NAiS

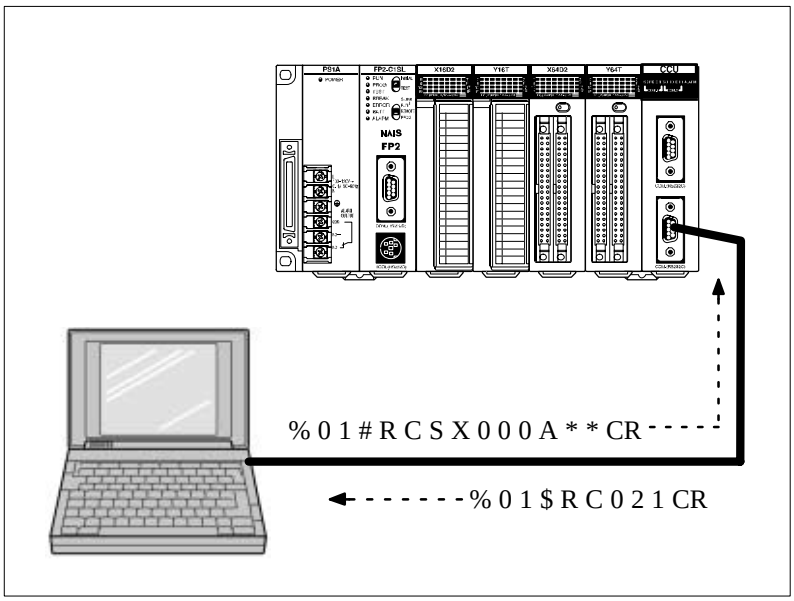
FP MEWTOCOL Protocol

Matsushita Electric Works (Europe) AG

ACGM0125V1.0END

7/2002

FP SERIES MEWTOCOL
Protocol



Smart Solutions by NAiS

BEFORE BEGINNING

Matsushita Electric Works, Ltd. pursues a policy of continuous improvement of the design and performance of its products, therefore, we reserve the right to change the manual/product without notice. In no event will Matsushita Electric Works, Ltd. be liable for direct, special, incidental, or consequential damage resulting from any defect in the product or its documentation, even if advised of the possibility of such damages.

Table of Contents

Chapter 1 MEWTOCOL Protocol

Chapter 2 MEWTOCOL–DAT Protocol

Chapter 3 MEWTOCOL Error Codes

Index

Record of Changes

Chapter 1

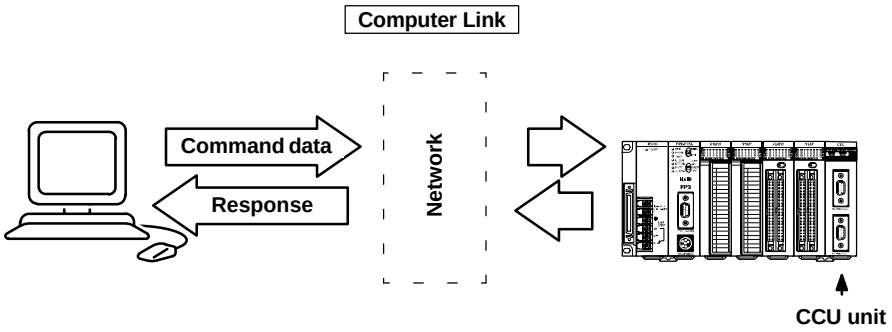
MEWTOCOL Protocol

1.1 MEWTOCOL Format, Introduction

MEWTOCOL is the communication protocol for FP series programmable controllers of which two types are supported as follows:

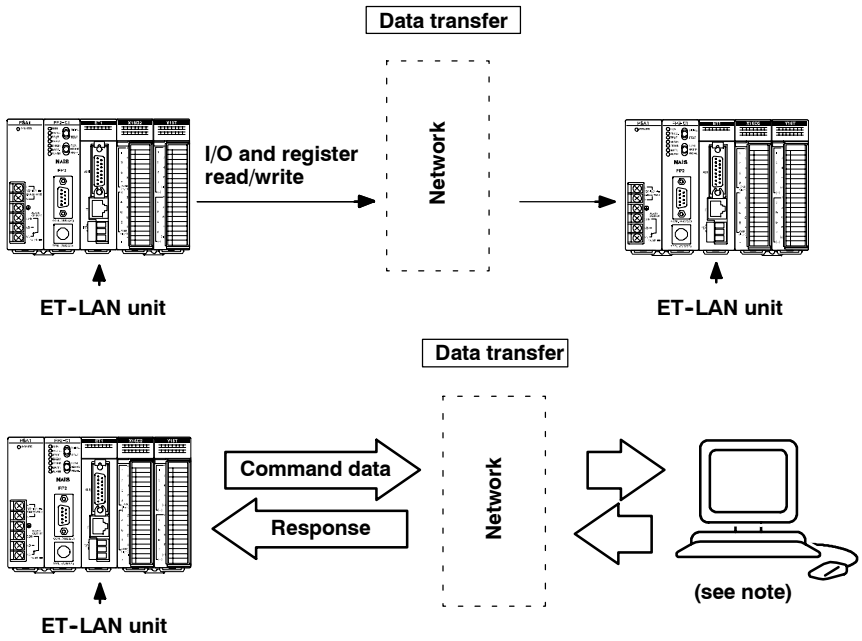
MEWTOCOL–COM Protocol:

MEWTOCOL–COM protocol is used for communication between an FP series programmable controller and a computer. Communication using MEWTOCOL–COM protocol is referred to as the computer link function. In the computer link, a computer always initiates communication by sending a MEWTOCOL–COM command message. The FP series programmable controller then returns back a response message to computer. Using this computer link, you do not need to create a communication program in the FP series programmable controller, but you do need to create a communication program in order to accommodate the MEWTOCOL–COM format. You can use any programming language such as BASIC or C to program the computer.



MEWTOCOL-DAT Protocol:

The MEWTOCOL-DAT protocol is used for communication between FP series programmable controllers or between an FP series programmable controller and a computer. Communication using the MEWTOCOL-DAT protocol is called the data transfer function. The data transfer function is performed through link units, such as the MEWNET-P, MEWNET-W, and ET-LAN unit, by executing the F145 (SEND)/P145 (PSEND) and F146 (RECV)/P146 (PRECV) instructions. During a data transfer, communication is usually initiated by the FP3, FP2, FP2SH, or FP10SH with a "link" unit (MEWNET Link P or W, MW2, ET1 or CCU unit) by executing the instructions, which means sending a MEWTOCOL-DAT command message. Then the response message is received from another FP3/FP2/FP2SH/FP10SH or a computer. When doing data transfers between FP series programmable controllers, you only need to create a program in one of the programmable controllers for executing the F145 (SEND)/P145 (PSEND) and F146 (RECV)/P146 (PRECV) instructions. You do not have to make a program in the other FP series programmable controller. For communication between an FP series programmable controller and a computer, you need to execute the F145 (SEND)/P145 (PSEND) and F146 (RECV)/P146 (PRECV) instructions in the FP series programmable controller and you need to create a program in order to accommodate the MEWTOCOL-DAT format. You can use any program language such as BASIC or C to program the computer.



It is also possible to receive the command and send a response from the computer side.

1.2 MEWTOCOL–COM Protocol

The MEWTOCOL–COM protocol is used for communication between a computer and an FP series programmable controller (computer link function). During computer link communication, the command is initiated from a computer and the FP series programmable controller sends a response message back to the computer in the MEWTOCOL–COM format.

All messages are transmitted as ASCII codes. Therefore, all characters you send to or receive from an FP series programmable controller should be converted to ASCII code.

☞ **Basic MEWTOCOL–COM terminology**

- Message**: A series of characters combining commands and text which are sent in one or more frames.**
- Command message**: A message from the computer to the FP series programmable controller.**
- Response message**: A message from the FP series programmable controller to the computer.**
- Frame**: A group of not more than 118 (when using % a header) or 2,048 (when using < as header) characters. Note that the < header is available only for high–level link units.**

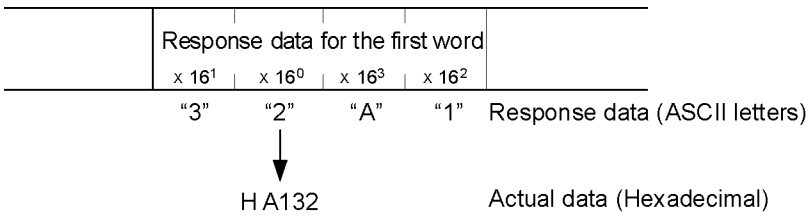
Number systems used in MEWTOCOL–COM messages

Three types of numbering systems are used for MEWTOCOL–COM messages as follows:

- Hexadecimal numbers

  Hexadecimal numbers are used for expressing the contents of the registers or error codes.

  [EXAMPLE]
  In the response message of the RD command, response data is expressed in hexadecimal numbers.

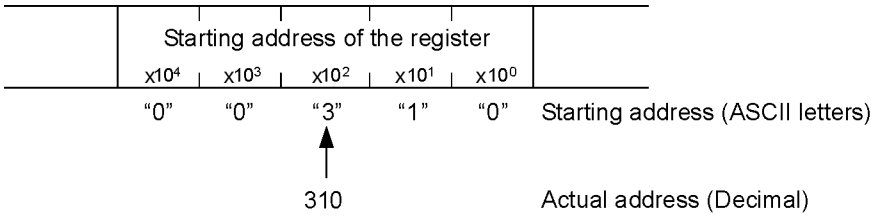


- Decimal numbers

  Decimal numbers are used for expressing the addresses for registers and timer/counter contacts.

  [EXAMPLE]

  In the command message of the RD command, the address is expressed in decimal numbers.

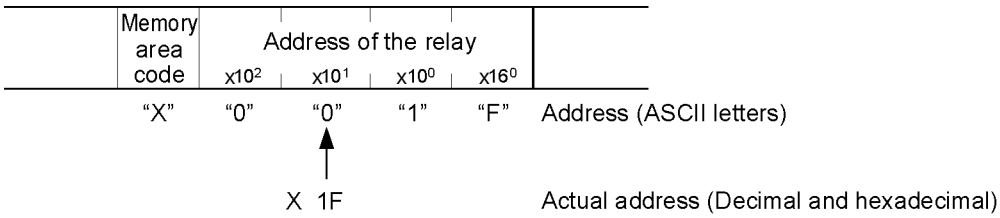


- Combination of decimal and hexadecimal numbers

  Address for relay bits (X, Y, R and L) are expressed as a combination of a word address (decimal) and a hexadecimal number for designating a specific bit. The right most digit is hexadecimal and the rest of the digits are decimal.

  [EXAMPLE]

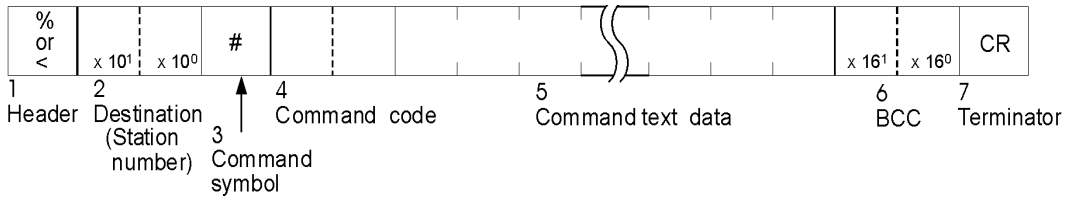
  In the RC command message for reading a single data bit, its address is expressed in a combination of decimal and hexadecimal numbers.



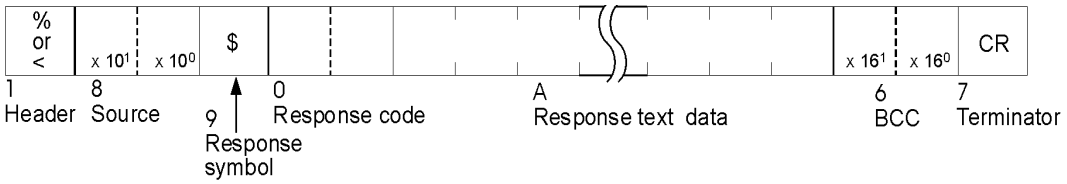
☞ **Since there are restrictions in the digits that can be used for expressing data and addresses, etc., in each command and response message, be sure to refer to each description of MEWTOCOL–COM commands and responses.**

1.2.1 Basic MEWTOCOL–COM Message Format

Command message format

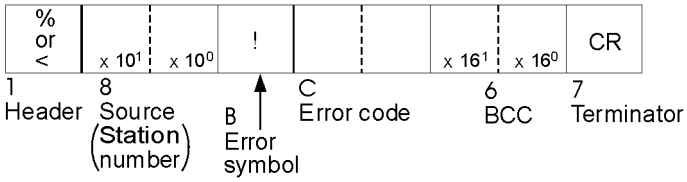


Response message format



Error response message

When an error occurs during data transmission, the following response will be returned by a programmable controller.



1. Header [“%“ (ASCII code : H25) or “<” (ASCII code: H3C)]
   The percent character “%“ is used for the header in both command and response messages frames, for up to and including 118 characters. The character “<“ is used for the header in both command and response message frames for up to and including 2,048 characters. The “<” header is available for high–level link units, such as the ET–LAN unit or MEWNET–H link unit.
2. Destination (Station number) [“01” through “63“ (decimals) or “FF“]
   The station that should read the command message is specified as 2 characters representing a decimal station number. Accordingly, the station number must be specified in the range of “01“ to “63“. You also can specify it as “FF“ to send the command message to all of the stations. In this case, no response message will be returned.

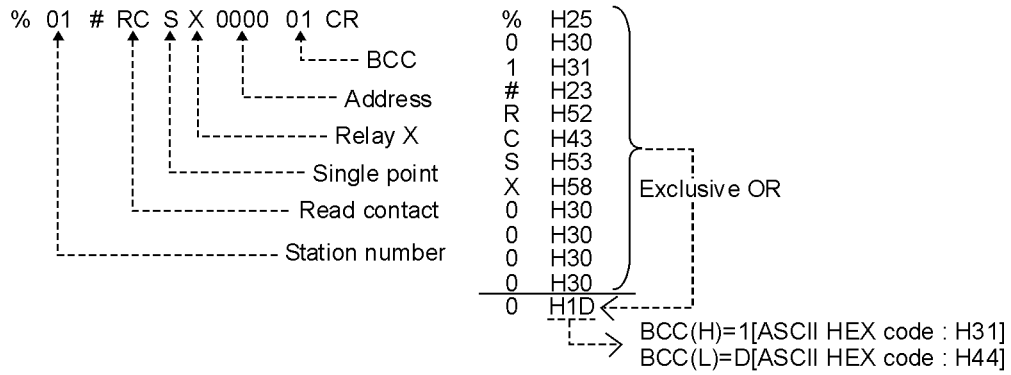
☞ **When the number is FF (ASCII code), it represents a global transfer (broadcast to all units). When a global transfer is performed, a response message for the command message is not returned. In this case, the global relay (CR97F) is turned ON after processing ends for the command at the PLC.**

3. Command symbol [“#“(ASCII code : H23)]
   The pound sign “#“ is used for the command symbol.
4. Command code [2 characters (capital letters)]
   The command code is specified as 2 uppercase characters. For details of the command codes, refer to page 135, “5) List of MEWTOCOL–COM Command/Response Codes“.

5. Command text data
   Depending on the command, the content of text data will vary.
   Information such as memory address that subjected to the data transmission, and data (if any), will be specified here.

6. Block Check Code (BCC) [2 characters]
   This code is used to detect errors in the message transmissions.
   If "**" is sent from a computer as the BCC, no block check will be performed on the command message. Even if a computer sending a command message has specified that no BCC is being sent, the receiving station will insert its own BCC in the response message.
   It is created by Exclusive ORing all of the codes from the header through the last text character, then translating the resulting 8–bit data into two ASCII characters.

Example :



7. Terminator [CR (ASCII code : H0D)]
   The carriage return "CR" is used as the terminator in both command and response messages.

8. Source (Station number) ["01" through "63" (decimals) or "FF"]
   The station number specified in the command message as the destination will be returned as source station number.

9. Response symbol ["$" (ASCII code: H24)]
   The dollar sign "$" is used in the response message. This indicates that a data transmission was successfully received.

10. Response code [2 characters (capital letters)]
    The same code as the one sent in the command message will be returned to indicate the programmable controller is responding to the command message.

11. Response text data
    When data must be returned in the response message, the response text data is added after the response code. For example, when a register read command (RD) is sent from a computer, the programmable controller will respond with text data.

12. Error symbol [!] (ASCII code: H16)]
    The exclamation character "!" is used to identify an error message. This indicates that a data transmission error occurred.

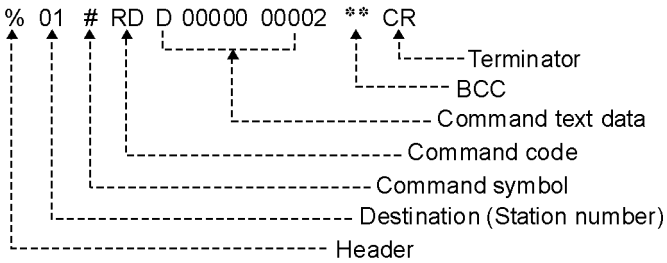
13. Error code [2 characters (hexadecimals)]
    The error code is specified as 2–character hexadecimal number expressed in ASCII format. For details about MEWTOCOL–COM error codes, see chapter 3.

Program example

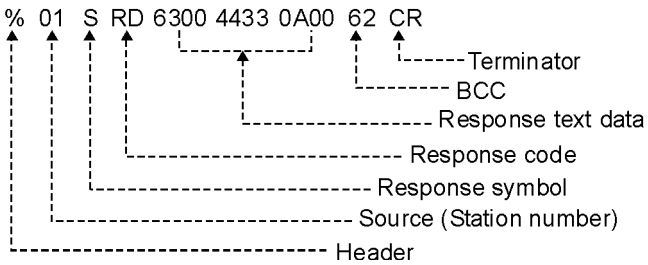
Reading data from data registers, DT0000 through DT0002 in a programmable controller which has assigned number is 01.

The data in the data registers are:
DT0000 0063 (Hexadecimal)
DT0001 3344 (Hexadecimal)
DT0002 000A (Hexadecimal)

Command message



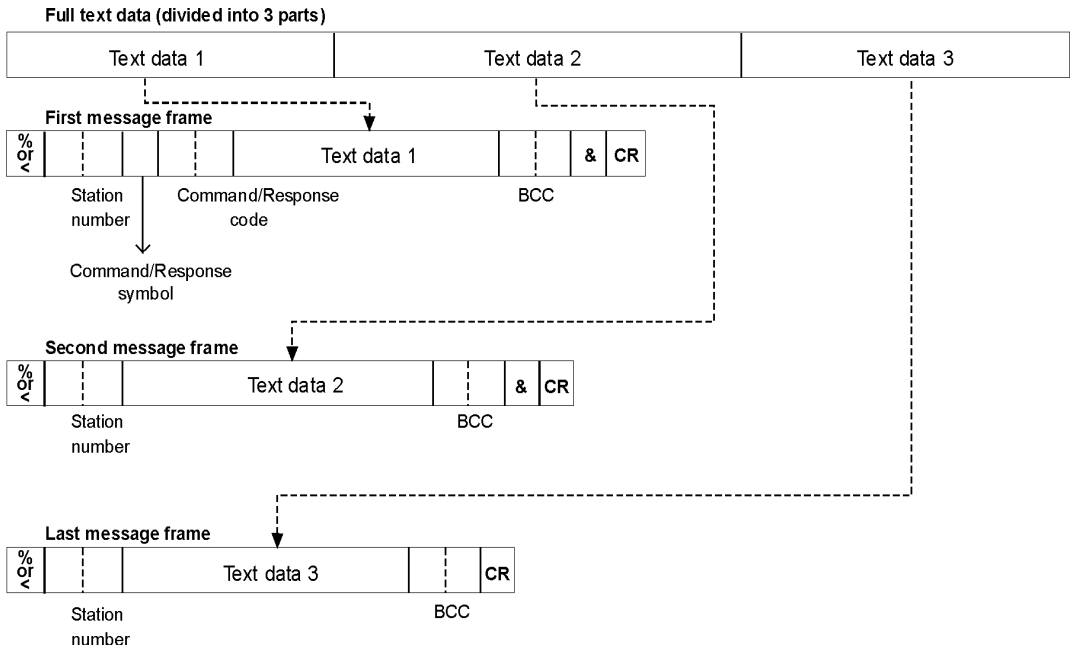
Response message



1.2.2 Multiple MEWTOCOL–COM Frames

The maximum of message length that the link unit can receive or send at one time is 118 characters when using the “%“ header and 2,048 characters when using the “<“ header. If the message to be sent exceeds specified limits, it must be divided into separate frames as shown below.

How to divide a message into multiple frames



The characters included in each frame are slightly different.

- 1st frame
  The delimiter character “&“ is added after the BCC.
  In all other respects it is just like a single frame message.

- 2nd (and 3rd, etc.) frames
  The second, third, etc. frames do not use the command or response symbols (“#“, “$“), but the second frame does require the “&“ character between the BCC and the terminator (CR).

- Last frame
  The last frame does not use the command or response symbols (“#“, “$”).
  It also does not include the “&“ delimiter character. In other words, it is just like a regular message frame, without a command or response symbol.

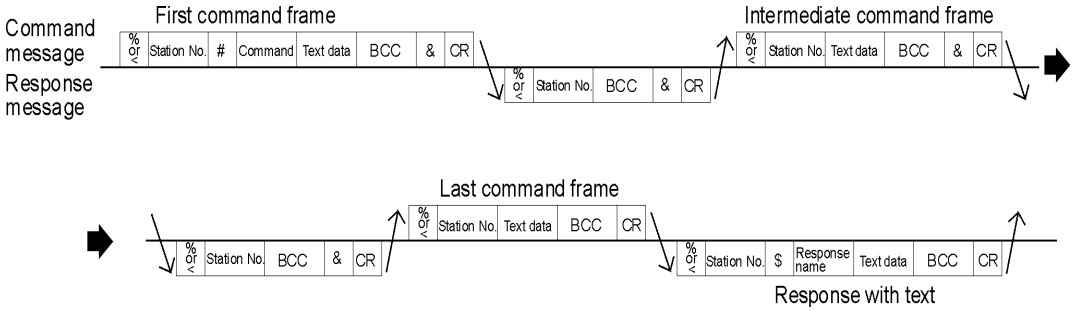
Data request message frame

When a programmable controller or a computer receives a message that contains an “&“ delimiter, they must send a data request message that contains the station number, the BCC and an “&“. For details, refer to the next sections.

Data flow using multiple frame

Using multiple frame command message

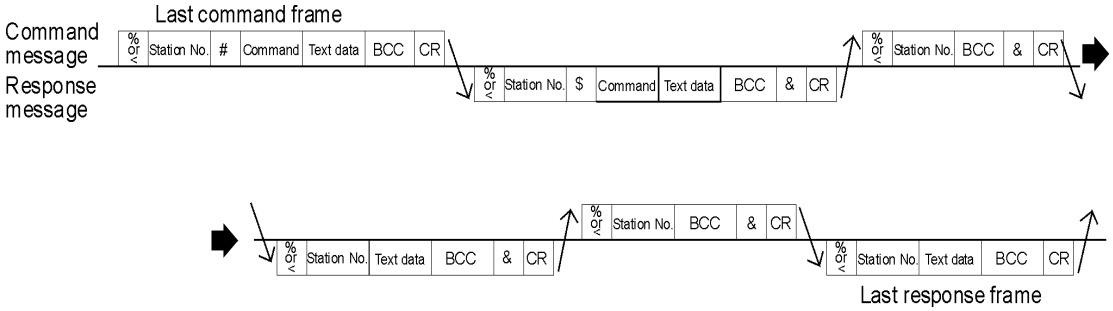
After each frame of the command message that contains an “&” delimiter is received, the programmable controller responds with its station number and the BCC. Then the programmable controller waits for the next piece of the command message.



The response message frame parentheses with text (“$“ response symbol/response code/text data) are not sent back to the computer until all of the command message frames with text have been sent to the programmable controller.

Using a multiple frame response message

After receiving each frame of a response message that contains an &[&] delimiter, the computer responds with the station number and the BCC. Then the computer waits for the next piece of the response message.



- **Command message frames without text (station number/BCC) are sent back to the programmable controller until all the response message frames have been received by computer.**
- **When a message is divided into multiple frames, the next frame can not be sent without first receiving a confirmation that the most recent frame was received correctly.**
- **As a message in multiple frames can not be interrupted without the abort (AB) command, it is recommended that the number of frames in one message should be limited to as small a number as possible.**

List of Main Symbols

Symbol name	Character	ASCII code (Hexadecimal)	Description
Header	%	H25	Indicates the start of a message frame.
Expansion header	<	H3C	Indicates the start of a message frame. This is available for high-level link units, such as the ET-LAN unit and MEWNET-H link unit.
Command symbol	#	H23	Indicates a command message.
Response symbol	$	H24	Indicates a normal response message frame.
Error symbol	!	H21	Indicates a response message when an error has occurred.
Terminator	CR	H0D	Indicates the end of a message frame.
Delimiter	&	H26	Indicates more to follow when a message is sent as several frames.

1.2.3 List of MEWTOCOL–COM Memory Area Codes

The memory area codes are specified as 1 or 2 characters (capital letters).
These codes are a little bit different from the names used in the programmable controller for the memory area in numbering or their specifications. Be sure to check the coincidence of each code before use.

Memory area name	Memory area code (ASCII HEX code)	Description	Applicable command
External input relay	X (H58)	This code is used when the external input relays in the memory area are specified. In the “RC” command, this code is used also to specify the word units address of the memory.	RC MC
	WX (H57) (H58)	This code is used only when the word external input relays are specified in the “MD” command. In other commands, the code “X” is used to specify also word external input relays.	MD
External output relay	Y (H59)	This code is used when the external output relays in the memory area are specified. In the “RC”, “WC”, and “SC” commands, this code is used also to specify the word units address of the memory.	RC WC SC MC
	WY (H57) (H59)	This code is used only when the word external output relays are specified in the “MD” command. In other commands, the code “Y” is used to specify also word external output relays.	MD
Internal relay	R (H52)	This code is used when the internal relays in the memory area are specified. In the “RC, “WC” and “SC” commands, this code is used also to specify the word units address of the memory.	RC WC SC MC
	WR (H57) (H52)	This code is used only when the word internal relays are specified in the “MD” command. In other commands, the code “R” is used to specify also word internal relays.	MD
Link relay	L (H4C)	This code is used when the link relays in the memory area are specified. In the “RC, “WC” and “SC” commands, this code is used also to specify the word units address of the memory.	RC WC SC MC
	WL (H57) (H4C)	This code is used only when the word link relays are specified in the “MD” command. In other commands, the code “L” is used to specify also word internal relays.	MD
Data register	D (H44)	This code is used when the data registers in the memory area are specified. Its addresses are expressed as a decimal number	RS, WD, SD, MD
File register	F (H46)	This code is used when the file registers in the memory area are specified. Its addresses are expressed as a decimal number.	RS, WD, SD, MD
Link data register	L (H4C)	This code is used when the link data registers in the memory area are specified. Its addresses are expressed as a decimal number.	RS, WD, SD, MD

Memory area name	Memory area code (ASCII HEX code)	Description	Applicable command
Index register (IX/IY)	IX (H49) (H58)	This code is used when the index register IX in the memory area is specified. (As each PLC has only one IX index register, the imaginary address of “0000” or “00000” is specified in the command message.)	RD WD MD
	IY (H49) (H59)	This code is used when the index register IY in the memory area is specified. (As each PLC has only one IY index register, the imaginary address of “0000” or “00000” is specified in the command message.)	RD WD MD
	ID (H49) (H44)	This code is used when both X type and Y type index registers in the memory area are specified. (As each PLC has only one set of index registers (IX and IY), the imaginary address of “0000” or “00000” is specified in the command message.)	RD WD
Timer/ Counter contact	T (H54)	This code is used when the timer contacts in the memory area are specified. (As they are expressed in decimal number, be sure to check its contact address when the address should be specified in word units.) Even if you specify “T” in the counter contact area address number, no error will occur.	RC MC
	C (H43)	This code is used when the counter contacts in the memory area are specified. (As they are expressed in decimal number, be sure to check its contact address when the address should be specified in word units.) Even if you specify “C” in the counter contact area address number, no error will occur.	RC MC
Timer/ Counter set value area	S (H53)	This code is used when the timer and/or counter set value areas in the memory area are specified in the “MD” command.	MD
Timer/ Counter elapsed value area	K (H4B)	This code is used when the timer and/or counter elapsed value areas in the memory area are specified in the “MD” command.	MD

1.2.4 List of MEWTOCOL–COM Command/Response Codes

The command/response codes are specified using two capital letters. The same code as the one sent in the command message will be returned to indicate that the programmable controller is responding to the command message.

Name	Command code (ASCII HEX code)	Description	Memory area code in MEWTOCOL–COM
Read contact	RC (H53) (H43)	Read the contents stored in external input and output relays, internal relays, link relays and timer/ counter contacts. Read–out data can be selected in single–bit units, an optional number of bits (up to 8) or word units.	External input relay: X External output relay: Y Internal relay: R Link relay: L Timer contact: T Counter contact: C
Write contact	WC (H57) (H43)	Writes data into external output, internal and link relays. Written data can be selected in single–bit units, an optional number of bits (up to 8) or word units.	External output relay: Y Internal relay: R Link relay: L
Set contact	SC (H53) (H43)	Sets a data pattern in external output, internal and link relays in word units.	External output relay: Y Internal relay: R Link relay: L
Read registers	RD (H52) (H44)	Reads the contents stored in data, link data, file and index registers	Data register: D Link data register: L File register: F Index register IX: IX Index register IX: IX Index register IX&IY: ID
Write registers	WD (H57) (H44)	Writes data into data, link data, file and index registers.	Data register: D Link data register: L File register: F Index register IX: IX Index register IX: IX Index register IX&IY: ID
Set registers	SD (H53) (H44)	Sets a data pattern in data, link data and file registers.	Data register: D Link data register: L File register: F
Read SV of a timer/counter	RS (H52) (H53)	Reads the set value are SV for the timer/counter.	No need to specify the memory area code.
Write a value of a timer/ counter to SV	WS (H57) (H53)	Writes data into the set value area SV for the timer/counter.	No need to specify the memory area code.
Read EV of a timer/counter	RK (H52) (H4B)	Reads the elapsed value area EV for the timer/counter.	No need to specify the memory area code.
Write a value of a timer/ counter to EV	WK (H57) (H4B)	Writes data into the elapsed value area DV for the timer/counter.	No need to specify the memory area code.
Specify contacts monitored	MC (H4E) (H43)	Registers or resets the addresses of external input and output relays, internal relays, link relays and timer/counter contacts, which will be monitored by the “MG” command.	External input relay: X External output relay: Y Internal relay: R Link relay: L Timer contact: T Counter contact: C

Name	Command code (ASCII HEX code)	Description	Memory area code in MEWTOCOL–COM
Abort a series of response messages	AB (H41) (H42)	Aborts a series of messages sent in multiple frames.	No need to specify the memory area code.
Specify registers monitored	MD (H4E) (H44)	Registers or resets the addresses of data, link data, file and index registers, word external input and output relays, word internal relays, and timer/counter set and elapsed value areas, which will be monitored by the “MG” command.	Data register: D Link data register: L File register: F Index register IX: IX Index register IY: IY Word external input relay: WR Word external output relay: WY Word internal relay: WR Timer/counter set value area: S Timer/counter elapsed value area: K
Monitoring start	MG (H4E) (H47)	Monitors the points specified in the “MC” and “MD” commands.	No need to specify the memory area code.
Read system registers	RR (H52) (H52)	Reads parameters stored in system registers of the PLC.	No need to specify the memory area code.
Write a value to system register	WR (H57) (H52)	Writes parameters into system registers of the PLCs.	No need to specify the memory area code.
Read the status of the PLC	RT (H52) (H54)	Reads the status of the PLC such as PLC type and program capacity.	No need to specify the memory area code.
Extended read status of the PLC	EXRT (H45) (H58) (H52) (H54)	Same as RT but can read additional statuses: e.g. the link unit, number of programs, acceptable port numbers, etc.	No need to specify the memory area code.
Read a program block of the PLC for backup	RP (H52) (H52)	Reads a program block stored in the PLC. The program read must be used only for backup purposes.	No need to specify the memory area code.
Write a program block read by WP command	WP (H57) (H52)	Write the program block read out by the “RP” command.	No need to specify the memory area code.
Change the mode of the PLC	RM (H52) (H4D)	Remotely controls the mode of the PLC (PROG. or RUN).	No need to specify the memory area code.
Abort a series of response messages	AB (H41) (H42)	Aborts a series of messages sent in multiple frames.	No need to specify the memory area code.

1.3 MEWTOCOL–COM Commands and Responses

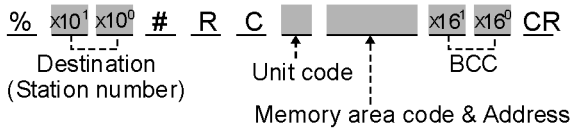
Descriptions for each MEWTOCOL–COM command and response messages are explained on the pages shown below.

RC Read contact (single point/plural points/word units)

Outline Reads the contents stored in external input relays, external output relays, internal relays, link relays and timer or counter contacts.

Basic message format

Command message



Response message

Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
x	–	x	–	x	–	x	–	–	–	–	–	–	–	x	x	–	–

x: available
–: not available

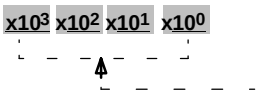
☞
- **The codes “X”, “Y”, “R“ and “L” are also used to read data in one word units (1 word = 16 bits).**
- **For details on memory area codes, see page 12.**

Unit codes

A computer can read a single bit of data, an optional number of bits (1 to 8 bits) or in units of words (1 word = 16 bits).
In order to set the data size for the “RC“ command, use the following unit codes.

Unit code	Description	Address numbering system	
		X, Y, R, L	T, C
S	Specify "S" to read a single bit of data.	Relay bit numbering (4-digit)	Decimal numbering (4-digit)
P	Specify "P" to read an optional number of bits (1 to 8 bits).	Relay bit numbering (4-digit)	Decimal numbering (4-digit)
C	Specify "C" to read data in units of words (1 word = 16 bits).	Word numbering (4-digit)	See note

☞ **You can read timer/counter contacts in units of words. However, since timer/counter contacts are not normally treated in units of words, it is recommended that you do not read them in units of words to avoid any numbering system confusion. When you specify the timer/counter contacts in this command, refer to the following:**

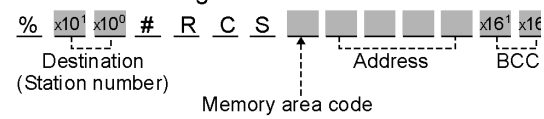


Setting	T/C contact number
0000	0 to 15
0001	16 to 31
⋮	⋮
0127	2032 to 2047

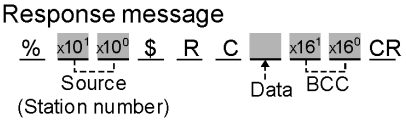
Description Reads the contents stored in external input relays, external output relays, internal relays, link relays and timer or counter contacts.
A computer can read a single bit of data, or an optional number of bits (1 to 8 bits) in one command message.
It can also read data in units of words (1 word = 16 bits).
Refer to following pages for detailed explanations.

■ When the unit code “S” is specified. [When you want to read a single bit of data.]

Command message

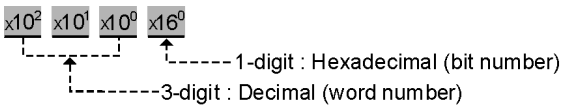


Response message

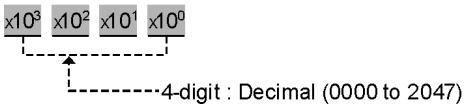


Explanation

1) Memory area code: Specify the memory area code of the programmable controller to be read from, referring to the codes given on page 17, “Memory area codes”.
2) Address: The address for X (external input relay), Y (external output relay), R (internal relay) and L (link relay) is expressed using a relay bit numbering system as follows:



The contact address for T (timer contact) and C (counter contact) is expressed using a decimal numbering system as follows:



When you read a timer contact, specify the contact with “T“ and when you read a counter contact, specify the contact with “C”. However, even if you specify “C” but then use a timer contact address or if you specify “T” and then a counter contact address, the computer will read the contents of the address specified in the command message.

3) Data: Contact data is specified as :
0 : OFF state
1 : ON state

Program example

Command message

% 0 1 # R C S X 0 0 0 A ✻ ✻ CR

Response message

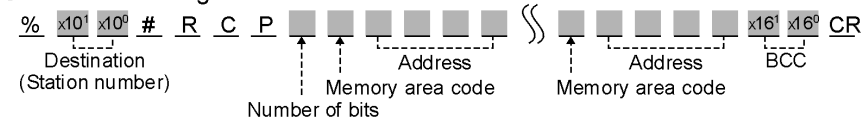
% 0 1 $ R C 0 2 1 C R

The contents of XA are read by the programmable controller whose station number is 01.

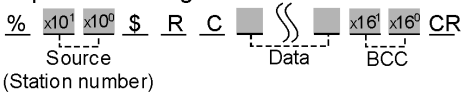
Command message
Destination: 01 station
Point: XA
Response message
Source: 01 station
Data: XA = 0 (OFF)

■ When the unit code “P” is specified. [To read one or more bits of data (1 to 8 bits).]

Command message



Response message



Explanation

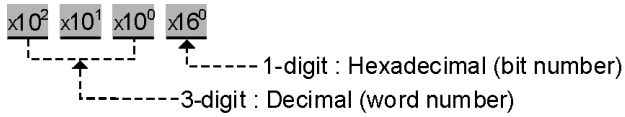
1) Number of bits: When you specify “P“ in the unit code, you must specify how many bits to read.
Specify a number in the range of 1 to 8.

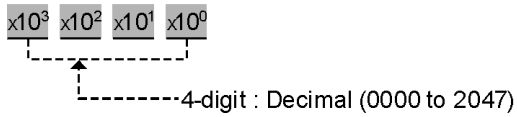
☞

- **You must specify a separate memory area code and address for each bit of data you want to access. Thus, you will have to give from 1 to 8 memory area codes and addresses depending on the number of bits you specified.**
- **A single bit can also be accessed with the unit code “S”.**

2) Memory area code: Specify the memory area code for the programmable controller to be read from, referring to the codes given on page 17, “Memory area codes“.

3) Address: The address for X (external input relay), Y (external output relay), R (internal relay) and L (link relay) is expressed using a relay bit numbering system as follows:





When you read a timer contact, specify the contact with “T“ and when you read a counter contact, specify the contact with “C”. However, even if you specify “C“ but then use a timer contact address or if you specify “T“ and then use a counter contact address, the computer will read the contents of the address specified in the command message.

4) Data: Contact data is specified as:
0: OFF state
1: ON state

Program example

Command message

% 0 1 # R C P 3 X 0 0 0 A Y 0 0 1 F T 0 0 0 5 ❊ ❊ CR

Response message

% 0 1 $ R C 1 0 0 2 0 CR

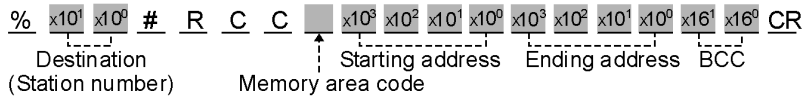
The contents of XA, Y1F and T5 will be read from the programmable controller whose station number is 01.

Command message
Destination: 01 station
Number of bits: 3 bits (XA, Y1F, T5)

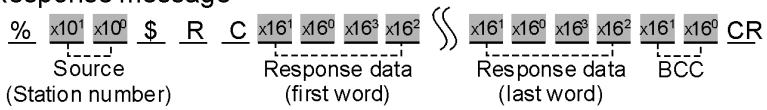
Response message
Source: 01 station
Data: XA = 1 (ON), Y1F = 0 (OFF), T5 = 0 (OFF)

■ When unit code “C” is specified. [To read bit data in units of words (1 word = 16 bits).]

Command message



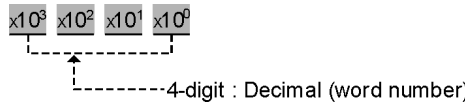
Response message



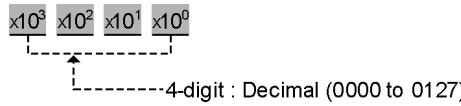
Explanation 1) Memory area code: Specify the memory area code for the programmable controller to read from, ad from, referring to the codes given on page 17, “Memory area codes”

☞ **The memory area codes used in this command do not have same name as those that are used in programming the programmable controller.**

2) Starting address
& Ending address: The starting and ending word addresses for X (external input relay), Y (external output relay), R (internal relay) and L (link relay) are expressed using a word numbering system as follows:



You can read timer/counter contacts in units of words. However, since timer/counter contacts are not normally treated in unit of words, it is recommended that you do not read them in units of words to avoid any numbering system confusion.
When you specify the timer/counter contacts in this command, refer to the following.



Setting	T/C contact number
0000	0 to 15
0001	16 to 31
⋮	⋮
0127	2032 to 2047

When you read a timer contact, specify the contact with “T“ and when you read a counter contact, specify the contact with “C”. However, even if you specify “C“ but then use a timer contact address or if you specify “T“ and then a counter contact address, the computer will read the contents of the address specified in the command message.

☞ **The ending address must be equal to or larger than the starting address.**

3) Response data: 4 characters are returned for each word relay address included in the command in the form shown below.
Data will be returned starting with the data stored in the starting word address specified in the command message.

Data received in the computer: 4-digit hexadecimal A 1 F C

Data in the programmable controller (word data):

Hexadecimal	F	C	A	1
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 1 1	1 1 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

☞ ● **The number of words of data that are returned is equal to the ending address minus the starting address plus one.**

- **The programmable controller stores words in low–byte, high–byte order. Thus, data returned by the programmable controller are in that order.**

Program example

Command message

% 0 1 # R C C X 0 0 0 0 0 0 0 2 ✻ ✻ CR

Response message

% 0 1 $ R C 6 3 0 0 4 4 3 3 0 A 0 0 6 2 CR

The contents of external input relays [WX0 to WX2 (X0 to X2F)] will be read from the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting address: WX0
Ending address: WX2
Read out range: WX0 to WX2 (X0 to X2F)

Response message:
Source: 01 station
Data received in
response message: H6300, H4433, H0A00
Actual data: WX0 = H0063, WX1 = H3344, WX2 = H000A

Data received in the computer — 4-digit hexadecimal: 6 3 | 0 0

Actual data in the programmable controller WX0

Hexadecimal	0	0	6	3
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 1 1 0	0 0 1 1

XF X0

Data received in the computer — 4-digit hexadecimal: 4 4 | 3 3

Actual data in the programmable controller WX1

Hexadecimal	3	3	4	4
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 1 1	0 0 1 1	0 1 0 0	0 1 0 0

X1F X10

Data received in the computer — 4-digit hexadecimal: 0 A | 0 0

Actual data in the programmable controller WX2

Hexadecimal	0	0	0	A
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	1 0 1 0

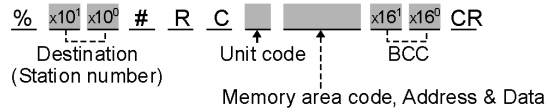
X2F X20

WC Write contact (single point/plural points/word units)

Outline Writes data into external output relays, internal relays and link relays.

Basic message format

Command message



Response message

Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
–	–	x	–	x	–	x	–	–	–	–	–	–	–	–	–	–	–

x: available
–: not available

- **The memory area code “X” (external input relay) can be speci ed only for the FP3.**
- **The codes ”X” (only for the FP3), ”Y”, ”R” and ”L” also are used to write data in units of words (1 word = 16 bits).**
- **For details on memory area codes, see page 12.**

Unit codes

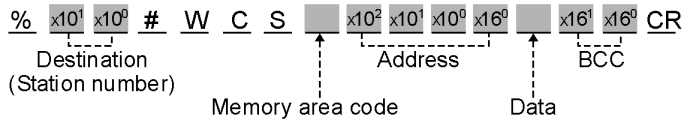
A computer can write a single bit of data, an optional number of bits (1 to 8 bits) or in units of words (1 word = 16 bits). In order to set the data size for “WC” command, use the following unit codes.

Unit code	Description	Address numbering system
		X, Y, R, L
S	Specify “S” to write a single bit of data.	Relay bit numbering (4-digit)
P	Specify “P” to write an optional number of bits (1 to 8 bits).	Relay bit numbering (4-digit)
C	Specify “C” to write data in units of words (1 word = 16 bits).	Word numbering (4-digit)

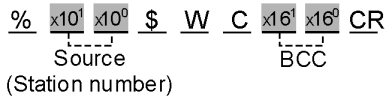
Description Writes data into external output relays, internal relays and link relays.
A computer can write a single bit of data, or an optional number of bits (1 to 8 bits) in one command message.
It can also write data in units of words (1 word = 16 bits).
Refer to the following pages for detailed explanations.

■ When the unit code “S” is specified. [When you want to write a single bit of data.]

Command message



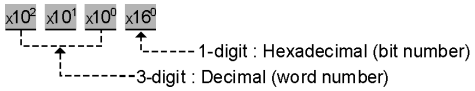
Response message



Explanation

1) Memory area code:
Specify the memory area code for the programmable controller to be written into, referring to the codes given on page 2 – 23, “Memory area codes”.

2) Address:
The address for Y (external output relay), R (internal relay) and L (link relay) is expressed using a relay bit numbering system as follows:



3) Data: Contact data is specified as :
0: OFF state
1: ON state

Program example

Command message

% 0 1 # W C S Y 0 0 0 A 1 ✻ ✻ CR

Response message

% 0 1 $ W C 1 4 CR

The data (1 = ON) is written to external output relay (YA) of the programmable controller whose station number is 01.

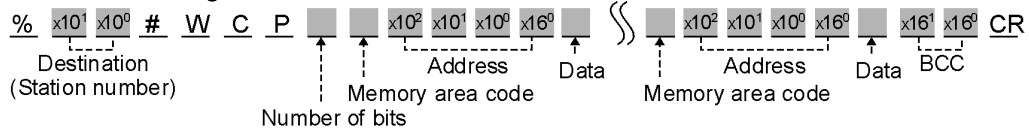
Command message
Destination: 01 station
Point: YA

Data written: 1 (ON)

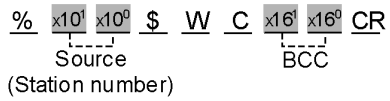
Response message
Source: 01 station

■ **When the unit code ”P” is specified. [To write one or more bits of data (1 to 8 bits).]**

Command message



Response message



Explanation

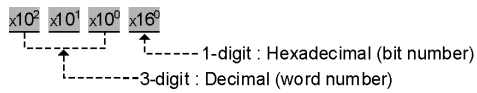
1) Number of bits:
When you specify ”P” in the unit code, you must specify how many bits to write.
Specify a number in the range of 1 to 8.

☞

- **You must specify a separate memory area code, address and data for each bit of data you want to access. Thus, you will have to give from 1 to 8 memory area codes, addresses and data depending on the number of bits you specified.**
- **A single bit can also be accessed with the unit code ”S”.**

2) Memory area code:
Specify the memory area code for the programmable controller to be written into, referring to the codes given on page 2 – 23, “Memory area codes”.

3) Address:
The address for Y (external output relay), R (internal relay) and L (link relay) is expressed using a relay bit numbering system as follows:



4) Data: Contact data is specified as :
0: OFF state
1: ON state

Program example

Command message

% 0 1 # W C P 3 Y 0 0 0 A 0 Y 0 0 1 F 1 R 0 0 0 5 0 ✲ ✲ CR

Response message

% 0 1 $ W C 1 4 CR

The data (0 = OFF, 1 = ON, 0 = OFF) are written to the external relays (YA and Y1F) and the internal relay (R5) of the programmable controller.

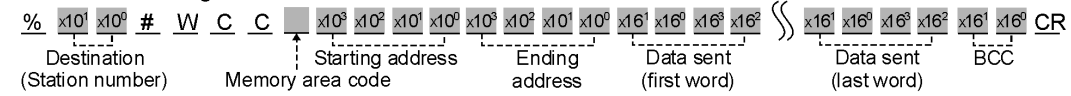
Command message:
Destination: 01 station
Number of bits: 3 bits (YA, Y1F, R5)

Data written: YA = 0 (OFF), Y1F = 1 (ON), R5 = 0 (OFF)

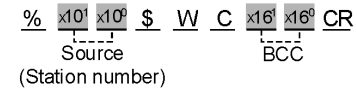
Response message:
Source: 01 station

■ **When the unit code “C” is specified. [To write data in units of words (1 word = 16 bits).]**

Command message



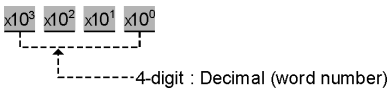
Response message



Explanation 1) Memory area code:
Specify the memory area code of the programmable controller to be written into, referring to the codes given on page 2 – 23, “Memory area codes”.

☞ **The memory area codes used in this command do not have same name as those that are used in programming the programmable controller.**

2) Starting address & Ending address:
The starting and ending word addresses for Y (external output relay), R (internal relay) and L (link relay) are expressed using a word numbering system as follows :



☞ **The ending address must be equal to or larger than the starting address.**

3) Data sent:
4 characters are used to write one of word data in the form shown below. Data will be sent to the programmable controller in order from the starting to the ending addresses.

Data sent 4-digit hexadecimal A 1 F C

Data set in the programmable controller (word data)

Hexadecimal	F	C	A	1
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 1 1	1 1 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

☞ ● **The number of words of data that are sent is equal to the ending address minus the starting address plus one.**

- **The programmable controller stores words in low–byte, high–byte order. Thus, data sent to the programmable controller must be in that order.**

Program example

Command message

% 0 1 # W C C R 0 0 0 0 0 0 0 2 6 3 0 0 4 4 3 3 0 A 0 0 ❊ CR

Response message

% 0 1 $ W C 1 4 CR

The data (H6300, H4433, H0A00) will be written into the address block [WR0 to WR2 (R0 to R2F)].

Command message
Destination: 01 station
Starting address: WR0
Ending address: WR2
Data write block: WR0 to WR2 (R0 to R2F)
Data sent: H6300, H4433, H0A00
Data set in programmable controller: WR0=H0063, WR1=H3344, WR2=H000A

Data sent — 4-digit hexadecimal: 6 3 0 0

Data set in the programmable controller WR0

	RF			R0
Hexadecimal	0	0	6	3
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 1 1 0	0 0 1 1

Data sent — 4-digit hexadecimal: 4 4 3 3

Data set in the programmable controller WR1

	R1F			R10
Hexadecimal	3	3	4	4
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 1 1	0 0 1 1	0 1 0 0	0 1 0 0

Data sent — 4-digit hexadecimal: 0 A 0 0

Data set in the programmable controller WR2

	R2F			R20
Hexadecimal	0	0	0	A
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	1 0 1 0

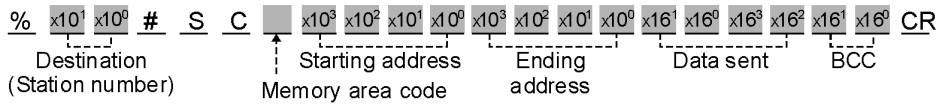
Response message
Source: 01 station

SC Set contact (word units)

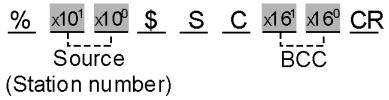
Outline Sets a data pattern (in word units) in external output relays, internal relays or link relays.

Basic message format

Command message



Response message



Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
–	–	x	–	x	–	x	–	–	–	–	–	–	–	–	–	–	–

x: available
–: not available

- **The codes “Y”, “R” and “L” are also used to write data patterns in units of words (1 word = 16 bits).**
- **For details on memory area codes, see page 12.**

Description Sets the data pattern in external input relays (only for the FP3), external output relays, internal relays or link relays.
The data pattern is written in units of words (one word = 16 bits).

Memory area codes

Specify the memory area code for the programmable controller to be written into, referring to the codes given above in “Memory area codes”.

The memory area codes used in this command do not have same name as those that are used in programming the programmable controller.

Starting address/Ending address

The starting and ending word addresses for X [(external input relay) only for the FP3], Y (external output relay), R (internal relay) and L (link relay) are expressed using a word numbering system as follows:

☞ **The ending address must be equal to or larger than the starting address.**

Data set

4 characters are used to set a data pattern in the form shown below.
Data will be sent to the programmable controller in order from the starting to the ending addresses.

Data sent

4-digit hexadecimal A 1 F C

Data set in the programmable controller (word data)

Hexadecimal	F	C	A	1
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 1 1	1 1 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

☞ **The programmable controller stores words in low–byte, high–byte order. Thus, data sent to the programmable controller must be in that order.**

Program example

Command message

% 0 1 # S C Y 0 0 0 0 0 0 3 0 A B C D ✻ ✻ CR

Response message

% 0 1 $ S C 1 0 CR

The data (HABCD) will be written to the address block (WY0000 to WY0030). The command and response messages are recognized as:

Command message
Destination: 01 station
Starting address: WY0
Ending address: WY30
Data set block: WY0 to WY30 (Y0 to Y30F)
Data sent: HABCD
Data set in programmable controller: HCDAB

Data sent

4-digit hexadecimal A B C D

Data pattern

Hexadecimal	C	D	A	B
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1

Data set in the programmable controller

Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Address				
WY0000	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0001	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0002	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0003	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0004	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
WY0026	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0027	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0028	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0029	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
WY0030	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1

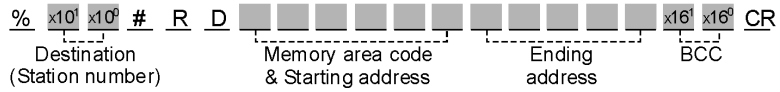
Response message
Source: 01 station

RD Read registers

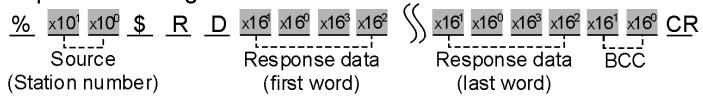
Outline Reads the contents stored in data registers, link data registers, file registers or index registers.

Basic message format

Command message



Response message



Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	A	A	A	A	A	A	N/A	N/A	N/A	N/A

A : Available
N/A: Not Available

☞
- **The memory area code ”ID” is used when both the “X” and the “Y” index registers.**
- **For details on memory area codes, see page 12.**

Description Reads the contents stored in data registers, link data registers, file registers, or index registers (IX or/and IY).
Since the memory area of each register is configured as 16 bits (one word), data from a register will be returned in the form of 4–digit hexadecimal.

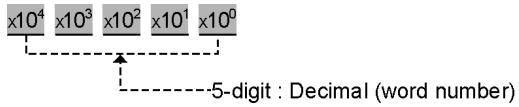
Memory area code

Specify the memory area code for the programmable controller to be read from, referring to the codes given above in “Memory area codes”.

☞ **The memory area codes used in this command do not have the same name as those that are used in programming the programmable controller.**

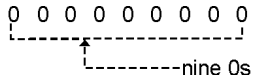
Starting address/Ending address

The starting and ending addresses for ”D” (data registers), “L” (link data registers) and “F” (file registers) are expressed using a word numbering system as follows:



The ending address must be equal to or larger than the starting address.

The “IX” (index register IX), “IY” (index register IY) and “ID” (index registers IX and IY) are specified with nine 0s instead of specifying the starting and ending addresses, as the index registers do not have their own numbers with them.



Response Data

4 characters are returned for each register address included in the command as shown below.
Data will be returned from the programmable controller starting with the starting to the ending address.

Data received in a computer

4-digit hexadecimal A 1 F C

Data set in the programmable controller (register data)

Hexadecimal	F	C	A	1
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 1 1	1 1 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

- **The number of words of data that are returned is equal to the ending address minus the starting address plus one.**
- **The programmable controller stores words in low–byte, high–byte order. Thus, data returned by the programmable controller are in that order.**

Program example

Command message

% 0 1 # R D D 0 1 1 0 5 0 1 1 0 7 ✲ ✲ CR

Response message

% 0 1 $ R D 6 3 0 0 4 4 3 3 0 A 0 0 6 2 CR

The contents of data registers (DT1105 to DT1107) will be read by the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting address: DT1105
Ending address: DT1107
Read out block: DT1105 to DT1107

Response message:
Source: 01 station
Data received: H6300, H4433, H0A00
Data set in programmable controller: DT1105 = H0063, DT1106 = H3344, DT1107 = H000A

• DT1105

Data received in the computer — 4-digit hexadecimal: 6 3 | 0 0

Actual data in the programmable controller DT1105

Hexadecimal	0	0	6	3
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 1 1 0	0 0 1 1

• DT1106

Data received in the computer — 4-digit hexadecimal: 4 4 | 3 3

Actual data in the programmable controller DT1106

Hexadecimal	3	3	4	4
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 1 1	0 0 1 1	0 1 0 0	0 1 0 0

• DT1107

Data received in the computer — 4-digit hexadecimal: 0 A | 0 0

Actual data in the programmable controller DT1107

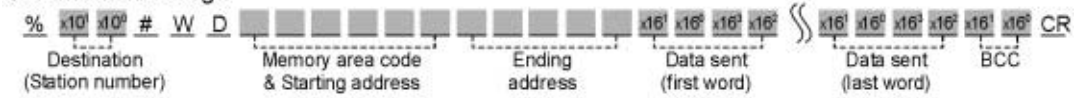
Hexadecimal	0	0	0	A
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	1 0 1 0

WD Write registers

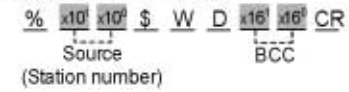
Outline Writes data into data registers, link data registers, file registers or index registers.

Basic message format

Command message



Response message



Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
–	–	–	–	–	–	–	–	x	x	x	x	x	x	–	–	–	–

x: available
–: not available

- **The memory area code ”ID” is used when both the “X” and the “Y” index registers.**
- **For details on memory area codes, see page 12.**

Description Writes data into data registers, link data registers, file registers or index registers (IX or/and IY) of the programmable controller.
Since the memory area of each register is configured as 16 bits (one word), data to a register will be written in the form of 4–digit hexadecimal.

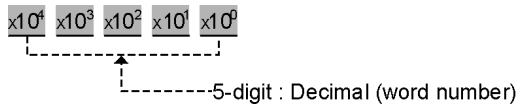
Memory area code

Specify the memory area code for the programmable controller to be written into, referring to the codes given above in “Memory area codes“.

The memory area codes used in this command do not have same name as those that are used in programming the programmable controller.

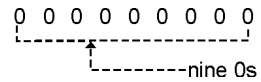
Starting address/Ending address

The starting and ending addresses for “D” (data registers), “L” (link data registers) and “F” (file registers) are expressed using a word numbering system as follows:



The ending address must be equal to or larger than the starting address.

The “IX” (index register IX), “IY“ (index register IY) and “ID” (index registers IX and IY) are specified with nine 0s instead of specifying the starting and ending addresses, as the index registers do not have their own numbers with them.



Data sent

4 characters are needed for each word of data (one word per register address) as shown below. Data will be sent to the programmable controller in order from the starting to the ending address.

Data received in a computer

4-digit hexadecimal A 1 F C

Data set in the programmable controller (register data)

Hexadecimal	F	C	A	1
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 1 1	1 1 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

- **The number of words of data that are sent is equal to the ending address minus the starting address plus one.**
- **The programmable controller stores words in low–byte, high–byte order. Thus, data sent to the programmable controller must be in that order.**
- **When the memory area code is ”ID”, two words of data (8 characters) should be sent in the order IX register data, IY register data.**

Program example

Command message

% 0 1 # W D D 0 0 0 0 1 0 0 0 0 3 0 5 0 0 0 7 1 5 0 0 0 9 ✻ ✻ CR

Response message

% 0 1 $ W D 1 3 CR

The data (H0500, H0715, H0009) will be sent to the specified registers (DT1, DT2, DT3) in the programmable controller.

Command message:
Destination: 01 station
Starting address: DT1
Ending address: DT3
Data write block: DT1 to DT3
Data sent: H0500, H0715, H0009
Data set in programmable controller: DT1 = H0005, DT2 = H1507, DT3 = H0900

Response message:
Source: 01 station

Data sent — 4-digit hexadecimal: 0 5 0 0

Data set in the programmable controller DT1

Hexadecimal	0	0	0	5
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 1

Data sent — 4-digit hexadecimal: 0 7 1 5

Data set in the programmable controller DT2

Hexadecimal	1	5	0	7
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 1	0 1 0 1	0 0 0 0	0 1 1 1

Data sent — 4-digit hexadecimal: 0 0 0 9

Data set in the programmable controller DT3

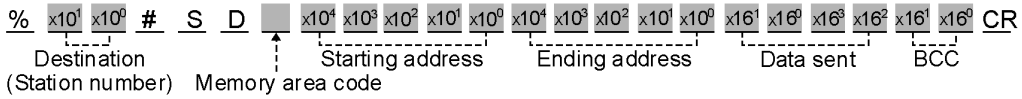
Hexadecimal	0	9	0	0
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	1 0 0 1	0 0 0 0	0 0 0 0

SD Set registers

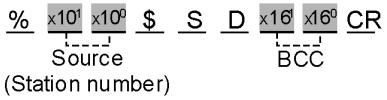
Outline Sets a data pattern in data registers, link data registers or file registers.

Basic message format

Command message



Response message



Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
–	–	–	–	–	–	–	–	x	x	x	–	–	–	–	–	–	–

x: available
–: not available

☞ **For details on memory area codes, see page 12.**

Description Sets a data pattern in data registers, link data registers or file registers in the programmable controller.
Since the memory area of each register is configured as 16 bits (one word), data to a register will be written in the form of 4–digit hexadecimal.

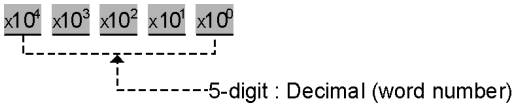
Memory area code

Specify the memory area code for the programmable controller to be written into, referring to the codes given above in “Memory area codes”.

☞ **The memory area codes used in this command do not have same name as those that are used in programming the programmable controller.**

Starting address/Ending address

The starting and ending addresses for “D” (data registers), “L” (link data registers) and “F“ (file registers) are expressed using a word numbering system as follows :



The ending address must be equal to or larger than the starting address.

Data sent

4 characters are needed for each word of data (one word per register address) as shown below.
Data will be sent to the programmable controller in order from the starting to the ending addresses.

Data sent

4-digit hexadecimal A 1 F C

Set pattern set in the programmable controller (set pattern)

Hexadecimal	F	C	A	1
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 1 1	1 1 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

The programmable controller stores words in low–byte, high–byte order. Thus, data sent to the programmable controller must be in that order.

Program example

Command message

% 0 1 # S D L 0 0 0 0 0 0 0 0 3 0 A B C D ❊ ❊ CR

Response message

% 0 1 $ S D 1 6 CR

The data [ABCD (H)] will fill the address block (WY0000 to WY0030).

Command message:

Destination: 01 station
Starting address: LD0
Ending address: LD30
Data set block: LD0 to LD30
Data sent: HABCD
Data set in programmable controller: HCDAB

Data sent 4-digit hexadecimal A B C D

Data pattern

Hexadecimal	C	D	A	B
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1

Data set in the programmable controller

Bit position		15 · · 12	11 · · 8	7 · · 4	3 · · 0
Address	LD0	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD1	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD2	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD3	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD4	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	.	.	.	.	.
	.	.	.	.	.
	.	.	.	.	.
	LD26	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD27	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD28	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD29	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1
	LD30	1 1 0 0	1 1 0 1	1 0 1 0	1 0 1 1

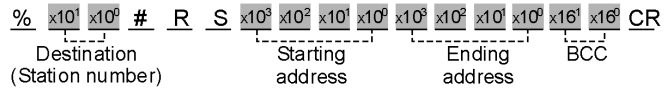
Response message
Source: 01 station

RS Read the set value from a timer/counter

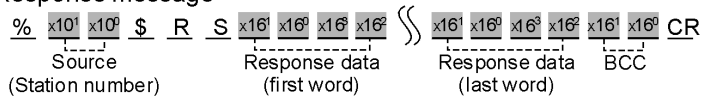
Outline Reads the timer/counter set value stored in the set value area.

Basic message format

Command message



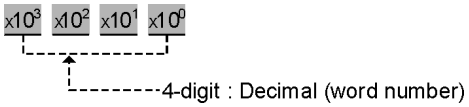
Response message



Description Reads the timer/counter set value stored in the set value area.
Since this command is dedicated to reading the timer/counter set value from the programmable controller, a memory area code is not required.

■ Starting address/Ending address

The starting and ending addresses for timer/counter set value are expressed using a word numbering system as follows:



☞ **The ending address must be equal to or larger than the starting address.**

■ Response data

4 characters are needed for each word of data (one word per “SV“ address) as shown below. Data will be read from the programmable controller in order from the starting to the ending addresses.

Data received 4-digit hexadecimal A 1 0 0

Data stored in the programmable controller

Hexadecimal	0	0	A	1
Decimal (K)	161			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

☞ **The programmable controller stores words in low–byte, high–byte order. Thus, data returned by the programmable controller are in that order.**

Program example

Command message

% 0 1 # R S 0 0 0 0 0 0 0 2 ❊ ❊ CR

Response message

% 0 1 $ R S 0 5 0 0 1 4 0 0 2 8 0 0 0 B CR

The contents of timer/counter set value area (SV0, SV1, SV2) will be returned by the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting address: SV0
Ending address: SV2
Read out block: SV0 to SV2

Response message
Source: 01 station
Data received: H0500, H1400, H2800
Data set in
programmable controller: SV0 = H0005 (K5), SV1 = H0014 (K20), SV2 = H0028 (K40)

Data received in the computer

4-digit hexadecimal 0 5 0 0

Actual data in the programmable controller "SV0"

Hexadecimal	0	0	0	5
Decimal (K)	5			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 1

Data received in the computer

4-digit hexadecimal 1 4 0 0

Actual data in the programmable controller "SV1"

Hexadecimal	0	0	1	4
Decimal (K)	20			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 1	0 1 0 0

Data received in the computer

4-digit hexadecimal 2 8 0 0

Actual data in the programmable controller "SV2"

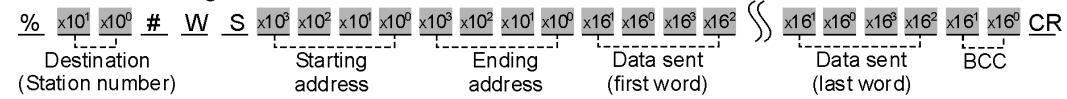
Hexadecimal	0	0	2	8
Decimal (K)	40			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 1 0	1 0 0 0

WS Write a data for a timer/counter set value area

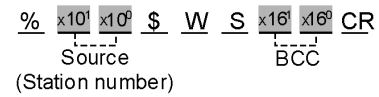
Outline Writes data into the timer/counter set value area in the programmable controller.

Basic message format

Command message



Response message



Description Writes the data into the specified timer/counter set value area.
Since this command is dedicated to writing the timer/counter set value into a set value area of the programmable controller, a memory area code is not required.

■ Starting address/Ending address

The starting and ending addresses for timer/counter set value are expressed using a word numbering system as follows:

☞ **The ending address must be equal to or larger than the starting address.**

■ Data sent

4 characters are needed for each word data (one word per “SV” address) as shown below.
Data will be sent to the programmable controller in order from the starting to the ending addresses.

Data sent 4-digit hexadecimal A 1 0 0

Data set in the programmable controller

Hexadecimal	0	0	A	1
Decimal (K)	161			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

☞ **The programmable controller stores words in low–byte, high–byte order. Thus, data sent to the programmable controller must be in that order.**

Program example

Command message

% 0 1 # W S 0 0 0 0 0 0 0 2 0 5 0 0 1 4 0 0 2 8 0 0 ❊ ❊ CR

Response message

% 0 1 $ W S 0 4 CR

The data (H0500, H1400, H2800) will be sent to the timer/counter set value areas (SV0, SV1, SV2) of the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting address: SV0
Ending address: SV2
Read out block: SV0 to SV2
Data sent: H0500 H1400, H2800
Data set in programmable controller: SV0 = H0005 (K5), SV1 = H0014 (K20), SV2 = H0028 (K40)

Response message:
Source: 01 station

Data sent 4-digit hexadecimal 0 5 0 0

Data set in the programmable controller "SV0"

Hexadecimal	0	0	0	5
Decimal (K)	5			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 1

Data sent 4-digit hexadecimal 1 4 0 0

Data set in the programmable controller "SV1"

Hexadecimal	0	0	1	4
Decimal (K)	20			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 1	0 1 0 0

Data sent 4-digit hexadecimal 2 8 0 0

Data set in the programmable controller "SV2"

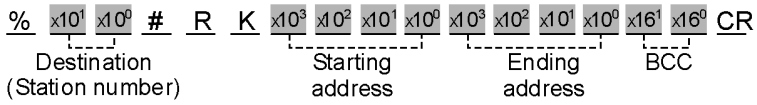
Hexadecimal	0	0	2	8
Decimal (K)	40			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 1 0	1 0 0 0

RK Read the elapsed value from a timer/counter

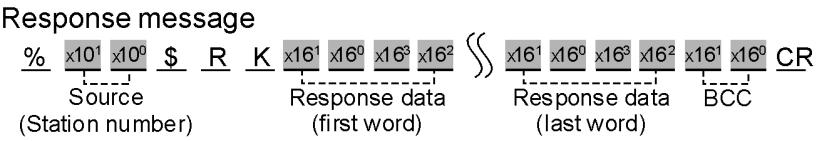
Outline Reads the timer/counter elapsed value stored in the elapsed value area.

Basic message format

Command message



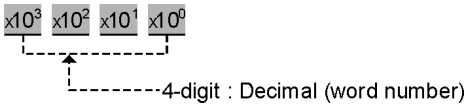
Response message



Description Reads the timer/counter elapsed value stored in the elapsed value area.
Since this command is dedicated to reading the timer/counter elapsed value from the programmable controller, a memory area code is not required.

■ Starting address/Ending address

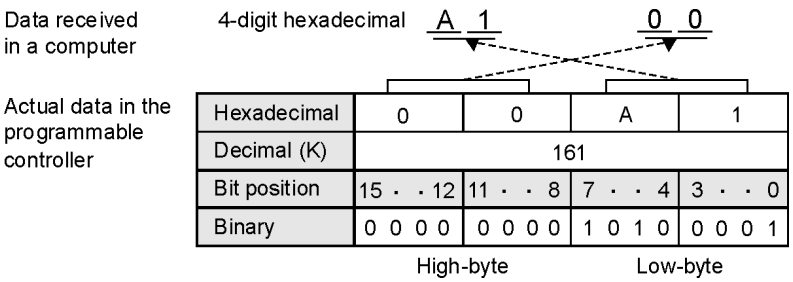
The starting and ending addresses for timer/counter elapsed value are expressed using a word numbering system as follows :



☞ **The ending address must be equal to or larger than the starting address.**

■ Response data

4 characters are needed for each word data (one word per ”EV” address) as shown below.
Data will be read from the programmable controller in order from the starting to the ending addresses.



☞ **The programmable controller stores words in low–byte, high–byte order. Thus, data returned by the programmable controller are in that order.**

Program example

Command message

% 0 1 # R K 0 0 0 0 0 0 0 2 ❊ ❊ CR

Response message

% 0 1 $ R K 0 5 0 0 1 4 0 0 2 8 0 0 1 F CR

The contents of timer/counter elapsed value area (EV0, EV1, EV2) will be returned by the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting address: EV0
Ending address: EV2
Read out block: EV0 to EV2

Response message:
Source: 01 station
Response data: H0500, H1400, H2800
Actual data in programmable controller: EV0 = H0005 (K5), EV1 = H0014 (K20), EV2 = H0028 (K40)

Data received in the computer — 4-digit hexadecimal: 0 5 | 0 0

Actual data in the programmable controller "EV0"

Hexadecimal	0	0	0	5
Decimal (K)	5			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 1

Data received in the computer — 4-digit hexadecimal: 1 4 | 0 0

Actual data in the programmable controller "EV1"

Hexadecimal	0	0	1	4
Decimal (K)	20			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 1	0 1 0 0

Data received in the computer — 4-digit hexadecimal: 2 8 | 0 0

Actual data in the programmable controller "EV2"

Hexadecimal	0	0	2	8
Decimal (K)	40			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 1 0	1 0 0 0

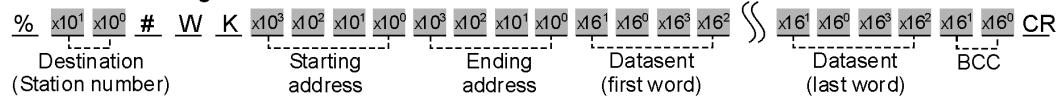
WK

Write a data for a timer/counter elapsed value area

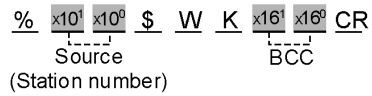
Outline Writes data into the timer/counter elapsed value area in the programmable controller.

Basic message format

Command message



Response message



Description Writes data into the specified timer/counter elapsed value area.
Since this command is dedicated to writing the timer/counter elapsed value into an elapsed value area of the programmable controller, a memory area code is not required.

■ Starting address/Ending address

The starting and ending addresses for timer/counter elapsed value are expressed using a word numbering system as follows:

☞ **The ending address must be equal to or larger than the starting address.**

■ Data sent

4 characters are needed for each word data (one word per “EV“ address) as shown below.
Data will be sent to the programmable controller in order from the starting to the ending addresses.

Data sent — 4-digit hexadecimal: A 1 0 0

Data set in the programmable controller

Hexadecimal	0	0	A	1
Decimal (K)	161			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 0 1 0	0 0 0 1
	High-byte		Low-byte	

☞ **The programmable controller stores words in low–byte, high–byte order. Thus, data sent to the programmable controller must be in that order.**

Program example

Command message

% 0 1 # W K 0 0 0 0 0 0 0 2 0 5 0 0 1 4 0 0 2 8 0 0 ❊ ❊ CR

Response message

% 0 1 $ W K 1 A CR

The data (H0500, H1400, H2800) will be sent to the timer/counter set value areas (EV0, EV1, EV2) of the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting address: EV0
Ending address: EV2
Read out block: EV0 to EV2
Data sent: H0500, H1400, H2800

Response message:
Source: 01 station
Actual data in programmable controller: EV0 = H0005 (K5),
EV1 = H0014 (K20),
EV2 = H0028 (K40)

Data sent — 4-digit hexadecimal 0 5 0 0

Data set in the programmable controller "EV0"

Hexadecimal	0	0	0	5
Decimal (K)	5			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	0 1 0 1

Data sent — 4-digit hexadecimal 1 4 0 0

Data set in the programmable controller "EV1"

Hexadecimal	0	0	1	4
Decimal (K)	20			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 1	0 1 0 0

Data sent — 4-digit hexadecimal 2 8 0 0

Data set in the programmable controller "EV2"

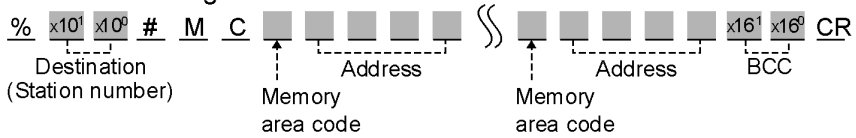
Hexadecimal	0	0	2	8
Decimal (K)	40			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 1 0	1 0 0 0

MC Specify contact addresses for monitoring Reset contact addresses that have been specified for monitoring

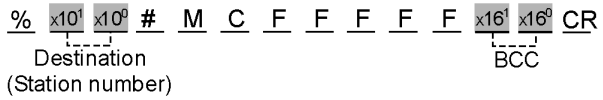
Outline Specifies the addresses of external input relays, external output relays, internal relays, link relays and timer or counter contacts.
Resets the points specified by previous “MC” commands.

Basic message format

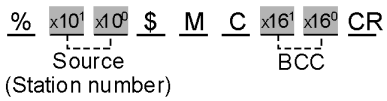
Command message



Reset



Response message



Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
x	–	x	–	x	–	x	–	–	–	–	–	–	–	x	x	–	–

x: available
–: not available

For details on memory area codes, see page 12.

Description Specifies addresses of external input relays, external output relays, internal relays, link relays and timer or counter contacts to be monitored, or it resets the points previously specified by an “MC” command.

- **A maximum of 20 contacts can be specified in one command message.**
- **A maximum of 80 points can be specified for one station.**

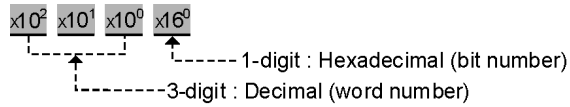
The points specified in an “MC” command are monitored by executing an “MG” command.

When specifying the contacts to be monitored

1. Memory area code:
   Specify the memory area code for the programmable controller contacts to be monitored, referring to the codes given on the previous page.

- **Memory area code: Specify the memory area code for the programmable controller contacts to be monitored, referring to the codes given in the previous page.**
- **When you want to specify plural points, you should specify each point with a combination of memory area codes and addresses.**
- **When you reset the points specified by “MC” commands, memory area codes are not required.**

2. Address setting:
The addresses for ”X” (external input relay), “Y” (external output relay), “R” (internal relay) and



The contact address for “T” (timer contact) and “C” (counter contact) are expressed using a decimal numbering system as follows:

“L” (link relay) are expressed using relay bit numbering system as follows :
When you specify a timer contact, specify the contact with “T” and when you specify a counter contact, specify the contact with “C”. However, even if you specify “C” but then use a timer contact address or if you specify “T” and then a counter contact address, the computer will read the contents of the address specified in the
command message.

Program example

Command message

% 0 1 # M C X 0 0 0 0 Y 0 0 1 A T 0 0 0 2 ✻ ✻ CR

Response message

% 0 1 $ M C 0 E CR

The points to be monitored (X0, Y1A, T2) will be specified.

■ To reset the points specified by a previous “MC” command

To reset the points specified by a previous “MC“ command, five “F”s are used in place of a memory area code and address as follows:

Program example

Command message

% 0 1 # M C F F F F F ✻ ✻ CR

Response message

% 0 1 $ M C 0 E CR

All points specified using the “MC” command will be cancelled.

MD

Specify registers, word relays or set or elapsed value area of timer/counter for monitoring
Reset the registers, word relays or timer/counter that have been specified for monitoring

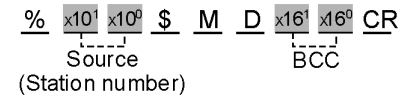
Outline Specifies the addresses of external input relays (word units), external output relays (word units), internal relays (word units), link relays (word units), data registers, link data registers, file registers, index registers (IX or IY) or timer/counter set/elapsed value which will be monitored.
Resets the points specified by previous “MD” commands.

Basic message format

Command message

Reset

Response message



Memory area codes

Relay								Register			Index register			Timer/Counter			
X	WX	Y	WY	R	WR	L	WL	D	L	F	IX	IY	ID	T	C	S	K
–	x	–	x	–	x	–	x	x	x	x	x	x	–	–	–	x	x

x: available
–: not available

☞ **For details on memory area codes, see page 12.**

Description Specifies the addresses of external input relays (word units), external output relays (word units), internal relays (word units), link relays (word units), data registers, link data registers, file registers or timer/counter set/elapsed value to be monitored, or it resets the points specified by a previous “MD” command.

☞

- **A maximum of 16 registers can be specified in one command message.**
- **A maximum of 16 points can be specified for one station.**

The points specified in an “MD” command are monitored by executing an “MG” command.

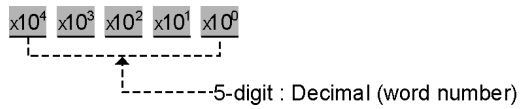
When specifying the points to be monitored

1. Memory area code:
   Specify the memory area code of the programmable controller to be monitored, referring to the codes given in the previous page.

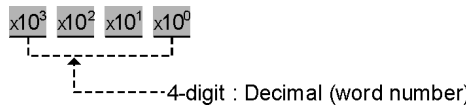
☞

- **You can specify several different memory area codes in one command message.**
- **When you want to specify plural points, you should specify each point with a combination of memory area codes and addresses.**
- **When you reset the points specified by “MD” commands, memory area codes are not required.**

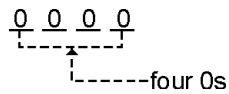
2. Address setting:
   The addresses for “D” (data registers), “L” (link data registers), “F” (file registers), “S” (timer/counter Set value) and “E” (timer/counter Elapsed value) are expressed using a 5–digit word numbering system as follows:



   The addresses for “WX” (word external input relays) and “WY” (word external output relays), “WR” (word internal relays) and “WL” (word link relays) are expressed using a 4–digit word
   numbering system as follows:



   The “IX” (X type index registers) and the “IY” (Y type index registers) are specified using four 0s instead of specifying an address since the index registers do not have their multiple addresses.



Program example

Command message

% 0 1 # M D W X 0 0 0 0 D 0 0 0 1 0 S 0 0 0 0 2 ✻ ✻ CR

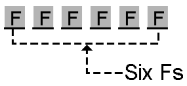
Response message

% 0 1 $ M D 0 9 CR

The points to be monitored [WX0 (X0 to XF), DT10, SV2] will be specified.

■ To reset the points specified by a previous “MD” command

To reset the points specified by a previous “MD” command, six “F”s are used in place of a memory area code and address as follows:



Program example

Command message

% 0 1 # M D F F F F F F ❊ ❊ CR

Response message

% 0 1 $ M D 0 9 CR

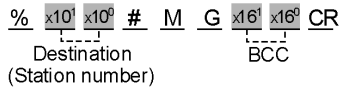
All points specified using the “MD” command will be cancelled.

MG Monitor the points specified in “MC” and “MD” commands

Outline Monitor the points specified in “MC” and “MD” commands.

Basic message format

Command message



Response message

Description The contacts and registers preset with the “MC” and “MD” commands are monitored.

■ Pace counter

The number of scans executed since last “MG” response message is returned.
If 1 to 9 scans, a one digit number (1 to 9) is returned.
If 10 scans or more, the character “A” is returned.

■ Number of characters for “MC” data

The total number of characters of data required to return information about each of the points specified in the “MC” command will be expressed as 2–digit hexadecimal number (H00 to H14).

☞ **Since a maximum of 80 points can be specified and 8 points are expressed using a 2–digit hexadecimal number, a maximum of 20 (H14) characters will be used to return this information.**

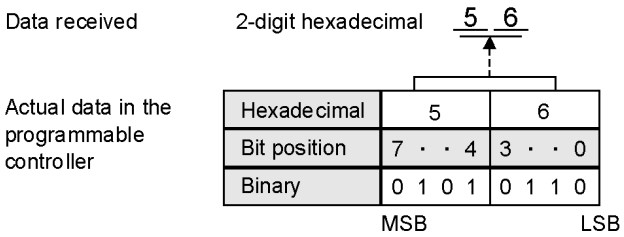
■ ”MC” data

8 bits of data will be returned as a 2–digit hexadecimal number using 2 characters as shown below.

Response data (Hex)	Binary	Specification
Lower digit	0th LSB	Status of the 1st set bit
	1st	Status of the 2nd set bit
	2nd	Status of the 3rd set bit
	3rd	Status of the 4th set bit
Upper digit	4th	Status of the 5th set bit
	5th	Status of the 6th set bit
	6th	Status of the 7th set bit
	7th MSB	Status of the 8th set bit

0 = OFF
1 = ON

Example: Response data: “56”



■ Number of characters for “MD” data

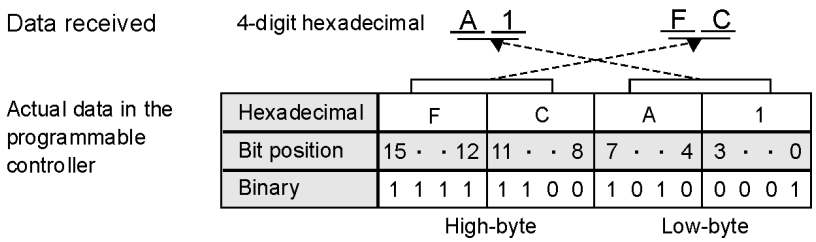
The total number of characters of data required to return information about each of the points specified in the “MD” command will be expressed as a 2–digit hexadecimal number (H00 to H40).

☞ **Since a maximum of 16 points can be specified and each point is expressed using a 4–digit hexadecimal number, a maximum of 64 (H40) characters will be used to return this information.**

■ “MD” data

Each data will be returned as hexadecimal number using 4 characters as shown below.

Example: Data received: “HA1FC”

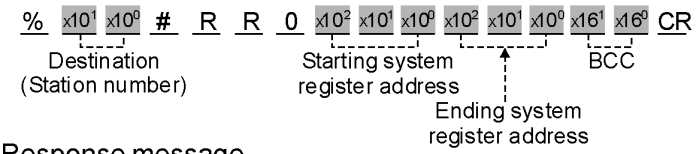


RR Read the contents of the system registers

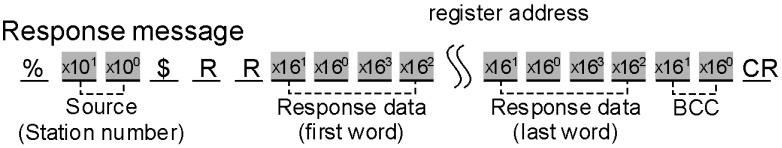
Outline Reads the contents stored in the system registers of the programmable controller.

Basic message format

Command message



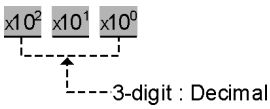
Response message



Description The contents of the system registers in the programmable controller are returned. "0" must be always placed between the command code and the starting system register number.

■ Starting/Ending system register addresses

The starting and ending system register addresses are expressed using a form as shown below :

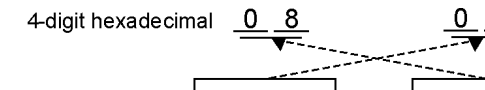


☞ **The ending system register address must be equal to or larger than the starting system register address.**

■ Response data

4 characters are needed for each system register data (one word per system register address) as shown below.
Data will be returned from the programmable controller in order from the starting to the ending system register addresses.

Data received



Actual data in the programmable controller

Hexadecimal	0	0	0	8
Decimal (K)	8			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	1 0 0 0
		High-byte	Low-byte	

Program example

Command message

% 0 1 # R R 0 0 0 5 0 0 7 ❊❊ CR

Response message

% 0 1 $ R R C 8 0 0 C 8 0 0 3 C 0 0 7 0 CR

The contents of system registers (numbers 5 to 7) will be returned by the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting number: System register 5
Ending number: System register 7

Response message:
Source: 01 station
Response data: HC800, HC800, H3C00
Actual data in programmable controller: System register 5 = H00C8,
System register 6 = H00C8,
System register 7 = H003C

Data received — 4-digit hexadecimal C 8 0 0

Actual data in the programmable controller system register 5

Hexadecimal	0	0	C	8
Decimal (K)	200			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 1 0 0	1 0 0 0

Data received — 4-digit hexadecimal C 8 0 0

Actual data in the programmable controller system register 6

Hexadecimal	0	0	C	8
Decimal (K)	200			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 1 0 0	1 0 0 0

Data received — 4-digit hexadecimal 3 C 0 0

Actual data in the programmable controller system register 7

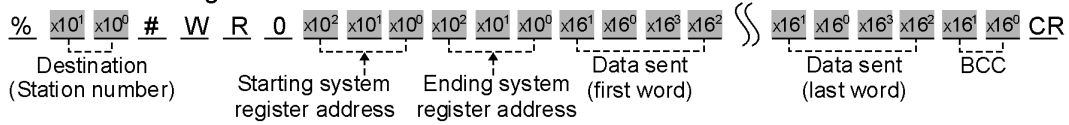
Hexadecimal	0	0	3	C
Decimal (K)	60			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 1 1	1 1 0 0

WR Write data into the system registers

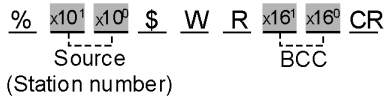
Outline Writes data into the system registers of the programmable controller.

Basic message format

Command message



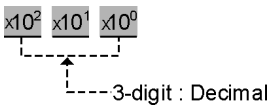
Response message



Description Data is written into the system registers of the programmable controller.
”0” must be always placed between the command code and the starting system register address.

■ Starting/Ending system register addresses

The starting and ending system register addresses are expressed using a form as shown below:



☞ **The ending system register address must be equal to or larger than the starting system register address.**

■ Data sent

4 characters are needed for each system register data (one word per system register address) as shown below.
Data will be sent to the programmable controller in order from the starting to the ending system register addresses.

Data sent 4-digit hexadecimal 0 8 0 0

Data set in the programmable controller

Hexadecimal	0	0	0	8
Decimal (K)	8			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 0 0	1 0 0 0
	High-byte		Low-byte	

Program example

Command message

% 0 1 # W R 0 0 0 5 0 0 7 C 8 0 0 C 8 0 0 3 C 0 0 ❊ ❊ CR

Response message

% 0 1 $ W R 0 5 CR

The data are written into the system registers (numbers 5 to 7) of the programmable controller whose station number is 01.

Command message:
Destination: 01 station
Starting number: System register 5
Ending number: System register 7
Data sent: HC800, HC800, H3C00
Data set in programmable controller: System register 5 = H00C8,
System register 6 = H00C8,
System register 7 = H003C

Data sent — 4-digit hexadecimal C 8 0 0

Data set in the programmable controller system register 5

Hexadecimal	0	0	C	8
Decimal (K)	200			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 1 0 0	1 0 0 0

Data sent — 4-digit hexadecimal C 8 0 0

Data set in the programmable controller system register 6

Hexadecimal	0	0	C	8
Decimal (K)	200			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	1 1 0 0	1 0 0 0

Data sent — 4-digit hexadecimal 3 C 0 0

Data set in the programmable controller system register 7

Hexadecimal	0	0	3	C
Decimal (K)	60			
Bit position	15 · · 12	11 · · 8	7 · · 4	3 · · 0
Binary	0 0 0 0	0 0 0 0	0 0 1 1	1 1 0 0

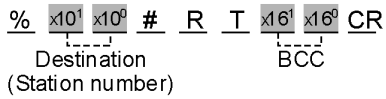
Response message
Source: 01 station

RT Read the status of the PLC

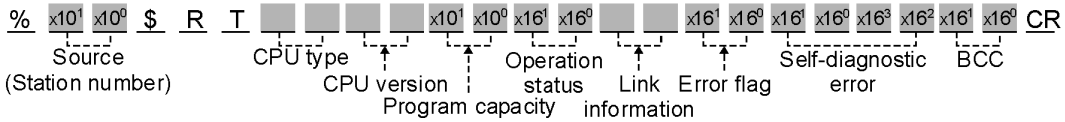
Outline Reads the status of the programmable controller.

Basic message format

Command message



Response message



Description The type of programmable controller, program capacity, operation mode and error flag status can be read with “RT” command.

CPU Type

Type of CPU which exists in the station specified in the command message, will be returned using 2 characters as shown below:

Code	CPU type	Code	CPU type
05	FP0 2.7K	03	FP3/C 10K
06	FP0 5K, 10K	13	FP3/C 16K
04	FP1/M 0.9K	02	FP5 16K
05	FP1/M 2.7K	12	FP5 24K
06	FP1/M 5K	20	FP10/10S 30K, FP10 60K
20	FPΣ	20	FP10SH 30K, 60K, 120K
50	FP2 16K, 32K	20	FP2SH 60K, 120K

CPU version

The version of the CPU which exists in the station specified in the command message, will be returned using 2 characters.

Code	CPU version
10	Version 1.0
11	Version 1.1
12	Version 1.2
•	•
•	•
35	Version 3.5
•	•
•	•
45	Version 4.5

Program capacity (for FP–C/FP3/FP5)

The program capacity will be returned using 2 characters when the destination station specified is FP–C, FP3 or FP5.

Code	Program capacity	Code	Program capacity
02	2 k (1,534) steps	14	14 k (13,822) steps
04	4 k (3,582) steps	16	16 k (15,870) steps
06	6 k (5,630) steps	18	18 k (17,918) steps
08	8 k (7,678) steps	20	20 k (19,966) steps
10	10 k (9,726) steps	22	22 k (22,014) steps
12	12 k (11,774) steps	24	24 k (24,062) steps

☞ **If the destination station is other than FP–C, FP3 or FP5, a code other than the one above is stored here.**

Operation status

The operation status is expressed as a 2–digit hexadecimal number. The contents of the operation status are the same as the data in the special relays (R9020 to R9027).

Response data (Hex)	Bit position	Specification	Content
Lower digit	0th LSB	Operation mode setting flag (R9020)	0 : In PROG. mode 1 : In RUN mode
	1st	Test run mode condition flag (R9021)	0 : Not in TEST RUN mode 1 : In TEST RUN mode
	2nd	Break execution flag (R9022)	0 : Break not executing 1 : Break executing
	3rd	Break condition flag (R9023)	0 : Break is in invalid condition 1 : Break is valid in TEST RUN mode
Upper digit	4th	Output enable/disable condition flag in the TEST RUN mode (R9024)	0 : Output disabled 1 : Output enabled in TEST RUN mode
	5th	Step run condition flag (R9025)	0 : Not in step run mode 1 : In step run mode
	6th	Message instruction [F149 (MSG) /P149 (PMSG)] flag (R9026)	0 : Message instruction not executing 1 : Message instruction executing
	7th MSB	Remote mode set flag (R9027)	0 : Not in REMOTE mode 1 : In REMOTE mode

Example: Response data: “81”

Data received 2-digit hexadecimal 8 1

Status of the programmable controller

Hexadecimal	8	1
Bit position	7 · · 4	3 · · 0
Binary	1 0 0 0	0 0 0 1
	MSB	LSB

Link information

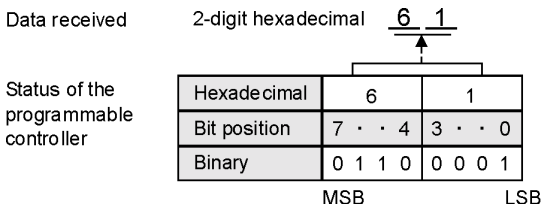
Link information is returned using 2 characters. However, as this information is meaningless for reading the status of the programmable controller, ignore this information.

Error flags (R9000 to R9007)

The status of 8 error flags will be returned using 2 characters.

Response data (Hex)	Bit position	Specification	Content
Lower digit	0th LSB	Self-diagnostic error flag (R9000)	0 : Self-diagnostic error has not occurred 1 : Self-diagnostic error occurred
	1st	Voltage dip detection flag (R9001)	0 : Voltage dip not detected 1 : Voltage dip detected
	2nd	Fuse blow detection flag (R9002)	0 : Fuse blow not detected 1 : Fuse blow detected
	3rd	Intelligent unit error flag (R9003)	0 : Intelligent unit error not detected 1 : Intelligent unit error detected
Upper digit	4th	I/O verify error flag (R9004)	0 : I/O verify error not detected 1 : I/O verify error detected
	5th	Battery voltage drop detection flag (Momentary flag) (R9005)	0 : No battery voltage drop detected 1 : Battery voltage drop detected
	6th	Battery voltage drop detection flag (Hold type flag) (R9006)	0 : No battery voltage drop detected 1 : Battery voltage drop detected
	7th MSB	Operation error flag (Momentary flag) (R9007)	0 : Operation error not detected 1 : Operation error detected

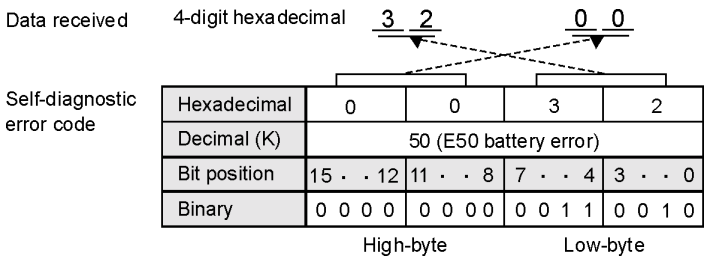
Example: Response data: “61”



Self–diagnostic errors DT9000, DT90000

Self–diagnostic error codes are read out from the programmable controller.
The contents of the codes are the same as those stored in special data register DT9000 or DT90000, depending on the PLC type used, and are expressed in hexadecimal numbers. Since they are usually expressed in decimal numbers from E1 through E299, you need to convert the hexadecimal code into decimal. If no self–diagnostic error is detected, H0000 is received.

Example: Response data: H3200

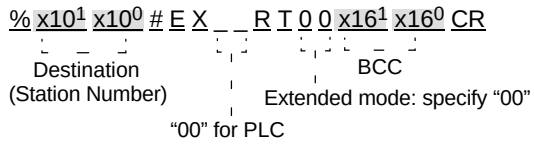


EXRT Extended read status of the PLC

Outline Reads the status of the programmable controller, but includes more information and has different CPU codes than the “RT” command.

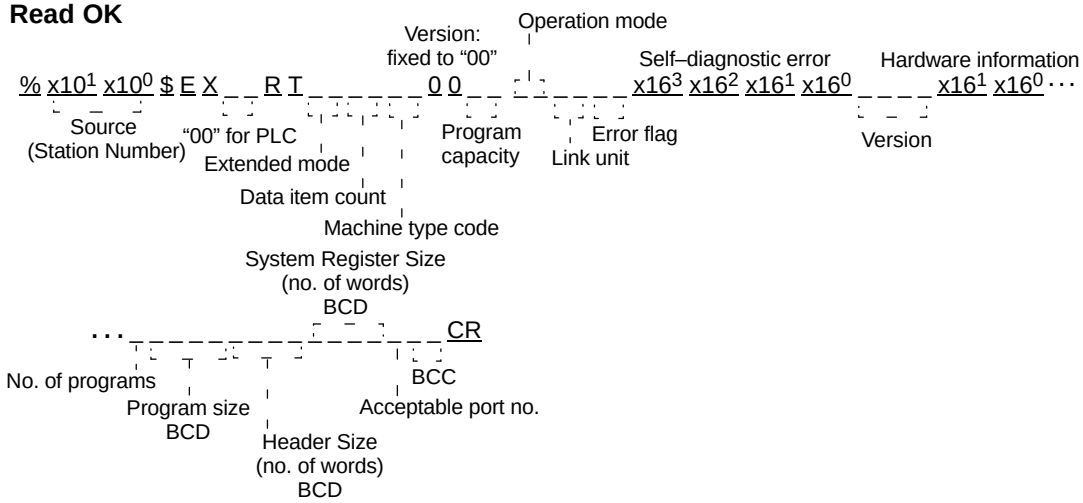
Basic message format

Command message

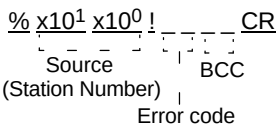


Response message

Read OK



Read error



- The details of conventional responses are the same as the details in conventional sequence status read.
- In the extended mode, specify “0” to issue commands.
- The number of bytes in the response (number of bytes from after the data item count to before BCC) is set as the data item count.

Description The “EXRT” command can read: type of programmable controller, program capacity, operation mode, the link unit, error flag status, self–diagnostic error, hardware information, number of programs, and acceptable port numbers.

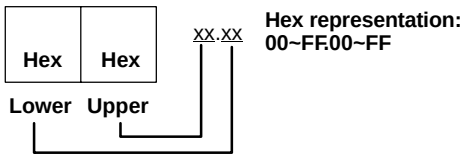
Explanation of extended responses

CPU type

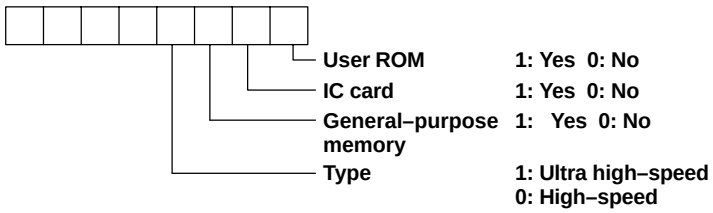
Type of CPU which exists in the station specified in the command message, will be returned using 2 characters as shown below. A "–" indicates that the PLC cannot process the "EXRT" command. Here, use the "RT" command instead.

Code	CPU type	Code	CPU type
40	FP0 2.7K	**–**	FP3/C 10K, 16K
41	FP0 5K	**–**	FP5 16K, 24K
42	FP0 10K	**–**	FP10/10S 30K, FP10 60K
–	FP1/M 0.9K, 2.7K, 5K	**30**	FP10SH 30K, 60K, 120K
43	FPΣ	**60**	FP2SH 60K, 120K
50	FP2 16K, 32K	**–**	—

Version



Hardware information



Number of programs

Number of programs	No.
No. 1 program area only	1
No. 1/No. 2 program areas	2

Program size (KB)

Program area, examples	Size
No. 1 program area (30 KB)	30
No. 1 program area (60 KB)	60
No. 1/No. 2 program areas	120

Acceptable port nos.

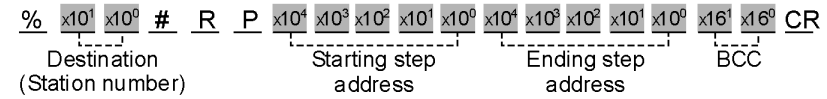
Acceptable port nos.	Code
Link unit	0
Tool port	0
Computer link port	2

RP Read a program stored in the PLC

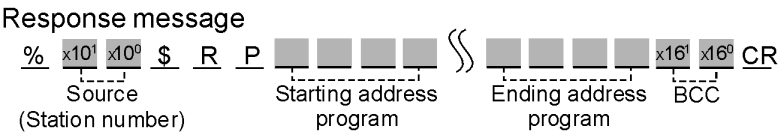
Outline Reads a program stored in the programmable controller.
This command is available only for program backup purposes.
(Note that the read–out program cannot be read using NPST–GR.)

Basic message format

Command message



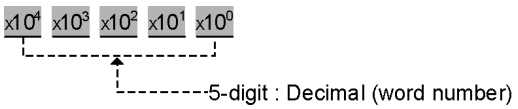
Response message



Description The program from the specified address is returned by the programmable controller.
This command should be used to save the program block only for backup purposes.

■ Starting step address/Ending step address

Starting and ending step addresses for the program are expressed as 5–digit decimal numbers as shown below:



☞ **The ending step address must be equal to or larger than the starting step address.**

■ Program

Each program step will be returned as 4 characters.

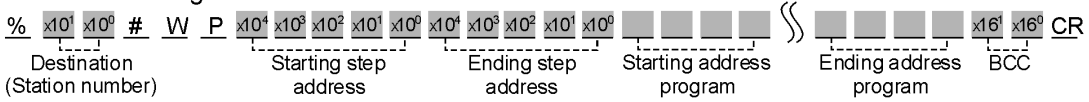
☞ **To avoid malfunctions in the programmable controller, it is recommended that you do not modify or review the program that is read out.**

WP Write a program which was saved by using the “RP” command back into the PLC

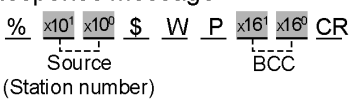
Outline Writes the program saved with the “RP“ command back into the programmable controller.
This command is available only for program downloading purposes.

Basic message format

Command message



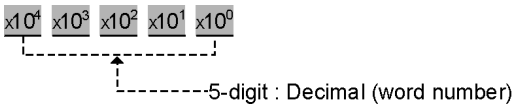
Response message



Description A program which was saved using the “RP” command is written back into the programmable controller.
This command should be used only for downloading the program block saved by using the “RP” command.

■ Starting step address/Ending step address

Starting and ending step addresses for the program are expressed using a 5–digit decimal as shown below:



☞ **The ending step address must be equal to or larger than the starting step address.**

■ Program

Each program step will require 4 characters to be written back into the programmable controller.

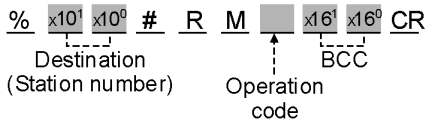
☞ **The program which is downloaded must be a program which was saved using the “RP” command.**
If you modify or revise the program, malfunction may occur.

RM Remote control of PLC operation mode

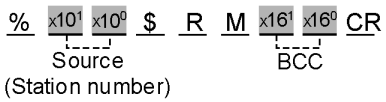
Outline Remotely controls the operation mode.
The operation mode is remotely set to the RUN or PROG. mode.

Basic message format

Command message



Response message



Description Controls the operation mode.
The operation mode is remotely set to the RUN or PROG. mode.

☞ **The "RM" command is only valid when the programmable controller is set to REMOTE mode. For details, refer to the Hardware manuals for each programmable controller.**

■ **Operation code**

Operation code	Specification
R	PROG. mode → RUN mode
P	RUN mode → PROG. mode

Program example

Command message

% 0 1 # R M R ❊ ❊ CR

Response message

% 0 1 $ R M 1 F CR

The operation mode of the programmable controller, whose station number is 01, is set to the RUN mode.

Command message:
Destination: 01 station
Data sent: PROG. mode .RUN mode

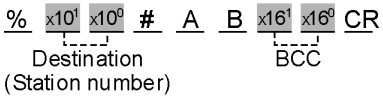
Response message:
Source: 01 station

AB Abort a series of response messages

Outline Aborts a series of messages.
This command is used to abort the reception of a response message sent in multiple frames.

Basic message format

Command message



Response message

No response message

Description This command cancels a message being sent in multiple frames. The cancellation occurs in the middle of the communication, when you want to stop receiving the response message for any reason.

Chapter 2

MEWTOCOL–DAT Protocol

2.1 MEWTOCOL–DAT Protocol

The MEWTOCOL–DAT protocol is used for communication (data transfer) between a computer and an FP series programmable controller. A command is initiated from a programmable controller (using instructions) to a computer and the computer sends a response message back to the programmable controller in the MEWTOCOL–DAT format.

All messages are transmitted in binary codes. Therefore, all data you receive from or send to an FP series programmable controller should be handled in binary code. For easier understanding, all descriptions in this section will be expressed in hexadecimal codes.

Basic terminology of MEWTOCOL–DAT

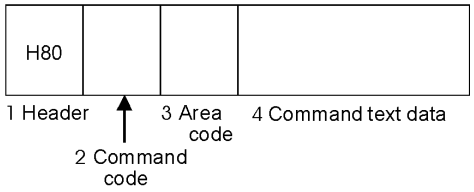
- **Message:**
  A series of binary data combining commands and text. A maximum of 1,020 words of data are available for text when communicating in a network with only high–level link units. A maximum of 16 words of data are available for text when communicating in a network with standard link units.
- **Command message:**
  A message which is sent to or from the programmable controller or computer. The programmable controller can issue command messages by executing the F145 (SEND)/P145 (PSEND) and F146 (RECV)/P146 (PRECV) instructions.

Item	Send/receive instruction	Command code of MEWTOCOL-DAT
Write data in word units	**F145 (SEND)**	H50
Write a bit data	**P145 (PSEND)**	H52
Read data in word units	**F146 (RECV)**	H51
Read a bit data	**P146 (PRECV)**	H53

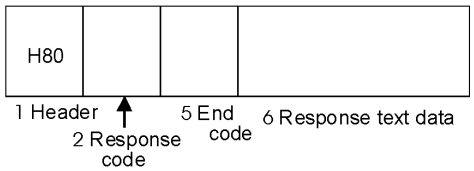
- **Response message:**
  A message which is issued by responding to a command message. When a computer issues a command message, the programmable controller sends it back to the computer. When a programmable controller issues a command message, the computer sends it back to the programmable controller.

2.1.1 Basic MEWTOCOL–DAT Message Format

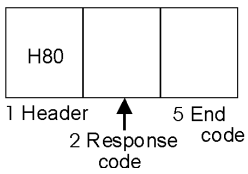
■ **Command message format**



■ **Response message format**



■ **Error response message format**



1. Header (H80)

H80 is used as the header in both the command and response messages.

2. Command codes (H50 to H53) and response codes (HD0 to HD3)

Command and response codes are specified using one byte as follows:

Command code	Corresponding response code	Description
H50	HD0	Command and response codes for writing data in word units.
H51	HD1	Command and response codes for reading data in word units.
H52	HD2	Command and response codes for writing a bit data.
H53	HD3	Command and response codes for reading a bit data.

3. Area codes

The operand is specified using one byte as follows:

Area code	Description
H00	Word link relays WL
H01	Word internal relays WR
H02	Word external output relays WY
H03	Word external input relays WX
H04	Timer/counter set value SV
H05	Timer/counter elapsed value EV
H06	Link data register LD
H07	Word special internal relays WR
H08	Special data register DT
H09	Data register DT
H0A	File register FL

4. Command text data

Depending on the command, the contents of text will vary.
Information such as memory addresses and data are specified here.

5. End codes (HFF or error codes)

The end code indicates the communication status using MEWTOCOL–DAT as follows:
– HFF: The operation has successfully completed.
– Other than HFF: An error was detected. For details about error codes, see chapter 3.

6. Response text data

When a command, which requests to have data sent back in a response message, is transmitted, it is followed by the end code of the response message.

2.2 MEWTOCOL–DAT Commands and Responses

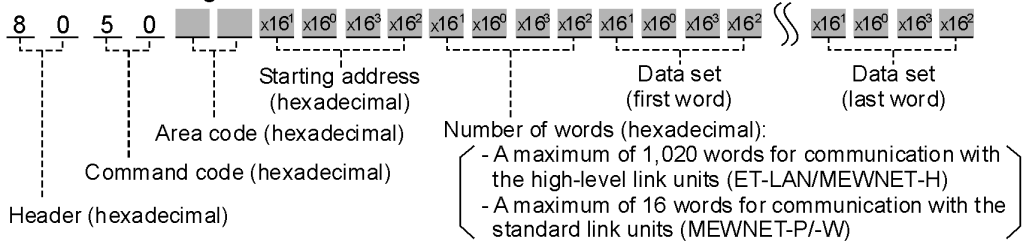
Descriptions for each MEWTOCOL–DAT command and response messages are explained on the pages shown below.

H50 Write data in word units

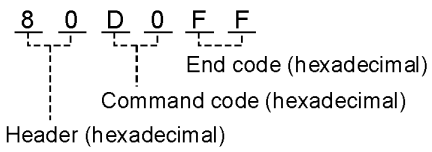
Outline Writes data into a specified area in word units.

Basic message format

Command message



Response message



List of memory area codes (hexadecimal)

Name of operands	Relay				Timer/ Counter		Register			Special internal relay	Special data register
	WX	WY	WR	WL	SV	EV	DT	LD	FL	WR	DT
Area code (HEX)	03	02	01	00	04	05	09	06	0A	07	08

Program example

Command message

8 0 5 0 0 9 0 1 0 0 0 3 0 0 5 0 0 1 0 3 0 0 0 0 0 2

Response message

8 0 D 0 F F

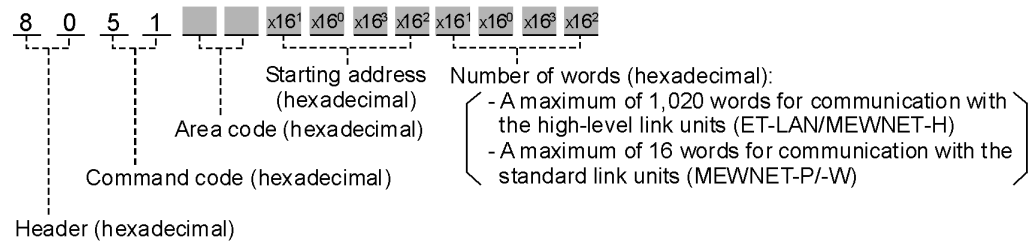
Data are transferred into data registers DT1, DT2, and DT3 as follows:
– Data set in DT1: H0150 (K336)
– Data set in DT2: H0003 (K3)
– Data set in DT3: H0200 (K512)

H51 Read data in word units

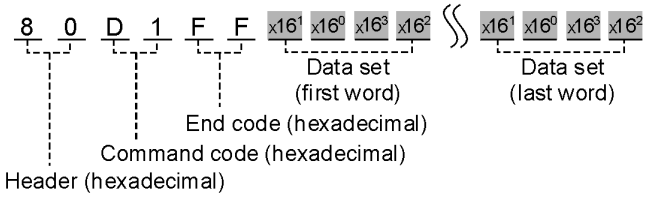
Outline Read data from a specified area in word units.

Basic message format

Command message



Response message



List of memory area codes (hexadecimal)

Name of operands	Relay				Timer/ Counter		Register			Special internal relay	Special data register
	WX	WY	WR	WL	SV	EV	DT	LD	FL	WR	DT
Area code (HEX)	03	02	01	00	04	05	09	06	0A	07	08

Program example

Command message

8 0 5 1 0 9 1 0 0 0 0 3 0 0

Response message

8 0 D 1 F F 5 0 0 1 0 3 0 0 0 0 0 2

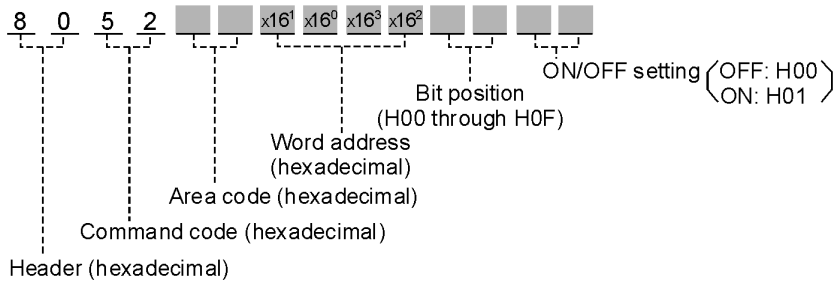
Data are transferred into data registers DT1, DT2, and DT3 as follows:
– Data read from DT16: H0150 (K336)
– Data read from DT17: H0003 (K3)
– Data read from DT18: H0200 (K512)

H52 Write a bit data

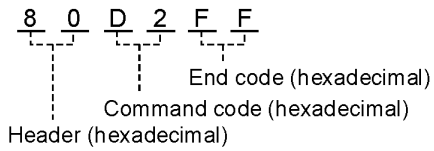
Outline Writes data into a bit of a specified word.

Basic message format

Command message



Response message



List of memory area codes (hexadecimal)

Name of operands	Relay				Timer/ Counter		Register			Special internal relay	Special data register
	WX	WY	WR	WL	SV	EV	DT	LD	FL	WR	DT
Area code (HEX)	03	02	01	00	04	05	09	06	0A	07	08

Program example

Command message

8 0 5 2 0 1 1 3 0 0 0 F 0 1

Response message

8 0 D 2 F F

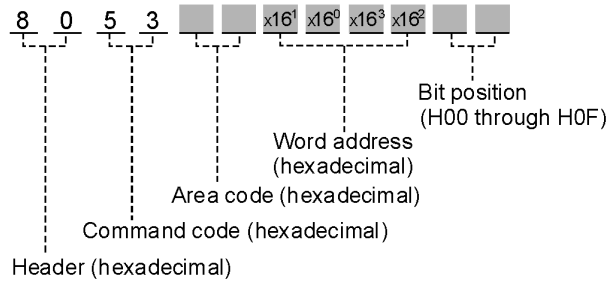
Bit position 15 of word internal relay WR19 (R19F) is turned ON.

H53 Read a bit data

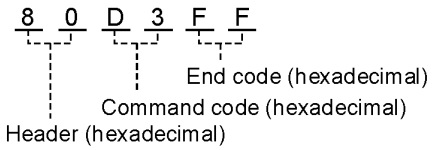
Outline Reads the bit of a specified word.

Basic message format

Command message



Response message



List of memory area codes (hexadecimal)

Name of operands	Relay				Timer/ Counter		Register			Special internal relay	Special data register
	WX	WY	WR	WL	SV	EV	DT	LD	FL	WR	DT
Area code (HEX)	03	02	01	00	04	05	09	06	0A	07	08

Program example

Command message

8 0 5 3 0 3 2 0 0 0 0 E

Response message

8 0 D 3 F F 0 1

The data in bit position 14 of word external input relay WX32 (X32E) is read out.

Chapter 3

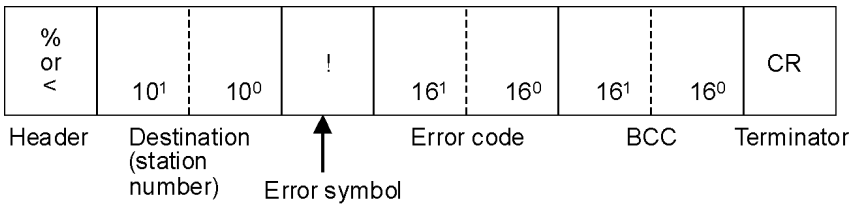
MEWTOCOL Error Codes

3.1 List of MEWTOCOL Error Codes

When an error occurs during a computer link and data transfer operation, the error code is sent back in the MEWTOCOL–COM or MEWTOCOL–DAT response message as follows:

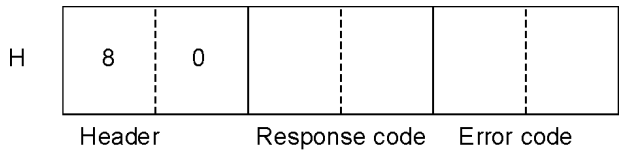
■ Computer link function (MEWTOCOL–COM)

The error code is stored in the response message as follows:



■ Data transfer function (MEWTOCOL–DAT)

The error code is stored in the response message as follows:



The same error code is also stored in special data registers as follows:
– FP3 (without C) and FP3C: DT9039
– FP10S and FP10: DT90039

3.2 MEWTOCOL Error Code Tables

MEWTOCOL error codes are usually expressed in hexadecimal in both MEWTOCOL–COM and MEWTOCOL–DAT response messages. The hexadecimal MEWTOCOL error codes are also expressed in ASCII HEX for convenience when reading MEWTOCOL–COM messages.

3.2.1 Table of Link Error Codes

Error code hexadecimal (ASCII code)	Name of error	Description	Steps to take
H21 (H32) (H31)	NACK error	Data error such as parity error and framing error occurred.	Check the communication format, cable connection and ambient noise level.
H22 (H32) (H32)	WACK error	Overflow of the receive–buffer occurred in the local node.	Reconfigure the receive–buffer size or send data size.
H23 (H32) (H33)	Source MEWTOCOL station number overlap	Source MEWTOCOL station number overlapped with that for another node, then the communication was shut down.	Reconfigure the MEWTOCOL station number in the network without overlap.
H24 (H32) (H34)	Transmission error	Data not conforming to the transmission protocol format was sent. Or a frame overflow or data error occurred.	Check the protocol format referring to the MEWTOCOL description.
H25 (H32) (H35)	Link unit hardware error	Communication parts of the link unit did not work properly.	Turn OFF the power for the system and then turn it ON again. If communication goes well, probably an abnormality caused by noise. Check the cable connection and ambient noise level. If communication cannot be performed well, probably a hardware abnormality. Replace the link unit with a new one.
H26 (H32) (H36)	MEWTOCOL station number setting error	The MEWTOCOL station number set for the source node was outside the specified range (ET–LAN system in the range of 1 to 64).	Set the MEWTOCOL station number within the specified range for the network.
H27 (H32) (H37)	Frame–over error	Data over the specified limit was transmitted. **Example** for MEWTOCOL–COM: 118 characters (using % header) or 2,048 characters (using < header in the high–level link unit).	Check the limitations of the frames for each link unit.
H28 (H32) (H38)	No response error	No response was sent back to the source station from the destination node.	Re–send the same data again.
H29 (H32) (H39)	Buffer close error	Data was transferred to or from the source node when its buffer was closed.	Open the buffer referring to the manual for each link unit.
H30 (H33) (H30)	Time–out error.	Data cannot be transferred.	Re–send the same data again.

Error code hexadecimal (ASCII code)	Name of error	Description	Steps to take
H32 (H33) (H32)	Transmission impossible error	Communication was shut down because of buffer overflow of the source node.	Reconfigure send data or buffer size so that the data size is within the limitation.
H33 (H33) (H33)	Communication stop	Network entry switch is OFF.	Turn the network entry switch ON.
H36 (H33) (H36)	No local station error	The source station does not exist in the network.	Check that the specified local station exists and re–send data again.
H38 (H33) (H38)	Other communication errors	Probably a transmission abnormality other than described above.	Redo the communication.

- **If an error occurs during communication in the 2nd or 3rd depth of the layers, an error response will not return.**
- **If a link error occurs, any other error (e. g., basic procedure error, processing error, or application error) will not be reported.**

3.2.2 Table of Basic Procedure Error Codes

Error code hexadecimal (ASCII code)	Name of error	Description	Steps to take
H40 (H34) (H30)	BCC error	BCC error occurred in the command data.	Check the connection of the cables and ambient noise level.
H41 (H34) (H31)	Format error	The command message does not match the protocol format.	Correct command message and resend the correct one.
H42 (H34) (H32)	Not–support error	The command not supported by the source or destination node was transmitted.	Check that the command message sent is supported by the source and destination nodes.
H43 (H34) (H33)	Procedure error	Another series of messages was sent to one node when a series of messages in multiple frames was being sent.	Change the program so that another message series is not sent while one series is still in progress.

3.2.3 Table of Processing Error Codes

Processing error codes are errors for the computer link function.

Error code hexadecimal (ASCII code)	Name of error	Description	Steps to take
H50 (H35) (H30)	Link setting error	[Computer link function error.] The route number where no link unit existed was specified in the computer link function.	Check the route number and set the correct one.
H51 (H35) (H30)	Simultaneous operation error	[Computer link function error.] The send–buffer overflowed while sending data to the local node in the computer link function.	Re–send data.
H52 (H35) (H32)	Sending disable error	[Computer link function error.] The sending operation to another node cannot be performed in the computer link function.	Turn OFF the power for the system and then turn it ON again. If communication goes well, probably an abnormality caused by noise. Check the cable connection and ambient noise level. If communication cannot be performed well, probably a hardware abnormality. Replace the link unit with a new one.
H53 (H35) (H33)	Busy error	[Computer link function error.] A new command was received from a local node while processing multiple frames.	Re–send the command again.

3.2.4 Table of Application Error Codes

Error code hexadecimal (ASCII code)	Name of error	Description	Steps to take
H60 (H36) (H30)	Parameter error	[Computer link function error.] The area code specified is not available for the CPU or the command in the computer link function.	Reset the correct area code.
H61 (H36) (H31)	Data error	[Computer link function error.] The specified data format, such as number system, data range, etc., was not correct.	Correct the data format referring to the description for MEWTOCOL–COM format.
H62 (H36) (H32)	Registration error	[Computer link function error.] The specified operands used for monitoring were not correct.	Set parameters for correct monitoring referring to the descriptions of MC and MD commands.
H63 (H36) (H33)	Mode error	[Computer link function error.] In the current operation mode of the PLC, operation of the command cannot be performed.	Change the operation mode.
H65 (H36) (H35)	Protect error	[Computer link function error.] The program was written to the PLC when writing to memory was prohibited.	It is impossible to write program into the PLC when the memory is protected.
H66 (H36) (H36)	Address error	[Computer link function error.] The address setting format, such as number system , address limitations, etc., was not correct.	Correct the address format referring to the description of the MEWTOCOL–COM format.
H67 (H36) (H37)	No data error	[Computer link function error.] The area without data was specified for reading.	Specify the correct area for reading.
H72 (H37) (H32)	Time–out error	[Data transfer function error] The CPU could not receive the answer within the specified time.	Re–send data.
H73 (H37) (H33)	Time–out error	[Data transfer function error] The receive–buffer did not become available within the specified time.	Re–send data.
H72 (H37) (H34)	Time–out error	[Data transfer function error] The response could not be received within the specified time.	Re–send data.

Index

W

Record of Changes

Manual No.	Date	Description of Changes
ACGM0125V1.0END	JULY 2002	Update of ACGM0127END V1.0, FP2 CCU Hardware Manual, MEWTOCOL. FP2 CCU information deleted because MEW released comprehensive FP2 CCU manual, ARCT1F319V10END.

GLOBAL NETWORK

North America	Europe	Asia Pacific	China	Japan
Aromat Corporation	**Matsushita Electric Works**	**Matsushita Electric Works**	**Matsushita Electric Works**	**Matsushita Electric Works Ltd. Automation Controls Group**

Europe

■ **Europe** — **Matsushita Electric Works (Europe) AG**
Rudolf–Diesel–Ring 2, D–83607 Holzkirchen, Tel. (08024) 648–0, Fax (08024) 648–111, www.mew–europe.com

■ **Austria** — **Matsushita Electric Works Austria GmbH**
Josef Madersperger Straße 2, A-2362 Biedermannsdorf, Austria, Tel. (02236) 2 68 46, Fax (02236) 46133, www.matsushita.at

■ **Benelux** — **Matsushita Electric Works Benelux B. V.**
De Rijn 4, (Postbus 211), 5684 PJ Best, (5680 AE Best), Netherlands, Tel. (0499) 37 2727, Fax (0499) 372185, www.matsushita.nl

■ **France** — **Matsushita Electric Works France S.A.R.L.**
B.P. 44, F-91371 Verrières le Buisson CEDEX, France, Tél. 01 60 13 57 57, Fax 01 60 13 57 58, www.matsushita–france.fr

■ **Germany** — **Matsushita Electric Works Deutschland GmbH**
Rudolf–Diesel–Ring 2, 83607 Holzkirchen, Germany, Tel. (08024) 648–0, Fax (08024) 648–555, www.matsushita.de

■ **Ireland** — **Matsushita Electric Works UK Ltd. Irish Branch Office**
Waverley, Old Naas Road, Bluebell, Dublin 12, Republic of Ireland, Tel. (01) 460 09 69, Fax (01) 460 11 31, www.matsushita.ie

■ **Italy** — **Matsushita Electric Works Italia s.r.l.**
Via del Commercio 3-5 (Z.I. Ferlina), I-37012 Bussolengo (VR), Italy, Tel. (045) 675 27 11, Fax (045) 670 04 44, www.matsushita.it

■ **Portugal** — **Matsushita Electric Works Portugal España S.A. Portuguese Branch Office**
Avda 25 de Abril, Edificio Alvorada 5º E, 2750-512 Cascais, Portugal, Tel. (21) 482 82 66, Fax (21) 482 74 21

■ **Scandinavia** — **Matsushita Electric Works Scandinavia AB**
Sjöängsvägen 10, 19272 Sollentuna, Sweden, Tel. +46 8 59 47 66 80, Fax (+46) 8 59 47 66 90, www.matsushita.se

■ **Spain** — **Matsushita Electric Works España S.A.**
Parque Empresarial Barajas, San Severo, 20, 28042 Madrid, España, Tel. (91) 329 38 75, Fax (91) 329 29 76, www.matsushita.es

■ **Switzerland** — **Matsushita Electric Works Schweiz AG**
Grundstrasse 8, CH-6343 Rotkreuz, Switzerland, Tel. (041) 799 70 50, Fax (041) 799 70 55, www.matsushita.ch

■ **UK** — **Matsushita Electric Works UK Ltd.**
Sunrise Parkway, Linford Wood East, Milton Keynes, MK14 6LF, England, Tel. (01908) 231 555, Fax (01908) 231 599, www.matsushita.co.uk

North & South America

■ **USA** — **Aromat Corporation Head Office USA**
629 Central Avenue, New Providence, N.J. 07974, USA, Tel. 1–908–464–3550, Fax 1–908–464–8513, www.aromat.com

Asia

■ **China** — **Matsushita Electric Works, Ltd. China Office**
2013, Beijing Fortune, Building 5, Dong San Huan Bei Lu, Chaoyang District, Beijing, China, Tel. 86–10–6590–8646, Fax 86–10–6590–8647

■ **Hong Kong** — **Matsushita Electric Works Ltd. Hong Kong**
Rm1601, 16/F, Tower 2, The Gateway, 25 Canton Road, Tsimshatsui, Kowloon, Hong Kong, Tel. (852) 2956–3118, Fax (852) 2956–0398

■ **Japan** — **Matsushita Electric Works Ltd. Automation Controls Group**
1048 Kadoma, Kadoma–shi, Osaka 571–8686, Japan, Tel. 06–6908–1050, Fax 06–6908–5781, www.mew.co.jp/e–acg/

■ **Singapore** — **Matsushita Electric Works (Asia Pacific) Pte. Ltd.**
101 Thomson Road, #25–03/05, United Square, Singapore 307591,Tel. (65) 6255–5473, Fax (65) 6253–5689

Specifications are subject to change without notice. Printed in Europe