on the "LADDER" screen.)

(7) Close the "ENTRY DEVICE" screen. (The "LADDER" screen will appear on the full screen.)

Press the ◁ menu key while the cursor is located on the "ENTRY DEVICE MONITOR" screen side.

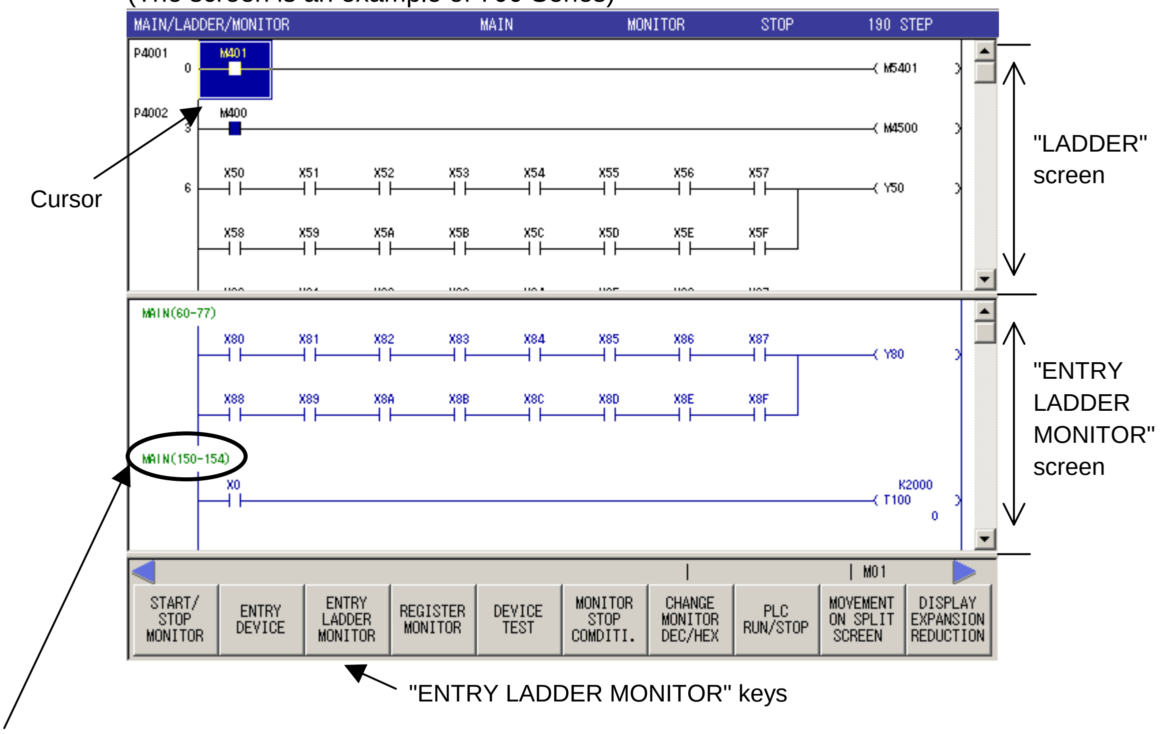
8.1.4 Ladder Entry Monitor (Split Screens)

700		70
Standard	Simple	
○	○	○

When [ENTRY LADDER MONITOR] is pressed, the screen is split, and the "ENTRY LADDER MONITOR" screen opens. The circuit monitor can be displayed and the randomly registered circuits can be monitored simultaneously.

(Note) When switching from the NC automatic update mode (white background color) circuit (program) to a local editing mode (light blue background) circuit (program), monitoring of the circuit registration on the split screen will end. Monitor the circuit registration for the circuit (program) opened from the NC's temporary memory.

(The screen is an example of 700 Series)



The name and number of steps for the program whose ladder has been registered.

(1) "ENTRY LADDER MONITOR" screen display

Select
- Standard operation mode: "MAIN" → [LADDER] → [MONITOR]
- Simple operation mode: "MAIN" → [LADDER MONITOR]
- 70 Series: "MAIN" → [LADDER MONITOR]

; then, press the [ENTRY LADDER MONITOR] menu key to change the ON/OFF status of the "ENTRY LADDER MONITOR" screen display.
The cursor position ("LADDER" screen or "ENTRY LADDER MONITOR" screen) changes the menu display.
When [◁] menu key is pressed on the "ENTRY LADDER MONITOR" screen, "ENTRY LADDER MONITOR" screen will be closed.

(2) Starting and stopping monitoring

Press the [START/STOP MONITOR] menu key.

8.1.5 Registering the Monitor

700		70
Standard	Simple	
○	○	○

Ladder registration with "ENTRY LADDER MONITOR" or device registration with "ENTRY DEVICE MONITOR" can be possible.
"ENTRY LADDER MONITOR" and "ENTRY DEVICE MONITOR" will be as shown below depending on the screen display status.

Screen display status	"REGISTER MONITOR" button	Registration operation
"LADDER" screen	"REGISTER MONITOR" button on the LADDER screen	The ladder specified with the "ENTRY LADDER MONITOR" is registered.
"LADDER" screen	"REGISTER MONITOR" button on each search screen.	Execution is not possible.
Split display of "LADDER" screen and "ENTRY LADDER MONITOR" screen	"REGISTER MONITOR" button on the LADDER screen and each search screen.	The ladder specified with the "ENTRY LADDER MONITOR" is registered.
Split display of "LADDER" screen and "DEVICE REGISTRATION MONITOR" screen	"REGISTER MONITOR" button on the LADDER screen and each search screen.	The device specified with the "DEVICE REGISTRATION MONITOR" is registered.

(1) Ladder registration with "ENTRY LADDER MONITOR"

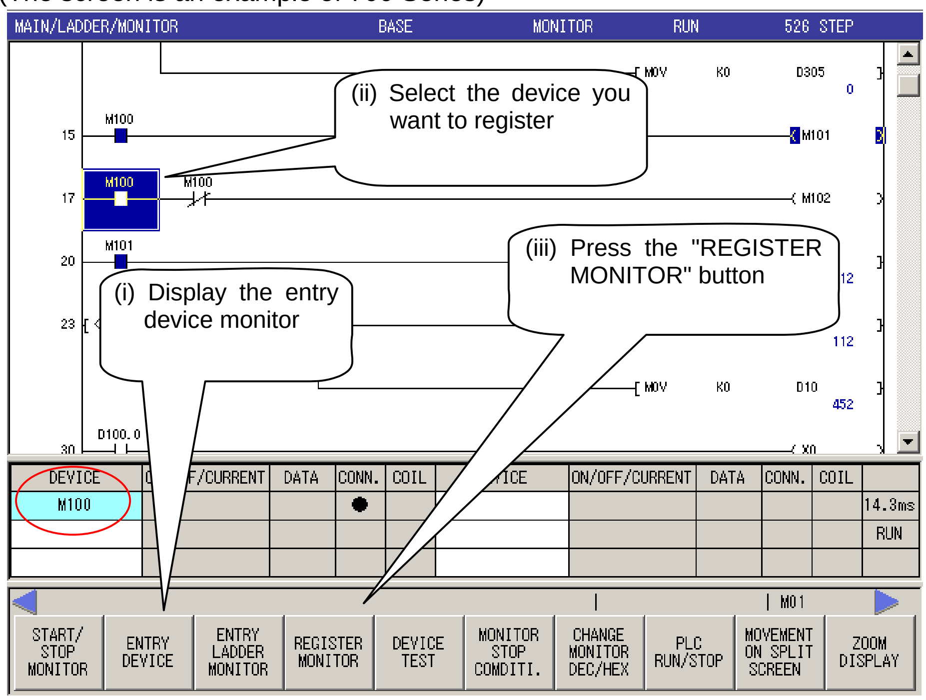
(a) Move the cursor in the "LADDER" screen to the circuit block you wish to register.

(b) Press the REGISTER MONITOR menu key. The circuit block at the cursor position is registered on the "ENTRY LADDER MONITOR" screen.

*The circuit block that you want to register can be searched and selected.
Press the REGISTER MONITOR menu key on the "FIND CONTACT OR COIL", "FIND DEVICE", "FIND INSTRUCTION", "FIND STEP NO." and "FIND STRING" popup screen for registration.

(2) Device registration with "DEVICE REGISTRATION MONITOR"

(The screen is an example of 700 Series)



(a) Press the ENTRY DEVICE MONITOR button to split and display the "ENTRY DEVICE MONITOR" screen.
*When the "ENTRY DEVICE MONITOR" screen is not displayed, "ENTRY LADDER MONITOR" screen will be displayed. So, always display the "ENTRY DEVICE MONITOR" screen.
(b) Point the cursor to the device circuit to be registered from the "LADDER" screen (upper part of a split screen).
(c) Press the REGISTER MONITOR button.
(d) When the REGISTER MONITOR button is pressed, the device on the cursor position in "LADDER" is displayed in the "ENTRY DEVICE MONITOR" screen (lower part of a split screen).

*Press the REGISTER MONITOR menu key on the "FIND CONTACT OR COIL", "FIND DEVICE", "FIND INSTRUCTION", "FIND STEP NO." and "FIND STRING" popup screen for device registration.

(Note that, however, this is only possible only when the "ENTRY DEVICE MONITOR" split screen is displayed.)

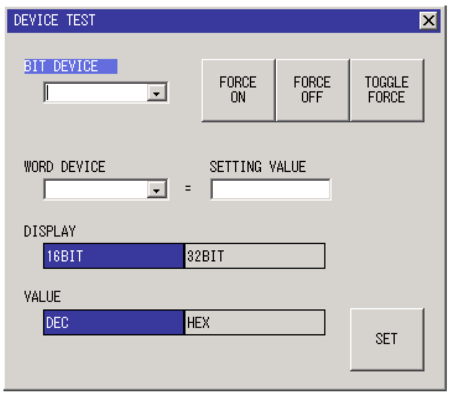
8.1.6 Testing the Devices

700		70
Standard	Simple	
○	○	○

The NC bit devices can be turned ON and OFF forcibly, and the word device's current value can be changed.

■ In 700 Series

"DEVICE TEST" popup screen



Menus corresponding to popup screen

(1) Testing the bit devices

(a) Select
Standard operation mode: "MAIN" → LADDER → MONITOR
Simple operation mode: "MAIN" → LADDER MONITOR
; then, DEVICE TEST menu key.
When selected, the "DEVICE TEST" popup screen will appear at the middle of the screen.
(b) Select the BIT DEVICE menu key, and input the device.
* When "DEVICE TEST" is selected while bit device exists on the cursor position, the device at the cursor position is displayed in "BIT DEVICE".
(c) To turn ON forcibly, select the FORCE ON menu key.
(d) To turn OFF forcibly, select the FORCE OFF menu key.
(e) To toggle between forced ON and OFF, select the TOGGLE FORCE menu key.
* When device test is executed, performance history is displayed in "BIT DEVICE".

(2) Testing the word devices

(a) Select { Standard operation mode: "MAIN" → LADDER → MONITOR ; Simple operation mode: "MAIN" → LADDER MONITOR } ; then, DEVICE TEST menu key.
When selected, the "DEVICE TEST" popup screen will appear at the middle of the screen.

(b) Select the DISPLAY menu key, and set the device's display format.

(c) Select the VALUE menu key, and set whether the device setting value is a decimal or hexadecimal.

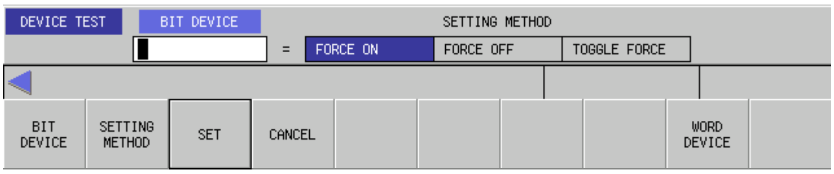
(d) Select the WORD DEVICE menu key, and input the device.
* When "DEVICE TEST" is selected while word device exists on the cursor position, the device at the cursor position is displayed in "WORD DEVICE". When multiple devices exist on the cursor position, they are displayed in the combo box; word device can be selected from there.

(e) Select the SETTING VALUE menu key, and input the value to set in the device.

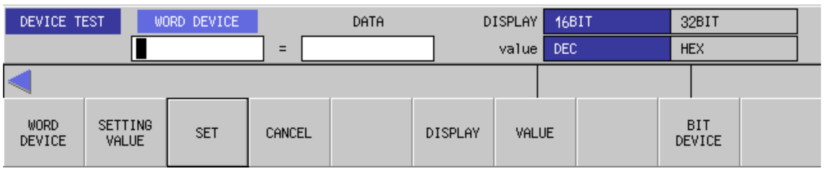
(f) Select the SET menu key, and change the word device's current value.
* When device test is executed, performance history is displayed in "WORD DEVICE".

■ In 70 Series

"DEVICE TEST" BIT DEVICE screen



"DEVICE TEST" WORD DEVICE screen



(1) Testing the bit devices

(a) Select "MAIN" → RADDER MONITOR then, DEVICE TEST menu key.
When "DEVICE TEST" is selected while bit device exists on the cursor position, the bit device screen where the device at the cursor position was displayed is displayed.

(b) Select the BIT DEVICE menu key, and input the device.

(c) To turn ON forcibly, select the FORCE ON menu key.

(d) To turn OFF forcibly, select the FORCE OFF menu key.

(e) To toggle between forced ON and OFF, select the TOGGLE FORCE menu key.

(Note 1) When WORD DEVICE menu key is pressed, the screen changes to the WORD DEVICE screen.

(2) Testing the word devices

(a) Select "MAIN" → RADDER MONITOR then, DEVICE TEST menu key.
When "DEVICE TEST" is selected while word device exists on the cursor position, the word device screen where the device at the cursor position was displayed is displayed.
(b) Select the DISPLAY menu key, and set the device's display format.
(c) Select the VALUE menu key, and set whether the device setting value is a decimal or hexadecimal.
(d) Select the WORD DEVICE menu key, and input the device.
(e) Select the SETTING VALUE menu key, and input the value to set in the device.
(f) Select the SET menu key, and change the word device's current value.

(Note 1) When BIT DEVICE menu key is pressed, the screen changes to the BIT DEVICE screen.

8.1.7 Changing the Current Value Monitor

700		70
Standard	Simple	
○	○	○

The current value displayed while monitoring the "LADDER" screen can be changed from decimal to hexadecimal and vice versa.

(Note) This function is valid only during monitoring.

(1) Changing the value

Select
- Standard operation mode: "MAIN" → LADDER → MONITOR
- Simple operation mode: "MAIN" → LADDER MONITOR
- 70 Series: "MAIN" → LADDER MONITOR

; then, CHANGE MONITOR DEC/HEX (10/16) menu key.

8.1.8 Movement on Split Screen

700		70
Standard	Simple	
○	○	○

Cursor is moved between split screens.

(1) Moving the cursor

Use the MOVEMENT ON SPLIT SCREEN menu key to move the cursor on the screen between "LADDER" screen and "ENTRY DEVICE MONITOR" screen, "ENTRY LADDER MONITOR" screen. When the screen is not split, the menu key is masked. The # key can be used as well for this operation.
When the "ENTRY LADDER MONITOR" screen is displayed in 100%, and the cursor is moved to the "LADDER" screen side, the split ratio will be changed to 50%, and the cursor stays on the "LADDER" screen side.

8.1.9 Searching

700		70
Standard	Simple	
○	○	○

Refer to "8.3 Searching and Replacing" for details.

8.1.10 Deleting All the Entry Ladders

700		70
Standard	Simple	
○	○	○

All the circuit registered with "ENTRY LADDER" function is deleted.

(1) Deleting all the entry ladders

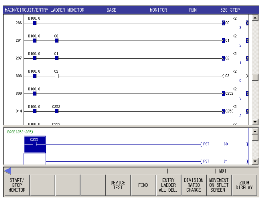
Select the ENTRY LADDER ALL DEL. menu key.
When executed, the following confirmation message appears.
"Deleting all the registered ladder. OK?　YES/NO"

8.1.11 Changing the Split Ratio

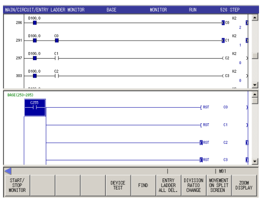
700		70
Standard	Simple	
○	○	○

The split ratio to display "ENTRY LADDER MONITOR" screen and "LADDER" screen is changed.
When "DIVISION RATIO CHANGE" menu key is pressed, the split ratio changes to "25%" → "50%" → "75%" → "100%" → "25%".
The default split ratio is 50%.
The split ratio set once will be held until ending the onboard.

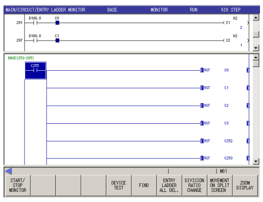
(The screen is an example of 700 Series)



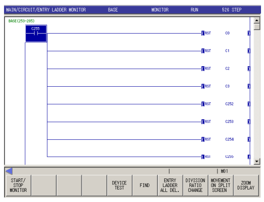
25%



50%



75%



100%

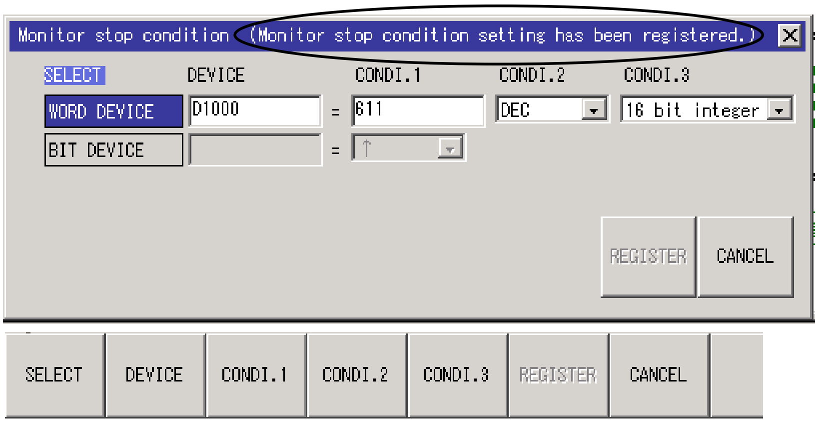
8.1.12 Setting the Monitor Stop Conditions

700		70
Standard	Simple	
○		

The conditions for stopping the "LADDER" screen monitor can be set.

Displays the validity of stop conditions

"Monitor stop condition" popup screen



Menus corresponding to popup screen

(1) Setting the monitor stop conditions for the device

(a) Select Standard operation mode: "MAIN" → LADDER → MONITOR ; then, select the MONITOR STOP CONDITI. menu key. When selected, "Monitor stop condition" popup screen will appear at the middle of the screen.

(b) Select the SELECT menu key, and select the word device or bit device.

(c) Select the DEVICE menu key, and input the device.

(d) Select the CONDI.1 , CONDI.2 , CONDI.3 menu keys, and set the stop conditions. (Only condition 1 is set for the bit device.)

(e) Select the REGISTER menu key, and register the stop conditions. When registered, the "Monitor stop condition" popup screen will automatically close.

(2) Canceling the monitor stop condition settings

(a) Select Standard operation mode: "MAIN" → LADDER → MONITOR ; then, select the MONITOR STOP CONDITI. menu key. When selected, the "Monitor stop condition" popup screen will appear at the middle of the screen.

(b) The monitor stop condition validity is displayed as follows when the monitor stop conditions are set.

Status	Display
No monitor stop conditions	Only screen title
Monitor stop conditions set	Screen title (Monitor stop condition setting has been registered)

(c) Select the CANCEL menu key. The monitor stop condition setting display will disappear.

(3) When monitor stop conditions are established

(a) A popup screen indicating "The monitor condition has been established" will appear.

(b) This popup screen will close when the INPUT key is pressed.

(4) Closing the popup screen

Press the ◁ menu key.

8.2 Editing

The PLC data circuit opened on the onboard editing area can be written, inserted and deleted.
During standard operation mode, PLC data must be retrieved from the NC or external device and opened in the onboard editing area to edit the data.
The range of editable PLC data in the simple operation mode is more and 70 Series limited than that of the standard operation mode.
Editable PLC data in each mode is shown below.

	700 Series		70 Series
	Standard operation mode	Simple operation mode	
Sequence program (Ladder)	○	○	○
PLC message data	○	×	○
Device comment	○ (Alphanumerical characters only)	×	×
Statement	○ (Alphanumerical characters only)	○	○
Note	○ (Alphanumerical characters only)	○	○

(Note 1) After editing the data, save it in the NC or external device. If not saved, the edited details will be lost.
(Note 2) If the password has not been released, the CONVERT LADDER button will appear in gray and will be disabled.

8.2.1 Changing to Circuit Editable Screen

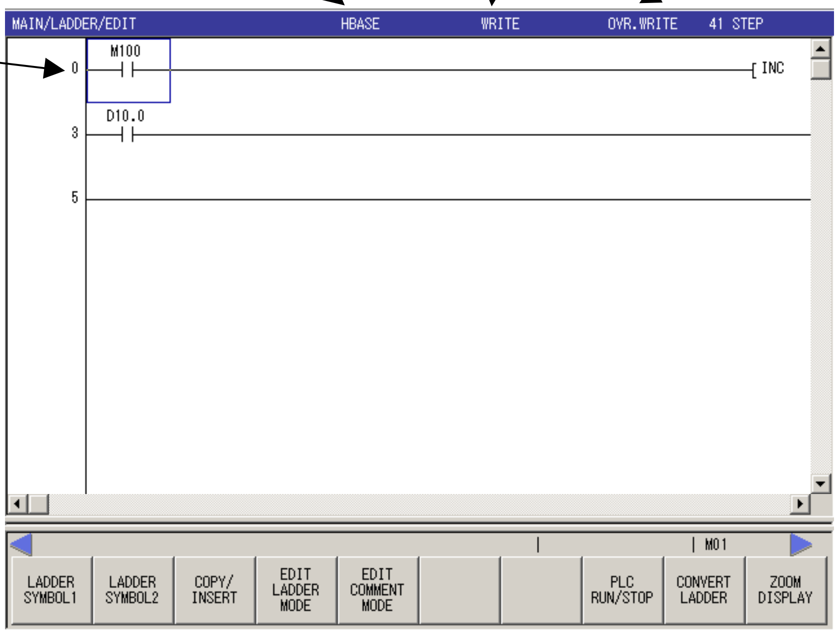
700		70
Standard	Simple	
○	○	○

(1) Changing to the circuit editable mode

Select
Standard operation mode: "MAIN" → LADDER → EDIT EDIT LADDER MODE
Simple operation mode: "MAIN" → LADDER EDIT
70 Series: "MAIN" → LADDER EDIT

to change the state to the circuit editable state. The current mode can be checked at the title bar.

(The screen is an example of 700 Series)



In standard operation mode, when the "EDIT" button under "LADDER" menu is pressed while the MONITOR mode is currently selected, the mode will be automatically changed to the EDIT LADDER MODE.

Before EDIT button in the ladder menu is pressed	After EDIT button in the ladder menu is pressed
MONITOR	Changed to EDIT LADDER MODE.
MONITOR STOP	Changed to EDIT LADDER MODE.
EDIT LADDER MODE (OVR.WRITE/INSERT)	Not changed (stays in EDIT LADDER MODE.)
EDIT COMMENT MODE	Not changed (stays in EDIT COMMENT MODE.)

(2) Moving the cursor

The cursor on the screen can be moved with the up, down, left and right arrow keys.
The cursor can also be moved in circuit units (ignoring cross bars) by pressing the |← and →| keys.

(3) Moving the screen

If the cursor is moved when it is at the top or bottom of the screen, the screen will follow the cursor and move.

To move the screen in page units, press the ⇧ and ⇩ keys.

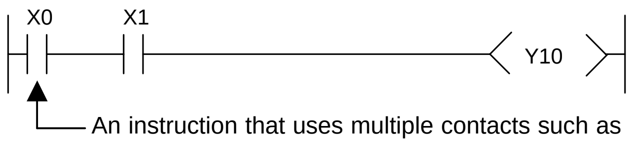
8.2.2 Restrictions

The restrictions which apply for editing circuits on the "LADDER DISPLAY" screen are shown below.

- Up to 24 lines can be edited in one circuit block.
- Up to 24 lines in one block and up to 48 lines in total can be edited.
- The maximum number of contacts in one circuit line can be changed with the "LADDER DISPLAY" setting.
- The master control (MC) symbol cannot be edited. The MC symbol is displayed during circuit monitoring. (The MC symbol does not appear during circuit editing.)
- If a series circuit exceeding the maximum number of contacts is created in one line, the line will automatically return and move to the next line. The return symbol is created with K0 to K99. The OUT (→) and IN (>-) return symbol No. always have to be the same.
- Another circuit cannot be inserted between the OUT (→) line and IN (>-) line of the return line.
- When writing the circuit, the return symbol is assigned with a serial number even if it is not in the same circuit block. Note that in the circuit block read out with the read function, the return number is assigned in order from No. 0.

- If the contact and coil to be overwritten extends over several contacts, the circuit cannot be edited with the write (overwrite) mode.

(Example)

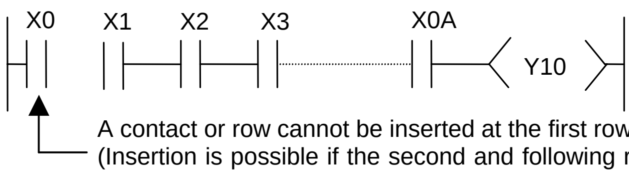


An instruction that uses multiple contacts such as "-[= D0 D1]-" at this position cannot be overwritten with the overwrite mode. (This is because the next instruction will also be overwritten.)

To make the above type of revision, insert "-[= D0 D1]-" beforehand with the write (insertion) mode, and then delete "LD X0" with the DELETE key.

- If a return results from the insertion of a contact in the first row of the circuit, the contact cannot be inserted.

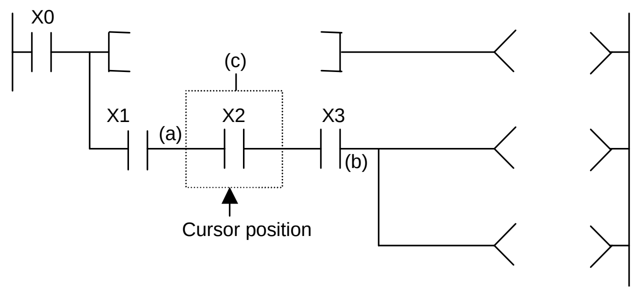
(Example)



A contact or row cannot be inserted at the first row of the circuit.
(Insertion is possible if the second and following rows are empty and a return will not result from the insertion.)

- Insertion of the circuit symbol is processed by right alignment and row insertion, so there may be causes when the symbol cannot be inserted because of the circuit shape.

  (Example)



(a) If the following conditions are established when inserting the circuit symbol at position (a), the message "Editing position is incorrect" will result, and the circuit will not be inserted.
There is no space between (a) and (b)
Insertion into row (c) is not possible

(Note) The position (b) is the closest position to the cursor position among the branch symbols and the coil-equivalent instructions.

- If one circuit block has two or more lines, and the instruction does not fit on one line, return the instruction before inputting it.
- Create the number of steps for one circuit block within approx. 4k steps. The NOP instruction in the circuit block is also included in the number of steps. The NOP instruction between circuit blocks is irrelevant.
- If there is an unconverted circuit, the screen movement range may be limited.

8.2.3 Inputting a Circuit

700		70
Standard	Simple	
○	○	○

A circuit can be written in or inserted.

(Note 1) The MITSUBISHI CNC compatible instructions (instructions usable only with MITSUBISHI CNC) are also checked when the circuit is input.

(1) Switching between circuit overwrite and insertion

The circuit can be edited by overwriting or by inserting.
Change between these methods with the INSERT key in standard operation mode, with the OVR.WRITE/INSERT menu key in simple operation mode. The overwrite and insertion state can be confirmed with "Mode 2" or cursor shape on the title bar.

(2) Inputting circuits (excluding cross bars, vertical bars and labels)

(a) Move the cursor to the position to input the circuit.

(b)

Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

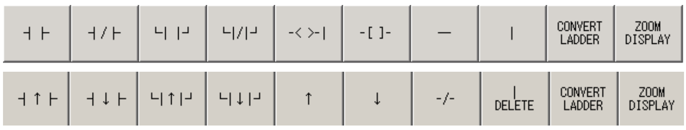
; then, select the LADDER SYMBOL 1 or LADDER SYMBOL 2 menu key. When selected, the "ENTER SYMBOL" popup screen will appear.

* In 700 Series, the "ENTER SYMBOL" popup screen appears as shown below so that it does not overlap the cursor line on the LADDER DISPLAY screen.

When cursor in LADDER screen is on upper half of screen	Displayed on lower part of screen
When cursor in LADDER screen is on lower half of screen	Displayed on upper part of screen

In 70 Series, the popup screen appears at the bottom of the circuit screen.

Circuit symbol 1 menus
Circuit symbol 2 menus

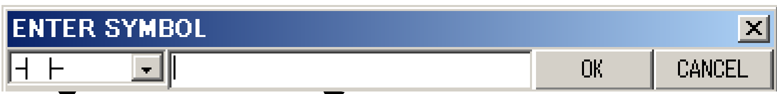


* These menus are shown in 700 Series. Some menus appear different in 70 Series.

* Some of the instructions in circuit symbol 2 are available only in the instruction extension mode.

* Instruction extension mode is set with bit selection 6452:BIT1.

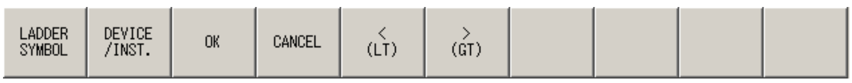
(c) Input the instruction on the "ENTER SYMBOL" popup screen.
(The screen is an example of 700 Series)



Circuit symbol selected with menu key

Input instruction (device) * Refer to circuit input patterns.

Menus corresponding to popup screen



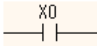
The < and > keys may not exist on the keyboard for some machine types. Use "<",">" menu button instead when inputting an instruction that includes "<",">".

(d) Select the OK menu key to set the input circuit.

(3) Circuit input patterns

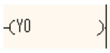
The basic input patterns for inputting circuits are shown below.

- **Inputting contact instructions**

(Example) For 

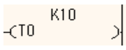
Circuit symbol	Instruction
-| |-	Device name **(Example)** X0

- **Inputting coil instructions**

(Example) For 

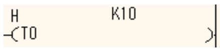
Circuit symbol	Instruction
-< >-|	Device name **(Example)** Y0

- **Inputting timer and counter coil instructions**

(Example) For 

* [SP] indicates a space code

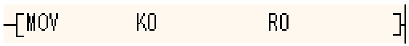
Circuit symbol	Instruction
-< >-|	Device [SP] device **(Example)** T0 K10

(Example) For 

* [SP] indicates a space code

Circuit symbol	Instruction
-< >-|	Instruction [SP] device [SP] device **(Example)** H T0 K10

- **Inputting function instructions**

(Example) For 

* [SP] indicates a space code

Circuit symbol	Instruction
-[ ]-	Instruction [SP] device [SP] device **(Example)** MOV K0 R0

- **Inputting labels**

(Example) For P4002

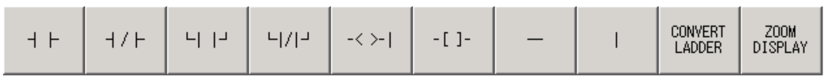
Circuit symbol	Instruction
(Empty)	Device **(Example)** P4002

(4) Inputting vertical bars and cross bars

(a) Move the cursor to the input position.

(b) Select
- Standard operation mode: "MAIN" → [LADDER] → [EDIT]
- Simple operation mode: "MAIN" → [LADDER EDIT]
- 70 Series: "MAIN" → [LADDER EDIT]

; then, select the [LADDER SYMBOL 1] [—] and [|] menu keys.

LADDER SYMBOL 1 menus 

(c) A vertical bar or cross bar is input at the cursor position.

(5) Deleting a circuit

(a) Move the cursor to the position to delete.

(b) Select the [DELETE] key. The circuit at the cursor position will be deleted.

(6) Deleting a vertical bar

(a) Move the cursor to the upper right of the vertical bar to be deleted.

(b) Select { Standard operation mode: "MAIN" → [LADDER] → [EDIT]
Simple operation mode: "MAIN" → [LADDER EDIT]
70 Series: "MAIN" → [LADDER EDIT] }

; then, select the [LADDER SYMBOL 2] → [| DELETE] menu keys.

LADDER SYMBOL 2 menus

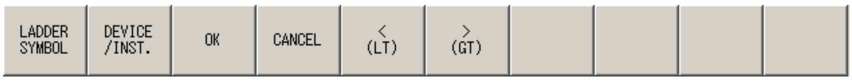
(c) The vertical bar at the lower left of the cursor will be deleted.

(7) Inputting a label

(a) Move the cursor to the position to input, and press the [INPUT] key.

(b) Input the label No. in the instruction field of the "ENTER SYMBOL" popup screen. (The circuit symbol does not need to be input.)

Menus corresponding to popup screen

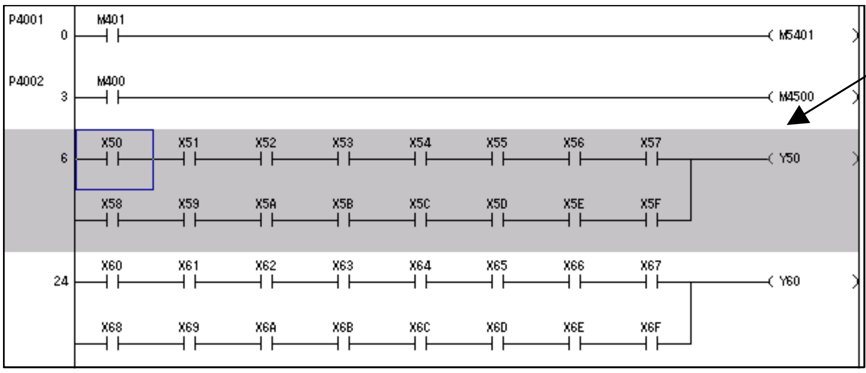


The [<] and [>] keys may not exist on the keyboard for some machine types. Use "<",">" menu button instead when inputting an instruction that includes "<",">".

(c) Select the [OK] menu key and set the input circuit.

(8) Displaying the revised circuit block

The circuit revised with edited operations is displayed as a circuit block unit with a gray background.



Circuit block being revised

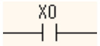
(9) Inputting circuits with list format

A list instruction character string can be directly input and edited. (Normally, the circuits should be input with the method given in (2) Circuit symbols.)

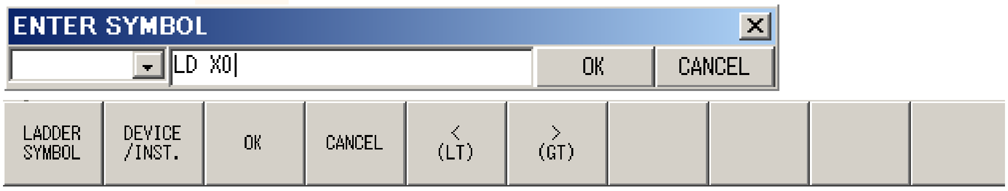
(a) Move the cursor to the position to input the circuit.

(b) When the [INPUT] key is pressed, the "ENTER SYMBOL" popup screen (with blank circuit symbol) will appear.

(c) Input the list instruction character string.

(Example) For 

Use a space code to delimit the instructions and devices



The [<] and [>] keys may not exist on the keyboard for some machine types. Use "<",">" menu button instead when inputting an instruction that includes "<",">".

(d) Select the [OK] menu key, and set the input circuit.

8.2.4 Inserting a Line

700		70
Standard	Simple	
○	○	○

A new line can be inserted at the cursor position to create a circuit.

(1) Inserting a line

(a) Move the cursor to the position to insert the line.

(b) Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select the COPY/INSERT and INSERT LINE menu key.

(c) A new line will be inserted above the cursor.

8.2.5 Deleting a Line

700		70
Standard	Simple	
○	○	○

The line where the cursor is at can be deleted.

(1) Deleting a line

(a) Move the cursor to the position to delete.

(b) Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select the COPY/INSERT and DELETE LINE menu key.

(c) The line where the cursor is will be deleted.

8.2.6 Designating the Range

700		70
Standard	Simple	
○	○	○

The range can be designated as a circuit block unit or as a circuit unit.

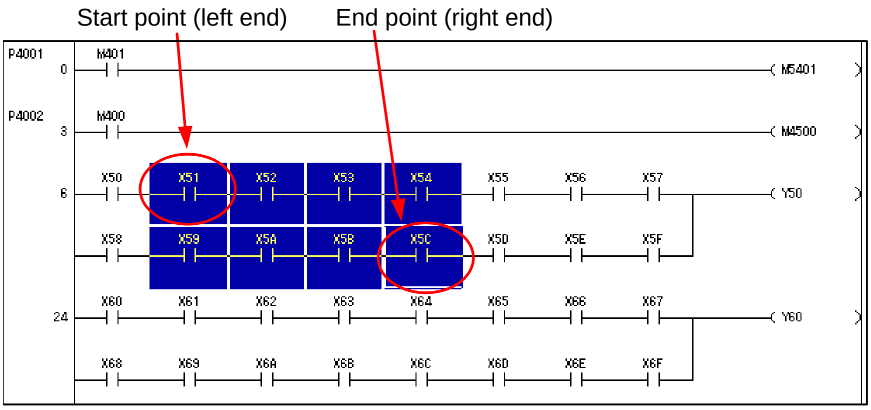
(1) Designating the range for a circuit unit

(a) Move the cursor to the left end of the circuit to be designated.

(b) Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select the COPY/INSERT and MARK menu key.

(c) Next, move the cursor to the right end of the range to be designated. The background of the selected range will change to blue.



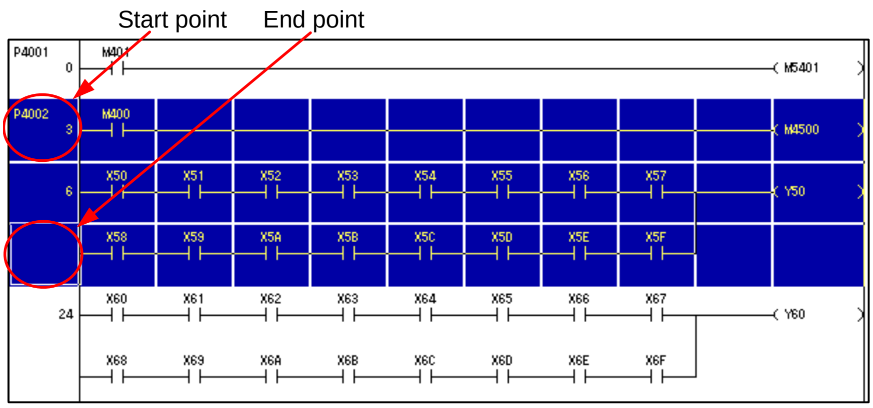
(2) Designating the range for a circuit block unit

(a) Move the cursor to the left end (position where label No. and step No. are displayed) of the circuit block to be designated.

(b) Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select the COPY/INSERT and MARK menu key.

(c) Move the cursor up and down, and move to the circuit block to be designated. The background of the selected range will change to blue.



(Note 1) In the NC automatic update mode (LADDER screen background is white), batch delete and copy & paste cannot be carried out on a circuit which has been designated.

(3) Canceling the range designation

(a) When the background of the selected range is blue,

Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select the COPY/INSERT and MARK menu key again.

(b) The background of the selected range will return to the original color.

8.2.7 Deleting in a Batch

700		70
Standard	Simple	
○	○	○

The circuits can be deleted in a batch.

(1) Deleting in a batch

(a) Refer to "8.2.6 Designating the Range", and designate the circuit range to delete.

(b) Press the DELETE key. The circuits in the designated range will all be deleted.

*When the range is the circuit block unit, conversion is also executed automatically.

(Note 1) When the background color of LADDER screen is white (auto renewal mode), a batch deletion in the circuit block unit is not possible. (An error message appears.)

8.2.8 Copy & Paste

700		70
Standard	Simple	
○	○	○

The circuits can be copied and pasted at another position or in another program.

(1) Copying

(a) Refer to "8.2.6 Designating the Range", and designate the range of circuits to be copied.
(b) Press the COPY menu key.

* If necessary memory could not be secured in the copied circuit, the error is displayed and the selected area to be copied is cleared. The error message "it is over the circuit size which can be copied" is displayed.

(Note 1) When a background color of LADDER screen is white (auto renewal mode), copy & paste in the circuit block unit is not possible. (An error message appears.)

(2) Pasting

(a) Move the cursor to the position to paste into.

* The PASTE key is displayed in gray until a circuit is copied.

* To paste to another program, change to that program.

(b) Press the PASTE menu key.

The paste operations differs according to the overwrite and insertion modes.

8.2.9 Converting a Program

700		70
Standard	Simple	
○	○	○

The program conversion operation differs according to the mode.

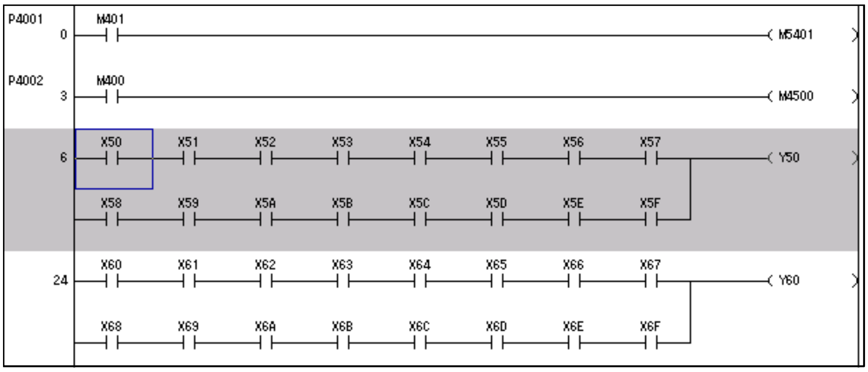
- NC automatic update mode
- Local editing mode

(1) Conversion operation for NC automatic update mode (LADDER screen background color is white) (700 Series/ 70 Series)

The background color is white if the program was opened from the temporary memory into the onboard editing area using "NC FILE", or when a "LADDER" for a program temporarily saved from the onboard editing area is displayed. After this type of program is converted, the circuit will automatically be saved in the program with the same name in the temporary memory.

(Note 1) Only the program currently being edited is updated.
(Note 2) The temporary memory is lost when the NC power is turned OFF. Refer to "12.4 Writing the Temporary Memory's PLC Data to the ROM", and always save the data on the ROM. (The message "ROM-Write incomplete" appears in the message area until the data is saved in the ROM.)
(Note 3) If the password has not been released, the CONVERT LADDER button will appear in gray and will be disabled.
(Note 4) Up to 512 steps can be converted at once.
(Note 5) When PLC is in RUN, a message appears to confirm whether to stop PLC to execute the operation.
When PLC is not stopped, programs will not be automatically written into the temporary memory. (An error message appears.)
After the conversion has been successfully completed, a message confirming whether to have PLC returned to the RUN state is displayed.
(Note that, however, if the conversion is executed during PLC STOP, this message will not be displayed.)
(Note 6) The circuit cannot be converted if it exceeds the temporary memory's capacity.) (An error will appear.)

Converting the NC automatic update mode (LADDER screen background is white) screen

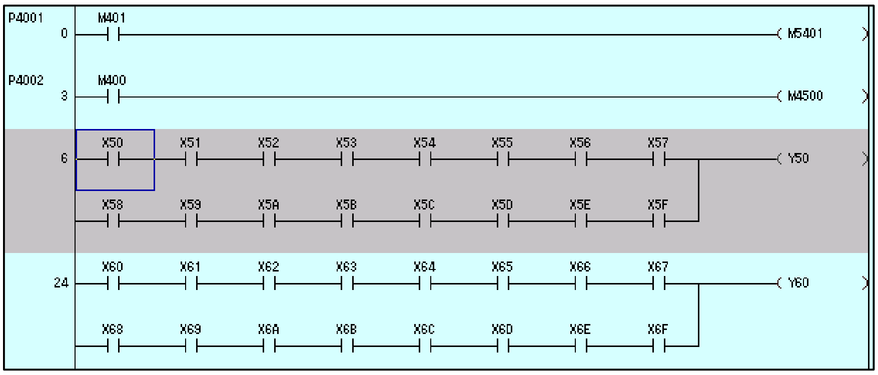


(2) Conversion operation for local editing mode (LADDER screen background is light blue) (700 Series)

The background color is light blue if the program was opened from an external device, or if a program to which data was added, the data name was changed or the program was initially set with the "11. FILE" operations is displayed on the "LADDER" screen. If this program is converted, only the program in the onboard editing area will be converted. (The program will not be automatically written into the temporary memory.)
To save the program in the temporary memory, save it with the "NC FILE" operations.

(Note 1) The program is not saved in the temporary memory after conversion.

Converting the local editing mode (LADDER screen background is light blue) screen



(3) Conversion method

Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select the CONVERT LADDER menu key, or

Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT
- 70 Series: "MAIN" → LADDER EDIT

; then, select
- LADDER SYMBOL 1
- LADDER SYMBOL 2
- COPY/INSERT

and the CONVERT LADDER menu key.

* The circuit block being edited will be converted. (The gray background will change to white when the data is correctly converted.)

* It may take slightly longer when saving to the temporary memory.

8.2.10 Editing a Statement

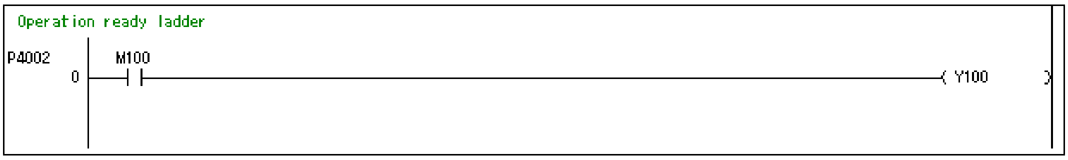
700		70
Standard	Simple	
○	○	○

Statements can be added to the program displayed on the "LADDER" screen.
A statement is character string data added to each circuit block to make it easy to understand the flow of the entire program.

(Note 1) If a user PLC containing an integrated statement is saved in the NC, a large user PLC memory will be required.
(Note 2) Only alphanumeric characters can be input on the onboard.

Comment type	Display	Edit	Input character range	Storage destination data type
Statement	○	○	64 one-byte alphanumeric characters	Program

- Example of interlinear integrated statement display



(1) Displaying a statement

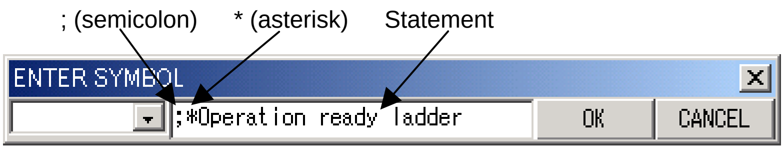
To display a statement, validate the statement display as explained in "8.4.2 Comment Display".

(2) Switching between integrated statement and peripheral statement.

Statements include integrated types and peripheral types. The integrated type and peripheral type is switched with the head character.

Head character is * (asterisk)	Peripheral statement
Head character is not an * (asterisk)	Integrated statement

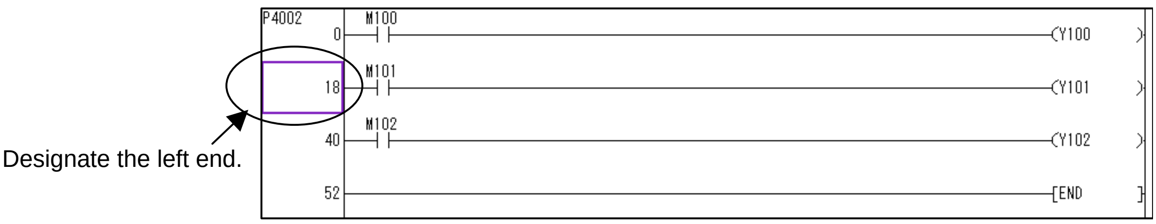
An example of inputting the peripheral statement is shown below.



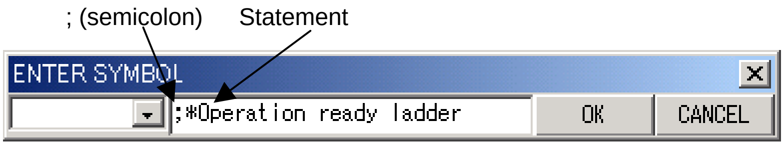
"ENTER SYMBOL" popup screen

(3) Inputting an interlinear statement

(a) Press "MAIN" → LADDER → EDIT and EDIT LADDER MODE menu key, and activate the circuit input mode.
(b) Press the INSERT key, and activate the insertion mode.
(c) Move the cursor on the "LADDER" screen to the left end of the position to input.



(d) Press the INPUT key, and the "ENTER SYMBOL" popup screen will appear.
(e) Input the statement on the "ENTER SYMBOL" popup screen. Add a ; (semicolon) at the head when inputting.

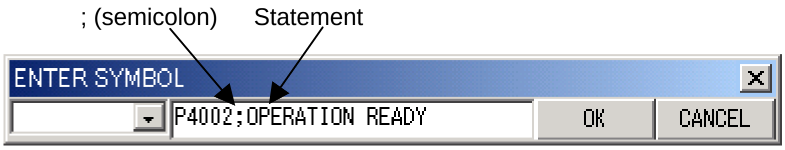


"ENTER SYMBOL" popup screen

(f) After setting the input, press the INPUT key or OK menu key. The statement will appear between the cursor lines.
(g) Select the CONVERT LADDER menu to complete the editing.

(4) Inputting a P statement

(a) Press "MAIN" → LADDER → EDIT and EDIT LADDER MODE menu key, and activate the circuit input mode.
(b) Move the cursor on the "LADDER" screen to the P label to be input, and then press the INPUT key. The "ENTER SYMBOL" popup screen will appear.
(c) Input a statement on the "ENTER SYMBOL" popup screen. Input the label, semicolon (;) and then statement.



(d) After setting the input, press the INPUT key or OK menu key. The P label statement will appear.
(e) Select the "CONVERT LADDER" menu to complete the editing.

700		70
Standard	Simple	
○	○	○

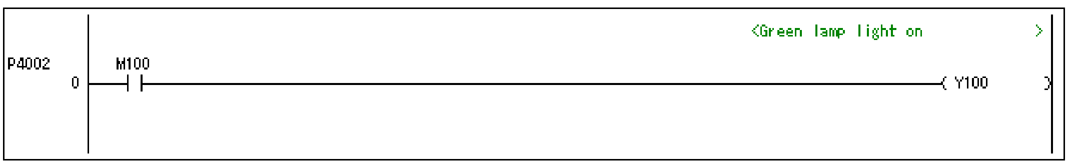
8.2.11 Editing a Note

Notes can be added to the program displayed on the "LADDER" screen.
A note, just like a statement, is character string data added to each coil and function instruction to make it easy to understand the flow of the entire program.

(Note 1) If a user PLC containing an integrated note is saved in the NC, a large user PLC memory will be required.
(Note 2) Only alphanumeric characters can be input on the onboard.

Comment type	Display	Edit	Input character range	Storage destination data type
Note	○	○	32 one-byte alphanumeric characters	Program

- Example of integrated note display



(1) Displaying a note

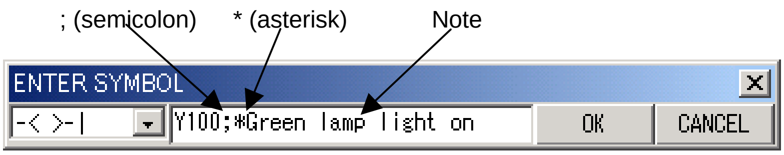
To display a note, validate the note display as explained in "8.4.2 Comment display".

(2) Switching between integrated note and peripheral note.

Notes include integrated types and peripheral types. The integrated type and peripheral type is switched with the head character.

Head character is * (asterisk)	Peripheral note
Head character is not an * (asterisk)	Integrated note

An example of inputting the peripheral note is shown below.



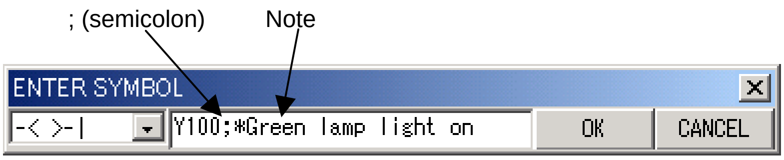
"ENTER SYMBOL" popup screen

(3) Inputting a note

(a) Press "MAIN" → LADDER → EDIT and EDIT LADDER MODE menu key, and activate the circuit edit mode.

(b) Move the cursor on the "LADDER" screen to the position of the coil or function instruction to be input, and press the INPUT key. The "ENTER SYMBOL" popup screen will appear.

(c) Input a note on the "ENTER SYMBOL" popup screen. Add a ; (semicolon) following the instruction when inputting.



"ENTER SYMBOL" popup screen

(d) After setting the input, press the INPUT key or OK menu key. The note will appear at the coil or function instruction at the cursor position.

(e) Select the CONVERT LADDER menu to complete the editing.

8.2.12 Editing a Comment

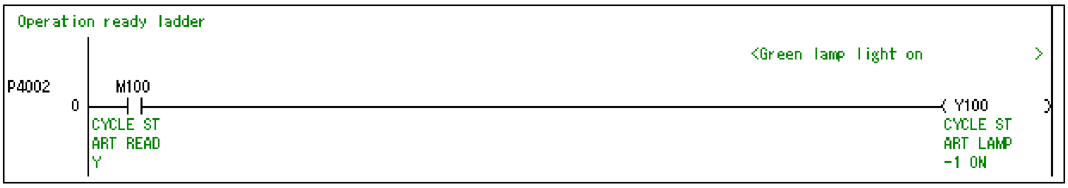
700		70
Standard	Simple	
○		

Comments can be added to each device. The program is easier to understand when meanings are assigned to the devices. Refer to "4.2 Types of Data" for details on the data.
The validity of comment display and editing, the input character range, types of data that can be saved, and a display example are given below.

(Note 1) Only alphanumeric characters can be input on the onboard.

Comment type	Display	Edit	Input character range	Storage destination data type
Comment	○	○	32 one-byte alphanumeric characters	Device comment

- Example of comment display



(1) Designating the device comment data containing comment to edit

The comment is stored in the device comment data file. Refer to "7.4.1 Changing Data (Program, Device Comment)", before editing, and designate the device comment data.

(2) Displaying a comment

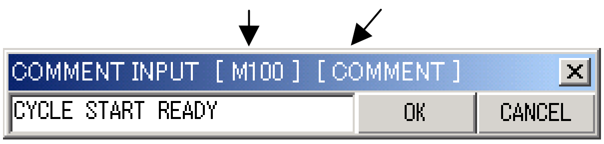
To display a comment, validate the comment display as explained in "8.4.2 Comment Display".

(3) Editing a comment

(a) Press "MAIN" → LADDER → EDIT and EDIT COMMENT MODE menu key. The line spacing on the "LADDER" screen will increase so that comments can be input.
(b) Move the cursor to the instruction for which a comment is to be input, and press the INPUT key. The popup screen for editing the comment will appear.
(c) Input a comment on the "COMMENT INPUT" popup screen.
(d) After setting the input, press the INPUT key or OK menu key. The comment will appear at the selected circuit.
(e) If there are multiple devices in the instruction, the process will shift to the input of the comment for the next device.

(Example) To add a comment to the X0 device on the "LADDER" screen

Device name   Device comment data name



"COMMENT INPUT" popup screen

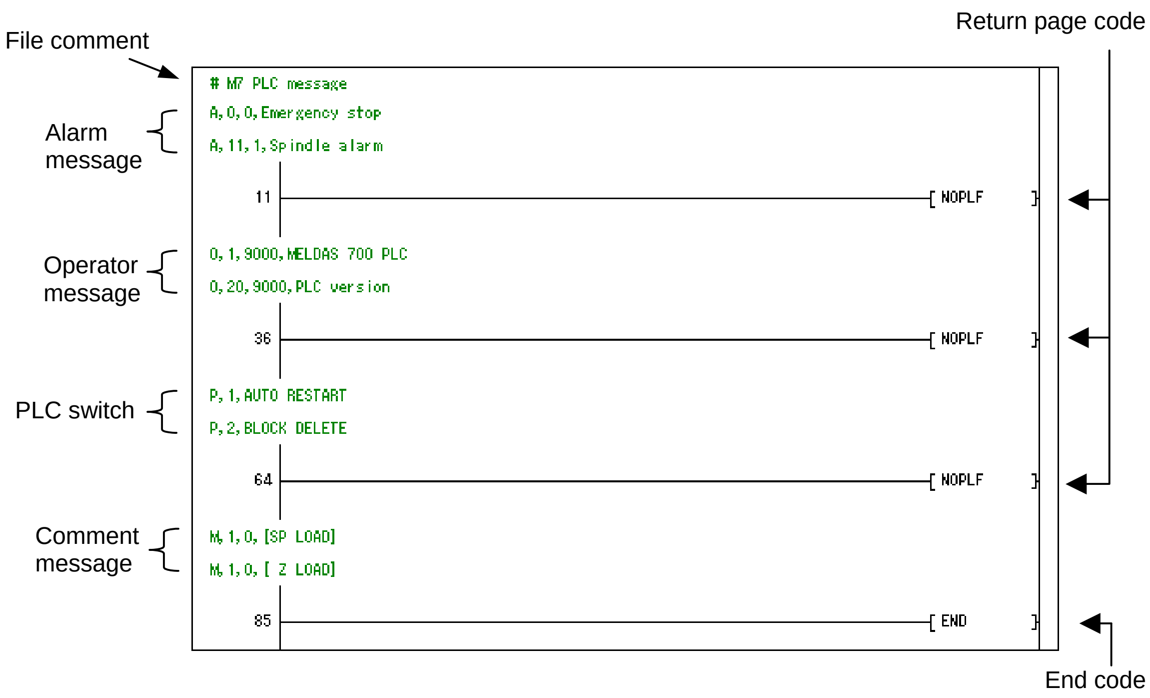
8.2.13 Editing a PLC Message

700		70
Standard	Simple	
○		○

The PLC messages (alarm messages, operator messages, PLC switches, comments) can be edited. These messages are created as programs with reserved names (example: M1xxxxx). The user PLCs, statements and notes cannot be written into this program.
Characters other than the alphanumeric character (Roman figure (ex. Ⅲ) etc.) can be edited by inputting the character-code.

(1) PLC message data description format

The PLC message data is created using the interlinear integrated statement editing function.



• Alarm, operator, PLC switch and comment messages

Details	Each message is described. One message is described on a line.
Description format	Alarm message ;A, "Index No. ", "data register No. ", "message character string" **(Example)** ;A,0,0,Emergency stop ;A,11,1,Spindle alarm
	Operator message ;O, "Index No. ", "data register No. ", "message character string" **(Example)** ;O,1,9000,MELDAS 700 PLC ;O,20,9000,PLC version
	PLC switch ;P, "Switch No. ", "message character string" **(Example)** ;P,1,AUTO RESTART ;P,2,BLOCK DELETE
	Comment message ;M, "Index No. ", "device No. ", "message character string" **(Example)** ;M,1,0,[SP LOAD] ;M,1,0,[Z LOAD]

• File comment

Details	The integrated statement which starts with a character other than A, O, P or M can be used as a file comment.
Description format	;"File comment" **(Example)** ;# M7 PLC message

• Return page code

Details	This is the circuit function instruction NOPLF instruction. This is created as a circuit instruction instead of a statement. **(Note)** One or more return page codes is created every 15 lines of setting areas or messages. (The message data could be skipped if a return code is not described.)

• End code

Details	This is the circuit END instruction, and is normally created automatically.

• Range of each message data input characters

Message type code	A: Alarm message, O: Operator message, P: PLC switch, M: Comment message
Index No.	One-byte numbers (0 to number of messages in setting area -1)
Switch No.	One-byte numbers (0 to number of messages in setting area -1)
Data register No.	One-byte numbers
Device	One-byte number (1 or 2)
Device No.	One-byte numbers (0 to 10)
Message character string	Alphanumeric character or other characters (Input by character-code (Unicode)) **(Note)** To display characters other than the alphanumeric character, the correspondence language should be able to be displayed.
; (semicolon)	Statement input code
, (comma)	Described element delimiter (only commas can be used to set a blank in the message character string)

• Maximum value of each message data

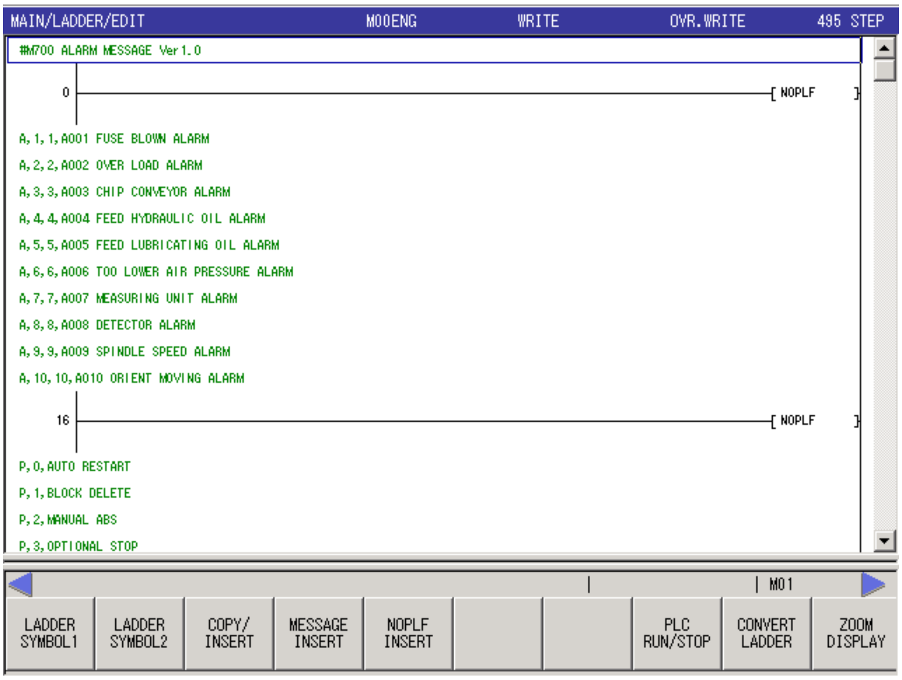
Message type	Maximum message length	Maximum number
Alarm message	32 Byte	1024 messages
Operator message	60 Byte	512 messages
PLC switch	14 Byte	32 messages
Comment messages	60 Byte	100 messages

(2) Designating the PLC message to edit

Refer to "8.4.1 Changing Data (Program, Device Comment)" for designating the PLC message.

(3) Displaying the PLC message

When the PLC message is set to be edited, the message editing screen is displayed as follows. No setting is needed to validate the display of statements or comments.



(4) Inputting one-byte alphanumeric characters in the PLC message

(a) Move the cursor to the message to edit, and press the INPUT key. The "ENTER SYMBOL" popup screen, as shown below, appears for editing the message.

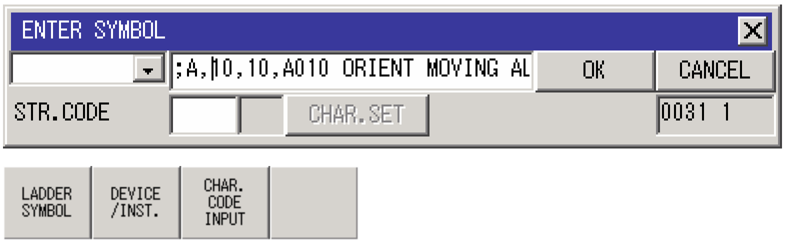
(b) Move the cursor to any position and input characters in the "DEVICE/INST" field on the "ENTER SYMBOL" popup screen.

(c) After confirming the input characters, press the INPUT key or the OK menu key. The edited message is displayed in the circuit.

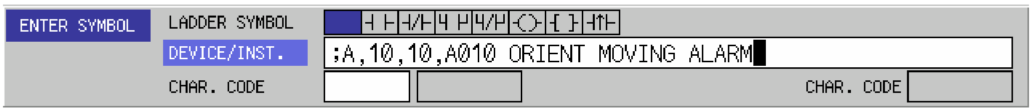
(d) Select the CONVERT LADDER menu key to complete the editing.

"ENTER SYMBOL" popup screen dedicated to messages

■In 700 Series



■In 70 Series



(5) Using character codes to input characters in the PLC message

(a) Move the cursor to the message to edit, and press the INPUT key. The "ENTER SYMBOL" popup screen, as shown below, appears for editing the message.

(b) Move the cursor to any position in the "DEVICE/INST" field on the "ENTER SYMBOL" popup screen. Then the right-bottom field in the "ENTER SYMBOL" popup screen displays the character code of the character at the cursor.

(c)Click the CHAR. CODE INPUT menu key and input a character code in the "STR. CODE" ("CHAR. CODE") entry field. The character corresponding to the entered character code is displayed. Pressing the ↑ / ↓ keys and page up/down keys can change the character code.

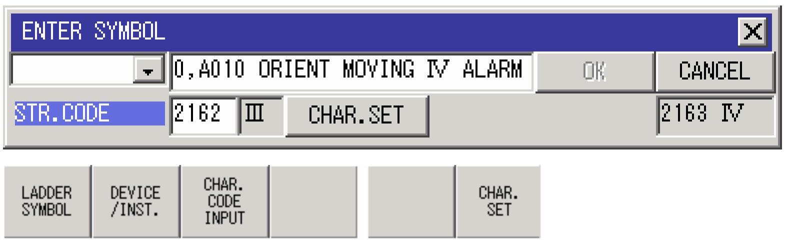
(d) Click the CHAR. SET menu key. The cursor position in the "DEVICE/INST." field displays the character as set in the "STR. CODE" ("CHAR. CODE") field. When canceling the input with character code, click the DEVICE/INST. menu key to move the cursor back to the "DEVICE/INST." field.

(e) After confirming the input characters, press the INPUT key or OK menu key. The edited message is displayed in the circuit.

(f) Select the CONVERT LADDER menu key to complete the editing.

"ENTER SYMBOL" popup screen dedicated to messages

■In 700 Series



■In 70 Series

8.2.14 Undoing the Last Editing Operation

700		70
Standard	Simple	
○	○	○

Cancels the last editing operation with circuit input mode and restores the previous state.
Undo operation is possible for only one most recent edit.
Menu is masked when undo operation is not possible.

(1) Procedures for undo operation

There are two ways of undo operations.

(a) Method with menu key selection:

Select
- Standard operation mode: "MAIN" → LADDER → EDIT
- Simple operation mode: "MAIN" → LADDER EDIT

COPY/INSERT - or -

Select
- 70 Series: "MAIN" → LADDER EDIT → LADDER SIMBOL 1 / LADDER SIMBOL 2 / COPY/INSERT

and then UNDO menu key.

(b) Press Ctrl, then Z key.

(2) Restorable edits and unrestorable editing

Restorable editing:
- Editing contacts/coils/lines etc. (add • change • delete)
- Editing statement/note
- Line insertion • line deletion
- Pasting in one circuit unit

Unrestorable editing:
- When a circuit is changed, the state prior to the change cannot be restored.
- When an unconverted circuit is discarded, the state prior to discarding cannot be restored.
- Editing with replacement, change of AB contacts or TC setting value
- Editing device comment with comment input mode
- Editing in a split display of a circuit

When pasted in a circuit block unit, the state prior to pasting cannot be restored.

8.3 Searching and Replacing

Contacts, coils, instructions, step Nos. and character strings can be searched for while editing and monitoring circuits. In addition, AB contacts can be changed, and devices, etc., can be replaced.
This search function is also used when selecting the circuit to register with the circuit register monitor.

(Note) If a circuit block is being edited in the editing mode, the following confirmation popup screen will appear.
"Unconverted circuit found. Okay to abort unconverted circuit?"
If YES is selected, the unconverted circuit will be aborted.
If NO is selected, the search will not be executed. (The unconverted circuit will remain.)

8.3.1 Searching for Ladder (Simple search)

700		70
Standard	Simple	
○	○	○

Move the cursor to the circuit to be searched and press the "INPUT" key, alphanumerical key, or ladder symbol menu key, and then popup screen for searching will appear.

(1) Displaying "FIND" screen

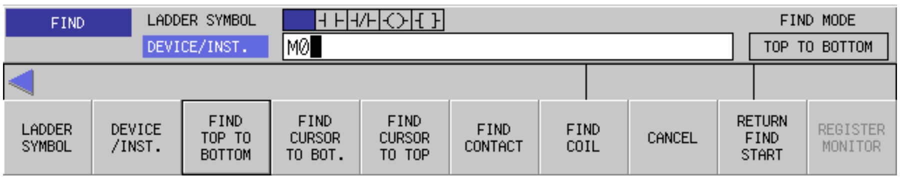
■ In 700 Series

"FIND" popup screen

Menus corresponding to popup screen

■ In 70 Series

"FIND" screen



(a) When pressing "INPUT", LADDER SYMBOL menu and alphabetical key on the LADDER MONITOR screen, "FIND" popup screen will appear.
- When "INPUT" is pressed while device exists on cursor position, the device on the cursor position is displayed.
- When LADDER SYMBOL menu is pressed, ladder symbol is displayed on the ladder symbol section.
- When alphabetical key is pressed, the pressed character is displayed and the instruction or device to be consecutively searched can be entered.
  The < and > keys may not exist on the keyboard for some machine types. Use "<",">" menu button instead when inputting an instruction that includes "<",">".

(b) With the simple operation mode, execute the following menu key operation to open the "FIND" popup screen.
- "MAIN" → "LADDER MONITOR" "FIND"
- "MAIN" → "LADDER EIDIT" "FIND"

(2) Search direction and types

Menu item	Search direction and type
FIND TOP TO BOTTOM	This searches the designated device or instruction from the top of program. When "INPUT" is pressed as soon as the popup screen is opened, search begins from the top, as well.
FIND CURSOR TO BOT.	This searches the designated device or instruction downward from the cursor position.
FIND CURSOR TO TOP	This searches the designated device or instruction upward from the cursor position.
FIND CONTACT	This searches the designated device contact instruction from the top of program. Ignore the status of "LADDER SYMBOL" at this time. (Example: When contact is searched in the "-()- M0" state, error will not occur and instead the contact of M0 will be searched.)
FIND COIL	This searches the designated device coil instruction from the top of program. Ignore the status of "LADDER SYMBOL" at this time.

(3) Consecutive searching method

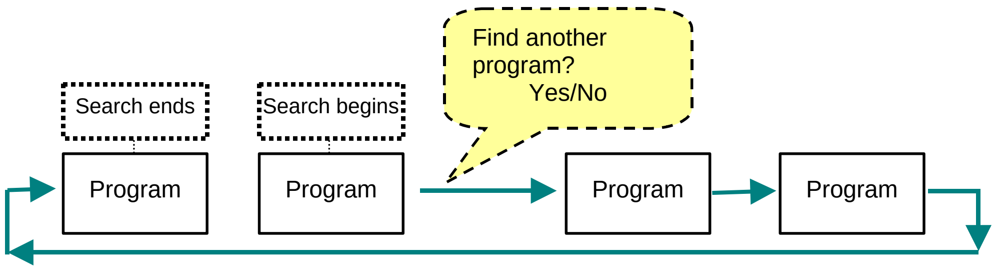
Once the search using menu key or "INPUT" is started, the menu key of selected search type turns to "FIND NEXT". When "FIND NEXT" menu key or "INPUT" is pressed, search continues consecutively. When multiple programs are opened, a program is searched to the end. Then, the next program will be searched consecutively.

(4) Consecutive searching of multiple programs

When multiple programs are opened, program search is performed to the end of a program, and then continued on to the next program in response to the following message.
When searched object was found in the first program:
"Find completed. Find another program. YES/NO"
When searched object was not found in the first program:
"The find target could not be found. Find another program. YES/NO"
(Message will not appear after the second program.)
(Search continues by simply pressing "INPUT" as the message focus is on "Yes")

If "Yes" is selected, the next program will be searched consecutively. When all the programs have been searched, "Find is complete" will appear.
If "No" is selected, search will be ended.

(Example)



(5) Ladder monitor registration method

Circuit block on the cursor position of the "LADDER" screen can be registered in the "ENTRY LADDER" screen.
Refer to "8.1.4 Ladder Entry Monitor (Split Screens)" for details.

(6) Device monitor registration method

Devices can be registered when "ENTRY DEVICE MONITOR" screen (lower part of screen) is displayed in a split screen format.

Refer to "8.1.3 (3)(a) when registering by using the "REGISTER MONITOR" button on the "LADDER" screen for details.

(7) Returning to the start (70 Series)

Returns to the step from which the search started.

8.3.2 Searching for Step No. (Simple search)

700		70
Standard	Simple	
○	○	○

(1) Displaying "FIND STEP NO." screen

(a) When number key is pressed in the LADDER EDIT screen or LADDER MONITOR screen, "FIND STEP NO." popup screen will appear.

(b) With the simple operation mode and 70 Series, execute the following menu key operation to open the "FIND STEP NO." popup screen.

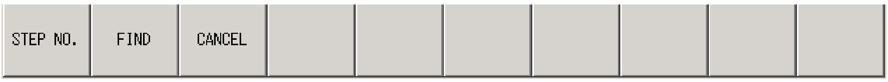
• "MAIN" → LADDER MONITOR FIND STEP NO. .

• "MAIN" → LADDER EDIT FIND STEP NO. .

* When numerical key is pressed while cursor is located on the ENTRY LADDER MONITOR (split screen) side, circuit "FIND" popup screen will appear.

"FIND STEP NO." popup screen

Menus corresponding to popup screen



(2) Search methods

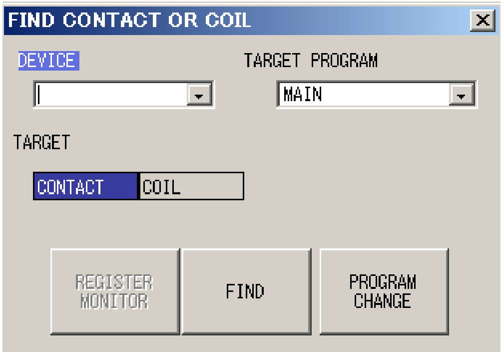
Input the step No. to be searched and press FIND menu key or INPUT key. Then, the specified step No. will be searched.

8.3.3 Searching for Contacts and Coils

700		70
Standard	Simple	
○		

Contact or coil devices can be searched from the circuits on the "LADDER" screen.
Also, in the same manner as the simple search, multiple programs can be searched consecutively.

"FIND CONTACT OR COIL" popup screen



Menus corresponding to popup screen

(1) Search methods

(a) Select "MAIN" → LADDER → FIND, REPLACE and FIND CONTACTOR COIL menu key.
When selected, the "FIND CONTACT OR COIL" popup screen will appear at the lower center of the screen.

(b) Select the DEVICE menu key, and input the device.
* The device to be searched can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Select contact or coil with the TARGET menu key. (Contact is selected as a default.)

(d) Press the FIND menu key.
* Search is always carried out downward from the head of the circuit.

(e) If the search target is found, the cursor will move to the searched circuit.

(2) Registering monitor

Registering the ladder specified with the entry ladder monitor or the device specified with the device registration monitor can be executed. Refer to "8.1.5 Registering the Monitor" for details.

(3) Closing the popup screen

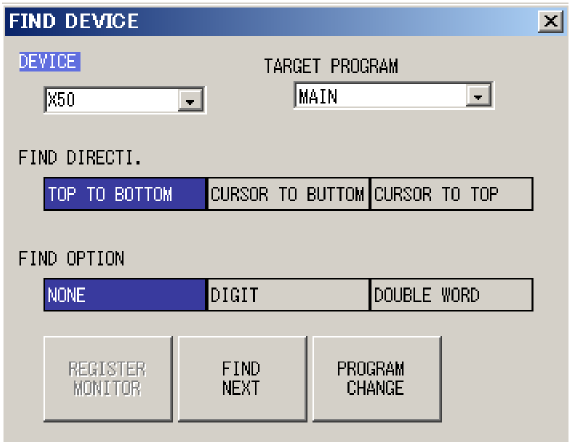
Press the ◁ menu key.

700		70
Standard	Simple	
○		

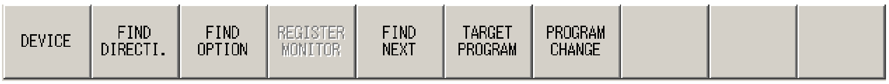
8.3.4 Searching for Device

A device can be searched from the circuits on the "LADDER" screen.

"FIND DEVICE" popup screen



Menus corresponding to popup screen



(1) Search methods

(a) Select "MAIN" → LADDER → FIND, REPLACE and FIND DEVICE menu key.
When selected, the "FIND DEVICE" popup screen will appear at the lower center of the screen.

(b) Select the DEVICE menu key, and input the device.
* The device to be searched can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Designate the search direction with the FIND DIRECTI. menu key. (As a default, the devices are searched downward from the head.)

(d) Designate the search options with the FIND OPTION menu key. (No default setting.)

(e) Press the FIND NEXT menu key.

(f) If the search target is found, the cursor will move to the searched circuit.

(g) If the FIND NEXT menu key is pressed again, the search will continue from the current position.

(2) Registering monitor

Registering the ladder specified with the entry ladder monitor or the device specified with the device registration monitor can be executed. Refer to "8.1.5 Registering the Monitor" for details.

(3) Closing the popup screen

Press the ◁ menu key.

(4) Consecutive searching of multiple programs

Refer to "8.3.1 Searching for Ladder (Simple search)" for details.

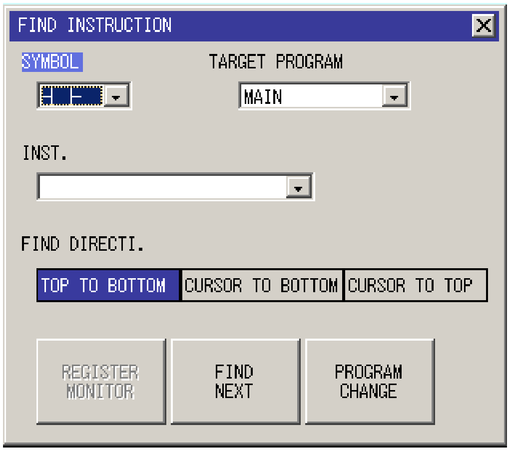
* Changing the search target program (ladder)
The program to be searched can be changed.
Refer to "8.4.1 Changing Data (Program, Device Comment)" for details on the operation methods.

8.3.5 Instruction Search

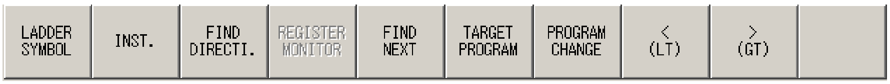
700		70
Standard	Simple	
○		

Instructions can be searched from the ladder on the "LADDER" screen.

"FIND INSTRUCTION" popup screen



Menus corresponding to popup screen



(Supplement)
The [<] and [>] keys may not exist on the keyboard for some machine types. Use "<",">" menu button instead when inputting an instruction that includes "<",">".

(1) Search methods

(a) Select "MAIN" → [LADDER] → [FIND, REPLACE] and [FIND INST.] menu key.
When selected, the "FIND INSTRUCTION" popup screen will appear at the lower center of the screen.

(b) Select the [LADDER SYMBOL] menu key, and select the circuit symbol from the list.
This can be omitted for a function instruction such as MOV.

(c) Input the search instruction with the [FIND INST.] menu key.
* The instruction to be searched can be directly input, or a previously input instruction can be selected from the list. (Note that the list is cleared when onboard is ended.)

(d) Designate the search direction with the [FIND DIRECTI.] menu key. (As a default, the devices are searched downward from the head.)

(e) Press the [FIND NEXT] menu key or [INPUT] key.

(f) If the search target is found, the cursor will move to the searched circuit.

(g) If the [FIND NEXT] menu key or [INPUT] key is pressed again, the search will continue from the current position.

(2) Registering monitor

Registering the ladder specified with the entry ladder monitor or the device specified with the device registration monitor can be executed. Refer to "8.1.5 Registering the Monitor" for details.

(3) Closing the popup screen

Press the [◁] menu key.

(4) Consecutive searching of multiple programs

Refer to "8.3.1 Searching for Ladder (Sample search)" for details.

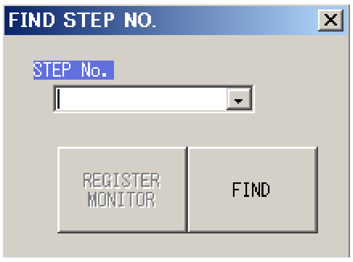
* Changing the search target program (ladder)
The program to be searched can be changed.
Refer to "8.4.1 Changing Data (Program, Device Comment)" for details on the operation methods.

8.3.6 Step No. Search

700		70
Standard	Simple	
○		

A circuit with a designated step No. can be searched from the circuits on the "LADDER" screen.

"FIND STEP NO." popup screen



Menus corresponding to popup screen

(1) Search method

(a) Select "MAIN" → LADDER → FIND, REPLACE and STEP NO. menu key.
When selected, the "FIND STEP NO." popup screen will appear at the lower center of the screen.

(b) Select the STEP NO. menu key, and set the step No.
* The step No. to be searched can be directly input, or a previously input step No. can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Press the FIND menu key.

(d) The circuit with the designated step No. will appear, and the popup screen will close.

(2) Registering monitor

Registering the ladder specified with the entry ladder monitor or the device specified with the device registration monitor can be executed. Refer to "8.1.5 Registering the Monitor" for details.

(3) Closing the popup screen

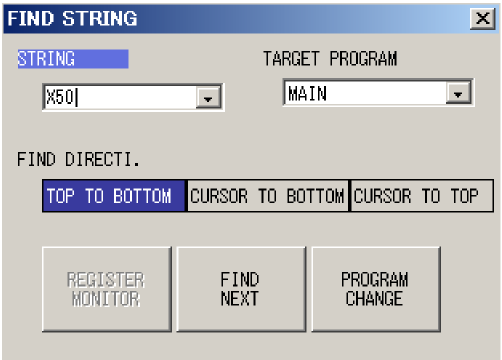
Press the ◁ menu key.

8.3.7 Character String Search

700		70
Standard	Simple	
○		

A designated character string can be searched from the circuit statements and notes on the "LADDER" screen.

"FIND STRING" popup screen



Menus corresponding to popup screen

(1) Search methods

(a) Select "MAIN" → LADDER → FIND, REPLACE and FIND STRING menu key.
When selected, the "FIND STRING" popup screen will appear at the lower center of the screen.

(b) Select the STRING menu key, and set the character string.
* The character string to be searched can be directly input, or a previously input character string can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Designate the search direction with the FIND DIRECTI. menu key. (As a default, the devices are searched downward from the head.)

(d) Press the FIND NEXT menu key.

(e) If the search target is found, the cursor will move to the searched circuit.

(f) If the FIND NEXT menu key is pressed again, the search will continue from the current position.

(2) Registering monitor

Registering the ladder specified with the entry ladder monitor or the device specified with the device registration monitor can be executed. Refer to "8.1.5 Registering the Monitor" for details.

(3) Closing the popup screen

Press the ◁ menu key.

(4) Consecutive searching of multiple programs

Refer to "8.3.1 Searching for Ladder (Simple search)" for details.

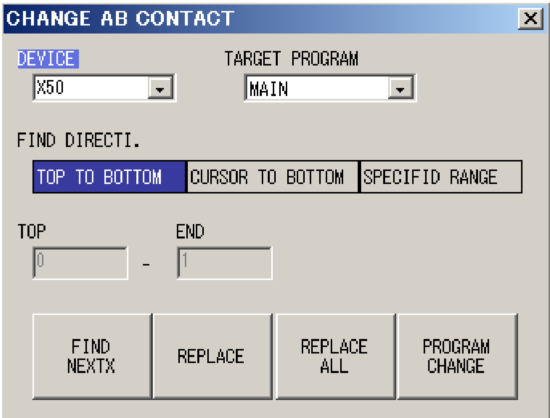
* Changing the search target program (ladder)
The program to be searched can be changed.
Refer to "8.4.1 Changing Data (Program, Device Comment)" for details on the operation methods.

8.3.8 Changing the AB Contacts

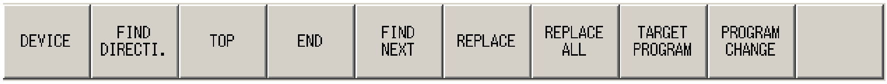
700		70
Standard	Simple	
○		

The A contact for a designated device in the circuits on the "LADDER" screen can be changed to a B contact and vice versa.

"CHANGE AB CONTACT" popup screen



Menus corresponding to popup screen



(1) Search methods

(a) Select "MAIN" → LADDER → FIND, REPLACE and CHANGE AB CONTACT menu key.
When selected, the "CHANGE AB CONTACT" popup screen will appear at the lower center of the screen.

(b) Select the DEVICE menu key, and set the device.
* The device to be changed can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Designate the search direction with the FIND DIRECTI. menu key. (As a default, the devices are searched downward from the head.)

(d) Press the FIND NEXT menu key.

(e) If the search target is found, the cursor will move to the searched circuit.

(f) If the FIND NEXT menu key is pressed again, the search will continue from the current position.

(2) Changing the AB contact

(a) Select "MAIN" → LADDER → FIND, REPLACE and CHANGE AB CONTACT menu key.
When selected, the "CHANGE AB CONTACT" popup screen will appear at the lower center of the screen.

(b) Select the DEVICE menu key, and set the device.
* The device to be changed can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Designate the search direction with the FIND DIRECTI. menu key. (As a default, the devices are searched downward from the head.)

(d) Press the FIND NEXT menu key.

(e) If the search target is found, the cursor will move to the searched circuit.

(f) Press the REPLACE key to replace the device.

(g) After replacing, the device search will continue, and the cursor will move accordingly.

(h) All devices can be replaced in a batch when the REPLACE ALL menu key is selected.

(Supplement) Device designation

Extensions and bit No. modifiers can be designated when designating the device, but index modifiers cannot be designated.

- Example of device designation
  X0Z3 cannot be designated (Cannot be designated because this is an index modifier)
- The A contact and B contact in the following instructions can be changed.
  A contact : LD,AND,OR,LDP,ANDP,ORP,EGP
  B contact : LDI,ANI,ORI,LDF,ANDF,ORF,EGF

(3) Closing the popup screen

Press the ◁ menu key.

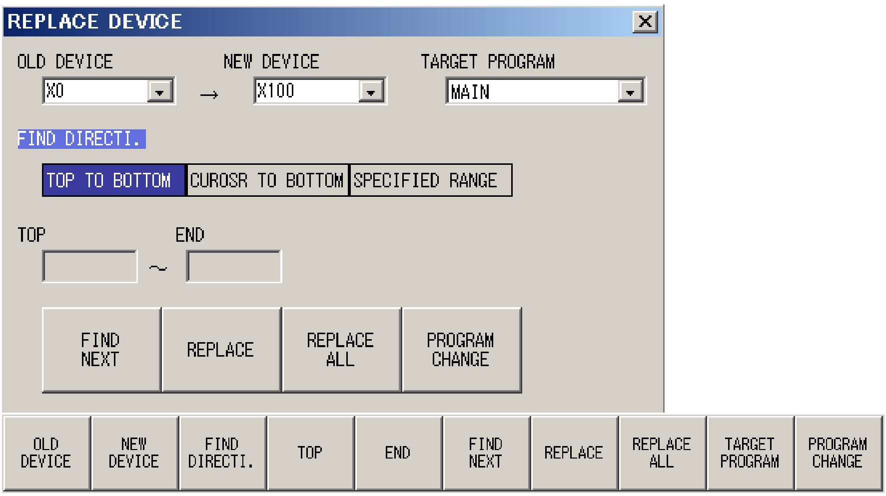
* Changing the search target program (ladder)
The program to be searched can be changed.
Refer to "8.4.1 Changing Data (Program, Device Comment)" for details on the operation methods.

8.3.9 Replacing Devices

700		70
Standard	Simple	
○		

Devices and character string constants, etc., on the "LADDER" screen can be replaced.

"REPLACE DEVICE" popup screen



Menus corresponding to popup screen

(1) Search methods

(a) Select "MAIN" → LADDER → FIND, REPLACE and REPLACE DEVICE menu key.
When selected, the "REPLACE DEVICE" popup screen will appear at the lower center of the screen.

(b) Select the OLD DEVICE menu key, and set the device to be replaced.
* The device to be changed can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Select the NEW DEVICE menu key, and set the new device.
* The device to be changed can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(d) Designate the search direction with the FIND DIRECTI. menu key. (As a default, the devices are searched downward from the head.)

(e) Press the FIND NEXT menu key.

(f) If the search target is found, the cursor will move to the searched circuit.

(g) If the FIND NEXT menu key is pressed again, the search will continue from the current position.

(2) Replacement methods

(a) Select "MAIN" → [LADDER] → [FIND, REPLACE] and [REPLACE DEVICE] menu key.
When selected, the "REPLACE DEVICE" popup screen will appear at the lower center of the screen.

(b) Select the [OLD DEVICE] menu key, and set the device.
* The device to be changed can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Select the [NEW DEVICE] menu key, and set the device.
* The device to be changed can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(d) Designate the search direction with the [FIND DIRECTI.] menu key. (As a default, the devices are searched downward from the head.)

(e) Press the [FIND NEXT] menu key.

(f) If the search target is found, the cursor will move to the searched circuit.

(g) Press the [REPLACE] key to replace the device.

(h) After replacing, the device search will continue, and the cursor will move accordingly.

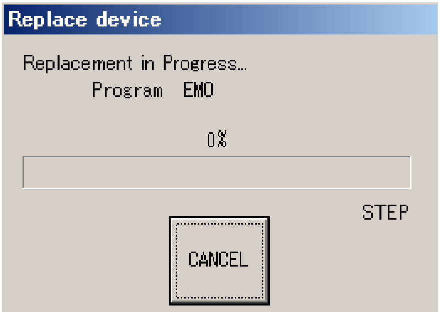
(i) All devices can be replaced in a batch when the [REPLACE ALL] menu key is selected.

(Supplement)
- Device designation
  The following devices can be replaced.
- Word devices to word devices
- Bit devices to bit devices
  Note that digits, indexes and indirect modifiers cannot be replaced.

(3) Closing the popup screen

Press the [◁] menu key.

* Changing the search target program (ladder)
  The program to be searched can be changed.
  Refer to "8.4.1 Changing Data (Program, Device Comment)" for details on the operation methods.
* When all replace is executed, the following progress bar will appear.

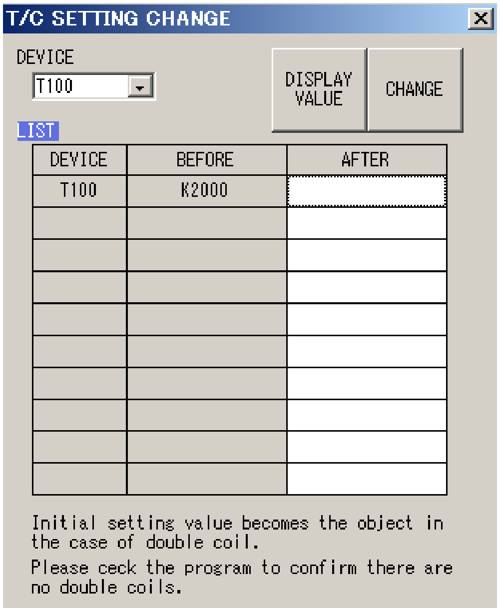


8.3.10 Changing the T/C Setting Value

700		70
Standard	Simple	
○		

The timer and counter setting values used in the program displayed on the "LADDER" screen can be listed, and the setting values can be changed in a batch.

"T/C SETTING CHANGE" popup screen



Menus corresponding to popup screen

(1) Displaying the timer and counter device list

(a) Select "MAIN" → [LADDER] → [FIND, REPLACE] and [T/C SETTING CHANGE] menu key.
When selected, the "T/C SETTING CHANGE" popup screen will appear at the lower center of the screen.

(b) Select the [DEVICE] menu key, and input the timer or counter device.
* The timer or counter device can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Select the [DISPLAY VALUE] menu key. When selected, the list of timer or counter devices will appear.

(2) Changing the timer and counter setting value

(a) List the timer or counter devices.

(b) Select the [LIST] menu key, move the cursor to the "AFTER" column of the device for which the setting value is to be changed, and then press the [INPUT] key. The cursor can be moved with the "up, down, left, right" arrow keys.

(c) Input the new setting value. (Example: To change the value to 10, input "K10".)

(d) Select the [CHANGE] menu. The setting will be changed, and the "T/C SETTING CHANGE" popup screen will automatically close.

(3) Closing the popup screen

Press the [◁] menu key.

8.4 Changing the Displayed Details

The program data and device comment data to be edited can be changed, and the comments can be displayed and set.

8.4.1 Changing Data (Program, Device Comment)

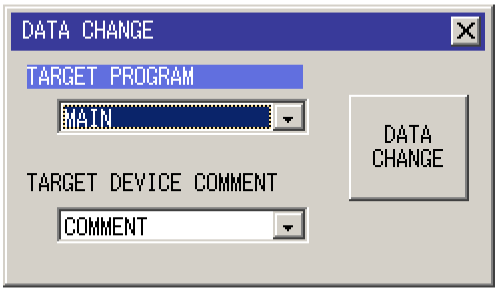
700		70
Standard	Simple	
○	○	

8.4.1.1 Data Changeover

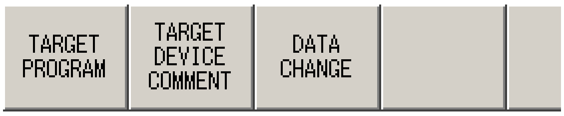
When reading out multiple program data and device comment data, the data to be displayed, edited and monitored can be changed.

(Note 1) If the data is changed to a circuit (program) for the local editing mode (LADDER screen background is light blue), monitoring on the "LADDER" screen will be disabled.
(The LADDER menu MONITOR and MONITOR menu will all be displayed in gray and disabled. Circuit monitoring and device registration monitoring on the LADDER screen will also be disabled.)

"DATA CHANGE" popup screen



Menus corresponding to popup screen



(1) Displaying the "DATA CHANGE" popup screen

(a) Standard operation mode:
• "MAIN" → LADDER → VIEW, TOOLS DATA CHANGE - or -
• "MAIN" → LADDER → EDIT → COPY/INSERT DATA CHANGE

(b) Simple operation mode:
• "MAIN" → LADDER MONITOR DATA CHANGE - or -
• "MAIN" → LADDER EDIT DATA CHANGE

When selected, the "DATA CHANGE" popup screen will appear at the lower center of the screen and will show the program and the device comment being edited.

(2) Changing the program data

(a) Select the TARGET DEVICE PROGRAM menu key, and designate the data to change.
(b) Select the DATA CHANGE menu key. When selected, the ladder circuit displayed on all screens will change to the designated program.

(3) Changing the device comment

(a) Select the TARGET DEVICE COMMENT menu key, and designate the data to change.
(b) Select the DATA CHANGE menu key. When selected and a comment is displayed, the details of the comment will change to the comment for the designated data.

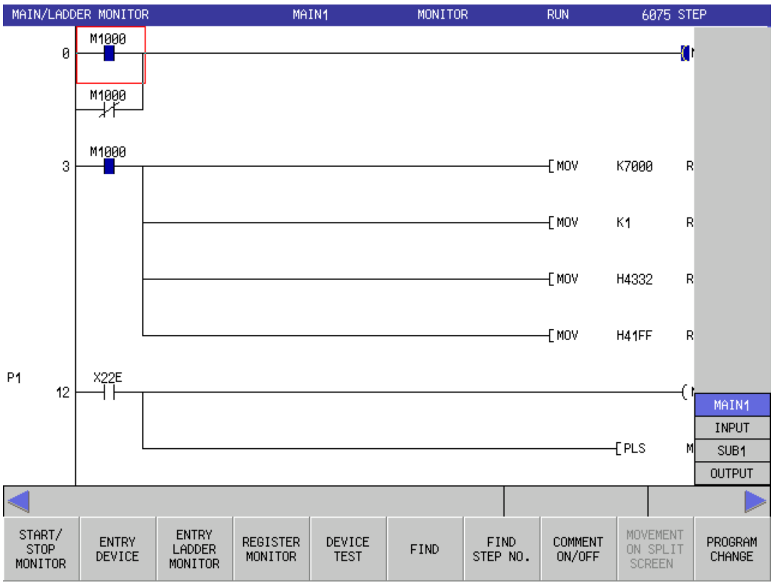
(4) Closing the popup screen

Press the ◁ menu key.

8.4.1.2 Program Changeover

700		70
Standard	Simple	
		○

The programs in the circuit display can be changed.



(1) Changing the displayed programs

(a) Select "MAIN" → [LADDER MONITOR] [PROGRAM CHANGE] - or -
"MAIN" → [LADDER EDIT] [PROGRAM CHANGE] menu.
When the menu key is pressed, the screen shows the list of the programs that have been read.

(b) Select the program with the arrow keys or page up/down key. The selected program is displayed by pressing the [INPUT] key.

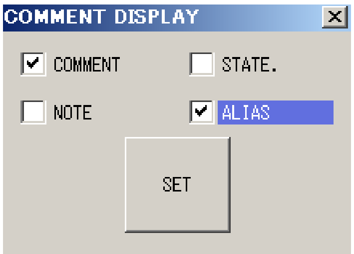
8.4.2 Comment Display

700		70
Standard	Simple	
○	○	

The validity of the comment display, statement display, note display and device name display can be set.
In 70 Series, these display settings are in "ENVIRON. SETTING" screen. Refer to "6.5 Comment Display Setting".

(Note 1) If an incompatible language is designated for the comment, it will not be displayed correctly.

"COMMENT DISPLAY" popup screen



Menus corresponding to popup screen

(1) Displaying a comment, statement, note or device name

* The method for displaying a comment is given as an example below.

(a)

Select { Standard operation mode: "MAIN" → LADDER → VIEW, TOOLS COMMENT DISPLAY menu key
Simple operation mode: "MAIN" → COMMENT DISPLAY menu key }

When selected, the "COMMENT DISPLAY" popup menu will appear at the lower center of the screen.

(b) Select the COMMENT DISPLAY menu key.
A check mark will appear at "COMMENT".

(c) When the COMMENT DISPLAY menu key is selected again, the check mark will be removed.

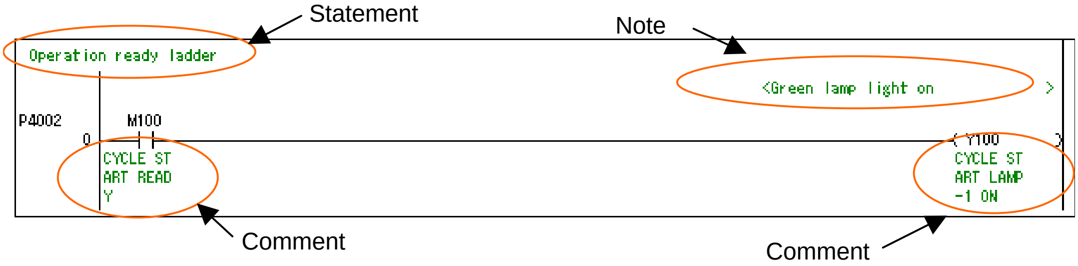
(d) Press the "SET" menu key. The "COMMENT DISPLAY" popup menu will close, and the comment display setting will change.
* Comment display set once will be held even after terminating the onboard.
When the onboard is started up again, the comment display is shown in the same state as it was set last.

(2) Closing the popup screen

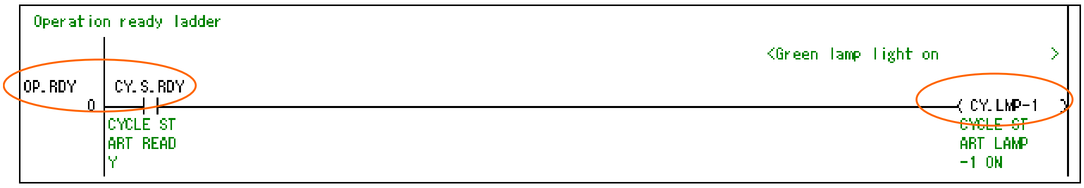
Press the ◁ menu key.

Examples of displaying various comments are given below.

- Example of displaying comment, statement and note



- Example of displaying device name (The device name appears at the device name display.)



(3) Comment ON/OFF by "EOB(;)"

Comment can be switched ON or OFF by using "EOB(;)".
Refer to "8.4.3 Comment ON/OFF" for details.

8.4.3 Comment ON/OFF

700		70
Standard	Simple	
○	○	○

Comment set with circuit's "COMMENT DISPLAY" can be switched ON or OFF.

(1) Switching by key

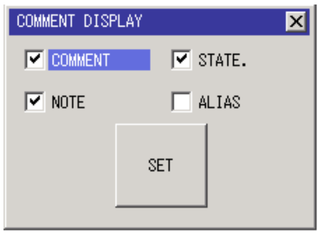
Comment can be switched ON or OFF by using "EOB(;)".

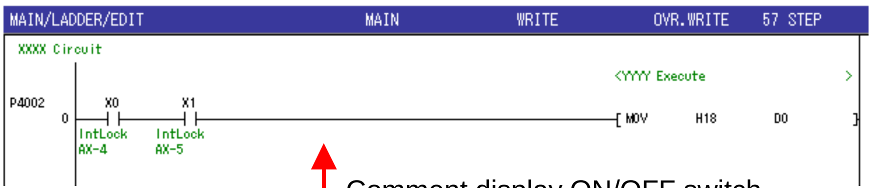
(2) Switching by menu key (Simple operation mode and 70 Series)

- "MAIN" → LADDER MONITOR COMMENT ON/OFF - or -
- "MAIN" → LADDER EDIT COMMENT ON/OFF

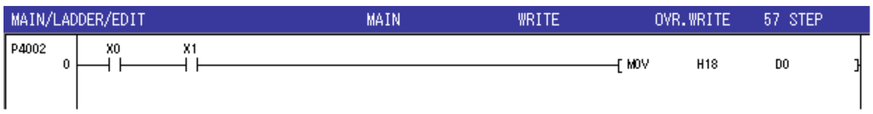
When comment display is switched ON while "COMMENT DISPLAY" box is unchecked, comment, statement and note will be checked.

(Example) "COMMENT ON/OFF" button operations when "COMMENT DISPLAY" is set as shown below.
(The screen is an example of 700 Series)





Comment display ON/OFF switch



8.4.4 Setting the Circuit Display Scale

700		70
Standard	Simple	
○	○	○

(1) Switching by menu key

The circuit's display scale can be switched by using [ZOOM DISPLAY] menu key.

(a) With standard operation mode

- "MAIN" → [LADDER] → [VIEW, TOOLS] [LADDER DISPLAY] [ZOOM DISPLAY]

(b) With simple operation mode and 70 Series

- "MAIN" → [LADDER MONITOR] [ZOOM DISPLAY] - or -
- "MAIN" → [LADDER EDIT] [ZOOM DISPLAY]

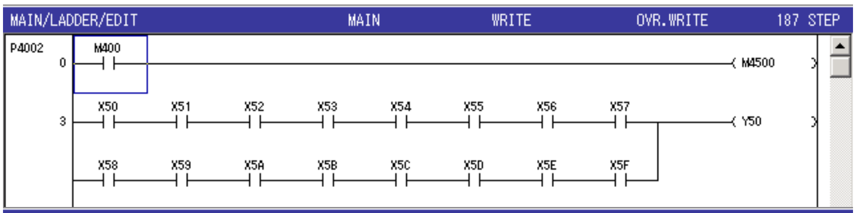
By pressing the menu key, the scale changes in 3 stages: "Reduction" → "Standard" → "Expansion" → "Reduction"

In 70 Series, "ENVIRON. SETTING" screen has the same setting menu. Refer to "6.4.2 Zoom Display".

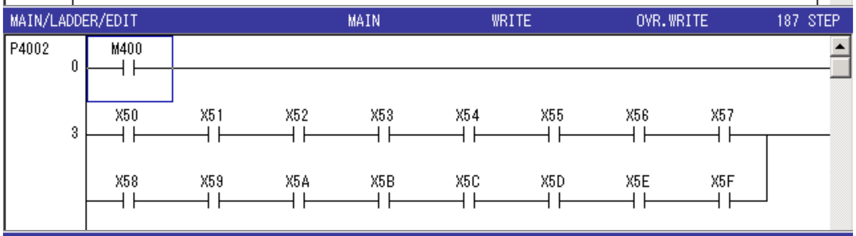
(Note1) The setting of circuit's expanded/reduced display and maximum number of contacts will be held even after terminating the onboard.
(When the onboard is started up again, the circuit will be displayed with the same settingas as before.)

■ In 700 Series

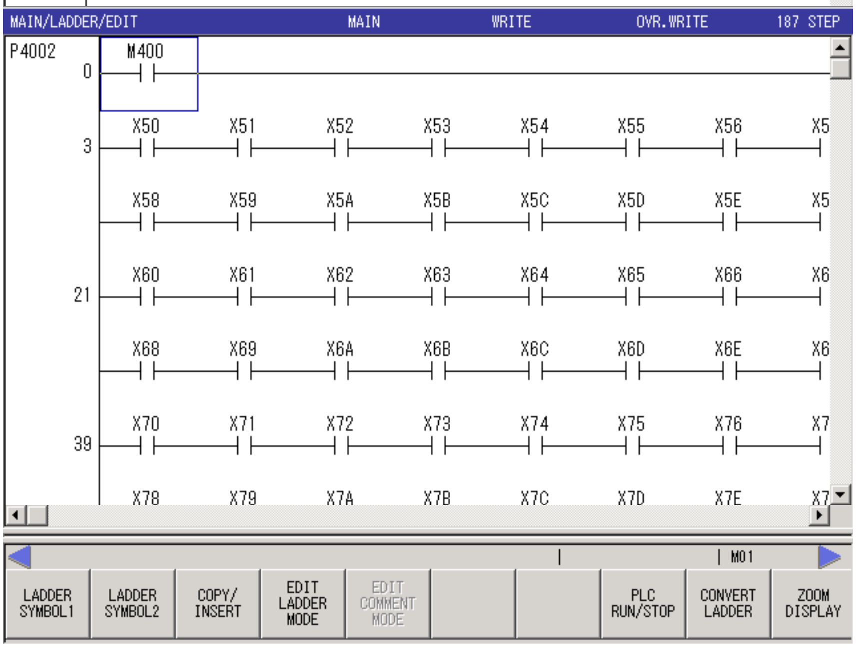
Reduced display



Standard display



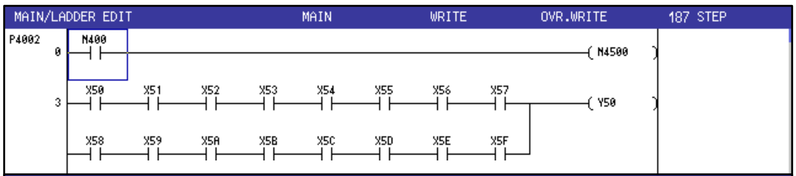
Expanded display



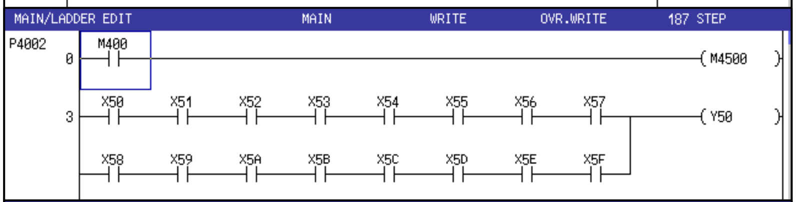
■ In 70 Series

<9 contacts display>

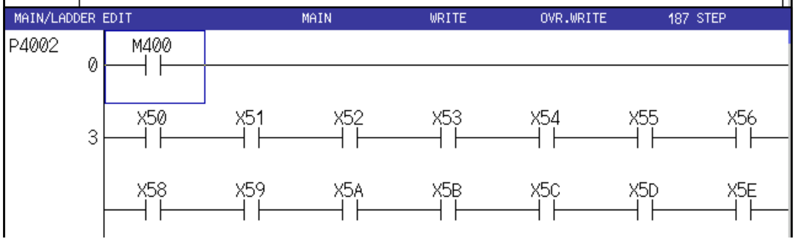
Reduced display



Standard display

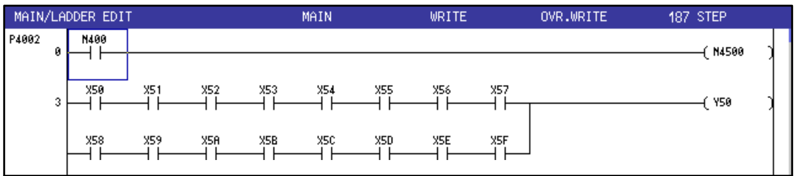


Expanded display

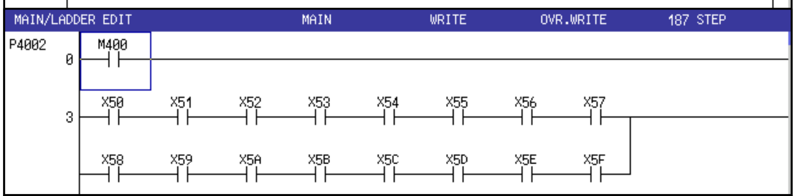


<11 contacts display>

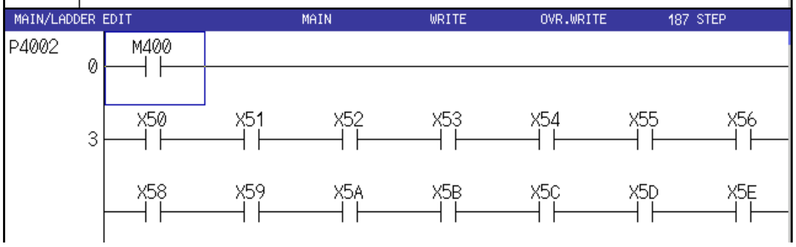
Reduced display



Standard display



Expanded display

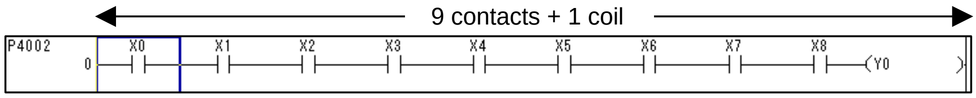


(Note 1) The ZOOM CURSOR menu focuses on the cursor display area in reduced display and enlarges the area to the standard size.

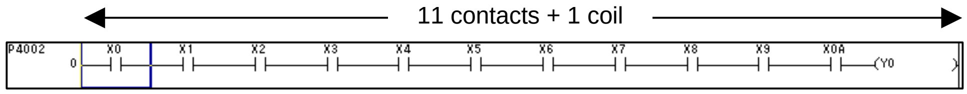
(2) Specifying the maximum number of contacts

In 700 Series, maximum number of contacts are determined by the resolution of the display unit. Maximum number of contacts will be changed from "LADDER DISPLAY" menu in the standard operation mode.
In the standard operation mode, select "MAIN" → LADDER → VIEW, TOOLS
LADDER DISPLAY NUMBER OF CONTACT menu key to switch display size.
In 70 Series, the maximum number of contacts is changed in "ENVIRON. SETTING" screen. Refer to "6.4.1 Maximum Number of Contacts" for details.

When screen resolution is VGA (640x480) (9 contacts)



When screen resolution is XGA (1024x768) (11 contacts)



(3) Displaying the circuit with horizontal scroll

There may be cases when the entire width of the circuit is not displayed because of the screen resolution, maximum number of contacts or the expansion/reduction display state. In this case, a scroll bar will appear at the bottom of the circuit screen. When the cursor is move to the left or right end of the circuit, the screen will follow the movement and display the hidden circuit. In 70 Series, the scroll bar is not displayed.

Cases in which horizontal scroll appears

Screen resolution	Maximum number of contacts	Reduced display	Standard display	Expanded display
VGA(640x480) (700 Series)	11 contacts	Horizontal scroll appears	Horizontal scroll appears	Horizontal scroll appears
	9 contacts	Entire circuit can be displayed	Horizontal scroll appears	Horizontal scroll appears
VGA(640x480) (70 Series)	11 contacts	Entire circuit can be displayed	Horizontal scroll appears	Horizontal scroll appears
	9 contacts	Entire circuit can be displayed	Entire circuit can be displayed	Horizontal scroll appears
XGA(1024x768)	11 contacts	Entire circuit can be displayed	Entire circuit can be displayed	Horizontal scroll appears
	9 contacts	Entire circuit can be displayed	Entire circuit can be displayed	Horizontal scroll appears

(Supplement)
The reduced circuit display can also be changed with the "DISPLAY EXPANSION REDUCTION" menu key on the EDIT menu.
The display scale is switched in the order of "Reduction" → "Standard" → "Expansion" → "Reduction" by pressing.

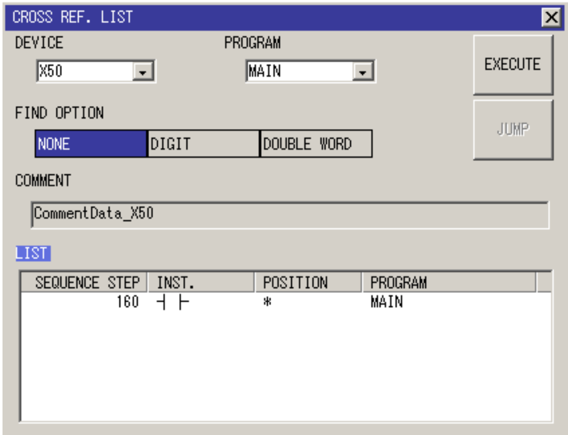
9. Other Functions

With the contact coil usage list function, step, instruction and position in which specified device is used can be displayed in a list.
With the device usage list function, device usage state in a program can be displayed per specified device.
By specifying device, the circuit sign, number of device usages in program and error state can be displayed.
Logical errors and input mistakes in the program can be checked with the program check function.

9.1 Contact Coil Usage List

700		70
Standard	Simple	
○		

The steps, instructions and position in which the designated device is used can be listed in the contact coil usage list.
(Note 1) Cannot jump to the device usage list executed before program change.



Menus corresponding to popup screen

(1) Displaying the usage list

(a) Select "MAIN" → LADDER → VIEW, TOOLS and CROSS REF. LIST menu key.
When selected, the "CROSS REF. LIST" popup screen will appear at the center of the screen.

(b) Select the DEVICE menu key, and designate the device.
* The device to be searched can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Press the PROGRAM menu key to select the area to be searched.
If "ALL PROGRAMS" is selected, all the programs opened in the onboard editing area is subject to search.

(d) Select the FIND OPTION menu key, and designate the search option. (No default setting.)

(e) Select the EXECUT menu key. The usage list of designated devices will appear.

(2) Jumping to the usage position

(a) Display the usage list as explained above.

(b) Select the LIST menu key, and select the position in the list to jump to.

(c) Press the JUMP menu key.
The circuit of the position designated in the list will appear at the head of the "LADDER" screen.

(3) Closing the popup screen

Press the ◁ menu key.

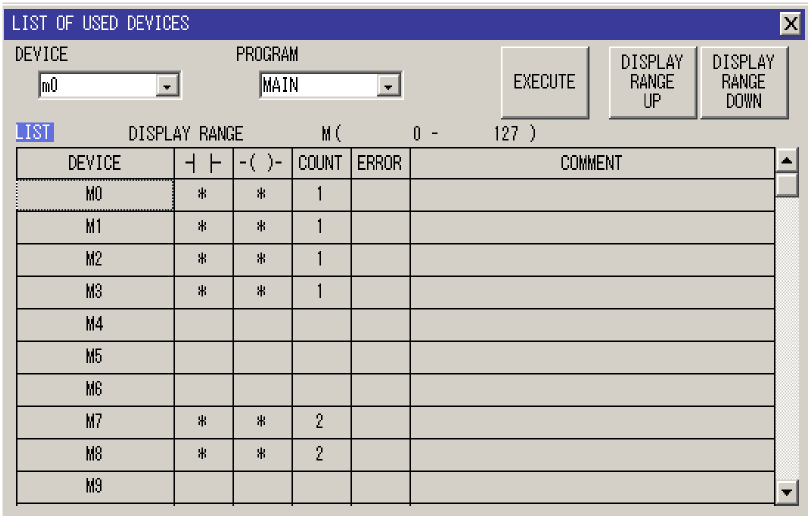
* The device usage list is held even if the popup window is closed. When the "CROSS REF. LIST" screen is displayed again, the previous list details will appear. (The list is cleared when onboard is ended.)

9.2 List of Used Devices

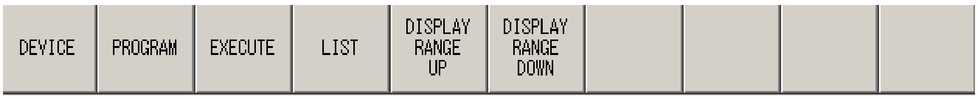
700		70
Standard	Simple	
○		

The usage state of devices in the program can be displayed for each designated device with the list of used devices. The circuit symbols, usage quantity and presence of errors can be displayed by designating the device.

"LIST OF USED DEVICE" popup screen



Menus corresponding to popup screen



(1) Displaying the list of used devices

(a) Select "MAIN" → LADDER → VIEW, TOOLS and LIST OF USED DEVICE menu key.
When selected, the "LIST OF USED DEVICE" popup screen will appear at the center of the screen.

(b) Select the DEVICE menu key, and designate the device.
* The device to be searched can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(c) Select the EXECUTE menu key. The usage list of designated devices will appear.

(2) Changing the displayed list page

(a) Display the list of used devices as explained above.

(b) Select the LIST menu key.

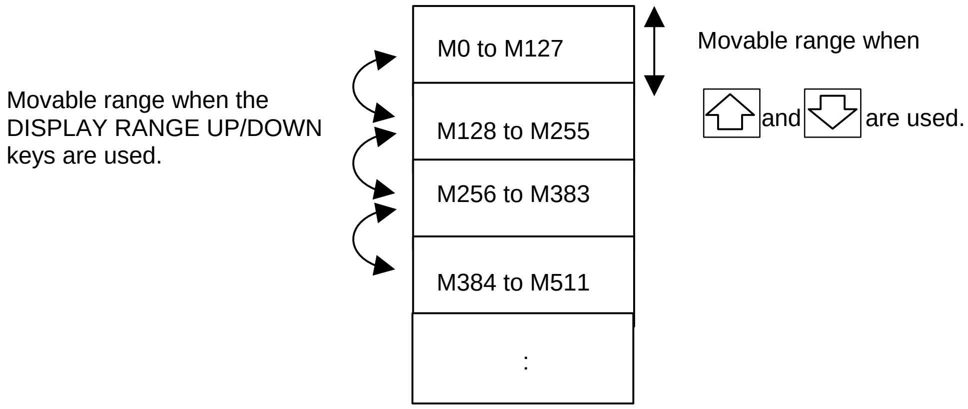
(c) Change the display in page units by pressing the ⇧ and ⇩ keys.
* Note that the range that can be moved with page changeover is limited to 512 devices. To move more, use "Changing the device display range" explained below.

(3) Changing the device display range

(a) The range of devices that can be displayed in the list is a unit of 128 devices.
(b) To move the display range upward, select the DISPLAY RANGE UP menu key.
(c) To move the display range downward, select the DISPLAY RANGE DOWN menu key.

* The relationship between the list's display page change and device display range change is shown below.

Example) In the case of M device



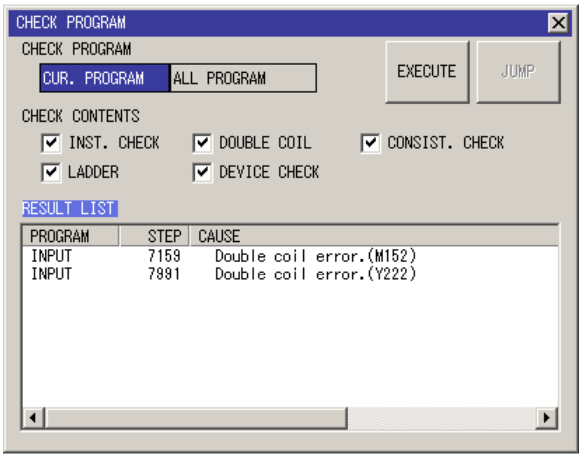
(4) Closing the popup screen

Press the ◁ menu key.

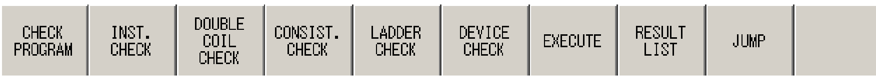
9.3 Program Check

700		70
Standard	Simple	
○		

Logical errors and input mistakes in the program can be checked with the program check function.



Menus corresponding to popup screen



(1) Checking a program (ladder circuit)

(a) Select "MAIN" → [LADDER] → [VIEW, TOOLS] and [PROGRAM CHECK] menu key.
When selected, the "PROGRAM CHECK" popup screen will appear at the lower center of the screen.

(b) Press the [CHECK PROGRAM] menu key to select the area to be checked.
If current program is selected, only the program which is currently subject to edit is checked.
If all the programs are selected, all the programs opened in the onboard editing area is subject to check.

(c) Select the menu key corresponding to the search option ([INST. CHECK], [DOUBLE COIL CHECK], [CONSIST. CHECK], [LADDER CHECK], [DEVICE CHECK]), and set the option validity.

(d) Select the [EXECUTE] menu key. The results of the program check will appear under RESULT LIST.

(2) Moving the RESULT LIST display up, down, left or right

(a) Execute the above program check.
(b) Select the [RESULT LIST] menu key.
(c) The listed display data can be moved in one line units with the [↑] and [↓] arrow keys.
(d) The listed display can be moved left and right with the [←] and [→] arrow keys.
(e) The display can be moved up or down in page units with the [⇧] and [⇩] keys.

(3) Jumping to the error section in the check results.

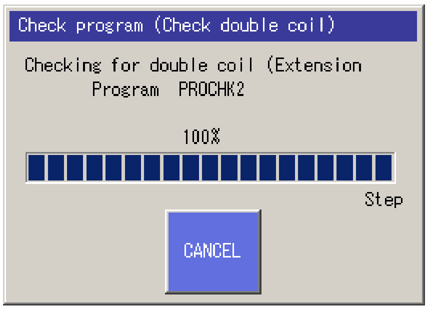
(a) Execute the above program check.
(b) Select the [LIST] menu key, and select the section in the list to jump to.
(c) Press the [JUMP] menu key.
The circuit of the position designated in the list will appear at the head of the "LADDER" screen.

(4) Closing the popup screen

Press the ◁ menu key.

* The check results list is held even if the popup window is closed. When the "PROGRAM CHECK" screen is displayed again, the previous list details will appear. (The list is cleared when onboard is ended.)

* A progress bar appears during program check as shown below.



10. Device Monitor Operations

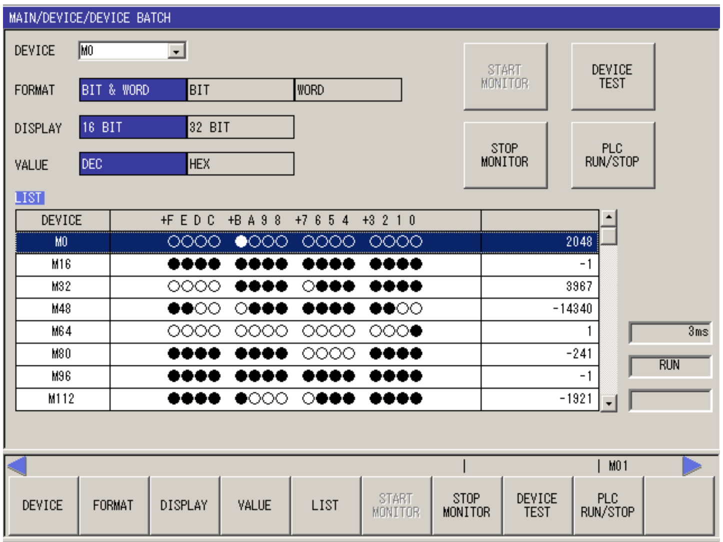
700		70
Standard	Simple	
○	○	○

10.1 Device Batch Monitor

With device batch monitor, one type of device is designated, and the continuing devices can be monitored with a list format.

(Note 1) If this screen is changed to another screen during monitoring, the monitoring operation will automatically stop.
(Note 2) The device batch monitor can be used in the NC automatic update mode (white background) or local editing mode (light blue background).
(Note 3) Number of device points when the monitor format is "bit & word" is displayed with 16 points.

"DEVICE BATCH" screen (The screen is an example of 700 Series)



(1) Displaying the "DEVICE BATCH" screen

Select { Standard operation mode: "MAIN" → [DEVICE] [DEVICE BATCH MONITOR] menu key
Simple operation mode and 70 Series: "MAIN" → [DEVICE BATCH MONITOR] menu key }

When selected, the "DEVICE BATCH" screen will appear on the full screen.

(2) Starting device batch monitoring

(a) Select the [DEVICE] menu key, and designate the head device to be monitored.
  * The device to be searched can be directly input, or a previously input device can be selected from the list. (Note that the list is cleared when onboard is ended.)

(b) Select the [FORMAT] menu key, and set the monitor format.

(c) Select the [DISPLAY] menu key, and set the display format.

(d) Select the [VALUE] menu key, and set decimal or hexadecimal.

(e) Select the [START MONITOR] menu key. When selected, monitoring of the devices designated in the list will start.

(3) Moving the list monitor data

(a) Start the device list monitoring.

(b) Select the [LIST] menu key.

(c) The listed display data can be moved in one line units with the [↑] and [↓] arrow keys.

(d) The display can be moved up or down in page units with the [⇧] and [⇩] keys.
  (Note) If the data is moved to a device number not included in the specifications, insignificant data may be displayed.

(4) Stopping the monitor

Select the $\boxed{\text{STOP MONITOR}}$ menu key. If there is a registered device, monitoring of the device will stop.

(5) Testing the device

Select the $\boxed{\text{DEVICE TEST}}$ menu key. When selected, the "DEVICE TEST" popup screen will appear. Refer to section "8.1.6 Testing the Devices" for details on the device test operations.

(6) Run and stop the PLC

Select the $\boxed{\text{PLC RUN/STOP}}$ menu key. When selected, the "PLC RUN/STOP" popup screen will open. Refer to "13.7 Controlling PLC RUN/STOP" for details on the PLC RUN/STOP operations.

(7) Closing the popup screen

Press the $\boxed{\triangleleft}$ menu key.

10.2 Device Registration Monitor

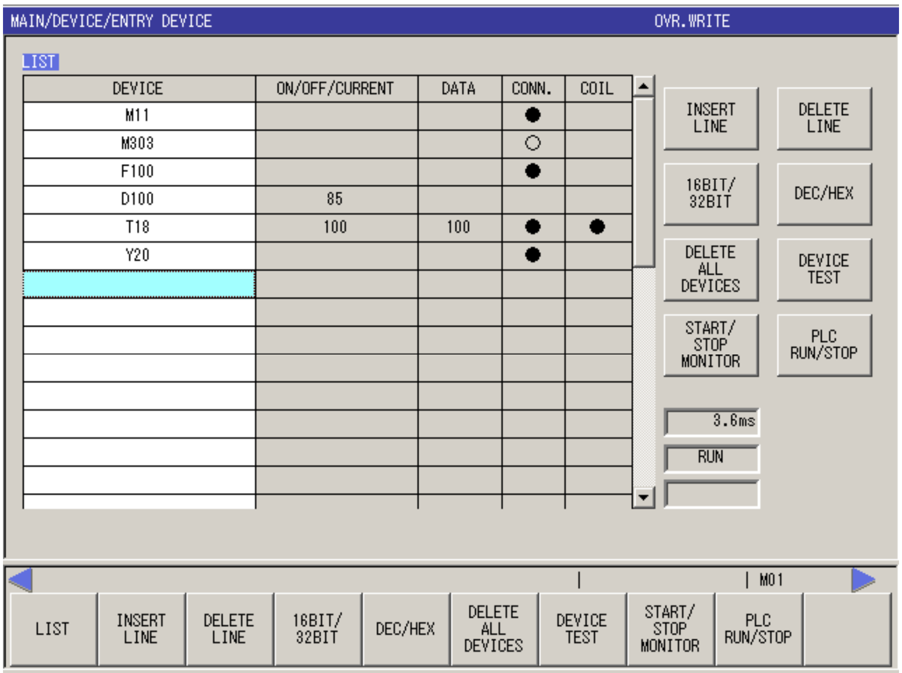
700		70
Standard	Simple	
○		

Devices at a separated position in the circuit or different types of devices can be monitored simultaneously on one screen.

(Note 1) If this screen is changed to another screen during monitoring, the monitoring operation will automatically stop.

(Note 2) The device registration monitor can be used in the NC automatic update mode (white background) or local editing mode (light blue background).

"ENTRY DEVICE" screen



(1) Displaying the "ENTRY DEVICE" screen

Select "MAIN" → DEVICE and ENTRY DEVICE menu key. The "ENTRY DEVICE" screen will appear on the full screen.

(2) Registering a device

(a) Move the cursor to the row of "device" on the "ENTRY DEVICE MONITOR" screen.
(b) Enter the input mode with alphabet or INPUT key.
(c) Enter the device name and press the INPUT key.

(3) Deleting a device

(a) Select the LIST menu key, and move the cursor to the device to be deleted.
(b) Select the DEVICE DELETION menu key. When selected, the device at the cursor position in the list will be deleted.

(4) Deleting all devices

(a) Select the DELETE ALL DEVICES menu key.
(b) A confirmation popup screen will open. To delete all devices, press the INPUT key.
(c) All devices displayed in the list will be deleted.

(5) Starting the monitor

Select the START MONITOR menu key. If devices are registered in the list, monitoring of the listed devices will start. (The device contact and data state are read out from the NC and displayed.)

(6) Stopping the monitor

Select the STOP MONITOR menu key. Monitoring of the registered devices in the list will stop.

(7) Testing the device

Select the DEVICE TEST menu key. When selected, the "DEVICE TEST" popup screen will appear. Refer to section "8.1.6 Testing the Devices" for details on the device test operations.

(8) Run and stop the PLC

Select the PLC RUN/STOP menu key. When selected, the "PLC RUN/STOP" popup screen will open. Refer to "13.7 Controlling PLC RUN/STOP" for details on the PLC RUN/STOP operations.

(9) Switching 16bit/32bit of registration device

*Only the word device can be changed.
Press the "16 BIT/32 BIT" button.

(10) Switching decimal/hexadecimal of registration device

*Only the word device can be changed.
Press the DEC/HEX button.

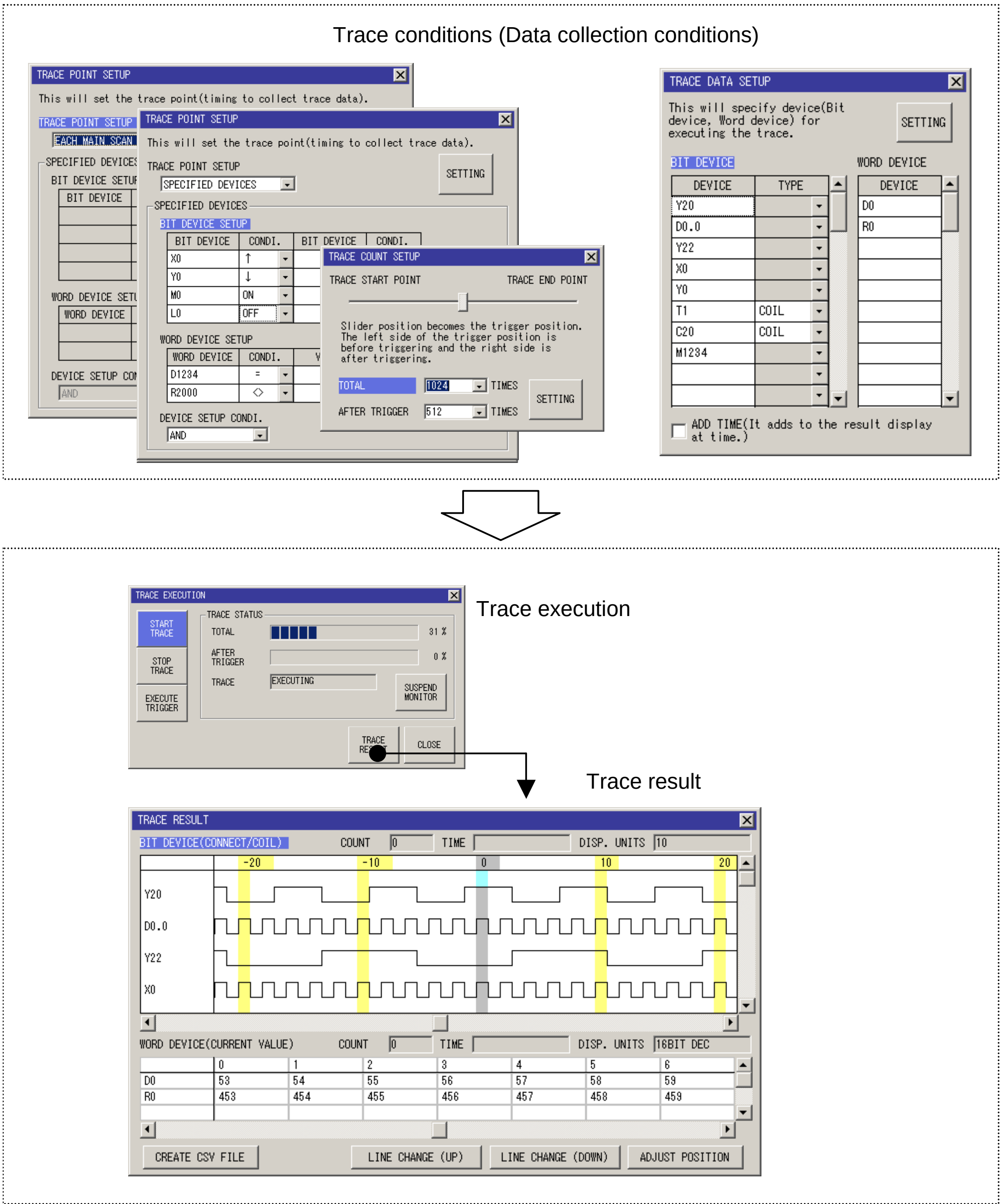
(11) Closing the popup screen

Press the ◁ menu key.

10.3 Sampling Trace

700		70
Standard	Simple	
○		

With the sampling trace function, the PLC device (bit device and word device) signals can be traced (data collection) and its result will be displayed in chronological order so that the operator can check and analyze the signal status.



	Details		
Trace condition (Data collection condition)	No. of traces (Number of times when data is collected)	The number between 1 and 8192 can be set. Note that, however, the total trace data size has to be smaller than 60kbyte. Refer to *1 for trace data size calculation.	
	No. of traces after trigger	Sets the number smaller than the No. of traces.	
	Trace point (Timing for collecting data)	High-speed Each scan	Trace is executed at every scan of PLC high-speed processing . *Up to 8 points of bit devices can be set for the trace data.
		Main Each scan	Trace is executed at every scan of PLC main processing.
		Trace data change	Trace is executed when the device set at the trace data at every scan of main processing is changed. (When changed from OFF->ON,ON->OFF for bit device; When the value is changed for word device.)
		Device designation	Trace is executed when the device setting condition at the time of device specification on the "TRACE POINT SETUP" screen has been established at every scan of the main processing. Refer to *2 for the settable devices.
	Trigger point (Point where trigger condition has been established.)	Screen input	Trigger is executed by pressing the "EXECUTE TRIGGER" button on the "EXECUTE TRIGGER" screen.
		Device designation	Trigger is executed by either one of the following two methods. • Trigger is executed by pressing the "EXECUTE TRIGGER" button on the "EXECUTE TRIGGER" screen. • Trigger is executed when the device setting condition at the time of device specification on the "TRIGGER POINT SETUP" screen has been established. Refer to *2 for the settable devices.
	Trace additional information	Time (displayed in the unit of hour, minute and second): Step No. and program name cannot be added.	
Trace data (Collected data)	Bit device	• 50 points of word device and 50 points of bit device can be set. Note that, however, trace data size has to be 60kbyte or less in total. Refer to *1 for calculation of trace data size. Refer to *3 for the settable devices.	
	Word device		
Trace execution	Start trace	Trace (data collection) is started by pressing the "START TRACE" button on the "TRACE EXECUTION" screen.	
	Stop trace	Trace (data collection) is stopped by pressing the "STOP TRACE" button on the "TRACE EXECUTION" screen.	

	Details	
Trace result display	Trace result display screen	The data on the number of traces (or the number of counts until trace is stopped) is display on the "TRACE RESULT" screen.
	Trace result output	Trace results are output to an external device in the CSV file format.
File input/output	Input	Trace file in the external device (trace condition and trace result) is input. (The trace file is included in the GX-Developer project.) Refer to *4 for the file compatibility with the GX-Developer.
	Output	Trace file in the external device (trace condition and trace result) is output. (The trace file is included in the GX-Developer project.) Refer to *4 for the file compatibility with the GX-Developer.
	Delete	Trace file in the external device (trace condition and trace result) is deleted. (The trace file is included in the GX-Developer project.)

*1 Calculation of trace data size

Set the number of traces and the trace devices so that the trace data size is 60kbyte (61440 byte) or smaller.
Trace data size will be calculated as follows.

[Trace data size (byte)] = [Size required for one trace (byte)] x [Number of traces]

Size required for one trace is calculated from word device points and bit device points of the trace device and the size required for one trace of each device.

Trace device type	Size (byte) required for one trace
Word device	2byte per 1 point
Bit device	2byte per 1 unit (1 unit = 16 points) 1 to 16 points → 1 unit → 2byte 17 to 32 points → 2 units → 4byte 33 to 48 points → 3 units → 6byte 49 to 50 points → 4 units → 8byte

Trace data size calculation example

No.	Bit device points	Word device points	Trace points	Trace data size	Trace execution
1	16 points (1 unit)	2 points	8192 points	(1x2+2x2)x8192 = 49152 byte	Possible
2	8 points (1 unit)	8 points	1000 points	(1x2+8x2)x1000 = 18000 byte	Possible
3	50 points (4 units)	50 points	568 points	(4x2+50x2)x568 = 61344 byte	Possible
4	50 points (4 units)	50 points	569 points	(4x2+50x2)x569 = 61452 byte	Not possible

*2 Applicable devices with trace/trigger point details setting

Applicable devices with trace/trigger point details setting
Bit device: X, Y, M, L, F, SB, B, SM, T(contact), ST(contact), C(contact) Word device: T(current value), ST(current value), C(current value), D, R, SW, SD, W Following qualifications are available for the above devices. • Bit device digit designation • Word device bit designation

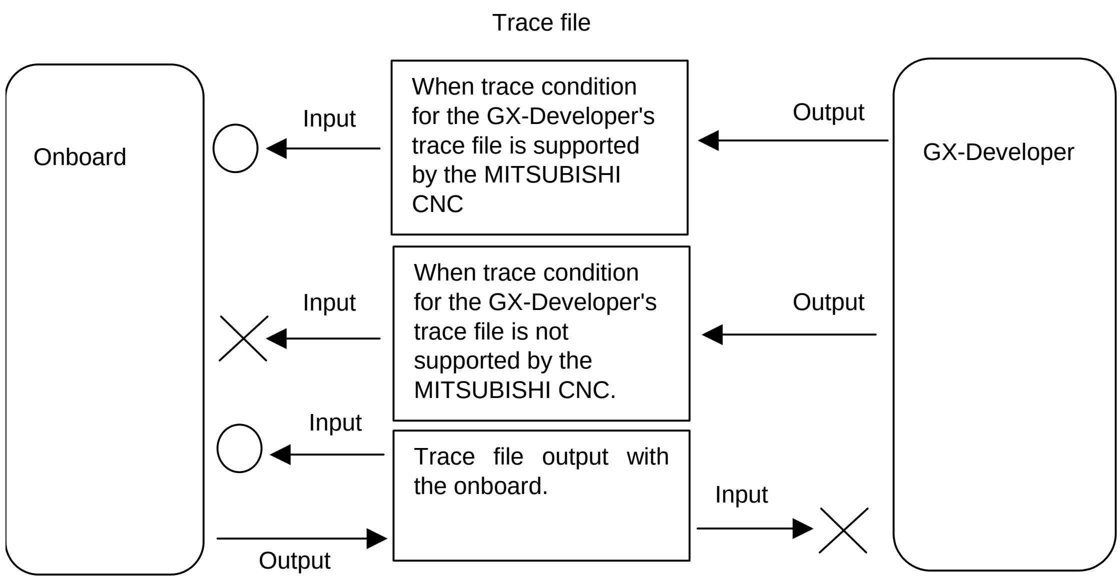
Process following an inapplicable device setting is as follows.
• If one or more applicable devices already exist, the setting of the inapplicable device will be ignored.
• If no applicable devices exist, error occurs when trace setting file is read upon trace execution.

*3 Applicable devices with device setting

Applicable devices with device setting
Bit device: X, Y, M, L, F, SB, B, SM T(contact), T(coil), ST(contact), ST(coil), C(contact), C(coil) Word device: T(current value), ST(current value), C(current value), D, R, SW, SD, W Following qualifications are available for the above devices. • Bit device digit designation • Word device bit designation

Process following an inapplicable device setting is as follows.
• If one or more applicable devices already exist, the setting of the inapplicable device will be ignored.
• If no applicable devices exist, error occurs when trace setting file is read upon trace execution.

*4 File compatibility of the trace file



Note 1) Deleting of trace file is possible only for the trace file output with the onboard.

*5 Trace file written into the NC using the GX-Developer
The trace file written into the NC with the GX-Developer is upper-compatible with the onboard trace file and can be read.

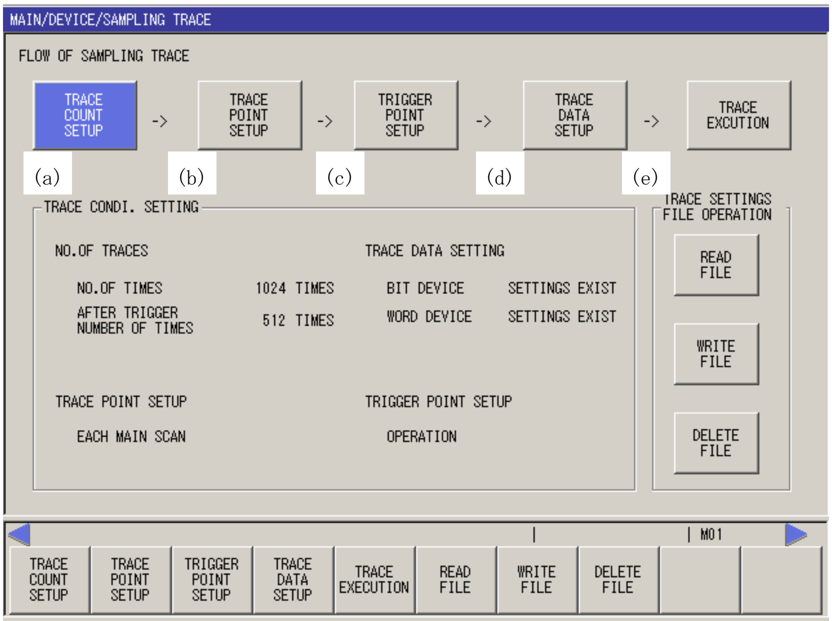
10.3.1 MAIN Screen

Sampling trace function is executed with this screen. Trace condition setting, trace execution, trace result display and file operations (READ FILE, WRITE FILE, DELETE FILE) can be performed.

(Note 1) The following operations cannot be performed during trace. (An error message appears.)

- Trace condition setting ("TRACE COUNT SETUP", "TRACE POINT SETUP", "TRIGGER POINT SETUP", "TRACE DATA SETUP")
- Reading and writing of the trace file in the file operation (Deleting of file is possible.)

(Note 2) Maximum size of trace data that can be registered is 60kbyte in total. If exceeded, an error message appears when the "TRACE EXECUTION" button is pressed. Refer to *1 in "10.3 " for calculation of trace data.



The flow chart of sampling trace operation is shown below.

(a) Set the number of traces.

↓

(b) Set the trace point (timing where trace data is collected).

↓

(c) Set the trigger (condition) to be the start position (0 point) at the time of trace execution.

↓

(d) Set the data (bit device, word device) to be traced.

↓

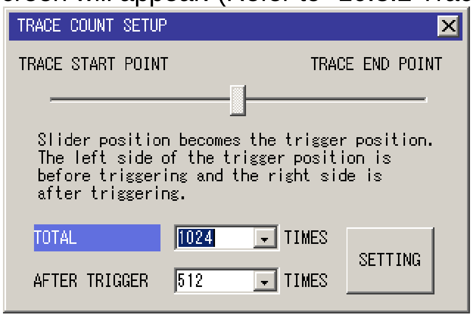
(e) Execute the trace.
Wait until trigger is executed and trace is completed.
:
Display the trace result.

(1) Displaying the "SAMPLING TRACE" screen

(a) Select "MAIN" → "DEVICE" → SAMPLING TRACE menu key. When selected, the "SAMPLING TRACE" screen will appear on the full screen.

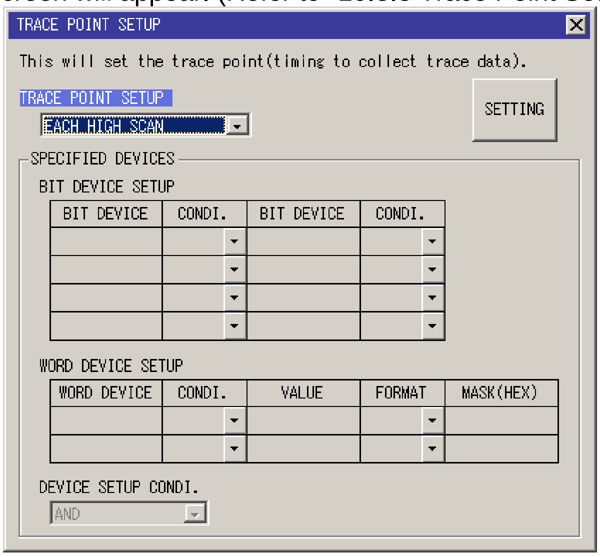
(2) Trace count setting

(a) Select the TRACE COUNT SETUP menu key. When selected, the "TRACE COUNT SETUP" popup screen will appear. (Refer to "10.3.2 Trace Count Setting" for details.)



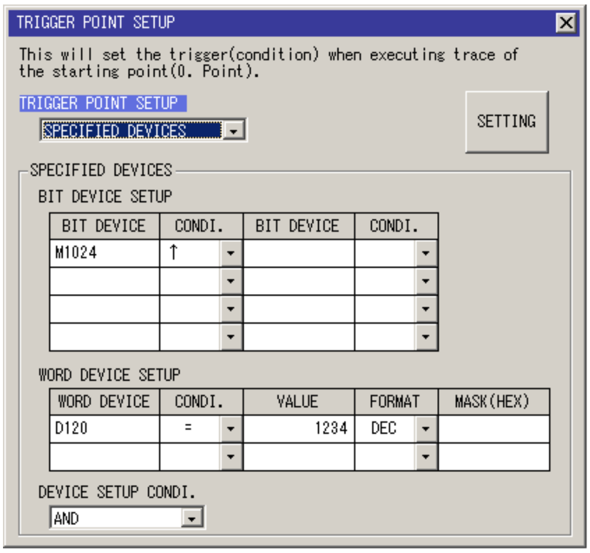
(3) Trace point setting

(a) Select the TRACE POINT SETUP menu key. When selected, the "TRACE POINT SETUP" popup screen will appear. (Refer to "10.3.3 Trace Point Setting" for details.)



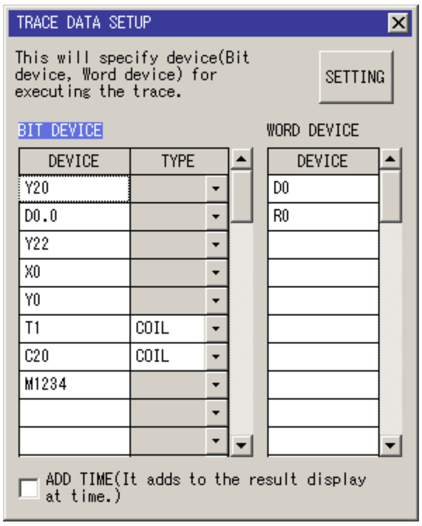
(4) Trigger point setting

(a) Select the TRIGGER POINT SETUP menu key. When selected, the "TRIGGER POINT SETUP" popup screen will appear. (Refer to "10.3.4 Trigger Point Setting" for details.)



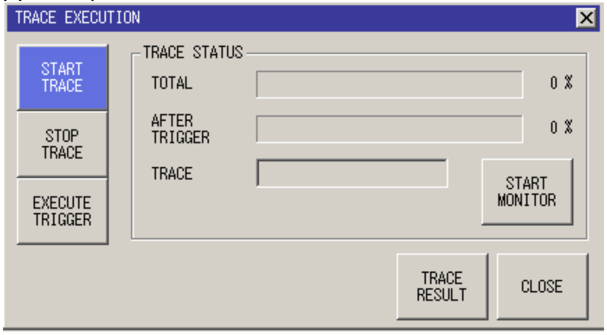
(5) Trace data setting

(a) Select the TRACE DATA SETUP menu key. When selected, the "TRACE DATA SETUP" popup screen will appear. (Refer to "10.3.5 Trace Data Setting" for details.)



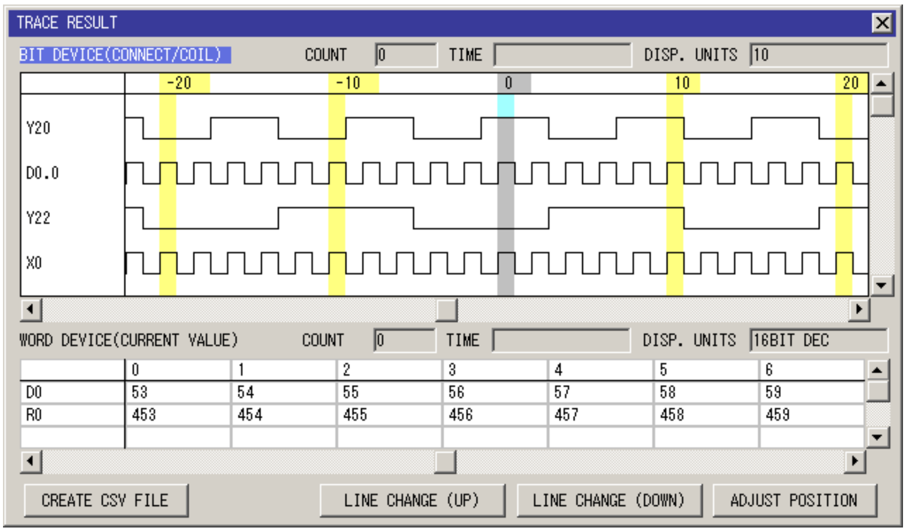
(6) Trace execution

(a) Select the TRACE EXECUTION menu key. When selected, the "TRACE EXECUTION" popup screen will appear. (Refer to "10.3.6 Trace Execution" for details.)



(7) Trace result display

(a) Select the TRACE RESULT menu key in the "TRACE EXECUTION" screen. When selected, the "TRACE RESULT" popup screen will appear. (Refer to "10.3.7 Trace Result Display" for details.)



(8) Trace condition setting display

Outline of trace condition is displayed in the main screen of "SAMPLING TRACE".

Number of traces	Total number	Total number of traces is shown.
	Number after trigger	Number of traces after trigger is shown.
Trace point setting	-	Trace point (timing for collecting trace data) setting method is shown.
Trigger point setting	-	Setting method for the start point (0 point) trigger at the execution of trace.
Trace data setting	-	Setting status for bit device and word device is shown.

(9) File operation (input, output, delete)

(a) Select "FILE OPERATION" → READ FILE, WRITE FILE, DELETE FILE menu key. When selected, a popup screen will appear.
Refer to the following sections for details.
"10.3.9 File Input", "10.3.10 File Output", "10.3.11 Deleting File"

(10) Return to the higher hierarchy of the menu

(a) Press the ◁ menu key.

10.3.2 Trace Count Setting

Number of traces (data collection) and number after trigger can be set.

Total number (Default: 1024)	Set the number of traces in the range from 1 to 8192.
Number after trigger (Default: 512)	Set the number of traces after the trigger establishment. The setting value should be less than the total number.

(Note 1) The setting of "NO. OF TRACES", etc. will not be valid until the "SETTING" menu key is pressed.
(Note 2) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.
(Note 3) Setting is not possible during trace. (An error message appears.)

(1) Displaying the "TRACE COUNT SETUP" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → TRACE COUNT SETUP menu. When selected, the "TRACE COUNT SETUP" popup screen will appear.

(2) Setting the total number

(a) Select the TOTAL menu key.

(b) Press the ↑/↓ key and specify whether to select the number of times from the list or to directly input.
When selecting from the list, the following settings are available.
"1024 times, 2048 times, 3072 times, 4096 times, 5120 times, 6144 times, 7168 times, 8192 times"

(3) Setting the number after trigger

(a) Select the AFTER TRIGGER menu key.

(b) Press the ↑/↓ key and specify whether to select the number of times from the list or to directly input.
When selecting from the list, the following settings are available.
"0 time, 1024 times, 2048 times, 3072 times, 4096 times, 5120 times, 6144 times, 7168 times, 8192 times"

(4) Establishment of trace count setting

(a) Select the SETTING menu key. When selected, the "TRACE COUNT SETUP" screen is closed and the changes will be validated.
*An error occurs if the number of traces is not correctly set. (The "TRACE COUNT SETUP" screen will not be closed.)

(5) Invalidate the changes and close the popup screen.

(a) Press the ◁ menu key.
*The changed settings will be invalid.

10.3.3 Trace Point Setting

Trace point (timing of collecting trace data) can be set.
Trace point can be specified with the method described below.

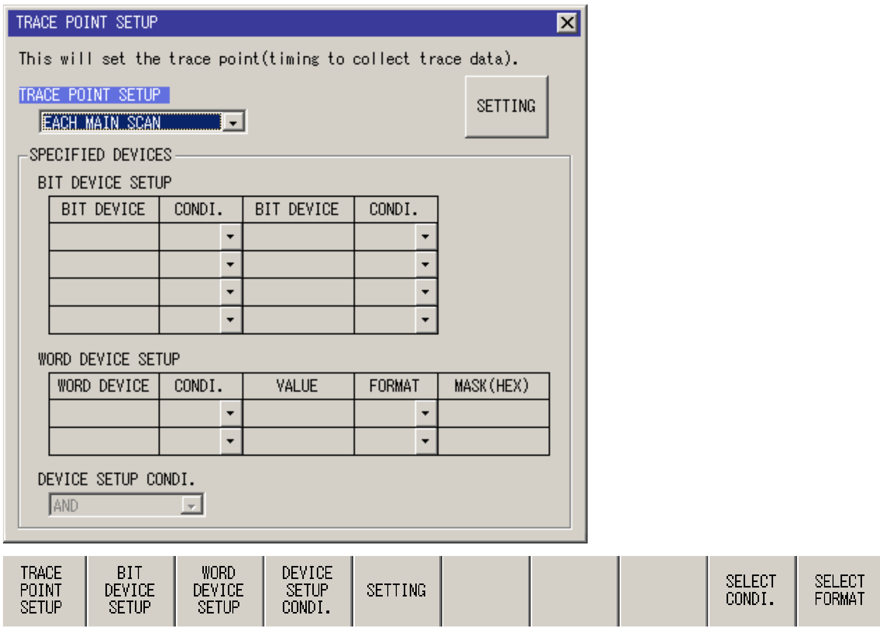
Every main scan (Default)	Trace data is collected every time after main scan is executed.
Every high speed scan	Trace data is collected every time after PLC high speed processing scan is executed. ***Up to 8 points of bit device can be set for the trace data.**
Trace data change	Trace data is collected when the device status set in the trace data setting at every scan of main processing is changed. When multiple devices are set in the trace data setting, trace data is collected even one of the devices is changed. ***Up to 8 points of bit device and 2 points of word device can be set for the trace data.**
Device designation	Trace data is collected when the device setting condition registered in the device specification area has been established at every scan of main processing. *8 points of bit device and 2 points of word device can be specified.

Trace point condition at the time of device designation is as follows.

	Condition	Establishment condition
Bit device setting	↑*1	When a bit device has changed from OFF to ON.
	↓	When a bit device has changed from ON to OFF.
	ON	When a bit device is ON.
	OFF	When a bit device is OFF.
Word device setting	= *1	When equal to the value
	<>	When differed from the value
	<	When smaller than the value
	>	When greater than the value
	<=	When equal to the value or less
	>=	When equal to the value or more
	Mask	After masking (AND) the traced device value with the mask value, establishment condition is checked. Example) Device D100 Current value 12A5(Hexadecimal) Trace condition = Trace condition value 1200(Hexadecimal) Trace condition mask FF00 (Hexadecimal) (1) Mask the trace point data. 12A5 & FF00 = 1200 (2) Check the trace condition. Equal (=) to the value? 1200 = 1200 ∴Condition established
Device setting condition	AND *1	All the conditions set with bit device and word device have been established.
	OR	Any one of the conditions set with bit device and word device has been established.

*The value for the word device setting is displayed in decimal and hexadecimal.
*1 is the default condition.

(Note 1) The setting of "TRACE POINT SETUP" etc. will not be valid until the "SETTING" menu key is pressed
(Note 2) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.
(Note 3) Setting is not possible during trace. (An error message appears.)



(1) Displaying the "TRACE POINT SETUP" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → TRACE POINT SETUP menu. When selected, the "TRACE POINT SETUP" popup screen will appear.

(2) Setting the trace point

(a) Select the TRACE POINT SETUP menu key.

(b) Press the ↑/↓ key and select a setting method from the list.

(3) Setting the bit device

*This is the trace point setting and set only when "SPECIFIED DEVICES" is selected.

(a) Select the BIT DEVICE SETUP menu key. When selected, the cursor is displayed in the table for bit device specification.

(b) Move the cursor to the bit device section and specify the device.

*Press the INPUT key after the device entry, and then "↑" is set automatically in the "CONDI." area.

*When the cursor is moved to another cell by using an arrow key, "↑" is set automatically in the "CONDI." area.

*An error occurs if an invalid device is specified.

(c) Select the SELECT CONDI. menu. When selected, the following conditions are changed in turns.

↑	Condition is established when a device has changed from OFF to ON.
↓	Condition is established when a device has changed from ON to OFF.
ON	Condition is established when a device is ON.
OFF	Condition is established when a device is OFF.

(4) Setting the word device

*This is the trace point setting and set only when "SPECIFIED DEVICES" is selected.

(a) Select the WORD DEVICE SETUP menu key. When selected, the cursor is moved to the table for word device specification.

(b) Move the cursor to the word device section and specify the device. Press the INPUT key or → key so that the cursor is moved automatically to the "CONDI." section and "=" is set as the condition.

*An error occurs if an invalid device is specified.

(c) Select the SELECT CONDI. menu. When selected, the following conditions are changed in turns.

=		Condition is established when a device is equal to the value.
<>		Condition is established when a device is differed from the value.
<		Condition is established when a device is smaller than the value.
>		Condition is established when a device is greater than the value.
<=		Condition is established when a device is equal to the device or less.
>=		Condition is established when a device is equal to the device or more

(d) Press the VALUE menu.

(e) Directly input the word device establishment condition. Press the INPUT key or → key to determine the entry.

*Decimal numbers are used in the initial setting. Enter the data in decimal.

(f) Mask data is input in hexadecimal. Mask data is used as shown below.

((Device current value) &(Mask data))[Setting condition] (Setting value)

Example) Condition establishment judgment in the following case.

Device D100

Current value 12A5 (Hexadecimal)

Setting condition =

Setting value 1200 (Hexadecimal)

Mask FF00 (Hexadecimal)

((Device current value) & (Mask data)) <Setting condition> (Setting value)

((12A5) & (FF00)) = (1200)

(1) Mask the current value.

12A5 & FF00 = 1200

(2) Check the trace condition.

Equal (=) to the value?

1200 = 1200

∴Condition established

(5) Deleting the bit device and word device
(a) Move the cursor to the device to be deleted from the table.
(b) Press the DELETE key to delete the device.
*Note that, however, deletion will not be valid until the SETTING menu key is pressed.

(6) Device setting condition
(a) The condition of combinations of bit device and word device is set.

AND condition	Trace (data collection) is executed when the condition of all the devices registered with bit device and word device has been established.
OR condition	Trace (data collection) is executed when the condition of any one of the devices registered with bit device and word device has been established.

(7) Determination of settings
(a) Select the SETTING menu key. When selected, the "TRACE POINT SETUP" screen is closed and the changes will be validated.
*An error occurs if the trace point is not correctly set. (The "TRACE POINT SETUP" screen will not be closed.)
*An error occurs if the trace point setting is "SPECIFIED DEVICES" and the device specification has not been made. (The "TRACE POINT SETUP" screen will not be closed.)

(8) Invalidate the changes and close the popup screen.
(a) Press the ◁ menu key.
*The changed settings will be invalid.

10.3.4 Trigger Point Setting

The trigger (condition) that can be the origin (0 point) at the time of trigger execution can be set.
Trigger point is specified in the following methods.

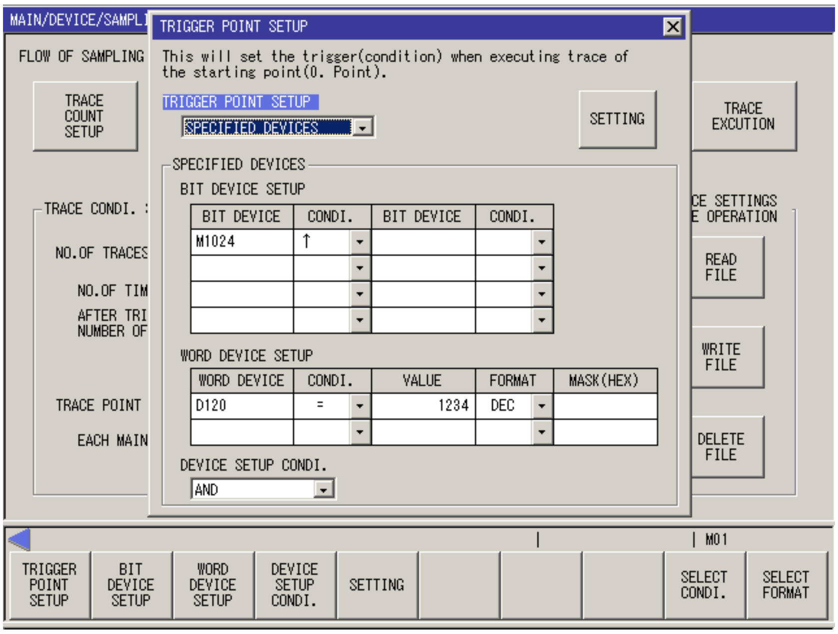
Screen input (Default)	Trigger is set with "EXECUTE TRIGGER" on the "TRACE EXECUTION" screen.
Device designation	The following two conditions must be satisfied for trigger setting: 1) The "EXECUTE TRIGGER" button in the "TRACE EXECUTION" screen is pressed 2) The setting condition of the device registered in the device specification area is established *8 points of bit device and 2 points of word device can be specified.

The following trace point conditions can be set at the time of device specification.

	Condition	Establishment condition
Bit device setting	↑*1	When a bit device has changed from OFF to ON.
	↓	When a bit device has changed from ON to OFF.
	ON	When a bit device is ON.
	OFF	When a bit device is OFF.
Word device setting	= *1	When equal to the value
	<>	When differed from the value
	<	When smaller than the value
	>	When greater than the value
	<=	When equal to the value or less
	>=	When equal to the value or more
	Mask	After masking (AND) the traced device value with the mask value, establishment condition is checked. Example) Device D100 Current value 12A5(Hexadecimal) Trace condition = Trace condition value 1200(Hexadecimal) Trace condition mask FF00 (Hexadecimal) (1) Mask the trace point data. 12A5 & FF00 = 1200 (2) Check the trace condition. Equal (=) to the value? 1200 = 1200 ∴Condition established
Device setting condition	AND *1	All the conditions set with bit device and word device have been established.
	OR	Any one of the conditions set with bit device and word device has been established.

* Word device setting value can be displayed both in decimal and hexadecimal.
*1 is the default condition.

(Note 1) The setting of "TRIGGER POINT SETUP", etc. will not be valid until the "SETTING" menu key is pressed.
(Note 2) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.
(Note 3) Setting is not possible during trace. (An error message appears.)



(1) Displaying the "TRIGGER POINT SETUP" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → TRIGGER POINT SETUP menu. When selected, the "TRIGGER POINT SETUP" popup screen will appear.

(2) Setting the trigger point

(a) Select the TRIGGER POINT SETUP menu key.

(b) Press the ↑/↓ key and select a setting method.

(3) Setting the bit device

*This is the trigger point setting and set only when "SPECIFIED DEVICES" is selected.

(a) Select the BIT DEVICE SETUP menu key. When selected, the cursor is displayed in the table for bit device specification.

(b) Move the cursor to the bit device section and specify the device.

* Press the INPUT key after the device entry, and the "↑" is set automatically in the "CONDI." area.

*When the cursor is moved to another cell by using an arrow key, "↑" is set automatically in the "CONDI." area.

*An error occurs if an invalid device is specified.

(c) Select the SELECT CONDI. menu. When selected, the following conditions are changed in turns.

↑	Condition is established when a device has changed from OFF to ON.
↓	Condition is established when a device has changed from ON to OFF.
ON	Condition is established when a device is ON.
OFF	Condition is established when a device is OFF.

(4) Setting the word device

* This is the trigger point setting and set only when "SPECIFIED DEVICES" is selected.

(a) Select the WORD DEVICE SETUP menu key. When selected, the cursor is displayed in the table for word device specification.

(b) Move the cursor to the word device section and specify the device.

Press the INPUT key or → key so that the cursor is moved automatically to the "CONDI." section and "=" is set as the condition.

*An error occurs if an invalid device is specified.

(c) Select the SELECT CONDI. menu. When selected, the following conditions are changed in turns.

=	Condition is established when a device is equal to the value.
<>	Condition is established when a device is differed from the value.
<	Condition is established when a device is smaller than the value.
>	Condition is established when a device is greater than the value.
<=	Condition is established when a device is equal to the device or less.
>=	Condition is established when a device is equal to the device or more

(d) Select the VALUE menu.

(e) Directly input the word device establishment condition. Press the INPUT key or → key to determine the entry.

*Decimal numbers are used in the initial setting. Enter the data in decimal.

(f) Mask data is input in hexadecimal. Mask data is used as shown below.

((Device value) AND (Mask data)) <Setting condition> (Setting value)

Example) Condition establishment judgment in the following case.

Device D100

Current value 12A5 (Hexadecimal)

Setting condition =

Setting value 1200 (Hexadecimal)

Mask FF00 (Hexadecimal)

((Device current value) & (Mask data)) <Setting condition> (Setting value)

((12A5) & (FF00)) = (1200)

(1) Mask the current value.

12A5 & FF00 = 1200

(2) Check the trace condition.

Equal (=) to the value?

1200 = 1200

∴Condition established

(5) Deleting the bit device and word device

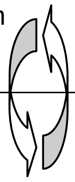
(a) Move the cursor to the device to be deleted from the table.

(b) Press the [DELETE] key to delete the device.

*Note that, however, deletion will not be valid until the [SETTING] menu key is pressed.

(6) Device setting condition

(a) The condition of combinations of bit device and word device is set.

AND condition		Trace (data collection) is executed when the condition of all the devices registered with bit device and word device has been established.
OR condition		Trace (data collection) is executed when the condition of any one of the devices registered with bit device and word device has been established.

(7) Establishment of setting

(a) Select the [SETTING] menu key. When selected, the "TRIGGER POINT SETUP" screen is closed and the changes will be validated.

*An error occurs if the trigger point is not correctly set. (The "TRIGGER OPINT SETUP" screen will not be closed.)

(8) Invalidate the changes and close the popup screen.

(a) Press the [◁] menu key.

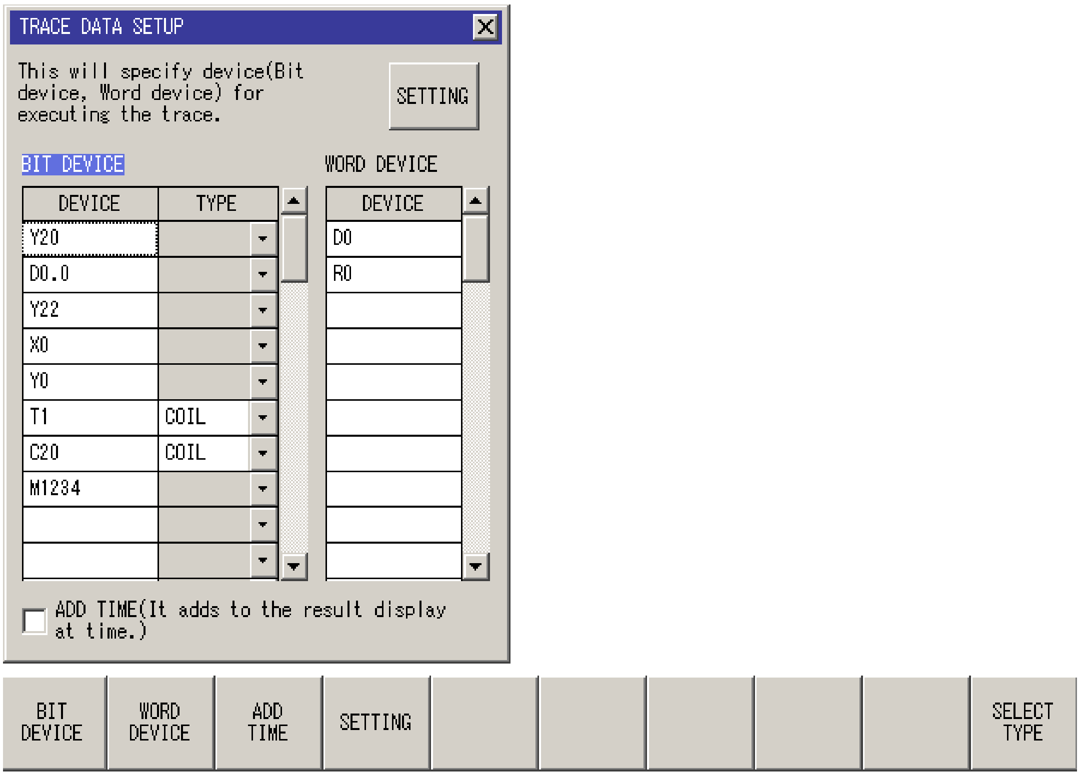
*The changed settings will be invalid.

10.3.5 Trace Data Setting

Devices (bit device, word device) to which trace is executed can be set.

Bit device	Up to 50 devices can be registered. **(Note 1)** When trace point is set in "every high speed scan", valid bit device is limited to 8 points or less. **(Note 2)** When trace point is set in "TRACE DATA CHANGE", valid bit device is limited to 8 points or less.
Word device	Up to 50 devices can be registered. **(Note 1)** When trace point is set in " TRACE DATA CHANGE ", valid word device is limited to 2 points or less.
Time addition (Time is added to the result information)	The time of trace point can be displayed on the "time" field on the trace result display screen. **(Note)** Note that if time addition is carried out, the size of traceable data is smaller than usual.

(Note 1) Changes will not be valid until the SETTING menu key is pressed.
(Note 2) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.
(Note 3) Setting is not possible during trace. (An error message appears.)



(1) Displaying the "TRACE DATA SETUP" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → TRACE DATA SETUP menu. When selected, the "TRACE DATA SETUP" popup screen will appear.

(2) Setting the bit device

(a) Select the BIT DEVICE menu key. When selected, the cursor moves to the table.

(b) Move the cursor and decide where to input the device.

(c) Directly enter the bit device name into the device section.

(d) After device name was entered, press the INPUT key or ↑/↓ key to determine the input.

*When T,ST and C devices are set, "COIL" is automatically set in the type section. If wish to change the type to "CONTACT", move the cursor to "COIL" and select the SELECT TYPE menu key.

*An error occurs if an invalid device is specified.

(3) Setting the word device

(a) Select the WORD DEVICE menu key. When selected, the cursor moves to the table.

(b) Move the cursor and decide where to input the device.

(c) Directly enter the word device name into the device section.

(d) After device name was entered, press the INPUT key or ↑/↓ key to determine the input.

*An error occurs if an invalid device is specified.

(4) Deleting the device in the table

(a) Move the cursor to the device to be deleted in the table.

(b) Press the DELETE key to delete the device.

*Deletion will not be valid until the SETTING menu key is pressed.

(5) Scrolling the table

(a) The cursor on the table can be moved to the top or bottom and then scrolled to the next line. Or, the Page UP/DOWN key can be used to scroll in one unit size.

(6) Addition of time (time is added to the result information) is carried out.

(a) Press the "ADD TIME" button. When pressed, time addition check mark is put. (If pressed once again, the check mark is removed.)

*Changes will not be valid until the SETTING menu key is pressed.

(7) Determination of settings

(a) Select the SETTING menu key. When selected, the "TRACE DATA SETUP" screen is closed and the changes will be valid.

*When trace data is not correctly set, an error occurs. (The "TRACE DATA SETUP" screen will not be closed.)

(8) Invalidate the changes and close the popup screen.

(a) Press the ◁ menu key.

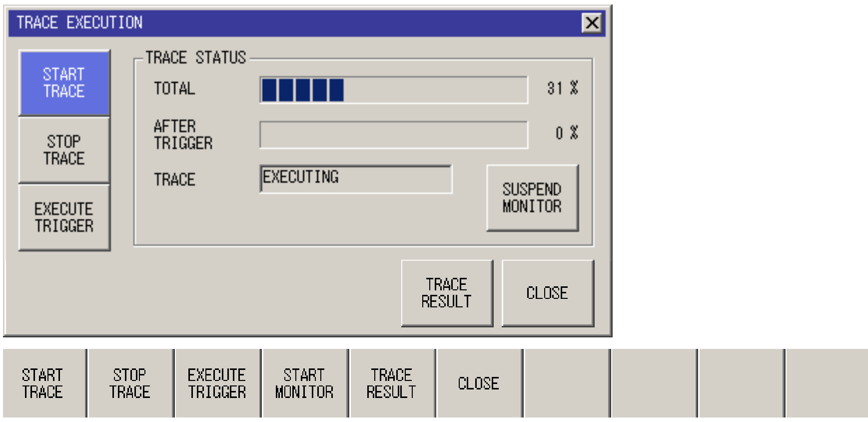
*The changed settings will be invalid.

10.3.6 Trace execution

Trace start/stop and trigger execution are carried out.

Starting trace	Trace is started upon automatically writing the trace condition set on the onboard into the NC control unit.
Stopping trace	Trace is stopped.
Compulsory trigger execution	Trigger can be compulsorily executed from the screen. Even when the trigger point setting is "SPECIFIED DEVICES", trigger can be compulsorily executed.
Monitor start/stop	Trace status monitor start/stop is carried out. When the "trace execution" screen is closed during trace, monitor display is automatically stopped. If you wish to display again, press the "monitor start" button again.
Trace result display	The "TRACE RESULT" screen is displayed.

(Note 1) If trace has been started already, an error occurs and trace will not be started.
(Note 2) If PLC is STOP, an error occurs and trace will not be started.
(Note 3) If trace condition (trace count, trace data) is not set correctly, trace will not be started. (An error message appears.)
(Note 4) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.
(Note 5) When trigger has not been executed and the trace count has been exceeded, trace data will be erased from the oldest one in order.
(Note 6) When trace point is set as "every high speed scan", number of points that is valid with trace data is limited to 8 or less bit device points. (if more than 8 points of device are set, only 8 points from the head can be target.)
(Note 7) If the trace point is "device change", valid number of points for trace data is limited to bit device 8 points or less, word device for 2 or less



(1) Displaying the "TRACE EXECUTION" screen

(a) Select "MAINT" → DEVICE → SAMPLING TRACE の TRACE EXECUTION menu. When selected, the "TRACE EXECUTION" popup screen will appear.

(2) Starting trace

(a) Select the START TRACE menu key.

When selected, execution status is displayed in "TRACE STATUS".

(3) Stopping trace

(a) Select the STOP TRACE menu key.

Trace is stopped.

(4) Executing trace

(a) Select the EXECUTE TRIGGER menu key.

*Compulsory trigger is executed from the screen.

(5) Monitor start/stop

(a) Select the START MONITOR / SUSPEND MONITOR menu key.

Display of trace status is started or sopped.

(6) Displaying trace result

(a) Select the TRACE RESULT menu key. When selected, the "TRACE EXECUTION" screen is closed and the "TRACE RESULT" screen is displayed. (Refer to "10.3.7 Trace Result Display" for details.)

(7) Trace status display

When trace status is being monitored, the button displays [Suspend monitor]. To stop monitoring, click the button. When not monitoring, the button displays [Start monitor]. To start monitoring, click the button.

[Trace] within [Trace status] includes the following four display items.

- Executing: Trace is being executed.
- Suspended: Trace has been stopped.
- Execution failed: Trace is not executed.
- Finished: Trace has been completed.

(8) Close the popup screen.

(a) Press the CLOSE menu key or the menu key.

When the screen is closed, trace status monitoring is stopped.

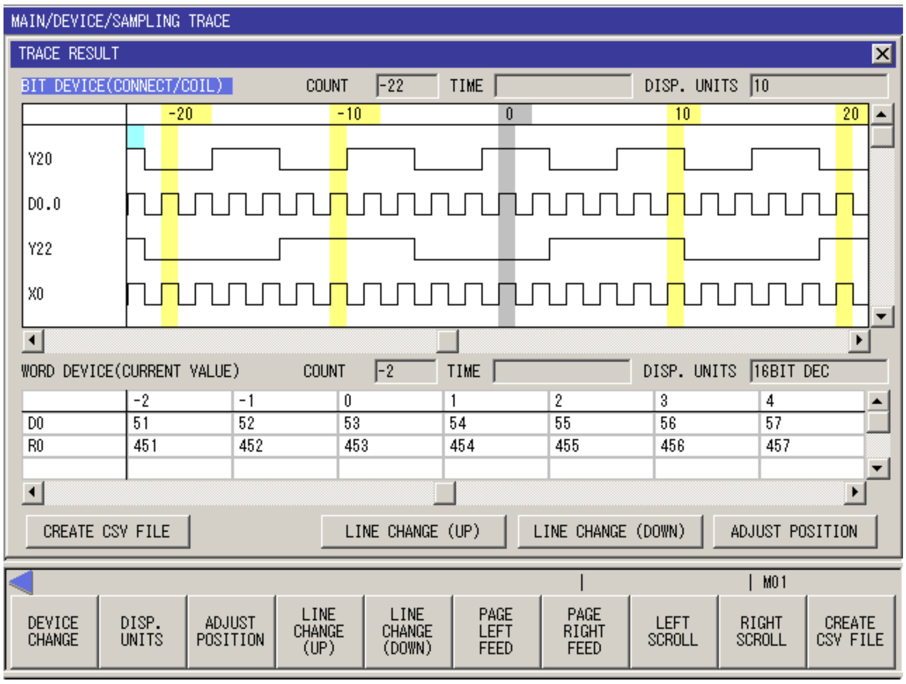
10.3.7 Trace Result Display

After trace execution, the collected trace result can be displayed in chronological order. Also, the output result can be output to an external device in the CSV file format.

Bit device	Device display area	The bit device (contact/coil) trace result is displayed on the upper part of a screen. Variation of device is displayed chronologically, regarding the trigger position as the base point (0).
	COUNT	Trace count at the cursor position in the bit device display section is displayed.
	TIME	Trace time (hour, min., sec.) at the cursor position in the bit device display section is displayed.
	DISP. UNITS	Display scale (display trace count) can be changed within the following range. 10 (default), 20, 50, 100
Word device	Device display area	The word device (current value) is displayed on the lower part of a screen. Variation of value is displayed chronologically, regarding the trigger position as the base point (0).
	COUNT	Trace count at the cursor position in the word device display section is indicated.
	TIME	Trace time (hour, min., sec.) at the cursor position in the word device display section is indicated.
	DISP. UNITS	16-bit in decimal (Default) 16-bit in hexadecimal 32-bit in decimal 32-bit in hexadecimal Note) In order to realize a 32-bit display, upper/lower level devices must be set in the trace data.

Common	DEVICE CHANGE	Move the cursor between bit device display area and word device display area. On the screen where the cursor exists, the details of result can be scrolled.
	DISP. UNITS	Change the display unit of the screen at the cursor position in the bit device or word device display area.
	ADJUST POSITION	Adjust the display position of one screen (ex. Word device display area) based on the cursor position of the screen where the cursor exists (ex. Bit device display area).
	LINE CHANGE (UP)	Move the device at the cursor position in the bit device or word device display area upward.
	LINE CHANGE (DOWN)	Move the device at the cursor position in the bit device or word device display area downward.
	PAGE LEFT FEED	Move the display position to one screen page left from the screen at the cursor position in the bit device or word device display area.
	PAGE RIGHT FEED	Move the display position to one screen page right from the screen at the cursor position in the bit device or word device display area.
	LEFT SCROLL	Move the display result of the screen at the cursor position in the bit device or word device display area to the left for one trace amount.
	RIGHT SCROLL	Move the display result of the screen at the cursor position in the bit device or word device display area to the right for one trace amount.
	CREATE CSV FILE	The trace result can be output to an external file in the CSV format.

(Note 1) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.



(1) Displaying the "TRACE RESULT" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → TRACE RESULT menu. When selected, the "TRACE RESULT" popup screen will appear.

(2) Switching cursor between bit device and word device

(a) Press the DEVICE CHANGE menu. When the DEVICE CHANGE menu is pressed, the cursor on the screen moves in turn between bit device section and word device section.

(3) Move the result in the bit device section and word device section vertically.

(a) Move the cursor to a desired screen with the DEVICE CHANGE menu.

(b)Move the cursor up and down with the ↑/↓ key. The screen is scrolled up/down according to the cursor movement.

(c) Press the PAGE UP / PAGE DOWN key to move one page up/down.

(4) Move the result in the bit device section and word device section horizontally.

(a) Move the cursor to a desired screen with the DEVICE CHANGE menu.

(b) Move the cursor right and left with the ← / → key. The screen is shifted to the right/left according to the cursor movement.

(c) Press the PAGE LEFT FEED / PAGE RIGHT FEED menu to move one page to the right/left.

(5) Move the result of bit device and word device for the amount of one trace data.

(a) Move the cursor to a desired screen with the DEVICE CHANGE menu.

(b) Press the LEFT SCROLL / RIGHT SCROLL menu.

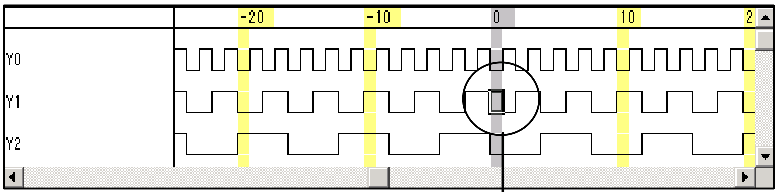
(6) Switching the display unit

(a) Move the cursor to a desired screen with the DEVICE CHANGE menu.

(b) Press the DISP.UNITS menu. Display unit is changed in turn.

(7) Align the bit device display position with the word device display position.

(a) Select the DEVICE CHANGE menu. When selected, the bit device display position and the word device display position can be aligned based on the cursor position in the currently selected table.



Data at the cursor position is displayed.

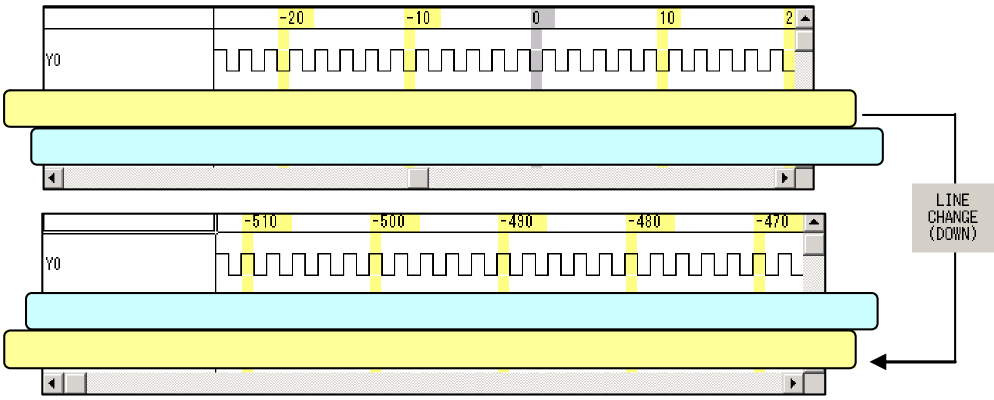
	0	1	2	3	4	5
D0	2096	2097	2098	2099	2100	2101
R20	10480	10480	10480	10480	10490	10500
D100	-2097	-2097	-2099	-2101	-2101	-2101

(8) Switch orders of devices in the list

(a) Move the cursor to a desired screen with the DEVICE CHANGE menu.

(b) Place the cursor at the device position to which you wish to move.

(c) Press the LINE CHANGE (UP) menu key to move one line up; press the LINE CHANGE (DOWN) menu key to move one line down.



(9) Output the trace result to an external device in the CSV format.

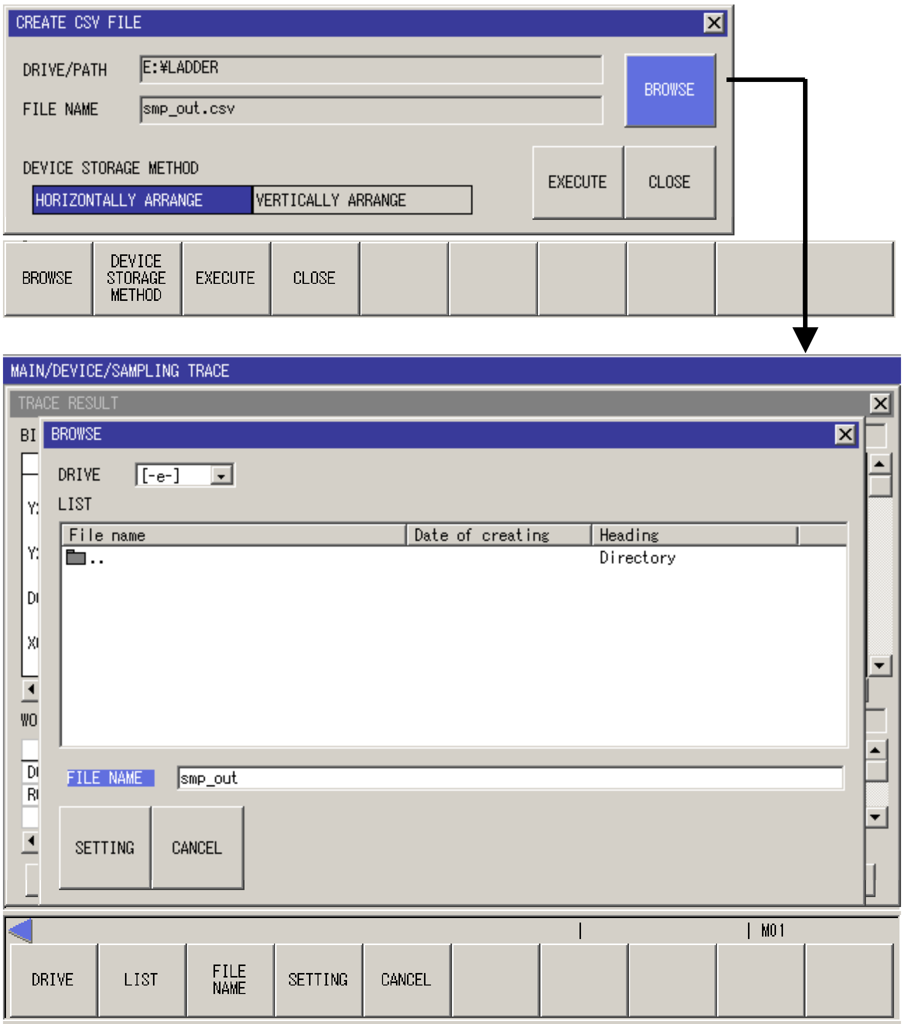
(a) Select the CREATE CSV FILE menu. When selected, the "CREATE CSV FILE" popup screen will appear. Refer to "10.3.8 Creating CSV File" for details.

(10) Close the "TRACE RESULT" screen.

(a) Select the CLOSE menu key or the ◁ menu key. When selected, the "TRACE RESULT" popup screen will appear.

10.3.8 Creating CSV File

Trace result can be output to an external device (IC card, etc.) in the CSV format.



(1) Displaying the "CREATE CSV FILE" screen

(a) Select "MAIN"→ DEVICE → SAMPLING TRACE → TRACE RESULT → CREATE CSV FILE menu.
When selected, the "CREATE CSV FILE" popup screen will appear.

(2) Displaying the "BROWSE" screen

(a) Select "MAIN"→ DEVICE → SAMPLING TRACE → TRACE RESULT → CREATE CSV FILE → BROWSE menu. When selected, the "BROWSE" popup screen will appear.

* If no path is set, set the route of the drive which runs the onboard as the path.

* When the "BROWSE" screen is displayed for the first time after starting the onboard, the same path as that of the external project will be the default path. Once a path has been set on the "BROWSE" screen, the set path will be the default path.

(3) Specifying the CSV file output destination

(a) Select the BROWSE menu. When selected, the "BROWSE" popup screen will appear and the path and file of the output destination can be specified.

(b) The drive and path of the output destination is selected on the "BROWSE" popup screen.

(c) File name is directly input by selecting the FILE menu on the "BROWSE" popup screen.

*Characters that can be used for a file name are as follows.

Item	Restrictions on input characters
File name	8 one-byte alphanumerical characters only. (Two-byte characters are prohibited.) (Signs and space codes cannot be used.)

(d) Select the SETTING menu on the "BROWSE" popup screen. When selected, the specified drive/path and file name are verified and the "BROWSE" popup screen is closed.

* When the CANCEL menu or the ◁ menu is selected on the "BROWSE" popup screen, the specified drive/path and file name will be disabled.

* When the set file name is illegal, an error appears. (The "BROWSE" screen will not be closed at this time.)

(4) Specifying the device storage method

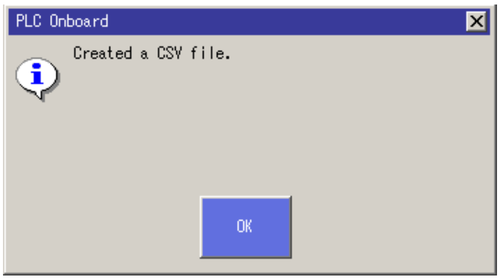
(a) When outputting in CSV, data alignment (vertical or horizontal) can be specified.

(b) Press the "DEVICE STORAGE METHOD" menu. When selected, vertical or horizontal way of device storage method can be changed.

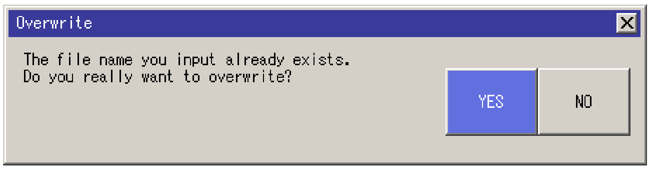
(5) Creating the CVS file

(a) Select the EXECUTE menu. When selected, the trace result data is created in the CSV file format with the specified drive/path and file name.

When successfully created, the following popup screen is displayed.

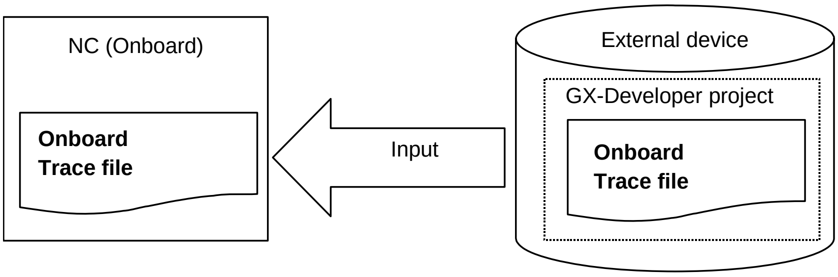


*When a file already exists under the same name, a message confirming the overwriting of this file appears. If selected "Yes", overwriting is executed.



10.3.9 File Input

Trace data created with the GX-Developer on the external device can be read into the onboard.

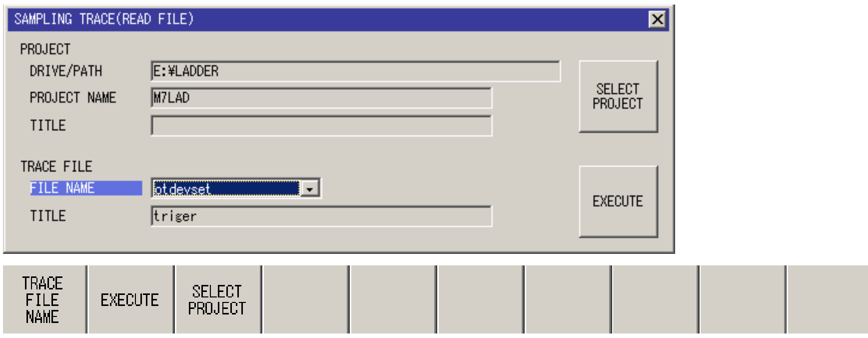


Input data	Trace condition	Trace count, trace point, trigger point, trace data, etc.
	Trace result	Trace data of bit device and word device
File format	GX-Developer project's trace file or the trace file written with the onboard **Note) Trace file format is differed between GX-Developer and onboard.**	

(Note 1) Trace data (trace condition + trace result) is discarded by turning the NC power OFF.

(Note 2) Input/output of trace file is not possible during trace.

(Note 3) Trace file created with the GX-Developer cannot be read if the trace file includes the trace conditions that are not supported by the MITSUBISHI CNC.



(1) Displaying the "SAMPLING FILE (WRITE FILE)" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → WRITE FILE menu. When selected, the "SAMPLING TRACE (WRITE FILE)" popup screen will appear.

(2) Inputting the sampling trace file

(a) Select the SELECT PROJECT menu and select the project to which sampling trace data is read. Refer to "10.3.12 Selecting Project" for details.
If a project has been opened already by the external file operation "OPEN PROJECT" or "SAVE PROJECT", that project will serve as the default project.

(b) Select the TRACE FILE NAME menu. When selected, the cursor is move to the file name section.

(c) Press the ↑/↓ key and specify the sampling trace file name to be input.

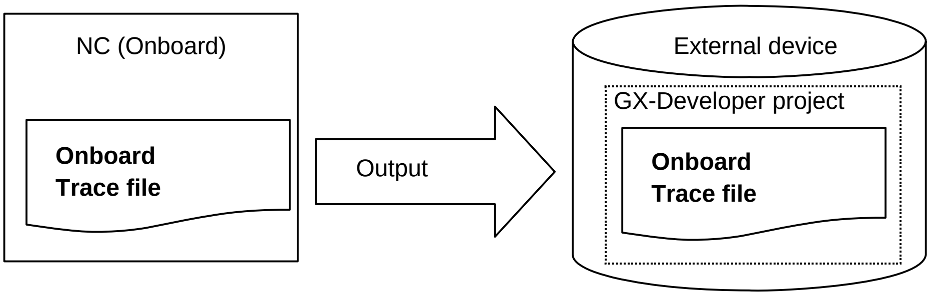
(d) Select the EXECUTE menu. When selected, the specified sampling trace file is read into the onboard. After reading, "SAMPLING TRACE (WRITE FILE)" screen is closed.

(3) Stop the file input and close the popup screen.

(a) Press the ◁ menu key. Writing of file will not be executed.

10.3.10 File Output

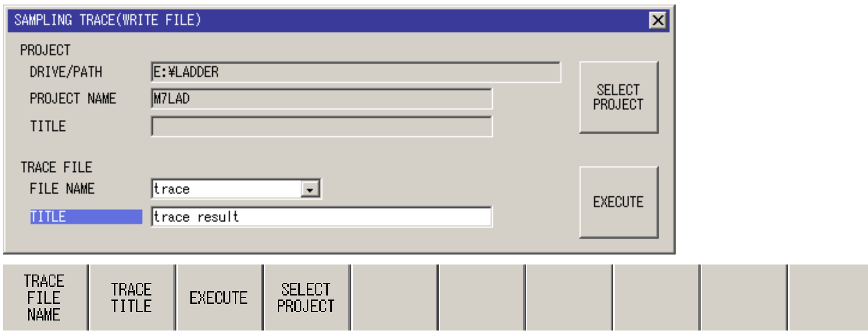
Trace data created with the onboard can be saved into the external device in the GX-Developer project format.



Output data	Trace condition	Trace count, trace point, trigger point, trace data, etc.
	Trace result	Trace data of bit device and word device
File format	Trace file that is unique to the onboard	

(Note 1) Input/output o trace file is not possible during trace.

(Note 2) The trace file written with the onboard cannot be read with the GX-Developer.
(There is no compatibility)



(1) Displaying the "SAMPLING TRACE (WRITE FILE)" screen

(a) Select "MAIN" → DEVICE → SAMPLING TRACE → WRITE FILE menu. When selected, the "SAMPLING TRACE (READ FILE)" popup screen will appear.

(2) Outputting the sampling trace file

(a) Select the SELECT PROJECT menu and specify the project to be output. Refer to "10.3.12 Selecting Project" for details.
If a project has been opened already by the external file operation "OPEN PROJECT" or "SAVE PROJECT", that project will serve as the default project.

(b) Select the TRACE FILE NAME menu. When selected, the cursor moves to the file name field.

(c) Press the ↑/↓ key to specify the file name of the sampling trace to be output.

(d) Select the TRACE TITLE menu to directly input the title.

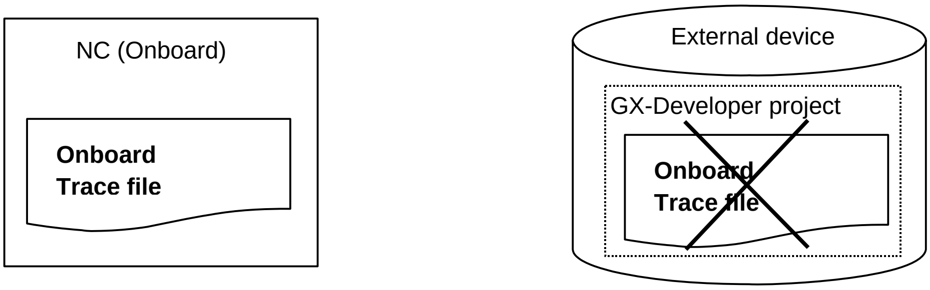
(e) Select the "EXECUTE" menu. When selected, output is executed with the specified project or trace file name. When output is done, the "SAMPLING TRACE (READ FILE)" screen is closed.
*If a trace file with the same name already exists in the output destination project, confirmation of overwriting is prompted before execution.

(3) Stop the file output and close the popup screen.

(a) Press the ◁ menu key. Reading of file will not be executed.

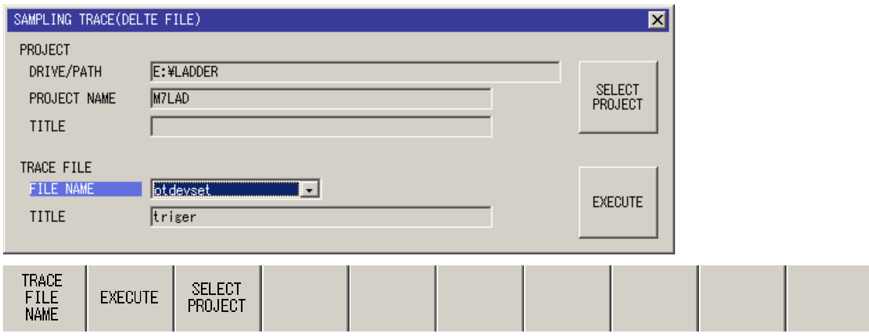
10.3.11 Deleting File

Trace data file on the external device can be deleted.



Deletion data	Trace condition	Trace count, trace point, trigger point, trace data, etc.
	Trace result	Trace data of bit device and word device
File format	Trace file output with the onboard	

(Note 1) Only the trace file that has been output with the onboard can be deleted. Trace file created with the GX-Developer cannot be deleted.



(1) Displaying the "SAMPLING TRACE (DELETE FILE)" screen

(a) Select "MAIN"→ DEVICE → SAMPLING TRACE → DELETE FILE menu. When selected, the "SAMPLING TRACE (DELETE FILE)" popup screen will appear.

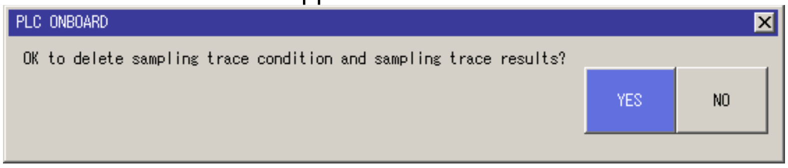
(2) Deleting the sampling trace file

(a) Select the SELECT PROJECT menu to select the project from which the sampling data should be deleted. Refer to "10.3.12 Selecting Project" for details.
If a project has been opened already by the external file operation "OPEN PROJECT" or "SAVE PROJECT", that project will serve as the default project.

(b) Select the TRACE FILE NAME menu. When selected, the cursor is moved to the file name field.

(c) Press the ↑/↓ key and specify the sampling trace file name to be input.

(d) Select the EXECUTE menu. When selected, a screen confirming whether the specified sampling trace file is OK to delete or not appears.



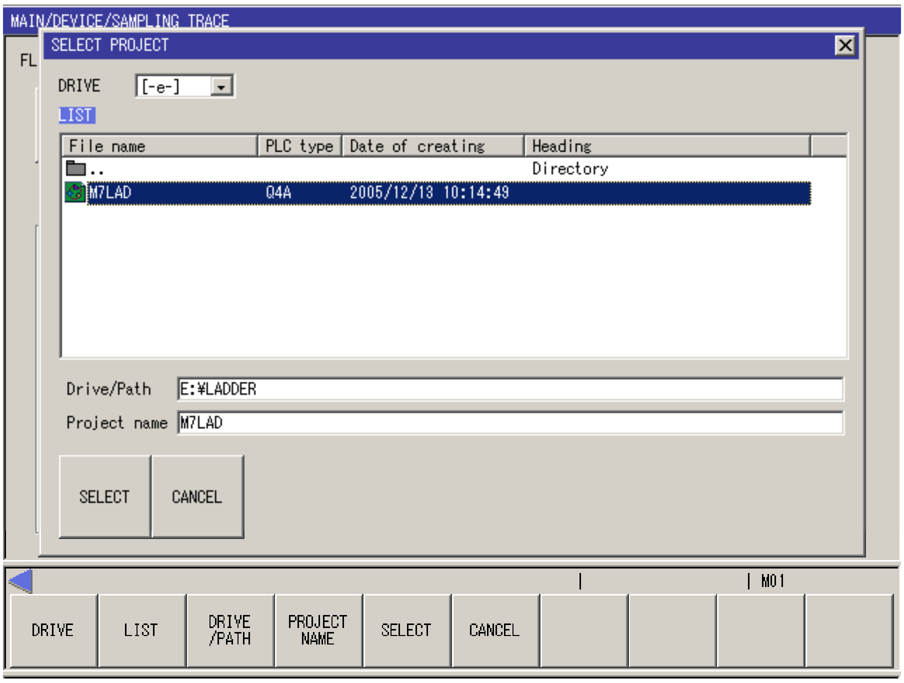
(e) If selected "YES" on the confirmation screen, deletion of trace file will be executed. If selected "NO", deletion will not be executed. After the execution, the SAMPLING TRACE (DELETE FILE) screen will be closed.

(3) Stop the file deletion and close the popup screen.

(a) Press the ◁ menu key. File deletion will not be executed.

10.3.12 Selecting Project

A project for when the sampling trace file operation "READ FILE", "WRITE FILE", or "DELETE FILE" is executed can be selected.



(1) Displaying the "SELECT PROJECT" screen

(a) This is displayed when "MAIN" → DEVICE → SAMPLING TRACE → READ FILE / WRITE FILE / DELETE FILE → SELECT PROJECT menu is pressed.

*If no path is set, set the route of the drive which runs the onboard as the path.

(2) Selecting the project

(a) Select the DRIVE menu to specify the drive.

(b) Select the SELECT PROJECT → LIST menu to specify the project.

By selecting the DRIVE/PATH menu, a direct input is possible, as well, to specify the drive and path.

(c) Select the PROJECT NAME menu to specify the project name.

(d) Press the SELECT menu. When selected, the "SELECT PROJECT" screen will be closed.

*When the CANCEL menu is pressed, the "SELECT PROJECT" screen is closed without selecting a project.

* An error message appears if the specified project name is illegal. (The "SELECT PROJECT" screen will not be closed at this time.)

(3) Stop the file deletion and close the popup screen.

(a) Press the ◁ menu key. File deletion will not be executed.

11. Setting the Parameters

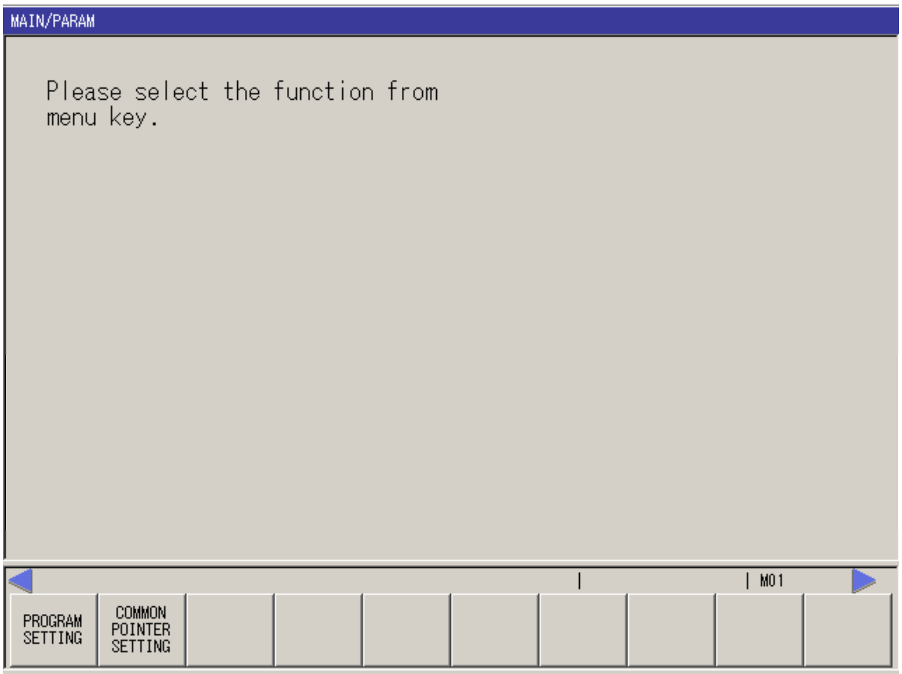
Parameters for controlling the PLC operation can be set.
These parameters are stored in the following data. Open the parameters onto the onboard with "OPEN" before starting.

Data type	Data name
Parameter	param

The parameters which can be set are shown below.

Parameter	Outline
PROGRAM SETTING	The execution order for executing multiple PLCs on the NC is designated.
COMMON POINTER SETTING	The common pointer settings for the multi-program method are displayed.

"PARAM." screen (The screen is an example of 700 Series)



(1) Displaying the "PARAM." screen

Select "MAIN" and PARAM. menu key. When selected, the "PARAM." screen will appear on the full screen.

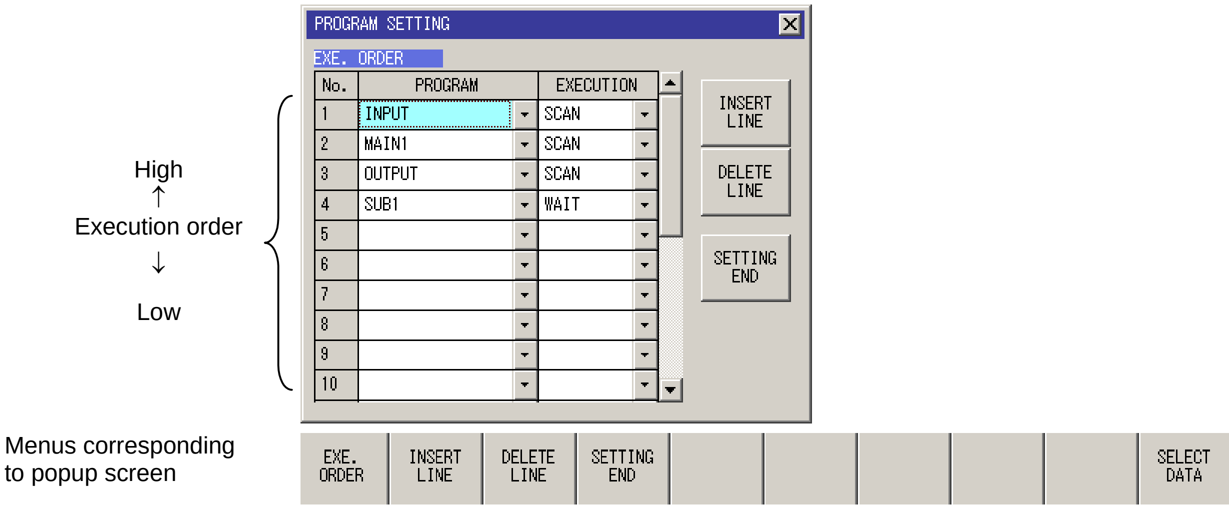
11.1 Setting the Program

700		70
Standard	Simple	
○	○	○

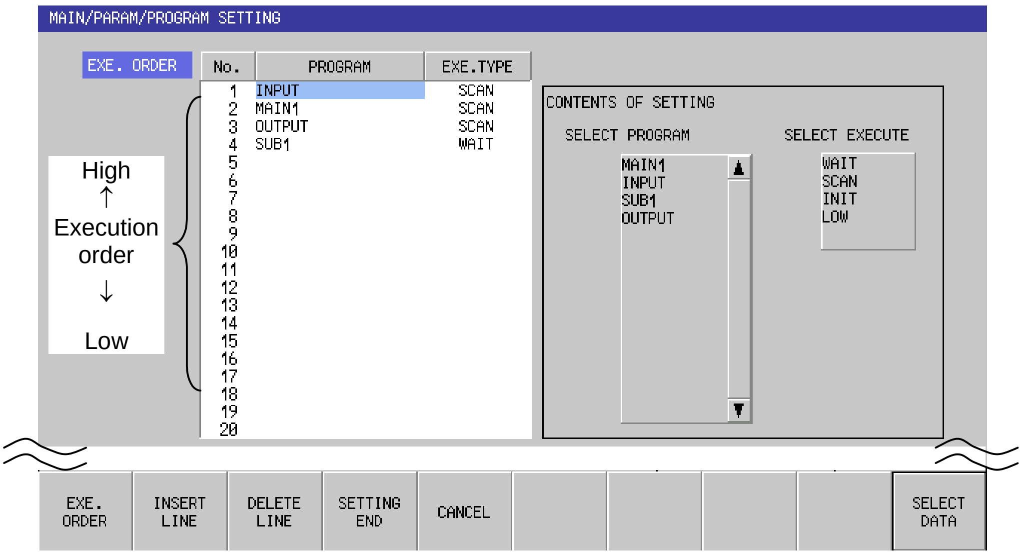
The execution order for executing multiple PLCs on the NC can be designated.
The execution order is registered under "EXE.ORDER" on the "PROGRAM SETTING" popup screen.
Refer to "II PROGRAMMING EXPLANATION 2.4 Multi-program Method" for details on execution order.
Up to 20 programs can be registered.

(Note 1) Open the parameter data and program to be set on the onboard editing area.
(Note 2) Do not register programs with the same name in the "EXE. ORDER" list.

"PROGRAM SETTING" popup screen (In 700 Series)



"PROGRAM SETTING" screen (In 70 Series)

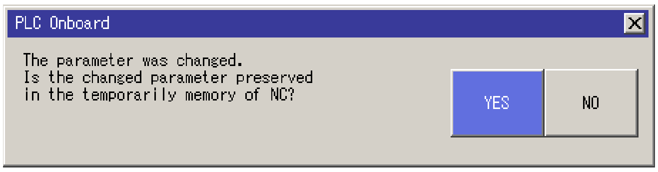


(1) Displaying the "PROGRAM SETTING" popup screen

Select "MAIN" → PARAM. and PROGRAM SETTING menu key. When selected, the "PROGRAM SETTING" popup screen will appear, and the program execution order will appear in the "EXE. ORDER" list.

(2) Designating the program execution order

(a) Press the EXE. ORDER menu key. Move the cursor on the list of "EXE. ORDER" to the row of "PROGRAM NAME".
(b) Press the "Input" menu key at the position where you wish to register. When the "Input" key is pressed, the list of program opened in the onboard edit area is displayed.
(c) Select a program name to set from the list of program and press the "Input" key.
(d) Move the cursor on the list of "EXE. ORDER" to the row of "EXECUTION TYPE".
(e) Press the INPUT key at the position where you wish to register. When the INPUT key is pressed, the list of execution type (SCAN (default), WAIT, INITIAL, LOW-SPEED) is displayed.
(f) Select a type to set from the list of execution type and press the "Input" key.
(g) Repeat the operation (a) to (f) and set the execution order of a program.
(h) Press the "SETTING END" button when all settings are completed.
*The changed contents will be discarded if the SETTING END button is not pressed.
(i) When the SETTING END button is pressed, the following popup screen appears.
When YES is selected, the parameter will be stored in the NC's temporary memory.
When NO is selected, the parameter will not be stored in the NC's temporary memory. (Note that the parameter in the onboard edit area has been changed.)



(3) Inserting a line

(a) Press the EXE. ORDER menu key. Move the cursor on the "EXE. ORDER" list to the line to insert.
(b) Press the INSERT LINE menu key. The line at the cursor position in the "EXE. ORDER" list moves one step down, and then a blank line will be inserted.
*If the maximum number of registerable programs has already been registered, a line cannot be inserted.
*The new settings will not be reflected in the parameter unless the SETTING END button is pressed.

(4) Deleting a line

(a) Press the EXE. ORDER menu key. Move the cursor on the "EXE. ORDER" list to the line to delete.
(b) Press the DELETE LINE menu key. The contents of the line at the cursor position in the "EXE. ORDER" list is deleted.
*The data below the cursor position will not be moved up to the deleted line.
*The new settings will not be reflected in the parameter unless the SETTING END button is pressed.

(5) Displaying the execution order details

Select the EXE. ORDER menu key. The cursor can be moved up and down in the "EXE. ORDER" list to confirm the details of numbers which are not displayed (No. 10 and following).

(6) Closing the popup screen

Press the ◁ menu key.

11.2 Common Pointer Setting

700		70
Standard	Simple	
○		○

The head P No. of the common pointer (which is used to call the sub-routine programs from all the programs) can be changed.
Refer to "II PROGRAMMING EXPLANATION 5.3.11 Pointer P" for details on the common pointer.

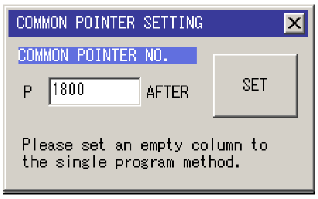
(Note 1) Set the common pointer within the range from P0 to P2047.

(Note 2) The head P No. of the common pointer is set to enable the program execution by the multi program method.

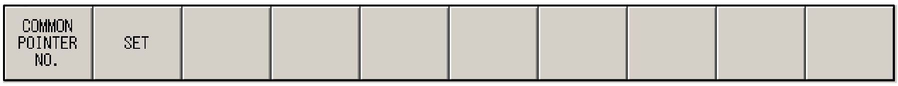
(Note 3) When executing the program with the conventional independent program method, leave it as blank and press the "SET" button.

(Note 4) When using the multi-program method, if the COMMON POINTER NO. setting area is blank (no data is set), the default value, P1800 and after, will be applied as the common pointer.

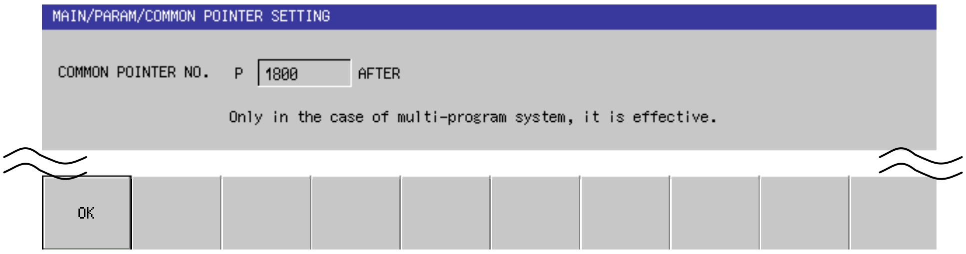
"COMMON POINTER SETTING" popup screen
(In 700 Series)



Menus corresponding to popup screen



"COMMON POINTER SETTING" screen (In 70 Series)

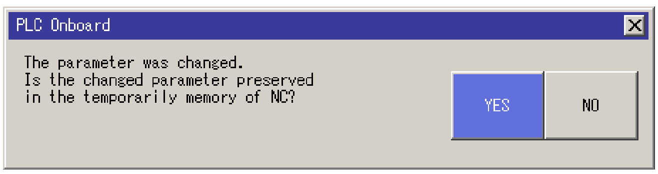


(1) Displaying the "COMMON POINTER SETTING" popup screen

Move to "MAIN" → "PARAMETER" screen, then select the [COMMON POINTER SETTING] menu key.
When selected, "COMMON POINTER NO." popup screen will appear and the common pointer head P No. will be displayed.

(2) Setting the common pointer

(a) Press the [COMMON POINTER No.] menu key. Enter the common pointer head P No.
(b) Press the [SET] menu key.
(c) When the [SET] button is pressed, the following popup screen appears.
When [YES] is selected, the parameter will be stored in the NC's temporary memory.
When [NO] is selected, the parameter will not be stored in the NC's temporary memory.
(Note that the parameter in the onboard edit area has been changed.)



(3) Closing the popup screen

Press the [◁] menu key.

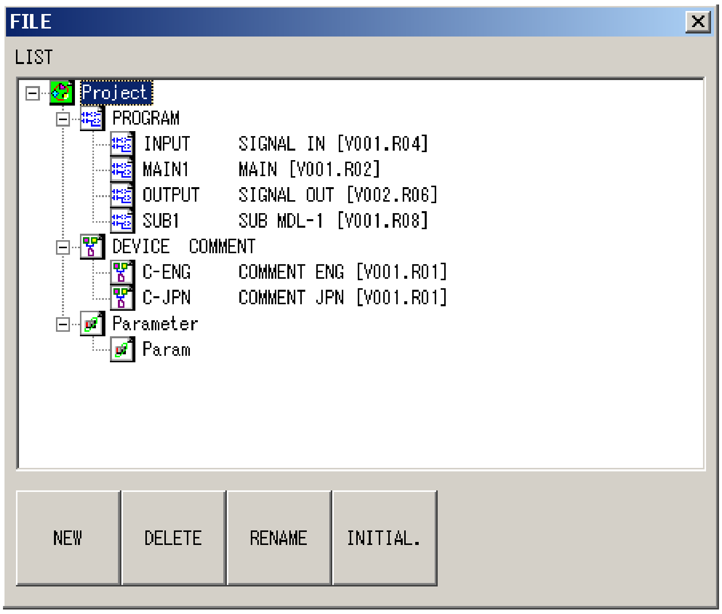
12. File Operations

700		70
Standard	Simple	
○		

In the onboard editing area, PLC data can be added, deleted and renamed, and also the initial settings can be made.

(Note 1) The parameter data cannot be added, deleted or renamed.

"FILE" popup screen



Menus corresponding to popup screen

(1) Displaying the "FILE" popup screen.

Select "MAIN" and FILE menu key. When selected, the "FILE" popup screen will appear.

(2) Displaying the list data

(a) Select the LIST menu key.

(b) If there are many PLC data items which do not fit on one screen, move the cursor in the list to display them all.

(Supplement)

If the DATA DELETE or DATA RENAME menu key is selected after selecting data from the list, the data name selected in the list will be set as the data name (old data name) in the respective popup screen.

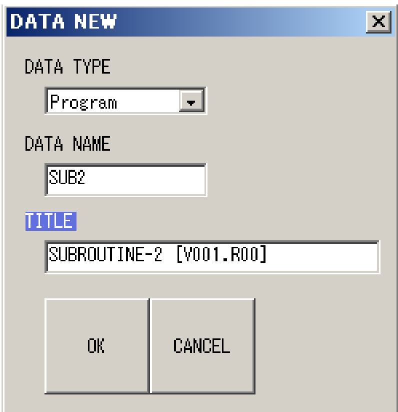
12.1 Adding New Data

700		70
Standard	Simple	
○		

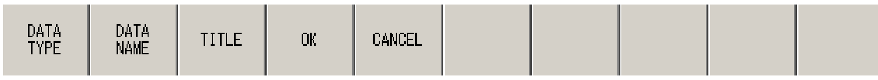
Open new data on the onboard editing area.

(Note 1) The newly added data should be saved with the operations given in "12.2 Saving PLC Data in Temporary Memory". After that, the data can be automatically updated to the temporary memory with the "circuit conversion" operation.
(Note 2) Program and device comment data can be created. Parameters cannot be newly created. When creating program data for a PLC message, refer to "4.2 Types of Data" and set the designated data name.
(Note 3) When there is no parameter data in the onboard editing area, default parameter data is created and used.
(Note 4) When the same name data already exists, a popup screen confirming an overwrite appears.

"DATA NEW" popup screen



Menus corresponding to popup screen



(1) Displaying the "DATA NEW" popup screen

Select "MAIN" → FILE and DATA NEW menu key. When selected, the "DATA NEW" popup menu will appear at the center of the screen.

(2) Adding data

(a) When the DATA NEW menu key is selected, the following "DATA NEW" popup screen will appear.
(b) Select the DATA TYPE menu key, and designate the data type (program or comment).
(c) Select the DATA NAME menu key, and input the name of the data to be newly created.
(d) Select the "TITLE" menu key, and input the data title. (The title can be omitted.)

Item	Limits to input characters
Data name	8 one-byte alphanumeric characters only (no full-byte characters)
Title	32 one-byte alphanumeric characters only (no full-byte characters)

(e) Select the OK menu. The "DATA NEW" popup menu will close, and the newly added PLC data will appear in the data list on the "FILE" popup screen.

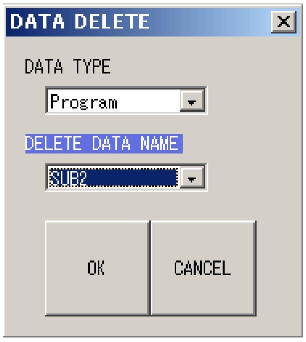
12.2 Deleting PLC Data

700		70
Standard	Simple	
○		

PLC in the onboard editing area can be deleted. (The data in the temporary memory is not deleted.)

(Note 1) The program and device comment data can be deleted. Parameters cannot be deleted.

"DATA DELETE" popup screen



Menus corresponding to popup screen

(1) Displaying the "DATA DELETE" popup menu.

Select "MAIN" → FILE and DATA DELETE menu keys. When selected, the "DATA DELETE" popup screen will appear at the center of the screen.

(2) Deleting PLC data

(a) When the DATA DELETE menu key is selected, the following "DATA DELETE" popup screen will appear.

(b) Select the DATA TYPE menu key, and designate the data type (program or comment).

(c) Select the DELETE DATA NAME menu key, and select the name of the data to be deleted. Select the data name from the list of data currently in the onboard editing area.

(d) Select the OK menu. The "DATA DELETE" popup menu will close, and the designated data will be deleted from the data list on the "FILE" popup screen.

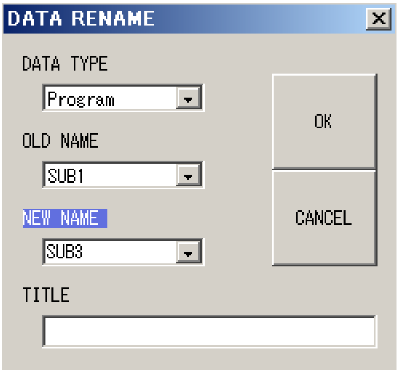
12.3 Renaming the PLC Data

700		70
Standard	Simple	
○		

The PLC data in the onboard editing area can be renamed.

(Note 1) After renaming the data, save it once with the operations explained in "12. 2 Saving PLC Data in Temporary Memory". After that, the data can be automatically updated to the temporary memory with the "circuit conversion" operation.
(Note 2) Program and device comment data can be renamed. Parameters cannot be renamed.
(Note 3) The name cannot be changed to an existing name. Delete the existing name before designating it as a new name.

" DATA RENAME " popup screen



Menus corresponding to popup screen

(1) Displaying the "DATA RENAME" popup screen

Select "MAIN" → FILE and DATA RENAME menu key. When selected, the "DATA RENAME" popup screen will appear at the center of the screen.

(2) Renaming the data

(a) When the DATA RENAME menu key is selected, the following "DATA RENAME" popup screen will appear.
(b) Select the DATA TYPE menu key, and designate the data type (program or comment).
(c) Select the OLD NAME menu key, and designate the old data name. Designate the old data name from the list of data currently in the onboard editing area.
(d) Select the NEW NAME menu key, and designate the new data name.
(e) Select the TITLE menu key, and input the new data title. (The title can be omitted.)

Item	Limits to input characters
New data name	8 one-byte alphanumeric characters only (no full-byte characters)
Title	32 one-byte alphanumeric characters only (no full-byte characters)

*"COMMENT" cannot be used for the device comment's data name.

(f) Select the OK menu. The "DATA RENAME" popup window will close, and the designated data name will be changed in the data list on the "FILE" popup screen.

(3) Closing the popup screen

Press the ◁ menu key.

12.4 Initialization

700		70
Standard	Simple	
○		

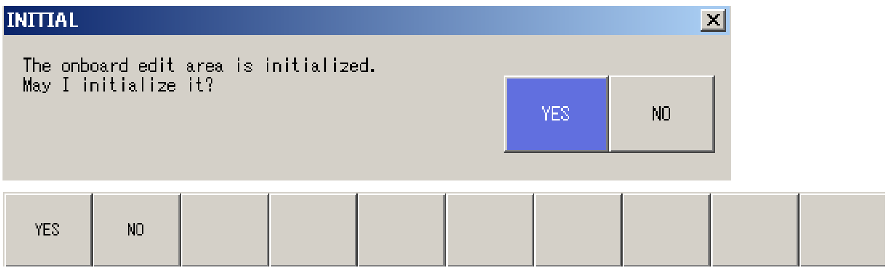
Initialize the onboard editing area.
When initialization is completed, the following PLC data will be created in the onboard editing area.

Data type	Data name	Details
PROGRAM	MAIN	User PLC (ladder) program containing only END instruction is created.
PARAM.	PC parameters (for Japanese)	PC parameter data set to the default value (program settings are not registered) is created.

* The device comment data is not created.

(Note 1) This function is an initial setting carried out to create new PLC (ladders) with the onboard.
(Note 2) The data created with initialization is located in the onboard editing area.
(Note 3) After creating the data with initialization, save it once with the operations explained in "12. 2 Saving PLC Data in Temporary Memory". After that, the data can be automatically updated to the temporary memory with the "circuit conversion" operation.

"INITIAL" popup screen



Menus corresponding to popup screen

(1) Displaying the "INITIAL" popup screen

Select "MAIN" → [FILE] and [INITIAL] menu keys. When selected, the "INITIAL" popup screen will appear at the center of the screen.

(2) Initialization

(a) Confirm the displayed message.
(b) Select the [YES] menu key to initialize.
Select the [NO] menu key to cancel the initialization.
* The popup window automatically closes when the [YES] or [NO] menu key is pressed.
(c) If [YES] is selected and initialization is completed, the data created with initialization will appear in the list on the "FILE" screen.

(3) Closing the popup screen

Press the [◁] menu key.

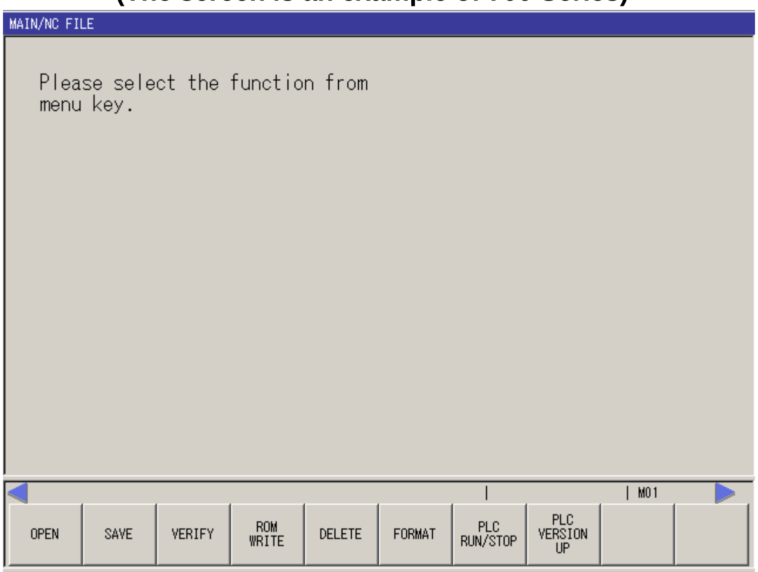
13. NC File Operations

The NC PLC can be RUN or STOP, the PLC data in the temporary memory can be read, written or verified, and the data can be written to the ROM.
Note that functions involving ladder editing or rewriting cannot be used unless the password is completed. (The buttons corresponding to the function are displayed in gray and are disabled.)
Input the password on the NC "MAINTE" screen to release the password.

Function	Outline	700 Series		70 Series
		Standard	Simple	
OPEN	The PLC data is opened from the temporary memory.	○	○*1	○
SAVE	The edited PLC data is saved in the temporary memory.	○	○*2	○*2
VERIFY	The PLC program in the onboard editing area is verified with the PLC program in the temporary memory.	○		
ROM WRITE	The PLC data in the temporary memory is written to the ROM.	○	○	○
DELETE	The PLC data in the temporary memory is deleted.	○		○
FORMAT	The temporary memory is formatted.	○		○
PLC RUN/STOP	The PLC RUN/STOP state can be controlled.	○	○	○
PROGRAM UPDATE	The temporary memory and ROM ladder versions are upgraded.	○		
LIST	The list of file size is displayed.			○

*1 Automatically executed at startup. Not exist in the menu.
*2 Automatically executed at conversion. Not exist in the menu.

"NC FILE" popup screen
(The screen is an example of 700 Series)



(1) Displaying the "NC FILE" screen

Select "MAIN" and the NC FILE menu key. When selected, the "NC FILE" screen will appear on the full screen.

(2) Operation during NC communication

A progress bar appears and shows "Reading..." while data is being exchanged between the onboard and NC.

13.1 Opening PLC Data from the Temporary Memory

700		70
Standard	Simple	
○	Automatic	○

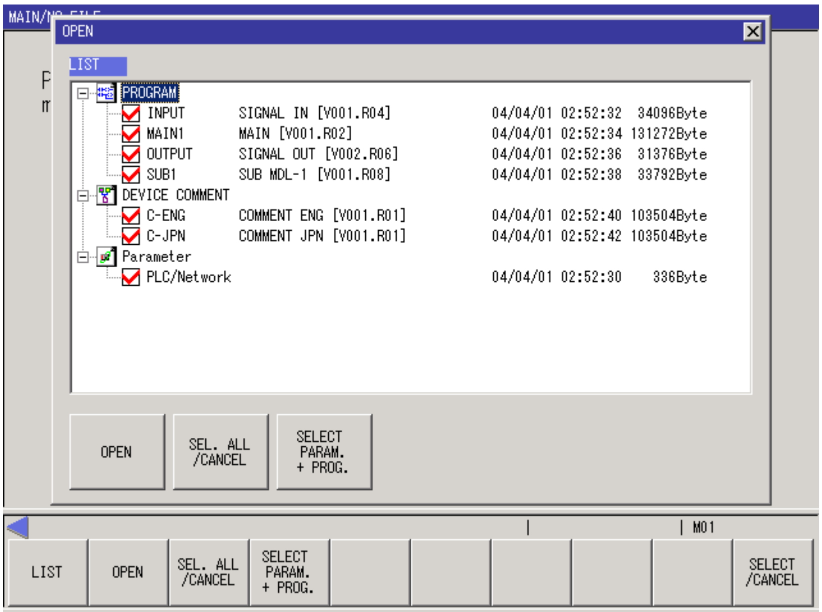
The PLC data can be opened from the temporary memory to the onboard where it can be edited and monitored and the parameters edited.

(Note 1) PLC data read in from the temporary memory and PLC data read in from an external device can both be displayed in the onboard editing area.

(Note 2) If there is no PLC data in the temporary memory, the list will be blank, and the OPEN button will be displayed in gray.

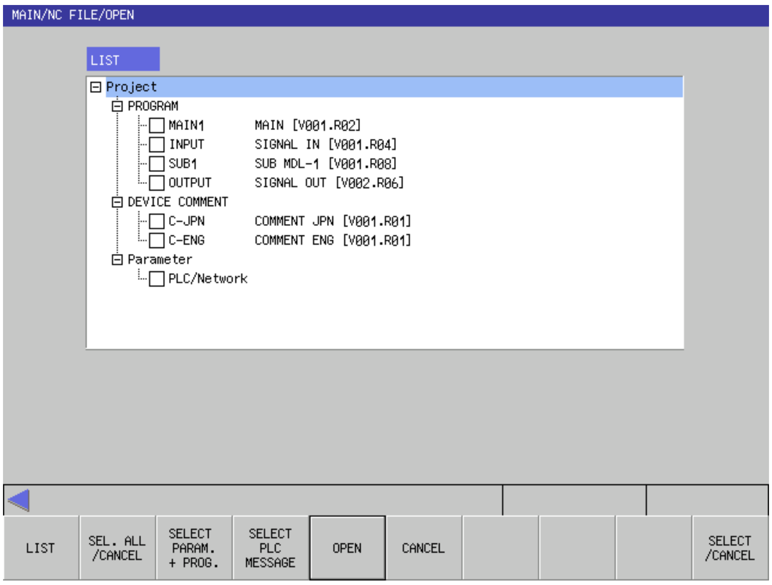
(Note 3) When opening parameter data, parameter in the onboard editing area is used if no parameter data had been selected. If there is no parameter data in the onboard editing area, default parameter data is created and used.

"OPEN" popup screen (The screen is an example of 700 Series)



Menus corresponding to popup screen

"OPEN" screen (The screen is an example of 70 Series)



(1) Displaying the "OPEN" popup screen

Select "MAIN" → NC FILE and the OPEN menu key. When selected, the "OPEN" popup screen will appear at the center of the screen.

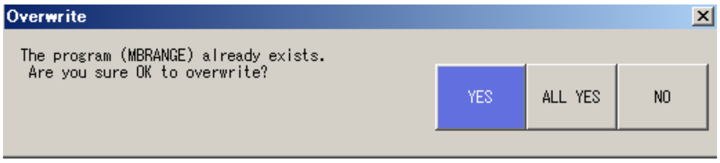
(2) Opening the PLC data

(a) Select the PLC data to be edited from the list. Invalidate the check for data which is not to be opened.

(Note 1) In 70 Series, these check boxes are all invalidated by default.

(b) After selecting the data, select the OPEN menu key. The selected PLC data will be read on the onboard.

* If data with the same name exists, an overwrite confirmation popup screen will appear. Select the operation to be taken.



Menus corresponding to popup screen

(3) Closing the popup screen

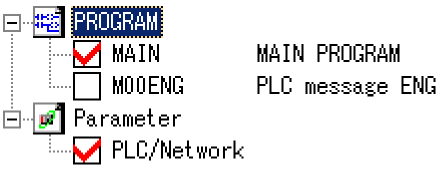
Press the ◁ menu key.

(4) SEL. ALL/CANCEL

Switches valid/invalid of the checkbox ☑ for all the data in the list.

(5) SELECT PARAM. + PROG.

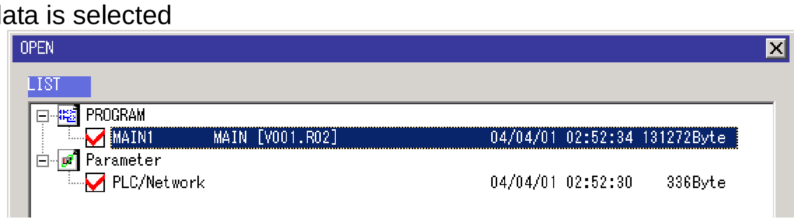
Makes the parameters and program data in the list selectable.
PLC message data is not selected at this time.
(Example)



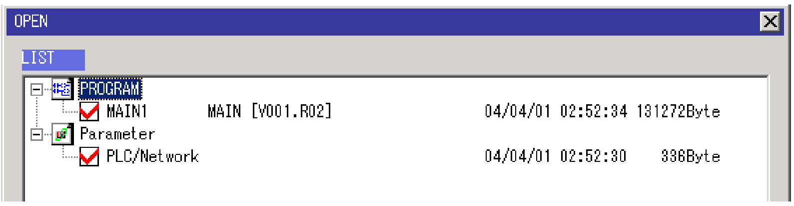
(6) SELECT/CANCEL

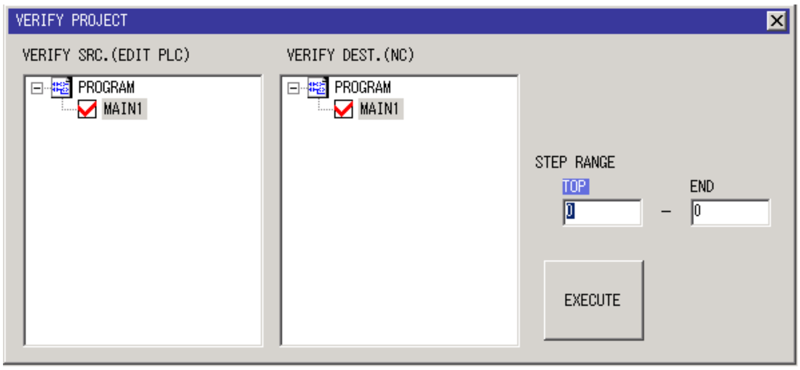
Switches valid/invalid of the checkbox ☑ for the selected data in the list.
"SP (space)" key can also be used for this operation.
When the listed data with a checkbox is not selected, no process occurs.

(Example)
• When data is selected



• When data is not selected





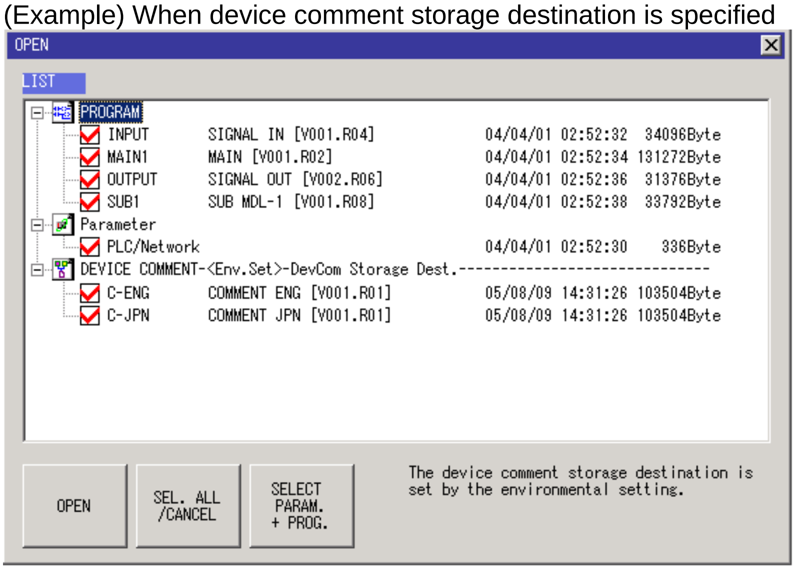
(7) PLC MESSAGE

Makes the PLC messages in the list selectable.

(8) Setting the storage destination of device comment

When the device comment storage destination is set to the location other than NC temporary memory, DEVICE COMMENT displays the status of the folders that were designated in "DevCom Storage Dest.". Refer to "6.2.1 Setting the Storage Destination of Device Comment" for details.

(Example) When device comment storage destination is specified



13.2 Saving PLC Data to the Temporary Memory

700		70
Standard	Simple	
○	Automatic	Automatic

PLC data edited or monitored, or for which the parameters were edited on the onboard, can be saved in the temporary memory.

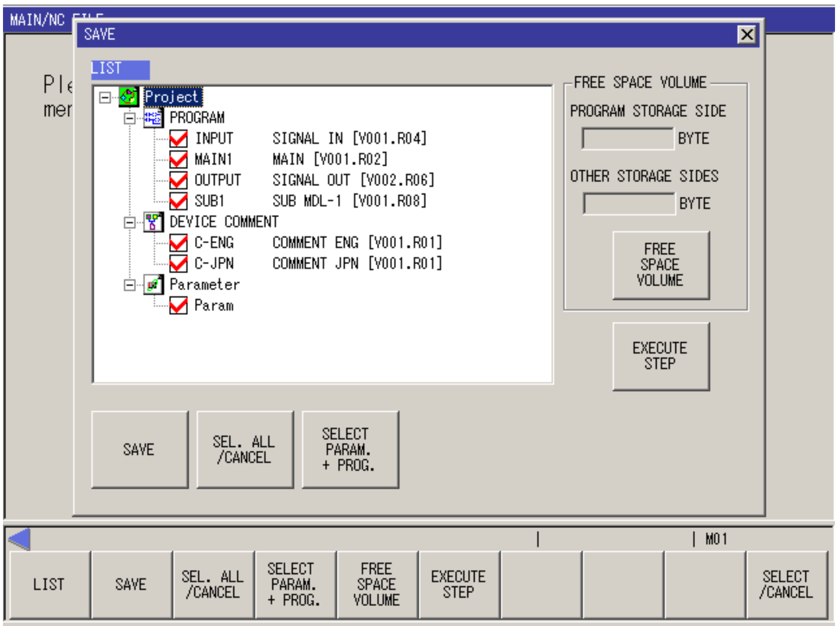
(Note 1) The PLC data in the temporary memory is lost when the NC power is turned OFF. Refer to "13.4 Writing the Temporary Memory's PLC Data to the ROM", and always save the data on the ROM. The message "ROM-Write incomplete" appears in the message area until the data is saved in the ROM.

(Note 2) If there is no PLC data in the onboard editing area, the list will be blank, and the SAVE button will be displayed in gray.

(Note 3) Save is not possible in the temporary memory during PLC RUN. (An error message appears.) When PLC is in RUN, a message appears to confirm whether to stop PLC to save the data. When PLC is not stopped, the PLC data cannot be saved in the temporary memory. (An error will appear.)
After successfully saved, a message confirming whether to have PLC returned to the RUN state is displayed.
(Note that, however, if saving to the temporary memory is executed during PLC STOP, this message will not be displayed.)

(Note 4) The program data saved in the temporary memory operates in the automatic update mode with white background of LADDER screen.

"SAVE" popup screen



Menus corresponding to popup screen

(1) Displaying the "SAVE" popup screen

Select "MAIN" → NC FILE and the SAVE menu key. When selected, the "SAVE" popup screen will appear at the center of the screen. All of the checks for the PLC data in the onboard editing area will be validated in the "List".

(2) Saving the PLC data in the temporary memory

(a) Select the PLC data to be saved from the list. Invalidate the check for data which is not to be saved.

(b) After selecting the data, select the SAVE menu key. The PLC data edited on the onboard will be saved in the temporary memory.

* When the data is saved in the temporary memory with this function, the "ROM-Write incomplete" message will appear. Refer to "13.4 Writing the Temporary Memory's PLC Data to the ROM", and always save the data on the ROM.
* If data with the same name exists, an overwrite confirmation popup screen will appear. Select the operation to be taken.

(3) Displaying the NC free space volume

Select the FREE SPACE VOLUME menu key. The free space volume in the temporary memory will appear.

(4) Closing the popup screen

Press the ◁ menu key.

(5) SEL. ALL/CANCEL

Switches valid/invalid of the checkbox ✓ for all the data in the list.

(6) SELECT PARAM. + PROG.

Makes the parameters and program data in the list selectable.

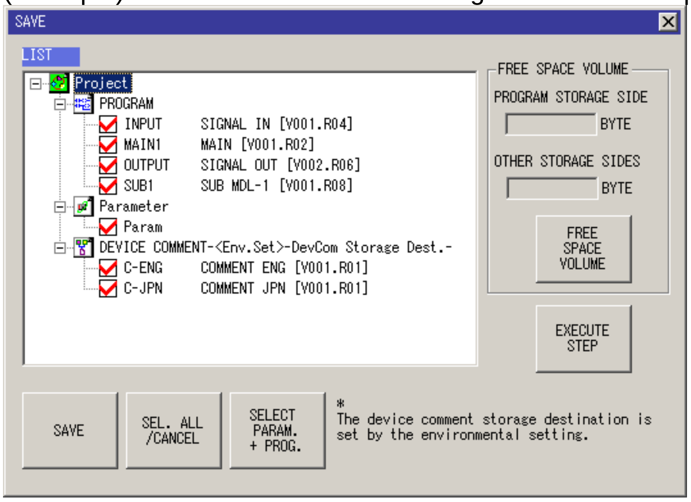
(7) SELECT/CANCEL

Switches valid/invalid of the checkbox ✓ for the selected data in the list.

(8) Device comment storage destination setting

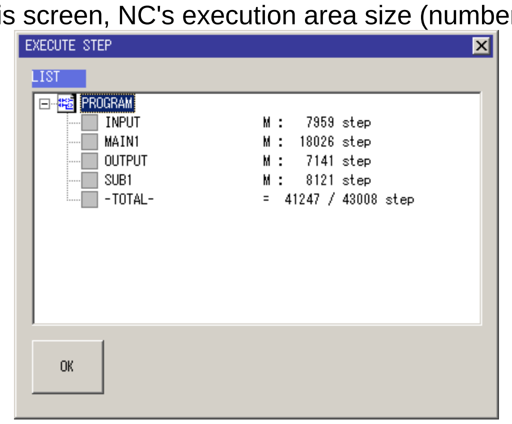
When the device comment storage destination is set to the location other than NC temporary memory, DEVICE COMMENT displays the status of the folders that were designated in "DevCom Storage Dest.". Refer to "6.2.1 Setting the Storage Destination of Device Comment" for details.

(Example) When device comment storage destination is specified



(9) EXECUTE STEP

The "EXECUTE STEP" screen is displayed.
In this screen, NC's execution area size (number of steps) is displayed.



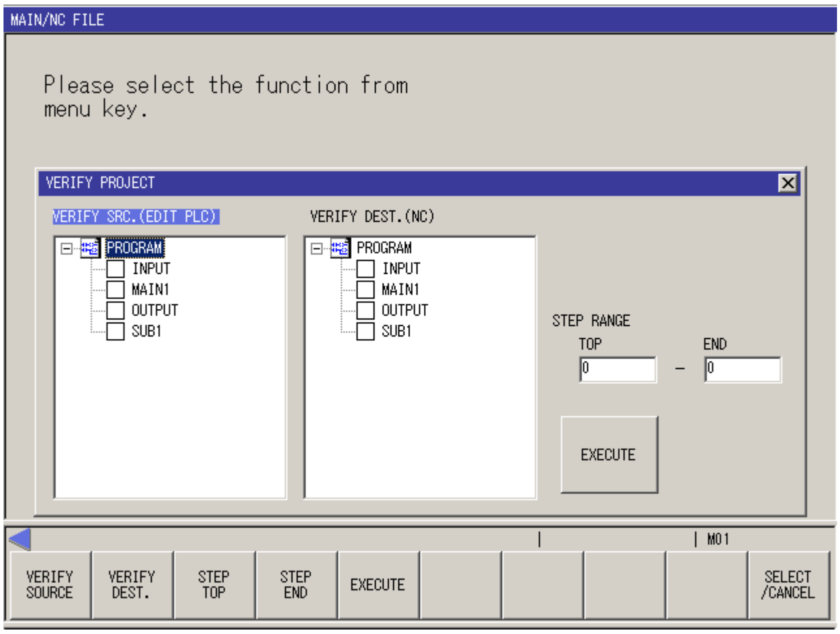
13.3 Verifying with the PLC Data in the Temporary Memory

700		70
Standard	Simple	
○		

The PLC data (programs) in the onboard editing area can be verified with the PLC data (program) in the temporary memory.

(Note 1) Only programs can be verified. (Device comments and parameters cannot be verified.)
(Note 2) Multiple programs cannot be verified simultaneously. (Always verify the programs one at a time.)
(Note 3) When there is no PLC data in either the onboard editing area or temporary memory, the list becomes blank and the "EXECUTE" button is displayed in gray.
(Note 4) With M720 series (when the display unit's OS is CE), drive cannot be selected.

" VERIFY PROJECT " popup screen



Menus corresponding to popup screen

(1) Displaying the "VERIFY PROJECT" popup screen

Select "MAIN" → NC FILE and the VERIFY PROJECT menu key. When selected, the "VERIFY PROJECT" popup screen will appear at the center of the screen.

(2) Verifying a PLC program

(a) Select the PLC data (program) to be verified from the VERIFY SRC. (EDIT PLC) and VERIFY DEST. (NC) lists. Multiple data items cannot be selected. (Select only one data item.)
(b) Select the "VERIFY" menu key. The selected PLC data (programs) will be verified, and the results will be displayed in the "VERIFY RESULT PROGRAM" popup screen.

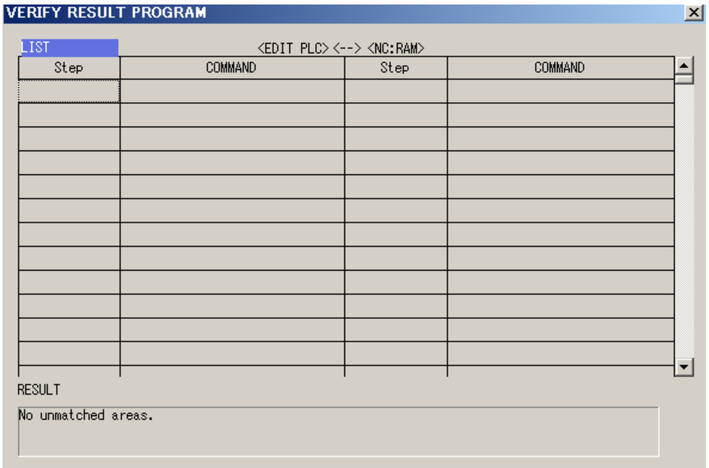
(3) Designating the verification step range

(a) Select the STEP TOP menu key, and designate the head step No. of the verification range.
(b) Select the STEP END menu key, and designate the end step No. of the verification range.
* The entire range will be verified if STEP TOP and STEP END are set to 0.

(4) Displaying the verification results

(a) When the PLC data is verified, the "VERIFY RESULT PROGRAM" popup screen will appear.
(b) Confirm the verification results, and revise the PLC data (program).
* This cannot jump to the verification results program in the same manner as GX Developer.

"VERIFY RESULT PROGRAM" popup screen



Menus corresponding to popup screen

(5) Closing the popup screen

Press the menu key.

(6) SELECT/CANCEL

Switches valid/invalid of the checkbox for the selected data in the list.

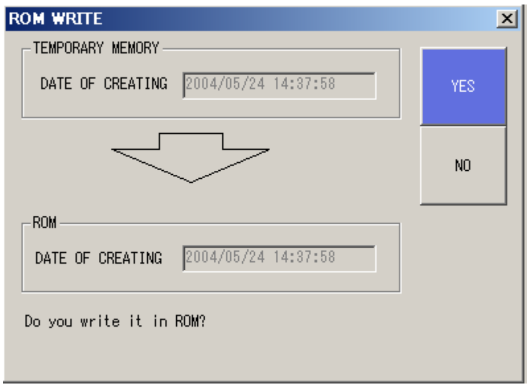
13.4 Writing PLC Data in Temporary Memory to ROM

700		70
Standard	Simple	
○	○	○

The PLC data in the temporary memory can be written to the ROM. The data in the temporary memory, including the date data, is copied to the ROM.

(Note 1) The data in the temporary memory is lost when the NC power is turned OFF, so always save the data in the ROM.
(Note 2) The PLC data in the onboard editing area cannot be written directory to the ROM.
(Note 3) ROM writing is not possible during PLC RUN. (An error message appears.)
When PLC is in RUN, a message appears to confirm whether to stop PLC to execute the ROM write operation.
ROM write is not possible if PLC is not stopped. (An error message appears.)
After ROM write operation is successfully completed, a message confirming whether to have PLC returned to the RUN state is displayed.
(Note that, however, when ROM write operation is executed during PLC STOP, a message confirming a return to the RUN state is not displayed.)
(Note 4) When the temporary memory is formatted (in the state where date of creation is not obtained), an error appears and the "YES" button is displayed in gray.
(Note 5) The date displayed on the "ROM WRITE" screen is the date that the PLC data was last saved in the temporary memory. The ROM date is also the date that the PLC data was last saved in the temporary memory.

"ROM WRITE" popup screen (The screen is an example of 700 Series)



Menus corresponding to popup screen

(1) Displaying the "ROM WRITE" popup screen

Select "MAIN" → NC FILE and the ROM WRITE menu key. When selected, the "ROM WRITE" popup screen will appear at the center of the screen.

(2) ROM-Write operations

Select the YES button. Select the NO menu key to cancel the operation.
When PLC is in RUN, a message appears to confirm whether to stop PLC to execute the ROM write operation.
When PLC is not stopped, ROM writing is not possible. (An error will appear.)

(3) Closing the popup screen

Press the ◁ menu key.

13.5 Deleting the PLC Data from the Temporary Memory

700		70
Standard	Simple	
○		○

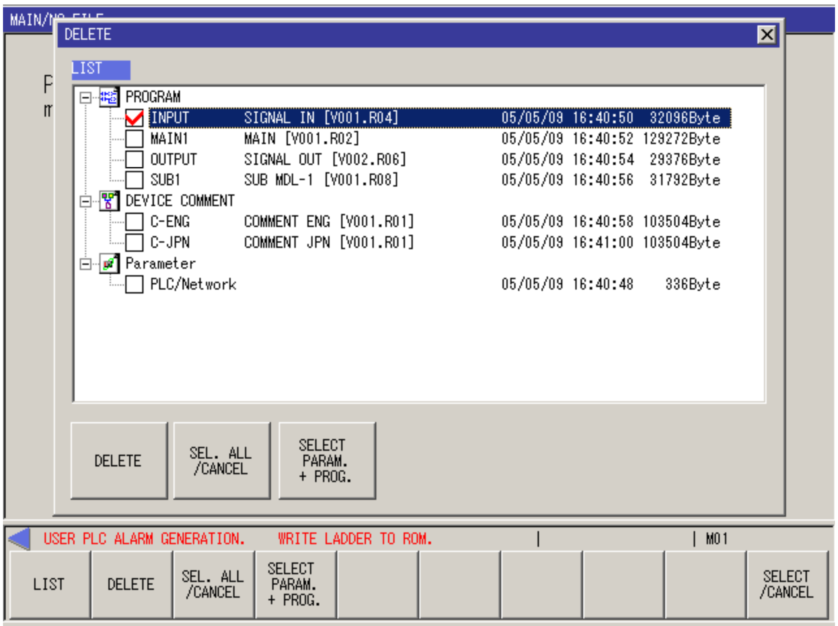
The PLC data in the temporary memory can be deleted.

(Note 1) If there is no PLC data in the temporary memory, the list will be blank, and the DELETE button will be displayed in gray.

(Note 2) PLC data in the temporary memory cannot be deleted during PLC RUN. (An error message appears.)
When PLC is in RUN, a message appears to confirm whether to stop PLC to delete the data.
When PLC is not stopped, the PLC data in the temporary memory cannot be deleted. (An error will appear.)
A message confirming whether to return the PLC to RUN state will be displayed after deletion is completed.
(Note that when PLC data in the temporary memory is deleted during PLC STOP, a message confirming whether to return to the RUN state will not be displayed.)

(Note 3) When the program in automatic update mode is deleted, the mode will be changed to local editing mode (LADDER screen background is blue) in 700 Series.

"DELETE" popup screen



Menus corresponding to popup screen

(1) Displaying the "DELETE" popup screen

Select "MAIN" → NC FILE and the DELETE menu key. When selected, the "DELETE" popup screen will appear at the center of the screen.

(2) Deleting the PLC data

(a) Select the PLC data to be deleted from the list of PLC data. Several data items can be selected.
(b) Press the "DELETE" menu key. When selected, the PLC data will be deleted.
* The deleted PLC data will also be deleted from the list.

(3) Closing the popup screen

Press the ◁ menu key.

(4) SEL. ALL/CANCEL

Switches valid/invalid of the checkbox ☑ for all the data in the list.

(5) SELECT PARAM. + PROG.

Makes the parameters and program data in the list selectable.

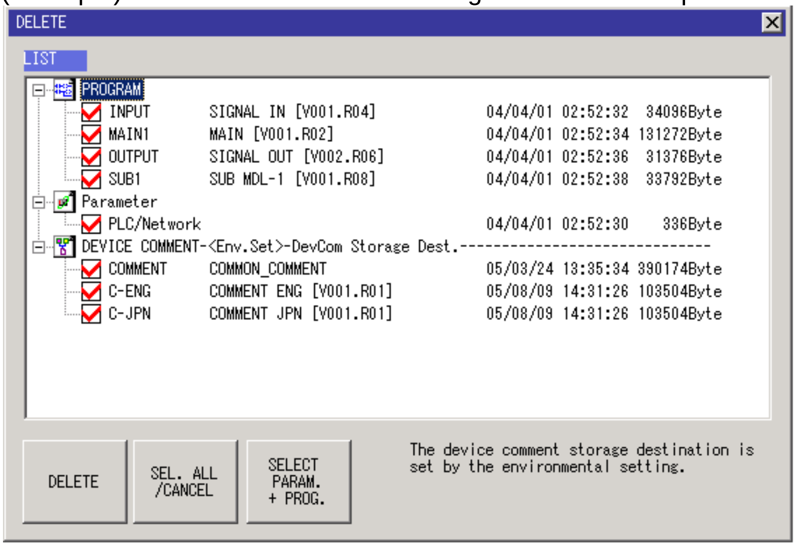
(6) SELECT/CANCEL

Switches valid/invalid of the checkbox for the selected data in the list.

(7) Device comment storage destination setting

When the device comment storage destination is set to the location other than NC temporary memory, the listed files display the status of the folders that were designated as storage destinations.
Refer to "6.2.1 Setting the Storage Destination of Device Comment" for details.

(Example) When device comment storage destination is specified



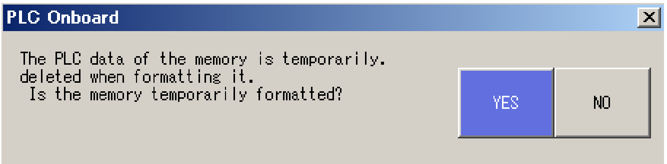
13.6 Formatting the Temporary Memory

700		70
Standard	Simple	
○		○

The temporary memory can be formatted (cleared of data).

(Note 1) The PLC data in the temporary memory is deleted when format is executed.
(Note 2) The temporary memory cannot be formatted during PLC RUN. (An error will appear.)
(Note 3) The mode in which a program is displayed changes from automatic update mode to local editing mode (LADDER screen background is light blue).

"FORMAT" popup screen (The screen is an example of 700 Series)



Menus corresponding to popup screen

(1) Displaying the "FORMAT" popup screen

Select "MAIN" → NC FILE and the FORMAT menu key. When selected, the "FORMAT" popup screen will appear at the center of the screen.

(2) Formatting operations

(a) Check the displayed message.
(b) Select the YES menu key to format the temporary memory. Select the NO menu key to cancel the operation.
* The popup window automatically closes when the YES or NO menu key is pressed.

(3) Closing the popup screen

Press the ◁ menu key.

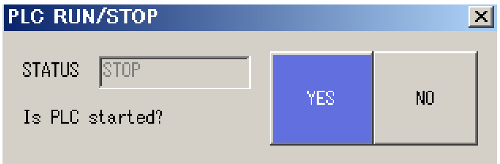
13.7 Controlling the PLC RUN/STOP

700		70
Standard	Simple	
○	○	○

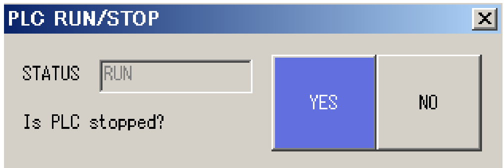
The PLC can be run or stopped.
The popup menu that appears changes according to the PLC RUN/STOP state. (The menu key correspondence is the same.)

"PLC RUN/STOP" popup screen (The screen is an example of 700 Series)

During PLC RUN



During PLC STOP



Menus corresponding to popup screen

(1) Displaying the "PLC RUN/STOP" popup screen

PLC RUN/STOP is carried out frequently, so the operations are carried out with the following menus.

[Standard operation mode]
- "MAIN" → NC FILE PLC RUN/STOP menu key
- "MAIN" → LADDER → EDIT menu PLC RUN/STOP menu key
- "MAIN" → LADDER → MONITOR menu PLC RUN/STOP menu key
- "MAIN" → DEVICE → "DEVICE BATCH MONITOR" screen PLC RUN/STOP menu key
- "MAIN" → DEVICE → "ENTRY DEVICE" screen PLC RUN/STOP menu key

[Simple operation mode]
- "MAIN" → NC FILE PLC RUN/STOP menu key
- "MAIN" → LADDER MONITOR PLC RUN/STOP menu key

[70 Series]
- "MAIN" → NC FILE PLC RUN/STOP menu key
- "MAIN" → EXTERNAL FILE PLC RUN/STOP menu key
- "MAIN" → LADDER MONITOR PLC RUN/STOP menu key
- "MAIN" → LADDER EDIT PLC RUN/STOP menu key
- "MAIN" → LADDER → "DEVICE BATCH" screen PLC RUN/STOP menu key

When selected, the "PLC RUN/STOP" popup screen will appear at the center of the screen.

(2) PLC STOP

(a) If the PLC RUN/STOP menu key is pressed during PLC RUN, the "STOP" popup screen will appear.
(b) When the YES menu key is selected, PLC STOP is executed, and the popup screen closes.
(c) When the NO menu key is selected, nothing takes place.

(3) PLC RUN

(a) If the PLC RUN/STOP menu key is pressed during PLC STOP, the "RUN" popup screen will appear.
(b) When the YES menu key is selected, PLC RUN is executed, and the popup screen closes.
(c) When the NO menu key is selected, nothing takes place.

(4) Closing the popup screen

Press the ◁ menu key.

13.8 Updating the PLC Version (Maintenance Function)

700		70
Standard	Simple	
○	○	

This function is used for maintenance. The following existing functions are carried out in a batch.
(1) The ladders are read out from the external device (IC card).
(2) The read ladders are saved in the NC's temporary memory.
(3) The data saved in the NC's temporary memory is written to the ROM.

(Note 1) Only programs, device comments and parameters that already exist in the NC can be upgraded. (A different program, device comment and parameter cannot be written to the NC.)
(Note 2) When PLC is in RUN, a message appears to confirm whether to stop PLC to upgrade the ladder. When PLC is not stopped, ladder cannot be upgraded. (An error will appear.)
A message confirming whether to return the PLC to RUN state will be displayed after upgrading of the ladder is completed.
(Note that when the ladder is upgraded during PLC STOP, a message confirming whether to return to the RUN state will not be displayed.)
(Note 3) This function cannot be used unless the password is released. (The PROGRAM UPDATE menu button will be displayed in gray.)
(Note 4) If there is no ladder file in the external device's "\lad" folder, "LIST" will be displayed blank and the "YES" and "LIST" button will appear in gray.
(Note 5) Among the PLC data to be upgraded, the data stored in project format is given priority.
(Note 6) When the device comment storage destination is specified as "storage destination specified", the device comment of the specified path will be the one to be upgraded.

13.8.1 Storing the Upgraded Data

The upgrading PLC data (program, device comment, parameter) can be stored in the following 2 ways.
• Method to store program, device comment and parameter within the same fixed file
• Method to store in the project format

(1) Method to store program, device comment and parameter within the same fixed file
The upgrading data must be stored in the external device (IC card) \lad folder (folder name is fixed).
* If there is data stored in the project format, that data shall be given priority.

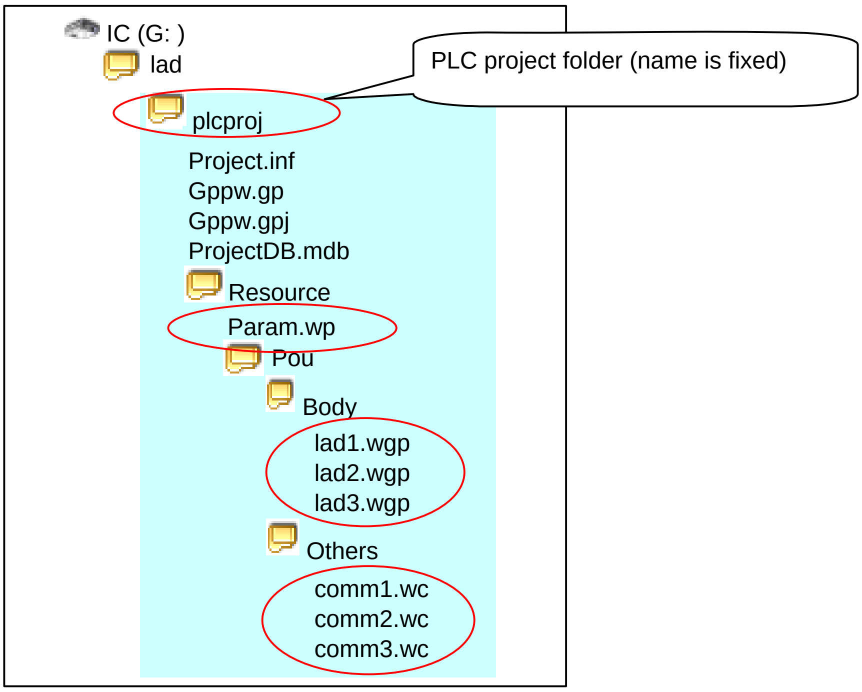
[IC CARD]

IC (G:)
lad
LAD1.wgp
LAD2.wgp
LAD3.wgp
} Program
comm1.wcd
comm2.wcd
comm3.wcd
} Device comment
param.wpa
} Parameter

(2) Method to store in the project format

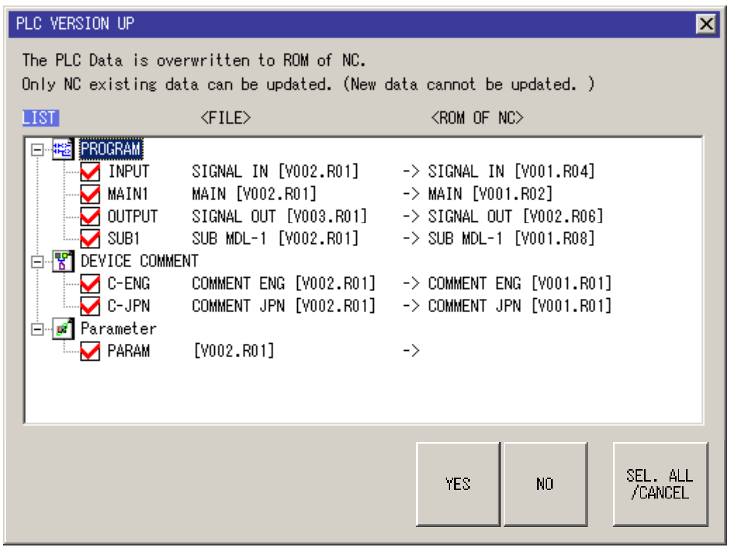
The upgrading PLC project folder (plcproj (folder name is fixed)) must be stored in the external device (IC card) \lad folder (folder name is fixed).

[IC CARD]



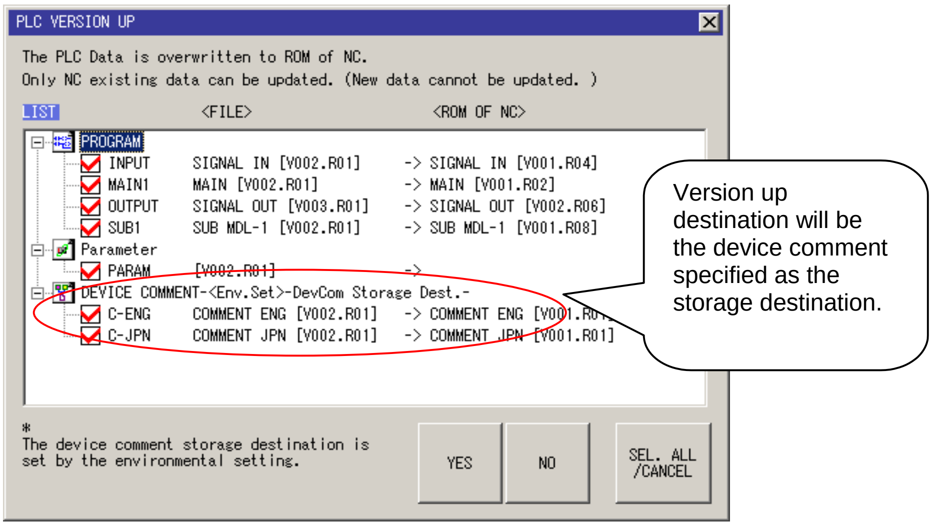
13.8.2 PLC VERSION UP Screen

(1) When the storage destination of device comment is specified as "NC temporary memory":



Menu corresponding to the popup screen

(2) When the storage destination of device comment is specified as "storage destination specified":



Menu corresponding to the popup screen

13.8.3 Operations of This Function

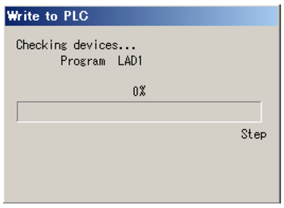
(1) Displaying the "PLC VERSION UP" popup screen

(a) Select "MAIN" → NC FILE screen → PLC VERSION UP menu key. When selected, the "PLC VERSION UP" popup screen will appear at the center of the screen.
*At this time, the program, device comment and parameter to be upgraded will be displayed in "LIST".

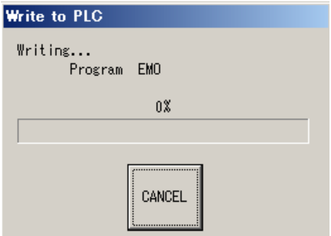
(2) Upgrading the ladder version

(a) Select the LIST menu key, and select the PLC data to be upgraded from the list. The checks are validated as the default, so invalidate the check for the ladder not to be upgraded. The check can be validated only for the program, device comment and parameter stored in the NC ROM and external device's \lad folder.

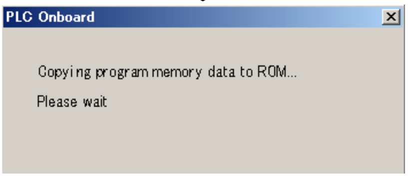
(b) Select the YES menu key. The "PLC VERSION UP" popup screen will close, and the ladder version upgrading will start.
The following popup screen appear in sequence during ladder version upgrading.



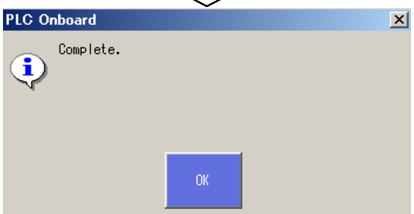
Start of program transmission to NC
The device is being checked.



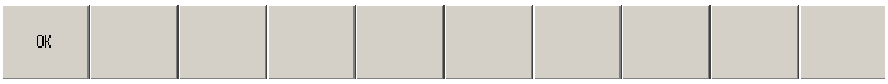
Transmission of programs to NC
The programs are being written.



The programs are written to the NC ROM.



Overwrite and save was completed normally.



Select the "OK" menu key to return to the "NC FILE" screen

(3) Closing the popup screen

Press the ◁ menu key.

(4) SEL.ALL/CANCEL

Switches valid/invalid of the checkbox ☑ for all the data in the list.

(5) SELECT/CANCEL

Switches valid/invalid of the checkbox ☑ for the selected data in the list.

13.9 Keyword

700		70
Standard	Simple	
○	○	○

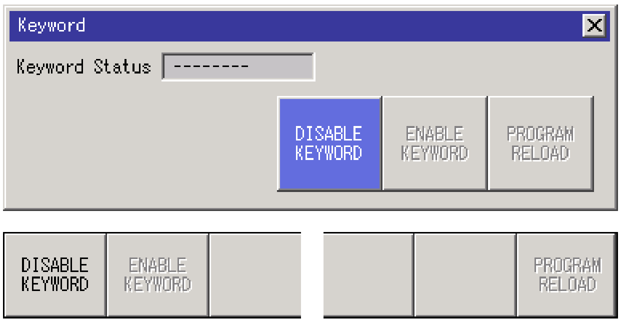
Keyword is used to protect the sequence programs stored in CNC. Data is protected from being read or overwritten by GX Developer or with the onboard PLC edit function.
A keyword protects a whole range of data. Only the files with particular names can be free from the keyword protection.
Only GX Developer can register and cancel the keyword. Onboard is used only to disable the keyword temporarily. For details of the keyword function, refer to "4.8 Keyword Registration" in the section "III. PERIPHERAL DEVELOPMENT ENVIRONMENT".
The following two ways are available to disable the keyword.

- Disabling the keyword on the KEYWORD screen
- Disabling the keyword as required at the read or write operation.

13.9.1 Disabling the Keyword on the KEYWORD Screen

The KEYWORD screen can be used to disable the keyword.

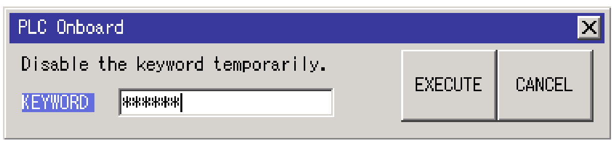
(1) Displaying the "KEYWORD" popup screen
Select "MAIN" → NC FILE and the KEYWORD menu key. The "KEYWORD" popup screen will appear.



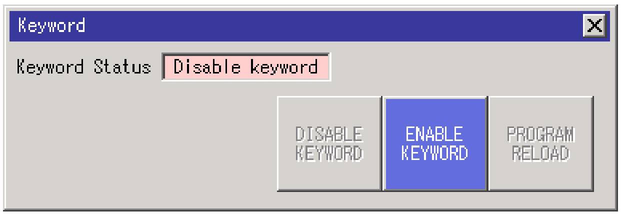
The details of the display of "Keyword Status" on this screen are as follows.

- "------" : Status is uncertain. (No keyword is disabled or registered.)
- "Disable keyword" : The keyword is being disabled.

(2) Disabling the keyword
Click the DISABLE KEYWORD menu key. The popup screen appears to input the keyword. Input the keyword and click the EXECUTE menu key.



If the keyword is successfully disabled, "Keyword Status" is changed from "------" to "Disable keyword". If not, the popup screen appears and shows the message "Keyword is not correct...".
(Note 1) Any "fictitious keyword" can also be disabled when no keyword is registered.

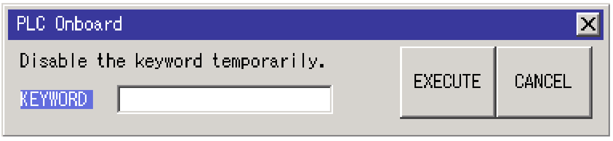


(3) Enabling the keyword
To undo the disabled keyword, click the ENABLE KEYWORD menu key. How to enable the keyword is different for each series.
700 Series:
End the onboard. The onboard can be ended from the main menu. The keyword runs again with "enabled" status after the usual onboard start-up. Turning the power OFF and ON also returns the status to "enabled".
70 Series:
Restart the onboard to return the keyword status to "enabled". Turning the power OFF and ON also returns the status to "enabled".
(Note 1) If the screen is changed to the CNC setting and display screen without the keyword enabled, the keyword stays disabled as long as the power is turned OFF and ON or the keyword is enabled again on onboard.

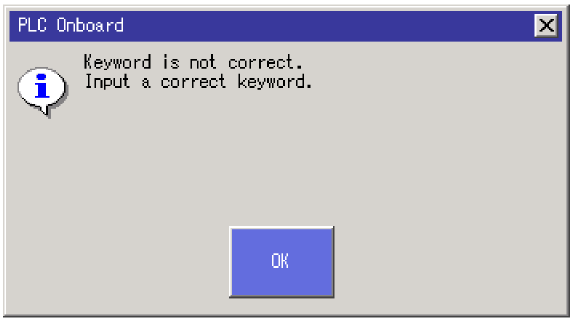
(4) Reading the programs again
When a keyword is set for the read protection, the onboard in 700 Series simple mode or in 70 Series does not read the sequence programs at its start-up. Disable the keyword and then click the PROGRAM RELOAD menu key in this case. The onboard restarts with the keyword disabled.

13.9.2 Disabling the Keyword as Required at the Read or Write Operation

If the prohibited operation is attempted to the protected data, the popup screen appears and requires disabling the keyword.



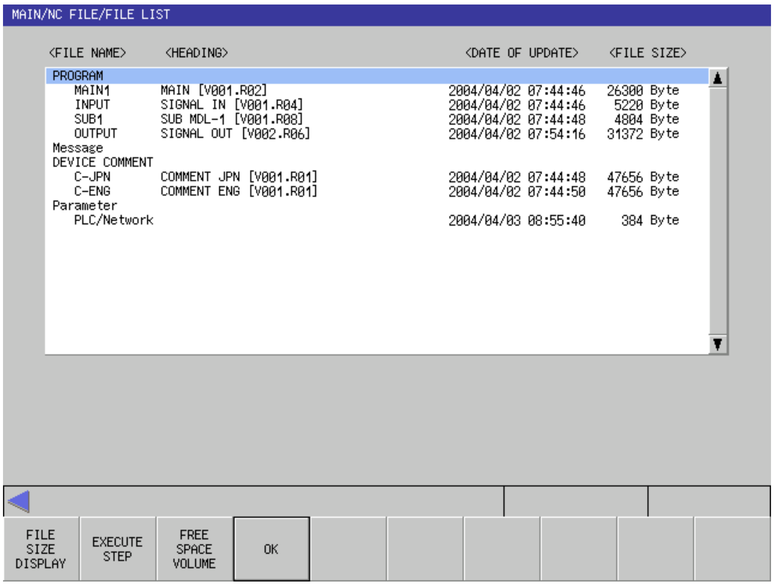
Input the keyword and click the EXECUTE menu key. The operation continues after the keyword has been successfully disabled.
The keyword, once disabled, stays disabled until any operation enables it.
The following popup screen appears when the keyword has not been disabled.



13.10 File list

700		70
Standard	Simple	
		○

The file list is displayed.



Three types of display are provided.

(1) FILE LIST

The "FILE SIZE DISPLAY" screen will appear.

(2) EXECUTE STEP

The "FILE SIZE DISPLAY" screen will appear. This menu is used to check the execution area size.

(3) FREE SPACE VOLUME

The "FILE SIZE DISPLAY" screen will appear.

14. External File Operations

700		70
Standard	Simple	
○		○

The PLC data can be opened from, saved to and verified with GX Developer projects in the external device, and projects can be deleted.

Example of GX Developer project

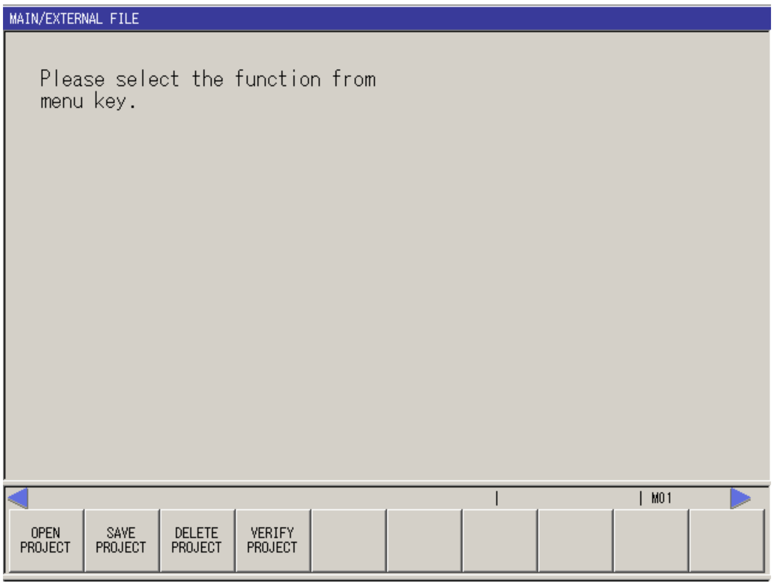
M7onbd\ ← The folder name is the project name.

Resource\Others\xxxx.wcd
Resource\POU\Body\xxxx.wpg
Resource\param.wpa
Gppw.wpj
gppw.wps
Project.inf

Data in folder (M7onbd)

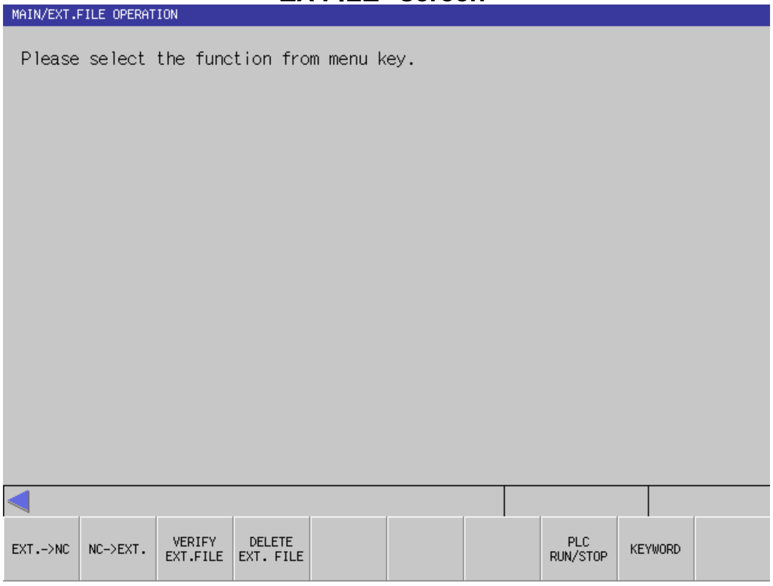
■ In 700 Series

"EX FILE" screen



■ In 70 Series

"EX FILE" screen



(1) Displaying the "EX FILE" screen

Select "MAIN" and the EXTERNAL FILE menu key. When selected, the "EX FILE" screen will appear on the full screen.

(2) Storage area of files

700 Series stores files in the onboard local editing area, in which the files are read, edited, saved and verified.
In 70 Series, files are read in the NC's temporary memory area and the same files can be simultaneously edited on onboard. Files in the temporary memory area is used as master files.

14.1 Opening PLC Data from a Project

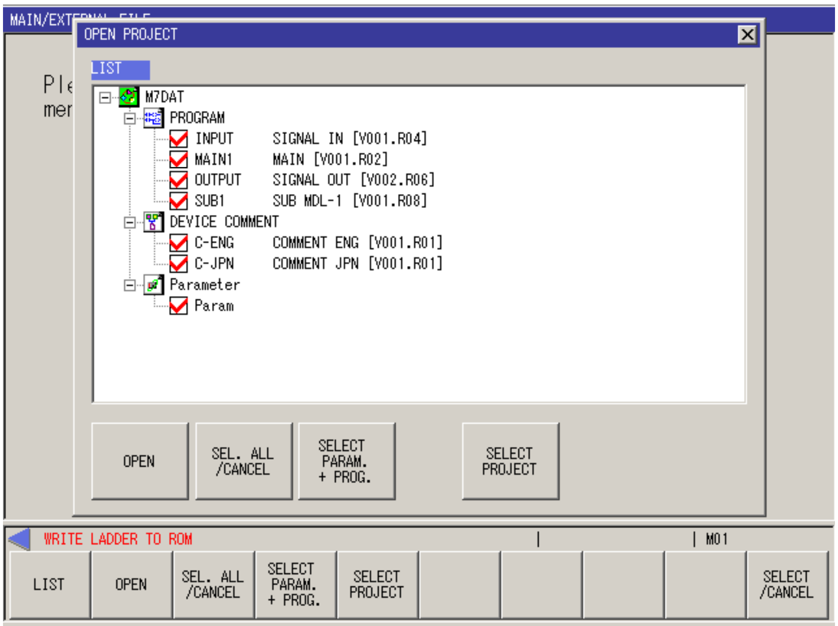
700		70
Standard	Simple	
○		○

The PLC data for a GX Developer project can be opened from the external device.

(Note 1) PLC data read in from the temporary memory and PLC data read in from an external device can both be displayed in the onboard editing area.
(Note 2) The OPEN button on the "OPEN PROJECT" popup screen is valid when the PLC data list is correctly read with "SELECT PROJECT" operations.
(Note 3) The drive cannot be selected for the M720 Series (display unit OS is CE).
(Note 4) When opening parameter data, parameter in the onboard editing area is used if no parameter data had been selected. If there is no parameter data in the onboard editing area, default parameter data is created and used.
(Note 5) SFC projected created with the GX-Developer, project of label program, and CPU with other than type Q4A cannot be opened normally. (An error occurs.)
(Note 6) The common comment created with the GX-Developer, "COMMENT", cannot be used on the onboard.
(Note 7) In 700 Series, a program opened from an external device will become the local edit mode.
(Note 8) In 70 Series, only compact flash (CF) cards are available as external device.

■ In 700 Series

"OPEN PROJECT" popup screen



Menus corresponding to popup screen

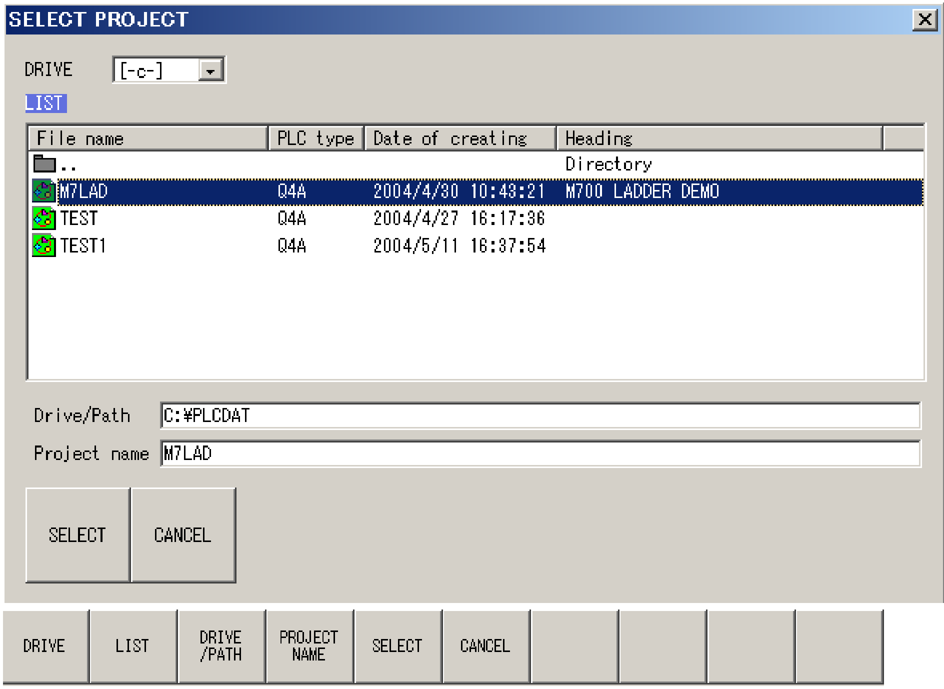
(1) Displaying the "OPEN PROJECT" popup

Select "MAIN" → EXTERNAL FILE and the OPEN PROJECT menu key. When selected, the "OPEN PROJECT" popup screen will appear at the center of the screen.

(2) Reading PLC data from an external device

(a) Press the SELECT PROJECT menu key. The "SELECT PROJECT" popup screen will open.
* The default drive is "D", and the path is "/(route)".

"SELECT PROJECT" popup screen



Menus corresponding to popup screen

(b) Select the DRIVE menu key, and designate the drive.

(c) Select the LIST menu key, and designate the path of the PLC data displayed under "LIST".

* The drive and path can be directly designated with the DRIVE/PATH menu key.

(d) Select the PROJECT NAME menu key, and designate the name of the file (project) to be read in.

* Normally when selected from the list, this is automatically set.

(e) Select the SELECT menu key. The "SELECT PROJECT" popup screen will close, and the check marks for the read PLC data will be validated on the "OPEN PROJECT" screen.

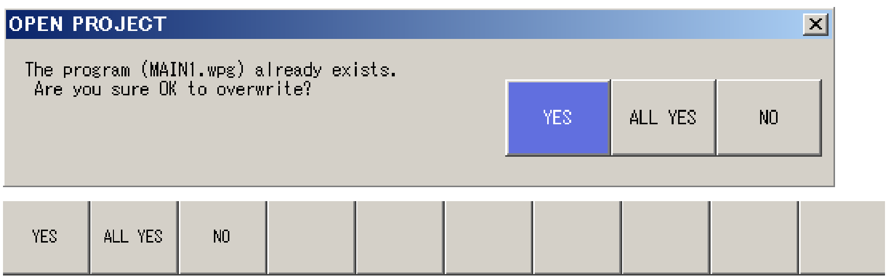
* Once set, the drive, path and project name are held until the onboard is ended.
* For some machine types, drive/path/project names are not retained.

(3) Opening the PLC data

(a) Select the PLC data to be edited from the list of PLC data read with the operations in (2). Several data items can be selected.

(b) Select the "OPEN" menu key. When selected, PLC data will be read on the onboard.

* If data with the same name exists, an overwrite confirmation popup screen will appear. Select the operation to be taken.



Menus corresponding to popup screen

(4) Closing the popup screen

Press the ◁ menu key.

(5) SEL. ALL/CANCEL

Switch valid/invalid of the checkbox for all the data in the list.

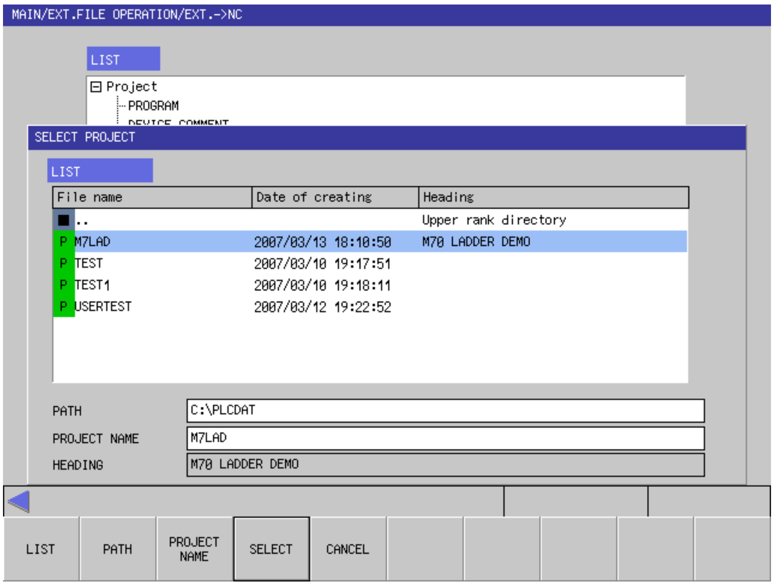
(6) SELECT PARAM. + PROG.

Make the parameters and program data in the list selectable.

(7) SELECT/CANCEL

Switch valid/invalid of the checkbox for the selected data in the list.

■ In 70 Series

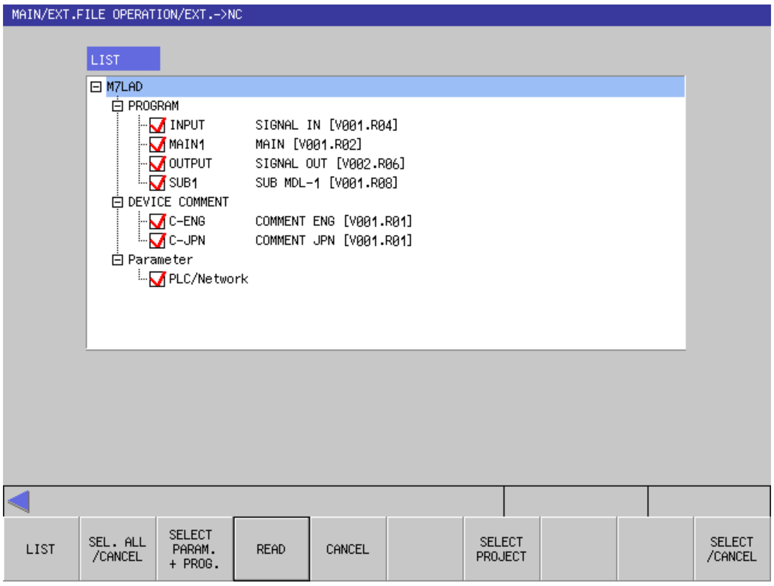


(1) Displaying the "OPEN PROJECT" popup screen

Select "MAIN" → EXTERNAL FILE EXT.->NC menu key.
Then the popup screen for selecting the project will appear at the center of the screen.

(2) Reading PLC data from an external device

Select the project to read, and then press the SELECT menu key or INPUT key.
When the key is pressed, the screen for selecting files will appear.



(3) Opening the PLC data

(a) Select the PLC data to be edited from the list of PLC data. Several data items can be selected.
(b) Press the READ menu key. The selected PLC data will be read on the onboard.

For the other operations, refer to the explanations in 700 Series.

14.2 Saving PLC Data from a Project

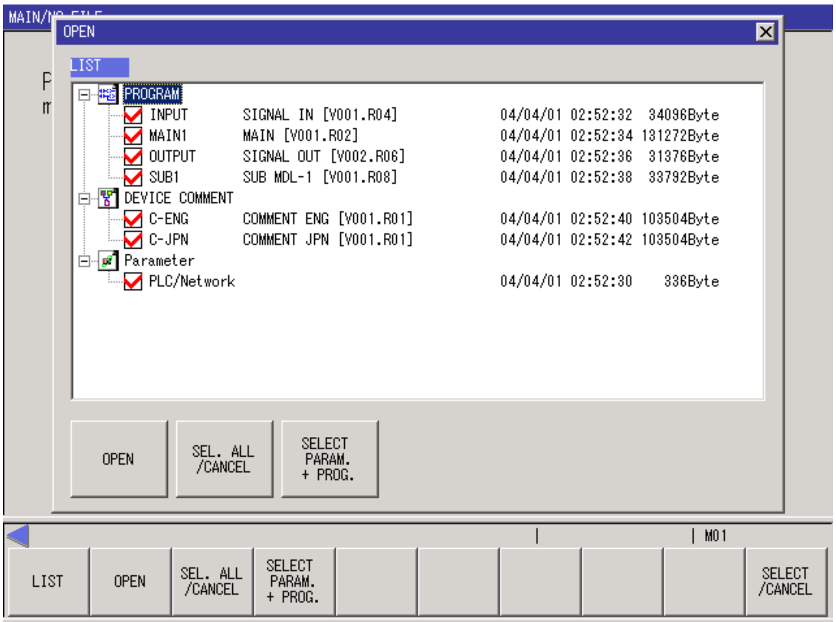
700		70
Standard	Simple	
○		○

The PLC data which has been edited and monitored, and the parameters set on the onboard, can be saved in the external device. The saved data can be read as a project with GX Developer.

(Note 1) If there is no PLC data in the onboard editing area, the list will be blank, and the "EXTERNAL SAVE" button will be displayed in gray.
(Note 2) The drive cannot be selected for the M720 Series (display unit OS is CE).
(Note 3) In 70 Series, only compact flash (CF) cards are available as external device.

■ In 700 Series

"SAVE PROJECT" popup screen



Menus corresponding to popup screen

(1) Displaying the "SAVE PROJECT" popup screen

Select "MAIN" → EXTERNAL FILE and the SAVE PROJECT menu key. When selected, the "SAVE PROJECT" popup screen will appear. The checks for the PLC data in the onboard editing area will be validated.

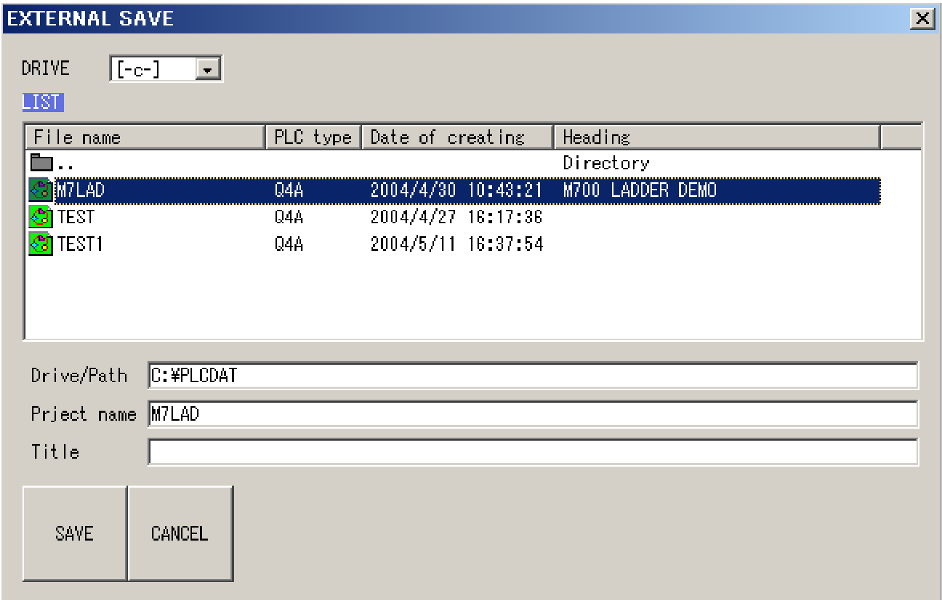
(2) Selecting the PLC data to save

Select the PLC data to save from the list of PLC data. Several data items can be selected.

(3) Saving the PLC data in the external device

(a) Select the EXTERNAL SAVE menu key. The "EXTERNAL SAVE" popup screen will appear.
* The default drive is "D", and the path is "/(route)".

"EXTERNAL SAVE" popup screen



Menus corresponding to popup screen

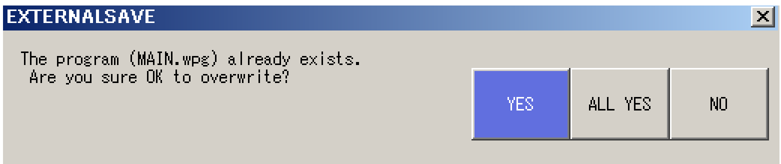
(b) Select the DRIVE menu key. Designate the save destination drive.

(c) Select the LIST menu key, and designate the path of the save destination displayed under "LIST".

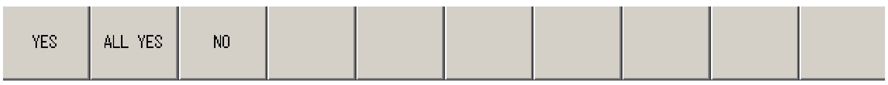
* The drive and path can be directly designated with the DRIVE/PATH menu key.

(d) Select the PROJECT NAME menu key, and designate the name of the file (project) to be saved.

(e) Select the SAVE menu key. The "EXTERNAL SAVE" popup screen will close, and the data will be saved with the designated drive, path and file name. If a file with the same name already exists, an overwrite confirmation popup screen will appear. Confirm and then save.



Menus corresponding to popup screen



* Once set, the drive, path and project name are held until the onboard is ended.

* For some machine types, drive/path/project names are not retained.

(4) Closing the popup screen

Press the ◁ menu key.

(5) SEL. ALL/CANCEL

Switch valid/invalid of the checkbox ☑ for all the data in the list.

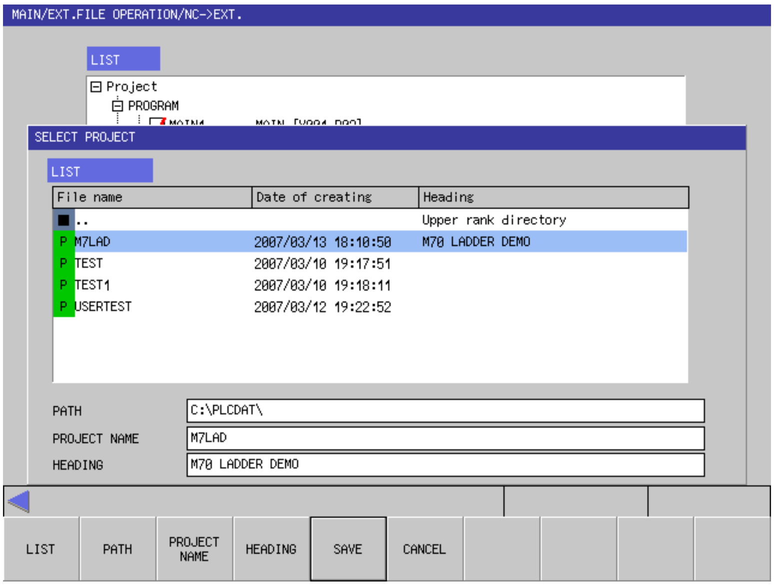
(6) SELECT PARAM. + PROG.

Make the parameters and program data in the list selectable.

(7) SELECT/CANCEL

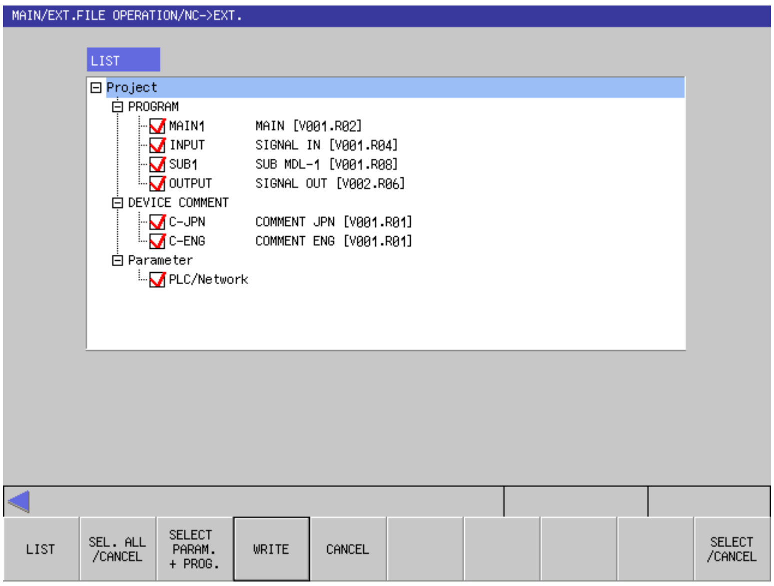
Switch valid/invalid of the checkbox for the selected data in the list.

■ In 70 Series



(1) Displaying the "SAVE PROJECT" popup screen

Select "MAIN" → EXTERNAL FILE NC ->EXT. menu key. When selected, the "SAVE PROJECT" screen will appear.



(2) Selecting the PLC data to save

Select the PLC data to be saved from the list of PLC data. Several data items can be selected.

(3) Saving the PLC data in the external device

(a) Press the WRITE menu key. Then the popup screen for selecting the project will appear.
(b) When overwriting the existing project, move the cursor to the file to overwrite. When creating a new project, enter the "PROJECT NAME" and "HEADING".
(c) Press the SAVE menu key. The project is saved.

For the other operations, refer to the explanations in 700 Series.

14.3 Deleting a Project

700		70
Standard	Simple	
○		○

A GX Developer project in the external device can be deleted.

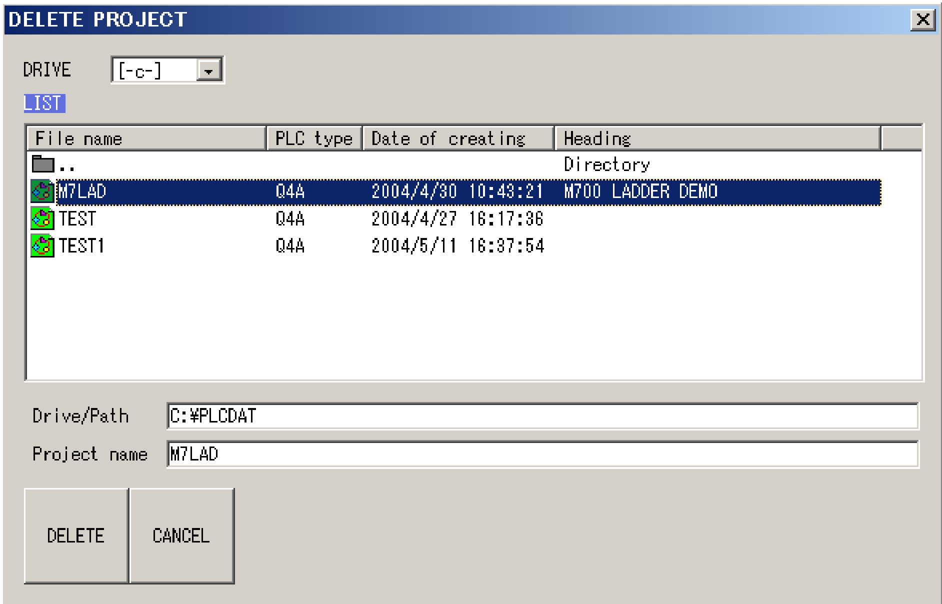
(Note 1) The drive cannot be selected for the M720 Series (display unit OS is CE).

(Note 2) The project currently opened on the onboard cannot be deleted. (An error message appears.)

(Note 3) In 70 Series, only compact flash (CF) cards are available as external device.

■ In 700 Series

"DELETE PROJECT" popup screen



Menus corresponding to popup screen

(1) Displaying the "DELETE PROJECT" popup screen

Select "MAIN" → EXTERNAL FILE and the DELETE PROJECT menu key. When selected, the "DELETE PROJECT" popup screen will appear.

* The default drive is "D", and the path is "/(route)".

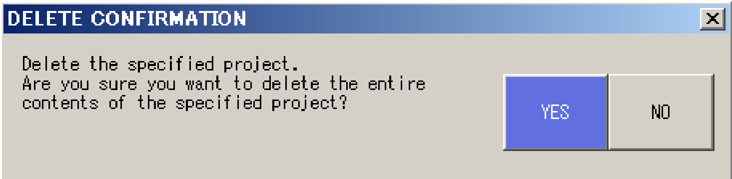
(2) Deleting a project

(a) Select the DRIVE menu key, and designate the drive.

(b) Select the LIST menu key, and designate the path of the PLC data displayed under "LIST".
* The drive and path can be directly designated with the DRIVE/PATH menu key.

(c) Select the PROJECT NAME menu key, and designate the name of the file (project) to be deleted.
* Normally when selected from the list, this is automatically set.

(d) Select the DELETE menu key. The "DELETE CONFIRMATION" popup screen will appear.

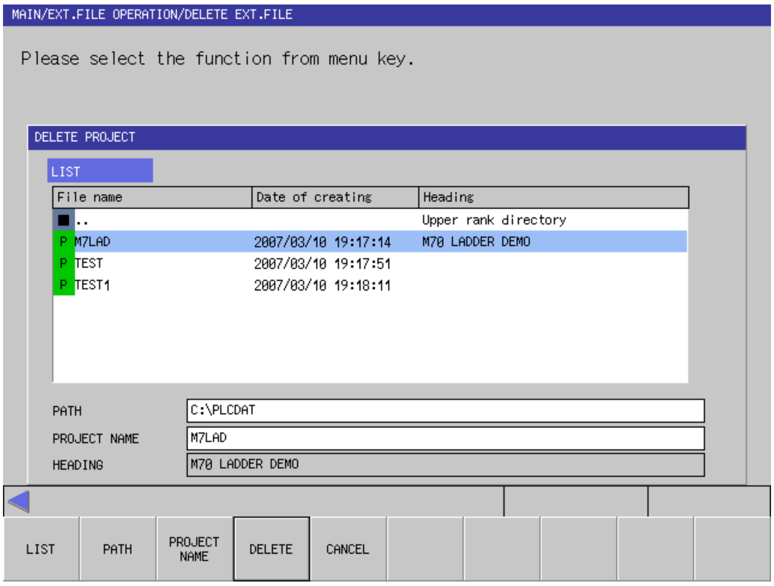


Menus corresponding to popup screen

(e) When the YES menu key is selected, deletion will be executed, and the "DELETE CONFIRMATION" popup screen and "DELETE PROJECT" popup screen will close.
When the NO menu key is selected, the deletion process will be canceled, and the "DELETE CONFIRMATION" popup screen will close.

* Drive/path/project names set once will be held even after terminating the onboard.
  When the onboard is started up again, the path specified last with external file operation is displayed.
* For some machine types, drive/path/project names are not retained.

■ In 70 Series



(1) Displaying the "DELETE PROJECT" popup screen

Select "MAIN" → EXTERNAL FILE DELETE EXT. FILE menu key. When selected, the "DELETE PROJECT" popup screen will appear.

(2) Deleting a project

(a) Specify the project to delete. Press the DELETE menu key. A popup screen will appear to confirm the deletion process.

(b) When the YES menu key is selected, deletion will be executed. When the NO menu key is selected, the deletion process will be canceled and the screen moves back to the "DELETE PROJECT" popup screen.

14.4 Verifying the Project PLC Data

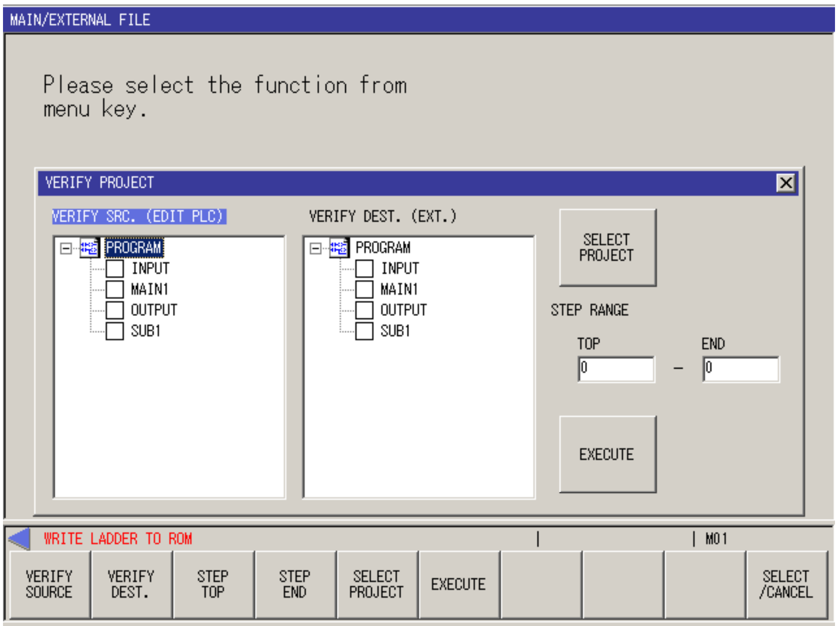
700		70
Standard	Simple	
○		○

The PLC data (programs) being edited the onboard editing area can be verified with the PLC data (program) in the external device.

(Note 1) Only programs can be verified. (Device comments and parameters cannot be verified.)
(Note 2) Multiple programs cannot be verified simultaneously. (Always verify the programs one at a time.)
(Note 3) The EXECUTE button on the "VERIFY PROJECT" popup screen is valid only after the PLC data list has been correctly read in with the "SELECT PROJECT" operations.
(Note 4) The EXECUTE button on the "VERIFY PROJECT" popup screen is displayed in gray if there is no PLC in the onboard editing area.
(Note 5) The drive cannot be selected for the M720 Series (display unit OS is CE).
(Note 6) In 70 Series, only compact flash (CF) cards are available as external device.

■ In 700 Series

"VERIFY PROJECT" popup screen



Menus corresponding to popup screen

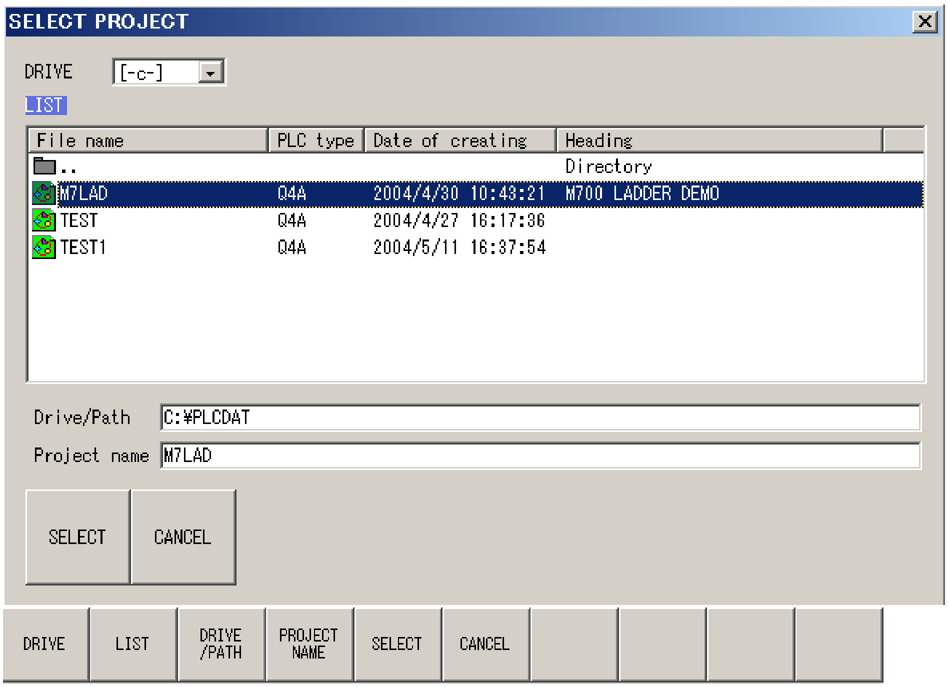
(1) Displaying the "VERIFY PROJECT" popup screen

Select "MAIN" → EXTERNAL FILE and the OPEN PROJECT menu key. When selected, the "VERIFY PROJECT" popup screen will appear.

(2) Reading the list of PLC data to be verified from the external device

(a) Select the SELECT PROJECT menu key. The "SELECT PROJECT" popup screen will appear.
* The default drive is "D", and the path is "/(route)".

"SELECT PROJECT" popup screen



Menus corresponding to popup screen

(b) Select the DRIVE menu key, and designate the drive.
(c) Select the LIST menu key, and designate the path of the PLC data displayed under "LIST".
* The drive and path can be directly designated with the DRIVE/PATH menu key.
(d) Select the PROJECT NAME menu key, and designate the name of the file (project) to be read in.
(e) Select the SELECT menu key. The "SELECT PROJECT" popup screen will close, and the read PLC data will appear under LIST on the "VERIFY PROJECT" screen.
The verification destination display will be "VERIFY DEST. (EXT.)".

* Drive/path/project names set once will be held even after terminating the onboard.
When the onboard is started up again, the path specified last with external file operation is displayed.
* For some machine types, drive/path/project names are not retained.

(3) Verifying the PLC data

(a) Select the PLC data (program) to be verified from the VERIFY SRC. (EDIT PLC) and VERIFY DEST. (NC/EXT.) lists. Multiple data items cannot be selected. (Select only one data item.)
(b) Select the "VERIFY" menu key. The selected PLC data (programs) will be verified, and the results will be displayed in the "VERIFY RESULT PROGRAM" popup screen.

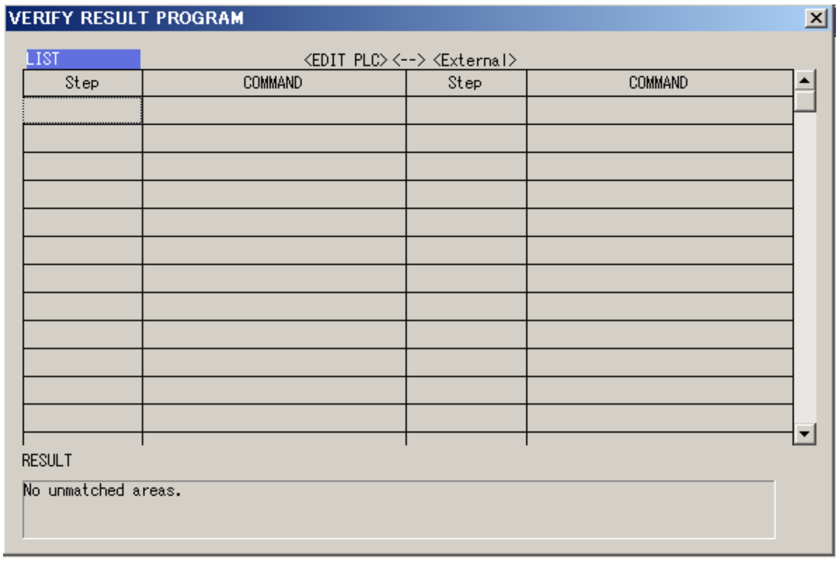
(4) Designating the verification step range

(a) Select the STEP TOP menu key, and designate the head step No. of the verification range.
(b) Select the STEP END menu key, and designate the end step No. of the verification range.
* The entire range will be verified if STEP TOP and STEP END are set to 0.

(5) Displaying the verification results

(a) When the PLC data is verified, the "VERIFY RESULT PROGRAM" popup screen will appear.
(b) Confirm the verification results, and revise the PLC data (program).
* This cannot jump to the verification results program in the same manner as GX Developer.

"VERIFY RESULT PROGRAM" popup screen



Menus corresponding to popup screen

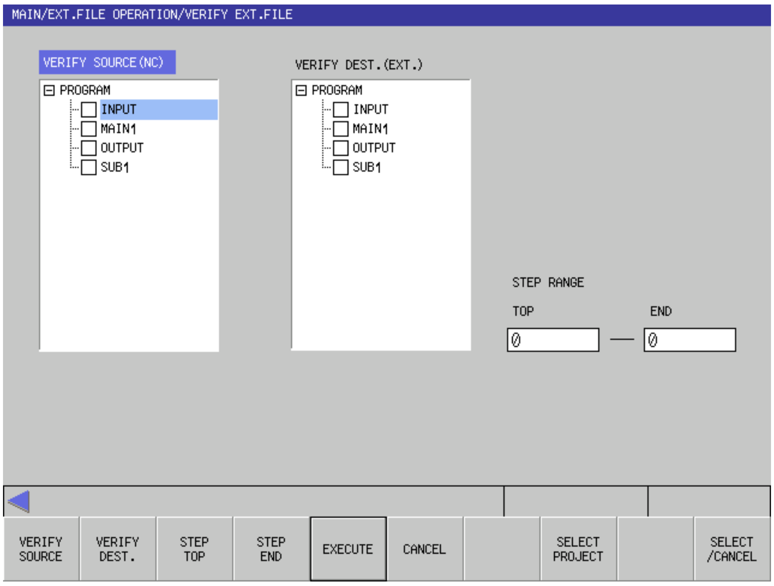
(6) Closing the popup screen

Press the menu key.

(7) SELECT/CANCEL

Switch valid/invalid of the checkbox for the selected data in the list.

■ In 70 Series



(1) Displaying the "VERIFY" popup screen

Select "MAIN" → EXTERNAL FILE VERIFY EXT. FILE menu key. Then the "VERIFY" screen will appear.

(2) Reading the list of PLC data to be verified from the external device

(a) Press the SELECT PROJECT menu key. The "PROJECT SELECT" popup screen will appear.

(b) Select the project to read and press the SELECT menu key or INPUT key. "SELECT PROJECT" popup screen will close, and the read PLC data will appear in the list on the "VERIFY" screen.

(3) Verifying the PLC data

(a) Select the PLC data (program) to be verified from VERIFY SOURCE (NC) and VERIFY DEST. (EXT.) lists. Multiple data items cannot be selected. (Select only one data item.)

(b) Select the VERIFY menu key. Then the selected PLC data (program) is verified and the result is displayed on the "VERIFY RESULT" popup screen.

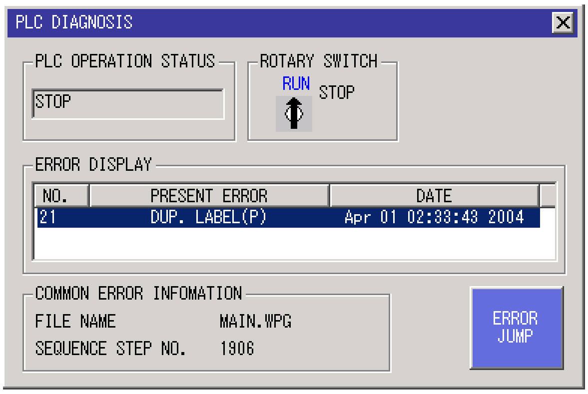
15. Diagnostics

700		70
Standard	Simple	
○	○	○

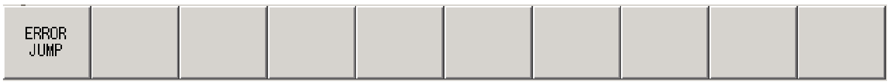
15.1 PLC Diagnostics

When an error occurs during program (ladder) execution, the detail of the error will be displayed on the popup screen. Refer to the following section for details on the error displayed in PLC diagnosis.
"III PERIPHERAL DEVELOPMENT ENVIRONMENT 8.2 PLC alarms on CNC Controller Side"

"VERIFY PROJECT" popup screen (The screen is an example of 700 Series)



Menus corresponding to popup screen



(1) Displaying "PLC DIAGNOSIS" popup screen

(a) Standard operation mode
- "MAIN" → DIAGNOS. → PLC DIAGNOSIS - or -
- "MAIN" → LADDER → PLC DIAGNOSIS

(b) Simple operation mode
- "MAIN" → PLC DIAGNOSIS

(c) 70 Series
- "MAIN" → PLC DIAGNOSIS

When menu key is pressed, "PLC DIAGNOSIS" popup screen is displayed and the error during program (ladder) execution is displayed.
When popup screen is opened, screen display is regularly (approximately every one second) updated.
When no error is occurring, "No error" is displayed.

(2) PLC operation status

PLC's RUN/STOP status is displayed.

(3) Rotary switch status

CNC unit rotary switch "CS2" status is displayed.

0(Normal)

1(PLC STOP mode)

(4) Error display

Error No., error type, error occurrence date and time (including seconds), file name, sequence step No., or parameter No. are displayed. The error with which file name, sequence step No. or parameter No. is not specified will be displayed in " -------- ".
Year is displayed in 4 digits. (Note: 2000 to 2099 only)

(5) Closing the popup screen

Press the ◁ menu key.

(6) ERROR JUMP

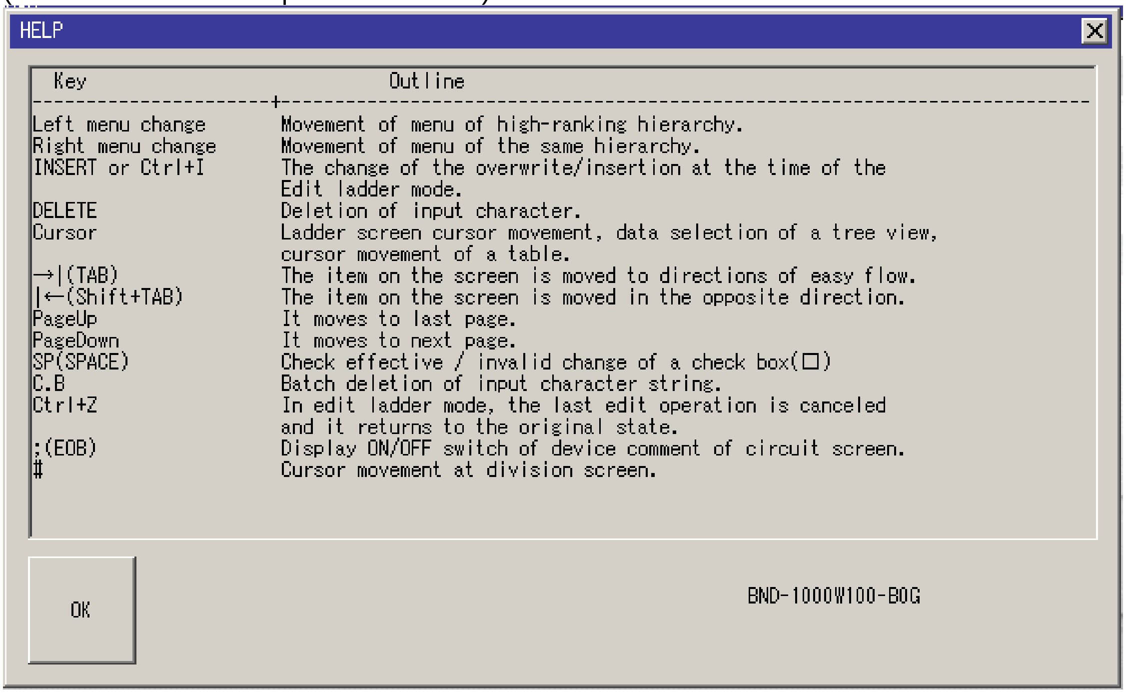
Current error jumps to the step of the ladder program shown in the common error information area.
When the conditions are where error jump function is not possible, the button and menu key are masked.
Refer to the following section for details on the error displayed in PLC diagnosis.
"III PERIPHERAL DEVELOPMENT ENVIRONMENT 8.2 PLC alarms on CNC Controller Side"

<Conditions where error jump is possible>
Error file name and step No. are displayed at error occurrence.
Target file has already been opened by the automatic update mode.

16. Help

700		70
Standard	Simple	
○	○	○

General description of the keys used in the onboard and the version information are displayed.
(The screen is an example of 700 Series)

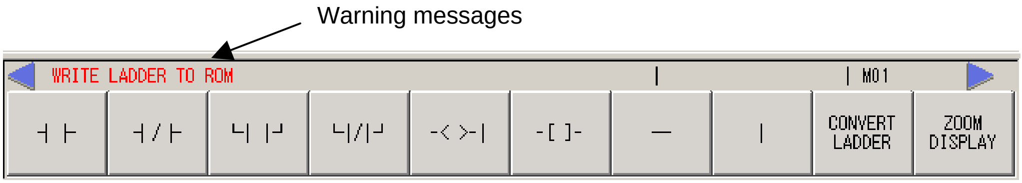


17. Error Messages

700		70
Standard	Simple	
○	○	○

17.1 Warning Messages

The warning messages are displayed in the message display area (above the menu key display).



Message	Warning details	Remedy
WRITE LADDER TO ROM	After updating the PLC data (saving, deleting or formatting) in the temporary memory, the data has not been written to the ROM. (The contents of the temporary memory are lost when the NC power is turned OFF.)	Save the contents of the temporary memory in the ROM with "ROM-Write". The message will disappear after the data is saved.

17.2 User PLC Alarm

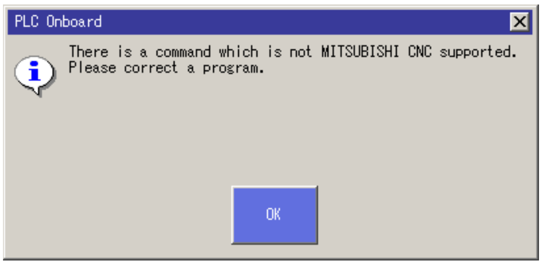
When user PLC alarm occurs, the alarm message appears in the message display area (above the area where menu key name is displayed).

Message	Error details	Remedy
USER PLC ALARM GENERATION	Various user PLC alarms	Carry out PLC DIAGNOSIS and resolve the cause of the error.

* When warning message and user PLC alarm occur at the same time, both are shown next each other in the message display area.

17.3 Error Message

Error message is displayed in the popup screen as shown below.



Message	Display timing
Command that is not supported in the MITSUBISHI CNC is included. Modify the program.	Whether the instruction is supported by MITSUBISHI CNC is checked after ladder entry. If not supported, this error is displayed.
Failed in writing into ROM.	Displays when ROM write has not been successfully performed in ROM write operation.
It is over the circuit size which can be copied.	Displays when memory required to copy circuit data is not secured.
PLC is in a RUN state. Is convert performed after stop PLC?	Displays when conversion has been executed at PLC RUN state during automatic update mode.
PLC is in a RUN state. Is preservation performed after stop PLC?	Displays when saving has been executed at PLC RUN state.
PLC is in a RUN state. Is deletion performed after stop PLC?	Displays when PLC has carried out a deletion process with NC file operation during RUN state.
PLC is in a RUN state. Is a ROM write performed after stop PLC?	Displays when PLC has carried out a ROM write process with NC file operation during RUN state.
PLC is in a RUN state. After stop PLC, is updating a program performed?	Displays when PLC has carried out a program update process with NC file operation during RUN state.
PLC is in a STOP state. Is PLC changed into a RUN state?	Displays when the operation has been successfully performed following PLC STOP after operation at PLC RUN state.

V APPENDIX

Appendix 1. Comparison of PLC Related Sections in Each Model

This section explains differences between the PLC4B development environment and M700 series PLC development environment.

Appendix 1.1 Development Tools, etc.

With the M700 Series, the user PLC development environment using the MELSEC PLC development tool is used. A comparison of each process is shown in "Table 2.1 List of development tool comparisons".
Refer to the respective Instruction Manuals for details on each tool.

Table 1.1 List of development tool comparisons

Development process		PLC4B development environment	MELSEC PLC development environment
Application from the old model	Tool	Ladder and message conversion tool (CHG4PB)	Ladder list converter (CNVM7) ∗1
	Hardware	PC9801/PC-AT	PC-AT
List -> ladder conversion	Tool	PLC development software (list section) (LIST4B)	GX Converter
	Hardware	PC9801/PC-AT	PC-AT
Ladder creation	Tool	PLC development software (ladder section) (PLC4B)	GX Developer
	Hardware	PC9801/PC-AT	PC-AT
Message creation	Tool	PLC development software (ladder section) (PLC4B)	Text editor → GX Converter → GX Developer
	Hardware	PC9801/PC-AT	PC-AT
Transfer to the CNC	Tool	PLC4B ⇔ FLD ⇔ M500 controller	GX Developer ⇔ RS232C/Ethernet ⇔ CNC controller
	Hardware	Via FLD	Via RS-232C/Ethernet
Monitor	Tool (1)		GX Developer
	Hardware		PC-AT ⇔ CNC controller
	Tool (2)	PLC onboard (ONBD)	PLC onboard
	Hardware	M500 controller	CNC controller
ROM writing	Tool	PLC onboard → F-ROM	PLC onboard → F-ROM
	Hardware	M500 controller	CNC controller
Print output	Tool (1)	PLC development software (ladder section) (PLC4B)	GX Developer
	Hardware	PC9801/PC-AT	PC-AT
	Tool (2)	PLC onboard (ONBD)	
	Hardware	M500 controller	

[Note] The arrows → and ⇔ in the list indicate the flow of information when using multiple tools.

Appendix 1.2 Devices and Device Assignments

Table 1.2.1 List of device differences (Machining center system)

Device name	M500M	M600M	M700	Remarks
X (PLC, machine input)	X0 to X4FF	X0 to XAFF	X0 to X1FFF	(Integrated to X and assignment changed)
U ($2 input)	U0 to U17F			
I ($3 and following input)	I0 to I4BF			
Y (PLC, machine output)	Y0 to Y57F	Y0 to YDFF	Y0 to Y1FFF	(Integrated to Y and assignment changed)
W ($2 output)	W0 to W1FF			
J ($3 and following output)	J0 to J63F			
S (spindle input/output)	S0 to S1FF	-	-	(Divided into X and Y and assignment changed)
M (temporary memory)	M0 to M5119	M0 to M5119	M0 to M10239	
G (temporary memory)	G0 to G3071	M5120 to M8191		
F (alarm message I/F)	F0 to F127	F0 to F255	F0 to F1023	
L (latch relay)	L0 to L255	←	←	
E (special relay)	E0 to E127	SM0 to SM127	SM0 to SM127	
Timer (10ms variable)	T0 to T15	T0 to T15	T0 to T703	The 10ms timer and 100ms timer are differentiated with the instructions The variable/fixed boundary is set with parameters.
(10ms fixed)	Q0 to Q39	T16 to T55		
(100ms variable)	T16 to T95	T56 to T135		
(100ms fixed)	Q40 to Q135	T136 to T231		
Integrated timer (100ms variable)	T96 to T103	T232 to R239	ST0 to ST63	Integrated timer **ST** has been newly added The variable/fixed boundary is set with parameters.
(100ms fixed)	Q136 to Q151	T240 to T255		
C (counter variable)	C0 to C23	C0 to C23	C0 to C255	The variable/fixed boundary is set with parameters.
B (counter fixed)	B0 to B103	C24 to C127		
D (data register)	D0 to D1023	←	D0 to D2047	
R (file register, NCI/F)	R0 to R8191	←	R0 to R13311	(Assignment changed)
A (accumulator)	A0, A1	-	-	Discontinued (replaced by D register)
Z (index register)	Z	Z0	Z0 to Z1	
V (index register)	V	Z1		
N (master control)	N0 to N7	←	←	
P (jump, call label)	P0 to P255	←	P0 to P2047 P4000 to P4005	Reserved label Local, common pointer
K (decimal constant)	K-32768 to K32767 K-2147483648 to K2147483647	←	←	
H (hexadecimal constant)	H0 to HFFFF H0 to HFFFFFFFF	←	←	

Table 1.2.2 List of device differences (lathe system)

Device name	M500L	M600L	M700	Remarks
X (PLC, machine input)	X0 to X4BF	X0 to X4BF	X0 to X1FFF	(Integrated to X and assignment changed)
U ($2 input)	U0 to U17F	X4C0 to X63F		
I ($3 to 8 input)	I0 to I3FF	X640 to XA3F		
S (No. 5, 6 spindle input)	S0 to S3F	XA40 to XA7F		
S (No. 3, 4 spindle input)	S80 to SBF	XA80 to XABF		
Y (PLC, machine output)	Y0 to Y53F	Y0 to Y53F	Y0 to Y1FFF	(Integrated to Y and assignment changed)
W($2 output)	W0 to W1FF	Y540 to Y73F		
J ($3 to 8 output)	J0 to J63F	Y740 to YD7F		
S (No. 5, 6 spindle output)	S40 to S7F	YD80 to YDBF		
S (No. 3, 4 spindle output)	SC0 to SFF	YDC0 to YDFF		
M (temporary memory)	M0 to M5119	M0 to M5119	M0 to M10239	
G (temporary memory)	G0 to G3071	M5120 to M8191		
F (alarm message I/F)	F0 to F127	F0 to F127	F0 to F1023	
L (latch relay)	L0 to L255	←	←	
E (special relay)	E0 to E127	SM0 to SM127	SM0 to SM127	
Timer (10ms variable)	T0 to T15	T0 to T15	T0 to T703	The 10ms timer and 100ms timer are differentiated with the instructions The variable/fixed boundary is set with parameters.
(10ms fixed)	Q0 to Q39	T16 to T55		
(100ms variable)	T16 to T95	T56 to T135		
(100ms fixed)	Q40 to Q135	T136 to T231		
Integrated timer (100ms variable)	T96 to T103	T232 to R239	ST0 to ST63	Integrated timer **ST** has been newly added The variable/fixed boundary is set with parameters.
(100ms fixed)	Q136 to Q151	T240 to T255		
C (counter variable)	C0 to C23	C0 to C23	C0 to C255	The variable/fixed boundary is set with parameters.
B (counter fixed)	B0 to B103	C24 to C127		
D (data register)	D0 to D1023	←	D0 to D2047	
R (file register, NCI/F)	R0 to R8191	←	R0 to R13311	(Assignment changed)
A (accumulator)	A0, A1	-	-	Discontinued (replaced by D register)
Z (index register)	Z	Z0	Z0 to Z1	
V (index register)	V	Z1		
N (master control)	N0 to N7	←	←	
P (jump, call label)	P0 to P255	←	P0 to P2047 P4000 to P4005	Reserved label Local, common pointer
K (decimal constant)	K-32768 to K32767 K-2147483648 to K2147483647	←	←	
H (hexadecimal constant)	H0 to HFFFF H0 to HFFFFFFFF	←	←	

Table 1.2.3 List of device differences (M60 Series)

Device name	M60 (PLC4B)	M60 (GX Developer)	M700	Remarks
X (PLC, machine input)	X0 to X4BF	X0 to X4BF	X0 to X1FFF	(Integrated to X and assignment changed)
U ($2 input)	U0 to U17F	X4C0 to X63F		
Y (PLC, machine output)	Y0 to Y53F	Y0 to Y53F	Y0 to Y1FFF	(Integrated to Y and assignment changed)
W ($2 output)	W0 to W1FF	Y540 to Y73F		
M (temporary memory)	M0 to M5119	M0 to M5119	M0 to M10239	
G (temporary memory)	G0 to G3071	M5120 to M8191		
F (alarm message I/F)	F0 to F127	F0 to F127	F0 to F1023	
L (latch relay)	L0 to L255	←	←	
E (special relay)	E0 to E127	SM0 to SM127	SM0 to SM127	
Timer (10ms variable)	T0 to T15	T0 to T15	T0 to T703	The 10ms timer and 100ms timer are differentiated with the instructions The variable/fixed boundary is set with parameters.
(10ms fixed)	Q0 to Q39	T16 to T55		
(100ms variable)	T16 to T95	R56 to T135		
(100ms fixed)	Q40 to Q135	T136 to R231		
Integrated timer (100ms variable)	T96 to T103	T232 to R239	ST0 to ST63	Integrated timer **ST** has been newly added The variable/fixed boundary is set with parameters.
(100ms fixed)	Q136 to Q151	T240 to T255		
C (counter variable)	C0 to C23	C0 to C23	C0 to C255	The variable/fixed boundary is set with parameters.
B (counter fixed)	B0 to B103	C24 to C127		
D (data register)	D0 to D1023	←	D0 to D2047	
R (file register, NCI/F)	R0 to R8191	←	R0 to R13311	(Assignment changed)
A (accumulator)	A0, A1	-	-	Discontinued (replaced by D register)
Z (index register)	Z	Z0	Z0 to Z1	
V (index register)	V	Z1		
N (master control)	N0 to N7	←	←	
P (jump, call label)	P0 to P255	←	P0 to P2047 P4000 to P4005	Reserved label Local, common pointer
K (decimal constant)	K-32768 to K32767 K-2147483648 to K2147483647	←	←	
H (hexadecimal constant)	H0 to HFFFF H0 to HFFFFFFFF	←	←	

Table 1.2.4 List of device differences (M60S Series)

Device name	M60S (PLC4B)	M60S (GX Developer)	M700	Remarks
X (PLC, machine input)	X0 to X4BF	X0 to X4BF	X0 to X1FFF	(Integrated to X and assignment changed)
U ($2 input)	U0 to U17F	X4C0 to X63F		
I ($3 to 8 input)	I0 to I3FF	X640 to XA3F		
S (No. 3 spindle input)	S0 to S1F	XA40 to XA5F		
S (No. 4 spindle input)	S40 to S5F	XA60 to XA7F		
S (No. 5 spindle input)	S80 to S9F	XA80 to XA9F		
S (No. 6 spindle input)	SC0 to SDF	XAA0 to XABF		
Y (PLC, machine output)	Y0 to Y53F	Y0 to Y53F	Y0 to Y1FFF	(Integrated to Y and assignment changed)
W ($2 output)	W0 to W1FF	Y540 to Y73F		
J ($3 to 8 output)	J0 to J63F	Y740 to YD7F		
S (No. 3 spindle output)	S20 to S3F	YD80 to YD9F		
S (No. 4 spindle output)	S60 to S7F	YDA0 to YDBF		
S (No. 5 spindle output)	SA0 to SBF	YDC0 to YDDF		
S (No. 6 spindle output)	SE0 to SFF	YDE0 to YDFF		
M (temporary memory)	M0 to M5119	M0 to M5119	M0 to M10239	
G (temporary memory)	G0 to G3071	M5120 to M8191		
F (alarm message I/F)	F0 to F127	F0 to F127	F0 to F1023	
L (latch relay)	L0 to L255	←	←	
E (special relay)	E0 to E127	SM0 to SM127	SM0 to SM127	
Timer (10ms variable)	T0 to T15	T0 to T15	T0 to T703	The 10ms timer and 100ms timer are differentiated with the instructions The variable/fixed boundary is set with parameters.
(10ms fixed)	Q0 to Q39	T16 to T55		
(100ms variable)	T16 to T95	T56 to T135		
(100ms fixed)	Q40 to Q135	T136 to T231		
Integrated timer (100ms variable)	T96 to T103	T232 to R239	ST0 to ST63	Integrated timer **ST** has been newly added The variable/fixed boundary is set with parameters.
(100ms fixed)	Q136 to Q151	T240 to T255		
C (counter variable)	C0 to C23	C0 to C23	C0 to C255	The variable/fixed boundary is set with parameters.
B (counter fixed)	B0 to B103	C24 to C127		
D (data register)	D0 to D1023	←	D0 to D2047	
R (file register, NCI/F)	R0 to R8191	←	R0 to R13311	(Assignment changed)
A (accumulator)	A0, A1	-	-	Discontinued (replaced by D register)
Z (index register)	Z	Z0	Z0 to Z1	
V (index register)	V	Z1		
N (master control)	N0 to N7	←	←	
P (jump, call label)	P0 to P255	←	P0 to P2047 P4000 to P4005	Reserved label Local, common pointer
K (decimal constant)	K-32768 to K32767 K-2147483648 to K2147483647	←	←	
H (hexadecimal constant)	H0 to HFFFF H0 to HFFFFFFFF	←	←	

Appendix 1.3 Instructions with Changed Designation Format

Comparison of PLC instruction format with the conventional machine type is shown in the "Table 1.3.1 Table of instruction correspondence". In M700 extended PLC instruction mode, a number of new instructions were added. Also, most of the conventional instructions extended the specifications of argument. Refer to "II PROGRAMMING EXPLANATION 6.2 Instruction List" for details.

Table 1.3.1 Table of instruction correspondence

	M500M/M500L/M60(*1)/M60S(*1)		M600M/M600L/M60(*2)/M60S(*2)		M700 (Compatible PLC Instruction mode)	
Function	**Inst. sign**	**Symbol**	**Inst. sign**	**Symbol**	**Inst. sign**	**Symbol/remarks**
Bit	DEFR	–[DEFR D]–	ANDP	D –┤↑├–	Discontinued	Alternative ladder is shown in the next page.
Average value	AVE	–[AVE S D n]┤	S. AVE	– [S.AVE S D n]┤	←	
Carry flag set	STC	–[STC]┤	S. STC	– [S.STC]┤	←	
Carry flag reset	CLC	–[CLC]┤	S. CLC	– [S.CLC]┤	←	
Right rotation	ROR	–[ROR n]┤	ROT	– [ROR D n]┤	←	Added D is a rotating word device (Same with RCR and DRCR.)
	DROR	–[DROR n]┤	DROR	– [DROR D n]┤	←	
Left rotation	ROL	–[ROL n]┤	ROL	– [ROL D n]┤	←	Added D is a rotating word device (Same with RCL and DRCL.)
	DROL	–[DROL n]┤	DROL	– [DROL D n]┤	←	
Search	SER	–[SER S1 S2 n]┤	SER	– [SER S1 S2 D n]┤	←	Added D is word device for storing search results
Quantity of 1	SUM	–[SUM S]┤	SUM	– [SUM S D]┤	←	Added D is word device for storing total number of bits
ATC	ATC	–[ATC Kn Rn Rm]-< Mm >┤	S. ATC	– [S.ATC Kn Rn Rm Mm]┤	←	
ROT	ROT	–[ROT Kn Rn Rm]-< Mm >┤	S. ROT	– [S.ROT Kn Rn Rm Mm]┤	←	
TSRH	TSRH	–[TSRH Rm Rn]-< Mn >┤	S. TSRH	– [S.TSRH Rm Rn Mn]┤	Discontinued	Replaced by PLC window
DDBA	DDBA	–[DDBA Rn / Dn]┤	S. DDBA	– [S.DDBA Rn / Dn]┤	Discontinued	Replaced by PLC window
DDBS	DDBS	–[DDBS Rn]┤	S. DDBS	– [S.DDBS Rn]┤	←	
C language release	CALL	–[CALL Pn]┤	S. CAL	– [S.CAL Pn]┤	S. CALL	– [S.CALL Kn]┤
BIT	LDBIT	├[BIT S1 n]–	LD<=	├ [<= S1 n]–	LD	S1.n (*3) ├┤├–
	ANDBIT	–[BIT S1 n]–	AND<=	– [= S1 n]–	AND	S1.n (*3) –┤├–
	ORBIT	└[BIT S1 n]┘	OR<=	└ [<= S1 n]┘	OR	S1.n (*3) └┤├┘
	LDBII	├[BII S1 n]–	LD<>	├ [> S1 n]–	LDI	S1.n (*3) ├┤/├–
	ANDBII	–[BII S1 n]–	AND<>	– [<> S1 n]–	ANDI	S1.n (*3) –┤/├–
	ORBII	└[BII S1 n]┘	OR<>	└ [<> S1 n]┘	ORI	S1.n (*3) └┤/├┘
Comparative instruction	LD=	├[= Dn/Kn Dn]–	←	←	LD=	├[= Dn/Kn Dn] – ├[= Dn Kn] – (*4)
Arithmetic operation instruction	+	–[+ Dn Dn/Kn Dn]┤	←	←	+	– [+ Dn Dn/Kn Dn]┤ – [+ Dn/Kn Dn Dn]┤ (*5)
Logical operation instruction	WAND	–[WAND Dn Dn/Kn Dn]┤	←	←	WAND	– [WAND Dn Dn/Kn Dn]┤ – [WAND Dn/Kn Dn Dn]┤ (*6)
Timer	OUT T	Kn/Dn –< Tn >┤	←	←	OUT T	(R register Kn/Dn/Rn correspondence) –< Tn >┤ (100ms dedicated)
Timer (10ms)					OUT H	H Kn/Dn –< Tn >┤ (10ms dedicated)
Timer setting value					S. TMOV	– [S.TMOV Tn D]┤

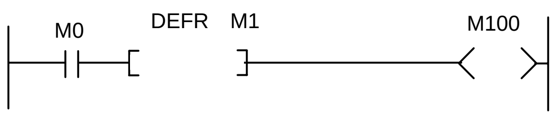
(*1) When PLC4B is used as development tool
(*2) When GX Developer is used as development tool
(*3) Follows word device bit designation format. (OUT, SET, and RST instructions also support same format)
(*4) Argument pattern for constant designation added. (AND=, OR=, LDD=, ANDD=, ORD=, LD>, and LD< system instructions also support same format)
(*5) Argument pattern for constant designation added. (D+, -, D-, *, D*, /, and D/ instructions also support same format)
(*6) Argument pattern for constant designation added. (WOR and WXOR instructions also support same format)

Appendix 1.3.1 Alternative Circuits Resulted from the Ban on DEFR Instruction

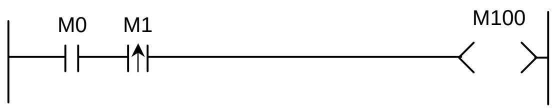
When using GX Developer as a development tool, DEFR instruction has to be substituted with an alternative instruction (ANDP), which has different performance specifications. DEFR instruction has to be substituted and banned because they are expected to become obstacles when expanding PLC functions in the future. So, Do not use DEFR instruction in M700 series.
In case DEFR instruction is currently used, substitute the ladder with another one, using the example of alternative ladders shown below.

DEFR instruction converts operation results up to DEFR instruction into pulse.
Conduction occurs when operation results prior to DEFR instruction is OFF→ON. At the same time, operation results prior to DEFR instruction will be saved in a specified device.
Examples of circuits and their movement are shown below.

(Ladder example 1) When using PLC4B as a development tool



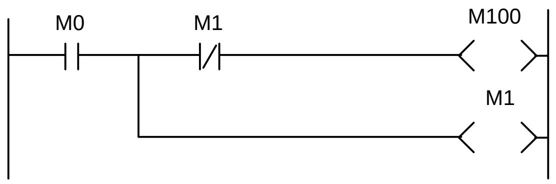
(Ladder example 2) When using GX Developer as a development tool



In the case of the ladder above, when M0 is turned OFF→ON (When M0 is ON and M1, or the operation results of one scan prior, is OFF), M100 will turn ON.
In the case of the ladder other than above, M100 will turn OFF. Also, the operation results prior to DEFR instruction (Operation result of LD M0 in the case above.) will be saved in M1.

An alternative circuit for the above circuits is shown below.

(Alternative ladder example)



Appendix 2. List of Instructions Usable with GX Developer

The following instruction lists are excerpts from the "QnACPU Programming Manual (Common Instructions)" (model name: QNACPU-P (KYOUTU) 13J522).
In these lists, the instructions "marked ✕" are unavailable for the CNC. (When written from GX Developer to the CNC, they are replaced by "NOP" instructions.)
Note that this list of instructions indicates the status in the extended instruction mode. In the compatible instruction mode, some instructions cannot be used even if there is no "x" mark.
For details on the usable instructions in each mode, refer to the list of instructions in "II PROGRAMMING EXPLANATION".

Appendix 2.1 Sequence Instructions

Contact instructions

Classification	Instruction sign	Symbol
Contact	LD	
	LDI	
	AND	
	ANI	
	OR	
	ORI	
	LDP	
	LDF	
	ANDP	
	ANDF	
	ORP	
	ORF	

Coupling instructions

Classification	Instruction sign	Symbol
Coupling	ANB	ANB
	ORB	ORB
	MPS	MPS
	MRD	MRD
	MPP	MPP
	INV	
	MEP	
	MEF	
	EGP	Vn
	EGF	Vn

Output instructions

Classification	Instruction sign	Symbol
Output	OUT	
	SET	SET D
	RST	RST D
	PLS	PLS D
	PLF	PLF D
	FF	FF D
	~~DELTA~~	DELTA D
	~~DELTAP~~	DELTAP D

Shift instructions

Classification	Instruction sign	Symbol
Shift	SFT	SFT D
	SFTP	SFTP D

Master control instructions

Classification	Instruction sign	Symbol
Master control	MC	MC n D
	MCR	MCR n

End instructions

Classifi-cation	Instruction sign	Symbol
Program end	FEND	FEND
	END	END

Other instructions

Classifi-cation	Instruction sign	Symbol
Stop	~~STOP~~	STOP
No opera-tion	NOP	
	NOPLF	NOPLF
	PAGE	PAGE n

Appendix 2.2 Comparison Operation Instructions

Comparison operation instructions

Classifi-cation	Instruction sign	Symbol
16-bit data compar-ison	LD=	= S1 S2
	AND=	= S1 S2
	OR=	= S1 S2
	LD<>	<> S1 S2
	AND<>	<> S1 S2
	OR<>	<> S1 S2
	LD>	> S1 S2
	AND>	> S1 S2
	OR>	> S1 S2
	LD<=	<= S1 S2
	AND<=	<= S1 S2
	OR<=	<= S1 S2
	LD<	< S1 S2
	AND<	< S1 S2
	OR<	< S1 S2
	LD>=	>= S1 S2
	AND>=	>= S1 S2
	OR>=	>= S1 S2

Comparison operation instructions (Continued)

Classifi-cation	Instruction sign	Symbol
32-bit data compar-ison	LDD=	D= S1 S2
	ANDD=	D= S1 S2
	ORD=	D = S1 S2
	LDD<>	D<> S1 S2
	ANDD<>	D <> S1 S2
	ORD<>	D<> S1 S2
	LDD>	D> S1 S2
	ANDD>	D> S1 S2
	ORD>	D> S1 S2
	LDD<=	D<= S1 S2
	ANDD<=	D <= S1 S2
	ORD<=	D<= S1 S2
	LDD<	D< S1 S2
	ANDD<	D < S1 S2
	ORD<	D< S1 S2
	LDD>=	D >= S1 S2
	ANDD>=	D >= S1 S2
	ORD>=	D>= S1 S2

Comparison operation instructions (Continued)

Classification	Instruction sign	Symbol
Real number data comparison	LDE=	E= S1 S2
	ANDE=	E= S1 S2
	ORE=	E= S1 S2
	LDE<>	E<> S1 S2
	ANDE<>	E<> S1 S2
	ORE<>	E<> S1 S2
	LDE>	E> S1 S2
	ANDE>	E> S1 S2
	ORE>	E> S1 S2
	LDE<=	E<= S1 S2
	ANDE<=	E<= S1 S2
	ORE<=	E<= S1 S2
	LDE<	E< S1 S2
	ANDE<	E< S1 S2
	ORE<	E< S1 S2
	LDE>=	E>= S1 S2
	ANDE>=	E>= S1 S2
	ORE>=	E>= S1 S2

Comparison operation instructions (Continued)

Classification	Instruction sign	Symbol
Character string data comparison	LD$=	$= S1 S2
	AND$=	$= S1 S2
	OR$=	$= S1 S2
	LD$<>	$<> S1 S2
	AND$<>	$<> S1 S2
	OR$<>	$<> S1 S2
	LD$>	$> S1 S2
	AND$>	$> S1 S2
	OR$>	$> S1 S2
	LD$<=	$<= S1 S2
	AND$<=	$<= S1 S2
	OR$<=	$<= S1 S2
	LD$<	$< S1 S2
	AND$<	$< S1 S2
	OR$<	$< S1 S2
	LD$>=	$>= S1 S2
	AND$>=	$>= S1 S2
	OR$>=	$>= S1 S2
Block data comparison	BKCMP=	BKCMP= S1 S2 D n
	BKCMP<>	BKCMP<> S1 S2 D n
	BKCMP>	BKCMP> S1 S2 D n
	BKCMP<=	BKCMP<= S1 S2 D n
	BKCMP<	BKCMP< S1 S2 D n
	BKCMP>=	BKCMP>= S1 S2 D n
	BKCMP=P	BKCMP=P S1 S2 D n
	BKCMP<>P	BKCMP<>P S1 S2 D n
	BKCMP>P	BKCMP>P S1 S2 D n
	BKCMP<=P	BKCMP<=P S1 S2 D n
	BKCMP<P	BKCMP<P S1 S2 D n
	BKCMP>=P	BKCMP>=P S1 S2 D n

Arithmetic operation instructions

Classification	Instruction sign	Symbol
BIN 16-bit addition/ subtraction	+	+ S D
	+P	+P S D
	+	+ S1 S2 D
	+P	+P S1 S2 D
	−	− S D
	−P	−P S D
	−	− S1 S2 D
	−P	−P S1 S2 D
BIN 32-bit addition/ subtraction	D+	D+ S D
	D+P	D+P S D
	D+	D+ S1 S2 D
	D+P	D+P S1 S2 D
	D−	D− S D
	D−P	D−P S D
	D−	D− S1 S2 D
	D−P	D−P S1 S2 D
BIN 16-bit multiplication/ division	*	* S1 S2 D
	*P	*P S1 S2 D
	/	/ S1 S2 D
	/P	/P S1 S2 D
BIN 32-bit multiplication/ division	D*	D* S1 S2 D
	D*P	D*P S1 S2 D
	D/	D/ S1 S2 D
	D/P	D/P S1 S2 D

Arithmetic operation instructions (Continued)

Classification	Instruction sign	Symbol
BCD 4-digit addition/ subtraction	~~B+~~	B+ S D
	~~B+P~~	B+P S D
	B+	B+ S1 S2 D
	B+P	B+P S1 S2 D
	~~B−~~	B− S D
	~~B−P~~	B−P S D
	B−	B− S1 S2 D
	B−P	B−P S1 S2 D
BCD 8-digit addition/ subtraction	~~DB+~~	DB+ S D
	~~DB+P~~	DB+P S D
	~~DB+~~	DB+ S1 S2 D
	~~DB+P~~	DB+P S1 S2 D
	~~DB−~~	DB− S D
	~~DB−P~~	DB−P S D
	~~DB−~~	DB− S1 S2 D
	~~DB−P~~	DB−P S1 S2 D
BCD 4-digit multiplication/ division	B*	B* S1 S2 D
	B*P	B*P S1 S2 D
	B/	B/ S1 S2 D
	B/P	B/P S1 S2 D
BCD 8-digit multiplication/ division	~~DB*~~	DB* S1 S2 D
	~~DB*P~~	DB*P S1 S2 D
	~~DB/~~	DB/ S1 S2 D
	~~DB/P~~	DB/P S1 S2 D

Arithmetic operation instructions (Continued)

Classification	Instruction sign	Symbol
Floating-point data addition/subtraction	E+	E+ S D
	E+P	E+P S D
	E+	E+ S1 S2 D
	E+P	E+P S1 S2 D
	E−	E− S D
	E−P	E−P S D
	E−	E− S1 S2 D
	E−P	E−P S1 S2 D
Floating-point data multiplication/division	E∗	E∗ S1 S2 D
	E∗P	E∗P S1 S2 D
	E/	E/ S1 S2 D
	E/P	E/P S1 S2 D
BIN block addition/subtraction	BK+	BK+ S1 S2 D n
	BK+P	BK+P S1 S2 D n
	BK−	BK− S1 S2 D n
	BK−P	BK−P S1 S2 D n
Character string data coupling	$+	$+ S D
	$+P	$+P S D
	$+	$+ S1 S2 D
	$+P	$+P S1 S2 D
BIN data increment	INC	INC D
	INCP	INCP D
	DINC	DINC D
	DINCP	DINCP D
	DEC	DEC D
	DECP	DECP D
	DDEC	DDEC D
	DDECP	DDECP D

Data conversion instructions

Classification	Instruction sign	Symbol
BCD conversion	BCD	BCD S D
	BCDP	BCDP S D
	DBCD	DBCD S D
	DBCDP	DBCDP S D

Data conversion Instructions (Continued)

Classification	Instruction sign	Symbol
BIN conversion	BIN	BIN S D
	BINP	BINP S D
	DBIN	DBIN S D
	DBINP	DBINP S D
BIN -> floating-point conversion	FLT	FLT S D
	FLTP	FLTP S D
	DFLT	DFLT S D
	DFLTP	DFLTP S D
Floating-point -> BIN conversion	INT	INT S D
	INTP	INTP S D
	DINT	DINT S D
	DINTP	DINTP S D
BIN 16-bit <-> 32-bit conversion	DBL	DBL S D
	DBLP	DBLP S D
	WORD	WORD S D
	WORDP	WORDP S D
BIN -> gray code conversion	GRY	GRY S D
	GRYP	GRYP S D
	DGRY	DGRY S D
	DGRYP	DGRYP S D
Gray code -> BIN conversion	GBIN	GBIN S D
	GBINP	GBINP S D
	DGBIN	DGBIN S D
	DGBINP	DGBINP S D
2's complement	NEG	NEG D
	NEGP	NEGP D
	DNEG	DNEG D
	DNEGP	DNEGP D
	ENEG	ENEG D
	ENEGP	ENEGP D
Block conversion	BKBCD	BKBCD S D n
	BKBCDP	BKBCDP S D n
	BKBIN	BKBIN S D n
	BKBINP	BKBINP S D n

Data transfer instructions

Classification	Instruction sign	Symbol
16-bit data transfer	MOV	MOV S D
	MOVP	MOVP S D
32-bit data transfer	DMOV	DMOV S D
	DMOVP	DMOVP S D
Floating-point data transfer	~~EMOV~~	EMOV S D
	~~EMOVP~~	EMOVP S D
Character string data transfer	~~$MOV~~	$MOV S D
	~~$MOVP~~	$MOVP S D
16-bit data NOT transfer	CML	CML S D
	CMLP	CMLP S D
32-bit data NOT transfer	DCML	DCML S D
	DCMLP	DCMLP S D
Block transfer	BMOV	BMOV S D n
	BMOVP	BMOVP S D n
Same data block transfer	FMOV	FMOV S D n
	FMOVP	FMOVP S D n
16-bit data change	XCH	XCH S D
	XCHP	XCHP S D
32-bit data change	DXCH	DXCH S D
	DXCHP	DXCHP S D
Block data change	~~BXCH~~	BXCH S D n
	~~BXCHP~~	BXCHP S D n
Upper/lower byte change	~~SWAP~~	SWAP D
	~~SWAPP~~	SWAPP D

Program branch instructions

Classification	Instruction sign	Symbol
Jump	CJ	CJ Pn
	~~SCJ~~	SCJ Pn
	JMP	JMP Pn
	GOEND	GOEND

Program execution control instructions

Classification	Instruction sign	Symbol
Interrupt disable	~~DI~~	DI
Interrupt enable	~~EI~~	EI
Interrupt disable/enable setting	~~IMASK~~	IMASK S
Return	~~IRET~~	IRET

I/O refresh instructions

Classification	Instruction sign	Symbol
I/O refresh	~~RFS~~	RFS D n

Other useful instructions

Classification	Instruction sign	Symbol
Up/down counter	~~UDCNT1~~	UDCNT1 S D n
	~~UDCNT2~~	UDCNT2 S D n
Teaching timer	~~TTMR~~	TTMR D n
Special timer	~~STMR~~	STMR S n D
Nearest access control	~~ROTC~~	ROTC S n1 n2 D
Ramp signal	~~RAMP~~	RAMP n1 n2 D1 n3 D2
Pulse density	~~SPD~~	SPD S n D
Pulse output	~~PLSY~~	PLSY n1 n2 D
Pulse width modulation	~~PWM~~	PWM n1 n2 D
Matrix input	~~MTR~~	MTR S D1 D2 n

Appendix 2.3 Application Instructions

Logical operation instructions

Classification	Instruction sign	Symbol
AND	WAND	WAND S D
	WANDP	WANDP S D
	WAND	WAND S1 S2 D
	WANDP	WANDP S1 S2 D
	DAND	DAND S D
	DANDP	DANDP S D
	DAND	DAND S1 S2 D
	DANDP	DANDP S1 S2 D
	~~BKAND~~	BKAND S1 S2 D n
	~~BKANDP~~	BKANDP S1 S2 D n
OR	WOR	WOR S D
	WORP	WORP S D
	WOR	WOR S1 S2 D
	WORP	WORP S1 S2 D
	DOR	DOR S D
	DORP	DORP S D
	DOR	DOR S1 S2 D
	DORP	DORP S1 S2 D
	~~BKOR~~	BKOR S1 S2 D n
	~~BKORP~~	BKORP S1 S2 D n
Exclusive OR	WXOR	WXOR S D
	WXORP	WXORP S D
	WXOR	WXOR S1 S2 D
	WXORP	WXORP S1 S2 D
	DXOR	DXOR S D
	DXORP	DXORP S D
	DXOR	DXOR S1 S2 D
	DXORP	DXORP S1 S2 D
	~~BKXOR~~	BKXOR S1 S2 D n
	~~BKXORP~~	BKXORP S1 S2 D n

Logical operation instructions (Continued)

Classification	Instruction sign	Symbol
NOT-exclusive OR	WXNR	WXNR S D
	WXNRP	WXNRP S D
	WXNR	WXNR S1 S2 D
	WXNRP	WXNRP S1 S2 D
	DXNR	DXNR S D
	DXNRP	DXNRP S D
	DXNR	DXNR S1 S2 D
	DXNRP	DXNRP S1 S2 D
	~~BKXNOR~~	BKXNOR S1 S2 D n
	~~BKXNORP~~	BKXNORP S1 S2 D n

Rotation instructions

Classification	Instruction sign	Symbol
Right rotation	ROR	ROR D n
	RORP	RORP D n
	RCR	RCR D n
	RCRP	RCRP D n
Left rotation	ROL	ROL D n
	ROLP	ROLP D n
	RCL	RCL D n
	RCLP	RCLP D n
Right rotation	DROR	DROR D n
	DRORP	DRORP D n
	DRCR	DRCR D n
	DRCRP	DRCRP D n
Left rotation	DROL	DROL D n
	DROLP	DROLP D n
	DRCL	DRCL D n
	DRCLP	DRCLP D n

Shift instructions

Classification	Instruction sign	Symbol
n-bit shift	SFR	SFR D n
	SFRP	SFRP D n
	SFL	SFL D n
	SFLP	SFLP D n
1-bit shift	~~BSFR~~	BSFR D n
	~~BSFRP~~	BSFRP D n
	~~BSFL~~	BSFL D n
	~~BSFLP~~	BSFLP D n
1-word shift	DSFR	DSFR D n
	DSFRP	DSFRP D n
	DSFL	DSFL D n
	DSFLP	DSFLP D n

Bit processing instructions

Classification	Instruction sign	Symbol
Bit set/reset	~~BSET~~	BSET D n
	~~BSETP~~	BSETP D n
	~~BRST~~	BRST D n
	~~BRSTP~~	BRSTP D n
Bit test	~~TEST~~	TEST S1 S2 D
	~~TESTP~~	TESTP S1 S2 D
	~~DTEST~~	DTEST S1 S2 D
	~~DTESTP~~	DTESTP S1 S2 D
Bit device Batch reset	~~BKRST~~	BKRST S n
	~~BKRSTP~~	BKRSTP S n

Data processing instructions

Classification	Instruction sign	Symbol
Data search	SER	SER S1 S2 D n
	SERP	SERP S1 S2 D n
	DSER	DSER S1 S2 D n
	DSERP	DSERP S1 S2 D n
Bit check	SUM	SUM S D
	SUMP	SUMP S D
	DSUM	DSUM S D
	DSUMP	DSUMP S D
Decode	DECO	DECO S D n
	DECOP	DECOP S D n
Encode	ENCO	ENCO S D n
	ENCOP	ENCOP S D n
7-segment decode	SEG	SEG S D
	SEGP	SEGP S D
Dissociation • Association	~~DIS~~	DIS S D n
	~~DISP~~	DISP S D n
	~~UNI~~	UNI S D n
	~~UNIP~~	UNIP S D n
	~~NDIS~~	NDIS S1 D S2
	~~NDISP~~	NDISP S1 D S2
	~~NUNI~~	NUNI S1 D S2
	~~NUNIP~~	NUNIP S1 D S2
	~~WTOB~~	WTOB S D n
	~~WTOBP~~	WTOBP S D n
	~~BTOW~~	BTOW S D n
	~~BTOWP~~	BTOWP S D n

Data processing instructions (Continued)

Classification	Instruction sign	Symbol
Retrieval	MAX	MAX S D n
	MAXP	MAXP S D n
	MIN	MIN S D n
	MINP	MINP S D n
	DMAX	DMAX S D n
	DMAXP	DMAXP S D n
	DMIN	DMIN S D n
	DMINP	DMINP S D n
Sort	SORT	SORT S1 n S2 D1 D2 S2:Number of data blocks to be compared at a time. D1:Device to be forced ON at sort completion D2:Used by system
	DSORT	DSORT S1 n S2 D1 D2 S2:Number of data blocks to be compared at a time. D1:Device to be forced ON at sort completion D2:Used by system
Total value calculation	WSUM	WSUM S D n
	WSUMP	WSUMP S D n
	DWSUM	DWSUM S D n
	DWSUMP	DWSUMP S D n

Table operation instructions

Classification	Instruction sign	Symbol
Table processing	FIFW	FIFW S D
	FIFWP	FIFWP S D
	FIFR	FIFR S D
	FIFRP	FIFRP S D
	FPOP	FPOP S D
	FPOPP	FPOPP S D
	FINS	FINS S D n
	FINSP	FINS S D n
	FDEL	FDEL S D n
	FDELP	FDELP S D n

Structuring instructions

Classification	Instruction sign	Symbol
Repeat	FOR	FOR n
	NEXT	NEXT
	BREAK	BREAK D Pn
	BREAKP	BREAKP D Pn
Sub-routine program call	CALL	CALL Pn S1~Sn
	CALLP	CALLP Pn S1~Sn
	RET	RET
	FCALL	FCALL Pn S1~Sn
	FCALLP	FCALLP Pn S1~Sn
	ECALL	SORT * Pn S1~Sn * : Program name
	ECALLP	ECALLP * Pn S1~Sn * : Program name
	EFCALL	EFCALL * Pn S1~Sn * : Program name
	EFCALLP	EFCALLP * Pn S1~Sn * : Program name
	COM	COM
Fixed index qualification	IX	IX S Device qualification circuit IXEND
	IXEND	
	IXDEV	IXDEV
	IXSET	IXSET Pn D Designation of qualification value

Appendix 3. List of Special Relays and Special Registers

Special relay SM and special register SD are the devices whose specifications are defined in the CNC. Specifications of the available devices are given below. Do not use the devices not indicated below as they are used in the system.

Appendix 3.1 Special Relay

Device	Name	Operation	Operation details	Setting side (Set time)
SM0	PLC error	OFF: No error ON: Error	• Turned ON when the PLC alarm (illegal PLC) has occurred. Reset is done by STOP→RUN.	System side (Error)
SM12	Carry flag	OFF: Carry OFF ON: Carry ON	Carry flag used during function instruction.	System side (State change)
SM16	H/W alarm (Temperature rise detection)	OFF: No error ON: Error	Used in some machine types. Refer to PLC I/F Manual, etc. for details.	System side (State change)
SM17	H/W alarm (DIO 24V illegal)			
SM18	H/W alarm (Power supply error)			
SM64	ATC display request flag	Refer to PLC I/F Manual, etc. for details.	Used in some machine types. Refer to PLC I/F Manual, etc. for details.	System side (State change)
SM65	Tool life management setting lock			
SM70	Key I/F related			
SM71	In ATC rotation			
SM80 : SM111	PLC switch	Refer to PLC I/F Manual, etc. for details.	Used in some machine types as the PLC switch. Refer to PLC I/F Manual, etc. for details.	System side (State change)
SM400	Always ON	ON OFF	• Always ON	System side (Every time END)
SM401	Always OFF	ON OFF	• Always OFF	System side (Every time END)
SM402	After RUN, turned ON by only 1 scan (Medium-speed ladder)	ON OFF 1 scan	• After RUN, turned ON by only 1 scan. • OFF during STOP. This contact can be used only in the medium-speed ladder.	System side (Every time END)
SM403	After RUN, turned OFF by only 1 scan (Medium-speed ladder)	ON 1 scan OFF	• After RUN, turned OFF by only 1 scan. • OFF during STOP. This contact can be used only in the medium-speed ladder.	System side (Every time END)
SM404	After RUN, turned ON by only 1 scan (High-speed ladder)	ON OFF 1 scan	• After RUN, turned ON by only 1 scan. • OFF during STOP. This contact can be used only in the high-speed ladder.	System side (Every time END)
SM405	After RUN, turned OFF by only 1 scan (High-speed ladder)	ON 1 scan OFF	• After RUN, turned OFF by only 1 scan. • OFF during STOP. This contact can be used only in the high-speed.	System side (Every time END)
SM410	0.1-second clock	0.05s 0.05s	• ON/OFF is repeated by the specified second. • Starts from OFF when the power is turned ON. • The clock operation is continued even in STOP. • Note that the ON-OFF status changes when the designated time has elapsed during the execution of the medium speed ladder.	System side (At high speed ladder processing) *1
SM411	0.2-second clock	0.1s 0.1s		
SM412	1-second clock	0.5s 0.5s		
SM413	2-second clock	1s 1s		
SM414	2n-second clock	n s n s	• ON/OFF is repeated according to the second specified with SD414. • Starts from OFF when the power is turned ON. • The clock operation is continued even in STOP. • If SD414 is 0 or smaller value, operates at n=30. • Note that the ON-OFF status changes when the designated time has elapsed during the execution of the medium speed ladder.	System side (At high speed ladder processing) *1

*1: Processing cycle of the high-speed ladder (sequence program) depends on the machine type.

Appendix 3.2 Special Register

Device	Name	Operation	Operation details	Setting side (Set time)
SD0	PLC error No.	PLC error No.	• Error No. is stored when PLC alarm (illegal PLC) occurs. 0 is set when normal. The error No. is cleared to 0 when executing PLC RUN next time. Refer to "Error code" in the list of PLC alarms for details on numbers.	System side (Error)
SD1	Time of PLC error occurrence	Time of occurrence (Year, month)	• Year (last 2 digits) and moth when SD0 data was updated is stored as the BCD 2-digit code. Upper 8 bits (B15-B8): Year (0-99) Lower 8 bits (B7-B0): Month (1-12)	System side (Error)
SD2		Time of occurrence (Day, hour)	• The day and hour when SD0 data was updated is stored as BCD 2-digit code. Upper 8 bits (B15-B8): Day of month (1-31) Lower 8 bits (B7-B0): Hour (0-23)	
SD3		Time of occurrence (Minute, second)	• The minute and second when SD0 data was updated is stored as BCD 2-digit code. Upper 8 bits (B15-B8): Minute (0-59) Lower 8 bits (B7-B0): Second (0-59)	
SD203	PLC operating status	PLC operating status	Operating status of PLC is stored. During RUN: 0, During STOP: 2	System side (Device read)
SD220	PLC error character string	Character string (15th/16th character)	• A message character string that is equivalent to an error No. is stored when PLC alarm (illegal PLC) occurs. (Diagnosis message displayed on the PLC diagnosis screen.) 0 is set when normal. The message character string is cleared to 0 when executing PLC RUN next time. Refer to "Diagnosis display character string" in the list of PLC alarms for details.	System side (State change)
SD221		Character string (13th/14th character)		
:		:		
SD226		Character string (3rd/4th character)		
SD227		Character string (1st/2nd character)		
SD290	Device allocation (Same as parameter contents)	Number of points allocated for X	• Stores the number of points currently set for each device (Note that number of allocated points does not always equal to the usable number of points.) SD290: Number of points allocated for X SD291: Number of points allocated for Y SD292: Number of points allocated for M SD293: Number of points allocated for L SD294: Number of points allocated for B SD295: Number of points allocated for F SD296: Number of points allocated for SB SD297: Number of points allocated for V SD298: Number of points allocated for S SD299: Number of points allocated for T SD300: Number of points allocated for ST SD301: Number of points allocated for C SD302: Number of points allocated for D SD303: Number of points allocated for W SD304: Number of points allocated for SW	System side (Initial)
SD291		Number of points allocated for Y		
SD292		Number of points allocated for M		
:		:		
SD302		Number of points allocated for D		
SD303		Number of points allocated for W		
SD304		Number of points allocated for SW		

Device	Name	Operation	Operation details	Setting side (Set time)
SD412	1-second counter	Count Nos. of 1 second unit	• After RUN, +1 is added every second of the medium speed ladder. • Count is repeated as follows. 0→32767→-32768→0 (decimal) 0→FFFF→0 (hexadecimal)	System side (At high speed ladder processing) *1
SD414	2n-second clock set	2n-second clock unit	• Stores the n of 2n-second clock (SM420). (Default value: 30) The setting range is from 1 to 32767.	User side
SD420	Scan counter (Medium-speed ladder)	Count Nos. by 1 scan	• After RUN, +1 is added every 1 scan of the medium speed ladder. • Count is repeated as follows. 0→32767→-32768→0 (decimal) 0→FFFF→0 (hexadecimal)	System side (Every time END)
SD430	Scan counter (High-speed ladder)	Count Nos. by 1 scan	• After RUN, +1 is added every 1 scan of the high speed ladder. • Count is repeated as follows. 0→32767→-32768→0 (decimal) 0→FFFF→0 (hexadecimal)	System side (Every time END)
SD520	Current scan time (Medium-speed ladder)	Current scan time (1ms unit)	• Stores the medium speed ladder current scan time in SD520 and SD521. (1µs unit is used for measurement) SD520: ms value is stored (Storage range: 0 to 14000) SD521: µs value is stored (Storage range: 0 to 999) (Example) When the current scan time is 23.6ms, it is stored as follows. SD520=23, SD521=600	System side (Every time END)
SD521	Current scan time (Medium-speed ladder)	Current scan time (1µs unit)		
SD524	Minimum scan time (Medium-speed ladder)	Minimum scan time (1ms unit)	• Stores the minimum value of the medium speed ladder scan time in SD524 and SD525. (1µs unit is used for measurement; 2nd scan and after is the target after RUN.) SD524: ms value is stored (Storage range: 0 to 14000) SD525: µs value is stored (Storage range: 0 to 999)	System side (Every time END)
SD525	Minimum scan time (Medium-speed ladder)	Minimum scan time (1µs unit)		
SD526	Maximum scan time (Medium-speed ladder)	Maximum scan time (1ms unit)	• Stores the maximum value of medium speed ladder scan time in SD526 and SD527. (1µs unit is used for measurement; 2nd scan and after is the target after RUN.) SD526: ms value is stored (Storage range: 0 to 14000) SD527: µs value is stored (Storage range: 0 to 999)	System side (Every time END)
SD527	Maximum scan time (Medium-speed ladder)	Maximum scan time (1µs unit)		

*1: Processing cycle of the high-speed ladder (sequence program) depends on the machine type.

Appendix 4. List of PLC Alarms

PLC diagnosis for each CNC PLC alarm, as well as the error details, causes and remedies are provided below.

NC alarm display (M700)				NC operating status	PLC diagnosis			
Message	Level	Sub-status 1	Sub-status 2		Error code SD0	Diagnosis display character string	File name	Step No.
U01 No user PLC	ALM (Red)	-	-	Emergency stop (S/W EMG)	-	-	-	-
U10 Illegal PLC (User PLC is illegal) "xx" in the lower 16 bits of the sub-status 1 indicates the program No.	ALM (Red)	0x04xx	Number of steps	Emergency stop (S/W EMG)	4	S/W INT. ERR	○	○
	ALM (Red)	0x20xx	Number of steps	Emergency stop (S/W EMG)	20	JUMP LABEL ERR	○	○
	ALM (Red)	0x21xx	Number of steps	Emergency stop (S/W EMG)	21	DUP. LABEL(P)	○	○
	ALM (Red)	0x22xx	-	Emergency stop (S/W EMG)	22	LOCAL LABEL OVER	○	
	ALM (Red)	0x23xx	-	Emergency stop (S/W EMG)	23	LABEL PARA. ERR		
	ALM (Red)	0x24xx	Number of steps	Emergency stop (S/W EMG)	24	RSV. LABEL ERR	○	○
	ALM (Red)	0x25xx	-	Emergency stop (S/W EMG)	25	PRG. PARA. ERR	○	

Error Contents and Cause	Remedy
Sequence program is included in F-ROM or temporary memory area. (1) Sequence program is not stored in F-ROM. (2) Sequence program is not written from the GX Developer or internal PLC edit function. (3) Sequence program cannot be read due to broken F-ROM.	(1) Write the sequence program from the GX Developer or internal PLC edit function, and then execute F-ROM writing. (2) If (1) does not solve the problem, there is a possibility of broken F-ROM.
Software instruction interruption illegal An error was found in data for the sequence program in execution. (1) Sequence program stored in F-ROM is broken. (2) Sequence program under development (before writing into F-ROM) is broken.	Contact Mitsubishi.
Label branching error (Before executing PLC) Occurs only when the bit selection parameter (#6452 bit6) "branch destination label check valid" is set to "1". (1) The CJ and CALL instructions were placed to a nonexistent label. (2) The CJ instruction was placed to the global label. (Branching is possible only with the CALL instruction.)	Check the branch destination of the CJ and CALL instructions existing in the steps occurred.
Label duplication error (Before executing PLC) (1) When using the multi-programming method: • Global labels are duplicated • Local labels are duplicated within the same file (2) When using the independent program method, labels are duplicated.	Correct the duplication of the labels existing in the steps occurred.
Local label over (Before executing PLC) The boundary value set with the PC parameter (global label boundary value) has been exceeded by the total number of local labels.	(1) Reduce the number of local labels used. • Use as sequentially as possible from P0. (2) Reset the PC parameter (global label boundary value).
Global label boundary value error (Before executing PLC) The contents of PC parameter (global label boundary value) is not normal. (1) When using the multi-programming method, a value greater than the maximum value is set. (2) When using the independent program method, the global label boundary value is set.	(1) When using the multi-programming method, correct the global label boundary value to an appropriate value. (2) When using the independent program method, delete the global label boundary value.
Reserved label error (Before executing PLC) (1) When using the multi-programming method, disabled reserved label exists. (2) When using the independent program method, reserved labels are duplicated.	(1) When using the multi-programming method, delete the reserved label. (2) When using the independent program method: • Delete the PC parameter program settings. • Correct the duplication of reserved labels
Program setting error (Before executing PLC) (1) When using the multi-programming method, PC parameter setting is not correct. • PC parameter (program setting) is not set. • Unstored program name is set. • More than the maximum number of programs that can be set (20 programs) are set (2) When using the independent program method, multiple programs are stored.	(1) When using the multi-programming method, check the PC parameter program settings. • Check the program settings and program name stored in the NC. • Set the number to 20 or less. (2) When using the independent program method: • Store only one program file.

NC alarm display (M700)				NC operating status	PLC diagnosis			
Message	Level	Sub-status 1	Sub-status 2		Error code SD0	Diagnosis display character string	File name	Step No.
U10 Illegal PLC (User PLC is illegal) "xx" in the lower 16 bits of the sub-status 1 indicates the program No.	ALM (Red)	0x26xx	-	Emergency stop (S/W EMG)	26	MISSING RET INS.	○	
	ALM (Red)	0x27xx	Number of steps	Emergency stop (S/W EMG)	27	LAD. CODE ERR	△	△
	ALM (Red)	0x28xx	-	Emergency stop (S/W EMG)	28	MISSING LAD(M)		
	ALM (Red)	0x29xx	-	Emergency stop (S/W EMG)	29	EXE. AREA OVER	○	
	ALM (Red)	0x30xx	Number of steps	Emergency stop (S/W EMG)	30	FOR INS. OVER	○	○
	ALM (Red)	0x31xx	Number of steps	Emergency stop (S/W EMG)	31	NEXT INS. ERR	△	△
	ALM (Red)	0x32xx	Number of steps	Emergency stop (S/W EMG)	32	BREAK INS. ERR	○	○
	ALM (Red)	0x400*	-	Emergency stop (S/W EMG)	40	PLC SYSTEM DOWN		
	WNG (Yellow)	0x80xx	Number of steps	PLC RUN	80	EXC.INT(BCD)	○	○

Error Contents and Cause	Remedy
RET instruction error (1) RET instruction was not executed at the branch destination of the CALL instruction. (2) RET instruction was executed without execution of CALL instruction.	Check the following matters for the entire sequence program to be executed. (1) Check if RET instruction is programmed at the end of sub-routine? (2) Check if diverged to the other operation in the middle of sub-routine and RET instruction is not executed. (3) Check if jumped to the END reservation label (P4005) in the middle of sub-routine. (4) Check if there is delimiter (FEND instruction) between adjacent program and sub-routine program.
Ladder code error (Before executing PLC) An error was found in data for the sequence program to be executed. (1) Disabled PLC instruction is used. (2) Sequence program stored in F-ROM is broken. (3) Sequence program under development (before writing into F-ROM) is broken.	Transferring, storing and F-ROM writing of the sequence program must be re-executed with the GX Developer or PLC onboard edit function.
No main processing ladders (Before executing PLC) Main processing program to be executed cannot be identified. (1) When using the multi-programming method, main processing "scan" is not set in the PC parameter (program setting). (2) When using the independent program method, no reservation ladder for the main processing ladder is available.	(1) When using the multi-programming method, check the PC parameter program settings. (2) When using the independent program method, add the reservation label (P4002) for the medium speed ladder.
Execution area over (Before executing PLC) The total number of steps for the ladder to be executed has exceeded the size of PLC processor execution area.	Check the PC parameter (program setting) and set so that the total number of steps for the ladder to be executed does not exceed the PLC processor execution area.
FOR instruction nesting over 17th level of nesting for FOR instruction was executed.	Check the number of FOR instruction's nestings in the steps generated, and keep the number to 16 or less.
NEXT instruction error (1) NEXT instruction was executed before FOR instruction. (2) After FOR instruction, END(FEND) was executed before NEXT instruction.	(1) Check the NEXT instruction existing in the number of steps generated and correct. (2) Check the ladder circuit of the program No. generated and correct. (Note that the number of steps at the error position is displayed as "0".) • Check if JMP,CALL,CJ instructions were executed between FOR and NEXT instruction, and NEXT instruction was jumped. • Check if FOR instruction and NEXT instruction are all paired.
BREAK instruction error BREAK was executed outside the range between FOR and NEXT instruction.	Check the BREAK instruction existing in the step generated and correct.
PLC system error	Contact Mitsubishi.
Software exceptional interruption (BCD instruction error) has occurred. With BCD and DBCD instructions, BIN value outside its input range was attempted to be converted into BCD.	Check the usage of BCD,DBCD instructions existing in the steps occurred.

NC alarm display (M700)				NC operating status	PLC diagnosis			
Message	Level	Sub-status			Error code SD0	Diagnosis display character string	File name	Step No.
		1	2					
U10 Illegal PLC (User PLC is illegal) "xx" in the lower 16 bits of the sub-status 1 indicates the program No.	WNG (Yellow)	0x81xx	Number of steps	PLC RUN	81	EXC.INT(BIN)	○	○
	ALM (Red)	0x82xx	Number of steps	Emergency stop (S/W EMG)	82	EXC.INT(D-BUS)	△	△
	ALM (Red)	0x83xx	Number of steps	Emergency stop (S/W EMG)	83	EXC.INT(INST.)	△	△
	ALM (Red)	0x84xx	Number of steps	Emergency stop (S/W EMG)	84	EXC.INT(I-FMT)	△	△
	ALM (Red)	0x85xx	Number of steps	Emergency stop (S/W EMG)	85	EXC.INT(I-BUS)	△	△
	ALM (Red)	0x86xx	Number of steps	Emergency stop (S/W EMG)	86	EXC.INT(CALL)	△	△
	ALM (Red)	0x87xx	Number of steps	Emergency stop (S/W EMG)	87	EXC.INT(MEM.)		
	ALM (Red)	0x88xx	Number of steps	Emergency stop (S/W EMG)	88	EXC.INT(ZERO)		
	ALM (Red)	0x89xx	Number of steps	Emergency stop (S/W EMG)	89	EXC.INT(DUP.)	△	△
	ALM (Red)	0x90xx	Number of steps	Emergency stop (S/W EMG)	90	EXC.INT(HALT)		
U50 Ladder stopped	WNG (Yellow)	-	-	Emergency stop (S/W EMG)	-		-	-

<table>
<tr><th>Error Contents and Cause</th><th>Remedy</th></tr>
<tr><td>Software exceptional interruption (BIN instruction error) has occurred.
(1) With BIN and DBIN instructions, BCD value outside its input range was attempted to be converted into BIN.
(2) With B+,B-,B*,B/ instructions, BCD value of argument 1 or 2 is outside the range between 0 and 9999.</td><td>Check the usage of BIN,DBIN,B+,B-,B*,B/ instructions existing in the steps occurred.</td></tr>
<tr><td>Software exceptional interruption (Bus error) has occurred.</td><td>Contact Mitsubishi.</td></tr>
<tr><td>Software exceptional interruption (Unmounted instruction error) has occurred.
(1) When HN102 card is not mounted and the bit selection parameter (#6452 bit6) "branch destination label check valid" is set to "0", jumped to an undefined label.
(2) Sequence program in execution is broken.</td><td>(1) Set the bit selection parameter (#6452 bit6) "branch destination label check valid" to "1" and check the branching step to the undefined label.
(2) Contact Mitsubishi.</td></tr>
<tr><td>Software exceptional interruption (Instruction format error) has occurred.</td><td>Contact Mitsubishi.</td></tr>
<tr><td>Software exceptional interruption (Instruction bus error) has occurred.
(1) When HN102 card is not mounted and the bit selection parameter (#6452 bit6) "branch destination label check valid" is set to "0", jumped to an undefined label.
(2) Sequence program in execution is broken.</td><td>(1) Set the bit selection parameter (#6452 bit6) "branch destination label check valid" to "1" and check the branching step to the undefined label.
(2) Contact Mitsubishi.</td></tr>
<tr><td>Software exceptional interruption (CALL/CALLS/RET instruction error) has occurred.</td><td rowspan="5">Contact Mitsubishi.</td></tr>
<tr><td>Software exceptional interruption (Memory area error) has occurred.</td></tr>
<tr><td>Software exceptional interruption (Zero interrupt error) has occurred.</td></tr>
<tr><td>Software exceptional interruption (Duplex exception occurrence error) has occurred.</td></tr>
<tr><td>Software exceptional interruption (HALT instruction stop) has occurred.</td></tr>
<tr><td>The ladder is stopped.</td><td>RUN the PLC.</td></tr>
</table>

Revision History

Date of revision	Manual No.	Revision details
Sept. 2004	IB(NA)1500036-A	First edition created.
Sept. 2005	IB(NA)1500036-B	Contents were revised to correspond to Mitsubishi CNC700 Series software version B1. Modifications were made to the followings in "II PROGRAMMING EXPLANATION". • 2.5 User Memory Area Configuration and Size • 4.2 Bit Selection Parameters • 4.3 Other Parameters • "DEFR" instruction was deleted from 6.1 Instruction List. • 6.4 Index Modifier • "DEFR" instruction was deleted from 7. Basic Instructions. • 10.5 Alarm Message Display Modifications were made to the followings in "III PERIPHERAL DEVELOPMENT • ENVIRONMENT". • Explanation of sampling trace function was added to 2.2 Function Support Conditions. • 4.4 File Name was split into two sections. • 4.4.1 File name rule for sequence program, parameter, and device comment • 4.4.2 File name rule for message data The following sections were added to "III PERIPHERAL DEVELOPMENT ENVIRONMENT". • 3.3 Preparation for Ethernet Communication • 4.5 Creating a Project • 5.7 Executing Sampling Trace on Device Modifications were made to the followings in "IV EXPLANATION OF BUILT-IN EDITING FUNCTION". • Descriptions on "Simple operation mode" were added overall. • 4.2 Type of Data • 5.1 Basic Operation Keys • 8.1.6 Testing the Devices • 8.3.4 Searching for Device The following sections were added to "IV EXPLANATION OF BUILT-IN EDITING FUNCTION". • 6. Environment Setting • 8.1.5 Registering Monitor • 8.1.8 Movement on Split Screen • 8.1.10 Deleting All the Entry Ladder • 8.1.11 Changing the Split Ratio • 8.1.12 Setting the Monitor Stop Conditions • 8.2.14 Undoing the Last Editing Operation • 8.3.1 Searching for Ladder (Simple search) • 8.3.2 Searching for Step No. (Simple search) • 8.4.3 Comment ON/OFF • 15. Diagnostics • 16. Help • 17.2 User PLC Alarm The following sections were deleted from "IV EXPLANATION OF BUILT-IN EDITING FUNCTION". • 14.5 Copying Project Data • 14.6 Reading Projects from the HD's Fixed Folder

<table>
<tr><th>Date of revision</th><th>Manual No.</th><th>Revision details</th></tr>
<tr><td>Jun. 2006</td><td>IB(NA)1500036-D</td><td>Contents were revised to correspond to Mitsubishi CNC700 Series software version C1.

Modifications were made to the followings in "II PROGRAMMING EXPLANATION".
• Drawing in "2.4.2 Program Execution Order" was improved.
• "2.6 Storing PLC Processing Program and Execution Mode" was added.
• "4.1 Description about PLC constant" was corrected.
• Parameters were added to "4.2 Bit Selection Parameters".
• SB, B, V, SW, SD, W were added to "5. Explanation of Devices".
• Description about GOEND instruction was added to "5.3.14.4 Reserved Pointers".

Description about extending PLC instructions was added throughout the following chapters.
• 6. Explanation of Instructions
• 7. Basic Instructions
• 8. Function Instructions

Modifications were made to the followings in "III PERIPHERAL DEVELOPMENT ENVIRONMENT".
• Description of "IC memory card A (RAM)" was added to "4.3 PLC Data Storage Areas"
• Description about "4.6.3 Setting the Number of Common Pointer Points" was corrected.
• 5.2.4 "Operations and Check Items at the Other Errors" was added.
• Sampling trace's applicable devices described in "5.7.2 Basic Specifications" were corrected.
• Addition was made to "8.1 List of Errors During GX Developer Online Operations"
• Addition was made to "8.2.5 List of Corresponding PLC Alarms"

Modifications were made to the followings in "IV EXPLANATION OF BUILT-IN EDITING FUNCTION".
• 9.1 Contact Coil Usage List
• 9.2 List of Used Devices
• 9.3 Program Check
• 10.2 Device Registration Monitor
• 11.1 Setting the Program
• 11.2 Common Pointer Setting
• 13.8 Updating the PLC Version (Maintenance Function)

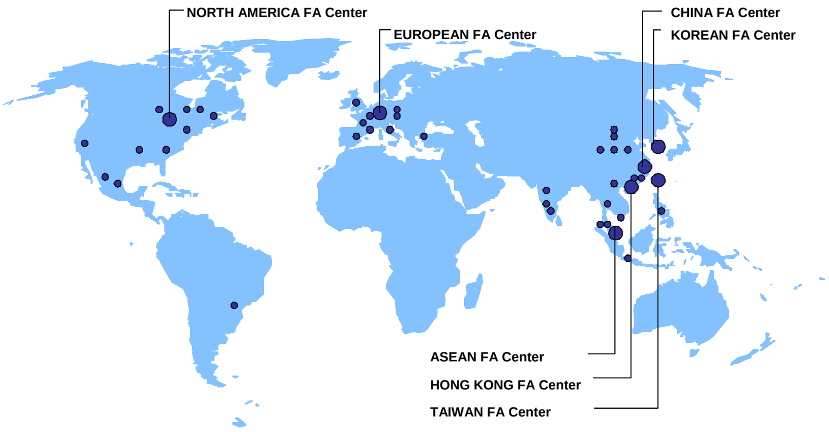
The following section was added to "IV EXPLANATION OF BUILT-IN EDITING FUNCTION".
• 10.3 Sampling Trace

Modifications were made to the followings in "V APPENDIX".
• Appendix 1.3 Instructions with Changed Designation Format
• Appendix 2. List of Instructions Usable with GX Developer

The following sections were added to "V APPENDIX".
• Appendix 3. List of Special Relays and Special Devices
• Appendix 4. List of PLC Alarms</td></tr>
<tr><td>Nov. 2006</td><td>IB(NA)1500036-E</td><td>Mistakes were corrected.</td></tr>
</table>

Date of revision	Manual No.	Revision details
Mar. 2007	IB(NA)1500036-F	Modifications were made to the followings in "II PROGRAMMING EXPLANATION". • 10.1 Tool Life Management (Machining Center System) • 10.3 PLC Axis Control The following sections were added to "III PERIPHERAL DEVELOPMENT ENVIRONMENT". • 4.8 Keyword Registration Modifications were made to the followings in "IV EXPLANATION OF BUILT-IN EDITING FUNCTION". • Descriptions on "70 Series" were added overall. The following sections were added to "IV EXPLANATION OF BUILT-IN EDITING FUNCTION". • 2.1.3 70 Series Startup • 5.2.2.3 Menu Keys in 70 Series • 5.2.3.3 Menu Keys in 70 Series • 6.4 Ladder Display Setting • 6.5 Comment Display Setting • 8.4.1.2 Program Changeover • 13.9 Keyword • 13.10 File list Mistakes were corrected.

Global service network



North America FA Center (MITSUBISHI ELECTRIC AUTOMATION INC.)

Illinois CNC Service Center
500 CORPORATE WOODS PARKWAY, VERNON HILLS, IL. 60061, U.S.A.
TEL: +1-847-478-2500 (Se FAX: +1-847-478-2650 (Se
California CNC Service Center
5665 PLAZA DRIVE, CYPRESS, CA. 90630, U.S.A.
TEL: +1-714-220-4796 FAX: +1-714-229-3818
Georgia CNC Service Center
2810 PREMIERE PARKWAY SUITE 400, DULUTH, GA., 30097, U.S.A.
TEL: +1-678-258-4500 FAX: +1-678-258-4519
New Jersey CNC Service Center
200 COTTONTAIL LANE SOMERSET, NJ. 08873, U.S.A.
TEL: +1-732-560-4500 FAX: +1-732-560-4531
Michigan CNC Service Satellite
2545 38TH STREET, ALLEGAN, MI., 49010, U.S.A.
TEL: +1-847-478-2500 FAX: +1-269-673-4092
Ohio CNC Service Satellite
62 W. 500 S., ANDERSON, IN., 46013, U.S.A.
TEL: +1-847-478-2608 FAX: +1-847-478-2690
Texas CNC Service Satellite
1000, NOLEN DRIVE SUITE 200, GRAPEVINE, TX. 76051, U.S.A.
TEL: +1-817-251-7468 FAX: +1-817-416-1439
Canada CNC Service Center
4299 14TH AVENUE MARKHAM, ON. L3R OJ2, CANADA
TEL: +1-905-475-7728 FAX: +1-905-475-7935
Mexico CNC Service Center
MARIANO ESCOBEDO 69 TLALNEPANTLA, 54030 EDO. DE MEXICO
TEL: +52-55-9171-7662 FAX: +52-55-9171-7698
Monterrey CNC Service Satellite
ARGENTINA 3900, FRACC. LAS TORRES, MONTERREY, N.L., 64720, MEXICO
TEL: +52-81-8365-4171 FAX: +52-81-8365-4171
Brazil MITSUBISHI CNC Agent Service Center
(AUTOMOTION IND. COM. IMP. E EXP. LTDA.)
ACESSO JOSE SARTORELLI, KM 2.1 18550-000 BOITUVA – SP, BRAZIL
TEL: +55-15-3363-9900 FAX: +55-15-3363-9911

European FA Center (MITSUBISHI ELECTRIC EUROPE B.V.)

Germany CNC Service Center
GOTHAER STRASSE 8, 40880 RATINGEN, GERMANY
TEL: +49-2102-486-0 FAX:+49-2102486-591
South Germany CNC Service Center
KURZE STRASSE. 40, 70794 FILDERSTADT-BONLANDEN, GERMANY
TEL: +49-711-3270-010 FAX: +49-711-3270-0141
France CNC Service Center
25, BOULEVARD DES BOUVETS, 92741 NANTERRE CEDEX FRANCE
TEL: +33-1-41-02-83-13 FAX: +33-1-49-01-07-25
Lyon CNC Service Satellite

U.K CNC Service Center
TRAVELLERS LANE, HATFIELD, HERTFORDSHIRE, AL10 8XB, U.K.
TEL: +44-1707-282-846 FAX:-44-1707-278-992
Italy CNC Service Center
ZONA INDUSTRIALE VIA ARCHIMEDE 35 20041 AGRATE BRIANZA, MILANO ITALY
TEL: +39-039-60531-342 FAX: +39-039-6053-206
Spain CNC Service Satellite
CTRA. DE RUBI, 76-80 -APDO.420 08190 SAINT CUGAT DEL VALLES, BARCELONA SPAIN
TEL: +34-935-65-2236 FAX:
Turkey MITSUBISHI CNC Agent Service Center
(GENEL TEKNIK SISTEMLER LTD. STI.)
DARULACEZE CAD. FAMAS IS MERKEZI A BLOCK NO.43 KAT2 80270 OKMEYDANI ISTANBUL, TURKEY
TEL: +90-212-320-1640 FAX: +90-212-320-1649
Poland MITSUBISHI CNC Agent Service Center (MPL Technology Sp. z. o. o)
UL SLICZNA 34, 31-444 KRAKOW, POLAND
TEL: +48-12-632-28-85 FAX:
Wroclaw MITSUBISHI CNC Agent Service Satellite (MPL Technology Sp. z. o. o)
UL KOBIERZYCKA 23, 52-315 WROCLAW, POLAND
TEL: +48-71-333-77-53 FAX: +48-71-333-77-53
Czech MITSUBISHI CNC Agent Service Center
(AUTOCONT CONTROL SYSTEM S.R.O.)
NEMOCNICNI 12, 702 00 OSTRAVA 2 CZECH REPUBLIC
TEL: +420-596-152-426 FAX: +420-596-152-112

ASEAN FA Center (MITSUBISHI ELECTRIC ASIA PTE. LTD.)

Singapore CNC Service Center
307 ALEXANDRA ROAD #05-01/02 MITSUBISHI ELECTRIC BUILDING SINGAPORE 159943
TEL: +65-6473-2308 FAX: +65-6476-7439
Thailand MITSUBISHI CNC Agent Service Center (F. A. TECH CO., LTD)
898/19,20,21,22 S.V. CITY BUILDING OFFICE TOWER 1 FLOOR 12,14 RAMA III RD BANGPONGPANG, YANNAWA, BANGKOK 10120. THAILAND
TEL: +66-2-682-6522 FAX: +66-2-682-6020
Malaysia MITSUBISHI CNC Agent Service Center
(FLEXIBLE AUTOMATION SYSTEM SDN. BHD.)
60, JALAN USJ 10/1B 47620 UEP SUBANG JAYA SELANGOR DARUL EHSAN MALAYSIA
TEL: +60-3-5631-7605 FAX: +60-3-5631-7636
JOHOR MITSUBISHI CNC Agent Service Satellite
(FLEXIBLE AUTOMATION SYSTEM SDN. BHD.)
NO. 16, JALAN SHAHBANDAR 1, TAMAN UNGKU TUN AMINAH, 81300 SKUDAI, JOHOR MALAYSIA
TEL: +60-7-557-8218 FAX: +60-7-557-3404
Indonesia MITSUBISHI CNC Agent Service Center
(PT. AUTOTEKNINDO SUMBER MAKMUR)
WISMA NUSANTARA 14TH FLOOR JL. M.H. THAMRIN 59, JAKARTA 10350 INDONESIA
TEL: +62-21-3917-144 FAX: +62-21-3917-164
India MITSUBISHI CNC Agent Service Center (MESSUNG SALES & SERVICES PVT. LTD.)
B-36FF, PAVANA INDUSTRIAL PREMISES M.I.D.C., BHOASRI PUNE 411026, INDIA
TEL: +91-20-2711-9484 FAX: +91-20-2712-8115
BANGALORE MITSUBISHI CNC Agent Service Satellite
(MESSUNG SALES & SERVICES PVT. LTD.)
S 615, 6TH FLOOR, MANIPAL CENTER, BANGALORE 560001, INDIA
TEL: +91-80-509-2119 FAX: +91-80-532-0480
Delhi MITSUBISHI CNC Agent Parts Center (MESSUNG SALES & SERVICES PVT. LTD.)
1197, SECTOR 15 PART-2, OFF DELHI-JAIPUR HIGHWAY BEHIND 32ND MILESTONE GURGAON 122001, INDIA
TEL: +91-98-1024-8895 FAX:
Philippines MITSUBISHI CNC Agent Service Center
(FLEXIBLE AUTOMATION SYSTEM CORPORATION)
UNIT No.411, ALABAMG CORPORATE CENTER KM 25. WEST SERVICE ROAD SOUTH SUPERHIGHWAY, ALABAMG MUNTINLUPA METRO MANILA, PHILIPPINES 1771
TEL: +63-2-807-2416 FAX: +63-2-807-2417
Vietnam MITSUBISHI CNC Agent Service Center (SA GIANG TECHNO CO., LTD)
47-49 HOANG SA ST. DAKAO WARD, DIST.1 HO CHI MINH CITY, VIETNAM
TEL: +84-8-910-4763 FAX: +84-8-910-2593

China FA Center (MITSUBISHI ELECTRIC AUTOMATION (SHANGHAI) LTD.)

China CNC Service Center
2/F., BLOCK 5 BLDG.AUTOMATION INSTRUMENTATION PLAZA, 103 CAOBAO RD. SHANGHAI 200233, CHINA
TEL: +86-21-6120-0808 FAX: +86-21-6494-0178
Shenyang CNC Service Center
TEL: +86-24-2397-0184 FAX: +86-24-2397-0185
Beijing CNC Service Satellite
9/F, OFFICE TOWER1, HENDERSON CENTER, 18 JIANGUOMENNEI DAJIE, DONGCHENG DISTRICT, BEIJING 100005, CHINA
TEL: +86-10-6518-8830 FAX: +86-10-6518-8030
China MITSUBISHI CNC Agent Service Center
(BEIJING JIAYOU HIGHTECH TECHNOLOGY DEVELOPMENT CO.)
RM 709, HIGH TECHNOLOGY BUILDING NO.229 NORTH SI HUAN ZHONG ROAD, HAIDIAN DISTRICT , BEIJING 100083, CHINA
TEL: +86-10-8288-3030 FAX: +86-10-6518-8030
Tianjin CNC Service Satellite
RM909, TAIHONG TOWER, NO220 SHIZILIN STREET, HEBEI DISTRICT, TIANJIN, CHINA 300143
TEL: -86-22-2653-9090 FAX: +86-22-2635-9050
Shenzhen CNC Service Satellite
RM02, UNIT A, 13/F, TIANAN NATIONAL TOWER, RENMING SOUTH ROAD, SHENZHEN, CHINA 518005
TEL: +86-755-2515-6691 FAX: +86-755-8218-4776
Changchun Service Satellite
TEL: +86-431-50214546 FAX: +86-431-5021690
Hong Kong CNC Service Center
UNIT A, 25/F RYODEN INDUSTRIAL CENTRE, 26-38 TA CHUEN PING STREET, KWAI CHUNG, NEW TERRITORIES, HONG KONG
TEL: +852-2619-8588 FAX: +852-2784-1323

Taiwan FA Center (MITSUBISHI ELECTRIC TAIWAN CO., LTD.)

Taichung CNC Service Center
NO.8-1, GONG YEH 16TH RD., TAICHUNG INDUSTIAL PARK TAICHUNG CITY, TAIWAN R.O.C.
TEL: +886-4-2359-0688 FAX: +886-4-2359-0689
Taipei CNC Service Satellite
TEL: +886-4-2359-0688 FAX: +886-4-2359-0689
Tainan CNC Service Satellite
TEL: +886-4-2359-0688 FAX: +886-4-2359-0689

Korean FA Center (MITSUBISHI ELECTRIC AUTOMATION KOREA CO., LTD.)

Korea CNC Service Center
1480-6, GAYANG-DONG, GANGSEO-GU SEOUL 157-200, KOREA
TEL: +82-2-3660-9631 FAX: +82-2-3664-8668

Notice

Every effort has been made to keep up with software and hardware revisions in the contents described in this manual. However, please understand that in some unavoidable cases simultaneous revision is not possible.
Please contact your Mitsubishi Electric dealer with any questions or comments regarding the use of this product.

Duplication Prohibited

MITSUBISHI CNC

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING,2-7-3 MARUNOUCHI,CHIYODA-KU,TOKYO 100-8310,JAPAN

MODEL	700/70 Series
MODEL CODE	008－334
Manual No.	IB-1500036(ENG)

Specifications subject to change without notice.

IB(NA)1500036(ENG)-F(0703)MEE

Printed in Japan on recycled paper.