YAMATAKE

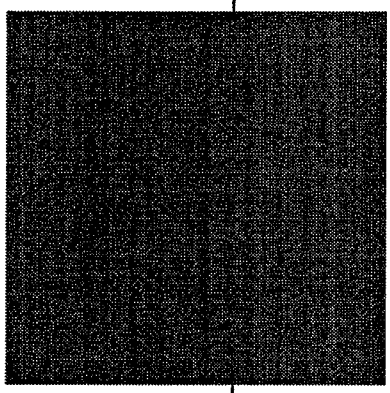
廃 番

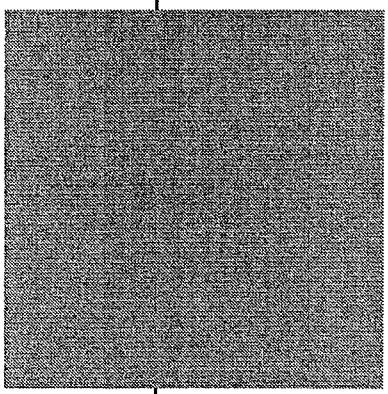
No. CP-UM-1562E

# MACHINE CONTROLLER MX SERIES

## MX200/MX100/MX50/MX30/MX20

# User's Manual Basic Programming





Thank you for purchasing the Machine Controller MX200/MX100/MX50/MX30/ MX20.
This manual contains information for ensuring correct use of the MX200/MX100/MX50/MX30/ MX20.
It also provides necessary information for installation, maintenance, and troubleshooting.
This manual should be read by those who design and maintain devices that use the MX200/MX100/MX50/MX30/ MX20.
Be sure to keep this manual nearby for handy reference.

Yamatake Corporation

廃 番

## RESTRICTIONS ON USE

This product has been designed, developed and manufactured for general-purpose application in machinery and equipment. Accordingly, when used in the applications outlined below, special care should be taken to implement a fail-safe and/or redundant design concept as well as a periodic maintenance program.

- Safety devices for plant worker protection
- Start/stop control devices for transportation and material handling machines
- Aeronautical/aerospace machines
- Control devices for nuclear reactors

Never use this product in applications where human safety may be put at risk.

## REQUEST

Ensure that this User's Manual is handed over to the user before the product is used.

Copying or duplicating this User's Manual in part or in whole is forbidden. The information and specifications in this User's Manual are subject to change without notice.

Considerable effort has been made to ensure that this User's Manual is free from inaccuracies and omissions.
If you should find any inaccuracies or omissions, please contact Yamatake Corporation.

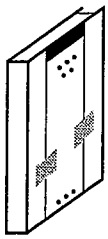
In no event is Yamatake Corporation liable to anyone for any indirect, special or consequential damages as a result of using this product.

廃 番

廃 番

# The Role of This Manual

In all, 7 manuals have been prepared for the Machine controller MX200/MX100/MX50/MX30/MX20. Read the manual according to your specific requirements. The following lists all the manuals that accompany this controller and gives a brief outline of the manual: If you do not have the required manual, contact Yamatake Corporation or your dealer.



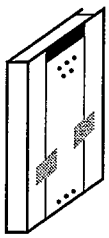
**MX200 Specifications & Installation Manual**

**Manual No. CP-UM-1604E**

This manual is required reading for first-time users of the Machine Controller MX200, those who design hardware for integrating the MX200 into operator control panels, and those who carry out maintenance.

It outlines the hardware configuration, product features and the other products used in combination with the MX200.

It also describes how to install and wire the MX200 for integrating into instruments, method of operation, maintenance and inspection, troubleshooting, and hardware specifications.



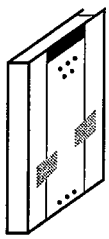
**MX100 Specifications & Installation Manual**

**Manual No. CP-UM-1257E**

This manual is required reading for first-time users of the Machine Controller MX100, those who design hardware for integrating the MX100 into operator control panels, and those who carry out maintenance.

It outlines the hardware configuration, product features and the other products used in combination with the MX100.

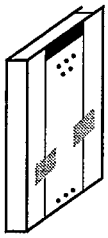
It also describes how to install and wire the MX100 for integrating into instruments, method of operation, maintenance and inspection, troubleshooting, and hardware specifications.



**MX50** **Manual No. CP-UM-1654E**

This manual is required reading for first-time users of the Machine Controller MX50, those who design hardware for integrating the MX50 into operator control panels, and those who carry out maintenance.

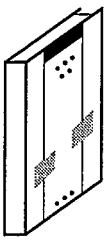
This manual describes how to install and wire the MX50 into a device, maintenance and inspection, troubleshooting and hardware specifications.



**MX30** **Manual No. CP-UM-1423E**

This manual is required reading for first-time users of the Machine Controller MX30, those who design hardware for integrating the MX30 into operator control panels, and those who carry out maintenance.

This manual describes how to install and wire the MX30 into a device, maintenance and inspection, troubleshooting and hardware specifications.

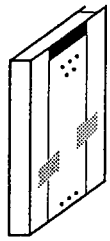


**Basic Programming** **Manual No. CP-UM-1562E**

This manual.

This manual describes the basic knowledge required for programming the MX series, the internal structure of MX200/MX100/MX50/MX30/MX20 registers and memory, and basic programming procedures.

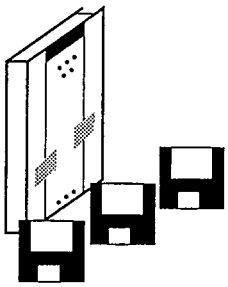
廃 番



**Programming Instruction Words** **Manual No. CP-UM-1563E**

This manual is required reading for programmers who write programs for the MX200/MX100/MX50/MX30/MX20 of machine controllers.

It gives detailed explanations of each instruction word and so can be used for reference.



**Personal Computer Loader Operation** **Manual No. CP-UM-1602E**

This manual is required reading for those who write programs for the MX200/MX100/MX50/MX30/MX20 series of machine controllers.

This manual comes with a loader software package supplied on floppy disk. The loader software package supports the MX200, MX100, MX50, MX30 and MX20, and runs on an IBM PC AT series computer or an IBM compatible machine. This manual describes how to create an "execution system disk" and how to operate the personal computer loader.

廃 番

# Organization of This User's Manual

This manual is organized as follows.

**Chapter 1 INTRODUCTION**

This chapter describes MX programming characteristics, system configuration and operation modes.

**Chapter 2 DEVICE TABLE & I/O ASSIGNMENTS**

This chapter describes the structure of device tables, devices, I/O assignments and the OP link.

**Chapter 3 PROGRAMMING NOTES**

This chapter describes factors in the ladder design and function instructions that require special attention.

**Chapter 4 DEBUGGING FUNCTION**

This chapter describes debugging by inserting breakpoints and debugging through step operations.

**Chapter 5 CONFIGURING & OPERATING PROCESSOR MODULES**

This chapter describes the internal configuration, execution operations and scan times of the processor module.

# Conventions Used in This Manual

The following conventions are used in this manual.

**! Handling Precautions**

: Handling Precautions: Handling Precautions indicate items that the user should pay attention to when handling the MX series.

廃 番

# Contents

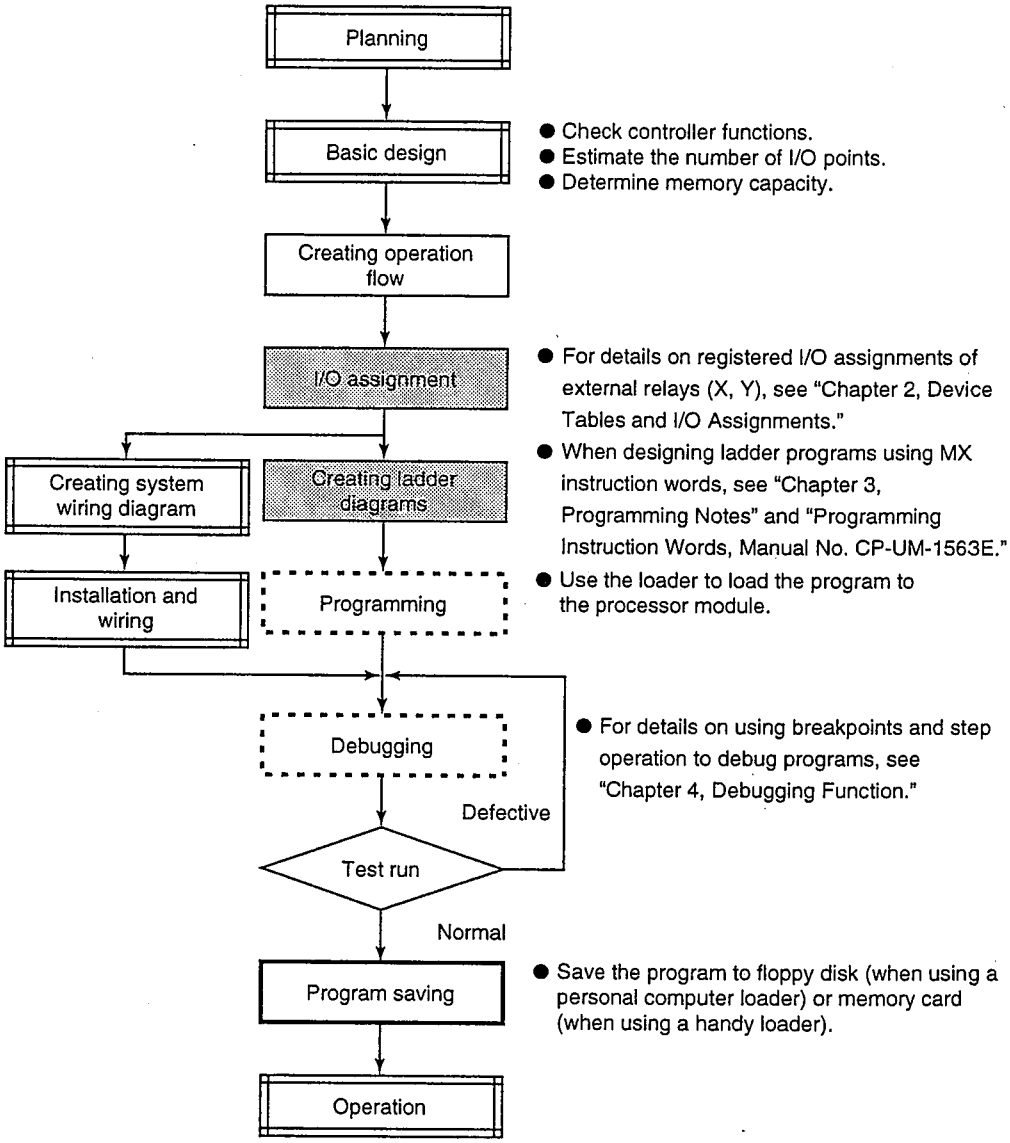
廃 番

廃番

# Chapter 1 INTRODUCTION

## 1 - 1 Design Procedures

The following flow chart describes each step involved in system design and provides references to related user manuals when using the MX series. The Basic Programming user manual mainly deals with I/O assignments and creation of ladder circuits.




— Indicates sections related to MX200/MX100/MX50/MX30/MX20 Programming Instruction Words (Manual No. CP-UM-1563E).

— Indicates sections related to MX200/MX100/MX50/MX30/MX20 Specifications & Installation Manual (Manual No. CP-UM-1604E).

— Indicates sections related to MX200/MX100/MX50/MX30/MX20 Personal Computer Loader Operation (Manual No. CP-UM-1602E).

廃 番

# 1 - 2 MX Series Programming Characteristics

Programming characteristics are as follows:

① A ladder language and mnemonic language are provided as standard.

② When a startup is standard during the processing of an instruction, the pulse can be specified by the instruction itself.

③ Data table instructions can be used to define preset multiple data items.

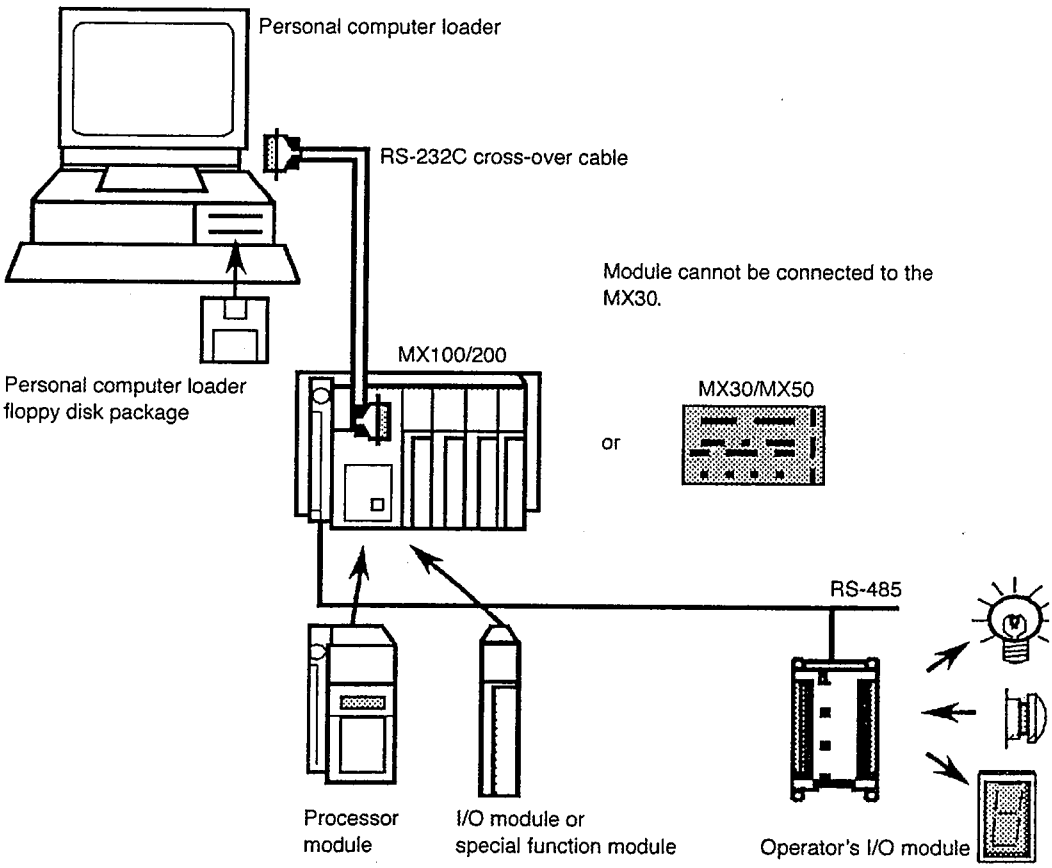
④ Strobe type instructions are provided to facilitate printer control and use of smart terminal (ST).

⑤ The use of conditional breakpoints allows real-time debugging something which was not possible with conventional controllers.

# 1 - 3 MX Series Programming Basics

## ■ Basic system

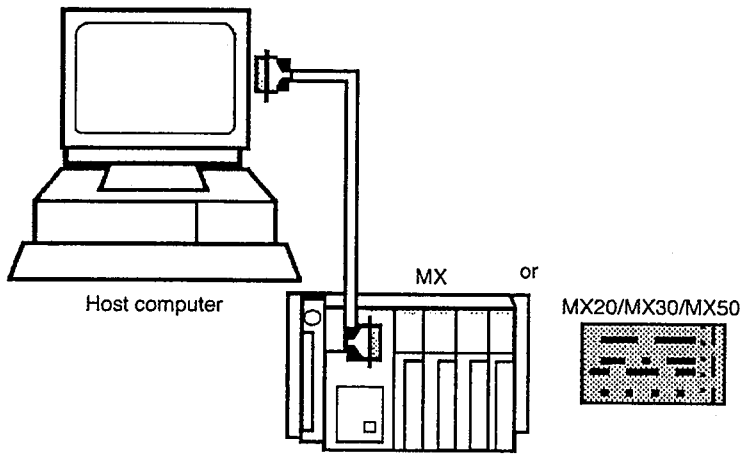
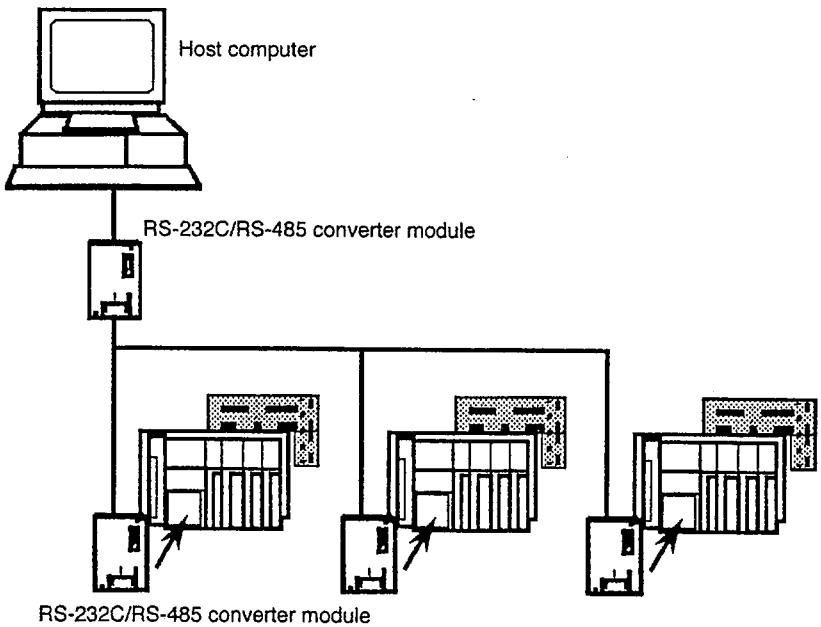
- Connect the processor module to the personal computer loader using an RS-232C crossover cable.
- If you connect operator's I/O to the processor module or MX50 using the RS-485 cable, only five wires will be required for operation button inputs and lamp outputs.



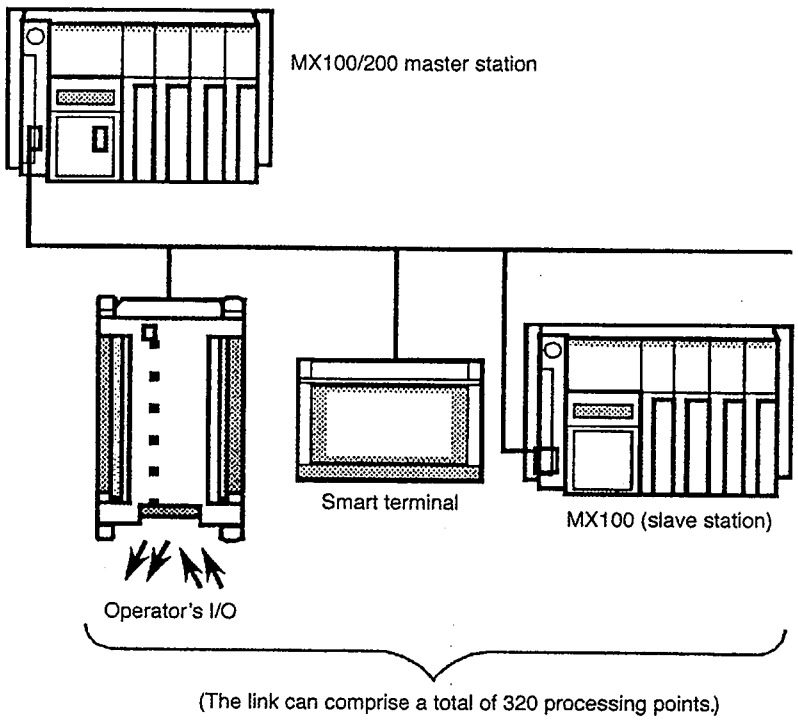
## ■ Communication systems

There are two types of communications: host communication and OP link communication.

① Host communication: Communication commands from a host computer are used for reading and writing MX data (I/O and alarms). The RS-232C port on the processor module is used for communications.
The following shows an example of a point-to-point and a multi-drop configuration that includes a host computer and an MX.

| Host communication | Description |
|---|---|
|  | Host computer: MX=1:1<br>The host computer can also be used as a personal computer loader. |
|  | Host computer: MX=1:N<br>(On the MX20, N=9, while on the MX30 N=1.) |

② OP link: An MX100/200 is used as the master and an operator's I/O (simplified remote I/O) module, ST (smart terminal) or an MX100/200 are used as slave stations in a data link that does not require a program.

| OP link | Description |
|---|---|
|  | An OP link (RS-485) can contain up to five of the following modules.<br>① Operator's I/O<br>② ST100/220<br>③ MX100/MX200<br>The link can comprise a total of 160 input processing points and 160 output processing points.<br>* ST100 has 16 input and 32 output points.<br>* ST220 has 80 input points and 64 output points.<br>(Notes)<br>A total of five MX100/200 modules can be connected as slave stations. |

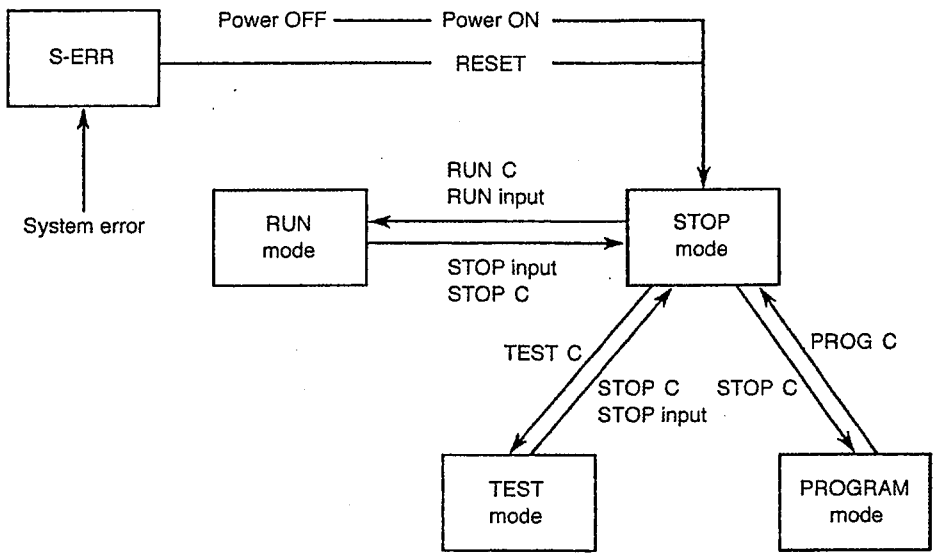
OP link: OP link stands for Operational Communication link.
This link is basically used for transmitting operational data between modules. It can be used by MX100 and MX200 controllers to establish a processor link (data link) as shown in the figure above.

# 1 - 4 MX Operation Modes

There are four MX operation modes:

1. RUN mode
   This mode is used to execute user programs
2. TEST mode
   Like the RUN mode, this mode is used for executing user programs and for real-time debugging using breakpoints during user program execution.
3. STOP mode
   The controller is in the STOP mode when the controller is turned ON or reset. User programs are stopped and I/O output is in OFF status in this mode. All other modes are accessed via this mode.
4. PROGRAM mode
   This mode is used to create programs. User programs are then stopped and I/O outputs are set to OFF.

Migration between the four modes is performed as follows.




Internal memory "device table" is held during migration from the RUN or TEST modes to the STOP mode, but is reset when the mode migrates from the STOP mode to the RUN or TEST modes.

## Handling Precautions

- When "forced set/reset" is used, the output conforms to the specified setting.
- When a "data change" is executed, output is for one scan only.
- RUN input and STOP input are contact inputs from the processor module terminal, while RUN C, STOP C and PROG C are mode selection commands input from the loader.
- The MX30/MX20 are not provided with contact inputs. They have an auto RUN function instead.
- The MX50 is not provided with contact inputs. They have an auto RUN function instead excluding when special settings are made.

廃番

# Chapter 2 DEVICE TABLE & I/O ASSIGNMENTS

## 2 - 1 Device Table Configuration

The MX series has an internal memory called a "device table." The data in the table are assigned numbers which the instructions use to perform reading and writing.

| | Device | Type | Address range | No. of points | Description |
|---|---|---|---|---|---|
| External relays | Input relay | MX200 | X000 to X19F | 320 | Assigns DI modules. |
| | | MX100 | X000 to X09F | 160 | |
| | | MX50 | X000 to X19F | 320 | When the high-speed counter is not provided |
| | | | X000 to X15F | 256 | When the high-speed counter is provided, addresses X160 to X19F are assigned for the high-speed counter. |
| | | MX30 | X000 to X03F | 64 | Assigns DI modules. |
| | | MX20 | X000 to X00B | 12 | |
| | OP link input relay | MX200 | X200 to X29F | 160 | Assigns OP I/O ST and other modules. |
| | | MX100 | | | |
| | | MX50 | | | |
| | Remote I/O input relay | MX30 | X200 to X29F | 160 | Assigns CBL remote inputs. |
| | Output relay | MX200 | Y000 to Y19F | 320 | Assigns DO modules. |
| | | MX100 | Y000 to Y09F | 160 | |
| | | MX50 | Y000 to Y19F | 320 | When the high-speed counter is not provided |
| | | | Y000 to Y15F | 256 | When the high-speed counter is provided, addresses Y160 to Y19F are assigned for the high-speed counter. |
| | | MX30 | Y000 to Y03F | 64 | Assigns DO modules. |
| | | MX20 | Y000 to Y00B | 12 | |
| | OP link output relay | MX200 | Y200 to Y29F | 160 | Assigns OP I/O ST and other modules. |
| | | MX100 | | | |
| | | MX50 | | | |
| | Remote I/O output relay | MX30 | Y200 to Y29F | 160 | Assigns CBL remote outputs. |
| Internal relays | Auxiliary relay | MX200 | M000 to M89F | 1440 | This area stores bit data. Bit data is cleared when the power is turned OFF and when the controller mode changes from STOP to RUN. |
| | | MX100 | M000 to M49F | 800 | |
| | | MX50 | M000 to M89F | 1440 | |
| | | MX30 | M000 to M49F | 800 | |
| | | MX20 | | | |
| | Latch relay | MX200 | L000 to L49F | 800 | This battery-backed up area stores bit data. |
| | | MX100 | | | |
| | | MX50 | | | |
| | | MX30 | | | |
| | | MX20 | | | |
| | Special relay | MX200 | M900 to M99F | 160 | Indicates alarm and error status. |
| | | MX100 | M900 to M94F<br>M980 to M99F | 112 | |
| | | MX50 | M900 to M99F | 160 | |
| | | MX30 | M900 to M94F<br>M980 to M99F | 112 | |
| | | MX20 | | | |
| | Timer/Counter | MX200 | T/C000 to T/C255 | 256 | Indicates when a set timer or counter value has been reached. Indicates set value when used with WTCS and RTCS instructions, and current value when used with WTCA and RTCA instructions. |
| | | MX100 | T/C000 to T/C199 | 200 | |
| | | MX50 | T/C000 to T/C255 | 256 | |
| | | MX30 | T/C000 to T/C199 | 200 | |
| | | MX20 | | | |

廃 番

| | Device | Type | Address range | No. of points | Description |
|---|---|---|---|---|---|
| Registers | Data register | MX200 | R0000 to R0499<br>R1000 to R4999 | 4500W | This battery-backed up area stores word data. |
| | | MX100 | R0000 to R0499 | 500W | |
| | | MX50 | R0000 to R0499<br>R1000 to R4999 | 4500W | |
| | | MX30<br>MX20 | R0000 to R0499 | 500W | |
| | Input register | MX200 | R0500 to R0519 | 20W | Assigned to special function modules |
| | | MX100 | R0500 to R0519 | 20W | |
| | | MX50 | R0500 to R0519 | 20W | When the high-speed counter is not provided |
| | | | R0508 to R0519 | 12W | When the high-speed counter is provided, addresses R0500 to R0507 are assigned for the high-speed counter. |
| | | MX30<br>MX20 | — | — | |
| | Output register | MX200 | R0600 to R0619 | 20W | Assigned to special function modules |
| | | MX100 | R0600 to R0619 | 20W | |
| | | MX50 | R0600 to R0619 | 20W | When the high-speed counter is not provided |
| | | | R0608 to R0619 | 12W | When the high-speed counter is provided, addresses R0600 to R0607 are assigned for the high-speed counter. |
| | | MX30<br>MX20 | — | — | |
| | Special register | MX200 | R0900 to R0999 | 100W | Registers all types of special data. |
| | | MX100 | R0900 to R0939 | 40W | |
| | | MX50 | R0900 to R0999 | 100W | |
| | | MX30<br>MX20 | R0900 to R0939 | 40W | |
| | Link register | MX200 | P0000 to P3999 | 4000W | CBL communication data register |
| | | MX100 | — | — | |
| | | MX50 | P0000 to P3999 | 4000W | |
| | | MX30<br>MX20 | — | — | |

# 2 - 2 Device Description

## ■ External relay area (X, Y)

| Relay | Type | Address range | No. of points |
|---|---|---|---|
| Input relay | MX200 | X000 to X19F | 320 |
| | MX100 | X000 to X09F | 160 |
| | MX50 | X000 to X19F | 320*1 |
| | MX30 | X000 to X03F | 64 |
| | MX20 | X000 to X00B | 12 |
| OP link output relay | MX200 | X200 to X29F | 160 |
| | MX100 | | |
| | MX50 | | |
| Remote I/O input relay | MX30 | X200 to X29F | 160 |
| Output relay | MX200 | Y000 to Y19F | 320 |
| | MX100 | Y000 to Y09F | 160 |
| | MX50 | Y000 to Y19F | 320*2 |
| | MX30 | Y000 to Y03F | 64 |
| | MX20 | Y000 to Y00B | 12 |
| OP link output relay | MX200 | Y200 to Y29F | 160 |
| | MX100 | | |
| | MX50 | | |
| Remote I/O output relay | MX30 | Y200 to Y29F | 160 |

*1 When the high-speed counter is provided, addresses X160 to X19F are assigned for the high-speed counter.

*2 When the high-speed counter is provided, addresses Y160 to Y19F are assigned for the high-speed counter.

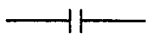
These bit devices are used in transactions between the MX and external devices. This area consists of the following relays:

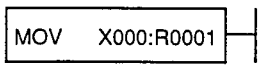
① DI/DO input and output relays
② OP link input and output relays
③ Remote I/O input and output relays

OP link input and output is used to link to operator's I/O or to MX slave stations.

- I/O assignment assigns each X and Y device to input and output modules and operator's I/O terminals. For details, see 2-4 I/O Registration (page 2-33).
- Non-I/O assigned areas, except X areas, can be used as internal relays.
- There is no limit to the number of a and b contacts for a given XY device in the program.
- The first bit address determines whether data in these areas is to be read as 1-bit, 4-bit, 8-bit, 16-bit or 32-bit data. The number of bits is identified by each instruction word.

Example X001




16 bits X000 to X00F

廃 番

## ■ Internal relay area

| Relay | Type | Address range | No. of points |
|---|---|---|---|
| Auxiliary relay | MX200 | M000 to M89F | 1440 |
| | MX100 | M000 to M49F | 800 |
| | MX50 | M000 to M89F | 1440 |
| | MX30 | M000 to M49F | 800 |
| | MX20 | | |
| Latch relay | MX200 | L000 to L49F | 800 |
| | MX100 | | |
| | MX50 | | |
| | MX30 | | |
| | MX20 | | |
| Special relay | MX200 | M900 to M99F | 160 |
| | MX100 | M900 to M94F<br>M980 to M99F | 112 |
| | MX50 | M900 to M99F | 160 |
| | MX30 | M900 to M94F<br>M980 to M99F | 112 |
| | MX20 | | |

Some internal MX relays cannot directly deal with external devices. There are no limits to the number of a and b contacts that can be used in a program.

- Auxiliary relay (M):
  This internal bit device is not battery-backed up, and data is not held when any of the following conditions occur;

  ① The MX power supply is turned OFF.
  ② STOP mode changes to RUN.
  ③ The RESET switch is pressed.
  ④ The device is reset (RST).

- Latch (hold) relay (L):
  This internal bit device is battery-backed up, and data is held when any of the following conditions occur:

  ① The MX power supply is turned OFF
  ② The RESET switch is pressed
  The latch relay is reset by issuing a latch clear from the loader.

- Special relay (M):
  These internal bit devices automatically store alarm and error conditions. M980 to M99F cannot be used by the user program. See 2-3 Special Relays and Registers for details.

## ■ Timer/counter area

| Device | Type | Address range | No. of points |
|---|---|---|---|
| Timer/Counter | MX200 | T/C000 to T/C255 | 256 |
| | MX100 | T/C000 to T/C199 | 200 |
| | MX50 | T/C000 to T/C255 | 256 |
| | MX30 | T/C000 to T/C199 | 200 |
| | MX20 | | |

The timer (T) and the counter (C) have a total of 256 points for the MX200 and a total of 200 points for the MX100/30/20. The timer and the counter cannot use the same numbers. Contacts, current values and set values are not distinguished by the numbers used but by the type of instruction used.

- Timer (T):
  Timer and counter area is used only for the ON delay timer (TMR), integrating timer (STM), timer/counter setting value write (WTCS), read (RTCS), writing current timer values (WTCA), and reading current timer values (RTCA) instruction words.

- Counters (C):
  Timer and counter area is used only for the addition counter (CNT), double-length addition counter (DCNT), up-down counter (UDC), double-length up-down counter (DUDC), timer counter set value write (WTCS), read (RTCS), current value write (WTCA) and read (RTCA) instruction words. The double-length addition counter and the double-length up-down counter use two areas each. For example, when C001 is used for the double-length addition counter, C002 is also occupied. As a result, C002 cannot be used by any other timer or counter.

The table below gives the range of timer and counter settings.

| Instruction word | Mnemonic | Setting range |
|---|---|---|
| ON delay timer | TMR | 0.01 to 9999 sec. |
| Integrating timer | STM | 0.01 to 9999 sec. |
| Addition counter | CNT | 1 to 99999 |
| Double-length addition counter | DCNT | 1 to 99999999 |
| Up-down counter | UDC | 0 to 9999 |
| Double-length up-down counter | DUDC | 0 to 99999999 |

## ■ Register area

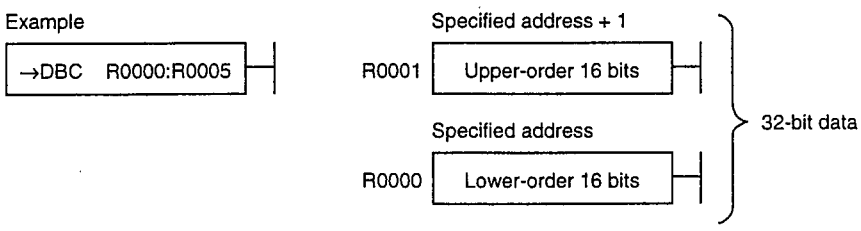
| Register | Type | Address range | No. of points |
|---|---|---|---|
| Data register | MX200 | R0000 to R0499<br>R1000 to R4999 | 4500W |
| | MX100 | R0000 to R0499 | 500W |
| | MX50 | R0000 to R0499<br>R1000 to R4999 | 4500W |
| | MX30/MX20 | R0000 to R0499 | 500W |
| Input register | MX200 | R0500 to R0519 | 20W |
| | MX100 | R0500 to R0519 | 20W |
| | MX50 | R0500 to R0519 | 20W*1 |
| | | R0508 to R0519 | 12W*2 |
| | MX30/MX20 | — | — |
| Output register | MX200 | R0600 to R0619 | 20W |
| | MX100 | R0600 to R0619 | 20W |
| | MX50 | R0600 to R0619 | 20W*1 |
| | | R0608 to R0619 | 12W*2 |
| | MX30/MX20 | — | — |
| Special register | MX200 | R0900 to R0999 | 100W |
| | MX100 | R0900 to R0939 | 40W |
| | MX50 | R0900 to R0999 | 100W |
| | MX30/MX20 | R0900 to R0939 | 40W |
| Link register | MX200 | P0000 to P3999 | 4000W |
| | MX100 | — | — |
| | MX50 | P0000 to P3999 | 4000W |
| | MX30/MX20 | — | — |

*1 When the high-speed counter is provided

*2 When the high-speed counter is provided, addresses R0500 to R0507 are assigned for the high-speed counter.

The register area, like the latch (hold) relay, is battery-backed up and the data it contains is not cleared when the power is turned OFF or the device is reset.

- Data register area:
  The data register area is a 16-bit device for storing data in the MX. So, two registers are required to store 32-bit data. The specified address is the lower-order 16 bits and the specified address +1 is the upper-order 16 bits.




- I/O register area:
  This is a 16-bit device for storing the word data of special function modules. This area is assigned by I/O assignment.

- Special register area:
  This is a 16-bit device for storing scan time data, module verification and stored data. These data items are stored automatically.

- Link register area:
  The link register is a data register for CBL communications. It is used for the following functions:
  • PC link
  • Remote I/O
  • Mail communication settings
  • Network information
  When CBL is not used, P0050 to P3199 can be used as a data register.

廃 番

# 2 - 3 Special Relays and Registers

## ■ Special relays

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M900 | Shift carry | R/W | Carry used for shift related instructions | ○ | ○ | ○ | ○ | ○ |
| M901 | Overflow | | Turns ON when operation result exceeds the allowable numeric range. | ○ | ○ | ○ | ○ | ○ |
| M902 | Underflow | | Turns ON when operation result is below the allowable numeric range. | ○ | ○ | ○ | ○ | ○ |
| M903 | Result of character string comparison | | Turns ON when a character string comparison indicates a match. Otherwise OFF. | ○ | | ○ | | |
| M904 | Remote transmission completed | | Turns ON when Y200 data onwards is transmitted to the MX200. Turns OFF when the data is moved to the CBL LSI oscillation buffer. | | | | ○ | |
| M905 | Remote data reception | | Turns ON when data received from MX200 is placed in X200 onwards. (When the END instruction has been executed, does not automatically turn OFF.) | | | | ○ | |
| | Library input conditions | | Input conditions for library instructions | ○ | | ○ | | |
| M906 | Excessive remote speed | | Turns ON when data received from MX200 is transmitted at a speed that exceeds the MX30 scan time. | | | | ○ | |
| | User flag 1 for library | | User flag 1 for library: the entry No. corresponds to bits 0 to 15 of R0960 and R0961. | ○ | | ○ | | |
| M907 | Undefined CBL instruction received | | Turns ON when the module receives a CBL command it does not support. | | | | ○ | |
| | User flag 2 for library | | User flag 2 for library: the entry No. corresponds to bits 0 to 15 of R0962 and R0963. | ○ | | ○ | | |
| M908 | Instantaneous power failure | | Turns ON after a recovery of an instantaneous power failure lasting longer than 20 ms is detected in the AC power supply and the MX power supply continued operation. | ○ | ○ | | | |
| M909 | Unsuccessful CBL ring | | Turns ON when there is an unsuccessful communication ring. | | | | ○ | |
| M90A | MT failure | | Turns ON when an MT problem causes a loopback. | | | | ○ | |
| M90B | MR failure | | Turns ON when an MR problem causes a loopback. | | | | ○ | |
| | User alarm | | The M92D turns ON when the contact is turned ON by the ladder. | ○ | | ○ | | |
| M90C | Unsuccessful CBL sub-ring | | Turns ON when a failure occurs in the sub-ring during a non-loopback state. | | | | ○ | |
| | Expansion program interface malfunction | | Turns ON when the expanded program receives a CBL or ASCII communication that it cannot directly handle. | ○ | | ○ | | |
| M90D | Special module retry occurred | | Turns ON when the READY wait time of a special function module is exceeded. | ○ | | | | |
| M90E | Operation error 1 | | Turns ON when an operation error occurs during the execution of a function instruction. Turns OFF when the error is recovered or no other instruction caused an error. | ○ | ○ | ○ | ○ | ○ |
| M90F | Operation error 2 | | Operation error 1 latch | ○ | ○ | ○ | ○ | ○ |

(Note)

The "Type" column means the following:

R/W: Reading and writing is supported in a user program

R Only reading is supported in a user program

M: Only monitoring is supported (from the loader)

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M910 | Transmission of CBL read command | R/W | CBL: Turns ON when a request for transmission of a read command is received. The system turns it OFF when a response is received. (See MX30 CBL specifications for details.) | | | | ○ | |
| M911 | Transmission of CBL write command | | CBL: Turns ON when a request for transmission of a write command is received. The system turns it OFF when a response is received. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 1. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M912 | Reception of CBL polling communication | | CBL: The system turns this contact OFF when a polling communication command is received. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 2. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M913 | Transmission of CBL polling communication | | CBL: Turns ON when a request for transmission of a polling communication command is received. The system turns it OFF when the transmission is completed. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 3. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M914 | Abnormal termination of CBL reception | | A CBL reception message terminated abnormally. | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 4. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M915 | Abnormal termination of CBL transmission | | A CBL transmission message terminated abnormally. | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 5. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M916 | Reception of CBL multiplex command | | Two or more WRITE commands and polling communications were received during one scan of MX30. (Processing during reception is determined by bit 15 of R0909.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 6. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M917 | Excessive CBL reception commands | | The amount of CBL data received exceeds processing capacity. | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 7. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M918 | Error in CPL read parameter | | CBL: The parameter setting in the read command is in error. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 8. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M919 | Error in CBL write parameter | | CBL: The parameter setting in the write command is in error. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 9. It automatically turns OFF after execution. | ○ | ○ | ○ | | |

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M91A | One-shot execution of OP link | R/W | When ON this contact issues a communication command to OP link station number 10. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M91B | Parameter error in CBL polling communication transmission | | CBL: The parameter setting in a transmitted polling communication is in error. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 11. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M91C | Error in CBL read response | | CBL: The read response is in error. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 12. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M91D | Error in CBL write response | | CBL: The write response is in error. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 13. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M91E | Error in CBL polling communication reception | | CBL: An error was found in a received polling communication. (See MX30 CBL specifications for details.) | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 14. It automatically turns OFF after execution. | ○ | ○ | ○ | | |
| M91F | CBL communication error | | OR output for CBL alarm related contacts M906, M907, M909, M90A, M90B, M90C, M90D, M914, M915, M916, M917, M918, M919, M91B, M91C, M91D, M91E | | | | ○ | |
| | One-shot execution of OP link | | When ON this contact issues a communication command to OP link station number 15. It automatically turns OFF after execution. | ○ | ○ | ○ | | |

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M920 | Normally ON | R/W | Turns ON when the power is turned ON. | ○ | ○ | ○ | ○ | ○ |
| M921 | Normally OFF | | Turns OFF when the power is turned ON. | ○ | ○ | ○ | ○ | ○ |
| M922 | ON during 1 scan | | Turns ON for only one scan after start of program in RUN/TEST mode. | ○ | ○ | ○ | ○ | ○ |
| M923 | OFF during 1 scan | | Turns OFF for only one scan after start of program in RUN/TEST mode. | ○ | ○ | ○ | ○ | ○ |
| M924 | Scan pulse | | Turns ON and OFF repeatedly for each scan in RUN/TEST mode. | ○ | ○ | ○ | ○ | ○ |
| M925 | 20 ms clock | | Turns ON and OFF repeatedly according to clock cycles. | ○ | ○ | ○ | ○ | ○ |
| M926 | 100 ms clock | | | | | | | |
| M927 | 1 second clock | | | | | | | |
| M928 | 1 minute clock | | | | | | | |
| M929 | OP link scan completed | R/W | Turns ON for only one scan when the OP link scan completes. | ○ | ○ | ○ | | |
| M92A | Remote I/O data valid | R/W | Turns ON when a remote I/O is set up and updated. | ○ | | ○ | | |
| M92B | — | — | Not used (always set to 0) | | | | | |
| M92C | | | | | | | | |
| M92D | Alarm OR output | R/W | Turns OR output caused by alarm. See table below for details. | ○ | ○ | | ○ | ○ |
| M92E | Loader monitor ON | | Turns ON when the loader is connected and the monitor is ON. | ○ | ○ | ○ | | |
| M92F | OP I/O connected | | Turns ON when an OP I/O is connected. | ○ | ○ | ○ | | |
| M930 | — | — | Not used (always set to 0) | | | | | |
| M931 | Battery failure | R/W | Turns ON when the voltage of the backup batteries is low and the batteries should be replaced. | ○ | ○ | ○ | ○ | ○ |
| M932 | Fuse blown | | Turns ON when a fuse in one of the modules in the slots is blown. | ○ | ○ | | | |
| M933 | Loader communication error | | A communication error occurred in the loader (a sum error or frame error). | ○ | ○ | ○ | ○ | ○ |
| M934 | OP I/O communication error | | Turns ON when a communication error occurred between an OP I/O and a processor module. | ○ | ○ | ○ | | |
| M935 | OP I/O verification error | | Turns ON when a registered OP I/O does not respond. | ○ | ○ | ○ | | |

M92D Description of alarm output OR

| | |
|---|---|
| MX200 | OR for M908, M90B, M90C, M90D, M90E, M90F, M931 to M935, (M936, M93A to M93F)*, M970 to M97E, M98C, M98D |
| MX100 | OR for M908, M90E, M90F, M931 to M935 |
| MX30 | M909, M90E, M90F, M931, M933 |
| MX20 | M90E, M90F, M931, M933 |

* When bit 4 of R0903 is ON, the alarm OR is canceled.

廃 番

| Address | Contact | Type | Description | |
|---|---|---|---|---|
| | | | Dedicated mode | General-purpose mode |
| M936 | ASCII parameter error | R/W | Turns ON when transmission and reception parameters exceed the register area.<br>Also M90E turns ON at this time. | Transmission and reception parameter error<br>Turns ON when there is a nonconformity in data stored in R0941 to R0949. |
| M937 | Request for forced termination of ASCII reception | R/W | Do not use this request. | Turns ON by the ladder program when ASCII reception is forcibly terminated.<br>Turns OFF automatically when ASCII reception is completed. |
| M938 | Request for ASCII transmission | R/W | Turns ON automatically when there is a request for ASCII communication (by execution of the SXRS or SXWS instructions). Turns OFF automatically when the transmission is completed. | Turns ON by the ladder program when there is a request for ASCII transmission.<br>Turns OFF automatically when ASCII transmission is completed. |
| M939 | Request for ASCII reception | R/W | Turns ON automatically when there is a request for ASCII communication (by execution of the SXRS or SXWS instructions). Turns OFF automatically when the reception is completed. | Turns ON by the ladder program when there is a request for ASCII reception.<br>Turns OFF automatically when ASCII reception is completed. |
| M93A | ASCII time-out error | R/W | Turns ON automatically when the response monitoring time set by the SNOD instruction is exceeded during ASCII communication (by execution of the SXRS or SXWS instructions) and the number of retries is completed. Turns OFF automatically when a new request for ASCII communication (by execution of the SXRS or SXWS instructions) is issued. | Turns ON when the time set from completion of ASCII transmission (M938 OFF) up to ASCII reception (M939 OFF) exceeds the monitoring time preset by R0941. Turns OFF automatically at the startup of the request for ASCII reception (M939). |
| M93B | ASCII reception buffer full | R/W | Normally OFF | Turns ON when the data in the reception buffer exceeds 1024 bytes or when the data exceeds 768 bytes during RTS control. |
| M93C | ASCII response error | R/W | Turns ON automatically when the response code for the reception data is a value other than "00". Turns OFF automatically when a new request for ASCII communication (by execution of the SXRS or SXWS instructions) is issued. | Normally OFF |
| M93D | ASCII reception data error | R/W | Turns ON when a parity, framing or overrun error is found in ASCII reception data. Turns OFF automatically when a new request for ASCII communication (by execution of the SXRS or SXWS instructions) is issued. | Turns ON when a parity, framing or overrun error is found in ASCII reception data. Turns OFF automatically when a request for ASCII reception (M939) starts. |
| M93E | ASCII reception checksum error | R/W | Turns ON when an incorrect checksum result is found in ASCII reception data. Turns OFF automatically when a new request for ASCII communication (by execution of the SXRS or SXWS instructions) is issued. | Turns ON when an incorrect checksum result is found in received data. Turns OFF automatically when a request for ASCII reception (M939) starts. |
| M93F | ASCII alarm | R/W | Turns ON when an address between M93A to M93E is ON and turns OFF when all addresses are OFF. | |

Note) M936 to M93F can only be used with the MX200/MX50.

| Address | Contact | Type | Description | | Applicable MX series device 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|---|
| M940 | STB instruction is being executed. | R/W | Y000 to 008 | ON during execution of STB instruction<br>OFF when instruction is not being executed | ○ | ○ | ○ | ○ | ○ |
| M941 | | | Y010 to 018 | | | | | | |
| M942 | | | Y020 to 028 | | | | | | |
| M943 | | | Y030 to 038 | | | | | | |
| M944 | | | Y040 to 048 | ON during execution of STB instruction<br>OFF when instruction is not being executed | ○ | ○ | ○ | | |
| M945 | | | Y050 to 058 | | | | | | |
| M946 | | | Y060 to 068 | | | | | | |
| M947 | | | Y070 to 078 | | | | | | |
| M948 | | | Y080 to 088 | | | | | | |
| M949 | | | Y090 to 098 | | | | | | |
| M94A to M94D | — | — | Not used (always set to 0) | | | | | | |
| M94E | Clock data setting | R/W | When the state of this contact changes from OFF to ON, the data in the R0994 to R0997 check registers is written to the internal clock. This is used to adjust the clock. | | ○ | | ○ | | |
| M94F | Fixed I/O | R/W | Turns ON when I/O registration data is used to fix the I/O configuration. | | ○ | ○ | ○ | ○ | |

| Address | Contact | Type | Description | | Applicable MX series device 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|---|
| M950 | Communication to one destination (response given)<br>Command transmission and reception flag | R/W | R/W buffer 0 | Conditions for turning ON:<br>Turns ON by the ladder program when a request for transmission of a read or write command is issued. Automatically turns ON at periodic intervals.<br><br>Conditions for turning OFF:<br>When a response is received<br>When the following errors occur:<br>Transmission parameter error<br>Transmission error<br>Time-out<br>Reception parameter error<br>Reception of error frame<br>Periodically occurring error | ○ | | ○ | | |
| M951 | | | buffer 1 | | | | | | |
| M952 | | | buffer 2 | | | | | | |
| M953 | | | buffer 3 | | | | | | |
| M954 | | | buffer 4 | | | | | | |
| M955 | | | buffer 5 | | | | | | |
| M956 | | | buffer 6 | | | | | | |
| M957 | | | buffer 7 | | | | | | |
| M958 | | | buffer 8 | | | | | | |
| M959 | | | buffer 9 | | | | | | |
| M95A | | | buffer 10 | | | | | | |
| M95B | | | buffer 11 | | | | | | |
| M95C | | | buffer 12 | | | | | | |
| M95D | | | buffer 13 | | | | | | |
| M95E | | | buffer 14 | | | | | | |
| M95F | | | buffer 15 | | | | | | |

| Address | Contact | Type | Description | | Applicable MX series device 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|---|
| M960 | Communication to one destination (no response) Polling communication command Transmission flag | R/W | Transmission buffer 0 | Conditions for turning ON:<br>Turns ON by the ladder program when a request for command transmission is issued.<br>Turns ON automatically at periodic intervals.<br>Conditions for turning OFF:<br>Turns OFF when transmission starts or when any of the following errors occur:<br>• Transmission error<br>• Transmission parameter error<br>• Periodically occurring error | ○ | | ○ | | |
| M961 | | | buffer 1 | | | | | | |
| M962 | | | buffer 2 | | | | | | |
| M963 | | | | | | | | | |
| M964 | Communication to one destination (no response) Polling communication command Reception flag | | Reception buffer 0 | Conditions for turning ON:<br>Turns ON by the ladder program to check reception.<br>Conditions for turning OFF:<br>Turns OFF when a command is received. | ○ | | ○ | | |
| M965 | | | buffer 1 | | | | | | |
| M966 | | | buffer 2 | | | | | | |
| M967 | | | | | | | | | |
| M968 | — | — | Not used (always set to 0) | | | | | | |
| M969 | | | | | | | | | |
| M96A | | | | | | | | | |
| M96B | | | | | | | | | |
| M96C | | | | | | | | | |
| M96D | | | | | | | | | |
| M96E | | | | | | | | | |
| M96F | | | | | | | | | |

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M970 | CBL communication error | R/W | Turns ON when any of addresses M971 to M97E is ON, and OFF when they are OFF. | ○ | | ○ | | |
| M971 | Mail communication error | | Turns ON when a mail communication error occurs. | ○ | | ○ | | |
| M972 | PC link error | | Turns ON when a PC link error occurs. | ○ | | ○ | | |
| M973 | Remote I/O error | | Turns ON when remote I/O is in error or when a registered remote I/O configuration is not matched. | ○ | | ○ | | |
| M974 | CBL address setting error | | Turns ON when the communication setting is in error. | ○ | | ○ | | |
| M975 | Unsuccessful ring | | Turns ON when communication with an adjacent CBL node is not possible. | ○ | | ○ | | |
| M976 | MT error | | Turns ON when there is a loopback on the MT. | ○ | | ○ | | |
| M977 | MR error | | Turns ON when there is a loopback on the MR. | ○ | | ○ | | |
| M978 | Loopback | | Turns ON when a loopback occurs somewhere in the local station and CBL. | ○ | | ○ | | |
| M979 | Sub-ring error | | Turns ON when a sub-ring error occurs somewhere in the local station and CBL. | ○ | | ○ | | |
| M97A | PC link P area error | | Turns ON when data is received from a link station other than the local station and in a different operation mode. | ○ | | ○ | | |
| M97B | Remote I/O P area error | | Turns ON when a data input or output of more than 26 words is specified in the single mode, when a data input or output of more than 51 words is specified in the dual mode or when data is received from a node with an address No. greater than 32 in the dual mode. | ○ | | ○ | | |
| M97C | — | — | Not used (always set to 0) | ○ | | ○ | | |
| M97D | Remote I/O EEPROM error | R/W | Turns ON when an error occurs in the EEPROM that stores the remote I/O assignment data. | ○ | | ○ | | |
| M97E | Duplicate CBL address error | | Turns ON when there is a node with the same address as the local address. | ○ | | ○ | | |
| M97F | CBL option | | Turns ON when a CBL option board is installed. | ○ | | ○ | | |

廃 番

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M980 | In compilation | M | In compilation | ○ | ○ | ○ | | |
| M981 | RUN mode | | In RUN mode | ○ | ○ | ○ | ○ | ○ |
| M982 | TEST mode | | In TEST mode | ○ | ○ | ○ | ○ | ○ |
| M983 | STOP mode | | In STOP mode | ○ | ○ | ○ | ○ | ○ |
| M984 | PROG mode | | In PROG mode | ○ | ○ | ○ | ○ | ○ |
| M985 | — | — | Not used (always set to 0) | | | | | |
| M986 | Error | M | Cause of error: OR output from address range M990 to M99F | ○ | ○ | ○ | ○ | ○ |
| M987 | Alarm | | OR output caused by alarm: same as for M92D | ○ | ○ | ○ | ○ | ○ |
| M988 | — | — | Not used (always set to 0) | | | | | |
| M989 | RUN terminal | M | Turns ON when the RUN terminal input is ON. | ○ | ○ | | | |
| M98A | — | — | Not used (always set to 0) | | | | | |
| M98B | STOP terminal | M | Turns ON when the STOP terminal input is ON. | ○ | ○ | | | |
| M98C | Special register EEPROM transmission error | M | A reset was performed during a data write to the EEPROM and data was corrupted. This is caused by a checksum error in the special register storage EEPROM. | ○ | | ○ | | |
| M98D | User EEPROM write error | | Turns ON when data is not properly written to a user EEPROM. | ○ | | | | |
| M98E | — | — | Not used (always set to 0) | | | | | |
| M98F | | | | | | | | |

| Address | Contact | Type | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| M990 | Bus error | M | Turns ON when an I/O bus line failure prevents normal access to all I/O modules. | ○ | ○ | | | |
| M991 | I/O verification error | M | Turns ON when the I/O module configuration does not match registered I/O (FIX). | ○ | ○ | ○ | | |
| M992 | I/O module error | M | Turns ON when the system contains an I/O module that cannot be identified or does not respond normally. | ○ | ○ | ○ | | |
| M993 | — | — | Not used (always set to 0) | | | | | |
| M994 | User ROM/memory card error | M | Defective user program ROM or memory card (MX100) | ○ | ○ | ○ | ○ | ○ |
| M995 | CBL communication LSI error | | Error occurred during communication LSI check. | ○ | | ○ | ○ | |
| M996 | — | — | Not used (always set to 0) | | | | | |
| M997 | | | | | | | | |
| M998 | User WDT error | M | User watchdog timer time-out | ○ | ○ | ○ | ○ | ○ |
| M999 | Program damaged | | Checksum verification error in user program | ○ | ○ | ○ | ○ | ○ |
| M99A | Program incomplete | | Syntax error in user program | ○ | ○ | ○ | ○ | ○ |
| M99B | Compile error | | Some code in user program cannot be compiled. | ○ | ○ | ○ | ○ | ○ |
| M99C | Object code size exceeded | | The size of the object code exceeds the capacity of the compiler RAM. | ○ | | ○ | | |
| M99D | — | — | Not used (always set to 0) | | | | | |
| M99E | AC power supply setting error | M | The power supply is set for 200V ac, but 100V ac is supplied. | ○ | | | | |
| M99F | Sub-processor failure | | Turns ON when an error occurs in the sub-processor, preventing data transactions between the sub-processor and the main processor. | ○ | | ○ | | |

廃 番

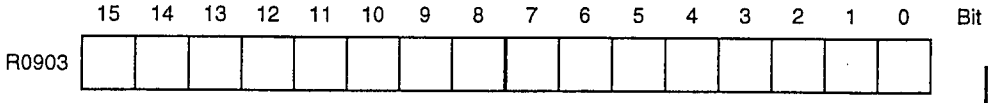
## ■ Special registers

| Address | Register | Bit configuration 15 12 | 11 8 | 7 4 | 3 0 | Description | Applicable MX series device 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| R0900 | Min. scan time | x10³ | x10² | x10 | x1 | Min. scan time value (ms)<br>Default = 9999 | ○ | ○ | ○ | ○ | ○ |
| R0901 | Last scan time | x10³ | x10² | x10 | x1 | Last scan time value<br>Default = 0000 | ○ | ○ | ○ | ○ | ○ |
| R0902 | Max. scan time | x10³ | x10² | x10 | x1 | Max. scan time value<br>Default = 0000 | ○ | ○ | ○ | ○ | ○ |
| R0903 | CBL address display LED | — | | LED display data | | Sets CBL node address in a lower-order byte at power up | | | | ○ | |
| | Hardware switching | | | | | Switching between OP link of OP I/O terminal and ASCII or between instantaneous alarm latch and clear (See Note 1). | ○ | | ○ | | |
| R0904 | Operation error 1 | Step No. | | | | The step No. where an operation error occurred. When the mode is changed from STOP to RUN or STOP to TEST, the oldest errors are cleared. Only four errors are stored at any one time.<br>Step number 1 is the oldest error and step number 4 is the latest. A binary value in the range #0000 to FFFF | ○ | ○ | ○ | ○ | ○ |
| R0905 | Operation error 2 | Step No. | | | | | | | | | |
| R0906 | Operation error 3 | Step No. | | | | | | | | | |
| R0907 | Operation error 4 | Step No. | | | | | | | | | |
| R0908 | Personal computer link computer setup data | Personal computer link data | | Not used | | Data for setting host communication with personal computer. (See Note 2.) | | | | ○ | ○ |
| | | Personal computer link data | | OP link data | | Data for setting host communication with personal computer and data for setting OP link communication. (See Note 2.) | ○ | ○ | ○ | | |
| R0909 | CBL set address | | Group | Set address | | CBL node address, group node address, etc. (See Note 3.) | | | | ○ | |

### Note 1) Data for setting hardware switching

The following table describes the meaning of the bits in register R0903.

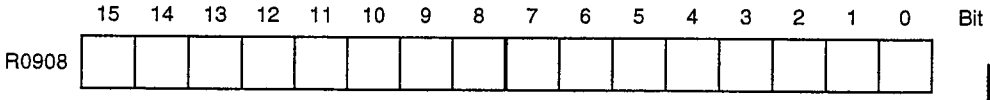
To change the setting of register R0903 in the ladder program, use the MOV (F-00) instruction. Note that data preset to the ladder program is enabled from the next scan whose setting was changed.




| Bit | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|
| | | 200 | 100 | 50 | 30 | 20 |
| 15 | 0: The OP I/O terminal is used by the OP link function.<br>1: The OP I/O terminal is used by the ASCII function. | ○ | | ○ | | |
| 14 | Not used (always set to 0) | | | | | |
| 13 | 0: Places battery alarm in M931 when battery voltage is too low.<br>1: Does not place battery alarm in M931 when battery voltage is too low. | ○ | | ○ | | |
| 12 | 0: Places alarm in M908 during detection of instantaneous alarm.<br>1: Does not place alarm in M908 during detection of instantaneous alarm. | ○ | | | | |
| 11 to 1 | Not used (always set to 0) | | | | | |
| 0 | 0: PC link errors are stored in M972 latch.<br>1: PC link errors are periodically cleared. | ○ | | ○ | | |

### Note 2) Data for setting host communications

The following table describes the meaning of the bits in register R0908.



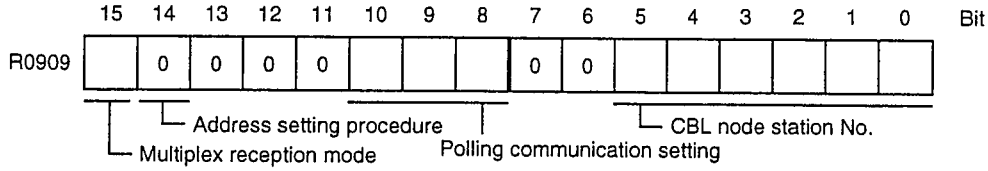

| Bit | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|
| | | 200 | 100 | 50 | 30 | 20 |
| 15<br>14 | Baud rate [0 0] 19200 bps [0 1] 9600 bps [1 0] 4800 bps [1 1] 2400 bps | ○ | ○ | ○ | ○ | ○ |
| 13 | Data format 0: 8 bit data, even parity, 1 stop bit<br>1: 8 bit data, no parity, 2 stop bits | ○ | ○ | ○ | ○ | ○ |
| 12 | 0: Restart after instantaneous power failure<br>1: Processing continues after instantaneous power failure | ○ | ○ | | | |
| 11<br>10<br>9<br>8 | Host address (0 to 15) MX100: 0 to 15<br>MX30: fixed address 1<br>MX20: 1 to 9<br>(Specified in R0950 for MX200/MX50) | | ○ | | ○ | ○ |
| 7<br>6 | Baud rate [0 0] 19200 bps [0 1] 9600 bps [1 0] 4800 bps [1 1] 2400 bps | ○ | ○ | ○ | | |
| 5 | Data format 0: 8 bit data, even parity, 1 stop bit<br>1: 8 bit data, no parity, 2 stop bits | ○ | ○ | ○ | | |
| 4 | 0: Set to OP link slave 1: Set to OP link master | ○ | ○ | ○ | | |
| 3<br>2<br>1<br>0 | OP link station (0 to 15) | ○ | ○ | ○ | | |

廃番

### Note 3) Address for setting CBL

Register R0909 is used only on the MX30.

The following table describes the meaning of the bits in register R0909.




| Bit | Description |
|---|---|
| 15 | 0: During polling reception when two or more write commands are received during one scan, all received data is directly written during the interrupt process. The simultaneity of data is lost, but all data is received.<br>1: During polling reception when two or more write commands are received during one scan, the second command onwards is ignored. The data of the second command is not received. |
| 14 | Procedure for setting address format of parameter setting area*<br>0: Decimal setting<br>1: BCD setting * Applicable to MX30 version S09 or higher. |
| 13 to 11 | Always set to 0 |
| 10 | Polling communication of destination group 3 (FAH)<br>0: Not received<br>1: Received |
| 9 | Polling communication of destination group 2 (F9H)<br>0: Not received<br>1: Received |
| 8 | Polling communication of destination group 1 (F8H)<br>0: Not received<br>1: Received |
| 7 | Always set to 0 |
| 6 | Always set to 0 |
| 5 to 0 | CBL node station No. (2 to 63)<br>The CBL function cannot be used when the station No. is 0.<br>When the station No. is 1, host communications from MX30 station Nos. 2 to 63 cannot be accessed via CBL. |

## Handling Precautions

When the R0909 setting is changed using the loader, the new setting is enabled after a system reset (by shutting down and turning ON the power again).

| Address | Register | Bit configuration (15 12 \| 11 8 \| 7 4 \| 3 0) | Description | Applicable MX series device: 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|
| R0910 | CBL: Parameter setting of read command | 0 to 499 (register No. setting) | Defines the parameter setting area required for reception and transmission using the CBL read command. | | | | ○ | |
| | OP link verification alarm | Verification failure address | This bit turns ON to indicate the address where registered OP link data and stored OP link data do not match. | ○ | ○ | ○ | | |
| R0911 | CBL: Parameter setting of write command | 0 to 499 (register No. setting) | Defines the parameter setting area required for reception and transmission using the CBL write command. | | | | ○ | |
| | Identifier indicating use/non-use of OP link | Used address | The bit corresponding to an address used for an OP link turns ON. | ○ | ○ | ○ | | |
| R0912 | CBL: Parameter setting of polling reception | 0 to 499 (register No. setting) | Defines the parameters required for polling reception. | | | | ○ | |
| | OP link verification data | 0000 0000 \| Address 1 (I \| O) | See R0913 to R0919. | ○ | ○ | ○ | | |
| R0913 | CBL: Parameter setting of polling transmission | 0 to 499 (register No. setting) | Defines the parameters required for polling transmission. | | | | ○ | |
| | OP link verification data | Address 2 (I \| O) \| Address 3 (I \| O) | The number of registered I/O words for OP link addresses are stored as 4-bit hexadecimals.<br>If not registered, stored data is used at reset. | ○ | ○ | ○ | ○ | |
| R0914 | OP link verification data | Address 4 (I \| O) \| Address 5 (I \| O) | | | | | | |
| R0915 | OP link verification data | Address 6 (I \| O) \| Address 7 (I \| O) | | | | | | |
| R0916 | OP link verification data | Address 8 (I \| O) \| Address 9 (I \| O) | | | | | | |
| R0917 | OP link verification data | Address 10 (I \| O) \| Address 11 (I \| O) | | | | | | |
| R0918 | OP link verification data | Address 12 (I \| O) \| Address 13 (I \| O) | | | | | | |
| R0919 | OP link verification data | Address 14 (I \| O) \| Address 15 (I \| O) | | | | | | |

廃 番

| Address | Register | Bit configuration (15 12 \| 11 8 \| 7 4 \| 3 0) | Description | Applicable MX series device: 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|
| R0920 | Remote communication default/variable length switching | 30H or other than 30H \| 0000 0000 | When the value of the upper-order byte is 30H, the number of remote I/O words is changed to the value specified in R0921. When the value of the upper-order byte is other than 30H, the default (3 word input/3 word output). | | | | ○ | |
| | Stored OP link data | 0000 0000 \| Address 1 (I / O) | The number of registered I/O words of each address of the OP link is stored as 4-bit hexadecimals. | ○ | ○ | ○ | | |
| R0921 | Remote communication data length setting | No. of input words (00H to 0AH) \| No. of output words (00H to 0AH) | When the upper-order byte is R0920 is 30H, upper-order byte in R0921 indicates the number of remote input words. The lower-order bytes in R0921 indicate remote output words. (When the entry exceeds 10 words, it is corrected to 10 words.) | | | | ○ | |
| | Stored OP link data | Address 2 (I / O) \| Address 3 (I / O) | The number of registered I/O words of each address of the OP link is stored as 4-bit hexadecimals. | ○ | ○ | ○ | | |
| R0922 | Stored OP link data | Address 4 (I / O) \| Address 5 (I / O) | | | | | | |
| R0923 | Stored OP link data | Address 6 (I / O) \| Address 7 (I / O) | | | | | | |
| R0924 | Stored OP link data | Address 8 (I / O) \| Address 9 (I / O) | | | | | | |
| R0925 | Stored OP link data | Address 10 (I / O) \| Address 11 (I / O) | | | | | | |
| R0926 | Stored OP link data | Address 12 (I / O) \| Address 13 (I / O) | | | | | | |
| R0927 | Stored OP link data | Address 14 (I / O) \| Address 15 (I / O) | | | | | | |
| R0928 | No. of CBL receptions | No. of CBL reception counter | The number of remote communication receptions from MX200 (binary) | | | | ○ | |
| R0929 | Blown fuse in I/O module | 0000 0000 \| Slot No. | The bit corresponding to the I/O module where the fuse was blown turns ON. | ○ | ○ | | | |

| Address | Register | Bit configuration 15 12 | 11 8 | 7 4 | 3 0 | Description | Applicable MX series device 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| R0930 | I/O module verification error | 0000 | 0000 | Slot No. | | The relevant bit turns ON when I/O module verification data and stored data differ. | ○ | ○ | ○ | ○ | ○ |
| R0931 | I/O bit/word identifier | 0000 | 0000 | 0000 | 0000 | Indicates that all I/Os are general-purpose I/Os (fixed data). | ○ | | ○ | | |
| | | Stored data | | Verification data | | The bit of each slot indicates the type of module.<br>0: General-purpose I/O module<br>1: Special I/O module | | ○ | | ○ | ○ |
| R0932 | I/O verification error | Slot 1 data | | Slot 2 data | | Indicates I/O verification data of each slot. | ○ | ○ | ○ | ○ | ○ |
| R0933 | | Slot 3 data | | Slot 4 data | | | | | | | |
| R0934 | | Slot 5 data | | Slot 6 data | | | | | | | |
| R0935 | | Slot 7 data | | Slot 8 data | | | | | | | |
| R0936 | Stored I/O data | Slot 1 data | | Slot 2 data | | Indicates stored I/O data of each slot. | ○ | ○ | ○ | ○ | ○ |
| R0937 | | Slot 3 data | | Slot 4 data | | | | | | | |
| R0938 | | Slot 5 data | | Slot 6 data | | | | | | | |
| R0939 | | Slot 7 data | | Slot 8 data | | | | | | | |

廃 番

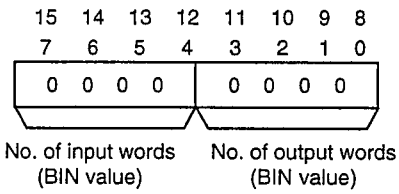
The data of I/O modules installed in each slot is stored in the following format:

● **Bit modules**

| Bit configuration | | | | | | | | Description |
|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Not defined |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 16-point input |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 32-point input |
| | α | (*) | | 0 | 0 | 1 | 1 | 32 point input + α (*) |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 16-point input/32-point output |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 16-point output |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 32-point output |
| | α | (*) | | 0 | 1 | 1 | 1 | 32-point output + α (*) |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | Not defined |
| 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 8-point input |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 24-point input |
| 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | Not defined |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 8-point input/18-point output |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 8-point output |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 24-point output |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | Not defined |

(*) α: In addition to a: 32-point input (output), a device code is added.
16I, 32I, ....
16O, 32O, ....

● **Word modules**



| Address | Register | Bit configuration (15 12 \| 11 8 \| 7 4 \| 3 0) | | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | 200 | 100 | 50 | 30 | 20 |
| R0940 | ASCII communication setting | Data defining ASCII communications | | Sets communication mode, RTS control, communication speed, data format, etc. | ○ | | ○ | | |
| R0941 | Monitoring time of ASCII communication response | Set data 0 to 9999 (specified in BCD) | | Sets the time (in 10 ms increments) for monitoring response to ASCII communication.<br>When set to 0, the response is not monitored.<br>Example: when R0941=100H, the response time is monitored for 1s. | | | | | |
| R0942 | Transmission data storage | 0 to 499, 1000 to 4999 (register Nos. are specified in BCD) | | Indicates where data transmitted in an ASCII communication comes from.<br>Example: when R0942=2000H, data stored in R2000 onwards is transmitted. | | | | | |
| R0943 | Transmission frame definition | Start code | End code | Sets the start and end code of data in an ASCII transmission. | | | | | |
| R0944 | Procedure for creating transmission checksum data | Procedure for adding/not adding checksum data | Position of checksum data | Procedure for adding/not adding transmission checksum data and how to create checksum data.<br>(Example:00 is 8 bit addition and 01 is 1's complement.)<br>It also sets the position of checksum data. | | | | | |
| R0945 | Range of transmission checksum calculation | Position for starting checksum calculation | Position for ending checksum calculation | Sets the start and end position of a transmission data checksum calculation. | | | | | |
| R0946 | Reception data storage | 0 to 499, 1000 to 4999 (register Nos. are specified in BCD) | | Indicates where data received in an ASCII communication is stored.<br>Example: when R0946=3000H, received data is stored in R3000 onwards. | | | | | |
| R0947 | Reception frame definition | Start code | End code | Sets the start and end code of data in an ASCII reception. | | | | | |
| R0948 | Procedure for creating reception checksum data | Procedure for adding/not adding checksum | Position of checksum data | Procedure for adding/not adding reception checksum data and how to create checksum data.<br>(Example:00 is 8 bit addition and 01 is 1's complement.)<br>It also sets the position of checksum data. | | | | | |
| R0949 | Range of reception checksum calculation | Position of checksum start data | Position of checksum end data | Sets the start and end position of a reception data checksum calculation. | | | | | |

廃 番

**Note) Data for setting host communication**

The following table describes the meaning of the bits in register R0940.

To change the setting of register R0903 in the ladder program, use the MOV (F-00) instruction. Note that data preset to the ladder program is enabled from the next scan whose setting was changed.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R0940 | | | 0 | | | | 0 | 0 | | | | | | | | 0 | |

| Bit | Description | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 15 | 0: ASCII communication function | | | | 1: Loader, host communication function | | | |
| 14 | 0: General-purpose mode | | | | 1: Dedicated mode | | | |
| 13 | Not used: Always set to 0 | | | | | | | |
| 12 | 0: RS-232C | | | | 1: RS-485 | | | |
| 11 | 0: 5-wire RS-485 (full duplex) | | | | 1: 3-wire RS-485 (half duplex) | | | |
| 10 | 0: No control | | | | 1: RTS control | | | |
| 9<br>8 | Not used: Always set to 0 | | | | | | | |
| 7<br>6<br>5 | 0<br>0<br>0 19200 bps | 0<br>0<br>1 9600 bps | 0<br>1<br>0 4800 bps | 0<br>1<br>1 2400 bps | 1<br>0<br>0 1200 bps | 1<br>0<br>1 600 bps | 1<br>1<br>0 Not used | 1<br>1<br>1 Not used |
| 4 | 0: 7 bits | | | | 1: 8 bits | | | |
| 3<br>2 | 0 No parity<br>0 | | 0 No parity<br>1 | | 1 Even parity<br>0 | | 1 Odd parity<br>1 | |
| 1 | 0: 1 stop bit | | | | 1: 2 stop bit | | | |
| 0 | Not used: Always set to 0 | | | | | | | |

(Note)

Half duplex: When data transmission is completed, the output driver is disabled within 500 μs.

RST control: When the reception buffer contains 768 or more data items, the RTS signal is automatically set to a non-active state.

| Address | Register | Bit configuration (15 12 \| 11 8 \| 7 4 \| 3 0) | Description | Applicable MX series device: 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|
| R0950 | MX address | 1 to 63, 7FH | MX address for CBL and host communication. When the MX address is set to 7H, the MX is set not to participlate in the CBL network. | ○ | | ○ | | |
| R0951 | Group class of polling communication reception | 0 to 7 | No reception when set to 0 (default: 0) | | | | | |
| R0952 | Single/dual mode setting | 0 to 1 | 0: Single mode<br>1: Dual mode (default) | | | | | |
| R0953 | Group address for destination of PC link transmission | 0 to 3 | No transmission when set to 0 (default: 0) | | | | | |
| R0954 | Group class for PC link reception | 0 to 7 | No reception when set to 0 (default: 0) | | | | | |
| R0955 | PC link monitoring time | 0 to 1 | 0: 500 ms (default)<br>1: 1s | | | | | |
| R0956 | No. of connected remote slave stations | 0 to 31 | 0: Default | | | | | |
| R0957 | — | — | Not used (always set to 0) | | | | | |
| R0958 | ROM version | Main CPU ROM version \| Sub-CPU ROM version | Indicates the ROM versions for the main and sub-CPUs in a 2-digit decimal format. | | | | | |
| R0959 | MX operation address | 1 to 63, 7FH | Copy of R0950 | | | | | |

Note) The following table describes the meaning of the bits in register R0950.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R0950 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |

| Bit | Description |
|---|---|
| 15 to 6 | Not used: Always set to 0 |
| 5 | Address 1 to 63 |
| 4 | |
| 3 | |
| 2 | |
| 1 | |
| 0 | |

The following table describes the meaning of the bits in register R0951.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R0951 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |

| Bit | Description |
|---|---|
| 15 to 3 | Not used: Always set to 0 |
| 2 | Polling communication of destination group 3 (FAH)<br>0: Not received<br>1: Received |
| 1 | Polling communication of destination group 2 (F9H)<br>0: Not received<br>1: Received |
| 0 | Polling communication of destination group 1 (F8H)<br>0: Not received<br>1: Received |

The following table describes the meaning of the bits in register R0954.

| | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| R0954 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |

| Bit | Description |
|---|---|
| 15 to 3 | Not used: Always set to 0 |
| 2 | PC link group 3<br>0: Not received<br>1: Received |
| 1 | PC link group 2<br>0: Not received<br>1: Received |
| 0 | PC link group 1<br>0: Not received<br>1: Received |

| Address | Register | Bit configuration (15 12 \| 11 8 \| 7 4 \| 3 0) | Description | Applicable MX series device | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 200 | 100 | 50 | 30 | 20 |
| R0960 | Library Internal flag M906: Save data 1 | Entry Nos. 15 to 0 correspond to bits 15 to 0. | Special contact used in the library. Save data of each M906 library | ○ | | ○ | | |
| R0961 | Library Internal flag M906: Save data 2 | Entry Nos. 31 to 16 correspond to bits 15 to 0. | | | | | | |
| R0962 | Library Internal flag M907: Save data 1 | Entry Nos. 15 to 0 correspond to bits 15 to 0. | Special contact used in the library. Save data of each M907 library | | | | | |
| R0963 | Library Internal flag M907: Save data 2 | Entry Nos. 31 to 16 correspond to bits 15 to 0. | | | | | | |
| R0964 | Library Entry data 1 | Entry Nos. 15 to 0 correspond to bits 15 to 0. | Indicates numbers of executable libraries. | | | | | |
| R0965 | Library Entry data 2 | Entry Nos. 31 to 16 correspond to bits 15 to 0. | | | | | | |
| R0966 | — | — | Not used (always set to 0) | | | | | |
| R0967 | | | | | | | | |
| R0968 | Incremented every 10 ms, BCD data | 0000 to 9999 | This register is incremented every 10 ms by adding a BCD value (substitute function of library timer). | | | | | |
| R0969 | Incremented every 10 ms, BIN data | 0 to 65535 | This register is incremented every 10 ms by adding a BIN value (substitute function of library timer). | | | | | |

廃 番

| Address | Register | Bit configuration (15 12 11 8 7 4 3 0) | Description | Applicable MX series device: 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|
| R0980 | Mounted memory device | 0000 **** 0000 00*0<br>*: 0 or 1 | Indicates whether memory for user program library stored in processor PCB is available or not.<br>0: Available 1: Not available | ○ | | ○ | | |
| R0981 | User program Stored memory device | 0000 00** 0000 00*1<br>*: 0 or 1 | Indicates whether the area assigned to user programs in the installed memory device is used or not.<br>0: Not used 1: Used | | | | | |
| R0982 | Memory device where user program error occured | 0000 00** 0000 00**<br>*: 0 or 1 | Indicates whether an error has occurred in the memory area assigned to user programs or not.<br>0: No error 1: Error occurred | | | | | |
| R0983 | Library storage memory device | 0000 00** 0000 0000<br>*: 0 or 1 | Indicates whether an area assigned for library use is used or not.<br>0: Not used 1: Used | | | | | |
| R0984 | Library memory device | 0000 **** 0000 0000<br>*: 0 or 1 | Indicates whether an error has occurred in the memory area assigned to the library or not.<br>0: No error 1: Error occurred | | | | | |
| R0985 | Ladder program ROM → RAM transfer setting | 0000 00** 0000 0000<br>*: 0 or 1 | Defines whether the ladder program is to be transferred from ROM to RAM or not.<br>0: Not transferred 1: Transferred | | | | | |
| R0986 | Data table ROM → RAM transfer setting | 0000 00** 0000 0000<br>*: 0 or 1 | Defines whether the data table is to be transferred from ROM to RAM or not.<br>0: Not transferred 1: Transferred | | | | | |
| R0987 | EEPROM write enable device | 0000 **** 0000 0000<br>*: 0 or 1 | Indicates whether writing to EEPROM is enabled or disabled.<br>0: Disabled 1: Enabled | | | | | |
| R0988<br>R0989 | — | — | Not used (always set to 0) | | | | | |

The meaning of each bit is described in the table on the next page.

The following table describes the meaning of the bits in registers R0980 to R0987.

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | Bit |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | | | |

| Bit | Description |
|---|---|
| 15 to 12 | Not used: Always set to 0 |
| 11 | Indicates state of EPROM/EEPROM (192 to 256 Kbytes) |
| 10 | Indicates state of EPROM/EEPROM (128 to 192 Kbytes) |
| 9 | Indicates state of EPROM/EEPROM (64 to 128 Kbytes) |
| 8 | Indicates state of EPROM/EEPROM (0 to 64 Kbytes) |
| 7 to 2 | Not used: Always set to 0 |
| 1 | Indicates state of expanded RAM |
| 0 | Indicates state of standard RAM |

廃 番

| Address | Register | Bit configuration (15 12 / 11 8 / 7 4 / 3 0) | Description | Applicable MX series device: 200 | 100 | 50 | 30 | 20 |
|---|---|---|---|---|---|---|---|---|
| R0990 | — | — | Not used (always set to 0) | ○ | ○ | ○ | ○ | ○ |
| R0991 | Basic display setting | | Sets the data normally displayed on the processor module<br>0: User mode<br>1: MX address<br>2: Error/Alarm | | | | | |
| R0992 | User display code 1 (7-segment LED) | 0 to F / 0 to F / 0 to F / 0 to F | Data displayed in the user mode by the 7-segment LED on the processor module | | | | | |
| R0993 | User display code 2 (8-bit data LED) | 00 to FF | Data displayed in the user mode by the 8-bit data LED on the processor module | | | | | |
| R0994 | Clock register | Year | Indicates the year<br>Example: 1993 is displayed as 1993[H] | | | ○ | | |
| R0995 | | Month / Day | Upper-order byte: Month,<br>Lower-order byte: Day<br>Example: November 5 is displayed as 1105[H] | | | | | |
| R0996 | | Hour / Minute | Upper-order byte: Hour,<br>Lower-order byte: Minute<br>Example: PM3:28 is displayed as 1528[H] | | | ○ | | |
| R0997 | | Second / Day of the week | Upper-order byte: Seconds,<br>Lower-order byte: Days of the week<br>Days of the week: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday → 0 1 2 3 4 5 6 | | | | | |
| R0998<br>R0999 | — | — | Not used (always set to 0) | | | | | |

# 2 - 4 I/O Registration

## ■ General-purpose I/O modules

The MX series processor module automatically reads the data (stored data) in I/O modules in the I/O slots when the power is turned ON or when the RESET switch is pressed. Assignment is based on how close an I/O module is to the processor module. Inputs start from X000 and outputs from Y000. Assignments are automatically left justified if there are any unoccupied slots.

| Processor module | 32 input points X000 X01F | 16 output points Y000 Y00F | Unoccupied | 16 input points X020 X02F | 32 output points Y010 Y02F | |
|---|---|---|---|---|---|---|

I/Os exceeding 512 points (a maximum of 320 input points and 320 output points) are ignored. When I/Os are registered by the loader, their data is registered as verification data, and a difference between stored data and the verification data is indicated as an error. If not registered, the stored data is taken as verification data, each time the power is turned ON.

Refer to the MX Series User's Manual "Personal Computer Loader" CP-UM-1602E for details.

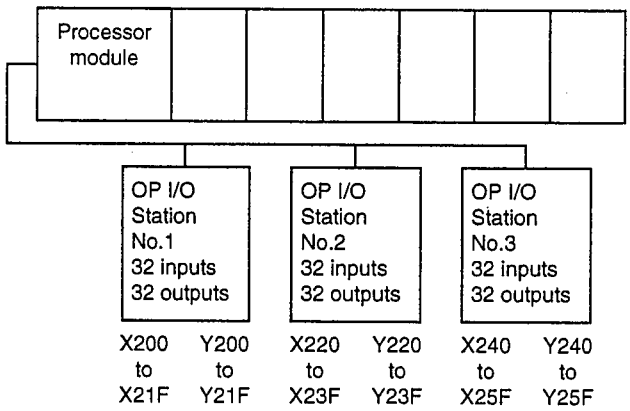
## ■ Operator's I/O

The I/O for the OP link are registered by setting the number of words required by the number of I/O points with the loader. Inputs are assigned from X200, while outputs are assigned from Y200. Note that inputs/outputs exceeding 160 points are ignored.

An alarm is displayed in the following cases:

- No OP I/O response at startup.
- Stored OP I/O data differs from registered data
- No OP I/O response during operation

Unless the OP I/Os are assigned with the loader they will not function even if they are connected.



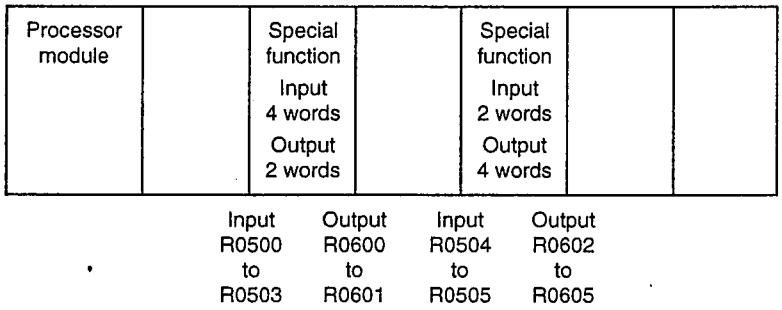

## ! Handling Precautions

The processor module does not automatically read the OP I/O data when the OP link I/O is not registered.

廃 番

## ■ Special I/O modules

MX special function modules are assigned to I/O registers in word units (16 bits.) Inputs are assigned to R0500 to R0519 and outputs are assigned to R0600 to R0619 starting from those closest to the processor module.
Assignment starts automatically when the power is turned ON or when the RESET switch is pressed. Special function modules may be mounted in any I/O slot.
After assignment, a missing I/O module is detected by comparing the I/O module data with registered I/O module data (if I/O assignments are registered in the personal computer loader) or with the data stored in the processor when the power was turned ON or when the RESET switch was pressed.

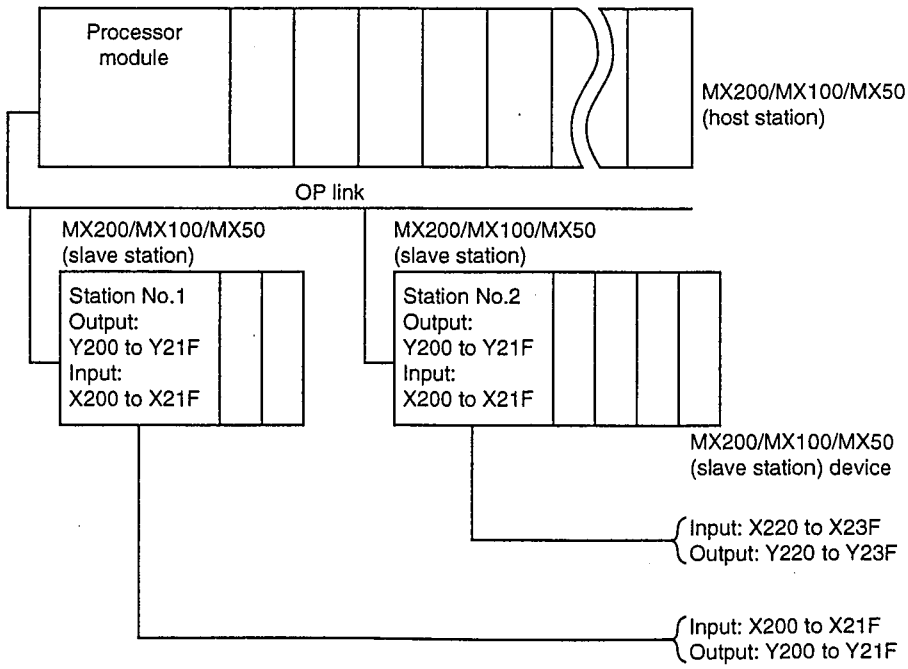



See the user manual supplied with the special module for details.
Special I/O modules refers to analog I/O modules and special function modules.

## ■ MX200/MX100/MX50

When MX200/MX100/MX50 devices are connected to an OP link (RS-485) to form a data link between host and slave stations, I/O registration is required just as in the case of operator's I/O connections.
Note that 32 inputs and 32 outputs (fixed values) can be assigned to an MX200/MX100/MX50 device used as a slave station.

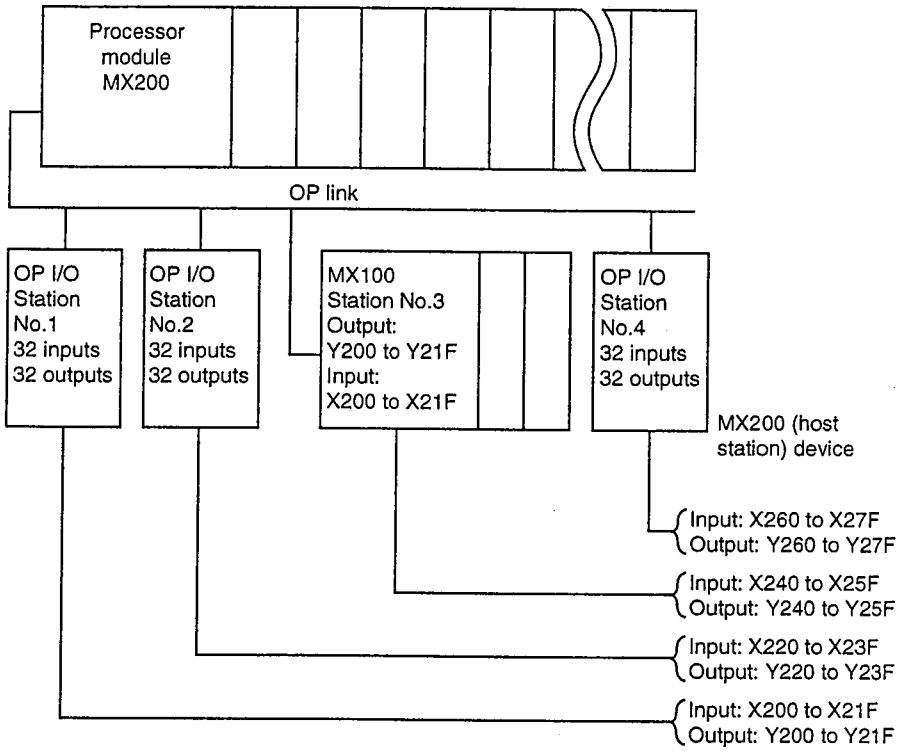
Example



## ■ Environment comprising operator's I/O and MX200/MX100/MX50 controllers

When operator's I/O and MX200/MX100/MX50 slave stations are connected to an OP link (RS-232C), they are assigned as normal operator's I/O. (In this environment, MX200/MX100/MX50 each has 32 input and 32 output points.)

Example

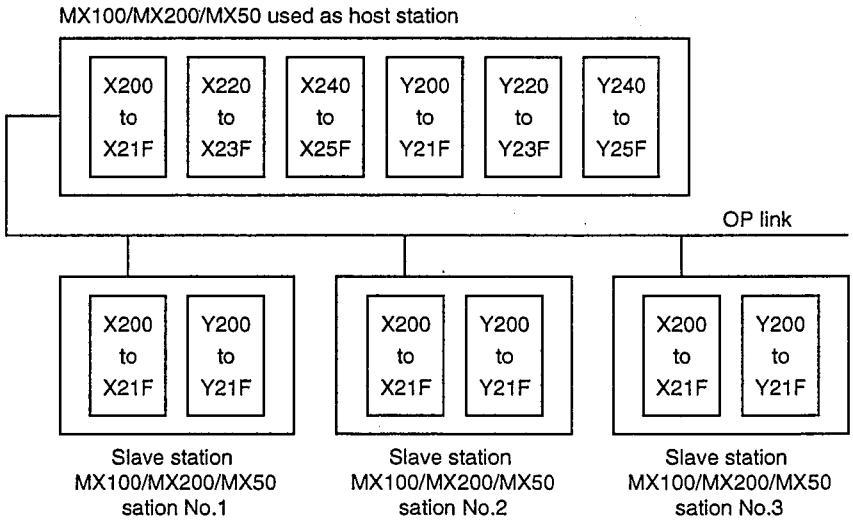


廃 番

# 2 - 5 OP Links

The following factors have to be kept in mind when data transactions are performed between MX100/MX200/MX50 host and slave stations connected to an OP link (RS-485).

1. The data link area in the MX100/MX200/MX50 host stations is the OP link area (X200 onwards , Y200 onwards).
2. The data link area for MX100/MX200/MX50 slave stations is fixed at X200 to X21F and Y200 to Y21F.
3. Host stations receive the values of slave station output relays (Y200 to Y21F) through the local station input relay (X200 onwards) and outputs the value of the local station output (Y200 onwards) to the slave station input relay (X200 to X21F).
4. The host (master) station, slave station and slave station numbers of an MX200 are set on a personal computer loader or processor module key operation. On the MX50 controller, the personal computer loader is used, and on the MX100 controller the DIP switches on the processor module are used to do the same. These are also the only procedure available.
5. Modules connected to an OP link must be I/O registered.




When Y200 on an MX100/MX200/MX50 controller used as slave station No.1 turns ON, the MX200 on host station MX200 also turns ON. Alternatively, when the Y200 on the host station turns ON, the X200 on slave station No.2 also turns ON.

廃番

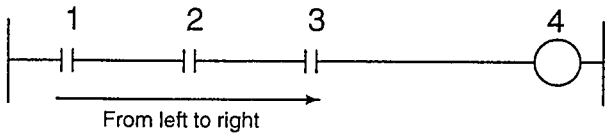
# Chapter 3 PROGRAMMING NOTES

## 3 - 1 Ladder Design Notes

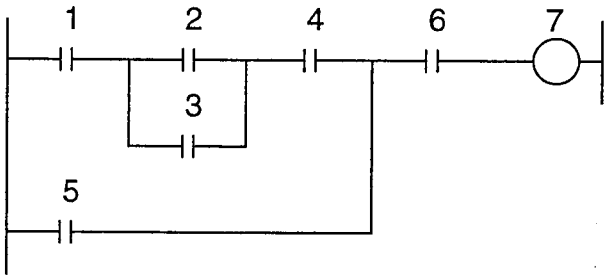
### ■ Program execution sequence

The sequence of program execution is shown by the mnemonics listed below.

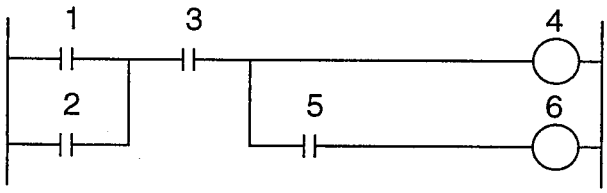
(Ladder) (Mnemonic)




```
LD    1
AND   2
AND   3
OUT   4
```




```
LD    1
LD    2
OR    3
ANB
AND   4
OR    5
AND   6
OUT   7
```

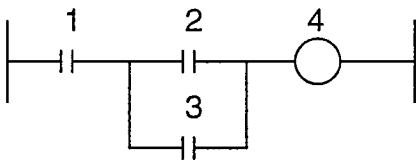



```
LD    4
OR    2
AND   3
LPS
OUT   4
LPP
AND   5
OUT   6
```

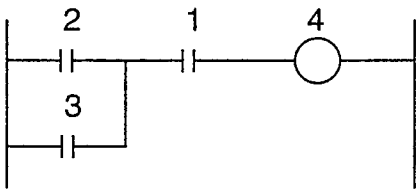
## ■ Program shortcuts

The position of the instruction will change the number of bytes needed when other factors in the ladder program are identical. The number of bytes can be reduced by placing contact instructions to the left and reducing the number of contacts as the column descends.

(Example 1) (Ladder) (Mnemonic)

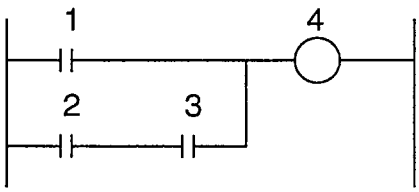



| | | No. of bytes |
|---|---|---|
| LD | 1 | 2 |
| LD | 2 | 2 |
| OR | 3 | 2 |
| ANB | | 2 |
| OUT | 4 | 2 |
| | | Total 10 |

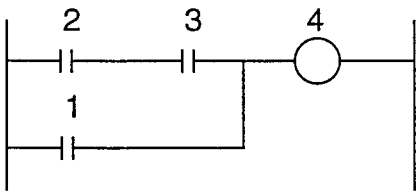



| | | |
|---|---|---|
| LD | 2 | 2 |
| OR | 3 | 2 |
| AND | 1 | 2 |
| OUT | 4 | 2 |
| | | Total 8 |

(Example 2) (Ladder) (Mnemonic)

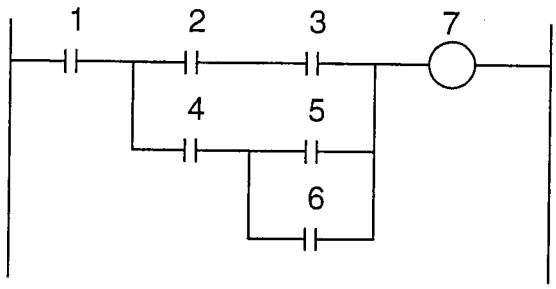



| | | No. of bytes |
|---|---|---|
| LD | 1 | 2 |
| LD | 2 | 2 |
| AND | 3 | 2 |
| ORB | | 2 |
| OUT | 4 | 2 |
| | | Total 10 |

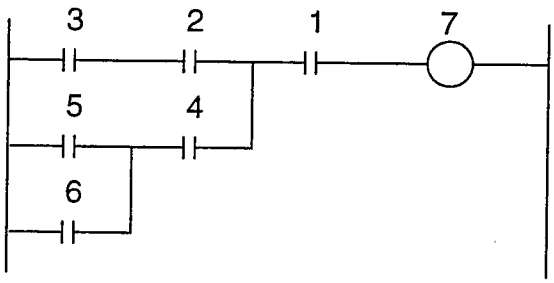



| | | |
|---|---|---|
| LD | 2 | 2 |
| AND | 3 | 2 |
| OR | 1 | 2 |
| OUT | 4 | 2 |
| | | Total 8 |

(Example 1) (Ladder) (Mnemonic)




| | | No. of bytes |
|---|---|---|
| LD | 1 | 2 |
| LD | 2 | 2 |
| AND | 3 | 2 |
| LD | 4 | 2 |
| LD | 5 | 2 |
| OR | 6 | 2 |
| ANB | | 2 |
| ORB | | 2 |
| ANB | | 2 |
| OUT | 7 | 2 |
| | | Total 20 |



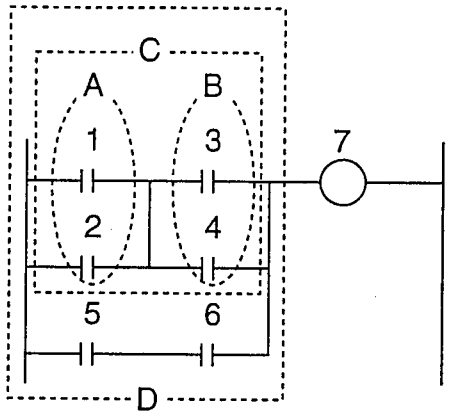

| | | |
|---|---|---|
| LD | 3 | 2 |
| AND | 2 | 2 |
| LD | 5 | 2 |
| OR | 6 | 2 |
| AND | 4 | 2 |
| ORB | | 2 |
| AND | | 2 |
| OUT | 7 | 2 |
| | | Total 16 |

## ■ Creating programs for serial and parallel circuits

Programs for serial circuits are created by dividing the circuit into blocks which are placed on top of each other.

(Example 1)

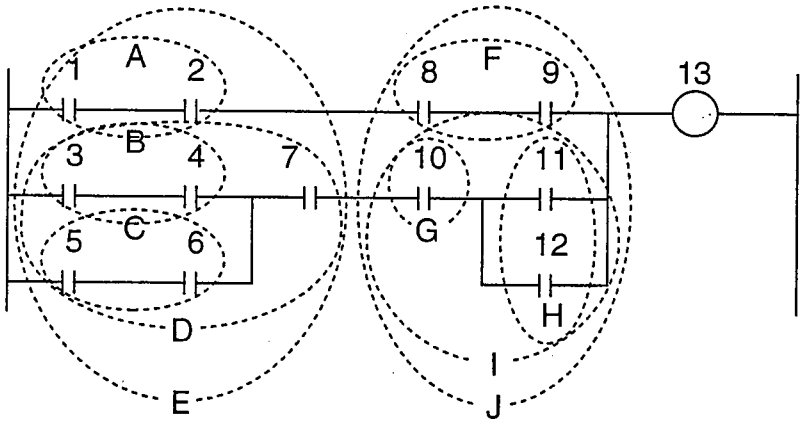
(Ladder)




(Mnemonic)

```
LD    1
OR    2
LD    3
OR    4
ANB       AND block for A and B above
LD    5
AND   6
ORB       OR block for C and D above
OUT   7
```

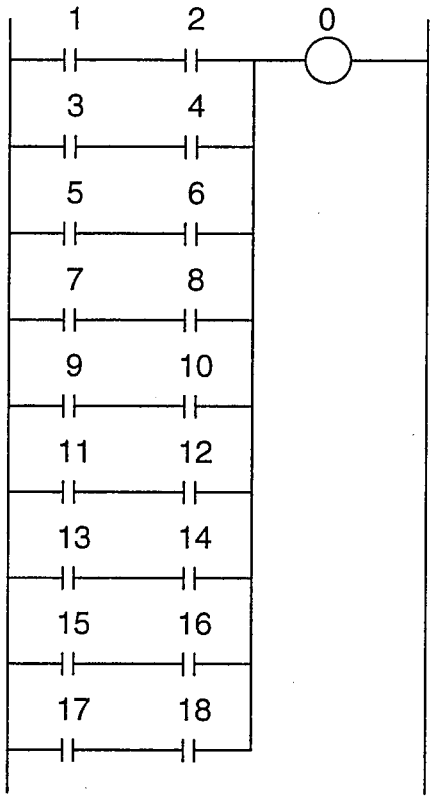
(Example 2)




```
LD    1
AND   2
LD    3
AND   4
LD    5
AND   6
ORB       OR block for B and C above
AND   7
ORB       OR block for A and D above
LD    8
AND   9
LD    10
LD    11
OR    12
ANB       AND block for G and H above
ORB       AND block for F and I above
ANB       AND block for E and J above
OUT   13
```

- Repeated entries of ANB or ORB can be made after repeated creation of blocks including LD or LDNOT. However, as only 8 blocks can be stored temporarily, it is not possible to create nine or more blocks at one time. If more blocks are required, use ANB (or ORB) after creating the first blocks, then add more blocks.

(Example)




More than 9 blocks so not executable

```
LD   1          LD   1
AND  2          AND  2
LD   3          LD   3
AND  4          AND  4
LD   5          ORB
AND  6          LD   5
LD   7          AND  6
AND  8          ORB
LD   9          LD   7
AND  10         AND  8
LD   11         ORB
AND  12         LD   9
LD   13         AND  10
AND  14         ORB
LD   15         LD   11
AND  16         AND  12
LD   17         ORB
AND  18         LD   13
ORB             AND  14
ORB             ORB
ORB             LD   15
ORB             AND  16
ORB             ORB
ORB             LD   17
ORB             AND  18
ORB             ORB
OUT  0          OUT  0
```

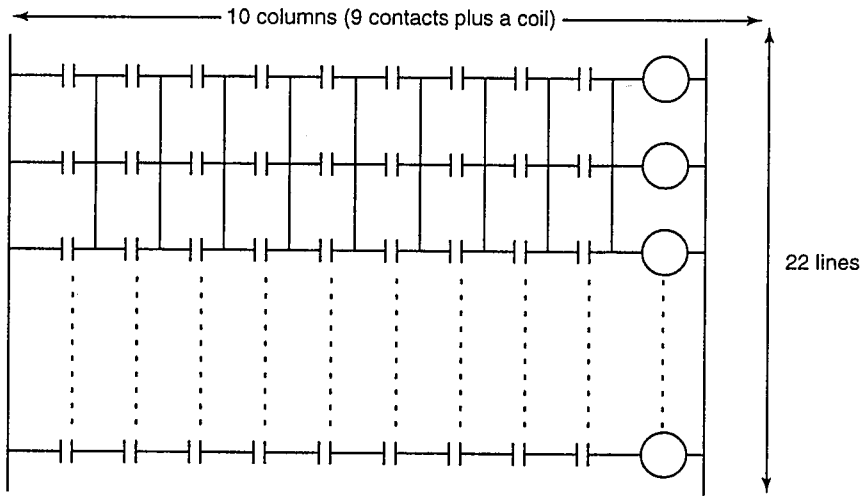
Program as indicated above.

## ■ Ladder display constraints

The following describes constraints that apply to writing ladder programs. They do not apply to the entry of mnemonics, so they are entry constraints and not execution constraints.

### ● Maximum range of circuit blocks

The maximum number of entries for a circuit block (section connected by the right and left buses) is 10 columns (nine contacts plus a coil) times 22 lines.

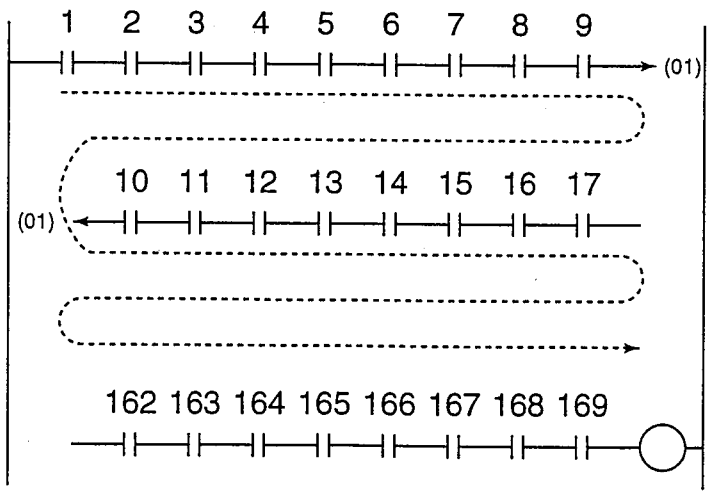



### [!] Handling Precautions

- A program written in mnemonics that exceed this constraint cannot be displayed as a ladder, but can be executed.
- A total of 10 LPSs (branch start) can be used in one circuit block.

### ● Line loopback

Up to 169 contacts can be connected in series in a ladder display.




### [!] Handling Precautions

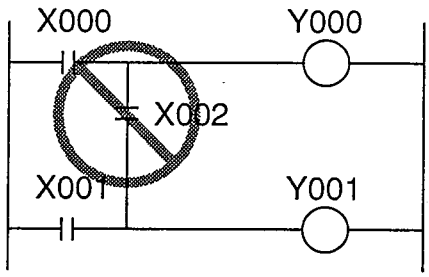
- A program written in mnemonics that exceed this constraint cannot be displayed as a ladder, but can be executed.
- Branch lines cannot be looped back using LPS (branch start), LRD and LPP.

## ■ Precautions to be taken in creating circuits

### ● Writing from relay panel circuit

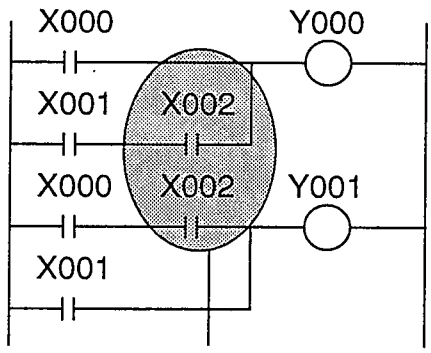
The current in a relay circuit panel flows in both directions. Such circuits are configured as follows. However, as such a circuit cannot be handled by the MX series of controllers it must be adapted as shown.

(Example)




This kind of circuit is not possible on the MX series.

Redraw the circuit as shown below.



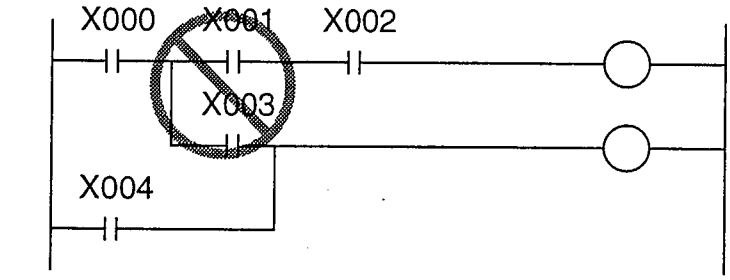

Take care to prevent ladder circuits from interfering with each other.

廃番

## ● Branch back circuit not permitted

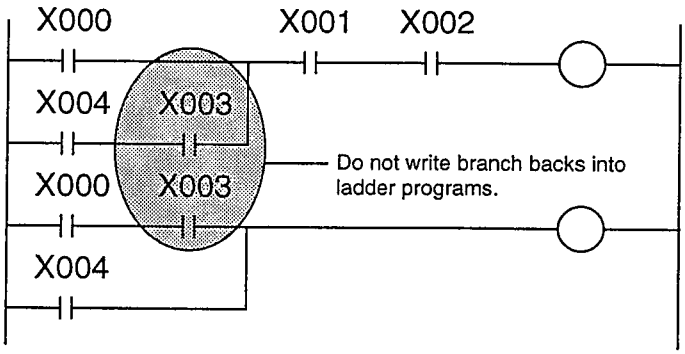
Do not write branch back circuits like the one shown below.

(Example)

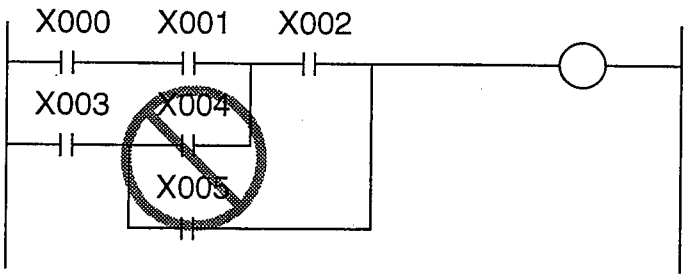



This kind of circuit is not possible on the MX series.

Redraw the circuit as shown below.

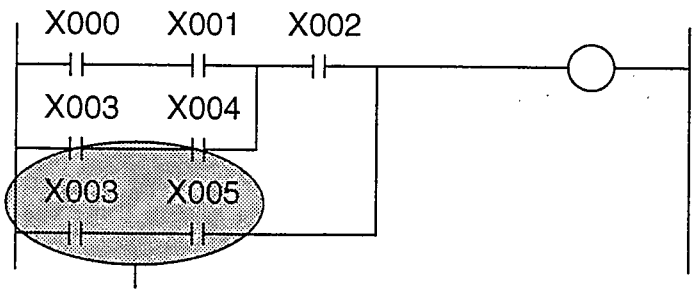



Do not write branch backs into ladder programs.




This kind of circuit is not possible on the MX series.

Redraw the circuit as shown below.

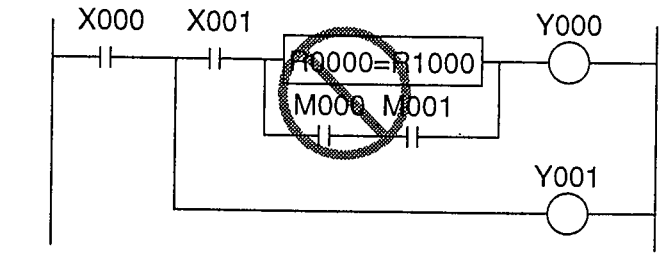



Enter by a branch from the bus.

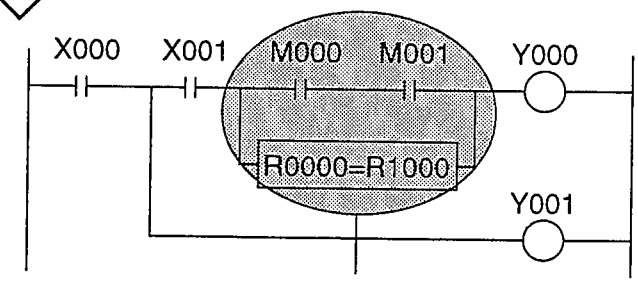
## ● Notes on the use of the compare instruction

Do not write branch back circuits like the one shown below.
Enter the compare instruction in parallel circuits under the ladder block.

(Example)



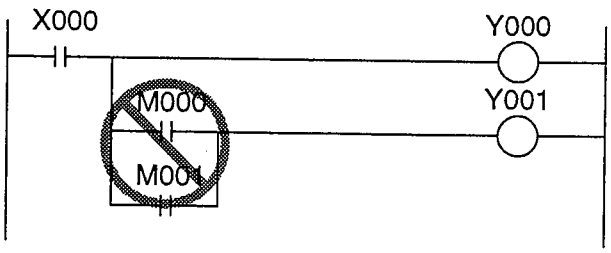

Redraw the circuit as shown below.



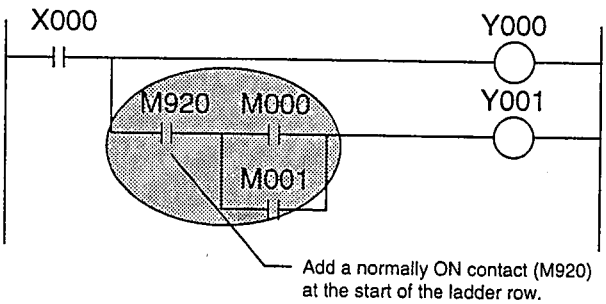

Enter the compare instruction in parallel circuits under the ladder block.

## ● Notes on the use of branches in parallel circuits

Do not write branch back circuits like the one shown below.

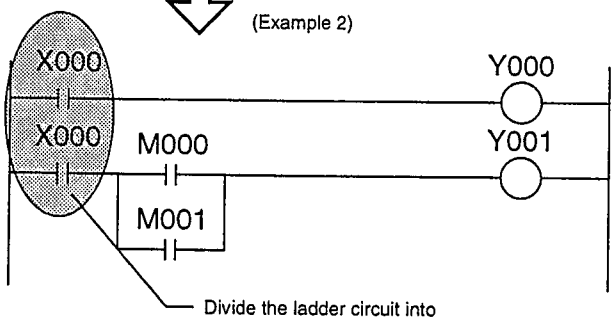



(Example 1)




Add a normally ON contact (M920) at the start of the ladder row.

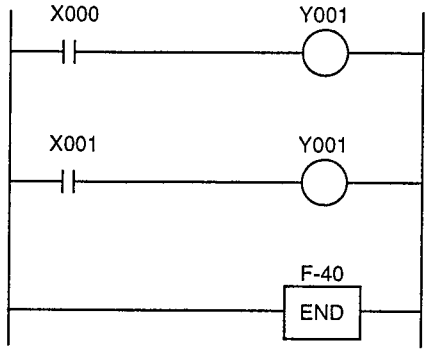
(Example 2)




Divide the ladder circuit into two rows.

## ● Duplicated output

When an OUT instruction with the same number is used twice, the loader displays a syntax alarm. The Y001 state is output when the END instruction is executed. The last written operation has priority.




← This instruction has priority.

Due to the state of the X100, Y001 is output.

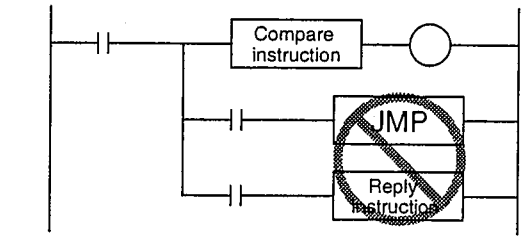
## ● Instructions which require care during changing of modes (STOP → RUN)

The pulse rise instruction is executed when the input condition during the change from STOP to RUN takes place. The instruction is not executed when the input condition is OFF.

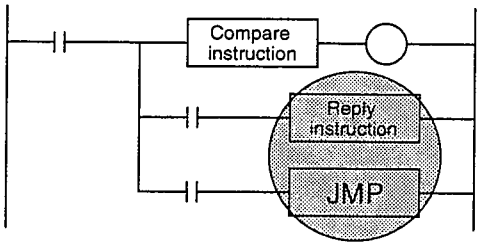
| Instruction \ Condition | Input ON | Input OFF |
|---|---|---|
| CNT<br>STB (F-08)<br>PLS↑ (F-20)<br>UDC (F-26)<br>DCNT (F-27)<br>DUDC (F-28)<br>SR (F-60)<br>Optional pulse instruction | Executed | Not executed |

## ● Notes on the use of branch instructions

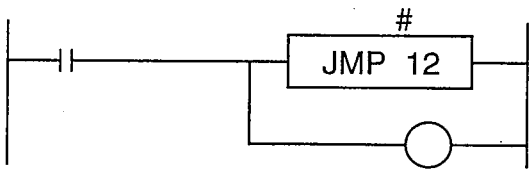
- Insert the JMP instruction after the branch instruction.




Redraw the circuit as shown below.

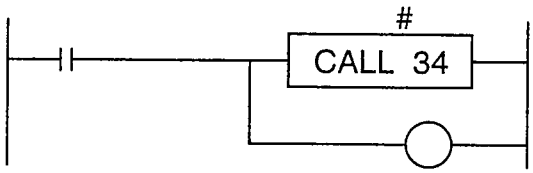



- When several outputs follow a JMP instruction, the JMP instruction is executed, but not the following JMP instruction.



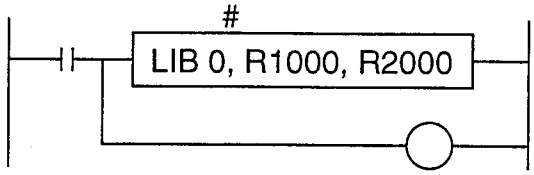

← Not executed when the input contact is ON.

- When several outputs follow a CALL instruction, the execution of the instruction following the CALL instruction is not guaranteed.




← Execution is not guaranteed.

- When several outputs follow a LIB instruction, the execution of the instruction following the LIB instruction is not guaranteed.




← Execution is not guaranteed.

# 3 - 2 Using Function Instructions

## ■ Range of numeric values

The MX series of controllers use instructions that handle both 16-bit and 32-bit numerics. The most significant bit (MSB) of 16-bit or 32-bit BIN data is used for the plus and minus sign bit (1: minus, 0: plus). Hexadecimal constants and BCD constants are all positive and have no sign bit. BIN data is expressed as decimal constants.

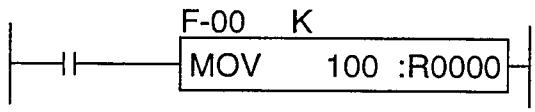
The range of numeric data is shown in the table below.

| | Symbol | 16-bit data | 32-bit data |
|---|---|---|---|
| Decimal constants (stored as BIN values) | K | -32768 to +32767 | -2147483648 to +2147483647 |
| Hexadecimal constants | # | 0000 to FFFF | 00000000 to FFFFFFFF |
| BCD constants | # | 0000 to 9999 | 00000000 to 99999999 |

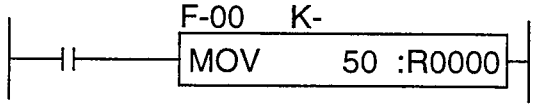
Handling Decimal Constants as Binary Data

Since the MSB of word data is a plus or minus identification bit, the decimal constant K is handled as follows:

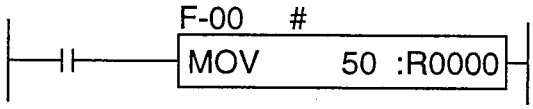

(Example)



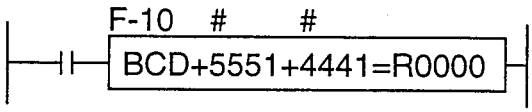

The decimal "100" is stored in register R0000 as a BIN value.



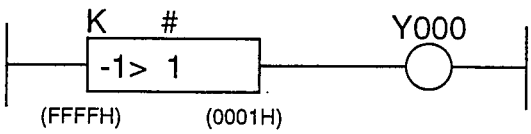

The decimal "-50" is stored in register R0000 as a BIN value.




The hexadecimal "0050" is stored in register R0000 as a hexadecimal value.




9992H, the result of adding BCD values "5551" and "4441", is stored in register R0000.

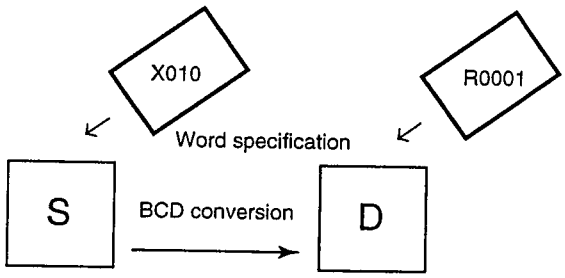



K-1 and #1 are compared. Since signs are ignored in the comparison, Y000 is set to ON.

## ■ Source and destination

- Source (S)
  This is the storage source of operation data. There are four types of sources: 4-bit, 8-bit, 16-bit and 32-bit data. Each source is specified by an instruction word.
- Destination (D)
  This is the storage destination of operation data. There are five types of destinations: 4-bit, 8-bit, 16-bit, 32-bit and 64-bit data. Each destination is specified by an instruction word.
- Bit device (Z)
  This is the storage destination for operation inputs or output bit data.
- Constants (n)
  This is the storage destination for operation constants (set values and numbers.)

MX instruction words store device data in the four data storage destinations.




The same device number can be used for both a source (S) and a destination (D).

## ■ Instruction types

MX instructions consist of an instruction part and a device.
The instruction part indicates the function of the instruction, and the device indicates the data that is used by the instruction. The combination of instruction parts and devices makes it possible to distinguish instructions as shown below.
(See page 3-13 for information on abbreviations such as S, D, Z or n.)

[1] Instruction part (only) ----- This is a type of instruction whose device does not change status.
It is mainly used for program control.
(Example) END and AND

[2] Instruction part + n ---- This type of instruction performs operation specified by constant (n) data.
(Example) MC and CALL

[3] Instruction part + Z ---- This type of instruction is used for turning bit devices ON and OFF.
(Example) AND, OUT and PLS ↑

[4] Instruction part + D ---- This type of instruction performs operations on destination (D) data and stores the result in destination (D).
(Example) BCDI, RST

[5] Instruction part + D + n ---- This type of instruction performs operations on constants (n) and stores the result in bit device (Z).
(Example) TMR and CNT and other timer and count instructions

[6] Instruction part + S + D ---- This type of instruction performs operations on source (S) data and stores the result in destination (D).
(Example) MOV → BCD and other transfer and conversion instructions

[7] Instruction part + S1 + S2 + D ---- This type of instruction performs operations on source 1 (S1) and Source 2 (S2) data and stores the result in destination (D).
(Example) Compare, BCD + and other compare and operation instructions

[8] Other instruction types

Instruction part + S + D1 + D2 ---- (Example) DMPX and other distribution and extraction instructions

Instruction part + n + D1 + D2 ---- (Example) SFL and other shift instructions

etc.

## ■ Specifying pulse

Function instructions are used for specifying pulses.
When the execution condition is in the input ON mode, some instructions are used to specify a "rise" (referred to as "pulse setting") for the execution status. Pulses are specified by inserting ↑ in the instruction sheet.
See User's Manual "Programming Instruction Words" CP-UM-1563E for information.

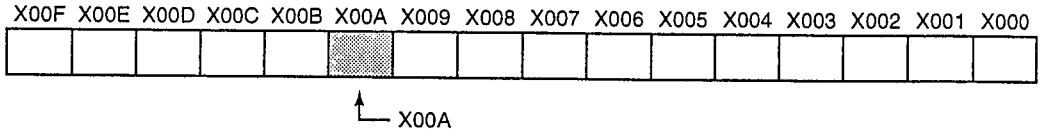
## ■ Specifying data

Data is specified using the source (S) and destination (D) of MX instructions as shown below.

### (1) Specifying I/O area (X, Y) and internal contact area (L, M)

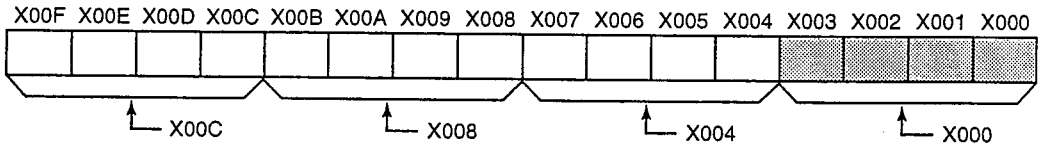
① Enter the relevant device number to specify a bit.

(Example)



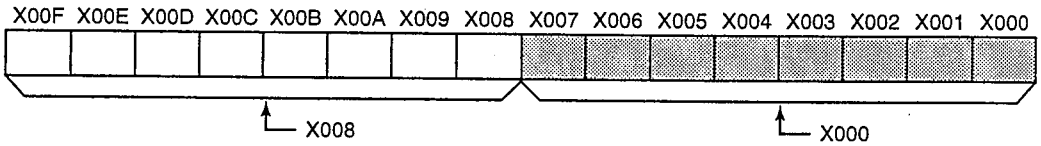

② Enter the first device number whose least significant digit is 0, 4, 8 or C to specify four bits.

(Example)



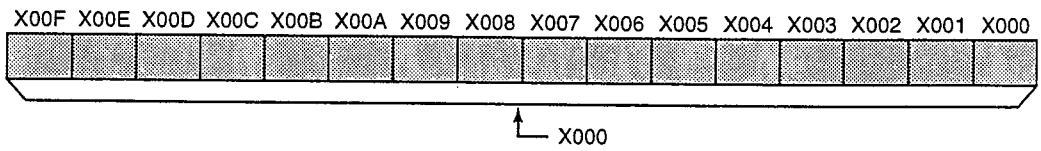

③ Enter the first device number whose least significant digit is 0 or 8 to specify eight bits.

(Example)



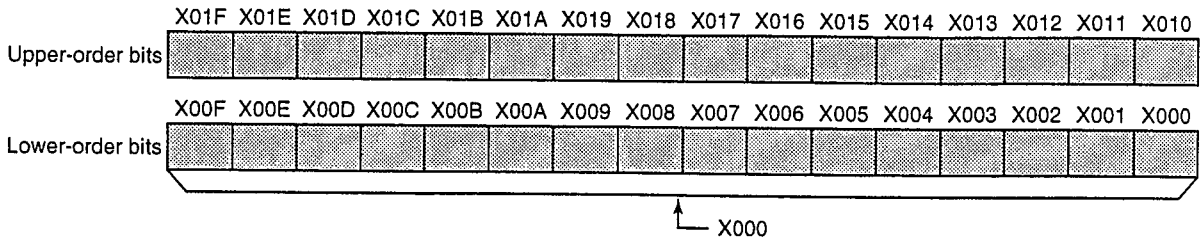

④ Enter the first device number whose least significant digit is 0 to specify 16 bits.

(Example)




⑤ Enter the first device number whose least significant digit is 0 to specify 32 bits double-length.

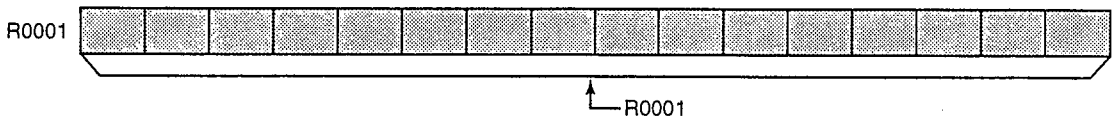
(Example)



### (2) Specifying register area (R)

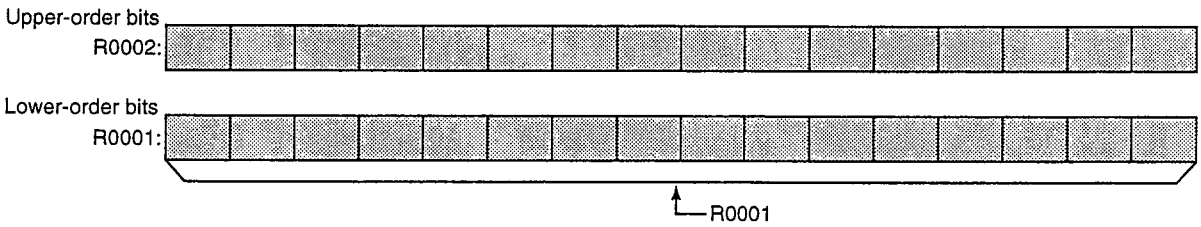
① Enter the relevant device number to specify 16 bits.

(Example)




② Enter the lower-order 16-bit device number to specify 32 bits.

(Example)




(Note) The least significant bit may be even or odd.

## ! Handling Precautions

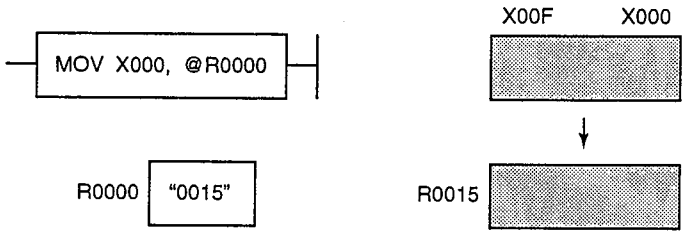
Data in the register area is generally handled in 16-bit units. However, as an exception, when a register area is specified in S or D, 4-bit data transfers (M4BT) or 8-bit data transfers (M8BT) may be handled as lower-order 4-bits or lower-order 8-bits.

## ■ Indirect addressing

Some instructions allow the use of the BCD value in register R as an indirect address in an MX application instruction instead of addressing register R itself. This can be achieved by prefixing register R with the "@" symbol in addressing with the loader.

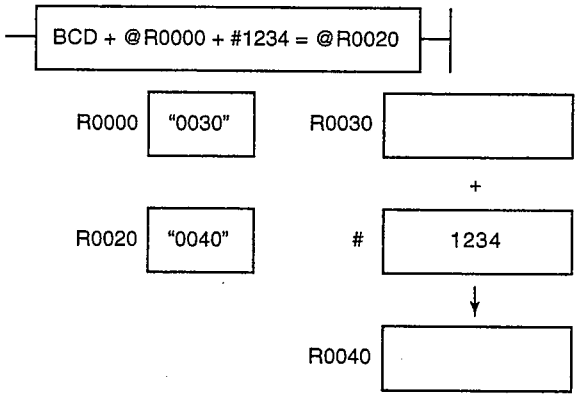
**(Example) MOV (data transfer) instruction**

When S: X000, D: @R0000, the X000 to X00F 16-bit data item is not moved to register R0000. Instead, if the R0000 value is a BCD value "0015", it is moved to register R0015.




**(Example) BCD + (BCD addition) instruction**

When S1@R0000, S2#1234, D: @R0020, the R0000 value and #1234 are not added directly. If the R0000 value is BCD value "0030" and the value in R0020 is "0040", the R0030 value and #1234 are added and the result is stored in R0040.




## ! Handling Precautions

- As shown in the above examples, indirect addressing can be used for several (2 or 3) registers of the same instruction.
- When the value in register R prefixed with "@" to indicate indirect accessing does not contain a BCD value or when the range of register R is exceeded, an operational error flag (M90E or M90F) is set.
  See the User's Manual "Programming Instruction Words" CP-UM-1563E for information on which instruction words allow indirect addressing.

## ■ Flags

The MX series of controllers have the following five type of flags:

1. Shift carry flag (M900)
2. Overflow flag (M901)
3. Underflow flag (M902)
4. Character string compare flag (M903) (MX200 only)
5. Operational error flag (M90E, M90F)

### ● Flag handling

- Flag operation
  Once a flag is set, it stays ON until an instruction that affects its operation is executed. However, the operational error latch type (M90F) stays ON when set to ON.
- Flag holding
  As stated earlier, flags are turned ON or OFF during scanning. So, in order to monitor the state using a loader, apply self-holder or breakpoints after the instruction that affects the flag. (See Chapter 4, "Debugging" for information on breakpoints.)

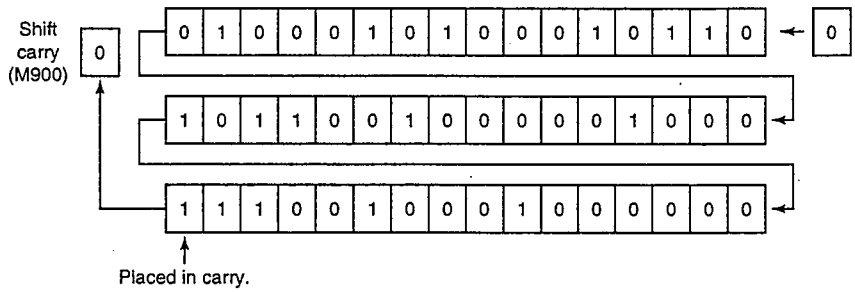
### ● The shift carry flag

The shift carry flag (M900) is used in the following cases.
Instruction words:

- SR (shift in both directions)
- SFL (left shift)
- SFR (right shift)
- RLC (leftward rotation including carry)
- RRC (rightward rotation including carry)
- RL (leftward rotation)
- RR (rightward rotation)

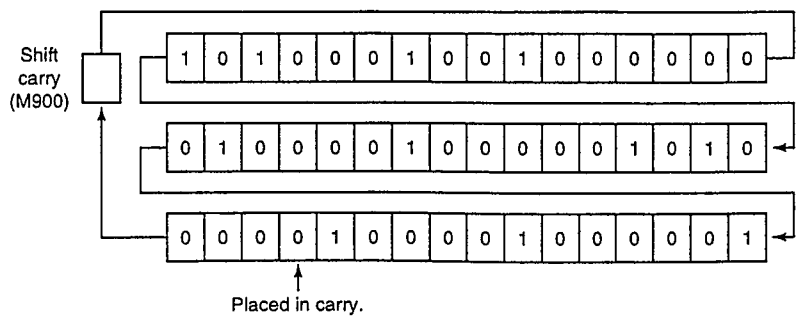
**(Example) SFL**

As shown below, when one bit is shifted leftwards, "1" is stored in the carry (M900).




**(Example) RLC**

As shown below, when four bits are rotated leftwards, "0" is stored in the carry (M900).



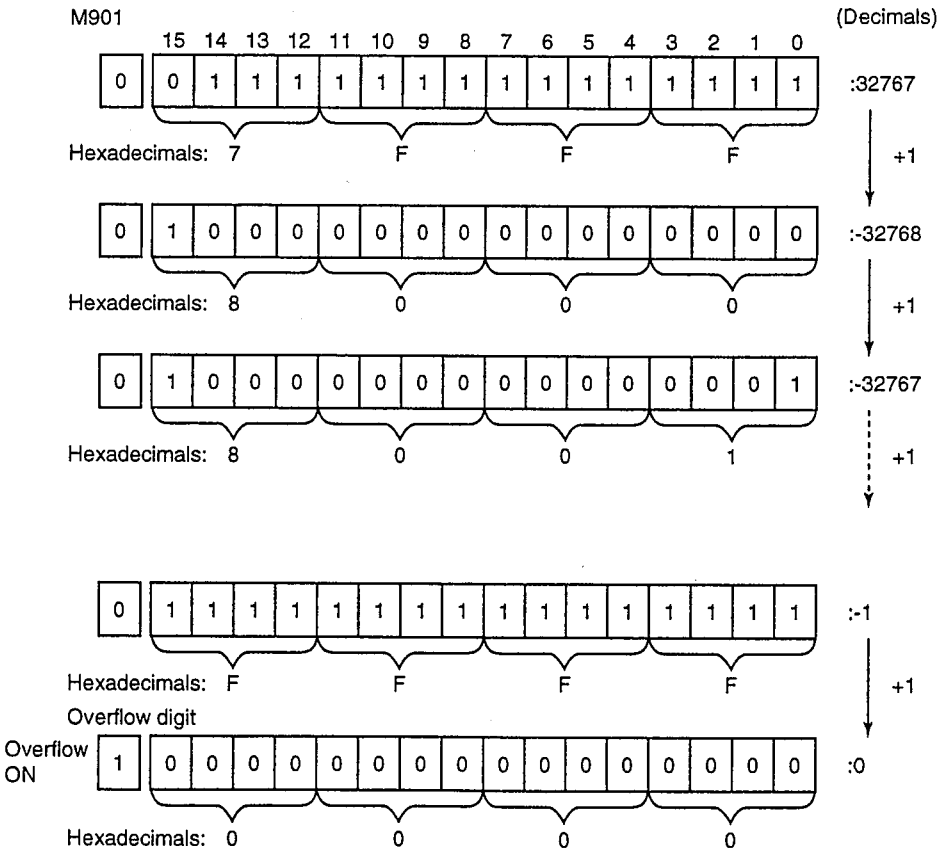
● **Overflow and underflow**

An overflow flag (M901) or an underflow flag (M902) is set in the following cases:

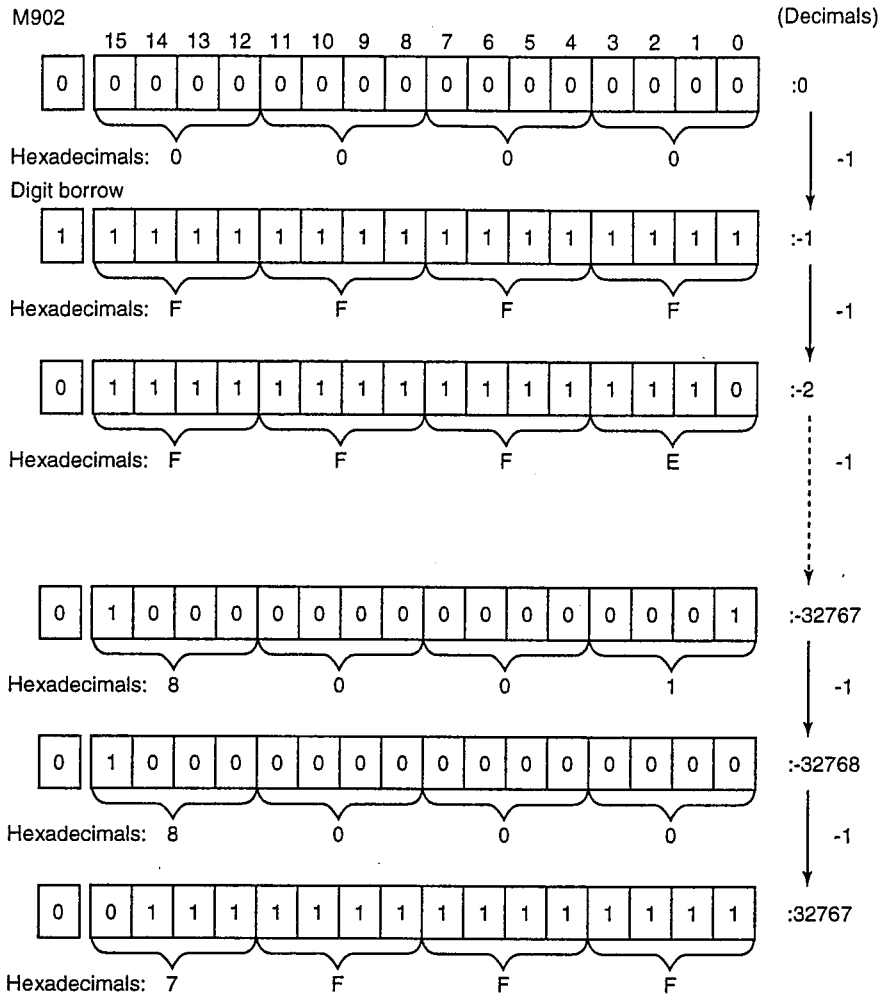
| | Instruction Words | When flag is set |
|---|---|---|
| Overflow flag | BIN addition/subtraction | When result exceeds #FFFF (see note) |
| | Double-length BIN addition/subtraction | When result exceeds #FFFFFFFF |
| | BIN increment | When result exceeds #FFFF |
| | BCD addition | When result exceeds 9999 |
| | Double-length BCD addition | When result exceeds 99999999 |
| | BCD increment | When result exceeds 9999 |
| Underflow flag | BIN addition/subtraction | When result is lower than #0000 |
| | Double-length BIN addition/subtraction | When result is lower than #00000000 |
| | BIN decrement | When result is lower than #0000 |
| | BCD subtraction | When result is negative |
| | Double-length BCD subtraction | When result is negative |
| | BCD decrement | When result is negative |

## ! Handling Precautions

As shown below, an overflow occurs when there are too many digits. BIN values has a minus sign bit, so the numeric range (32767) it can handle is not an overflow.



The same is true for underflow conditions.




See the User's Manual "Programming Instruction Words" CP-UM-1563E for information on character string compare result flags.

● **Operational error flags**

The table below shows examples when error flags (M90E: unlatched flags and M90F: latched flags) are set.

| | Operational error | Example |
|---|---|---|
| 1 | No BCD data can be found. | • The operand of a BCD operation is not a BCD value.<br>• The offset data of a data distribution instruction or extraction instruction is not a BCD value.<br>• The write data of a timer/counter set value write or current value write instruction is not a BCD value.<br>• The table specified value of a data table read instruction is not a BCD value.<br>• The conversion object of a double-length BCD to BIN conversion is not a BCD value. |
| 2 | A digit-over condition results when data is converted to BCD values. | • The result of a BIN to BCD conversion instruction exceeds 9999.<br>• The result of a double-length BIN to BCD conversion instruction exceeds 99999999. |
| 3 | Division by 0 executed. | • BIN division or double-length BIN division by 0 is executed.<br>• BCD division or double-length BCD division by 0 is executed. |
| 4 | The offset of the distribution destination, extraction source or storage destination data is too great. | • The address of distribution destination D1 + (D2) of data instruction exceeds the area to which D1 belongs.<br>• The address of extraction source S1 + (S2) of data instruction exceeds the area to which S1 belongs.<br>• The data table read instruction was specified for an area where there was no table.<br>• The storage destination D + (S2) of data table read instruction exceeds the area to which D belongs, and S2 is not a constant (#). |
| 5 | Although 1 is to be raised, there is no such rise. | • A 16 to 4 encode instruction is used on an operand that is all 0's. |
| 6 | A value other than a BCD value or a value outside the address range is stored in the register when an indirect instruction is specified. | • #FFFF is transferred to the register by MOV. |
| 7 | A value other than a character string was specified in the process of a character string processing instruction | • The specified data string exceeds 256.<br>• The result of the character string process exceeds 256. |

## ! Handling Precautions

- When several operational errors occur, check the special registers (R0904 to R0907) to locate the step where the error occurred.
- Latch type (M90F) operational errors are reset with a table change or forced set/reset from the loader.

## ● Syntax errors

The conditions listed in the table below cause a syntax error. This causes a syntax error (M99A) to be set. A syntax error message is also displayed on the loader. Such an error is detected during the program check performed at powering up or during the loader program check.

| Code | Error | Cause of Error |
|---|---|---|
| 02 | Parameter error | Incorrect parameter value (device No., constant #, constant K) |
| 03 | Device area exceeded | The device No. value exceeds the defined area.<br>(Example) X000 to X09F → X100<br>X200 to X29F → X11F<br>(Correct) (Wrong) |
| 04 | Illegal constant # | ① The value of constant # exceeds the specified number of digits.<br>② A decimal point (timer values are excluded) has been set.<br>③ Numeric values A to F, a to f are used. |
| 05 | Illegal constant K | ① Symbols other than "-", "+" and "0 to 9" are used in a K value.<br>② The value of a double-length instruction -2747483648 ≤ to ≤ 2147483647.<br>③ The value of K (other than double-length instruction) -32768 ≤ to ≤ 32767. |
| 06 | Same area error | The parameter value is not the same area.<br>① The D1 and D2 values of DCPY (FUN07), SR (FUN60), SFL (FUN61), SFR (FUN62), RCL (FUN63), RRC (FUN64), RL (FUN65) and RR (FUN66) are in the same area, and D1 must be equal to or less than D2.<br>● General-purpose Y output (Y00 to Y090) and OP I/O output (Y200 to Y290) are in different areas. Register R (R0000 to R0499) and output register (R0600 to R0619) are also in different areas.<br>② S1 and S2 of BMOV (01) are in the same area, and S1 must be equal to or less than S2.<br>● General-purpose X output (X00 to X090) and OP I/O input (X200 to X290) are in different areas. General-purpose Y input and OP I/O input are in different areas, too.<br>● Register R (R0000 to R0499) and input register (R0600 to R0619) are in different areas.<br>● Only register (R0000 to R0499) can use indirect addressing which is assumed to be in the same area as register (R0000 to R0499). When indirect addressing is used for S1 and S2, an S1 ≤ S2 check is not performed. |
| 07 | Parameter area exceeded | Parameter values (device No., constant #, constant K) exceed the set area.<br>① BMOV (FUN01) is exceeded<br>The number of D words exceed the S2 - S1 + 1 word value of device set in D.<br>(Example) BMOV S1: X000 9 - 0 + 1 = 10 words<br>(FUN01) S2: X090<br>D: R498498 + 9 > 499 (error)<br>② Words (n + 1)/2 + 1 (all decimals are rounded off) were added to a device specified in STB (FUN08) area over S and the area was exceeded.<br>(Example) STB n: #11 → (11 - 1)/2 = 6 words<br>(FUN08) S: X080<br>D: Y000 → 8 + 5 = 13 > 9 (error) |
| 12 | Multiple defective outputs | Several output instructions (OUT, MOV, CALL, etc.) were issued after a CALL and LIB instruction. (This occurs because the line logic after a CALL instruction is the value of the corresponding RET value. The line logic after execution of a LIB instruction becomes undefined depending on the content of the library.) |
| 20 | Undefined instruction | An undefined instruction was found in the program. |

| Code | Error | Cause of Error |
|---|---|---|
| 21 | Circuit error | The following errors occurred on the circuit block.<br>① The total of ANB, ORB plus 1 does not match the total of LD, LDNOT, LD compare and DLD compare.<br>② The total of LD, LDNOT, LD compare and DLD compare exceeds 8.<br>● In internal processing of LD, LDNOT, LD compare and DLD compare, the internal count is +1; with ANB and ORB the internal count is -1. When the internal count is more than 9 or less than 0, an error has occurred.<br>③ There is an ORB or ANB instruction following an output instruction.<br>(Note) Output instructions: OUT, TMR, CNT, INV (FUN22), LPS (FUN34), LRD (FUN35) and LPP (FUN36)<br>④ There is an output instruction immediately after JPE (FUN33), RBP (FUN41), SBR (FUN43) and RET (FUN44).<br>However, an output instruction is allowed immediately after MCR (FUN31), JPE (FUN33), RBP (FUN41), BSR (FUN43), RET (FUN44) and END (FUN40)<br>⑤ There is no LD instruction at all in front of an output instruction.<br>However, this is possible with MCR (FUN31), JPE (FUN33), RBP (FUN41), BSR (FUN43), RET (FUN44) and END (FUN40). |
| 22 | CALL, SBR, RET error | ① There is no SBR (FUN43) having the No. called by CALL (FUN42).<br>② There is no RET (FUN44) having the same No. as an SBR (FUN43).<br>③ There are CALL (FUN42) instructions having the same Nos. as SBR (FUN43) and RET (FUN44).<br>④ There is no RBP (FUN41) or RET (FUN44) immediately before SBR (FUN43). |
| 23 | MC, MCR error | ① There is no MC (FUN30) No. corresponding to MCR (FUN31).<br>② There is no MCR (FUN31) having a No. of #n or less between MC (FUN30) #n and END (FUN40). |
| 24 | JMP, JPE error | There is no JPE (FUN33) with the same No. as JMP (FUN32). |
| 25 | Multi-input instruction error | ① The total number of LD, LDNOT, LD compare and DLD compare output instructions preceding a multi-input instruction (see note 1) does not match the total number of ANB and ORB instructions. This condition is the cause of the error.<br>2 inputs: The total number of LD, LDNOT and other instructions must equal 2 + (the total number of ANB and ORB instructions).<br>3 inputs: The total number of LD, LDNOT and other instructions must equal 3 + (the total number of ANB and ORB instructions).<br>4 inputs: The total number of LD, LDNOT and other instructions must equal 4 + (the total number of ANB and ORB instructions).<br>(Note 1) Multi-input instructions are instructions that are used for entering inputs other than execution conditions such as counter and shift.<br>2 inputs: CNT, STM (FUN25)<br>3 inputs: UDC (FUN26), DUDC (FUN28)<br>4 inputs: SR (FUN40)<br>② There is an output instruction following the multi-input instruction. |
| 26 | STOP instruction error | There is no LD or LDNOT input immediately before the STOP (FUN94) instruction. |
| 27 | LPS, LRD, LPP error | ① LPS (FUN34) count is not equal to LPP (FUN36) count in a block.<br>② There is no output instruction before an LRD (FUN35) or LPP (FUN36).<br>③ The LPS (FUN34) count in a block exceeds 32. |
| 30 | Timer and counter using the same number | The same No. is used by the timer and counter instructions.<br>Timer instruction: TMR, STM (FUN25)<br>Counter instruction: CNT and UDC (FUN26), DCNT (FUN27), DUDC (FUN28)<br>(Note 2) DCNT (FUN27) and DUDC (FUN28) occupy specified devices and specified devices plus 1. |
| 31 | JPE duplicated | There are two or more JPEs (FUN33) having the same No. |
| 32 | SBR duplicated | There are two or more SBRs (FUN43) having the same No. |

| Code | Error | Cause of Error |
|---|---|---|
| 33 | Pulse specification exceeded | More than 2000 pulse instructions are used.<br>Pulse instructions<br>● PLS[↑] (FUN20) ● PLS[↓] (FUN21)<br>● STB (FUN08) ● SR (FUN60)<br>● Function instructions using pulse options |
| 34 | DTBL instruction (MX100, MX30) | The instruction immediately preceding the DTBL instruction (FUN98) is not an END instruction (FUN40).<br>(An error is also caused if it is preceded by a NOT instruction.)<br>The set value n (number of data items) does not meet the following requirement:<br>2n ≤ (total byte count) - (used byte count including this instruction + 2) |
| 35 | RTBL instruction error | ① Although the RTBL instruction (FUN99) is used, the DTBL instruction (FUN98) has not been written.<br>② The DTBL instruction caused an error.<br>③ When the S1 constant is used, the set count in the DTBL instruction (FUN98) does not meet the following requirement: n ≥ S1 + 1<br>④ When the S1 and S2 constants are used, the set count in the DTBL instruction (FUN98) does not meet the following requirement: n ≥ S1 + S2 |

## Handling Precautions

- Multi-input instructions are instructions such as counter and shift that are used for entering inputs other than execution conditions.
  2 inputs: CNT, STM (FUN25)
  3 inputs: UDC (FUN26), DUDC (FUN28)
  4 inputs: SR (FUN40)
- When two or more operational errors occur, special registers (R0904 → R0907) store data identifying which step the error occurred.
- When a latch type (M90F) is reset to clear an operational error by changing operation modes (→ STOP → RUN) or loader operation (table operation or forced set/reset).

The following conditions do not produce an error but an alarm. When they occur, the program cannot be transferred to an MX controller or to disk.

| Code | Error | Cause of Error |
|---|---|---|
| 40 | Duplicated coil used | Y, M and L having the same number are used in the output instruction (OUT), PLS[↑] (FUN20), PLF[↓] (FUN21), SET (FUN23) and RST (FUN24) |
| 41 | No JMP to JPE | There is no JMP (FUN32) having the same number as in JPE (FUN33). |
| 42 | No CALL to SBR (MX100, MX30) | There is no CALL instruction (FUN42) having the same number as in SBR (FUN43). |
| 43 | No END instruction (MX100, MX30) | There is no END instruction (FUN40) in the program. |

廃番

# Chapter 4 DEBUGGING FUNCTION

## 4 - 1 Outline

Programs often do not run as intended when they have been written. This is often caused by an incorrect understanding or the setting of instruction words, errors in programming sequence and other errors that are referred to as "bugs". "Debugging" is the process of removing these bugs to enable the program to run smoothly.

The MX series of controllers are provided with the following functions to allow efficient debugging:

- Functions for observing internal data
- Functions for manipulating the program execution process
- Functions for directly changing internal data from outside

### ● Functions for observing internal data

- Hot-line monitor
- Multi-point monitor

  These functions are ideal for monitoring ON and OFF conditions of "contacts", "coils" and other relay image devices. These monitors allow you to execute the program from the first step to the END instruction as the internal data is read out, and to check the result of execution.
- Time chart display

  This is a chart displaying time series changes of bit data.

### ● Functions for directly changing internal data from outside

- Data changes
- Forced setting/forced resetting

  These functions allow you to change the data in the user program from outside without having to operate the program itself which greatly facilitates the debugging process.

### ● Functions for manipulating the program execution

- Breakpoints
- Step execution

  You must be able to monitor the result produced after the execution of an instruction in programs that repeatedly operate on data in several data registers to be able to judge whether the program is operating normally. To make this possible, the program is executed in separate steps which allow you to monitor the result produced by execution.

# 4 - 2 Forced Setting, Resetting and Data Changing

## ■ Forced setting, resetting and data changing

The MX series of controllers are provided with the following two ways of changing a device table. Both operations are performed on the loader.

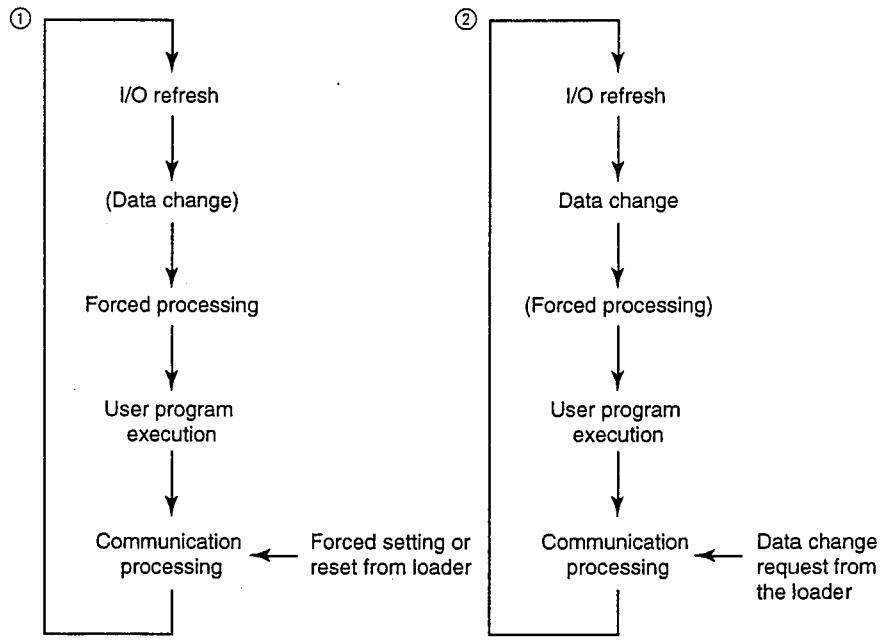
- Forced setting and resetting
- Data change

### ● Forced setting and resetting

The states of the device table can be forcibly set or reset before the user program is executed. For example, if a forced Y001 setting is made, the program is set to ON before the execution of the next scan program even if the user program is OFF.

### ● Data changes

Like forced setting and resetting, the states of the device table can be forcibly changed. For example, even if Y001 is set to ON as a result of a table change, the program stops at the next scan if an OFF condition is set in the user program.



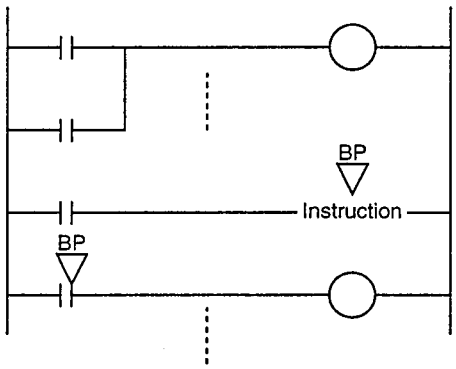

## ! Handling Precautions

- Forced setting, resetting and table changes can be set in the device table before execution. As a result, the result of the user program may not match the result produced by the forced result.
- Communication processing and table changes are processed off-line. (See figure above.)

# 4 - 3 Breakpoints

Breakpoints allow you to interrupt (break) a program when execution reaches a preset instruction step (the breakpoint), and monitor the result of execution produced at that point. This speeds up debugging of programs that perform a great number of data operations. A total of 16 breakpoints can be inserted in a program.
The conditions set at each breakpoint to stop the program can be set separately. The table on the following page describes break conditions.

(Example)
When a program does not operate as intended, the instruction words that are used to operate a device are called up and their changes are traced using the following procedure.




① Set the controller to the TEST mode.
② Insert a breakpoint before the instruction word whose operation is to be checked.
③ Make a monitor setting for the target device.
④ Set the execution (RUN) mode and specify start.
⑤ Execution stops when the program reaches the instruction with the breakpoint. The device data at this time are the values immediately before execution of the instruction.
⑥ Restart execution by specifying continue.
⑦ The program now stops (breaks) immediately after executing the instruction with the breakpoint. The device data at this point is the result of executing the instruction.
⑧ Check the data immediately before and after execution of the instruction to make sure that the intended process is taking place.

This procedure allows to check operation of a selected part of a program that runs erratically and identify the bugs (program errors) in them.

## ● Break types

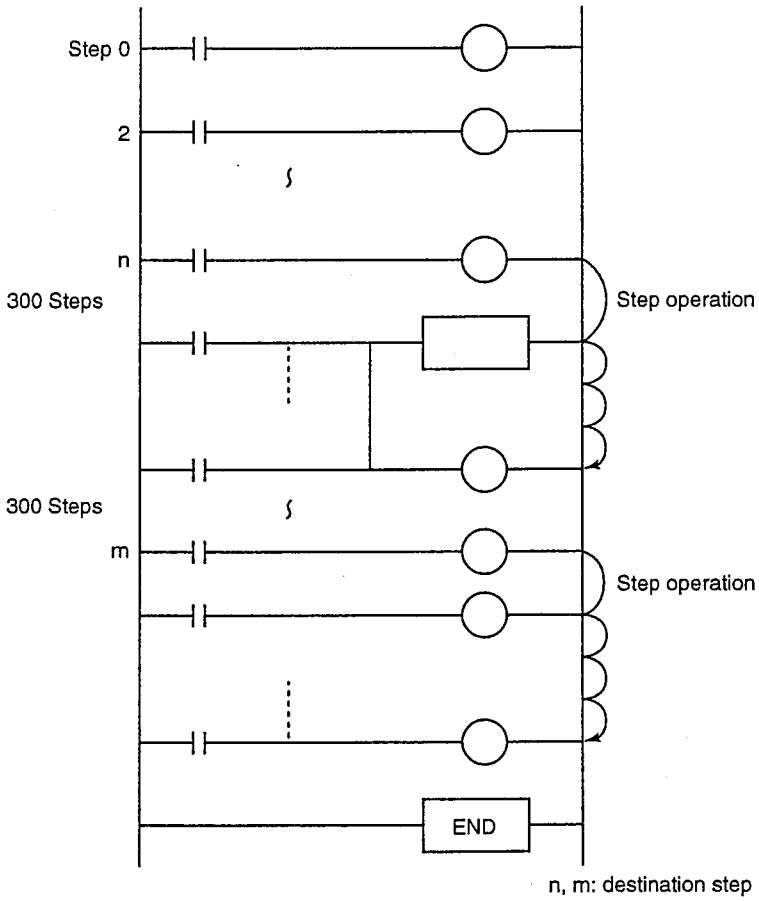
The table below lists the nine break conditions. Set one of the following conditions.

| | Break condition | Description |
|---|---|---|
| ① | Normal break processing | User program execution is stopped at positions where breakpoints have been inserted. |
| ② | Line logic ON | When the user program reaches the breakpoint, the line logic turns ON and break processing starts. |
| ③ | Line logic OFF | When the user program reaches the breakpoint, the line logic turns OFF and break processing starts. |
| ④ | Line logic OFF → ON | When the user program reaches the breakpoint, the line logic turns from OFF to ON and break processing starts. |
| ⑤ | Line logic ON → OFF | When the user program reaches the breakpoint, the line logic turns from ON to OFF and break processing starts. |
| ⑥ | Operational error 1 (M90E) is ON | When the user program reaches the breakpoint, break processing starts if operation error 1 (M90E) is ON. |
| ⑦ | Shift carry (M900) is ON | When the user program reaches the breakpoint, break processing starts if shift carry (M900) is ON. |
| ⑧ | Overflow flag (M901) is ON | When the user program reaches the breakpoint, break processing starts if the overflow flag (M901) is ON. |
| ⑨ | Underflow flag (M902) is ON | When the user program reaches the breakpoint, break processing starts if the underflow flag (M902) is ON. |

# 4 - 4 Step Execution

This function is used to execute a program in instruction steps.
Step execution is used after program execution is stopped at a breakpoint, and enables detailed analysis of operation results.
Step execution allows 300 steps to be set at two locations or a total of 600 steps.

Step execution is performed as follows:




① Set the controller to the TEST mode.
② Specify step execution at the start of a program step range you want to check.
③ Make monitor setting for device that is to be observed.
④ Set step execution to "enable".
⑤ Set the controller to the execution (RUN) mode and specify start.
⑥ Execution stops (breaks) at the start of step where step execution was set. The device data at this point are the values of the instruction indicated by the cursor before execution.
⑦ Restart execution by specifying continue.
⑧ The instruction word indicated by the cursor is executed and then execution stops. The cursor now indicates the next instruction to be executed. The device data at this point is the result of execution of the instruction in step ⑥.
⑨ Steps ⑦ and ⑧ are repeated.

This procedure allows to check operation of a selected range part of a program that runs erratically and identify the bugs (program errors) in them.

## Handling Precautions

Care is required in inserting breakpoints and performing step execution.

- No outputs are updated to the output module during break operation. An update is made when the END instruction is executed.
- Inputs from input modules are accepted during break processing.

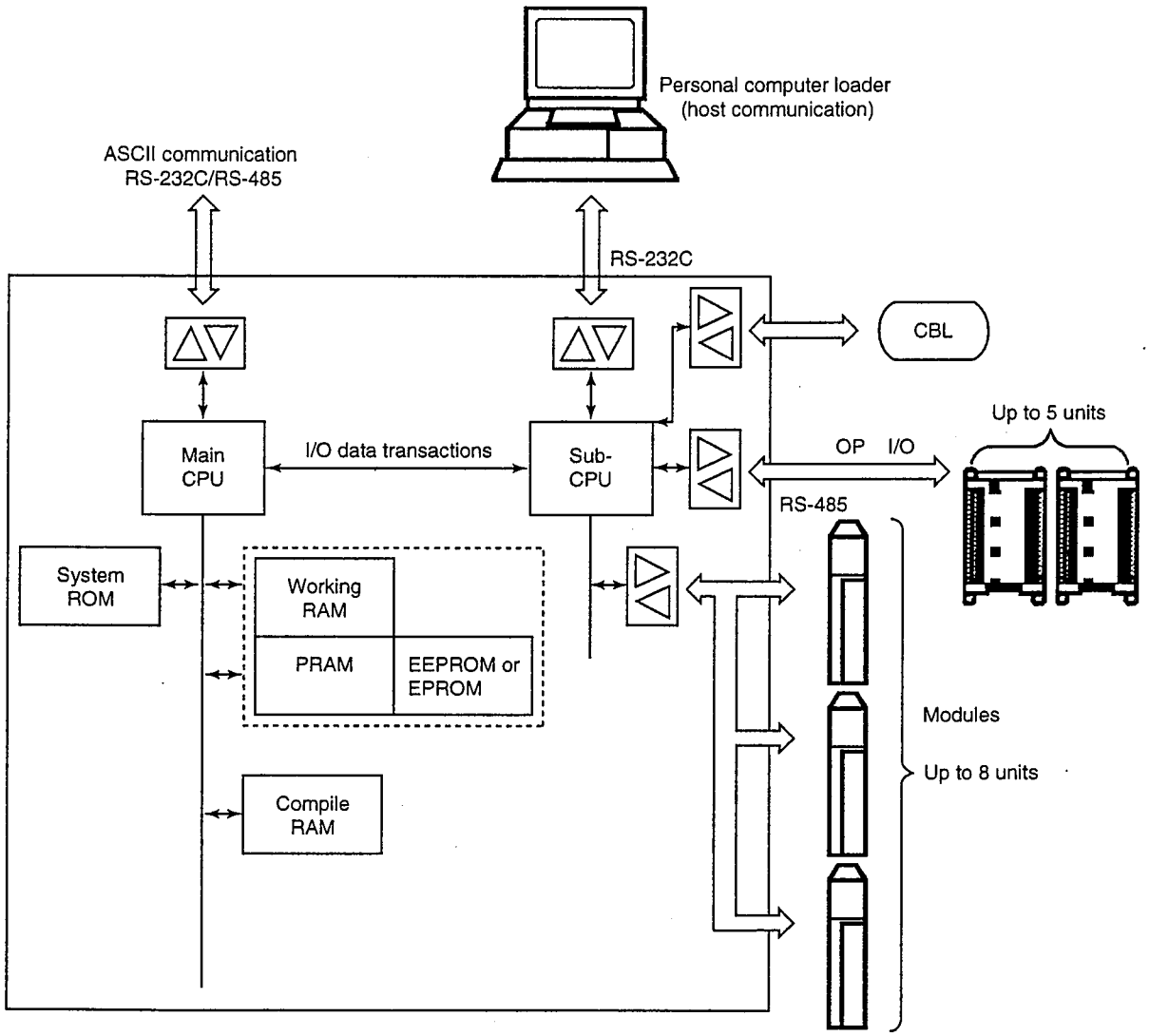
廃 番

廃番

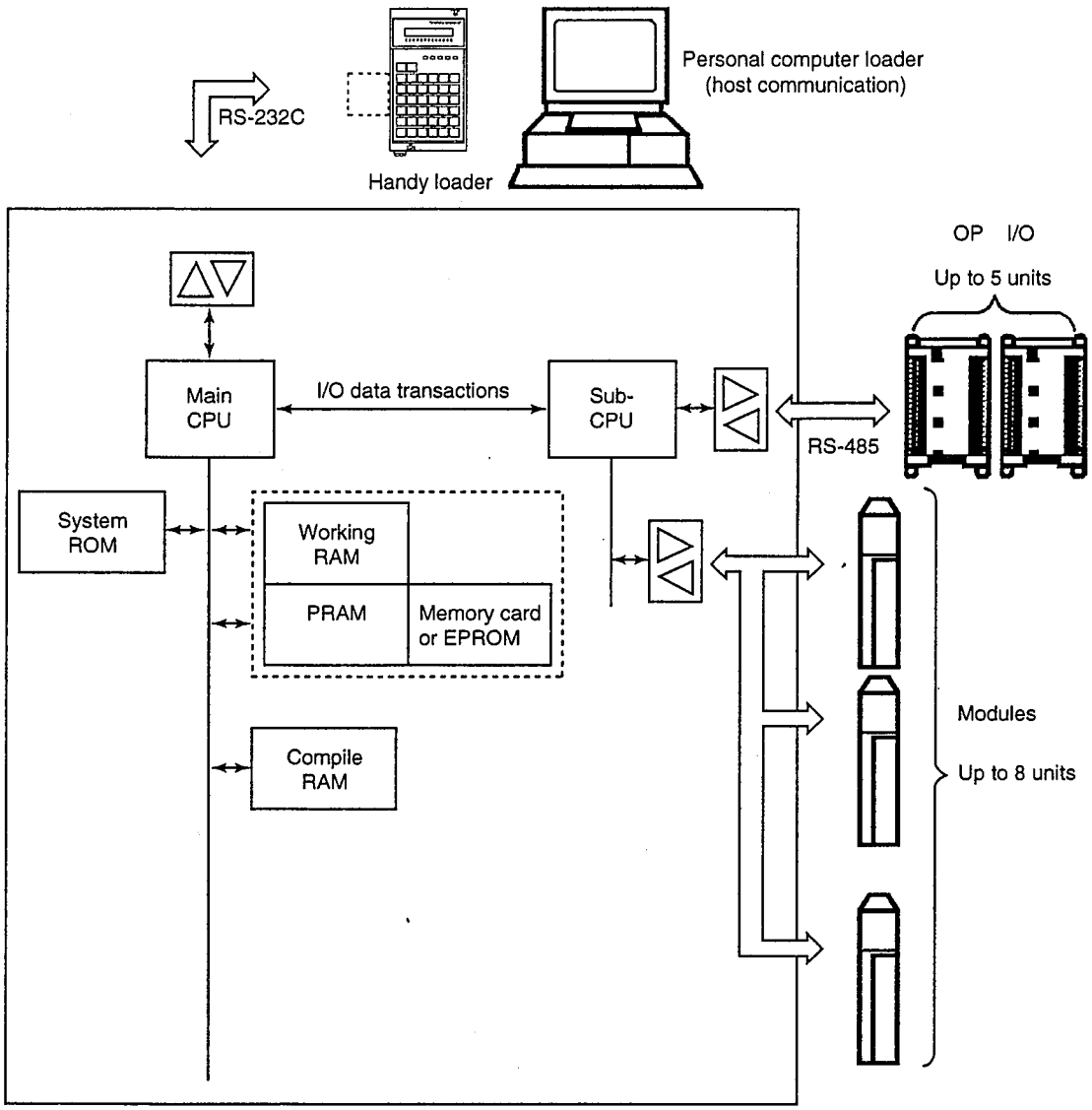
# Chapter 5 CONFIGURING AND OPERATING PROCESSOR MODULES

## 5 - 1 Processor Module Configuration

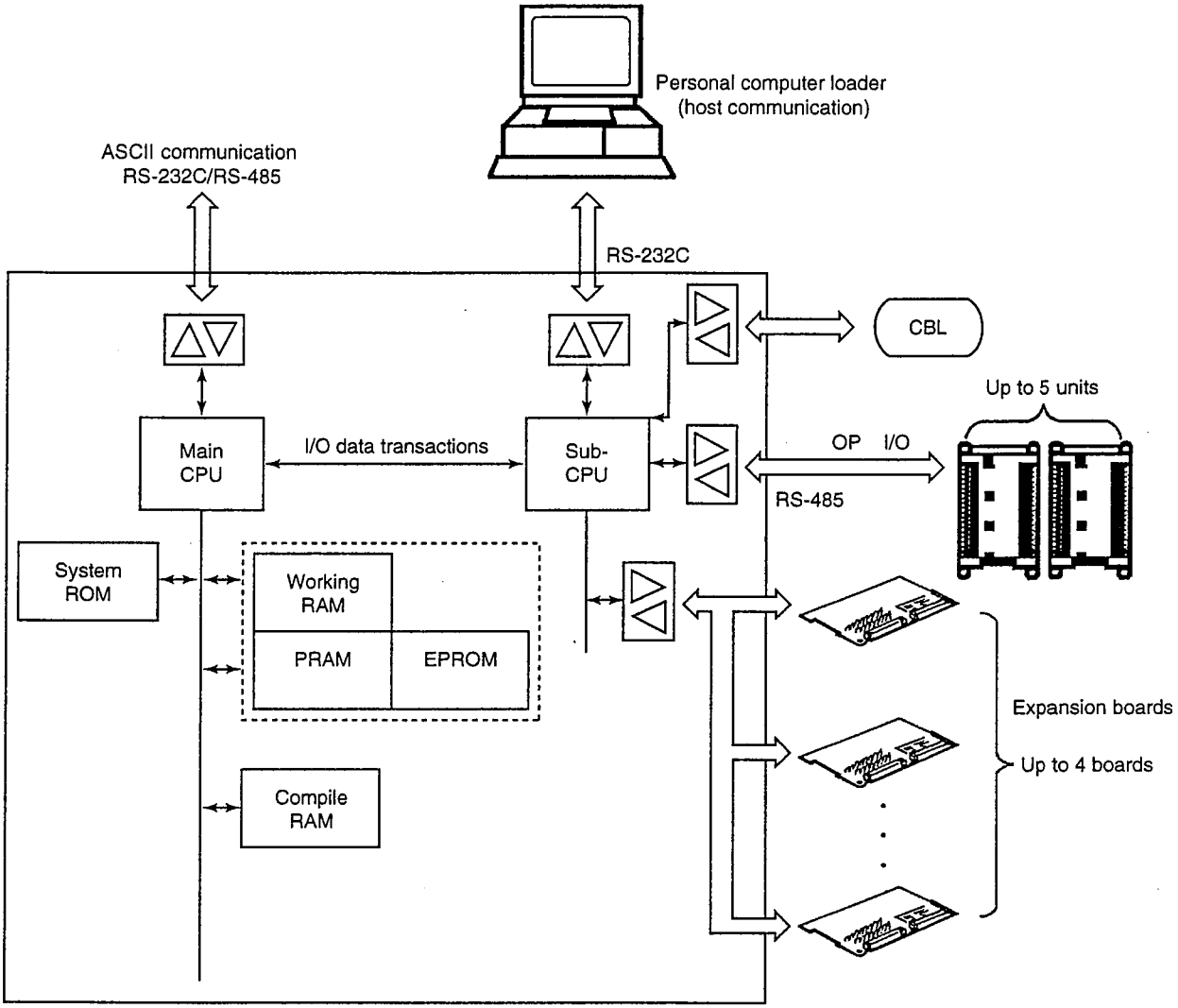
### ■ MX200



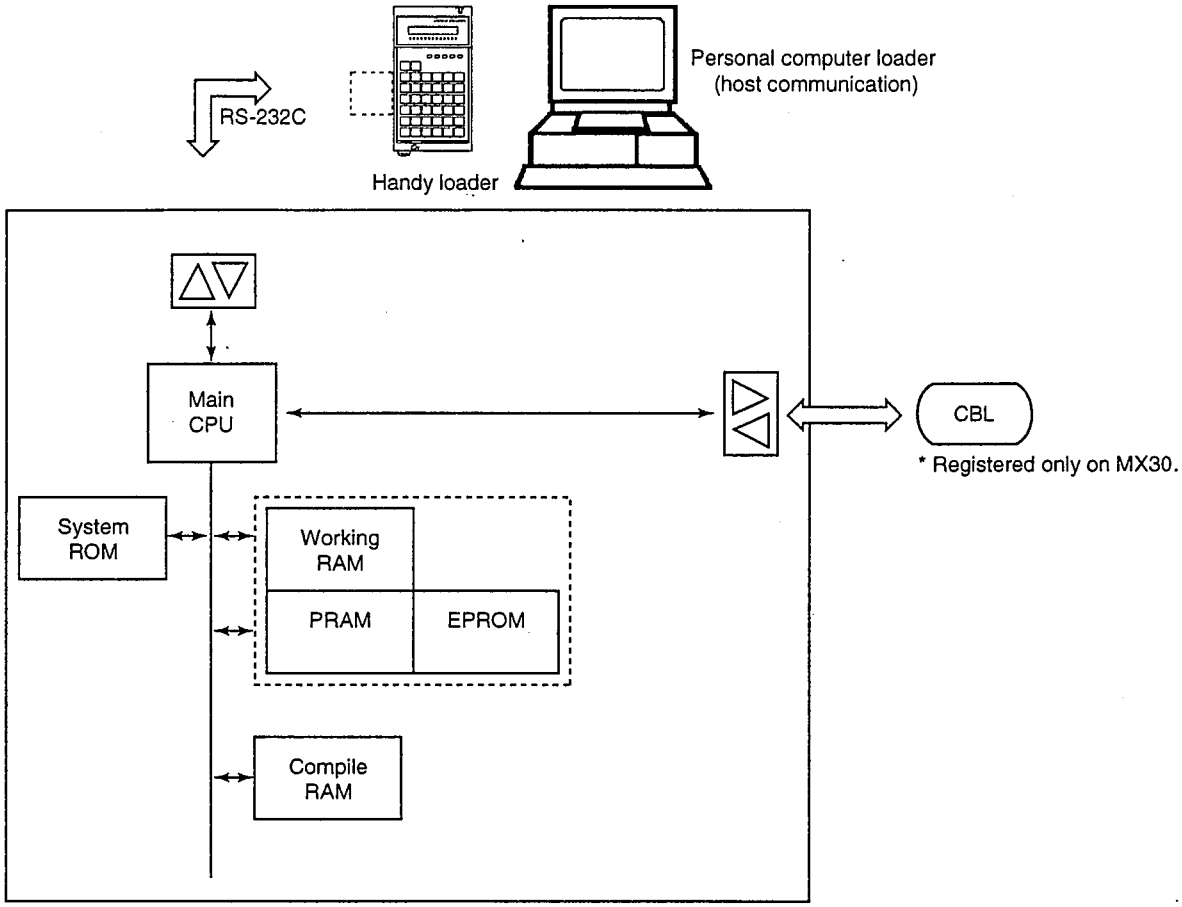
## ■ MX100


RS-232C
Handy loader
Personal computer loader
(host communication)
OP I/O
Up to 5 units
Main CPU
I/O data transactions
Sub-CPU
RS-485
System ROM
Working RAM
PRAM
Memory card or EPROM
Compile RAM
Modules
Up to 8 units

## ■ MX50


Personal computer loader
(host communication)
ASCII communication
RS-232C/RS-485
RS-232C
CBL
Up to 5 units
Main
CPU
I/O data transactions
Sub-
CPU
OP I/O
RS-485
System
ROM
Working
RAM
PRAM
EPROM
Compile
RAM
Expansion boards
Up to 4 boards

## ■ MX30/MX20

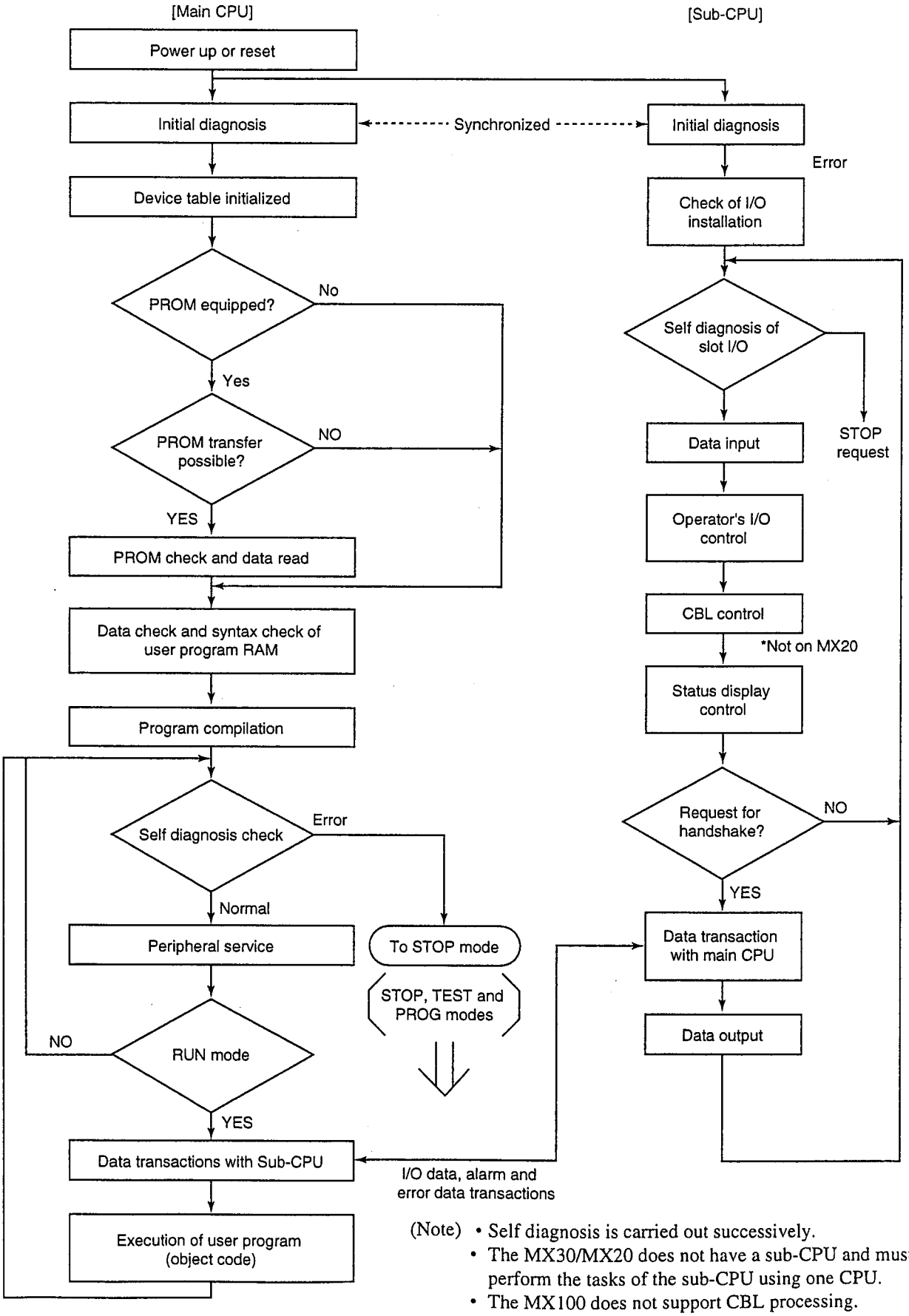



| Device | Description |
|---|---|
| Main CPU | This is the main CPU. This CPU is in charge of overall control of the processor module. It conducts I/O transactions with the sub-CPU and handles self diagnosis, program execution and peripheral services for ASCII communication. |
| Sub-CPU | This CPU controls I/O transactions between CBL, I/O modules and operator's I/O terminals. It also provides the main CPU with data of its operations and diagnoses key inputs, I/O bus checks, etc. |
| Working RAM | Stores the device table data.<br>(The latch relay, timer/counter and register data are stored in registers that are battery backed up to ensure that it lasts through a power failure.) |
| PRAM | This RAM memory (31 K) stores user program source code.<br>User programs from personal loaders, handy loaders* and PROM cards are written to this memory as source code. |
| Compile RAM | PRAM compiles stored user program source code i.e. converts it to machine language (object code) and stores it in this form.<br>The CPU reads user programs stored in the compile RAM during execution to fetch required machine code data.<br>This ensures speedy processing. |
| System ROM | This system ROM is used for self diagnosis. |

* Only for MX100/MX30/MX20.

# 5 - 2 Processor Module Operation

## ■ Operation flow




(Note) • Self diagnosis is carried out successively.
• The MX30/MX20 does not have a sub-CPU and must perform the tasks of the sub-CPU using one CPU.
• The MX100 does not support CBL processing.

廃番

## ■ Main CPU operations

1. Initial diagnosis
   The CPU performs a ROM sumcheck and a read and write test of RAM.

2. Initialization of device table
   The CPU checks the status of the device table when the MX controller is powered up or when the operation mode is changed from STOP to RUN/TEST modes.

| Device | | Devise state | | | |
|---|---|---|---|---|---|
| | | Master (MX200) | | Slave (MX100, OP, ST100, etc.) | |
| | | Used | Not used | Used | Not used |
| X | M000 to X19F | Clear → Input data | Clear (no module) | Clear → Input data | Clear (no module) |
| | X200 to X29F | Clear → Input data | Clear (no module) | Clear → Master data from X200 to 21F | Clear (no module) |
| Y | Y000 to Y19F | Clear | Clear → Can be used as M contact | Clear | Clear → Can be used as M contact |
| | Y200 to Y29F | Clear | Clear → Can be used as M contact | Clear | Clear → Can be used as M contact |
| M000 to M89F | | Clear | | | |
| T/C contact | | | | | |
| L000 to L49F | | Hold | | | |
| R0000 to R0499<br>R1000 to R4999 | | | | | |
| Current T/C value | | Cleared when used with a TMR instruction.<br>Held when used with CNT, STM, UDC, DCNT and DUDC instructions. | | | |
| T/C contact | | | | | |
| R0500 to R0519 | | Hold → Input data | Hold | Hold → Input data | Hold |
| R0600 to R0619 | | Clear → Input data | Clear → When used as general-purpose register | Clear → Input data | Clear → When used as general-purpose register |
| * P0050 to P3199 | | Hold → PC link for remote I/O data | Hold → When used as general-purpose register | Hold → PC link for remote I/O data | Hold → When used as general-purpose register |

* P0000 to P0049 is CBL network data and P3800 to P3999 is mail communication setting data.
See Chapter 2, "Device Table and I/O Assignment" for information on device tables.

3. Check of user program (RAM)
   A sumcheck of the user program is performed, and a program damage error (M999) is generated if an error is found.

4. Program check
   Users programs undergo a syntax check before they are compiled. If a syntax error is found, the syntax error flag (M99A) is set and the program is not compiled.

5. Program compile
   Compiling turns the user program into machine code which normally takes about 2 seconds.

6. Self diagnosis check
   The result of the self diagnosis determines the operation mode.

7. Peripheral service
   Data transactions are performed between the loader (Handy Loader or personal computer loader) or a host computer.

8. Data transactions with the sub-CPU
   I/O, alarms and error data is passed between the main and sub-CPU.

## ■ Sub-CPU operations

1. Initial diagnosis
   The sub-CPU performs a ROM sumcheck and a read and write test of RAM.

2. Registered I/O check
   The sub-CPU compares registered I/O modules and operator's I/O information with registered I/O modules and operator's I/O data. A verification error is output if they do not match.

3. Slot I/O self diagnosis
   The sub-CPU issues a STOP request if a slot I/O self diagnosis discovers an error.

4. Data input
   The sub-CPU fetches data from input modules.

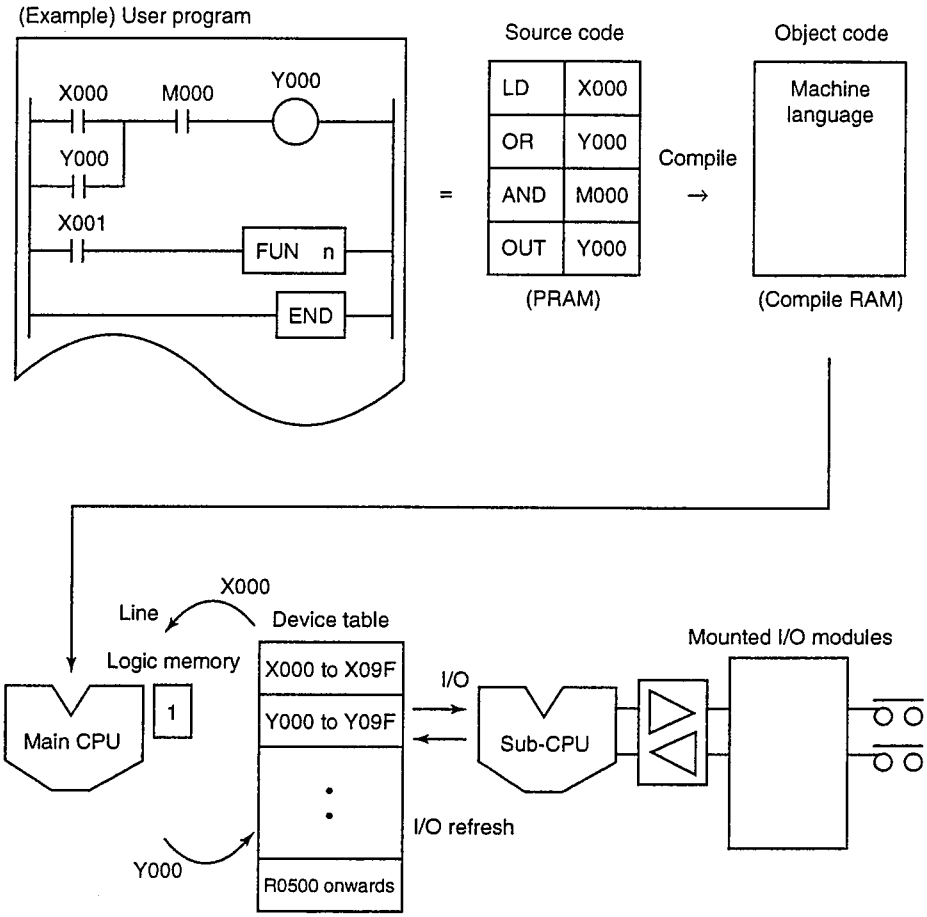
5. Control of operator's I/O
   The sub-CPU controls data transactions with operator's I/O.

6. Status display control
   The sub-CPU outputs a status display on the front panel of the processor module.

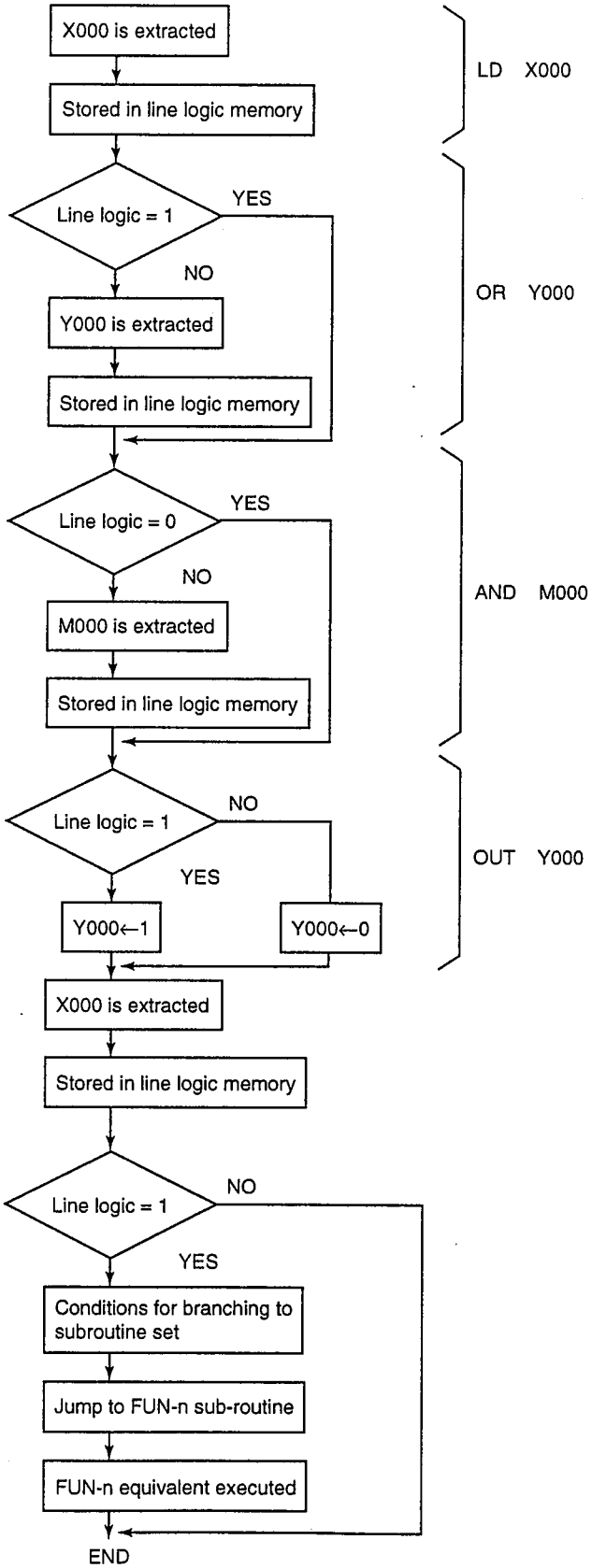
7. Data output
   The sub-CPU outputs data to output modules.

## ■ Ladder processing flow

### ● Internal flow



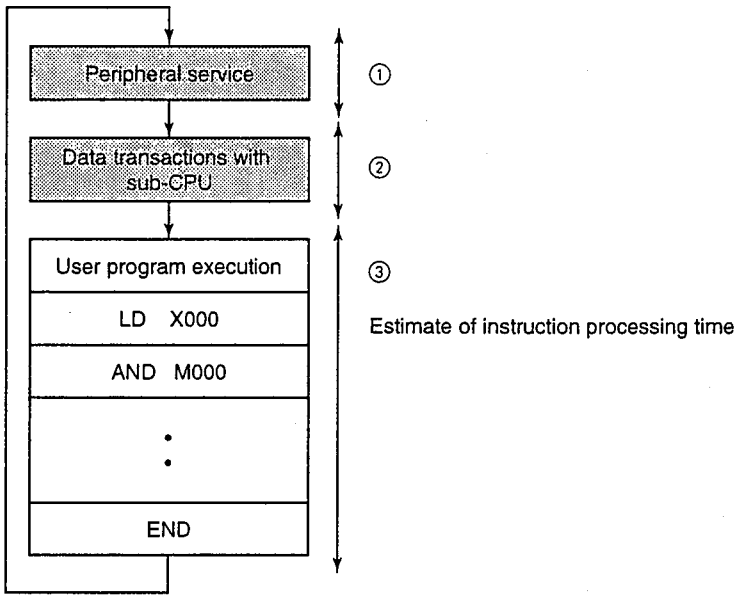
## ● Process flow



# 5 - 3 Scan Time and Response Time

## ■ Scan time

The scan time is the sum of the times required for ① execution of peripheral services, ② data transactions with the sub-CPU, and ③ execution of the user program.



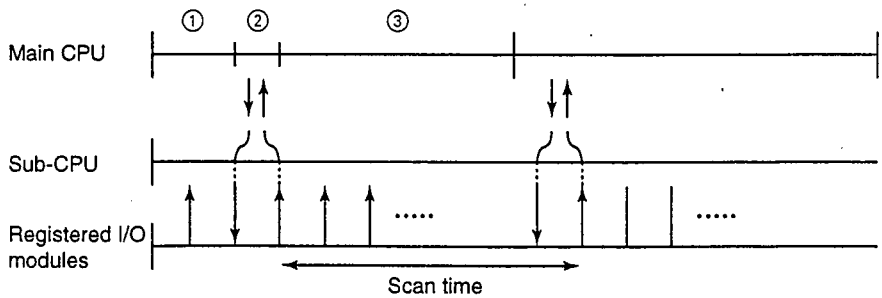

① Peripheral service execution:
This is the time taken to perform data transactions between the loader (personal computer or Handy Loader) or host computer.

② Data transactions with sub-CPU:
This is the time taken to perform data transactions between the sub-CPU and registered I/O modules, operator's I/O and alarm and other data.

③ User program execution:
This is the time taken to process each instruction word.

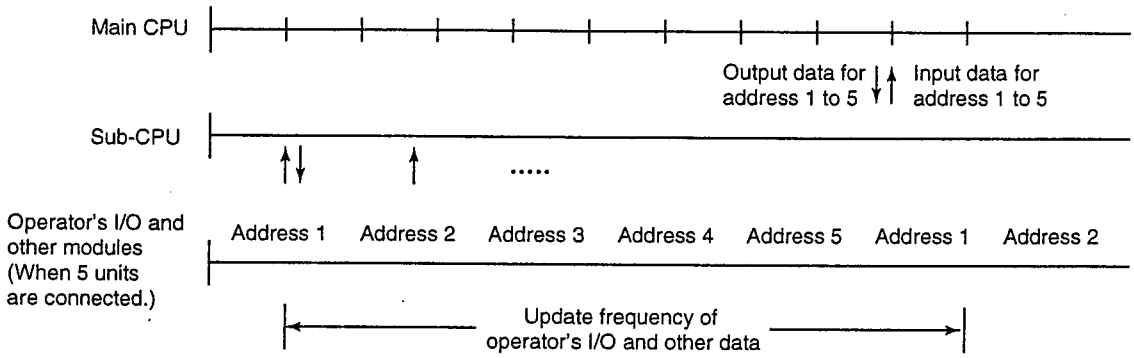
### ● Update frequency of registered I/O module data

The sub-CPU continues to enter data in the registered I/O modules until the main CPU issues a handshake request. The data that the main CPU fetches from the sub-CPU is the latest data.



## ● Update frequency of operator's I/O data

Operator's I/O data is not input or output with the same frequency as data from registered I/O modules. As shown in the diagram below, the sub-CPU inputs and outputs data to and receives data from the operator's I/O once every several scans. When data from all connected operator's I/O modules is fetched, the sub-CPU issues a handshake request to the main CPU and passes all address data to the main CPU while it receives the data of all addresses that should be output.




(Notes)
- Update cycles are reflected in special contact M929.
- The MX30 and MX20 does not have an OP link function.

## ● How to Estimate Scan Time for MX200

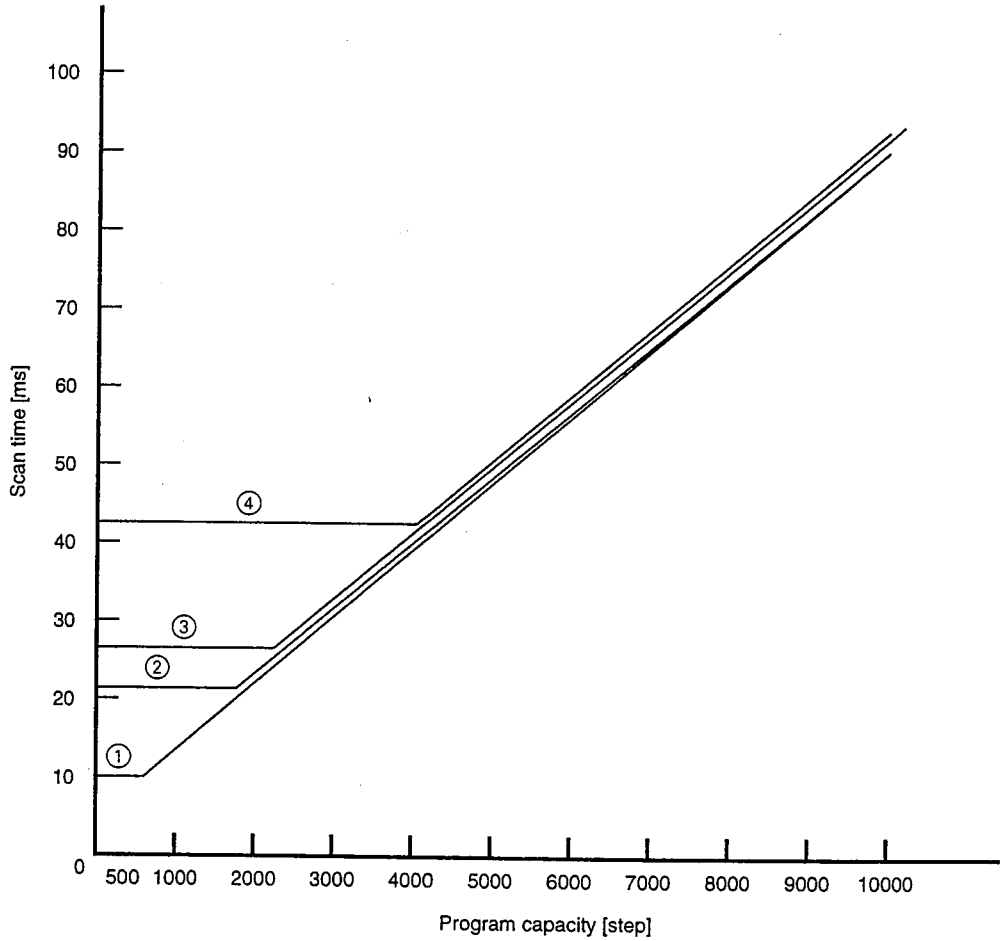
The scan time is the sum of the following:

① peripheral service time + ② data transactions with the sub-CPU + ③ user program execution

(Scan time estimate)

The minimum scan time for the MX200 depends on whether or not CBL is used, operation methods and I/O configuration.

The following graph shows the relationship between program size and scan time (reference value).




Factors that affect scan time

① No CBL but DI/O

| | |
|---|---|
| Input module | : 32 IN x 3, 64 IN x 1 |
| Output module | : 32 OUT x 3, 64 OUT x 1 |

② No CBL, but DI/O and special module

| | |
|---|---|
| Input module | : 32 IN x 1, 64 IN x 1 |
| Output module | : 32 OUT x 1, 64 OUT x 1 |
| Special module (IN/OUT) | : AI (4w/2w), AO (2w/2w), HSC (3w/3w), PPM (6w/6w) |

③ CBL and DI/O

| | |
|---|---|
| CBL | : Five MX200 modules operating in dual mode; PC link execution |
| Input module | : 32 IN x 3, 64 IN x 1 |
| Output module | : 32 OUT x 3, 64 OUT x 1 |

④ CBL, DI/O and special module

| | |
|---|---|
| CBL | : Five MX200 modules operating in dual mode; PC link execution |
| Input module | : 32 IN x 1, 64 IN x 1 |
| Output module | : 32 OUT x 1, 64 OUT x 1 |
| Special module (IN/OUT) | : AI (4w/2w), AO (2w/2w), HSC (3w/3w), PPM (6w/6w) |

**① to ④ shared conditions**

① The ladder program uses a combination of the following ten instructions (1 instruction takes an average of 9 µS):

| | |
|---|---|
| Contact instruction | 4 |
| Output instruction | 2 |
| Compare instruction | 1 |
| BCD addition (F-10) | 1 |
| BIN subtraction (F-71) | 1 |
| 8-word block transfer | 1 |

② The loader is used for monitoring multi-contacts.

③ Data is changed according to the ASCII connector loader settings.

④ No OP link

⑤ Calculate maximum scan time.

## ● Notes on estimating scan time

- When a jump instruction is executed, the area of the program that is bypassed is not executed, but the scan time can be shortened proportional to the size of the skipped area. Master control instructions, on the other hand, execute the program between MC and MCR which means that the scan time does not change.
- The execution of instructions (ISS and IOS) that refresh I/O in a user program lengthens the scan time by 2 to 3 msec per instruction.

## ● Monitoring scan times

- The scan time is managed by a watchdog timer. When the scan time exceeds the value set in the watchdog timer, a scan time-out (M998) is generated. The program then enters the STOP mode. The initial value set in the watchdog timer is 100 ms. This can be changed with the watchdog timer refresh WDT instruction. Execute this instruction regularly.
- The MIN., LAST and MAX. scan time values are stored in the R0900 to R0902 special registers
- When the processor module is in the STOP mode, the value indicated by special register R0901 is the sub-CPU scan time.
  This value indicates the minimum (=0) scan that is required by any step.

廃番

## ■ I/O response time

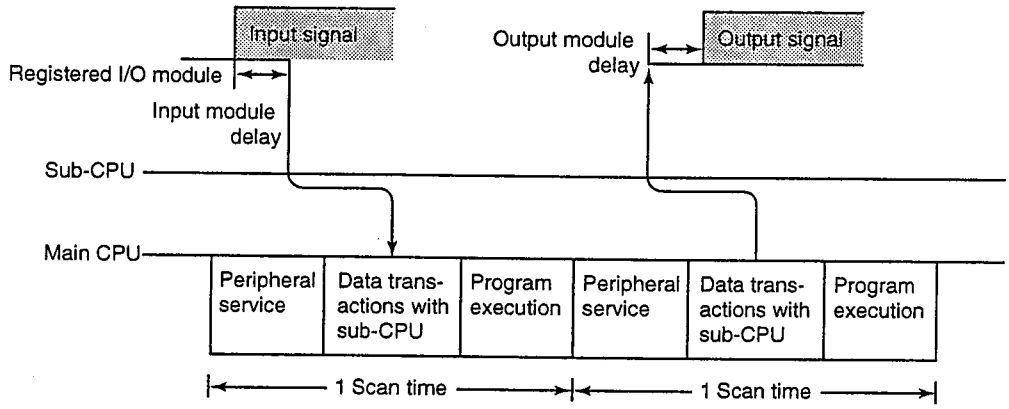
When there is a change in an external input signal, the time it takes from when the signal is output until a result is produced on the output terminal is referred to as the I/O response time.

① Minimum value: When the sub-CPU fetches a registered I/O module change immediately before a data transaction with the main CPU.

② Maximum value: When the sub-CPU fetches a registered I/O module change immediately after a data transaction with the main CPU.

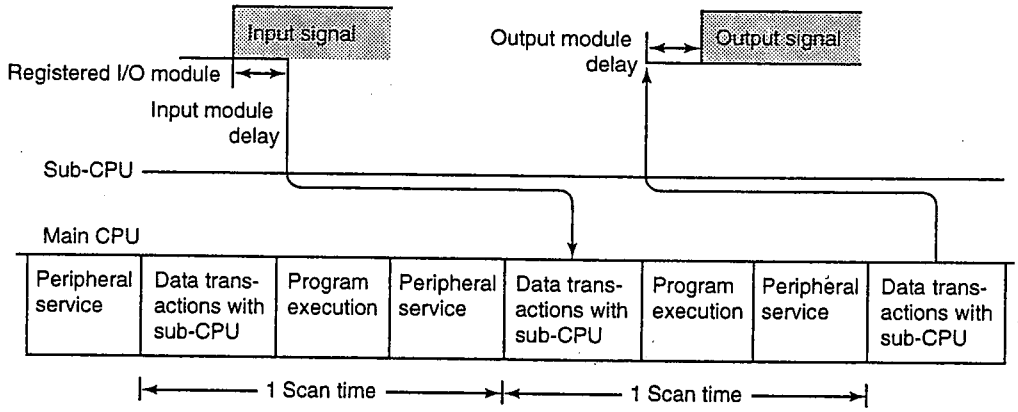
- **Minimum scan time:**

I/O response time is the sum of (input module response time) + 1 scan + (output module response time).




- **Maximum scan time:**

I/O response time is the sum of (input module response time) + 2 scans + (output module response time).



廃 番

# Index

廃 番

廃 番

# Revision History

| Printed Date | Manual Number | Edition | Revised pages | Description |
|---|---|---|---|---|
| **94-12** | **CP-UM-1562E** | **1st Edition** | | |
| **98-06** | | **2nd Edition** | Entire manual | Descriptions for MX50 and MX20 were added on. |
| **02-10** | | **3rd Edition** | | Company name changed.<br>RESTRICTIONS ON USE changed. |
| | | | | |

廃 番

*Specifications are subject to change without notice.*

YAMATAKE

**Yamatake Corporation**
Control Products Division
Head office : Totate International Building
2-12-19 Shibuya Shibuya-ku Tokyo 150-8316 Japan
***Inquiries to :*** International Business Division
Phone : 81-3-3486-2331, Fax : 81-3-3486-2300 (Sales)
Phone : 81-466-20-2307, Fax : 81-466-27-9264 (Customer Service)
http://www.yamatake.com

*This has been printed on recycled paper.*

Printed in Japan.
1st Edition: Issued in Nov., 1989
3rd Edition: Issued in Oct., 2002(M)