MITSUBISHI ELECTRIC

Programmable Controller

MELSEC iQ-F series

MELSEC iQ-F
FX5 User's Manual
(MELSEC Communication Protocol)

SAFETY PRECAUTIONS

(Read these precautions before use.)

Before using this product, please read this manual and the relevant manuals introduced in this manual carefully and pay full attention to safety in order to handle the product correctly.

This manual classifies the safety precautions into two categories: [⚠WARNING] and [⚠CAUTION].

⚠ **WARNING**	Indicates that incorrect handling may cause hazardous conditions, resulting in death or severe injury.
⚠ **CAUTION**	Indicates that incorrect handling may cause hazardous conditions, resulting in minor or moderate injury or property damage.

Depending on the circumstances, procedures indicated by [⚠CAUTION] may also cause severe injury.

It is important to follow all precautions for personal safety.

Store this manual in a safe place so that it can be read whenever necessary. Always forward it to the end user.

[DESIGN PRECAUTIONS]

⚠ WARNING

- Make sure to set up the following safety circuits outside the PLC to ensure safe system operation even during external power supply problems or PLC failure. Otherwise, malfunctions may cause serious accidents.
  (1) Note that when the CPU module detects an error, such as a watchdog timer error, during self-diagnosis, all outputs are turned off. Also, when an error that cannot be detected by the CPU module occurs in an input/output control block, output control may be disabled. External circuits and mechanisms should be designed to ensure safe machinery operation in such a case.
- When executing control (data change) to a running other station programmable controller by connecting the external device to the MC protocol compatible device, configure interlock circuits in the program of the other station programmable controller to ensure that the entire system operates safely at any time.
  For other controls to a running other station programmable controller (such as program modification or operating status change), read relevant manuals carefully and ensure the safety before the operation. Especially, in the case of a control from an external device to a remote other station programmable controller, immediate action cannot be taken for a problem on the programmable controller due to a communication failure.
  Determine the handling method as a system when communication failure occurs along with configuration of interlock circuit on other station PLC program, by considering external equipment and other station PLC.
- Do not write any data into the "system area" or "write protect area" of the buffer memory in the MC protocol compatible device or intelligent function module. Also, do not output (ON) any "use prohibited" signals among the signals which are output to the MC protocol compatible device and intelligent function device. Executing data writing to the "system area" or "write protect area", or outputting "use prohibited" signals may cause malfunction of the programmable controller alarm.

[SECURITY PRECAUTIONS]

⚠WARNING

- To maintain the security (confidentiality, integrity, and availability) of the programmable controller and the system against unauthorized access, denial-of-service (DoS) attacks, computer viruses, and other cyberattacks from unreliable networks and devices via network, take appropriate measures such as firewalls,virtual private networks (VPNs), and antivirus solutions.

[STARTUP AND MAINTENANCE PRECAUTIONS]

⚠WARNING

- Before modifying the program in operation, forcible output, running or stopping the PLC, read through this manual carefully, and ensure complete safety. An operation error may damage the machinery or cause accidents.
- Do not change the program in the PLC from two or more peripheral equipment devices at the same time. (i.e. from an engineering tool and a GOT)
  Doing so may cause destruction or malfunction of the PLC program.

[STARTUP AND MAINTENANCE PRECAUTIONS]

⚠CAUTION

- Read relevant manuals carefully and ensure the safety before performing online operations (operation status change) with peripheral devices connected to the running MC protocol compatible device or CPU modules of other stations. Improper operation may damage machines or cause accidents.

INTRODUCTION

This manual explains the specifications and settings related to the MC protocol of the MELSEC iQ-F Series.
It should be read and understood before attempting to install or use the module.
Always forward it to the end user.

Regarding use of this product

- This product has been manufactured as a general-purpose part for general industries, and has not been designed or manufactured to be incorporated in a device or system used in purposes related to human life.
- Before using the product for special purposes such as nuclear power, electric power, aerospace, medicine or passenger movement vehicles, consult Mitsubishi Electric.
- This product has been manufactured under strict quality control. However when installing the product where major accidents or losses could occur if the product fails, install appropriate backup or failsafe functions in the system.

Note

- If in doubt at any stage during the installation of the product, always consult a professional electrical engineer who is qualified and trained to the local and national standards. If in doubt about the operation or use, please consult the nearest Mitsubishi Electric representative.
- Since the examples indicated by this manual, technical bulletin, catalog, etc. are used as a reference, please use it after confirming the function and safety of the equipment and system. Mitsubishi Electric will accept no responsibility for actual use of the product based on these illustrative examples.
- This manual content, specification etc. may be changed without a notice for improvement.
- The information in this manual has been carefully checked and is believed to be accurate; however, if you notice a doubtful point, an error, etc., please contact the nearest Mitsubishi Electric representative. When doing so, please provide the manual number given at the end of this manual.

CONTENTS

RELEVANT MANUALS

Manual name <manual number>	Description
MELSEC iQ-F FX5S/FX5UJ/FX5U/FX5UC User's Manual (Hardware) <SH-082452ENG>	Describes the details of hardware of the FX5 CPU module, including performance specifications, wiring, installation, and maintenance.
MELSEC iQ-F FX5 User's Manual (Application) <JY997D55401>	Describes the basic knowledge required for program design, functions of the CPU module, devices/labels, and parameters.
MELSEC iQ-F FX5 Programming Manual (Program Design) <JY997D55701>	Describes the specifications of ladders, ST, FBD/LD, and other programs and labels.
MELSEC iQ-F FX5 Programming Manual (Instructions, Standard Functions/Function Blocks) <JY997D55801>	Describes the specifications of instructions and functions that can be used in programs.
MELSEC iQ-F FX5 User's Manual (Serial Communication) <JY997D55901>	Describes the N:N network, Parallel link, MELSEC Communication protocol, inverter communication, non-protocol communication, and predefined protocol support.
MELSEC iQ-F FX5 User's Manual (MELSEC Communication Protocol) <JY997D60801> (This manual)	Explains methods for the device that is communicating with the CPU module by MC protocol to read and write the data of the CPU module.
MELSEC iQ-F FX5 User's Manual (Ethernet Communication) <JY997D56201>	Describes the Ethernet communication function of the CPU module built-in and the Ethernet module.
MELSEC iQ-F FX5 User's Manual (SLMP) <JY997D56001>	Explains methods for the device that is communicating with the CPU module by SLMP to read and write the data of the CPU module.
GX Works3 Operating Manual <SH-081215ENG>	Describes the system configuration, parameter settings, and online operations of GX Works3.

TERMS

Unless otherwise specified, this manual uses the following terms.
For details on the FX3 devices that can be connected with the FX5, refer to the User's Manual (Hardware) of the CPU module to be used.

Term	Description
Engineering tool	The product name of the software package for the MELSEC programmable controllers
Connected station (host station)	Connected station (host station) indicates a station directly connected to an external device. Other station indicates a station connected to the connected station (host station) on the network.
Other station	
Relay station	A station that includes two or more network modules. Transient transmission is performed through this station to stations on other networks.
Buffer memory	Memory areas of intelligent function modules for storing setting values and monitor values.

GENERIC TERMS AND ABBREVIATIONS

Unless otherwise specified, this manual uses the following generic terms and abbreviations.

Generic term/abbreviation	Description
FX3	A generic term for FX3S, FX3G, FX3GC, FX3U, and FX3UC programmable controllers
FX5	A generic term for FX5S, FX5UJ, FX5U, and FX5UC programmable controllers
FX5 CPU module	A generic term for FX5S CPU module, FX5UJ CPU module, FX5U CPU module, and FX5UC CPU module
FX5S CPU module	A generic term for FX5S-30MR/ES, FX5S-40MR/ES, FX5S-60MR/ES, FX5S-80MR/ES[*1], FX5S-30MT/ES, FX5S-40MT/ES, FX5S-60MT/ES, FX5S-80MT/ES[*1], FX5S-30MT/ESS, FX5S-40MT/ESS, FX5S-60MT/ESS, and FX5S-80MT/ESS[*1]
FX5U CPU module	A generic term for FX5U-32MR/ES, FX5U-32MT/ES, FX5U-32MT/ESS, FX5U-64MR/ES, FX5U-64MT/ES, FX5U-64MT/ESS, FX5U-80MR/ES, FX5U-80MT/ES, FX5U-80MT/ESS, FX5U-32MR/DS, FX5U-32MT/DS, FX5U-32MT/DSS, FX5U-64MR/DS, FX5U-64MT/DS, FX5U-64MT/DSS, FX5U-80MR/DS, FX5U-80MT/DS, and FX5U-80MT/DSS
FX5UC CPU module	A generic term for FX5UC-32MT/D, FX5UC-32MT/DSS, FX5UC-64MT/D, FX5UC-64MT/DSS, FX5UC-96MT/D, FX5UC-96MT/DSS, FX5UC-32MT/DS-TS, FX5UC-32MT/DSS-TS, and FX5UC-32MR/DS-TS
FX5UJ CPU module	A generic term for FX5UJ-24MR/ES, FX5UJ-24MT/ES, FX5UJ-24MT/ESS, FX5UJ-40MR/ES, FX5UJ-40MT/ES, FX5UJ-40MT/ESS, FX5UJ-60MR/ES, FX5UJ-60MT/ES, and FX5UJ-60MT/ESS
GX Works3	The product name of the software package, SWnDND-GXW3, for the MELSEC programmable controllers (The 'n' represents a version.)

Generic term/abbreviation	Description
MC protocol	The abbreviation for the MELSEC communication protocol. A protocol for accessing MC protocol-compatible devices and programmable controllers that are connected to MC protocol-compatible devices from external devices.
MC protocol-compatible device	A generic term for devices that can receive MC protocol messages.
SLMP	The abbreviation for Seamless Message Protocol. A protocol for accessing SLMP-compatible devices and programmable controllers that are connected to SLMP-compatible devices from external devices.
Intelligent function module	A generic term for FX5 intelligent function modules and FX3 intelligent function modules
Intelligent module	The abbreviation for intelligent function modules
External device	A generic term for devices of communication target (such as personal computer, HMI)
Serial port	A generic term for the four ports consisting of the built-in RS-485 port of the CPU module (CH1), communication board (CH2), communication adapter 1 (CH3), and communication adapter 2 (CH4).
Module access device	A generic term for the module access devices of the MELSEC iQ-R series/MELSEC iQ-F series and intelligent function module devices of the MELSEC-Q/L series

*1 Area-specific model

1 OUTLINE

This manual describes the method for reading or writing data in a CPU module with the data communication function of the external device using MC protocol (serial communication).
When transferring data using MC protocol, always refer to ☞ Page 12 MC PROTOCOL DATA COMMUNICATION.

1.1 Outline of MC Protocol

MC protocol (MELSEC communication protocol) is a protocol used for access from a CPU module or an external device (such as a personal computer or an HMI) to an MC protocol compatible device.
MC protocol communications are available among devices that can transfer messages by MC protocol.
In case of a serial port of FX5, communication is possible by 1C frame compatible with A and 3C/4C frame compatible with QnA of the MC protocol.
Device data in a CPU module can be written or read from a personal computer or an HMI by using MC protocol.
Writing and reading the device allows operation monitoring, data analyzing, and production managing of a CPU module by a personal computer or an HMI.
The following shows the flow for starting MC protocol communication.

1. Connect cables and external devices.

Configure the connection for the MC protocol communication.
For details, refer to the following manual.
📖MELSEC iQ-F FX5 User's Manual (Serial Communication)

2. Set parameters.

Set parameters with engineering tool.

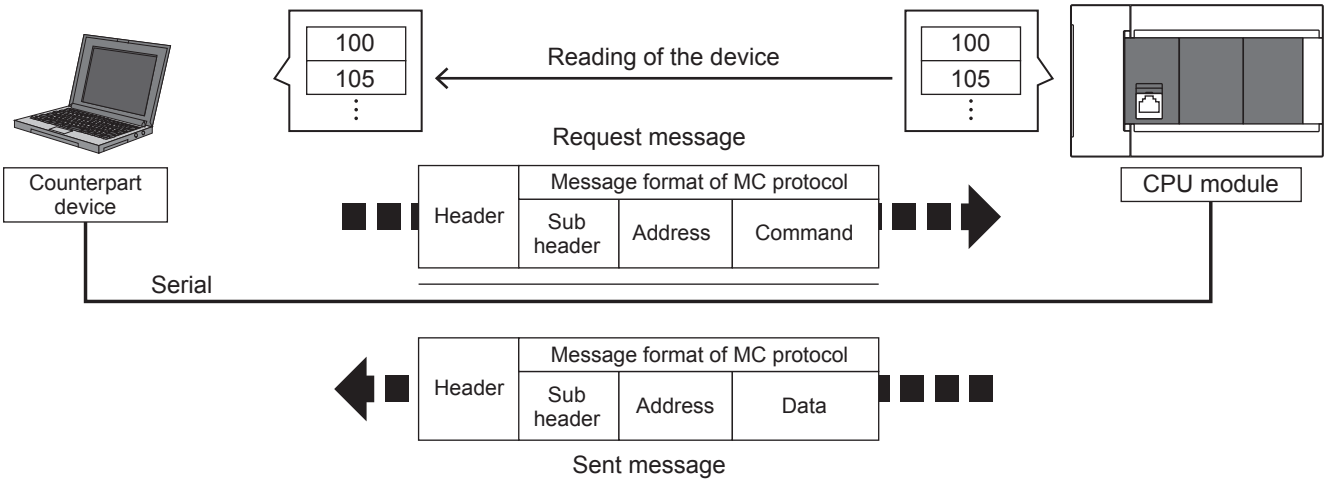
3. Write the set parameters to the CPU module.

Write set parameters to a CPU module. Validate the parameters by turning off to on or resetting the system.

1.2 Features of MC Protocol

System monitoring from an external device (such as personal computer, HMI)

An external device can send a request message in MC protocol message format to a CPU module to enable device read, allowing system monitoring.
Using MC protocol allows not only device data reading but also device data writing and resetting an CPU module.



MEMO

2 MC PROTOCOL DATA COMMUNICATION

This chapter describes the MC protocol data communication by which the external device reads or writes data to a CPU module.

2.1 Type and Application of the Data Communication Frame

This section describes the type and application of the frame (data communication message) by which the external device accesses a CPU module with MC protocol.

When the external device accesses a CPU module using serial communication, the data communication is executed by sending or receiving a request message (access request) and response message of the following frame.

Target communication method	Applicable communication frames	Features and purposes	Communication data code	Section of control procedure
serial communication	4C frame	Accessible from external devices with the maximum access range.	ASCII code or binary code	Page 16 MESSAGE FORMAT
	3C frame	These message formats are simplified compared to the 4C frame.	ASCII code	Page 16 MESSAGE FORMAT
	1C frame*1	These frames have the same message structures as when accessing the CPU module using an FX3 or MELSEC-A series computer link module.	ASCII code	Page 16 MESSAGE FORMAT

*1 For the corresponding version of 1C frame for the FX5U/FX5UC CPU module, refer to ☞ Page 123 Added and Enhanced Functions.

Point

FX5 CPU module supports 3E/1E frame (Ethernet communication) of MC protocol.

The message format of 3E/1E frame of MC protocol is the same as that of the 3E/1E frame of SLMP.

For details on 3E/1E frame of SLMP, refer to the following manual.

- MELSEC iQ-F FX5 User's Manual (SLMP)
- MELSEC iQ-F FX5 User's Manual (Ethernet Communication)

2.2 Concept of Control Procedure of MC Protocol

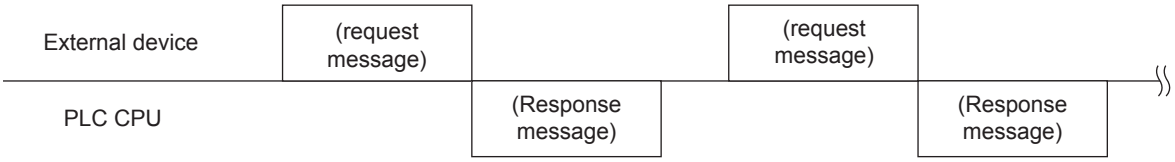
This section describes the concept of the procedure (control procedure) when the external device accesses a CPU module with MC protocol.

Sending a request message

Data communication using MC protocol communication is executed in half-duplex communication.

To access the CPU module, send the next request message after receiving a response message for the preceding request message from the CPU module.

(Until the receiving of the response message is completed, the next request message cannot be sent.)



When a response message of completion for a request message cannot be received

■When a response message of completion with an error is received

Take corrective actions depending on the error code in the response message.

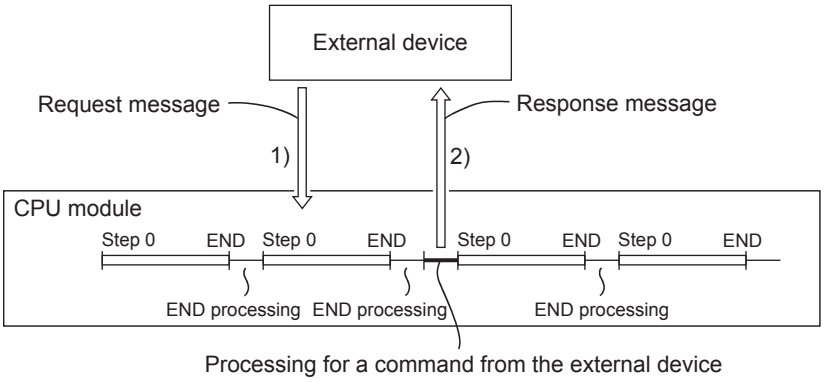
■When a response message or all messages cannot be received

Resend a request message after the monitoring time of the response monitoring timer elapses.
Change the set value of the monitoring time as needed.

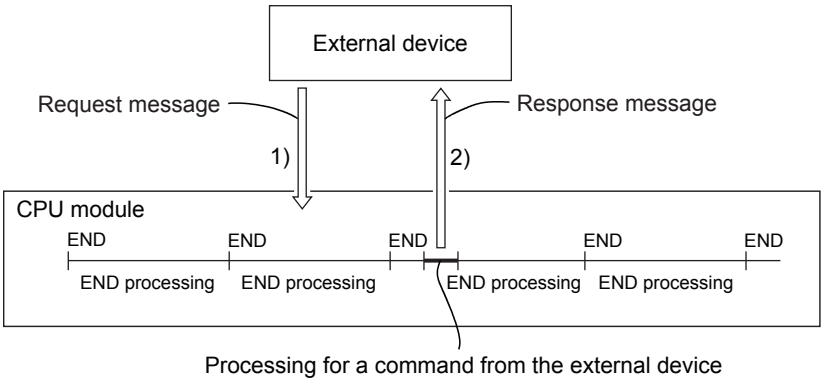
2.3 Access Timing of the CPU Module Side

The following shows the access timing of the CPU module side when the CPU module is accessed from the external device using the serial communication port.

• RUN



• STOP



1. To send a read request or a write request to the CPU module side from the external device, a request message is sent.

2. The CPU module reads or writes the data according to the description requested from the external device when the END instruction of the CPU module is executed and sends a response message including the processing result to the external device of the request source.

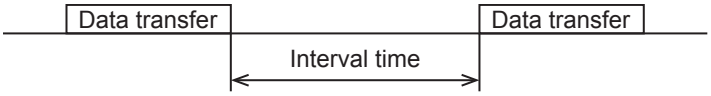
Point

- Access between the external device and CPU module is processed at each END processing when the CPU module is running for a command request. (The scan time becomes longer by the processing time of the command request.)
- When accesses are requested simultaneously to the CPU module from multiple external device, the processing requested from the external device may be on hold until several END processings take place depending on the request timing.

2.4 Transfer Time

Link time

■Data transfer



■Data transfer time

R: Number of read data points, W: Number of written data points, T: Time to send or receive one character, V: Interval time, S: Max Scan Time of PLC, D: Message waiting time

(1) 1C Frame

Time to read continuous word devices (data registers etc.) in one station (ms)

$=(21^{*1} + 4 \times R^{*2}) \times T\ (ms) + V + S\ (SD8012) \times 3 + D$

Time to write continuous word devices (data registers etc.) in one station (ms)

$=(20^{*1} + 4 \times W^{*2}) \times T\ (ms) + V + S\ (SD8012) + D$

*1 This is the number of characters when format1 is used and the sum check is not provided.
When format4 is used, add "4" to this value.
Further, when the sum check is provided, add "4" to this value also.

*2 The number of points is counted in 1-word units.

(2) 3C Frame

Time to read continuous word devices (data registers etc.) in one station (ms)

$=(43^{*3} + 4 \times R^{*4}) \times T\ (ms) + V + S\ (SD524) \times 3 + D$

Time to write continuous word devices (data registers etc.) in one station (ms)

$=(42^{*3} + 4 \times W^{*4}) \times T\ (ms) + V + S\ (SD524) \times 3 + D$

*3 This is the number of characters when format1 is used and the sum check is not provided at the time of execution of batch read/write command.
When format4 is used, add "4" to this value.
Further, when the sum check is provided, add "4" to this value also.
Further, when specifying an extension, add "+7" to this value also.

*4 The number of points is counted in 1-word units.

(3) 4C Frame: In case of ASCII code (When format1 to format4 are used)

Time to read continuous word devices (data registers etc.) in one station (ms)

$=(49^{*5} + 4 \times R^{*6}) \times T\ (ms) + V + S\ (SD524) \times 3 + D$

Time to write continuous word devices (data registers etc.) in one station (ms)

$=(48^{*5} + 4 \times W^{*6}) \times T\ (ms) + V + S\ (SD524) \times 3 + D$

*5 This is the number of characters when format1 is used and the sum check is not provided at the time of execution of batch read/write command.
When format4 is used, add "4" to this value.
Further, when the sum check is provided, add "4" to this value also.
Further, when specifying an extension, add "+7" to this value also.

*6 The number of points is counted in 1-word units.

(4) 4C Frame: In case of binary code (When format5 is used)

Time to read continuous word devices (data registers etc.) in one station (ms)

$=(42^{*7} + 4 \times R^{*8}) \times T\ (ms) + V + S\ (SD524) \times 3 + D$

Time to write continuous word devices (data registers etc.) in one station (ms)

$=(40^{*7} + 4 \times W^{*8}) \times T\ (ms) + V + S\ (SD524) \times 3 + D$

*7 This is the number of characters when format5 is used and the sum check is not provided at the time of execution of batch read/write command.
Further, when the sum check is provided, add "4" to this value also.
Further, when specifying an extension, add "+7" to this value also.
When "10H" exists in the data area, since DLE "10H" is added just before "10H", add “+ "10H numeral”.

*8 The number of points is counted in 1-word units.

■Time to send or receive one character

The table below shows the time required to send or receive one character when the start bit is 1-bit, the data length is 7-bit, the parity is 1-bit, and the stop bit is 1-bit (total 10-bits).

Transmission speed (baud rate) (bps)	Time to send or receive 1 character (ms)
300	33.34
600	16.67
1200	8.34
2400	4.17
4800	2.08
9600	1.04
19200	0.52
38400	0.26
57600	0.17
115200	0.08

The tables below show the data transfer times depending on the number of continuously read or written word devices at transmission speeds of 9600 bps and 19200 bps when the message waiting time is 0 ms[*1], the maximum scan time is 20 ms, and the interval time is 100 ms.

• When the transmission speed is 9600 bps (Unit: Second)

Number of data points	Number of stations		
	Station No. 1	Station No. 8	Station No. 16
10 points	0.3	1.9	3.7
32 points	0.4	2.6	5.2
64 points	0.5	3.7	7.3

• When the transmission speed is 19200 bps (Unit: Second)

Number of data points	Number of stations		
	Station No. 1	Station No. 8	Station No. 16
10 points	0.2	1.6	3.2
32 points	0.3	2.0	3.9
64 points	0.4	2.5	5.0

When the types of read or written devices increase, "Data transfer time shown in above table × Number of device types" is required.

When the number of read or written points exceeds "64", the transfer time increases.

Accordingly, for achieving efficient data transfer, it is recommended to decrease the number of types of transferred devices and use as many continuous device numbers as possible.

*1 When RS-485 one-pair wiring using FX-485PC-IF is adopted, the message waiting time (for every exchange) must be 70 to 150ms. When RS-485 two-pair wiring or RS-232C is adopted, the message waiting time becomes 0ms.

3 MESSAGE FORMAT

This chapter describes the message data format, the data specification method, and limitations etc. when performing MC protocol data communication using the 3C/4C frame to the serial communication port.

3.1 Types and Purposes of Messages

The messages of MC protocol can be classified as shown in the following table depending on the supported device and its intended purpose.

Formats and codes

There are five formats for the message that can be used for serial communication module.

Format	Code of communication data	Remarks	Reference	Corresponding of FX5	GX Works3 setting
Format 1	ASCII code	—	Page 17 Format 1	○	Message Pattern: Pattern 1 (X, Y OCT), Pattern 1 (X, Y HEX)*1
Format 2	ASCII code	Format with block number appended	—	×	—
Format 3	ASCII code	Format enclosed with STX and ETX	—	×	—
Format 4	ASCII code	Format with CR and LF appended at the end	Page 18 Format 4	○	Message Pattern: Pattern 4 (X, Y OCT), Pattern 4 (X, Y HEX)*1
Format 5	Binary code	Can be used by 4C frame.	Page 19 Format 5	○	Message Pattern: Pattern 5

○: Applicable, ×: Not applicable

*1 For supported versions of each format for the FX5U/FX5UC CPU module, refer to ☞ Page 123 Added and Enhanced Functions.
Set the format with the module parameter of GX Works3.

Point

Communication using binary code shorten the communication time since the amount of communication data is reduced by approximately half as compared to the one using ASCII code.

Frame

This section explains the types and purposes of the frames (data communication messages) used by the external device to access the supported devices using MC protocol.
The frames for MC protocol (serial communication) are as follows:

Frame	Features and purposes	Compatible message format	Format	Corresponding of FX5
4C frame	Accessible from external devices with the maximum access range.	Dedicated protocols for MELSEC-QnA series serial communication modules (QnA extension frame).	Formats 1 to 5	○ Corresponding to formats 1, 4, and 5.
3C frame	These message formats are simplified compared to the 4C frame. Data communication software for MELSEC-QnA series programmable controllers can be used.	Dedicated protocols for MELSEC-QnA series serial communication modules (QnA frame).	Formats 1 to 4	○ Corresponding to formats 1 and 4.
2C frame		Dedicated protocols for MELSEC-QnA series serial communication modules (QnA simplified frame).		×
1C frame	These frames have the same message structures as when accessing the CPU module using an FX3 or MELSEC-A series computer link module. Data communication software for FX3 or MELSEC-A series programmable controllers can be used.	Dedicated protocols for MELSEC-A series computer link modules		○ Corresponding to formats 1 and 4.

○: Applicable, ×: Not applicable

3.2 Message Formats of Each Protocol

This section explains the message format and setting data per each format.

Format 1

There are 2 types of formats: format 1 (X, Y OCT) and format 1 (X, Y HEX).

The specification method of the device number for the X (input) and Y (output) to be accessed is different from each other. (☞ Page 34 Device Access)

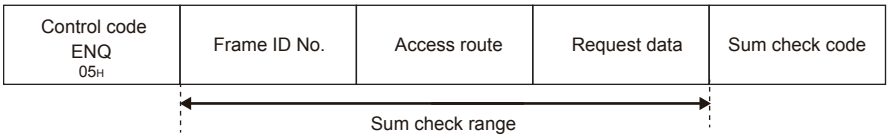
- Format 1 (X, Y OCT): octal
- Format 1 (X, Y HEX): hexadecimal

Unless otherwise specified, the both of them are described as format 1.

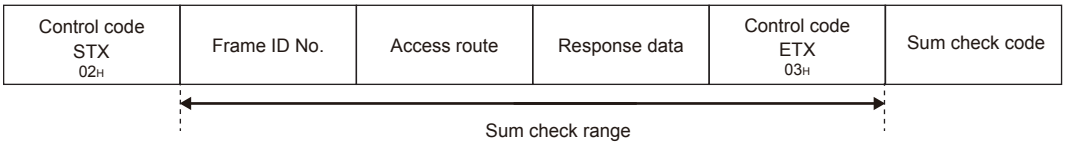
Format 1 (X, Y HEX) cannot be used depending on the version. For supported versions of each format for the FX5U/FX5UC CPU module, refer to ☞ Page 123 Added and Enhanced Functions.

Message format

■Request message

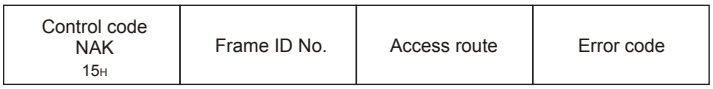


■Response message (Normal completion: Response data)



■Response message (Normal completion: No response data)

■Response message (Abnormal completion)



Setting data

Set the following items.

Item	Description	Reference
Control code (ENQ, STX, ACK, NAK, ETX)	A code is defined for control.	Page 20 Control code
Frame ID No.	Specify the frame to be used.	Page 22 Frame ID No.
Access route	Specify the access route.	Page 25 Accessible Ranges and Settable Data for Each Frame
Request data	Set the command that indicates the request content. Refer to "Request data" rows of each command.	Page 31 COMMANDS
Response data	Store the read data for the command. Refer to "Response data" rows of each command.	
Sum check code	The value of the lower one byte (8 bits) of the additional result regarding the data in the sum check target range as a binary data.	Page 22 Sum check code
Error code	Error code indicates the content of occurred error.	Page 24 Error code

Format 4

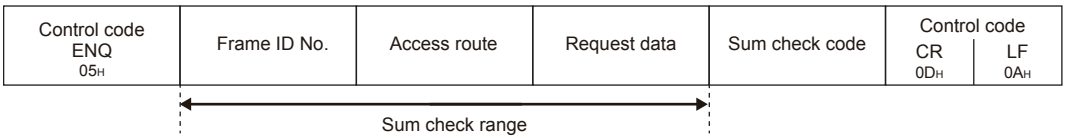
There are 2 types of formats: format 4 (X, Y OCT) and format 4 (X, Y HEX).
The specification method of the device number for the X (input) and Y (output) to be accessed is different from each other. (☞ Page 34 Device Access)

- Format 4 (X, Y OCT): octal
- Format 4 (X, Y HEX): hexadecimal

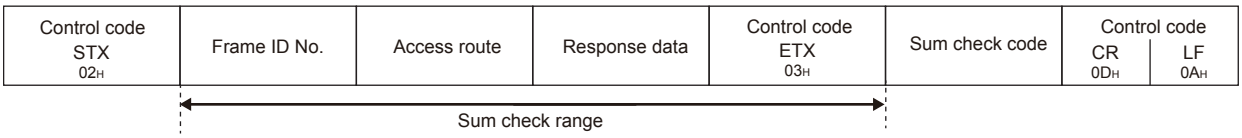
Unless otherwise specified, the both of them are described as format 4.
Format 4 (X, Y HEX) cannot be used depending on the version. For supported versions of each format for the FX5U/FX5UC CPU module, refer to ☞ Page 123 Added and Enhanced Functions.

Message format

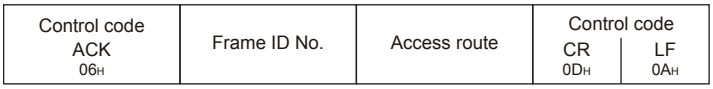
■Request message



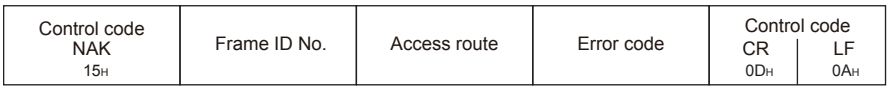
■Response message (Normal completion: Response data)



■Response message (Normal completion: No response data)



■Response message (Abnormal completion)



Setting data

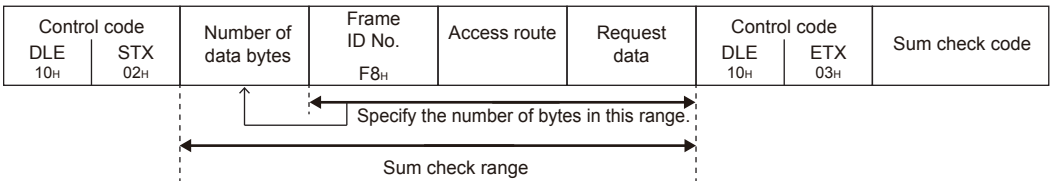
Set the following items.

Item	Description	Reference
Control code (ENQ, STX, ACK, NAK, ETX, CR, LF)	A code is defined for control.	Page 20 Control code
Frame ID No.	Specify the frame to be used.	Page 22 Frame ID No.
Access route	Specify the access route.	Page 25 Accessible Ranges and Settable Data for Each Frame
Request data	Set the command that indicates the request content. Refer to "Request data" rows of each command.	Page 31 COMMANDS
Response data	Store the read data for the command. Refer to "Response data" rows of each command.	
Sum check code	The value of the lower one byte (8 bits) of the additional result regarding the data in the sum check target range as a binary data.	Page 22 Sum check code
Error code	Error code indicates the content of occurred error.	Page 24 Error code

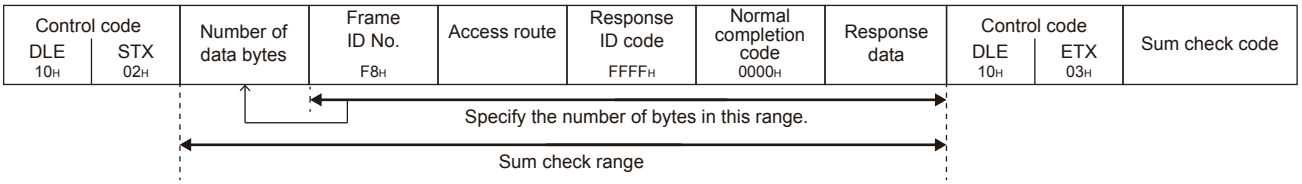
Format 5

Message format

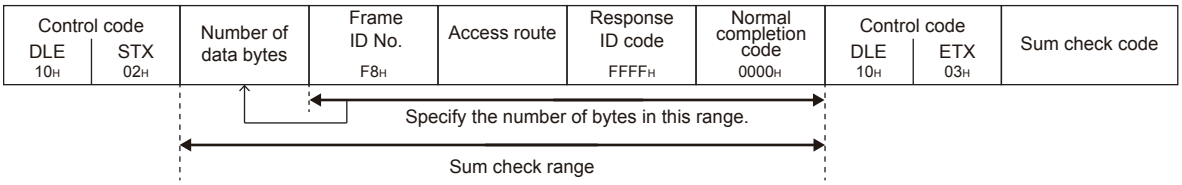
■Request message



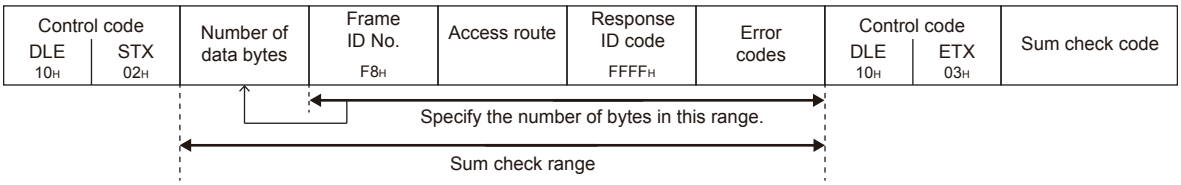
■Response message (Normal completion: Response data)



■Response message (Normal completion: No response data)



■Response message (Abnormal completion)



Setting data

Set the following items.

Item	Description	Reference
Control code (DLE, STX, ETX)	A code is defined for control.	Page 20 Control code
Number of data bytes	The number of bytes from the frame ID No. to control code (DLE, ETX).	Page 21 Number of data bytes
Frame ID No.	Specify the frame to be used.	Page 22 Frame ID No.
Access route	Specify the access route.	Page 25 Accessible Ranges and Settable Data for Each Frame
Request data	Set the command that indicates the request content. Refer to "Request data" rows of each command.	Page 31 COMMANDS
Response data	Store the read data for the command. Refer to "Response data" rows of each command.	
Sum check code	The value of the lower one byte (8 bits) of the additional result regarding the data in the sum check target range as a binary data.	Page 22 Sum check code
Response ID code	This indicates a response message. The 2-byte numerical value, 'FFFH' is stored.	—
Normal completion code	This indicates the processing is completed normally. The 2-byte value, '000H' is stored.	—
Error code	Error code indicates the content of occurred error.	Page 24 Error code

3.3 Details of Setting Data (Format)

This section explains how to specify the common data items and their content in each message.

Control code

Control code is a data that has special meaning (such as head data of a message) for transmission control.

Control code used in a message (format 1 to format 4) in ASCII code

The control code used for a message in ASCII code (format 1 to format 4) is shown in the following table.

Symbol name	Description	Code (hexadecimal)
STX	Start of Text	02H
ETX	End of Text	03H
EOT	End of Transmission	04H
ENQ	Enquiry	05H
ACK	Acknowledge	06H
LF	Line Feed	0AH
CL	Clear	0CH
CR	Carriage Return	0DH
NAK	Negative Acknowledge	15H

■EOT(04H), CL(0CH)

EOT and CL are codes for initializing the transmission sequence for data communications in ASCII code using the MC protocol and for placing CPU module into wait state to receive commands from an external device.

The transmission sequence is initialized with the command (command code: 1615) when binary code (format 5) is used.

When performing the following at an external device, send the EOT/CL to the CPU module depending on the format used.

- Canceling a read/write request by command previously sent. (If a write request is issued, the write request cannot be canceled when the data has already written to the CPU module.)
- Placing CPU module into the wait state to receive commands before commands are sent.
- Placing CPU module into the state where it has been started up when data communication cannot be performed normally.

The message structure when sending EOT, CL is shown below.

Only the following data is sent. The station No. and PC No. are not required.

Format	EOT	CL
Format 1	EOT 04H	CL 0CH
Format 4	EOT 04H, CR 0DH, LF 0AH	CL 0CH, CR 0DH, LF 0AH

When CPU module receives EOT or CL, it proceeds as follows.

- Terminates any read/write processing performed upon request from the external device. In this case, CPU module does not send a response message to the command previously received.
- CPU module initializes the transmission sequence using the MC protocol and placing CPU module into wait state to receive commands from an external device.
- CPU module does not send a response message to the EOT or CL reception. (It does not send anything to external devices.)

Control code used in a message (format 5) in binary code

The control code used for a message in binary code (format 5) is shown in the table below.

Symbol name	Description	Code (hexadecimal)
STX	Start of Text	02H
ETX	End of Text	03H
DLE	Data Link Escape	10H

■Additional code (10H)

The additional code is added to distinguish the data when the control code (10H) is the same as the setting data in frame 5.
When '10H' is included in the data from "Number of data bytes" and "Request data" in the request message, the additional code '10H' is added in front of the data.
When '10H' is included in the data from "Number of data bytes" and "Response data" in the response message, the additional code '10H' is added.
('10H' is transmitted as '10H' + '10H'.)

Point

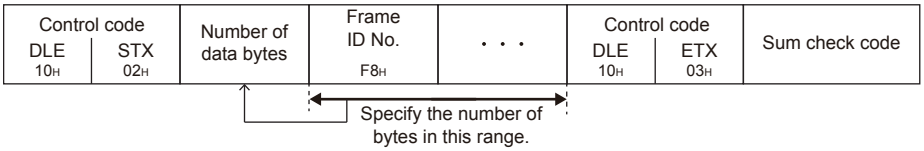
Calculate the following value except for the additional code.
- Number of data bytes (setting item of format 5)
- Sum check code

Number of data bytes

A number of data bytes indicates the total number of bytes from the frame ID No. to control code.

Range

Calculate the data in the range from frame ID No. before DLE (10H) except for the additional code. (☞ Page 21 Additional code (10H))



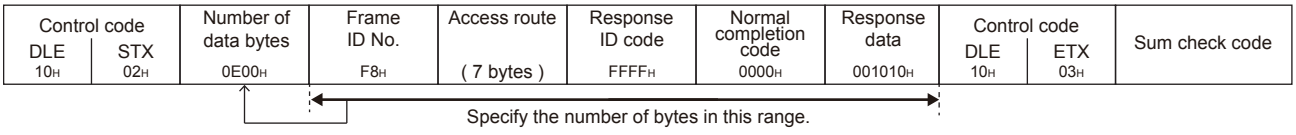
Setting method

Set the data in binary code (format 5) at data communication.
Send 2-byte numerical value from the lower byte (L: bits 0 to 7).

Ex.

Response message (Normal completion: Response data)
- Frame ID No.: 1 byte
- Access route: 7 byte
- Response ID code, normal completion code: 4 bytes
- Response data: 2 bytes + additional code (10H) 1 byte

Number of data bytes = 1 + 7 + 4 + 2 = 14 (0EH)



Block number

Block number is an arbitrary number defined by an external device and used for data defragmentation.
Block number converts data to 2-digit (hexadecimal) ASCII code within the range of '00H' to 'FFH' and sends them from the upper digits.
CPU module only checks if the block number is specified within the correct range. It does not check whether the block numbers are sent in order.

Frame ID No.

Specify the frame to be used.

Type	Setting value
4C frame	F8
3C frame	F9
1C frame	— (Not required)

Setting method

■Data communication in ASCII code

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

■Data communication in binary code

Send 1-byte numerical value.

Ex.
For 4C frame (F8)

ASCII code	Binary code
F 8 46H 38H	F8H

Sum check code

Set the sum check code when performing sum check.
For sum check code, set the value to be calculated from the data with the range of sum check for error detection.

Sum check

Sum check is a function for detecting error when data changes while data transmission.
Set the sum check existence by Engineering tool.

■When sum check code is set to "Exist"

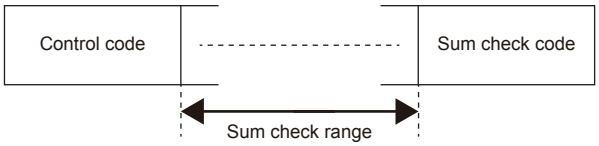
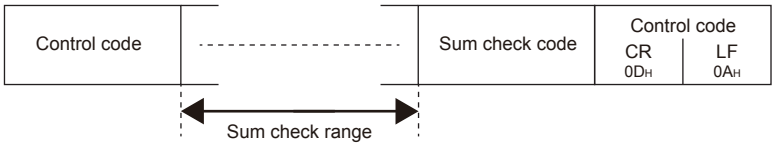
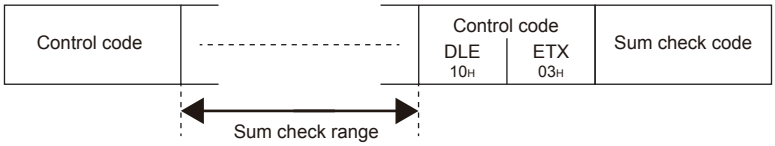
Attach a sum check code to the request message.
CPU module checks the sum check code. The sum check code is added to the response message.

■When sum check code is set to "None"

The sum check code is not required for the request message.
CPU module does not check the sum check code. The sum check code is not added to the response message.

Sum check range

The sum check range of each message format is as follows:

Format	Message structure	Reference
Format 1		Page 17 Format 1
Format 4		Page 18 Format 4
Format 5		Page 19 Format 5

Calculation of a sum check code

For sum check code, set the numerical values of the lower 1 byte (8 bits) of the added result (sum) as binary data within the sum check range.

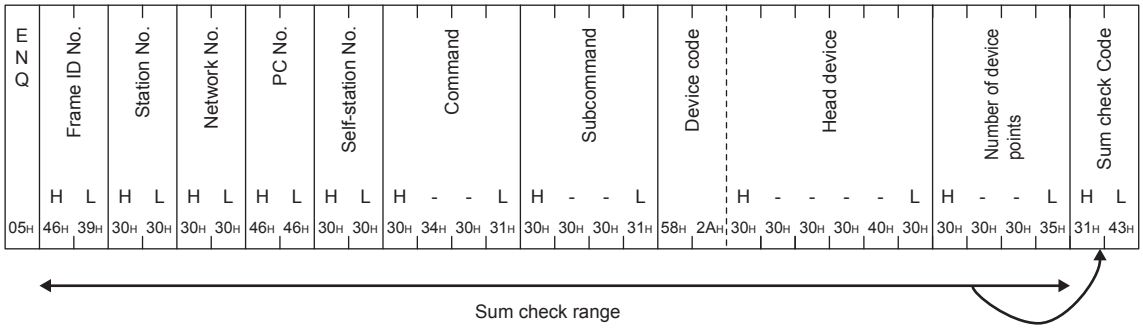
Calculate sum check code except for the additional code. (☞ Page 21 Additional code (10H))

Ex.

In the following case of 3C frame format 1, the sum check code will be '1C'.

Formula: 46H + 39H + 30H + 30H + 30H + 30H + 46H + 46H + 30H + 30H + 30H + 34H + 30H + 31H + 30H + 30H + 30H + 31H + 58H + 2AH + 30H + 30H + 30H + 30H + 34H + 30H + 30H + 30H + 30H + 35H = 61CH

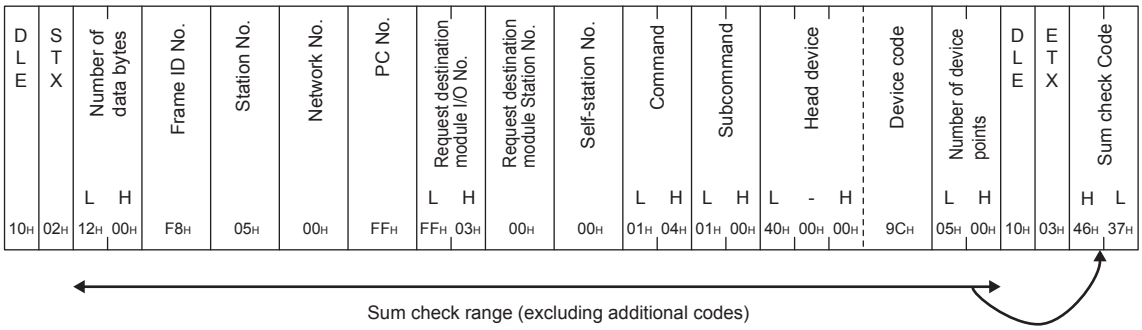
Sum check code: '1C' (ASCII code 31H, 43H)



In the following case of 4C frame format 5, the sum check code will be 'F7'.

Formula: 12H + 00H + F8H + 05H + 00H + FFH + FFH + 03H + 00H + 00H + 01H + 04H + 01H + 00H + 40H + 00H + 00H + 9CH + 05H + 00H = 3F7H

Sum check code: 'F7' (ASCII code 46H, 37H)



Setting method

■Data communication in ASCII code

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

■Data communication in binary code

The same as data communication in ASCII code, use the numerical value converted to 2 digit ASCII code (hexadecimal).

Send 2-byte numerical value from the lower byte (L: bits 8 to 15).

Ex.

Sum check code: 'F7' (ASCII code 46H, 37H)

ASCII code, binary code


Error code

Error code indicates the content of occurred error.

If more than one error occurs at the same time, the error code detected first is returned.

For the content of error code and its corrective action, refer to the following manual.

MELSEC iQ-F FX5 User's Manual (Serial Communication)

MELSEC iQ-F FX5 User's Manual (Application)

Setting method

■Data communication in ASCII code

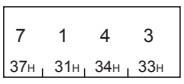
Convert the numerical value to 4-digit ASCII code (hexadecimal) and send it from upper digits.

■Data communication in binary code

Send 2-byte numerical value from the lower byte (L: bits 0 to 7).

Ex.

When error code 7143H is returned

ASCII code	Binary code
	

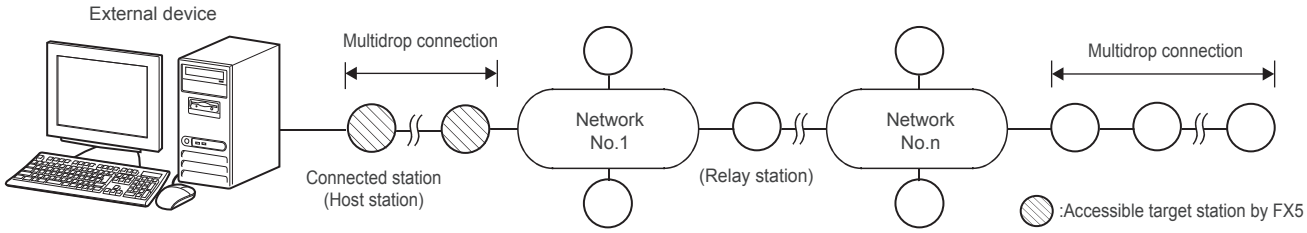
3.4 Accessible Ranges and Settable Data for Each Frame

The accessible range of each frame and the data items to set an access route are as shown below.

4C frame

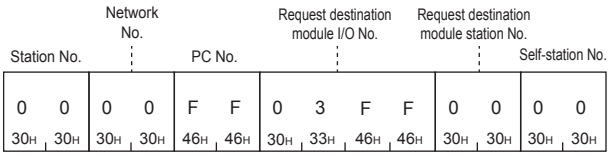
Accessible range of 4C frame

The following ranges can be accessed.

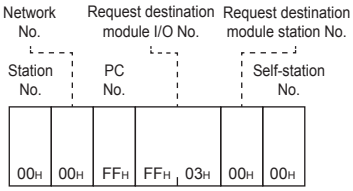


Message format (Setting example for accessing connected station (host station))

■Data communication in ASCII code (Format 1, Format 4)



■Data communication in binary code (Format 5)



Data to be set

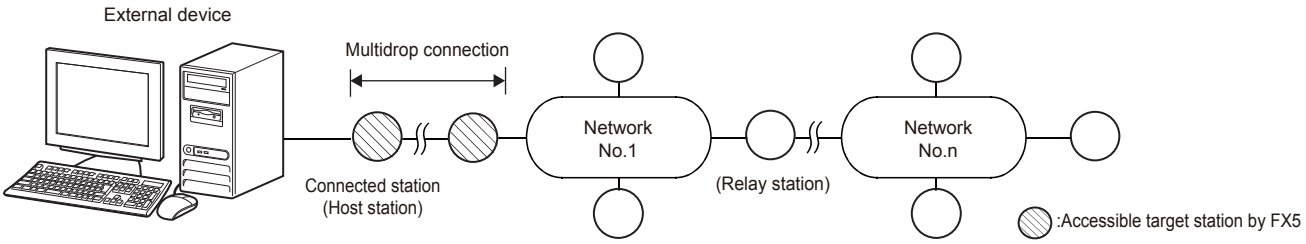
Set the following items.

Item	Description	Reference
Station No.	Specify the station to be connected from an external device.	Page 27 Station No.
Network No.	Specify the access target network No.	Page 28 Network No., PC No.
PC No.	Specify the network No. number of the access target.	
Request destination module I/O No.	Specify the start I/O number of a connection source module (relay station) of multidrop connection via network.	Page 29 Request destination module I/O No., request destination module station No.
Request destination module station No.	Specify the station No. of an access target module of multidrop connection via network.	
Self-station No.	At the time of m:n multidrop connection, specify the station No. of a request source external device.	Page 30 Self-station No.

3C frame

Accessible range of 3C frame

The following ranges can be accessed.



Message format (Setting example for accessing connected station (host station))

Station No.		Network No.		PC No.		Self-station No.	
0	0	0	0	F	F	0	0
30H	30H	30H	30H	46H	46H	30H	30H

Data to be set

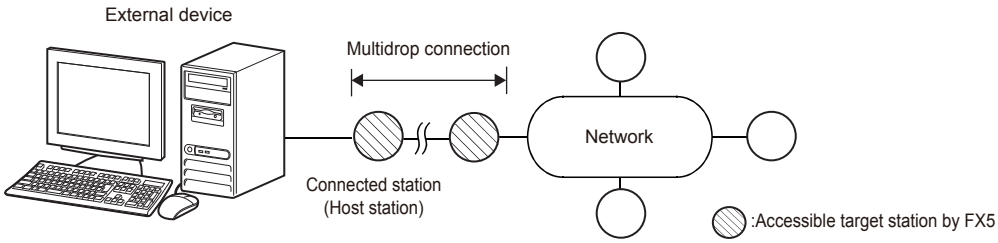
Set the following items.

Item	Description	Reference
Station No.	Specify the station to be connected from an external device.	Page 27 Station No.
Network No.	Specify the access target network No.	Page 28 Network No., PC No.
PC No.	Specify the network station No. of the access target.	
Self-station No.	At m:n multidrop connection, specify the station No. of a request source external device.	Page 30 Self-station No.

1C frame

Accessible range of 1C frame

The following ranges can be accessed.



Message format (Setting example for accessing connected station (host station))

Station No.		PC No.	
0	0	F	F
30H	30H	46H	46H

Data to be set

Set the following items.

Item	Description	Reference
Station No.	Specify the station to be connected from an external device.	Page 27 Station No.
PC No.	Specify the network station No. of the access target.	Page 28 Network No., PC No.

3.5 Details of Setting Data (Frame)

This section explains the content and specification method of the data items to set the access route.

Item	4C frame	3C frame	1C frame	Reference
Station No.	○	○	○	Page 27 Station No.
Network No.	○	○	—	Page 28 Network No., PC No.
PC No.			○	
Request destination module I/O No.	○	—	—	Page 29 Request destination module I/O No., request destination module station No.
Request destination module station No.				
Self-station No.	○	○	—	Page 30 Self-station No.

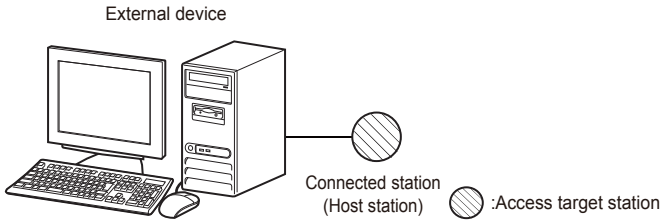
○: Necessary, —: Unnecessary

Station No.

Specify the station accessed from an external device.

Accessing connected station (host station)

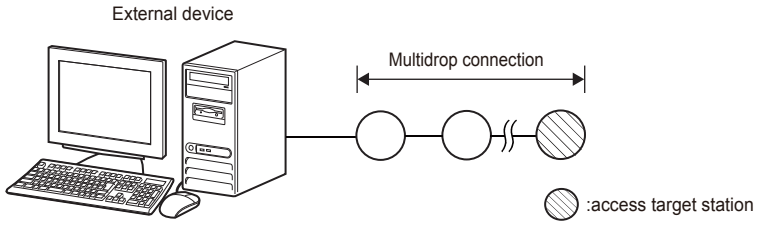
Specify '0' when accessing connected station (host station).



Accessing multidrop connection station

For the multidrop connection, specify the station No. to be accessed from 0 to 15 (00H to 0FH).

When all stations are specified as targets by the global function of 1C frame, specify FFH.



Setting method

The station No. is specified by the following parameter items of engineering tool, and writes the "module parameter" in the CPU module.

- GX Works3: "Station Number Settings" in "Module Parameter"

■Data communication in ASCII code

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

■Data communication in binary code

Send 1-byte numerical value.

Ex.

When the station No. setting for CPU module to be accessed is '5'

ASCII code	Binary code
0 5 30H 35H	05H

Network No., PC No.

Specify the network No. and station No. that are set with the parameters for the access target network module.

Specify a fixed value when accessing the connection station.

Network No. setting is not required for 1C frame.

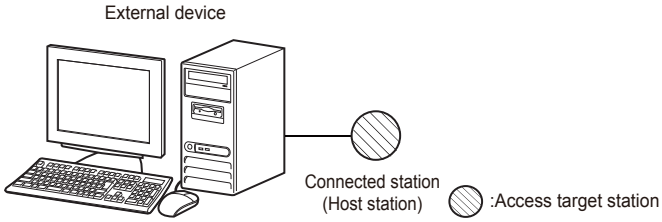
Point

Specify the network No. with the value shown below.

Specifying improper value may result in no response returned.

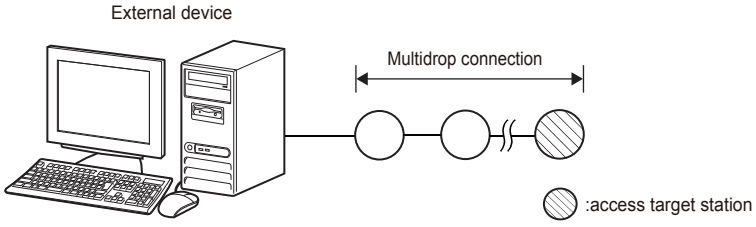
Accessing connected station (host station)

Specify '0' for the network No., and 'FF' for the PC No.



Accessing multidrop connection station

Specify '0' for the network No., and 'FF' for the PC No.



Setting method

■Data communication in ASCII code

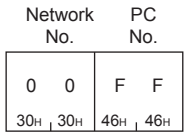
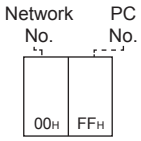
Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

■Data communication in binary code

Send 1-byte numerical value.

Ex.

Accessing connected station (host station) or multidrop connection station

ASCII code	Binary code
	

Request destination module I/O No., request destination module station No.

The following fixed value is specified in FX5 CPU module.

Request destination module I/O No.	Request destination module station No.
03FFH	00H

Setting method

■Data communication in ASCII code

For the request destination module I/O No., convert the numerical value to 4-digit ASCII code (hexadecimal) and send it from upper digits.

For the request destination module station No., convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

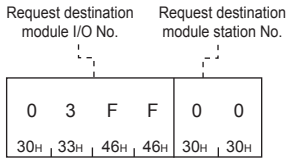
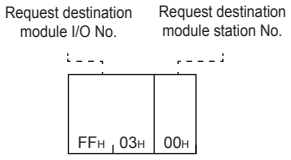
■Data communication in binary code

For the request destination module I/O No., the 2-byte value is sent from the lower byte (L: bit 0 to 7).

For the request destination module station No., the 1-byte value is sent.

Ex.

Accessing connected station (host station)

ASCII code	Binary code
	

Self-station No.

The following fixed value is specified in FX5 CPU module.

Self-station No.
00H

Setting method

■Data communication in ASCII code

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

■Data communication in binary code

Send 1-byte numerical value.

Ex.

When 00H is specified

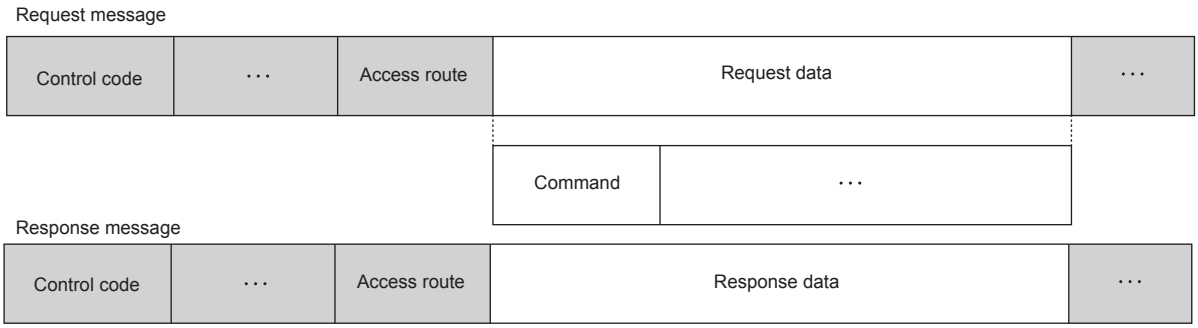
ASCII code	Binary code
	

4 COMMANDS

This chapter explains commands of MC protocol.

4.1 List of Commands and Functions

The functions of a message is defined by each command. The message format for request data and response data varies with commands. Depending on the type of frame to be used, the specific value is assigned to a command. The value of command is specified at the head of a request data.



The explanation of each command in Part 3, the message format of request data and response data are explained.
For the message formats other than request data and response data, refer to the following sections.
☞ Page 16 MESSAGE FORMAT

Command List

1C frame

For the commands for 1C frame, refer to the ☞ Page 93 Command and Function Lists for 1C Frame.

3C/4C frame

Name	Commands	Sub-command	Contents of processing	Number of points processed in one-time update
Batch read	0401H	0001H	Reads data in 1-point units from bit devices or word devices.	ASCII: 3584 points BIN: 3584 points
		0000H	• Reads data in 16-point units from bit devices. • Reads data in 1-word unit from word devices.	960 words (15360 points)
		0081H	• Reads data in 1-bit unit from buffer memory of intelligent unit. • Reads data in 1-bit unit from a device indirectly specified in the index register.	ASCII: 3584 points BIN: 3584 points
		0080H	• Reads data in 1-word unit from buffer memory of intelligent unit. • Reads data in 1-word unit from a device indirectly specified in the index register.	960 words (15360 points)
		0083H	• Reads data in 1-bit unit from buffer memory of intelligent unit. • Reads data in 1-bit unit from a device indirectly specified in the index register.	ASCII: 3584 points BIN: 3584 points
		0082H	• Reads data in 1-word unit from buffer memory of intelligent unit. • Reads data in 1-word unit from a device indirectly specified in the index register.	960 words (15360 points)
Batch write	1401H	0001H	Writes data in 1-bit units to bit devices.	ASCII: 3584 points BIN: 3584 points
		0000H	• Writes data in 16-bit units to bit devices. • Writes data in 1-word units to bit devices.	960 words (15360 points)
		0081H	• Writes data in 1-bit unit to MC protocol compatible devices or buffer memory of intelligent unit. • Indirectly specifies bit devices, word devices and buffer memory in the index register.	ASCII: 3584 points BIN: 3584 points
		0080H	Writes data in 1-word (16-bits) unit to MC protocol compatible devices or buffer memory of intelligent unit.	960 words (15360 points)
		0083H	Writes data in 1-bit unit to MC protocol compatible devices or buffer memory of intelligent unit.	ASCII: 3584 points BIN: 3584 points
		0082H	Writes data in 1-word (16-bits) unit to MC protocol compatible devices or buffer memory of intelligent unit.	960 words (15360 points)
Random read	0403H	0000H	Reads a word device in 1-word unit or 2-word unit by randomly specifying the device number.	192 points
		0080H	Reads data in 1-word (16-bit) unit from MC protocol compatible devices or buffer memory of intelligent unit.	192 points
		0082H	Reads data in 1-word (16-bit) unit from MC protocol compatible devices or buffer memory of intelligent unit.	192 points
Random write	1402H	0001H	Writes data in 1-bit unit to a bit device by randomly specifying the device number.	188 points
		0000H	• Writes data in 16-bit unit to a bit device by randomly specifying the device number. • Writes data in 1-word unit or 2-word unit to a word device by randomly specifying the device number.	(Number of word access points)×12+(number of double word access points)×14≤1920
		0081H	• Writes data in 1-bit unit to MC protocol compatible devices or buffer memory of intelligent unit. • Indirectly specifies a buffer memory in the index register.	188 points
		0080H	Writes data in 1-word (16-bits) or 2-word unit to MC protocol compatible devices or buffer memory of intelligent unit.	(Number of word access points)×12+(number of double word access points)×14≤1920*1
		0083H	Writes data in 1-bit unit to MC protocol compatible devices or buffer memory of intelligent unit.	188 points
		0082H	Writes data in 1-word (16-bits) unit or 2-word unit to MC protocol compatible devices or buffer memory of intelligent unit.	(Number of word access points)×12+(number of double word access points)×14≤1920*1

Name	Commands	Sub-command	Contents of processing	Number of points processed in one-time update
Batch read multiple blocks	0406H	0000H	Assumes an n point part of a bit device or word device as 1-block and reads data by randomly specifying the multiple blocks. (When specifying bit devices, 16-bit is intended in 1-point.)	960 points
		0080H	Assumes an n point part of a MC protocol compatible devices or buffer memory of intelligent unit as 1-block and reads data by randomly specifying the multiple blocks. (When specifying bit devices, 16-bit is intended in 1-point.)	960 points
		0082H	Assumes an n point part of a MC protocol compatible devices or buffer memory of intelligent unit as 1-block and reads data by randomly specifying the multiple blocks.	960 points
Batch write multiple blocks	1406H	0000H	Assumes an n point part of a bit device or word device as 1-block and writes data by randomly specifying the multiple blocks. (When specifying bit devices, 16-bit is intended in 1-point.)	760 points
		0080H	Assumes an n point part of a MC protocol compatible devices or buffer memory of intelligent unit as 1-block and writes data by randomly specifying multiple blocks. (When specifying bit devices, 16-bit is intended in 1-point.)	760 points*1
		0082H	Assumes an n point part of a MC protocol compatible devices or buffer memory of intelligent unit as 1-block and writes data by randomly specifying multiple blocks.	760 points*1
Remote RUN	1001H	0000H	Requests remote RUN to a device.	—
Remote STOP	1002H	0000H	Requests remote STOP to a device.	—
Remote PAUSE	1003H	0000H	Requests remote PAUSE to a device.	—
Remote latch clear	1005H	0000H	Requests remote latch clear when a device is in STOP mode.	—
Remote RESET	1006H	0000H	Requests remote reset to cancel error stop mode of a device.	—
Read CPU model name	0101H	0000H	Reads a processor module name code (processor type) of a device.	—
Loopback test	0619H	0000H	Checks if normal communication is possible.	—
LED OFF, error code initialization	1617H	0000H	Clears all errors in batches and turns OFF LED.	—

*1 With device extension specification, the number of points that can be set become fewer. When using the device extension specification, calculate doubling the number of access points.

4.2 Device Access

This section explains the control procedure specification method and shows a specification example when the device memory is read and written.

Data to be Specified in Commands

This section explains the contents and specification methods for data items which are set in each command related to device access.

Subcommand

Subcommands are data for specifying the unit for reading and writing, device type to be specified, and the data reading condition.

The following table shows the details of setting items.

Setting item		Description
Data size specification	Word units	• The target data is read or written in word units. • Select "0" even when the reading data or writing data does not exist in arguments of the command.
	Bit units	The target data is read or written in bit units.
Device specification format	2 digit code/6 digit number specification	Data or items related to the address specifications are expressed in the following sizes, which are the same as the existing setting. • Device code: 1 byte in binary • Device number: 3 bytes in binary
	4 digit code/8 digit number specification	Data or items related to the address specifications are extended to the following size. • Device code: 2 byte in binary • Device number: 4 bytes in binary
Device memory extension specification	Not specified	Set this when specifying devices of a CPU module. * Set this when not using the device memory extension specification.
	Specified	• Set this for the buffer memory specification of the intelligent function module. • This setting corresponds to the buffer memory indirect specification with index register.

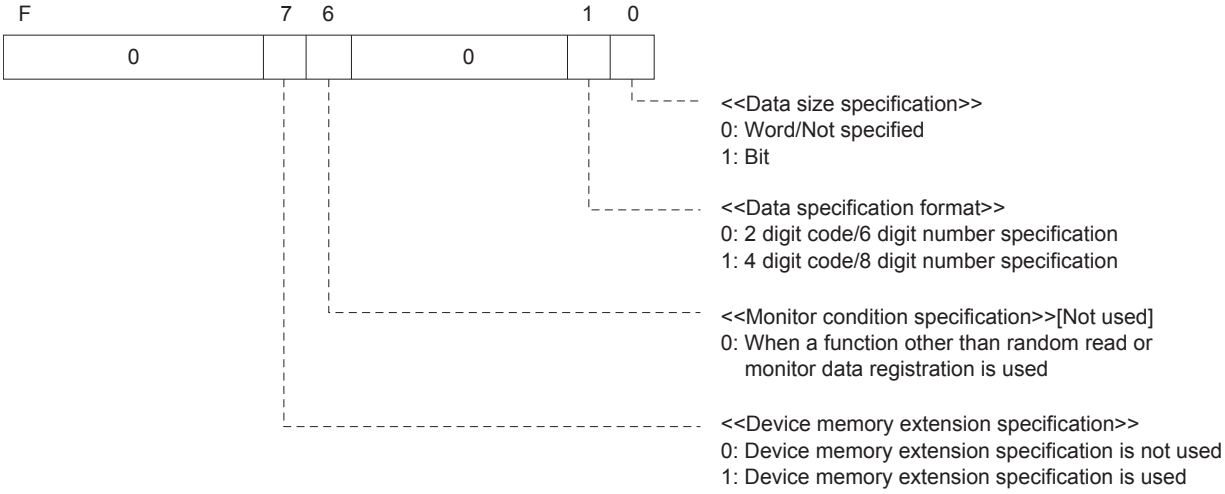
❶When communicating data in ASCII code

The value 0000H(0), or the following value, is converted to a 4 digit (hexadecimal) ASCII code and sequentially transmitted beginning from the most significant digit ("0").

❷When communicating data in binary code

The value 0000H, or the following 2-byte value, is used for transmission.

❸The following figure shows the specification contents of the subcommand.



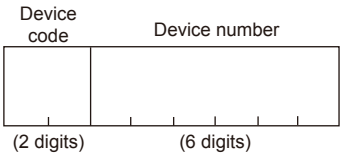
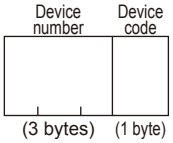
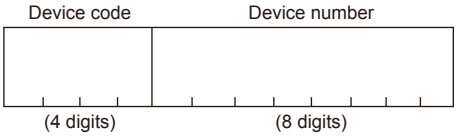
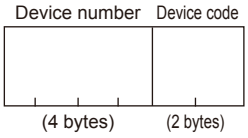
❹In the following cases, the subcommand is 0000H or 0001H.

- When neither monitor condition nor device memory extension is specified.
- When using a command that cannot select monitor condition specification and device memory extension specification.

Devices

Specify the device to be accessed by device code and device number.

- The data order differs between ASCII code and binary code.
- The data size to set up changes with setting of the device specification format of subcommand.

Device specification format of subcommand	ASCII code	Binary code
2 digit code/6 digit number specification		
4 digit code/8 digit number specification		

Point

When accessing any of the following devices, use the device extension specification (subcommand: 008☐).

- Module access device

For the message format for device extension specification, refer to the following section.

☞ Page 112 Device Memory Extension Specification

Device codes

Specify the device name to be accessed.

Specify the device within the range of the access target module.

For the values of each device code, refer to the following section.

☞ Page 37 Device code list

■Data communication using ASCII code

Convert the numerical value to 2-digit or 4-digit ASCII code (hexadecimal), and send it from the upper digits.

- For 2 digit code/6 digit number specification: 2-digit ASCII code
- For 4 digit code/8 digit number specification: 4-digit ASCII code

The '*' in a device code can also be specified with a space (code: 20H).

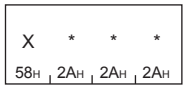
■Data communication using binary code

Send the 1-byte or 2-byte numerical value from the lower byte (L: bits 0 to 7).

- For 2 digit code/6 digit number specification: 1 byte
- For 4 digit code/8 digit number specification: 2 bytes

Ex.

For input (X)

Device specification format of subcommand	ASCII code	Binary code
2 digit code/6 digit number specification		
4 digit code/8 digit number specification		

Device number

Specify the number of the device to be accessed.

Specify the device number within the range of the access target module.

■Data communication using ASCII code

Convert the numerical value to 6-digit or 8-digit ASCII code, and sent it from the upper digits.

Specify the device number in octal, decimal or hexadecimal, depending on the device type. (☞ Page 37 Device code list)

- For 2 digit code/6 digit number specification: 6-digit ASCII code
- For 4 digit code/8 digit number specification: 8-digit ASCII code (10 digits at device extension specification)

The '0' in the upper digits can also be specified with a space (code: 20H).

■Data communication using binary code

Send the 3-byte or 4-byte numerical value in order from the lower byte (L: bit 0 to 7).

For a device of which device number is in decimal, convert it to hexadecimal and specify.

- For 2 digit code/6 digit number specification: 3 bytes*1
- For 4 digit code/8 digit number specification: 4 bytes*1

*1 The additional code may be added. (☞ Page 21 Additional code (10H))

Ex.

For link relay (B) 1234 (a device of which device number is in hexadecimal)

Device specification format of subcommand	ASCII code	Binary code
2 digit code/6 digit number specification	0 0 1 2 3 4 30H 30H 31H 32H 33H 34H	34H 12H 00H
4 digit code/8 digit number specification	0 0 0 0 1 2 3 4 30H 30H 30H 30H 31H 32H 33H 34H	34H 12H 00H 00H

For internal relay (M) 1234 (a device of which device number is in decimal)

For binary code, convert the device number to hexadecimal. '1234' (decimal) → '4D2' (hexadecimal)

Device specification format of subcommand	ASCII code	Binary code
2 digit code/6 digit number specification	0 0 1 2 3 4 30H 30H 31H 32H 33H 34H	D2H 04H 00H
4 digit code/8 digit number specification	0 0 0 0 1 2 3 4 30H 30H 30H 30H 31H 32H 33H 34H	D2H 04H 00H 00H

For internal relay (M) 16 (with additional code)

For CPU module binary code, specify '10H' as '10H + 10H'. (☞ Page 21 Additional code (10H))

Device specification format of subcommand	Binary code
2 digit code/6 digit number specification	DLE 10H 10H 00H 00H
4 digit code/8 digit number specification	DLE 10H 10H 00H 00H 00H

Device code list

The table below shows devices and device number range that can handled in commands used in communication by MC protocol.

Specify devices and device number range that are there in the targeted unit for performing data reading, writing etc.

1C frame

For device code used in 1C frame, refer to ☞ Page 94 Data to be specified in command.

3C/4C frame

In 3C/4C frame, in the following "Device Code", specify a device of the access point.

Division	Device		Type	Device Code[*1] (Device specification format: Long)		Device No.		FX5 device available[*2]
				ASCII code	Binary code			
Internal user devices	Input		Bit	X* (X***)	9CH (9C00H)	Specify within the range of device numbers that the unit at the access point bears.	[*3]	○
	Output			Y* (Y***)	9DH (9D00H)		[*3]	○
	Internal relay			M* (M***)	90H (9000H)		Decimal	○
	Latch relay			L* (L***)	92H (9200H)		Decimal	○
	Annunciator			F* (F***)	93H (9300H)		Decimal	○
	Edge relay			V* (V***)	94H (9400H)		Decimal	—
	Link relay			B* (B***)	A0H (A000H)		Hexadecimal	○
	Step relay			S* (S***)	98H (9800H)		Decimal	○
	Data register		Word	D* (D***)	A8H (A800H)		Decimal	○
	Link register			W* (W***)	B4H (B400H)		Hexadecimal	○
	Timer	Contact	Bit	TS (TS**)	C1H (C100H)		Decimal	○
		Coil		TC (TC**)	C0H (C000H)			○
		Present value	Word	TN (TN**)	C2H (C200H)			○
	Long Timer	Contact	Bit	— (LTS*)	51H (5100H)		Decimal	—
		Coil		— (LTC*)	50H (5000H)			—
		Present value	Double word	— (LTN*)	52H (5200H)			—
	Retentive timer	Contact	Bit	SS (STS*)	C7H (C700H)		Decimal	○
		Coil		SC (STC*)	C6H (C600H)			○
		Present value	Word	SN (STN*)	C8H (C800H)			○
	Long Retentive Timer	Contact	Bit	— (LSTS)	59H (5900H)		Decimal	—
		Coil		— (LSTC)	58H (5800H)			—
		Present value	Double word	— (LSTN)	5AH (5A00H)			—
	Counter	Contact	Bit	CS (CS**)	C4H (C400H)		Decimal	○
		Coil		CC (CC**)	C3H (C300H)			○
		Present value	Word	CN (CN**)	C5H (C500H)			○
	Long counter	Contact	Bit	— (LCS*)	55H (5500H)		Decimal	○
		Coil		— (LCC*)	54H (5400H)			○
		Present value	Double word	— (LCN*)	56H (5600H)			○
	Link special relay		Bit	SB (SB**)	A1H (A100H)		Hexadecimal	○
	Link special register		Word	SW (SW**)	B5H (B500H)		Hexadecimal	○

Division	Device	Type	Device Code*1 (Device specification format: Long)		Device No.		FX5 device available*2
			ASCII code	Binary code			
System device	Special relay	Bit	SM (SM**)	91H (9100H)	Specify within the range of device numbers that the unit at the access point bears.	Decimal	○
	Special Register	Word	SD (SD**)	A9H (A900H)		Decimal	○
	Command input	Bit	—	—	—	Hexadecimal	—
	Command output		—	—		Hexadecimal	—
	Function register	Word	—	—		Decimal	—
Index register		16 bit	Z* (Z***)	CCH (CC00H)	Specify within the range of device numbers that the unit at the access point bears.	Decimal	○
		32 bit	LZ (LZ**)	62H (6200H)		Decimal	○
File register		Word	R* (R***)	AFH (AF00H)		Decimal	○
			ZR (ZR**)	B0H (B000H)		Decimal	—
Unit access device*4	Link register	Word	W* (W***)	B4H (B400H)		Hexadecimal	—
	Link special register		SW (SW**)	B5H (B500H)		Hexadecimal	—
	Module access device		G* (G***)	ABH (AB00H)		Decimal	○

*1 [ASCII code]
When a device code is less than the specified number of characters, add "*" (ASCII code: 2AH), or <space> (ASCII code: 20H) at the end of the device code.
[Binary code]
When a device code is less than the specified size, add "00H" at the end of the device code.

*2 ○: FX5 device
—: No FX5 device

*3 Depends on the message format. See below.
Format 1 (X, Y OCT), format 4 (X, Y OCT): octal
Format 1 (X, Y HEX), format 4 (X, Y HEX), format 5: hexadecimal

*4 It is necessary to make "Device memory extension specification" of the sub-command to ON (1).

Number of device points

Specify the number of device points to be read or written.

Setting method

■Data communication using ASCII code

Convert the numerical value to 4-digit ASCII code (hexadecimal), and send it from the upper digits.

Use capitalized code for alphabetical letter.

■Data communication using binary code

Send the 2-byte numerical value*1 in order from the lower byte (L: bit 0 to 7).

*1 The additional code may be added. (☞ Page 21 Additional code (10H))

Ex.

For 5 points and 20 points

Number of device points	ASCII code	Binary code
5 points	0 0 0 5 30H 30H 30H 35H	05H 00H
20 points	0 0 1 4 30H 30H 31H 34H	14H 00H

Access points

Specify the number of device points to be accessed in word unit, double word unit, or bit unit.

It specifies within the number of points processed which can be performed by the one communication shown in the table (☞ Page 32) of the command list.

Setting method

■Data communication using ASCII code

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Use capitalized code for alphabetical letter.

■Data communication using binary code

Send the 1-byte[*1] numerical value (hexadecimal).

*1 The additional code may be added. (☞ Page 21 Additional code (10H))

Ex.

For 5 points and 20 points

Number of device points	ASCII code	Binary code
5 points	0 5 30H 35H	05H
20 points	1 4 31H 34H	14H

Number of bit access points

Specify the number of device points to be accessed in bit units.

Number of word access points, number of double word access points

Specify the number of device points to be accessed in word unit or double word unit.

Number of blocks

Specify the number of blocks of the device to be accessed in hexadecimal.

Set each number of blocks within the following range.

• Number of word device blocks + Number of bit device blocks ≤ 120

Point

In the following case, calculate it as number of blocks × 2.

• When accessing by setting device extension specification (subcommand: 008□)

Setting method

■Data communication using ASCII code

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Use capitalized code for alphabetical letter.

■Data communication using binary code

Send the 1-byte[*1] numerical value (hexadecimal).

*1 The additional code may be added. (☞ Page 21 Additional code (10H))

Ex.

For 5 points and 20 points

Number of device points	ASCII code	Binary code
5 points	0 5 30H 35H	05H
20 points	1 4 31H 34H	14H

Number of word device blocks

Specify the number of blocks of the word device.

Number of bit device blocks

Specify the number of blocks of the bit device.

Read data, write data

The read device value is stored for reading, and the data to be written is stored for writing.

The data order differs between bit units or word units.

For bit units

The following shows the data to be read and written in bit units.

■Data communication using ASCII code

The ON/OFF status of each device are represented with single-digit ASCII code.

- For ON: '1' (31H)
- For OFF: '0' (30H)

■Data communication using binary code

Represent the ON/OFF status of each device in 4-bit per 1 point.

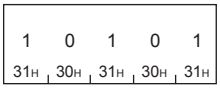
- For ON: '1'
- For OFF: '0'

When the number of points is odd, the lowest 4 bits are set to '0'.

Ex.

When indicating ON/OFF status of five points from M10

M10	M11	M12	M13	M14
ON	OFF	ON	OFF	ON

ASCII code	Binary code*1
	

*1 The additional code may be added. (☞ Page 21 Additional code (10H))

For word units (16-point unit for bit device)

The following shows the data to be read and written in word units.

When handling data other than bit data, refer to the following section.

☞ Page 46 Considerations for handling real number data and character string data

■Data communication using ASCII code

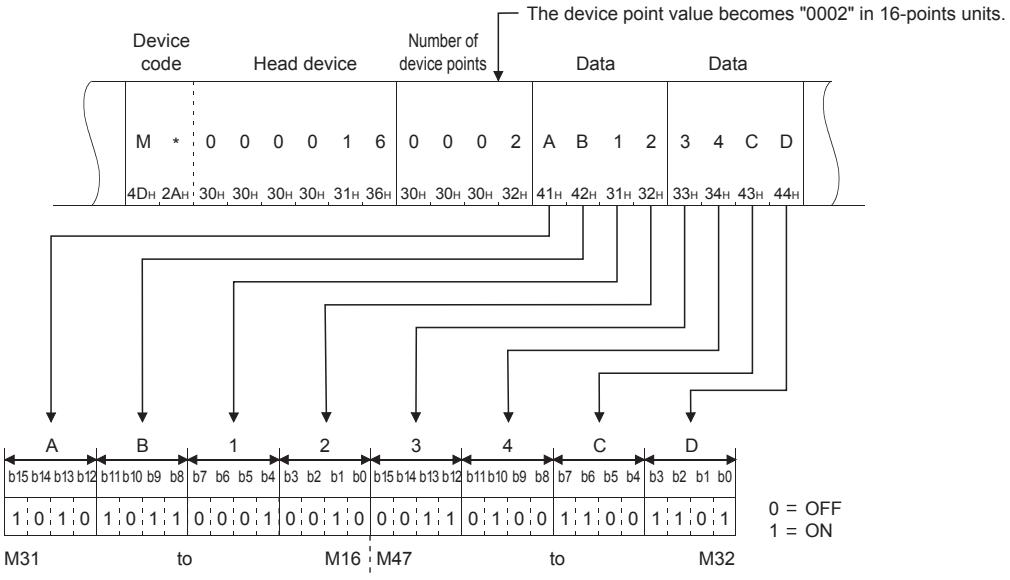
Convert the 1-word(16 points of bit device) numerical value to 4-digit ASCII code (hexadecimal), and send it from the upper digits.

Use capitalized code for alphabetical letter.

The ON/OFF status of bit device is a value of hexadecimal 1-digit in 4-point units.

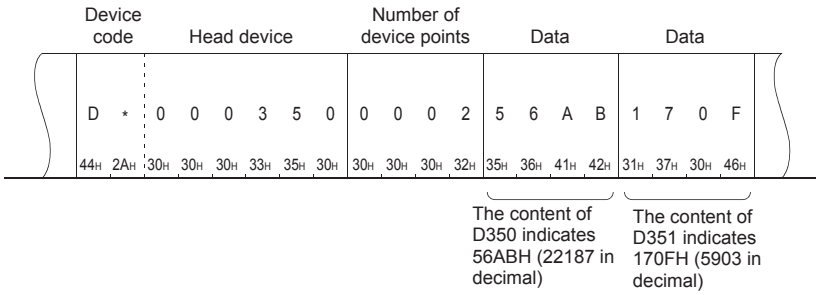
Ex.

When indicating ON/OFF status of 32 points from M16



Ex.

When indicating the stored data of D350 and D351

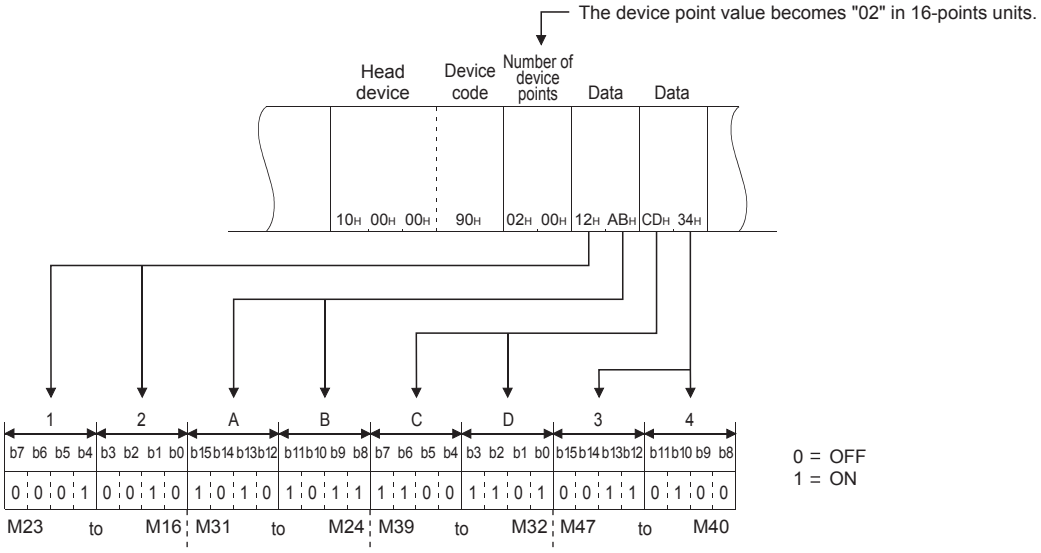


■Data communication using binary code

Send the numerical value in order from the lower byte (L: bit 0 to 7) by handling 16 points unit as 2 bytes.

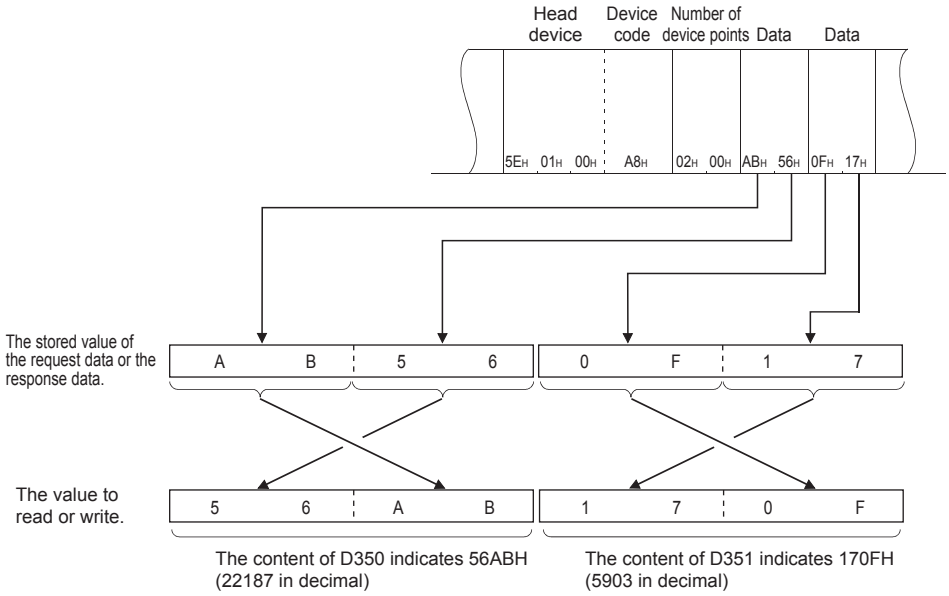
Ex.

When indicating ON/OFF status of 32 points from M16



Ex.

When indicating the stored data of D350 and D351



For double word unit (32-point unit for bit device)

The following shows the data to be read and written in double word units.

■Data communication using ASCII code

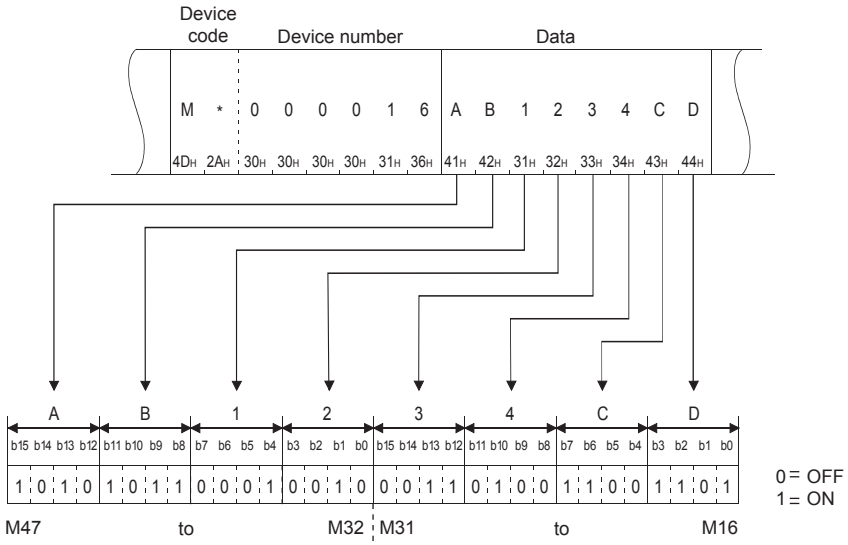
Convert the 2-word numerical value (32 points of bit device) to 8-digit ASCII code (hexadecimal), and send it from the upper digits.

Use capitalized code for alphabetical letter.

The ON/OFF status of the bit device is 1-digit hexadecimal value in 4-point units.

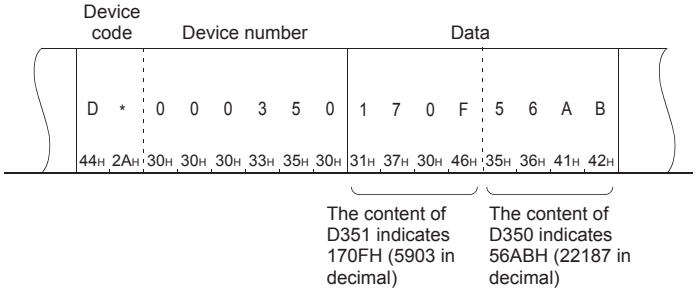
Ex.

When indicating ON/OFF status of 32 points from M16



Ex.

When indicating the stored data of D350 (D351)

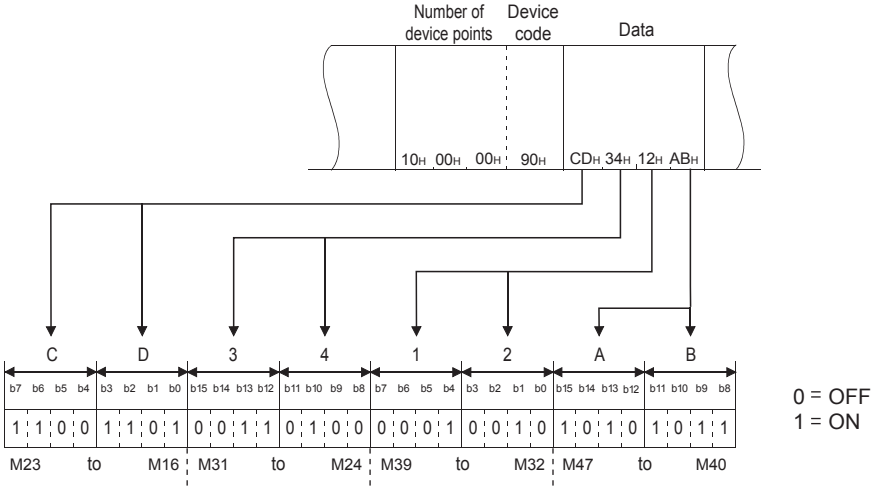


■Data communication using binary code

Send the numerical value in order from the lower byte (L: bit 0 to 7) by handling 32 points unit as 4 bytes.

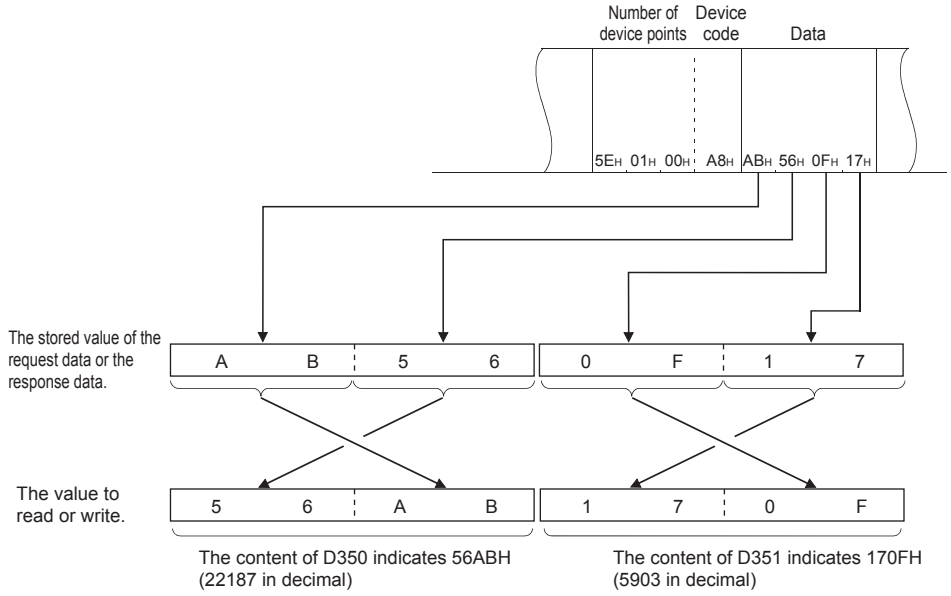
Ex.

When indicating ON/OFF status of 32 points from M16



Ex.

When indicating the stored data of D350 (D351)



Considerations for handling real number data and character string data

The word data and double word data are handled as integer value (16-bit data or 32-bit data).

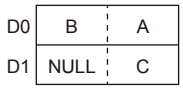
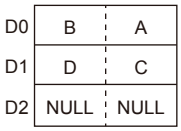
When data other than integer (real number, character string) is stored in a device, the stored value is read as integer value.

- When real number (0.75) is stored in D0 and D1: D0 = 0000H, D1 = 3F40H
- When character string ('12AB') is stored in D2 and D3: D2 = 3231H, D3 = 4241H

For data to be used as real number or character string data in the instructions of the programmable controller, write it to the device/label according to the defined data specification method. For more details on how to specify data used in instructions, refer to the MELSEC iQ-F FX5 Programming Manual (Instructions, Standard Functions/Function Blocks).

■For character string data

The following shows how character string data is stored.

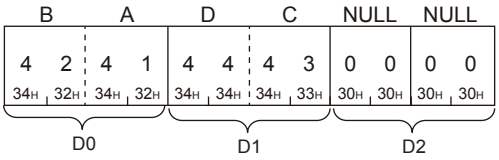
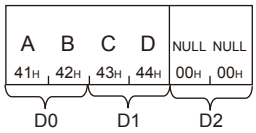
Item	For ASCII code character string	
Character string to be stored	'ABC'	'ABCD'
Character code	'41H', '42H', '43H'	'41H', '42H', '43H', '44H'
Image when character string data is stored from D0	NULL indicates 00H. 	NULL indicates 00H. 

Ex.

Write ASCII code character string data used in the instructions which handle character strings to word device

Store the character string ('ABCD') to D0 and D1: D0 = 4241H ('BA'), D1 = 4443H ('DC')

Specify the following data for write data.

ASCII code	Binary code
	

Point

When communicating ASCII code character string data in ASCII code, data is rearranged every two characters and stored.

Device memory extension specification (subcommand: bit7)

For details, refer to ☞ Page 112 Device Memory Extension Specification.

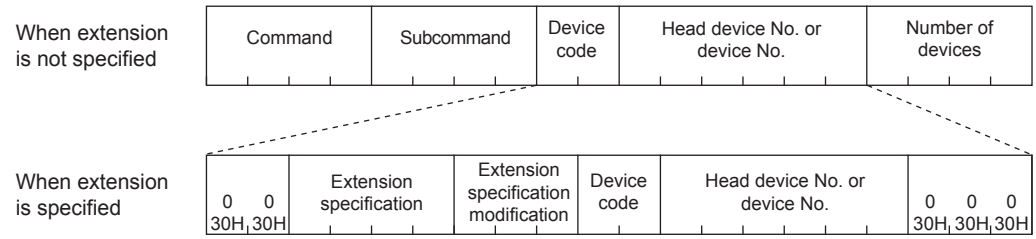
This section explains how to read or write from/to a device to/from module access device areas and how to specify a device indirectly by using index register.

❶Message format

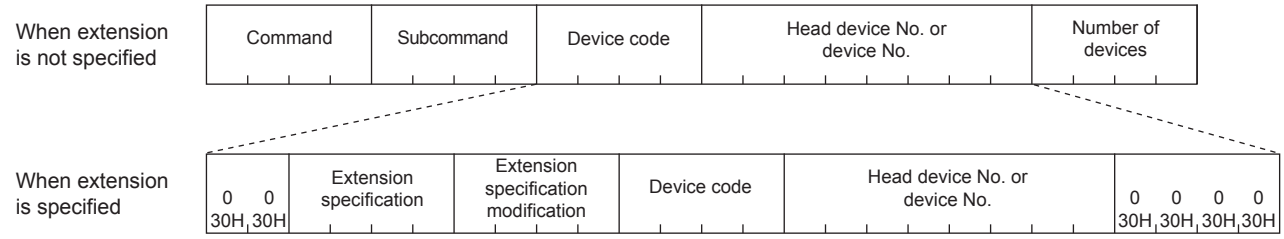
Response messages are extended as well.

■When communicating data in ASCII code

2 digit code/6 digit number specification

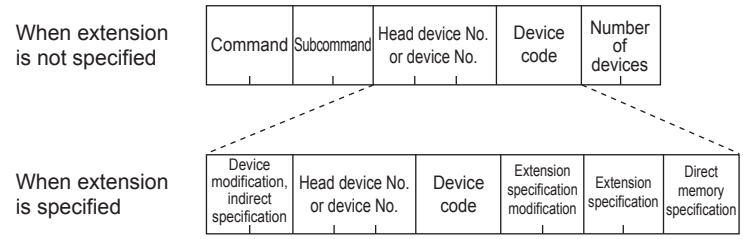


4 digit code/8 digit number specification

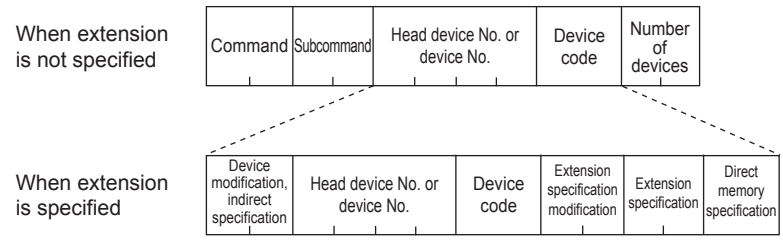


■When communicating data in binary code

2 digit code/6 digit number specification

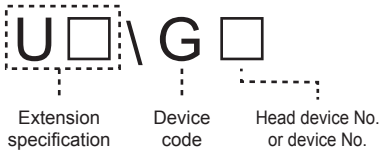


4 digit code/8 digit number specification



❷Module access device specification

The following shows the approach for module access device specification in programming and request data.



■Extension specification

Specify the module number of intelligent function modules.

ASCII code	Binary code
Specify the module number in hexadecimal (3-digit ASCII code). When described with 4-digits, specify the module number with the upper 3-digits. Example 001 U □ □ □ (55H) U 0 0 1 (55H 30H 30H 31H)	Specify the module number in hexadecimal (2 bytes). When described with 4-digits, specify the module number with the upper 3-digits. Example 001 □□H □□H 01H 00H

■Device code

Specify the module access device in the device code list.

■Head device No. or device No.

The format is the same as the message when extension is not specified.

■Direct memory specification (only when communicating in binary code)

The type (intelligent function module device) of access device is specified.

Module access device: F8H is specified

Set/reset

Specify the ON/OFF status of bit device.

• For ON: '1'

Device specification format of subcommand	ASCII code	Binary code
2 digit code/6 digit number specification	0 1 (30H 31H)	01H
4 digit code/8 digit number specification	0 0 0 1 (30H 30H 30H 31H)	01H 00H

• For OFF: '0'

Device specification format of subcommand	ASCII code	Binary code
2 digit code/6 digit number specification	0 0 (30H 30H)	00H
4 digit code/8 digit number specification	0 0 0 0 (30H 30H 30H 30H)	00H 00H

Device Read (Batch)

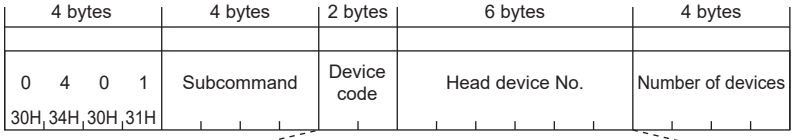
Data in devices are read in a batch.

Request data

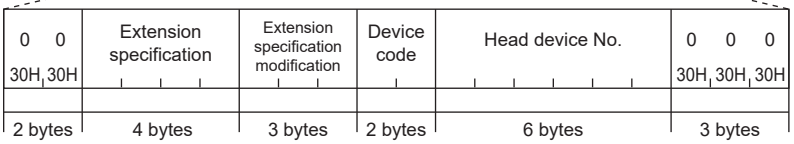
■When communicating data in ASCII code

2 digit code/6 digit number specification

When extension is not specified

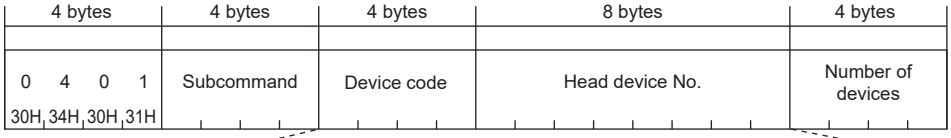


When extension is specified

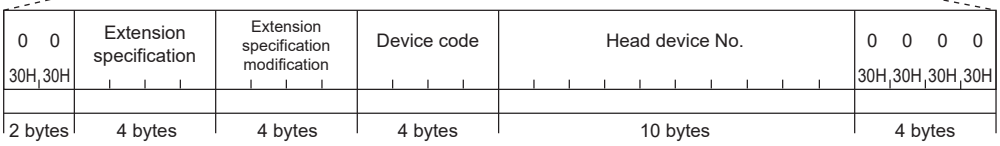


4 digit code/8 digit number specification

When extension is not specified



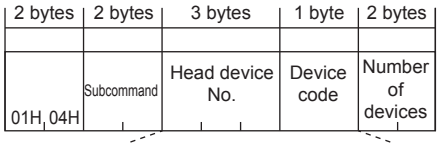
When extension is specified



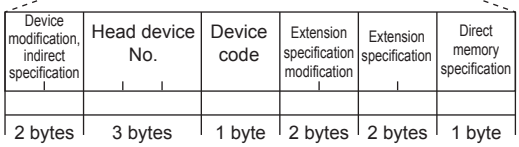
■When communicating data in binary code

2 digit code/6 digit number specification

When extension is not specified

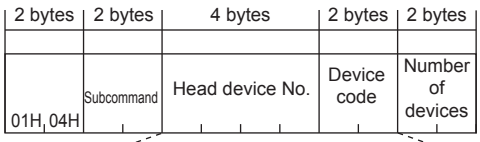


When extension is specified

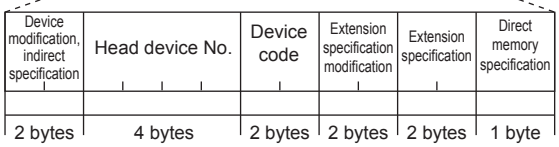


4 digit code/8 digit number specification

When extension is not specified



When extension is specified



■Subcommand

Specify the subcommand selected from the item.

Item			Subcommand					
Data size specification	Device specification format	Device memory extension specification	ASCII code (Upper column: characters, lower column: character code)				Binary code	
Bit units	2 digit code/6 digit number specification	Not specified	0	0	0	1	01H	00H
			30H	30H	30H	31H		
		Specified	0	0	8	1	81H	00H
			30H	30H	38H	31H		
	4 digit code/8 digit number specification	Specified	0	0	8	3	83H	00H
			30H	30H	38H	33H		
Word units	2 digit code/6 digit number specification	Not specified	0	0	0	0	00H	00H
			30H	30H	30H	30H		
		Specified	0	0	8	0	80H	00H
			30H	30H	30H	30H		
	4 digit code/8 digit number specification	Specified	0	0	8	2	82H	00H
			30H	30H	38H	32H		

■Device code

Specify the device code that corresponds to the device type to be read. Refer to the device code list (☞ Page 37).

The double word device and the long index register (LZ) are not supported.

■Device No.

Specify the head number of target device of reading.

■Number of devices

Specify the number of target device points of reading.

Item	Number of devices	
	ASCII code	Binary code
When reading data in bit units	1 to 3584 points	1 to 3584 points
When reading data in word units	1 to 960 points	1 to 960 points

Response data

The read device value is stored in hexadecimal. The data order differs depending on the type of code, ASCII code or binary code.

Read data

Communication example

■When reading data in bit units

M100 to M107 are read.

• When communicating data in ASCII code

(Request data)

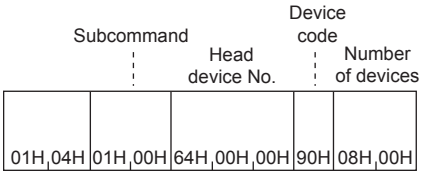
	Subcommand	Device code	Head device No.	Number of devices
0 4 0 1	0 0 0 1	M *	0 0 0 1 0 0	0 0 0 8
30H 34H 30H 31H	30H 30H 30H 31H	4DH 2AH	30H 30H 30H 31H 30H 30H	30H 30H 30H 38H

(Response data)

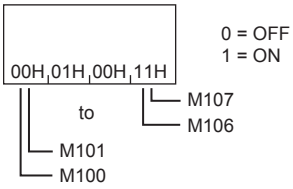
0 0 0 1 0 0 1 1	0 = OFF 1 = ON
30H 30H 30H 31H 30H 30H 31H 31H	
M100 to M107	

• When communicating data in binary code

(Request data)



(Response data)

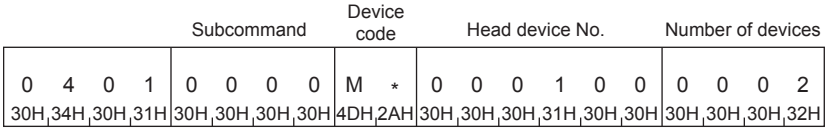


■When reading data in word units (bit device)

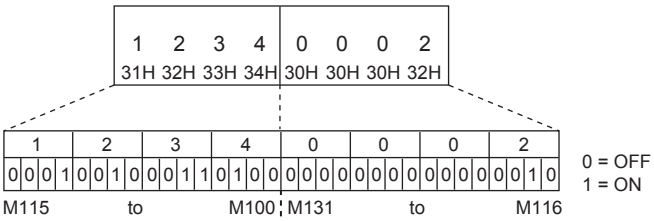
M100 to M131 (2-word) are read.

• When communicating data in ASCII code

(Request data)

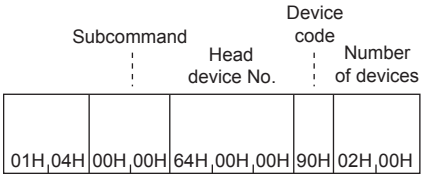


(Response data)

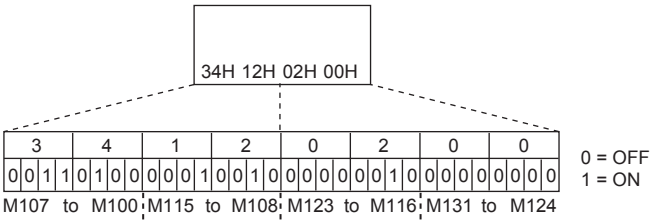


• When communicating data in binary code

(Request data)



(Response data)



■When reading data in word units (word device)

Values in T100 to T102 are read.

It is supposed that 4660 (1234H) is stored in T100, 2 (2H) is stored in T101, and 7663 (1DEFH) is stored T102.

• When communicating data in ASCII code

(Request data)

	Subcommand	Device code	Head device No.	Number of devices
0 4 0 1	0 0 0 0	T N	0 0 0 1 0 0	0 0 0 3
30H 34H 30H 31H	30H 30H 30H 30H	54H 4EH	30H 30H 30H 31H 30H 30H	30H 30H 30H 33H

(Response data)

1 2 3 4	0 0 0 2	1 D E F
31H 32H 33H 34H	30H 30H 30H 32H	31H 44H 45H 46H
T100	T101	T102

• When communicating data in binary code

(Request data)

Subcommand	Head device No.	Device code	Number of devices
01H 04H	00H 00H	64H 00H 00H	C2H	03H 00H

(Response data)

34H 12H	02H 00H	EFH 1DH
T100	T101	T102

Device Write (Batch)

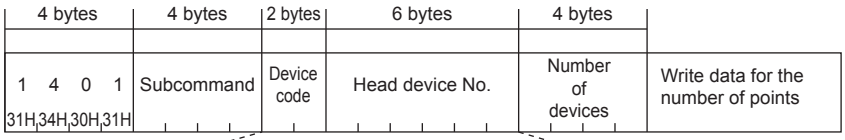
Data in devices are written in a batch.

Request data

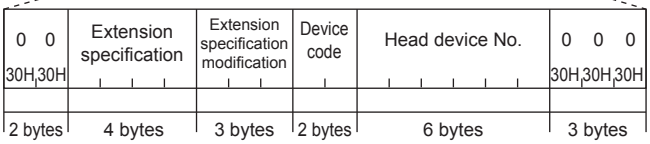
■When communicating data in ASCII code

2 digit code/6 digit number specification

When extension is not specified

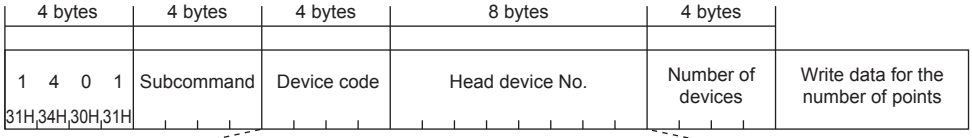


When extension is specified

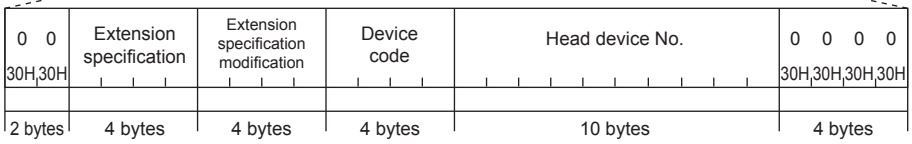


4 digit code/8 digit number specification

When extension is not specified



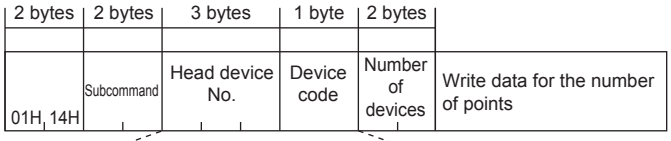
When extension is specified



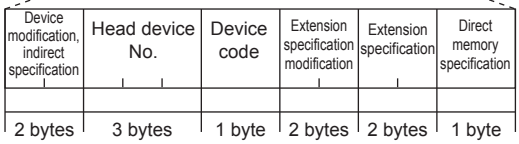
■When communicating data in binary code

2 digit code/6 digit number specification

When extension is not specified

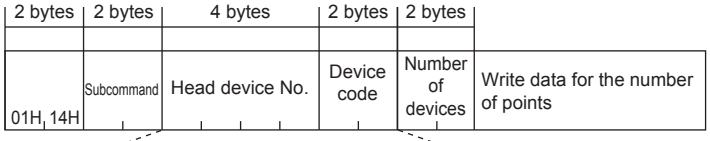


When extension is specified

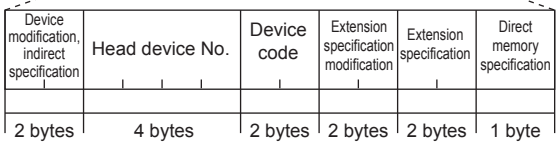


4 digit code/8 digit number specification

When extension is not specified



When extension is specified



■Subcommand

Specify the subcommand selected from the item.

Item			Subcommand					
Data size specification	Device specification format	Device memory extension specification	ASCII code (Upper column: characters, lower column: character code)				Binary code	
Bit units	2 digit code/6 digit number specification	Not specified	0	0	0	1	01H	00H
			30H	30H	30H	31H		
		Specified	0	0	8	1	81H	00H
			30H	30H	38H	31H		
	4 digit code/8 digit number specification	Specified	0	0	8	3	83H	00H
			30H	30H	38H	33H		
Word units	2 digit code/6 digit number specification	Not specified	0	0	0	0	00H	00H
			30H	30H	30H	30H		
		Specified	0	0	8	0	80H	00H
			30H	30H	38H	30H		
	4 digit code/8 digit number specification	Specified	0	0	8	2	82H	00H
			30H	30H	38H	32H		

■Device code

Specify the device code that corresponds to the device type to be written. Refer to the device code list (☞ Page 37).

The double word device and the long index register (LZ) are not supported.

■Device No.

Specify the head number of target device of writing.

■Number of devices

Specify the number of target device points of writing.

Item	Number of devices	
	ASCII code	Binary code
When writing data in bit units	1 to 3584 points	1 to 3584 points
When writing data in word units	1 to 960 points	1 to 960 points

■Write data

Specify value to be written to a device for the number of points specified in "Device point".

Response data

There is no response data for the Device Write command.

Communication example

■When writing data in bit units

Values are written to M100 to M107.

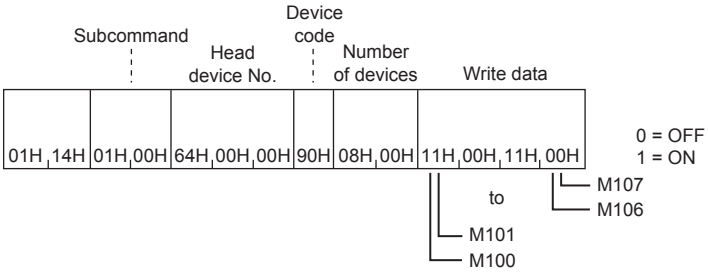
• When communicating data in ASCII code

(Request data)

	Subcommand	Device code	Head device No.	Number of devices	Write data
1 4 0 1	0 0 0 1	M *	0 0 0 1 0 0	0 0 0 8	1 1 0 0 1 1 0 0
31H 34H 30H 31H	30H 30H 30H 31H	4DH 2AH	30H 30H 30H 31H 30H 30H	30H 30H 30H 38H	31H 31H 30H 30H 31H 31H 30H 30H
					M100 to M107 0 = OFF 1 = ON

• When communicating data in binary code

(Request data)

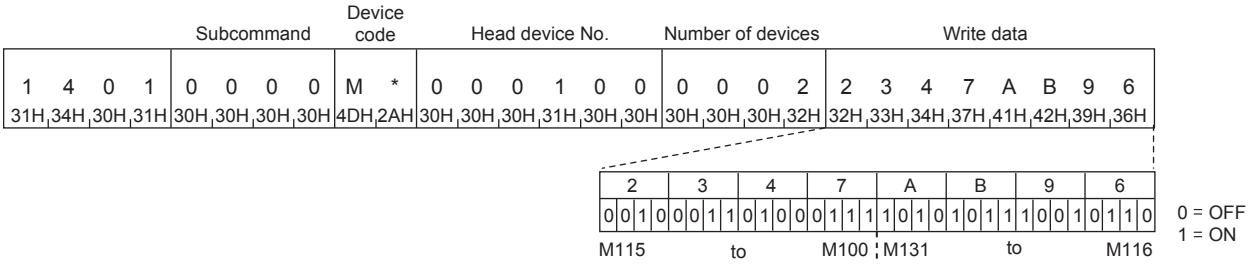


■When writing data in word units (bit device)

Values are written to M100 to M131 (2-word).

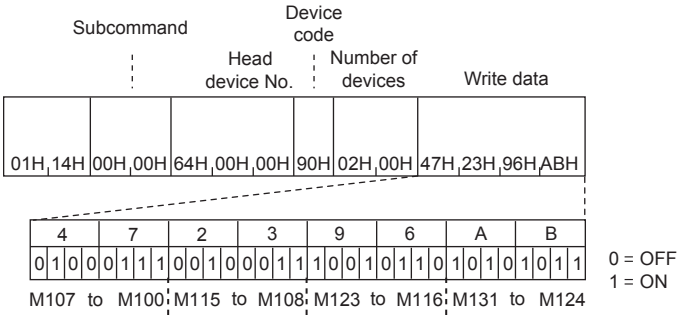
• When communicating data in ASCII code

(Request data)



• When communicating data in binary code

(Request data)

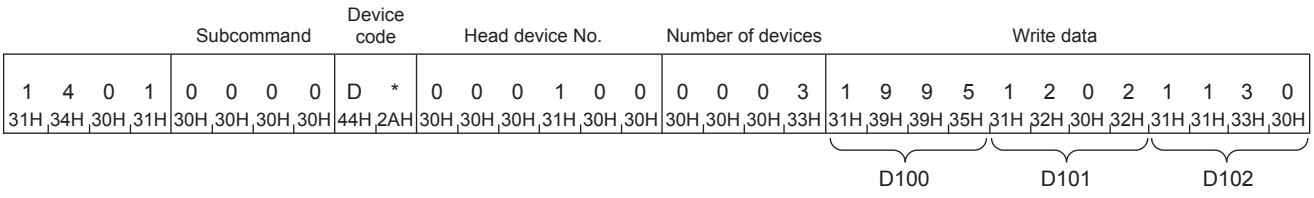


■When writing data in word units (word device)

6549 (1995H) is written in D100, 4610 (1202H) is written in D101, and 4400 (1130H) is written in D102.

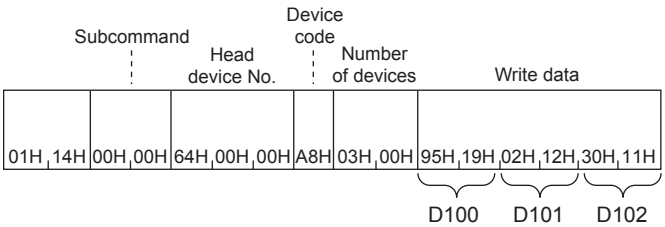
• When communicating data in ASCII code

(Request data)



• When communicating data in binary code

(Request data)



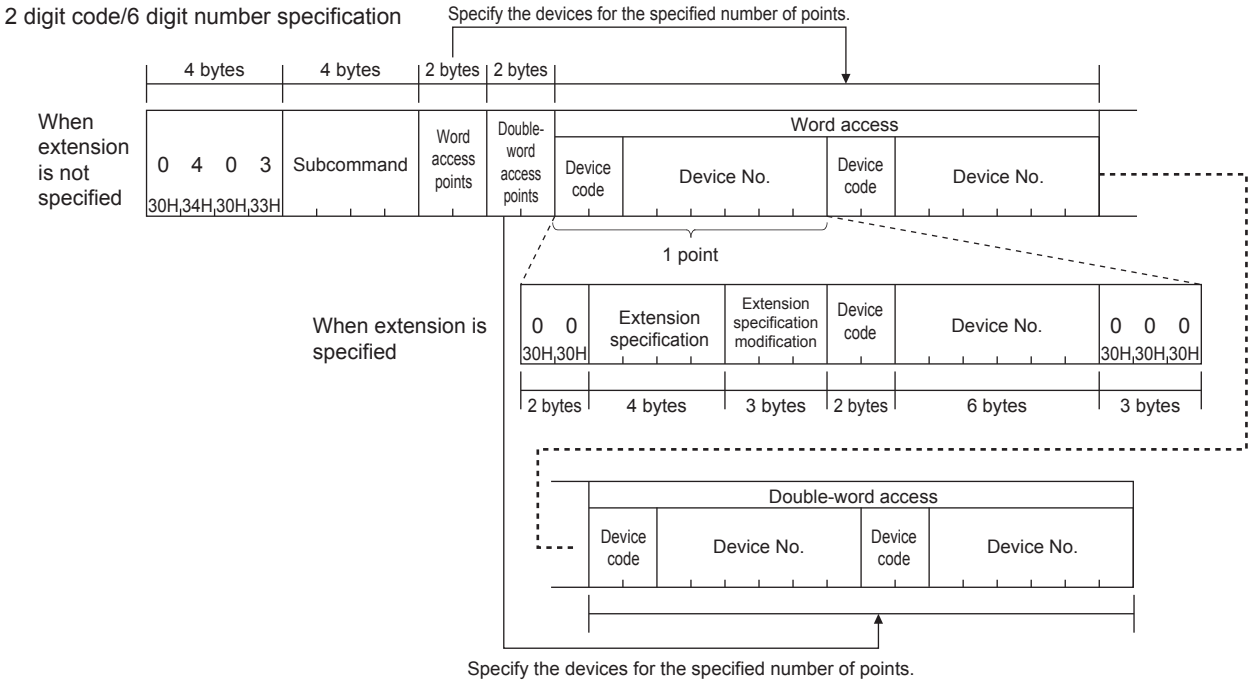
Random read

This command specifies the device No. randomly and reads the device value.

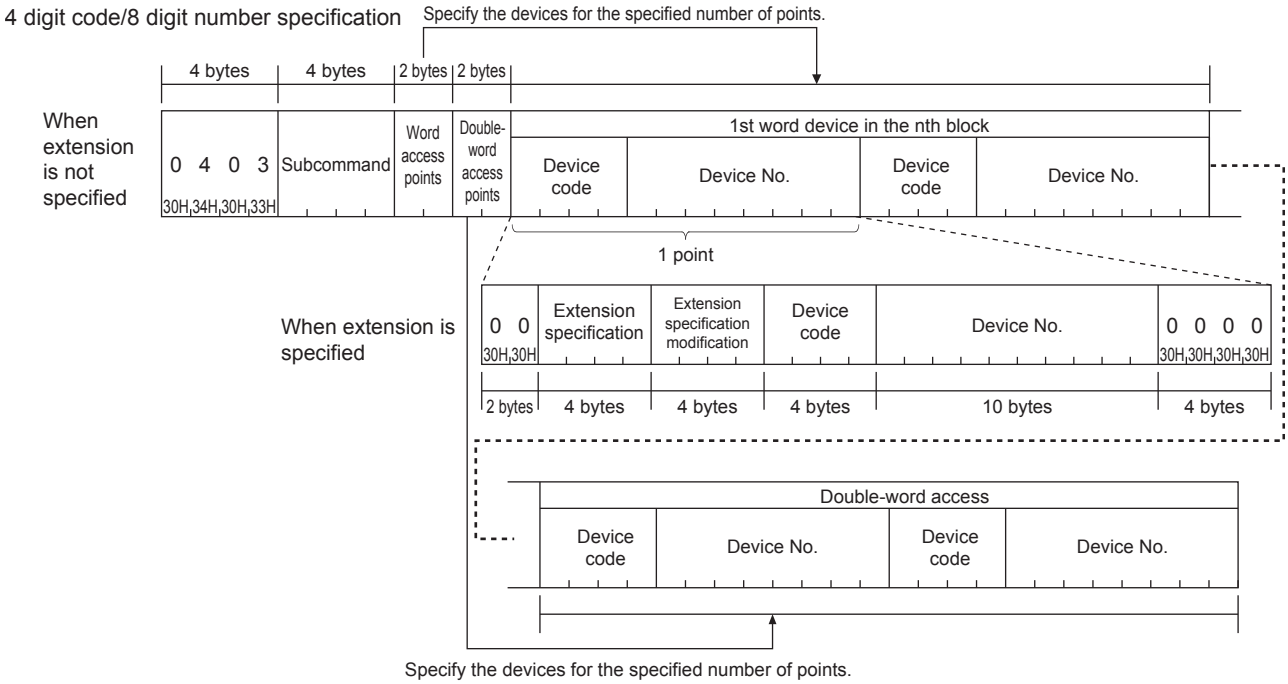
Request data

■When communicating data in ASCII code

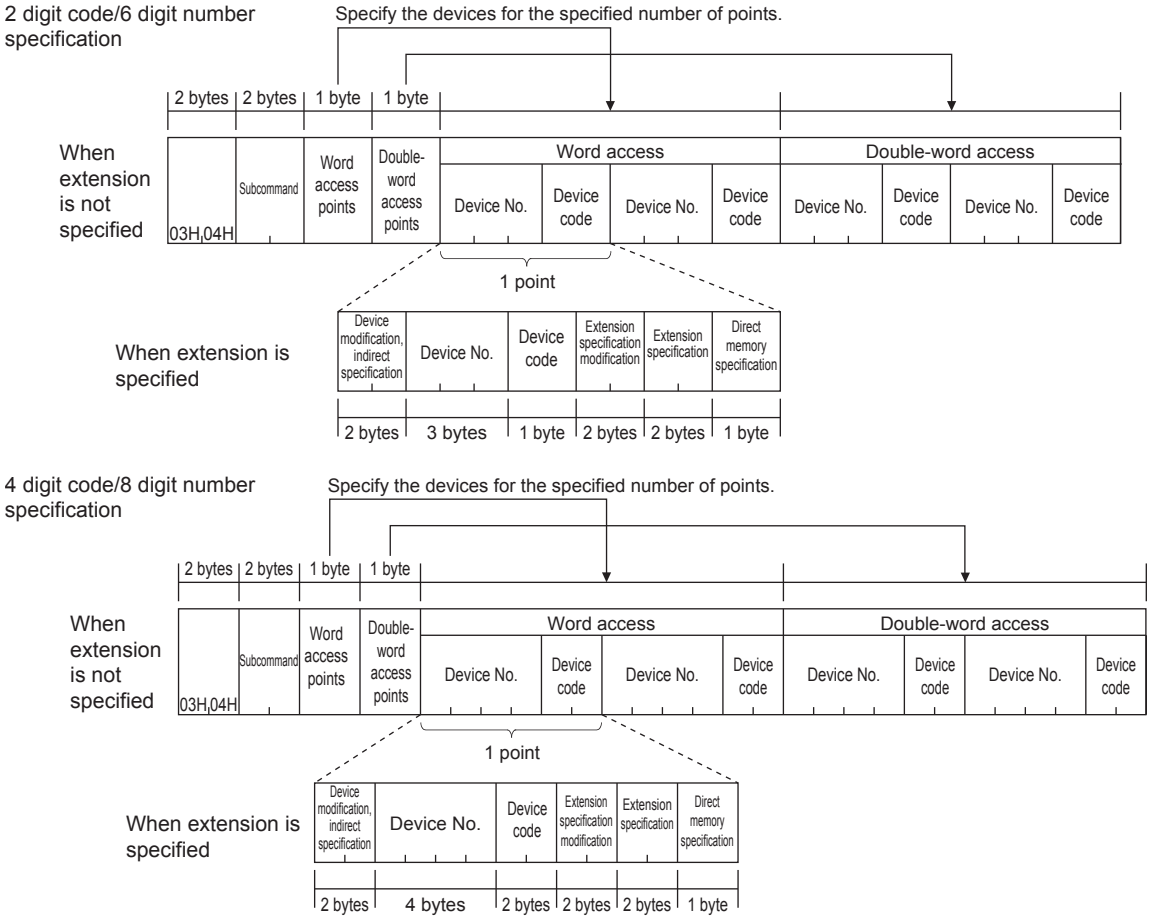
2 digit code/6 digit number specification



4 digit code/8 digit number specification



■When communicating data in binary code



■Subcommand

Specify the subcommand selected from the item.

Item			Subcommand					
Data size specification	Device specification format	Device memory extension specification	ASCII code (Upper column: characters, lower column: character code)				Binary code	
Word units	2 digit code/6 digit number specification	Not specified	0	0	0	0	00H	00H
			30H	30H	30H	30H		
		Specified	0	0	8	0	80H	00H
			30H	30H	38H	30H		
	4 digit code/8 digit number specification	Specified	0	0	8	2	82H	00H
			30H	30H	38H	32H		

■Word access points, double-word access points

Specify the number of target device points of reading.

Item	Description	Number of points	
		ASCII code	Binary code
Word access points	Specify the number of points to be accessed in one-word units. The bit device is 16-point units, the word device is one-word units.	1 ≤ word access points + double-word access points ≤ 192 When device memory extension specification is used, double the number of the access points.	
Double-word access points	Specify the number of points to be accessed in two-word units. The bit device is 32-point units, the word device is two-word units.		

■Device code, device No.

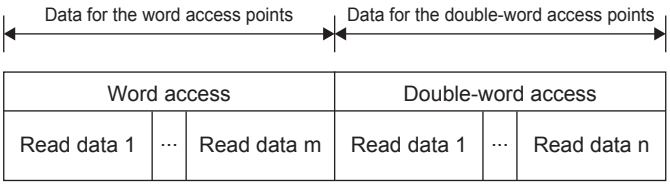
Specify the target device of reading.

Item	Description
Word access	Specify the device points specified as word access points. The specification is not necessary when the word access points are zero.
Double-word access	Specify the device points specified as double-word access points. The specification is not necessary when the double-word access points are zero.

Set up in order of word access device → double word access device.

Response data

The read device value is stored in hexadecimal. The data order differs depending on the type of code, ASCII code or binary code.



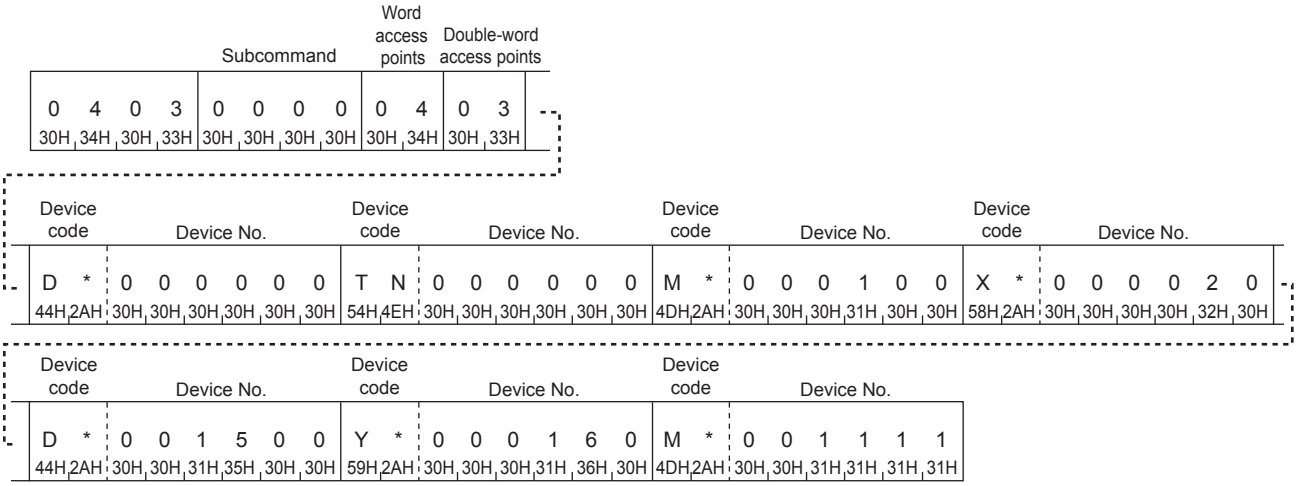
Communication example

Read D0, T0, M100 to M115, X20 to X37 by word access, and D1500 to D1501, Y160 to Y217, M1111 to M1142 by double-word access.

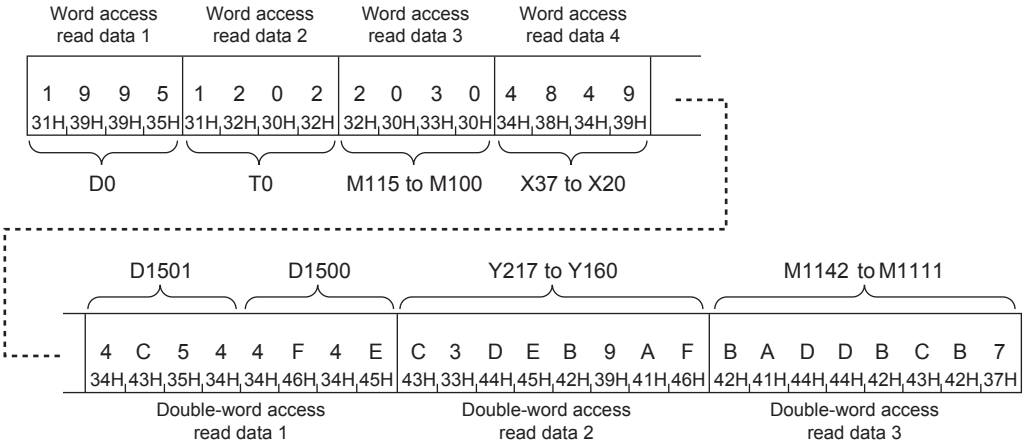
It is supposed that 6549 (1995H) is stored in D0, 4610 (1202H) is stored in T0, 20302 (4F4EH) is stored in D1500, 19540 (4C54H) is stored in D1501.

■When communicating data in ASCII code (X, Y OCT)

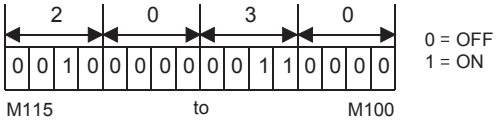
• Request data



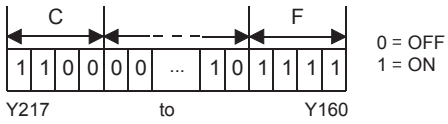
• Response data



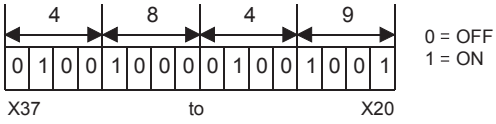
Word access read data 3



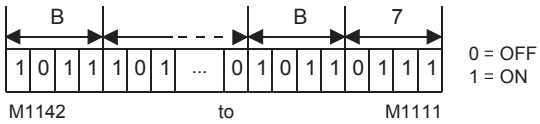
Double-word access read data 2



Word access read data 4

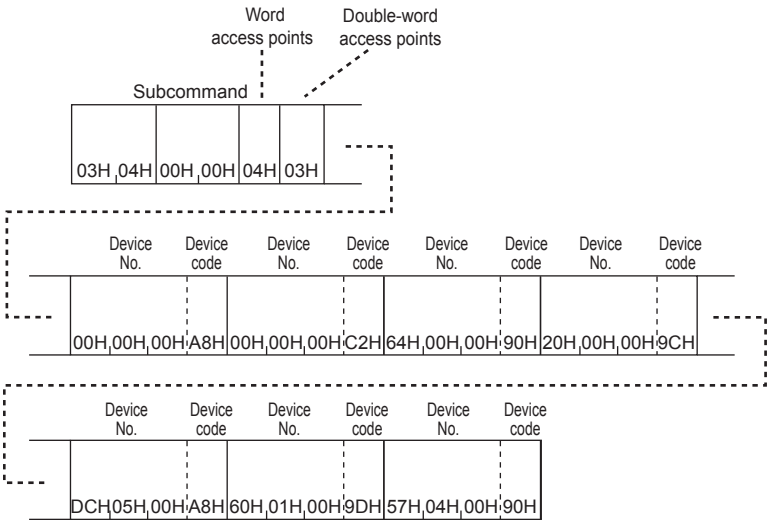


Double-word access read data 3

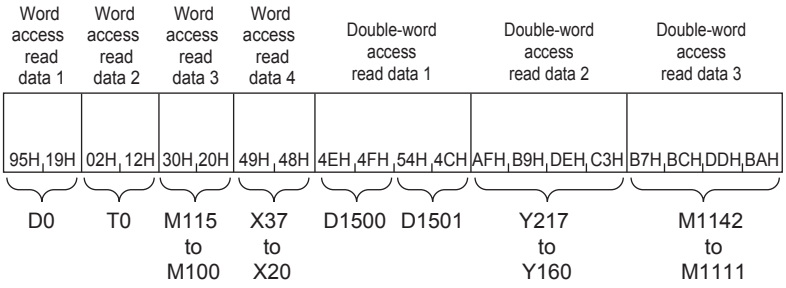


■When communicating data in binary code

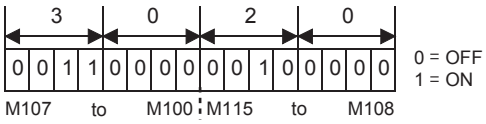
• Request data



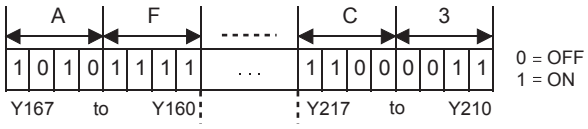
• Response data



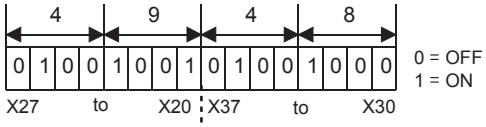
Word access read data 3



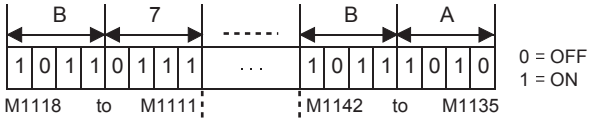
Double-word access read data 2



Word access read data 4



Double-word access read data 3



Random write

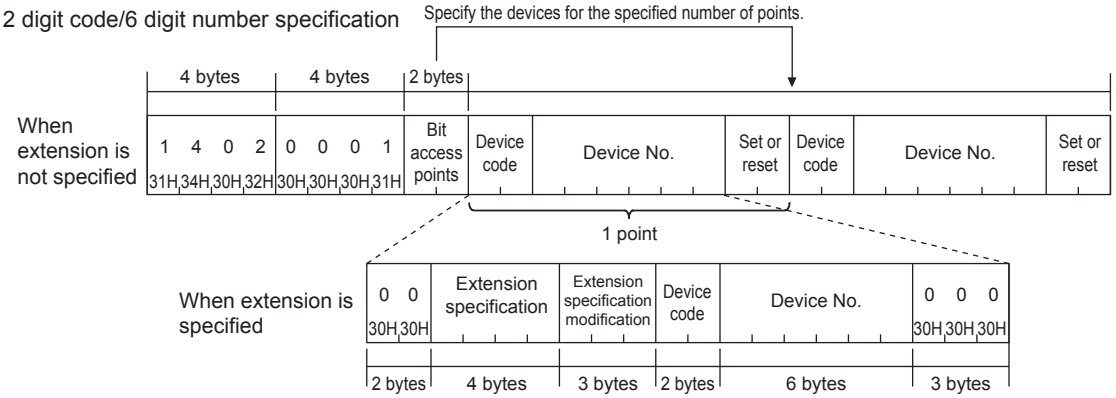
This command specifies the device No. randomly and writes the data.

Request data

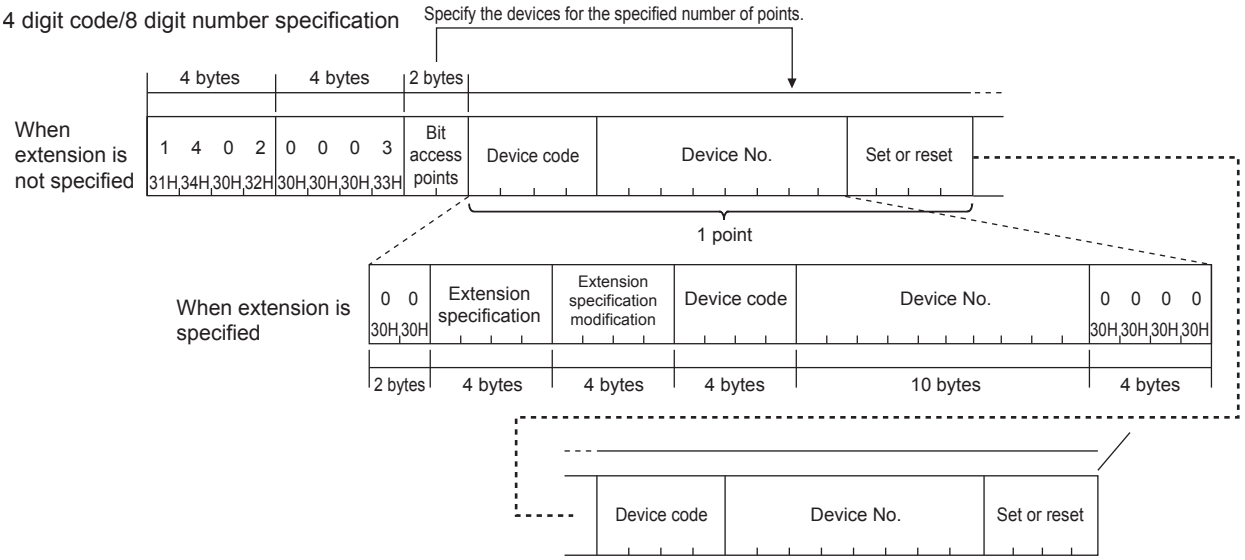
■When writing data in bit units

- When communicating data in ASCII code

2 digit code/6 digit number specification

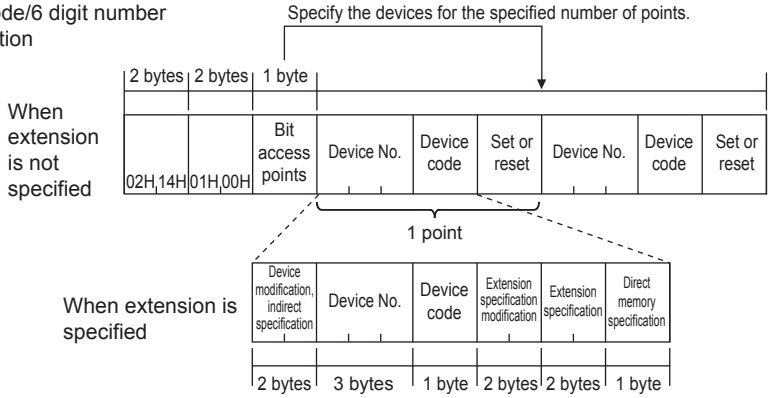


4 digit code/8 digit number specification

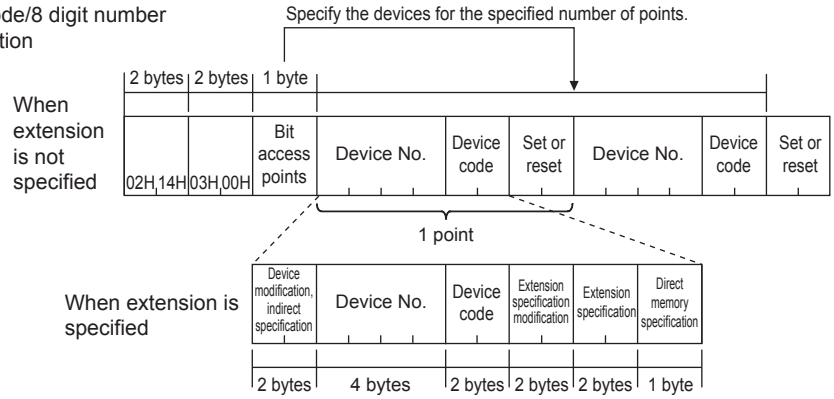


• When communicating data in binary code

2 digit code/6 digit number specification

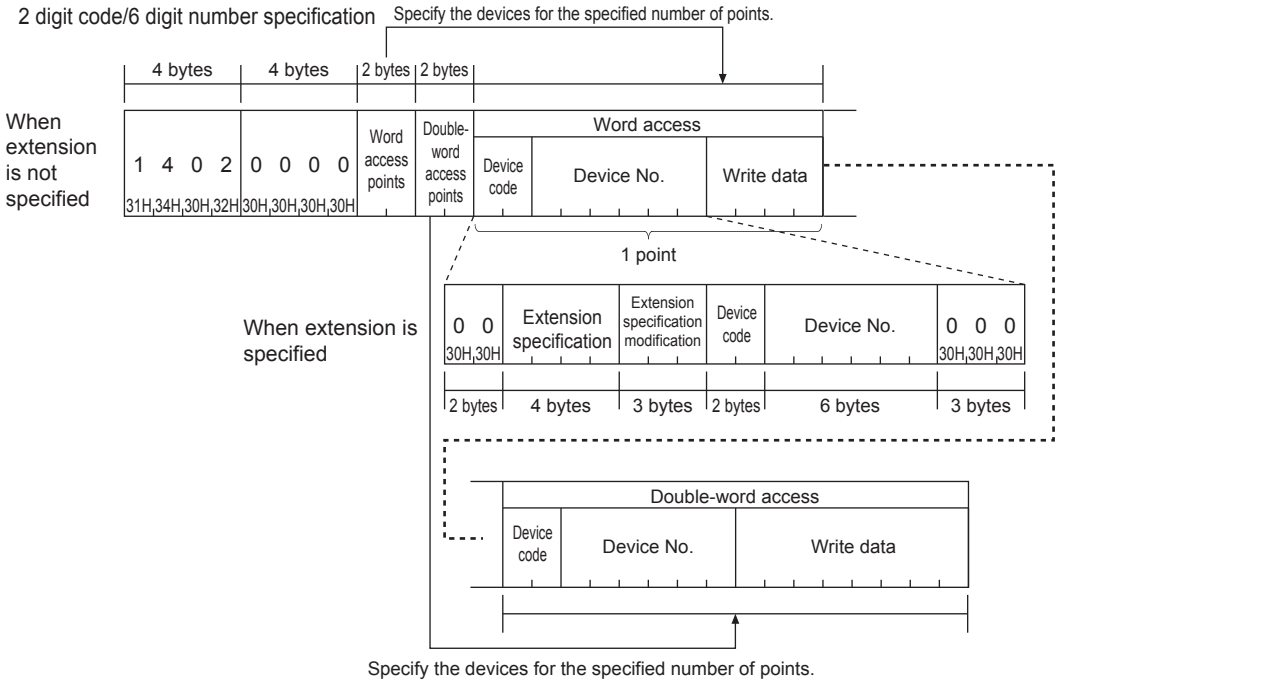


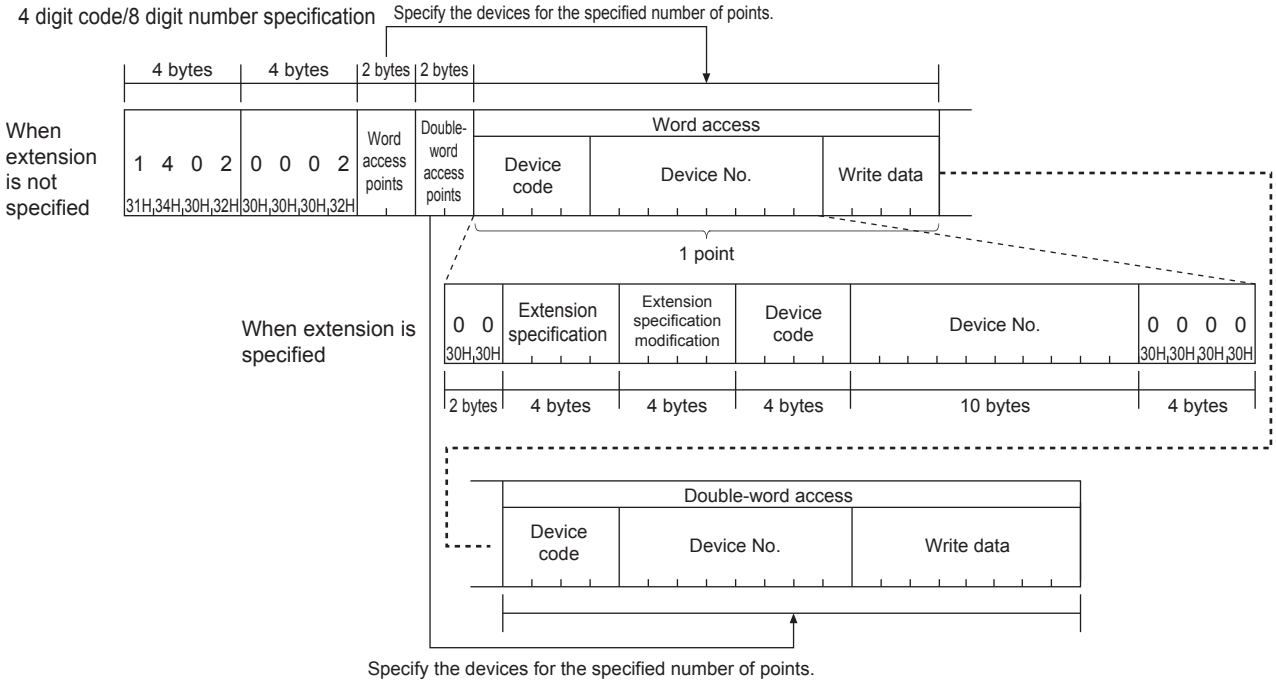
4 digit code/8 digit number specification



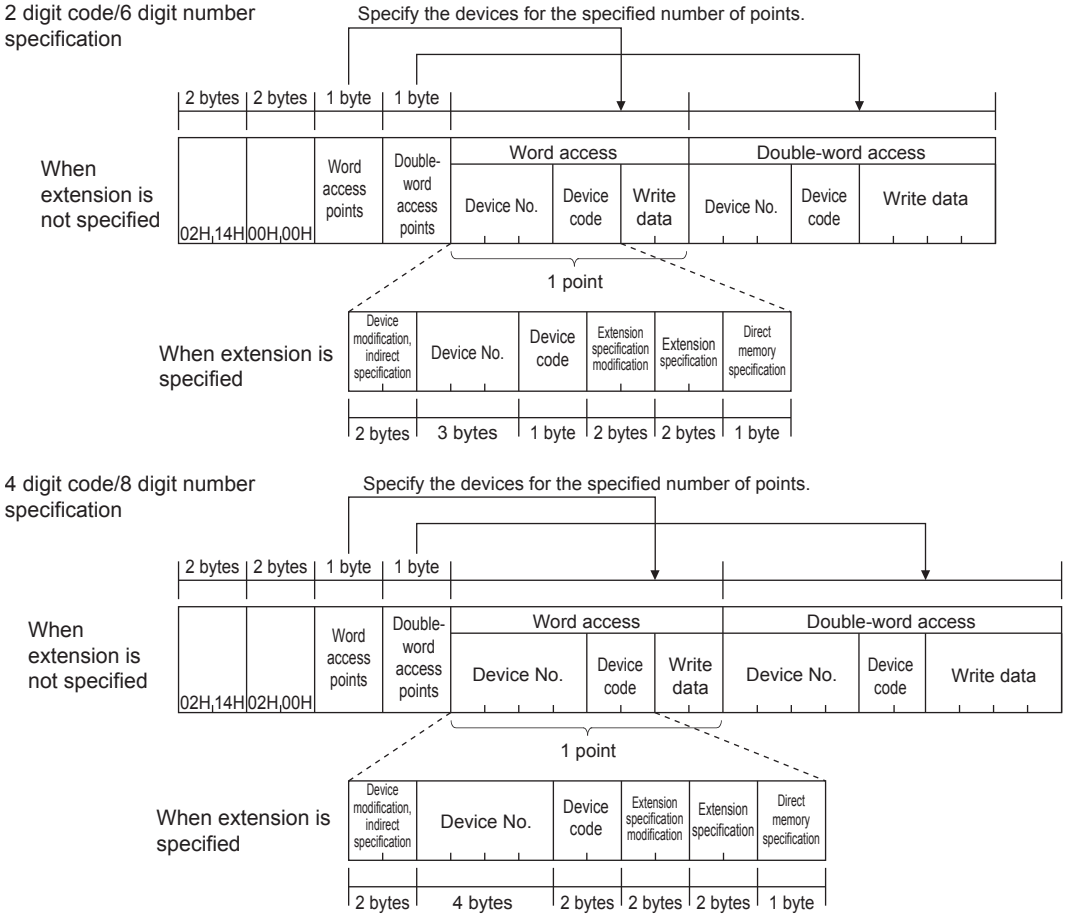
■When writing data in word units

• When communicating data in ASCII code





• When communicating data in binary code



■Subcommand

Specify the subcommand selected from the item.

Item			Subcommand					
Data size specification	Device specification format	Device memory extension specification	ASCII code (Upper column: characters, lower column: character code)				Binary code	
Bit units	2 digit code/6 digit number specification	Not specified	0	0	0	1	01H	00H
			30H	30H	30H	31H		
		Specified	0	0	8	1	81H	00H
			30H	30H	38H	31H		
	4 digit code/8 digit number specification	Specified	0	0	8	3	83H	00H
			30H	30H	38H	33H		
Word units	2 digit code/6 digit number specification	Not specified	0	0	0	0	00H	00H
			30H	30H	30H	30H		
		Specified	0	0	8	0	80H	00H
			30H	30H	38H	30H		
	4 digit code/8 digit number specification	Specified	0	0	8	2	82H	00H
			30H	30H	38H	32H		

■Bit access points, word access points, double-word access points

Item	Description	Number of points	
		ASCII code	Binary code
Bit access points	Specify the number of bit device points in one-point units.	1 to 188 When device memory extension specification is used 1 to 94	

Item	Description	Number of points	
		ASCII code	Binary code
Word access points	Specify the number of points to be accessed in one-word units. The bit device is 16-point units, the word device is one-word units.	1 ≤ word access points × 12 + double-word access points × 14 ≤ 1920 When device memory extension specification is used, double the number of the access points.	
Double-word access points	Specify the number of points to be accessed in two-word units. The bit device is 32-point units, the word device is two-word units.		

■Device code, device No., write data

Specify the target device of writing.

The data is specified in hexadecimal number.

Item	Description
Word access	Specify the device points specified as word access points. The specification is not necessary when the word access points are zero.
Double-word access	Specify the device points specified as double-word access points. The specification is not necessary when the double-word access points are zero.

■Set or reset

Specify ON/OFF of the bit device.

• 2 digit code/6 digit number specification

Item	Data to write		Remark
	ON	OFF	
ASCII code	"01"	"00"	Two characters will be sent in order from "0".
Binary code	01H	00H	The one-byte numerical value shown left will be sent.

• 4 digit code/8 digit number specification

Item	Data to write		Remark
	ON	OFF	
ASCII code	"0001"	"0000"	Four characters will be sent in order from "0".
Binary code	0001H	0000H	The two-byte numerical value shown left will be sent.

Response data

There is no response data for the Random write command.

Communication example

■When writing data in bit units

Turn off M50 and turn on Y27.

• When communicating data in ASCII code (X, Y OCT)

(Request data)

	Subcommand	Bit access points	Device code	Device No.	Set or reset	Device code	Device No.	Set or reset
1 4 0 2	0 0 0 1	0 2	M *	0 0 0 0 5 0	0 0	Y *	0 0 0 0 2 7	0 1
31H 34H 30H 32H	30H 30H 30H 31H	30H 32H	4DH 2AH	30H 30H 30H 30H 35H 30H	30H 30H	59H 2AH	30H 30H 30H 30H 32H 37H	30H 31H

• When communicating data in binary code

(Request data)

Subcommand	Bit access points	Device code	Device No.	Set or reset	Device code	Device No.	Set or reset
02H 14H	01H 00H	02H	32H 00H 00H	90H 00H		27H 00H 00H	9DH 01H

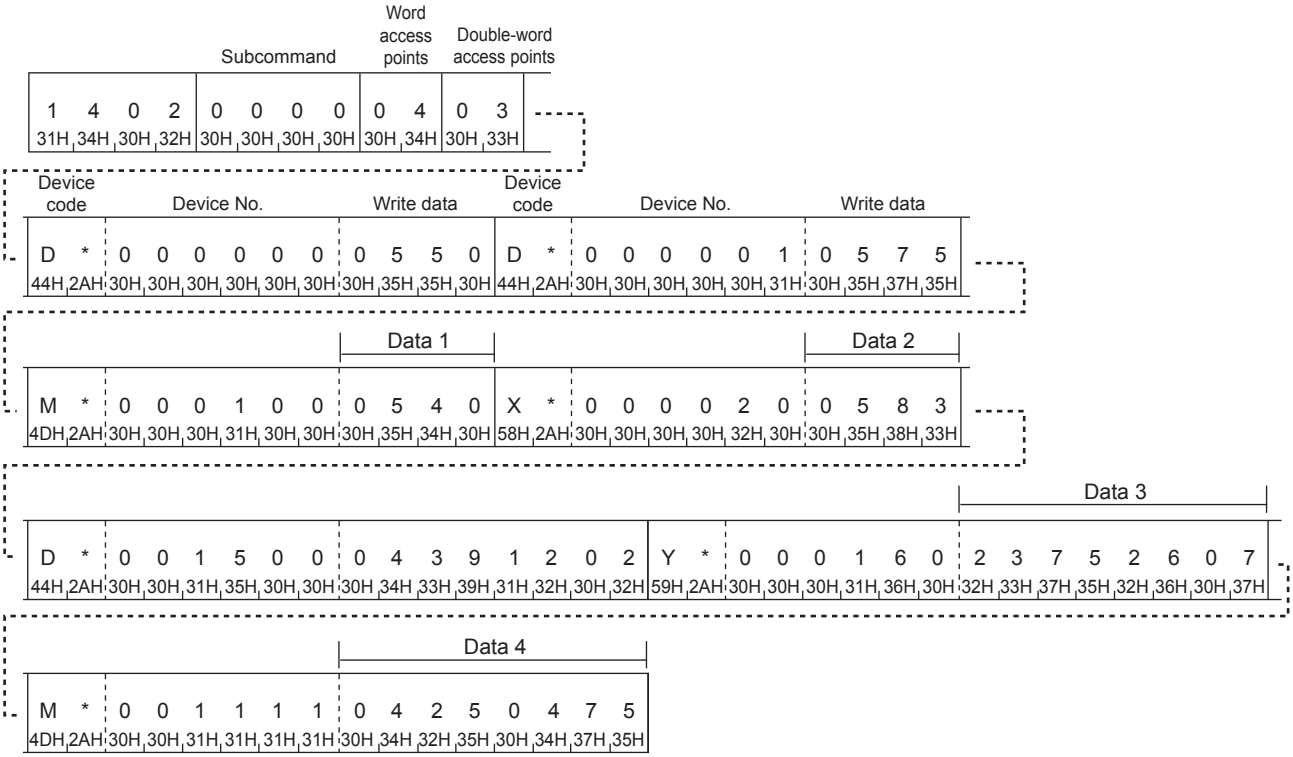
■When writing data in word units

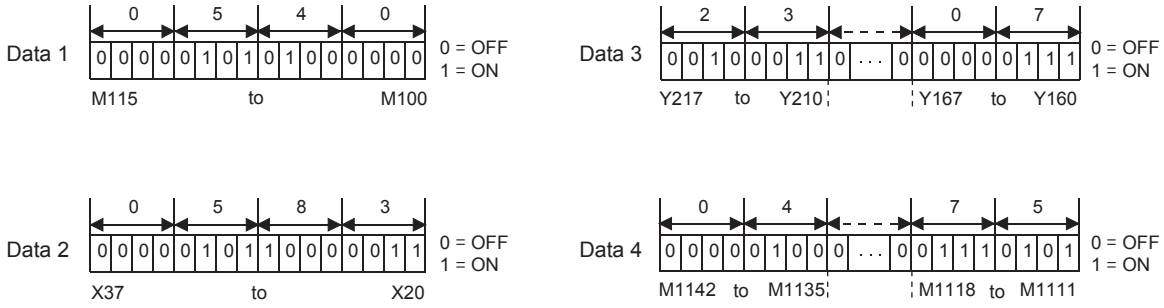
Write the value in a device as follows.

Item	Target device
Word access	D0, D1, M100 to M115, X20 to X37
Double-word access	D1500 to D1501, Y160 to Y217, M1111 to M1142

• When communicating data in ASCII code (X, Y OCT)

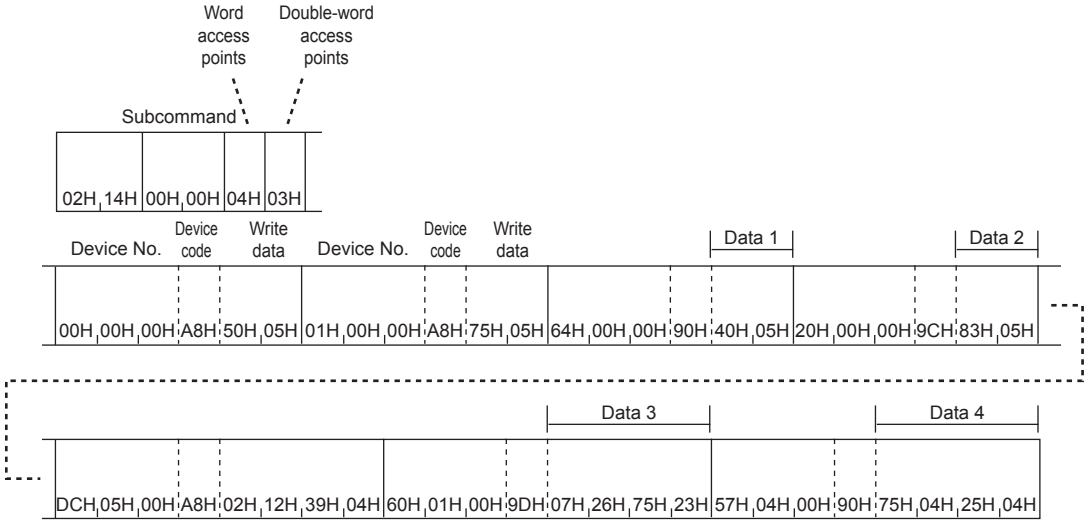
(Request data)

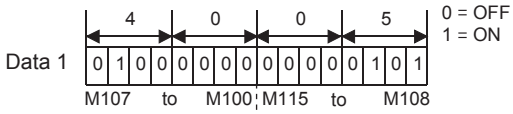


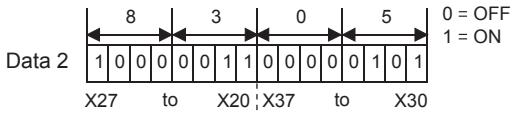


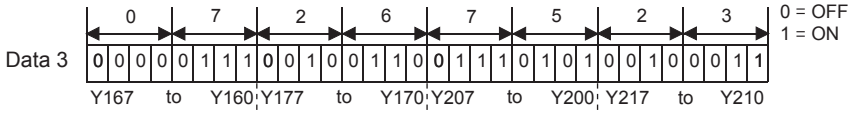
• When communicating data in binary code

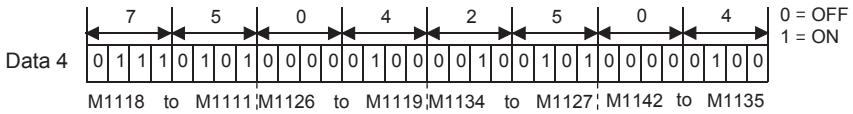
(Request data)











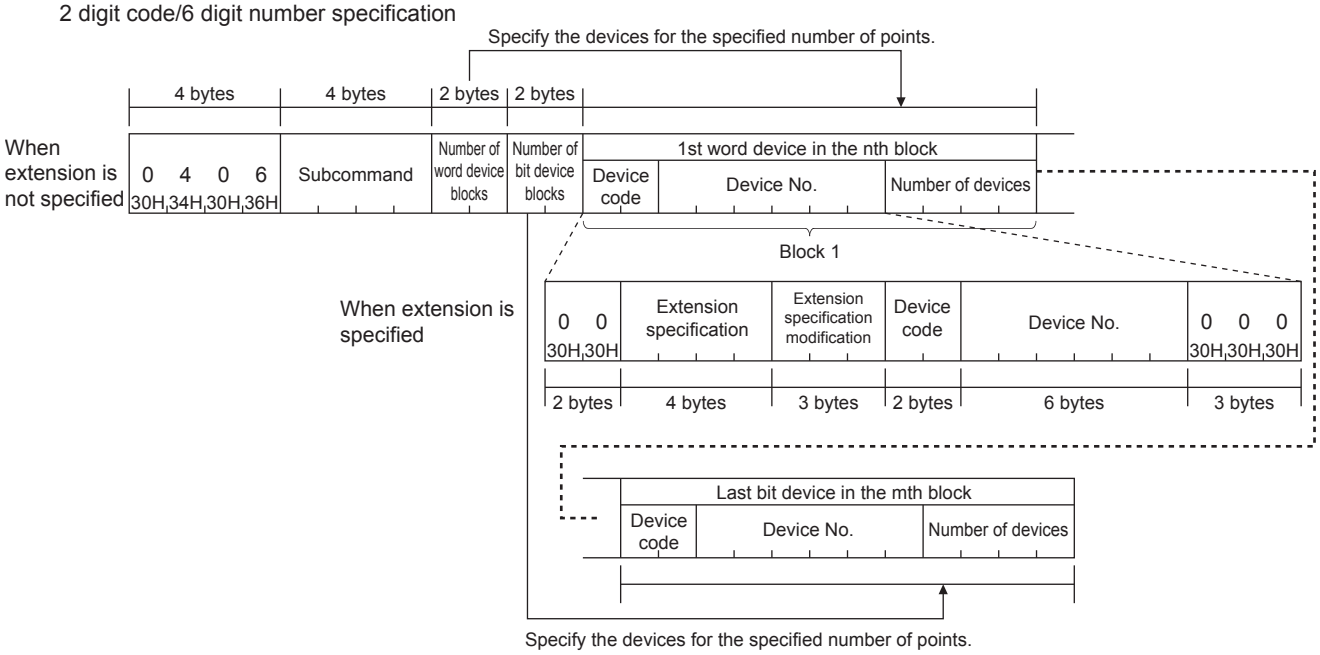
Batch read multiple blocks

The examples shown in this section explain the control procedure for reading by randomly specifying multiple blocks, where 1 block consists of n point(s) of bit device memory (one point is specified by 16-bit) or word device memory (one point is specified by 1-word).

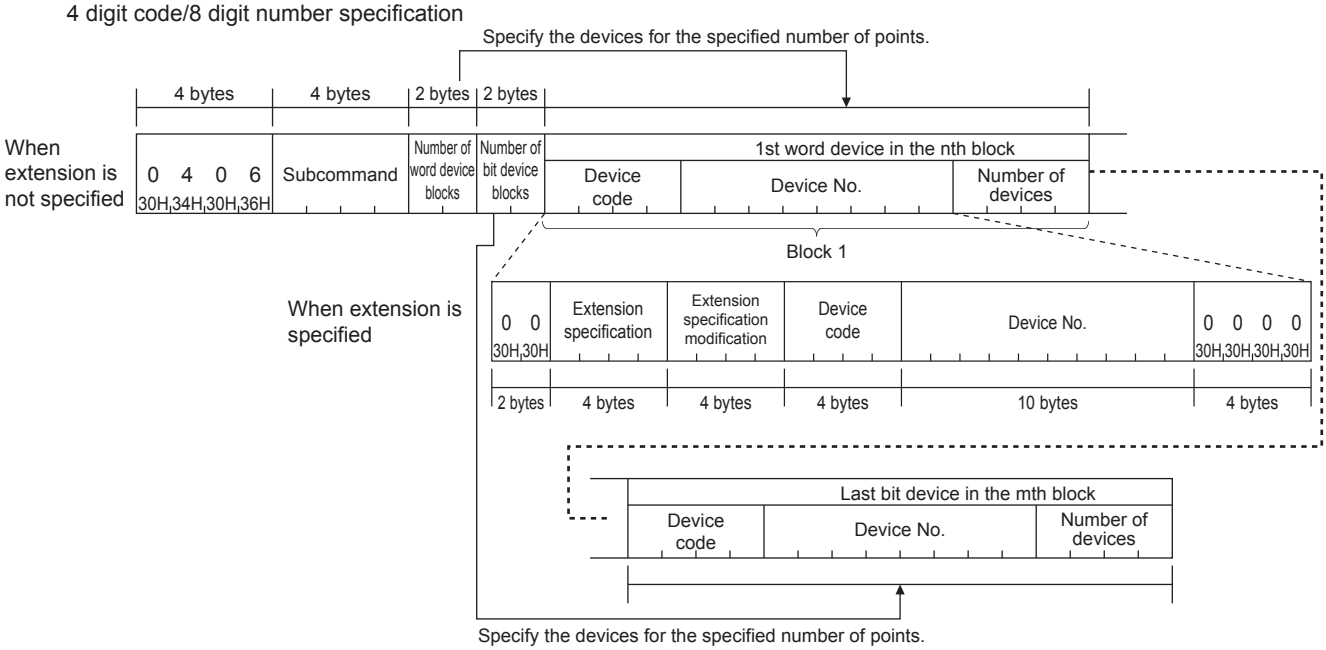
Request data

■When communicating data in ASCII code

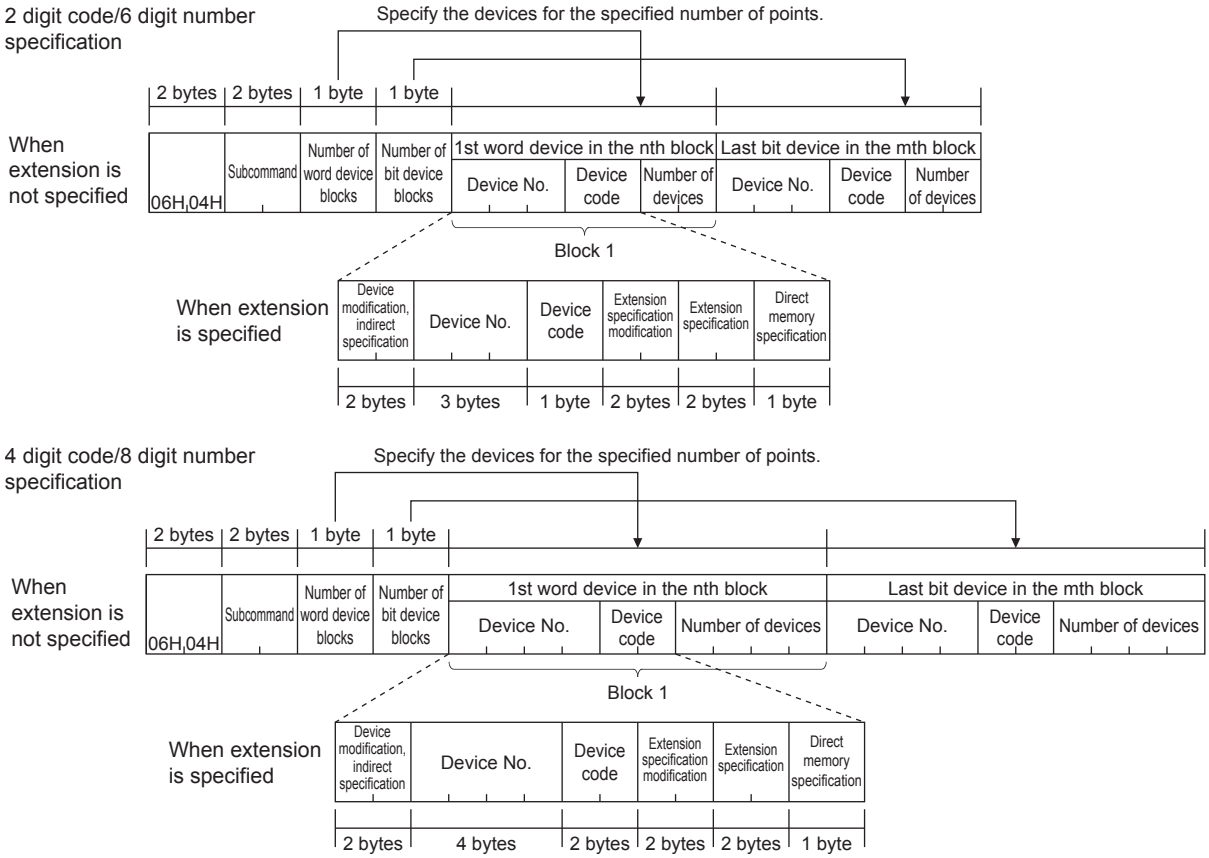
2 digit code/6 digit number specification



4 digit code/8 digit number specification



■When communicating data in binary code



❶Subcommand

Specify the subcommand selected from the item.

Item			Subcommand					
Data size specification	Device specification format	Device memory extension specification	ASCII code (Upper column: characters, lower column: character code)				Binary code	
Word units	2 digit code/6 digit number specification	Not specified	0	0	0	0	00H	00H
			30H	30H	30H	30H		
		Specified	0	0	8	0	80H	00H
			30H	30H	38H	30H		
	4 digit code/8 digit number specification	Specified	0	0	8	2	82H	00H
			30H	30H	38H	32H		

❷Number of word device blocks and number of bit device blocks

Specify the number of blocks of the device to be read in hexadecimal.

Item	Description	Number of points	
		ASCII code	Binary code
Number of word device blocks	Specify the number of blocks of the word device to be read.	Number of word device blocks + number of bit device blocks ≤120 When device memory extension specification is used, double the number of the block points.	
Number of bit device blocks	Specify the number of blocks of the bit device to be read.		

❸Device code, device No., number of device points

Specify the device points while satisfying the following conditions:

Total number of points for all word device blocks + total number of points for all bit device blocks ≤ 960

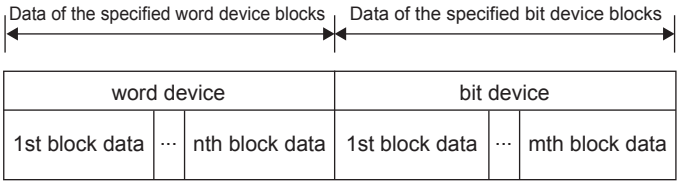
Item	Description
Word device	Specify the device points specified in "Number of word device blocks". When "Number of word device blocks" is set to 0, this specification is unnecessary.
Bit device	Specify the device points specified in "Number of bit device blocks". When "Number of bit device blocks" is set to 0, this specification is unnecessary.

Point

When specifying a contact or coil of a timer, retentive timer, or counter, use the bit device block.

Set up in order of word device → bit device.

Response data



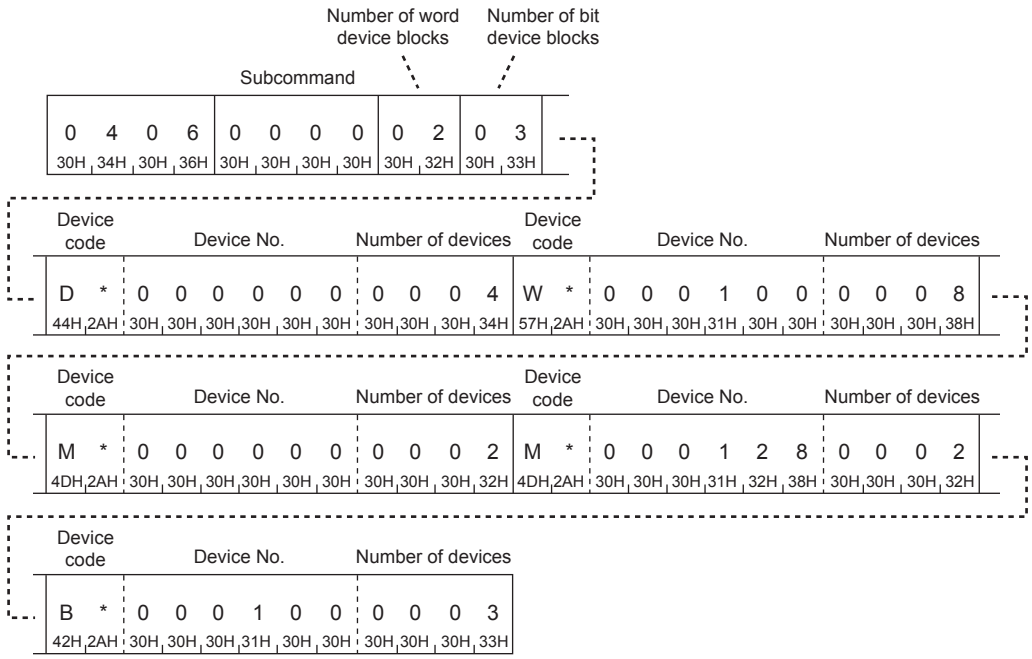
Communication example

Values are read from devices as follows.

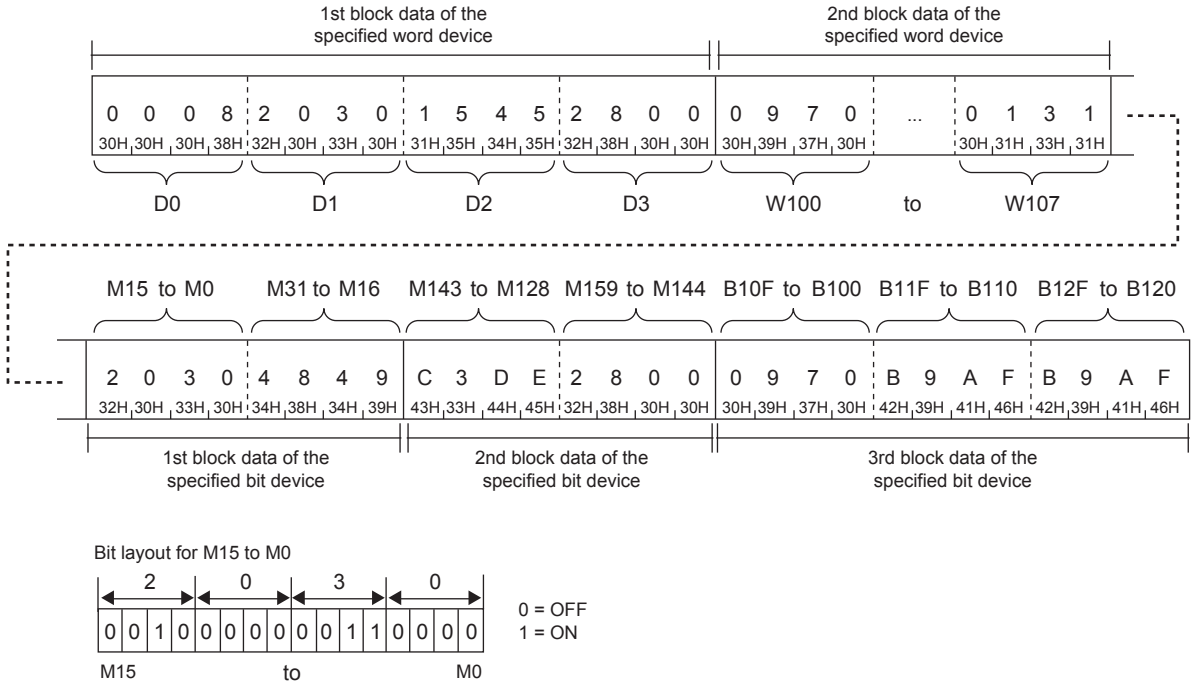
Item	Read contents
Word device	• Block 1: D0 to D3 (4 points) • Block 2: W100 to W107 (8 points)
Bit device	• Block 1: M0 to M31 (2 points) • Block 2: M128 to M159 (2 points) • Block 3: B100 to B12F (3 points)

■When communicating data in ASCII code

(Request data)

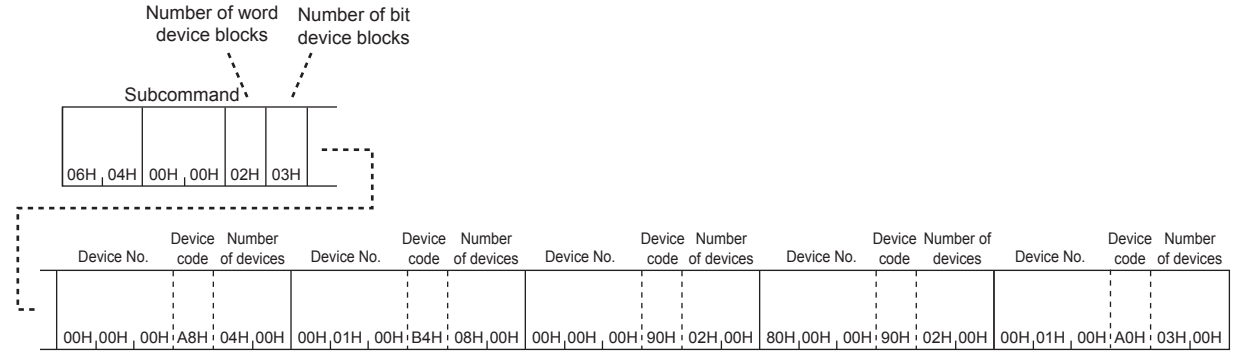


(Response data)

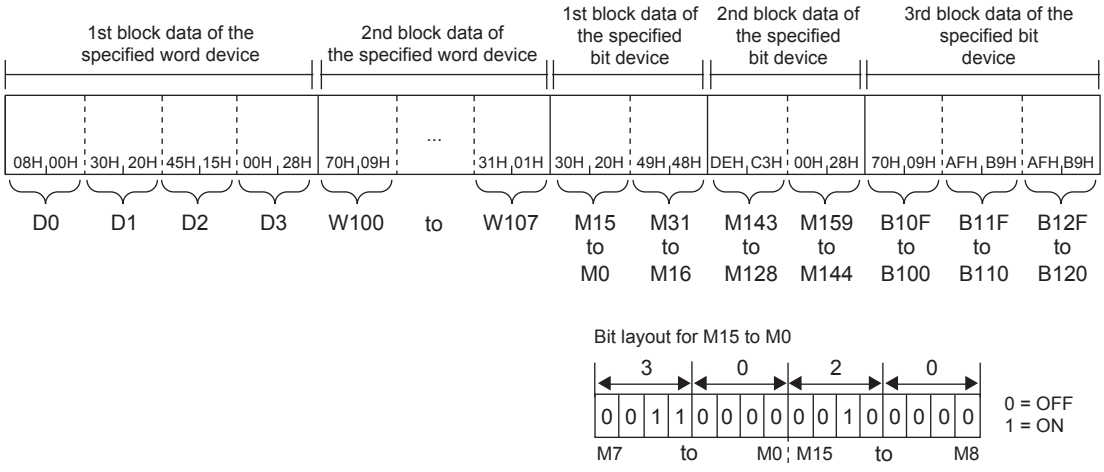


■When communicating data in binary code

(Request data)



(Response data)

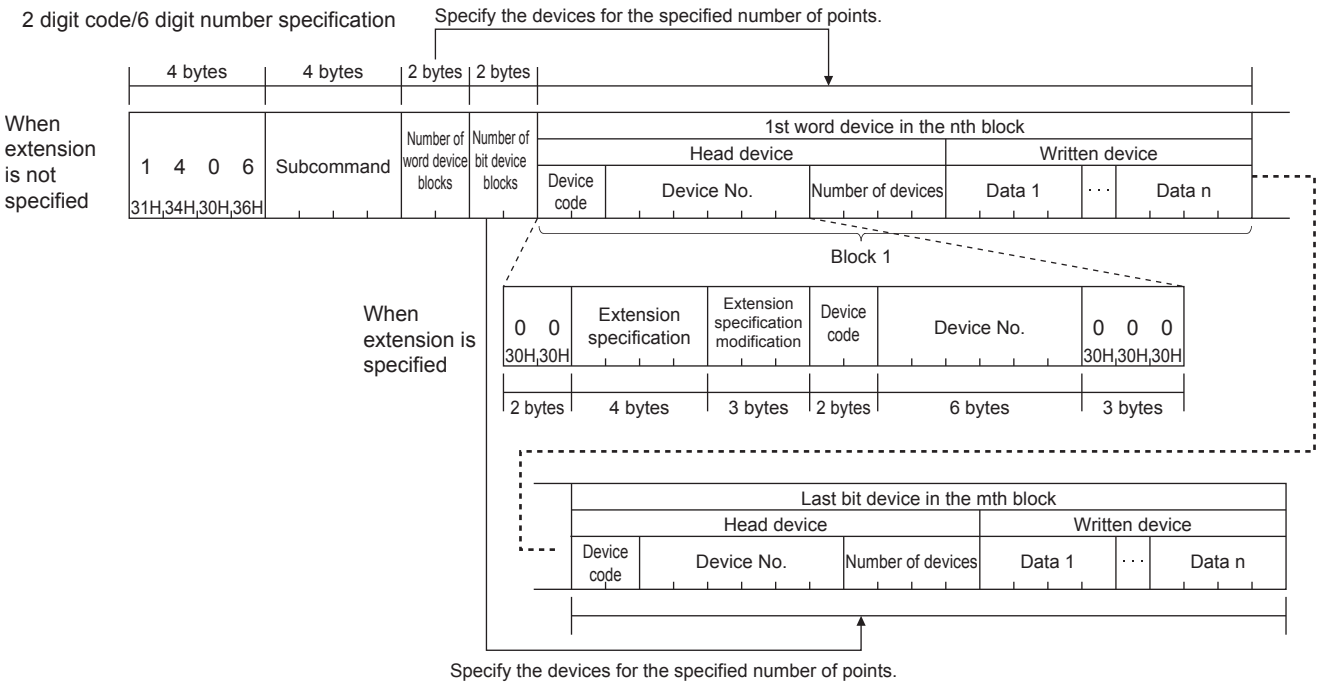


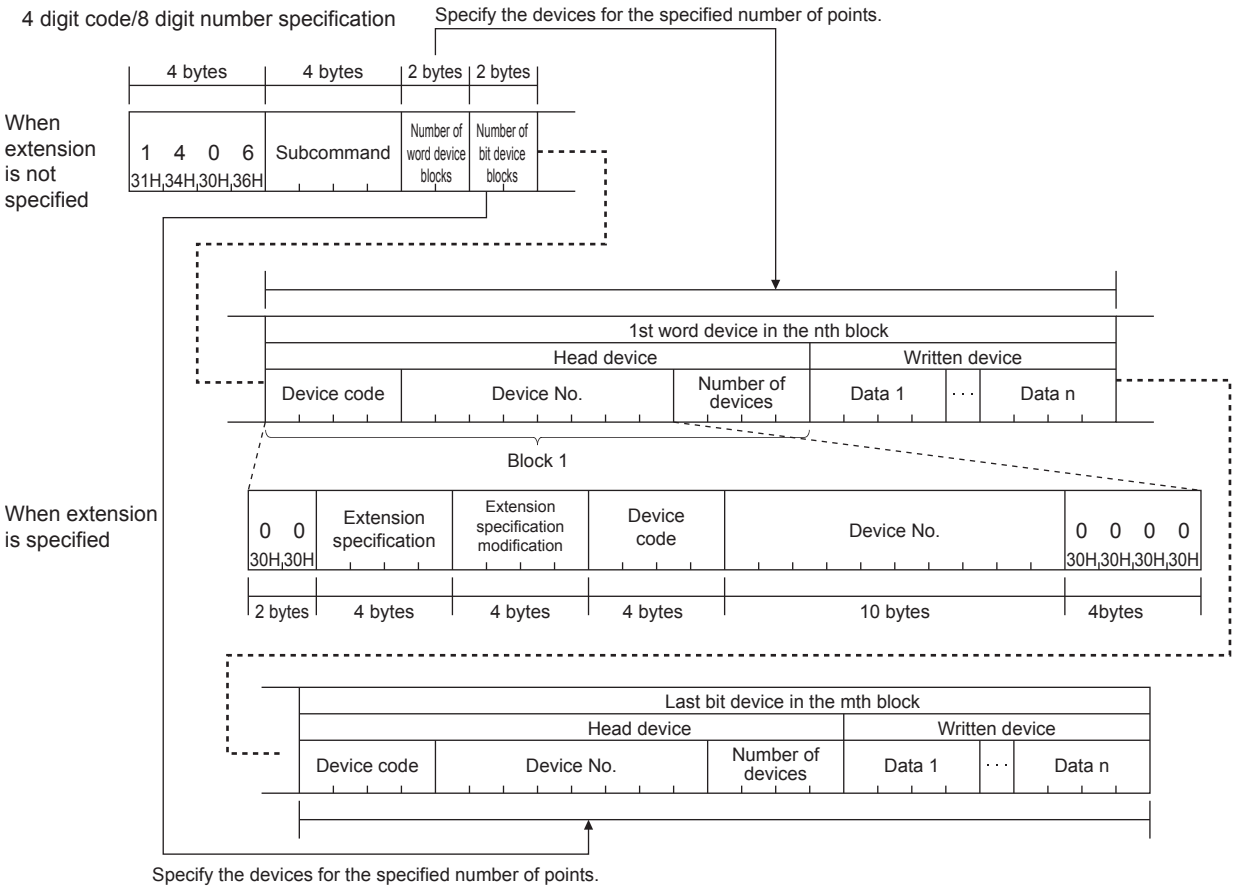
Batch write multiple blocks

The examples shown in this section explain the control procedure for writing by randomly specifying multiple blocks, where 1 block consists of n point(s) of a bit device memory (one point is specified by 16-bit) and a word device memory (one point is specified by 1-word).

Request data

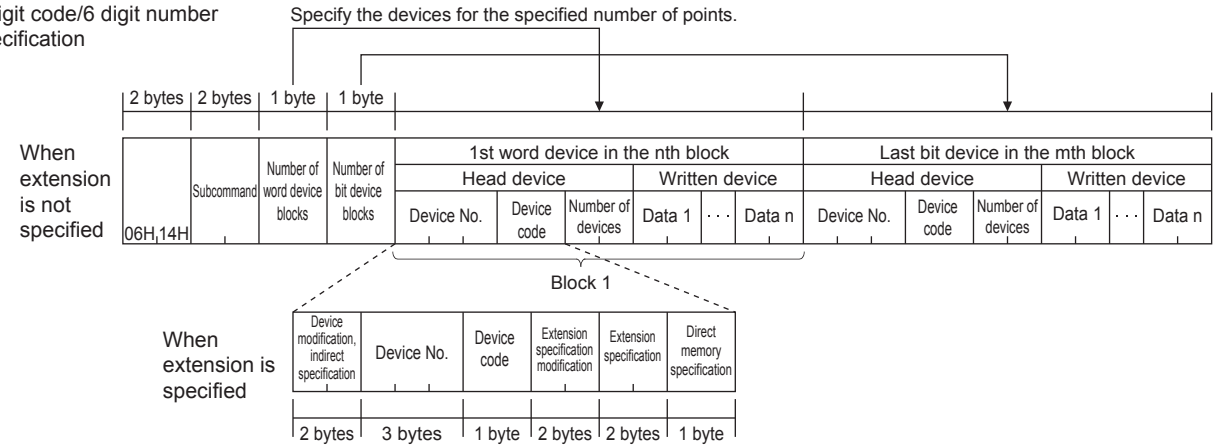
■When communicating data in ASCII code



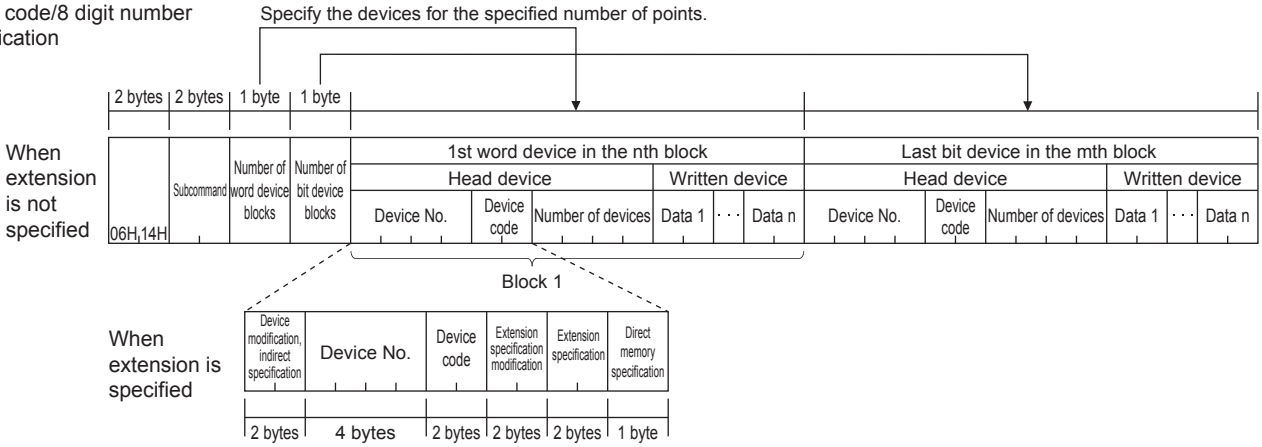


■When communicating data in binary code

2 digit code/6 digit number specification



4 digit code/8 digit number specification



❶Subcommand

Specify the subcommand selected from the item.

Item			Subcommand					
Data size specification	Device specification format	Device memory extension specification	ASCII code (Upper column: characters, lower column: character code)				Binary code	
Word units	2 digit code/6 digit number specification	Not specified	0	0	0	0	00H	00H
			30H	30H	30H	30H		
		Specified	0	0	8	0	80H	00H
			30H	30H	38H	30H		
	4 digit code/8 digit number specification	Specified	0	0	8	2	82H	00H
			30H	30H	38H	32H		

❷Number of word device blocks and number of bit device blocks

Specify the number of blocks of the device to be written in hexadecimal.

Item	Description	Number of points	
		ASCII code	Binary code
Number of word device blocks	Specify the number of blocks of the word device to be written.	Number of word device blocks + number of bit device blocks ≤120 When device memory extension specification is used, double the number of the block points.	
Number of bit device blocks	Specify the number of blocks of the bit device to be written.		

❸Device code, device No., number of device points

Specify the device points while satisfying the following conditions:

(number of word device blocks + number of bit device blocks) × 4 + total number of points for all word device blocks + total number of points for all bit device blocks ≤ 760

Item	Description
Word device	Specify the device of the points specified in "Number of word device blocks". When "Number of word device blocks" is set to 0, this specification is unnecessary.
Bit device	Specify the device of the points specified in "Number of bit device blocks". When "Number of bit device blocks" is set to 0, this specification is unnecessary.

Point

When specifying a contact or coil of a timer, retentive timer, or counter, use the bit device block.

Set up in order of word device → bit device.

Response data

There is no response data for the batch write multiple blocks command.

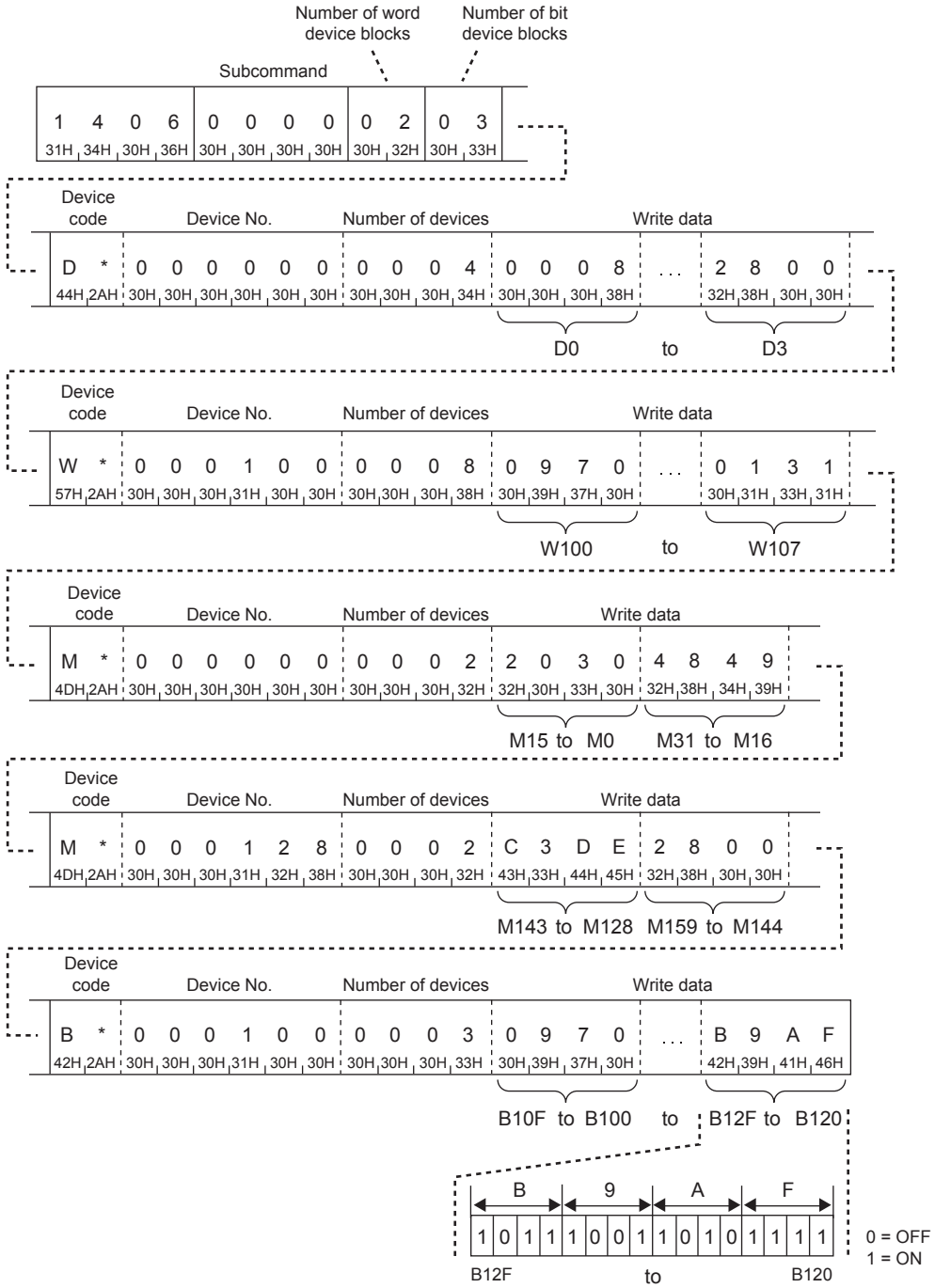
Communication example

Write values from devices as follows.

Item	Write contents
Word device	• Block 1: D0 to D3 (4 points) • Block 2: W100 to W107 (8 points)
Bit device	• Block 1: M0 to M31 (2 points) • Block 2: M128 to M159 (2 points) • Block 3: B100 to B12F (3 points)

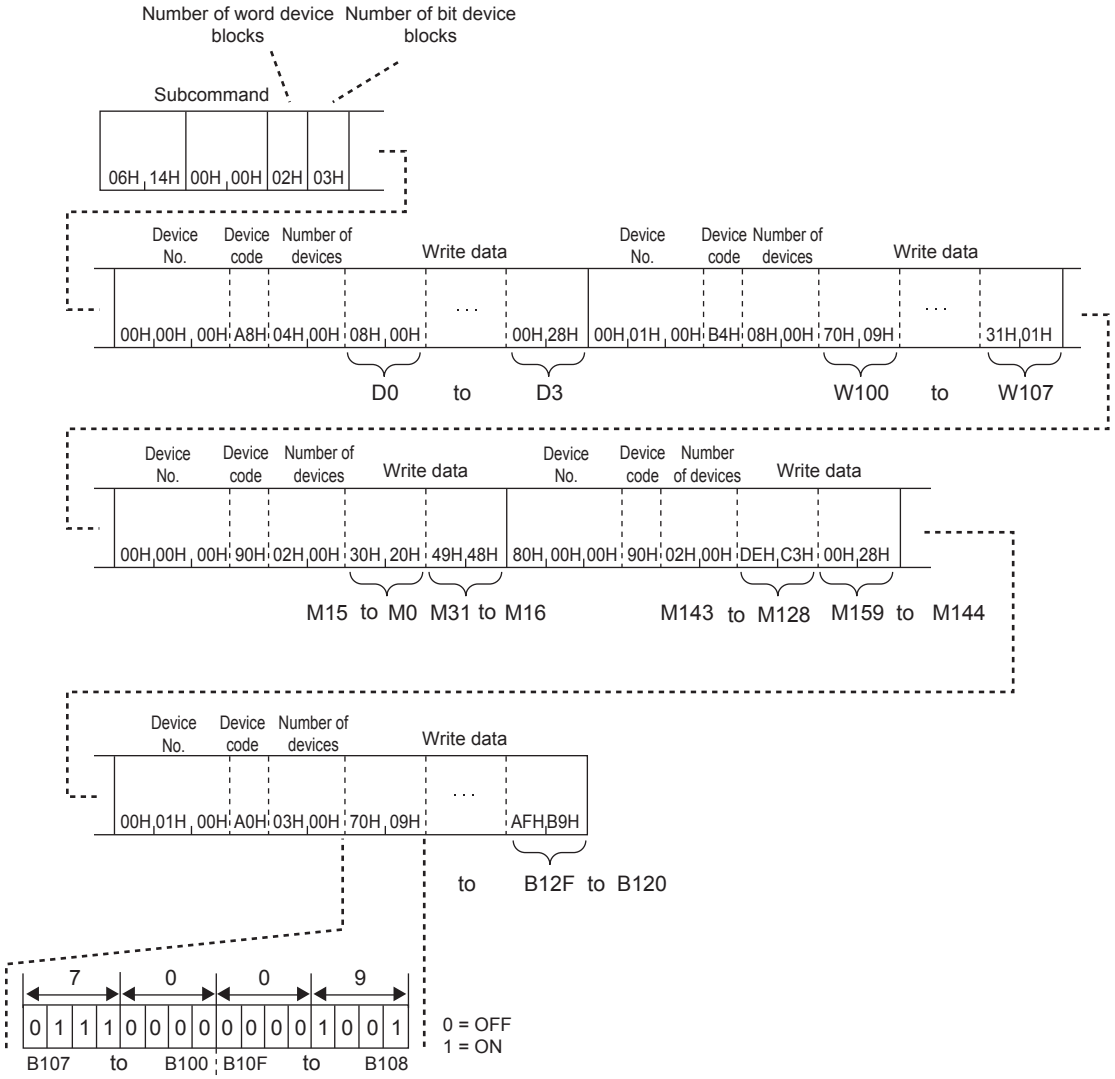
■When communicating data in ASCII code

(Request data)



■When communicating data in binary code

(Request data)



4.3 Remote Control

This section describes the command to set the MC protocol compatible device or CPU module to the RUN status or STOP status by a message from the external device.

Before the remote operation

When the accessed device or module is turned from off to on or the system is reset after the remote operation

The information about the remote operation will be deleted.

Ex.

Even if the Remote STOP is executed when the switch of the CPU module is in the RUN status, the operation will return to the RUN status after resetting the module.

When a remote password of the CPU module of the access destination is enabled

Remote operation from the external device is not available. An error will occur at the access destination, and an abnormal response will be sent back to the external device. Unlock the remote password of the CPU module side, and resend the request message.

Operable station in one command

Only one station can be operated remotely by one command.

Remote RUN

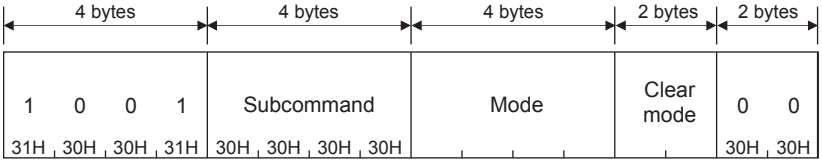
This command executes Remote RUN to the access destination module.

Point

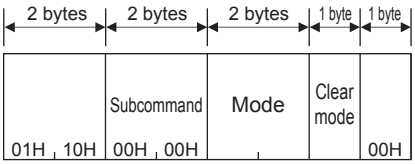
Remote RUN can be executed when the switch of the access destination module is in the RUN status. Even if the switch is in the STOP status, Remote RUN (command: 1001H) will be completed normally. However, the access destination does not change to the RUN status.

Request data

■When communicating data in ASCII code



■When communicating data in binary code



■Mode

This mode specifies whether Remote RUN can be executed forcibly by a device other than the external device which performed Remote STOP or Remote PAUSE. If forced execution is not allowed, Remote RUN can be executed only by the external device which performed Remote STOP or Remote PAUSE.

Forced execution is used when the external device which performed the remote operation cannot execute Remote RUN because of a problem with the device.

Item	Mode	
	ASCII code	Binary code
Forced execution not allowed (Remote RUN cannot be executed when other device executes Remote STOP or Remote PAUSE.)	0 0 0 1 30H 30H 30H 31H	01H 00H
Forced execution allowed (Remote RUN can be executed when other device executes Remote STOP or Remote PAUSE.)	0 0 0 3 30H 30H 30H 33H	03H 00H

■Clear mode

This mode specifies whether the clear (initialization) processing of device is executed when operation starts after Remote RUN.

Only 00H is valid.

Item	Mode	
	ASCII code	Binary code
Do not clear device	0 0 30H 30H	00H

Response data

There is no response data for the Remote RUN command.

Communication example

Set mode to "Forced execution not allowed", and set clear mode to "Clear all devices including that in the latch range" when executing Remote RUN.

• When communicating data in ASCII code

(Request data)

												Mode				Clear mode			
1	0	0	1	0	0	0	0	0	0	0	1	0	2	0	0
31H	30H	30H	31H	30H	30H	30H	30H	30H	30H	30H	31H	30H	32H	30H	30H

• When communicating data in binary code

(Request data)

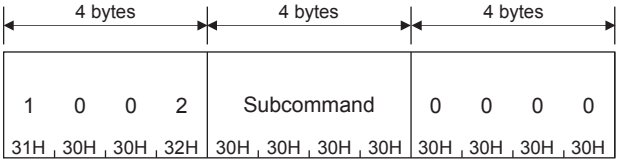
														Mode		Clear mode			
01H	10H	00H	00H	01H	00H	02H	00H

Remote STOP

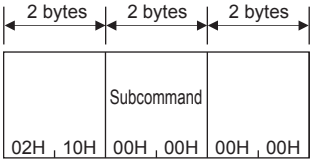
This command executes Remote STOP to the access destination module.

Request data

■When communicating data in ASCII code



■When communicating data in binary code



Response data

There is no response data for the Remote STOP command.

Communication example

Send request messages from the external device by using the message format shown in the request data above.

Remote PAUSE

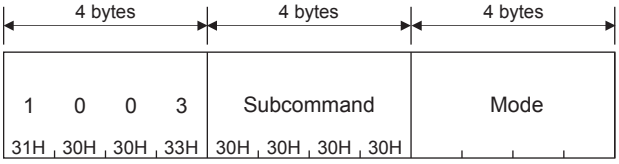
This command executes Remote PAUSE to the access destination module.

Point

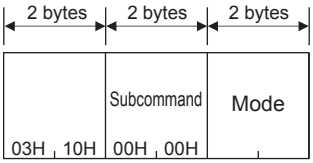
Remote PAUSE can be executed when the switch of the access destination module is in the RUN status. Even if the switch is in the STOP status, Remote PAUSE (command: 1003H) will be completed normally. However, the access destination does not change to the PAUSE status.

Request data

■When communicating data in ASCII code



■When communicating data in binary code



■Mode

This mode specifies whether Remote PAUSE can be executed forcibly by a device other than the external device which performed Remote STOP or Remote PAUSE. If forced execution is not allowed, Remote PAUSE can be executed only by the external device which performed Remote STOP or Remote PAUSE.

Forced execution is used when the external device which performed the remote operation cannot execute Remote PAUSE because of a problem with the device.

Item	Mode	
	ASCII code	Binary code
Forced execution not allowed (Remote RUN cannot be executed when other device executes Remote STOP or Remote PAUSE.)	0 0 0 1 30H 30H 30H 31H	01H 00H
Forced execution allowed (Remote RUN can be executed when other device executes Remote STOP or Remote PAUSE.)	0 0 0 3 30H 30H 30H 33H	03H 00H

Response data

There is no response data for the Remote PAUSE command.

Communication example

Set mode to "Forced execution not allowed" when executing Remote PAUSE.

■When communicating data in ASCII code

(Request data)

	Mode	
1 0 0 3 31H 30H 30H 33H	0 0 0 0 30H 30H 30H 30H	0 0 0 1 30H 30H 30H 31H

■When communicating data in binary code

(Request data)

	Mode	
03H 10H	00H 00H	01H 00H

Remote latch clear

This command executes remote latch clear to the access destination module.

Point

Before executing the remote latch clear, set the status of the access destination module to STOP.

While the access destination is stopped or paused remotely by request from another external device:

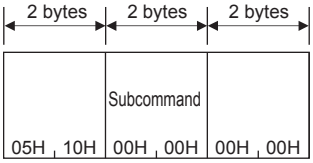
- The remote latch clear cannot be executed. Abnormal completion of the command will occur.
- Cancel the Remote STOP or Remote PAUSE before executing the command.

Request data

■When communicating data in ASCII code

4 bytes	4 bytes	4 bytes
1 0 0 5 31H 30H 30H 35H	Subcommand 30H 30H 30H 30H	0 0 0 0 30H 30H 30H 30H

■When communicating data in binary code



Response data

There is no response data for remote latch clear command.

Communication example

Send request messages from the external device by using the message format shown in the request data above.

Remote RESET

This command executes Remote RESET to the access destination module. Remote RESET is used to restore when an error occurred in the MC protocol compatible device.

Point

Before executing Remote RESET, perform the following.

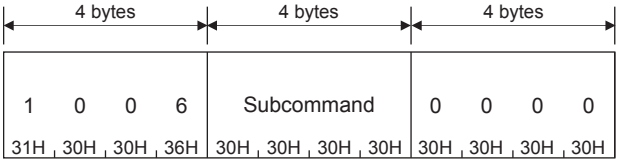
- When the access destination module has a Remote RESET enable/disable setting, go to GX Works3 Navigation window ⇨ [Parameter] ⇨ Module model ⇨ [CPU Parameter] ⇨ [Operation Related Setting] ⇨ [Remote Reset Setting], and select "Enable" for "Remote Reset". (Default: Disable)
- Set the status of the access destination module to STOP.

Precautions

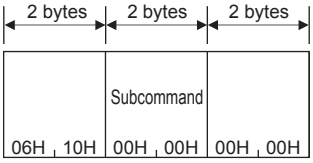
- In some cases, Remote RESET cannot be executed because of hardware error, etc.
- The response message when Remote RESET is executed may not be sent back to the external device since the access destination is reset.

Request data

■When communicating data in ASCII code



■When communicating data in binary code



Response data

There is no response data for the Remote RESET command.

Communication example

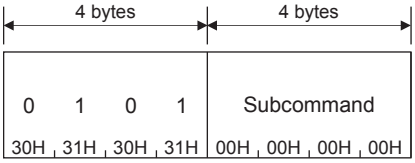
Send request messages from the external device by using the message format shown in the request data above.

Read CPU model nome

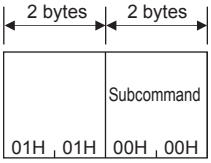
This command reads the processor module name code (processor type) of the access destination module.

Request data

■When communicating data in ASCII code

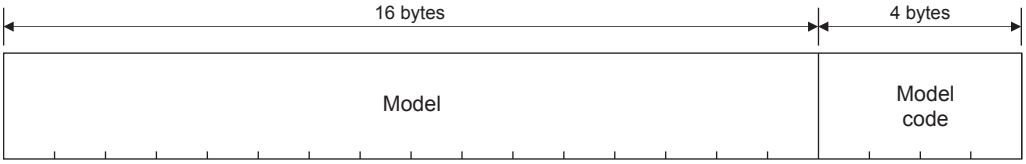


■When communicating data in binary code

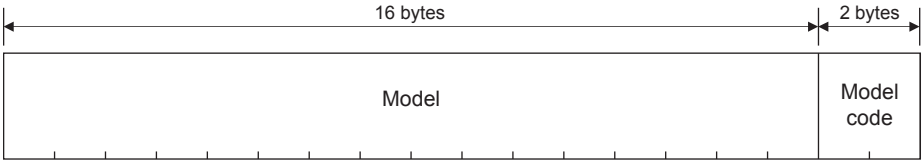


Response data

■When communicating data in ASCII code



■When communicating data in binary code



■Model

The characters of the module model are stored for 16 characters from the upper byte.

If the model to be read is less than 16 characters, space (20H) is stored for the remaining characters. Even when communicating data in binary code, the module model is stored in ASCII code.

■Model code

The following model codes will be stored.

When communicating in binary code, the data is stored in order from the lower byte to the upper byte.

Model	Model code (hexadecimal)
FX5U-32MR/ES	4A21H
FX5U-64MR/ES	4A23H
FX5U-80MR/ES	4A24H
FX5U-32MT/ES	4A29H
FX5U-64MT/ES	4A2BH
FX5U-80MT/ES	4A2CH
FX5U-32MT/ESS	4A31H
FX5U-64MT/ESS	4A33H
FX5U-80MT/ESS	4A34H
FX5U-32MR/DS	4A41H
FX5U-64MR/DS	4A43H
FX5U-80MR/DS	4A44H
FX5U-32MT/DS	4A49H
FX5U-64MT/DS	4A4BH
FX5U-80MT/DS	4A4CH
FX5U-32MT/DSS	4A51H
FX5U-64MT/DSS	4A53H
FX5U-80MT/DSS	4A54H
FX5UC-32MT/D	4A91H
FX5UC-64MT/D	4A92H
FX5UC-96MT/D	4A93H
FX5UC-32MT/DSS	4A99H
FX5UC-64MT/DSS	4A9AH
FX5UC-96MT/DSS	4A9BH
FX5UC-32MR/DS-TS	4AA9H
FX5UC-32MT/DS-TS	4AB1H
FX5UC-32MT/DSS-TS	4AB9H
FX5UJ-24MR/ES	4B0DH
FX5UJ-40MR/ES	4B0EH
FX5UJ-60MR/ES	4B0FH
FX5UJ-24MT/ES	4B14H
FX5UJ-40MT/ES	4B15H
FX5UJ-60MT/ES	4B16H
FX5UJ-24MT/ESS	4B1BH
FX5UJ-40MT/ESS	4B1CH
FX5UJ-60MT/ESS	4B1DH
FX5S-30MR/ES	4B4EH
FX5S-40MR/ES	4B4FH
FX5S-60MR/ES	4B50H
FX5S-80MR/ES*1	4B51H
FX5S-30MT/ES	4B55H
FX5S-40MT/ES	4B56H
FX5S-60MT/ES	4B57H
FX5S-80MT/ES*1	4B58H
FX5S-30MT/ESS	4B5CH
FX5S-40MT/ESS	4B5DH
FX5S-60MT/ESS	4B5EH
FX5S-80MT/ESS*1	4B5FH

*1 Area-specific model

Point

The model of the CPU module is identified by the model code.

Communication example

■When communicating data in ASCII code

(Request data)

0	1	0	1	0	0	0	0
30H	31H	30H	31H	30H	30H	30H	30H

(Response data)

F	X	5	U	-	3	2	M	R	/	E	S					4	A	2	1
46H	58H	35H	55H	2DH	33H	32H	4DH	52H	2FH	45H	53H	20H	20H	20H	20H	34H	41H	32H	31H

■When communicating data in binary code

(Request data)

01H	01H	00H	00H

(Response data)

F	X	5	U	-	3	2	M	R	/	E	S					
46H	58H	35H	55H	2DH	33H	32H	4DH	52H	2FH	45H	53H	20H	20H	20H	20H	21H 4AH

4.4 Clear Error

This function turns off ERR LED of the FX5CPU from the external device and/or initializes the communication error information or error code stored in the buffer memory.
This function is used to initialize the current error information due to an abnormal response for a request message and return it to the normal state or initialize the error code storage area of the buffer memory.
The order and description of the data item differ depending on the frame and pattern in a communication.

Point

This function can be used only for the FX5CPU which is connected with the external device.
This function cannot be used for the FX5CPU of another station via the network system.

The data part of the command and control procedure when the display LEDs of the FX5CPU are turned off and the communication error information is initialized from the external device is described.

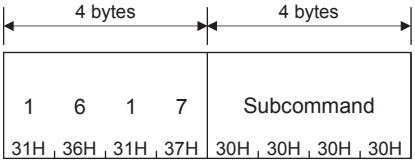
Command

Function	Command (Subcommand)	Processing content	CPU module status		
			STOP	RUN	
				Write allow setting	Write prohibit setting
Clear Error	1617(0000)	Turns off the display LEDs, initializes the error code, and others.	○	○	○

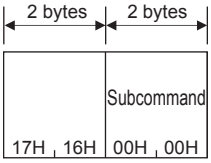
○: The function can be executed.

Request data

■When communicating data in ASCII code



■When communicating data in binary code



Response data

There is no response data for the Clear Error command.

Communication example

Send request messages from the external device by using the message format shown in the request data above.

4.5 Self-Test

This function tests whether the communication function between the external device and FX5CPU operates normally or not. The control procedure when this function is used is described with examples.

Point

- At the startup of the FX5CPU or when trouble occurs, this function can check whether the connection between the external device and FX5CPU is correct and/or whether the data communication function operates normally.
- This function can be used only for the FX5CPU which is connected with the external device (including a multi-drop connection station). This function cannot be used for the FX5CPU of another station via the network system.

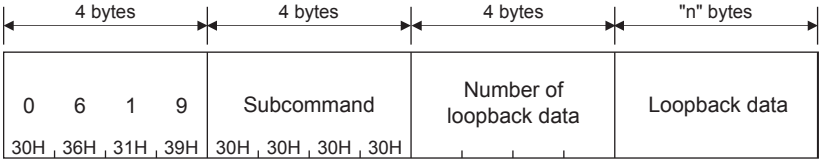
Command

Function	Command (Subcommand)	Processing content	CPU module status		
			STOP	RUN	
				Write allow setting	Write prohibit setting
Self-Test	0619(0000)	Checks whether data communication is executed normally.	○	○	○

○: The function can be executed.

Request data

■When communicating data in ASCII code



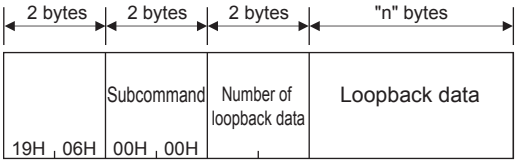
- Number of loopback data (number of bytes)

The number of the bytes is converted into a four-digit ASCII code (hexadecimal) and data is sent from the upper digit ("0").

- Loopback data (user data)

The order of character strings for up to 960 1-byte characters ("0" to "9", "A" to "F") is sent from the head.

■When communicating data in binary code



- Number of loopback data (number of bytes)

The two-byte numerical value which indicates the number of the bytes is used and data is sent from the low byte (L: bit 0 to 7).

- Loopback data (user data)

Data is sent for up to 960 bytes from the head by treating each character code ("0" to "9", "A" to "F") as a 1 byte value.

Response data

The same number of the loopback data and loopback data which the external device sent are sent back to the external device.

Communication example

Send request messages from the external device by using the message format shown in the request data (☞ Page 87).

■When executing the Self-Test by communicating in ASCII code

(Request data)

Command	Subcommand	Number of loopback data	Loopback data
0 6 1 9	0 0 0 0	0 0 1 2	a b c d e f g h i j k l
30H 36H 31H 39H	30H 30H 30H 30H	30H 30H 31H 32H	61H 62H 63H 64H 65H 66H 67H 68H 69H 6AH 6BH 6CH

(Response data)

Number of loopback data	Loopback data
0 0 1 2	a b c d e f g h i j k l
30H 30H 31H 32H	61H 62H 63H 64H 65H 66H 67H 68H 69H 6AH 6BH 6CH

■When executing the Self-Test by communicating in binary code

(Request data)

Command	Subcommand	Number of loopback data	Loopback data
			a b c d e f g h i j k l
19H 06H	00H 00H	12H 00H	61H 62H 63H 64H 65H 66H 67H 68H 69H 6AH 6BH 6CH

(Response data)

Number of loopback data	Loopback data
	a b c d e f g h i j k l
12H 00H	61H 62H 63H 64H 65H 66H 67H 68H 69H 6AH 6BH 6CH

5 COMMUNICATING USING 1C FRAMES

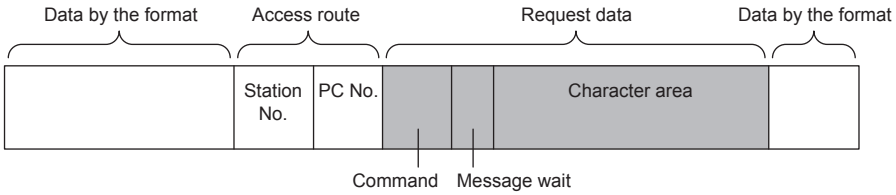
This chapter explains the functions when accessing using 1C frame and their message format.
1C frame is compatible with the communication function of the dedicated protocols supported by computer link of FX3 and A series computer link modules.
Only the commands for 1C frame explained in this chapter can be used for 1C frame.

5.1 Message Format

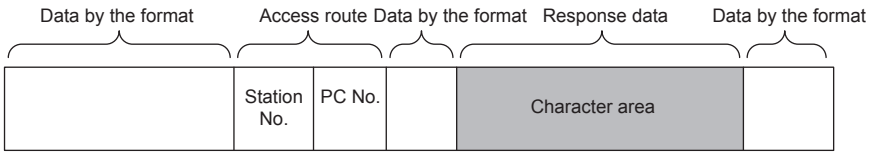
This section explains the message format when communicating data using 1C frame.

Message format

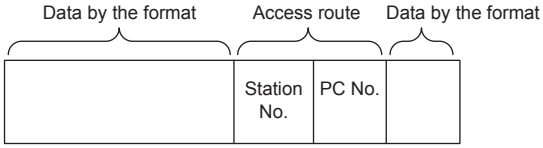
■Request message



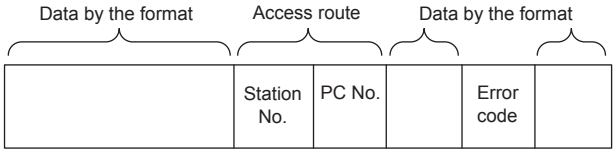
■Response message (Normal completion: Response data)



■Response message (Normal completion: No response data)



■Response message (Abnormal completion)



Setting data

Set the following items.

Item		Description	Reference
Data by format		The message formats differ depending on the set format (Format 1, Format 4).	Page 17 Message Formats of Each Protocol
Access route	Station No.	Specify the station to be connected from an external device.	Page 27 Station No.
	PC No.	Specify the network station No. of the access target.	Page 28 Network No., PC No.
Request data	Command	Specify the function to request such as read or write.	Page 90 Command
	Message wait	A data to generate a delay time for response transmission. Specify the wait time within the range of 0 to 150 ms in 10 ms units.	Page 91 Message wait
	Character area	A data that instructs the FX5 CPU module to execute a request specified by command. The content of the character areas differs depending on the command.	Page 91 Character area
Response data	Character area	A data that FX5 CPU module returns to a request specified by a command. The content of the character areas differs depending on the command.	
Error code		Error code indicates the content of occurred error.	Page 92 Error code

5.2 Details of Setting Data

This section explains how to specify the common data items and their content in each message.

Command

Set the command type. (☞ Page 93 Command and Function Lists for 1C Frame)

The setting values for each command are as follows.

Function	Command		Reference
	Symbol	ASCII code	
Device memory read and write	BR	42H, 52H	Page 97 Batch read (bit units) (command: BR)
	WR	57H, 52H	Page 98 Batch read (word units) (command: WR, QR)
	QR	51H, 52H	
	BW	42H, 57H	Page 100 Batch write (bit units) (command: BW)
	WW	57H, 57H	Page 101 Batch write (word units) (command: WW, QW)
	QW	51H, 57H	
	BT	42H, 54H	Page 103 Test (random write) (bit units) (command: BT)
	WT	57H, 54H	Page 104 Test (random write) (word units) (command: WT, QT)
	QT	51H, 54H	
Remote RUN, Remote STOP	RR	52H, 52H	Page 106 Remote RUN, Remote STOP (command: RR, RS)
	RS	52H, 53H	
Read CPU model name	PC	50H, 43H	Page 107 Read CPU model name (command: PC)
Global	GW	47H, 57H	Page 109 Global signal ON/OFF (command: GW)
Loopback test	TT	54H, 54H	Page 110 Loopback test (Command: TT)

Setting method

Use the commands by converting to 2-digit (hexadecimal) ASCII codes.

Ex.

Device memory batch read (BR) in bit unit

B	R
42H	52H

Message wait

Message wait is a data to generate a delay time for response transmission.

Some external devices may take time to become receiving status after sending a command.

Specify the minimum wait time to send the result after FX5 CPU module is received a command from an external device.

Specify the wait time in accordance with the specifications of the external device.

Setting method

Specify the wait time within the range of 0 to 150 ms in 10 ms units.

Convert 0H to FH (0 to 15) to 1-digit (hexadecimal) ASCII codes regarding 10 ms as 1H.

Ex.

When the message wait time is 100 ms

If the following value is set to message wait in request message, after passing 100 ms or more, transmission of a response message will be started.

A
41H

Character area

The content of the character areas differs depending on the command.

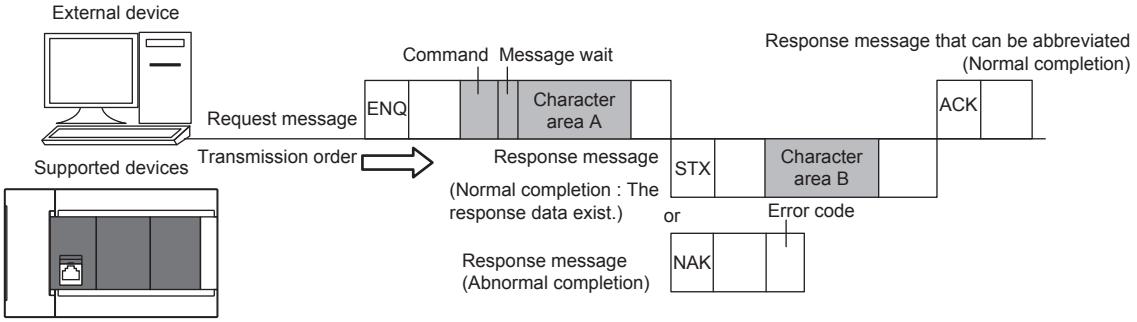
The character area of request data is equivalent to the character A area and the character C area of the dedicated protocols for computer link of FX3 and A series computer link module. The character are of response data is equivalent to the character B area of a dedicated protocols.

- Character area A: A data to instruct the FX5 CPU module to perform the read request specified by a command.
- Character area B: A data that FX5 CPU module returns to a request specified by a command.
- Character area C: A data to instruct the FX5 CPU module to perform the write request specified by a command.

When reading data (Response data)

The following shows the image when the response data (character B area of the dedicated protocol) is included in the response message.

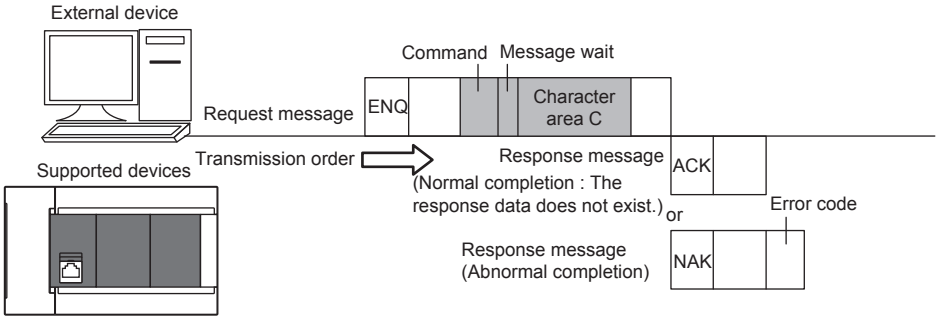
(The head of the message data in the figure is a control code of format 1. ☞ Page 17 Message Formats of Each Protocol)



When writing data (No response data)

The following shows the image when the response data is not included in the response message.

(The head of the message data in the figure is a control code of format 1. ☞ Page 17 Message Formats of Each Protocol)



Error code

Error code indicates the content of occurred error.

If more than one error occurs at the same time, the error code detected first is returned.

For the content of error code and its corrective action, refer to the followig manual.

MELSEC iQ-F FX5 User's Manual (Application)

MELSEC iQ-F FX5 User's Manual (Serial Communication)

Setting method

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Ex.

For error code 05H

0	5
30H	35H

5.3 Command and Function Lists for 1C Frame

Use the following commands for data communication using 1C frame.

Name			Commands		Contents of processing	Number of points processed in onetime update
			Symbol	ASCII code		
Device memory	Batch reading	Bit units	BR	42H, 52H	Reads data in 1-point units from bit devices	256 points
		Word units	WR	57H, 52H	Reads data in 16-point units from bit devices.	32 word (512 points)
					Reads data in 1-word unit from word devices.	64 points
			QR	51H, 52H	Reads data in 16-point units from bit devices.	32 word (512 points)
					Reads data in 1-word unit from word devices.	64 points
	Batch writing	Bit units	BW	42H, 57H	Writes data in 1-points units to bit devices.	160 points
		Word units	WW	57H, 57H	Writes data in 16-points units to bit devices.	10 word (160 points)
					Writes data in 1-points units to bit devices.	64 points
			QW	51H, 57H	Writes data in 16-points units to bit devices.	10 word (160 points)
					Writes data in 1-points units to bit devices.	64 points
	Test (random write)	Bit units	BT	42H, 54H	Writes data in 1-bit unit to a bit device by randomly specifying the device number.	20 points
		Word units	WT	57H, 54H	Writes data in 16-bit unit to a bit device by randomly specifying the device number.	10 word (160 points)
					Writes data in 1-word unit to a word device by randomly specifying the device number.	10 points
			QT	51H, 54H	Writes data in 16-bit unit to a bit device by randomly specifying the device number.	10 word (160 points)
					Writes data in 1-word unit to a word device by randomly specifying the device number.	10 points
PLC	Remote RUN		RR	52H, 52H	Requests remote RUN to a device.	—
	Remote STOP		RS	52H, 53H	Requests remote STOP to a device.	
	CPU model name reading		PC	50H, 43H	Reads PLC model name.	—
Global			GW	47H, 57H	Turns global signal ON or OFF (SM8126 in FX5) in all PLCs connected in computer link.	1 point
Loopback test			TT	54H, 54H	Returns received characters back to the computer as they are.	254 bytes

Command

Command is a communication function issued by MC protocol, and the command is accessible.

When accessing modules other than FX5 CPU module, there is a restriction for the accessible device range.

☞ Page 94 Considerations when accessing devices other than FX5 CPU module

5.4 Device Memory Read and Write

This section explains the specification content and examples of the control procedure when reading from/writing to the device memory are as shown below.

For the message formats other than request data and response data, refer to the following sections.

☞ Page 89 Message Format, Page 90 Details of Setting Data

Considerations

The considerations when reading/writing device memory using the commands described in this section.

Considerations when accessing devices other than FX5 CPU module

■Accessible devices

Only the devices with the same names that exist in FX5 CPU can be accessed within the device range.

☞ Page 95 Accessible device range

■Special relays and special registers

Special relays and special registers can be accessed within the following range.

- Access SM8000 to SM8511 by specifying M8000 to M8511.
- Access SD8000 to SD8511 by specifying D8000 to D8511.
- Access LC0 to LC55 by specifying CS200 to CS255 or CN200 to CN255.

Data to be specified in command

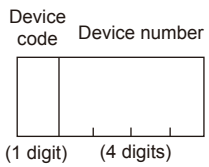
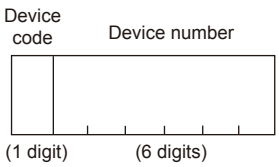
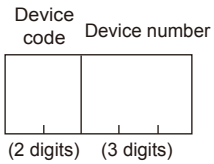
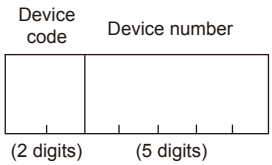
Device codes, device numbers

The settings of each device when reading/writing device memory can be performed using device code and device number as shown in the following figure.

Specify the device to be accessed by a device code and a device number.

The data size to be set differs depending on the command.

The setting data size differ when the device type is timer or counter.

Device type	BR, BW, BT, WR, WW, WT command	QR, QW, QT command
Other than timer and counter		
Timer, counter		

■BR, BW, BT, WR, WW, WT commands

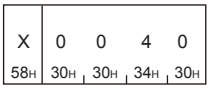
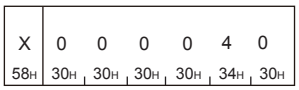
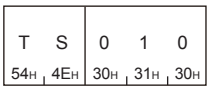
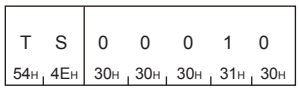
- Device code: Convert the device name to 1-digit ASCII code (2-digits for timer or counter), and send it from the upper digits.
- Device number: Convert the numerical value to 4-digit ASCII code (3-digits for timer or counter), and send it from the upper digits.

■QR, QW, QT commands

- Device code: Convert the device name to 1-digit ASCII code (2-digits for timer or counter), and send it from the upper digits.
- Device number: Convert the numerical value to 6-digit ASCII code (5-digits for timer and counter), and send it from the upper digits.

Ex.

Current value of input (X) 40 and timer (T) 10

Device	BR, BW, BT, WR, WW, WT command	QR, QW, QT command
Input (X) 40		
Current value of timer (T) 10		

Accessible device range

The following table shows the devices and device number range that can be specified when accessing the device memory. Access the CPU module within the range of device number that can be used by commands and the range of the device number that can be used in the access target CPU. (☞ Page 94 Considerations when accessing devices other than FX5 CPU module)

Device		Type	Device code	Device number	Representation	Available commands
Input relay		Bit	X	X0000 to X0377	Octal/Hexadecimal*1	BR, BW, BT, WR, WW, WT
				X000000 to X000377	Octal/Hexadecimal*1	QR, QW, QT
Output relay		Bit	Y	Y0000 to Y0377	Octal/Hexadecimal*1	BR, BW, BT, WR, WW, WT
				Y000000 to Y000377	Octal/Hexadecimal*1	QR, QW, QT
Internal relay		Bit	M	M0000 to M7679	Decimal	BR, BW, BT, WR, WW, WT
				M000000 to M007679	Decimal	QR, QW, QT
Step relay		Bit	S	S0000 to S4095	Decimal	BR, BW, BT, WR, WW, WT
				S000000 to S004095	Decimal	QR, QW, QT
Timer	Contact	Bit	TS	TS000 to TS511	Decimal	BR, BW, BT, WR, WW, WT
				TS00000 to TS00511	Decimal	QR, QW, QT
	Present value	Word	TN	TN000 to TN511	Decimal	WR, WW, WT
				TN00000 to TN00511	Decimal	QR, QW, QT
Counter	Contact	Bit	CS*2	CS000 to CS255	Decimal	BR, BW, BT, WR, WW, WT
				CS00000 to CS00255	Decimal	QR, QW, QT
	Present value	Word	CN*3	CN000 to CN255	Decimal	WR, WW, WT
				CN00000 to CN00255	Decimal	QR, QW, QT
Data register		Word	D	D0000 to D7999	Decimal	WR, WW, WT
				D000000 to D007999	Decimal	QR, QW, QT
Extended register		Word	R	R0000 to R9999	Decimal	WR, WW, WT
				R000000 to R032767	Decimal	QR, QW, QT
Special relay		Bit	M*4	M8000 to M8511	Decimal	BR, BW, BT, WR, WW, WT
				M008000 to M008511	Decimal	QR, QW, QT
Special Register		Word	D*5	D8000 to D8511	Decimal	WR, WW, WT
				D008000 to D008511	Decimal	QR, QW, QT

*1 It depends on the message format. See below.
Format 1 (X,Y OCT), format 4 (X,Y OCT): octal
Format 1 (X,Y HEX), format 4 (X,Y HEX): hexadecimal

*2 Access LCS0 to LCS55 by specifying CS200 to CS255 (CS00200 to CS00255).

*3 Access LCN0 to LCN55 by specifying CN200 to CN255 (CN00200 to CN00255).

*4 Access SM8000 to SM8511 by specifying M8000 to M8511 (M008000 to M008511).

*5 Access SD8000 to SD8511 by specifying D8000 to D8511 (D008000 to D008511).

Precautions

- When using the command specifying the word unit (WR, WW, WT, QR, QW, QT), make sure to set a multiple of 8 as the head device address of the bit device.
- Special relays and special registers are divided into three, the ones for read only, write only, and the system. If the write is performed outside the writeable range, an error may occur in the PLC. For details of the special relays and special registers, refer to MELSEC iQ-F FX5 User's Manual (Application).
- Do not set the setting which overlaps the data register and the special register.
- In the FX5 CPU module, even if the device value is changed, the device initial value is not changed.

Number of device points

Specify the number of device points to be read or written.

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the number of device points in one command within the device points that can be processed in one communication.

☞ Page 93 Command and Function Lists for 1C Frame

Specify '00' for 256 points.

Ex.

5 points, 20 points, 256 points

Number of device points	ASCII code
5 points	0 5 30H 35H
20 points	1 4 31H 34H
256 points	0 0 30H 30H

Read data, write data

The data storage method is the same as reading/writing data with device access of 4C/3C frame. (☞ Page 41 Read data, write data)

Batch read (bit units) (command: BR)

Reads bit devices in batch.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

Command	Message wait	Head device	Number of device points

■Response data

The value of read device is stored in bit units. (☞ Page 41 Read data, write data)

Data specified by request data

■Command

BR command

B	R
42H	52H

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Head device

Specify the head device. (☞ Page 94 Device codes, device numbers)

■Number of device points

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the number of device points within the following range:

- 1 ≤ Number of device points ≤ 256 (for 256 points, specify '00H')
- Head device No. + Number of device points - 1 ≤ Maximum device No.

Communication example

Read data in bit units under the following conditions.

- Message wait: 100 ms
- Head device: X40
- Number of device points: 5 points

(Request data)

■When using BR

Message wait

Head device

Number of device points

B	R	A	X	0	0	4	0	0	5
42H	52H	41H	58H	30H	30H	34H	30H	30H	35H

(Response data)

Data read

0	1	1	0	1
30H	31H	31H	30H	31H
(X40)	(X41)	(X42)	(X43)	(X44)

Batch read (word units) (command: WR, QR)

Reads bit devices in 16-point units.

Reads word devices in 1-point units

Message format

The following shows the message format of the request data and response data of the command.

■Request data

Command	Message wait	Head device	Number of device points (Number of words)

■Response data

The value of read device is stored in word units. (☞ Page 41 Read data, write data)

Data specified by request data

■Command

WR command	QR command
W R 57H 52H	Q R 51H 52H

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Head device

Specify the head device. (☞ Page 94 Device codes, device numbers)

■Number of device points

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the number of device points within the following range:

- For bit device: 1 ≤ Number of device points ≤ 32
- For bit device: Head device No. + Number of device points ×16 - 1 ≤ Maximum device No.
- For word device: 1 ≤ Number of device points ≤ 64
- For word device: Head device No. + Number of device points - 1 ≤ Maximum device No.

When reading CN200 to CN255 (CN00200 to CN00255), since 1 device point is 2 words, the number of device points becomes 32 points.

Point

- When specifying bit devices, set the head device No. in multiples of 8.

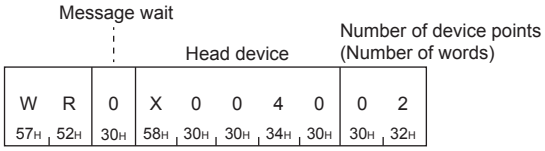
Communication example (Reading bit device memory)

Read bit devices in 16-point units under the following conditions.

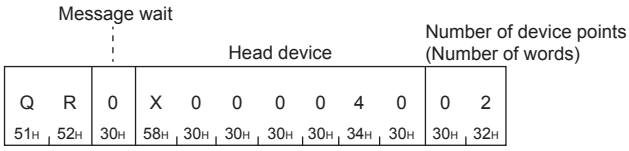
- Message wait: 0 ms
- Head device: X40
- Number of device points: 32 points (2 words)

(Request data)

■When using WR



■When using QR



(Response data)

Data read

0001 (Hexadecimal : 1) 31H	0010 (Hexadecimal : 2) 32H	0011 (Hexadecimal : 3) 33H	0100 (Hexadecimal : 4) 34H	1010 (Hexadecimal : A) 41H	1011 (Hexadecimal : B) 42H	1100 (Hexadecimal : C) 43H	1101 (Hexadecimal : D) 44H
(X57) to (X54)	(X53) to (X50)	(X47) to (X44)	(X43) to (X40)	(X77) to (X74)	(X73) to (X70)	(X67) to (X64)	(X63) to (X60)

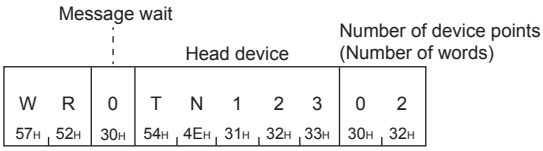
Communication example (Reading word device memory)

Read word devices in 1-point units under the following conditions.

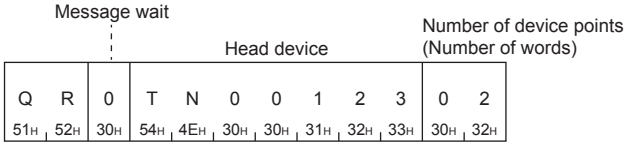
- Message wait: 0 ms
- Head device: Current value of T123
- Number of device points: 2 points (2 words)

(Request data)

■When using WR



■When using QR



(Response data)

Data read

7 B C 9	1 2 3 4
37H 42H 43H 39H	31H 32H 33H 34H
(T123)	(T124)

Batch write (bit units) (command: BW)

Writes bit devices in batch.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

Command	Message wait	Head device	Number of device points	Write data for the number of device points

■Response data

There is no response data for this command.

Data specified by request data

■Command

BW command

B	W
42H	57H

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Head device

Specify the head device. (☞ Page 94 Device codes, device numbers)

■Number of device points

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the number of device points within the following range:

- 1 ≤ Number of device points ≤ 160
- Head device No. + Number of device points - 1 ≤ Maximum device No.

■Write data for the number of device points

Store the data to be written in batch. (☞ Page 96 Read data, write data)

Communication example

Write data in bit units in batch under the following conditions.

- Message wait: 0 ms
- Head device: M903
- Number of device points: 5 points

(Request data)

■When using BW

Message wait

Number of device points (Number of words)

Head device

Write data for the number of device points

B	W	0	M	0	9	0	3	0	5	0	1	1	0	1
42H	57H	30H	4DH	30H	39H	30H	33H	30H	35H	30H	30H	30H	31H	30H
										(M903)	(M904)	(M905)	(M906)	(M907)

Batch write (word units) (command: WW, QW)

Write data to bit devices in 16-point units.

Write data to word devices in 1-point units.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

Command	Message wait	Head device	Number of device points	Write data for the number of device points

■Response data

There is no response data for this command.

Data specified by request data

■Command

WW command	QW command
W (57H) W (57H)	Q (51H) W (57H)

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Head device

Specify the head device. (☞ Page 94 Device codes, device numbers)

■Number of device points

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the number of device points within the following range:

- For bit device: 1 ≤ Number of device points ≤ 10
- For bit device: Head device No. + Number of device points × 16 - 1 ≤ Maximum device No.
- For word device: 1 ≤ Number of device points ≤ 64
- For word device: Head device No. + Number of device points - 1 ≤ Maximum device No.

When reading CN200 to CN255 (CN00200 to CN00255), since 1 device point is 2 words, the number of device points becomes 32 points.

■Write data for the number of device points

Store 4-digit data per one device point. (☞ Page 96 Read data, write data)

Point

When specifying bit devices, set the head device No. in multiples of 8.

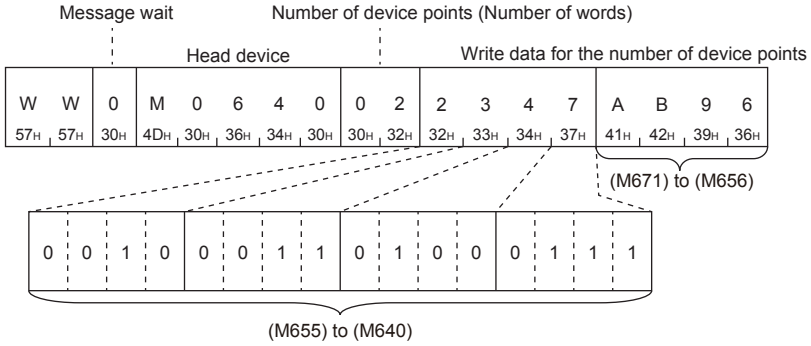
Communication example (Writing to word bit memory)

Write data to bit devices in 16-point units under the following conditions.

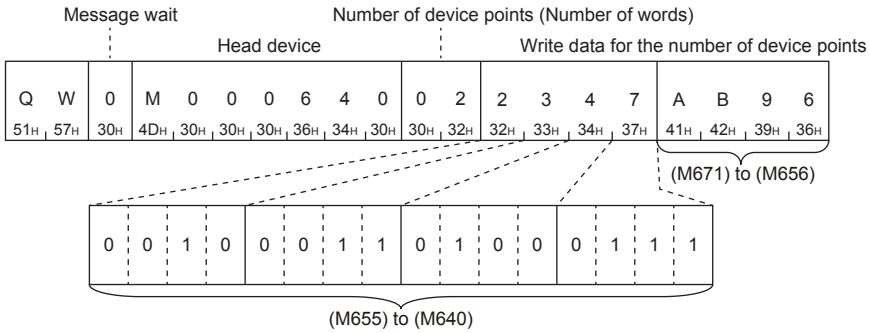
- Message wait: 0 ms
- Head device: M640
- Number of device points: 32 points (2 words)

(Request data)

■When using WW



■When using QW



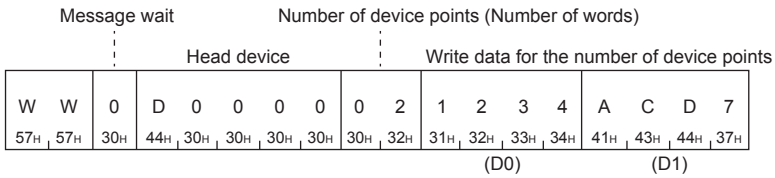
Communication example (Writing to word device memory)

Write data to word devices in 1-point units under the following conditions.

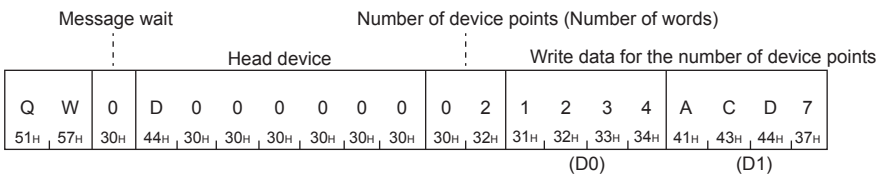
- Message wait: 0 ms
- Head device: D0
- Number of device points: 2 points (2 words)

(Request data)

■When using WW



■When using QW



Test (random write) (bit units) (command: BT)

Set/reset devices and device numbers to bit devices by specifying them randomly in 1 point units.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

Command	Message wait	Number of device points (n points)	Device (first point)	Set/reset (first point)	…	Device (nth point)	Set/reset (nth point)

■Response data

There is no response data for this command.

Data specified by request data

■Command

BT command

B	T
42H	54H

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Number of device points

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the number of device points within the following range:

- 1 ≤ Number of device points ≤ 20

■Device

Specify the device to test. (☞ Page 94 Device codes, device numbers)

■Set/Reset

- 0 (30H): Reset (OFF)
- 1 (31H): Set (ON)

Communication example

Perform the test in bit units under the following conditions.

- Message wait: 0 ms
- Number of device points: 3 points
- Device: Turn ON M50, turn OFF S100, and turn ON Y1

(Request data)

■When using BT

	Message wait	Number of device points (Number of words)	Device	Set/reset	Device	Set/reset	Device	Set/reset
B T	0	0 3	M 0 0 5 0	1	S 0 1 0 0	0	Y 0 0 0 1	1
42H 54H	30H	30H 33H	4DH 30H 30H 35H 30H	31H	53H 30H 31H 30H 30H	31H	59H 30H 30H 30H 31H	31H

Test (random write) (word units) (command: WT, QT)

Set/reset devices and device numbers to bit devices by specifying them randomly in 16 point units.
Write devices and device numbers to word devices by specifying them randomly in 1 point units.
A mixture of word devices and bit devices (16 bit units) can be specified. However, LC0 to LC55 (CN200 to CN255, and CN00200 to CN00255) cannot be used for the word device.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

Command	Message wait	Number of device points (n points)	Device (first point)	Write data (first point)	···	Device (nth point)	Write data (nth point)

■Response data

There is no response data for this command.

Data specified by request data

■Command

WT command	QT command
W T 57H 54H	Q T 51H 54H

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Number of device points

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.
Specify the number of device points within the following range:

- 1 ≤ Number of device points ≤ 10 (for bit device : 10 (16 points are designated as 1))

■Device

Specify the device to test. (☞ Page 94 Device codes, device numbers)

■Write data

Store 4-digit data per one device point. (☞ Page 96 Read data, write data)

Point

When specifying bit devices, set the head device No. in multiples of 8.

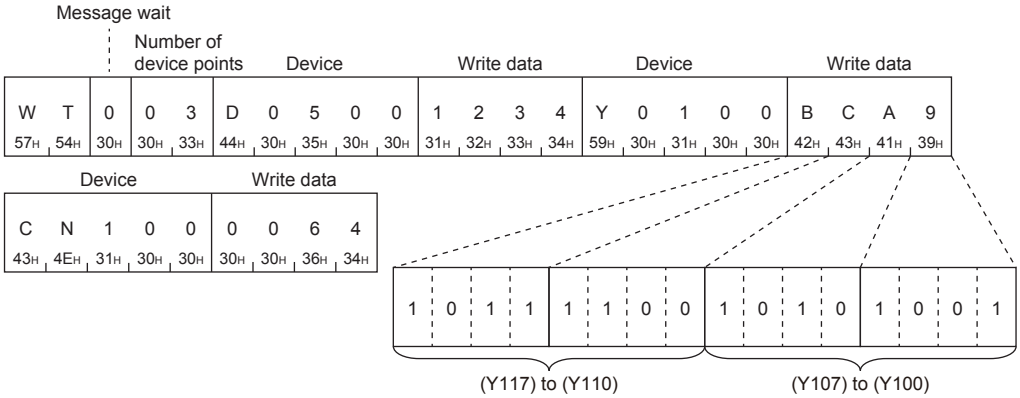
Communication example

Write data with mixture specification of word devices and bit devices (16-point unit) under the following conditions.

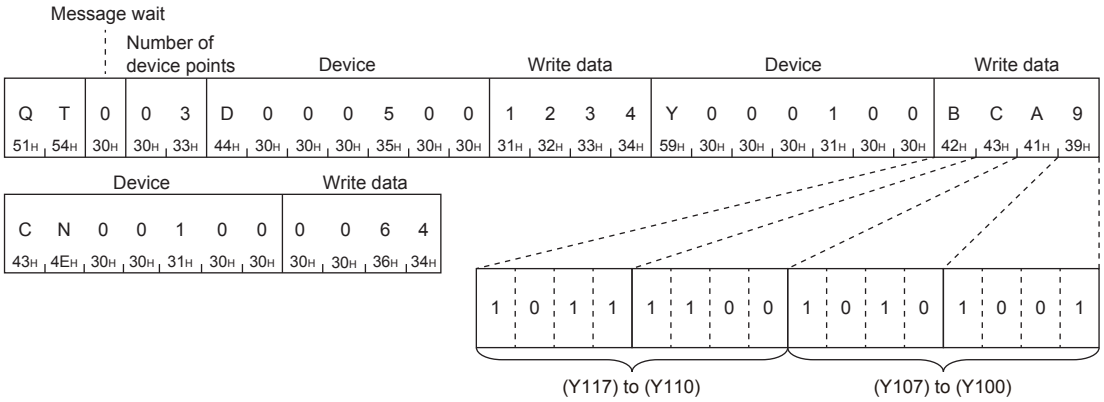
- Message wait: 0 ms
- Number of device points: 3 points (3 words)
- Device: Set 1234H to D500, BCA9H from Y100 to Y117, and 64H to current values of C100

(Request data)

■When using WT



■When using QT



5.5 Remote Operation

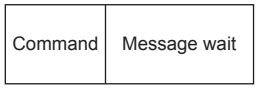
Remote RUN, Remote STOP (command: RR, RS)

Perform remote RUN (RR) and remote STOP (RS) of the FX5 PLC from the target device.

Message format

The following shows the message format of the request data and response data of the command.

■Request data



■Response data

There is no response data for RR and RS command.

Data specified by request data

■Command

RR command	RS command
	

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

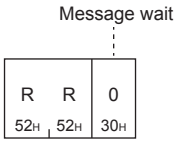
Communication example

Perform remote RUN or remote STOP of the PLC under the following conditions.

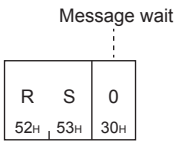
- Message wait: 0 ms

(Request data)

■When using RR



■When using RS



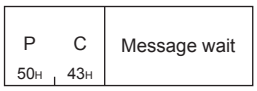
Read CPU model name (command: PC)

Read the model name of the PLC which links to the target device.

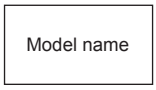
Message format

The following shows the message format of the request data and response data of the command.

■Request data



■Response data



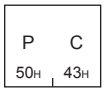
The read contents can be interpreted as the model names of PLC according to the following.

PLC model (CPU module)	Model code (hexadecimal)		PLC model (CPU module)	Model code (hexadecimal)
FX1S	F2H		A2ACPU-S1	93H
FX0N	8EH		A2CCPU	9AH
FX2(FX), FX2C	8DH		A2USCPU	82H
FX1N, FX1NC	9EH		A2CPU-S1, A2USCPU-S1	83H
FX2N, FX2NC	9DH		A3CPU, A3NCPU	A3H
FX3S	F5H		A3ACPU	94H
FX3G, FX3GC	F4H		A3HCPU, A3MCPU	A4H
FX3U, FX3UC	F3H		A3UCPU	84H
FX5S, FX5UJ, FX5U, FX5UC	F3H		A4UCPU	85H
A0J2HCPU	98H		A52GCPU	9AH
A1CPU, A1NCPU	A1H		A73CPU	A3H
A1SCPU, A1SJCPU	98H		A7LMS-F	A3H
A2CPU(-S1), A2NCPU(-S1), A2SCPU	A2H		AJ72P25/R25	ABH
A2ACPU	92H		AJ72LP25/BR15	8BH

Data specified by request data

■Command

PC command



■Message wait

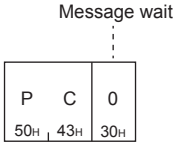
Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

Communication example

Read the model name of the PLC under the following conditions.

- Message wait: 0 ms

(Request data)



Model code setting

The model code read by the PC command can be changed freely using a special relay or register.

■Corresponding devices

The following devices are used for setting the model code.

Device number	Description	Default
SM8125	ON: The lower 8 bits of SD8126 are used as the model name. OFF: F3H (fixed value) is used as the model name.	OFF
SD8126	Store the model code to be changed to the lower 8 bits.	0

■Setting method

The procedure for changing the model code is shown below.

1. Set the model code in the lower 8 bits of SD8126.
2. Turn on SM8125.

The lower 8 bits of SD8126 are set in the response data.

5.6 Global Function

Global signal ON/OFF (command: GW)

Turn on and off the special relay of all stations or the specified station of the PLC at the multi drop link from the target device.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

G (47H)	W (57H)	Message wait	Factor number

■Response data

There is no response data for this command.

Data specified by request data

■Command

GW command
G (47H) W (57H)

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Factor number

For the special relay corresponding to the following CH, in the case of 1 (31H), turn on the special relay, and in the case of 0 (30H), turn off the special relay.

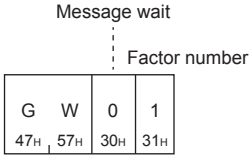
CH	Special relay corresponding to CH
1	SM8680
2	SM8690
3	SM8700
4	SM8710

Communication example

Turn on the special relay of the target station under the following conditions.

- Message wait: 0 ms

(Request data)



Point

When specifying all stations, specify FFH to the station number.
If the value other than FFH is specified, the special relay of the specified station number is turned on and off.

5.7 Loopback Test

A loopback test checks whether the communication function between an external device and FX5 CPU module operates normally.

Loopback test (Command: TT)

Return the characters received from an external device to the external device unchanged.

Message format

The following shows the message format of the request data and response data of the command.

■Request data

T 54H	T 54H	Message wait	Character length	Loopback data

■Response data

Character length	Loopback data

Data specified by request data

■Command

TT command

T 54H	T 54H

■Message wait

Specify the delayed time of the response transmission. (☞ Page 91 Message wait)

■Character length

Convert the numerical value to 2-digit ASCII code (hexadecimal), and send it from the upper digits.

Specify the character length within the following range:

- 1 ≤ Character length ≤ 254

■Loopback data

Store the loopback data for character length.

Data stored by response data

■Character length

The same data as request data is stored.

■Loopback data

The same data as request data is stored.

Point

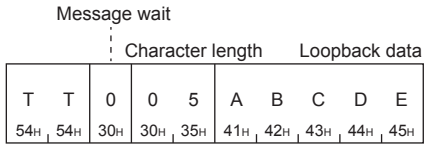
Specify 'FF' for PC No.

Communication example

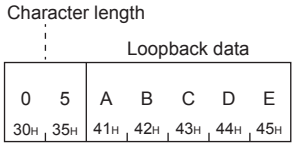
Return 5-digit data received from an external device to the external device unchanged under the following conditions.

- Message wait: 0 ms
- Character length: 5-digit
- Loopback data: 'ABCDE'

(Request data)



(Response data)



APPENDIX

Appendix 1 Device Memory Extension Specification

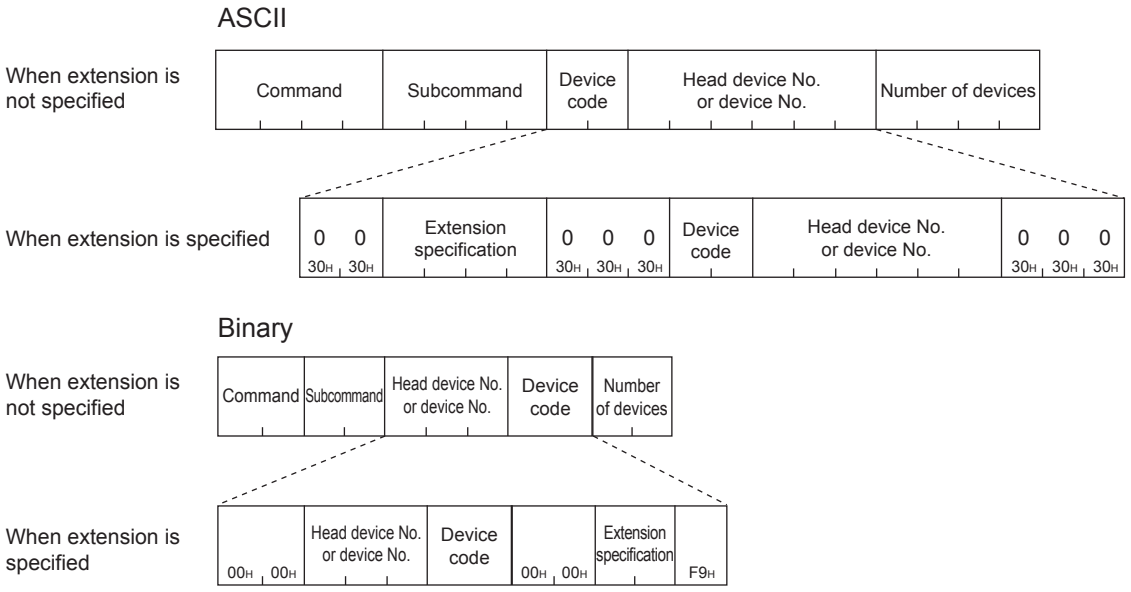
The following accesses are available by setting the subcommand of request data to 008☐.

- Access to module access device
- Access with indirect specification of the device No. by using index register or long index register
- Access with indirect specification of the device No. by using values stored in word device

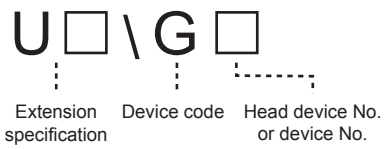
Access to module access device

Access to the buffer memory of MC protocol compatible devices or intelligent function modules.

Request data



The following shows the module access device and request data.



Point

Devices described in Page 37 Device number can be accessed by specifying 0 in "extension specification" of commands which can specify multiple devices. However, when specifying 008☐ in "subcommand", specify the device in the message format shown above. Message formats when extension is not specified and message formats when extension is specified cannot coexist in the same message.

■Command

The following commands can be used for accessing.

Function	Command
Batch read	0401
Batch write	1401
Random read	0403
Random write	1402
Batch read multiple blocks	0406
Batch write multiple blocks	1406

■Subcommand

Subcommand	
ASCII code	**Binary code**
0 0 8 0 30H 30H 38H 30H	80H 00H
0 0 8 2 30H 30H 38H 32H	82H 00H

■Extension specification

Specify the module number of intelligent function modules.

ASCII code	Binary code
Specify the module number in hexadecimal (ASCII code 3-digits). When described with 4-digits, specify the module number with the upper 3-digits. U □ □ □ 55H Example 001 U 0 0 1 55H 30H 30H 31H	Specify the module number in hexadecimal (2 bytes). When described with 4-digits, specify the module number with the upper 3-digits. □□H □□H Example 001 01H 00H

■Device code

Specify the following device codes.

Type	Device code				Device No. range	
	ASCII code*1		Binary code			
	2 digit code/ 6 digit number specification	4 digit code/ 8 digit number specification	2 digit code/ 6 digit number specification	4 digit code/ 8 digit number specification		
Word	G*	G***	ABH	AB00H	Specify within the device No. range of the module for access destination.	Decimal

1 For ASCII codes, the device code is specified with 2 characters. If the device text is one character only, add "" (ASCII code: 2AH) or a space (ASCII code: 20H) after the device text.

■Head device or device No.

Specify the head device or device No. in decimal, with the same format as the message when extension is not specified.

Point

Indirect specification of the access target device No. can be performed by using the CPU module index register (Z) or long index register (LZ). (☞ Page 115 Access with indirect specification of the device No. by using index register or long index register)

Response data

The same as when extension is not specified.

Communication example

Access to the buffer memory (Address: 1) of the intelligent function module whose module number is 003H.

• When communicating data in ASCII code

(Request data)

Subcommand				Extension specification							Device code		Head device No. or device No.								
0	0	8	0	0	0	U	0	0	3	0	0	0	G	*	0	0	0	0	0	1	0	0	0
30H	30H	38H	30H	30H	30H	55H	30H	30H	33H	30H	30H	30H	47H	2AH	30H	30H	30H	30H	30H	31H	30H	30H	30H

• When communicating data in binary code

(Request data)

Subcommand		Head device No. or device No.			Device code	Extension specification		
80H	00H	00H	00H	01H	00H	00H	ABH	00H	00H	03H	00H	F8H

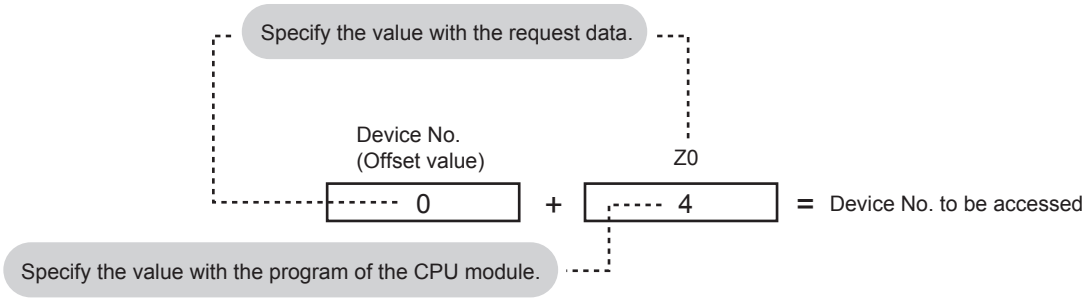
Access with indirect specification of the device No. by using index register or long index register

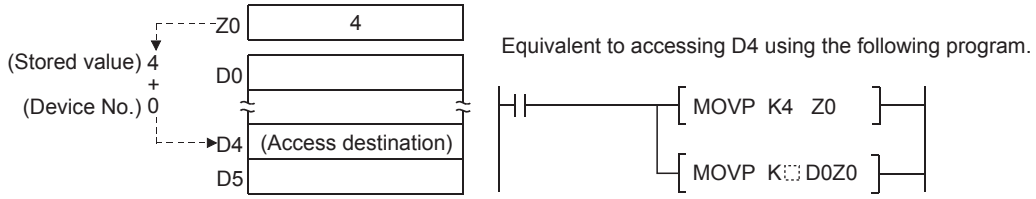
Indirect specification of the device No. can be performed by using the index register or long index register when accessing the device.

The access destination can be switched with one message, by changing the value of the index register or long index register in CPU module programs.

Ex.

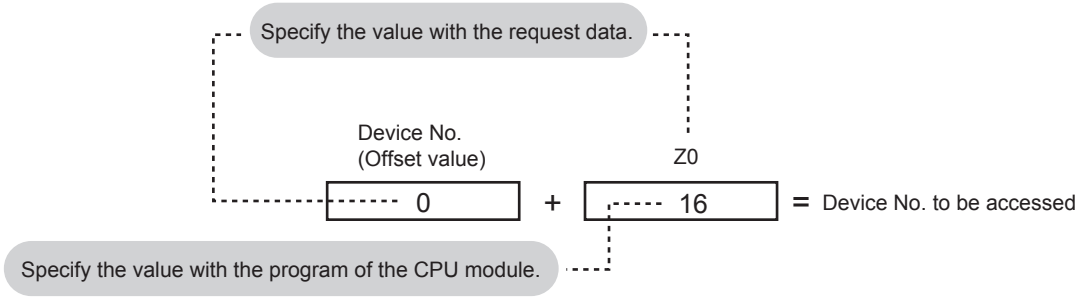
When accessing D4 with D0 and Z0 specifications

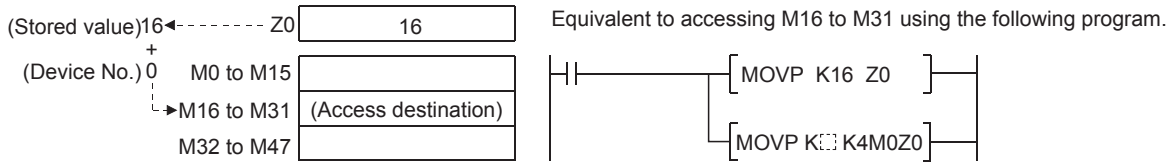




Ex.

When accessing M16 to M31 with M0 and Z0 specifications (Word units)



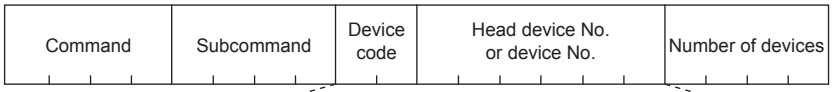


A

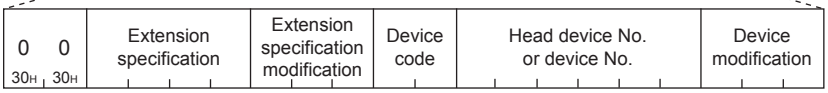
Request data

ASCII

When extension is not specified

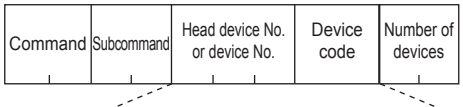


When extension is specified

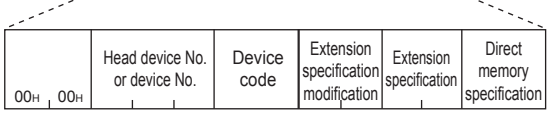


Binary

When extension is not specified

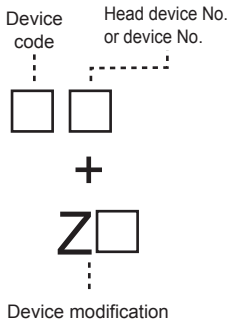


When extension is specified

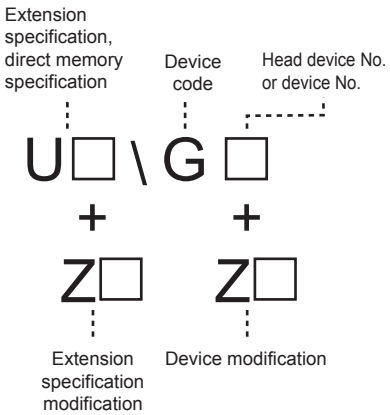


The following shows the approach for devices, index registers, long index registers and request data.

- Other than the module access device



- Module access device



Point

When specifying 008□ in "subcommand", specify the device with the message format shown above.
Message formats when extension is not specified and message formats when extension is specified cannot coexist in the same message.

■Command

The following commands can be used for accessing.

Function	Command
Random read	0403
Random write	1402

■Subcommand

Item	Subcommand	
	ASCII code	Binary code
When accessing in bit units	0 0 8 1 30H 30H 38H 31H	81H 00H
	0 0 8 3 30H 30H 38H 33H	83H 00H
When accessing in word units	0 0 8 0 30H 30H 38H 30H	80H 00H
	0 0 8 2 30H 30H 38H 32H	82H 00H

■Extension specification

Specify the module number.

The values specified in this item turn to the offset value when performing indirect specification of the module number in "extension specification modification".

Item	ASCII code	Binary code
Module access device	Specify the module number in hexadecimal (2 bytes). U □ □ □ 55H Example 001 U 0 0 1 55H 30H 30H 31H	Specify the module number in hexadecimal (2 bytes). □□H □□H Example 001 01H 00H
Devices other than the above	Specify 0. 0 0 0 0 30H 30H 30H 30H	Specify 0. 00H 00H

■Extension specification modification

Treat the value specified in "extension specification" as the offset value. Specify the index register or long index register number when performing indirect specification of the module number with index register or long index register.

- The following value is specified when the access point is a module of the MELSEC iQ-R/iQ-F Series.

Subcommand	ASCII code	Binary code
0083 0082	Specify the number of the index register in decimal (2-digit ASCII code). • FX5S/FX5U/FX5UC CPU module: 0 to 23 • FX5UJ CPU module: 0 to 19 Z □ □ 5AH 20H	Specify the number of the index register (Z) in hexadecimal. • FX5S/FX5U/FX5UC CPU module: 00H to 17H • FX5UJ CPU module: 00H to 13H □□H 40H
0081 0080	Specify the number of the index register in decimal (2-digit ASCII code). • FX5S/FX5U/FX5UC CPU module: 0 to 23 • FX5UJ CPU module: 0 to 19 Z □ □ 5AH	Specify the number of the index register (Z) in hexadecimal. • FX5S/FX5U/FX5UC CPU module: 00H to 17H • FX5UJ CPU module: 00H to 13H □□H 40H

• The following value is specified when the access point is a module of the MELSEC Q/L Series.

ASCII code	Binary code
Specify the number of the index register in decimal (2-digit ASCII code). (Specification range: 0 to 15) 	Specify the number of the index register in hexadecimal. (Specification range: 0 to F) 

Point

The long index register (LZ) can not be used in the extension specification modification.

■Device code

Specify the code of the device to be accessed. (☞ Page 37 Device code list)

Specify the following device code when accessing the module access device.

Type	Device code				Device No. range	
	ASCII code*1		Binary code			
	2 digit code/6 digit number specification	4 digit code/8 digit number specification	2 digit code/6 digit number specification	4 digit code/8 digit number specification		
Word	G*	G***	ABH	AB00H	Specify within the device No. range of the module for access destination.	Decimal

1 For ASCII codes, the device code is specified with 2 characters. If the device text is one character only, add "" (ASCII code: 2AH) or a space (ASCII code: 20H) after the device text.

■Head device or device No.

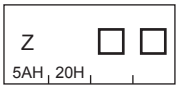
Specify the head device or device No. with the same format as the message when extension is not specified.

The values specified in this item turn to the offset value when performing indirect specification of the device No. in "device modification".

■Device modification

Treat the value specified in "Head device or device No." as the offset value. Specify the index register or long index register number when performing indirect specification of the device No. with index register or long index register.

• The following value is specified when the access point is a module of the MELSEC iQ-R/iQ-F Series.

Subcommand	ASCII code	Binary code
0083 0082	Specify the number of the index register in decimal (2-digit ASCII code).*1 • FX5S/FX5U/FX5UC CPU module: 0 to 23 • FX5UJ CPU module: 0 to 19 Specify the number of the long index register (LZ) in decimal (2-digit ASCII code). • FX5S/FX5U/FX5UC CPU module: 0 to 11 • FX5UJ CPU module: 0, 1  	Specify the number of the index register (Z) in hexadecimal.*1 • FX5S/FX5U/FX5UC CPU module: 00H to 17H • FX5UJ CPU module: 00H to 13H Specify the number of the long index register (LZ) in hexadecimal. • FX5S/FX5U/FX5UC CPU module: 00H to 0BH • FX5UJ CPU module: 00H, 01H  
0081 0080	Specify the number of the index register in decimal (2-digit ASCII code). • FX5S/FX5U/FX5UC CPU module: 0 to 23 • FX5UJ CPU module: 0 to 19 	Specify the number of the index register (Z) in hexadecimal. • FX5S/FX5U/FX5UC CPU module: 00H to 17H • FX5UJ CPU module: 00H to 13H 

*1 The device modification range of the index register (Z) is -32768 to 32767. When the device modification range is not within -32768 to 32767, use the long index register (LZ).

• The following value is specified when the access point is a module of the MELSEC Q/L Series.

ASCII code	Binary code
Specify the number of the index register in decimal (2-digit ASCII code). (Specification range: 0 to 15) 	Specify the number of the index register in hexadecimal. (Specification range: 0 to F) 

■Direct memory specification (only when communicating in binary code)

Specify the device type when accessing the module access device.

Item	Binary code
Module access device	Specify F8H.
Other than the above	Specify 00H.

Response data

The same as when extension is not specified.

Communication example

Accessing the device of D100 + Z4.

• When communicating data in ASCII code

(Request data)

Subcommand	Extension specification	Extension specification modification	Device code	Head device No. or device No.	Device modification
0 0 8 0	0 0	0 0 0	D *	0 0 0 1 0 0	Z 0 4
30H 30H 38H 30H	30H 30H	30H 30H 30H 30H	30H 30H 30H	44H 2AH	30H 30H 30H 31H 30H 30H	5AH 30H 34H

• When communicating data in binary code

(Request data)

Subcommand	Device modification	Head device No. or device No.	Device code	Extension specification modification	Extension specification	Direct memory specification
80H 00H	04H 40H	64H 00H 00H	A8H	00H 00H	00H 00H	00H

Access with indirect specification of the device No. by using the values stored in word device

Access the device corresponding to the address stored in word device (for 2 points).

Ex.

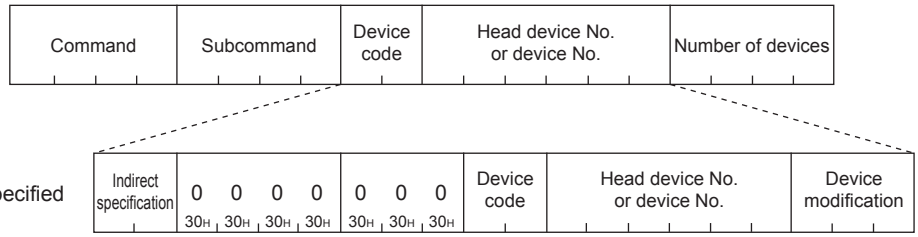
When storing the address of D100 in D0, and trying to access D100 from external devices by accessing "@D0"

The ADRSET instruction is used on the CPU module side and the address of D100 is stored in D0.

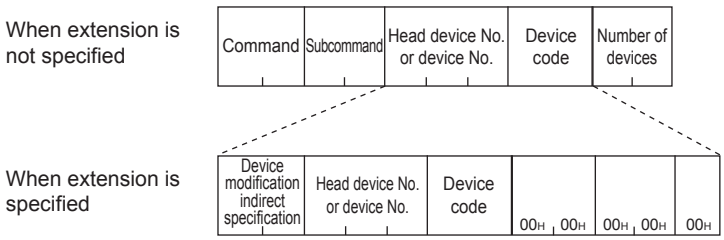
D100 can be indirectly accessed by specifying "@D0" with the request data.

Request data

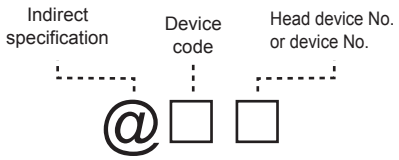
ASCII



Binary



The following shows the indirect specification devices and request data.



Point

- When specifying 008☐ in "subcommand", specify the device with the message format shown above. Message formats when extension is not specified and message formats when extension is specified cannot coexist in the same message.
- The indirect specification and the device modification using index registers can not be set simultaneously.

■Command

The following commands can be used for accessing.

Function	Command
Random read	0403
Random write	1402

■Subcommand

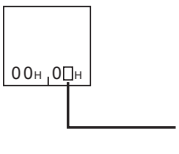
ASCII code	Binary code
0 0 8 0 30H 30H 38H 30H	80H 00H

■Indirect specification, Device modification

Specify the "@" part of the indirect specification device. Indirect specification can be specified only for word devices.

When communicating data in ASCII code

When communicating data in binary code



■Device code (Only word device codes can be specified at indirect specification)

Specify the code of the device to be accessed. (☞ Page 37 Device code list)

■Head device or device No.

Specify the head device or device No. with the same format as the message when extension is not specified.

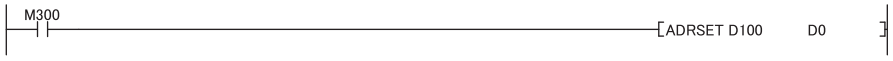
Response data

The same as when extension is not specified.

Communication example

Access to @D0. (Consider @D0 indirect specification of D100.)

At command execution, store the D100 address in D0 with the following programs.



• When communicating data in ASCII code

(Request data)

Subcommand				Indirect specification		Device code		Head device No. or device No.						Device modification		
0	0	8	0	0	@	0	0	0	0	0	0	0	D	*	0	0	0	0	0	0	0	0	0
30H	30H	38H	30H	30H	40H	30H	30H	30H	30H	30H	30H	30H	44H	2AH	30H	30H	30H	30H	30H	30H	30H	30H	30H

• When communicating data in binary code

(Request data)

Subcommand	indirect specification	Head device No. or device No.	Device code			
80H 00H	00H 08H	00H 00H 00H	A8H	00H 00H	00H 00H	00H

Appendix 2 CPU Module Processing Time of MC Protocol

When accessing the CPU module from an external device using MC protocol communication, the following "intervention time to the scan time" and "number of scans for processing" of the CPU module side are required. On the request from the external device using MC protocol communication, the CPU module processes a specified number of points during each END processing in case the CPU module is running.

Item	Command	Subcommand	Access points 1) / 2)	Intervention time [ms]*2 (extension of scan time)				Number of scans required for processing
				Access point 1)		Access point 2)		
				FX5S	FX5UJ/ FX5U/ FX5UC	FX5S	FX5UJ/ FX5U/ FX5UC	
Batch read	0401	0001	1/3584	0.04	0.04	1.17	1.17	1
		0000	1/960	0.04	0.04	1.15	0.96	1
Batch write	1401	0001	1/3584	0.04	0.04	1.26	1.26	1
		0000	1/960	0.05	0.05	1.21	1.16	1
Random read	0403	0000	1/192	0.06	0.06	3.33	2.60	1
Random write	1402	0001	1/188	0.04	0.04	2.86	2.08	1
		0000	1/160*1	0.05	0.04	2.90	2.28	1
Batch read multiple blocks	0406	0000	1/960	0.05	0.05	1.16	0.96	1
Batch write multiple blocks	1406	0000	1/770	0.05	0.04	1.05	0.94	1
Read CPU model name	0101	0000	(one station)	0.03	0.03	—	—	1

*1 This is the processing time when accessing with only word access points specified.

*2 This is the processing time when 1 is set to [CPU Parameter] - [Service Processing Setting] - [Device/Label Access Service Processing Setting] - [Set Processing Counts] of GX Works3.

Point

• Number of scans required for processing

The CPU module processes only one command during an END processing. If GX Works3 or other modules are also accessing the CPU module simultaneously, the number of scans required for processing may increase due to the waiting time.

• Method of reducing the intervention time to the scan time

Adjust the service process execution count of the CPU module in [CPU Parameter] - [Service Processing Setting] - [Device/Label Access Service Processing Setting] to reduce the intervention time to the scan time. (For details, refer to the MELSEC iQ-F FX5 User's Manual (Application))

• When extension of scan time affects the control

Access multiple times with less points.

Appendix 3 Added and Enhanced Functions

The functions added or changed with the CPU module and engineering tool, and the supported CPU modules' firmware version and engineering tool software version are given below.

The firmware version can be confirmed with module diagnosis (CPU diagnosis). Refer to the following manuals for details on diagnosing the module (CPU diagnosis).

MELSEC iQ-F FX5S/FX5UJ/FX5U/FX5UC User's Manual (Hardware)

Refer to the GX Works3 Operating Manual for details on the software version.

FX5S CPU module

Add/Change Function	CPU module firmware version	Engineering tool software version	Reference
MC protocol	First released product or later	"1.080J" or later	—

FX5UJ CPU module

Add/Change Function	CPU module firmware version	Engineering tool software version	Reference
MC protocol	First released product or later	"1.060N" or later	—

FX5U/FX5UC CPU module

Add/Change Function	CPU module firmware version	Engineering tool software version	Reference
Message format: format 1 (X, Y HEX), format 4 (X, Y HEX)	"1.040" or later	"1.030G" or later	Page 16
1C frame	"1.110" or later	"1.050C" or later	Page 12 Page 89

A

INDEX

MEMO

REVISIONS

Revision date	Revision	Description
February 2015	A	First Edition
October 2016	B	■Added models FX5U-32MR/DS, FX5U-32MT/DS, FX5U-32MT/DSS, FX5U-64MR/DS, FX5U-64MT/DS, FX5U-64MT/DSS, FX5U-80MR/DS, FX5U-80MT/DS, FX5U-80MT/DSS, FX5UC-64MT/D, FX5UC-64MT/DSS, FX5UC-96MT/D, FX5UC-96MT/DSS ■Added function Message format: format 1 (X, Y HEX), format 4 (X, Y HEX) ■Added or modified parts SAFETY PRECAUTIONS, RELEVANT MANUALS, TERMS, Section 1.2, 2.4, 3.1, 3.2, 3.3, 3.5, 4.2, 4.3, Appendix 1, 3, WARRANTY
October 2018	C	■Added models FX5UC-32MT/DS-TS, FX5UC-32MT/DSS-TS ■Added function 1C frame ■Added or modified parts RELEVANT MANUALS, TERMS, Section 1.1, 2.1, 2.4, 3.1, 3.3, 3.4, 3.5, 4.1, 4.3, Chapter 5, Appendix 3
October 2019	D	■Added models FX5UJ-24MR/ES, FX5UJ-24MT/ES, FX5UJ-24MT/ESS, FX5UJ-40MR/ES, FX5UJ-40MT/ES, FX5UJ- 40MT/ESS, FX5UJ-60MR/ES, FX5UJ-60MT/ES, FX5UJ-60MT/ESS, FX5UC-32MR/DS-TS ■Added or modified parts RELEVANT MANUALS, TERMS, Section 2.1, 3.1, 3.2, 3.4, 4.1, 4.2, 4.3, 5.5, Appendix1, 2, 3, TRADEMARKS
May 2020	E	■Added or modified parts RELEVANT MANUALS, TERMS, Section 2.1, TRADEMARKS
September 2020	F	■Added or modified parts SAFETY PRECAUTIONS, WARRANTY
April 2022	G	■Added model FX5S CPU module ■Added or modified parts RELEVANT MANUALS, TERMS, GENERIC TERMS AND ABBREVIATIONS, Section 4.3, 5.5, Appendix 1, 3

This manual confers no industrial property rights or any rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

©2015 MITSUBISHI ELECTRIC CORPORATION

WARRANTY

Please confirm the following product warranty details before using this product.

1. Gratis Warranty Term and Gratis Warranty Range

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the sales representative or Mitsubishi Service Company. However, if repairs are required onsite at domestic or overseas location, expenses to send an engineer will be solely at the customer's discretion. Mitsubishi shall not be held responsible for any re-commissioning, maintenance, or testing on-site that involves replacement of the failed module.

[Gratis Warranty Term]

The gratis warranty term of the product shall be for one year after the date of purchase or delivery to a designated place. Note that after manufacture and shipment from Mitsubishi, the maximum distribution period shall be six (6) months, and the longest gratis warranty term after manufacturing shall be eighteen (18) months. The gratis warranty term of repair parts shall not exceed the gratis warranty term before repairs.

[Gratis Warranty Range]

(1) The range shall be limited to normal use within the usage state, usage methods and usage environment, etc., which follow the conditions and precautions, etc., given in the instruction manual, user's manual and caution labels on the product.

(2) Even within the gratis warranty term, repairs shall be charged for in the following cases.
   1. Failure occurring from inappropriate storage or handling, carelessness or negligence by the user. Failure caused by the user's hardware or software design.
   2. Failure caused by unapproved modifications, etc., to the product by the user.
   3. When the Mitsubishi product is assembled into a user's device, Failure that could have been avoided if functions or structures, judged as necessary in the legal safety measures the user's device is subject to or as necessary by industry standards, had been provided.
   4. Failure that could have been avoided if consumable parts (battery, backlight, fuse, etc.) designated in the instruction manual had been correctly serviced or replaced.
   5. Relay failure or output contact failure caused by usage beyond the specified life of contact (cycles).
   6. Failure caused by external irresistible forces such as fires or abnormal voltages, and failure caused by force majeure such as earthquakes, lightning, wind and water damage.
   7. Failure caused by reasons unpredictable by scientific technology standards at time of shipment from Mitsubishi.
   8. Any other failure found not to be the responsibility of Mitsubishi or that admitted not to be so by the user.

2. Onerous repair term after discontinuation of production

(1) Mitsubishi shall accept onerous product repairs for seven (7) years after production of the product is discontinued.
Discontinuation of production shall be notified with Mitsubishi Technical Bulletins, etc.

(2) Product supply (including repair parts) is not available after production is discontinued.

3. Overseas service

Overseas, repairs shall be accepted by Mitsubishi's local overseas FA Center. Note that the repair conditions at each FA Center may differ.

4. Exclusion of loss in opportunity and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to:

(1) Damages caused by any cause found not to be the responsibility of Mitsubishi.

(2) Loss in opportunity, lost profits incurred to the user by Failures of Mitsubishi products.

(3) Special damages and secondary damages whether foreseeable or not, compensation for accidents, and compensation for damages to products other than Mitsubishi products.

(4) Replacement by the user, maintenance of on-site equipment, start-up test run and other tasks.

5. Changes in product specifications

The specifications given in the catalogs, manuals or technical documents are subject to change without prior notice.

6. Product application

(1) In using the Mitsubishi MELSEC programmable controller, the usage conditions shall be that the application will not lead to a major accident even if any problem or fault should occur in the programmable controller device, and that backup and fail-safe functions are systematically provided outside of the device for any problem or fault.

(2) The Mitsubishi programmable controller has been designed and manufactured for applications in general industries, etc. Thus, applications in which the public could be affected such as in nuclear power plants and other power plants operated by respective power companies, and applications in which a special quality assurance system is required, such as for railway companies or public service purposes shall be excluded from the programmable controller applications.
In addition, applications in which human life or property that could be greatly affected, such as in aircraft, medical applications, incineration and fuel devices, manned transportation, equipment for recreation and amusement, and safety devices, shall also be excluded from the programmable controller range of applications.
However, in certain cases, some applications may be possible, providing the user consults their local Mitsubishi representative outlining the special requirements of the project, and providing that all parties concerned agree to the special circumstances, solely at the user's discretion.

(3) Mitsubishi shall have no responsibility or liability for any problems involving programmable controller trouble and system trouble caused by DoS attacks, unauthorized access, computer viruses, and other cyberattacks.

TRADEMARKS

Anywire and AnyWireASLINK are either registered trademarks or trademarks of Anywire Corporation.
The company names, system names and product names mentioned in this manual are either registered trademarks or trademarks of their respective companies.
In some cases, trademark symbols such as '™' or '®' are not specified in this manual.

Manual number: JY997D60801G
MODEL: FX5-U-MCPRO-E
MODEL CODE: 09R556

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS : 1-14 , YADA-MINAMI 5-CHOME , HIGASHI-KU, NAGOYA , JAPAN

When exported from Japan, this manual does not require application to the Ministry of Economy, Trade and Industry for service transaction permission.

Specifications subject to change without notice.