MITSUBISHI ELECTRIC

Programmable Controller

MELSEC iQ-R series

# MELSEC iQ-R Ethernet User's Manual (Application)

-RJ71EN71
-R00CPU
-R01CPU
-R02CPU
-R04CPU
-R04ENCPU
-R08CPU
-R08ENCPU
-R08PCPU
-R08PSFCPU
-R08SFCPU
-R16CPU
-R16ENCPU
-R16PCPU
-R16PSFCPU
-R16SFCPU
-R32CPU
-R32ENCPU
-R32PCPU
-R32PSFCPU
-R32SFCPU
-R120CPU
-R120ENCPU
-R120PCPU
-R120PSFCPU
-R120SFCPU

# SAFETY PRECAUTIONS

(Read these precautions before using this product.)

Before using this product, please read this manual and the relevant manuals carefully and pay full attention to safety to handle the product correctly. If the equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

The precautions given in this manual are concerned with this product only. For the safety precautions of the programmable controller system, refer to the MELSEC iQ-R Module Configuration Manual.

In this manual, the safety precautions are classified into two levels: "⚠WARNING" and "⚠CAUTION".

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

Under some circumstances, failure to observe the precautions given under "⚠CAUTION" may lead to serious consequences.

Observe the precautions of both levels because they are important for personal and system safety.

Make sure that the end users read this manual and then keep the manual in a safe place for future reference.

## [Design Precautions]

### ⚠WARNING

- Configure safety circuits external to the programmable controller to ensure that the entire system operates safely even when a fault occurs in the external power supply or the programmable controller. Failure to do so may result in an accident due to an incorrect output or malfunction.
  (1) Emergency stop circuits, protection circuits, and protective interlock circuits for conflicting operations (such as forward/reverse rotations or upper/lower limit positioning) must be configured external to the programmable controller.
  (2) When the programmable controller detects an abnormal condition, it stops the operation and all outputs are:
    - Turned off if the overcurrent or overvoltage protection of the power supply module is activated.
    - Held or turned off according to the parameter setting if the self-diagnostic function of the CPU module detects an error such as a watchdog timer error.
  (3) All outputs may be turned on if an error occurs in a part, such as an I/O control part, where the CPU module cannot detect any error. To ensure safety operation in such a case, provide a safety mechanism or a fail-safe circuit external to the programmable controller. For a fail-safe circuit example, refer to "General Safety Requirements" in the MELSEC iQ-R Module Configuration Manual.
  (4) Outputs may remain on or off due to a failure of a component such as a relay and transistor in an output circuit. Configure an external circuit for monitoring output signals that could cause a serious accident.
- In an output circuit, when a load current exceeding the rated current or an overcurrent caused by a load short-circuit flows for a long time, it may cause smoke and fire. To prevent this, configure an external safety circuit, such as a fuse.
- Configure a circuit so that the programmable controller is turned on first and then the external power supply. If the external power supply is turned on first, an accident may occur due to an incorrect output or malfunction.
- Configure a circuit so that the external power supply is turned off first and then the programmable controller. If the programmable controller is turned off first, an accident may occur due to an incorrect output or malfunction.
- For the operating status of each station after a communication failure, refer to manuals for the network used. For the manuals, please consult your local Mitsubishi representative. Incorrect output or malfunction due to a communication failure may result in an accident.

## [Design Precautions]

### ⚠WARNING

- When connecting an external device with a CPU module or intelligent function module to modify data of a running programmable controller, configure an interlock circuit in the program to ensure that the entire system will always operate safely. For other forms of control (such as program modification, parameter change, forced output, or operating status change) of a running programmable controller, read the relevant manuals carefully and ensure that the operation is safe before proceeding. Improper operation may damage machines or cause accidents. When a Safety CPU is used, data cannot be modified while the Safety CPU is in SAFETY MODE.
- Especially, when a remote programmable controller is controlled by an external device, immediate action cannot be taken if a problem occurs in the programmable controller due to a communication failure. To prevent this, configure an interlock circuit in the program, and determine corrective actions to be taken between the external device and CPU module in case of a communication failure.
- Do not write any data to the "system area" and "write-protect area" of the buffer memory in the module. Also, do not use any "use prohibited" signals as an output signal from the CPU module to each module. Doing so may cause malfunction of the programmable controller system. For the "system area", "write-protect area", and the "use prohibited" signals, refer to the user's manual for the module used. For areas used for safety communications, they are protected from being written by users, and thus safety communications failure caused by data writing does not occur.
- If a communication cable is disconnected, the network may be unstable, resulting in a communication failure of multiple stations. Configure an interlock circuit in the program to ensure that the entire system will always operate safely even if communications fail. Failure to do so may result in an accident due to an incorrect output or malfunction. When safety communications are used, an interlock by the safety station interlock function protects the system from an incorrect output or malfunction.

## [Design Precautions]

### ⚠CAUTION

- Do not install the control lines or communication cables together with the main circuit lines or power cables. Doing so may result in malfunction due to electromagnetic interference. Keep a distance of 100mm or more between those cables.
- During control of an inductive load such as a lamp, heater, or solenoid valve, a large current (approximately ten times greater than normal) may flow when the output is turned from off to on. Therefore, use a module that has a sufficient current rating.
- After the CPU module is powered on or is reset, the time taken to enter the RUN status varies depending on the system configuration, parameter settings, and/or program size. Design circuits so that the entire system will always operate safely, regardless of the time.
- Do not power off the programmable controller or reset the CPU module while the settings are being written. Doing so will make the data in the flash ROM and SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM and SD memory card again. Doing so also may cause malfunction or failure of the module.
- When changing the operating status of the CPU module from external devices (such as the remote RUN/STOP functions), select "Do Not Open by Program" for "Opening Method" of "Module Parameter". If "Open by Program" is selected, an execution of the remote STOP function causes the communication line to close. Consequently, the CPU module cannot reopen the line, and external devices cannot execute the remote RUN function.
- The base unit and modules mounted on the base unit must be powered by a power supply that complies with Safety Extra-Low Voltage (SELV) and Limited Energy Circuit (LIM) or one that complies with UL 1310 Class 2.

## [Security Precautions]

### ⚠WARNING

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

## [Installation Precautions]

### ⚠WARNING

- Shut off the external power supply (all phases) used in the system before mounting or removing the module. Failure to do so may result in electric shock or cause the module to fail or malfunction.

## [Installation Precautions]

### ⚠CAUTION

- Use the programmable controller in an environment that meets the general specifications in the MELSEC iQ-R Module Configuration Manual. Failure to do so may result in electric shock, fire, malfunction, or damage to or deterioration of the product.
- To mount a module, place the concave part(s) located at the bottom onto the guide(s) of the base unit, and push in the module until the hook(s) located at the top snaps into place. Incorrect interconnection may cause malfunction, failure, or drop of the module.
- To mount a module with no module fixing hook, place the concave part(s) located at the bottom onto the guide(s) of the base unit, push in the module, and fix it with screw(s). Incorrect interconnection may cause malfunction, failure, or drop of the module.
- When using the programmable controller in an environment of frequent vibrations, fix the module with a screw.
- Tighten the screws within the specified torque range. Undertightening can cause drop of the component or wire, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction. For the specified torque range, refer to the MELSEC iQ-R Module Configuration Manual.
- When using an extension cable, connect it to the extension cable connector of the base unit securely. Check the connection for looseness. Poor contact may cause malfunction.
- When using an SD memory card, fully insert it into the SD memory card slot. Check that it is inserted completely. Poor contact may cause malfunction.
- Securely insert an extended SRAM cassette or a battery-less option cassette into the cassette connector of the CPU module. After insertion, close the cassette cover and check that the cassette is inserted completely. Poor contact may cause malfunction.
- Beware that the module could be very hot while power is on and immediately after power-off.
- Do not directly touch any conductive parts and electronic components of the module, SD memory card, extended SRAM cassette, battery-less option cassette, or connector. Doing so can cause malfunction or failure of the module.

## [Wiring Precautions]

### ⚠WARNING

- Shut off the external power supply (all phases) used in the system before installation and wiring. Failure to do so may result in electric shock or cause the module to fail or malfunction.
- After installation and wiring, attach a blank cover module (RG60) to each empty slot before powering on the system for operation. Also, attach an extension connector protective cover[*1] to each unused extension cable connector as necessary. Directly touching any conductive parts of the connectors while power is on may result in electric shock.
  *1 For details, please consult your local Mitsubishi Electric representative.

## [Wiring Precautions]

### ⚠CAUTION

- Individually ground the FG and LG terminals of the programmable controller with a ground resistance of 100 ohms or less. Failure to do so may result in electric shock or malfunction.
- Use applicable solderless terminals and tighten them within the specified torque range. If any spade solderless terminal is used, it may be disconnected when the terminal screw comes loose, resulting in failure.
- Check the rated voltage and signal layout before wiring to the module, and connect the cables correctly. Connecting a power supply with a different voltage rating or incorrect wiring may cause fire or failure.
- Connectors for external devices must be crimped or pressed with the tool specified by the manufacturer, or must be correctly soldered. Incomplete connections may cause short circuit, fire, or malfunction.
- Securely connect the connector to the module. Poor contact may cause malfunction.
- Do not install the control lines or communication cables together with the main circuit lines or power cables. Doing so may result in malfunction due to noise. Keep a distance of 100mm or more between those cables.
- Place the cables in a duct or clamp them. If not, dangling cables may swing or inadvertently be pulled, resulting in malfunction or damage to modules or cables.
  In addition, the weight of the cables may put stress on modules in an environment of strong vibrations and shocks.
  Do not clamp the extension cables with the jacket stripped. Doing so may change the characteristics of the cables, resulting in malfunction.
- Check the interface type and correctly connect the cable. Incorrect wiring (connecting the cable to an incorrect interface) may cause failure of the module and external device.
- Tighten the terminal screws or connector screws within the specified torque range. Undertightening can cause drop of the screw, short circuit, fire, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, fire, or malfunction.
- When disconnecting the cable from the module, do not pull the cable by the cable part. For the cable with connector, hold the connector part of the cable. For the cable connected to the terminal block, loosen the terminal screw. Pulling the cable connected to the module may result in malfunction or damage to the module or cable.
- When wiring, shield the module using a cover or similar means to prevent foreign matter such as dust or wire chips from entering the module. Such foreign matter can cause a fire, failure, or malfunction. Remove the shield for heat dissipation before operating the system.
- When a protective film is attached to the top of the module, remove it before system operation. If not, inadequate heat dissipation of the module may cause a fire, failure, or malfunction.
- Programmable controllers must be installed in control panels. Connect the main power supply to the power supply module in the control panel through a relay terminal block. Wiring and replacement of a power supply module must be performed by qualified maintenance personnel with knowledge of protection against electric shock. For wiring, refer to the MELSEC iQ-R Module Configuration Manual.
- For Ethernet cables to be used in the system, select the ones that meet the specifications in the user's manual for the module used. If not, normal data transmission is not guaranteed.

## [Startup and Maintenance Precautions]

### ⚠WARNING

- Do not touch any terminal while power is on. Doing so will cause electric shock or malfunction.
- Correctly connect the battery connector. Do not charge, disassemble, heat, short-circuit, solder, or throw the battery into the fire. Also, do not expose it to liquid or strong shock. Doing so will cause the battery to produce heat, explode, ignite, or leak, resulting in injury and fire.
- Shut off the external power supply (all phases) used in the system before cleaning the module or retightening the terminal screws, connector screws, or module fixing screws. Failure to do so may result in electric shock.

## [Startup and Maintenance Precautions]

### ⚠CAUTION

- When connecting an external device with a CPU module or intelligent function module to modify data of a running programmable controller, configure an interlock circuit in the program to ensure that the entire system will always operate safely. For other forms of control (such as program modification, parameter change, forced output, or operating status change) of a running programmable controller, read the relevant manuals carefully and ensure that the operation is safe before proceeding. Improper operation may damage machines or cause accidents.
- Especially, when a remote programmable controller is controlled by an external device, immediate action cannot be taken if a problem occurs in the programmable controller due to a communication failure. To prevent this, configure an interlock circuit in the program, and determine corrective actions to be taken between the external device and CPU module in case of a communication failure.
- Do not disassemble or modify the modules. Doing so may cause failure, malfunction, injury, or a fire.
- Use any radio communication device such as a cellular phone or PHS (Personal Handy-phone System) 25cm or more away in all directions from the programmable controller. Failure to do so may cause malfunction.
- Shut off the external power supply (all phases) used in the system before mounting or removing the module. Failure to do so may cause the module to fail or malfunction.
- Tighten the screws within the specified torque range. Undertightening can cause drop of the component or wire, short circuit, or malfunction. Overtightening can damage the screw and/or module, resulting in drop, short circuit, or malfunction.
- After the first use of the product, do not perform each of the following operations more than 50 times (IEC 61131-2/JIS B 3502 compliant).
  Exceeding the limit may cause malfunction.
  - Mounting/removing the module to/from the base unit
  - Inserting/removing the extended SRAM cassette or battery-less option cassette to/from the CPU module
  - Mounting/removing the terminal block to/from the module
  - Connecting/disconnecting the extension cable to/from the base unit
- After the first use of the product, do not insert/remove the SD memory card to/from the CPU module more than 500 times. Exceeding the limit may cause malfunction.
- Do not touch the metal terminals on the back side of the SD memory card. Doing so may cause malfunction or failure of the module.
- Do not touch the integrated circuits on the circuit board of an extended SRAM cassette or a battery-less option cassette. Doing so may cause malfunction or failure of the module.

## [Startup and Maintenance Precautions]

### ⚠CAUTION

- Do not drop or apply shock to the battery to be installed in the module. Doing so may damage the battery, causing the battery fluid to leak inside the battery. If the battery is dropped or any shock is applied to it, dispose of it without using.
- Startup and maintenance of a control panel must be performed by qualified maintenance personnel with knowledge of protection against electric shock. Lock the control panel so that only qualified maintenance personnel can operate it.
- Before handling the module, touch a conducting object such as a grounded metal to discharge the static electricity from the human body. Wearing a grounded antistatic wrist strap is recommended. Failure to discharge the static electricity may cause the module to fail or malfunction.
- After unpacking, eliminate static electricity from the module to prevent electrostatic discharge from affecting the module. If an electrostatically charged module comes in contact with a grounded metal object, a sudden electrostatic discharge of the module may cause failure.
  For details on how to eliminate static electricity from the module, refer to the following.
  Safety Precautions For Static Electricity Before Using MELSEC MX Controller MX-R Model and MELSEC iQ-R Series Models (FA-A-0368)
- Use a clean and dry cloth to wipe off dirt on the module.

## [Operating Precautions]

### ⚠CAUTION

- When changing data and operating status, and modifying program of the running programmable controller from an external device such as a personal computer connected to an intelligent function module, read relevant manuals carefully and ensure the safety before operation. Incorrect change or modification may cause system malfunction, damage to the machines, or accidents.
- Do not power off the programmable controller or reset the CPU module while the setting values in the buffer memory are being written to the flash ROM in the module. Doing so will make the data in the flash ROM and SD memory card undefined. The values need to be set in the buffer memory and written to the flash ROM and SD memory card again. Doing so can cause malfunction or failure of the module.

## [Disposal Precautions]

### ⚠CAUTION

- When disposing of this product, treat it as industrial waste.
- When disposing of batteries, separate them from other wastes according to the local regulations. For details on battery regulations in EU member states, refer to the MELSEC iQ-R Module Configuration Manual.

## [Transportation Precautions]

### ⚠CAUTION

- When transporting lithium batteries, follow the transportation regulations. For details on the regulated models, refer to the MELSEC iQ-R Module Configuration Manual.
- The halogens (such as fluorine, chlorine, bromine, and iodine), which are contained in a fumigant used for disinfection and pest control of wood packaging materials, may cause failure of the product. Prevent the entry of fumigant residues into the product or consider other methods (such as heat treatment) instead of fumigation. The disinfection and pest control measures must be applied to unprocessed raw wood.

# CONDITIONS OF USE FOR THE PRODUCT

(1) MELSEC programmable controller ("the PRODUCT") shall be used in conditions;
i) where any problem, fault or failure occurring in the PRODUCT, if any, shall not lead to any major or serious accident; and
ii) where the backup and fail-safe function are systematically or automatically provided outside of the PRODUCT for the case of any problem, fault or failure occurring in the PRODUCT.

(2) The PRODUCT has been designed and manufactured for the purpose of being used in general industries.
MITSUBISHI ELECTRIC SHALL HAVE NO RESPONSIBILITY OR LIABILITY (INCLUDING, BUT NOT LIMITED TO ANY AND ALL RESPONSIBILITY OR LIABILITY BASED ON CONTRACT, WARRANTY, TORT, PRODUCT LIABILITY) FOR ANY INJURY OR DEATH TO PERSONS OR LOSS OR DAMAGE TO PROPERTY CAUSED BY the PRODUCT THAT ARE OPERATED OR USED IN APPLICATION NOT INTENDED OR EXCLUDED BY INSTRUCTIONS, PRECAUTIONS, OR WARNING CONTAINED IN MITSUBISHI ELECTRIC USER'S, INSTRUCTION AND/OR SAFETY MANUALS, TECHNICAL BULLETINS AND GUIDELINES FOR the PRODUCT.
("Prohibited Application")
Prohibited Applications include, but not limited to, the use of the PRODUCT in;
- Nuclear Power Plants and any other power plants operated by Power companies, and/or any other cases in which the public could be affected if any problem or fault occurs in the PRODUCT.
- Railway companies or Public service purposes, and/or any other cases in which establishment of a special quality assurance system is required by the Purchaser or End User.
- Aircraft or Aerospace, Medical applications, Train equipment, transport equipment such as Elevator and Escalator, Incineration and Fuel devices, Vehicles, Manned transportation, Equipment for Recreation and Amusement, and Safety devices, handling of Nuclear or Hazardous Materials or Chemicals, Mining and Drilling, and/or other applications where there is a significant risk of injury to the public or property.

Notwithstanding the above restrictions, Mitsubishi Electric may in its sole discretion, authorize use of the PRODUCT in one or more of the Prohibited Applications, provided that the usage of the PRODUCT is limited only for the specific applications agreed to by Mitsubishi Electric and provided further that no special quality assurance or fail-safe, redundant or other safety features which exceed the general specifications of the PRODUCTs are required. For details, please contact the Mitsubishi Electric representative in your region.

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

# INTRODUCTION

Thank you for purchasing the Mitsubishi Electric MELSEC iQ-R series programmable controllers.
This manual describes the functions, programming, and troubleshooting of the relevant products listed below.
Before using this product, please read this manual and the relevant manuals carefully and develop familiarity with the functions and performance of the MELSEC iQ-R series programmable controller to handle the product correctly.
When applying the program examples provided in this manual to an actual system, ensure the applicability and confirm that it will not cause system control problems.
Note that the menu names and operating procedures may differ depending on an operating system in use and its version. When reading this manual, replace the names and procedures with the applicable ones as necessary.
Please make sure that the end users read this manual.

## Relevant products

RJ71EN71, CPU module

**Point**

Unless otherwise specified, the buffer memory addresses in this manual are for when the P1 connecter of the RJ71EN71 or RnENCPU is used.
Check the corresponding buffer memory addresses in the list and use the correct addresses when using the following: (☞ Page 486 Buffer Memory)

- CPU module (built-in Ethernet port part)
- P2 connector of the RJ71EN71
- RJ71EN71 (network type: Q-compatible Ethernet)

# CONTENTS

SAFETY PRECAUTIONS . . . . . 1
CONDITIONS OF USE FOR THE PRODUCT . . . . . 10
INTRODUCTION . . . . . 11
RELEVANT MANUALS . . . . . 17
TERMS . . . . . 18
GENERIC TERMS AND ABBREVIATIONS . . . . . 20

**CHAPTER 1 FUNCTIONS 21**

**1.1 Connection with MELSOFT Product and GOT . . . . . 21**
Connection via a hub . . . . . 22
Direct connection . . . . . 28
**1.2 SLMP Communications . . . . . 31**
Applications . . . . . 31
Communication structure . . . . . 32
Data communication procedures . . . . . 32
List of valid commands . . . . . 34
Dedicated instructions . . . . . 35
**1.3 Communications Using the Predefined Protocol . . . . . 36**
Applicable connections . . . . . 36
Data communication procedures . . . . . 37
Protocol communication type . . . . . 44
Packet elements . . . . . 45
Execution conditions of predefined protocol communications . . . . . 52
Example of predefined protocol communications . . . . . 54
**1.4 Socket Communications . . . . . 63**
Setting procedure . . . . . 63
Applicable dedicated instructions . . . . . 64
Applicable connections . . . . . 64
Communication structure . . . . . 65
Communications using TCP/IP . . . . . 66
Communications using UDP/IP . . . . . 69
Broadcast communications . . . . . 70
Precautions . . . . . 71
**1.5 Communications Using the Fixed Buffer . . . . . 72**
Differences between the "Procedure Exist" and "No Procedure" control methods . . . . . 72
Setting procedure . . . . . 72
Applicable dedicated instructions . . . . . 73
Applicable connections . . . . . 73
Communication structure . . . . . 73
Send procedure . . . . . 75
Receive procedure . . . . . 77
Pairing open . . . . . 81
Broadcast communications . . . . . 82
Precautions . . . . . 82
Data format . . . . . 83
Example of communications using the fixed buffer . . . . . 87
**1.6 Communications Using the Random Access Buffer . . . . . 97**
Setting procedure . . . . . 97

Communication structure . . . 98
Procedure for reading from external device . . . 98
Procedure for writing from external device . . . 99
Physical address and logical address of random access buffer . . . 100
Precautions . . . 100
Data format . . . 100
Example of communications with random access buffer . . . 109
**1.7 Communications Using MODBUS/TCP . . . 110**
MODBUS/TCP master function . . . 110
MODBUS/TCP slave function . . . 110
**1.8 Link Dedicated Instruction Communication . . . 118**
Data communication procedures . . . 118
**1.9 File Transfer Function (FTP server) . . . 119**
Data communication procedures . . . 120
Files that can be transferred with FTP . . . 121
FTP command . . . 122
Precautions . . . 135
**1.10 File Transfer Function (FTP Client) . . . 138**
Transferable files . . . 139
Procedure for executing the file transfer function (FTP client) . . . 140
Precautions . . . 142
**1.11 Time Setting Function (SNTP Client) . . . 144**
**1.12 Web Server Function . . . 146**
JavaScript objects . . . 147
CGI object . . . 161
Error message . . . 175
**1.13 Security . . . 177**
IP filter . . . 177
Remote password . . . 178
**1.14 Simple CPU Communication Function (RJ71EN71, RnENCPU (Network Part)) . . . 185**
Setting procedure . . . 187
Checking the simple CPU communication status . . . 215
Communication example . . . 218
Precautions . . . 224
**1.15 Simple CPU Communication Function (CPU Module (Built-in Ethernet Port Part)) . . . 228**
Setting procedure . . . 229
Checking the simple CPU communication status . . . 249
Simple CPU communication in a redundant system . . . 251
Precautions . . . 256
**1.16 Simple Device Communication Function . . . 259**
Setting procedure . . . 259
Checking the simple device communication status . . . 262
Communication example . . . 263
Operation . . . 268
Precautions . . . 289
**1.17 IP Address Change Function . . . 293**
IP address of the Ethernet-equipped module . . . 294
Usage methods . . . 296
Checking the IP address . . . 299
Checking the operating status . . . 299
Precautions . . . 300

**1.18 Redundant System Function . . . 302**
System configuration . . . 302
System switching request . . . 303
System switching cause . . . 304
Redundant group setting . . . 306
Communication path bypass function . . . 308
Both systems identical IP address setting function . . . 309
Functions restricted in a redundant system . . . 312
Setting example . . . 318
**1.19 Discard Received Data at CPU STOP . . . 325**

**CHAPTER 2 PARAMETER SETTINGS 326**

**2.1 Setting Parameters . . . 326**
**2.2 Basic Settings . . . 326**
Own Node Settings . . . 327
External Device Configuration . . . 329
Transmission Port Settings . . . 332
**2.3 Application Settings . . . 333**
Frame Settings . . . 336
Communication Speed . . . 337
FTP Server Settings . . . 338
FTP Client Settings . . . 339
DNS Settings . . . 339
MODBUS/TCP Settings . . . 340
Simple CPU communication setting (RJ71EN71, RnENCPU (network part)) . . . 342
Simple CPU communication setting (CPU module (built-in Ethernet port part)) . . . 345
Simple Device Communication Setting . . . 347
Time Setting . . . 400
Timer Settings for Data Communication . . . 401
Security . . . 404
Gateway Parameter Settings . . . 404
Network/Station No. <-> IP information setting . . . 407
Interrupt Settings . . . 414
IP Packet Transfer Setting . . . 414
Network Dynamic Routing . . . 415
Module Operation Mode . . . 415
Redundant System Settings . . . 416
iQ Sensor Solution Settings . . . 419
SLMPSND Instruction Settings . . . 419

**CHAPTER 3 TROUBLESHOOTING 420**

**3.1 Checking with LED . . . 420**
**3.2 Checking the Module Status . . . 422**
Module communication test . . . 424
**3.3 Checking the Network Status . . . 426**
Ethernet diagnostics . . . 426
Simple CPU communication diagnostics . . . 434
Error log area . . . 436
**3.4 Troubleshooting by Symptom . . . 437**
Communications cannot be performed with the external device . . . 437

A message sent from the external device cannot be received frequently. . . . 439
Direct connection to the engineering tool is not possible. . . . 439
CPU module search on the network cannot be performed. . . . 440
Communications using the SLMP cannot be performed. . . . 440
Communications using the predefined protocol cannot be performed. . . . 441
Communications using the random access buffer cannot be performed. . . . 442
Socket communications/communications using the fixed buffer cannot be performed. . . . 443
Communications using MODBUS/TCP cannot be performed. . . . 444
A dedicated instruction is not completed. . . . 445
Communications using a link dedicated instruction cannot be performed. . . . 445
Access using the file transfer function (FTP server) cannot be performed. . . . 445
Access using the file transfer function (FTP client) cannot be performed. . . . 446
The time setting function (SNTP client) cannot be performed normally. . . . 446
The access cannot be allowed/denied correctly by the IP filter. . . . 446
Remote password does not work. . . . 446
Communications using the built-in Ethernet port are slow. . . . 447
When the simple CPU communication function is used . . . . 448
When the simple device communication function is used . . . . 450
Data communications cannot be performed with the device supporting iQ Sensor Solution. . . . 450
When the redundant system function is used . . . . 451
**3.5 List of Error Codes . . . . 453**
**3.6 List of Parameter Numbers. . . . 475**
**3.7 Event List . . . . 477**
**3.8 End Codes Returned to an External Device During Data Communications . . . . 479**

## APPENDICES 481

**Appendix 1 Module Label . . . . 481**
**Appendix 2 I/O Signals . . . . 482**
List of I/O signals . . . . 482
Details of I/O signals . . . . 484
**Appendix 3 Buffer Memory . . . . 486**
List of buffer memory addresses . . . . 486
Details of buffer memory addresses (RJ71EN71, RnENCPU (network part)). . . . 505
Details of buffer memory addresses (CPU module (built-in Ethernet port part)) . . . . 528
**Appendix 4 Dedicated Instruction . . . . 536**
Open/close processing instructions. . . . 537
Instruction for SLMP communications . . . . 537
Instructions for predefined protocol communications . . . . 537
Socket communications instructions . . . . 538
Instructions for communications using the fixed buffer . . . . 538
Link dedicated instructions . . . . 539
File transfer function instruction. . . . 539
Other dedicated instructions . . . . 540
**Appendix 5 Initial Processing . . . . 541**
Initial processing . . . . 541
Re-initial processing . . . . 542
**Appendix 6 TCP/IP Communications, UDP/IP Communications . . . . 546**
TCP/IP communications . . . . 546
UDP/IP communications . . . . 551
**Appendix 7 Communications with Different Networks. . . . 553**

CONTENTS

**Appendix 8 Processing Time** . . . . . **556**
Performance list of simple CPU communication function . . . . . 560
**Appendix 9 Mounting the Module with a Remote Head Module** . . . . . **562**
Restrictions on functions and specifications . . . . . 562
**Appendix 10 Using the Module in the Redundant System with Redundant Extension Base Unit** . . . . . **563**
Restrictions on functions and specifications . . . . . 563
Precautions . . . . . 564
Example of communications using the fixed buffer . . . . . 565
**Appendix 11 Port Numbers Used by Ethernet-equipped Module** . . . . . **571**
**Appendix 12 Operation Image and Data Structure of Predefined Protocol** . . . . . **573**
Operation image of each communication type of protocol . . . . . 573
Verification operation of receive packet . . . . . 578
Example of packet element data . . . . . 579
**Appendix 13 Example of External Device Program** . . . . . **582**
**Appendix 14 How to Turn Off ERR LED** . . . . . **583**
**Appendix 15 Software Licenses and Copyrights** . . . . . **584**
**Appendix 16 Added and Enhanced Functions** . . . . . **586**
RJ71EN71, RnENCPU (network part) . . . . . 586
CPU module (built-in Ethernet port part) . . . . . 587

**INDEX** **588**

REVISIONS . . . . . 590
WARRANTY . . . . . 593
INFORMATION AND SERVICES . . . . . 594
TRADEMARKS . . . . . 594

# RELEVANT MANUALS

The following manuals are relevant to this product.

| Manual name [manual number] | Description |
|---|---|
| MELSEC iQ-R Ethernet User's Manual (Application) [SH-081257ENG] (this manual) | Functions, parameter settings, programming, troubleshooting, I/O signals, and buffer memory of Ethernet |
| MELSEC iQ-R Module Configuration Manual [SH-081262ENG] | The combination of the MELSEC iQ-R series modules, common information on the installation/wiring in the system, and specifications of the power supply module, base unit, SD memory card, and battery |
| MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup) [SH-081256ENG] | Specifications, procedures before operation, system configuration, wiring, and communication examples of Ethernet, CC-Link IE Controller Network, and CC-Link IE Field Network |
| MELSEC iQ-R CPU Module User's Manual (Startup) [SH-081263ENG] | Performance specifications, procedures before operation, and troubleshooting of the CPU module |
| MELSEC iQ-R CPU Module User's Manual (Application) [SH-081264ENG] | Memory, functions, devices, and parameters of the CPU module |
| MELSEC iQ-R Programmable Controller CPU Module User's Manual [SH-082488ENG] | Procedures before operation, specifications, devices, memory, functions, parameters, and troubleshooting of the programmable controller CPU |
| MELSEC iQ-R Process CPU Module User's Manual [SH-082493ENG] | Procedures before operation, specifications, devices, memory, functions, parameters, and troubleshooting of the Process CPU |
| MELSEC iQ-R SIL2 Process CPU Module User's Manual [SH-082503ENG] | Procedures before operation, specifications, devices, memory, functions, parameters, and troubleshooting of the SIL2 Process CPU |
| MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks) [SH-081266ENG] | Instructions for the CPU module and standard functions/function blocks |
| MELSEC iQ-R Programming Manual (Module Dedicated Instructions) [SH-081976ENG] | Dedicated instructions for the intelligent function modules |
| MELSEC iQ-R Ethernet, CC-Link IE, and MELSECNET/H Function Block Reference [BCN-P5999-0381] | Specifications of the following MELSEC iQ-R series module FBs: Ethernet-equipped module FBs, CC-Link IE TSN module FBs, CC-Link IE Controller Network module FBs, CC-Link IE Field Network module FBs, and MELSECNET/H network module FBs |
| GX Works3 Operating Manual [SH-081215ENG] | System configuration, parameter settings, and online operations of GX Works3 |
| MELSEC iQ-R MODBUS and MODBUS/TCP Reference Manual [BCN-P5999-1060] | The protocol used for data reading or writing from an external device to the Ethernet-equipped module |
| SLMP Reference Manual [SH-080956ENG] | The protocol used to access an SLMP-compatible device from an external device (such as a personal computer or HMI (Human Machine Interface)) or an SLMP-compatible module (such as the Ethernet-equipped module or modules on CC-Link IE TSN). |
| iQ Sensor Solution Reference Manual [SH-081133ENG] | Operation methods of the online functions for iQ Sensor Solution |
| MELSEC iQ-R Simple Device Communication Library Reference Manual [SH-082515ENG] | Simple device communication library |

# TERMS

Unless otherwise specified, this manual uses the following terms.

| Term | Description |
|---|---|
| Buffer memory | Memory in an intelligent function module to store data such as setting values and monitor values. For CPU modules, it refers to memory to store data such as setting values and monitor values of the Ethernet function, or data used for data communication of the multiple CPU system function. |
| Control CPU | A CPU module that controls connected I/O modules and intelligent function modules.<br>In a multiple CPU system, a control CPU can be set for each module. |
| Control system | A system that controls a redundant system and performs network communications in a redundant system |
| Control System IP Address | An IP address common to system A and system B in a redundant system. (☞ Page 309 Both systems identical IP address setting function) |
| CPU module (built-in Ethernet port part) | A built-in Ethernet port part of a CPU module (CPU part for an RnENCPU). |
| Device | A memory of a CPU module to store data. Devices such as X, Y, M, D, and others are provided depending on the intended use. |
| Engineering tool | A tool used for setting up programmable controllers, programming, debugging, and maintenance |
| Global label | A label that is valid for all the program data when multiple program data are created in the project.<br>There are two types of global label: a module specific label (module label), which is generated automatically by GX Works3, and an optional label, which can be created for any specified device. |
| Intelligent function module | A module that has functions other than an input or output, such as an A/D converter module and D/A converter module |
| iQSS | iQ Sensor Solution. Cooperation of sensors, programmable controllers, HMI (Human Machine Interface), and engineering tool to facilitate the start-up or maintenance of sensors. (📖 iQ Sensor Solution Reference Manual) |
| MODBUS/TCP | A protocol for using MODBUS protocol messages on a TCP/IP network |
| Module label | A label that represents one of memory areas (I/O signals and buffer memory areas) specific to each module in a given character string.<br>For the module used, GX Works3 automatically generates this label, which can be used as a global label. |
| New control system | A system that has switched to control system from standby system after system switching |
| OPS | An operation station using MELSOFT and EZSocket partner products. |
| Predefined protocol support function | A function of GX Works3. This function sets protocols appropriate to each external device and reads/writes protocol setting data. |
| Process CPU | A CPU module that performs process control and sequence control. Process control function blocks and the online module change function can be executed.<br>This module is also used with a redundant function module as a pair and configures a redundant system.<br>The Process CPU models include the R08PCPU, R16PCPU, R32PCPU, and R120PCPU. |
| Process CPU (redundant mode) | A Process CPU operating in redundant mode.<br>A redundant system is configured with this CPU module. Process control function blocks and the online module change function can be used even in this mode. |
| Redundant function module | A module to configure a redundant system by using it in combination with a process CPU (redundant mode).<br>The SIL2 function module model name is R6RFM. |
| Redundant system | A system consisting of two systems that have same configuration (CPU module, power supply module, network module, and other modules). Even after an error occurs in one of the two system, the other system takes over the control of the entire system. |
| Redundant system with redundant extension base unit | A redundant system that is configured using extension base unit(s) |
| Relay station | A station that relays data link to other station with mounting more than one network modules on one programmable controller. |
| RnENCPU (CPU part) | A module on the left-hand side of the RnENCPU (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup)) |
| RnENCPU (network part) | A module on the right-hand side of the RnENCPU (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup)) |
| Routing | A process of selecting paths for communication with other networks. There are two types of routing: dynamic routing that auto-selects the communication routes, and static routing where communication routes are arbitrarily set. |
| SIL2 Process CPU | This module is used with a SIL2 function module as a pair, and performs both standard control and safety control.<br>This module is also used with a redundant function module as a pair and configures a redundant system.<br>The SIL2 Process CPU models include the R08PSFCPU, R16PSFCPU, R32PSFCPU, and R120PSFCPU. |
| SLMP | A SeamLess Message Protocol.<br>This protocol is used to access an SLMP-compatible device or a programmable controller connected to an SLMP-compatible device from an external device. |
| Standby system | A backup system in a redundant system |

| Term | Description |
|---|---|
| Subnet mask | A number used to logically divide one network into multiple subnetworks and manage them easily. The following Ethernet network systems can be configured:<br>• A small-scale Ethernet network system in which multiple network devices are connected.<br>• A medium- or large-scale network system in which multiple small-scale network systems are connected via routers or other network communication devices. |
| System A | A system that is set as system A to distinguish two systems in a redundant system.<br>It is used to distinguish connected two systems (system A and system B). |
| System B | A system that is set as system B to distinguish two systems in a redundant system. It is used to distinguish connected two systems (system A and system B). |
| Transient transmission group number | Number that is assigned for transient transmission to any given stations.<br>By specifying a group of stations as transient transmission target, data can be sent to the stations of the same group number. |

# GENERIC TERMS AND ABBREVIATIONS

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

| Generic term/abbreviation | Description |
|---|---|
| ARP | An abbreviation for Address Resolution Protocol. This protocol is used to obtain the MAC address of Ethernet from an IP address. |
| BUFRCV instruction | A generic term for the dedicated instructions GP.BUFRCV and ZP.BUFRCV |
| BUFRCVS instruction | A generic term for the dedicated instructions G.BUFRCVS and Z.BUFRCVS |
| BUFSND instruction | A generic term for the dedicated instructions GP.BUFSND and ZP.BUFSND |
| CLOSE instruction | A generic term for the dedicated instructions GP.CLOSE, ZP.CLOSE, GP.CONCLOSE, and SP.SOCCLOSE |
| ECPRTCL instruction | A generic term for the dedicated instructions GP.ECPRTCL and SP.ECPRTCL |
| ERRCLEAR instruction | A generic term for the dedicated instructions GP.ERRCLEAR and ZP.ERRCLEAR |
| ERRRD instruction | A generic term for the dedicated instructions GP.ERRRD and ZP.ERRRD |
| Ethernet device | A generic term for the devices supporting IP communication (such as a personal computer, a vision sensor, and a bar code reader) |
| Ethernet-equipped module | A generic term for the following modules:<br>• RJ71EN71 (when the Ethernet function is used)<br>• CPU module (when the Ethernet function is used) |
| ICMP | An abbreviation for Internet Control Message Protocol. This protocol is used to exchange messages of errors in an IP network or other information related to an Ethernet network. |
| MELSECNET/10 | An abbreviation for the MELSECNET/10 network system |
| MELSECNET/H | An abbreviation for the MELSECNET/H network system |
| Network module | A generic term for the following modules:<br>• Ethernet interface module<br>• A module on CC-Link IE TSN (the RJ71GN11-T2 and a module on a remote station)<br>• CC-Link IE Controller Network module<br>• A module on CC-Link IE Field Network (a master/local module, and a module on a remote I/O station, a remote device station, and an intelligent device station)<br>• MELSECNET/H network module<br>• MELSECNET/10 network module<br>• RnENCPU (network part) |
| OPEN instruction | A generic term for the dedicated instructions GP.OPEN, ZP.OPEN, GP.CONOPEN, and SP.SOCOPEN |
| Programmable controller CPU | A generic term for the R00CPU, R01CPU, R02CPU, R04CPU, R04ENCPU, R08CPU, R08ENCPU, R16CPU, R16ENCPU, R32CPU, R32ENCPU, R120CPU, R120ENCPU |
| READ instruction | A generic term for the dedicated instructions JP.READ and GP.READ |
| RECV instruction | A generic term for the dedicated instructions JP.RECV and GP.RECV |
| RECVS instruction | A generic term for the dedicated instructions G.RECVS and Z.RECVS |
| Remote head module | An abbreviation for the RJ72GF15-T2 CC-Link IE Field Network remote head module |
| REQ instruction | A generic term for the dedicated instructions J.REQ, JP.REQ, G.REQ, and GP.REQ |
| RnENCPU | A generic term for the R04ENCPU, R08ENCPU, R16ENCPU, R32ENCPU, and R120ENCPU |
| SEND instruction | A generic term for the dedicated instructions JP.SEND and GP.SEND |
| SOCRCV instruction | A generic term for the dedicated instructions GP.SOCRCV and SP.SOCRCV |
| SOCRCVS instruction | A generic term for the dedicated instructions G.SOCRCV and S.SOCRCV |
| SOCSND instruction | A generic term for the dedicated instructions GP.SOCSND and SP.SOCSND |
| SREAD instruction | A generic term for the dedicated instructions JP.SREAD and GP.SREAD |
| SWRITE instruction | A generic term for the dedicated instructions JP.SWRITE and GP.SWRITE |
| UINI instruction | A generic term for the dedicated instructions G.UINI, GP.UINI, Z.UINI, and ZP.UINI |
| WRITE instruction | A generic term for the dedicated instructions JP.WRITE and GP.WRITE |
| ZNRD instruction | A generic term for the dedicated instructions J.ZNRD and JP.ZNRD |
| ZNWR instruction | A generic term for the dedicated instructions J.ZNWR and JP.ZNWR |

# 1 FUNCTIONS



## 1.1 Connection with MELSOFT Product and GOT

Programming and monitoring of the programmable controller with the engineering tool, and monitoring and testing of the programmable controller from the GOT can be performed via Ethernet. This function enables remote operations using Ethernet's long-distance connection and high-speed communication.

The section describes the methods of connecting the Ethernet-equipped module, MELSOFT product (such as engineering tool and MX Component), and GOT.

○: Connection available, ×: Connection not available

| Connection method | Purpose | Availability | | | |
|---|---|---|---|---|---|
| | | MELSOFT products | | GOT | |
| | | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
| Connection via a hub (Connection by specifying the IP address) | • To connect to an Ethernet-equipped module that has no network number and station number<br>• To connect multiple MELSOFT products | ○ | ○ | × | ○ |
| Connection via a hub (Connection by specifying the network number and station number) | • To connect by using network number and station number<br>• To connect multiple MELSOFT products and GOTs | ○ | ○*2 | ○ | ○*2 |
| Direct connection (Connection without specifying the IP address, network number, or station number)*1 | • To connect without hub using one Ethernet cable for one-on-one communication with the external device<br>• To connect to an Ethernet-equipped module whose IP address in unknown | ○ | ○ | × | × |

*1 This connection method is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".

*2 Before the connection, check the firmware version of the CPU module. (☞ Page 586 Added and Enhanced Functions)

**Point**

For the procedures to connect the Ethernet-equipped module and GOT, refer to the following.

Manual for the GOT used

**Restriction**

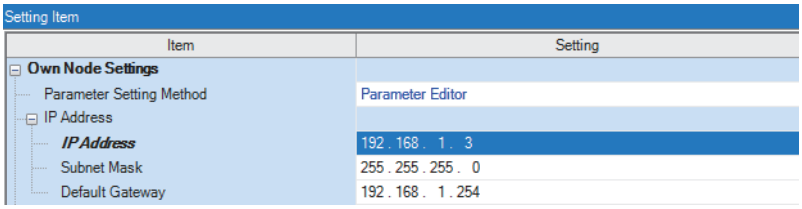
If multiple RJ71EN71s with the same network number are installed to one base unit (main base unit or extension base unit), the Ethernet-equipped module cannot connect to a MELSOFT product or GOT via an RJ71EN71. When the RJ71EN71 with the same network number as a relay RJ71EN71 is set in a relay network or target network, the Ethernet-equipped module cannot connect to other stations and other networks.

# Connection via a hub

## Setting procedure

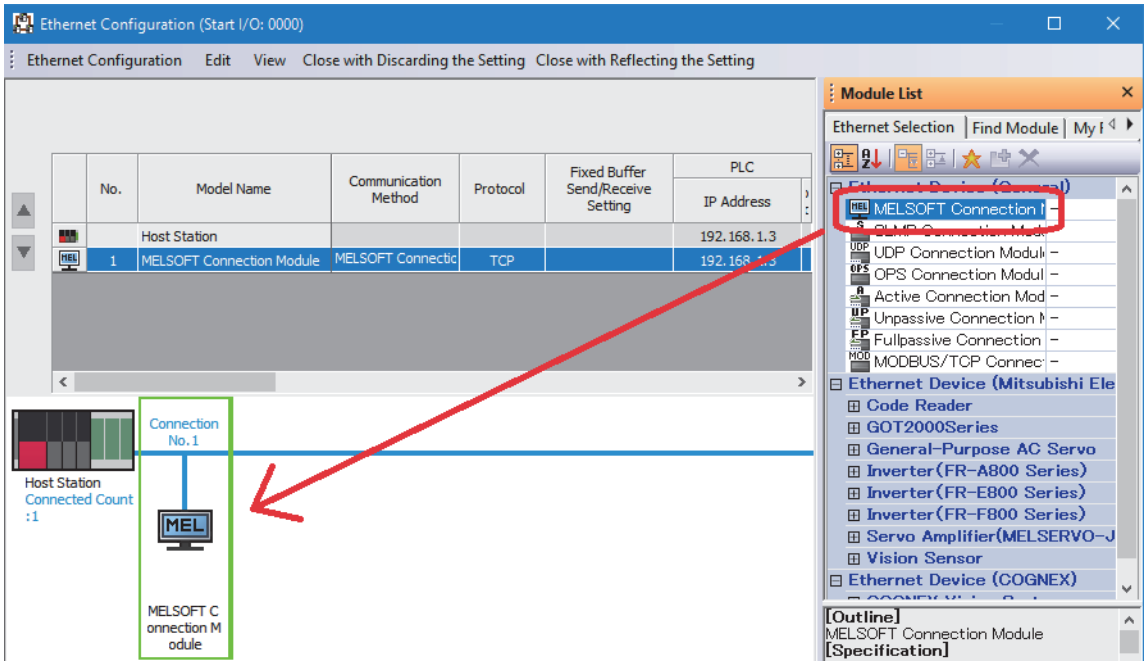
### ■Setting in the Ethernet-equipped module side

***1.*** Set the IP address of the Ethernet-equipped module in "Own Node Settings" under "Basic Settings".




***2.*** When connecting by specifying the network number and station number, set the network number and station number in "Own Node Settings" under "Basic Settings".

***3.*** When using a TCP/IP connection, add as many MELSOFT connection modules as the number of devices to be connected in "External Device Configuration" under "Basic settings". When using a UDP/IP connection, up to the maximum number of connectable devices can be connected without setting "External Device Configuration".*1

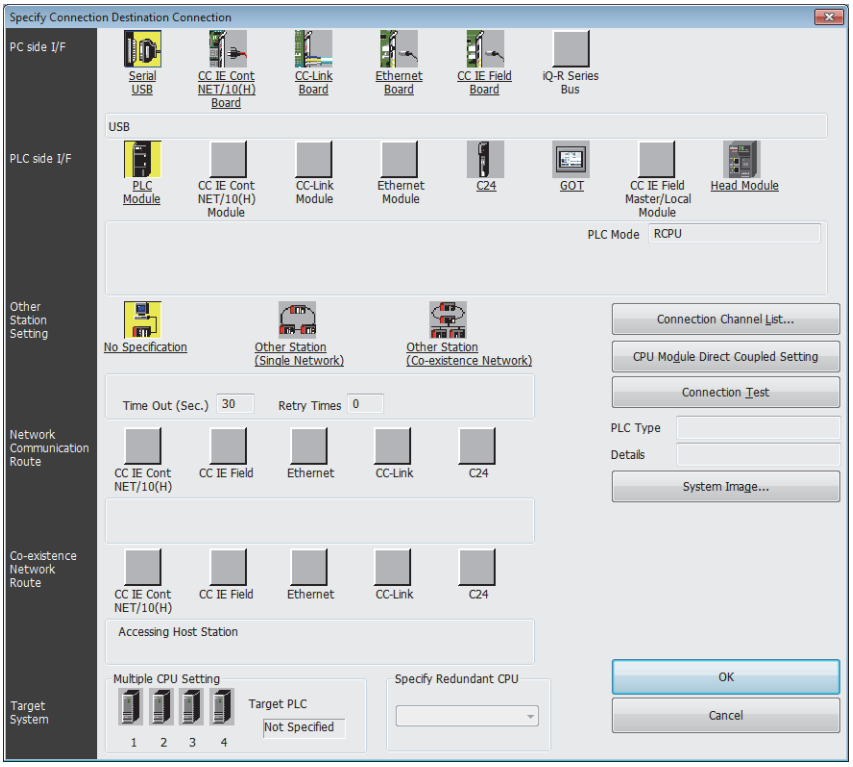


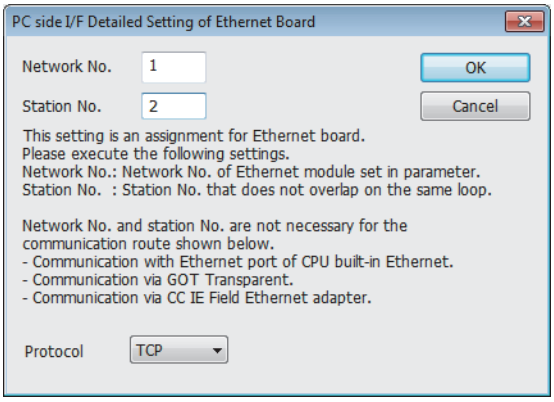

*1 Up to 16 CPU modules (built-in Ethernet port part) are supported. Up to 64 RJ71EN71 and RnENCPU (network part) are supported.

## ■Settings on the engineering tool side

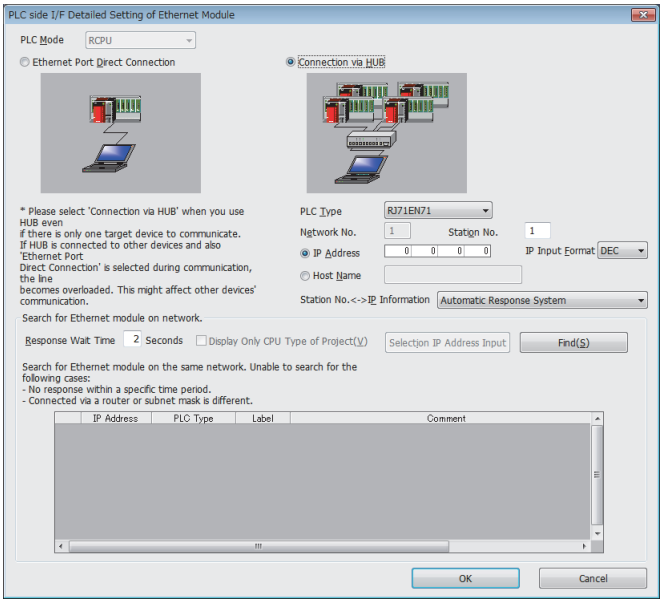
Set in the "Specify Connection Destination Connection" window.

[Online] ⇨ [Current Connection Destination]




1. Set "PC side I/F" to "Ethernet Board".
2. Double-click "Ethernet Board", and open the "PC side I/F Detailed Setting of Ethernet Board" window.
3. Set the network number, station number, and protocol of the personal computer. (Set the network number and protocol according to the settings for the Ethernet-equipped module. Set the station number so that it is not the same as a station number assigned to other Ethernet devices.)*1
4. Set the "PLC side I/F" to the module to be connected.




***5.*** Double-click the icon set in step 4, and open the detailed setting window.

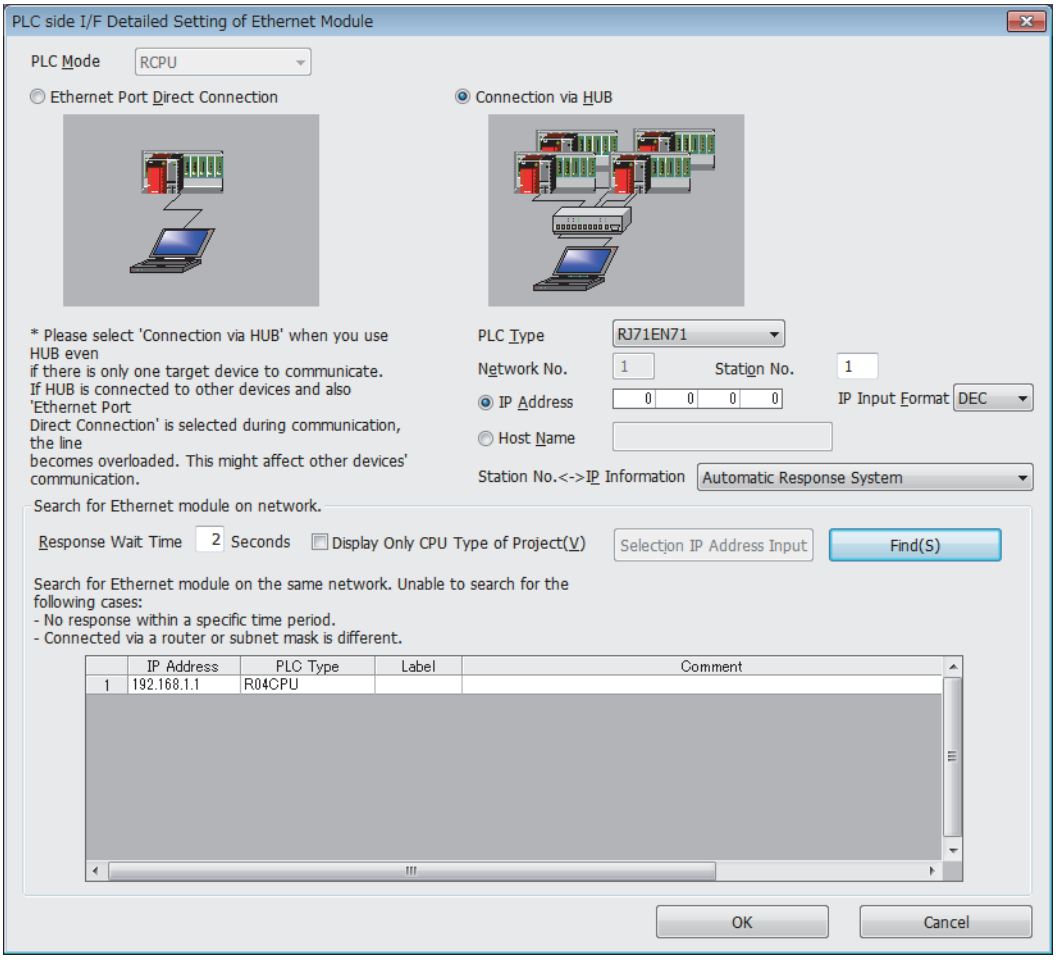
***6.*** Select "Connection via HUB" for the connection method, and enter the station number and IP address or host name for the Ethernet-equipped module. Select "RJ71EN71" to connect the RnENCPU (network part).

***7.*** Set "Other Station Setting" or "Network Communication Route" if necessary.

*1 The network number and station number do not need to be set when connecting with the CPU module (built-in Ethernet port part).

## Searching modules on the network

When connecting with a hub, a list of modules that can be searched for will appear by clicking the [Find] button on the detailed setting window.




### ■Search target modules

- The control CPU of the RJ71EN71 or the CPU module connected to the same hub as the engineering tool
- The control CPU of the RJ71EN71 or the CPU module connected to cascade-connected hub
- The remote head module that controls the RJ71EN71 connected to the same hub as the engineering tool
- The remote head module that controls the RJ71EN71 connected to a cascade-connected hub

**Point**

- By setting "Do Not Respond to CPU Module Search" in "Security" under "Application Settings" to "Do Not Respond", the modules will not be listed even if a search is performed.
- Only the MELSEC iQ-R Series Ethernet-equipped modules are searched.
- The RJ71EN71 in which the network type is set to "Q Compatible Ethernet" cannot be searched.

### ■When module does not appear after search

If a connected Ethernet-equipped module does not appear in the list after searching the modules on the network, check the following items.

- Search is not performed if it is disabled with the IP filter.
- Modules connected via a router cannot be searched.
- If the module is connected via a wireless LAN, packet loss can prevent the Ethernet communication from stabilizing, and may inhibit the module search.
- If there are modules with the same IP address in the list, review the IP address parameter settings for the Ethernet-equipped module.
- If the service processing load of the search-target CPU module is high, it may not be possible to search for the corresponding module. If the search cannot be performed, increase the response wait time in the search dialog, and perform the search again.

## Precautions

### ■Remote operation

If remote STOP or remote PAUSE has been executed from the engineering tool to the CPU module on another station when the CPU module (built-in Ethernet port part) and engineering tool are connected with an Ethernet cable, perform the following before turning the power off or resetting the CPU module.

- Remote RUN
- Remote RESET

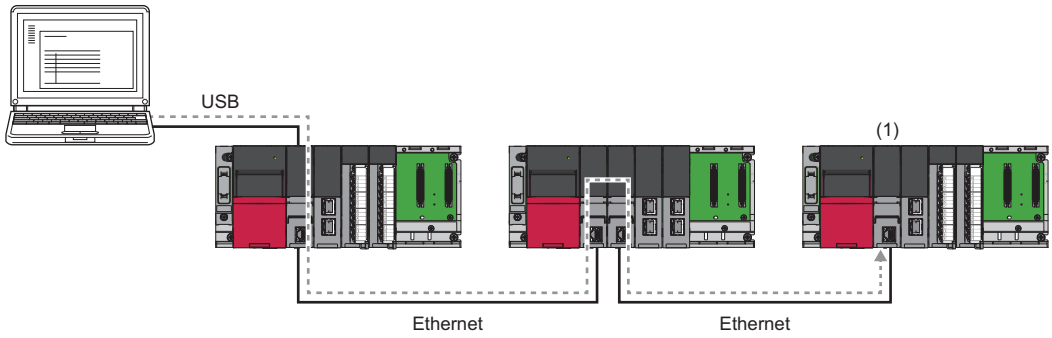
### ■Functions incompatible with connection via a hub

The following functions cannot be used for connection via a hub. To use the following functions, connect the CPU module (built-in Ethernet port part) directly or with a USB cable.

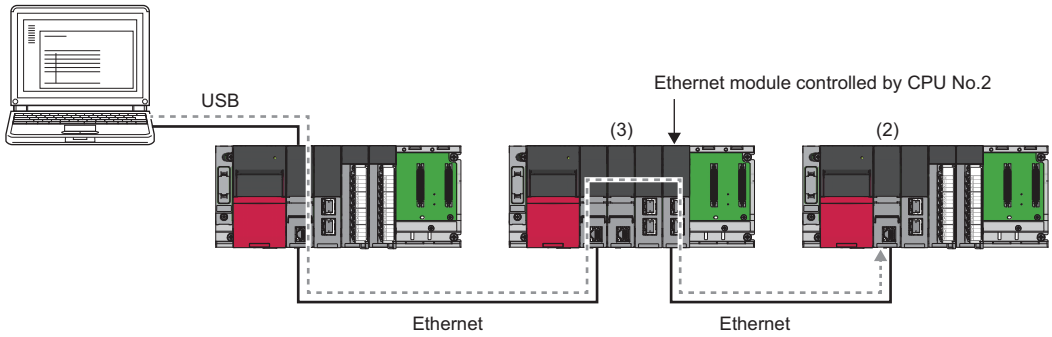
- Ethernet Diagnostics
- CC-Link IE Field Diagnostics
- CC-Link IE Control Diagnostics

## ■Communications by network number/station number by using the CPU module (built-in Ethernet port part)

- The UDP/IP protocol is used for the connection and data is always exchanged as binary codes.
- When accessing another station, set the network number for the CPU module (built-in Ethernet port part) so that it is unique among the network numbers of the other network. Also, set the station number that is unique among the station numbers set for the other modules on the same network.
- When configuring the target station or relay station as a multiple CPU system, ensure that the CPU modules listed below have the firmware supporting the communications by network number/station number.
  - Target station
  - CPU modules working as the relay path
  - CPU module that controls the Ethernet module working as the relay path




When communicating with CPU module shown as (1), ensure that all CPU modules have the firmware of the version supporting the communications by network number/station number. Also, set the network number and station number to all CPU modules.




When communicating with CPU module shown as (2), ensure also that all CPU modules have the firmware of the version supporting the communications by network number/station number. Note, however, that the communication is possible even when the network number and station number are not set to CPU module shown as (3) of the CPU No.2.

# Direct connection

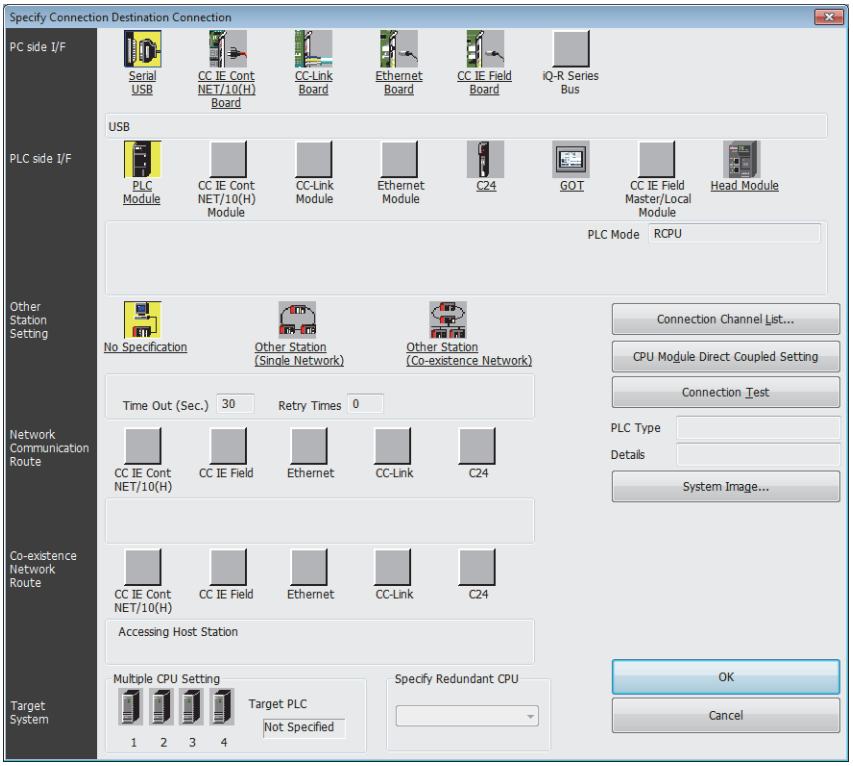
The Ethernet-equipped module and engineering tool can be directly connected with one Ethernet cable without using a hub. When direct connection is made, communication is possible without setting the IP address or host name in the "Specify Connection Destination Connection" window. (Communicate using broadcast communications)

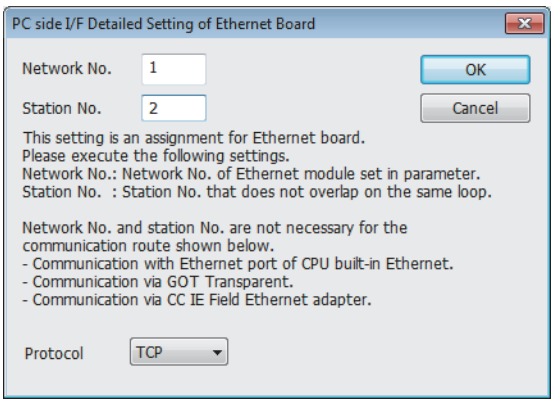
**Point**

- To prohibit direct connection with the Ethernet, set "Disable" for "Disable Direct Connection with MELSOFT" in "Security" under the "Application Settings".
- A direct connection is not possible when the RJ71EN71 network type is set to "Q Compatible Ethernet".

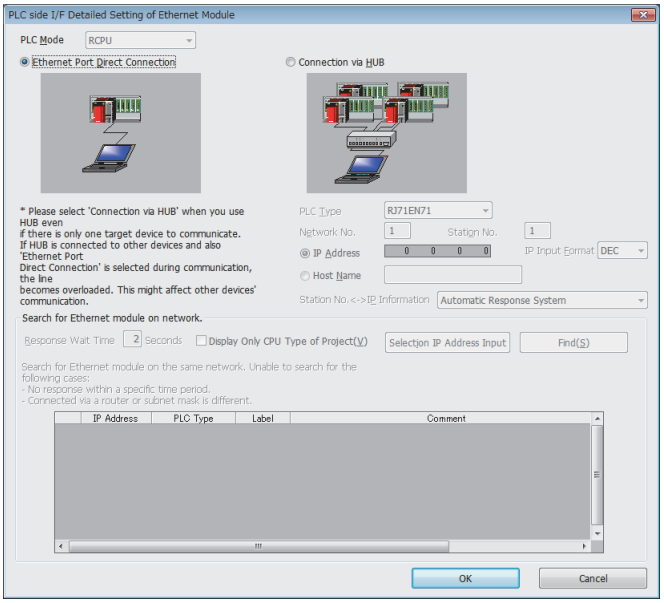
Set in the "Specify Connection Destination Connection" window.

[Online] ⇨ [Current Connection Destination]




1. Set "PC side I/F" to "Ethernet Board".
2. Double-click "Ethernet Board", and open the "PC side I/F Detailed Setting of Ethernet Board" window.
3. Set the network number, station number, and protocol of the personal computer. (Set the network number and protocol according to the settings for the Ethernet-equipped module. Set the station number so that it is not the same as a station number assigned to other Ethernet devices.)
4. Set the "PLC side I/F" to the module to be connected.




5. Double-click the icon set in step 4, and open the detailed setting window.
6. Select "Ethernet Port Direct Connection" for the connection method.

**Point**

When connecting directly with the CPU module (built-in Ethernet port part), setting is also possible by clicking the [CPU Module Direct Coupled Setting] button of the "Specify Connection Destination Connection" window.

## Precautions

### ■Connection with LAN line

Do not connect with a LAN line and set direct connection. Data will be sent to all external devices on the LAN line, so this setting will cause the line load to increase and will affect communication with other external devices.

### ■Connections that are not direct connections

- Do not use a configuration in which the Ethernet-equipped module and external device are connected with a hub. A direct connection is not established when the devices are connected with a hub.
- When creating a network connection on the personal computer side, communication with a direct connection is not possible if two or more Ethernet ports are set to "Enable". Review the personal computer settings so that only the Ethernet port for the direct connection is set to "Enable", and the other Ethernet ports are set to "Disable".

### ■Settings incompatible with direct connection

When using the RJ71EN71 or the RnENCPU (network part), a direct connection cannot be established if the following setting is made in the "Specify Connection Destination Connection" window of the engineering tool.

- When "Other Station (Co-existence Network)" is selected for "Other Station Setting"
- When "Other Station (Single Network)" is selected for "Other Station Setting", and "Other station in the same loop or access to multilevel system" is selected in the "Network Communication Route Detailed Setting of Ethernet" window

### ■Functions incompatible with direct connection

The following functions cannot be used when the RJ71EN71 or the RnENCPU (network part) is directly connected. To use the following functions, connect the CPU module (built-in Ethernet port part) directly or connect the CPU module with a USB cable.

- CC-Link IE Field Diagnostics
- CC-Link IE Control Diagnostics

## ■Conditions that cannot communicate with direct connection

Communication with a direct connection may be disabled if the following conditions apply. If connection is not possible, review the settings for the Ethernet-equipped module and personal computer.

- When all bits of the IP address for the Ethernet-equipped module that correspond with the 0 section of the subnet mask for the personal computer are on or off

Ex.

IP address for the Ethernet-equipped module: 64.64.255.255

IP address for the personal computer: 64.64.1.1

Subnet mask for the personal computer: 255.255.0.0

- When all bits of the IP address for the Ethernet-equipped module that correspond with the host address of each class in the IP address for the personal computer are on or off

Ex.

IP address for the personal computer: 192.168.0.1 ← 192.x.x.x, class C and the host address is the fourth octet.

Subnet mask for the personal computer: 255.0.0.0

IP address for the Ethernet-equipped module: 64.64.255.255 ← each bit turns on because of the fourth octet is 255

**Point**

The IP address for each class are as follow.

- Class A: 0.x.x.x to 127.x.x.x
- Class B: 128.x.x.x to 191.x.x.x
- Class C: 192.x.x.x to 223.x.x.x

The host address for each class is the 0 section shown below.

- Class A: 255.0.0.0
- Class B: 255.255.0.0
- Class C: 255.255.255.0

# 1.2 SLMP Communications

SLMP communications are available among devices that can receive/send messages with the SLMP control procedure. The Ethernet-equipped module processes and transfers data following instructions (command) from the external device, so the programmable controller only needs the open/close processing and does not require a program for data communication. For SLMP communications, refer to the following.

SLMP Reference Manual

## Applications

This section describes the applications of SLMP communications.

### Data read/write

Data read/write can be executed for the following data. With this, the external device can monitor the operation of the Ethernet-equipped module, analyze data, and control production.

- Device or global label of the CPU module connected with the RJ71EN71 (When the Ethernet function of the RJ71EN71 or the RnENCPU (network part) is used)
- Device or global label of the CPU module (When the Ethernet function of the CPU module (CPU part for the RnENCPU) is used)
- Buffer memory of the intelligent function module

### File read/write

Files such as parameter files stored in a CPU module can be read/written. Files in a CPU module can be managed on an external device.

### Remote control of a CPU module

A CPU module can be remotely controlled from the external device using remote operations.

### Remote password lock/unlock

The remote password can be locked and unlocked from the external device.

### Access to the programmable controller on another station via other network

The programmable controller on another station can be accessed from the external device via each network.
For details, refer to the following.

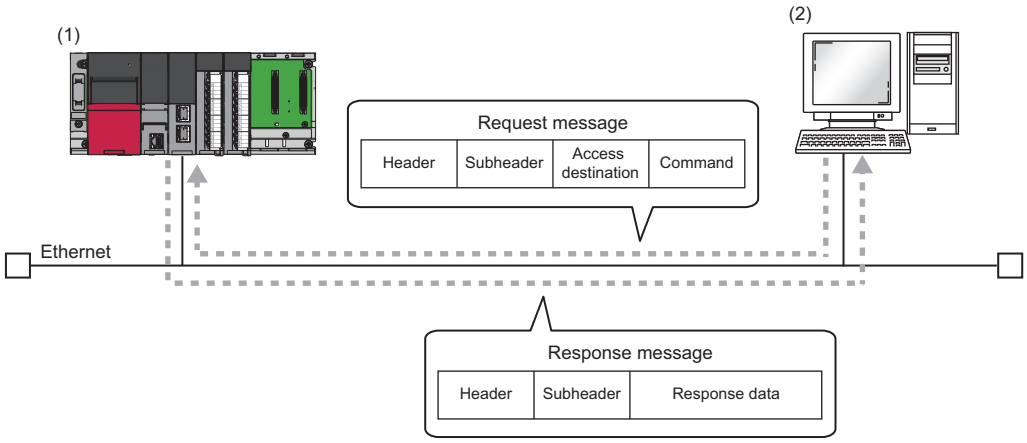
SLMP Reference Manual

**Point**

Accessing to another station via the port 1 and port 2 of the RJ71EN71 is possible for the module with the firmware version of "18" or later.

# Communication structure

When a message is sent from the external device to the Ethernet-equipped module using the SLMP message format, the Ethernet-equipped module executes a processing corresponding to the received message. During communication, the Ethernet-equipped module functions as a server and the external device (terminals such as a personal computer) functions as the client. The server (Ethernet-equipped module) automatically returns a response message suitable for the request message received from the client.




(1) Server side: Ethernet-equipped module
(2) Client side: External device

**Point**

- The simple CPU communication function enables reading from and writing to the external device that acts as the SLMP client.

☞ Page 185 Simple CPU Communication Function (RJ71EN71, RnENCPU (Network Part))
☞ Page 228 Simple CPU Communication Function (CPU Module (Built-in Ethernet Port Part))

- The Ethernet-equipped module can be used as the client by sending commands to the external device through the predefined protocol function.

☞ Page 36 Communications Using the Predefined Protocol

# Data communication procedures

This section describes the procedures for communicating with SLMP.

1. After the module parameters are set, the system checks that the initial processing of the Ethernet-equipped module has ended normally. ('Initial status' (Un\G1900024.0): On)
2. Perform the open processing to establish a connection between the Ethernet-equipped module and external device. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
3. After the connection is established, the SLMP messages are sent from the external device.
4. Close the connection when communication is finished.

**Point**

In the following case, the Ethernet-equipped module performs a remote password check when the external device is accessing the programmable controller. If communication is not possible, unlock the remote password. (☞ Page 180 Access permit processing (Unlock processing))

- When remote password is set for the CPU module.
- When connection for exchanging data with external device is set as a remote password check target.

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

1. Select "SLMP Connection Module" in "Module List" and drag it to "List of devices" or "Device map area".
2. Set the other items to the connection if required.

## Communications using an auto-open UDP port

The auto-open UDP port is used for communication with SLMP.

The auto-open UDP port is a UDP/IP port that automatically opens and closes at the following timing. When this port is used, communication is enabled when the initial processing is completed. Communication can be performed without a program regardless of the connection's open status.

### ■Open/close timing

After the Ethernet-equipped module initial processing completes, the port automatically opens according to the registered parameter settings. The port automatically closes when the power for the Ethernet-equipped module station turns off or is reset.

**Point**

- When the initial processing ends normally, the Ethernet-equipped module enables communications using an automatic open UDP port. The module waits for a communication request to the Ethernet-equipped module on the own station. (Automatic open)
- The Ethernet-equipped module accepts and processes requests from anywhere as long as they are addressed to the Ethernet-equipped module itself.
- If a communication request is received from an external device, the corresponding port number is occupied until that processing ends. Even if another communication request is accepted during this time, the communication processing will be waited.

# List of valid commands

The following table lists the commands that can be executed from the external device to the Ethernet-equipped module.
"☐" in the "Sub-command" field differs according to the specified device.
For details on each command, refer to the following.
SLMP Reference Manual

| Item | | Command | Sub command | Description |
|---|---|---|---|---|
| Type | Operation | | | |
| Device | Read | 0401 | 00☐1 | Reads value from the bit devices (consecutive device No.) in one-point units. |
| | | | 00☐0 | • Reads values from the bit device (consecutive device No.) in 16-point units.<br>• Reads value from the word devices (consecutive device No.) in one-word units. |
| | | | 00☐3 | Reads value from the bit devices (consecutive device No.) in one-point units. |
| | | | 00☐2 | • Reads values from the bit device (consecutive device No.) in 16-point units.<br>• Reads value from the word devices (consecutive device No.) in one-word units. |
| | Write | 1401 | 00☐1 | Writes value to the bit devices (consecutive device No.) in one-point units. |
| | | | 00☐0 | • Writes value to the bit devices (consecutive device No.) in 16-point units.<br>• Writes value to the word devices (consecutive device No.) in one-word units. |
| | | | 00☐3 | Writes value to the bit devices (consecutive device No.) in one-point units. |
| | | | 00☐2 | • Writes value to the bit devices (consecutive device No.) in 16-point units.<br>• Writes value to the word devices (consecutive device No.) in one-word units. |
| | Read Random | 0403 | 00☐0 | Specifies the device number and reads value from the word devices in one-word units or two-word units. This can be specified with inconsecutive device No. |
| | | | 00☐2 | Specifies the device number and reads value from the word devices in one-word units or two-word units. This can be specified with inconsecutive device No. |
| | Write Random | 1402 | 00☐1 | Specifies the device No. to bit device in one-point units and writes value. This can be specified with inconsecutive device No. |
| | | | 00☐0 | • Specifies the device No. to bit device in 16-point units and writes value. This can be specified with inconsecutive device No.<br>• Specifies the device No. to word device in one-word units or two-word units and writes value. This can be specified with inconsecutive device No. |
| | | | 00☐3 | Specifies the device No. to bit device in one-point units and writes value. This can be specified with inconsecutive device No. |
| | | | 00☐2 | • Specifies the device No. to bit device in 16-point units and writes value. This can be specified with inconsecutive device No.<br>• Specifies the device No. to word device in one-word units or two-word units and writes value. This can be specified with inconsecutive device No. |
| | Entry Monitor Device | 0801 | 00☐0 | Registers the device to be read by Execute Monitor (command: 0802). |
| | | | 00☐2 | |
| | Execute Monitor | 0802 | 0000 | Reads the value of device registered by Entry Monitor Device (command: 0801). |
| | Read Block | 0406 | 00☐0 | Reads data by treating n points of word devices or bit devices (one point is equivalent to 16 bits) as one block and specifying multiple blocks. This can be specified with inconsecutive device No. |
| | | | 00☐2 | |
| | Write Block | 1406 | 00☐0 | Writes data by treating n points of word devices or bit devices (one point is equivalent to 16 bits) as one block and specifying multiple blocks. This can be specified with inconsecutive device No. |
| | | | 00☐2 | |
| Label | Array Label Read | 041A | 0000 | Reads data from array type labels or labels whose structure members are the array. |
| | Array Label Write | 141A | 0000 | Writes data to array type labels or labels whose and structure members are the array. |
| | Read Random | 041C | 0000 | Specifies labels and reads the data. |
| | Write Random | 141B | 0000 | Specifies labels and writes data. |
| Memory | Read | 0613 | 0000 | Reads the buffer memory data of own station (SLMP-compatible device). |
| | Write | 1613 | 0000 | Writes the data in the buffer memory of own station (SLMP-compatible device). |
| Extend Unit | Read | 0601 | 0000 | Reads the data in the buffer memory of intelligent function module. |
| | Write | 1601 | 0000 | Writes the data in the buffer memory of intelligent function module. |

| Item | | Command | Sub command | Description |
|---|---|---|---|---|
| Type | Operation | | | |
| Remote Control | Remote Run | 1001 | 0000 | Executes the remote RUN to the access destination module. |
| | Remote Stop | 1002 | 0000 | Executes the remote STOP to the access destination module. |
| | Remote Pause | 1003 | 0000 | Executes the remote PAUSE to the access destination module. |
| | Remote Latch Clear | 1005 | 0000 | Executes the Remote Latch Clear to the access destination module. |
| | Remote Reset | 1006 | 0000 | Executes the Remote RESET to the access destination module. |
| | Read Type Name | 0101 | 0000 | Reads the model name and model code of the access destination module. |
| Remote Password | Lock | 1631 | 0000 | Specifies the remote password to disable the communication with other devices. (The locked state is activated from the unlocked state.) |
| | Unlock | 1630 | 0000 | Specifies the remote password to enable communication with other devices. (The unlocked state is activated from the locked state.) |
| File | Read Directory/ File | 1810 | 0040 | Reads file list information. |
| | Search Directory/ File | 1811 | 0040 | Reads the presence of the specified file, file No., and file size. |
| | New File | 1820 | 0040 | Reserves storage area for the specified file. |
| | Delete File | 1822 | 0040 | Deletes a file. |
| | Copy File | 1824 | 0040 | Copies the specified file. |
| | Change File State | 1825 | 0040 | Changes file attributes. |
| | Change File Date | 1826 | 0040 | Changes the file creation date. |
| | Open File | 1827 | 0040 | Locks a file so that the content of the file is not changed by other devices. |
| | Read File | 1828 | 0000 | Reads the data of a file. |
| | Write File | 1829 | 0000 | Writes the contents in a file. |
| | Close File | 182A | 0000 | Cancels the file lock by open processing. |
| Self Test | | 0619 | 0000 | Tests whether the communication with external devices is normally executed or not. |

# Dedicated instructions

SLMP communications can be executed using the dedicated instructions. (☞ Page 537 Instruction for SLMP communications)

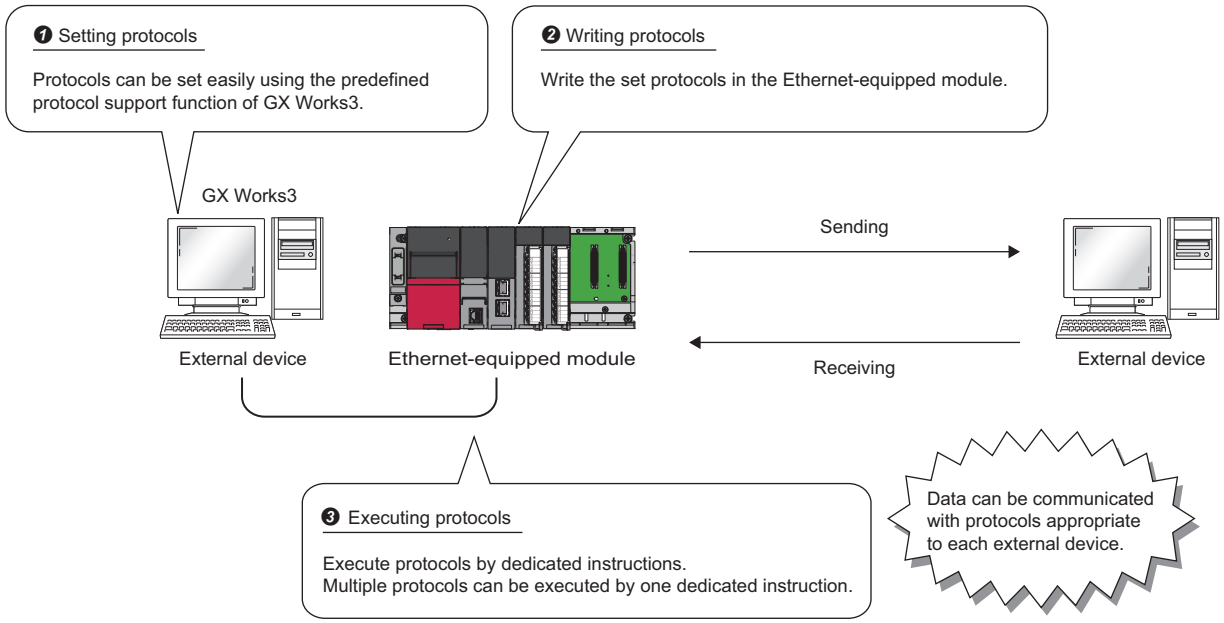
# 1.3 Communications Using the Predefined Protocol

Data can be exchanged between the external device (such as measuring instrument and bar code reader) and the CPU module following the protocol of the device.

Data that varies according to communication session can be handled by incorporating a device or buffer memory into the communication packet.

Sets the protocol required for communication with the external device using the engineering tool.

The protocol can be set by selecting from the predefined protocol library, or it can be created and edited.




**Point**

The number of protocols and packets that can be registered is as follow.

- Protocols: 128 maximum
- Packets: 256 maximum
- Packet data area size: 12288 bytes maximum

When the number of packets reaches the upper limit, protocols cannot be added even if the number of protocols has not reached the upper limit. If the packet data area size reaches the upper limit, protocols and packets cannot be added even if the number of protocols and packets has not reached the upper limit.

## Applicable connections

The connections No.1 to 16 of the P1 connector can be used for the communications using the predefined protocol.

Communications using the predefined protocol cannot be used with the P2 connector.

**Point**

- When two or more ECPRTCL instructions are executed at the same time for the same connection, the later instruction is ignored and not executed until the earlier instruction is completed.
- When, for a pairing-set connection, two or more ECPRTCL instructions are executed at the same time by specifying a pair of two different connections, the later instruction is completed with an error until the earlier instruction is completed.

# Data communication procedures



When the predefined protocol support function is used, data can be exchanged with the external device using the following procedure.

1. Select, create or edit the protocol with the predefined protocol support function, and write the protocol setting data. (☞ Page 37 Creating the protocol setting data)
2. Set the module parameter. (☞ Page 43 Setting procedure)
3. Write the parameters to the CPU module, and check that initial processing of the Ethernet-equipped module completed successfully. ('Initial status' (Un\G1900024.0): On)
4. Perform the open processing to establish a connection between the Ethernet-equipped module and external device. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
5. Execute the protocol with the ECPRTCL instruction. (☞ Page 537 Instructions for predefined protocol communications)
6. Close the connection when communication is finished.

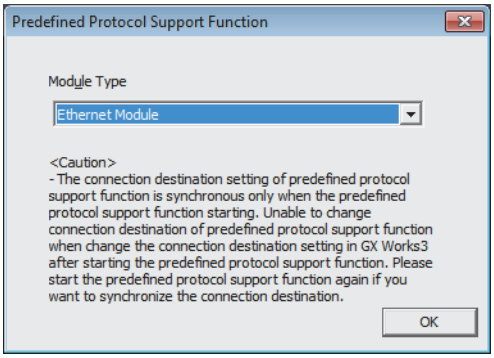
**Point**

The communication data code is binary code communication regardless of the selected settings.

## Creating the protocol setting data

Use the predefined protocol support function to create the protocol setting data.

[Tool] ⇨ [Predefined Protocol Support Function]

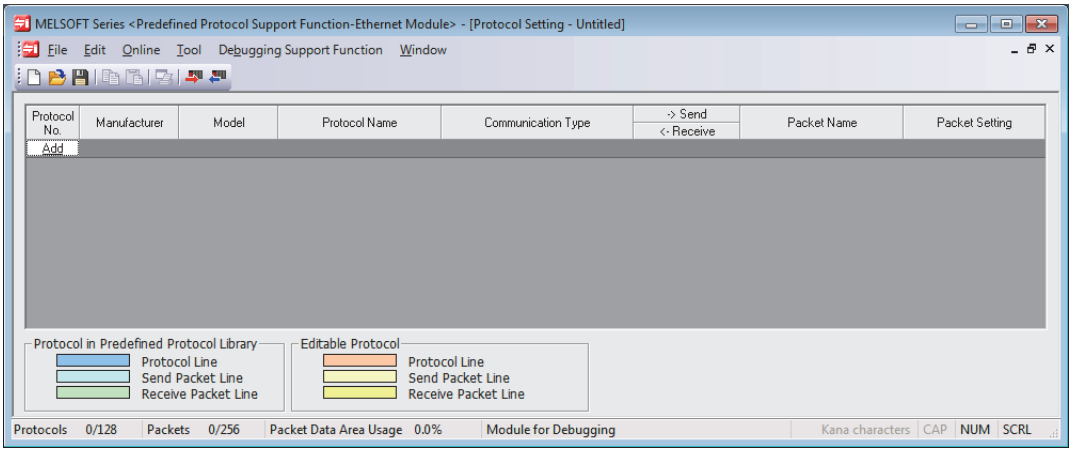



1. Select the module for which to create the protocol setting data. Select the following items when using the RnENCPU.
   - CPU part: "Built-in Ethernet CPU"
   - Network part: "Ethernet Module"

## ■Newly creating the protocol setting data

Newly create the protocol setting data.

[File] ⇨ [New]

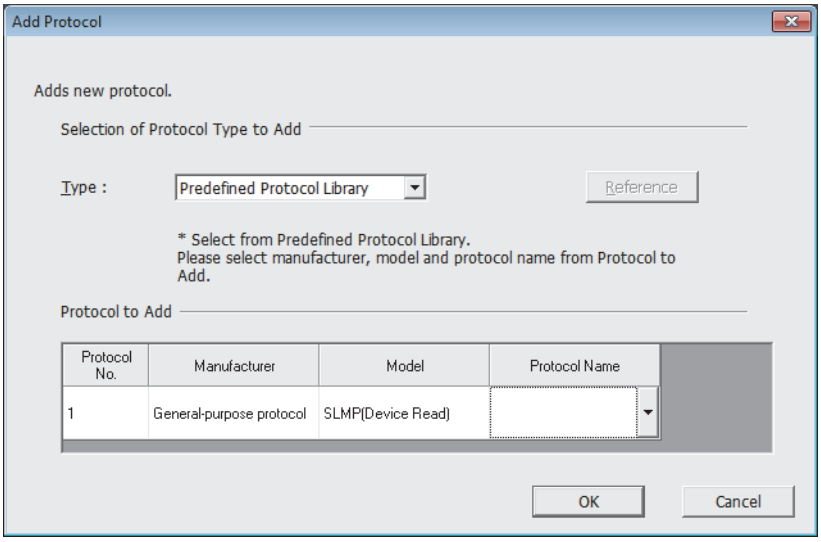



| Item | Description |
|---|---|
| Protocol No. | Displays the protocol number used with the dedicated instruction in the program. |
| Manufacturer | Displays the name of the manufacturer of the device for which the protocol is being set. |
| Model | Displays the model of the protocol to be set. |
| Protocol Name | Displays the name of the protocol to be set. |
| Communication Type | Displays the communication type of the protocol to be set.<br>Send Only: Sends one send packet once.<br>Receive Only: If there is a matching packet up to 16 registered and received packets, it is received.<br>Send & Receive: After sending one send packet, if there is a matching packet up to 16 registered and received packets, it is received. |
| →Send/←Receive | Displays the packet send direction.<br>→: For send<br>←(1) to (16): For receive, the received packet number is displayed in parentheses. |
| Packet Name | Displays the packet name. |
| Packet Setting | Displays the validity of variables in the packet elements and the variable setting state.<br>If the variable is not set, there are no elements, or there is an element error, the protocol is not written to the Ethernet-equipped module.<br>No Variable: When there is no variable in the elements<br>Variable Set: Only when all variables have been set<br>Variable Unset: When there is even one unset variable<br>Element Unset: When there are no elements in an editable protocol<br>Element error: When elements do not satisfy requirements |

## ■Adding protocol

Add protocol.

[Edit] ⇨ [Add Protocol]



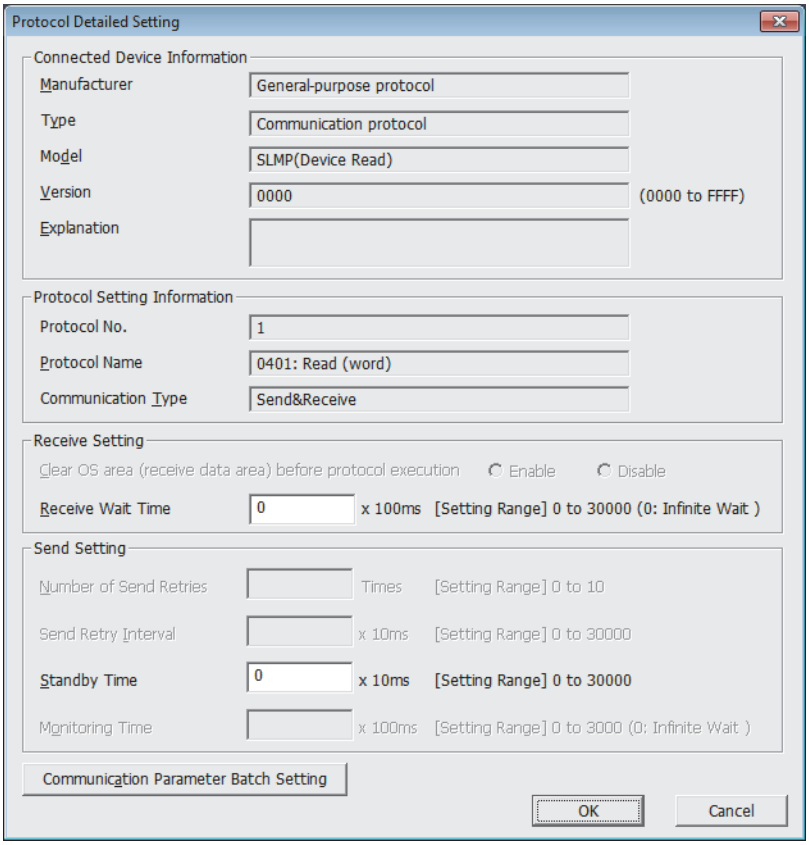

| Item | Description | Setting range |
|---|---|---|
| Type | Select the type of protocol to be added. | • Predefined Protocol Library<br>• User Protocol Library<br>• Add New |
| Protocol No. | Select the protocol number to be added. | 1 to 128 |
| Manufacturer*1 | Set the maker of the protocol to be added. | — |
| Model*1 | Set the type of protocol to be added. | — |
| Protocol Name*1 | Set the name of the protocol to be added. | — |

*1 The name can be set only when "Predefined Protocol Library" is selected for "Type".

## ■Protocol Detailed Setting

Set the protocol send/receive parameters.

"Protocol Setting" window ⇨ Select a protocol ⇨ [Edit] ⇨ [Protocol Detailed Setting]




| Item | | Description |
|---|---|---|
| Connected Device Information*1 | Manufacturer | Set the protocol maker name. |
| | Type | Set the protocol device type. |
| | Model | Set the protocol model. |
| | Version | Set the protocol device version. |
| | Explanation | Set a description of the protocol device. |
| Protocol Setting Information*1 | Protocol No. | The protocol number for the selected protocol is displayed. |
| | Protocol Name | Set the protocol name. |
| | Communication Type | Set the protocol communication type. |
| Receive Setting | Receive Wait Time | Set the time for wait after the module enters the receive data wait state.<br>If communication with the external device is disabled because of a disconnection and matching packet data is not received within the specified time, the module judges that an error has occurred and cancels the receive data wait state. |
| Send Setting | Standby Time | Set the time to wait from when the protocol set for the module enters the execution state to when the data is actually sent. The time for the external device to enter the receive enable state can be adjusted with this in respect to the module's send timing. |

*1 The setting cannot be changed if the protocol was selected from the predefined protocol library.

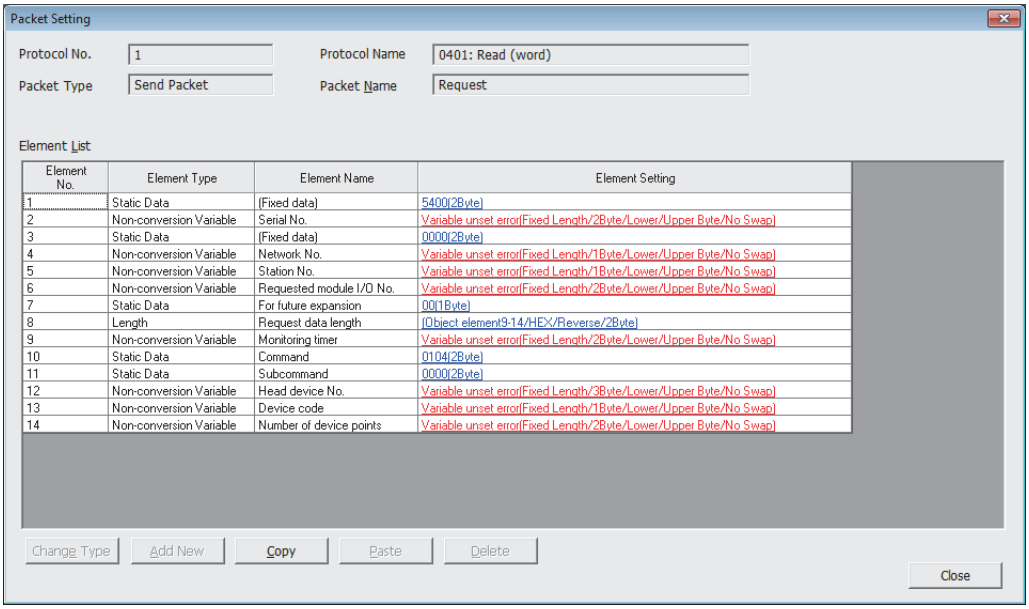
**Point**

Send/receive parameters can be set for multiple protocols by clicking the [Communication Parameter Batch Setting] button and setting the range of the set protocol numbers, receive settings, and send settings.

## ■Packet setting

Set the configuration of the send/receive packets on the "Packet Setting" window.

"Protocol Setting" window ⇨ Packet to be set




The above window opens when "Predefined Protocol Library" is selected on the "Add Protocol" window.

When "Add New" or "User Protocol Library" has been selected, configure the packets with the [Change Type] button and [Add New] button.

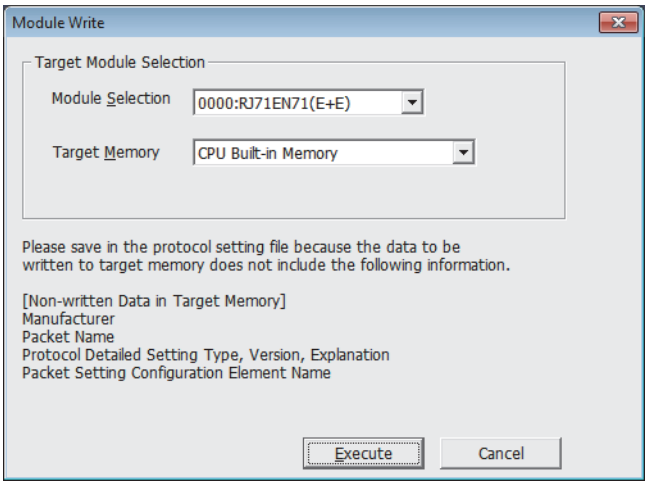
For details on the packet elements, refer to the following.

☞ Page 45 Packet elements

## ■Writing the protocol setting data

Write the protocol setting data to the Ethernet-equipped module.

[Online] ⇨ [Write to Module]




Select the module and memory into which the protocol data is to be written, and execute write.

When writing to a CPU module, the protocol setting data is written into the module extension parameters.

**Point**

The following data is not written as the protocol setting data so it will not be displayed even when read.

However, when the protocol is selected from the predefined protocol library, the following can be displayed.

- Manufacturer
- Packet Name
- Protocol Detailed Setting Type, Version, Explanation
- Packet Setting Configuration Element Name

After writing the protocol setting data, the setting data are enabled at the following timing.

- When the system is powered off and on
- When the CPU module is reset
- When the status of the CPU module changed from STOP to RUN state

The predefined protocol settings written in the SD memory card can be transferred to the CPU module memory by using boot operation.

For details on boot operation, refer to the manual for the CPU module used.

### ■When the protocol setting data are written into multiple target memory

When the protocol setting data are written into multiple target memory, the following operation will take place.

○: The protocol setting data are written., ×: The predefined protocol data are not written.

| Target memory | | | Operation |
|---|---|---|---|
| CPU module (built-in Ethernet port part) | SD memory card | RJ71EN71, RnENCPU (network part) | |
| ○ | ○ | × | Operation follows settings in "Module Extended Parameter" under "Setting of File/Data Use or Not in Memory Card" of the "Memory Card Parameter" window. |
| ○ | × | ○ | As soon as the protocol setting data are enabled, the predefined protocol settings of the RJ71EN71 or the RnENCPU (network part) are overwritten by the predefined protocol settings in the CPU module or SD memory card. |
| × | ○ | ○ | |
| ○ | ○ | ○ | Operation follows settings in "Module Extended Parameter" under "Setting of File/Data Use or Not in Memory Card" of the "Memory Card Parameter" window.<br>As soon as the protocol setting data are enabled, the predefined protocol settings of the RJ71EN71 or the RnENCPU (network part) are overwritten by the predefined protocol settings in the CPU module or SD memory card. |

**Point**

By writing the protocol setting data to the CPU built-in memory, the same setting can be used even after module change.

When capacity of the CPU built-in memory is insufficient, write data to an SD memory card.

Although the protocol setting data can also be written to the intelligent function module, the data must be written again when the module is changed.

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

***1.*** Select the external device to be connected in "Module List" and drag it to "List of devices" or "Device map area".

| External device name | Description |
|---|---|
| UDP Connection Module | Select to communicate with the external device using UDP/IP. |
| Active Connection Module | Select to perform the open processing to the external device from the Ethernet-equipped module (Active open) and communicate using TCP/IP. |
| Unpassive Connection Module | Select to receive the open processing from an unspecified external device (Unpassive open) and communicate using TCP/IP. |
| Fullpassive Connection Module | Select to receive the open processing from the specified external device (Fullpassive open) and communicate using TCP/IP. |

***2.*** Set "Communication Method" for the external device to "Predefined Protocol".

***3.*** Set the other parameters required for communication in the connection.

# Protocol communication type

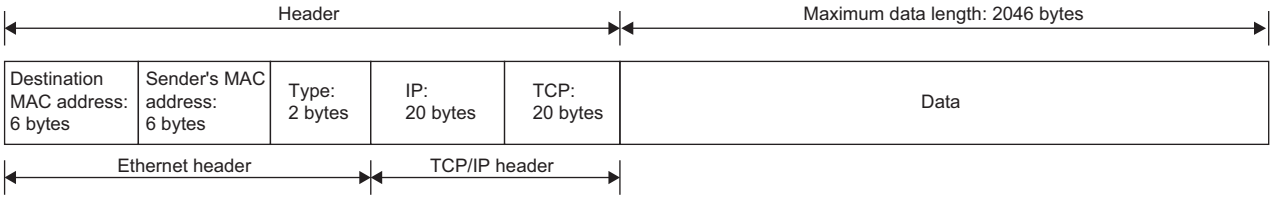
The packets sent to the external device when a processing is executed and the external device's receive packets are registered in the protocol.

The packet elements set with the predefined protocol support function are the data section of the packets that are actually sent and received.

This section describes an example of the packet configuration. For details on the packet elements, refer to the following.

☞ Page 573 Operation Image and Data Structure of Predefined Protocol

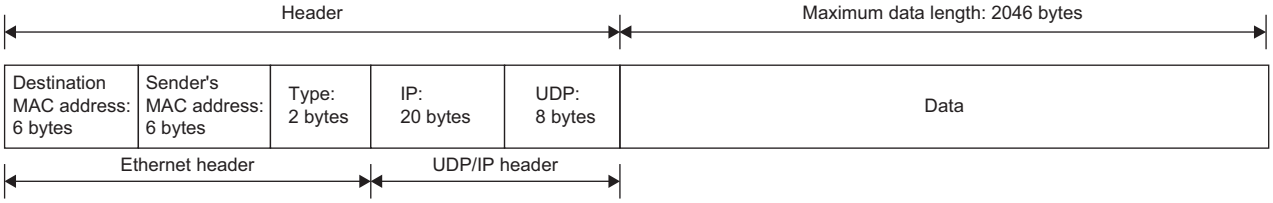
## For TCP/IP




### ■Precautions for Passive open

- When the CPU module is connected to external device in Passive open state, the IP address of the connected external device or port number of the external device can be obtained with the SP.SOCCINF instruction. (☞ Page 537 Instructions for predefined protocol communications)

## For UDP/IP




With the predefined protocol support function, data is exchanged with the external device with the procedures (communication type) shown below.

For details on the communication type operation, refer to the following.

☞ Page 573 Operation image of each communication type of protocol

| Communication type | Description |
|---|---|
| Send Only | The send packet is sent once. |
| Receive Only | If there is a packet that matches within the maximum of 16 registered receive packets, the packet is received. |
| Send & Receive | After sending the send packets, if there are packets that match the up to 16 registered receive packets, the packets are received. |

### ■Precautions for UDP/IP communications

- Use the SP.SOCCSET instruction to change the external device by the CPU module. (☞ Page 537 Instructions for predefined protocol communications)

### ■Precautions for broadcast communications

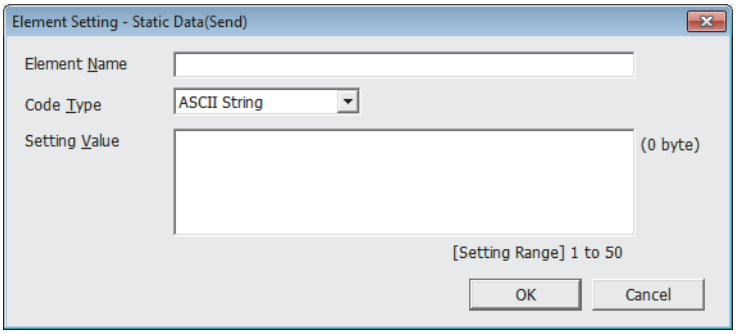
- When the CPU module receives data by receiving the broadcast, the IP address of the external device which send data and port number of the external device can be obtained with the SP.SOCCINF instruction. (☞ Page 537 Instructions for predefined protocol communications)

# Packet elements



The packet is created with a combination of packet elements.
Up to 32 elements can be set in one packet. One packet can have a maximum data length of 2046.
This section describes the details of the packet elements.
For examples of the packet element data, refer to the following.
☞ Page 573 Operation Image and Data Structure of Predefined Protocol

## Static data



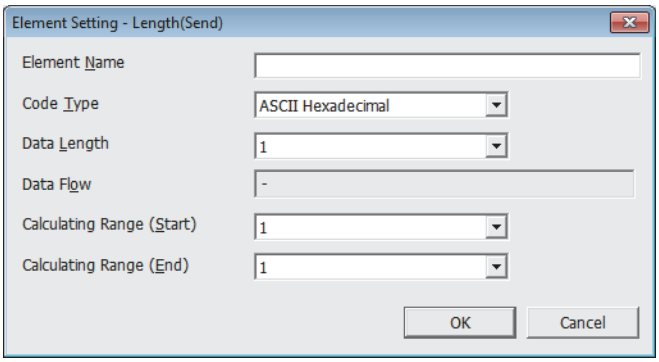

Use when there are specific codes and character strings, such as commands, in the packet.

- When sending: The specified code and character string are sent.
- When receiving: The received data is verified.

Multiple static data elements can be placed anywhere in the data part.
The following table lists the items.

| Item | Description | Remarks |
|---|---|---|
| Element Name | Set the element name. | — |
| Code Type | Select a data type of the setting value.<br>ASCII String/ASCII Control Code/HEX | — |
| Setting Value | Set data within 1 to 50 bytes.<br>Code type and setting range are as follows:<br>• ASCII String: 20H to 7EH<br>• ASCII Control Code: Control code of 00H to 1FH and 7FH<br>• HEX: Hexadecimal data of 00H to FFH | Setting example<br>ASCII string: "ABC"<br>ASCII control code: STX<br>HEX: FFFF |

## Length




The length code is used when there is an element that indicates the data length in the packet.

- When sending: Automatically calculates the data length in the specified range, and adds it to the packet.
- When receiving: From the received data, the data (value) corresponding to the length is verified as the specified range's data length.

Length elements can be placed anywhere in the data part.

Multiple length elements can be set placed in one packet.

The following table lists the items.

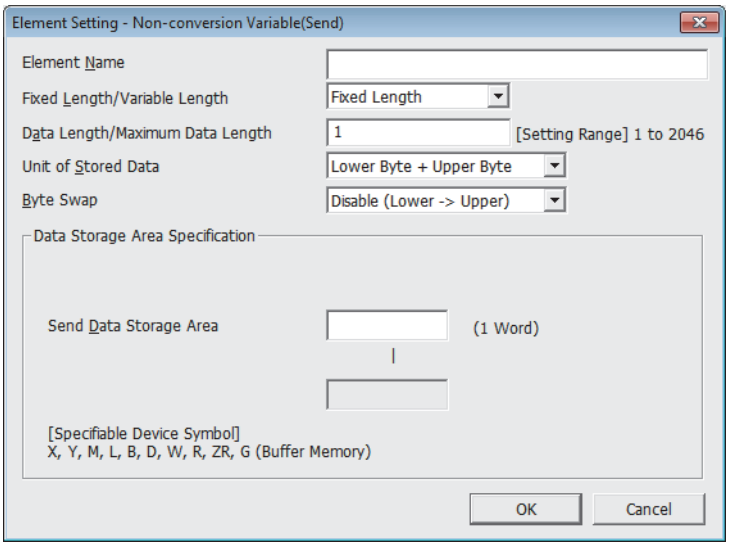
| Item | | Description | | Remarks |
|---|---|---|---|---|
| Element Name | | Set the element name. | | — |
| Code Type | | Select the data length type.<br>ASCII Hexadecimal/HEX | | — |
| Data Length | | Select the data length on the line.<br>The range is 1 to 4 bytes. | | — |
| Data Flow | | Forward Direction<br>(Upper Byte → Lower Byte) | When sending: Sends the calculated length in order from the upper byte.<br>When receiving: Receives the data in order from the upper byte. | This cannot be set if the data length is 1 byte. |
| | | Reverse Direction<br>(Lower Byte → Upper Byte) | When sending: Sends the calculated length in order from the low-order byte.<br>When receiving: Receives the data in order from the low-order byte. | |
| | | Byte Swap (by Word)*1 | When sending: Interchanges the bytes in word units and sends the calculated length.<br>When receiving: Interchanges the bytes in word units and receives the data. | |
| Calculating Range | Start | Select the start packet element number for the range to be calculated.<br>The range is 1 to 32. | | — |
| | End | Select the end packet element number for the range to be calculated.<br>The range is 1 to 32. | | |

*1 This can be selected only when the data length is set to 4 bytes.

**Point**

- If there are no elements other than length, an element error occurs. (When using length, one or more elements other than length are required.)
- If the calculation result exceeds the number of digits set with "Data Length", the excessive digit value is discarded (invalidated). For example, if the data length is 2 bytes and the data size calculation results are "123" bytes, the data length will be "23".
- If there is a non-conversion variable (variable length)/non-verified reception (character length variable) after the length, and that section is not included in the length calculating range, arrange the static data immediately after the non-conversion variable/non-verified reception.
- When the code type setting is "ASCII Hexadecimal", a mismatch will occur if a character string other than "0" to "9", "A" to "F", and "a" to "f" is received.
- Use "0" to "9" or "A" to "F" when converting to ASCII characters during send.
- When arranging multiple length elements, none of the length calculating range may overlap.
- When arranging multiple length elements, the previous length calculating range may not exceed the arranged length.
- A length element cannot be arranged at the final position of the packet elements.

## Non-conversion variable




Use this to send the CPU module device or buffer memory data as part of the send packet, or to store part of the received packet in the CPU module device or buffer memory.

Multiple non-conversion variable can be arranged in one packet.

The following table lists the items.

| Item | Description | |
|---|---|---|
| Element Name | Set the element name. | |
| Fixed Length/Variable Length | Fixed Length | The data whose length is fixed is sent and received. |
| | Variable Length | When sending: The data length is specified at the time of the protocol execution and the data is sent.<br>When receiving: The data whose length is variable is received. |
| Data Length/Maximum Data Length | Set the data length of the send/receive data.<br>(For a variable length, set the maximum data length that can be specified for the data length storage area.)<br>The range is 1 to 2046. | |
| Unit of Stored Data | Lower Byte + Upper Byte | When sending: Each one word (2 bytes) data in the data storage area is sent in the order of the lower byte to the upper byte.<br>When receiving: The receive data is stored to the data storage area in the order of the lower byte to the upper byte. |
| | Lower Bytes Only | When sending: Each lower byte data in the data storage area is sent. The Ethernet-equipped module ignores the upper byte data.<br>When receiving: The receive data is stored to each lower byte in the data storage area. The Ethernet-equipped module stores 00H in the upper byte. |
| Byte Swap | Disable (Lower → Upper)/Enable (Upper → Lower) | When sending:<br>When "Enable (Upper → Lower)" is selected, data in the upper byte and lower byte are swapped by one word (2 bytes) and sent. When "Unit of Stored Data" is "Lower Byte + Upper Byte" and "Data Length" is an odd number of bytes, the upper byte is sent at transmission of the last byte. When "Unit of Stored Data" is "Lower Bytes Only" and "Data Length" is an odd number of bytes, data without any byte swap is sent at transmission of the last byte.<br>When receiving:<br>When "Enable (Upper → Lower)" is selected, data in the upper byte and lower byte are swapped by word and sent. When "Unit of Stored Data" is "Lower Byte + Upper Byte" and "Data Length" is an odd number of bytes, the last byte is stored to the upper byte. When "Unit of Stored Data" is "Lower Bytes Only" and "Data Length" is an odd number of bytes, the last byte is stored without any byte swap. |

| Item | Description |
|---|---|
| Data Storage Area Specification | Specify the start device for storing the variable value.<br>The settable devices are listed below.<br>Inner user[*1*2]<br>• Input (X)<br>• Output (Y)<br>• Internal relay (M)<br>• Latch relay (L)<br>• Link relay (B)<br>• Data register (D)<br>• Link register (W)<br>File register[*2]<br>• File register (R, ZR)<br>Buffer memory<br>• G device (G) ('Send/receive area for predefined protocol support function' (Un\G1902000 to Un\G1904047))[*3*4] |

*1 Do not set local devices.

*2 Set within the device range specified with "Device/Label Memory Area Setting" in "Memory/Device Setting" of "CPU Parameter".

*3 This cannot be set if the target module is a CPU module.

*4 Un\G18432 to Un\G20479 if the RJ71EN71 network type is set to "Q Compatible Ethernet".

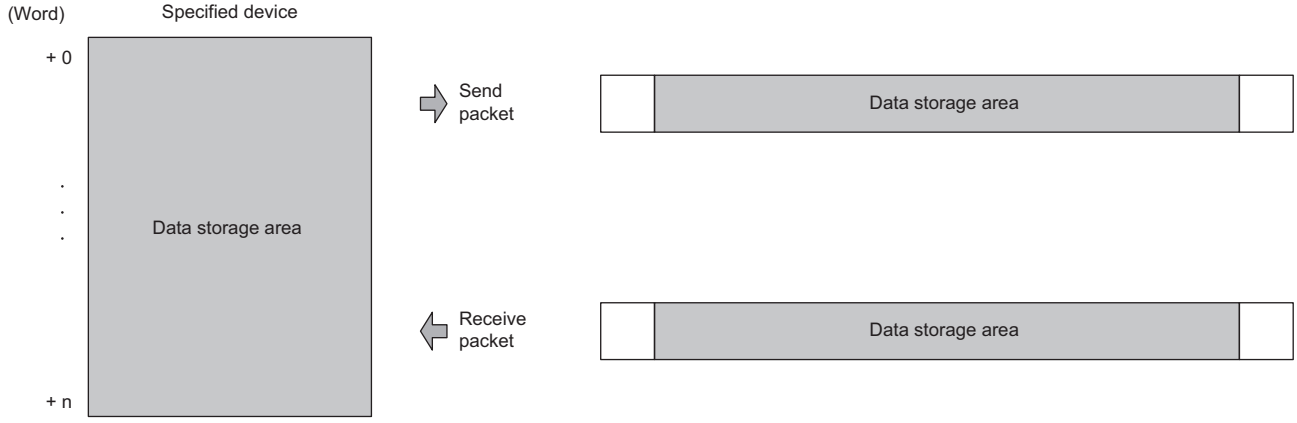
The following figures show the configuration of the data storage area.

### ■When "Fixed Length/Variable Length" is "Fixed Length"

The area after the device number specified on the "Element Setting" window becomes the data storage area.

The occupied data storage area differs according to the "Unit of Stored Data".

- When "Lower Byte + Upper Byte" is selected, the same size as the data length is occupied. (However, when the data length of a send packet is an odd number, the upper byte (lower byte for "Byte Swap") of the end device is not sent. When the data length of a receive packet is an odd number, the last data is stored with one byte of 00H.)
- When "Lower Bytes Only" is selected, a size double the data length is occupied.




For send packet: Send data is stored by the program

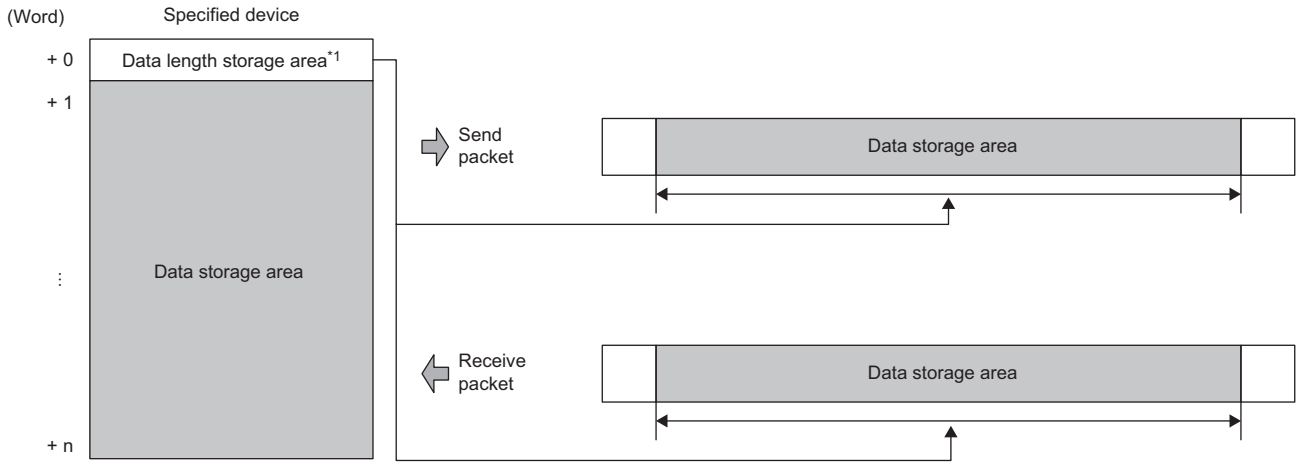
For receive packet: Receive data is stored by the Ethernet-equipped module

### ■When "Fixed Length/Variable Length" is "Variable Length"

The area after the device number specified on the "Element Setting" window + 1 becomes the data storage area.

The occupied data storage area differs according to the "Unit of Stored Data".

- When "Lower Byte + Upper Byte" is selected, the same size as the data length + one word (length for the data length storage area) are occupied. (However, when the data length of a send packet is an odd number, the upper byte (lower byte for "Byte Swap") of the end device is not sent. When the data length of a receive packet is an odd number, the last data is stored with one byte of 00H.)
- When "Lower Bytes Only" is selected, a size double the data length + one word (for data length storage area) is occupied.




For send packet: Send data is stored by the program

For receive packet: Receive data is stored by the Ethernet-equipped module

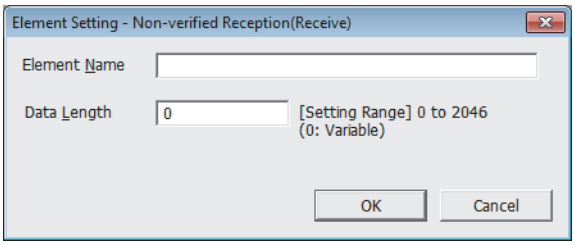
*1 The data length unit is byte fixed

**Point**

When "Fixed Length/Variable Length" is "Variable Length" and the configuration is set as follows, an error occurs:

- An element other than static data is placed behind a non-conversion variable element when non-conversion variable is out of the length calculating range or when there is no length element (except for when non-conversion variable is placed at the end of the packet elements).
- Multiple non-conversion variable elements are placed in the length calculating range, while a length element is not placed.
- A non-conversion variable element is placed before a length element in the length calculating range.

## Non-verified reception




Use this when receive data include unnecessary data.

If the receive packet contains non-verified reception, Ethernet-equipped module skims over the specified number of characters.

Multiple non-verified reception elements can be set in one packet.

The following table lists the items.

| Item | Description | | Remarks |
|---|---|---|---|
| Element Name | Set the element name. | | — |
| Data Length | 0 (Number of characters variable) | Set when the number of characters that are not verified differs between each communication session. | — |
| | 1 to 2046 (number of character specification) | Set the number of characters that are not verified. | |

**Point**

When "Data Length" is set to 0, an error will occur if the following layout is used.

- An element other than static data is placed behind a non-verified reception element when non-verified reception is out of the length calculating range or when there is no length element (except for when non-verified reception is placed at the end of the packet elements).
- Multiple non-verified reception elements are placed in the length calculating range, while a length element is not placed.
- A non-verified reception element is placed before a length element in the length calculating range.

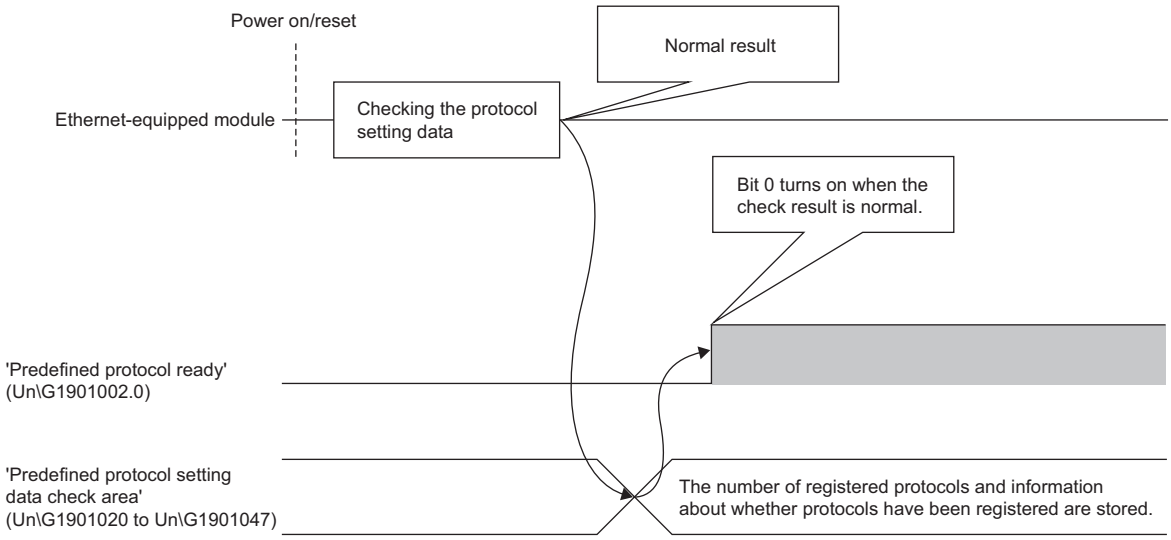
# Execution conditions of predefined protocol communications

The predefined protocol communications can be executed when 'Predefined protocol ready' (Un\G1901002.0) is on.
This section describes the operation of 'Predefined protocol ready' (Un\G1901002.0).

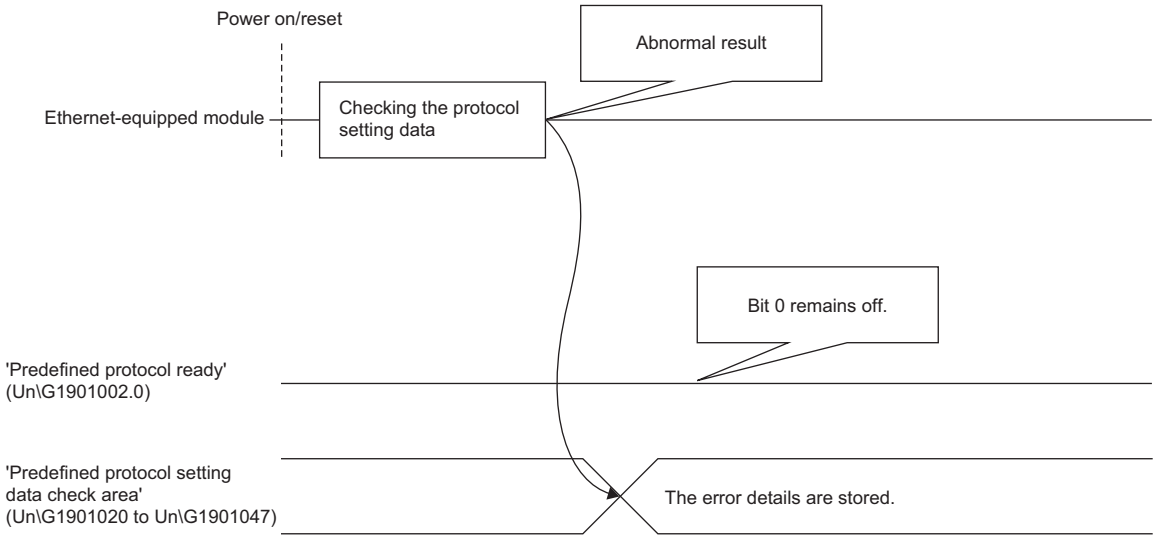
## When the system is powered on or reset

If protocol setting data is written in, the Ethernet-equipped module checks the protocol setting data when the system is powered on or reset.
If the protocol setting data is normal, the Ethernet-equipped module turns on 'Predefined protocol ready' (Un\G1901002.0), and enables execution of the protocol.
'Predefined protocol ready' (Un\G1901002.0) is used as the interlock signal for executing the protocol.
If the protocol setting data is abnormal, 'Predefined protocol ready' (Un\G1901002.0) stays off, and the details of the error are stored in 'Predefined protocol setting data check area' (Un\G1901020 to Un\G1901047).
If protocol setting data is not written in, the protocol setting data is not checked, and 'Predefined protocol ready' (Un\G1901002.0) remains off.
Whether the protocol setting data is registered or not can be checked with 'Number of registered predefined protocols' (Un\G1901024) and 'Predefined protocol registration' (Un\G1901032 to Un\G1901047).

### ■When protocol setting data is normal




### ■When protocol setting data is abnormal




## When executing UINI instruction

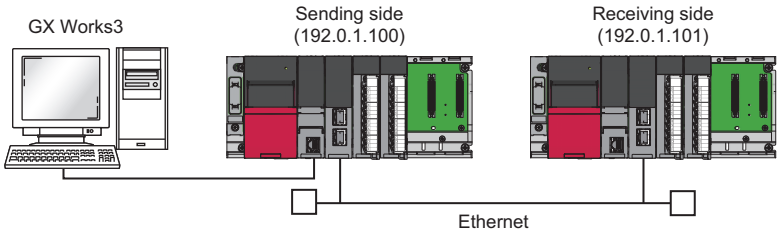
When the RJ71EN71 network type is set to "Q Compatible Ethernet" and the UINI instruction is executed, the protocol setting data will not be checked.

Predefined protocol ready maintains the state before the UINI instruction was executed.

# Example of predefined protocol communications

This section describes an example of predefined protocol communications using UDP/IP.

## System configuration



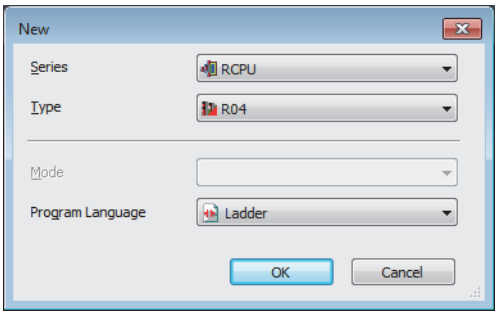

## Parameter settings

Connect the engineering tool to the CPU module and set the parameters.

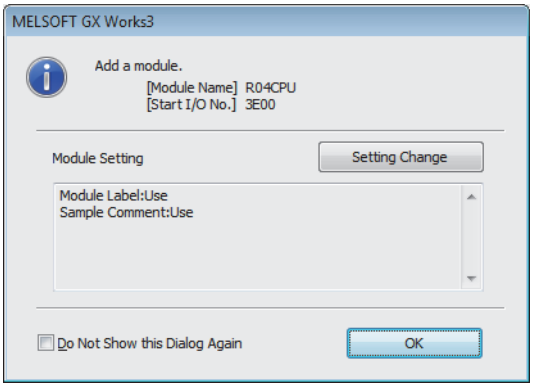
### ■Sending side

***1.*** Set the CPU module as follows.

[Project] ⇨ [New]



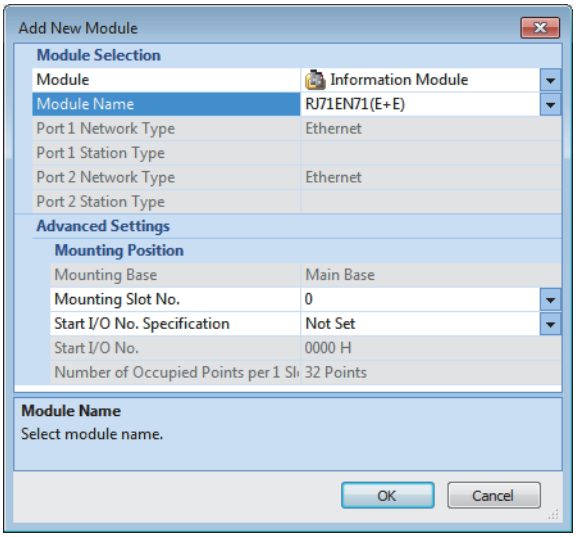

***2.*** Click the [Setting Change] button in the following window and set the item to use module labels.



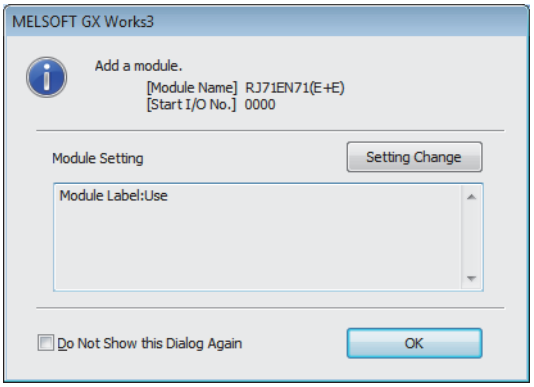

***3.*** Click the [OK] button to add the module labels of the CPU module.

**4.** Set the RJ71EN71 as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]

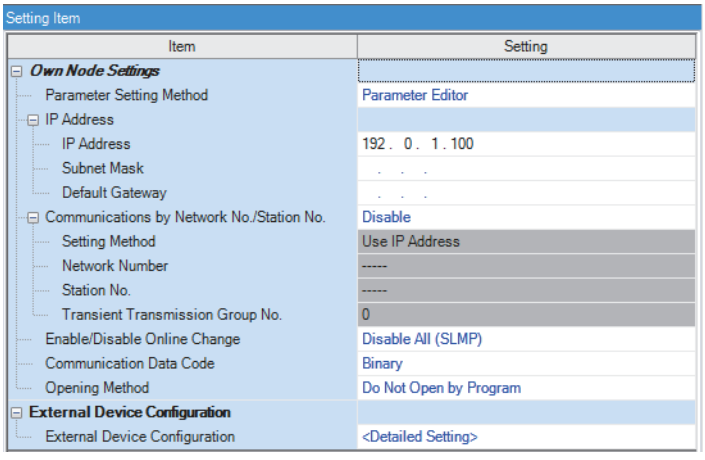



**5.** Click the [OK] button to add the module labels of the RJ71EN71.



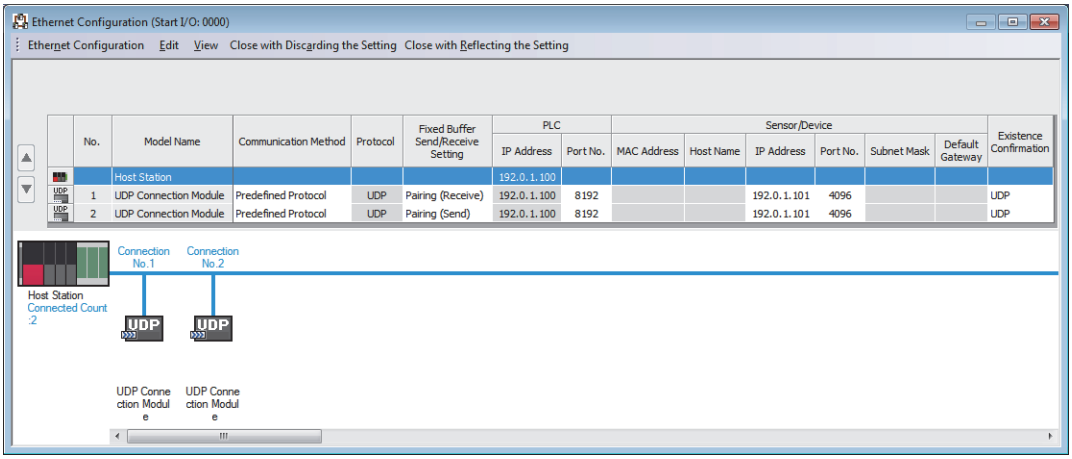

**6.** Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings]



**7.** Set the network configuration as follows.

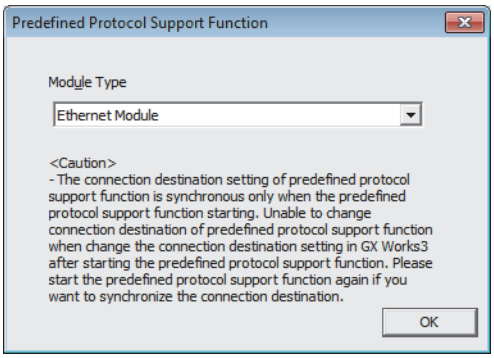
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [External Device Configuration]




**8.** Start the predefined protocol support function.

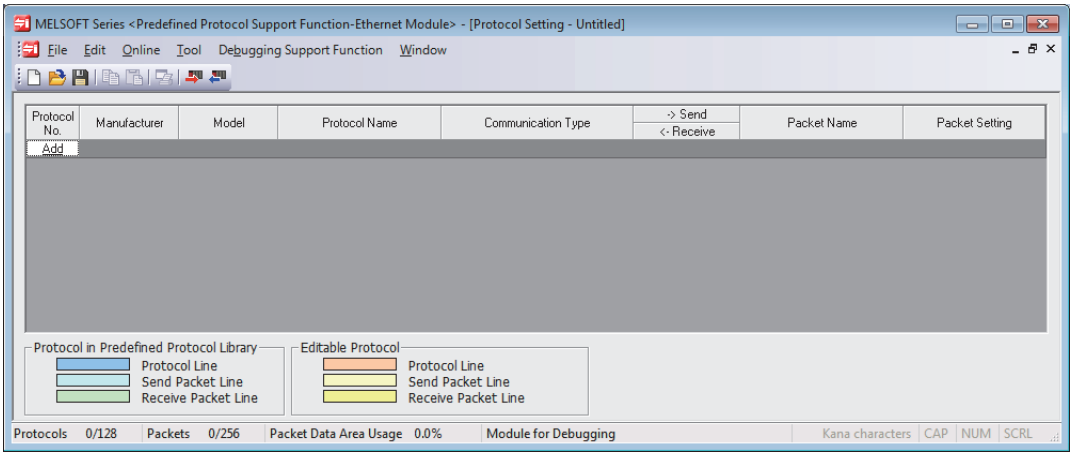
[Tool] ⇨ [Predefined Protocol Support Function]

**9.** Select "Ethernet Module" for "Module Type" and click the [OK] button.



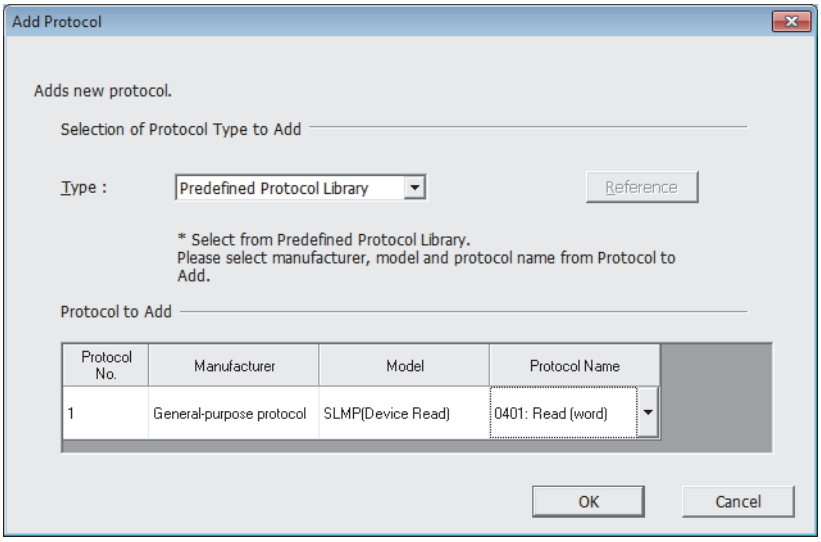

**10.** Newly create the protocol setting.

[File] ⇨ [New]



**11.** Set the protocol as follows.

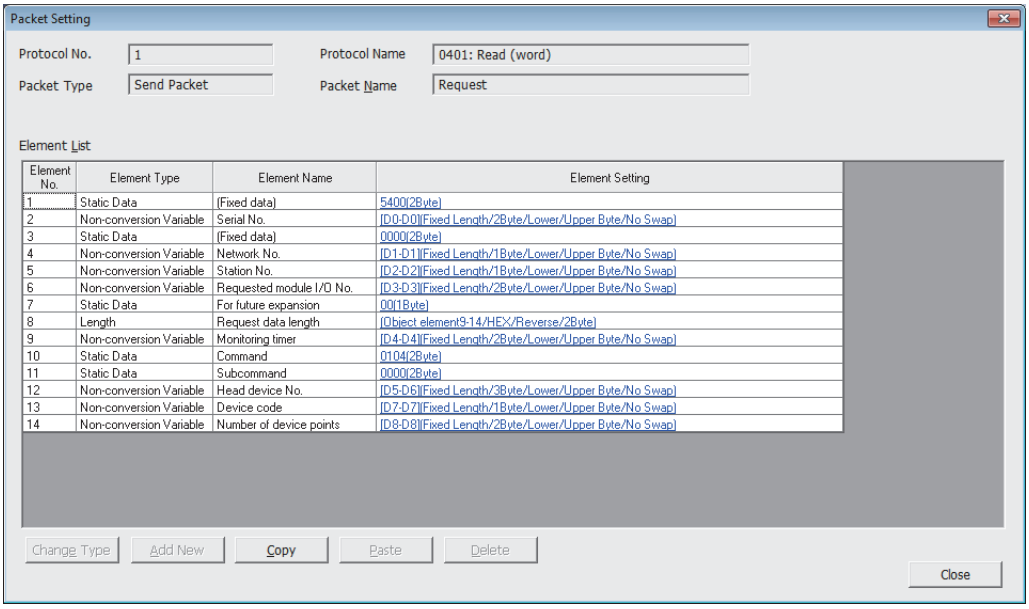
[Edit] ⇨ [Add Protocol]



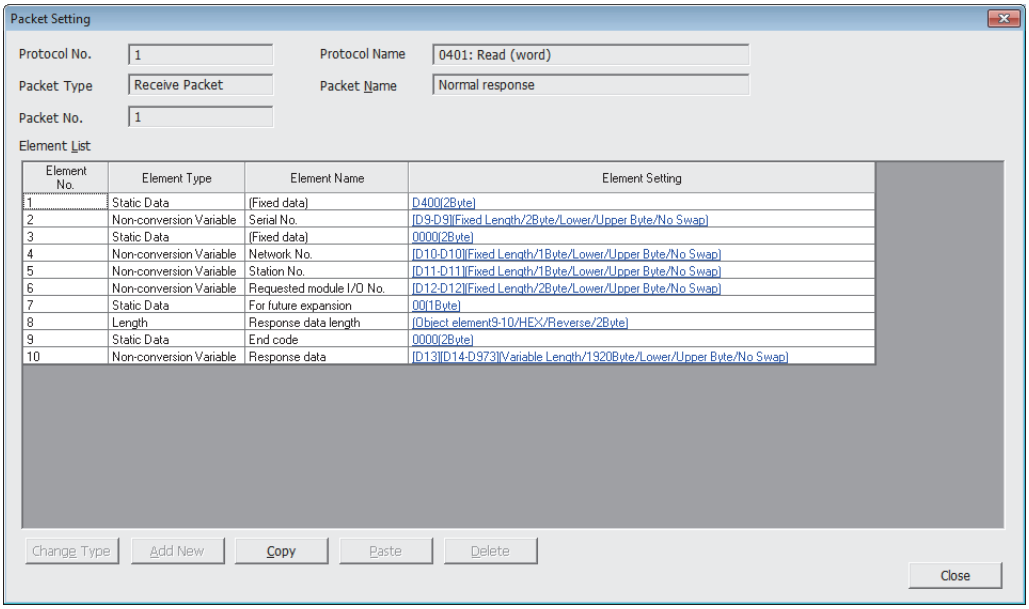

**12.** Set each packet as follows.

"Protocol Setting" window ⇨ Packet to be set

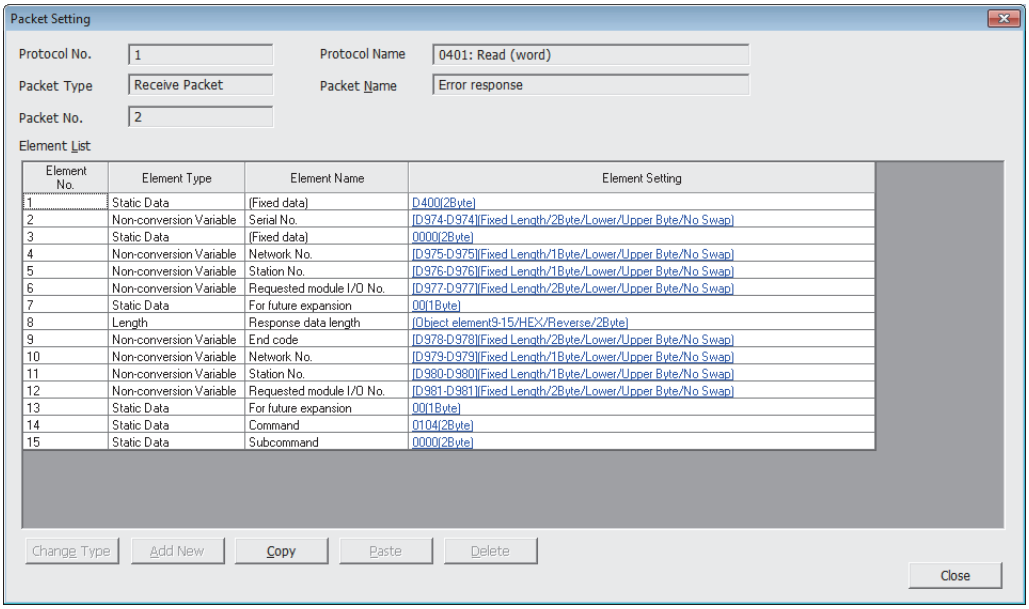
- Request



• Normal response

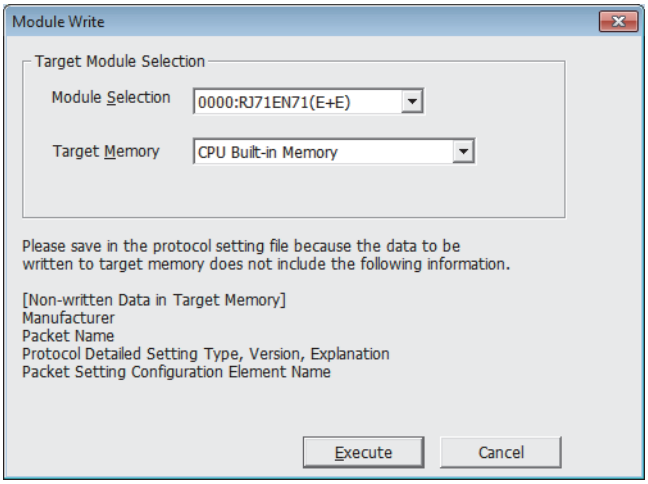



• Error response



**13.** Write the protocol setting data to the CPU module.

[Online] ⇨ [Write to Module]




**14.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

## ■Receiving side

**1.** Set the CPU module and add the module labels of the CPU module. The setting method of the CPU module and addition method of the module label are the same as those of when setting the sending side. (☞ Page 54 Sending side)

**2.** Set the RJ71EN71 and add the module labels of the RJ71EN71. The setting method of the RJ71EN71 and addition method of the module label are the same as those of when setting the sending side. (☞ Page 54 Sending side)

**3.** Set the items in "Basic Settings" as follows.

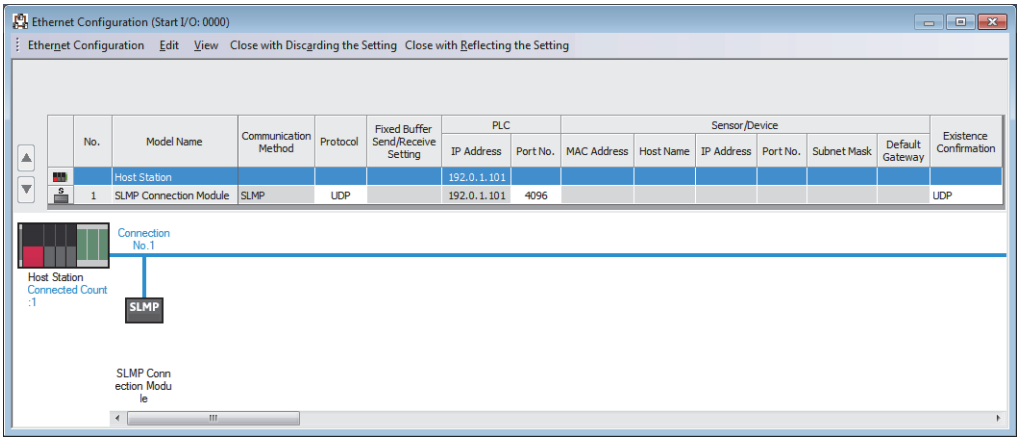
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings]

Setting Item

| Item | Setting |
|---|---|
| ***Own Node Settings*** | |
| Parameter Setting Method | Parameter Editor |
| IP Address | |
| IP Address | 192 . 0 . 1 . 101 |
| Subnet Mask | . . . |
| Default Gateway | . . . |
| Communications by Network No./Station No. | Disable |
| Setting Method | Use IP Address |
| Network Number | ----- |
| Station No. | ----- |
| Transient Transmission Group No. | 0 |
| Enable/Disable Online Change | Disable All (SLMP) |
| Communication Data Code | Binary |
| Opening Method | Do Not Open by Program |
| **External Device Configuration** | |
| External Device Configuration | <Detailed Setting> |

**4.** Set the network configuration as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [External Device Configuration]




**5.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

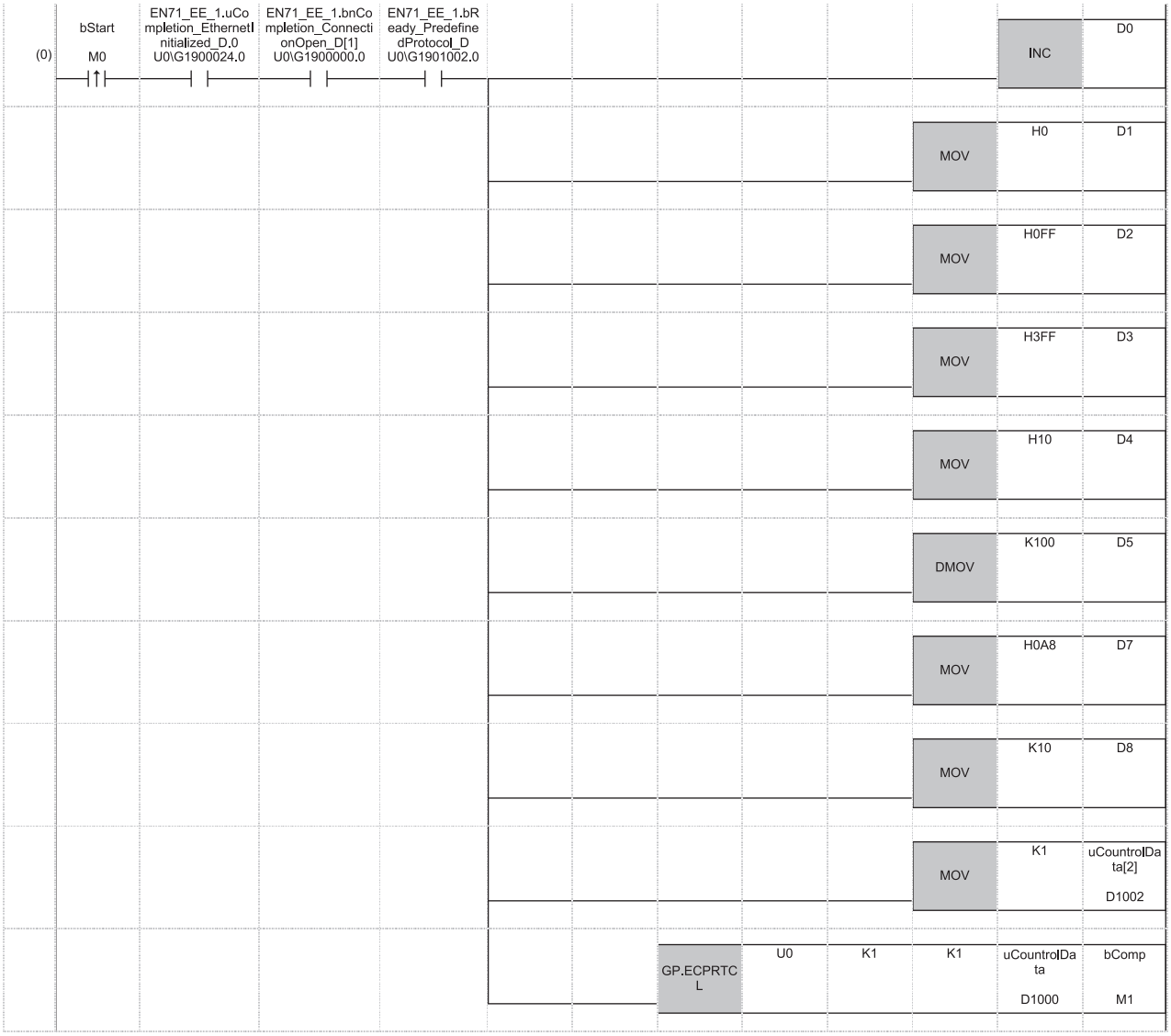
**Point**

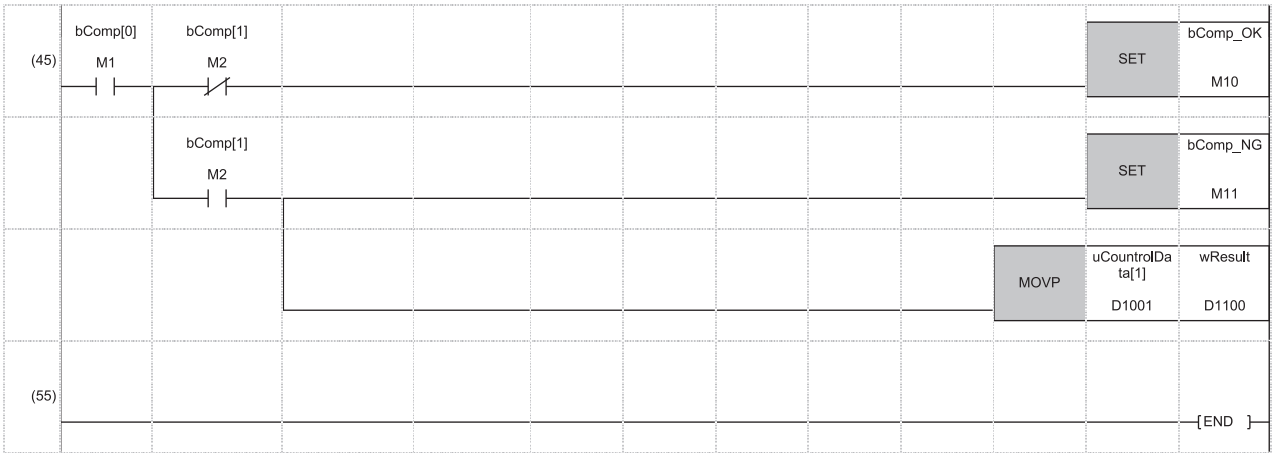
In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

## Program examples

| Classification | Label name | Description | Device |
|---|---|---|---|
| Module label | EN71_EE_1.bnCompletion_ConnectionOpen_D[1] | Open completion signal (connection No.1) | U0\G1900000.0 |
| | EN71_EE_1.uCompletion_EthernetInitialized_D.0 | Initial status | U0\G1900024.0 |
| | EN71_EE_1.bReady_PredefinedProtocol_D | Predefined protocol ready | U0\G1901002.0 |
| Label to be defined | Define global labels as shown below: | | |

| Label Name | Data Type | | Class | Assign (Device/Label) |
|---|---|---|---|---|
| bStart | Bit | ... | VAR_GLOBAL | M0 |
| bComp | Bit(0..1) | ... | VAR_GLOBAL | M1 |
| bComp_OK | Bit | ... | VAR_GLOBAL | M10 |
| bComp_NG | Bit | ... | VAR_GLOBAL | M11 |
| uCountrolData | Word [Unsigned]/Bit String [16-bit](0..17) | ... | VAR_GLOBAL | D1000 |
| wResult | Word [Signed] | ... | VAR_GLOBAL | D1100 |






(0) When 'bStart' (M0) is turned on, the protocol is executed by the dedicated instruction GP.ECPRTCL, and D100 to D109 of the receiving side CPU module is read and stored in D14 to D23 of the sending side CPU module.
'bComp_OK' (M10) is turned on when the protocol completed successfully.

D0 to D8 stores the following as a protocol setting data.

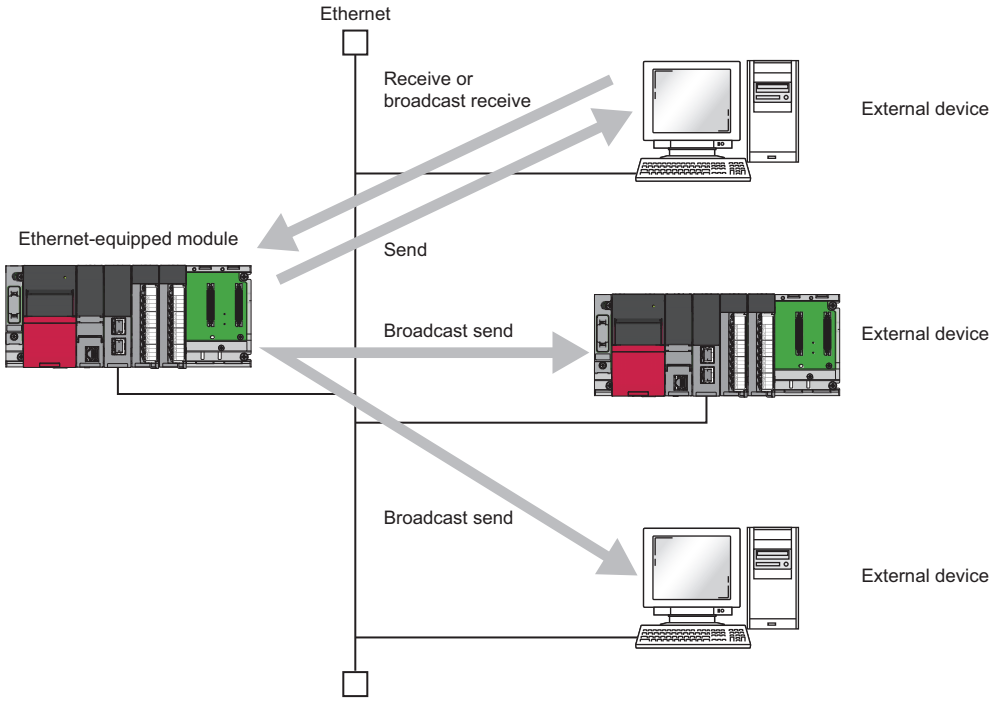
| Device number | Element name | Setting value | Description |
|---|---|---|---|
| D0 | Serial No. | — | The number increases one by one each time the 'bStart' (M0) is turned on. |
| D1 | Network No. | 0H | Because the access destination is the connected station (own station), 0H is stored as the request destination network number. |
| D2 | Station No. | FFH | Because the access destination is the connected station (own station), FFH is stored as the request destination station number. |
| D3 | Requested module I/O No. | 3FFH | Because the access destination is the receiving side CPU module (control CPU of the receiving side RJ71EN71), 3FFH is stored. |
| D4 | Monitoring timer | 10H | The monitoring timer is set to 4 seconds. (Unit: 250ms) |
| D5 to D6 | Head device No. | 100 | To read D100 to D109, start device number is set to 100. |
| D7 | Device code | A8H | To read D100 to D109, device code is set to A8H. |
| D8 | Number of device points | 10 | To read 10 points from D100 to D109, number of device points is set to 10. |

**Point**

In this program example, a receiving side program is not required.

# 1.4 Socket Communications

Using dedicated instructions, arbitrary data can be exchanged with an external device connected by Ethernet over TCP/IP or UDP/IP. (☞ Page 538 Socket communications instructions)
Use this for bidirectional communication one-on-one with an external device.




**Point**

For examples of socket communications, refer to the following.
MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup)

## Precautions

Socket communications cannot be used when the RJ71EN71 network type is set to "Q Compatible Ethernet".

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

***1.*** Select the external device to be connected in "Module List" and drag it to "List of devices" or "Device map area".

| External device name | Description |
|---|---|
| UDP Connection Module | Select to communicate with the external device using UDP/IP. |
| Active Connection Module | Select to perform the open processing to the external device from the Ethernet-equipped module (Active open) and communicate using TCP/IP. |
| Unpassive Connection Module | Select to receive the open processing from an unspecified external device (Unpassive open) and communicate using TCP/IP. |
| Fullpassive Connection Module | Select to receive the open processing from the specified external device (Fullpassive open) and communicate using TCP/IP. |

***2.*** Set "Communication Method" for the external device to "Socket Communication".

***3.*** Set the other parameters required for communication in the connection.

# Applicable dedicated instructions

For the dedicated instructions used for socket communications, refer to the following.

- ☞ Page 537 Open/close processing instructions
- ☞ Page 538 Socket communications instructions

# Applicable connections

The following connections can be used for data exchange with socket communications.

| Module | | Usable connections |
|---|---|---|
| RJ71EN71 | P1 connector | Connection No.17 to 64 |
| | P2 connector | Connection No.1 to 64*1 |
| RnENCPU (network part) | P1 connector | Connection No.17 to 64 |
| CPU module (built-in Ethernet port part) | | Connection No.1 to 16 |

*1 Although the engineering tool shows connection No.1 to 64 as available for port 2 of the RJ71EN71, the internal processing of the RJ71EN71 treats the connections as connection No.65 to 128. To use connections for port 2 of the RJ71EN71 for dedicated instructions, buffer memory, or other purposes, specify connection No.65 to 128.

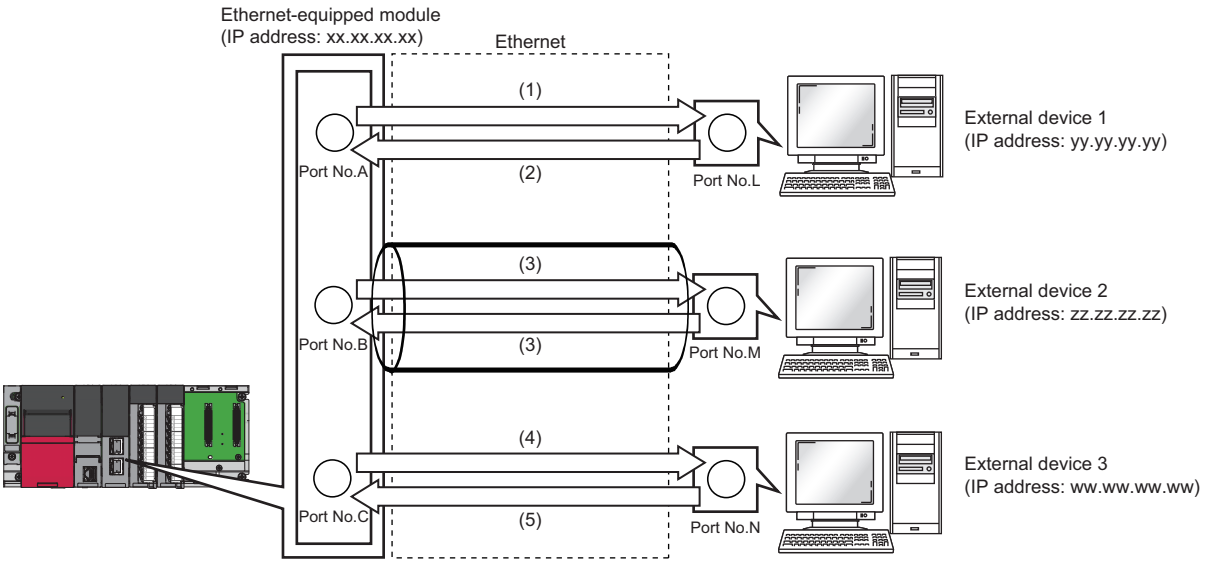
# Communication structure



With socket communications, port numbers that identify the communication are used to enable multiple communication sessions with the external device. These are used for both TCP/IP and UDP/IP.

For send: Specify send source Ethernet-equipped module's port number and the send destination external device's port number.

For receive: Specify the Ethernet-equipped module's port number, and read the data sent to that port.




(1) Sending UDP data from Ethernet-equipped module's port number A to external device 1's port number L
(2) Sending UDP data from external device 1's port number L to Ethernet-equipped module's port number A
(3) Sending data with TCP/IP connection
(4) Sending UDP data from Ethernet-equipped module's port number C to external device 3's port number N
(5) Sending UDP data from external device 3's port number N to Ethernet-equipped module's port number C

# Communications using TCP/IP

TCP/IP protocol establishes a connection between the external device's port number for reliable data exchange.
Check the following items before performing socket communications using TCP/IP.

- IP addresses and port numbers on external device side
- IP addresses and port numbers on the Ethernet-equipped module side
- Which side, the external device side or Ethernet-equipped module side, will open a connection (Active open or Passive open)

## TCP/IP connection operation

TCP/IP connection includes Active open and Passive open.
First, the side with the TCP/IP connection executes Passive open with the specified port number
The side with TCP/IP connection specifies the port number waiting in the Passive open side, and executes Active open.
This enables the TCP/IP connection, the connection is established, and data can be exchanged.
For details on Active open and Passive open, refer to the following.
(☞ Page 546 TCP/IP communications)

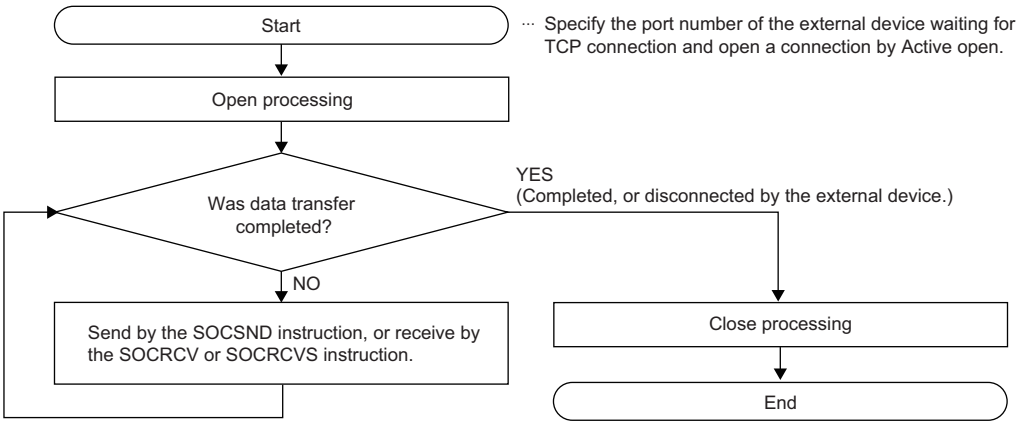
**Point**

The Active open and Passive open expression may differ according to the external device.

- Active open: TCP/IP connection side, client side, connect side, and others
- Passive open: TCP/IP connection wait side, server side, listen side, and others

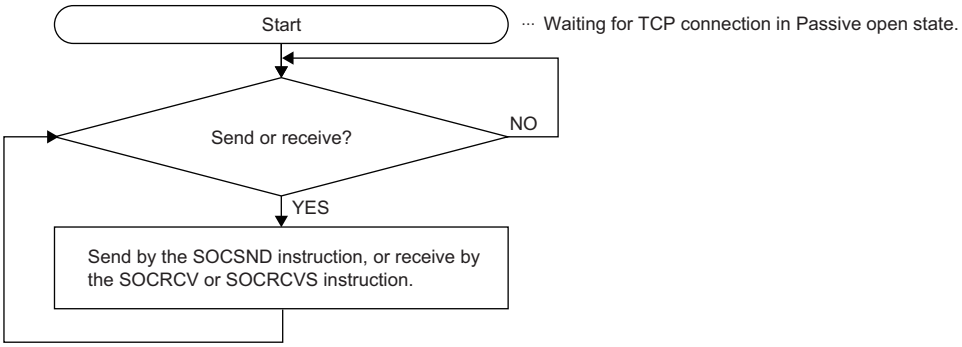
### ■Active open

The following figure shows the flow of data exchange using Active open.




### ■Passive open

The following figure shows the flow of data exchange using Passive open.



## Precautions for TCP/IP communications

### ■Conditions for closing

In addition to when close is requested from the external device, the TCP/IP communications processing will be closed in the following cases if 'Open completion signal' (Un\G1900000 to Un\G1900007) turns off.

- When alive check function times out
- When forced close is received from external device

### ■TCP/IP connection elements

The TCP/IP connection is managed with the following four elements. Only one connection containing the same four elements can be created at one time. To use multiple TCP/IP connections simultaneously, ensure that one of the four elements is different.

- IP address of the Ethernet-equipped module side
- Port number of the Ethernet-equipped module side
- IP address of the external device side
- Port number of the external device side

Note that when "Unpassive Connection Module" or "Fullpassive Connection Module" is selected, the port number of the Ethernet-equipped module side has to be different. In addition, one of the other three element also has to be different.

### ■Reconnecting with same connection

After closing the connection during TCP/IP communications, wait at least 500ms before reconnecting to a connection with the same external device (IP address), own station port number, and external device port number.

If a wait interval cannot be provided before reconnecting, changing the own station port number on the Active open side and connecting is recommended.

### ■Connection close wait time setting

With the TCP/IP communications, if the external device sends the request for closing the connection immediately after sending data, the system may fail to read the receive data that is immediately before the close processing on the Ethernet-equipped module side. In this case, the close processing timing of the Ethernet-equipped module can be delayed using 'Connection close wait time setting area for socket communications (TCP/IP)' (Un\G760, Un\G761[*1]) to read the receive data.

*1 The buffer memory address of the CPU module (built-in Ethernet port part)

For details on the connection close wait time setting, refer to the following.

☞ Page 533 Connection close wait time setting area for socket communications (TCP/IP)

**Restriction**

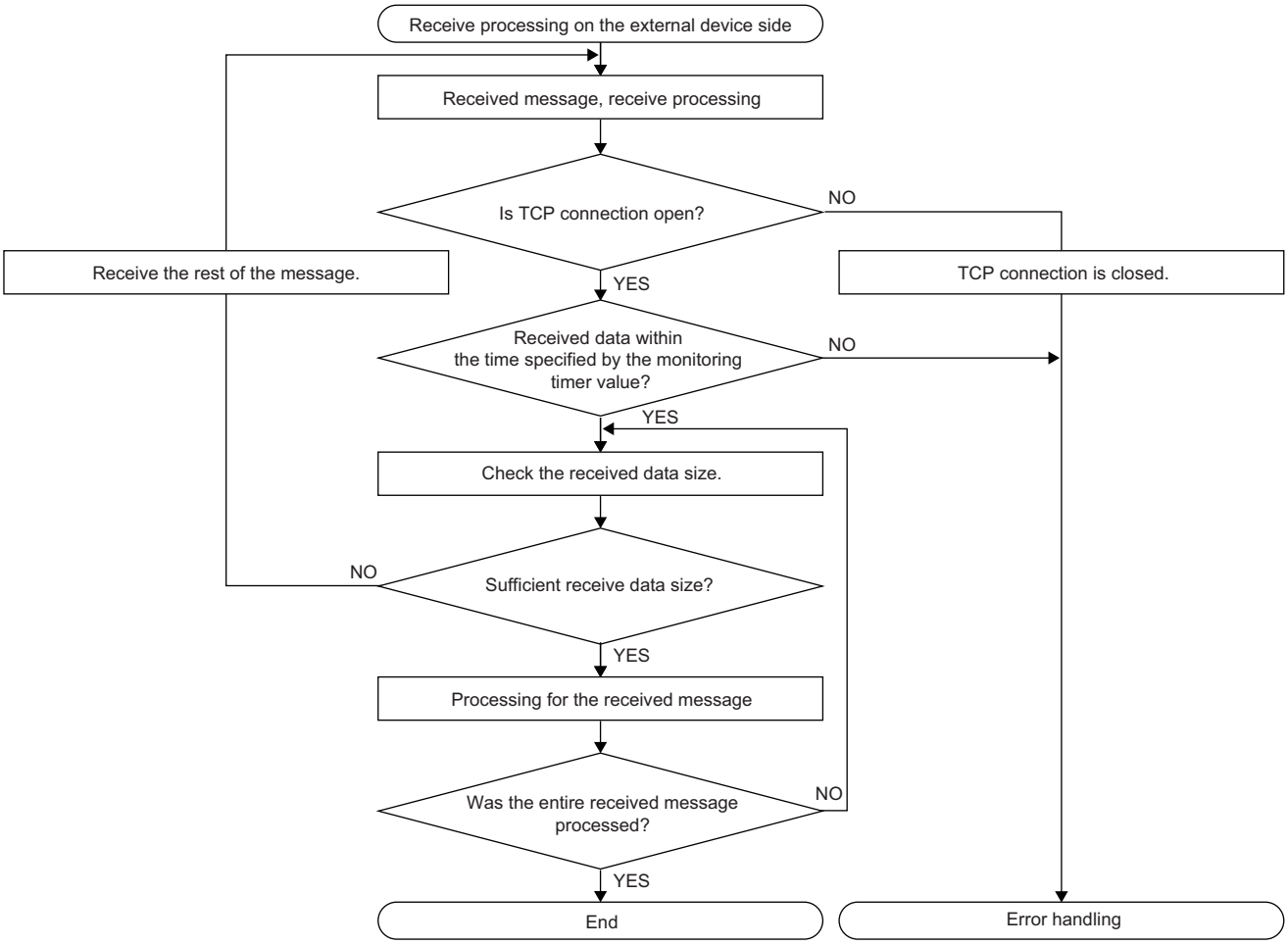
Before using the connection close wait time setting, check the version of the CPU module used. (☞ Page 586 Added and Enhanced Functions)

## ■Checking the receive data length

There is no concept of delimiting the communication data during communication with TCP/IP. Thus, the continuously sent data may be merged on the received side, or the data sent in a group may be split on the receive side. If necessary, the receiving side must check the receive data length and perform the processing.

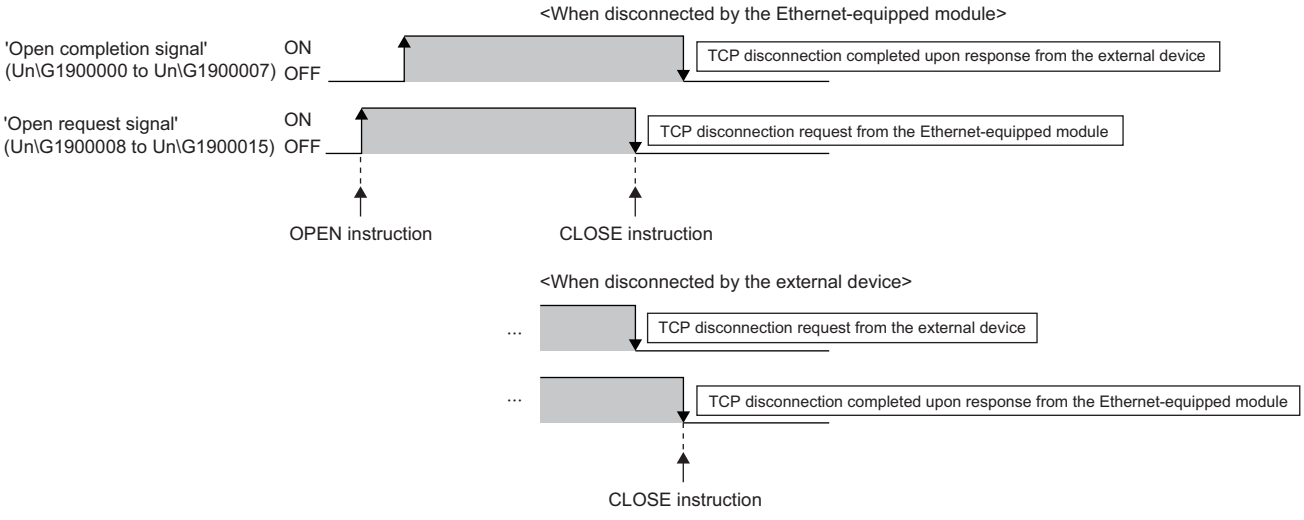
If the data length is determined when receiving with the Ethernet-equipped module side, using the fixed-length mode is recommended.

When receiving on the external device side, check the receive data length and perform the processing as shown below.



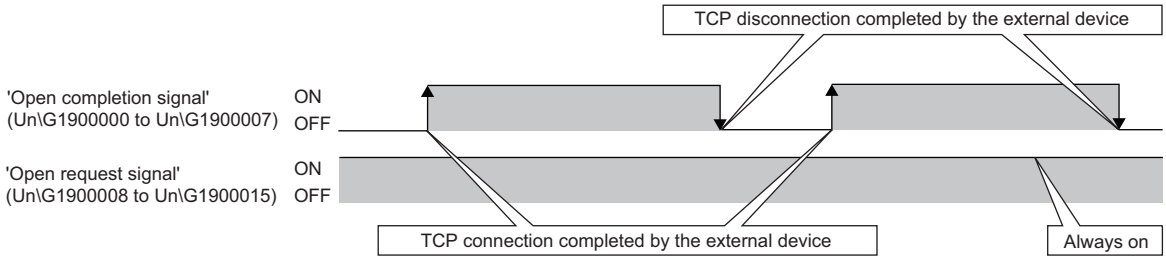

## ■Precautions for Active open

Use 'Open completion signal' (Un\G1900000 to Un\G1900007) and 'Open request signal' (Un\G1900008 to Un\G1900015) in the program to create an interlock circuit. The on/off timing for the open completion signal and open request signal is shown below.



### ■Precautions for Passive open

- Use 'Open completion signal' (Un\G1900000 to Un\G1900007) and 'Open request signal' (Un\G1900008 to Un\G1900015) in the program to create an interlock circuit. The on/off timing for the open completion signal and open request signal is shown below.




- When the CPU module is connected to the external device with Passive open, the connected external device's IP address or the external device's port number can be retrieved with the SP.SOCCINF instruction.
- Connect from the external device after the Ethernet-equipped module enters the open standby state. The TCP/IP connection request received from the external device between the time from CPU startup completion to open wait state causes an error, and force close connection is returned to the external device. In this case, wait for the Ethernet-equipped module to enter the open wait state and then retry from the external device.
- Do not execute the GP.CONCLOSE instruction or SP.SOCCLOSE instruction in the program. If the GP.CONCLOSE instruction or SP.SOCCLOSE instruction is executed, the open completion signal and open request signal for the corresponding connection will turn off. The close processing will be executed and send/receive will be disabled. To re-open a closed connection, execute the GP.CONOPEN instruction or SP.SOCOPEN instruction.

# Communications using UDP/IP

Communication with UDP/IP uses a simple protocol without order control or re-send control.
Check the following items before performing socket communications using UDP/IP.

- IP addresses and port numbers on external device side
- IP addresses and port numbers on the Ethernet-equipped module side

## Precautions for UDP/IP communications

- Loss of data, data arrival order interchange, and others could be occur. Consider using TCP/IP if there are problems.
- Even if the communication line between the CPU module and external device is not connected because of a connected cable disconnection and others, the data send processing may end normally. Thus, providing a communication procedure and sending/receiving data is recommended.
- Use the SP.SOCCSET instruction to change the external device with the CPU module.
- 'Open completion signal' (Un\G1900000 to Un\G1900007) and 'Open request signal' (Un\G1900008 to Un\G1900015) for the connection set to UDP/IP is always on.
- Do not execute the GP.CONCLOSE instruction or SP.SOCCLOSE instruction in the program. If the GP.CONCLOSE instruction or SP.SOCCLOSE instruction is executed, the open completion signal and open request signal for the corresponding connection will turn off. The close processing will be executed and send/receive will be disabled. To re-open a closed connection, execute the GP.CONOPEN instruction or SP.SOCOPEN instruction.
- Even if 'Open completion signal' (Un\G1900000 to Un\G1900007) is turned on, data sending may fail. If data sending fails, send the data again.

# Broadcast communications

Broadcast communications is a communication method that does not specify the external device. Data is exchanged between all Ethernet-equipped module stations and external devices on the same Ethernet to which the Ethernet-equipped modules are connected.

| Item | Description |
|---|---|
| Broadcast sending | The same data is sent to all Ethernet devices on the same Ethernet. |
| Broadcast receiving | The data sent with broadcast send is received. |

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

1. Select "UDP Connection Module" in "Module List" and drag it to "List of devices" or "Device map area".
2. Set "Communication Method" for the external device to "Broadcast Send" or "Broadcast Receive".
3. Set the other parameters required for communication in the connection.

## Precautions

- Decide the dedicated port number for broadcast communication in the system, and use that number.
- Access via a router is not permitted when using broadcast send.
- The external device connected on the same Ethernet must carry out a read/discard processing if the message received with broadcast receiving is not required.
- When the CPU module receives the data with broadcast receiving, the IP address of the sending external device and the external device's port number can be retried with the SP.SOCCINF instruction.

# Precautions



This section describes the precautions for exchanging data with socket communications.

## Port number

Port numbers 1 to 1023 are typically reserved port numbers (WELL KNOWN PORT NUMBERS) and 61440 to 65534 are used by other communication functions, so 1024 to 4999 or 5010 to 61439 should be used for the own station port numbers. In addition, if ports are used for other functions, do not use the ports for this function. (☞ Page 571 Port Numbers Used by Ethernet-equipped Module)

## Reading received data

If 'Socket/fixed buffer reception status signal' (Un\G1900016 to Un\G1900023) is on, read the received data. The communication could be affected if large amounts of data are not read out for a while.

## Accessing a file during communication

The CPU module prioritizes the file access processing over the Ethernet communication processing. Thus, if the file is accessed with FTP, the engineering tool, and so on, during socket communications, the socket communications processing could be delayed.

To access a file while monitoring the response time with the external device with socket communications, add the time required for accessing the file to the monitoring time.

## Module FB and dedicated instruction

- When performing the open processing using the module FB or dedicated instruction, start sending and receiving data after the module FB or dedicated instruction is completed.
- When executing multiple module FBs or dedicated instructions simultaneously for one connection, refer to the following.

☞ Page 536 When dedicated instructions or others are simultaneously executed for the same connection

# 1.5 Communications Using the Fixed Buffer

Communications using the fixed buffer uses TCP/IP and UDP/IP to send and receive arbitrary data with the external device connected to the Ethernet with dedicated instructions in the same manner as exchanging data with socket communications. Exchanges arbitrary data between the CPU module and external device using the fixed buffer of the RJ71EN71 and the RnENCPU (network part).

The following table lists the differences with socket communications are given below.

| Item | Differences | |
|---|---|---|
| | Socket communications | Communications using the fixed buffer |
| Connection send/receive | Send/receive is possible with one connection | Specify send or receive for one connection (Two connections are required for send and receive) |

## Precautions

The CPU module (CPU part for the RnENCPU) cannot exchange data with the fixed buffer.

## Differences between the "Procedure Exist" and "No Procedure" control methods

"Procedure Exist" and "No Procedure" control methods can be used for fixed buffer communication. The following table lists the differences between "Procedure Exist" and "No Procedure".

| Item | Differences | |
|---|---|---|
| | Procedure Exist | No Procedure |
| Message format | Data is sent and received with the predetermined data format. | Data is sent and received according to the external device's message format. |
| Response for received data | A response is sent for the received data. | No response is sent for the received data. |
| Data Code | Data can be exchanged with binary code or ASCII code. | Data is exchanged only with binary codes. |
| Data length specified with dedicated instructions | Specify with a number of words. | Specify with a number of bytes. |
| Amount of application data per data exchange session*1 | Maximum 5113 words (binary code)<br>Maximum 2556 words (ASCII code) | Maximum 10238 bytes |

*1 The following value is used if the RJ71EN71 network type is set to "Q Compatible Ethernet".
Procedure Exist: Maximum 1017 words (binary code), maximum 508 words (ASCII code)
No Procedure: Maximum 2046 bytes

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

***1.*** Select the external device to be connected in "Module List" and drag it to "List of devices" or "Device map area".

| External device name | Description |
|---|---|
| UDP Connection Module | Select to communicate with the external device using UDP/IP. |
| Active Connection Module | Select to perform the open processing to the external device from the Ethernet-equipped module (Active open) and communicate using TCP/IP. |
| Unpassive Connection Module | Select to receive the open processing from an unspecified external device (Unpassive open) and communicate using TCP/IP. |
| Fullpassive Connection Module | Select to receive the open processing from the specified external device (Fullpassive open) and communicate using TCP/IP. |

***2.*** Set "Communication Method" for the external device to "Fixed Buffer (Procedure Exist)" or "Fixed Buffer (No Procedure)".

***3.*** Set the "Fixed Buffer Send/Receive Setting".

***4.*** Set the other parameters required for communication in the connection.

# Applicable dedicated instructions

For the dedicated instructions used for communications using the fixed buffer, refer to the following.

- ☞ Page 537 Open/close processing instructions
- ☞ Page 538 Instructions for communications using the fixed buffer

GP.CONOPEN and GP.CONCLOSE of the open/close processing instructions are not available when "Q Compatible Ethernet" is set to network type.

# Applicable connections

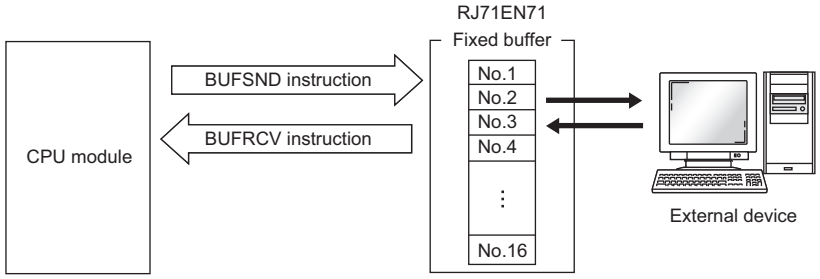
The connections No.1 to 16 of the P1 connector can be used for the communications using the fixed buffer.
Communications using the fixed buffer cannot be used with the P2 connector.

# Communication structure

This section describes the mechanism of communication with fixed buffer communication.

## Data flow

Dedicated instructions are used to send and receive data in fixed buffer communication (☞ Page 73 Applicable dedicated instructions)
With "Procedure Exist", the CPU module and external device exchange data one-on-one. A handshake is established with the external device when sending data form the CPU module and receiving data from the external device.
With "No Procedure", data is sent from the CPU module and received from the external device without a procedure.

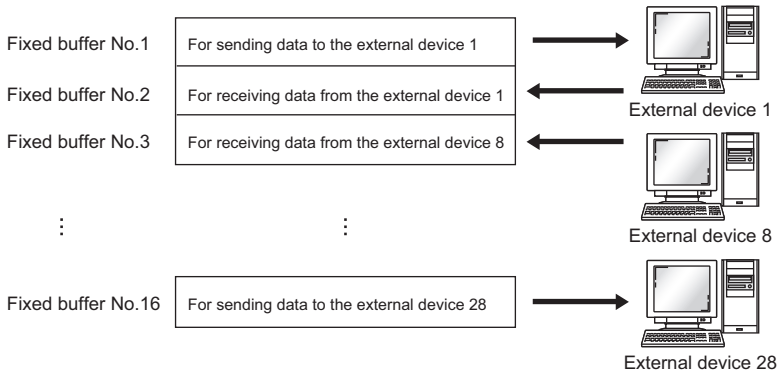


## External devices capable of data exchange

Data can be exchanged with the following external devices.

- Devices in Ethernet to which the RJ71EN71 is connected
- Devices in Ethernet to which the RnENCPU (network part) is connected
- Device connected via router

As the following figure shows, the external device for communication and the working application (for send/receive and "Procedure Exist"/"No Procedure") are set in "External Device Configuration" under "Basic Settings" using each fixed buffer (No.1 to No.16) to fix the external device for each buffer.




Pay attention to the following when changing the external device.

- During TCP/IP communications, the external device can be changed only when a connection is not established with the external device (when the open completion signal is off).
- During UDP/IP communications, the external device can be changed regardless of the connection status with the external device.
- When changing the external device, do not use the pairing open or alive check function.

## Processing during data send/receive

### ■During data send

When the BUFSND instruction is executed, the RJ71EN71 and the RnENCPU (network part) send data from the corresponding fixed buffer to the external device set in the specified connection.

### ■During data reception

If the data is being received from an external device set in the specified connection, the RJ71EN71 and the RnENCPU (network part) process the reception.

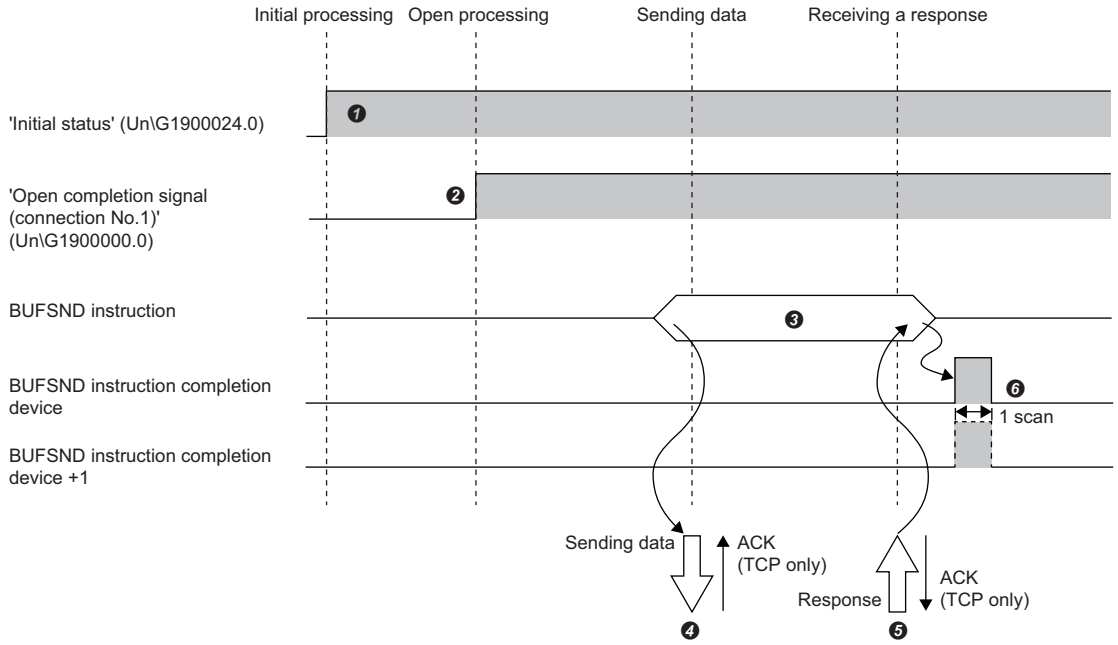
If data is being received from an external device that is not set in the specified connection, the RJ71EN71 and the RnENCPU (network part) will ignore the received data.

# Send procedure

The following figure shows the processing when sending data from the RJ71EN71 or the RnENCPU (network part) to the external device.

## Procedure exists

The following figure shows the send processing for the fixed buffer No.1 area corresponding to the connection No.1.




❶ Normal completion of the initial processing is checked. ('Initial status' (Un\G1900024.0): On)

❷ A connection is established between the external device and the RJ71EN71 or the RnENCPU (network part), and normal completion of the connection No.1 open processing is checked. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)

❸ The BUFSND instruction is executed. (Sends data)

❹ The data length amount of send data in the fixed buffer No.1 area is sent to the external device.

❺ When the external device receives the data from the RJ71EN71 or the RnENCPU (network part), returns a response to the RJ71EN71 or the RnENCPU (network part).

❻ When a response is received from the external device, the RJ71EN71 or the RnENCPU (network part) finishes data send. If the response is not returned within the response monitor timer value, a data send error occurs.*1 If the data send completes abnormally, execute the BUFSND instruction again and start the send processing.

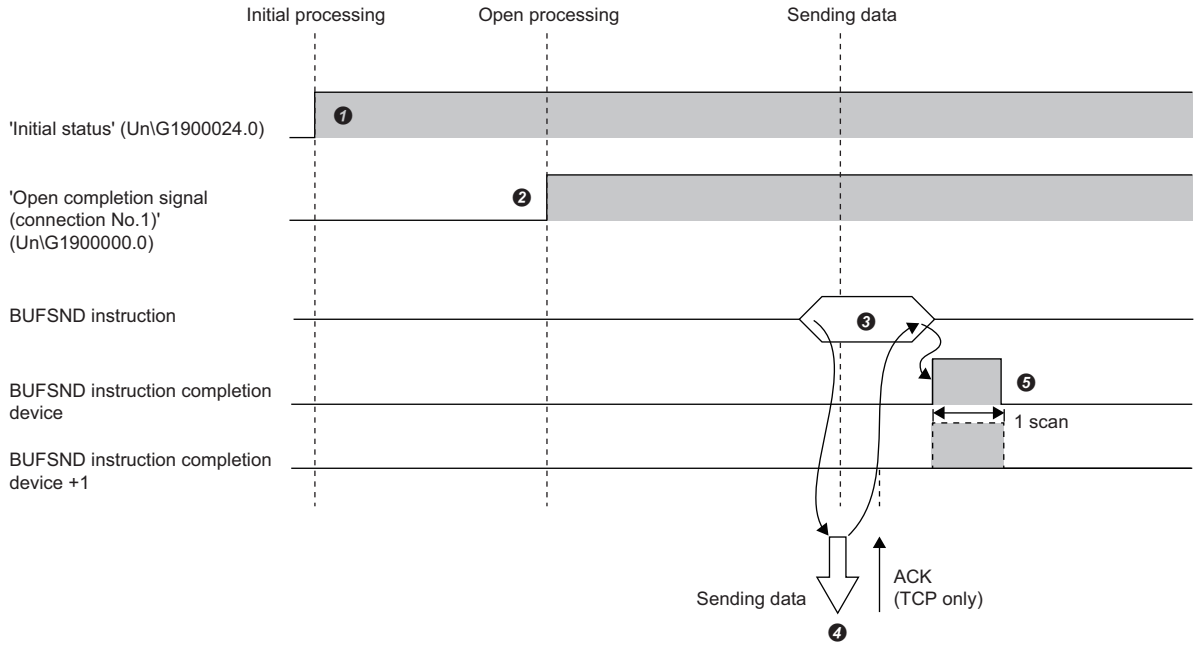
*1 Adjust the monitor timer value with the parameters. (☞ Page 401 Timer Settings for Data Communication)

**Point**

- The details of the open setting are enabled at the rising edge of the RJ71EN71 and the RnENCPU (network part) open completion signal.
- Send the next data (command) after the data exchange has been completed for the previous data (command) send.
- When sending or receiving data to multiple external devices, the data can be sent sequentially. However, to avoid communication trouble, it is recommended to switch the external device and send/receive the data. When using a connection opened with UDP/IP, the setting value in the communication address setting area can be changed before sending or receiving to switch the external device.

## No procedure

The following figure shows the send processing for the fixed buffer No.1 area corresponding to the connection No.1.




❶ Normal completion of the initial processing is checked. ('Initial status' (Un\G1900024.0): On)
❷ A connection is established between the external device and the RJ71EN71 or the RnENCPU (network part), and normal completion of the connection No.1 open processing is checked. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
❸ The BUFSND instruction is executed. (Sends data)
❹ The data length amount of send data in the fixed buffer No.1 area is sent to the external device.
❺ When a response is received from the external device, the RJ71EN71 or the RnENCPU (network part) finishes data send. If the data send completes abnormally, execute the BUFSND instruction again and start the send processing.

**Point**

During UDP/IP communications, if the internal processing of the RJ71EN71 and the RnENCPU (network part) completed normally, the data send processing may end normally even if the communication line between the CPU module and external device is disconnected because of a connection cable disconnection or other causes. Thus, providing a communication procedure and sending/receiving data is recommended.

# Receive procedure

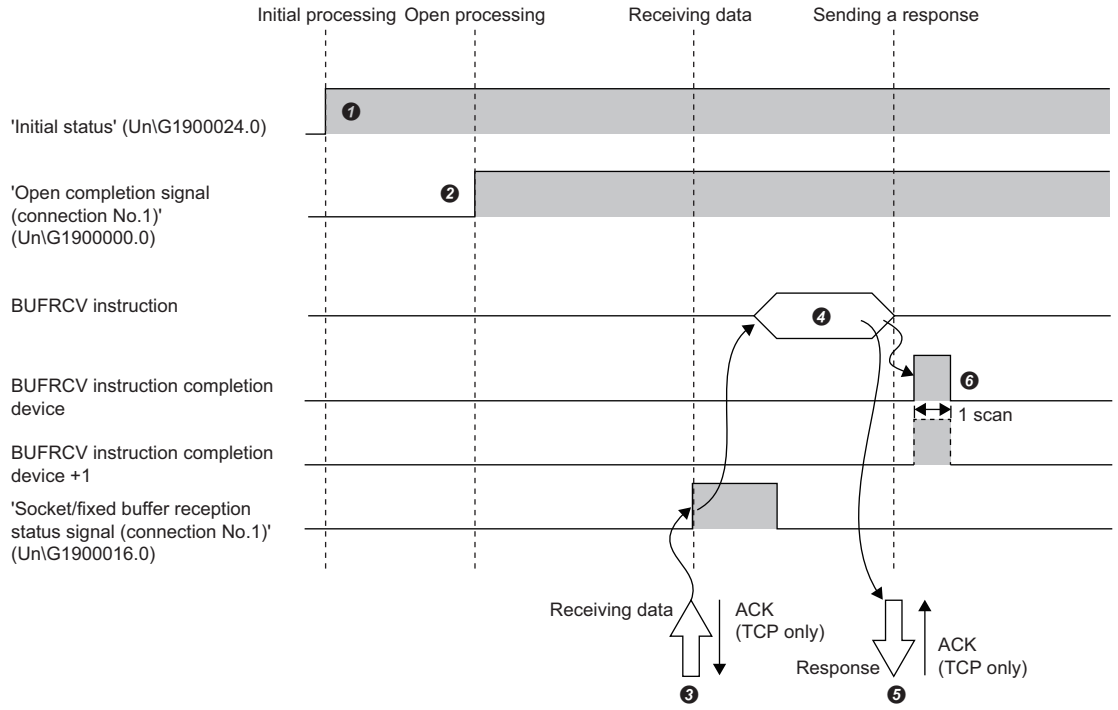


The following figure shows the processing for the RJ71EN71 or the RnENCPU (network part) to receive data from the external device. The following receive methods can be used.

- Receiving with main program (BUFRCV instruction)
- Receiving with interrupt program (BUFRCVS instruction)

## Receiving with main program (procedure exists)

The following figure shows the receive processing for the fixed buffer No.1 area corresponding to the connection No.1.



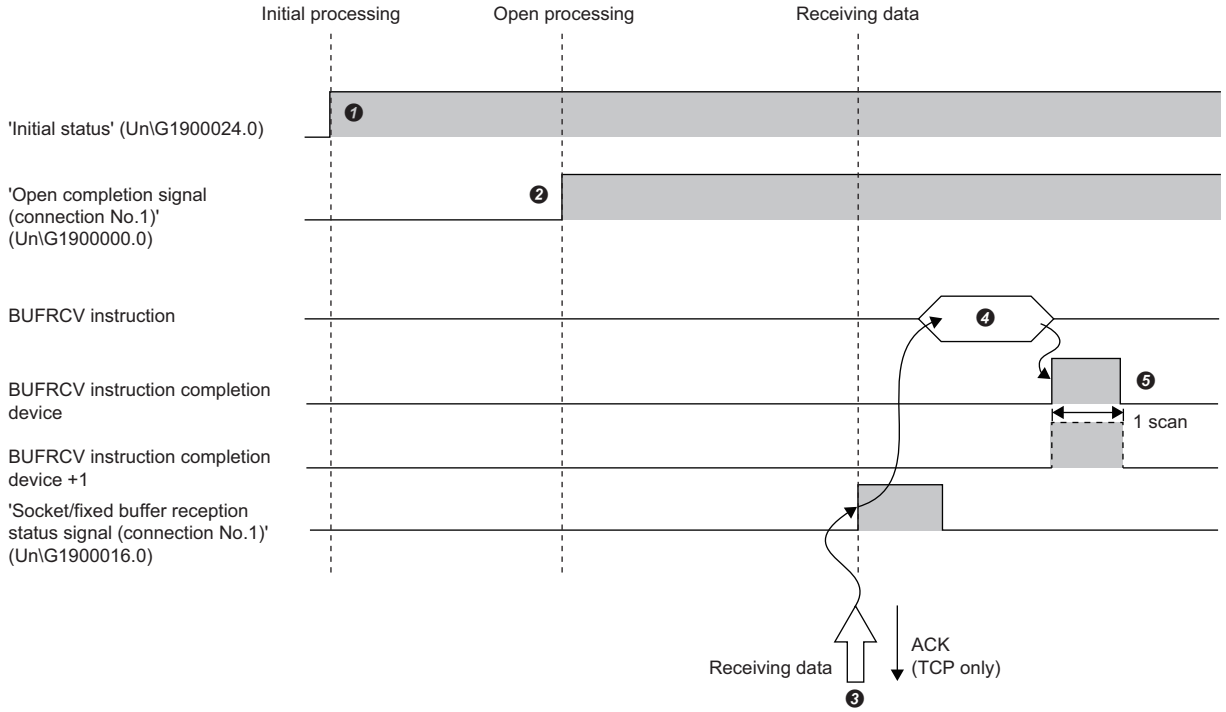

❶ Normal completion of the initial processing is checked. ('Initial status' (Un\G1900024.0): On)
❷ A connection is established between the external device and the RJ71EN71 or the RnENCPU (network part), and normal completion of the connection No.1 open processing is checked. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
❸ Data is received from the external device. ('Socket/fixed buffer reception status signal (connection No.1)' (Un\G1900016.0): On)
❹ The BUFRCV instruction is executed, and the receive data length and receive data are read from the fixed buffer No.1. ('Socket/fixed buffer reception status signal (connection No.1)' (Un\G1900016.0): Off)
❺ When reading of the receive data length and receive data is completed, a response is returned to the external device.
❻ The receive processing ends. If the data reception completes abnormally, execute the BUFRCV instruction again and start the receive processing.

**Point**

- The details of the open setting are enabled at the rising edge of the RJ71EN71 and the RnENCPU (network part) open completion signal.
- Execute the BUFRCV instruction when the socket/fixed buffer reception status signal changes from off to on.
- The socket/fixed buffer reception status signal does not turn on when abnormal data is received. In addition, data is not stored in the fixed buffer No.1 area.

## Receiving with main program (no procedure)

The following figure shows the receive processing for the fixed buffer No.1 area corresponding to the connection No.1.




❶ Normal completion of the initial processing is checked. ('Initial status' (Un\G1900024.0): On)
❷ A connection is established between the external device and the RJ71EN71 or the RnENCPU (network part), and normal completion of the connection No.1 open processing is checked. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
❸ Data is received from the external device. ('Socket/fixed buffer reception status signal (connection No.1)' (Un\G1900016.0): On)
❹ The BUFRCV instruction is executed, and the receive data length and receive data are read from the fixed buffer No.1. ('Socket/fixed buffer reception status signal (connection No.1)' (Un\G1900016.0): Off)
❺ The receive processing ends. If the data reception completes abnormally, execute the BUFRCV instruction again and start the receive processing.

**Point**

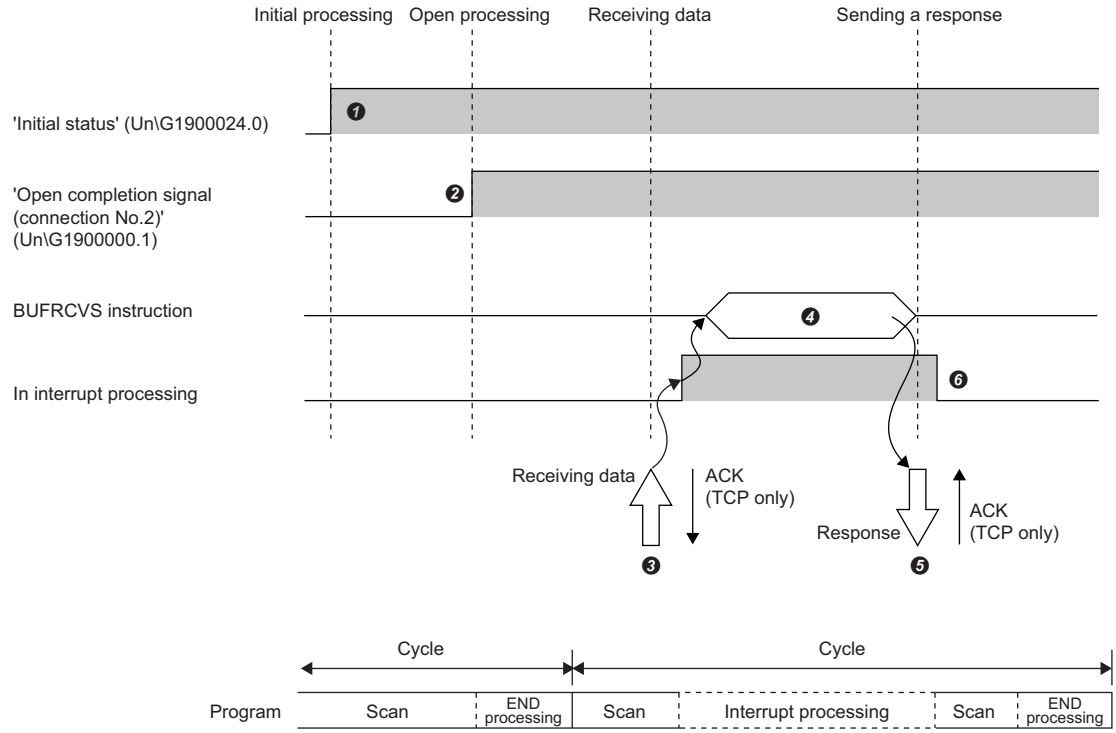
- The details of the open setting are enabled at the rising edge of the RJ71EN71 and the RnENCPU (network part) open completion signal.
- Execute the BUFRCV instruction when the socket/fixed buffer reception status signal changes from off to on.
- The socket/fixed buffer reception status signal does not turn on when abnormal data is received. In addition, data is not stored in the fixed buffer No.1 area.

## Reception with interrupt program (procedure exists)

Use the BUFRCVS instruction for receiving data with the interrupt program. The interrupt program is started when data is received from the external device. It enables the reading of receive data to the CPU module.

The interrupt settings are required to use the interrupt program. (☞ Page 414 Interrupt Settings)

The following figure shows the receive processing for the fixed buffer No.2 area corresponding to the connection No.2.




❶ Normal completion of the initial processing is checked. ('Initial status' (Un\G1900024.0): On)

❷ A connection is established between the external device and the RJ71EN71 or the RnENCPU (network part), and normal completion of the connection No.2 open processing is checked. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)

❸ The CPU module is requested to start the interrupt program, and data is received from the external device. ('Socket/fixed buffer reception status signal (connection No.2)' (Un\G1900016.1): On)

❹ The interrupt program starts. The BUFRCVS instruction is executed, and the receive data length and receive data are read from the fixed buffer No.2. ('Socket/fixed buffer reception status signal (connection No.2)' (Un\G1900016.1): Off)

❺ When reading of the receive data length and receive data is completed, a response is returned to the external device.*1

❻ Execution of the interrupt program ends, and execution of the main program resumes.

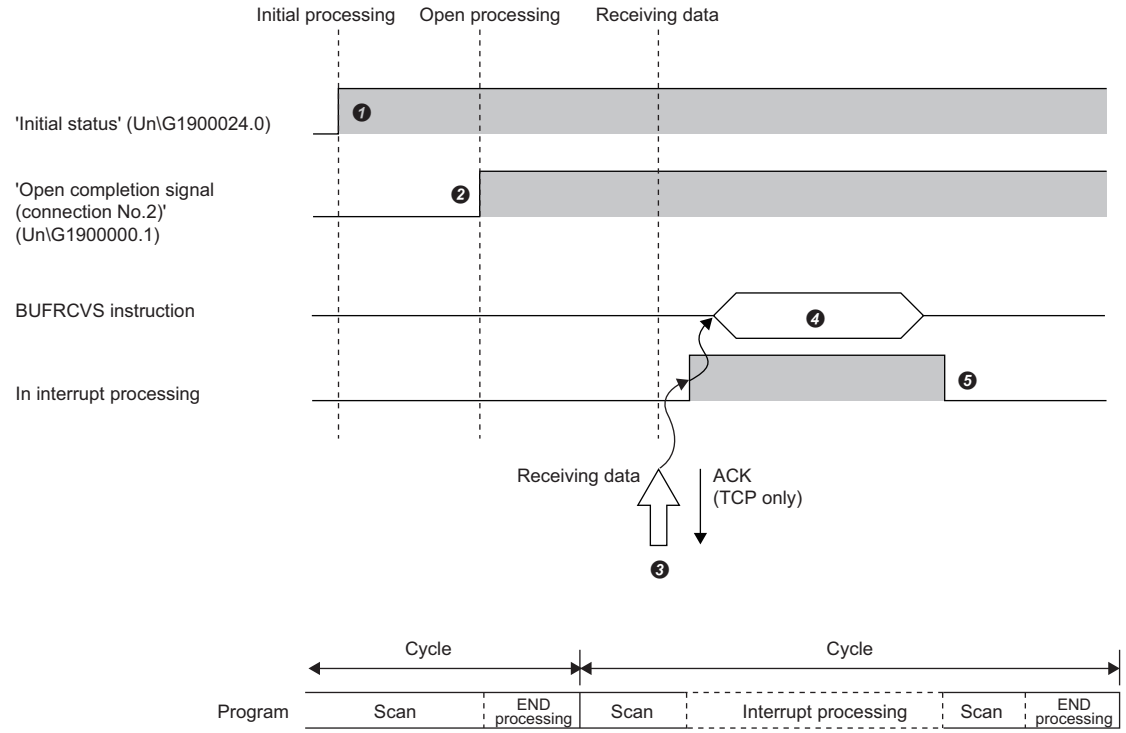
*1 A response is not returned when the execution completes abnormally.

## Reception with interrupt program (no procedure)

Use the BUFRCVS instruction for receiving data with the interrupt program. The interrupt program is started when data is received from the external device. It enables the reading of receive data to the CPU module.

The interrupt settings are required to use the interrupt program. (☞ Page 414 Interrupt Settings)

The following figure shows the receive processing for the fixed buffer No.2 area corresponding to the connection No.2.




❶ Normal completion of the initial processing is checked. ('Initial status' (Un\G1900024.0): On)

❷ A connection is established between the external device and the RJ71EN71 or the RnENCPU (network part), and normal completion of the connection No.2 open processing is checked. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)

❸ The CPU module is requested to start the interrupt program, and data is received from the external device. ('Socket/fixed buffer reception status signal (connection No.2)' (Un\G1900016.1): On)

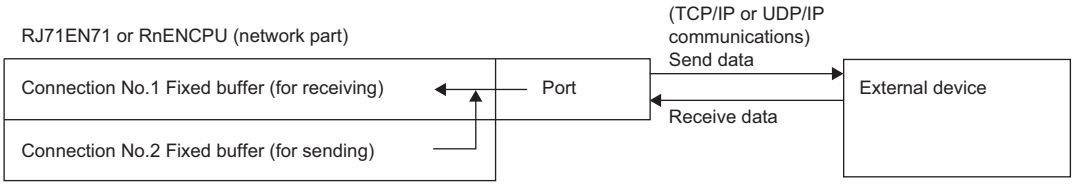
❹ The interrupt program starts. The BUFRCVS instruction is executed, and the receive data length and receive data are read from the fixed buffer No.2. ('Socket/fixed buffer reception status signal (connection No.2)' (Un\G1900016.1): Off)

❺ Execution of the interrupt program ends, and execution of the main program resumes.

# Pairing open



Pairing open is an opening method that pairs a fixed buffer communication reception connection and send connection, and establishes a connection using one port each from the own station and external device.
When pairing open is specified, data can be exchanged with two connections using an open processing for one port.




## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

1. Select the external device to be connected in "Module List" and drag it to "List of devices" or "Device map area".
2. Set "Communication Method" for the external device.
3. Set the "Fixed Buffer Send/Receive Setting" with the external device to "Pairing (Receive)".*1
4. Set the other parameters required for communication in the connection.
5. Select the same external device as step 1 from the "Module List", and drag to the next connection number of the external device set in step 1.
6. Set "Communication Method" for the external device to the same as the external device set in step 1.
7. Set the "Fixed Buffer Send/Receive Setting" with the external device to "Pairing (Send)".
8. Set the other parameters to the same values as the external device set in step 1.

*1 Set "Pairing (Receive)" to connection No.1 to 7, or No.9 to 15.

**Point**

- Only the external devices in the Ethernet to which the RJ71EN71 or the RnENCPU (network part) is connected and devices connected via a router can communicate data with pairing open.
- The open/close processing for the next connection (send connection) is executed automatically using the open/close processing on the receive connection side set to pairing open.

# Broadcast communications

Broadcast communications is a communication method that does not specify the external device. Data is exchanged between all Ethernet-equipped module stations and external devices on the same Ethernet to which the Ethernet-equipped modules are connected.

| Item | Description |
|---|---|
| Broadcast sending | The same data is sent to all Ethernet devices on the same Ethernet. |
| Broadcast receiving | The data sent with broadcast send is received. |

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

1. Select "UDP Connection Module" in "Module List" and drag it to "List of devices" or "Device map area".
2. Set "Communication Method" for the external device to "Broadcast Send" or "Broadcast Receive".
3. Set the other parameters required for communication in the connection.

## Precautions

- Decide the dedicated port number for broadcast communication in the system, and use that number.
- Access via a router is not permitted when using broadcast send.
- The external device connected on the same Ethernet must carry out a read/discard processing if the message received with broadcast receiving is not required.

# Precautions

This section describes the precautions for communications using the fixed buffer.

## Precautions for UDP/IP communications

Even if 'Open completion signal' (Un\G1900000 to Un\G1900007) is turned on, data sending may fail. If data sending fails, send the data again.

## Module FB and dedicated instruction

- When performing the open processing using the module FB or dedicated instruction, start sending and receiving data after the module FB or dedicated instruction is completed.
- When executing multiple module FBs or dedicated instructions simultaneously for one connection, refer to the following.

☞ Page 536 When dedicated instructions or others are simultaneously executed for the same connection

# Data format

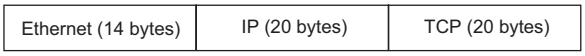


This section describes the data format used during communications using the fixed buffer.
The communication data is configured of the header and application data.

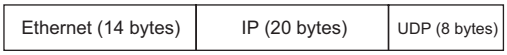
## Header

The header is for TCP/IP or UDP/IP. The RJ71EN71 or the RnENCPU (network part) automatically adds and deletes the header, the setting is not required.
The contents of the header are shown below.

- TCP/IP

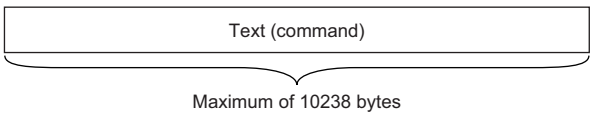



- UDP/IP




## Application data

If the communication procedure is "Fixed Buffer (No Procedure)", the application data expresses the following data code with binary codes. Data is exchanged with binary codes regardless of the communication data code setting.



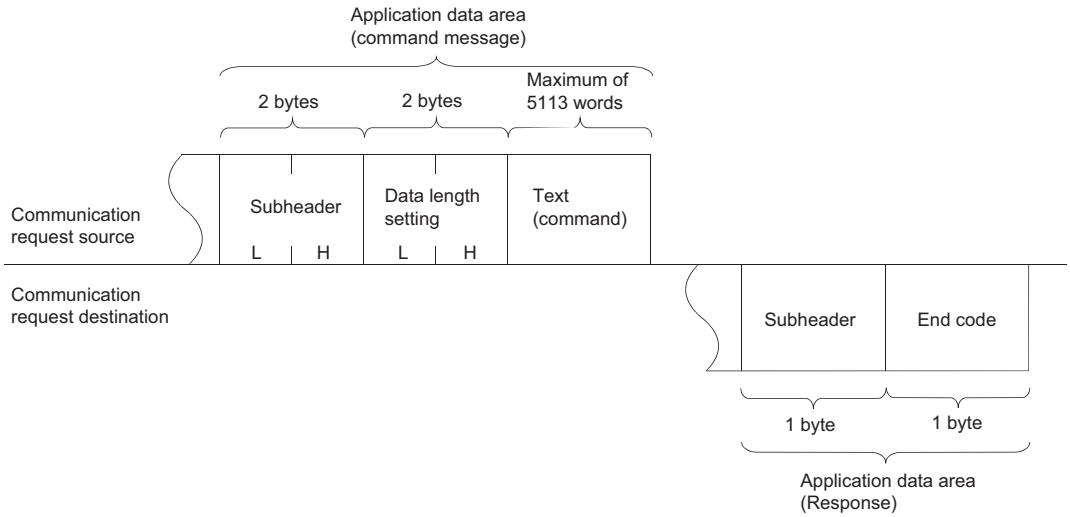

**Point**

With nonprocedural, the subheader and data length added for procedural are not used, so the data is all handled as valid text. The RJ71EN71 and the RnENCPU (network part) turn on the fixed buffer reception status signal after storing the size of the received message (packet) in the receive data length storage area. Providing a check procedure including the data length, data type code, and so on, in the message's application data is recommended so that the application data's byte size and data type can be seen on the receiving side.

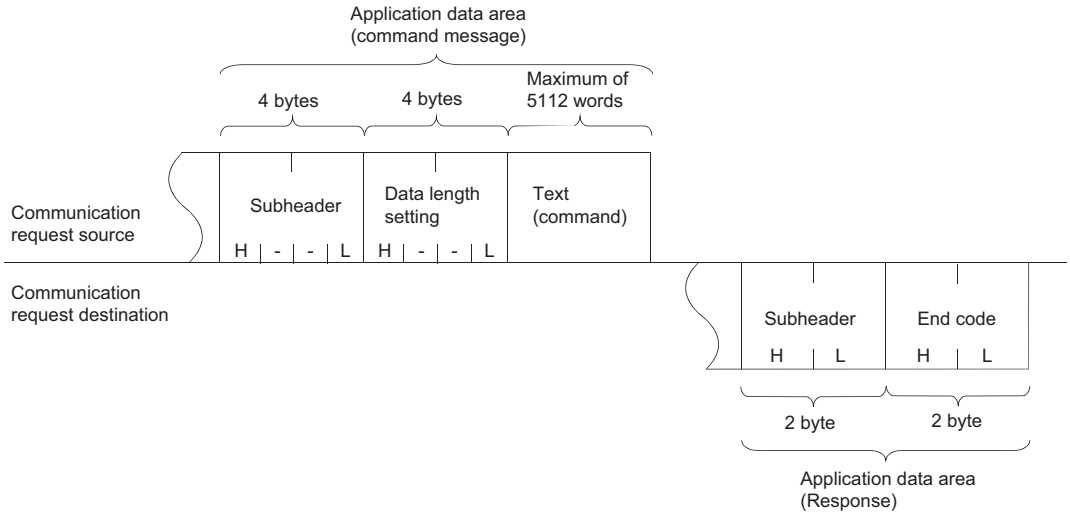
The following figure shows the configuration of the application data when the communication procedure is set to "Fixed Buffer (Procedure Exist)".

## ■Format

• When exchanging data with binary codes

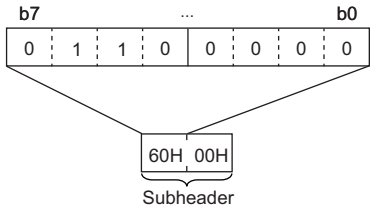
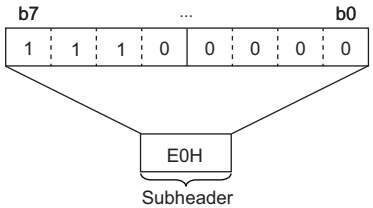
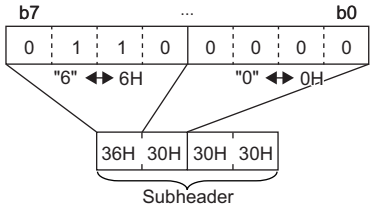
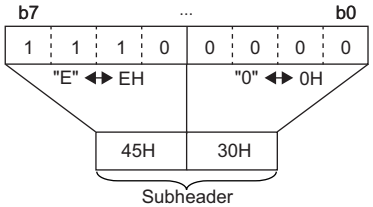



• When exchanging data with ASCII codes




## ■Subheader

The RJ71EN71 or the RnENCPU (network part) automatically adds and deletes the subheader, the setting is not required.

| Data Format | Command (external device → RJ71EN71 or RnENCPU (network part)) | Response (RJ71EN71 or RnENCPU (network part) → external device) |
|---|---|---|
| Binary code |  |  |
| ASCII code |  |  |

## ■Data length setting

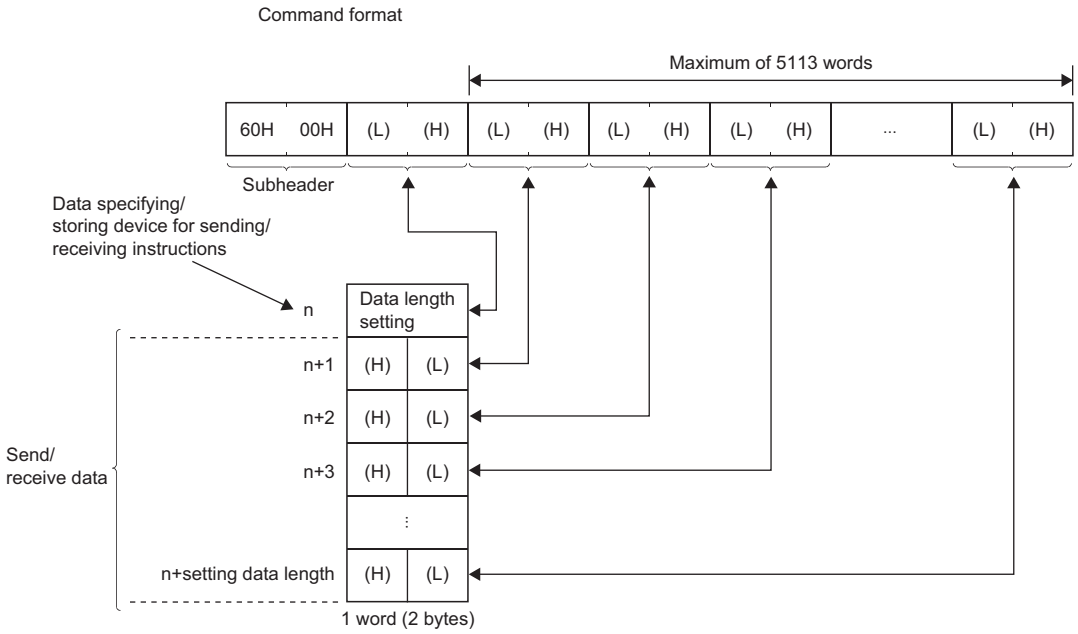
Shows the amount of data in the text (command) section.

- When exchanging data with binary codes: Maximum 5113 words
- When exchanging data with ASCII codes Maximum 2556 words

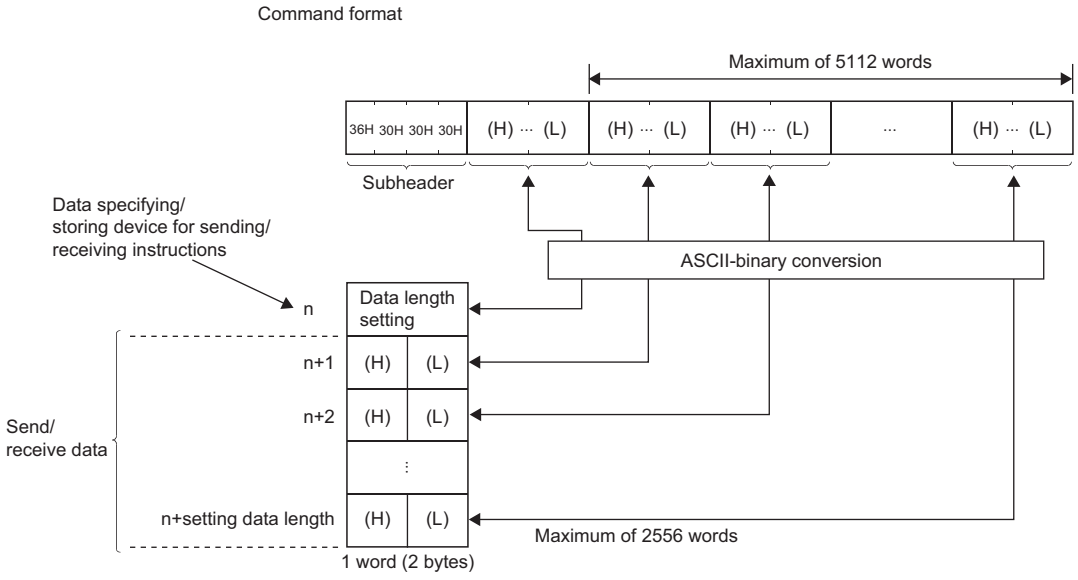
## ■Text (command)

Shows the format of the command/response.

- When exchanging data with binary codes




- When exchanging data with ASCII codes



## ■End code

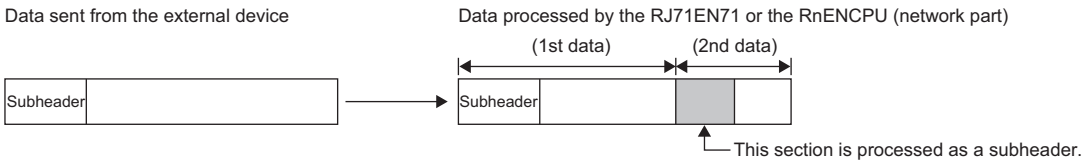
The error code is stored in the end command added to the response. (☞ Page 479 End Codes Returned to an External Device During Data Communications)

The end code is also stored in the BUFSND instruction and BUFRCV instruction completion status area (inside control data).

The following cases may apply if an error code for communications using the SLMP or random access buffer is stored even when executing communications using the fixed buffer.

| Description | Remedy method |
|---|---|
| The data length specified in the application data section of the message sent from the external device to the RJ71EN71 or the RnENCPU (network part) differs from the actual data size in the text section. | Specify the actual data size in the text section as the data length in the application data section. (Refer to the following descriptions.) |
| The subheader of the message sent from the external device to the RJ71EN71 or the RnENCPU (network part) is incorrect. | Review the subheader specified in the application data section. |

The communication data may be split and exchanged due to buffer limitations to the own station or external station. The data that is split and received is restored (reassembled) by the RJ71EN71 or the RnENCPU (network part) and exchanged. (The received data is restored (reassembled) based on the data length in the communication data.) The RJ71EN71 and the RnENCPU (network part) processing that take place when the data in the communication data is incorrect are shown below.

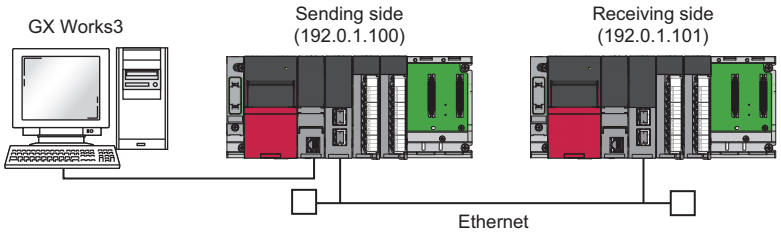
| Communication Method | Description |
|---|---|
| Fixed Buffer (Procedure Exist), Random Access Buffer | When data length specified immediately after subheader < text data volume<br>• The data immediately after the text corresponding to the data length specified immediately after the subheader is handled as the second message.<br>• The start of each statement becomes the subheader, so the RJ71EN71 and the RnENCPU (network part) execute a processing corresponding to the subheader code.<br>• If the subheader is not a code supported by the RJ71EN71 and the RnENCPU (network part), an abnormal completion response is sent to the external device.<br><br>In the above case, the code processed as the subheader with the uppermost bit set as 1 is returned as the response.<br>For example, if the command's subheader section is 65H, the response's subheader is E5H. |
| | When data length specified immediately after subheader > text data volume<br>The RJ71EN71 and the RnENCPU (network part) wait to receive the insufficient remaining data.<br>If the remaining data is received within the response monitor timer value, the RJ71EN71 and the RnENCPU (network part) execute a processing corresponding to the subheader code.<br>If the remaining data is not received within the response monitor timer value, the RJ71EN71 and the RnENCPU (network part) execute the following processing.<br>• The ABORT(RST) instruction is sent to the external device, and the line is closed.<br>• The error code is stored in 'Connection status storage area' (Un\G100 to Un\G163). |
| Fixed Buffer (No Procedure) | During nonprocedural, there is no message data length, so the received data is stored as is into the receive buffer area. Providing a check procedure including the data length, data type code, and so on, in the message's application data is recommended so that the application data's byte size and data type can be seen on the receiving side. |

# Example of communications using the fixed buffer



This section describes the socket communications examples using Active open of TCP/IP communications.

## System configuration



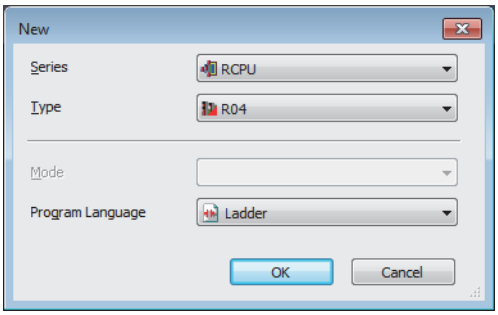

## Parameter settings

Connect the engineering tool to the CPU module and set the parameters.

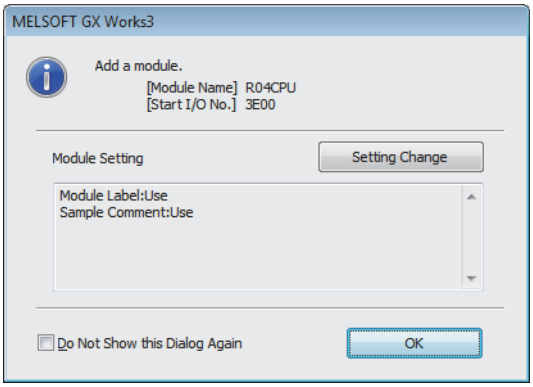
### ■Sending side

***1.*** Set the CPU module as follows.

[Project] ⇨ [New]



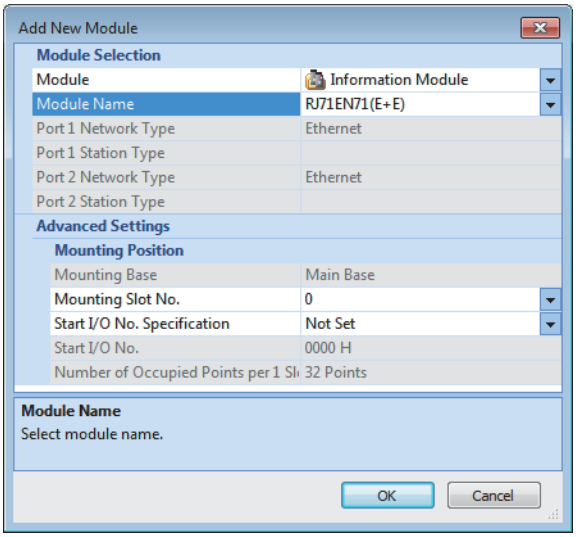

***2.*** Click the [Setting Change] button in the following window and set the item to use module labels.



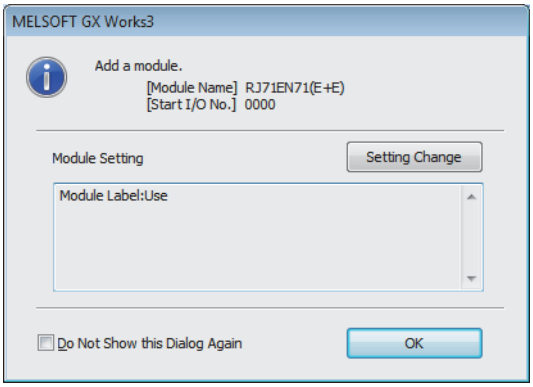

***3.*** Click the [OK] button to add the module labels of the CPU module.

**4.** Set the RJ71EN71 as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]

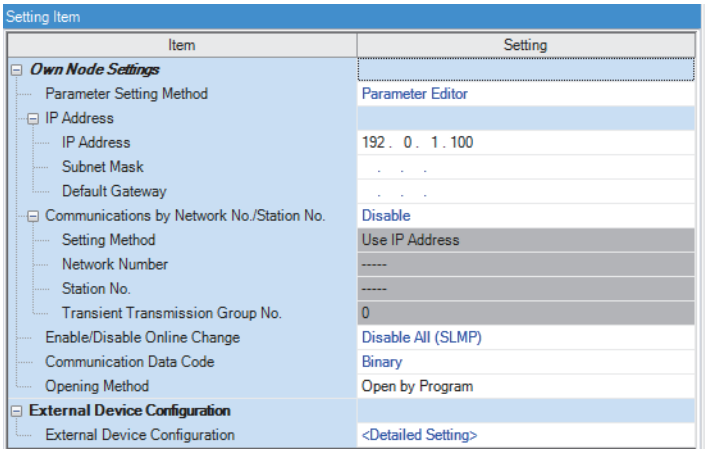



**5.** Click the [OK] button to add the module labels of the RJ71EN71.



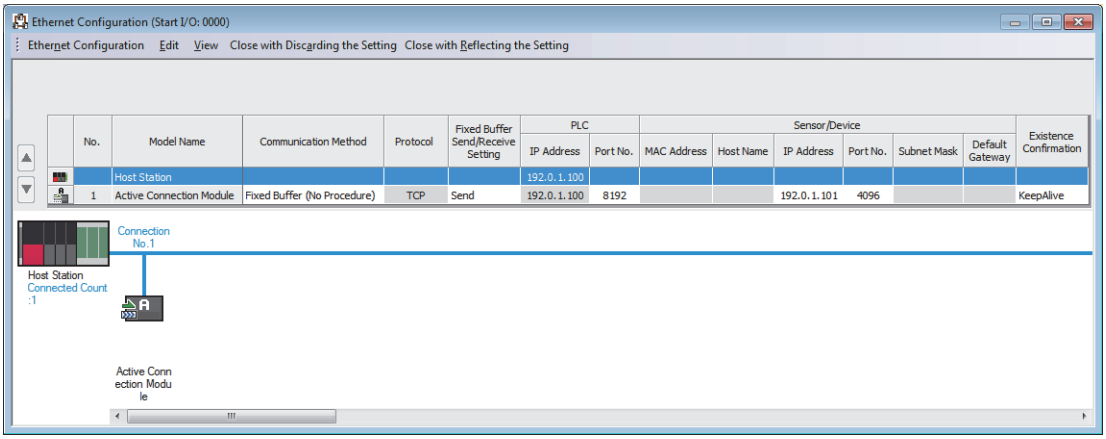

**6.** Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings]



**7.** Set the network configuration as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [External Device Configuration]




**8.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

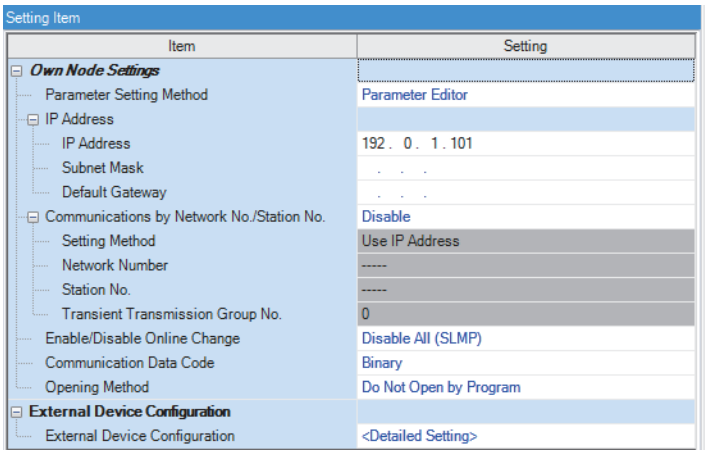
## ■Receiving side

**1.** Set the CPU module and add the module labels of the CPU module. The setting method of the CPU module and addition method of the module label are the same as those of when setting the sending side. (☞ Page 87 Sending side)

**2.** Set the RJ71EN71 and add the module labels of the RJ71EN71. The setting method of the RJ71EN71 and addition method of the module label are the same as those of when setting the sending side. (☞ Page 87 Sending side)

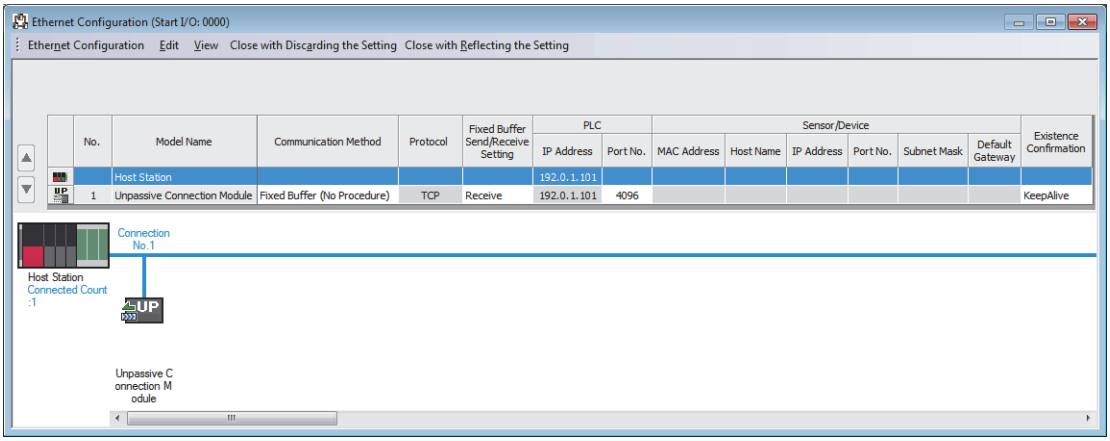
**3.** Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings]



**4.** Set the network configuration as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [External Device Configuration]




**5.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

## Program example (sending side)

### ■Labels

| Classification | Label name | Description | Device |
|---|---|---|---|
| Module label | RCPU.stSM.bAlways_ON | Always on | SM400 |
| | EN71_EE_1.stPort1.uStatus_HUB_Connection_D.0 | Connection status | U0\G5192.0 |
| | EN71_EE_1.bnCompletion_ConnectionOpen[1] | Open completion signal (connection No.1) | U0\G1900000.0 |
| | EN71_EE_1.bnStatus_ConnectionOpenExecution[1] | Open request signal (connection No.1) | U0\G1900008.0 |
| | EN71_EE_1.uCompletion_EthernetInitialized.0 | Initial status | U0\G1900024.0 |
| Label to be defined | Define global labels as shown below: | | |

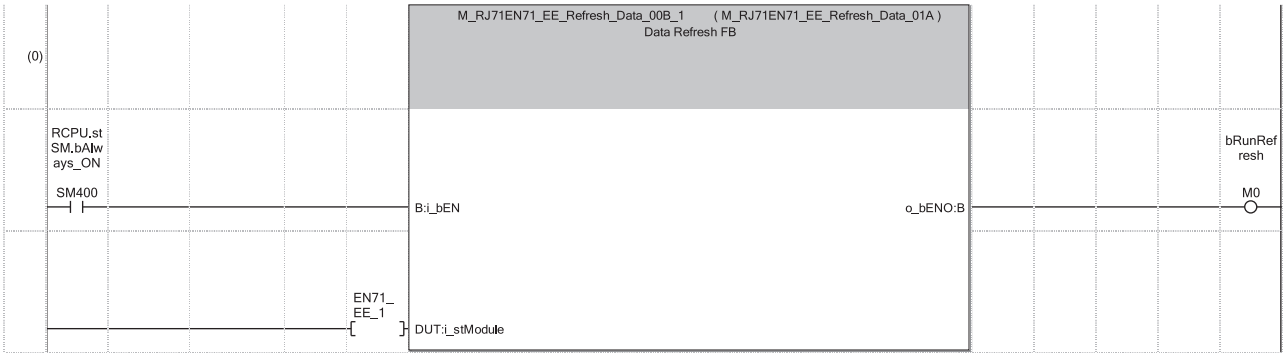
| Label Name | Data Type | | Class | Assign (Device/Label) |
|---|---|---|---|---|
| bRunRefresh | Bit | ... | VAR_GLOBAL | M0 |
| bStartOpen | Bit | ... | VAR_GLOBAL | M1 |
| bRunOpen | Bit | ... | VAR_GLOBAL | M2 |
| bOpen_OK | Bit | ... | VAR_GLOBAL | M3 |
| bOpen_NG | Bit | ... | VAR_GLOBAL | M4 |
| bStartSend | Bit | ... | VAR_GLOBAL | M5 |
| bRunSend | Bit | ... | VAR_GLOBAL | M6 |
| bSend_OK | Bit | ... | VAR_GLOBAL | M7 |
| bSend_NG | Bit | ... | VAR_GLOBAL | M8 |
| bStartClose | Bit | ... | VAR_GLOBAL | M13 |
| bRunClose | Bit | ... | VAR_GLOBAL | M14 |
| bClose_OK | Bit | ... | VAR_GLOBAL | M15 |
| bClose_NG | Bit | ... | VAR_GLOBAL | M16 |
| bStartOpenFB | Bit | ... | VAR_GLOBAL | M20 |
| bStartCloseFB | Bit | ... | VAR_GLOBAL | M21 |
| uOpenErrID | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D0 |
| uSendErrID | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D10 |
| uCloseErrID | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D30 |
| uSendData | Word [Unsigned]/Bit String [16-bit](0..3) | ... | VAR_GLOBAL | D1000 |

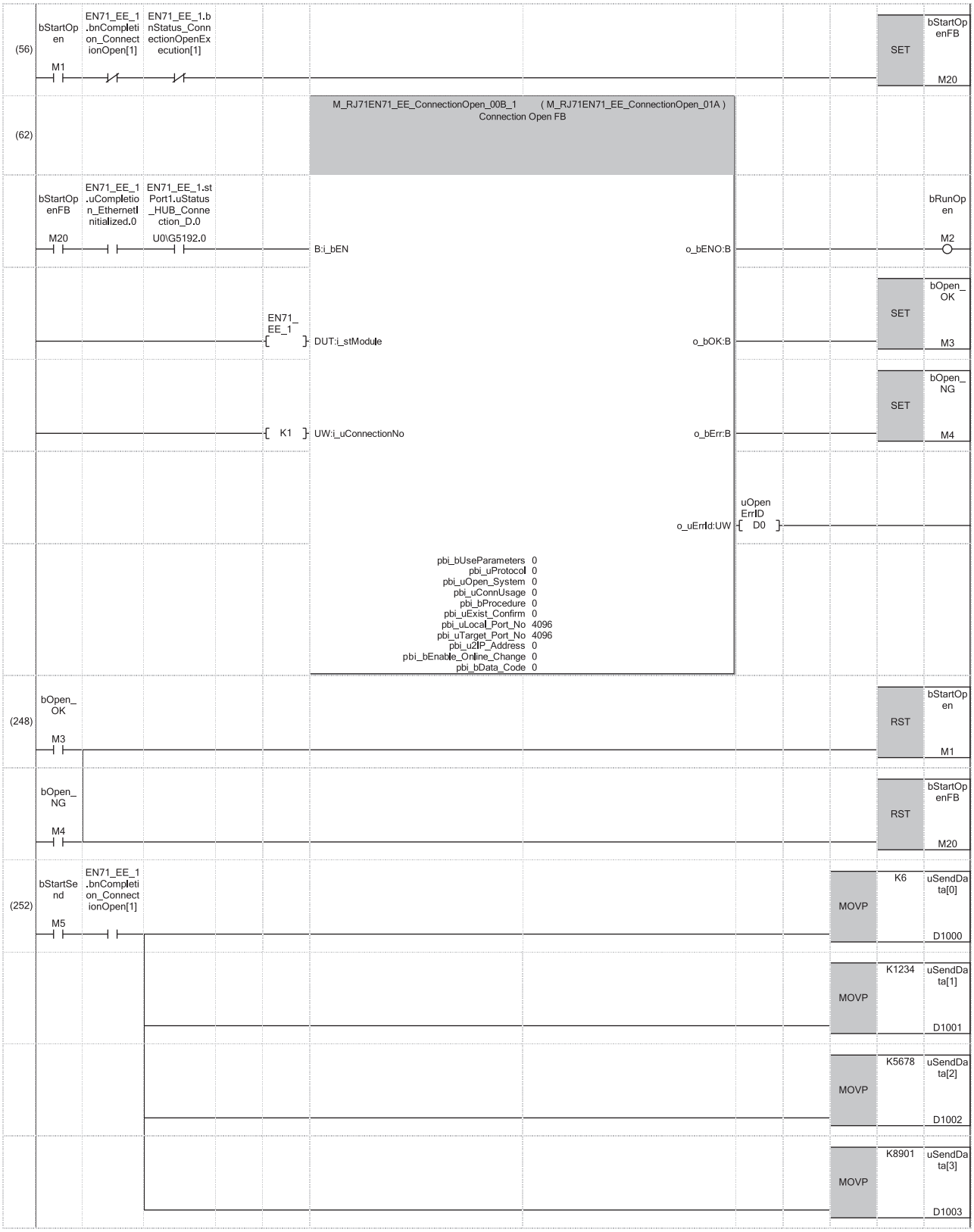
### ■FBs

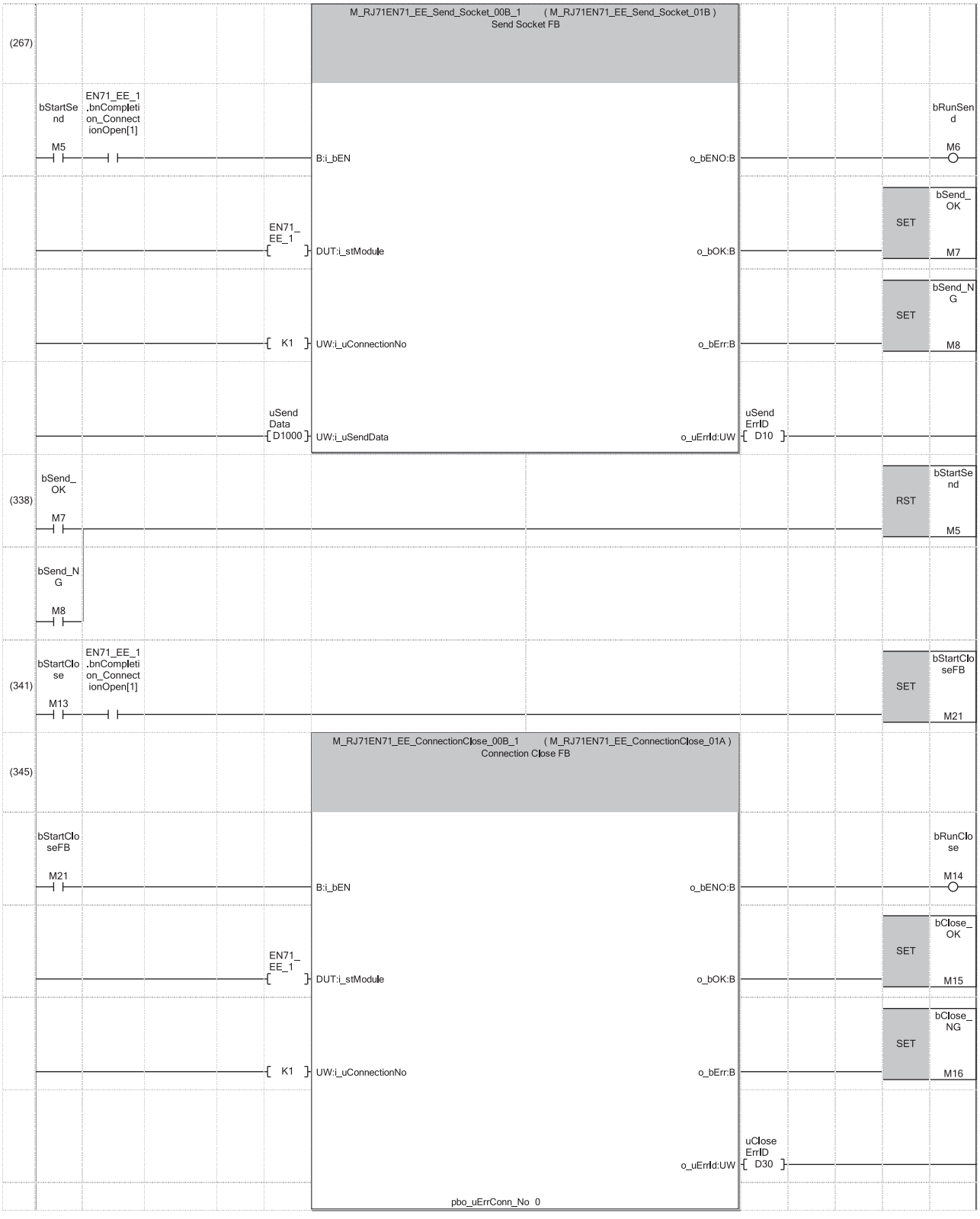
The following shows FBs to be used in the program example.

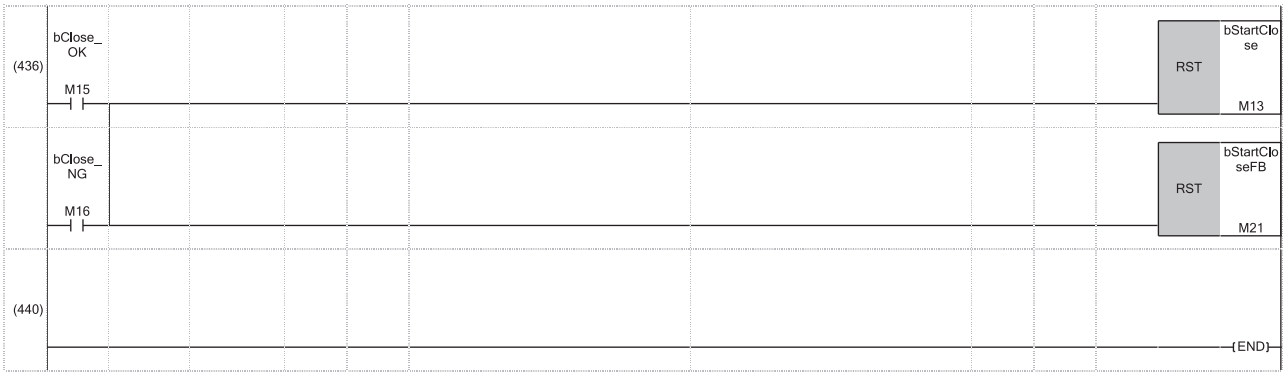
- M_RJ71EN71_EE_Refresh_Data
- M_RJ71EN71_EE_ConnectionOpen
- M_RJ71EN71_EE_Send_Socket
- M_RJ71EN71_EE_ConnectionClose

### ■Program




(56)
bStartOpen
M1
EN71_EE_1.bnCompletion_ConnectionOpen[1]
EN71_EE_1.bnStatus_ConnectionOpenExecution[1]
SET
bStartOpenFB
M20
(62)
M_RJ71EN71_EE_ConnectionOpen_00B_1 ( M_RJ71EN71_EE_ConnectionOpen_01A )
Connection Open FB
bStartOpenFB
M20
EN71_EE_1.uCompletion_EthernetInitialized.0
EN71_EE_1.stPort1.uStatus_HUB_Connection_D.0
U0\G5192.0
B:i_bEN
o_bENO:B
bRunOpen
M2
EN71_EE_1
DUT:i_stModule
o_bOK:B
SET
bOpen_OK
M3
K1
UW:i_uConnectionNo
o_bErr:B
SET
bOpen_NG
M4
o_uErrId:UW
uOpen ErrID
D0
pbi_bUseParameters 0
pbi_uProtocol 0
pbi_uOpen_System 0
pbi_uConnUsage 0
pbi_bProcedure 0
pbi_uExist_Confirm 0
pbi_uLocal_Port_No 4096
pbi_uTarget_Port_No 4096
pbi_u2IP_Address 0
pbi_bEnable_Online_Change 0
pbi_bData_Code 0
(248)
bOpen_OK
M3
RST
bStartOpen
M1
bOpen_NG
M4
RST
bStartOpenFB
M20
(252)
bStartSend
M5
EN71_EE_1.bnCompletion_ConnectionOpen[1]
MOVP
K6
uSendData[0]
D1000
MOVP
K1234
uSendData[1]
D1001
MOVP
K5678
uSendData[2]
D1002
MOVP
K8901
uSendData[3]
D1003


(267)
M_RJ71EN71_EE_Send_Socket_00B_1 ( M_RJ71EN71_EE_Send_Socket_01B )
Send Socket FB
bStartSend
M5
EN71_EE_1.bnCompletionConnectionOpen[1]
B:i_bEN
o_bENO:B
bRunSend
M6
EN71_EE_1
DUT:i_stModule
o_bOK:B
SET
bSend_OK
M7
K1
UW:i_uConnectionNo
o_bErr:B
SET
bSend_NG
M8
uSend Data
D1000
UW:i_uSendData
o_uErrId:UW
uSend ErrID
D10
(338)
bSend_OK
M7
RST
bStartSend
M5
bSend_NG
M8
(341)
bStartClose
M13
EN71_EE_1.bnCompletionConnectionOpen[1]
SET
bStartCloseFB
M21
(345)
M_RJ71EN71_EE_ConnectionClose_00B_1 ( M_RJ71EN71_EE_ConnectionClose_01A )
Connection Close FB
bStartCloseFB
M21
B:i_bEN
o_bENO:B
bRunClose
M14
EN71_EE_1
DUT:i_stModule
o_bOK:B
SET
bClose_OK
M15
K1
UW:i_uConnectionNo
o_bErr:B
SET
bClose_NG
M16
o_uErrId:UW
uClose ErrID
D30
pbo_uErrConn_No 0




(0) The refresh processing of the module label is performed. (The processing is required for when using the module function block.)
When the refresh processing is completed, 'bRunRefresh' (M0) is turned on.

(56) When 'bStartOpen' (M1) is turned on, the open processing of connection No.1 is performed.
When the open processing is completed successfully, 'bOpen_OK' (M3) is turned on.

(252) When 'bStartSend' (M5) is turned on, the send data is stored and sent to the receiving side.
When the data send is completed successfully, 'bSend_OK' (M7) is turned on.

(341) When 'bStartClose' (M13) is turned on, the close processing of connection No.1 is performed.
When the close processing is completed successfully, 'bClose_OK' (M15) is turned on.

## Program example (receiving side)

### ■Labels

| Classification | Label name | Description | Device |
|---|---|---|---|
| Module label | RCPU.stSM.bAlways_ON | Always on | SM400 |
| | EN71_EE_1.bnCompletion_ConnectionOpen[1] | Open completion signal (connection No.1) | U0\G1900000.0 |
| | EN71_EE_1.bnCompletion_ReceiveSocket_FixedBuffer[1] | Socket/fixed buffer reception status signal (connection No.1) | U0\G1900016.0 |
| | EN71_EE_1.uCompletion_EthernetInitialized.0 | Initial status | U0\G1900024.0 |
| Label to be defined | Define global labels as shown below: | | |

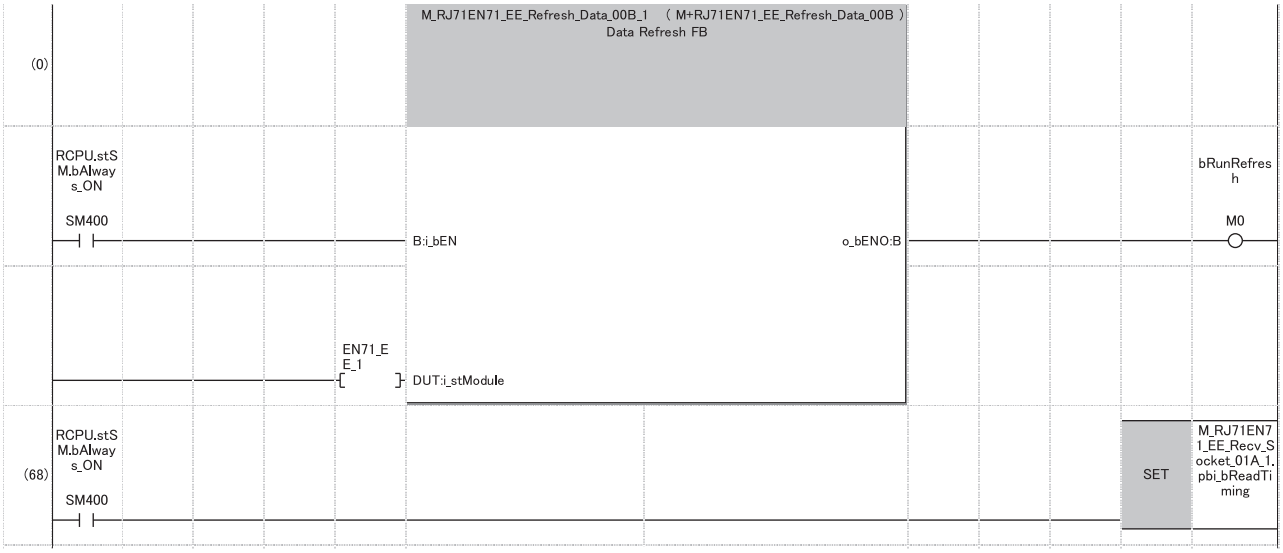
| Label Name | Data Type | | Class | Assign (Device/Label) |
|---|---|---|---|---|
| bRunRefresh | Bit | ... | VAR_GLOBAL | M0 |
| bStartRecv | Bit | ... | VAR_GLOBAL | M9 |
| bRunRecv | Bit | ... | VAR_GLOBAL | M10 |
| bRecv_OK | Bit | ... | VAR_GLOBAL | M11 |
| bRecv_NG | Bit | ... | VAR_GLOBAL | M12 |
| bStartRecvFB | Bit | ... | VAR_GLOBAL | M17 |
| uRecvErrID | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D20 |
| uRecvData | Word [Unsigned]/Bit String [16-bit](0..5119) | ... | VAR_GLOBAL | D2000 |

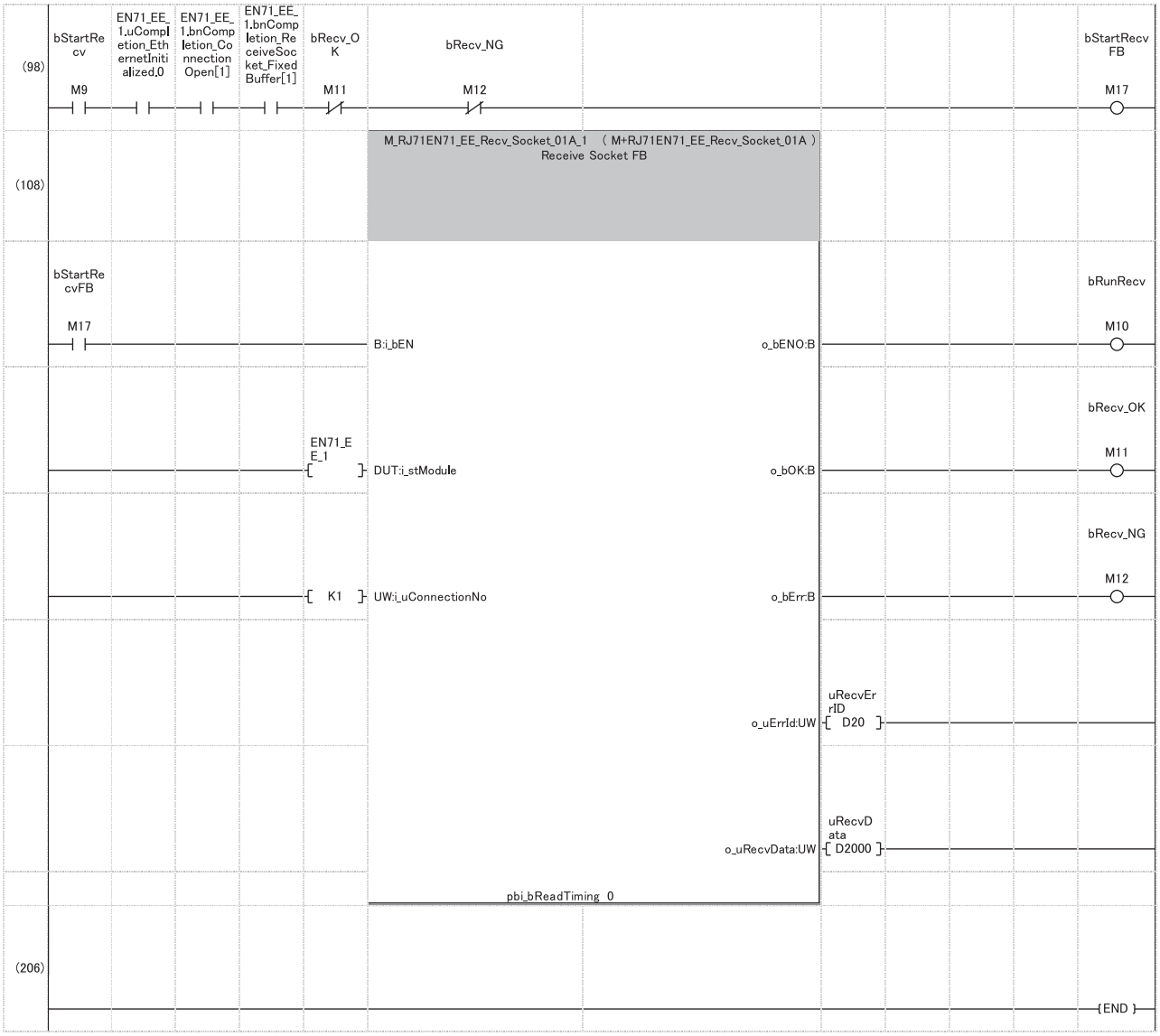
### ■FBs

The following shows FBs to be used in the program example.

- M_RJ71EN71_EE_Refresh_Data
- M_RJ71EN71_EE_Recv_Socket

### ■Program






(0) The refresh processing of the module label is performed. (The processing is required for when using the module function block.)
When the refresh processing is completed, 'bRunRefresh' (M0) is turned on.

(98) When 'bStartRecv' (M9) is turned on, the data sent from the sending side is received and stored in 'uRecvData' (D2000).
(The device range in which data is stored varies depending on the data length of the received data.)
When the data receive is completed successfully, 'bRecv_OK' (M11) is turned on.

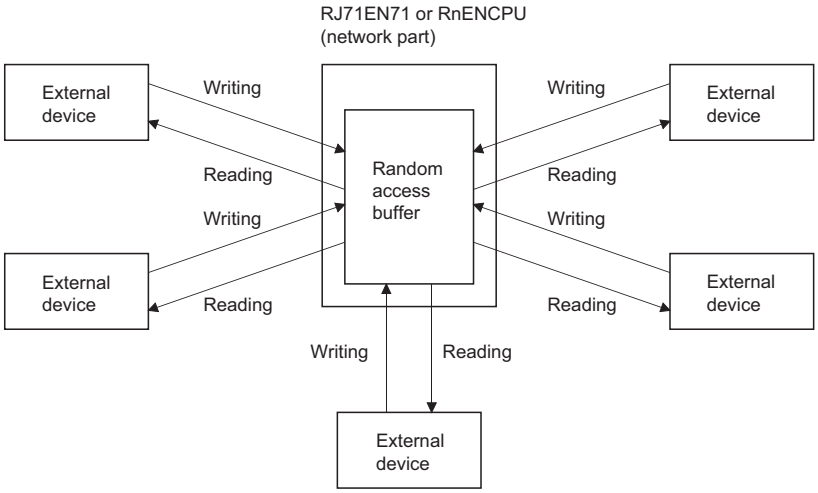
**Point**

- Secure sufficient device areas according to the maximum length of data sent from the send source to prevent the device areas used for other purposes from being overwritten by the receive data.
- When the data receive is consecutively executed, turn on pbi_bReadTiming (read timing) as shown in the above program.
- To receive data at shorter intervals than the scan time of the CPU module, add the normally closed contact of 'bRecv_OK' (M11) and 'bRecv_NG' (M12) to the execution conditions of FB for receiving as shown in the above program. When there is no normally closed contact of 'bRecv_OK' (M11) and 'bRecv_NG' (M12), 'bStartRecvFB' (M17) is not turned off and on and the FB for receiving may not be executed.

# 1.6 Communications Using the Random Access Buffer



With communications using the random access buffer, data can be freely read and written between any external device (excluding Ethernet-equipped module) and the RJ71EN71 or the RnENCPU (network part). The external device does not need to be fixed. The random access buffer is used as the common buffer area for external devices connected to the Ethernet.




## Precautions

The CPU module (CPU part for the RnENCPU) cannot exchange data with the random access buffer.

## Setting procedure

Set "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

***1.*** Select the external device to be connected in "Module List" and drag it to "List of devices" or "Device map area".

| External device name | Description |
|---|---|
| UDP Connection Module | Select to communicate with the external device using UDP/IP. |
| Active Connection Module | Select to perform the open processing to the external device from the Ethernet-equipped module (Active open) and communicate using TCP/IP. |
| Unpassive Connection Module | Select to receive the open processing from an unspecified external device (Unpassive open) and communicate using TCP/IP. |
| Fullpassive Connection Module | Select to receive the open processing from the specified external device (Fullpassive open) and communicate using TCP/IP. |

***2.*** Set "Communication Method" for external device to "Random Access Buffer".

***3.*** Set the IP address of the external device.

***4.*** Set the other parameters required for communication in the connection. (☞ Page 329 External Device Configuration)

# Communication structure

This section describes the mechanism of communications using the random access buffer.

## Data flow

The following figure shows the flow of data communications using the random access buffer.
A dedicated packet is used to exchange data between the external device and the RJ71EN71 or the RnENCPU (network part).
A program is used to access the random access buffer from the CPU module.

**Point**

The process is executed asynchronously with the program, so if synchronization is required, use socket communications or communications using the fixed buffer.

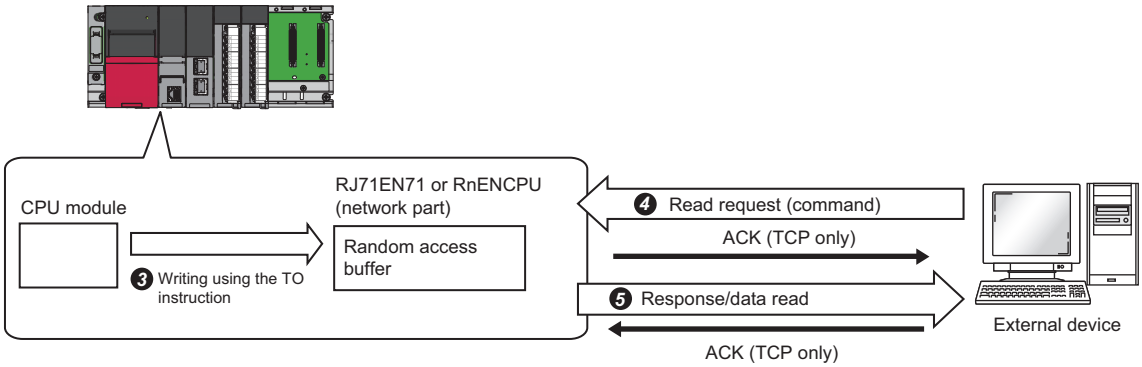
## External devices capable of data exchange

Data can be exchanged with the following external devices.

- Devices in Ethernet to which the RJ71EN71 is connected
- Devices in Ethernet to which the RnENCPU (network part) is connected
- Device connected via router

# Procedure for reading from external device

The following figure shows the processing when sending data from the RJ71EN71 or the RnENCPU (network part) in response to a read request from the external device.




1. After the module parameters are set, check that the initial processing of the RJ71EN71 or the RnENCPU (network part) has completed normally. ('Initial status' (Un\G1900024.0): On)
2. The open processing is executed to establish a connection between the RJ71EN71 or the RnENCPU (network part) and external device. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
3. The program writes the data into the random access buffer of the RJ71EN71 or the RnENCPU (network part).
4. The read request is sent from the external device to the RJ71EN71 or the RnENCPU (network part) asynchronously from the above processing. (RJ71EN71 or RnENCPU (network part) side: Receives command)
5. When the read request is received from the external device, the RJ71EN71 or the RnENCPU (network part) send the data written into the random access buffer to the external device. (RJ71EN71 or RnENCPU (network part) side: Sends response)
6. The connection is closed when communication is finished.

# Procedure for writing from external device



The following shows the procedure when writing data from the external device to the random access buffer of the RJ71EN71 or the RnENCPU (network part).

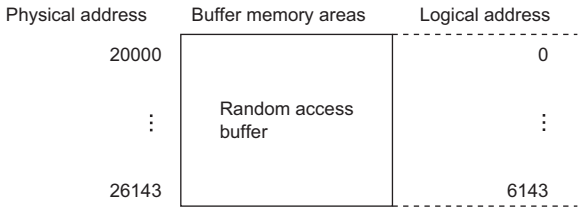
1. After the module parameters are set, check that the initial processing of the RJ71EN71 or the RnENCPU (network part) has completed normally. ('Initial status' (Un\G1900024.0): On)
2. The open processing is executed to establish a connection between the RJ71EN71 or the RnENCPU (network part) and external device. (☞ Page 546 TCP/IP Communications, UDP/IP Communications)
3. Data is written from the external device to the random access buffer of the RJ71EN71 or the RnENCPU (network part). (RJ71EN71 or RnENCPU (network part) side: Receives command)
4. The RJ71EN71 or the RnENCPU (network part) execute the write processing requested by the external device, and returns the write results to the external device that sent the write request. (RJ71EN71 or RnENCPU (network part) side: Sends response)
5. The data written in the random access buffer is read asynchronously from the above the processing by the program.
6. Close the connection when communication is finished.

# Physical address and logical address of random access buffer

This section describes the start address of the random access buffer of the RJ71EN71 or the RnENCPU (network part) specified in the command.

The address specified for the random access buffer differs from the address specified by the external device and the address specified with the FROM/TO instruction.

- Physical address: Address specified with program's FROM/TO instruction
- Logical address: Address specified by external device in start address item of command




# Precautions

The following section lists the precautions for communications using the random access buffer.

## Precautions for programming

### ■Initial processing and open processing completion

The initial processing and connection open processing must be completed.

### ■Send request from CPU module

Send cannot be requested from the CPU module. Completion of sending to CPU module is not checked. If the data send/receive must be synchronized between the CPU module and external device, use fixed buffer communication.

### ■Random access buffer address

The address specified by the external device is different from the address specified with the FROM/TO instruction. For details, refer to the following.

(☞ Page 100 Physical address and logical address of random access buffer)

# Data format

The communication data is configured of the header and application data.

## Header

The header is for TCP/IP or UDP/IP. The RJ71EN71 or the RnENCPU (network part) automatically adds and deletes the header, the setting is not required.

### ■Details of header section size

The details of the header section data format and size are shown below.

- TCP/IP

| Ethernet (14 bytes) | IP (20 bytes) | TCP (20 bytes) |
|---|---|---|

- UDP/IP

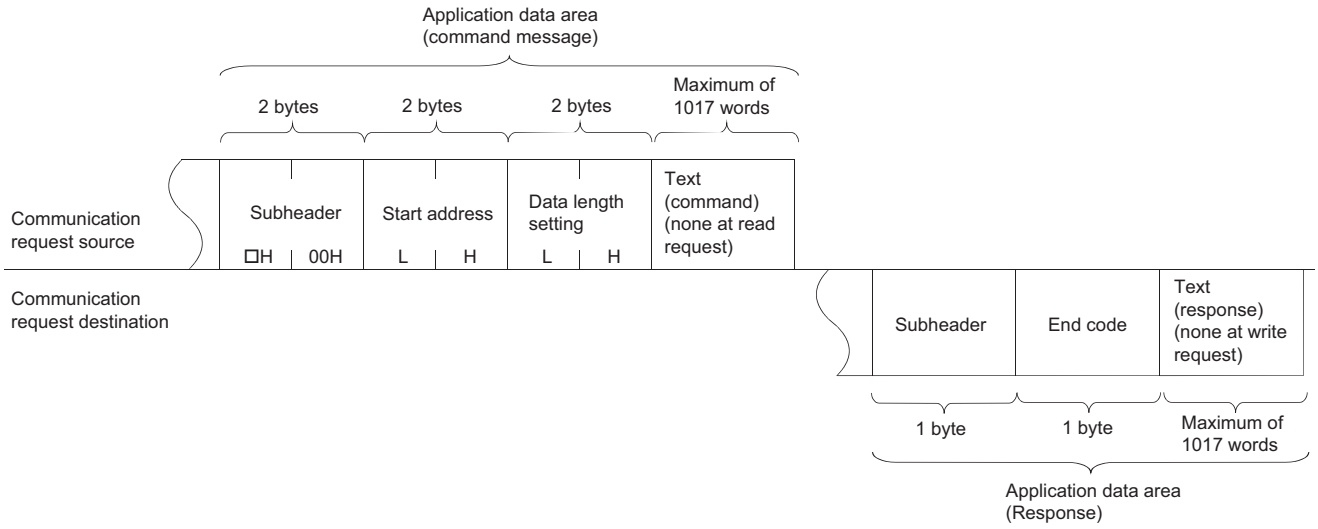
| Ethernet (14 bytes) | IP (20 bytes) | UDP (8 bytes) |
|---|---|---|

## Application data

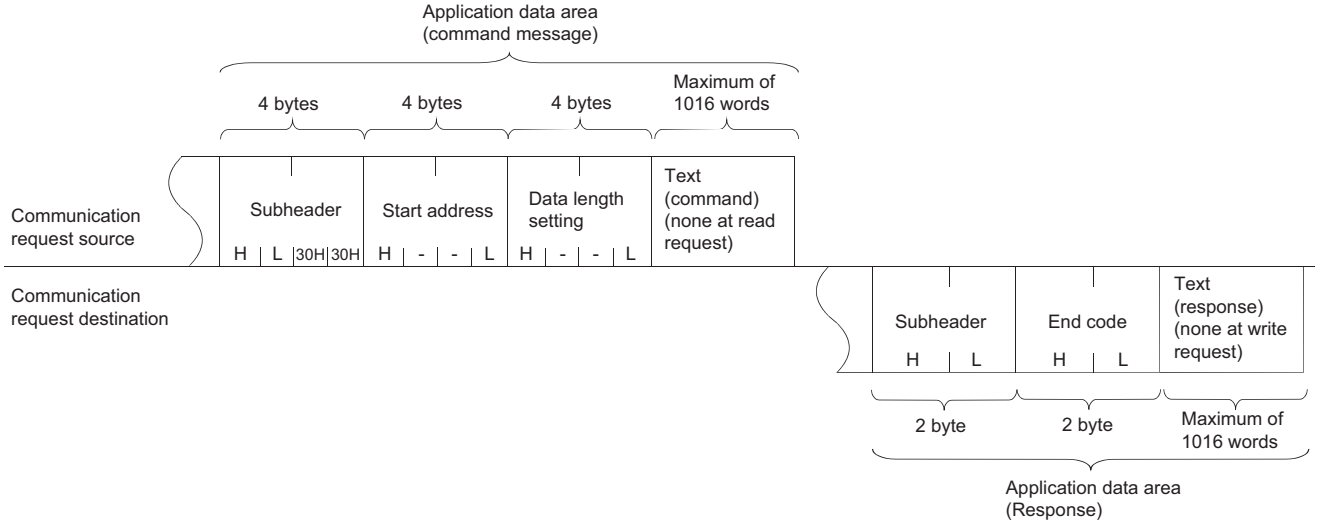
The application data expresses the following data code as binary code or ASCII code. Switch between the binary code and ASCII code with "Own Node Settings" under "Basic Settings". (☞ Page 327 Own Node Settings)

### ■Format

• When exchanging data with binary codes

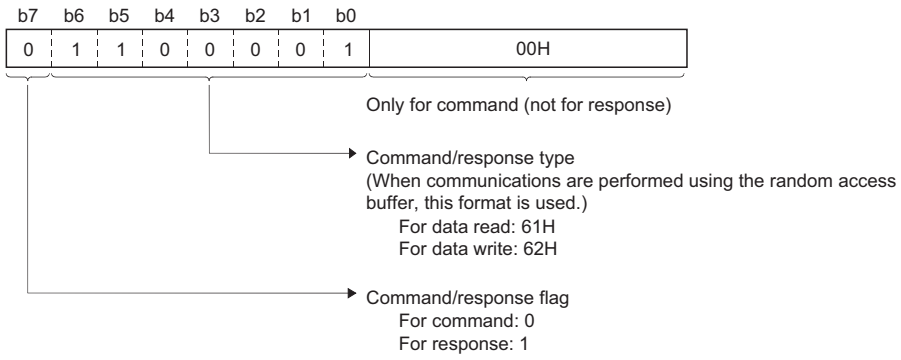



• When exchanging data with ASCII codes

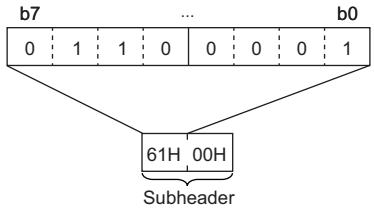
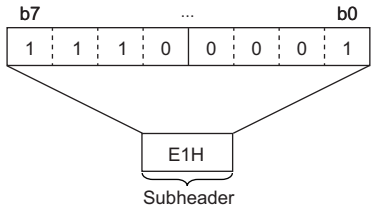
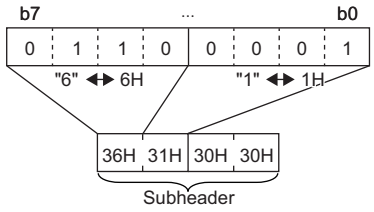
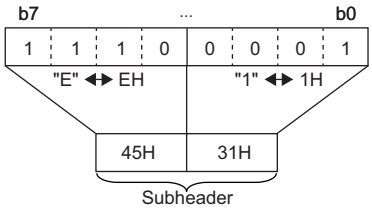



### ■Subheader

The RJ71EN71 or the RnENCPU (network part) automatically adds and deletes the subheader, the setting is not required.



• During read

| Data format | Command (external device → RJ71EN71 or RnENCPU (network part)) | Response (RJ71EN71 or RnENCPU (network part) → external device) |
|---|---|---|
| Binary code |  |  |
| ASCII code |  |  |

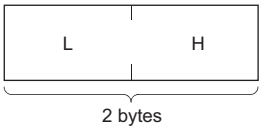
## ■Start Address

The start address of the random access buffer range where data read/write is executed is expressed as a logical address.

(☞ Page 100 Physical address and logical address of random access buffer)

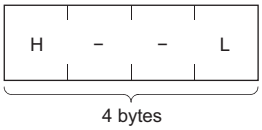
• When exchanging data with binary codes

Specify the head address with a binary value.




• When exchanging data with ASCII codes

Specify the ASCII code for expressing the start address as a hexadecimal.

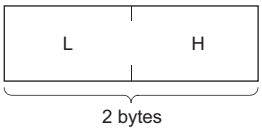



## ■Data length setting

Express the number of read/write data words with the random access buffer range.

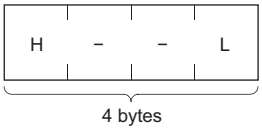
• When exchanging data with binary codes

Specify the number of words as a binary value. (Maximum 1017 words)



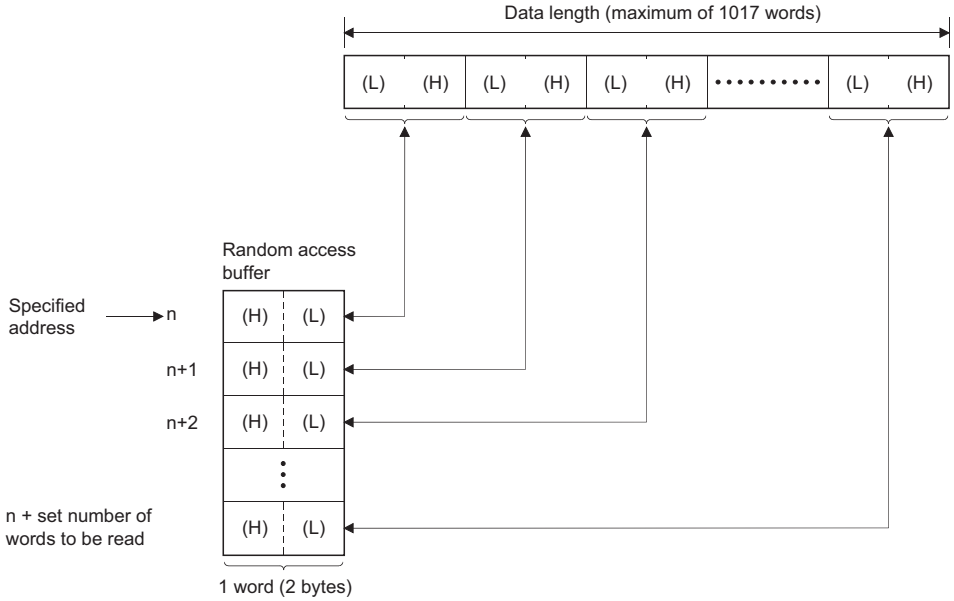

• When exchanging data with ASCII codes

Specify the ASCII code when expressing the number of words as a hexadecimal. (Maximum 508 words)

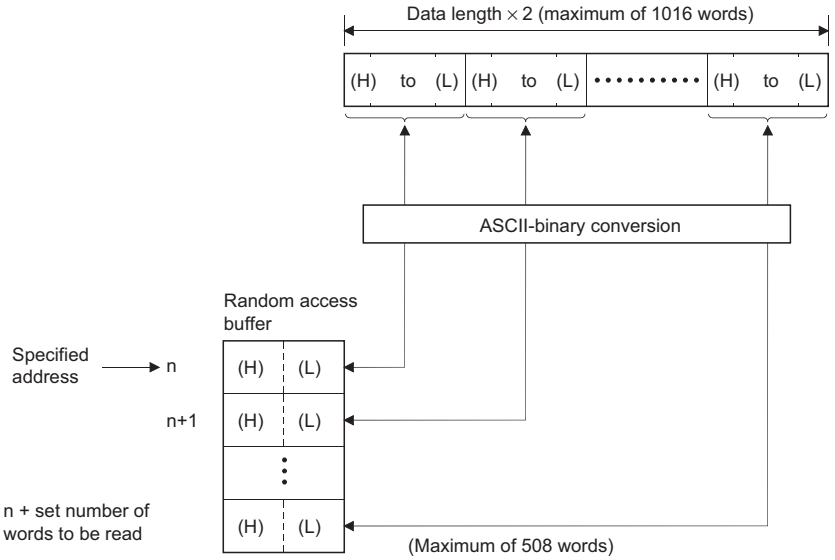


## ■Text

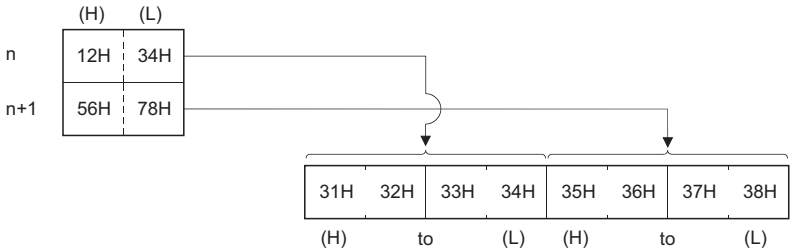
• When exchanging data with binary codes




• When exchanging data with ASCII codes




**Ex.**




## ■End code

The error code is stored in the end command added to the response. (☞ Page 479 End Codes Returned to an External Device During Data Communications)

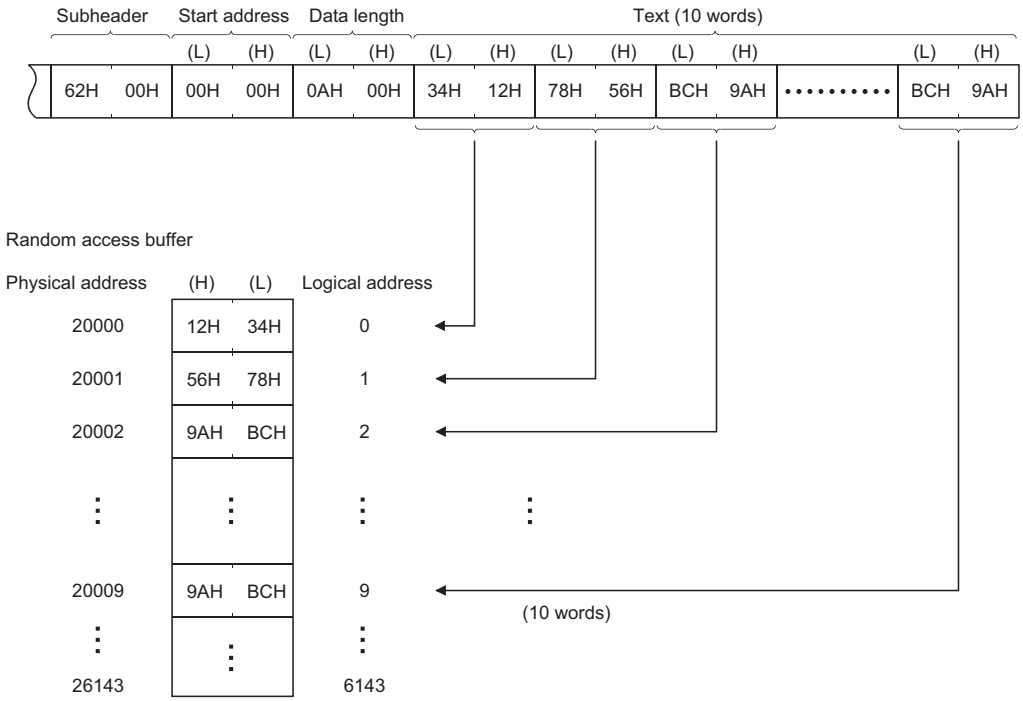
## Example of command/response format

The following section shows an example of the command and response format.

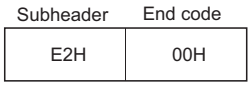
### ■Writing to buffer with write request from external device

• When exchanging data with binary codes

Command format (external device → RJ71EN71)

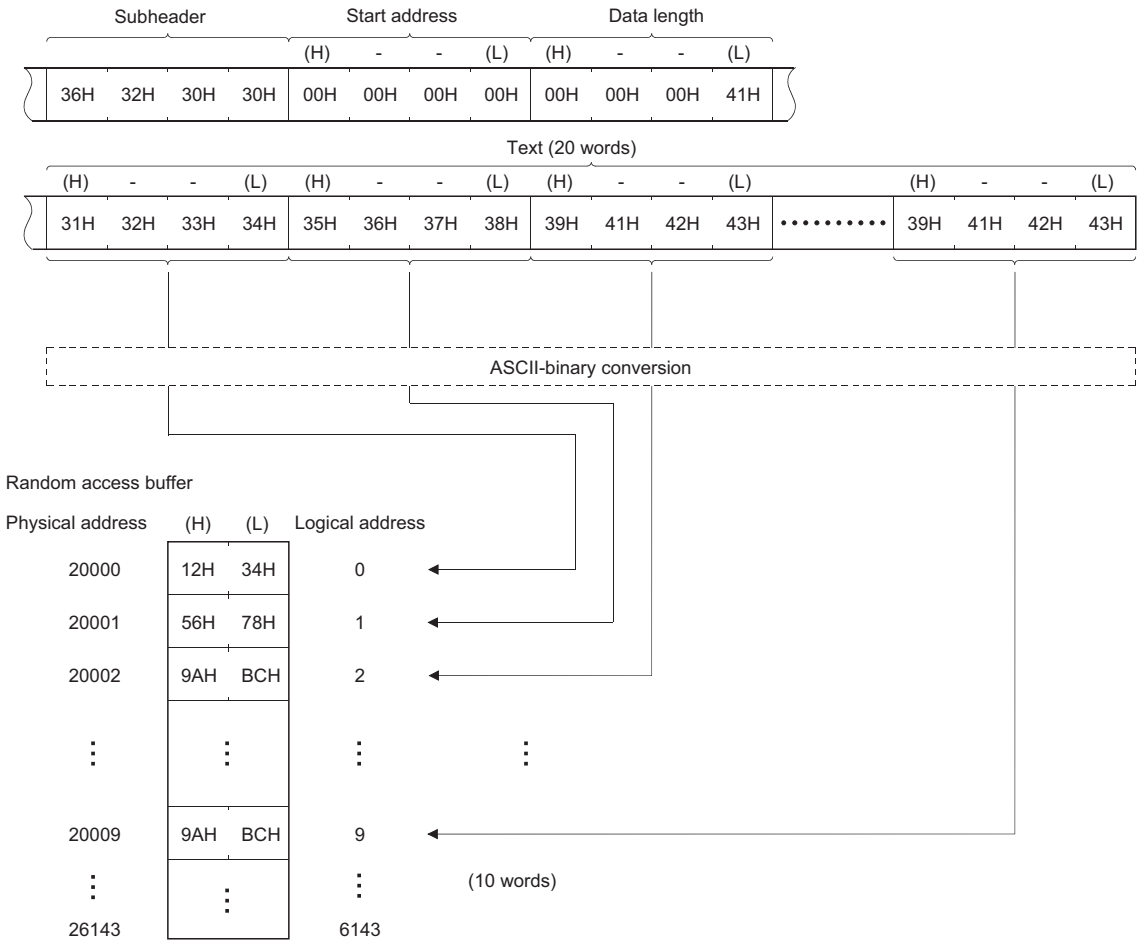



Response format (RJ71EN71 → external device)

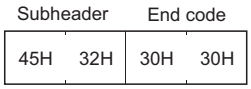


• When exchanging data with ASCII codes

Command format (external device → RJ71EN71)



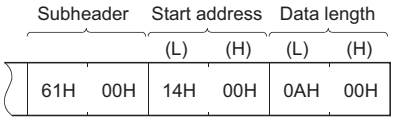

Response format (RJ71EN71 → external device)



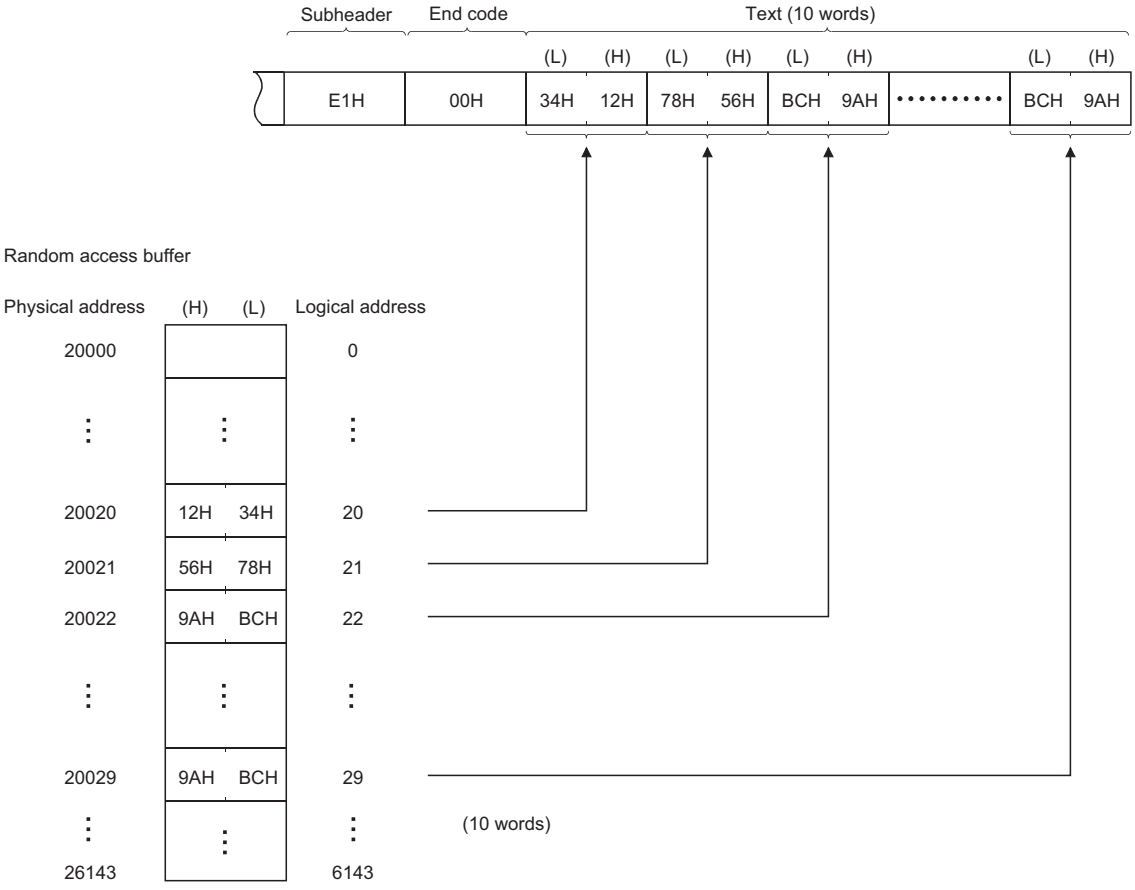
## ■Reading from buffer with read request from external device

• When exchanging data with binary codes

Command format (external device → RJ71EN71)

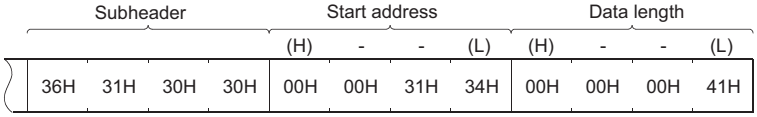



Response format (RJ71EN71 → external device)

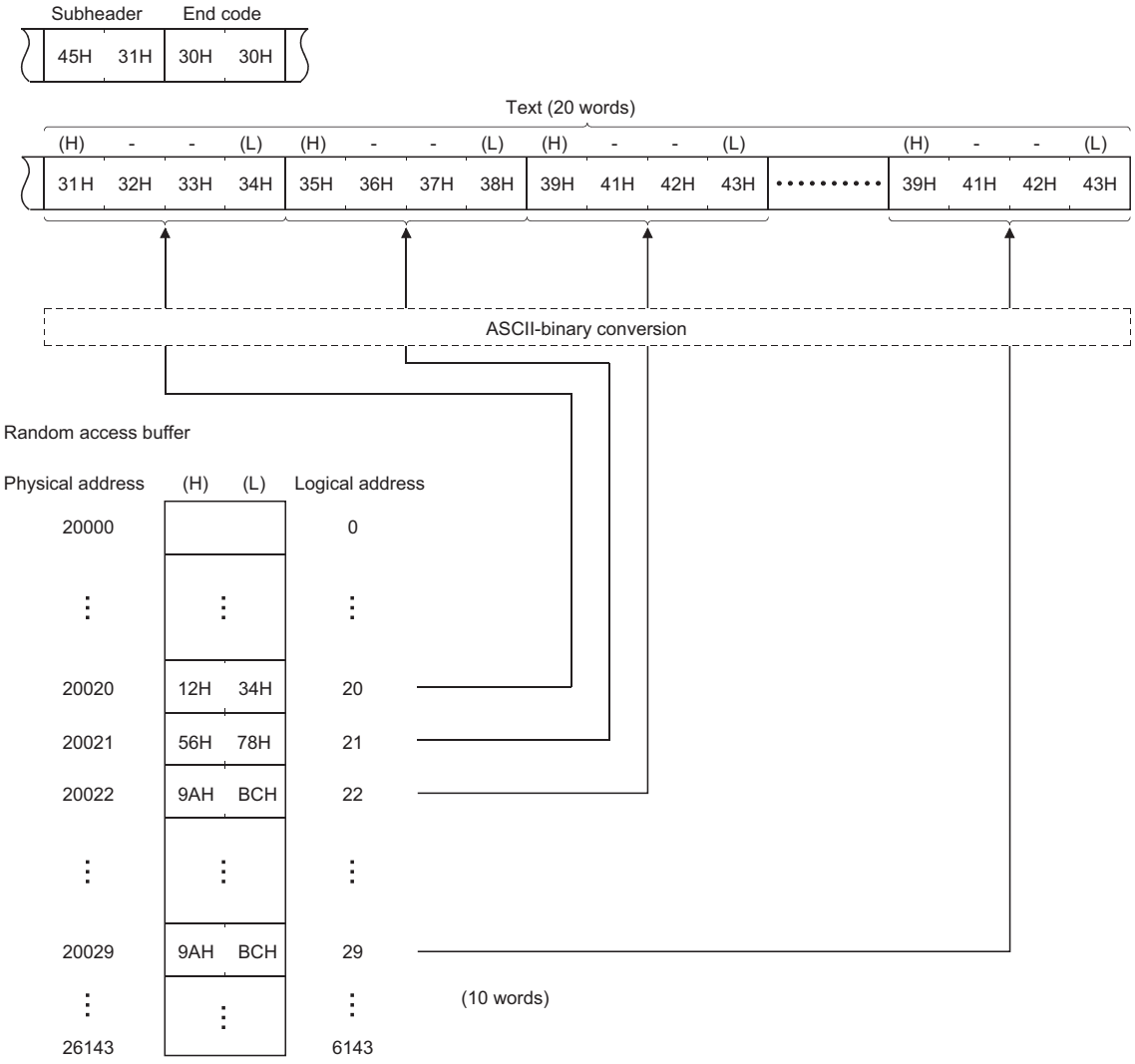


• When exchanging data with ASCII codes

Command format (external device → RJ71EN71)



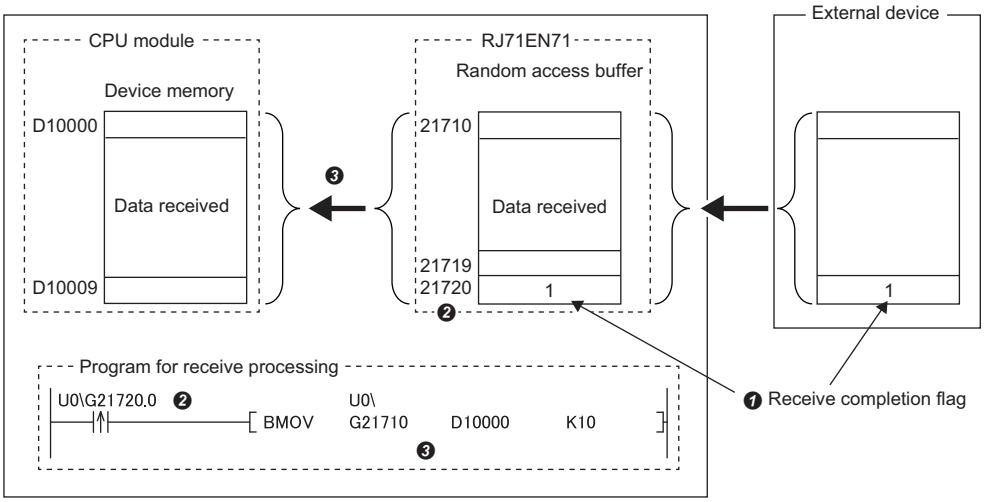

Response format (RJ71EN71 → external device)



# Example of communications with random access buffer



The following figure shows an example of the write operation from the external device.




❶ The receive completion flag is set at the end of the received data.
❷ The receive completion flag is monitored.
❸ When the receive completion flag turns on, the data is transferred from the random access buffer to the CPU module.

# 1.7 Communications Using MODBUS/TCP

MODBUS protocol messages are sent/received via the TCP/IP network.

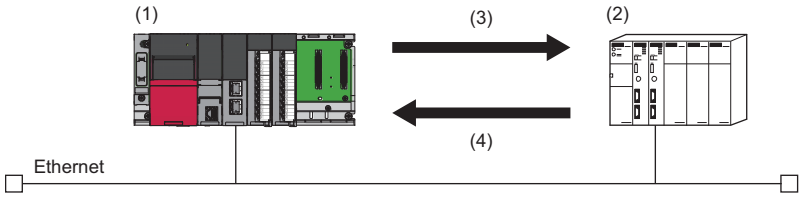
For communications using MODBUS/TCP, refer to the following.

📖 MELSEC iQ-R MODBUS and MODBUS/TCP Reference Manual

## MODBUS/TCP master function

A module communicates with a MODBUS/TCP slave device as a master station using one of the following functions.

| Function | Description | Available module | Reference source |
|---|---|---|---|
| Communications using the predefined protocol | Executes the protocol using the dedicated instructions when a MODBUS/TCP is selected from the predefined protocol library. | • CPU module (built-in Ethernet port part)<br>• RJ71EN71, RnENCPU (network part) | Page 36 Communications Using the Predefined Protocol<br>📖Ethernet Predefined Protocol Library (Standard Equipped Protocol) Reference Manual (for MODBUS/TCP) [EPM-M002] |
| Simple CPU communication function | Issues a request message for reading/writing to a MODBUS/TCP slave device automatically. | RJ71EN71, RnENCPU (network part) | Page 185 Simple CPU Communication Function (RJ71EN71, RnENCPU (Network Part)) |
| Simple device communication function | Executes the protocol when a MODBUS/TCP is selected from the simple device communication library. | RJ71EN71, RnENCPU (network part) | Page 259 Simple Device Communication Function<br>📖 MELSEC iQ-R Simple Device Communication Library Reference Manual |

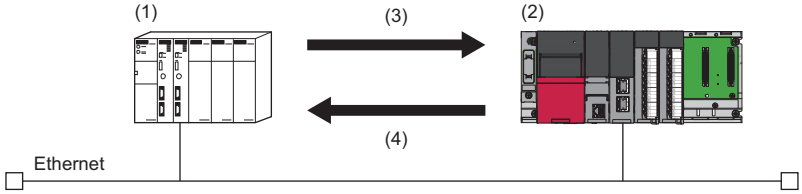



(1) CPU module (built-in Ethernet port part), RJ71EN71, RnENCPU (network part)
(2) MODBUS/TCP slave device
(3) Request message
(4) Response message

## MODBUS/TCP slave function

A module communicates with a MODBUS/TCP master device as a slave station using the automatic operations described in the following pages.

- ☞ Page 111 Automatic response to MODBUS/TCP master device
- ☞ Page 112 Automatic conversion from MODBUS device




(1) MODBUS/TCP master device
(2) RJ71EN71 or RnENCPU (network part)
(3) Request message
(4) Response message

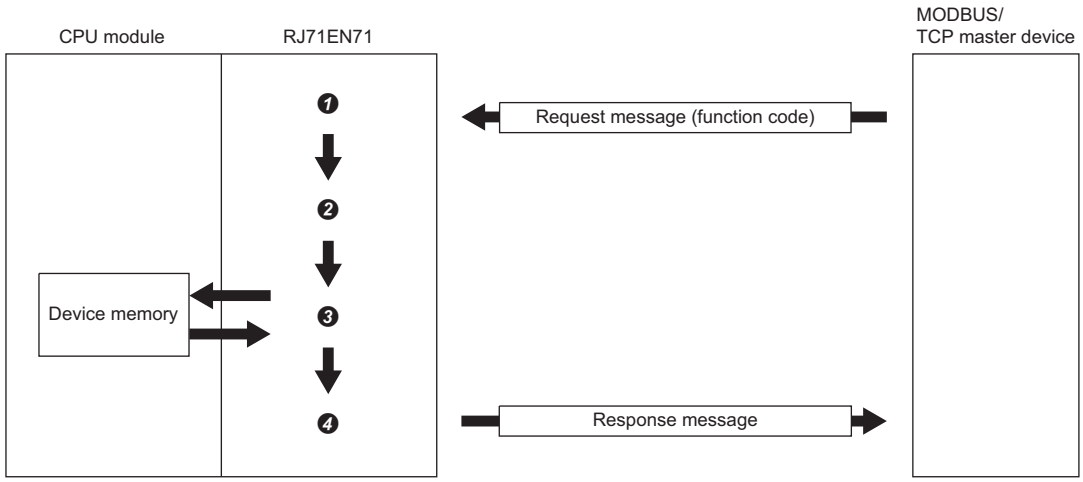
**Restriction**

- Before using this function, check the firmware version of the module. (☞ Page 586 Added and Enhanced Functions)
- This function is not available when the network type is set to "Q Compatible Ethernet" for the RJ71EN71.

## Automatic response to MODBUS/TCP master device

When receiving a request message from the MODBUS/TCP master device, the RJ71EN71 or RnENCPU (network part) automatically executes a necessary processing depending on the function code and sends a response message to the MODBUS/TCP master device.

No program is required because the function of automatic conversion from MODBUS device is used. (☞ Page 112 Automatic conversion from MODBUS device)




❶ A request message is received.
❷ The function code is checked.
❸ The requested processing is executed.
❹ The response message is sent.

This section lists the function codes (MODBUS standard functions) that can be executed from the MODBUS/TCP master device to the RJ71EN71 and RnENCPU (network part). (📖 MELSEC iQ-R MODBUS and MODBUS/TCP Reference Manual)

| Function code (FC) | Sub-code (SC) | Function name |
|---|---|---|
| 01H | — | Read coil |
| 02H | — | Read input |
| 03H | — | Read holding register |
| 04H | — | Read input register |
| 05H | — | Write single coil |
| 06H | — | Write single register |
| 0FH | — | Write multiple coils |
| 10H | — | Write multiple register |
| 14H | 06H | Read extended file register |
| 15H | 06H | Write extended file register |
| 16H | — | Mask-write holding register |
| 17H | — | Read/write multiple register |

### ■Setting procedure

Set the automatic response function in "External Device Configuration" under "Basic Settings". (☞ Page 329 External Device Configuration)

1. Select "MODBUS/TCP Connection Module" in "Module List", and drag it to "List of devices" or "Device map area".
2. Set "Port No." and "Existence Confirmation".

When multiple devices are set in "MODBUS/TCP Connection Module", set the same value to "Port No." of all devices.

3. Select [Close with Reflecting the Setting] to finish the external device configuration settings.

### ■Open/close timing

After the initial processing of the RJ71EN71 and RnENCPU (network part) completes, the port automatically opens according to the parameter settings. The port automatically closes when the station on which the RJ71EN71 and RnENCPU (network part) are mounted is turned off or reset.

### ■Precautions

When a dedicated instruction and the automatic communication function are simultaneously used with the QJ71MT91 set as the MODBUS/TCP master device, set two devices to "MODBUS/TCP Connection Module" in "External Device Configuration".

For dedicated instructions and the automatic communication function of the QJ71MT91, refer to the following.

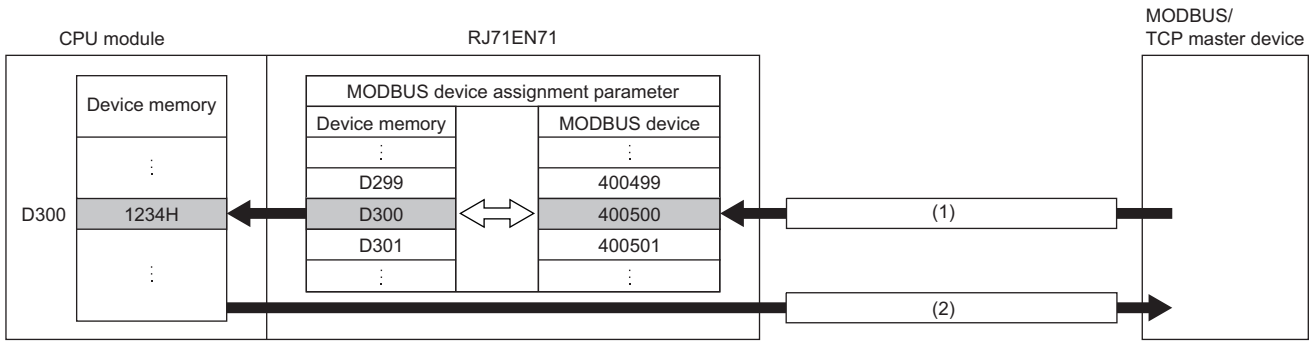
📖 MODBUS/TCP Interface Module User's Manual

## Automatic conversion from MODBUS device

This function automatically converts the access to the MODBUS devices of the RJ71EN71 and RnENCPU (network part) into the access to the devices of the CPU module.

Use the MODBUS device assignment parameters to create the association between the MODBUS devices and the device memory areas of the CPU module.

This allows the RJ71EN71 and RnENCPU (network part) to convert the access to the MODBUS device specified in the request message from the MODBUS/TCP master device into the access to the corresponding device of the CPU module.




(1) Request message (request for reading the value in holding register 400500)

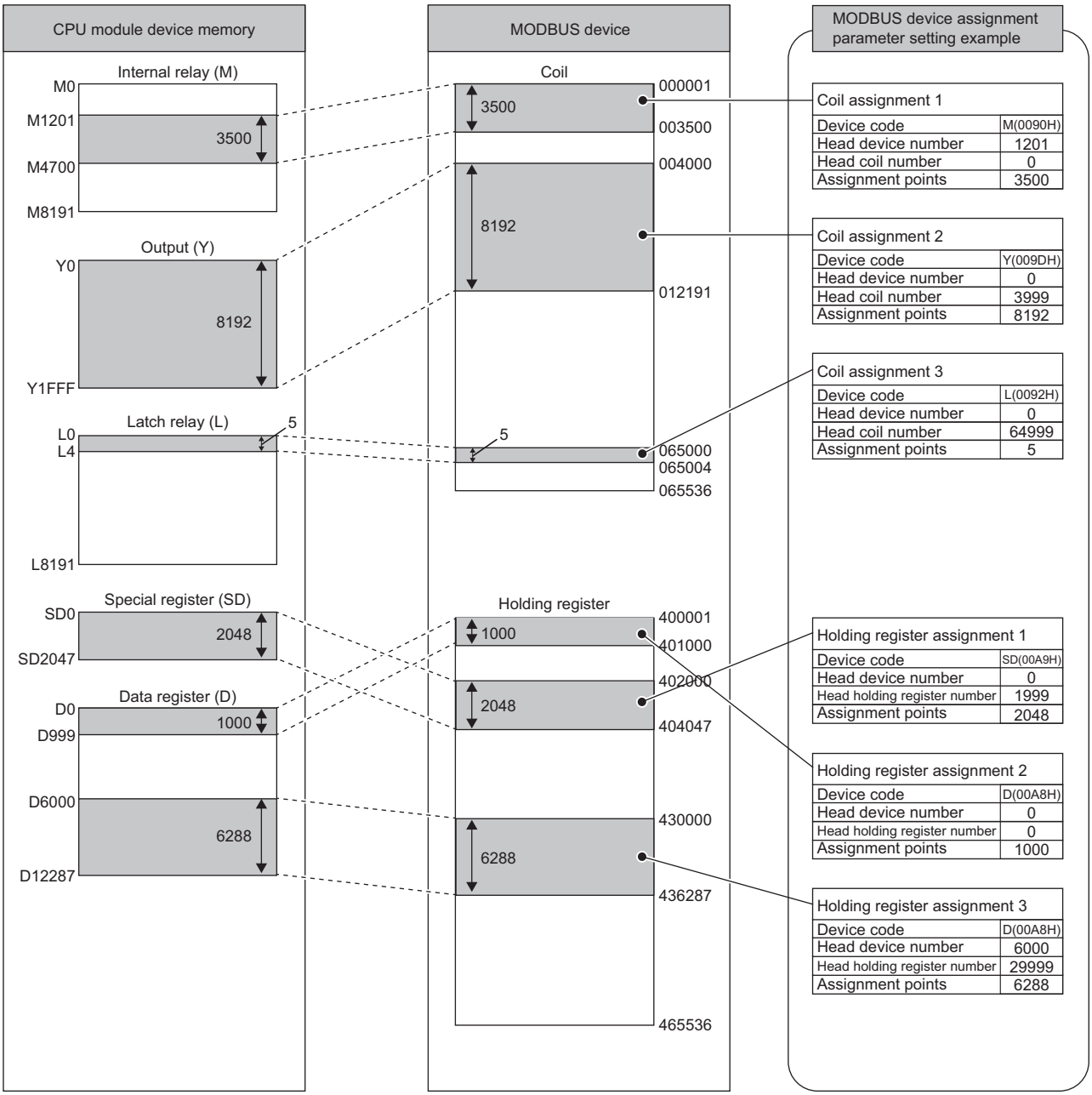
(2) Response message (response with the value in holding register 400500 (1234H))

### ■Setting procedure

Set the assignment to the MODBUS devices in "MODBUS/TCP Settings" under "Application Settings". (☞ Page 340 MODBUS/TCP Settings)

## ■MODBUS device assignment parameters

The parameters are used to associate the MODBUS devices with the device memory areas of the CPU module. Using the parameters enables direct access from a MODBUS/TCP master device to the device memory areas of the CPU module.



## ■MODBUS devices

The following table lists the MODBUS devices supported by the RJ71EN71 and RnENCPU (network part).

| MODBUS device type | Unit | Read/write | Number of access points | MODBUS device No. |
|---|---|---|---|---|
| Coil | Bit | Read/write | 65536 points | 1 to 65536 |
| Input | Bit | Read | 65536 points | 100001 to 165536 |
| Input register | Word | Read | 65536 points | 300001 to 365536 |
| Holding register | Word | Read/write | 65536 points | 400001 to 465536 |
| Extended file register | Word | Read/write*1 | 10027008 points | File number: 0 to 1002*2*3<br>Device number: 600000 to 609999 |

*1 For the CPU module, read-only memory cannot be set in the file register.

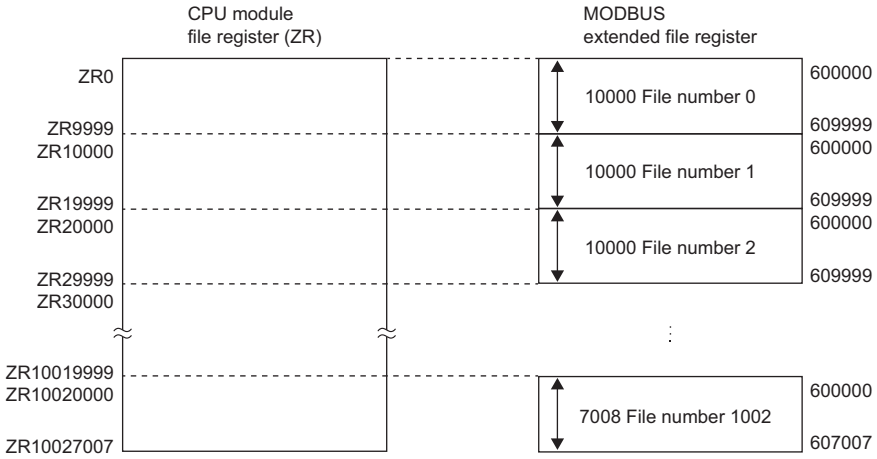
*2 The maximum values of the number of access points and the file number for the extended file register depend on the size of the file register (ZR) set in the CPU module. (☞ Page 114 MODBUS extended file register assignment)

*3 The number of access points for the extended file register is the maximum size that can be set in the CPU module (when the R120CPU + NZ2MC-16MBS are mounted).

## ■MODBUS extended file register assignment

The assignment of the MODBUS extended file register to the CPU module is fixed to the file register (ZR).

The following shows the assignment to the file register of the CPU module. (The number of points for the file register shows the maximum size that can be set in the CPU module (when the R120CPU + NZ2MC-16MBS are mounted)).




The size of the MODBUS extended file register depends on the size of the file register (ZR) set in the CPU module.

For details, refer to the manual for the CPU module used.

## ■Applicable devices

The following table lists the devices of the CPU module that can be set in the MODBUS devices.

| Device name | Symbol | MODBUS device (○: Can be assigned, ×: Cannot be assigned) | | | | |
|---|---|---|---|---|---|---|
| | | Coil | Input | Input register | Holding register | Extended file register |
| Special relay | SM | ○ | ○ | ○*1 | ○*1 | × |
| Special register | SD | × | × | ○ | ○ | × |
| Input | X | ○ | ○ | ○*1 | ○*1 | × |
| Output | Y | ○ | ○ | ○*1 | ○*1 | × |
| Internal relay | M | ○ | ○ | ○*1 | ○*1 | × |
| Latch relay | L | ○ | ○ | ○*1 | ○*1 | × |
| Annunciator | F | ○ | ○ | ○*1 | ○*1 | × |
| Edge relay | V | ○ | ○ | ○*1 | ○*1 | × |
| Link relay | B | ○ | ○ | ○*1 | ○*1 | × |
| Data register | D | × | × | ○ | ○ | × |
| Link register | W | × | × | ○ | ○ | × |
| Timer (contact) | TS | ○ | ○ | ○*1 | ○*1 | × |
| Timer (coil) | TC | ○ | ○ | ○*1 | ○*1 | × |
| Timer (current value) | TN | × | × | ○ | ○ | × |
| Long timer (current value) | LTN | × | × | ○*4 | ○*4 | × |
| Retentive timer (contact) | STS | ○ | ○ | ○*1 | ○*1 | × |
| Retentive timer (coil) | STC | ○ | ○ | ○*1 | ○*1 | × |
| Retentive timer (current value) | STN | × | × | ○ | ○ | × |
| Long retentive timer (current value) | LSTN | × | × | ○*4 | ○*4 | × |
| Counter (contact) | CS | ○ | ○ | ○*1 | ○*1 | × |
| Counter (coil) | CC | ○ | ○ | ○*1 | ○*1 | × |
| Counter (current value) | CN | × | × | ○ | ○ | × |
| Long counter (contact) | LCS | ○ | ○ | ○*1 | ○*1 | × |
| Long counter (coil) | LCC | ○ | ○ | ○*1 | ○*1 | × |
| Long counter (current value) | LCN | × | × | ○*2 | ○*2 | × |
| Link special relay | SB | ○ | ○ | ○*1 | ○*1 | × |
| Link special register | SW | × | × | ○ | ○ | × |
| Step relay*5 | S | ○ | ○ | ○*1 | ○*1 | × |
| Direct input | DX | ○ | ○ | ○*1 | ○*1 | × |
| Direct output | DY | ○ | ○ | ○*1 | ○*1 | × |
| Index register | Z | × | × | ○ | ○ | × |
| Long index register | LZ | × | × | ○*2 | ○*2 | × |
| File register | R | × | × | ○ | ○ | × |
| | ZR | × | × | × | × | ○*3 |
| Refresh data register | RD | × | × | ○ | ○ | × |
| Buffer memory | H*6 | × | × | ○ | ○ | × |

*1 When setting a bit type device to the input register or output register, set it so that the number of assigned points will be a multiple of 16. Otherwise, an error occurs in the engineering tool.

*2 The long counter (current value) and the long index register use two points of MODBUS devices.

*3 The assignment of the extended file register is fixed to the file register (ZR).

*4 The long timer (current value) and long retentive timer (current value) use four points of MODBUS devices. (☞ Page 116 Long timer and long retentive timer)

*5 The step relay is read-only. An attempt of writing will cause an error.

*6 When the buffer memory is used, use H (hexadecimal constant) as the device symbol and set the address in hexadecimal notation. The setting range is 1D5000H to 1D6FFFH. (☞ Page 525 User setting area for MODBUS/TCP function (Un\G1921024 to Un\G1929215))

### ■Long timer and long retentive timer

Note that the following items cannot be set for reading the long timer and long retentive timer.

- Long timer (contact)
- Long timer (coil)
- Long retentive timer (contact)
- Long retentive timer (coil)

When, however, the current value of the long timer or long retentive timer is set in units of four words, the contact and coil will be read simultaneously.

| Response data | Description |
|---|---|
| 1st word | The current value is stored. |
| 2nd word | |
| 3rd word | • b0: The value of the coil is stored.<br>• b1: The value of the contact is stored.<br>• b2 to b15: Used by the system. |
| 4th word | Used by the system. |

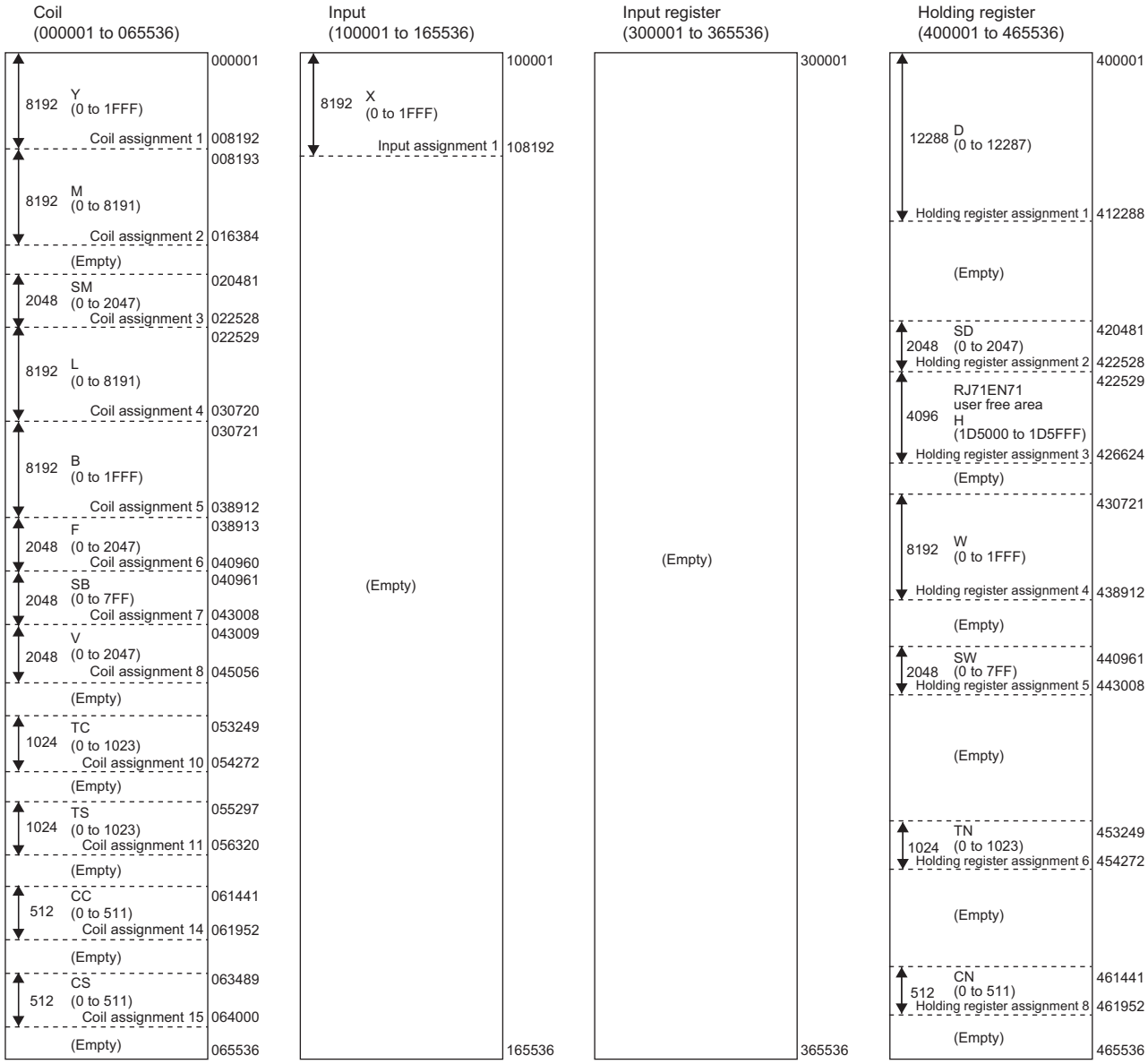
As shown above, the long timer and long retentive timer use the data configuration of four words per device. Consequently, specify the number of device points for a request data in units of four points.

Note that the write of long timer and long retentive timer cannot be set.

## ■Default assignment parameters

The MODBUS device assignment parameters include default assignment parameters as initial values.

The following diagram shows the MODBUS device assignment using the default assignment parameters.




**Point**

- In the MODBUS device assignment parameters, start MODBUS device number is last five digits of the setting target MODBUS device number - 1.
- The default values set for the MODBUS device assignment parameters cannot be used for creating a project for the engineering tool using the remote head module. Clear the values first before starting the setting.

# 1.8 Link Dedicated Instruction Communication

The RJ71EN71 or the RnENCPU (network part) execute the transient transmission to the programmable controller of the other station on the same Ethernet using the link dedicated instructions. (☞ Page 539 Link dedicated instructions)
A station on a network other than Ethernet can be accessed. (☞ Page 553 Communications with Different Networks)

### Precautions

The CPU module (CPU part for the RnENCPU) cannot exchange data with the link dedicated instruction.

## Data communication procedures

This section describes the procedure for communication with link-dedicated instructions.

1. After setting the module parameters, check that the initial processing of the own station, relay stations, and the RJ71EN71 and RnENCPU (network part) of access stations have completed successfully. ('Initial status' (Un\G1900024.0): On)
2. Data is exchanged with link dedicated instructions. (☞ Page 539 Link dedicated instructions)

**Point**

Communication with the link dedicated instructions uses the MELSOFT transmission port (UDP/IP), so the open/close processing is not required.

### Setting procedure

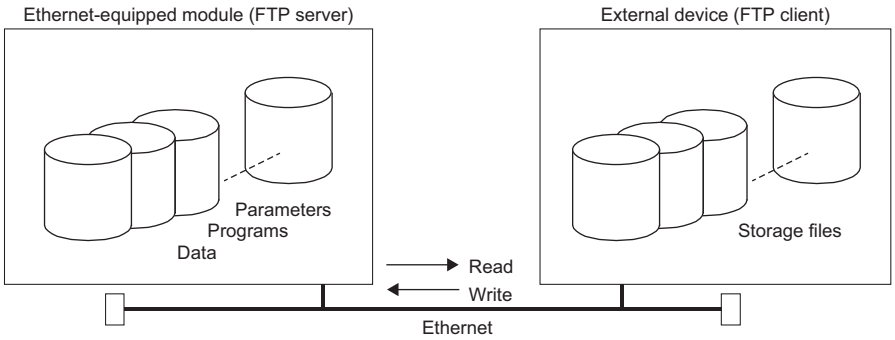
1. Set the network number and station number in "Own Node Settings" under "Basic Settings". (☞ Page 328 Details of items)
2. Set the send destination station information in "Network/Station number <-> IP Related information setting" under "Application Settings". (☞ Page 407 Network/Station No. <-> IP information setting)*1

*1 This setting is not required when the target station is specified with an IP address using control data of the control data of READ instruction, SREAD instruction, WRITE instruction, or SWRITE instruction.

# 1.9 File Transfer Function (FTP server)



The server function of FTP (File Transfer Protocol) used to transfer files with the external device is supported. The external device with FTP client function can easily and directly access files in the CPU module.
The following operations can be executed on the CPU module files from the external device with FTP client function.




- Reading of file from CPU module (download)
- Writing of file to CPU module (upload)
- Browsing of file names in CPU module

**Point**

When using the file transfer function (FTP server) with multi-CPU system configuration, the following CPU module is accessed immediately after logging in. To change the access destination CPU module, execute the cpuchg command.

- Control CPU (when the RJ71EN71 is connected)
- CPU module connected with Ethernet cable (when CPU module is connected)

# Data communication procedures

The following section describes the settings for FTP communication.

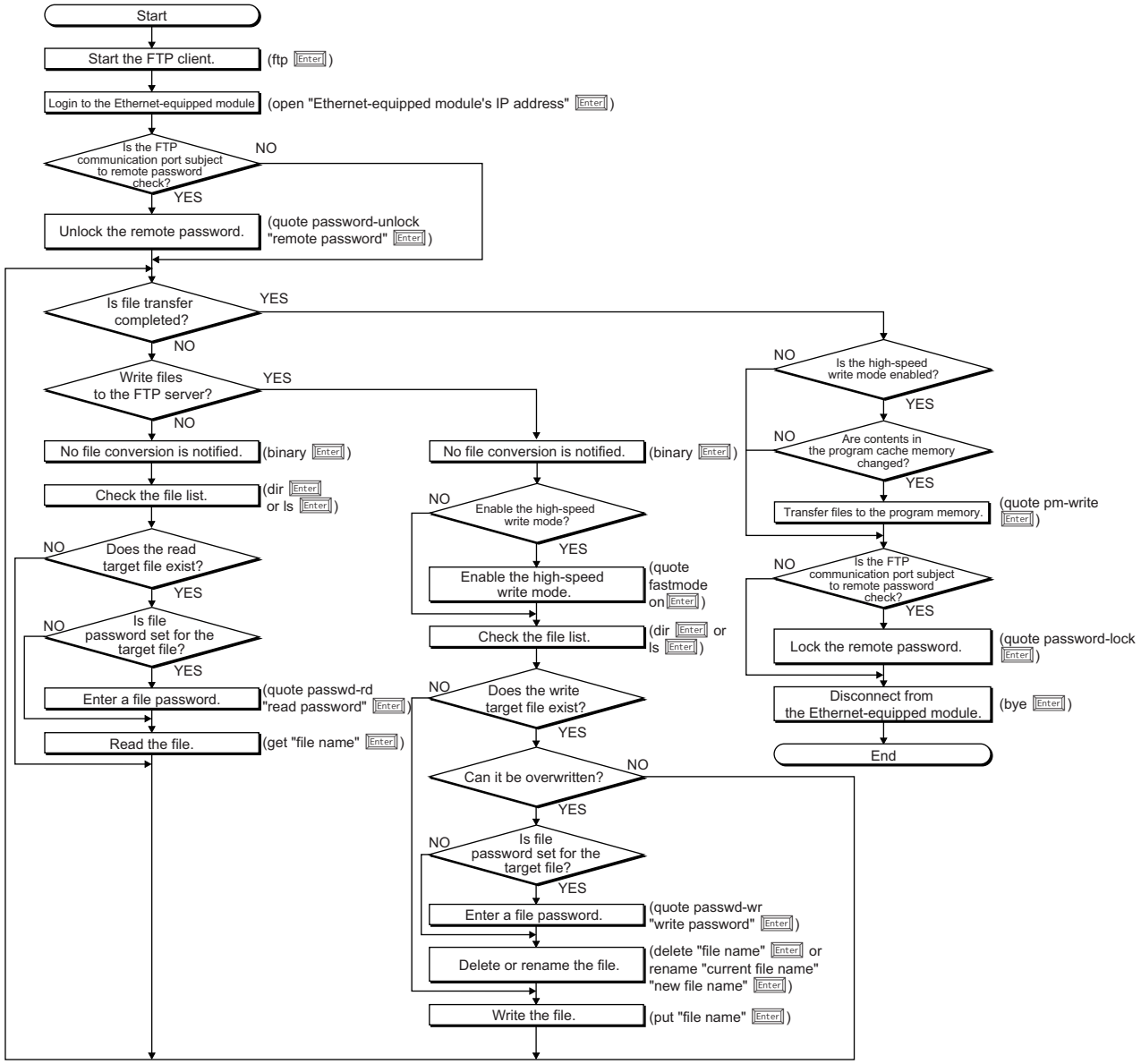
## Setting in the Ethernet-equipped module side

To use the file transfer function (FTP server), set "Use" for "FTP Server" in "FTP Server Settings" under "Application Settings". For details on other setting items, refer to the following.

☞ Page 338 FTP Server Settings

## Operations on external device (FTP client) side

This section describes the procedures and required processes on the external device side for using the Ethernet-equipped module's file transfer function (FTP server). The FTP commands and input format used for the operation are shown in the explanation. ("Enter" means to input CR, Enter or the Return key.)

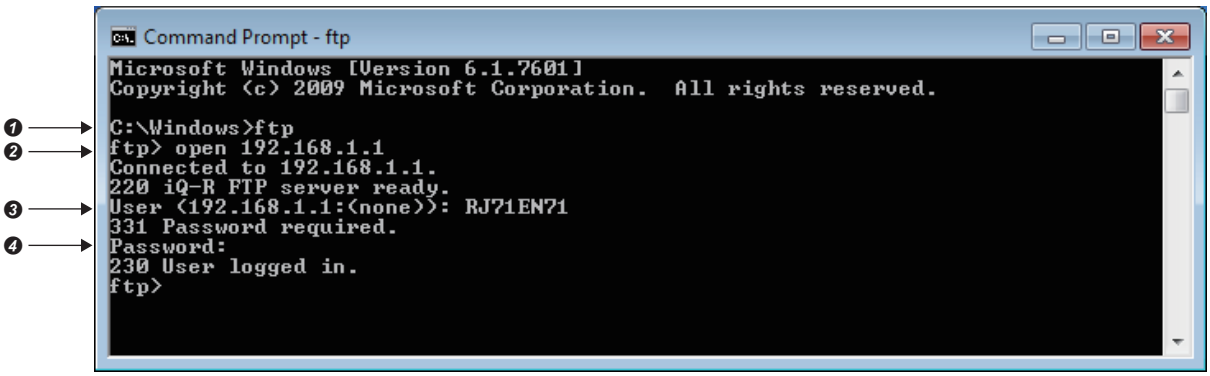


## Logging into Ethernet-equipped module

This section describes the steps from starting FTP and logging into the Ethernet-equipped module.

Ex.

Start FTP from the Microsoft® Windows® command prompt.




❶ FTP start (FTP Enter)
❷ Connect with FTP server (open Ethernet-equipped module IP address Enter)
❸ Specify login name (login name Enter)
❹ Specify password (password Enter)

Use the login name and password that are set in "FTP Server Settings" under "Application Settings". When the Ethernet-equipped module (FTP server) receives the login name and password from the external device (FTP client), it checks that the login name and password are correct.

If the login name and password are correct, transfer of the files to the Ethernet-equipped module is permitted. If incorrect, file transfer is not permitted.

## Locking and unlocking the remote password

If the FTP communication port is specified as a remote password check target with the remote password setting, unlock the remote password with the following command. (quote password-unlock remote password Enter) When finished, lock the remote password with the following command. (quote password-lock Enter)

**Point**

If the FTP communication port is specified as a remote password check target, some commands cannot be used until the remote password is unlocked.

For details on the commands that can be used in the locked state, and those that can only be used in the unlock state, refer to the following.

(☞ Page 122 FTP command list)

## Inputting the file password

If a file password is set for the target file, the file password must be input with the following command before the file can be accessed.

- Write password (quote passwd-wr write password Enter)
- Read password (quote passwd-rd read password Enter)

# Files that can be transferred with FTP

For files that can be transferred (read, written, or deleted) with the file transfer function (FTP server), refer to the file operation descriptions in the manual for the CPU module used.

# FTP command

## FTP command list

The FTP client side commands supported by the Ethernet-equipped module are shown below.

○: Executable[*1], ×: Not executable[*2]

| Command | Function | CPU module status | | | Remote password | |
|---|---|---|---|---|---|---|
| | | STOP | RUN | | Unlocked[*4] | Locked[*4] |
| | | | Enable[*3] | Disable[*3] | | |
| binary[*5] | Notifies that the file will be transferred without conversion. | ○ | ○ | ○ | ○ | × |
| bye | Closes and ends the connection with the FTP server. | ○ | ○ | ○ | ○ | ○ |
| close | Closes the connection with the FTP server. | ○ | ○ | ○ | ○ | ○ |
| delete | Deletes the CPU module file. | ○ | ○[*6] | × | ○ | × |
| dir | Displays the CPU module file information. | ○ | ○ | ○ | ○ | × |
| get | Reads a file from the CPU module. | ○ | ○ | ○ | ○ | × |
| ls | Displays the CPU module file name. | ○ | ○ | ○ | ○ | × |
| mdelete | Deletes the CPU module file. | ○ | ○[*6] | × | ○ | × |
| mdir | Stores the CPU module file information in the file. | ○ | ○ | ○ | ○ | × |
| mget | Reads a file from the CPU module. | ○ | ○ | ○ | ○ | × |
| mls | Reads a file from the CPU module. | ○ | ○ | ○ | ○ | × |
| mput | Writes the file to the CPU module. | ○ | ○ | × | ○ | × |
| open | Connects to the FTP server. | ○ | ○ | ○ | ○ | ○ |
| put | Writes the file to the CPU module. | ○ | ○ | × | ○ | × |
| pwd | Displays the current directory of the CPU module. | ○ | ○ | ○ | ○ | × |
| quit | Closes and ends the connection with the FTP server. | ○ | ○ | ○ | ○ | ○ |
| quote | Sends the FTP server's subcommand.[*7] | ○ | ○ | ○ | ○ | ○ |
| rename | Changes the CPU module file name. | ○ | ○ | × | ○ | × |
| user | Inputs the user name and password for the Ethernet-equipped module. | ○ | ○ | ○ | ○ | ○ |

*1 The command may not be executed depending on the file type. (☞ Page 121 Files that can be transferred with FTP)

*2 If executed, the process completes abnormally.

*3 Shows the "Allow Online Change" setting in "FTP Server Settings" under "Application Settings". If an illegal command is executed while write is prohibited during RUN, the process completes abnormally.

*4 Shows whether the command can be executed when the FTP communication port is specified as a remote password check target port with the Ethernet-equipped module mounted station's remote password setting. For details on the remote password, refer to the following.
(☞ Page 178 Remote password)

*5 The Ethernet-equipped module is set to transfer the file automatically without conversion. (Binary code fixed)
Therefore, the file is transferred with binary code regardless of the "Communication Data Code" setting in "Own Node Settings" under "Basic Settings".

*6 The parameter file and program file cannot be deleted when the CPU module is in the RUN state.

*7 Only the subcommands dedicated for the Ethernet-equipped module can be used. For the subcommands can be used, refer to the following.
(☞ Page 123 Subcommands usable with quote command)

## ■Subcommands usable with quote command

This section describes the Ethernet-equipped module dedicated commands added to the quote command and used.

When executing this command from the FTP client, input the subcommand after the quote command.

("Enter" means to input CR, Enter or the Return key.)

**Ex.**

Executing the STOP command

Input the following at the command prompt.

quote stop Enter

The following table lists the subcommands can be used.

○: Executable, ×: Not executable*1

| Command | Function | CPU module status | | | Remote password | |
|---|---|---|---|---|---|---|
| | | STOP | RUN | | Unlocked | Locked |
| | | | Write enable | Write prohibit | | |
| change | Shows or changes the CPU module file attributes. | ○ | ○ | × | ○ | × |
| cpuchg*2 | Specifies the access destination CPU module when using a multi-CPU system configuration. | ○ | ○ | ○ | ○ | × |
| fastmode*5 | Enables or disables the high-speed write mode for the CPU module, or displays its status. | ○ | ○ | ○ | ○ | × |
| passwd-rd | Sets, shows or clears the file password (read password). | ○ | ○ | ○ | ○ | × |
| passwd-wr | Sets, shows, or clears the file password (write password). | ○ | ○ | ○ | ○ | × |
| password-lock | Changes the remote password from the unlock state to the lock state. | ○ | ○ | ○ | ○ | ×*4 |
| password-unlock | Changes the remote password from the lock state to the unlock state. | ○ | ○ | ○ | ○ | ○ |
| path-delimiter*5 | Changes the directory delimiter of the file path of the CPU module from "\" to "/".*6 | ○ | ○ | ○ | ○ | × |
| path-delimiter off*5 | Turn off file setting changing for the file path name when a file is specified by the path-delimiter.*6 | ○ | ○ | ○ | ○ | × |
| path-delimiter status*5 | Displays the setting status of the file path name when a file is specified.*6 | ○ | ○ | ○ | ○ | × |
| status | Displays the CPU module operation information. | ○ | ○ | ○ | ○ | × |
| run | Sets the CPU module to the RUN state. | ○*3 | ○ | ○ | ○ | × |
| stop | Sets the CPU module to the STOP state. | ○ | ○ | ○ | ○ | × |
| pm-write | Writes the program memory. | ○ | × | × | ○ | × |

*1 If executed, the process completes abnormally.

*2 This subcommand cannot be used when the connection destination is a CPU module.

*3 This subcommand is invalid when the RUN/STOP/RESET switch of the CPU module is set to STOP.

*4 Even if the subcommand is executed, the remote password remains locked with no error occurred.

*5 This subcommand can be used only when the connection destination is a programmable controller CPU. Before using this subcommand, check the versions of the CPU module used. (☞ Page 586 Added and Enhanced Functions)

*6 For function details, refer to the following.

☞ Page 125 FTP server function file path name switching

## Specifying an FTP command

This section describes the method of specifying the files specified with the FTP command on the FTP client (external device side) supported by the Ethernet-equipped module.

With the CPU module, the drive name and file name are distinguished when specifying the file.

When specifying a file with FTP, specify the target file with the following arrangement.*1

| Item | Description |
|---|---|
| Specification format | [Drive name:\]Folder name*2\File name.Extension |
| Example | 3:\MAINSEQ1.QDR (other than drive 2)<br>2:\LOGGING\LOG01\00000001\LOG01_00000001.CSV (drive 2) |
| Specification details | Refer to the following.<br>☞ Page 124 Drive name (drive No.),Page 124 Folder name, file name, and extension |

*1 The default directory delimiter is "\". To change the directory delimiter to "/", refer to the following.
☞ Page 125 FTP server function file path name switching

*2 The file name can be specified only when specifying a file stored in drive 2.

**Point**

The section enclosed in square brackets [ ] in the specification format can be omitted.

### ■Drive name (drive No.)

Specify the drive name of the target memory for transferring the file.

For the correspondence of the target memory and drive name, refer to the manual for the CPU module used.

Omitting the drive name is the same as specifying the program memory (drive 4). If, however, FTP server function file path name switching is enabled, the setting is valid. (☞ Page 125 FTP server function file path name switching)

### ■Folder name, file name, and extension

Specify the folder name and file name according to the rules provided in the manual for the CPU module used.

The name determined for the CPU module is assigned for the extension.

When using an FTP command that can be used for multiple files, specify the file name and extension with the wild card "*" or "?". (Depending on the FTP client, there may be additional restrictions to the characters that can be used for the file name.)

*: All files with the arbitrary character string (including none) are targeted from the position specified with "*".

?: All files with the arbitrary character string (excluding none) are targeted from the position specified with "?". ("?" can be used multiple times.)

## FTP server function file path name switching

The directory delimiter of the file path specified using the FTP command is changed from "\" to "/" in the CPU module (built-in Ethernet port part).

In addition, the omission of the drive number in the file path name can be set.

| Change procedure | Description |
|---|---|
| Using the special relay/special register | Changing the special relay SM1512 (FTP server function file path name switching setting enable flag) changes the directory delimiter.<br>This method is used when FTP clients use only Linux OS and "/" is to be set for the directory delimiter. |
| Using the FTP command | Specifying the subcommand (path-delimiter) for the quote command changes the directory delimiter of the file path name of the CPU module.<br>This method is used when FTP clients use Windows OS or Linux OS and the directory delimiter is to be changed every time the server is accessed by the client. |

The following table lists how to specify CPU module files after changing the directory delimiter to "/".

| Item | Description |
|---|---|
| Specification format | [Drive name:/]File name.Extension |
| Example | 3:/MAINSEQ1.QDR |

- After the directory delimiter is changed to "/", directory delimiter commands including "\" can be received by the CPU module. However, the directory delimiter that is sent from the CPU module is "/".
- When changing the directory delimiter again, the contents set later are valid.
- When executing the FTP command to specify a file path name using an FTP client with Linux OS, the same folder configuration must be used for the CPU module and the personal computer.

**Restriction**

Only the programmable controller CPU can be used for the function. Before using this function, check the version of the CPU module used. ☞ Page 586 Added and Enhanced Functions

### ■Using the special relay/special register

- Turning on the SM1512 (FTP server function file path name switching setting enable flag) changes the directory delimiter of the CPU module file path name to "/". Storing the setting value in the SD1512 (FTP server function file path name switching setting) enables omission of the drive number in the file path name.
- While the SM1512 is on, the directory delimiter change is valid. Turning off the SM1512 clears the change and returns the directory delimiter from "/" to "\".
- When an improper value is set, the SM1513 (FTP server function file path name switching setting error) is turned on and the error cause is stored in the SD1513 (FTP server function file path name switching setting error cause). When there is no error in the setting value, the SM1513 is turned off.
- The current setting status is stored in the SD1514 (FTP server function file path name switching setting status) When the SM1512 is off, 0 is stored in the SD1514.

| Special relay | Name |
|---|---|
| SM1512 | FTP server function file path name switching setting enable flag |
| SM1513 | FTP server function file path name switching setting error |

| Special register | Name |
|---|---|
| SD1512 | FTP server function file path name switching setting |
| SD1513 | FTP server function file path name switching setting error cause |
| SD1514 | FTP server function file path name switching setting status |

For details on the special relay and special register, refer to the manual for the CPU module used.

**Restriction**

Do not turn the SM1512 off and on or on and off while the FTP command is executing. Doing so may cause malfunction.

## ■Using the FTP command

- When executing the quote command with a subcommand of the path-delimiter [mode], the directory delimiter of the CPU module file path name is changed from "\" to "/". Inputting the setting value to [mode] enables omission of the drive number in the file path.
- Executing the path-delimiter off command clears the setting.
- Executing the path-delimiter status command enables the current settings to be checked.

| FTP command (Sub command of the quote command) | Function |
|---|---|
| path-delimiter | Changes the setting of the file path name when a file is specified.<br>(After executing the command, the SM1512 is turned on and the current setting value is stored in the SD1514.) |
| path-delimiter off | Turns off file setting changing for the file path name when a file is specified. |
| path-delimiter status | Displays the setting status of the file path name when a file is specified. |

For details on the quote command, refer to the following.

☞ Page 130 Ethernet-equipped module dedicated subcommand

**Point**

The current settings can also be checked using the SM1512 (FTP server function file path name switching setting enable flag) and the SD1514 (FTP server function file path name switching setting status).

## Details of FTP command

This section describes the FTP commands on the FTP client side supported by the Ethernet-equipped module, and the methods of using those commands.

**Point**

- Note that depending on the client side FTP application, some of the FTP commands may not operate as described in this manual. Refer to the manual for the FTP client, and check the functions, operation methods, and so on.
- The specification format and result of the command may differ depending on the FTP client used.
- The section enclosed in square brackets [ ] in the specification format can be omitted.

### ■FTP server support command

| Command name | Description | |
|---|---|---|
| binary | Function | Notifies the FTP server that the file will be transferred without conversion. The return code and kanji codes are also not converted.<br>These settings are automatically applied to the Ethernet-equipped module. |
| | Specification format | binary (abbreviated: bin) |
| bye | Function | Closes the connection with the FTP and quits the FTP. |
| | Specification format | bye |
| | Same function | quit |
| close | Function | Closes the connection with the FTP server. |
| | Specification format | close |
| delete | Function | Deletes files stored in the CPU module. |
| | Specification format | delete "file path name" |
| | Example | When deleting files stored in the SD memory card<br>delete 2:\MAINSEQ1.PRG |
| | Similar command | mdelete |
| dir | Function | Displays the file name, date of creation and volume of files stored in the CPU module. |
| | Specification format | dir [drive name:\] |
| | Example | When displaying the detailed information of files stored in the data memory<br>dir 4:\ |
| | Similar command | ls |
| get | Function | Reads a file from the CPU module. |
| | Specification format | get "source file path name" [destination file path name] |
| | Example 1 | When reading files stored in the SD memory card and store with same file name<br>get 2:\MAINSEQ1.PRG |
| | Example 2 | When reading files stored in the SD memory card and store with different file name<br>get 2:\SEQ1BAK.PRG SEQ\SEQ10LD.PRG |
| | Caution | • If the destination file path name (FTP client side) is not specified, the file is stored in the FTP client side with the same file name as the source file name (CPU module side).<br>• The transfer destination is in the currently connected directly when FTP is started and connected.<br>• When the command is executed in an FTP client with Linux OS, the drive number in the source file path name must be omitted. Set the drive number in the file path name to be omitted in advance using the FTP server function file path name switching, and specify as follows. (☞ Page 125 FTP server function file path name switching)<br>Example: "get /SEQ10LD.PRG" |
| ls | Function | Displays the names of files stored in the CPU module. |
| | Specification format | ls [drive name:\] |
| | Example | When displaying the detailed information of files stored in the data memory<br>ls 4:\ |
| | Similar command | dir |
| | Caution | When the command is executed in an FTP client with Linux OS, the results are the same as the dir command. (The mls command corresponds to the ls command for an FTP client with Windows OS.) |

| Command name | Description | |
|---|---|---|
| mdelete | Function | Deletes files stored in the CPU module.<br>When deleting multiple files, specify the file name and extension in the file path name with wild cards (*, ?). |
| | Specification format | mdelete "file path name" (abbreviated: mdel) |
| | Example | When deleting all files with "CSV" extension from files stored in SD memory card<br>mdelete 2:*.CSV |
| | Caution | When the command is executed in an FTP client with Linux OS, omit the drive number in the file path name and add "/" to the beginning of the file path name. Set the drive number in the file path name to be omitted in advance using the FTP server function file path name switching, and specify as follows. (☞ Page 125 FTP server function file path name switching)<br>Example: "mdelete /*.csv" |
| | Similar command | delete |
| mdir | Function | Stores the detailed information (file name, date of creation, volume) of the file stored in the CPU module in the FTP client side file as log data. |
| | Specification format | mdir "source drive name":\"destination file path name" |
| | Example | When storing the detailed information of file stored in data memory into 20120702.LOG file<br>mdir 4:\20120702.LOG |
| | Caution | • Always specify "\" immediately after the source drive name.<br>• Always specify the source drive name when specifying the destination file path name (FTP client side).<br>• If the destination file path name is not specified, the file is stored with the file name determined by the FTP client's FTP application.<br>• The transfer destination is in the currently connected directly when FTP is started and connected. |
| | Similar command | mls |
| mget | Function | Reads a file from the CPU module.<br>When reading multiple files, specify the file name and extension in the file path name with wild cards (*, ?).<br>When reading multiple files, receive is confirmed before transferring each file. |
| | Specification format | mget "file path name" |
| | Example | When reading all files with "CSV" extension from files stored in SD memory card<br>mget 2:*.CSV |
| | Caution | • The read file is stored with the same file name in the FTP client side. The storage destination is in the current connection directory when the FTP is started and connected.<br>• When the command is executed in an FTP client with Linux OS, omit the drive number in the file path name and add "/" to the first of the file path name. Set the drive number in the file path name to be omitted in advance using the FTP server function file path name switching, and specify as follows. (☞ Page 125 FTP server function file path name switching)<br>Example: "mget /*.csv" |
| mls | Function | Stores the file name of the file stored in the CPU module in the FTP client side file as log data. |
| | Specification format | mls "source drive name":\"destination file path name" |
| | Example | When storing the file name of file stored in SD memory card into 20120702.LOG file<br>mls 2:\20120702.LOG |
| | Caution | • Always specify "\" immediately after the source drive name.<br>• Always specify the source drive name when specifying the destination file path name (FTP client side).<br>• If the destination file path name is not specified, the file is stored with the file name determined by the FTP client's FTP application.<br>• The transfer destination is in the currently connected directly when FTP is started and connected. |
| | Similar command | mdir |
| mput | Function | Writes the file to the CPU module.<br>When writing multiple files, specify the file name and extension in the file path name with wild cards (*, ?).<br>When writing multiple files, send is confirmed before transferring each file. |
| | Specification format | mput "source file path name" |
| | Example | When writing all files with "PRG" extension<br>mput*.PRG |
| | Caution | • The storage destination file name is the same as the FTP client side.<br>• The transmission destination is the data memory (drive 4 "4:\"). When the file path name is changed using the FTP server function file path name switching, the setting is valid. (☞ Page 125 FTP server function file path name switching) |

| Command name | Description | |
|---|---|---|
| open | Function | Specifies the host name or IP address and port number on the FTP server side, and connects with the FTP server. |
| | Specification format | open "host name" [port number]<br>open "IP address" [port number]<br>• Host name: Host name set with Microsoft® Windows® hosts file<br>• IP address: IP address of the Ethernet-equipped module side<br>• Port number: Port number to be used (If omitted, port number 21 is used for operation) |
| | Example 1 | When specifying the host name and connecting to the FTP server<br>open HOST |
| | Example 2 | When specifying the IP address and connecting to the FTP server<br>open 192.0.1.254 |
| | Caution | The IP address can be specified to create a connection when starting the FTP. |
| put | Function | Writes the file to the CPU module. |
| | Specification format | put "source file path name" [destination file path name] |
| | Example 1 | When writing the param.PRM file to the SD memory card with the same file name<br>put param.PRM 2:\param.PRM |
| | Example 2 | When writing the param.PRM file to the SD memory card with a different file name<br>put param.PRM 2:\param1.PRM |
| | Caution | • If the directory is not specified with the source file path name (FTP client side), the file in the current connection directory when the FTP server is started and connected is written.<br>• If the destination file path name (FTP server side) is not specified, the file is transferred to the data memory (drive 4 "4:\"). |
| pwd | Function | Displays current directory name of the CPU module. |
| | Specification format | pwd |
| | Caution | "\" is displayed as the pwd command execution results. |
| quit | Function | Closes the connection with the FTP and quits the FTP. |
| | Specification format | quit |
| | Similar command | bye |
| quote | Function | Sends the FTP server's subcommand (Ethernet-equipped module dedicated subcommand). |
| | Specification format | quote |
| | Example | quote password-lock |
| | Caution | Only the Ethernet-equipped module dedicated subcommand can be specified. Refer to the following.<br>(☞ Page 130 Ethernet-equipped module dedicated subcommand) |
| rename | Function | Changes the CPU module file name. |
| | Specification format | rename "old file path name" "new file path name"<br>(abbreviated: ren) |
| | Example | When changing the name of a file stored in the SD memory card<br>rename 2:\param.PRM 2:\param1.PRM |
| | Caution | The following response code appears when the file is successfully renamed.<br>350 Need more info.<br>250 Rename successful. |
| user | Function | Inputs the user name and password for the connected FTP server. |
| | Specification format | user "user name" [FTP password]<br>• User name: Login name set with Ethernet-equipped module parameters<br>• FTP password: FTP password set with Ethernet-equipped module parameters |
| | Example 1 | When specifying the user name<br>user RJ71EN71 |
| | Example 2 | When specifying the user name and password<br>user RJ71EN71 RJ71EN71 |

## ■Ethernet-equipped module dedicated subcommand

This section describes the Ethernet-equipped module dedicated subcommands that are added to the FTP command's quote command and sent.

| Command name | Description | |
|---|---|---|
| change | Function | Shows or changes the attributes of files stored in the CPU module. |
| | Specification format 1 | When displaying the file attributes<br>quote change file path name<br>One of the following is displayed as the execution results when the process ends normally.<br>• When specified file is read-only: -----R<br>• When specified file is writable and readable: -----W |
| | Specification format 2 | When changing the file attributes<br>quote change "file path name" "attribute"<br>Specify the attributes with one of the following methods.<br>• When changing to a read-only file: r<br>• When changing to a writable and readable file: w |
| | Example 1 | When displaying the attributes of files stored in the SD memory card<br>quote change 2:\PRG1.PRG |
| | Example 2 | When changing the attributes of files stored in the SD memory card<br>quote change 2:\PRG1.PRG r |
| cpuchg | Function | Specifies the access destination CPU module when using a multi-CPU system configuration.<br>This may also display the current access destination CPU module. |
| | Specification format | quote cpuchg [access destination CPU module]<br>Access destination CPU module: Specify the CPU module from which files are to be accessed.<br>Shows the method for specifying the access destination CPU module.<br>• No input: Checks the current access destination CPU module.<br>• controlPLC: Specifies the control CPU as the access destination CPU module.<br>• no1: Specifies the CPU No.1 as the access destination CPU module.<br>• no2: Specifies the CPU No.2 as the access destination CPU module.<br>• no3: Specifies the CPU No.3 as the access destination CPU module.<br>• no4: Specifies the CPU No.4 as the access destination CPU module.<br>The execution results when the access is completed normally are shown in the following manner.<br>• When setting access destination CPU module: 200 Command Okey<br>• When displaying access destination CPU module: 200 Target cpu is "access destination CPU module".<br>The following appears if the specified access destination CPU module is not found.<br>• 557 Cpu not found |
| | Example 1 | When specifying control CPU as access destination CPU module<br>quote cpuchg controlPLC |
| | Example 2 | When displaying the currently set access destination CPU module<br>quote cpuchg |
| | Caution | The access destination CPU module after logging into the RJ71EN71 with the open/user command is the RJ71EN71 control CPU. |

| Command name | Description | |
|---|---|---|
| fastmode | Function | Enables the high-speed write mode.<br>Since both of the following processing are not performed in high-speed write mode, the program (.PRG) and FB file (.PFB) can be transferred to the "$MELPRJ$" folder at high speed.<br>• To create a temporary file (FTP_I***_#.TMP)<br>• To automatically transfer files that have been written in the program cache memory to the program memory. |
| | Specification format | quote fastmode [mode]<br>Mode: Specify whether to enable or disable the high-speed write mode.<br>• on: Enabled<br>• off: Disabled<br>One of the following is displayed as the execution results when the process ends normally.<br>• When the status of high-speed write mode is switched to on (enabled): 200 Command successful<br>• When the status of high-speed write mode is switched to off (disabled): 200 Command successful<br>• If status is displayed when high-speed write mode is enabled: 200 Fastmode is ON<br>• If status is displayed when high-speed write mode is disabled: 200 Fastmode is OFF |
| | Example 1 | When enabling the high-speed write mode<br>quote fastmode on |
| | Example 2 | When disabling the high-speed write mode<br>quote fastmode off |
| | Example 3 | When displaying status of the high-speed write mode<br>quote fastmode |
| | Caution | • The high-speed write mode is disabled when logging into the CPU module.<br>• In high-speed write mode, files are not automatically transferred when contents in the program cache memory are changed with the file write command (put, mput), file delete command (delete, mdelete), or file rename command (rename). Therefore, execute the file transfer command (to program memory) (quote pm-write) after changing the contents. If the CPU module is reset without executing the file transfer command (to program memory) after the changing, incorrect program files remain in the program memory. In this case, delete the incorrect program files with the file delete command (delete, mdelete) and transfer files again. |
| passwd-rd | Function | Sets the read password (file password) registered as the file transfer target file to the Ethernet-equipped module.<br>Shows/clears the read password set in FTP.<br>Use this command only when a read password is registered for the file transfer target file. The CPU module checks the password when accessing the specified file. |
| | Specification format | quote passwd-rd [read password]<br>The following appears as the execution results when the command ends normally.<br>• When setting a read password: 200 Command successful<br>• When displaying the read password: 200 Read-password is "read password"<br>• When clearing the read password: 200 Command successful<br>• When displaying the state with a read password not set: 200 Read-password is not set. |
| | Example 1 | When specifying the read password (ABCD1234@efgh)<br>quote passwd-rd ABCD1234@efgh |
| | Example 2 | When displaying the read password currently set in FTP<br>quote passwd-rd |
| | Example 3 | When clearing the read password currently set in FTP<br>quote passwd-rd c, or quote passwd-rd C |
| | Caution | • One read password can be set for the FTP of the Ethernet-equipped module. When the file transfer target file changes and when a read password is registered for the change destination file, reset the read password for the target file.<br>• The read password is initialized (cleared) when logging into the Ethernet-equipped module. |

| Command name | Description | |
|---|---|---|
| passwd-wr | Function | Sets the write password (file password) registered in the file transfer target file to the Ethernet-equipped module.<br>Shows/clears the write password set in FTP.<br>Use this command only when a write password is registered for the file transfer target file. The CPU module checks the password when accessing the specified file. |
| | Specification format | quote passwd-wr [write password]<br>The following appears as the execution results when the command ends normally.<br>• When setting the write password: 200 Command successful<br>• When displaying the write password: 200 Write-password is "Write password"<br>• When clearing the write password: 200 Command successful<br>• When displaying the state with the write password not set: 200 Write-password is not set. |
| | Example 1 | When specifying the write password (ABCD1234@efgh)<br>quote passwd-wr ABCD1234@efgh |
| | Example 2 | When displaying the write password currently set in the FTP<br>quote passwd-wr |
| | Example 3 | When clearing the write password currently set in the FTP<br>quote passwd-wr c, or quote passwd-wr C |
| | Caution | • One write password can be set for the FTP of the Ethernet-equipped module. When the file transfer target file changes and when a write password is registered for the change destination file, reset the write password for the target file.<br>• The write password is initialized (cleared) when logging into the Ethernet-equipped module. |
| password-unlock | Function | Specifies the remote password set for the CPU module and unlocks the password.<br>This command is used when FTP communication port is specified as a remote password check target port. |
| | Specification format | quote password-unlock [remote password]<br>• Remote password: Specifies the remote password set in the CPU module parameters.<br>The following appears as the execution results when the command ends normally.<br>200 Command Okey<br>The following appears when the remote password does not match.<br>556 Password Error<br>The following appears if another command is requested before the remote password is unlocked.<br>555 Password Locked |
| | Example | When specifying a remote password (123456)<br>quote password-unlock 123456 |
| | Caution | • If the FTP communication port is specified as a remote password check target port when logging in, the password will be locked.<br>• The CPU module files can be accessed by executing this command and unlocking before starting the various FTP operation.<br>• If the FTP communication port is not specified as a remote password check target port, the processing will complete normally when the remote password is unlocked. |
| password-lock | Function | Locks the remote password function set for the CPU module.<br>This command is executed when the FTP communication port is specified as a remote password check target port. |
| | Specification format | quote password-lock<br>The following appears as the execution results when the command ends normally.<br>200 Command Okey |
| | Example | When locking the remote password<br>quote password-lock |

| Command name | Description | |
|---|---|---|
| path-delimiter | Function | Changes the directory delimiter of the file path name when a file is specified.<br>Also, the drive number can be omitted depending on the mode specification. |
| | Specification format | quote path-delimiter [mode]<br>Mode: Specify one of the following 4 digit hexadecimal number.<br><br>0 \| 0 \| 0/2/3/4 \| 0/2<br><br>0000 or blank:"/" is set for the directory delimiter of the file path name. (When the FTP clients use Linux OS and other OS)<br>0022: When "/" is set for the directory delimiter of the file path name and the file path is specified from the FTP client, the drive number can be omitted. When the drive number is omitted, Drive 2 (SD memory card) is used.<br>0032: When "/" is set for the directory delimiter of the file path name and the file path is specified from the FTP client, the drive number can be omitted. When the drive number is omitted, Drive 3 (device/label memory) is used.<br>0042: When "/" is set for the directory delimiter of the file path name and the file path is specified from the FTP client, the drive number can be omitted. When the drive number is omitted, Drive 4 (data memory) is used.<br>When drive number omission is set and the CPU module receives the command with the specified drive number, the specified drive number in the received command is prioritized.<br>The following appears as the execution results when the command ends normally.<br>200 path-delimiter is ON [mode]<br>The following appears when the mode setting value is out of range.<br>504 Parameter not accepted, not implemented.<br>To clear the setting, use the path-delimiter off command. |
| | Example 1 | When "/" is set for the directory delimiter<br>quote path-delimiter 0000 |
| | Example 2 | When "/" is set for the directory delimiter and the drive number in the file path name is omitted<br>quote path-delimiter 0022: Drive 2<br>quote path-delimiter 0032: Drive 3<br>quote path-delimiter 0042: Drive 4 |
| path-delimiter off | Function | Clears the setting change of the file path name by the path-delimiter. |
| | Specification format | quote path-delimiter off<br>The following appears.<br>200 path-delimiter is OFF |
| path-delimiter status | Function | Displays the setting status of the file path name when a file is specified. |
| | Specification format | quote path-delimiter status<br>The following appears as the execution results when the command ends normally.<br>• When ON is set: 200 path-delimiter is ON (mode)<br>• When OFF is set: 200 path-delimiter is OFF |
| | Caution | The 4 digit hexadecimal number is displayed as the setting value for "ON (mode)" when ON is set. |
| run | Function | Sets the CPU module to the RUN state (remote RUN). Clearing of the device memory can be specified when entering the RUN state. |
| | Specification format | quote run [mode [clear mode]]<br>Mode: Specify whether to forcibly execute remote RUN<br>• 0: Normal RUN (default)<br>• 1: Forced RUN<br>Clear mode: Specify to clear (initialize) CPU module device memory at start of operation during remote RUN<br>• 0: Do not clear device (default)<br>• 1: Clear other than latch area<br>• 2: Clear all including latch range<br>The following message appears as the execution results when the process ends normally.<br>200 Command successful |
| | Example 1 | When executing the remote RUN with normal RUN and do not clear device memory specified<br>quote run |
| | Example 2 | When executing the remote RUN with forced RUN and do not clear device memory specified<br>quote run 1 |
| | Example 3 | When executing the remote RUN with forced RUN and clear other than latch area memory specified<br>quote run 11 |
| | Caution | • Forced RUN is used only when remote RUN of the CPU module is not possible due to trouble in the device that remote STOPPED the CPU module, and the CPU needs to be forcibly remote RUN with another device. With normal RUN, a module that is in the STOP/PAUSE state executed by another device cannot be set to the RUN state.<br>• Specify clearing of the device memory by the clear mode at the start of operation according to the system specifications. After the specified clear is executed, the CPU module will RUN according to the device's default settings. |

| Command name | Description | |
|---|---|---|
| status | Function | Displays the CPU module operation information.<br>This command is used to check the operation information when transferring files. |
| | Specification format | quote status<br>One of the following is displayed as the execution results when the process ends normally.<br>When CPU module is in RUN state: "RUN"<br>When CPU module is in STOP state: "STOP"<br>When CPU module is in PAUSE state: "PAUSE" |
| stop | Function | Set the CPU module to the STOP state (remote STOP). |
| | Specification format | quote stop<br>The following message appears as the execution results when the process ends normally.<br>200 Command successful |
| | Caution | Write to the program memory after setting the CPU module to the STOP state with this command. |
| pm-write | Function | Transfers the files written in the program cache memory to the program memory so that the files are held even at power-off. |
| | Specification format | quote pm-write<br>The following message appears as the execution results when the process ends normally.<br>200 Command successful |
| | Caution | • The MELSEC iQ-R series CPU module automatically transfers the files to program memory after writing them in the program cache memory. Therefore, the files are held at power-off without using this command. If this command is used in the FTP client program, the program does not need be modified when replacing the MELSEC-Q series.<br>• Set the CPU module to the STOP state before writing. |

# Precautions



## Precautions for designing system

Design the system (such as configuration of interlock circuits in the program) so that the entire system always functions properly during file transfer to the operating system and during status control of the programmable controller.

## FTP client

- The FTP command specifications may differ from this manual depending on the FTP client. In this case, refer to the manual for the FTP client and check the functions and operation methods.
- FTP operations are not possible from Microsoft® Internet Explorer®. If attempted, Internet Explorer® will issue an error.
- Specify the IP address for the FTP command without zero fill. (Do not set "1" as "001".)

## Processing on CPU module side

- Only the drive of the CPU module on the own station can access files.
- Do not power off or reset it, or insert/eject the SD memory card during file access. The file could be damaged if these are attempted.
- Do not manipulate the files from a peripheral, such as an engineering tool, while accessing the files. (This also applies to online operations such as writing during RUN as the files are manipulated.) If the file is manipulated from another device during the file transfer function (FTP server) operation, the peripheral may issue an error. If the processing has been halted due to an error, re-execute the processing before quitting the FTP function.

## Communication processing

- If a timeout error occurs during file transfer, the TCP/IP connection will be closed. Log into the Ethernet-equipped module with the FTP client again before resuming file transfer.
- The existence of the external device is checked with the FTP connection.
- The file transfer processing time will differ according to the following factors: the load rate (line congestion status) of the Ethernet line, the number of connections used simultaneously (communication processing of other connections), and the system configuration (distance between the FTP server and FTP client, and how to access the CPU module).
- Only one external device (FTP client) can log into the Ethernet-equipped module at one time. If a connection is attempted from another FTP client in the login state, an error will occur without establishing the connection.
- If another communication function (MELSOFT connection or SLMP) is simultaneously executed with UDP/IP during file transfer with FTP, an error such as timeout may occur. Either communicate after the file is transferred, or communicate with TCP/IP.

## Writing files

- An existing file cannot be overwritten. Delete or rename an existing file with the file delete command (delete, mdelete) or the file rename command (rename) before writing files.
- A read-only file or a file locked by the function except for FTP cannot be written. If attempted, a write error occurs.
- A file cannot be transferred when the SD memory card used is protected. If attempted, a write error occurs.
- A temporary file (FTP_I***_#.TMP) is automatically created at a file writing. (*** is first three digits of start I/O number, and # is the hardware port number to which FTP client is connected.) The file is renamed the write target file name when a writing is completed, however, the temporary file may remain if the system is powered off or the CPU module is reset during a file writing. In this case, delete the temporary file with the file delete command (delete, mdelete).
- In high-speed write mode, a temporary file is not created at a file writing. Therefore, if the system is powered off or the CPU module is reset during a file writing, a file in the middle of the writing remains. In this case, delete this file with the file delete command (delete, mdelete) and write a file again.
- When the file register is set as a refresh device, do not perform any writing or deletion to the corresponding drive.
- When writing a large file to the SD memory card, set the CPU module to STOP. If writing is performed in the RUN state, a communication error may occur.
- The number of files that can be written into the specified drive is maximum [maximum number of files that can be written] - 1 file. For details on the maximum number of files that can be written to the CPU module, refer to the manual for the CPU module used.

## Deleting files

- Decide the timing for deleting the files for the entire system including the CPU module and engineering tool.
- Files with read-only attributes and files that are locked by a function other than FTP cannot be deleted. An error will occur if attempted.
- The file cannot be deleted if the SD memory card is protected. An error will occur if attempted.

## FTP password

The FTP password can be reset with the following procedure when it is lost.

1. Read the parameters from the CPU module with the engineering tool.
2. Click the [Restore the Default Settings] button in "Application Settings" to return all "Application Settings" to the default values.
3. Set the "FTP Server Settings" and "Application Settings" again.
4. Write the set parameters to the CPU module.
5. Enable the parameters by powering off and on or resetting the CPU module.

**Point**

When the FTP password is lost, all items set in "Application Settings" must be reset in addition to the "FTP Server Settings".

## Setting a firewall on the FTP client side

If the FTP communication is blocked by a firewall on the FTP client side, data cannot be exchanged from the FTP server. Check the firewall settings, enable FTP communication and then access the FTP server.

## Monitoring timer

When operation of files such as program (.PRG) and FB file (.PFB) is performed in a CPU module with a large program capacity, a timeout may occur due to long processing time on the CPU module. If a timeout occurs, increase the response monitoring timer value. (☞ Page 339 Response Monitoring Timer)

## Execution of the file transfer function (FTP server) during execution of another function

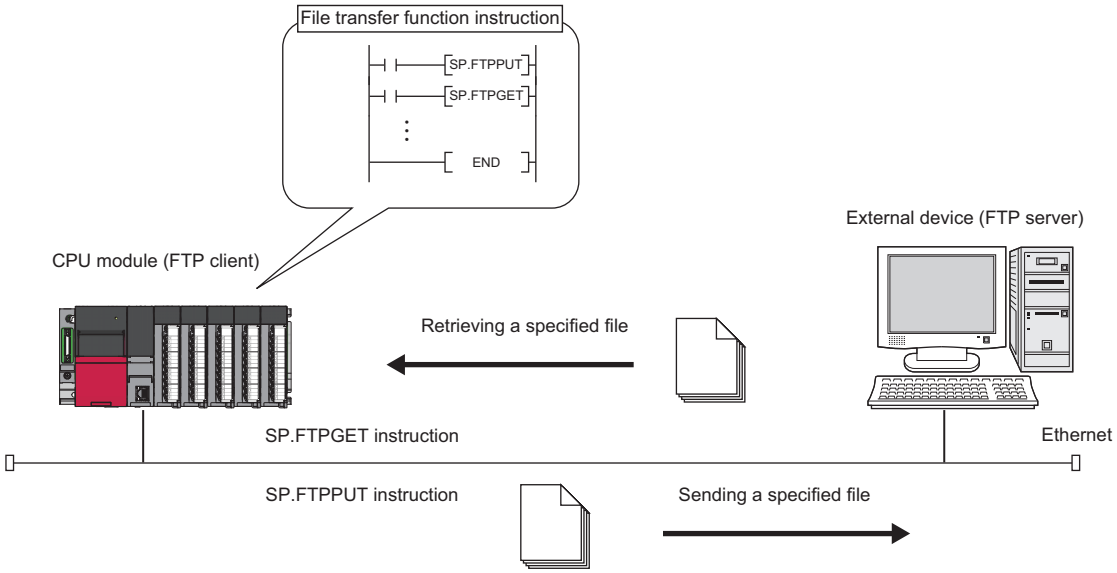
For some file transfer functions (FTP server), there are some functions that cannot be simultaneously executed. Check that the function that cannot be simultaneously executed is not being executed and then execute the file transfer function (FTP server).

| File transfer function (FTP server) | | Functions that cannot be simultaneously executed |
|---|---|---|
| File transfer function (FTP server) of the RJ71EN71 and RnENCPU (network part) | • Writing a file (put, mput, pm-write)<br>• Deleting a file (delete, mdelete)<br>• Changing a file name (rename)<br>• Changing a file attribute (change) | • CPU module data backup/restoration function<br>• iQ Sensor Solution data backup/restoration function |
| File transfer function (FTP server) of the CPU module (built-in Ethernet port part) | | |

# 1.10 File Transfer Function (FTP Client)

The CPU module becomes an FTP client and can execute file transfer with the FTP server connected to Ethernet using the file transfer function instruction.*1

The FTP server is specified with a module parameter or a control data of the file transfer function instruction.*2




*1 To execute this function, an FTP server is required. For details on an FTP server, refer to the manual for the server used.

*2 When the FTP server is specified with a control data of the file transfer function instruction, check the version of the CPU module. (Manual for the CPU module used)
For details, refer to the following.
MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks)

**Point**

When the data logging function is used, a data logging file can be transferred to the FTP server by using the data logging file transfer function. (Manual for the CPU module used)

**Restriction**

Before using the file transfer function (FTP client), check the version of the CPU module used. (Page 586 Added and Enhanced Functions)

## File transfer specifications

The following table summarizes the file transfer specifications of the file transfer function (FTP client).

| Item | Description |
|---|---|
| FTP server whose operation has been checked by Mitsubishi Electric*1 | Microsoft®Internet Information Services(IIS)<br>Target OS:<br>• Microsoft®Windows®10<br>• Microsoft®Windows®8.1<br>• Microsoft®Windows®8<br>• Microsoft®Windows®7 |
| Number of connectable FTP servers | 1 |
| FTP transfer mode | Binary mode |

*1 This item indicates that the FTP server operation has been verified when the FTP server is specified using a module parameter. For the FTP server operation verified when the FTP server is specified with the control data of the file transfer function instruction, refer to the following.
MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks)

# Transferable files



The following table lists the files that can be transferred using the file transfer function (FTP client).

○: Transferable, ×: Not transferable

| File type | | CPU built-in memory | | | SD memory card | File name and extension |
|---|---|---|---|---|---|---|
| | | Program memory | Device/label memory | Data memory | | |
| | | Drive 0 | Drive 3 | Drive 4 | Drive 2 | |
| Program | | × | × | × | ○ | ANY_STRING.PRG |
| FB file | | × | × | × | ○ | ANY_STRING.PFB |
| CPU parameter | | × | × | × | ○ | CPU.PRM |
| System parameter | | × | × | × | ○ | SYSTEM.PRM |
| Module parameter | | × | × | × | ○ | UNIT.PRM |
| Module extension parameter | | × | × | × | ○ | • UEXmmmnn.PRM<br>• UEXmmm00.PPR |
| Module-specific backup parameter | | × | × | × | ○ | UBPmmmnn.BPR |
| Memory card parameter | | × | × | × | × | MEMCARD.PRM |
| Device comment | | × | × | × | ○ | ANY_STRING.DCM |
| Device initial value | | × | × | × | ○ | ANY_STRING.DID |
| Global label setting file | | × | × | × | ○ | GLBLINF.IFG |
| Initial label value file | Initial global label value file | × | × | × | ○ | GLBLINF.LID |
| | Initial local label value file | × | × | × | ○ | PROGRAM_NAME.LID |
| File register | | × | ○*1*2 | × | ○ | ANY_STRING.QDR |
| Event history | | × | × | × | × | • EVENT.LOG<br>• EVEN2.LOG |
| Device data storage file | | × | × | ○*2 | ○ | DEVSTORE.QST |
| General-purpose data | | × | × | ○*3 | ○*3 | ANY_STRING.ANY_STRING |
| Data logging setting file | Common setting file | × | × | × | × | LOGCOM.LCS |
| | Individual setting file | × | × | × | × | LOGnn.LIS |
| Memory dump setting file | | × | × | × | ○ | MEMDUMP.DPS |
| Remote password | | × | × | × | ○ | 00000001.SYP |
| Abnormal database check file | | × | × | × | ○ | ErrorDB.txt |
| System file for backing up CPU module data | | × | × | × | ○ | $BKUP_CPU_INF.BSC |
| Backup data file for backing up CPU module data | | × | × | × | ○ | BKUP_CPU.BKD |
| Device/label data file for backing up CPU module data | | × | × | × | ○ | BKUP_CPU_DEVLAB.BKD |
| System file for the backup/restoration function for the iQ Sensor Solution | | × | × | × | ○ | $BKUP_UNIT_INF.BSI |
| Backup data file for the backup/restoration function for the iQ Sensor Solution | | × | × | × | ○ | Depends on a connected device.QBR |
| ODBC server setting file | | × | × | ○ | × | netserver.cfg |
| Database path file | | × | × | × | ○ | dbmaintainpath.txt |
| Recording setting file | | × | × | × | ○ | RECCFGn.RSI |

*1 The file can be transferred only from the FTP client to the FTP server. (Only the SP.FTPPUT instruction can be executed.)

*2 When the SP.FTPPUT instruction is executed with bit 2 (Transfer completion file delete setting) of (s1)+0 set to 1 (Delete), the file cannot be transferred.

*3 The file that has more than one period cannot be transferred. The file that has one period or no period (only file name) can be transferred.

# Procedure for executing the file transfer function (FTP client)

To execute the file transfer function (FTP client), set an FTP client and an FTP server in parameters, and execute a file transfer function instruction.
Follow the procedures described below to specify the FTP server with the module parameter.

## Setting an FTP server

**1.** Set the login name, password, and home directory for the FTP client to the FTP server. In addition, give users a right to access (read/write) files. (📖 Manual for the server used)

## Setting an FTP client

**1.** Set the IP address of the CPU module (built-in Ethernet port part) to be used as an FTP client. (☞ Page 327 Own Node Settings)

[Navigation window] ⇨ [Parameter] ⇨ Target module ⇨ [Module Parameter] ⇨ [Basic Settings] ⇨ [Own Node Settings] ⇨ [IP Address]

For the Process CPU (redundant mode), set the IP addresses of system A/system B because communications are performed using the IP addresses. Set the IP address of system B by the following procedure. (☞ Page 416 Redundant System Settings)

[Navigation window] ⇨ [Parameter] ⇨ Target module ⇨ [Module Parameter] ⇨ [Application Settings] ⇨ [Redundant System Settings] ⇨ [System B IP Address]

**2.** Set "Use" for "To Use or Not to Use FTP Client Settings" in "FTP Client Settings" under "Application Settings".

For details on other setting items in "FTP Client Settings", refer to the following.

☞ Page 339 FTP Client Settings

**3.** Set "Use" for "To Use or Not to Use DNS Server Settings" in "DNS Settings" under "Application Settings" as needed.

For details on other setting items in "DNS Settings", refer to the following.

☞ Page 339 DNS Settings

**4.** Set a timer for the file transfer function (FTP client) in "Timer Settings for Data Communication" under "Application Settings" as needed. (☞ Page 401 Timer Settings for Data Communication)

| Item | Setting |
|---|---|
| Timeout period | • TCP Resend Timer<br>• TCP ULP Timer |
| Alive check interval and count | • Destination Alive Check Start Interval Timer<br>• Destination Alive Check Interval Timer<br>• Destination Alive Check Resend Count |

## Executing a file transfer function instruction

Use the dedicated instructions to execute the file transfer function (FTP client). (☞ Page 539 File transfer function instruction)

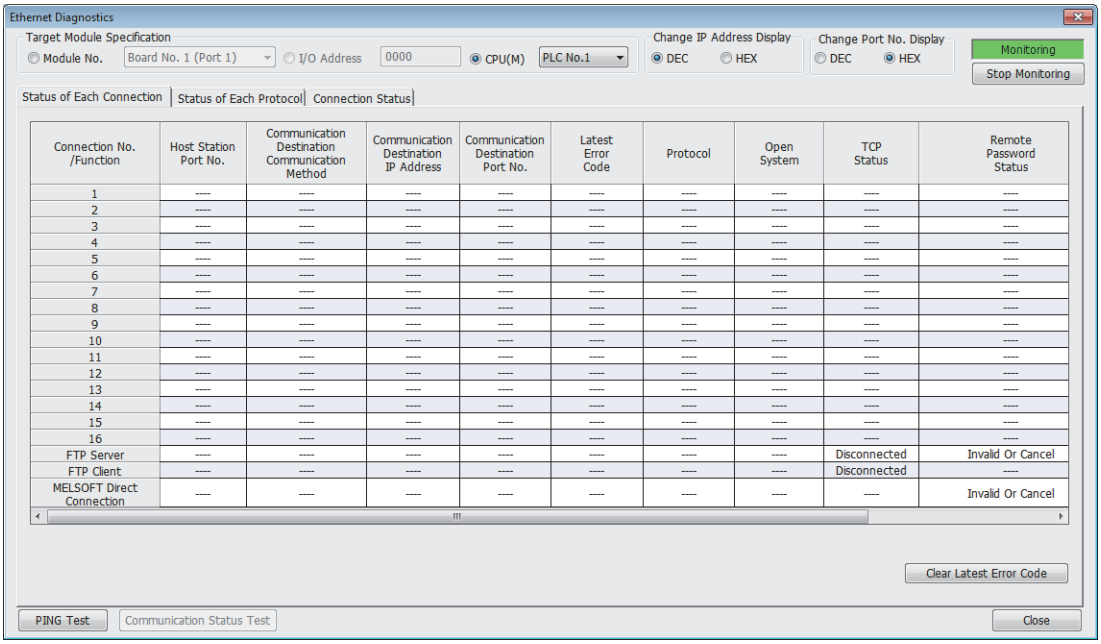
### ■Precautions

- To retrieve the file register file from the FTP server, execute the put command of the FTP server function while the CPU module is in a STOP state.
- To send/retrieve files stored in the CPU built-in memory, such as a sequence program file, to/from the FTP server, execute the CPU module data backup/restoration function together with the instruction. (📖 Manual for the CPU module used)

| File transfer function | Procedure |
|---|---|
| Sending files to the FTP server | (1) Back up transfer target files stored in memory areas to the SD memory card.<br>(2) Execute the SP.FTPPUT instruction. |
| Retrieving files from the FTP server | (1) Execute the SP.FTPGET instruction.<br>(2) Restore the retrieved files from the SD memory card to memory areas. |

## Checking the file transfer function (FTP client) status

The status of the function can be checked with "Status of Each Connection" on the "Ethernet Diagnostics" window. (☞ Page 428 Status of Each Connection)



# Precautions

## Connection with the FTP server

If a response to an alive check request is not returned from the FTP server, the CPU module (FTP client) determines that the FTP server is not alive and disconnects the connection.
The connection may also be disconnected if the FTP server does not support the TCP KeepAlive function (response to a KeepAlive ACK message).

## Communication error when retrieving transfer files

If a large capacity file is retrieved by execution of the SP.FTPGET instruction, a communication error may occur due to an increase of the transfer processing load. When transferring a large capacity file, extend a connection time (timeout period) in parameters for the FTP server so that no communication error occurs.

## Execution of the SD memory card forced disable function

If the SD memory card forced disable function is executed, use of the SD memory card is disabled in the middle of the transfer processing, causing an immediate stop of the processing. Check that the transfer processing has been completed, and execute the function.

## Execution of a file transfer function instruction during execution of another function

A file transfer function instruction cannot be executed during execution of the following functions. Check that the following function is not being executed, and then execute the instruction.

- CPU module data backup/restoration function
- iQ Sensor Solution data backup/restoration function

## File operation from the external device during file transfer

Do not manipulate the files from an external device, such as an engineering tool, while transferring the files.
If the file is manipulated from another device while transferring the files, the external device may issue an error. If the processing has been halted due to an error, re-execute the processing after the file transfer operation.

## Communication processing

If another communication function (MELSOFT connection or SLMP) is simultaneously executed with UDP/IP during file transfer with FTP, an error such as timeout may occur. Either communicate after the file is transferred, or communicate with TCP/IP.

## Deleting unnecessary files

If a cable is disconnected, system is powered off, or the CPU module is reset during the file transfer, delete unnecessary files (such as a temporary file and undefined files) in the FTP server. Then, transfer files again.

## When the process CPU (redundant mode) is used

### ■System switching during the file transfer

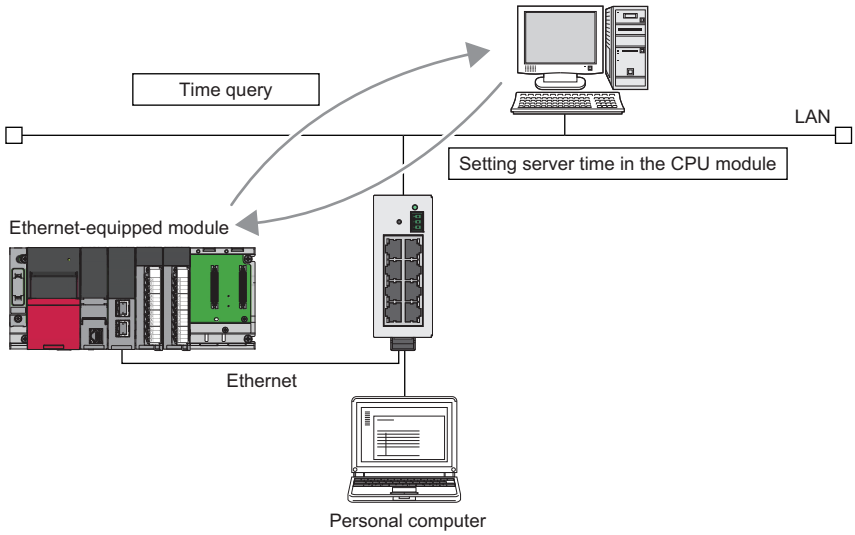
- When the system switching occurs during the file transfer by an engineering tool, the system switching instructions, or the system switching request from the network module, the system will be switched after completed the processing.
- When the system switching occurs because of power-off, reset, hardware failure, or stop error during the file transfer, the connections between the FTP server and old control system cannot be disconnected properly. In this case, communications may not be performed with the new control system until disconnection of the old control system is recognized due to FTP server timeout. To reconnect with the FTP server, keep an attempt to execute until the connection with the FTP server becomes open. In addition, delete unnecessary files on the FTP server when the files are generated. The files are not remain in the FTP server since the FTP connection are not connected even when the system switching is executed other than during the file transfer.

### ■SP.FTPPUT instruction immediately after system switching

When the SP.FTPPUT instruction is executed immediately after the system switching, execute the instruction with transfer files existing (such as generated files) in the new control system.

# 1.11 Time Setting Function (SNTP Client)

Time information is collected from the time information server (SNTP server) connected on the LAN at the specified timing, and the CPU module's time is automatically set.




**Restriction**

This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".

**Point**

An SNTP server (time information server) must be provided on the LAN line to use this function.

## Time setting execution timing

Time setting is executed in the following timing.

- At powering off and on resetting the CPU module
- At each set time (execution at set cycle interval)
- At set time (execution at set time)
- At programmed arbitrary timing*1

*1 The time is set when 'Time setting function (SNTP client) execution' (Un\G11009.0 for the RJ71EN71 and RnENCPU (network part), Un\G299.0 for the CPU module (built-in Ethernet port part)) is turned on.

**Point**

When setting the time at the CPU module power on or reset, check the hub or external device connection before setting.

## Setting procedure

Set with "Time Setting (SNTP client)" under "Application Settings". (☞ Page 400 Time Setting)

## Confirming the execution results

The time setting execution results can be checked with the following module label.

### ■For the RJ71EN71 and RnENCPU (network part)

- 'Operation result' (Un\G11000)
- 'Execution time' (Un\G11001 to Un\G11007)
- 'Required response time' (Un\G11008)

### ■For the CPU module (built-in Ethernet port part)

- 'Operation result' (Un\G290)
- 'Execution time' (Un\G291 to Un\G297)
- 'Required response time' (Un\G298)

## Precautions

### ■Time to be set

The time to be set in the CPU module is the time which is obtained by adjusting the time information (UTC) retrieved from the SNTP server (time information server) based on the time zone set in "Clock Related Setting" under "Operation Related Setting" of "CPU Parameter".

### ■Communication timeout

If a response is not received from the SNTP server (time information sever) 20 seconds after the time setting is executed, the communication times out. An error does not occur when the communication times out. Instead, the timeout occurrence appears in the event history.

### ■Delay by communication time

The time set in the CPU module may be delayed by the time required to communicate with the SNTP server's personal computer. To reduce this delay, specify an SNTP server personal computer on the network that is as close to the Ethernet-equipped module as possible.

### ■When using multi-CPU system configuration

Execute the time setting only for the CPU No.1. Even if the time is set for a CPU module other than the CPU No.1, the clock data for the CPU No.1 is set automatically.

### ■Setting the execution time

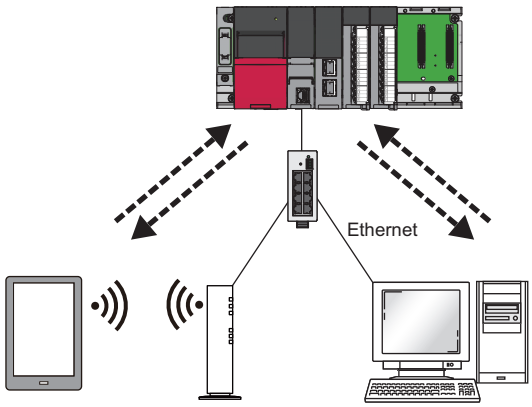
The execution time can be set in the range of 1980 to 2079.

### ■Daylight saving time function

This function is supported by the RJ71EN71 Ver.68 or later.

# 1.12 Web Server Function

The Web server function can read/write device data from/to the CPU module by using a Web browser. The device data can be read/written from/to the CPU module by using a general-purpose Web browser of an Ethernet-connected personal computer and also by using a general-purpose Web browser of a tablet or smartphone via Ethernet. An original Web page can be created by users and displayed as well.




For the specifications, parameter settings, procedures before operation, system configuration, functions, and troubleshooting of the Web server function, refer to the following.

MELSEC iQ-R/MELSEC iQ-F Web Server Function Guide Book

This manual describes the JavaScript objects and CGI objects that can be used in the user Web page.

**Point**

Incorporate measures other than this function if the programmable controller system's safety must be maintained against illegal access from an external device. Mitsubishi shall not be held liable for any system problems that may occur from illegal access. The user authentication of the Web server function is one method of preventing illegal access (such as program or data destruction) from an external device. It does not completely prevent illegal access.

Examples of measures for illegal access are given below.

- Install a firewall
- Install a personal computer as a relay station, and control the relay of send/receive data with an application program
- Install an external device for which the access rights can be controlled as a relay station (Contact the network provider or equipment dealer for details on the external devices for which access rights can be controlled.)

# JavaScript objects



JavaScript objects in the library of user Web pages enable device data to be read or written easily on the user Web page. The following table lists the objects in the JavaScript object library for the user Web page (RUserWebLib.js).

| Object name (function name) | Description | Reference |
|---|---|---|
| Data block (WSDatblk) | Displays the specified device data in a table format. | Page 150 Data block (WSDatblk) |
| Level display (WSLevel) | Displays what percentage the device value accounts for of the whole (the range between the upper and lower limit values). | Page 152 Level display (WSLevel) |
| Figure display (WSFigure) | Displays the specified figure when the device value reaches the set value. | Page 154 Figure display (WSFigure) |
| Image display (WSPicture) | Displays the specified image file when the device value reaches the set value. | Page 156 Image display (WSPicture) |
| Historical graph (WSHstgrp) | Displays the device data in a time-series line graph. | Page 157 Historical graph display (WSHstgrp) |
| Write button (WSWtrBtn) | Writes the specified value to the specified device. | Page 159 Write button (WSWtrBtn) |
| Logoff button (WSLogoutBtn) | Performs the logoff operation. | Page 160 Logoff button (WSLogoutBtn) |

## Common information on JavaScript objects

### ■Style sheet

Descriptions for the styles such as character size, color, line color, the background color of the user Web page can be omitted. If descriptions are omitted or incorrect, the following default display according to the style sheet (UserWebStyle.css) is applied.

| Item | Default |
|---|---|
| Character color | Black |
| Background color | White |
| Line color | Black |
| Fill color of level display | Blue |
| Graph line color | Blue |
| Character size | 20 |
| Button character color | According to the setting of the Web browser |
| Button background color | According to the setting of the Web browser |
| Button line color | According to the setting of the Web browser |

The character font differs depending on the terminal used to display the user Web page.

**Restriction**

When editing the style sheet in the library of the user Web page, do not change the class names in the style sheet. In addition, do not define the style classes with the same name.

## ■Parameter settings of JavaScript objects

- When an optional parameter setting is omitted, the object will be displayed as default. Also, when the optional parameter setting is abnormal, the object will be displayed as default.
- Set parameters using the specified data formats. A parameter error occurs if a parameter is set in the data format other than the specified one (for example, when a character string such as '1' is set in the parameter that should be a numeric value).
- If the X- and Y- coordinates are omitted, the object will be placed at the upper left end (coordinates 0, 0).
- For an RGB value specified for parameters or HTML specifications such as a color name, the range check is not performed. The operation for an abnormal setting differs depending on the browser.
- When specifying a device name which includes \, such as Un\G and Jn\X, repeat \ as Un\\G and Jn\\X. (\ is an escape sequence.)
- The following table lists formats and positional notations available for the devices which can be set for JavaScript objects.

○: Can be set, ×: Cannot be set

| Classification | Type | Device name | 16-bit signed<br>16-bit unsigned | 32-bit signed<br>32-bit unsigned | Single-precision real number<br>Double-precision real number | Bit |
|---|---|---|---|---|---|---|
| User device | Bit | Input (X) | × | × | × | ○ |
| | | Output (Y) | × | × | × | ○ |
| | | Internal relay (M) | × | × | × | ○ |
| | | Latch relay (L) | × | × | × | ○ |
| | | Link relay (B) | × | × | × | ○ |
| | | Annunciator (F) | × | × | × | ○ |
| | | Link special relay (SB) | × | × | × | ○ |
| | | Edge relay (V) | × | × | × | ○ |
| | | Step relay (S) | × | × | × | ○ |
| | Bit/word | Timer (T) | ○: Unsigned decimal/ hexadecimal | × | × | × |
| | | Retentive timer (ST) | ○: Unsigned decimal/ hexadecimal | × | × | × |
| | Bit/double word | Long timer (LT) | × | ○: Unsigned decimal/ hexadecimal | × | × |
| | | Long retentive timer (LST) | × | ○: Unsigned decimal/ hexadecimal | × | × |
| | Bit/word | Counter (C) | ○: Unsigned decimal/ hexadecimal | × | × | × |
| | Bit/double word | Long counter (LC) | × | ○: Unsigned decimal/ hexadecimal | × | × |
| | Word | Data register (D) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| | | Link register (W) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| | | Link special register (SW) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| | Bit | Direct access input (DX) | × | × | × | ○ |
| | | Direct access output (DY) | × | × | × | ○ |
| System device | Bit | Special relay (SM) | × | × | × | ○ |
| | Word | Special register (SD) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| Link direct device | Bit | Link input (Jn\X) | × | × | × | ○ |
| | Bit | Link output (Jn\Y) | × | × | × | ○ |
| | Bit | Link relay (Jn\B) | × | × | × | ○ |
| | Bit | Link special relay (Jn\SB) | × | × | × | ○ |
| | Word | Link register (Jn\W) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| | Word | Link special register (Jn\SW) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |

| Classification | Type | Device name | 16-bit signed<br>16-bit unsigned | 32-bit signed<br>32-bit unsigned | Single-precision real number<br>Double-precision real number | Bit |
|---|---|---|---|---|---|---|
| Module access device | Word | Module access device (Un\G) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| CPU buffer memory access device | Word | CPU buffer memory access device (U3En\G, U3En\HG) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| Index register | Word | Index register (Z) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| | Double word | Long index register (LZ) | × | ○: Decimal/hexadecimal | ○: Decimal for single-precision real number | × |
| File register | Word | File register (R) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| | | File register (ZR) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| Refresh data register | Word | Refresh data register (RD) | ○: Decimal/hexadecimal | ○: Decimal/hexadecimal | ○: Decimal | × |
| Other devices | — | SFC block device (BL) | × | × | × | ○ |
| | — | SFC step device (BLn\S) | × | × | × | ○ |

## ■Refreshing cycle

Set the refreshing cycle of the user Web page to the variable updateInterval in HTML. This refreshing cycle applies to all the objects in the user Web page. When the variable updateInterval is not described in HTML or it is out of range between 1 and 120 *1, 5-second interval (default) applies to the operation.

*1 The setting range depends on the firmware version of the CPU module. For details, refer to the following.
MELSEC iQ-R/MELSEC iQ-F Web Server Function Guide Book

Ex.

When refreshing the user Web page in 10-second interval

```
<script>
  var updateInterval = 10;
  </script>
```

## ■Message display language

Set the message display language to the variable dspLanguage in HTML. The following table lists the settings of the variable dspLanguage and message display languages.

| dspLanguage | Message display language |
|---|---|
| No description | English |
| ja-JP | Japanese |
| en-US | English |
| zh-CN | Chinese (Simplified) |
| Other than the above (out of range) | English |

## ■Available files

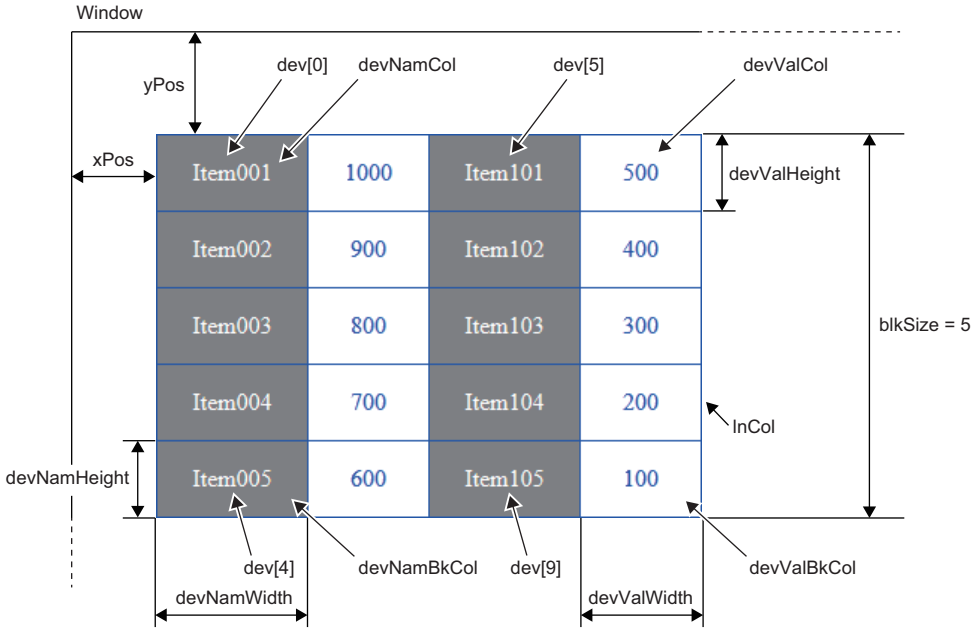
The following files are available in the user Web page.

| File | Extension | MIME type |
|---|---|---|
| HTML | .html | text/html |
| | .htm | text/htm |
| JavaScript | .js | text/javascript |
| CSS | .css | text/css |
| GIF image | .gif | image/gif |
| PNG image | .png | image/png |
| JPG/JPEG image | .jpg(.jpeg) | image/jpeg |

## Data block (WSDatblk)

This object displays the specified device data in a table format. The current values of the devices are displayed as center aligned and updated in the fixed interval.

• direction = 0 (Vertical)




• direction = 1 (Horizontal)




Set dev[n] for the devices to be displayed. (n: 0 to 31)

○: Required, —: Optional

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| dev[0].dsp | Device display name 1 | A name to be displayed with the data block | ○*1 |
| dev[0].name | Device name 1 | Device type + device number | ○ |
| dev[0].base | The positional notation of the device 1 | K: Decimal<br>H: Hexadecimal<br>B: Binary | ○ |
| dev[0].format | The data format of the device 1 | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number<br>6: Bit | ○ |
| ⋮ | | | |
| dev[31].dsp | Device display name 32 | A name to be displayed with the data block | — |
| dev[31].name | Device name 32 | Device type + device number | — |
| dev[31].base | The positional notation of the device 32 | K: Decimal<br>H: Hexadecimal<br>B: Binary | — |

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| dev[31].format | The data format of the device 32 | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number<br>6: Bit | — |
| direction | Display direction | 0: Vertical<br>1: Horizontal | —<br>(If omitted, it is 0.) |
| blkSize | Block size | 1 to 32<br>(The number of lines is displayed when the display direction is set to vertical, and the number of columns is displayed when the display direction is set to horizontal.) | ○ |
| devNamDisp | Device name display | 0: Not display<br>1: Display | —<br>(If omitted, it is 1.) |
| devNamCol | Character color of device name | The RGB value or color name<br>Example: #FF0000 (RGB value) or red (color name) | — |
| devNamBkCol | Background color of device name | | |
| devNamWidth | Cell width of device name | Positive real numbers (unit: px)<br>(For the horizontal display direction, the setting value of the cell width of device value takes priority, and for the vertical display direction, the setting value of the cell height of device value does.) | ○*1 |
| devNamHeight | Cell height of device name | | |
| devValCol | Character color of device value | The RGB value or color name | — |
| devValBkCol | Background color of device value | | |
| devValWidth | Cell width of device value | Positive real numbers (unit: px) | ○ |
| devValHeight | Cell height of device value | | |
| lnCol | Line color | The RGB value or color name | — |
| xPos | X-coordinate | The horizontal coordinate of the data block at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the data block at the upper left end (unit: px) | — |

*1 When the device name is not displayed (devNamDisp = 0), no error occurs without this setting.

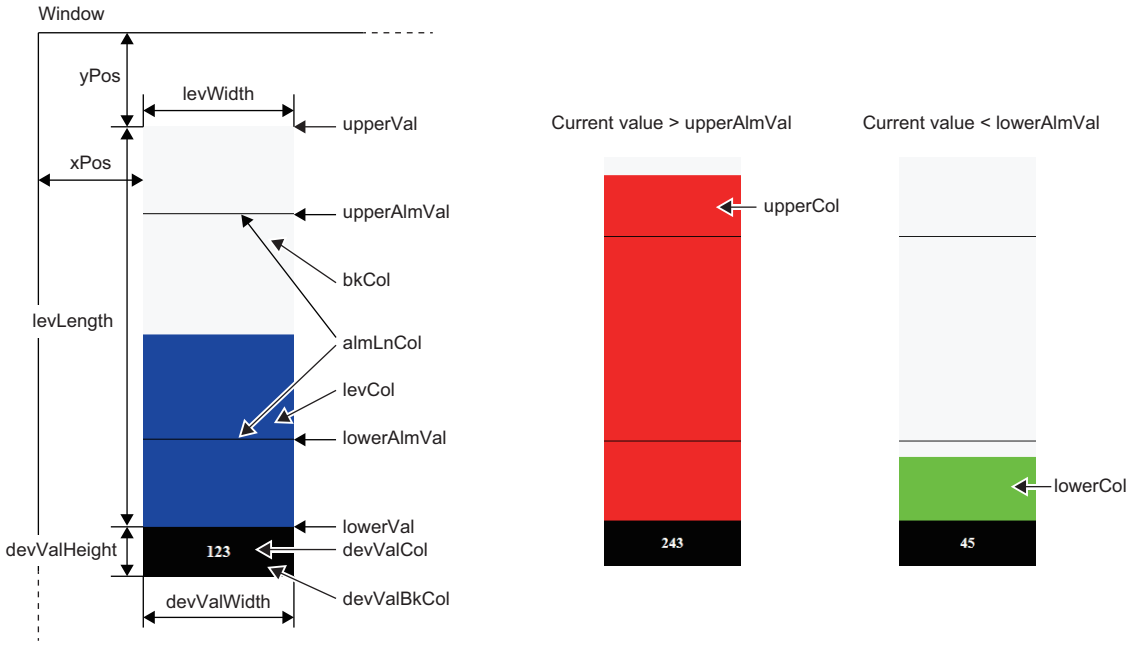
## ■Precautions

- Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.
- If the data format is the single-precision real number or double-precision real number, the positional notation will be the decimal.

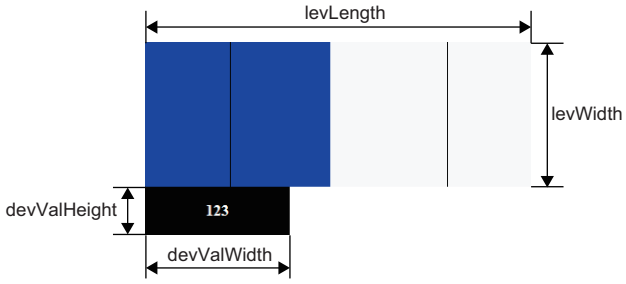
## Level display (WSLevel)

This object displays the current value of the specified device as a level for the range between the upper and lower limit values. The displayed color can be changed when the current value is above or below the specified value. The current values of the devices are updated in the fixed interval.

• direction = 0 (Vertical)




• direction = 1 (Horizontal)




○: Required, —: Optional

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| devName | Device name | Device type + device number | ○ |
| direction | Level direction | 0: Vertical<br>1: Horizontal | —<br>(If omitted, it is 0.) |
| upperVal | Upper limit value | A numeric value (decimal) within the range of the specified data format. | ○ |
| lowerVal | Lower limit value | | |
| upperAlmVal | Upper limit of alarm value | | |
| lowerAlmVal | Lower limit of alarm value | | |
| dspAlmLn | Display of alarm value line | 0: Not display<br>1: Display | —<br>(If omitted, it is 1.) |
| dspVal | Display of current value | | |
| valFormat | Data format | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number | —<br>(If omitted, it is 0.) |
| levCol | Level display color | The RGB value or color name | — |

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| upperCol | Display color when the current value is above the upper limit of alarm value. | The RGB value or color name | —<br>(If omitted, it is the same as the level display color.) |
| lowerCol | Display color when the current value is below the lower limit of alarm value. | | |
| bkCol | Background color | The RGB value or color name | — |
| almLnCol | Line color of alarm value | | — |
| levLength | Level length | Positive real numbers (unit: px) | ○ |
| levWidth | Level width | | |
| xPos | X-coordinate | The horizontal coordinate of the level display at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the level display at the upper left end (unit: px) | — |
| devValWidth | Cell width of current value | Positive real numbers (unit: px) | ○ |
| devValHeight | Cell height of current value | | |
| devValCol | Character color of current value | The RGB value or color name | — |
| devValBkCol | Background color of current value | | |

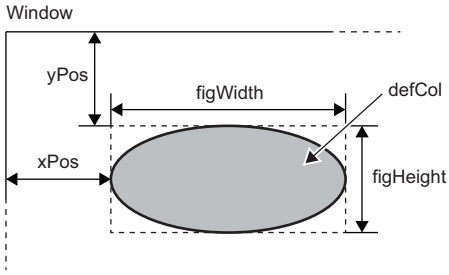
## ■Precautions

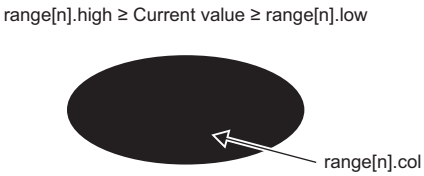
Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.

## Figure display (WSFigure)

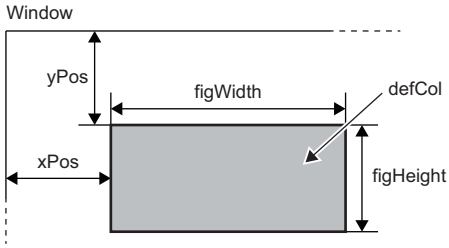
When the device value is within the specified range, a figure is displayed in the specified color. The device value is monitored in the fixed interval.

• figType = 'Oval' (oval)



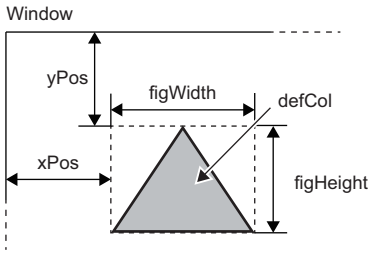




• figType = 'Rect' (rectangle)




• figType = 'Tri' (triangle)

When a negative value is set for the figure height (figHeight), the downward triangle is displayed.




Set range[n] for the range of the display color to be displayed. (n: 0 to 4)

○: Required, —: Optional

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| devName | Device name | Device type + device number | ○ |
| devFormat | The data format of the device | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number<br>6: Bit | ○ |
| figType | Figure type | Oval: Oval (When specifying a circle, set the figure height = width.)<br>Rect: Rectangle (When specifying a square, set the figure height = width.)<br>Tri: Triangle | ○ |
| figHeight | Figure height | Real numbers (unit: px)<br>When a negative value is set, the value is treated as an absolute value. | ○ |
| figWidth | Figure width | | |
| defCol | Default display color | The RGB value or color name<br>(This item is displayed when the device value is out of the setting range.) | ○ |
| rangeNum | Number of setting ranges | 1 to 5 | ○ |
| range[0].low | Lower limit of setting range 1 | A numeric value (decimal) within the range of the specified data format. | ○ |
| range[0].high | Upper limit of setting range 1 | | |

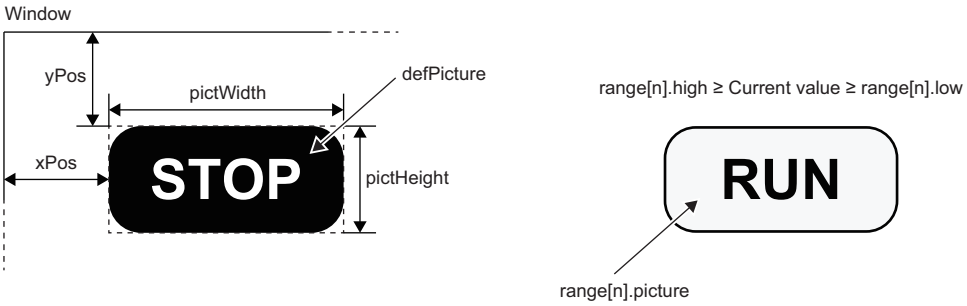
| Parameter | Item | Description | Requirement |
|---|---|---|---|
| range[0].col | Display color when device value is within setting range 1 | The RGB value or color name | ○ |
| range[1].low | Lower limit of setting range 2 | A numeric value (decimal) within the range of the specified data format. | — |
| range[1].high | Upper limit of setting range 2 | | |
| range[1].col | Display color when device value is within setting range 2 | The RGB value or color name | — |
| ⋮ | | | |
| range[4].low | Lower limit of setting range 5 | A numeric value (decimal) within the range of the specified data format. | — |
| range[4].high | Upper limit of setting range 5 | | |
| range[4].col | Display color when device value is within setting range 5 | The RGB value or color name | — |
| xPos | X-coordinate | The horizontal coordinate of the figure display at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the figure display at the upper left end (unit: px) | |

## ■Precautions

- Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.
- If the setting ranges overlap, the figure with the lower number of the setting range is displayed. For example, when the device value is in both of the setting range 1 and 2, the figure of the setting range 1 is displayed.

## Image display (WSPicture)

When the device value is within the specified range, the specified image file can be displayed. The device value is monitored in the fixed interval.




Set range[n] for the range in which the image is to be displayed. (n: 0 to 4)

○: Required, —: Optional

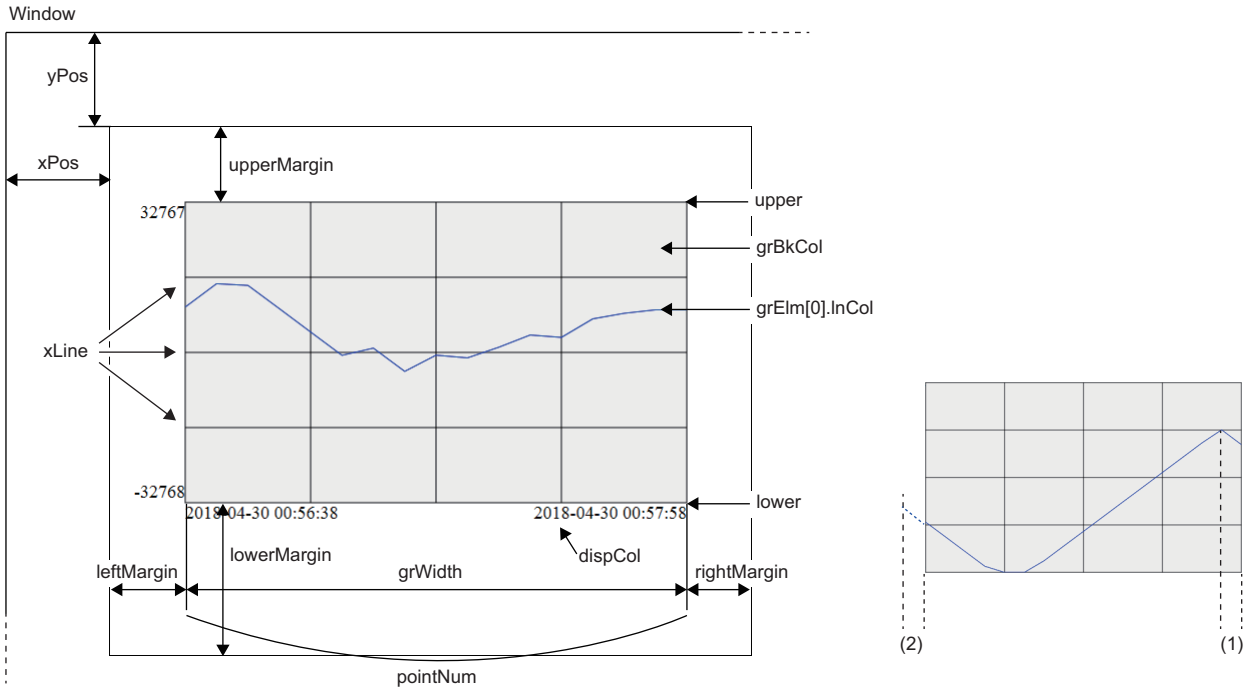
| Parameter | Item | Description | Requirement |
|---|---|---|---|
| devName | Device name | Device type + device number | ○ |
| devFormat | The data format of the device | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number<br>6: Bit | ○ |
| pictHeight | Display range height of image file | Real numbers (unit: px)<br>When a negative value is set, the value is treated as an absolute value. | ○ |
| pictWidth | Display range width of image file | | |
| defPicture | Name of image file to be displayed as default | Extension: .jpg, .jpeg, .gif, .png<br>(This item is displayed when the device value is out of the setting range.) | ○ |
| rangeNum | Number of setting ranges | 1 to 5 | ○ |
| range[0].low | Lower limit of setting range 1 | A numeric value (decimal) within the range of the specified data format. | ○ |
| range[0].high | Upper limit of setting range 1 | | |
| range[0].picture | Name of image file displayed when device value is within setting range 1 | Extension: .jpg, .jpeg, .gif, .png | ○ |
| range[1].low | Lower limit of setting range 2 | A numeric value (decimal) within the range of the specified data format. | — |
| range[1].high | Upper limit of setting range 2 | | |
| range[1].picture | Name of image file displayed when device value is within setting range 2 | Extension: .jpg, .jpeg, .gif, .png | — |
| ⋮ | | | |
| range[4].low | Lower limit of setting range 5 | A numeric value (decimal) within the range of the specified data format. | — |
| range[4].high | Upper limit of setting range 5 | | |
| range[4].picture | Name of image file displayed when device value is within setting range 5 | Extension: .jpg, .jpeg, .gif, .png | — |
| xPos | X-coordinate | The horizontal coordinate of the figure display range at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the figure display range at the upper left end (unit: px) | — |

### ■Precautions

- Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.
- The total size of the image file on one screen should be 100K bytes or smaller.
- If the setting ranges overlap, the image with the lower number of the setting range is displayed. For example, when the device value is in both of the setting range 1 and 2, the figure of the setting range 1 is displayed.

## Historical graph display (WSHstgrp)

This object displays the device value in a time-series line graph. The device value is read in every refreshing cycle. After the number of records is reached, the display shifts to left and old records are deleted.




(1) Refreshing cycle
(2) Data in a refreshing cycle is out of the display range.

Set grElm[n] for the devices to be displayed. (n: 0 to 31)

○: Required, —: Optional

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| grElmNum | Number of graph elements | 1 to 32 | ○ |
| devFormat | The data format of the device | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number<br>6: Bit | ○ |
| grElm[0].devName | Device name | Device type + device number | ○ |
| grElm[0].lnCol | Graph line color | The RGB value or color name<br>Example: #FF0000 (RGB value) or red (color name) | — |
| grElm[1].devName | Device name | Device type + device number | — |
| grElm[1].lnCol | Graph line color | The RGB value or color name | — |
| ⋮ | | | |
| grElm[31].devName | Device name | Device type + device number | — |
| grElm[31].lnCol | Graph line color | The RGB value or color name | — |
| grBkCol | Graph background color | The RGB value or color name | — |
| dspCol | Character color | | |
| pointNum | Number of records | 5 to 100 | ○ |
| upper | Upper limit value | A numeric value (decimal) within the range of the specified data format. | ○ |
| lower | Lower limit value | | |
| yLine | Vertical axis interval (record) | 0 to 99 | ○ |
| xLine | Number of horizontal axes | | |
| grHeight | Graph height | Positive real numbers (unit: px) | ○ |
| grWidth | Graph width | | |

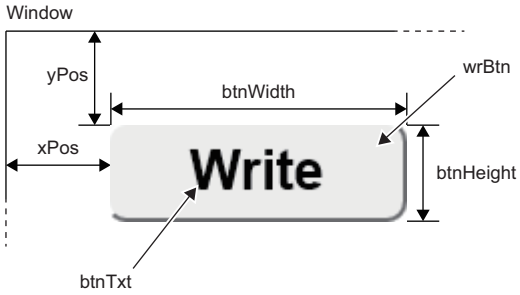
| Parameter | Item | Description | Requirement |
|---|---|---|---|
| xPos | X-coordinate | The horizontal coordinate of the historical graph display at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the historical graph display at the upper left end (unit: px) | |
| rightMargin | Right margin | Positive real numbers (unit: px) | — (If omitted, it is 0.) |
| leftMargin | Left margin | | |
| upperMargin | Upper margin | | |
| lowerMargin | Lower margin | | |

## ■Precautions

- Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.
- The numerical value used in the historical graph is fixed to decimal.
- If the communication load is high for a while, the device value may be lost.

## Write button (WSWtrBtn)

This object writes the specified value to the specified device.




○: Required, —: Optional

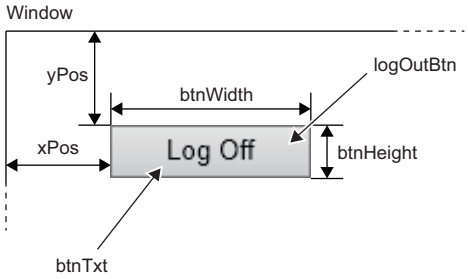
| Parameter | Item | Description | Requirement |
|---|---|---|---|
| devName | Device name | Device type + device number | ○ |
| devBase | Positional notation of device | K: Decimal<br>H: Hexadecimal<br>B: Binary | ○ |
| devFormat | The data format of the device | 0: 16-bit signed<br>1: 16-bit unsigned<br>2: 32-bit signed<br>3: 32-bit unsigned<br>4: Single-precision real number<br>5: Double-precision real number<br>6: Bit | ○ |
| wrVal | Write value | The setting range depends on the positional notation and data format of the device.<br>Specify a string for the input value.<br>Example: wrVal: '1' | ○ |
| wrBtn | Write button | Class element name<br>It is used for the selector of the style sheet. | — |
| btnTxt | Text displayed on the button | Optional | —<br>(If omitted, it is empty.) |
| btnWidth | Button width | Positive real numbers (unit: px) | ○ |
| btnHeight | Button height | | |
| wrComfirm | Write confirmation message | 0: Not display<br>1: Display | —<br>(If omitted, it is 0.) |
| language | Message language | 0: Japanese<br>1: English<br>2: Chinese (Simplified) | —<br>(If omitted, it is 1.) |
| xPos | X-coordinate | The horizontal coordinate of the write button at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the write button at the upper left end (unit: px) | — |

### ■Precautions

- Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.
- If no parameter is specified for the write button, the default style specified in UserWebStyle.css is applied.

## Logoff button (WSLogoutBtn)

This object places the button to log off. When the logoff button is clicked, the user login page (Log-in_User.html) is displayed.




○: Required, —: Optional

| Parameter | Item | Description | Requirement |
|---|---|---|---|
| logOutBtn | Logoff button | Class element name of button object<br>(It is used for the selector of the style sheet.) | — |
| btnTxt | Text displayed on the button | Optional | —<br>(If omitted, it is empty.) |
| btnWidth | Button width | Positive real numbers (unit: px) | ○ |
| btnHeight | Button height | | |
| xPos | X-coordinate | The horizontal coordinate of the logoff button at the upper left end (unit: px) | — |
| yPos | Y-coordinate | The vertical coordinate of the logoff button at the upper left end (unit: px) | — |

### ■Precautions

- Write all the necessary parameters. If the necessary parameters are not written or the setting value is out of range, an error occurs.
- When using the logoff button, always store the user login page (Log-in_User.html) in the SD memory card.
- If no parameter is specified for the logoff button, the default style specified in UserWebStyle.css is applied.

# CGI object



By using the CGI object, a simple and small file size user Web page can be created.
For an overview of CGI, refer to reference books on the market.
For the user Web page, the following CGI objects dedicated to read/write data from/to devices can be used.

| File name | Item | Description | Reference |
|---|---|---|---|
| RdDevRnd.cgi | Device read CGI | Reads the current value of the specified device. | Page 163 Device read CGI |
| WrDevRnd.cgi | Device write CGI | Writes the specified value to the specified device. | Page 169 Device write CGI |

The data acquisition from the Web server or the operation to the Web server is executed by passing it as the CGI request from the client and returning the execution result as a response to the client.

## Data specified for CGI object

The data used in the CGI object are listed below.

### ■Device name

The following table lists the devices that can be accessed with the device read CGI and device write CGI.

| Type | Device |
|---|---|
| Bit device | X, Y, M, L, B, F, SB, V, DX, DY, SM, Jn\X, Jn\Y, Jn\B, Jn\SB, BL, BLn\S |
| Word device | T (Current value), ST (Current value), C (Current value), D, W, SW, SD, Jn\W, Jn\SW, Un\G, U3En\G, U3En\HG, Z, R, ZR, RD |
| Double-word device | LT (Current value), LST (Current value), LC (Current value), LZ |

When specifying a device name which includes \, such as Un\G and Jn\X, repeat \ as Un\\G and Jn\\X. (\ is an escape sequence.)

## ■Device size

Applicable device size differs depending on the device.

For the notation of the device number, specify decimal or hexadecimal depending on the device.

○: Available in both device read CGI and device write CGI, ×: Not available

| Device | Notation | Device size | | | |
|---|---|---|---|---|---|
| | | B: Bit | W: Word | D: Double word | Q: Quad word |
| X | Hexadecimal | ○ | × | × | × |
| Y | Hexadecimal | ○ | × | × | × |
| M | Decimal | ○ | × | × | × |
| L | Decimal | ○ | × | × | × |
| B | Hexadecimal | ○ | × | × | × |
| F | Decimal | ○ | × | × | × |
| SB | Hexadecimal | ○ | × | × | × |
| V | Decimal | ○ | × | × | × |
| DX | Hexadecimal | ○ | × | × | × |
| DY | Hexadecimal | ○ | × | × | × |
| SM | Decimal | ○ | × | × | × |
| Jn\X | Hexadecimal | ○ | × | × | × |
| Jn\Y | Hexadecimal | ○ | × | × | × |
| Jn\B | Hexadecimal | ○ | × | × | × |
| Jn\SB | Hexadecimal | ○ | × | × | × |
| BL | Decimal | ○ | × | × | × |
| BLn\S | Decimal | ○ | × | × | × |
| T (current value) | Decimal | × | ○ | ○ | ○ |
| ST (current value) | Decimal | × | ○ | ○ | ○ |
| C (current value) | Decimal | × | ○ | ○ | ○ |
| D | Decimal | × | ○ | ○ | ○ |
| W | Hexadecimal | × | ○ | ○ | ○ |
| SW | Hexadecimal | × | ○ | ○ | ○ |
| SD | Decimal | × | ○ | ○ | ○ |
| Jn\W | Hexadecimal | × | ○ | ○ | ○ |
| Jn\SW | Hexadecimal | × | ○ | ○ | ○ |
| Un\G | Decimal | × | ○ | ○ | ○ |
| U3En\G | Decimal | × | ○ | ○ | ○ |
| U3En\HG | Decimal | × | ○ | ○ | ○ |
| Z | Decimal | × | ○ | ○ | ○ |
| R | Decimal | × | ○ | ○ | ○ |
| ZR | Decimal | × | ○ | ○ | ○ |
| RD | Decimal | × | ○ | ○ | ○ |
| LT (current value) | Decimal | × | × | ○ | ○ |
| LST (current value) | Decimal | × | × | ○ | ○ |
| LC (current value) | Decimal | × | × | ○ | ○ |
| LZ | Decimal | × | × | ○ | Device read CGI only |

## ■Device value

The notation of the device value used in the CGI object should be as follows:

- The CGI object treats device values are treated as a string type in the hexadecimal notation. When using the decimal number or real number in the Web page, convert hexadecimal ↔ decimal or real number by using JavaScript.
- Do not start with 0x. Do not perform the 0 interpolation. For example, when reading/writing the value 0x012F, set the string 12F as the device value.
- The case of the hexadecimal alphanumeric character is ignored. For example, the device value can be expressed either as 12F or 12f to read/write the value 0x012F.

## Device read CGI

Reads the current value of the specified device.

### ■Access method and access information

| Item | Description |
|---|---|
| Access method | POST |
| Access destination information (URL) | /cgi/RdDevRnd.cgi |

### ■Request specifications

The following table lists the parameters used for the request.

| Parameter name | Data type | Description | Setting range |
|---|---|---|---|
| NUM | string | Hexadecimal string of the number of read device points (n: 1 to 20) | Set this parameter so that the total number of device points specified to read/write in one Web page is within 32 points. |
| DEV1 | string | Device name of the first point | Up to 16 alphanumeric characters<br>(This parameter is not case-sensitive. The indirect specification, bit specification, digit specification, or index modification cannot be performed.)<br>☞ Page 161 Device name |
| TYP1 | string | Device size of the first point | B: Bit<br>W: Word<br>D: Double word<br>Q: Quad word<br>☞ Page 162 Device size |
| ⋮ | | | |
| DEV(n) | string | Device name of the nth point (n: 1 to 32) | Up to 16 alphanumeric characters<br>(This parameter is not case-sensitive. The indirect specification, bit specification, digit specification, or index modification cannot be performed.) |
| TYP(n) | string | Device size of the nth point | B: Bit<br>W: Word<br>D: Double word<br>Q: Quad word |

The format of the request data is the query string. Specify DEV(n) and TYP(n) with the serial number. Otherwise, an error occurs.

**Ex.**

When reading 10 points of device from D0, M100, ···, SD0

NUM=A&DEV1=D0&TYP1=D&DEV2=M100& ··· &DEV10=SD0&TYP10=W

## ■Response specifications

The following table lists the parameters used for the response.

| Parameter name | Data type | Description |
|---|---|---|
| RET | string | Execution result (hexadecimal string)<br>0000: Normal<br>0001: Not logged on<br>0005: Incorrect request source (Referer)<br>4***: Error code (Manual for the CPU module used) |
| DATA | string | Reading value (array)<br>Hexadecimal string |

The format of the response data is JSON.

Ex.

Response data of the device read CGI

```
{
   "RET" : "0",
   "DATA" : [
      "100",
         :
      "FABC"
   ]
}
```

The above response is transferred in the following format on a message.

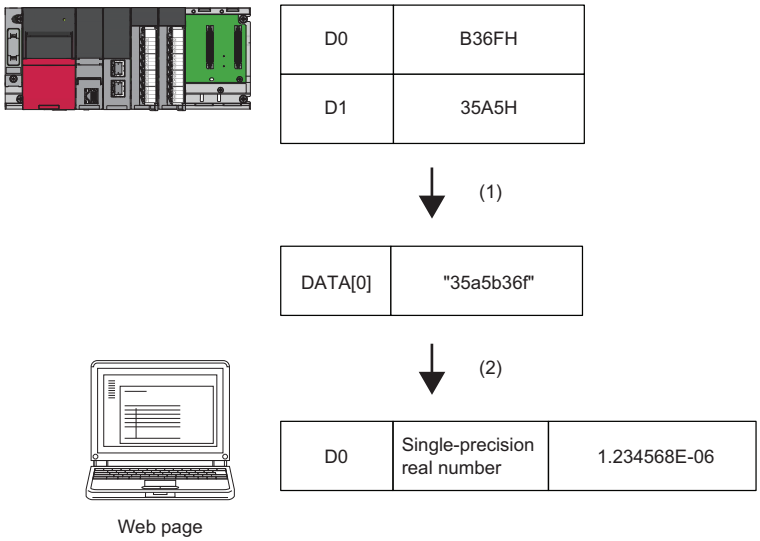
{"RET":"0","DATA":["100", ··· ,"FABC"]}

When the execution result is abnormal, only RET is transferred.

{"RET":"4031"}

**Point**

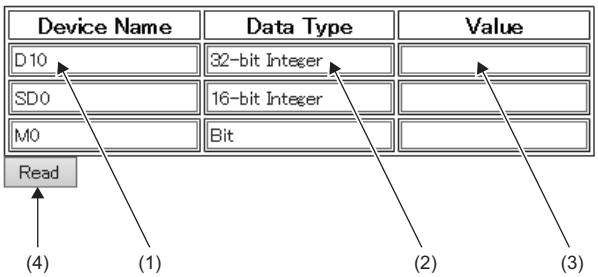
To display the device value in the real number format on the Web page, specify D: Double word for the device size and execute a request (1). Convert the read data to the real number format by using JavaScript (2).



## ■Application example

An example for reading values by using the device read CGI is shown below.

- Display of HTML




(1) Set DEV1.
(2) Set TYP1.
(3) Set VAL1.
(4) Click to call the ReadDeviceRandomTbl function of JavaScript. (Set the table ID for the argument.)

- HTML example

```
<form id="devform" name="readdev" method="post">
  <table id="devtbl" class="devtbl" border="1">
    <tr>
      <th>Device Name</th><th>Data Type</th><th>Value</th>
    </tr>
    <tbody>
    <tr>
      <td><input type="text" id="DEV1" name=" DEV1" class="input"value='D10' /></td>
      <td><input type="text" id="TYP1" name=" TYP1" class="input"value='32-bit Integer' /></td>
      <td><input type="text" id="VAL1" name=" VAL1" class="read-input"/></td>
    </tr>
    <tr>
      <td><input type="text" id="DEV2" name="DEV2" class="input" value='SD0'/ ></td>
      <td><input type="text" id="TYP2" name="TPY2" class="input" value='16-bit Integer'/ ></td>
      <td><input type="text" id="VAL2" name="VAL2" class="read-input"/></td>
    </tr>
    <tr>
      <td><input type="text" id="DEV3" name="DEV3" class="input"value='M0'/ ></td>
      <td><input type="text" id="TYP3" name="TYP3" class="input"value='Bit'/ ></td>
      <td><input type="text" id="VAL3" name="VAL3" class="read-input"/></td>
    </tr>
    </tbody>
  </table>
  <input type="button" value="Read" class="input" onclick=" ReadDeviceRandomTbl(devtbl)"/>
  </form>
```

- JavaScript example

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <!-- charset setting*Set UTF-8 which is set in the Web server setting -->
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
    <!-- Set the title name -->
    <title>Sample</title>
    <link href="./css/UserWebStyle.css" rel="stylesheet" media="all" />
    <!-- Library JavaScript reading (change the path depending on the storage position) -->
    <script src="./RUserWebLib.js"></script>
    <!-- Write the original JavaScript hereinafter -->
    <!-- Write the JavaScript of the user here -->
    <script>
      // The function for a CGI request
      function ReadDeviceRandomTbl(devtblid) {
        var devtblitem = document.getElementById(devtblid);
        var i, devitem, typitem;
        var tblrows = devtblitem.rows.length;
        var param;

        // Number of device points setting
        param = "NUM=" + (tblrows - 1) + '&';
        for (i = 1; i < tblrows; i++) {
          // Parameter setting of the device name
          devitem = document.getElementById(devtblitem.rows[i].cells[0].childNodes[0].id);
          param += devitem.name + "=" + devitem.value + '&';

          // Parameter setting of the device size
          typitem = document.getElementById(devtblitem.rows[i].cells[1].childNodes[0].id);
          if( "Bit" == typitem.value) {
            param += typitem.name + "=" + 'B';
          }
          else if("16-bit Integer" == typitem.value) {
            param += typitem.name + "=" + 'W';
          }
          else if("32-bit Integer" == typitem.value) {
            param += typitem.name + "=" + 'D';
          }
          else {
            param += typitem.name + "=" + 'Q';
          }
          if( i < (tblrows - 1)) param += '&';
        }

        // Request to the CGI
        xhr = new XMLHttpRequest();
        xhr.open('POST', "/cgi/RdDevRnd.cgi", true);
        xhr.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded');
        var FUNC = function() { ReadDeviceRandomTbl_Response(xhr, devtblid); }; // Response analysis function setting
        xhr.onreadystatechange = FUNC;
        xhr.send(param);
      }
```

```
// The function for analyzing a response
  function ReadDeviceRandomTbl_Response(xhr, devtblid) {
    // XMLHttpRequest Client status check
    // 0:UNSENT 1:OPENED 2:HEADERS_RECEIVED 3:LOADING 4:DONE
    if( 4 != xhr.readyState ) {
      // End the processing if the status 4 is other than DONE (operation complete).
      return;
    }

    // HTTP Response code check
    if ( 200 != xhr.status ) {
      // Display the error dialog box if the response code is other than "200 OK".
      alert("HTTP STATUS ERROR=" + xhr.status );
      return;
    }

    var i, dataitem;
    var devtblitem = document.getElementById(devtblid);
    var tblrows = devtblitem.rows.length; // Obtain the number of the table lines (including the header).
    var res = JSON.parse( xhr.response ); // Analysis processing of JSON string

    // Judgment from the CGI
    if( res.RET != "0000" ) {
      // Display the error dialog box if the result is abnormal.
      alert("ERROR=" + res.RET);
    }
    else {
      // Reflect the obtained value to the table if the result is normal.
      for ( i = 1, m = 0; i < tblrows; i++, m++) {
        dataitem = document.getElementById(devtblitem.rows[i].cells[2].childNodes[0].id);
        // Set the read result to the value of the table (convert the hexadecimal string to numerical value).
        dataitem.value = parseInt(res.DATA[m],16);
      }
      alert("read complete");
    }
  }
  </script>
</head>
```

```
<body>
    <form>
      <table id="devtbl" class="devtbl" border="1">
        <tr>
          <th>Device Name</th>
          <th>Data Type</th>
          <th>Value</th>
        </tr>
        <tbody>
          <tr>
            <td><input type="text" id="DEV1" name="DEV1" class="input" value="D10"/></td>
            <td><input type="text" id="TYP1" name="TYP1" class="input" value="16-bit Integer"/></td>
            <td><input type="text" id="DATA1" name="DATA1" class="read-input" ></td>
          </tr>
          <tr>
            <td><input type="text" id="DEV2" name="DEV2" class="input" value="D11"/></td>
            <td><input type="text" id="TYP2" name="TYP2" class="input" value="32-bit Integer"/></td>
            <td><input type="text" id="DATA2" name="DATA2" class="read-input" /></td>
          </tr>
          <tr>
            <td><input type="text" id="DEV3" name="DEV3" class="input" value="M0"/></td>
            <td><input type="text" id="TYP3" name="TYP3" class="input" value="Bit"/></td>
            <td><input type="text" id="DATA3" name="DATA3" class="read-input" /></td>
          </tr>
        </tbody>
      </table>
      <input type="button" value="Read" class="input" onclick="ReadDeviceRandomTbl('devtbl')"/>
    </form>
  </body>
</html>
```

| Device Name | Data Type | Value |
|---|---|---|
| D10 | 16-bit Integer | |
| D11 | 32-bit Integer | |
| M0 | Bit | |

Read

(5)

(5) Display the read result.

**Point**

In the above example, the request parameter is as follows:
NUM=3&DEV1=D10&TYP1=D&DEV2=SD0&TPY2=W&DEV3=M0&TYP3=B

## Device write CGI

Writes the specified value to the specified device.

### ■Access method and access information

| Item | Description |
|---|---|
| Access method | POST |
| Access destination information (URL) | /cgi/WrDev.cgi |

### ■Request specifications

The following table lists the parameters used for the request.

| Parameter name | Data type | Description | Setting range |
|---|---|---|---|
| NUM | string | Number of write device points (1) | Set 1.<br>(Set this parameter so that the total number of device points specified to read/write per Web page is within 32 points.) |
| DEV1 | string | Device name | Up to 16 alphanumeric characters<br>(This parameter is not case-sensitive. The indirect specification, bit specification, digit specification, or index modification cannot be performed.)<br>☞ Page 161 Device name |
| TYP1 | string | Device size | B: Bit<br>W: Word<br>D: Double word<br>Q: Quad word<br>☞ Page 162 Device size |
| DATA1 | string | Write value | Hexadecimal string |

The format of the request data is the query string.

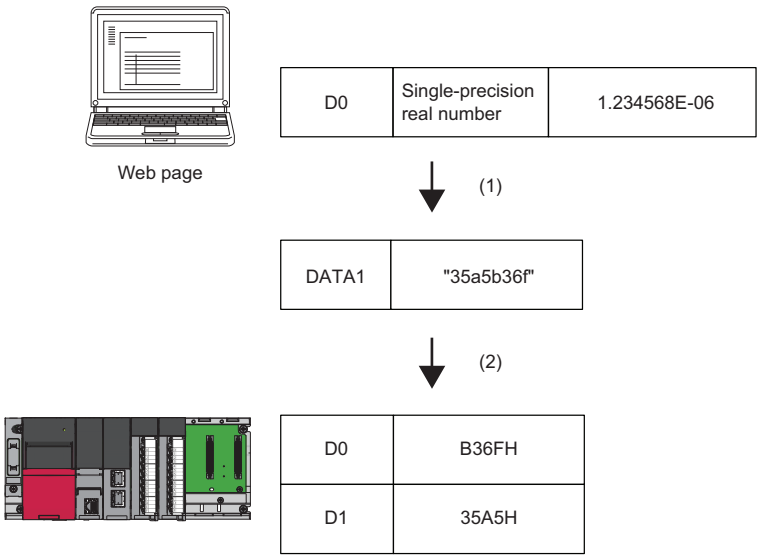
Ex.

Writing FFFFH to D0

NUM=1&DEV1=D0&TYP1=W&DATA1=FFFF

**Point**

When writing the device value which is input to the Web page in the real number format, the input data needs to be converted to the hexadecimal format by using JavaScript (1). For the device size, D: Double word needs to be specified and performing a request (2) are required as well.



## ■Response specifications

The following table lists the parameters used for the response.

| Parameter name | Data type | Description |
|---|---|---|
| RET | string | Execution result (hexadecimal string)<br>0000: Normal<br>0001: Not logged on<br>0002: No permission (A user without device write permission executed the CGI.)<br>0005: Incorrect request source (Referer)<br>4***: Error code (Manual for the CPU module used) |
| DATA | string | The read value of the target device (array)<br>Hexadecimal string |

The format of the response data is JSON.

**Ex.**

Response data of the device write CGI

```
{
   "RET" : "0",
   "DATA" : [
      "100"
   ]
}
```

The above response is transferred in the following format on a message.

```
{"RET":"0","DATA":["100"]}
```

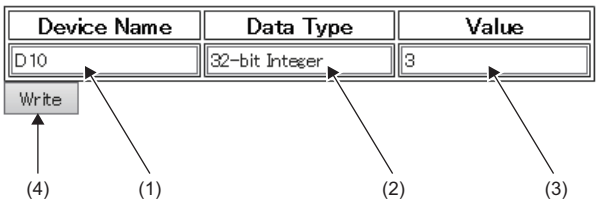
When the execution result is abnormal, data is not written and only the RET is transferred.

```
{"RET":"4031"}
```

## ■Application example

An example for writing a value by using the device write CGI is shown below.

- Display of HTML




(1) Set DEV1.
(2) Set TYP1.
(3) Set VAL1.
(4) Click it to call the WriteDeviceBlockTbl function of JavaScript. (Set the table ID, the line number where the write starts, and the number of write points 1 for the arguments.)

- HTML example

```
<form id="devform" name="readdev" method="post">
  <table id="devtbl" class="devtbl" border="1">
    <tr>
      <th>Device Name</th><th>Data Type</th><th>Value</th>
    </tr>
    <tbody>
    <tr>
      <td><input type="text" id="DEV1" name=" DEV1" class="input"value='D10'/ ></td>
      <td><input type="text" id="TYP1" name=" TYP1" class="input"value='32-bit Integer'/ ></td>
      <td><input type="text" id="DATA1" name=" DATA1" class="input"value='3'/></td>
    </tbody>
  </table>
  <input type="button" value="Write" class="input" onclick=" WriteDeviceBlockTbl(devtbl,1,1)"/>
</form>
```

- JavaScript example

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <!-- charset setting*Set UTF-8 which is set in the Web server setting -->
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
    <!-- Set the title name -->
    <title>Device write CGI sample</title>
    <link href="./css/UserWebStyle.css" rel="stylesheet" media="all" />
    <!-- Library JavaScript reading (change the path depending on the storage position) -->
    <script src="./RUserWebLib.js"></script>
    <!-- Write the original JavaScript hereinafter -->
    <!-- Write the JavaScript of the user here -->
    <script>
      function WriteDeviceBlockTbl(devtblid, row) {
      var dataitem;
      var xhr;
      var devtblitem = document.getElementById(devtblid);
      // Number of device points setting (fixed to one point)
      var param = 'NUM=1&';
      // Parameter setting of the device name
      var devitem = document.getElementById(devtblitem.rows[row].cells[0].childNodes[0].id);
      param += 'DEV1=' + devitem.value + '&';

      // Parameter setting of the device size
      var typitem = document.getElementById(devtblitem.rows[row].cells[1].childNodes[0].id);
      if( 'Bit' == typitem.value){
        param += 'TYP1=' + 'B';
      }
      else if( '16-bit Integer' == typitem.value) {
        param += 'TYP1=' + 'W';
      }
      else if( '32-bit Integer' == typitem.value) {
        param += 'TYP1=' + 'D';
      }
      else {
        param += 'TYP1=' + 'Q';
      }
      param += '&';

      // Parameter setting of the data
      var dataitem = document.getElementById(devtblitem.rows[row].cells[2].childNodes[0].id);
      param += 'DATA1=' + parseInt(dataitem.value).toString(16)

      // Request to the CGI
      xhr = new XMLHttpRequest();
      xhr.open('POST', "/cgi/WrDev.cgi", true);
      xhr.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded');
      var FUNC = function() { WriteDeviceBlockTbl_Response(xhr, typitem, dataitem); }; // Response analysis function setting
      xhr.onreadystatechange = FUNC;
      xhr.send(param);
    }
```

```
// The function for analyzing a response
        function WriteDeviceBlockTbl_Response(xhr, typitem, dataitem) {
            // XMLHttpRequest Client status check
            // 0:UNSENT 1:OPENED 2:HEADERS_RECEIVED 3:LOADING 4:DONE
            if( 4 != xhr.readyState ) {
                // End the processing if the status 4 is other than DONE (operation complete).
                return;
            }
            // HTTP Response code check
            if ( 200 != xhr.status ) {
                // Display the error dialog box if the response code is other than "200 OK".
                alert("HTTP STATUS ERROR=" + xhr.status );
                return;
            }

            var value;
            var res = JSON.parse( xhr.response ); // Analysis processing of JSON string
            // Judgment from the CGI
            if( res.RET != "0000" ) {
                // Display the error dialog box if the result is abnormal.
                alert("ERROR=" + res.RET);
            }
            else {
                // Reflect the write result value if the result is normal.
                dataitem.value = parseInt(res.DATA[0],16);
                alert("write complete");
            }
        }
    </script>
</head>
```

```
<body>
  <form>
    <table id="devtbl" class="devtbl" border="1">
      <tr>
        <th>Device Name</th>
        <th>Data Type</th>
        <th>Value</th>
      </tr>
      <tbody>
        <tr>
          <td><input type="text" id="DEV1" name="DEV1" class="input" value="D10"/></td>
          <td><input type="text" id="TYP1" name="TYP1" class="input" value="16-bit Integer"/></td>
          <td><input type="text" id="DATA1" name="DATA1" class="input" value="3"/></td>
          <td><input type="button" value=" Write" class="input" onclick="WriteDeviceBlockTbl('devtbl',1)"/></td>
        </tr>
        <tr>
          <td><input type="text" id="DEV2" name="DEV2" class="input" value="D11"/></td>
          <td><input type="text" id="TYP2" name="TYP2" class="input" value="32-bit Integer"/></td>
          <td><input type="text" id="DATA2" name="DATA2" class="input" value="10"/></td>
          <td><input type="button" value=" Write" class="input" onclick="WriteDeviceBlockTbl('devtbl',2)"/></td>
        </tr>
        <tr>
          <td><input type="text" id="DEV3" name="DEV3" class="input" value="M0"/></td>
          <td><input type="text" id="TYP3" name="TYP3" class="input" value="Bit"/></td>
          <td><input type="text" id="DATA3" name="DATA3" class="input" value="1"/></td>
          <td><input type="button" value=" Write" class="input" onclick="WriteDeviceBlockTbl('devtbl',3)"/></td>
        </tr>
      </tbody>
    </table>
    </form>
  </body>
</html>
```

**Point**

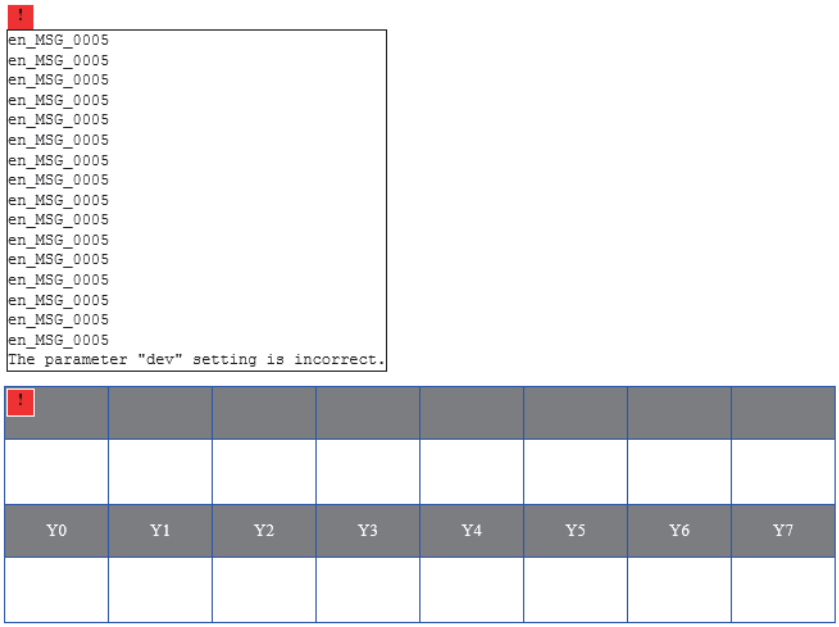
In the above example, the request parameter is as follows:
NUM=1&DEV1=D10&TYP1=D&DATA1=3

# Error message



## Display example of error messages

If an object of the user Web page is used incorrectly, a button to notify the user of an error is displayed at the upper left of the object when the user Web page is displayed. The error message related to the common processing which is not limited to the object is displayed at the upper left of the window.




Click the button to display the detailed information of the error.

## ■List of error messages of the user Web page

The following table lists error messages, causes, and action of the user Web page.

| Description | Cause | Action |
|---|---|---|
| A required parameter "XXX" has not been set. | A required parameter has not been set. | Check the contents of the HTML file. |
| The setting of the parameter "XXX" is incorrect. | Parameters are out of the setting range. | |
| | Set values of parameters that should be numeric values are specified with other than numeric values. | |
| The data format of the device "XXX" is incorrect. | Positional notation of the device, data format, device name | |
| The receive data contains errors. | The receive data contains errors. | Check that the CPU module and the client terminal are properly connected and update the Web page. |
| You were logged off. Log on and try again. | The user has been logged off. (Communications are disconnected.) | Check that the CPU module and the terminal are properly connected and log on again. |
| Specified devices are invalid. The following are the possible causes.<br>• A device such as pointer and constant<br>• Unsupported device | Device names are incorrect. | Check parameters of each object to see if any device name is specified incorrectly. |
| Specified device No. is invalid. The following are the possible causes.<br>• A device over the settable device range of CPU parameter | Device numbers and buffer memory addresses are out of range or not numeric values. The device range is incorrect. | Check parameters of each object to see if any device number and buffer memory address are specified incorrectly. |
| Specified modules do not exist. Check the modules. | The module numbers are incorrect.<br>An error has occurred in communications with intelligent function modules. | Check parameters of each object to see if any module number is specified incorrectly.<br>Check that there are no power-offs or errors in extension units. |

For details on the other errors, refer to the following.

📖 MELSEC iQ-R/MELSEC iQ-F Web Server Function Guide Book

📖 Manual for the CPU module used

# 1.13 Security

Security depending on the network environment can be structured by restricting access by each communication path to the CPU module.

For the restriction methods for access, refer to the following.

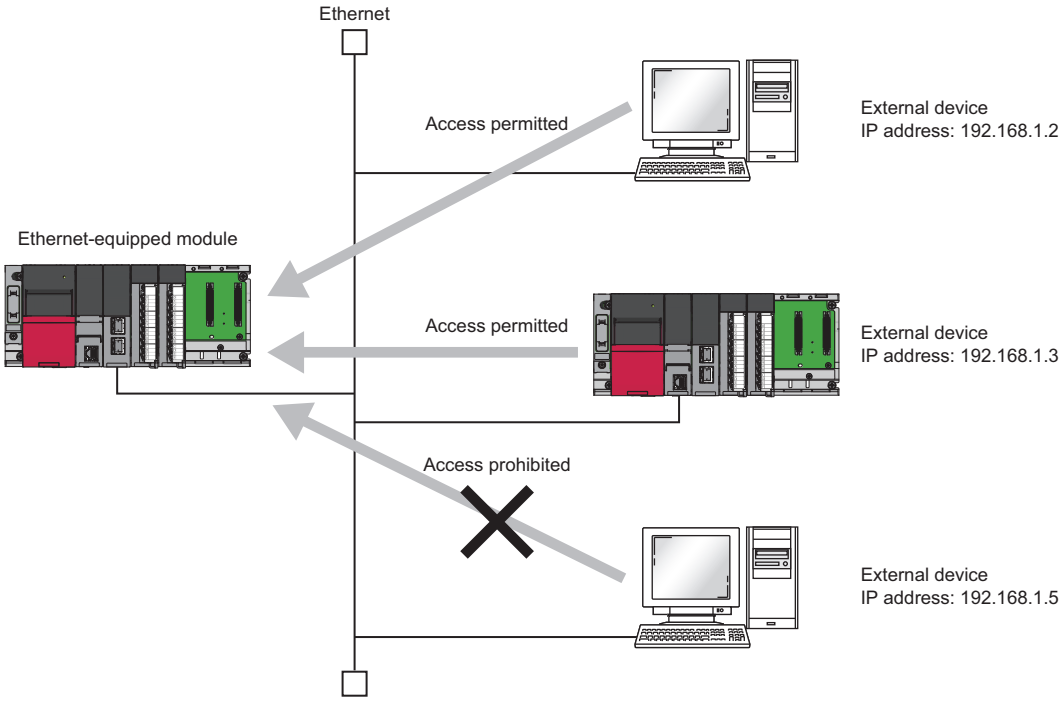
- Page 177 IP filter
- Page 178 Remote password

## IP filter

Identifies the IP address of the access source, and prevents access from an illegal IP address.

The IP address of the external device to be allowed or denied is set in the parameters, and access from external devices is restricted.

Use of this function is recommended when using in an environment connected to a LAN line.




**Restriction**

- This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".
- Access cannot be restricted when the network other than Ethernet is set as the communication path.

**Point**

The IP filter is one method of preventing illegal access (such as program or data destruction) from an external device. It does not completely prevent illegal access. Incorporate measures other than this function if the programmable controller system's safety must be maintained against illegal access from an external device. Mitsubishi shall not be held liable for any system problems that may occur from illegal access.

Examples of measures for illegal access are given below.

- Install a firewall
- Install a personal computer as a relay station, and control the relay of send/receive data with an application program
- Install an external device for which the access rights can be controlled as a relay station (Contact the network provider or equipment dealer for details on the external devices for which access rights can be controlled.)

### Setting procedure

1. Set the IP address to be allowed or denied in "IP Filter Settings" of "Security" under "Application Settings". (☞ Page 404 IP Filter Settings)
2. Write the module parameters to the CPU module.
3. The IP filter is enabled when the CPU module power is turned off and on or reset.

**Point**

Even if the connection is established as set with the Ethernet-equipped module's "External Device Configuration" under "Basic Settings" or the program, access from the external device is either allowed or denied following "IP Filter Settings" of "Security" under "Application Settings".
Therefore, if the IP address set in the Ethernet-equipped module's "External Device Configuration" under "Basic Settings" is set to be denied with "IP Filter Settings" of "Security" under "Application Settings", the IP filter is enabled and communication with the external device is denied.

### Precautions

If there is a proxy server in the LAN line, deny the IP address for the proxy server. If the IP address is allowed, it will not be possible to prevent access from personal computers that access the proxy server.

## Remote password

Permits or prohibits access from the external device to the CPU module via the Ethernet-equipped module. This function can prevent illegal access of the CPU module from a remote location.

**Point**

The remote password is one method of preventing illegal access (such as program or data destruction) from an external device. It does not completely prevent illegal access. Incorporate measures other than this function if the programmable controller system's safety must be maintained against illegal access from an external device. Mitsubishi shall not be held liable for any system problems that may occur from illegal access.
Examples of measures for illegal access are given below.

- Install a firewall
- Install a personal computer as a relay station, and control the relay of send/receive data with an application program
- Install an external device for which the access rights can be controlled as a relay station (Contact the network provider or equipment dealer for details on the external devices for which access rights can be controlled.)

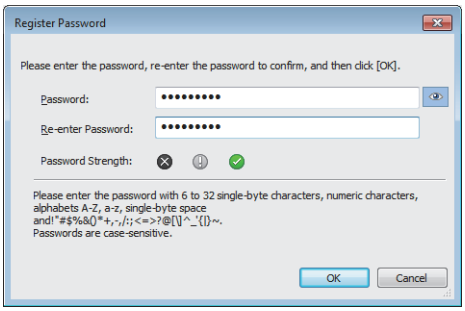
### Number of settable modules

Up to eight modules can be set for remote passwords.
When using the multiple CPU system configuration, up to eight modules can be set for each CPU module.

## Setting procedure

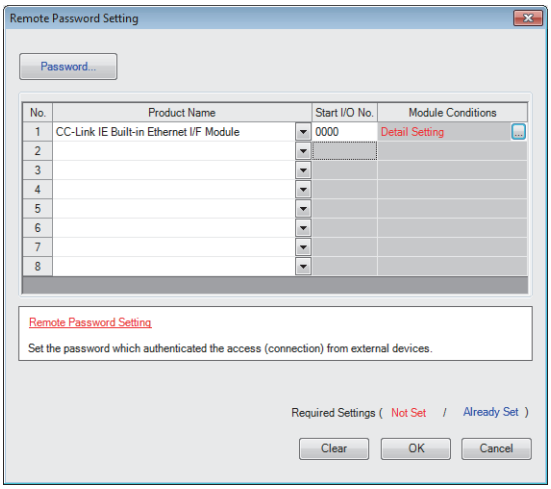
Set the remote password as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Remote Password]



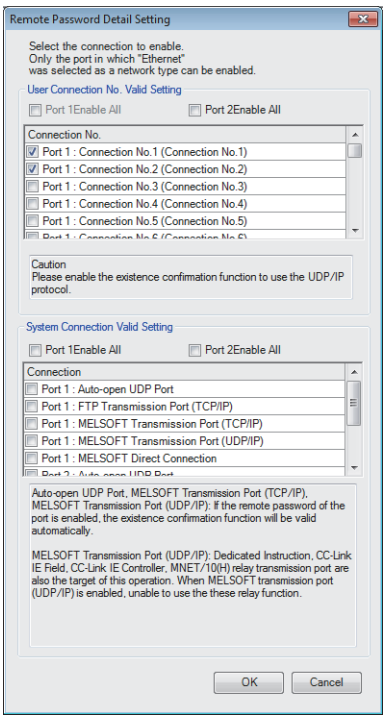

**1.** Register the remote password on the "Register Password" window.

[Password] button




**2.** Select the module for which the remote password is to be applied, and set the start I/O number. Select the following items when using the RnENCPU.

- CPU part: "CPU Module (Built-in Ethernet Function)"
- Network part: "CC-Link IE Built-in Ethernet I/F Module"




**3.** Set the target connection on the "Remote Password Detail Setting" window.

"Detail Setting" for the target module.

**4.** Write the remote password to the CPU module.

**5.** The remote password is enabled when the CPU module is powered off and on or reset.

## Access permit/prohibit processing operation

This section describes the processing for permitting or prohibiting access of the CPU module with remote password by the external device.

### ■Access permit processing (Unlock processing)

The external device trying to communicate unlocks the remote password set for the connected Ethernet-equipped module. If the password is not unlocked, the Ethernet-equipped module to which the external device is connected prohibits access, so an error occurs in the external device.

The unlocking method is shown below.

- SLMP dedicated command (Remote Password Unlock)
- Dedicated command (password-unlock) for file transfer function (FTP server)
- Input password from engineering tool

If unlock processing (password authentication) fails a certain number of times, the lockout function provided will deny password authentication for a certain period of time. The lockout time corresponding to the number of password input times is shown below.

| Number of password input errors*1 | Lockout time*2 |
|---|---|
| 1st to 5th time | 0 minutes |
| 6th time | 1 minute |
| 7th time | 5 minutes |
| 8th time | 15 minutes |
| 9th time or more | 60 minutes |

*1 The number of password input errors is reset when the correct password is input.

*2 During lockout, the number of password input errors are not counted. Therefore, even if the 7th password input error occurs within one minute of the 6th password input error, the lockout time will not be extended by one minute.

### ■Access processing

Access to the specified station is possible when the remote password is correctly unlocked. Execute the arbitrary access.

### ■Access prohibit processing (Lock processing)

When access to the specified station ends, lock the remote password from the external device to disable subsequent access.

The locking method is shown below.

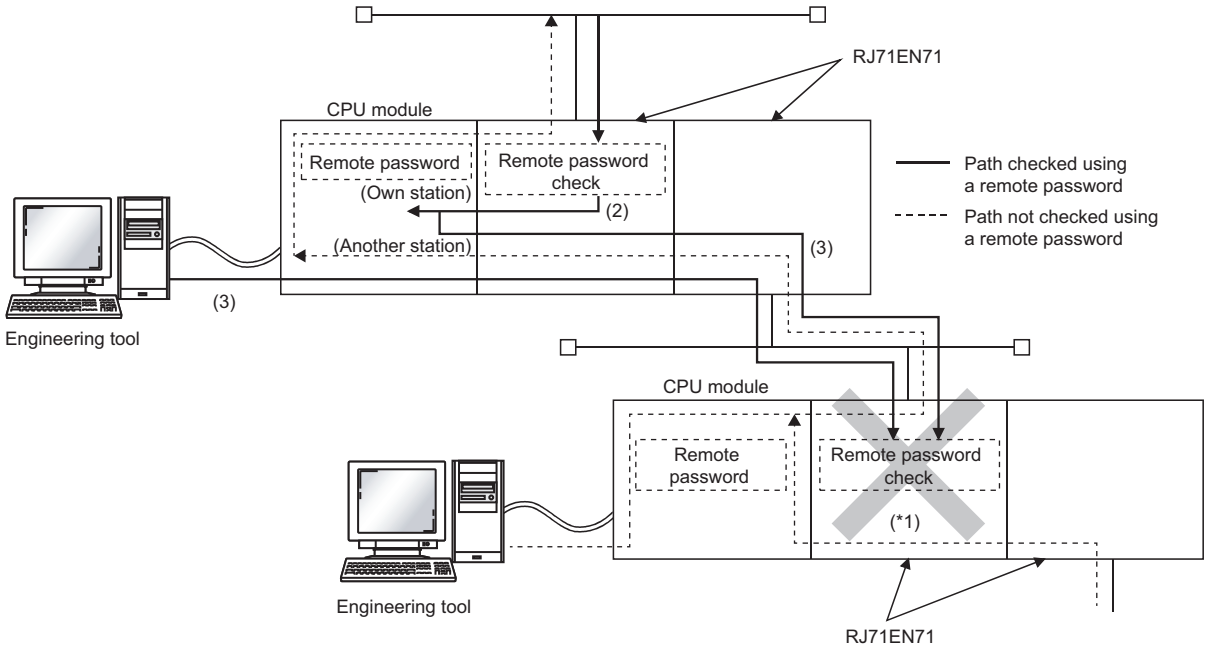
- SLMP dedicated command (Remote Password Lock)
- Dedicated command (password-lock) for file transfer function (FTP server)
- Lock with engineering tool (executed automatically)

## Remote password check operation

### ■Communication that is checked

The Ethernet-equipped module checks the remote password for a communication request made to the own station or other station received from the external device.

When checking a remote password for modules with multiple connections, the connection for which the remote password is set.



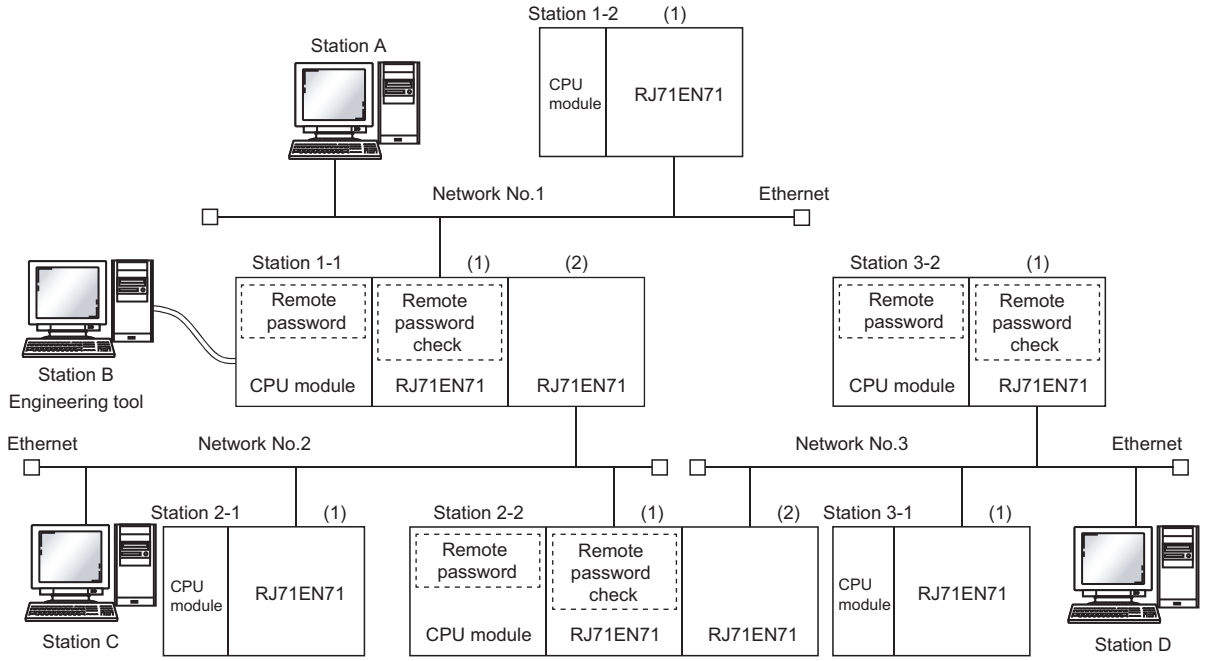

*1 The remote password check is set, so the communication request from the external device is not accepted. If the remote password check is not set, the communication request can be accepted and data can be exchanged from the external device.

## ■Accessible station

The station accessible from the external device when a remote password is set for the CPU module and the station that can unlock/lock the remote password are limited to those in the same network number.

The following figure shows an example of when the remote password is set for multiple stations in the system.

Ex.




*1 The password can be unlocked and locked by the following external devices.
1-1 station (1): A station only
2-2 station (1): C station only
3-2 station (1): D station only

○: Station accessible from external device after remote password is unlocked

△: Station accessible from external device even if remote password is not unlocked

×: Station that cannot be accessed from external device

| External device*2 (Request source) | Target programmable controller (request source) | | | | | |
|---|---|---|---|---|---|---|
| | 1-1 station CPU | 1-2 station CPU | 2-1 station CPU | 2-2 station CPU | 3-1 station CPU | 3-2 station CPU |
| A station | ○ | △ | ○ | × | × | × |
| B station | △ | △ | △ | × | × | × |
| C station | △ | △ | △ | ○ | ○ | × |
| D station | △ | △ | △ | △ | △ | ○ |

*2 The A station accesses the ○ station after the remote password for 1-1 station (1) is unlocked and access the △ station if the communication line is open.
The B station accesses the △ station if the communication line is open.
The C station accesses the ○ station after the remote password for 2-2 station (1) is unlocked and access the △ station if the communication line is open.
The D station accesses the ○ station after the remote password for 3-2 station (1) is unlocked and access the △ station if the communication line is open.

## Precautions

The following section lists the precautions for using remote password.

### ■Set connection

Set the remote password for the connection used to exchange data with an external device that can execute the unlock/lock processing.

### ■When remote password is set for UDP/IP connection

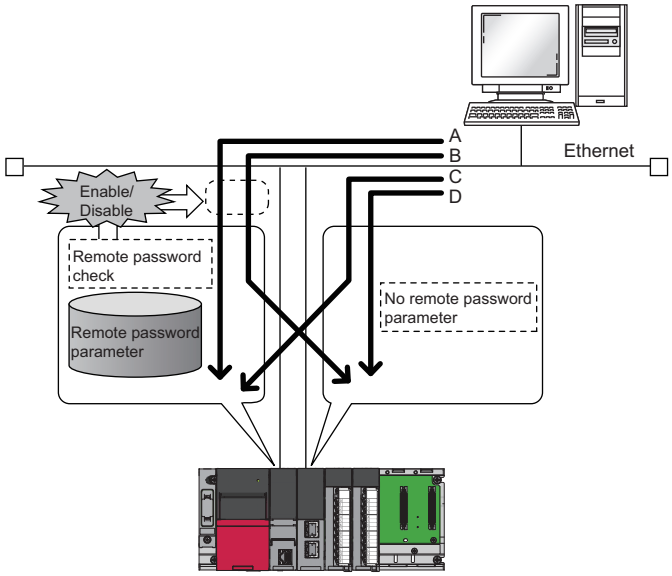
- Determine the external device to communicate with and exchange the data. (With UDP/IP, after the remote password is unlocked data can be exchanged with devices other than the unlocked external device too. Determine the communication device before starting use.)
- Always lock the remote password after data communication is finished. (If the remote password is not locked, the unlocked state is held until timeout occurs.)
- Set "UDP" for "Existence Confirmation" of the target connection in "External Device Configuration" under "Basic Settings". Set as small a value as possible for the "Destination Alive Check Start Interval Timer" and "Destination Alive Check Interval Timer" in "Timer Settings for Data Communication" under "Application Settings".

### ■TCP/IP close processing

If the TCP/IP is closed before the TCP/IP is locked, the CPU module will automatically start the lock processing.

### ■Remote password valid range

The remote password is valid only for access from the Ethernet-equipped module for which the parameters are set. When using multiple CPU modules in a multi-CPU system, set a remote password for each CPU module for requiring a remote password.




The remote password is checked when accessing with path A or B.
The remote password is not checked when accessing with path C or D.

### ■Connection for nonprocedural fixed buffer communication

The remote password is not checked during communication for nonprocedural fixed buffer communication, so do not set the remote password check.

### ■Accessing another station's programmable controller

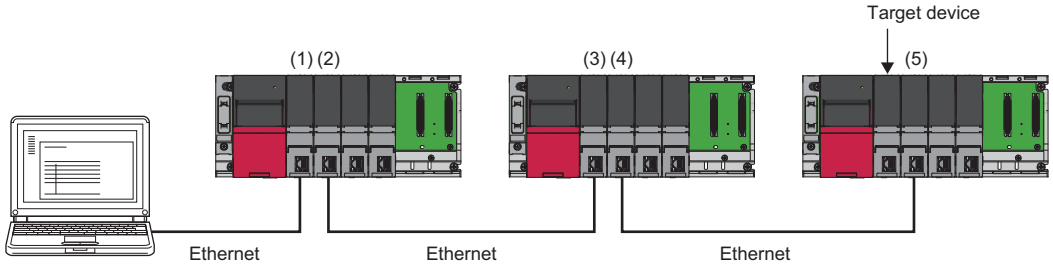
When the external device is accessing another station's programmable controller via the Ethernet-equipped module, it may not be possible to access the programmable controller if a remote password is set for the CPU module at the relay station or access station.

### ■When the CPU module is used as a relay station for the connection between the MELSOFT product and GOT

A remote password is enabled when the following items are set in "System Connection" in "Remote Password Detail Setting".

| "PLC side I/F Detailed Setting of PLC Module" of "Current Connection Destination" | "System Connection" of "Remote Password Detail Setting" |
|---|---|
| Ethernet port direct connection | MELSOFT direct connection |
| Connection via a hub | MELSOFT transmission port (UDP/IP) |

The following figure shows the availability of the access when a remote password is set for a relay station (CPU module).




When a remote password is set for (1), the target device becomes accessible after the remote password is unlocked.
When remote passwords are set for (2) to (5), the target device is accessible even when the remote passwords are not unlocked.

# 1.14 Simple CPU Communication Function (RJ71EN71, RnENCPU (Network Part))



This function allows data communications between specified devices at the specified timing just by making simple settings of the RJ71EN71 and RnENCPU (network part) from an engineering tool. Send/receive is performed with a specified communication destination on a one-to-one basis.

Use this function for communicating device data with programmable controllers.




| Communication destination | Type | Description |
|---|---|---|
| MELSEC iQ-R series | CPU module, Ethernet module | ☞ Page 189 Module type |
| MELSEC-Q series | CPU module, Ethernet module | |
| MELSEC-L series | CPU module, Ethernet module | |
| MELSEC iQ-F series | CPU module | |
| MELSEC iQ-L series | CPU module | |
| MELSEC-A/AnS series | Ethernet module | |
| MELSEC-F series | Ethernet block/adapter | |
| Programmable controllers of other companies | | |
| SLMP-compatible device | QnA-compatible 3E frame | |
| MODBUS/TCP slave device | | |

**Point**

Access via routers is also available. For the access, set the subnet mask and default gateway. (☞ Page 327 Own Node Settings)

**Restriction**

Before using the simple CPU communication function, check the versions of the module and engineering tool used. (☞ Page 226 Restrictions applicable depending on versions)

## Comparison of the simple CPU communication function

The simple CPU communication function differs between the RJ71EN71 and RnENCPU (network part) and the CPU module (built-in Ethernet port part).

For details on the simple CPU communication function of the CPU module (built-in Ethernet port part), refer to the following.

☞ Page 228 Simple CPU Communication Function (CPU Module (Built-in Ethernet Port Part))

| Item | | RJ71EN71, RnENCPU (network part)*2 | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| Number of connectable modules | | 64 | 64 |
| Number of settings | | 512*1 | 64 |
| Number of points available for communication | One setting | 1024 words maximum<br>(word device: 512 points maximum, bit device: 8192 points maximum) | 512 words maximum<br>(word device: 512 points maximum, bit device: 8192 points maximum) |
| | All settings | 524288 words maximum | 8192 words maximum |
| Communication pattern | | Write/Read | Write/Read |
| Communication setting | | Fixed interval/On request*1 | Fixed interval/On request*1 |
| Communication destination device | | • MELSEC iQ-R (Built-in Ethernet)<br>• MELSEC-Q (Built-in Ethernet)<br>• MELSEC-L (Built-in Ethernet)<br>• MELSEC iQ-F (Built-in Ethernet)*1<br>• MELSEC iQ-L (Built-in Ethernet)*1<br>• MELSEC iQ-R (Ethernet Module)<br>• MELSEC-Q (Ethernet Module)<br>• MELSEC-L (Ethernet Module)<br>• MELSEC-A/AnS (Ethernet Module)*1<br>• MELSEC-FX3 (Ethernet Block/Adapter)*1<br>• SLMP Corresponding Device (QnA Compatible 3E Frame)<br>• OMRON (CS/CJ series)<br>• OMRON (NJ/NX series)*1<br>• KEYENCE (KV series)<br>• Panasonic (FP2SH series)<br>• Panasonic (FP7 series)<br>• YASKAWA MP3000 series<br>• YASKAWA MP2000 series<br>• Yokogawa FA-M3 series<br>• MODBUS/TCP-compatible device<br>• SIEMENS S7 series*1<br>• SIEMENS S7 series (extension)*1<br>• Fuji Electric MICREX-SX series*1<br>• JTEKT TOYOPUC series*1 | • MELSEC iQ-R (Built-in Ethernet)<br>• MELSEC MX Controller (MX-R Model) (Built-in Ethernet)*1<br>• MELSEC MX Controller (MX-F Model) (Built-in Ethernet)*1<br>• MELSEC-Q (Built-in Ethernet)<br>• MELSEC-L (Built-in Ethernet)<br>• MELSEC iQ-F (Built-in Ethernet)*1<br>• MELSEC iQ-L (Built-in Ethernet)*1<br>• MELSEC iQ-R (Ethernet Module)*1<br>• MELSEC-Q (Ethernet Module)*1<br>• MELSEC-L (Ethernet Module)*1<br>• MELSEC-A/AnS (Ethernet Module)*1<br>• MELSEC-FX3 (Ethernet Block/Adapter)*1<br>• SLMP Corresponding Device (QnA Compatible 3E Frame) |
| Target device (own station)*1 | Bit device | X, Y, M, L, F, B, TS, TC, STS, STC, CS, CC, SB, SM | X, Y, M, L, F, B, TS, TC, STS, STC, CS, CC, SB, SM |
| | Word device | D, W, SW, TN, STN, CN, SD, R, ZR, RD, G (buffer memory) | D, W, SW, TN, STN, CN, SD, R, ZR, RD |
| Execution interval setting range | | 1ms to 65535ms | 10ms to 65535ms |
| Monitoring time setting range at error | | 1s to 300s | 1s to 300s |

*1 The function differs depending on the firmware version of the module. (☞ Page 586 Added and Enhanced Functions)

*2 The specifications are for one port.

# Setting procedure

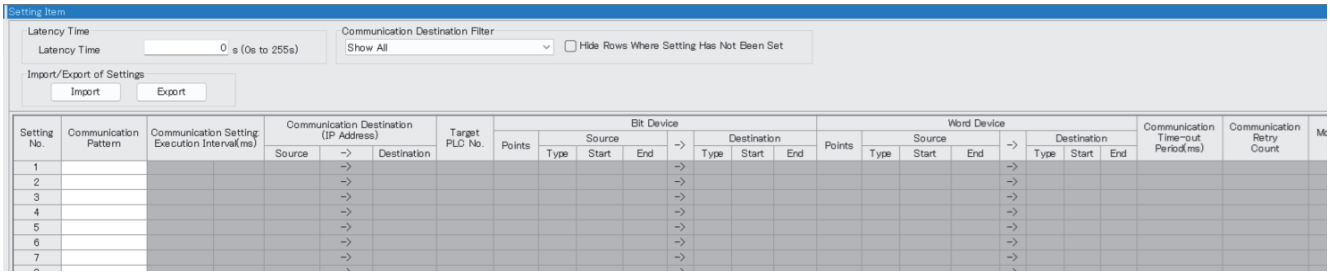


The following section describes the settings for using this function.

1. Set "Enable" for "To Use or Not to Use Simple CPU Communication Setting" in "Simple CPU Communication Setting" under "Application Settings".

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple CPU Communication Setting]

2. Double-click "<Detailed Setting>" of the "Simple CPU Communication Setting" to display the following window. (☞ Page 342 Simple CPU communication setting (RJ71EN71, RnENCPU (network part)))




3. After setting the items, write the module parameters to the CPU module on the own station, and reset or power off and on the CPU module to start the simple CPU communication.

**Point**

When "On Request" is set to the communication setting, refer to the following.

☞ Page 209 Operations when the communication setting is "On Request"

## Communication Pattern

Select the communication pattern from the following items.

| Setting item | Description |
|---|---|
| Read | Read the data of the specified destination device (transmission source) to the specified device of the own station (transmission destination). |
| Write | Write the data of the specified device of the own station (transmission source) to the specified destination device (transmission destination). |

## Communication Setting: Execution Interval

Select the communication timing from the following items.

| Setting item | | Description | Setting range |
|---|---|---|---|
| Fixed Interval | — | Data are communicated between the devices at a specified execution interval. | — |
| | Execution Interval (ms) | Execution interval is set. | 1ms to 65535ms (in increments of 1ms) |
| On Request | | Data are communicated between the devices upon request only. | — |

**Point**

- Actual time of execution interval may be longer than the value of the setting because the time is affected by the specified communication destinations or Ethernet line congestion. For the preventive measures, refer to the following.

☞ Page 448 When the simple CPU communication function is used

- If latency time is set, data communication is started after the latency time has passed. (☞ Page 212 Latency Time)

## Communication Destination (IP Address)

Click "Destination" to open the "Communication Destination Setting" window and specify the communication destination. Up to 64 communication destinations can be registered.

**Point**

In the engineering tool, up to 512 communication destinations can be set. Note that the settable number is different.

Devices having different settings in any of the items below are counted as different communication destination.

- Module type
- IP address
- UDP/TCP
- Port No.
- Own station port number

Note that the following modules are counted as a single module type even if the series are different.

- SIEMENS S7 series and SIEMENS S7 series (extension)
- OMRON (CS/CJ series) and OMRON (NJ/NX series)

## ■Module type

Select the module type of the communication destination.

| Module type | Communication destination | Communication specification |
|---|---|---|
| MELSEC iQ-R (Built-in Ethernet) | RnCPU, RnENCPU[*1], RnPCPU, RnSFCPU, RnPSFCPU | MELSOFT connection (protocol: UDP) |
| MELSEC-Q (Built-in Ethernet) | Built-in Ethernet port QCPU | |
| MELSEC-L (Built-in Ethernet) | Built-in Ethernet port LCPU | |
| MELSEC iQ-F (Built-in Ethernet) | MELSEC iQ-F series CPU module[*4] | MELSOFT connection (protocol: TCP) |
| MELSEC iQ-L (Built-in Ethernet) | LnHCPU | MELSOFT connection (protocol: UDP) |
| MELSEC iQ-R (Ethernet Module) | RnENCPU[*2], RJ71EN71 | |
| MELSEC-Q (Ethernet Module) | QJ71E71-100 | |
| MELSEC-L (Ethernet Module) | LJ71E71-100 | |
| MELSEC-A/AnS (Ethernet Module) | AJ71E71N3-T, AJ71QE71N3-T, A1SJ71E71N3-T, A1SJ71QE71N3-T | MC protocol (A-compatible 1E frame) |
| MELSEC-FX3 (Ethernet Block/Adapter) | FX3U-ENET-ADP, FX3U-ENET-L | |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | • SLMP-compatible device (No serial number)<br>• MC protocol 3E frame-compatible device | SLMP (MC protocol QnA-compatible 3E frame)<br>• Read (0401H)<br>• Write (1401H) |
| OMRON (CS/CJ series) | Supported model for SYSMAC CS/CJ series | FINS |
| OMRON (NJ/NX series) | Supported model for SYSMAC NJ/NX series | FINS |
| KEYENCE (KV series) | Supported model for KV series | SLMP (MC protocol QnA-compatible 3E frame) |
| Panasonic (FP2SH series) | Supported model for FP2 series | MEWTOCOL |
| Panasonic (FP7 series) | Supported model for FP7 series | MEWTOCOL7 |
| YASKAWA MP3000 series | Supported model for MP3000 series | Extended MEMOBUS |
| YASKAWA MP2000 series | Supported model for MP2000 series | |
| Yokogawa FA-M3 series | Supported model for FA-M3 series | Personal computer link |
| MODBUS/TCP-compatible device | MODBUS/TCP slave device | MODBUS/TCP |
| SIEMENS S7 series[*3] | Supported model for S7 series | S7 Communication |
| SIEMENS S7 series (extension) | Supported model for S7 series | S7 Communication |
| Fuji Electric MICREX-SX series | • Supported model for MICREX-SX (CPU module) series<br>• Supported model for MICREX-SX (Ethernet module) series | Loader command |
| JTEKT TOYOPUC series | • Supported model for TOYOPUC PC10 (CPU module) series<br>• Supported model for TOYOPUC Plus (CPU module) series<br>• Supported model for TOYOPUC Plus (Ethernet module) series | Computer link |

*1 When the RnENCPU is used for the connection to the Ethernet port of the CPU part

*2 When the RnENCPU is used for the connection to the Ethernet port of the network part

*3 This module type is provided because it can be used for the setting of Communication Destination (IP Address) in the engineering tool with the Ver.1.085P or earlier.
And thus, when newly setting this item, set SIEMENS S7 series (extension) as the module type.

*4 For the FX5U CPU module and FX5UC CPU module, the module with the firmware version of "1.110" or later can be used.

## ■IP address, UDP/TCP, port number

Set the IP address of the communication destination, communication protocol, port number, and own port number.

| Module type | Setting range | | | |
|---|---|---|---|---|
| | IP address | UDP/TCP | Port No. | Own station port number |
| MELSEC iQ-R (Built-in Ethernet) | 0.0.0.1 to 223.255.255.254 | UDP | Cannot be set. | 1 to 4999, 5010 to 65534 |
| MELSEC-Q (Built-in Ethernet) | | UDP | | |
| MELSEC-L (Built-in Ethernet) | | UDP | | |
| MELSEC iQ-F (Built-in Ethernet) | | TCP | | |
| MELSEC iQ-L (Built-in Ethernet) | | UDP | | |
| MELSEC iQ-R (Ethernet Module) | | UDP | | |
| MELSEC-Q (Ethernet Module) | | UDP | | |
| MELSEC-L (Ethernet Module) | | UDP | | |
| MELSEC-A/AnS (Ethernet Module) | | UDP, TCP | 256 to 65534 | |
| MELSEC-FX3 (Ethernet Block/Adapter) | | UDP, TCP | 1025 to 5548, 5552 to 65534 | |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | | UDP, TCP | 1 to 65534 | |
| OMRON (CS/CJ series) | | UDP | | |
| OMRON (NJ/NX series) | | UDP | | |
| KEYENCE (KV series) | | UDP, TCP | | |
| Panasonic (FP2SH series) | | UDP, TCP | | |
| Panasonic (FP7 series) | | UDP, TCP | | |
| YASKAWA MP3000 series | | UDP, TCP | | |
| YASKAWA MP2000 series | | UDP, TCP | | |
| Yokogawa FA-M3 series | | UDP, TCP | | |
| MODBUS/TCP-compatible device | | TCP | | |
| SIEMENS S7 series | | TCP | 102 | |
| SIEMENS S7 series (extension) | | TCP | | |
| Fuji Electric MICREX-SX series | | TCP | 251 to 65531 | |
| JTEKT TOYOPUC series | | TCP | 1025 to 65535 | |

**Point**

- When a multiple CPU system is configured, specify the IP address of the CPU module to which the Ethernet cable is connected.
- Port numbers 1 to 1023 are typically reserved port numbers (WELL KNOWN PORT NUMBERS) and 61440 to 65534 are used by other communication functions, so it is recommended to use 1024 to 4999 or 5010 to 61439 for the own station port numbers.
- If the communication destination is Mitsubishi Electric programmable controllers (except MELSEC-A/AnS and MELSEC-FX3), the number between 61696 and 65534 is dynamically used for the own station port number. Consequently, do not specify the number between 61696 and 65534 for the own station port number in the OPEN instruction and the GP.CONOPEN instruction. Otherwise, the instruction may not be completed properly.
- When the same communication destination is specified in multiple settings, it is recommended to set a different value for each own station port number. If the number of the connections in the external device has exceeded the upper limit, reduce the number of connections by setting the own station port numbers to same values.

### ■Option (hexadecimal)

Set the option setting value of the communication destination.

| Module type | Setting range |
|---|---|
| MELSEC iQ-R (Built-in Ethernet) | Cannot be set. |
| MELSEC-Q (Built-in Ethernet) | |
| MELSEC-L (Built-in Ethernet) | |
| MELSEC iQ-F (Built-in Ethernet) | |
| MELSEC iQ-L (Built-in Ethernet) | |
| MELSEC iQ-R (Ethernet Module) | |
| MELSEC-Q (Ethernet Module) | |
| MELSEC-L (Ethernet Module) | |
| MELSEC-A/AnS (Ethernet Module) | |
| MELSEC-FX3 (Ethernet Block/Adapter) | |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | |
| OMRON (CS/CJ series) | Specify the FINS node address of the own station. (When 0000H is specified, the module operates with node address 1.)*1<br>Setting range: 0000H to 00FDH (default: 0000H) |
| OMRON (NJ/NX series) | |
| KEYENCE (KV series) | Cannot be set. |
| Panasonic (FP2SH series) | |
| Panasonic (FP7 series) | |
| YASKAWA MP3000 series | |
| YASKAWA MP2000 series | |
| Yokogawa FA-M3 series | |
| MODBUS/TCP-compatible device | Set the setting value for the module ID of the MODBUS application header.<br>■When communicating with a MODBUS/TCP-compatible device<br>00H or FFH*2<br>■When communicating with a MODBUS RTU/ASCII-compatible device via a gateway device<br>• 00H (broadcast)<br>• 01H to F7H (station number of a MODBUS RTU/ASCII-compatible device) |
| SIEMENS S7 series | Cannot be set. |
| SIEMENS S7 series (extension) | |
| Fuji Electric MICREX-SX series | |
| JTEKT TOYOPUC series | |

*1 Before using the module, check the versions of the module and engineering tool to be used. (☞ Page 226 Restrictions applicable depending on versions)

*2 Depending on the specifications of the MODBUS/TCP-compatible device, communication may not be possible with 00H. If communication is not possible, specify FFH.

## Precautions

The following are the precautions when 00H (broadcast) is set for communication with a MODBUS RTU/ASCII-compatible device via a gateway device.

- Since the slave station does not response, a timeout error (CFB0H) occurs. Do not specify the read.
- Do not specify both a bit device and a word device in one setting value. Specify only one of them.
- When setting multiple devices, set a different own station port number.

For the communication time-out period, communication retry count, and monitoring time at error, set as follows. When setting the following values, the second communication is executed after Communication Time-out Period, and the communication of a third or later is executed at the cycle of Communication Time-out Period × 2.

- Communication Retry Count = 0
- Communication Time-out Period = Monitoring Time At Error

## Setting for communication destination

The settings differ depending on the communication destination.

Specify the protocol and own station port number for each setting to be the same as those set in the simple CPU communication settings.

| Communication destination | Necessary setting |
|---|---|
| MELSEC iQ-R (Built-in Ethernet)*1 | Since the MELSOFT connection is used for communications, no setting is required. |
| MELSEC-Q (Built-in Ethernet)<br>MELSEC-L (Built-in Ethernet) | Set the following items (for the number of devices) in the "Open Setting" window.<br>• Protocol: "UDP"<br>• Open System: "MELSOFT Connection" |
| MELSEC iQ-F (Built-in Ethernet)*2 | Since the MELSOFT connection is used for communications, no setting is required.*5 |
| MELSEC iQ-L (Built-in Ethernet)<br>MELSEC iQ-R (Ethernet Module)*1 | Since the MELSOFT connection is used for communications, no setting is required. |
| MELSEC-Q (Ethernet Module)<br>MELSEC-L (Ethernet Module) | Set the network number and station number. |
| MELSEC-A/AnS (Ethernet Module) | ■Programming tool<br>Create an initial processing program, and set the following items.<br>• IP address and port number<br>• Communication method (UDP/TCP)<br>• Bit 15 and Bits 0 to 7 of Area for communication instruction during stop (Un\G496) (Turn on the bits corresponding to the connection numbers to be used.)*3<br>■DIP switch on the front of the module<br>Set the following items.<br>• SW2 (data code setting): Off<br>• SW7 (CPU communication timing setting): On*4 |
| MELSEC-FX3 (Ethernet Block/Adapter) | Set "Communication Data Code" of "Ethernet Port" to "Binary Code".<br>Set the following items (for the number of devices) in the "Open Setting" window.<br>• Protocol: "UDP or "TCP"<br>• Open system: "MC protocol"<br>• IP address, port number: Settings made for the own station (RJ71EN71 or RnENCPU) |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | Configure the settings so that the SLMP (QnA-compatible 3E frame of the MC protocol) can be used on the SLMP-compatible device side. (📖 Manual for the device used)<br>Set the communication data code to binary. |
| OMRON (CS/CJ series)<br>OMRON (NJ/NX series)<br>KEYENCE (KV series)<br>Panasonic (FP2SH series)<br>Panasonic (FP7 series)<br>YASKAWA MP3000 series<br>YASKAWA MP2000 series<br>Yokogawa FA-M3 series | Refer to the manual for the programmable controller of other manufacturer to be used. |
| MODBUS/TCP-compatible device | Set the MODBUS/TCP as a slave station. (📖 Manual for the device used) |
| SIEMENS S7 series<br>SIEMENS S7 series (extension)<br>Fuji Electric MICREX-SX series<br>JTEKT TOYOPUC series | Refer to the manual for the programmable controller of other manufacturer to be used. |

*1 Up to 64 devices can be connected to each port of the CPU module (built-in Ethernet port part) or the Ethernet module.

*2 Up to nine devices can be connected.

*3 Set this item to execute this function while the communication destination CPU module is in the STOP state or stop error state.

*4 Turn on this switch to execute this function while the communication destination CPU module is in the RUN state.

*5 For either of the cases below, add as many MELSOFT connections (TCP) as the number of the RJ71EN71s to the external device configuration setting.
When communicating with MELSOFT connection modules using TCP
When communicating from multiple RJ71EN71s to the MELSEC iQ-F (built-in Ethernet) using simple CPU communication

## Target PLC No.

When the communication destination is one of the following modules and a multiple CPU system is used, specify the "Target PLC No." of the communication destination.

| Communication destination | Target PLC No. |
|---|---|
| MELSEC iQ-R (built-in Ethernet): RnCPU, RnENCPU, RnPCPU, RnSFCPU | PLC No.1 to 4 |
| MELSEC-Q (built-in Ethernet): QnUDE(H)CPU, QnUDVCPU, QnUDPVCPU (📖 QCPU User's Manual (Multiple CPU System)) | PLC No.1 to 4 |
| MELSEC iQ-R (Ethernet Module): RJ71EN71, RnENCPU | PLC No.1 to 4 |
| MELSEC-Q (Ethernet): QJ71E71-100, QJ71E71 (📖 QCPU User's Manual (Multiple CPU System)) | PLC No.1 to 4 |
| Yokogawa FA-M3 series*1 | PLC No.1 to 4 |
| Fuji Electric MICREX-SX series | PLC No.1 to 8 |

*1 When the communication destination is Yokogawa FA-M3 series and "Not Specified" is set to "Target PLC No.", the CPU No.1 is automatically regarded as the target. (When the firmware version of the RJ71EN71 and the RnENCPU (network part) is "42" or later)

## Devices that can be specified

The devices that can be specified as transmission source and transmission destination vary depending on the communication destination. Bit device and word device can be set together for each setting number.
For devices that can be specified, the size of a device is specified in units of 16 points for a bit device and one point for a word device. Use 0 or multiples of 16 to specify the device number of a bit device.

**Point**

- Set the number of device points to the total number of setting No.1 to 512 (524288 words maximum).
- The maximum number of points per setting is 1024 words (bit device 8192 points maximum, word device 512 points maximum).
- Set the devices within the device range specified with the parameter of the CPU module.

### ■Devices that can be specified on the own station

The maximum number of points when the NZ2MC-16MBS is connected to the R120CPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 161882111 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to 9A61FFFH | — |
| | TS | 0 to 8993439 | This device can be specified only when "Write" is set to "Communication Pattern". |
| | TC | 0 to 8993439 | This device can be specified only when "Write" is set to "Communication Pattern". |
| | STS | 0 to 8993439 | This device can be specified only when "Write" is set to "Communication Pattern". |
| | STC | 0 to 8993439 | This device can be specified only when "Write" is set to "Communication Pattern". |
| | CS | 0 to 8993439 | This device can be specified only when "Write" is set to "Communication Pattern". |
| | CC | 0 to 8993439 | This device can be specified only when "Write" is set to "Communication Pattern". |
| | SB | 0H to 9A61FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 10117631 | A local device cannot be specified. |
| | W | 0H to 9A61FFH | — |
| | TN | 0 to 8993439 | — |
| | STN | 0 to 8993439 | — |
| | CN | 0 to 8993439 | — |
| | SW | 0H to 9A61FFH | — |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10027007 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |
| | G | PORT1: 722500 to 1246787<br>PORT2: 2722500 to 3246787 | Buffer memory for the RJ71EN71 and RnENCPU (network part).<br>A G device is expressed as Un\G. The search for the device requires the search string to be prefixed by Un\G. When the search for a G device is attempted while the RJ71EN71 and RnENCPU (network part) are not mounted, U0\G is prefixed for the search. (An error for unmounted module is not detected.) |

### ■When the communication destination is the MELSEC iQ-R (built-in Ethernet) or MELSEC iQ-R (Ethernet module)

The maximum number of points when the NZ2MC-16MBS is connected to the R120CPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 161882111 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to 9A61FFFH | — |
| | TS | 0 to 8993439 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | TC | 0 to 8993439 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STS | 0 to 8993439 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STC | 0 to 8993439 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CS | 0 to 8993439 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CC | 0 to 8993439 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | SB | 0H to 9A61FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 10117631 | A local device cannot be specified. |
| | W | 0H to 9A61FFH | — |
| | TN | 0 to 8993439 | — |
| | STN | 0 to 8993439 | — |
| | CN | 0 to 8993439 | — |
| | SW | 0H to 9A61FFH | — |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10027007 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |

### ■When the communication destination is the MELSEC-Q (built-in Ethernet) or MELSEC-Q (Ethernet module)

The maximum number of points when the Q4MCA-8MBS is connected to the Q26UDVCPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 1FFFH | — |
| | Y | 0H to 1FFFH | — |
| | M | 0 to 61439 | — |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to EFFFH | — |
| | TS | 0 to 32767 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | TC | 0 to 32767 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STS | 0 to 32767 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STC | 0 to 32767 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CS | 0 to 32767 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CC | 0 to 32767 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | SB | 0H to 7FFFH | — |
| | SM | 0 to 2047 | — |
| Word device | D | 0 to 4910079 | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | W | 0H to 4AEBFFH | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | TN | 0 to 32767 | — |
| | STN | 0 to 32767 | — |
| | CN | 0 to 32767 | — |
| | SW | 0H to 7FFFH | — |
| | SD | 0 to 2047 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 4849663 | Data is read or written following the file register settings of the communication destination. |

### ■When the communication destination is the MELSEC-L (built-in Ethernet) or MELSEC-L (Ethernet module)

The following table lists the maximum number of points for the L26CPU. The number of points varies depending on the CPU module used.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 1FFFH | — |
| | Y | 0H to 1FFFH | — |
| | M | 0 to 61439 | — |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to EFFFH | — |
| | TS | 0 to 25471 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | TC | 0 to 25471 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STS | 0 to 25471 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STC | 0 to 25471 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CS | 0 to 25471 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CC | 0 to 25471 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | SB | 0H to 7FFFH | — |
| | SM | 0 to 2047 | — |
| Word device | D | 0 to 421887 | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | W | 0H to 66FFFH | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | TN | 0 to 25471 | — |
| | STN | 0 to 25471 | — |
| | CN | 0 to 25471 | — |
| | SW | 0H to 6FFFH | — |
| | SD | 0 to 2047 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 393215 | Data is read or written following the file register settings of the communication destination. |

## ■When the communication destination is the MELSEC iQ-F (built-in Ethernet)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0 to 1777 | Octal notation is used. |
| | Y | 0 to 1777 | Octal notation is used. |
| | M | 0 to 32767 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to 7FFFH | — |
| | TS | 0 to 1023 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | TC | 0 to 1023 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STS | 0 to 1023 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STC | 0 to 1023 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CS | 0 to 1023 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CC | 0 to 1023 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | SB | 0H to 7FFFH | — |
| | SM | 0 to 9999 | — |
| Word device | D | 0 to 7999 | A local device cannot be specified. |
| | W | 0H to 7FFFH | — |
| | TN | 0 to 1023 | — |
| | STN | 0 to 1023 | — |
| | CN | 0 to 1023 | — |
| | SW | 0H to 7FFFH | — |
| | SD | 0 to 11999 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |

## ■When the communication destination is the MELSEC iQ-L (built-in Ethernet)

The maximum number of points when the L16HCPU is connected. The number of points varies depending on the CPU module used.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 14065663 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to D69FFFH | — |
| | TS | 0 to 781407 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | TC | 0 to 781407 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STS | 0 to 781407 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | STC | 0 to 781407 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CS | 0 to 781407 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | CC | 0 to 781407 | This device can be specified only when "Read" is set to "Communication Pattern". |
| | SB | 0H to D69FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 879103 | A local device cannot be specified. |
| | W | 0H to D69FFH | — |
| | TN | 0 to 781407 | — |
| | STN | 0 to 781407 | — |
| | CN | 0 to 781407 | — |
| | SW | 0H to D69FFH | — |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 819199 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |

### ■When the communication destination is the MELSEC A/AnS (Ethernet module)

The maximum number of points for the A4UCPU. The number of points varies depending on the CPU module used. (📖 MELSEC Communication Protocol Reference Manual)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 7FFH | — |
| | Y | 0H to 7FFH | — |
| | M | 0 to 8191, 9000 to 9255 | To specify 9000 to 9255 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used.<br>To specify 9000 to 9255, use 9000 + a multiple of 16. |
| | F | 0 to 2047 | — |
| | B | 0H to FFFH | — |
| | TS | 0 to 2047 | — |
| | TC | 0 to 2047 | — |
| | CS | 0 to 1023 | — |
| | CC | 0 to 1023 | — |
| Word device | D | 0 to 6143, 9000 to 9255 | To specify 9000 to 9255 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used. |
| | W | 0H to FFFH | — |
| | TN | 0 to 2047 | — |
| | CN | 0 to 1023 | — |
| | R | 0 to 8191 | — |
| | Rn | 10000 to 488191 | Upper two digits are 1 to 48 (block No.), and lower four digits are 0 to 8191.<br>The range of 8192 to 9999 must not be specified.<br>The usable range varies depending on the CPU module used and parameter settings. (📖 User's manual of the CPU module used) |

### ■When the communication destination is the MELSEC-FX3 (Ethernet block/adapter)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0 to 377 | Octal notation is used. |
| | Y | 0 to 377 | Octal notation is used. |
| | M | 0 to 7679, 8000 to 8511 | To specify 8000 to 8511 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used. |
| | S | 0 to 4095 | The ranges where write is limited with an allowable number of times and the ranges where write is automatically limited are included. (📖 FX3S/FX3G/FX3GC/FX3U/FX3UC SERIES PROGRAMMING MANUAL - Basic & Applied Instructions Edition) |
| | TS | 0 to 511 | — |
| | CS | 0 to 191 | — |
| Word device | D | 0 to 8511 | • To specify 8000 to 8511 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used.<br>• The ranges where write is limited with an allowable number of times are included. (📖 FX3S/FX3G/FX3GC/FX3U/FX3UC SERIES PROGRAMMING MANUAL - Basic & Applied Instructions Edition) |
| | TN | 0 to 511 | The ranges where write is limited with an allowable number of times are included. (📖 FX3S/FX3G/FX3GC/FX3U/FX3UC SERIES PROGRAMMING MANUAL - Basic & Applied Instructions Edition) |
| | CN | 0 to 199 | The ranges where write is limited with an allowable number of times are included. (📖 FX3S/FX3G/FX3GC/FX3U/FX3UC SERIES PROGRAMMING MANUAL - Basic & Applied Instructions Edition) |
| | R | 0 to 32767 | — |

## ■When the communication destination is an SLMP-compatible device (QnA-compatible 3E frame)

Applicable devices differ depending on the device. For applicable devices, refer to the manual for the device used.
The device types that can be set are X, Y, M, L, F, B, TS*1, TC*1, STS*1, STC*1, CS*1, CC*1, SB, SM, D, W, TN, STN, CN, SW, SD, R, and ZR. The accessible device range of the communication destination is the range which can be specified with the subcommand (0000) of the SLMP (QnA-compatible 3E frame of the MC protocol).

*1 This device can be specified only when "Read" is set to "Communication Pattern".

**Point**

The following SLMP commands are sent to the external device.

• When "Read" is set to "Communication Pattern"

| Subheader | Request destination network No. | Request destination station No. | Request destination module I/O No. | Request destination multidrop station No. | Request data length | Monitoring timer | Command | Subcommand | Device code | Start device No. | Number of device points |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 50H 00H | 00H | FFH | FFH 03H | 00H | 0CH 00H | 00H 00H | 01H 04H | 00H 00H | | | |

• When "Write" is set to "Communication Pattern"

| Subheader | Request destination network No. | Request destination station No. | Request destination module I/O No. | Request destination multidrop station No. | Request data length | Monitoring timer | Command | Subcommand | Device code | Start device No. | Number of device points | Write data |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 50H 00H | 00H | FFH | FFH 03H | 00H | | 00H 00H | 01H 14H | 00H 00H | | | | |

## ■When the communication destination is OMRON (CS/CJ series)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | .. | 0 to 6143 | Channel I/O |
| | AR | 0 to 447 | The setting is available only when "Read" is set to "Communication Pattern". |
| | AR | 448 to 959 | — |
| | HR | 0 to 511 | — |
| | WR | 0 to 511 | — |
| Word device | DM | 0 to 32767 | — |
| | TIM | 0 to 4095 | — |
| | CNT | 0 to 4095 | — |
| | DR | 0 to 15 | — |
| | TK | 0 to 31 | The setting is available only when "Read" is set to "Communication Pattern". |
| | EM | 0 to 32767 | — |
| | EMn_ | 0 to 32767 | "n" represents 0H to FH or 10H to 18H. (25 devices in total) |

**Point**

- In the IP address table on the communication destination side, set the IP address and node address to be assigned to the RJ71EN71 or RnENCPU (network part). Set the node address to the value specified in Option. If the node address cannot be specified in Option because of restrictions (for example, module version limitations), set the node address to 1.
- When using the RJ71EN71 in a redundant system, set the control system IP address in "Redundant System Settings" under "Application Settings" on the own station. Then, set the control system IP address for the IP address to be assigned to the RJ71EN71 or RnENCPU (network part) in the IP address table of the item on the communication destination.

### ■When the communication destination is OMRON (NJ/NX series)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | .. | 0 to 6143 | Channel I/O |
| | HR | 0 to 511 | — |
| | WR | 0 to 511 | — |
| Word device | DM | 0 to 32767 | — |
| | EMn_ | 0 to 32767 | "n" represents 0H to FH or 10H to 18H. (25 devices in total) |

When communicating with the NJ/NX series, only the area provided as memory for the CJ module can be accessed.
The following modules can use the memory for the CJ module.

- NJ series CPU module
- NX series NX502 CPU module
- NX series NX102 CPU module
- NX series NX1P2 CPU module
- NX series NX701-☐☐20

When communicating with the NX series, use the memory settings for CJ series modules in Sysmac Studio to enable the area and range to be used. (Settings are not required for the NJ series.)
When changing the variable definition in the enabled area, assign each device as follows. (☐ represents the address.)

| Type | Device | Assignment destination |
|---|---|---|
| Bit device | CIO☐ | %☐ |
| | WR☐ | %W☐ |
| | HR☐ | %H☐ |
| Word device | DM☐ | %D☐ |
| | EMn☐ | %En☐ (n = 0H to 18H) |

Reading from or writing to any of the following areas may cause an error or unintended operation.

- Area out of range for each device
- Area whose memory is not enabled

**Point**

- In the IP address table on the communication destination side, set the IP address and node address to be assigned to the RJ71EN71 or RnENCPU (network part). Set the node address to the value specified in Option. If the node address cannot be specified in Option because of restrictions (for example, module version limitations), set the node address to 1.
- When using the RJ71EN71 in a redundant system, set the control system IP address in "Redundant System Settings" under "Application Settings" on the own station. Then, set the control system IP address for the IP address to be assigned to the RJ71EN71 or RnENCPU (network part) in the IP address table of the item on the communication destination.

### ■When the communication destination is KEYENCE (KV series)

The format of devices other than B device consists of the last two digits of bit specification and the upper digits of word specification.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of bits. One point corresponds to one bit.) | R | 0 to 199915 | Specify 00 in the last two digits for the start. |
| | B | 0H to 7FFFH | Specify 0 or a multiple of 16 for the start. |
| | MR | 0 to 399915 | Specify 00 in the last two digits for the start. |
| | LR | 0 to 99915 | Specify 00 in the last two digits for the start. |
| | CR | 0 to 7915 | Specify 00 in the last two digits for the start. |
| | T | 0 to 3999 | — |
| | C | 0 to 3999 | — |
| Word device | CM | 0 to 5999 | — |
| | DM | 0 to 65534 | — |
| | EM | 0 to 65534 | — |
| | FM | 0 to 32767 | — |
| | ZF | 0 to 524287 | — |
| | W | 0H to 7FFFH | — |
| | T | 0 to 3999 | — |
| | C | 0 to 3999 | — |

### ■When the communication destination is Panasonic (FP2SH series)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | WX | 0 to 511 | The setting is available only when "Read" is set to "Communication Pattern". |
| | WY | 0 to 511 | — |
| | WR | 0 to 886 | — |
| | WL | 0 to 639 | — |
| Word device | LD | 0 to 8447 | — |
| | DT | 0 to 10239 | — |
| | FL | 0 to 32764 | — |

**Point**

Setting item of communication destination: Set the MEWTOCOL station number. At this time, set 1 to the external device side (FP2) and 2 to the own station side.

## ■When the communication destination is Panasonic (FP7 series)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | WX | 0 to 511 | — |
| | WY | 0 to 511 | — |
| | WR | 0 to 2047 | — |
| | WL | 0 to 1023 | — |
| | _WX | 001000000 to 468000511 | The upper three digits of the device number show the program block number, and the lower six digits show the device number. |
| | _WY | 001000000 to 468000511 | The upper three digits of the device number show the program block number, and the lower six digits show the device number. |
| | _WR | 001000000 to 468002047 | The upper three digits of the device number show the program block number, and the lower six digits show the device number. |
| | _WL | 001000000 to 468001023 | The upper three digits of the device number show the program block number, and the lower six digits show the device number. |
| Word device | LD | 0 to 16383 | — |
| | DT | 0 to 999423 | — |
| | _LD | 001000000 to 468016383 | The upper three digits of the device number show the program block number, and the lower six digits show the device number. |
| | _DT | 001000000 to 468065534 | The upper three digits of the device number show the program block number, and the lower six digits show the device number. |

## ■When the communication destination is YASKAWA MP3000 series

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | IB | 0H to 21FFFH | The setting is available only when "Read" is set to "Communication Pattern". |
| | OB | 0H to 21FFFH | — |
| | MB | 0 to 1048575 | — |
| | GB | 0 to 2097151 | — |
| | SB | 0 to 65534 | — |
| Word device | IW | 0H to 21FFFH | The setting is available only when "Read" is set to "Communication Pattern". |
| | OW | 0H to 21FFFH | — |
| | MW | 0 to 1048575 | — |
| | GW | 0 to 2097151 | — |
| | SW | 0 to 65534 | — |

## ■When the communication destination is YASKAWA MP2000 series

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | IB | 0H to FFFH | The setting is available only when "Read" is set to "Communication Pattern". |
| | MB | 0 to 4095 | — |
| Word device | IW | 0H to 7FFFH | The setting is available only when "Read" is set to "Communication Pattern". |
| | MW | 0 to 65534 | — |

### ■When the communication destination is Yokogawa FA-M3 series

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | X | 201 to 71664 | The setting is available only when "Read" is set to "Communication Pattern". The setting range of the last two digits is 01 to 64. (Only 01, 17, 33, or 49 can be set as the start device. Only 16, 32, 48, or 64 can be set as the end device.) |
| | Y | 201 to 71664 | The setting range of the last two digits is 01 to 64. (Only 01, 17, 33, or 49 can be set as the start device. Only 16, 32, 48, or 64 can be set as the end device.) |
| | I | 1 to 165535 | Use 1 or a multiple of 16 +1 to set the start device. |
| | E | 1 to 4096 | |
| | L | 1 to 78192 | |
| | M | 1 to 9984 | |
| Word device | D | 1 to 65535 | — |
| | R | 1 to 4096 | — |
| | V | 1 to 256 | — |
| | B | 1 to 262144 | — |
| | W | 1 to 78192 | — |
| | Z | 1 to 1024 | — |
| | TP | 1 to 3072 | — |
| | CP | 1 to 3072 | — |
| | TS | 1 to 3072 | — |
| | CS | 1 to 3072 | — |
| | TI | 1 to 3072 | — |
| | CI | 1 to 3072 | — |

### ■When the communication destination is a MODBUS/TCP-compatible device

Applicable devices and the number of writable/readable points differ depending on the device. For applicable devices and the number of writable/readable points, refer to the manual for the device used.

The device types that can be set are Coil, Input, Input Register, and Holding Register.

| Type | Applicable device | | |
|---|---|---|---|
| | Maximum number of write points | Maximum number of read points | Remarks |
| Coil | 1968 | 2000 | — |
| Input | — | 2000 | Setting is possible only when the "Communication Pattern" is "Read". |
| Input Register | — | 125 | Setting is possible only when the "Communication Pattern" is "Read". |
| Holding Register | 123 | 125 | — |

**Point**

Specify the value obtained by subtracting 1 from the MODBUS device number.

The simple CPU communication uses the function codes shown below. Communicate with the supporting MODBUS/TCP-compatible devices.

| Function code | Function name | Corresponding setting on the simple CPU communication | |
|---|---|---|---|
| | | "Communication pattern" | Settable data |
| 01H | Read coil | "Read" | Bit device |
| 02H | Read input | "Read" | Bit device |
| 03H | Read holding register | "Read" | Word device |
| 04H | Read input register | "Read" | Word device |
| 0FH | Write multiple coils | "Write" | Bit device |
| 10H | Write multiple register | "Write" | Word device |

### ■When the communication destination is SIEMENS S7 series

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one byte.) | I | 0 to 255 | Start digit: An even number must be specified.<br>Last digit: An odd number must be specified. |
| | Q | 0 to 255 | |
| | M | 0 to 255 | |
| Word device | DB | 001000 to 255511 | Upper 3 digits: Block No., Lower 3 digits: Address range<br>An even number must be specified to the start of the lower 3 digits and an odd number must be specified to the last of the lower 3 digits. (The size for 1 point is 1 byte. Two points are required to set for word device.) |

For the settings for communications with the SIEMENS S7 series, set as follows.

Ex.

When reading one word of the SIEMENS S7 series devices starting from I100, specify I100 for the start address and I101 for the last address in the simple CPU communication.

For details, refer to the manual for the external device.

**Point**

Access is executed in units of words. The upper byte and lower byte of reading/writing value may be interchanged depending on the setting in the communication destination.

### ■When the communication destination is SIEMENS S7 series (extension)

The accessible devices of SIEMENS S7 series (extension) are the same as those of SIEMENS S7 series. Moreover, the accessible ranges of the devices are widened when the communication destination is SIEMENS S7 series (extension).

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one byte.) | I | 0 to 65535 | Start digit: An even number must be specified.<br>Last digit: An odd number must be specified. |
| | Q | 0 to 65535 | |
| | M | 0 to 65535 | |
| Word device | DB | 1.0 to 65535.65535 | Specify the range in the following format: <block No.>.<address range>. (Separate block No. from address range by placing a dot (.) between them.)<br>For the start of address range, specify an even number. For the last of address range, specify an odd number. (Because the size for 1 point is 1 byte, two points must be set for word device.)<br>Match the first and last block numbers. |

For the settings for communications with the SIEMENS S7 series (extension), set as follows.

Ex.

When reading one word of the SIEMENS S7 series (extension) devices starting from I100, specify I100 for the start address and I101 for the last address in the simple CPU communication.

For details, refer to the manual for the external device.

## ■When the communication destination is Fuji Electric MICREX-SX series

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | MX1 | 0000000 to 1703935 | — |
| | MX3 | 000000 to 262143 | — |
| | MX10 | 0000 to 1023 | — |
| Word device | MW1 | 0000000 to 1703935 | — |
| | MW3 | 000000 to 262143 | — |
| | MW10 | 0000 to 1023 | — |

## ■When the communication destination is JTEKT TOYOPUC series

Specify the program number (1 to 3) using the first digit for the standard area (X, Y, M, K, M, V, D, R, S, and N).

The program number does not required to be set for the extended area (EX, EY, EM, EK, EL, EV, GX, GY, GM, U, ES, EN, H, EB, and FR).

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device (The device is displayed in units of words. One point corresponds to one word.) | X | 10000H to 3007FH | Upper 1 digit: Program No., Lower 4 digits: Address range<br>The following lists the communicable address range of lower 4 digits.<br>• X/Y (input/output): 0000 to 007FH<br>• M (internal relay): 0000 to 007FH, 0100 to 017FH<br>• K (keep relay): 0000 to 002FH<br>• L (link relay): 0000 to 007FH, 0100 to 02FFH<br>• V (special relay): 0000 to 000FH, 0100 to 017FH |
| | Y | 10000H to 3007FH | |
| | M | 10000H to 3017FH | |
| | K | 10000H to 3002FH | |
| | L | 10000H to 302FFH | |
| | V | 10000H to 3017FH | |
| | EX | 00000H to 0007FH | — |
| | EY | 00000H to 0007FH | — |
| | EM | 00000H to 001FFH | — |
| | EK | 00000H to 000FFH | — |
| | EL | 00000H to 001FFH | — |
| | EV | 00000H to 000FFH | — |
| | GX | 00000H to 00FFFH | — |
| | GY | 00000H to 00FFFH | — |
| | GM | 00000H to 00FFFH | — |
| Word device | D | 10000H to 32FFFH | Upper 1 digit: Program No., Lower 4 digits: Address range<br>The following lists the communicable address range of lower 4 digits.<br>• D (data register): 0000 to 2FFFH<br>• R (link register): 0000 to 07FFH<br>• S (special register): 0000 to 03FFH, 1000 to 13FFH<br>• N (present value register): 0000 to 01FFH, 1000 to 17FFH |
| | R | 10000H to 307FFH | |
| | S | 10000H to 313FFH | |
| | N | 10000H to 317FFH | |
| | U | 00000H to 1FFFFH | — |
| | ES | 00000H to 007FFH | — |
| | EN | 00000H to 007FFH | — |
| | H | 00000H to 007FFH | — |
| | EB | 00000H to 3FFFFH | — |
| | FR | 00000H to 1FFFFH | — |

Note the following when configuring settings for communications with the JTEKT TOYOPUC series.

Ex.

When the JTEKT TOYOPUC series devices are P2-M01230 (bit device specification), they are represented as M20123 in the simple CPU communication.

For details, refer to the manual for the external device.

## Operation during the simple CPU communication

### ■Operation of the own station

Since the own station operates without depending on the operating status of the CPU module, the simple CPU communication is performed even when the CPU module is in the STOP state because of stop error. When an initial communication error occurs, communications of the corresponding setting numbers are stopped and ones of the setting numbers, where no error occurs will operate.

If the communication destination sends a response, data is transferred to the system area for the simple CPU communication. Data inconsistency will not occur in one setting since processing is executed for each setting.

### ■Operation of the communication destination

The operation differs depending on the communication destination.

| Communication destination | Data operation in one setting |
|---|---|
| MELSEC iQ-R (Built-in Ethernet)<br>MELSEC-Q (Built-in Ethernet)<br>MELSEC-L (Built-in Ethernet) | The data specified with a bit device or a word device in one setting is communicated at the same timing. However, if both a bit device and a word device are set for the same setting number, the data of each device may be communicated at different timing.<br>For the following cases, the data is received at different timing.<br>• The total of bit device points and word device points exceeds 960 words.<br>• TS, TC, STS, STC, CS, or CC is specified as a bit device. |
| MELSEC iQ-F (Built-in Ethernet) | The data specified with a bit device or a word device in one setting is communicated at the same timing. However, if both a bit device and a word device are set for the same setting number, the data of each device may be communicated at different timing. |
| MELSEC iQ-L (Built-in Ethernet)<br>MELSEC iQ-R (Ethernet Module) | The data specified with a bit device or a word device in one setting is communicated at the same timing. However, if both a bit device and a word device are set for the same setting number, the data of each device may be communicated at different timing.<br>For the following cases, the data is received at different timing.<br>• The total of bit device points and word device points exceeds 960 words.<br>• TS, TC, STS, STC, CS, or CC is specified as a bit device. |
| MELSEC-Q (Ethernet Module)<br>MELSEC-L (Ethernet Module) | The data specified with a bit device or a word device in one setting is communicated at the same timing. However, if both a bit device and a word device are set for the same setting number, the data of each device may be communicated at different timing. |
| MELSEC-A/AnS (Ethernet Module) | When the data size in one setting is as follows, data inconsistency will not occur.<br>• Read: Bit device 2048 points or less, word device 256 points or less<br>• Write: Bit device 640 points or less, word device 256 points or less |
| MELSEC-FX3 (Ethernet Block/Adapter) | When the data size in one setting is as follows, data inconsistency will not occur.<br>• Read: Bit device 512 points or less, word device 64 points or less<br>• Write: Bit device 160 points or less, word device 64 points or less |
| SLMP Corresponding Device (QnA Compatible 3E Frame)<br>OMRON (CS/CJ series)<br>OMRON (NJ/NX series)<br>KEYENCE (KV series)<br>Panasonic (FP2SH series)<br>Panasonic (FP7 series)<br>YASKAWA MP3000 series<br>YASKAWA MP2000 series<br>Yokogawa FA-M3 series<br>MODBUS/TCP-compatible device<br>SIEMENS S7 series<br>SIEMENS S7 series (extension)<br>Fuji Electric MICREX-SX series | Depends on the communication destination. |

## Operations when the communication setting is "On Request"

### ■How to start communications

Power off and on the system or reset the CPU module first. Turn off and on the buffer memory area of Request to start communication at request, and communications start.

**Point**

Once the CPU module is reset, the system starts the preparation processing. The preparation processing checks the parameter settings (own station port numbers). If an error occurs, data will not be communicated. In this case, the buffer memory area of Simple CPU communication status stores AH (Communications impossible).

### ■Each status after communications start

The following table lists the buffer memory areas that stores each status of communications from its start to completion.

| Status | Item | Address | Buffer memory status |
|---|---|---|---|
| During the latency time*1 | Simple CPU communication status for each setting number | Un\G721936 to Un\G721999<br>Un\G1247460 to Un\G1247907 | 1H: Preparing |
| After a lapse of the latency time | Preparation completion for each setting number | Un\G721912 to Un\G721915<br>Un\G1247412 to Un\G1247439 | On: Ready |
| | Simple CPU communication status for each setting number | Un\G721936 to Un\G721999<br>Un\G1247460 to Un\G1247907 | 2H: Waiting for the request |
| Request to start communication at request issued | Simple CPU communication status for each setting number | Un\G721936 to Un\G721999<br>Un\G1247460 to Un\G1247907 | 3H: Communicating |
| | Execution status flag for each setting number | Un\G721908 to Un\G721911<br>Un\G1247384 to Un\G1247411 | On: Communicating |
| Data communication completed*2 | Execution status flag for each setting number | Un\G721908 to Un\G721911<br>Un\G1247384 to Un\G1247411 | Off: Communication stop |

*1 During the latency time, another start request (turning off and on the buffer memory area of Request to start communication at request) is ignored.

*2 After the data communications has been completed, 2H (Waiting for the request) is stored in the buffer memory area of Simple CPU communication status by turning on and off the buffer memory area of Request to start communication at request, and one session completes.

**Point**

Once a start request of On Request communications is detected, another start request (turning off and on the buffer memory area of Request to start communication at request) from a program is ignored until the buffer memory area of Execution status flag turns off (Communication stop). (This symptom is not treated as an error.)

### ■Behavior on errors

After data is sent to the communication destination, if no response is received from the communication destination or an error occurs (or an error response is received), the system makes retries until the timeout time.

During retries, the buffer memory area of Simple CPU communication status stores 5H (Retry being executed).

The system retries the operation for the number of times specified in the parameter plus one additional time.

### ■Behavior when an abnormal state is detected

If an error response is received after the retries, an abnormal state is detected and the buffer memory area of Simple CPU communication error code stores error codes.

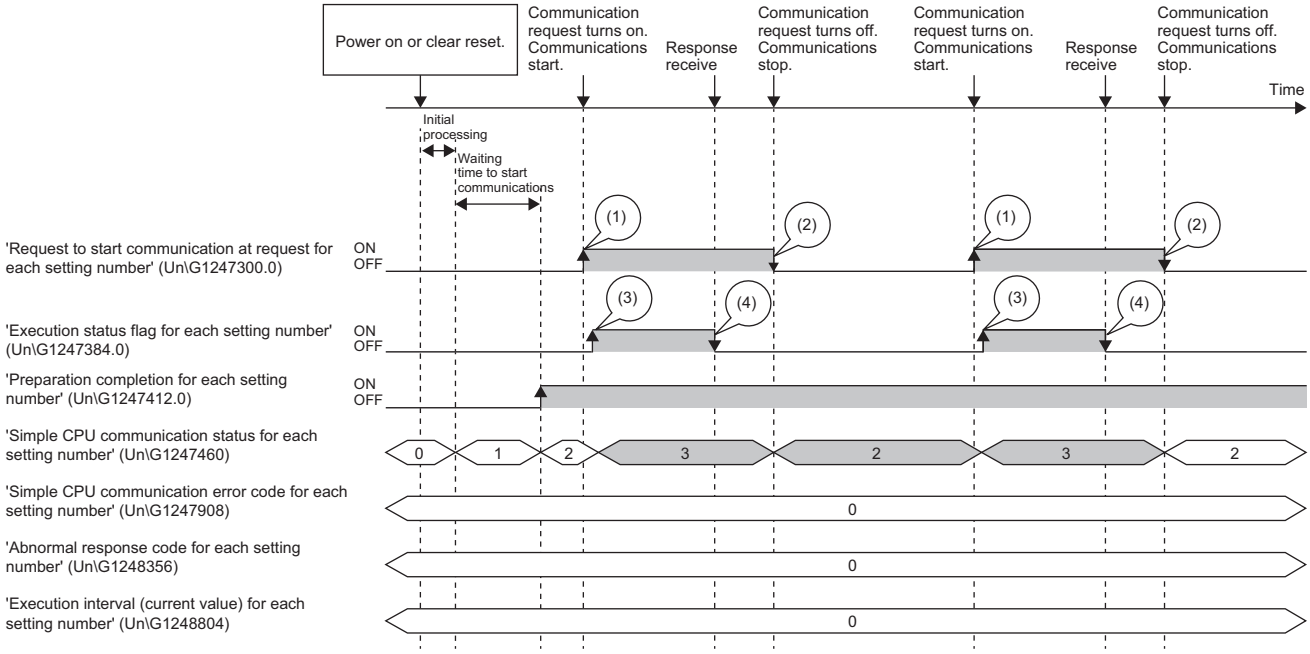
In an abnormal state, the buffer memory area of Execution status flag turns off (Communication stop). By programmatically turning on and off the buffer memory area of Request to start communication at request, 2H (Waiting for the request) is stored in the buffer memory area of Simple CPU communication status.

**Point**

- While the communications of one setting number is in an abnormal state, the value in the buffer memory areas of Simple CPU communication status for other setting numbers, in which the same communication destination is set, remains unchanged (2H (Waiting for the request)).
- If no response is received from the communication destination, the buffer memory area of Simple CPU communication error code stores error codes.
- If an error response is received from the communication destination, the buffer memory area of Simple CPU communication error code stores error codes and the buffer memory area of Abnormal response code stores abnormal response codes. (The abnormal response code here indicates any codes received from the destination device.)

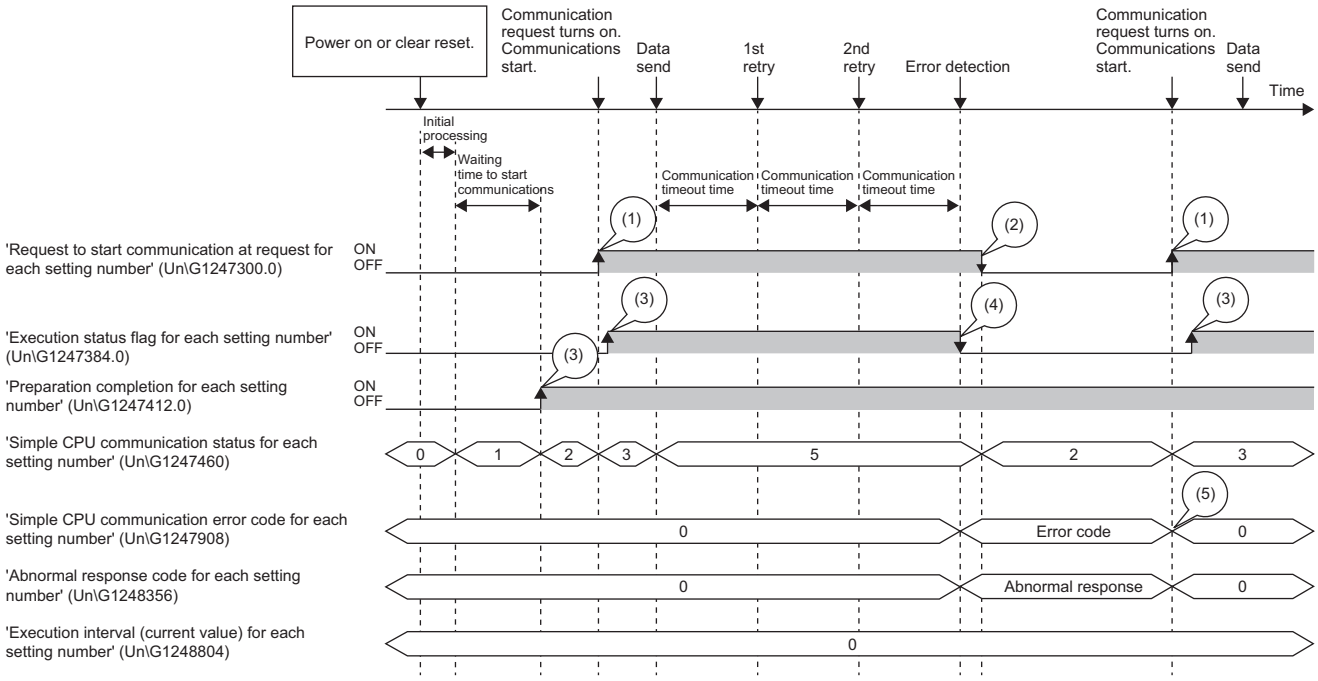
## ■Timing chart

• Operation of on-request communications of setting No.65 (Normal)




(1) The buffer memory area is turned on by the program.
(2) The buffer memory area is turned off by the program.
(3) The buffer memory area is turned on by the system.
(4) The buffer memory area is turned off by the system.

• Operation of on-request communications of setting No.65 (Abnormal state)




(1) The buffer memory area is turned on by the program.
(2) The buffer memory area is turned off by the program.
(3) The buffer memory area is turned on by the system.
(4) The buffer memory area is turned off by the system.
(5) The value is cleared to 0 with a clear request from the engineering tool.

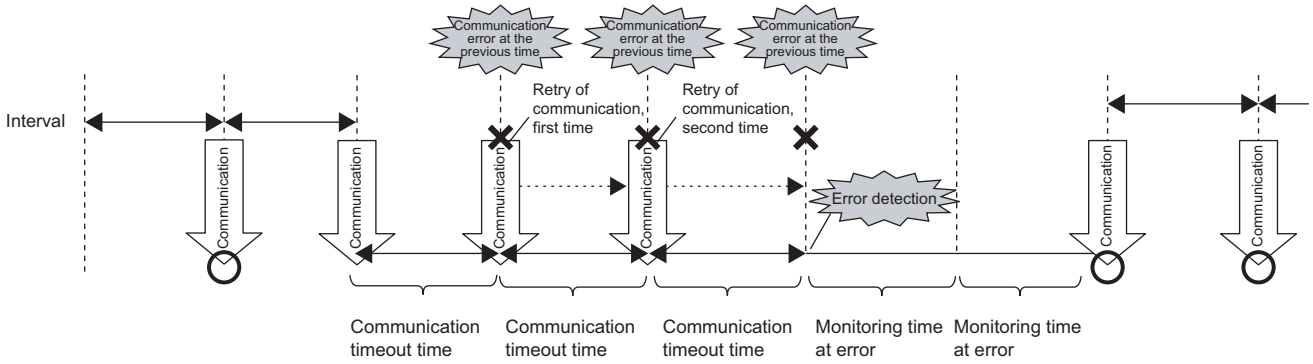
## Communication Time-out Period/Communication Retry Count/Monitoring Time At Error

When the communication destination does not respond or returns an error response, the communication will be retried (resent) after the time-out period has passed.

At this time, after communications are performed for the communication retry count + 1 times, the fixed interval communication is performed during the monitoring time at error. If the communication destination responds after these communications, the communications at the execution interval specified with the parameters are restarted.

Ex.

Error detection timing when 2 is set for the communication retry count




**Point**

The error is detected if the communication destination does not respond or communications fail by when the following time has passed: (communication retry count + 1) × communication time-out period

## Latency Time

By setting the latency time, the start timing in the communications can be switched in the following cases.

- To prevent an error due to the overlap of the start timing in the communications
- To start communications after the communication destination is ready for the communications

During the latency time, "Preparing" is indicated in the communication status.

**Point**

Communications at a specified execution interval (when "Fixed Interval" is set for the communication setting) is ignored during the latency time.

## Import/Export of Settings

Import/export the simple CPU communication setting parameters from/to a CSV file.

### ■Export

1. Click the [Export] button in the "Setting Item" window under "Simple CPU Communication Setting".
2. Input the file name and click the [Save] button in the "Export to File" window.
3. Edit the exported CSV file as necessary. The description displayed the "Setting Item" window under "Simple CPU Communication Setting" can be edited.

**Point**

The following describes the detailed format in the CSV file.

- The file format is Unicode (UTF-16 Little Endian with BOM).
- "\t" (tab) delimits between the items.
- Each item enclosed in double quotation marks (").
- If the item includes double quotation marks ("), double quotation marks (") in the items are displayed as two double quotation marks ("").
- A return is set at the end of the line. The return code is CR+LF.

### Precautions

In the CSV file, a string displayed in the "Setting Item" window under "Simple CPU Communication Setting" may be replaced with another string. The following table lists the items to be replaced with another string.

| Item | Engineering tool | CSV file |
|---|---|---|
| IP Address Input Format*1 | Decimal | DEC |
| | Hexadecimal | HEX |
| Device Assignment Method*2 | Points/Start | Points/Start |
| | Start/End | Start/End |
| Initial Communication Setting | —(Not supported) | Unsupported |
| Communication Pattern | Read | Read |
| | Write | Write |
| Communication Setting | On Request | 0 |
| | Fixed Interval | 1 |

| Item | Engineering tool | CSV file |
|---|---|---|
| Communication Destination Setting: Device Type | MELSEC iQ-R (Built-in Ethernet) | 16 |
| | MELSEC iQ-L (Built-in Ethernet) | 20 |
| | MELSEC iQ-F (Built-in Ethernet) | 19 |
| | MELSEC-Q (Built-in Ethernet) | 17 |
| | MELSEC-L (Built-in Ethernet) | 18 |
| | MELSEC iQ-R (Ethernet Module) | 21 |
| | MELSEC-Q (Ethernet Module) | 22 |
| | MELSEC-L (Ethernet Module) | 23 |
| | MELSEC-A/AnS (Ethernet Module) | 25 |
| | MELSEC-FX3 (Ethernet Block/Adapter) | 26 |
| | SLMP Corresponding Device (QnA Compatible 3E Frame) | 30 |
| | OMRON (CS/CJ series) | 32 |
| | OMRON (NJ/NX series) | 45 |
| | KEYENCE (KV series) | 33 |
| | Panasonic (FP2SH series) | 34 |
| | Panasonic (FP7 series) | 35 |
| | YASKAWA MP3000 series | 36 |
| | YASKAWA MP2000 series | 37 |
| | Yokogawa FA-M3 series | 38 |
| | MODBUS/TCP-compatible device | 39 |
| | SIEMENS S7 series | 40 |
| | SIEMENS S7 series (extension) | 44 |
| | Fuji Electric MICREX-SX series | 41 |
| | JTEKT TOYOPUC series | 42 |
| Target PLC No. | Not Specified | 0 |
| | PLC No.1 to 8 | 1 to 8 |

*1 The IP address is imported in positional notation of the IP address input format of the CSV file.
*2 The device range (number of points/start number/end number) of the transfer source device is set in the device assignment method of the CSV file.

### ■Import

1. Click the [Import] button in the "Setting Item" window under "Simple CPU Communication Setting".
2. Select the file name and click the [Save] button in the "Import to File" window.
3. Click the [OK] button.
4. The data in the CSV file is reflected in the "Setting Item" window under "Simple CPU Communication setting".

**Point**

For the file format at import, follow the following format.
- "\t" (tab) can be used as a delimiter between items.
- Enclose items including delimiters and return codes in double quotation marks.

## Precautions

- When imported, all the original settings are cleared. Export and save the settings as necessary.
- Do not insert or delete lines or change the value of the item to an invalid value in the CSV file. An error message is displayed and the settings cannot be imported.
- Import the CSV file whose data has been set from the 10th line onward.

# Checking the simple CPU communication status



The simple CPU communication status can be checked with the buffer memory or diagnostic functions.

## Checking with the buffer memory

The simple CPU communication status can be checked with the storage status of the corresponding setting number in the following buffer memory areas.

| Item | | Address | Remarks |
|---|---|---|---|
| Request to start communication at request | | Un\G721896 to Un\G721899<br>Un\G1247300 to Un\G1247327 | • Setting No.1: Un\G721896.0<br>⋮<br>• Setting No.64: Un\G721899.F<br>• Setting No.65: Un\G1247300.0<br>⋮<br>• Setting No.512: Un\G1247327.F |
| Request to stop fixed interval communication | | Un\G721900 to Un\G721903<br>Un\G1247328 to Un\G1247355 | • Setting No.1: Un\G721900.0<br>⋮<br>• Setting No.64: Un\G721903.F<br>• Setting No.65: Un\G1247328.0<br>⋮<br>• Setting No.512: Un\G1247355.F |
| Request to restart fixed interval communication | | Un\G721904 to Un\G721907<br>Un\G1247356 to Un\G1247383 | • Setting No.1: Un\G721904.0<br>⋮<br>• Setting No.64: Un\G721907.F<br>• Setting No.65: Un\G1247356.0<br>⋮<br>• Setting No.512: Un\G1247383.F |
| Execution status flag | | Un\G721908 to Un\G721911<br>Un\G1247384 to Un\G1247411 | • Setting No.1: Un\G721908.0<br>⋮<br>• Setting No.64: Un\G721911.F<br>• Setting No.65: Un\G1247384.0<br>⋮<br>• Setting No.512: Un\G1247411.F |
| Ready | | Un\G721912 to Un\G721915<br>Un\G1247412 to Un\G1247439 | • Setting No.1: Un\G721912.0<br>⋮<br>• Setting No.64: Un\G721915.F<br>• Setting No.65: Un\G1247412.0<br>⋮<br>• Setting No.512: Un\G1247439.F |
| Simple CPU communication status | 0H: Unset | Un\G721936 to Un\G721999<br>Un\G1247460 to Un\G1247907 | • Setting No.1: Un\G721936<br>⋮<br>• Setting No.64: Un\G721999<br>• Setting No.65: Un\G1247460<br>⋮<br>• Setting No.512: Un\G1247907 |
| | 1H: Preparing | | |
| | 2H: Waiting for the request | | |
| | 3H: Communicating | | |
| | 4H: Communication stop | | |
| | 5H: Retry being executed | | |
| | 6H: Monitoring at error | | |
| | AH: Communications impossible | | |
| Simple CPU communication error code | | Un\G722000 to Un\G722063<br>Un\G1247908 to Un\G1248355 | • Setting No.1: Un\G722000<br>⋮<br>• Setting No.64: Un\G722063<br>• Setting No.65: Un\G1247908<br>⋮<br>• Setting No.512: Un\G1248355 |
| Abnormal response code | | Un\G722064 to Un\G722127<br>Un\G1248356 to Un\G1248803 | • Setting No.1: Un\G722064<br>⋮<br>• Setting No.64: Un\G722127<br>• Setting No.65: Un\G1248356<br>⋮<br>• Setting No.512: Un\G1248803 |

| Item | Address | Remarks |
|---|---|---|
| Execution interval (current value) | Un\G722128 to Un\G722191<br>Un\G1248804 to Un\G1249251 | • Setting No.1: Un\G722128<br>⋮<br>• Setting No.64: Un\G722191<br>• Setting No.65: Un\G1248804<br>⋮<br>• Setting No.512: Un\G1249251 |

For details on the values to be stored in the buffer memory, refer to the following.

☞ Page 516 Simple CPU communication area

### ■Communication stop/restart request

When 'Simple CPU communication status for each setting number' (Un\G721936 to Un\G721999, Un\G1247460 to Un\G1247907) stores any of the following values, communications can be stopped by turning off and on 'Request to stop fixed interval communication for each setting number' (Un\G721900 to Un\G721903, Un\G1247328 to Un\G1247355). (If the value is other than the following, the stop request is invalid.)

- 3H: Communicating
- 5H: Retry being executed
- 6H: Monitoring

**Point**

When 'Request to stop fixed interval communication for each setting number' (Un\G721900 to Un\G721903, Un\G1247328 to Un\G1247355) is turned off and on, the communications will stop at the timing of sending data in the next execution interval.

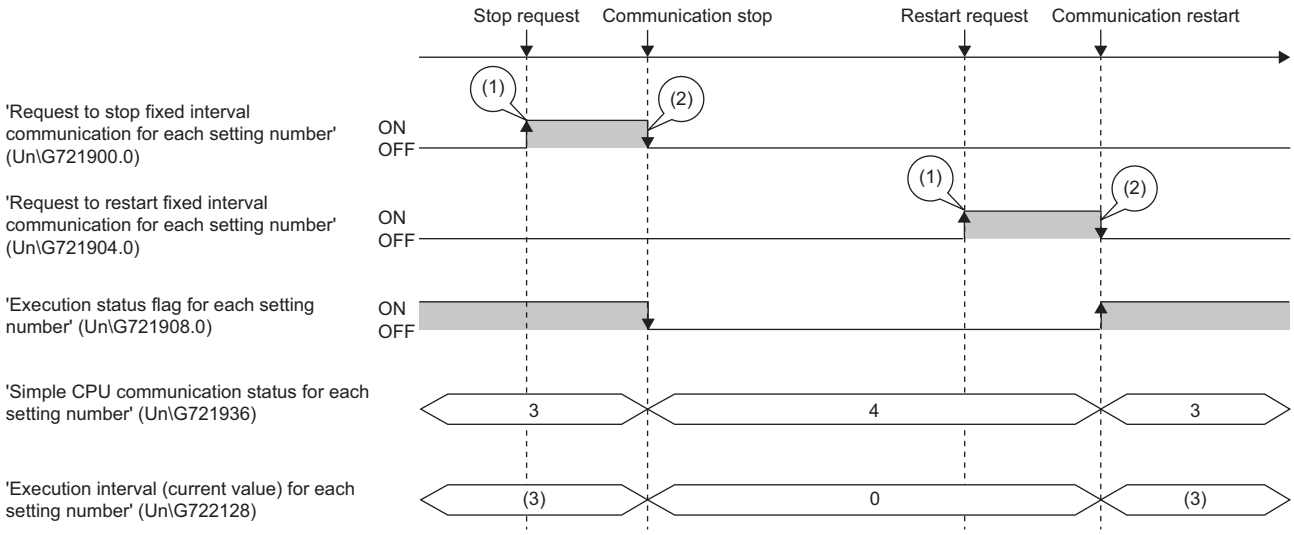
Therefore, if the communication destination does not respond with the stop request accepted, the communications will stop after the communication time-out period has passed. When the retry is being executed, the communications will stop after the communication time-out period has passed without depending on the retry count.

When 'Simple CPU communication status for each setting number' (Un\G721936 to Un\G721999, Un\G1247460 to Un\G1247907) is in the "4H: Communication stop" state, communications can be restarted by turning off and on 'Request to restart fixed interval communication for each setting number' (Un\G721904 to Un\G721907, Un\G1247356 to Un\G1247383). (If the status is other than "4H: Communication stop", the restart request is invalid.)

When the communication setting is "On Request" or no setting is configured, no operation is performed by turning off and on 'Request to restart fixed interval communication for each setting number' (Un\G721904 to Un\G721907, Un\G1247356 to Un\G1247383) and 'Request to stop fixed interval communication for each setting number' (Un\G721900 to Un\G721903, Un\G1247328 to Un\G1247355). Therefore, the buffer memory areas will remain on without being turned on and off by the system.

Ex.

The operation at communication stop/restart request for the setting No.1




(1) The buffer memory area is turned on by the program.
(2) The buffer memory area is turned off by the system.
(3) The current value is stored.

## Checking with "simple CPU communication diagnostics"

The communication status of the simple CPU communication function can be checked with "Simple CPU Communication Diagnostics".

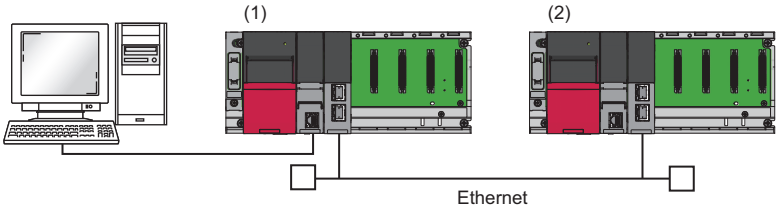
For details, refer to the following.

☞ Page 434 Simple CPU communication diagnostics

# Communication example

In this example, B0 to B1FFF of own station are transferred to M0 to M8191 of the communication destination using the simple CPU communication function. Also, W0 to W1FF of own station are transferred to D0 to D511 of the communication destination.

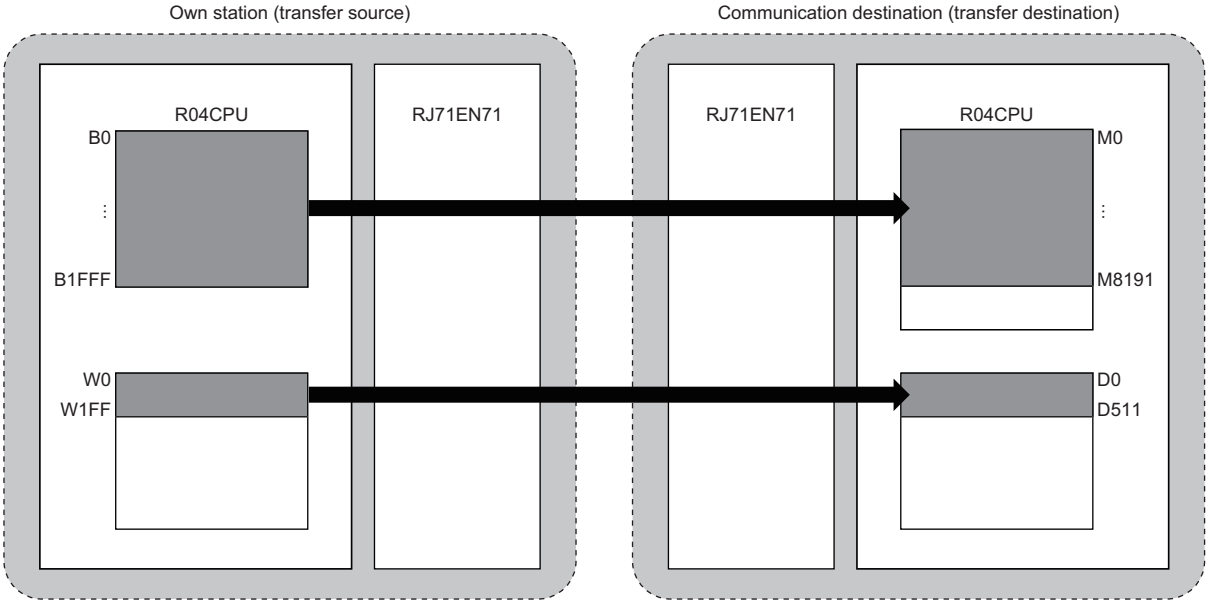
## System configuration




(1) Own station (transfer source): RJ71EN71 (IP address: 192.168.3.40)
(2) Communication destination (transfer destination): RJ71EN71 (IP address: 192.168.3.50)

## Device assignment

The following figure shows the assignment of devices that communicates data in the simple CPU communication.



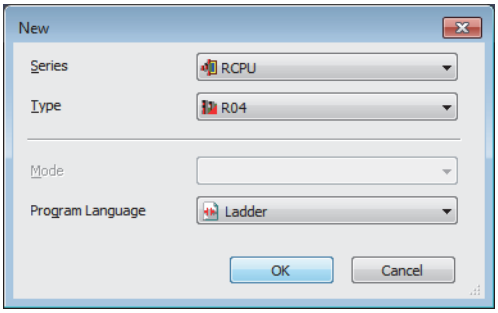
## Parameter settings

Connect the engineering tool to the CPU module and set the parameters.

### ■Settings for the transfer source

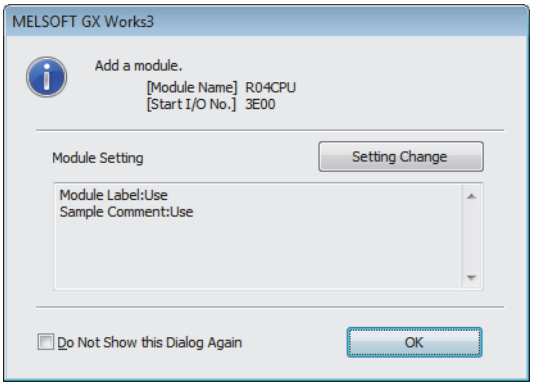
**1.** Set the CPU module as follows.

[Project] ⇨ [New]



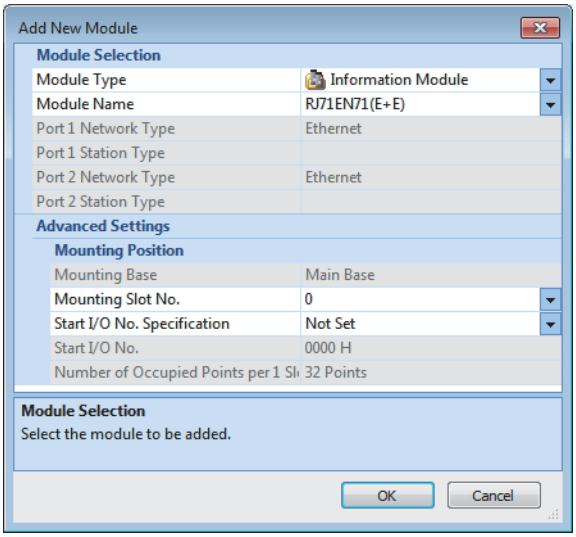

**2.** Click the [Setting Change] button and set to use module labels.

**3.** Click the [OK] button to add the module labels of the CPU module.

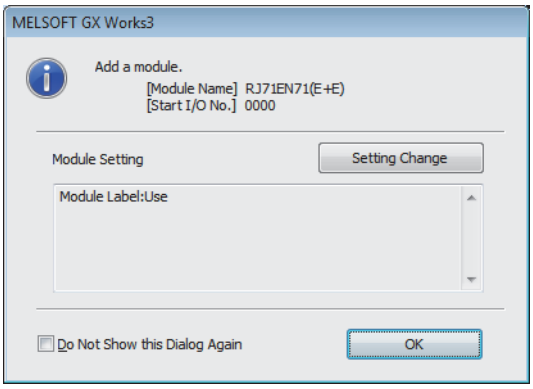



**4.** Set the RJ71EN71 as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]



**5.** Click the [OK] button to add the module labels of the RJ71EN71.




**6.** Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [Own Node Settings]

Setting Item

| Item | Setting |
|---|---|
| ***Own Node Settings*** | |
| Parameter Setting Method | Parameter Editor |
| IP Address | |
| IP Address | 192 . 168 . 3 . 40 |
| Subnet Mask | . . . |
| Default Gateway | . . . |
| Communications by Network No./Station No. | Disable |
| Setting Method | Use IP Address |
| Network No. | ----- |
| Station No. | ----- |
| Transient Transmission Group No. | 0 |
| Enable/Disable Online Change | Disable All (SLMP) |
| Communication Data Code | Binary |
| Opening Method | Do Not Open by Program |
| **External Device Configuration** | |
| External Device Configuration | <Detailed Setting> |

**7.** Set the items in "Application Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple CPU Communication Setting]

Setting Item

| Item | Setting |
|---|---|
| **Simple CPU Communication Setting** | |
| ***To Use or Not to Use Simple CPU Communication*** | Use |
| CPU Response Monitoring Timer | 5 |
| Unit | s |
| Simple CPU Communication Setting | <Detailed Setting> |
| **Simple Device Communication Setting** | |
| To Use or Not to Use Simple Device Communication | Not to Use |
| Resource Setting | Normal |
| Instruction Initial Value to Start/Stop Function | Start |
| CPU Response Monitoring Timer | 5 |
| Unit | s |
| Communication Destination Setting | <Detailed Setting> |
| Device List | <Confirmation> |

**8.** Set the items in "Simple CPU Communication Setting" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple CPU Communication Setting] ⇨ [Detailed Setting]

| Setting No. | Communication Pattern | Communication Setting: Execution Interval(ms) | | Communication Destination (IP Address) Source | -> | Destination | Target PLC No. |
|---|---|---|---|---|---|---|---|
| 1 | Write | On Request | | Host Station(192 . 168 . 3 . 40) | -> | MELSEC iQ-R(Ether)(192 . 168 . 3 . 50) | Not Specified |
| 2 | | | | | -> | | |
| 3 | | | | | -> | | |

| Bit Device | | | | | | | | Word Device | | | | | | | | Communication Time-out Period(ms) | Communication Retry Count | Monitoring Time At Error(s) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Points | Source | | | -> | Destination | | | Points | Source | | | -> | Destination | | | | | |
| | Type | Start | End | | Type | Start | End | | Type | Start | End | | Type | Start | End | | | |
| 8192 | B | 00000 | 01FFF | -> | M | 0 | 8191 | 512 | W | 00000 | 001FF | -> | D | 0 | 511 | 1000 | 3 | |
| | | | | -> | | | | | | | | -> | | | | | | |
| | | | | -> | | | | | | | | -> | | | | | | |

**9.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

## ■Settings for the transfer destination

1. Set the CPU module and the RJ71EN71. Steps to take are similar to those for the transfer source, except no setting for module labels is required. (☞ Page 219 Settings for the transfer source)
2. Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [Own Node Settings]

Setting Item

| Item | Setting |
|---|---|
| ***Own Node Settings*** | |
| Parameter Setting Method | Parameter Editor |
| IP Address | |
| IP Address | 192 . 168 . 3 . 50 |
| Subnet Mask | . . . |
| Default Gateway | . . . |
| Communications by Network No./Station No. | Disable |
| Setting Method | Use IP Address |
| Network No. | ----- |
| Station No. | ----- |
| Transient Transmission Group No. | 0 |
| Enable/Disable Online Change | Disable All (SLMP) |
| Communication Data Code | Binary |
| Opening Method | Do Not Open by Program |
| **External Device Configuration** | |
| External Device Configuration | <Detailed Setting> |

3. Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

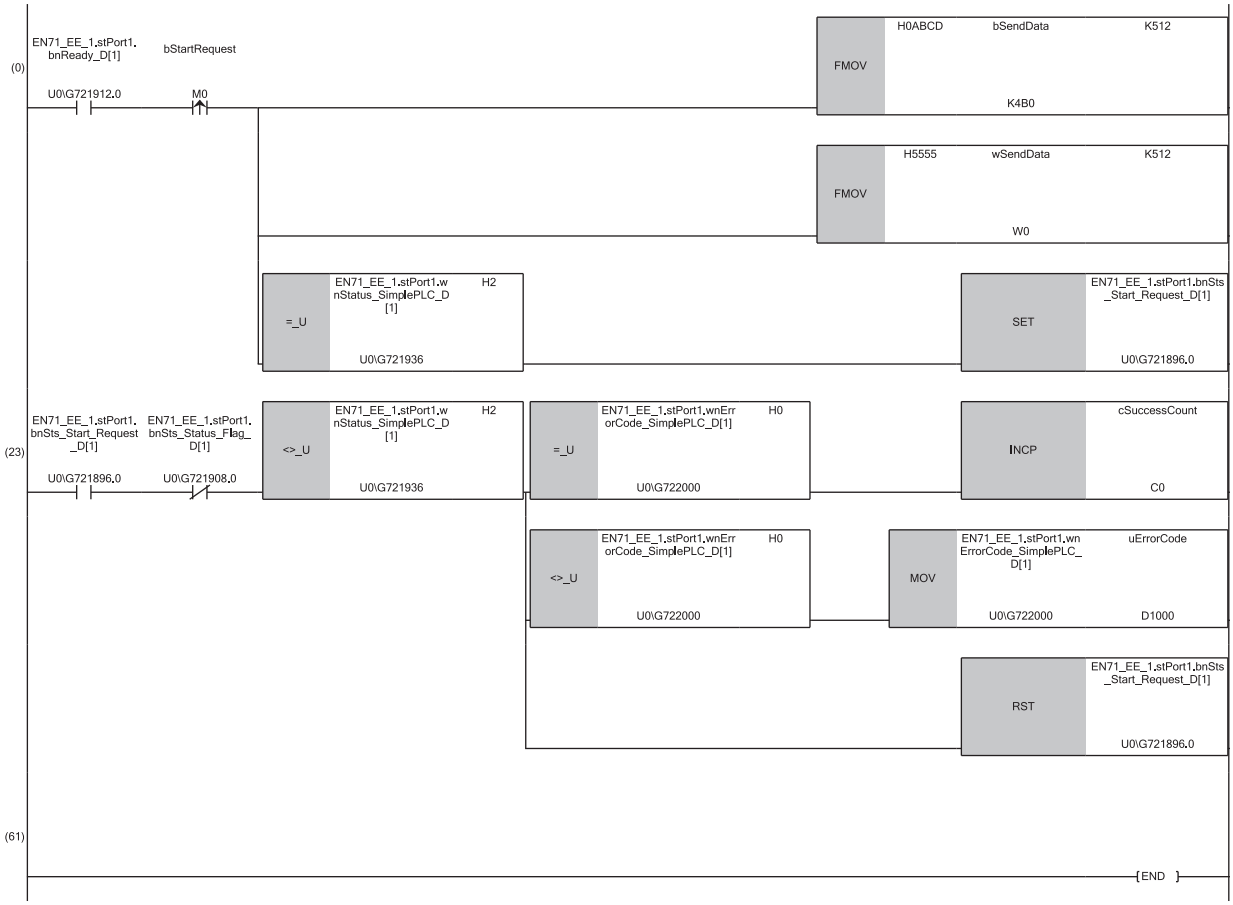
## Program examples

Write the following program to the transfer source CPU module.

In this example, ABCDH is stored in a bit device and 5555H is stored in a word device, and transfer them.

| Classification | Label name | Description | Device |
|---|---|---|---|
| Module label | EN71_EE_1.stPort1.bnSts_Start_Request_D[1] | Request to start communication at request of Simple CPU communication setting No.1 | Un\G721896.0 |
| | EN71_EE_1.stPort1.bnSts_Status_Flag_D[1] | Execution flag of Simple CPU communication setting No.1 | Un\G721908.0 |
| | EN71_EE_1.stPort1.bnReady_D[1] | Preparation completion flag of Simple CPU communication setting No.1 | Un\G721912.0 |
| | EN71_EE_1.stPort1.wnStatus_SimplePLC_D[1] | Communication status storage location of Simple CPU communication setting No.1 | Un\G721936 |
| | EN71_EE_1.stPort1.wnErrorCode_SimplePLC_D[1] | Error code storage location of Simple CPU communication setting No.1 | Un\G722000 |
| Label to be defined | Define global labels as shown below: | | |

| Label Name | Data Type | | Class | Assign (Device/Label) |
|---|---|---|---|---|
| bStartRequest | Bit | ... | VAR_GLOBAL | M0 |
| bSendData | Word [Signed] | ... | VAR_GLOBAL | K4B0 |
| wSendData | Word [Signed] | ... | VAR_GLOBAL | W0 |
| cSuccessCount | Counter | ... | VAR_GLOBAL | C0 |
| uErrorCode | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D1000 |




(0) Turn on 'Request execution device of the simple CPU communication setting No.1' (M0) and store the transfer data ABCDH to 512 points of data from bSendData (K4B0), and the transfer data (5555H) to 512 points of data from wSendData (W0).
Turn on 'Request to start communication at request of Simple CPU communication setting No.1' (Un\G721896.0) when 'Communication status storage location of Simple CPU communication setting No.1' (Un\G721936) is 2H.

(23)Start counting 'Number of successful communications' (C0) when communications start successfully.
Store an error code in 'Simple CPU communication setting No.1 error code' (D1000) when the abnormal state is detected.

# Precautions

This section provides precautions for the simple CPU communication function.

## Incomplete data reception and timeout

Because the load of Ethernet communication becomes heavy during execution of the simple CPU communication, if another communication function (such as MELSOFT connection, SLMP communication, socket communication function, and simple device communication function) is simultaneously performed by using UDP, the data reception of UDP may not be completed, resulting in a timeout error or other errors. Therefore, to perform other communications during the simple CPU communication, the communication using TCP is recommended.

## Execution interval setting

The execution interval of the simple CPU communication function and time required for completing the communications will differ according to the load rate of the Ethernet line, operating status of other communication functions (such as MELSOFT connection, SLMP communication, socket communication function, and simple device communication function), scan time of the communication destination, and system configuration. Thus, the set execution interval or time required for completing the communications may be increased.

### ■Cause of increased execution interval of the simple CPU communication

- The scan time of the control CPU module of the RJ71EN71 where the simple CPU communication function is set is long.
- The RJ71EN71 where the simple CPU communication function is set is communicating by using a function other than the simple CPU communication function, and the processing of the communication by using a function other than the simple CPU communication function takes time.
- The load on the Ethernet line is high.
- The number of settings in "Simple CPU Communication Setting" under "Application Settings" is large.
- Response from the communication destination is delayed.
- More than one RJ71EN71 where the simple CPU communication function is set is mounted on a base unit (main base unit or extension base unit).

To perform communications at the set execution interval, set a sufficient execution interval. If checking the execution interval is required, perform actual communications and check it with the "Simple CPU Communication Diagnostics" window.

## Duplication of own station port numbers

The simple CPU communication cannot be performed in the following cases.

- The own station port number set in the simple CPU communication and the own station port number of the CPU module set in the external device configuration are the same.
- The own station port number set in the simple CPU communication and the one set in another function such as the socket communication are the same. ('Simple CPU communication status for each setting number' (Un\G721936 to Un\G721999) is "AH: Communications impossible".)

Set a unique own station port number. For the setting numbers other than the one in which a communication error occurs, communications are performed after the preparation processing is completed normally.

## IP address and module type of communication destination

Check the IP address and module type of the communication destination before setting the parameters. The simple CPU communication is not performed when the communication destination does not exist (an incorrect IP address is specified) or when the preparation processing is not completed. ('Simple CPU communication status for each setting number' (Un\G721936 to Un\G721999) is AH: Communications impossible.)

## Communication destination device

Check the type and range of the device to be read or written in the communication destination.

Especially, when "Write" is set to "Communication Pattern", the control data of the communication destination may be overwritten, resulting in malfunction.

## Security

The simple CPU communication is not performed when a remote password is set for the communication destination. Clear the remote password.
If the IP filter has been set in the communication destination, check the setting and correct it if necessary.
If any security function specific to the communication destination (programmable controller products manufactured by other companies), clear the function.

## When using the file register (R, ZR)

When using the file register (R, ZR) for the device on the own station, always assign the device within the setting range. If the devices are not assigned, an error occurs (error code: 4031), and the simple CPU communication of the setting number which exceeds the setting range cannot be performed.

## Setting of the network and station numbers

When the communication destination is the MELSEC iQ-R (Ethernet module), MELSEC-Q (Ethernet module), or MELSEC-L (Ethernet module), the network number and station number should be set in the own node settings of the Ethernet parameters.

## Usable connections

Up to 64 connections can be used in the simple CPU communication settings. If, however, connection has been used for other functions, the maximum number of connections is the difference calculated by subtracting the number of connections used for other function from 64.

## Measures against errors occurring during communications

When an error occurs during writing communications, data writing may not be completed properly. Check the error details, take appropriate measures, and restart communications.

## Restrictions when certain devices are used

When using the following devices, use modules with the firmware version of "62" or later.

- F, T (TS, TC, TN)
- ST (STS, STC, STN)
- C (CS, CC, CN)

## Re-initial processing

When performing re-initial processing while using simple CPU communication, check the following.
(Check the communication status using 'Simple CPU communication status for each setting number' (Un\G721936 to Un\G721999, Un\G1247460 to Un\G1247907).)

### ■When the communication setting is "Fixed Interval"

Perform re-initial processing when the communication status is "4H: Communication stop" due to a communication stop request.
During re-initial processing, the communication status becomes "1H: Preparing". After the re-initial processing is completed, the communication status returns to "4H: Communication stop" when communication through the port to be used becomes available.

### ■When the communication setting is "On Request"

Perform re-initial processing while the communication status is "2H: Waiting for the request".
During re-initial processing, the communication status becomes "1H: Preparing". After the re-initial processing is completed, the communication status returns to "2H: Waiting for the request" when communication through the port to be used becomes available.

## Redundant system

Simple CPU communication operates only in the control system. Communication is not performed in the standby system.
In addition, when using simple CPU communication in a redundant system configuration, check the following.
☞ Page 418 Precautions on simple CPU communication

## Restrictions applicable depending on versions

The simple CPU communication function is available for the RJ71EN71 or RnENCPU (network part) whose firmware version is "35" or later and that uses the engineering tool with version 1.050C or later.
The terms of "Fixed Interval" and "On Request" in the list are used for the communication setting in "Simple CPU Communication Setting" under "Application Settings".

| Firmware version of the RJ71EN71 or the RnENCPU (network part) | Engineering tool version | Description |
|---|---|---|
| "35" to "38" | 1.050C | • Number of settable modules: 1 to 64<br>• Communication setting: "Fixed Interval" only<br>• Communication destination: MELSEC iQ-R (built-in Ethernet), MELSEC-Q (built-in Ethernet), MELSEC-L (built-in Ethernet), MELSEC iQ-F (built-in Ethernet), MELSEC iQ-R (Ethernet module), MELSEC-Q (Ethernet module), MELSEC-L (Ethernet module), SLMP-compatible device (QnA-compatible 3E frame), OMRON (CS/CJ series), KEYENCE (KV series), Panasonic (FP2SH series), Panasonic (FP7 series), YASKAWA MP3000 series, YASKAWA MP2000 series, Yokogawa FA-M3 series, MODBUS/TCP-compatible device<br>• Option setting: Cannot be set (the setting value is regarded as 0. The FINS node address is regarded as 1). |
| | 1.052E or later | ■Simple CPU communication settings No.1 to 64<br>Same as the engineering tool with version 1.050C[*1*2]<br>■Simple CPU communication settings No.65 to 512<br>Simple CPU communication impossible[*3] |
| | 1.065T or later | ■Simple CPU communication settings No.1 to 64<br>• Same as the engineering tool with version 1.050C[*1*2]<br>• The setting value is not reflected even if the option is set. (The setting value is regarded as 0. The FINS node address is regarded as 1).<br>■Simple CPU communication settings No.65 to 512<br>Simple CPU communication impossible[*3] |
| "39" to "41" | 1.050C | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 35 to 38. |
| | 1.052E or later | • Number of settable modules: 1 to 512<br>• Communication setting: "Fixed Interval" or "On Request"<br>• Communication destination: MELSEC-A/AnS (Ethernet module) and MELSEC-FX3 (Ethernet block/adapter) are added to the communication destination that uses the engineering tool with version 1.050C.<br>• Option setting: Cannot be set (the setting value is regarded as 0. The FINS node address is regarded as 1). |
| | 1.065T or later | • Same as the engineering tool with versions 1.052E and later<br>• The setting value is not reflected even if the option is set. (The setting value is regarded as 0. The FINS node address is regarded as 1). |
| "42" to "46" | 1.050C | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 35 to 38. |
| | 1.052E or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 39 to 41. |
| | 1.060N or later | • Communication destination: MELSEC iQ-L (built-in Ethernet), SIEMENS S7 series, Fuji Electric MICREX-SX series, and JTEKT TOYOPUC series are added to the communication destination that uses the engineering tool with version 1.052E.<br>• Redundant system<br>• Option setting: Cannot be set (the setting value is regarded as 0. The FINS node address is regarded as 1). |
| | 1.065T or later | • Communication destination: MELSEC iQ-L (built-in Ethernet), SIEMENS S7 series, Fuji Electric MICREX-SX series, and JTEKT TOYOPUC series are added to the communication destination that uses the engineering tool with version 1.052E.<br>• Redundant system<br>• The setting value is not reflected even if the option is set. (The setting value is regarded as 0. The FINS node address is regarded as 1). |

| Firmware version of the RJ71EN71 or the RnENCPU (network part) | Engineering tool version | Description |
|---|---|---|
| "47" to "61" | 1.050C | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 35 to 38. |
| | 1.052E or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 39 to 41. |
| | 1.060N or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 42 to 46. |
| | 1.065T or later | Option setting: Configurable for MODBUS/TCP-compatible devices. FINS node addresses cannot be set. (They are regarded as 1.) |
| "62" to "87" | 1.050C | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 35 to 38. |
| | 1.052E or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 39 to 41. |
| | 1.060N or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 42 to 46. |
| | 1.065T or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 47 to 61. |
| | 1.085P or later | • Devices that can use F, T, ST, and C: own station, MELSEC iQ-R (built-in Ethernet, Ethernet module), MELSEC-Q (built-in Ethernet, Ethernet module), MELSEC-L (built-in Ethernet, Ethernet module), MELSEC iQ-F (built-in Ethernet), MELSEC iQ-L (built-in Ethernet), SLMP-compatible device (QnA-compatible 3E frame)<br>• Devices that can use F, T, and C: MELSEC-A/AnS (Ethernet module)<br>• Devices that can use S, T, and C: MELSEC-FX3 (Ethernet block/adapter)<br>• Communication destination: SIEMENS S7 series (extension) is added.<br>• Option setting: Configurable for MODBUS/TCP-compatible devices. FINS node addresses cannot be set. (They are regarded as 1.) |
| "88" or later | 1.050C | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 35 to 38. |
| | 1.052E or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 39 to 41. |
| | 1.060N or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 42 to 46. |
| | 1.065T or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 47 to 61. |
| | 1.085P or later | Same as the RJ71EN71 or RnENCPU (network part) with firmware versions 62 to 87. |
| | 1.125F or later | • Communication destination: OMRON (NJ/NX series) is added.<br>• Option setting: FINS node addresses of the own station can be set while an OMRON module is selected for the module type. |

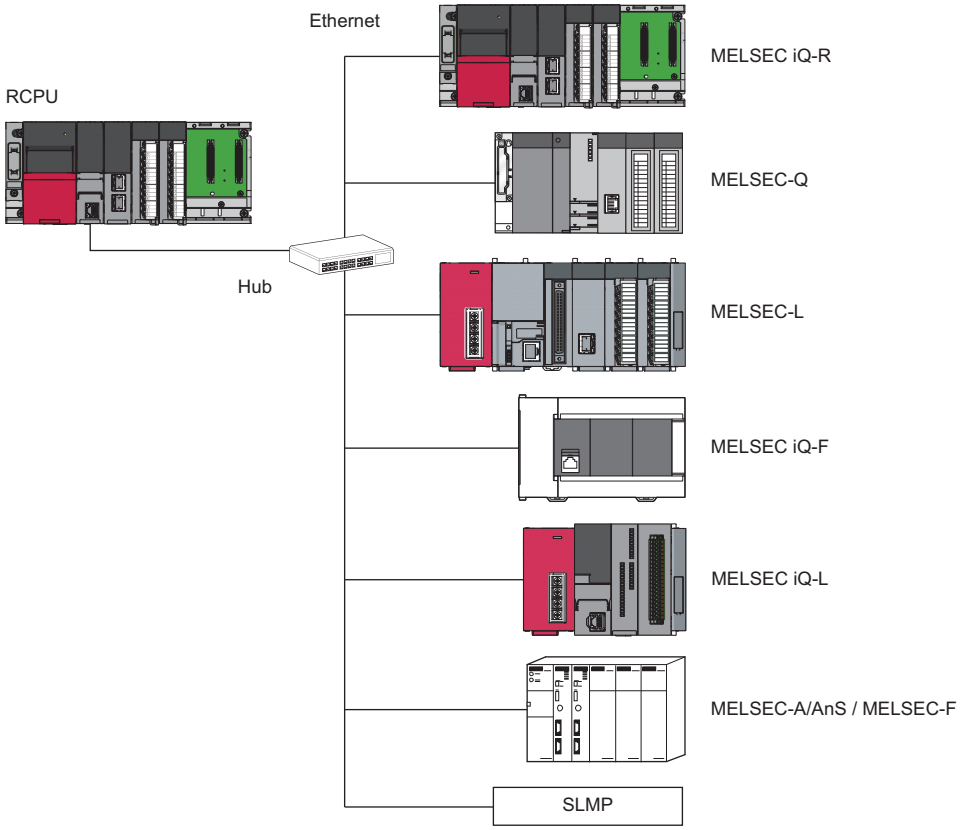
*1 The simple CPU communication of the setting number with "On Request" set to the communication setting is not executed and a simple CPU communication error (error code: CFBFH) occurs.

*2 When a duplicate port number is set to the own stations, a duplication error of own port number (error code: CFB2H) occurs.
The simple CPU communication of the setting number with an unsupported communication destination is not executed, and a simple CPU communication error (CFB0H or CFBFH) occurs.

*3 Setting No.65 to 512 are prohibited to use. Using these areas may overwrite the values in the G device, resulting in malfunction of the module. To use setting No.65 to 512, use RJ71EN71 or RnENCPU (network part) with the firmware version of "39" or later.

# 1.15 Simple CPU Communication Function (CPU Module (Built-in Ethernet Port Part))

This function allows data communications between specified devices at the specified timing just by making simple settings of the CPU module (built-in Ethernet port part) from an engineering tool. Send/receive is performed with a specified communication destination on a one-to-one basis.




| Communication destination | Type | Description |
|---|---|---|
| MELSEC iQ-R series | CPU module (Built-in Ethernet, Ethernet Module*1) | ☞ Page 230 Module type |
| MELSEC MX Controller MX-R Model*1 | Controller (Built-in Ethernet) | |
| MELSEC MX Controller MX-F Model*1 | Controller (Built-in Ethernet) | |
| MELSEC-Q series | CPU module (built-in Ethernet), Ethernet module*1 | |
| MELSEC-L series | CPU module (built-in Ethernet), Ethernet module*1 | |
| MELSEC iQ-F series*1 | CPU module (built-in Ethernet) | |
| MELSEC iQ-L series*1 | CPU module (built-in Ethernet) | |
| MELSEC-A/AnS series*1 | Ethernet module | |
| MELSEC-F series*1 | Ethernet block/adapter | |
| SLMP-compatible device | QnA-compatible 3E frame | |

*1 For the communication destinations and types that can be set, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

**Point**

Access via routers is also available. For the access, set the subnet mask and default gateway. (☞ Page 327 Own Node Settings)

**Restriction**

Before using the simple CPU communication function, check the versions of the CPU module used. (☞ Page 586 Added and Enhanced Functions)

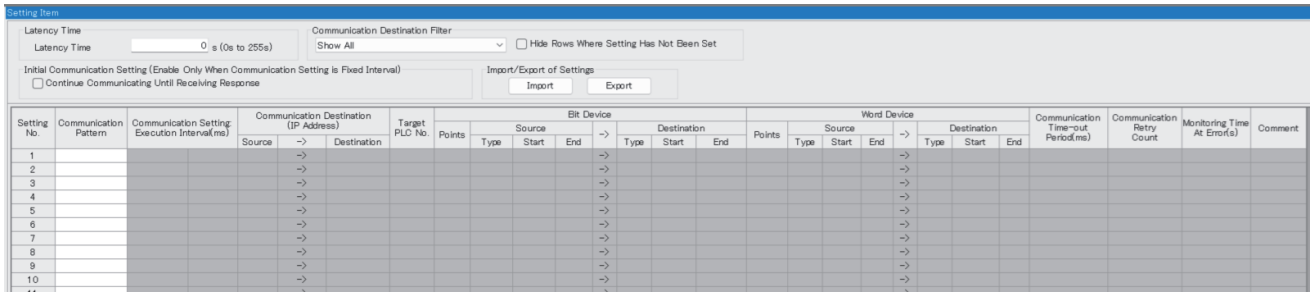
# Setting procedure

The following section describes the settings for using this function.

1. Set "Enable" for "To Use or Not to Use Simple CPU Communication Setting" in "Simple CPU Communication Setting" under "Application Settings".

[Navigation window] ⇨ [Parameter] ⇨ Target module ⇨ [Module Parameter] ⇨[Application Settings] ⇨ [Simple CPU Communication Setting]

2. Double-click "<Detailed Setting>" of the "Simple CPU Communication Setting" to display the following window. (☞ Page 345 Simple CPU communication setting (CPU module (built-in Ethernet port part)))




3. After setting the items, write the module parameters to the CPU module of the own station, and reset or power off and on the CPU module to start the simple CPU communication.

## Communication Pattern

Select the communication pattern from the following items.

| Setting item | Description |
|---|---|
| Read | Read the data of the specified destination device (transmission source) to the specified device of the own station (transmission destination). |
| Write | Write the data of the specified device of the own station (transmission source) to the specified destination device (transmission destination). |

## Communication Setting: Execution Interval

Select the communication timing from the following items.

| Setting item | | Description | Setting range |
|---|---|---|---|
| Fixed Interval | — | Data are communicated between the devices at a specified execution interval. | — |
| | Execution Interval (ms) | Execution interval is set. | 10ms to 65535ms (in increments of 1ms) |
| On Request*1 | | Data are sent and received only upon request. | — |

*1 To set "On Request", check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

**Point**

- Actual time of execution interval may be longer than the value of the setting because the time is affected by the specified communication destinations or Ethernet line congestion. For the preventive measures, refer to the following.

☞ Page 448 When the simple CPU communication function is used

- If latency time is set, data communication is started after the latency time has passed. (☞ Page 242 Latency Time)

## Communication Destination (IP Address)

Click "Destination" to open the "Communication Destination Setting" window and specify the communication destination. Up to 64 communication destinations can be registered.

### ■Module type

Select the module type of the communication destination from the following list.

| Module type | Communication destination | Communication specification |
|---|---|---|
| MELSEC iQ-R (Built-in Ethernet) | MELSEC iQ-R series CPU module*1 | MELSOFT connection (protocol: UDP) |
| MELSEC MX Controller (MX-R Model) (Built-in Ethernet)*5 | MELSEC MX Controller MX-R Model | |
| MELSEC MX Controller (MX-F Model) (Built-in Ethernet)*5 | MELSEC MX Controller MX-F Model | |
| MELSEC-Q (Built-in Ethernet) | Built-in Ethernet port QCPU | |
| MELSEC-L (Built-in Ethernet) | Built-in Ethernet port LCPU | |
| MELSEC iQ-F (Built-in Ethernet) | MELSEC iQ-F series CPU module*2 | |
| MELSEC iQ-L (Built-in Ethernet) | MELSEC iQ-L series CPU module | |
| MELSEC iQ-R (Ethernet Module)*5 | RCPU on which the RJ71EN71 can be mounted, MELSEC MX Controller MX-R Model*3, or RnENCPU | |
| MELSEC-Q (Ethernet Module)*5 | QCPU*3 on which an Ethernet module can be mounted | |
| MELSEC-L (Ethernet Module)*5 | LCPU*3 on which an Ethernet module can be mounted | |
| MELSEC-A/AnS (Ethernet Module) | ACPU*4 on which an Ethernet module can be mounted | MC protocol (A-compatible 1E frame) |
| MELSEC-FX3 (Ethernet Block/Adapter) | • FX3U and FX3UC CPUs with which an Ethernet Block can be mounted<br>• FX3S, FX3G, FX3GC, FX3U, and FX3UC CPUs with which an Ethernet Adapter can be mounted | |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | • SLMP-compatible device (No serial number)<br>• MC protocol 3E frame-compatible device | SLMP (MC protocol QnA-compatible 3E frame)<br>Read (0401H)<br>Write (1401H) |

*1 The RnENCPU can communicate when the Ethernet port of its CPU part is connected.
*2 For the FX5U CPU module and FX5UC CPU module, the module with the firmware version of "1.110" or later can be used.
*3 The CPU module that controls the Ethernet module is the communication destination.
*4 The AnCPU, AnSCPU, and QCPU (A mode) are included.
*5 For the CPU modules and their available versions, refer to the following.
☞ Page 586 Added and Enhanced Functions

### ■IP address, TCP/UDP, port number, and own station port number

Set the IP address of the communication destination, communication protocol, port number, and own port number.

| Module type | Setting range | | | |
|---|---|---|---|---|
| | IP address | TCP/UDP | Port No. | Own station port number |
| MELSEC iQ-R (Built-in Ethernet) | 0.0.0.1 to 223.255.255.254 | UDP | Cannot be set. | Cannot be set. |
| MELSEC MX Controller (MX-R Model) (Built-in Ethernet) | | | | |
| MELSEC MX Controller (MX-F Model) (Built-in Ethernet) | | | | |
| MELSEC iQ-L (Built-in Ethernet) | | | | |
| MELSEC iQ-F (Built-in Ethernet) | | | | |
| MELSEC-Q (Built-in Ethernet) | | | | |
| MELSEC-L (Built-in Ethernet) | | | | |
| MELSEC iQ-R (Ethernet Module) | | | | |
| MELSEC-Q (Ethernet Module) | | | | |
| MELSEC-L (Ethernet Module) | | | | |
| MELSEC-A/AnS (Ethernet Module) | | | 256 to 65534 | 1 to 4999, 5010 to 65534 |
| MELSEC-FX3 (Ethernet Block/Adapter) | | | 1025 to 5548, 5552 to 65534 | |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | | | 1 to 4999, 5010 to 65534 | |

**Point**

- When a multiple CPU system is configured, specify the IP address of the CPU module to which the Ethernet cable is connected.
- In communications where the communication destination other than MELSEC-A/AnS, MELSEC-FX3, and SLMP Corresponding Device is set, port numbers 61696 to 65534 are dynamically used for the own station port numbers. Therefore, when using the simple CPU communication function, do not set 61696 to 65534 to the own station port numbers for establishing a connection instruction (SP.SOCOPEN) of the socket communication function. If set, the instruction may not be completed properly.
- Port numbers 1 to 1023 are typically reserved port numbers (WELL KNOWN PORT NUMBERS) and 61440 to 65534 are used by other communication functions, so 1024 to 4999 or 5010 to 61439 should be used for the own station port numbers.
- When specifying the same communication destination in multiple settings, set different own station port numbers.

## Setting for communication destination

The settings differ depending on the communication destination.

| Communication destination | Necessary setting |
|---|---|
| MELSEC iQ-R (Built-in Ethernet) | Since MELSOFT Transmission Port UDP/IP is used for communications, no setting is required. |
| MELSEC MX Controller (MX-R Model) (Built-in Ethernet) | |
| MELSEC MX Controller (MX-F Model) (Built-in Ethernet) | |
| MELSEC-Q (Built-in Ethernet) | Set the following items (for the number of devices) in the "Open Setting" window.<br>• Protocol: "UDP"<br>• Open System: "MELSOFT Connection" |
| MELSEC-L (Built-in Ethernet) | |
| MELSEC iQ-F (Built-in Ethernet) | Since MELSOFT Transmission Port UDP/IP is used for communications, no setting is required. |
| MELSEC iQ-L (Built-in Ethernet) | |
| MELSEC iQ-R (Ethernet Module) | Set the network number and station number. |
| MELSEC-Q (Ethernet Module) | |
| MELSEC-L (Ethernet Module) | |
| MELSEC-A/AnS (Ethernet Module) | ■Programming tool<br>Create an initial processing program, and set the following items.<br>• IP address and port number<br>• Communication method: "UDP"<br>• Bit 15 and Bits 0 to 7 of Area for communication instruction during stop (Un\G496) (Turn on the bits corresponding to the connection numbers to be used.)*1<br>For the initial processing program, refer to the following.<br>📖 For A Ethernet Interface Module User's Manual<br>■DIP switch on the front of the module<br>Set the following items.<br>• SW2 (data code setting): Off<br>• SW7 (CPU communication timing setting): On*2 |
| MELSEC-FX3 (Ethernet Block/Adapter) | Set "Communication Data Code" of "Ethernet Port" to "Binary Code".<br>Set the following items (for the number of devices) in the "Open Setting" window.<br>• IP address and port number<br>• Protocol: "UDP"<br>• Open system: "MC protocol" |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | Configure the settings so that the SLMP (QnA-compatible 3E frame of the MC protocol) can be used on the SLMP-compatible device side. (📖 Manual for the device used)<br>Set the communication data code to binary.<br>Set the protocol to UDP. |

*1 Set this item to execute this function while the communication destination CPU module is in the STOP state or stop error state.

*2 Turn on this switch to execute this function while the communication destination CPU module is in the RUN state.

## Target PLC No.

When the communication destination is one of the following modules and a multiple CPU system is used, specify the "Target PLC No." of the communication destination.

- MELSEC iQ-R (built-in Ethernet): RnCPU, RnENCPU, RnPCPU, RnSFCPU
- MELSEC-Q (built-in Ethernet): QnUDE(H)CPU, QnUDVCPU, QnUDPVCPU (📖 QCPU User's Manual (Multiple CPU System))

## Devices that can be specified

The devices that can be specified as transmission source and transmission destination vary depending on the communication destination. Bit device and word device can be set together for each setting number.
For devices that can be specified, the size of a device is specified in units of 16 points for a bit device and one point for a word device. Use 0 or multiples of 16 to specify the device number of a bit device.

**Point**

- Set the number of device points to the total number of setting No.1 to 64 (8192 words maximum).
- The maximum number of points per setting is 512 words (bit device 8192 points maximum, word device 512 points maximum).

**Restriction**

A safety device cannot be specified.

### ■Devices that can be specified on the own station

The maximum number of points when the NZ2MC-16MBS is connected to the R120CPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 161882111 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F*1 | 0 to 32767 | — |
| | B | 0H to 9A61FFFH | — |
| | TS/TC*1 | 0 to 8993439 | • A local device cannot be specified.<br>• This device can be specified only when "Write" is set for "Communication Pattern". |
| | STS/STC*1 | 0 to 8993439 | • A local device cannot be specified.<br>• This device can be specified only when "Write" is set for "Communication Pattern". |
| | CS/CC*1 | 0 to 8993439 | • A local device cannot be specified.<br>• This device can be specified only when "Write" is set for "Communication Pattern". |
| | SB | 0H to 9A61FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 10117631 | A local device cannot be specified. |
| | W | 0H to 9A61FFH | — |
| | TN*1 | 0 to 8993439 | A local device cannot be specified. |
| | STN*1 | 0 to 8993439 | A local device cannot be specified. |
| | CN*1 | 0 to 8993439 | A local device cannot be specified. |
| | SW | 0H to 9A61FFH | — |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10027007 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

### ■When the communication destination is the MELSEC iQ-R (built-in Ethernet)

The maximum number of points when the NZ2MC-16MBS is connected to the R120CPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 161882111 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F*1 | 0 to 32767 | — |
| | B | 0H to 9A61FFFH | — |
| | TS/TC*1 | 0 to 8993439 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC*1 | 0 to 8993439 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC*1 | 0 to 8993439 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 9A61FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 10117631 | A local device cannot be specified. |
| | W | 0H to 9A61FFH | — |
| | TN*1 | 0 to 8993439 | A local device cannot be specified. |
| | STN*1 | 0 to 8993439 | A local device cannot be specified. |
| | CN*1 | 0 to 8993439 | A local device cannot be specified. |
| | SW | 0H to 9A61FFH | — |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10027007 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

### ■When the communication destination is the MELSEC MX Controller (MX-R Model) (Built-in Ethernet)

The following table lists the maximum number of points.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 167747583 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to 9FF9FFFH | — |
| | TS/TC | 0 to 9319295 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC | 0 to 9319295 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC | 0 to 9319295 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 9FF9FFFH | — |
| | SM | 0 to 4495 | — |
| Word device | D | 0 to 10484223 | A local device cannot be specified. |
| | W | 0H to 9FF9FFH | — |
| | TN | 0 to 9319295 | A local device cannot be specified. |
| | STN | 0 to 9319295 | A local device cannot be specified. |
| | CN | 0 to 9319295 | A local device cannot be specified. |
| | SW | 0H to 9FF9FFH | — |
| | SD | 0 to 4495 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10452991 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 4194303 | — |

### ■When the communication destination is the MELSEC MX Controller (MX-F Model) (Built-in Ethernet)

The following table lists the maximum number of points.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 67084287 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to 3FF9FFFH | — |
| | TS/TC | 0 to 3726879 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC | 0 to 3726879 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC | 0 to 3726879 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 3FF9FFFH | — |
| | SM | 0 to 9999 | — |
| Word device | D | 0 to 4192767 | A local device cannot be specified. |
| | W | 0H to 3FF9FFH | — |
| | TN | 0 to 3726879 | A local device cannot be specified. |
| | STN | 0 to 3726879 | A local device cannot be specified. |
| | CN | 0 to 3726879 | A local device cannot be specified. |
| | SW | 0H to 3FF9FFH | — |
| | SD | 0 to 9999 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 4161535 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 262143 | — |

## ■When the communication destination is the MELSEC iQ-R (Ethernet module)

The following table lists the maximum number of points.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 167747583 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F | 0 to 32767 | — |
| | B | 0H to 9FF9FFFH | — |
| | TS/TC | 0 to 9319295 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC | 0 to 9319295 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC | 0 to 9319295 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 9FF9FFFH | — |
| | SM | 0 to 4495 | — |
| Word device | D | 0 to 10484223 | A local device cannot be specified. |
| | W | 0H to 9FF9FFH | — |
| | TN | 0 to 9319295 | A local device cannot be specified. |
| | STN | 0 to 9319295 | A local device cannot be specified. |
| | CN | 0 to 9319295 | A local device cannot be specified. |
| | SW | 0H to 9FF9FFH | — |
| | SD | 0 to 4495 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10452991 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 4194303 | — |

### ■When the communication destination is the MELSEC-Q (built-in Ethernet) or MELSEC-Q (Ethernet module)

The maximum number of points when the Q4MCA-8MBS is connected to the Q26UDVCPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 1FFFH | — |
| | Y | 0H to 1FFFH | — |
| | M | 0 to 61439 | — |
| | L | 0 to 32767 | — |
| | F*1 | 0 to 32767 | — |
| | B | 0H to EFFFH | — |
| | TS/TC*1 | 0 to 32767 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC*1 | 0 to 32767 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC*1 | 0 to 32767 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 7FFFH | — |
| | SM | 0 to 2047 | — |
| Word device | D | 0 to 4910079 | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | W | 0H to 4AEBFFH | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | TN*1 | 0 to 32767 | — |
| | STN*1 | 0 to 32767 | — |
| | CN*1 | 0 to 32767 | — |
| | SW | 0H to 7FFFH | — |
| | SD | 0 to 2047 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 4849663 | Data is read or written following the file register settings of the communication destination. |

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

### ■When the communication destination is the MELSEC-L (built-in Ethernet) or MELSEC-L (Ethernet module)

The following table lists the maximum number of points for the L26CPU. The number of points varies depending on the CPU module used.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 1FFFH | — |
| | Y | 0H to 1FFFH | — |
| | M | 0 to 61439 | — |
| | L | 0 to 32767 | — |
| | F*1 | 0 to 32767 | — |
| | B | 0H to EFFFH | — |
| | TS/TC*1 | 0 to 25471 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC*1 | 0 to 25471 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC*1 | 0 to 25471 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 7FFFH | — |
| | SM | 0 to 2047 | — |
| Word device | D | 0 to 421887 | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | W | 0H to 66FFFH | The maximum number of points when D (extended data register) or W (extended link register) is set. |
| | TN*1 | 0 to 25471 | — |
| | STN*1 | 0 to 25471 | — |
| | CN*1 | 0 to 25471 | — |
| | SW | 0H to 6FFFH | — |
| | SD | 0 to 2047 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 393215 | Data is read or written following the file register settings of the communication destination. |

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

## ■When the communication destination is the MELSEC iQ-F (built-in Ethernet)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0 to 1777 | Octal notation is used. |
| | Y | 0 to 1777 | Octal notation is used. |
| | M | 0 to 32767 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F*1 | 0 to 32767 | — |
| | B | 0H to 7FFFH | — |
| | TS/TC*1 | 0 to 1023 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC*1 | 0 to 1023 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC*1 | 0 to 1023 | This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to 7FFFH | — |
| | SM | 0 to 9999 | — |
| Word device | D | 0 to 7999 | A local device cannot be specified. |
| | W | 0H to 7FFFH | — |
| | TN*1 | 0 to 1023 | — |
| | STN*1 | 0 to 1023 | — |
| | CN*1 | 0 to 1023 | — |
| | SW | 0H to 7FFFH | — |
| | SD | 0 to 11999 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

## ■When the communication destination is the MELSEC iQ-L (built-in Ethernet)

The maximum number of points when the L16HCPU is connected. The number of points varies depending on the CPU module used.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 14065663 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | F*1 | 0 to 32767 | — |
| | B | 0H to D69FFFH | — |
| | TS/TC*1 | 0 to 781407 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | STS/STC*1 | 0 to 781407 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | CS/CC*1 | 0 to 781407 | • A local device cannot be specified.<br>• This device can be specified only when "Read" is set for "Communication Pattern". |
| | SB | 0H to D69FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 879103 | A local device cannot be specified. |
| | W | 0H to D69FFH | — |
| | SW | 0H to D69FFH | — |
| | TN*1 | 0 to 781407 | A local device cannot be specified. |
| | STN*1 | 0 to 781407 | A local device cannot be specified. |
| | CN*1 | 0 to 781407 | A local device cannot be specified. |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 819199 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

### ■When the communication destination is the MELSEC A/AnS (Ethernet module)

The maximum number of points for the A4UCPU. The number of points varies depending on the CPU module used. (📖 MELSEC Communication Protocol Reference Manual)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0 to 7FFh | — |
| | Y | 0 to 7FFh | — |
| | M | 0 to 8191, 9000 to 9255 | To specify 9000 to 9255 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used.<br>To specify 9000 to 9255, use 9000 + a multiple of 16. |
| | B | 0 to FFFh | — |
| Word device | D | 0 to 6143, 9000 to 9255 | To specify 9000 to 9255 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used. |
| | W | 0 to FFFh | — |
| | R | 0 to 8191 | — |

### ■When the communication destination is the MELSEC-FX3 (Ethernet block/adapter)

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0 to 377 | Octal notation is used. |
| | Y | 0 to 377 | Octal notation is used. |
| | M | 0 to 7679, 8000 to 8511 | To specify 8000 to 8511 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used. |
| Word device | D | 0 to 8511 | To specify 8000 to 8511 with "Communication Pattern" set to "Write", refer to the manual of the CPU module used. |
| | R | 0 to 32767 | — |

### ■When the communication destination is an SLMP-compatible device (QnA-compatible 3E frame)

Applicable devices differ depending on the device. For applicable devices, refer to the manual for the device used.
The device types that can be set are X, Y, M, L, F[*1], B, T (TS[*1*2], TC[*1*2], TN[*1]), ST (STS[*1*2], STC[*1*2], STN[*1]), C (CS[*1*2], CC[*1*2], CN[*1]), SB, SM, D, W, SW, SD, R, and ZR. The accessible device range of the communication destination is the range which can be specified with the subcommands 00□1 and 00□0 of the SLMP (QnA-compatible 3E frame of the MC protocol).

*1 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)
*2 This device can be specified only when "Read" is set for "Communication Pattern".

## Communication Time-out Period/Communication Retry Count/Monitoring Time At Error

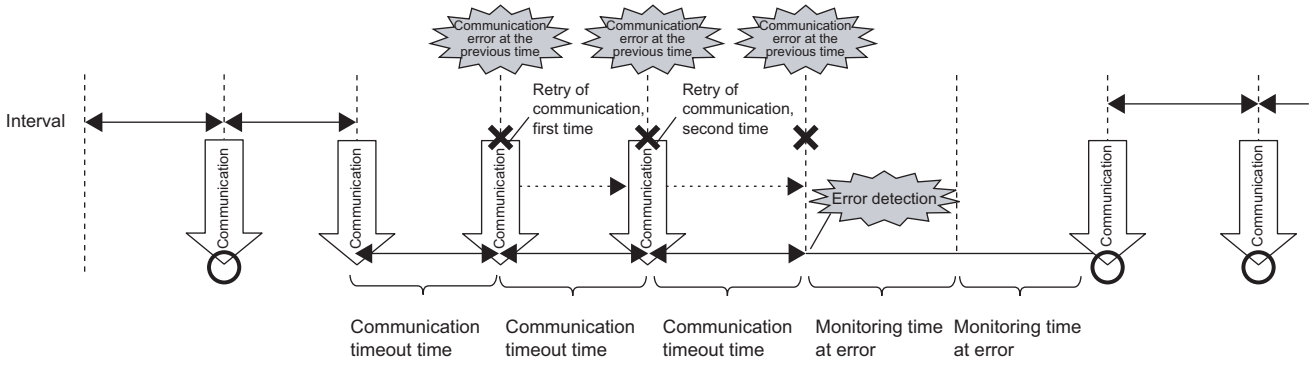
When the communication destination does not respond or returns an error response, the communication will be retried (resent) after the time-out period has passed.

When Communication Setting is "Fixed Interval", an error is detected after communications are performed for the communication retry count + 1 times, then the fixed interval communication is performed during the monitoring time at error. If the communication destination responds after these communications, the communications at the execution interval specified with the parameters are restarted.

When Communication Setting is "On Request", an error is detected after communications are performed for the communication retry count + 1 times, then simple CPU communication status becomes "2H: waiting for the request".

**Ex.**

Error detection timing when 2 is set for the communication retry count




**Point**

The error is detected if the communication destination does not respond or communications fail by when the following time has passed: (communication retry count + 1) × communication time-out period

## Latency Time

By setting the latency time, the start timing in the communications can be switched in the following cases.

- To prevent an error due to the overlap of the start timing in the communications
- To start communications after the communication destination is ready for the communications

During the latency time, "Preparing" is indicated in the communication status.

**Point**

When the latency time is set, the communications of the setting numbers where Communication Setting is set to "Fixed Interval" will not be performed during the latency time, and the communications will start after the latency time has elapsed. For the setting numbers where "Communication Setting" is set to "On Request", if Request to start communication at request is turned on during the latency time, the communications will start after the latency time has elapsed.

## Initial Communication Setting

The operation can be set when the communication destination does not respond (including that the communication destination has the different IP address or is not started) at the initial communication.
By using this setting, communications can be started at the timing when the external device is connected or powered on after starting simple CPU communications even if the external device is not connected or powered off at starting the simple CPU communications.

**Point**

This setting is disabled when the communication setting is set to "On Request". This setting is also disabled when the communication setting is set to "Fixed Interval" if simple CPU communication status is other than the following.

- 3H: Communicating
- 5H: Retry being executed
- 6H: Monitoring

**Restriction**

Before using the initial communication setting, check the versions of the CPU module used. (☞ Page 586 Added and Enhanced Functions)

## Import/Export of Settings

Import/export the simple CPU communication setting parameters from/to a CSV file.

### ■Export

1. Click the [Export] button in the "Setting Item" window under "Simple CPU Communication Setting".
2. Input the file name and click the [Save] button in the "Export to File" window.
3. Edit the exported CSV file as necessary. The description displayed the "Setting Item" window in "Simple CPU Communication Setting" can be edited.

**Point**

The following describes the detailed format in the CSV file.

- The file format is Unicode (UTF-16 Little Endian with BOM).
- "\t" (tab) delimits between the items.
- Each item enclosed in double quotation marks (").
- If the item includes double quotation marks (), double quotation marks (") in the items are displayed as two double quotation marks ("").
- A return is set at the end of the line. The return code is CR+LF.

### Precautions

In the CSV file, a string displayed in the "Setting Item" window under "Simple CPU Communication Setting" may be replaced with another string. The following table lists the items to be replaced with another string.

| Item | Engineering tool | CSV file |
|---|---|---|
| IP Address Input Format*1 | Decimal | DEC |
| | Hexadecimal | HEX |
| Device Assignment Method*2 | Points/Start | Points/Start |
| | Start/End | Start/End |
| Initial Communication Setting | With check for the initial communication setting | Enable |
| | Without check for the initial communication setting | Disable |
| Communication Pattern | Read | Read |
| | Write | Write |
| Communication Setting | On Request | 0 |
| | Fixed Interval | 1 |
| Communication Destination Setting: Device Type | MELSEC iQ-R (Built-in Ethernet) | 16 |
| | MELSEC iQ-L (Built-in Ethernet) | 20 |
| | MELSEC iQ-F (Built-in Ethernet) | 19 |
| | MELSEC-Q (Built-in Ethernet) | 17 |
| | MELSEC-L (Built-in Ethernet) | 18 |
| | MELSEC-Q (Ethernet Module) | 22 |
| | MELSEC-L (Ethernet Module) | 23 |
| | MELSEC-A/AnS (Ethernet Module) | 25 |
| | MELSEC-FX3 (Ethernet Block/Adapter) | 26 |
| | SLMP Corresponding Device (QnA Compatible 3E Frame) | 30 |
| Target PLC No. | Not Specified | 0 |
| | PLC No.1 to 8 | 1 to 8 |

*1 The IP address is imported in positional notation of the IP address input format of the CSV file.

*2 The device range (number of points/start number/end number) of the transfer source device is set in the device assignment method of the CSV file.

### ■Import

1. Click the [Import] button in the "Setting Item" window under "Simple CPU Communication Setting".
2. Select the file name and click the [Save] button in the "Import to File" window.
3. Click the [OK] button.
4. The data in the CSV file is reflected in the "Setting Item" window under "Simple CPU Communication setting".

**Point**

For the file format at import, follow the following format.

- "\t" (tab) can be used as a delimiter between items.
- Enclose items including delimiters and return codes in double quotation marks.

#### Precautions

- When imported, all the original settings are cleared. Export and save the settings as necessary.
- Do not insert or delete lines or change the value of the item to an invalid value in the CSV file. An error message is displayed and the settings cannot be imported.
- Import the CSV file whose data has been set from the 10th line onward.

## Operations when the communication setting is "Fixed Interval"

### ■Operation

After the system is powered off and on or the CPU module is reset, communications start.

After data is sent to the communication destination, if no response is received from the communication destination or an error occurs (or an error response is received), the system makes retries when the timeout time has elapsed. (☞ Page 242 Communication Time-out Period/Communication Retry Count/Monitoring Time At Error)

**Point**

- After the system is powered off and on or the CPU module is reset, the system starts the preparation processing. If an error occurs in the preparation processing, data will not be sent or received. In this case, the CPU buffer memory area of Simple CPU communication status stores AH (Communications impossible).
- When the latency time is set, the communications will start after the latency time has elapsed. (☞ Page 242 Latency Time)
- The operations in the initial communication vary depending on the setting details of the initial communication setting. When the initial communication setting is not set and no response is received from the communication destination, an error occurs and data are not sent or received. (The fixed interval communication is not performed during the monitoring time at error.) *1 (☞ Page 243 Initial Communication Setting)
- If multiple setting numbers are set for the same communication destination, the operations are as follows: When Communication Setting is set to "Fixed Interval" and retries occur in the processing of one setting number, communications on the other setting numbers set for the same communication destination will not be performed at the execution interval. When Communication Setting is set to "On Request", communications cannot be performed while the CPU buffer memory area of Simple CPU communication status is 6H (monitoring).

*1 For the RnPCPU, the initial communication setting is already set (enable) by default. The setting cannot be changed because "Continue Communicating Until Receiving Response" in the setting item is selected and grayed out.
Communications start once the communication destination device is connected (powered on).

## Operations when the communication setting is "On Request"

### ■Operation

After the system is powered off and on or the CPU module is reset, communications start when the CPU buffer memory area of Request to start communication at request is turned on. After the data are sent and received, the system automatically turns off the CPU buffer memory area of Request to start communication at request, and changes Simple CPU communication status to 2H (waiting for the request).

After data is sent to the communication destination, if no response is received from the communication destination or an error occurs (or an error response is received), the system makes retries when the timeout time has elapsed. (☞ Page 242 Communication Time-out Period/Communication Retry Count/Monitoring Time At Error)

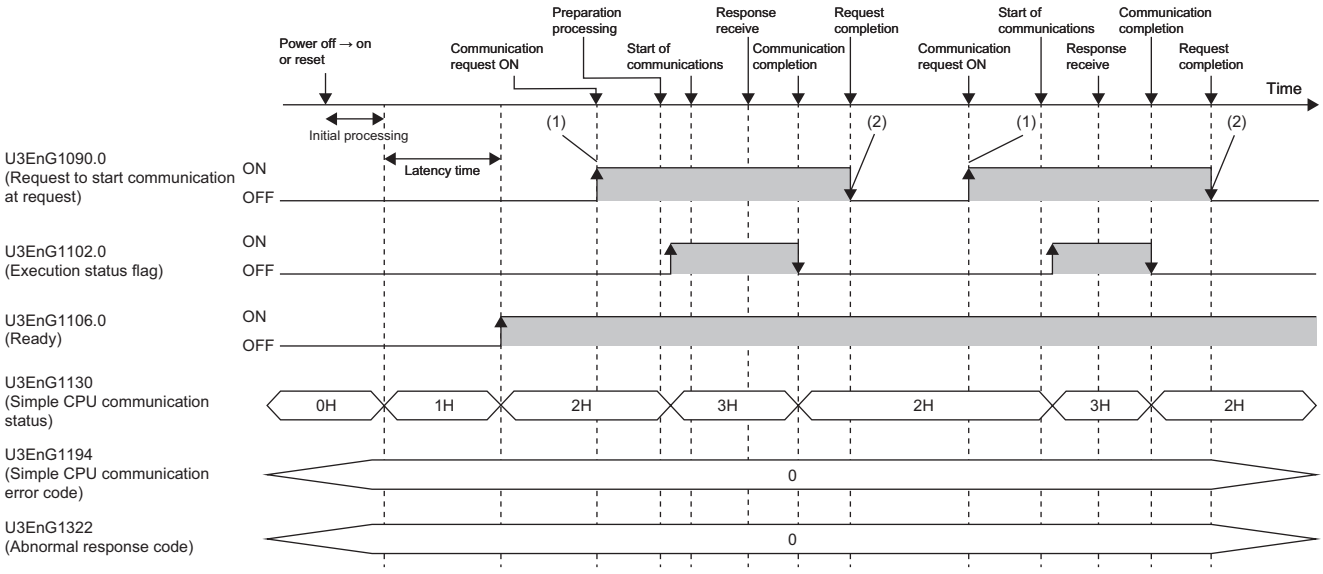
**Point**

- After the system is powered off and on or the CPU module is reset, the system starts the preparation processing. If an error occurs in the preparation processing, data will not be sent or received. In this case, the CPU buffer memory area of Simple CPU communication status stores AH (Communications impossible).
- When the latency time is set, the communications will start after the latency time has elapsed. During the latency time, the CPU buffer memory area of Request to start at request is ignored even when it is turned on. (☞ Page 242 Latency Time)
- During the initial execution of communications*1, the processing time is longer because it includes the preparation processing, such as checking the availability of the communication destination.
- After Request to start communication at request is turned on, another request will be ignored even when the CPU buffer memory area of Request to start communication at request is turned off and on by a program.
- The updates for the following operations do not occur simultaneously: when the execution status flag is turned on, when Request to start communication at request is turned on, and when Simple CPU communication status becomes 3H (communicating). The updates for the following operations may not occur simultaneously if the scan time is long: when the execution status flag is turned off, when Request to start communication at request is turned off, and when Simple CPU communication status becomes 2H (waiting for the request).
- If the scan time is long, the on and off of the execution status flag are updated simultaneously, which may disable the program to detect the flags.
- When multiple setting numbers are set for the same communication destination, if an error is detected in one of the settings, the error will also occur in the other setting numbers set for the same communication destination. In addition, when "Communication Setting" of the same communication destination is set to "Fixed Interval", communications cannot be performed while the CPU buffer memory area of Simple CPU communication status is 6H (monitoring).

*1 This refers to any initial executions with the communication setting "On Request" where the same communication destination is set, or the execution immediately following one that completed with an error. In addition, this also refers to the execution under the following conditions: Another setting number set for the same communication destination with its communication setting "Fixed Interval" exists, and all the "Fixed Interval" settings are in the stop status. (This does not apply when the communication setting "Fixed Interval" is 3H (communicating).)

Ex.

Operation of Request to start communication at request of setting No.1




(1) The buffer memory area is turned on by the program.

(2) The buffer memory area is turned off by the system.

## Operation during the simple CPU communication

### ■Operation of the own station

Since the own station operates without depending on the operating status of the CPU module, the simple CPU communication is performed even when the CPU module is in the STOP state because of stop error. When an initial communication error occurs, communications of the corresponding setting numbers are stopped and ones of the setting numbers, where no error occurs will operate.

If the communication destination responds at the timing of the END processing of the CPU module, data is transferred to the system area for the simple CPU communication.

### ■Operation of the communication destination

The operation differs depending on the communication destination.

| Communication destination | Data operation in one setting |
|---|---|
| MELSEC iQ-R (Built-in Ethernet) | The data in one setting is sent simultaneously, except under the following conditions:<br>• A bit device (TS, TC, STS, STC, CS, and CC) and a word device are specified simultaneously.*1 (In this case, the data are sent at different timing.) |
| MELSEC MX Controller (MX-R Model) (Built-in Ethernet) | |
| MELSEC MX Controller (MX-F Model) (Built-in Ethernet) | |
| MELSEC-Q (Built-in Ethernet) | |
| MELSEC-L (Built-in Ethernet) | |
| MELSEC iQ-F (Built-in Ethernet) | |
| MELSEC iQ-L (Built-in Ethernet) | |
| MELSEC iQ-R (Ethernet Module) | |
| MELSEC-Q (Ethernet Module) | The data specified with a bit device or a word device in one setting are sent simultaneously.<br>However, if both a bit device and a word device are set for the same setting number, the bit device and word device are sent at different timing. |
| MELSEC-L (Ethernet Module) | |
| MELSEC-A/AnS (Ethernet Module) | The data are sent simultaneously when the data size in one setting is as follows*1:<br>• When "Write" is set to "Communication Pattern": Bit device of 640 points or less, word device of 256 points or less<br>• When "Read" is set to "Communication Pattern": Bit device of 2048 points or less, word device of 256 points or less |
| MELSEC-FX3 (Ethernet Block/Adapter) | The data are sent simultaneously when the data size in one setting is as follows*1:<br>• When "Write" is set to "Communication Pattern": Bit device of 160 points or less, word device of 64 points or less<br>• When "Read" is set to "Communication Pattern": Bit device of 512 points or less, word device of 64 points or less |
| SLMP Corresponding Device (QnA Compatible 3E Frame) | The data specified with a bit device or a word device in one setting are sent simultaneously.<br>However, if both a bit device and a word device are set for the same setting number, the data of each device is communicated at different timing.*1 |

*1 The order of a word device and a bit device for read/write is as follows:
·When "Communication Pattern" is set to "Read": Word device → Bit device
·When "Communication Pattern" is set to "Write": Bit device → Word device

# Checking the simple CPU communication status



The simple CPU communication status can be checked with the buffer memory or diagnostic functions.

## Checking with the buffer memory

The simple CPU communication status can be checked with the storage status of the corresponding setting number in the following CPU buffer memory areas.

| Item | | Address | Remarks |
|---|---|---|---|
| Request to start communication at request | | U3En\G1090 to 1093 | 1090.0: Setting No.1 to 1093.F: Setting No.64 |
| Request to stop fixed interval communication | | U3En\G1094 to 1097 | 1094.0: Setting No.1 to 1097.F: Setting No.64 |
| Request to restart fixed interval communication | | U3En\G1098 to 1101 | 1098.0: Setting No.1 to 1101.F: Setting No.64 |
| Execution status flag | | U3En\G1102 to 1105 | 1102.0: Setting No.1 to 1105.F: Setting No.64 |
| Ready | | U3En\G1106 to 1109 | 1106.0: Setting No.1 to 1109.F: Setting No.64 |
| Simple CPU communication status | 0H: Unset | U3En\G1130 to 1193 | 1130: Setting No.1 to 1193: Setting No.64 |
| | 1H: Preparing | | |
| | 2H: Waiting for the request | | |
| | 3H: Communicating | | |
| | 4H: Communication stop | | |
| | 5H: Retry being executed | | |
| | 6H: Monitoring | | |
| | AH: Communications impossible | | |
| Simple CPU communication error code | | U3En\G1194 to 1257 | 1194: Setting No.1 to 1257: Setting No.64 |
| Execution interval (current value) | | U3En\G1258 to 1321 | 1258: Setting No.1 to 1321: Setting No.64 |
| Abnormal response code | | U3En\G1322 to 1385 | 1322: Setting No.1 to 1385: Setting No.64 |

For details on the values to be stored in the CPU buffer memory, refer to the following.

☞ Page 496 CPU module (built-in Ethernet port part)

**Point**

When the "MELSEC-A/AnS (Ethernet Module)" or "MELSEC-FX3 (Ethernet Block/Adapter)" is set to the communication destination, if an error response is received from the communication destination, the buffer memory area of Simple CPU communication error code stores the error code, CFB4H, and the buffer memory area of Abnormal response code stores the codes returned from the external device.

### ■Request to stop fixed interval communication/Request to restart fixed interval communication

When the simple CPU communication status of the corresponding setting number in the CPU buffer memory is either of the following, communications can be stopped by turning off and on the stop request of the corresponding setting number.

- 3H: Communicating
- 5H: Retry being executed
- 6H: Monitoring

**Point**

- This setting is disabled when the communication setting is set to "On Request". This setting is also disabled if Simple CPU communication status is other than the above when the communication setting is set to "Fixed Interval".
- When the stop request is turned off and on, the communications will stop at the timing of sending data in the next execution interval. Therefore, if the communication destination does not respond with the stop request accepted, the communications will stop after the communication time-out period has passed. When the retry is being executed, the communications will stop after the communication time-out period has passed without depending on the retry count.
- When an error occurs during stop in another setting numbers set for the same communication destination, the value will not be stored in the simple CPU communication error code of the setting number itself.

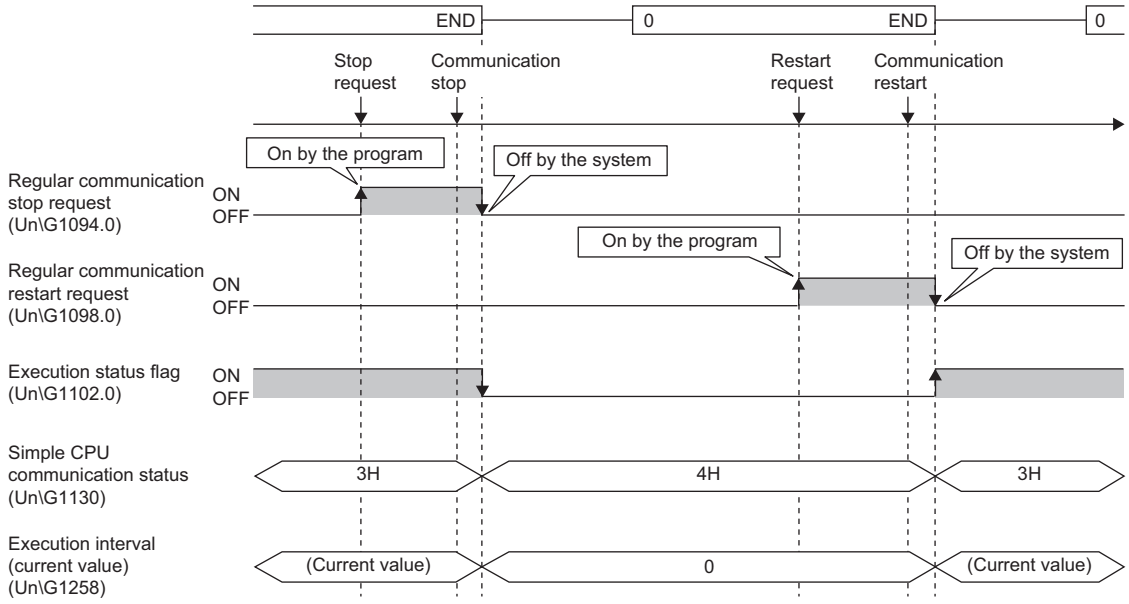
When the simple CPU communication status of the corresponding setting number in the CPU buffer memory is "4H: Communication stop", communications can be restarted by turning off and on the restart request for the corresponding setting number. (If the status is other than "4H: Communication stop", the restart request is invalid.)

**Point**

While another setting number that is set for the same communication destination has an error, the communication cannot be restarted.

Ex.

The operation at communication stop/restart request for the setting No.1




## Checking with "simple CPU communication diagnostics"

The communication status of the simple CPU communication function can be checked with "Simple CPU Communication Diagnostics".

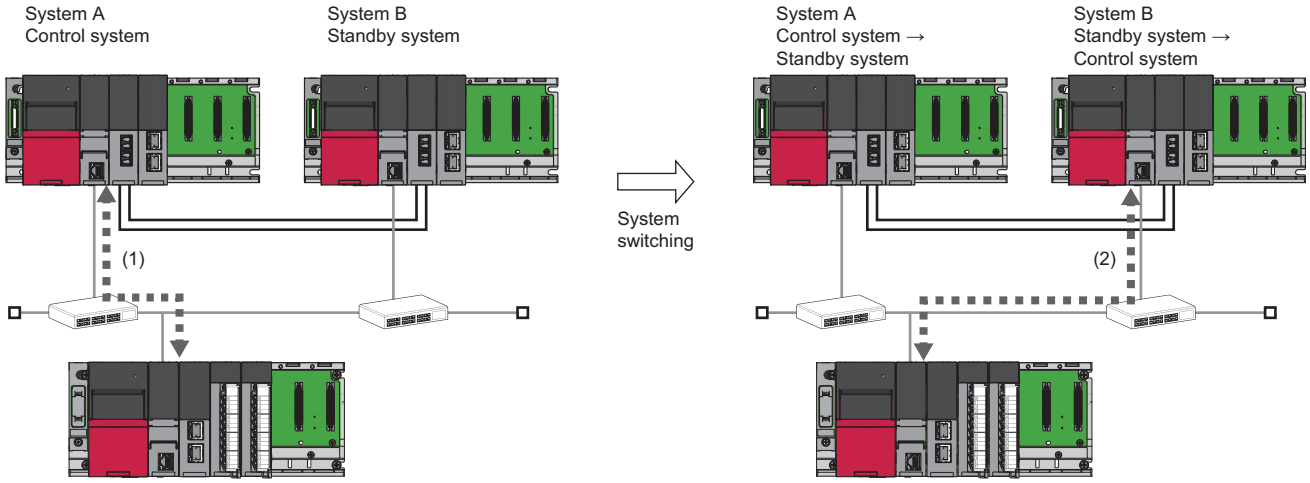
For details, refer to the following.

☞ Page 434 Simple CPU communication diagnostics

# Simple CPU communication in a redundant system



For the Process CPU (redundant mode), the control system CPU communicates with a communication destination on a 1:1 (one-to-one) basis to send/receive device data. (The standby system does not communicate with external devices.)
For system switching, communication by the old control system is interrupted, and the new control system connects to the communication destination to continue communication.




(1) "IP Address" (system A) of "Own Node Settings" is used to communicate.
(2) "System B IP Address" of "Redundant System Settings" is used to communicate.

**Point**

The CPU module (built-in Ethernet port part) cannot issue a system switching request.
To execute system switching when a communication error is detected, the communication error should be detected by a program that will execute the system switching instruction. For details on sample programs, refer to the following.
MELSEC iQ-R Process CPU Module User's Manual

## Setting procedure

### ■Own station Ethernet setting

For the Process CPU (redundant mode), set "IP Address" (system A) of "Own Node Settings" in "Basic Settings" and "System B IP Address" of "Redundant System Settings" in "Application Settings".

Page 327 Own Node Settings
Page 416 Redundant System Settings

### ■External device Ethernet setting

To set a redundant system IP address using the external device Ethernet settings (such as the SLMP setting and security setting), set both the system A IP address and system B IP address.

### ■Simple CPU communication setting

Refer to the following for this setting.

Page 229 Setting procedure
Page 345 Simple CPU communication setting (CPU module (built-in Ethernet port part))

### ■When the communication destination is the MELSEC-FX3 (Ethernet block/adapter) or MELSEC-A/AnS (Ethernet module)

For the MELSEC-FX3 (Ethernet block/adapter) or MELSEC-A/AnS (Ethernet module), the same port number cannot be set for different IP addresses. Therefore, for the communication settings, set a different port number to each IP address.

Ex.

Communication destination communication setting

| | | Protocol | Open System | Host Station Port No. | Destination IP Address | Destination Port No. |
|---|---|---|---|---|---|---|
| For the system A → | 1 | UDP | MC Protocol | 3001 | 192.168. 3. 39 | 2001 |
| For the system B → | 2 | UDP | MC Protocol | 3002 | 192.168. 3. 41 | 2002 |
| | 3 | | | | | |
| | 4 | | | | | |

Set communication separately for system A and system B, according to the communication settings of the external device.

Ex.

Process CPU (redundant mode) simple CPU communication setting

| Setting No. | Communication Pattern | Communication Setting: Execution Interval(ms) | | Communication Destination (IP Address) Source | -> | Destination |
|---|---|---|---|---|---|---|
| 1 | Read | Fixed | 100 | MELSEC-FX3 (Ether)(192 . 168 . 3 . 50) | -> | Host Station(192 . 168 . 3 . 39) |
| 2 | Read | Fixed | 100 | MELSEC-FX3 (Ether)(192 . 168 . 3 . 50) | -> | Host Station(192 . 168 . 3 . 39) |
| 3 | | | | | -> | |

Setting No.1 (setting for the system A)

| | Item | Setting |
|---|---|---|
| | IP Address Input Format | DEC |
| | Device Type | MELSEC-FX3 (Ethernet Block/Adapter |
| | IP Address | 192 . 168 . 3 . 50 |
| | TCP/UDP | UDP |
| | Port No. | 3001 |
| (1) | Host Station Port No. | 2001 |
| | Options(Hexadecimal) | |

Setting No.2 (setting for the system B)

| | Item | Setting |
|---|---|---|
| | IP Address Input Format | DEC |
| | Device Type | MELSEC-FX3 (Ethernet Block/Adapter |
| | IP Address | 192 . 168 . 3 . 50 |
| | TCP/UDP | UDP |
| | Port No. | 3002 |
| (1) | Host Station Port No. | 2002 |
| | Options(Hexadecimal) | |

(1) When specifying the same communication destination, set different own station port numbers.

After the above setting operation, communication is performed using setting No.1 when system A is the control system, and setting No.2 when system B is the control system.

## Operation

For the Process CPU (redundant mode) simple CPU communication, communication with external devices is performed by the control system only, regardless of the operation mode.

- Communication is not performed if the system is not yet assigned or by the standby system.
- Communication stops temporarily during system switching. After system switching is completed, the new control system restarts communicating with the external device.
- Communication is performed by connecting to the external device using the system A IP address and system B IP address.

**Point**

For a program that was executed in both systems, to write the operation results of execution in the standby system of the Process CPU (redundant mode) to the communication destination, write the operation results to the control system device by executing the CONTWR(P) instruction with the program executed in both systems on the standby system, and then write the results to the communication destination device from the control system by using the simple CPU communication function.

## ■Operation at power-on (system not yet assigned)

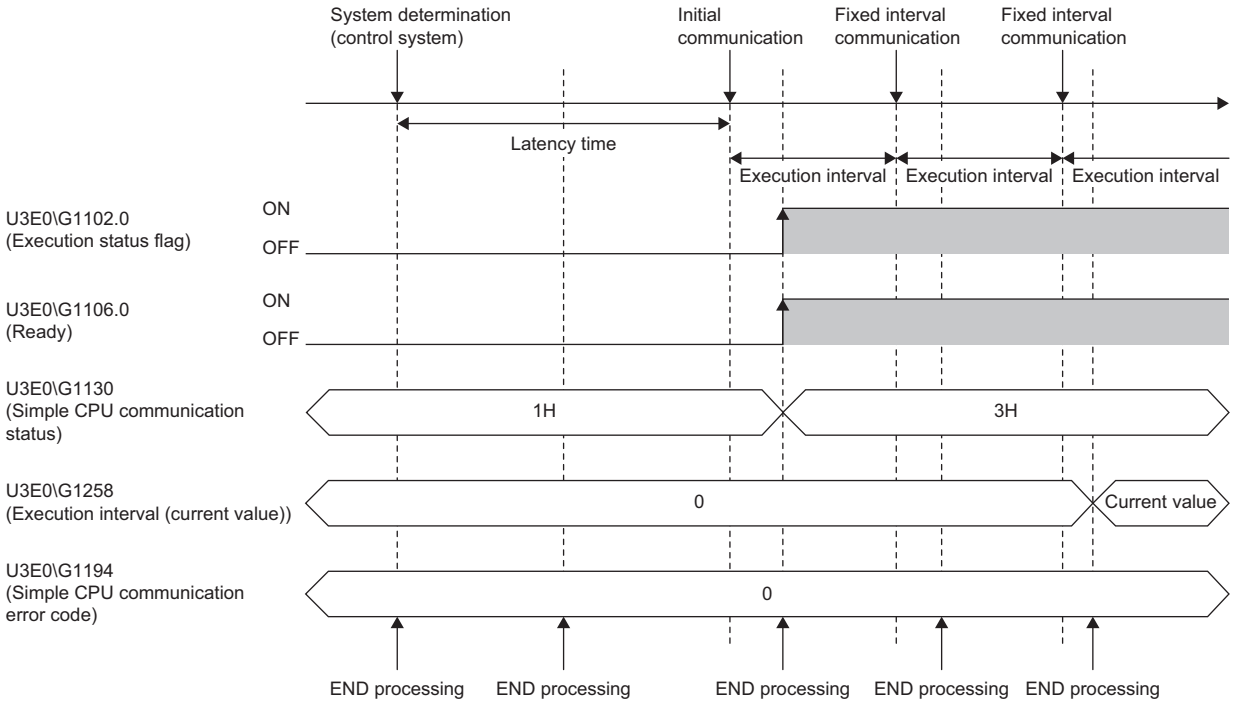
At power-on, after the Ethernet port is initialized according to the simple CPU communication setting parameters, the state of preparing (waiting for a system to be assigned) arises and continues until a system is assigned, and simple CPU communication does not start for that duration. Measurement of the latency time does not start until a system is assigned.

## ■Operation after a system is assigned

After a system is assigned, the control system starts measuring the latency time, and after the latency time elapses, it starts the simple CPU communication.

Ex.

Operation of the control system (for setting No.1)




The standby system does not start simple CPU communication, even after the latency time elapses, and stands by until it becomes the control system due to system switching.

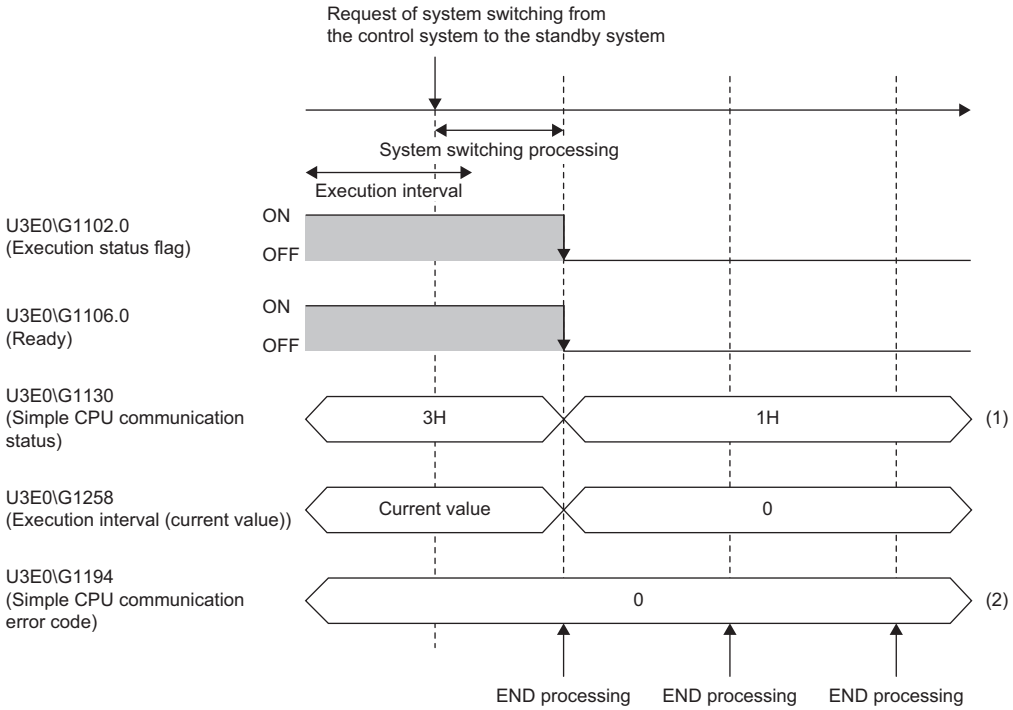
## ■Operation during system switching

When system switching occurs, the simple CPU communication of the old control system is interrupted, and simple CPU communication of the new control system is started, to continue communication with the external device. (However, if system switching fails, the simple CPU communication of the control system is not interrupted.)

In addition, even if the simple CPU communication status is AH (communications impossible), as long as communications are possible when the standby system switches to the control system, the simple CPU communication restarts.

Ex.

When the system switches from the control system to the standby system (for setting No.1)

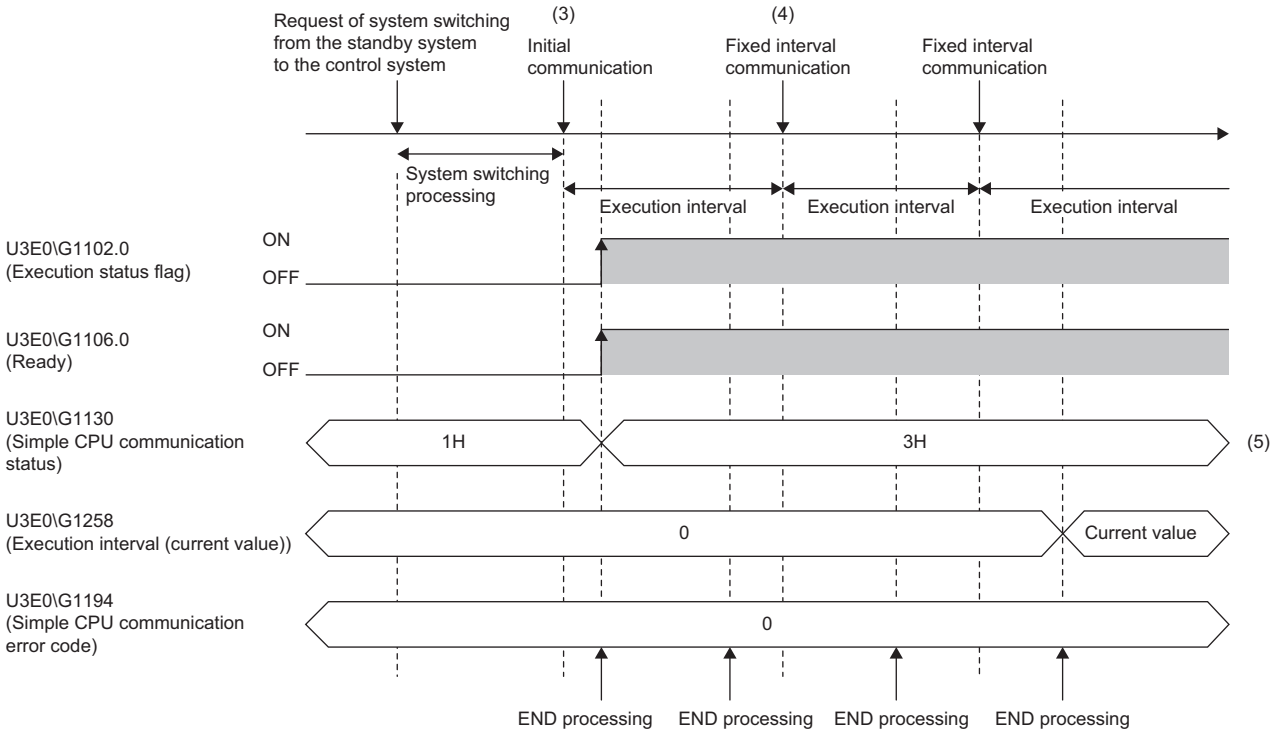



(1) When the status before system switching is 0H (unset), 1H (preparing), or AH (communications impossible), the status is not updated. Otherwise, the status becomes 1H (preparing) at the time of system switching.

(2) Diagnostic information before system switching is held.

Ex.

When the system switches from the standby system to the control system (for setting No.1)




(3) When the standby system becomes the control system while waiting for communications to start, measurement of the latency time continues, and when the latency time elapses, the simple CPU communication starts.

(4) Measurement of the execution interval for fixed interval communication starts from point 0 after system switching is completed. For that reason, the first fixed interval communication is performed after both system switching and after the set execution interval elapses.

(5) The status is updated according to the communication status of the first communication. However, when communication was stopped in the old control system, the status becomes 4H (communication stop).

## Communication stop and restart request

For the Process CPU (redundant mode), only the control system can stop communication and make a restart request. When communication is stopped or a restart request is made by the standby system or while the system is not yet assigned, the request is ignored, and the stop request/restart request for the corresponding setting No. in the CPU buffer memory is turned off automatically. If system switching occurs while communication is stopped, the stop status of the old control system is transferred to the new control system.

# Precautions

This section provides precautions for the simple CPU communication function.

## Incomplete data reception and timeout

Because the load of Ethernet communication becomes heavy during the simple CPU communication, if the other communication (MELSOFT connection or SLMP) is simultaneously performed by using UDP, the data reception of UDP may not be completed, resulting in a timeout error. Therefore, to perform other communications during the simple CPU communication, the communication using TCP is recommended.

## Execution interval setting

The execution interval (Communication Setting: "Fixed Interval") and the time until communication completion (Communication Setting: "On Request") in the simple CPU communication function will differ according to the load rate of the Ethernet line, operating status of other communication functions (such as MELSOFT connection, SLMP communication, socket communication function, CC-Link IE Field Network Basic), scan time of the communication destination, and system configuration. Thus, the set execution interval or time required for completing the communications may be increased.
To perform communications at the set execution interval, set a sufficient execution interval. If checking the execution interval is required, perform actual communications and check it with the "Simple CPU Communication Diagnostics" window.

## Duplication of own station port numbers

The simple CPU communication cannot be performed in the following cases.

- The own station port number set in the simple CPU communication and the own station port number of the CPU module set in the external device configuration are the same.
- The own station port number set in the simple CPU communication and the one set in another function such as the socket communication are the same.

Set different own station port numbers. For the setting numbers other than the one in which a communication error occurs, communications are performed after the preparation processing is completed normally.

## Module type and IP address of communication destination

Check the module type and IP address of the communication destination before setting the parameters. The simple CPU communication is not performed when the preparation processing is not completed because the communication destination does not exist or an incorrect module type or IP address is specified.

## Communication destination device

Check the type and range of the device to be read or written in the communication destination.
Especially, when "Write" is set to "Communication Pattern", the control data of the communication destination may be overwritten, resulting in malfunction.

## Remote password

The simple CPU communication is not performed when a remote password is set for the communication destination. Clear the remote password.

## Same communication destination setting

If multiple setting numbers are set for the same communication destination and an error occurs in one of the settings, the error also occurs in other setting numbers set for the same communication destination, resulting in the disabling of communications.

If an error occurs in a module set to "Fixed Interval" during communications and "6H: monitoring" in Simple CPU communication status of the setting number while both "Fixed Interval" and "On Request" are set in the "Communication Setting", the communication will not start even when Request to start communication at request of the error setting number is turned on.

However, Simple CPU communication status of the setting number, where "Communication Setting" is set to "Fixed Interval", is either "4H: communication stop" or "AH: communications impossible", the communication will start by turning on Request to start communication at request of the setting number where "Communication Setting" set to "On Request".

In addition, when the RnPCPU (in redundant mode) is used, an individual IP address is set for each Ethernet port of the system A and B, where the MELSEC-FX3 (Ethernet block/adapter) or MELSEC-A/AnS (Ethernet module) is set as the same communication destination. Therefore, an error will not be detected between such communication settings.

## File register (R, ZR)

When using the file register (R, ZR) for the device on the own station, always assign the device within the setting range. The simple CPU communication of the setting number which exceeds the setting range due to a stop error of the own station cannot be performed.

When a stop error occurs, communications continue for the setting numbers that do not exceed the setting range and those with a different communication destination.

## Special relay (SM) and special register (SD)

Do not write any data to the special relay (SM) and the special register (SD) that are set on the system side. Doing so may cause a system failure or communication failure.

## When the communication destination in a redundant system

In this case, use "Control System IP Address" of "Redundant System Settings" for the Process CPU (redundant mode) side, and set the control IP address on the "Communication Destination Setting" window in the simple CPU communication.

## Redundant system latency time

For the Process CPU (redundant mode), each system measures the latency time separately. For that reason, when the control system is started up first, then if the standby system is started and system switching occurs before the latency time elapses, regardless of the elapsed latency time for the old control system, the new control system waits before starting communications. The waiting time for communications to start may become twice the latency time at the maximum.

## Redundant system restrictions

- If system switching occurs due to an operation error in the control system program, before system switching is completed, the operation result (device value) immediately after the occurrence of the operation error may be written to the external device.
- For the built-in Ethernet of a CPU module, even if a communication error is detected, system switching by the system is not executed. To execute system switching when a communication error is detected, the communication error should be detected by a program that will execute the system switching instruction. For details on sample programs, refer to the following.

MELSEC iQ-R Process CPU Module User's Manual

## Communication with a MELSEC MX Controller with user authentication enabled

In the user authentication exemption settings for the MELSEC MX Controller (communication destination), exclude device/label read and write operations.

For the user authentication exemption settings, refer to the following.

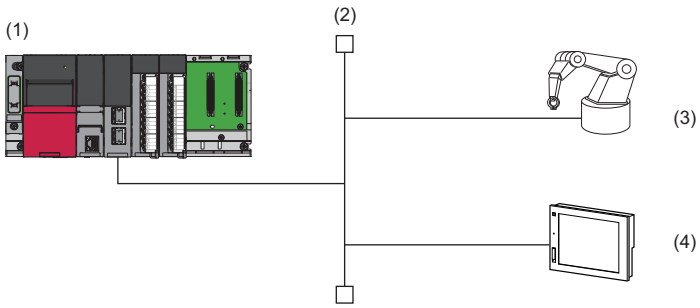
📖 Manual for the controller used

## If a device that requires a long startup time is included

Among multiple communication destinations, if a device takes a long time to start up, the period during which the simple CPU communication status of the device remains "1H: Preparing" may be longer than that of the other communication destinations. Power on the CPU module after that device has already started.

# 1.16 Simple Device Communication Function

This function allows data communications following the protocol of the Ethernet-connected communication destination only by making simple settings of the RJ71EN71 and RnENCPU (network part) from an engineering tool.
Use this function for communicating device-specific information (memory areas) with an external device (such as an industrial robot).




(1)RJ71EN71
(2)Ethernet
(3)External device (robot)
(4)External device (temperature controller)

**Restriction**

Before using the simple device communication function, check the versions of the module and engineering tool used. (☞ Page 292 Restrictions applicable depending on versions)

## Setting procedure

The following section describes the settings for using this function.

1. Register a simple device communication library file into the simple device communication library, if such a file is obtained. (☞ Page 259 Registration into simple device communication library)
2. Set the simple device communication setting. (☞ Page 260 Simple device communication setting)
3. After completing the settings, write the module parameters to the CPU module of the own station, and reset or power off and on the CPU module to start the simple device communication.

**Point**

For the simple device communication library file, please consult your local Mitsubishi representative.

### Registration into simple device communication library

1. From the menus of the engineering tool, perform registration into the simple device communication library.

[Tool] ⇨ [Register Simple Device Communication Library]

2. Select a simple device communication library file (*.sdcl) of the external device that should be newly added or whose protocol information should be updated, and then click the [Open] button.

**Point**

- When a file to be newly added is selected, a selectable external device (manufacturer, device type, and model) is added.
- When a file whose protocol information should be updated (file of the same model as the file that has been already registered) is selected, the existing registration is overwritten with new registration and the protocol is added.
- For details on the protocol for each manufacturer, refer to the following.

MELSEC iQ-R Simple Device Communication Library Reference Manual

## Simple device communication setting

**1.** Set "Use" for "To Use or Not Use Simple Device Communication" in "Simple Device Communication Setting" under "Application Settings".

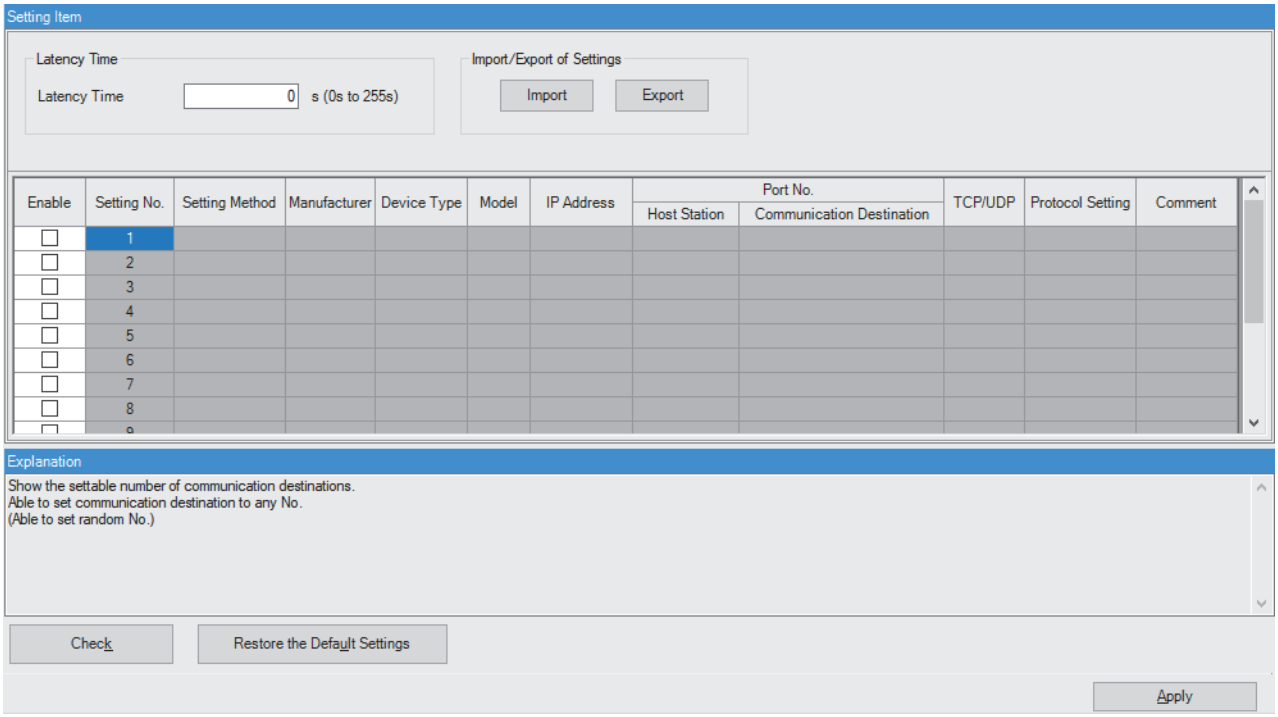
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple Device Communication Setting]

**2.** Set "Resource Setting", "Instruction Initial Value to Start/Stop Function", and "CPU Response Monitoring Timer" if necessary.

If "Resource Setting" is set to "Extension 1", set "To Use or Not Use Simple Device Communication" of the other ports to "Not Use".

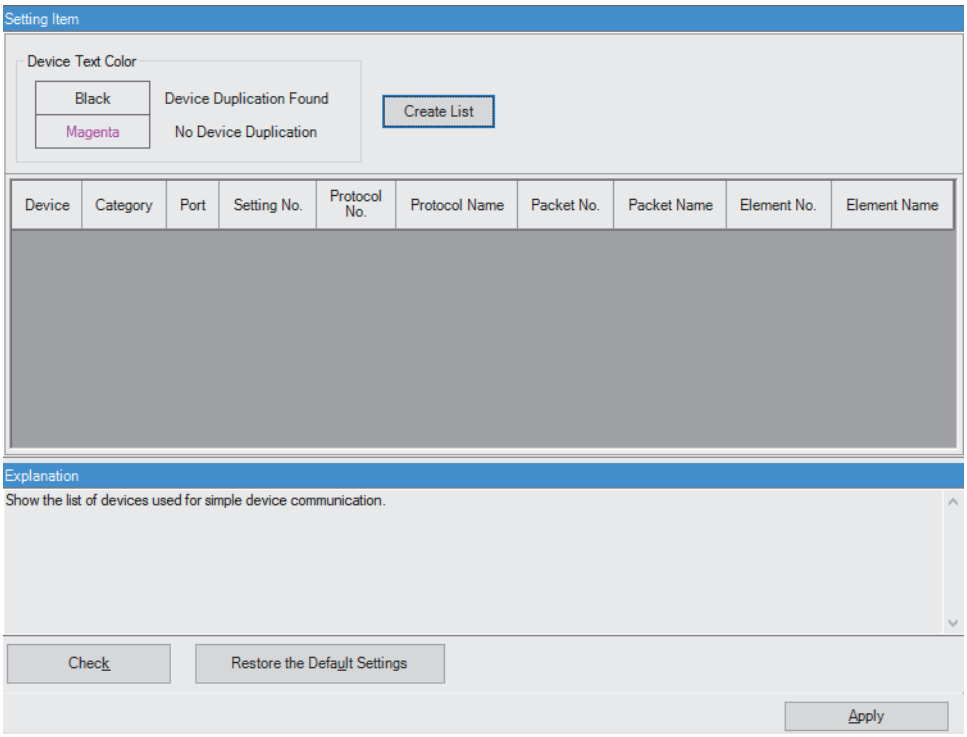
**3.** From the window that can be displayed by the following operation, set "Communication Destination Setting". (☞ Page 349 "Communication Destination Setting" window)

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple Device Communication Setting] ⇨ [Communication Destination Setting] ⇨ [Detailed Setting]



**4.** From the window that can be displayed by the following operation, check "Device List". (☞ Page 399 "Device List" window)

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple Device Communication Setting] ⇨ [Device List] ⇨ [Confirmation]



## Devices that can be specified

For devices that can be specified, the size of a device is specified in units of 16 points for a bit device and one point for a word device. Use 0 or multiples of 16 to specify the device number of a bit device.

The following table lists the maximum number of points when the NZ2MC-16MBS is connected to the R120CPU. The number of points differs depending on the CPU module used, whether to use or not an extended SRAM cassette, or the type of the extended SRAM cassette.

| Type | Applicable device | | |
|---|---|---|---|
| | Symbol | Range | Remarks |
| Bit device | X | 0H to 2FFFH | — |
| | Y | 0H to 2FFFH | — |
| | M | 0 to 161882111 | A local device cannot be specified. |
| | L | 0 to 32767 | — |
| | B | 0H to 9A61FFFH | — |
| | SB | 0H to 9A61FFFH | — |
| | SM | 0 to 4095 | — |
| Word device | D | 0 to 10117631 | A local device cannot be specified. |
| | W | 0H to 9A61FFH | — |
| | SW | 0H to 9A61FFH | — |
| | SD | 0 to 4095 | — |
| | R | 0 to 32767 | Data is read or written following the file register settings of the communication destination. |
| | ZR | 0 to 10027007 | Data is read or written following the file register settings of the communication destination. |
| | RD | 0 to 1048575 | — |
| | G | • P1: 722500 to 1246787<br>• P2: 2722500 to 3246787 | Buffer memory for the RJ71EN71 or RnENCPU (network part).<br>A G device is expressed as Un\Gn device. To search for a device, specify the Un\Gn device.<br>When the search for a G device is attempted while the RJ71EN71 or RnENCPU is not mounted, the device will be searched for with U0\Gn. No error message appears for indicating that the RJ71EN71 or RnENCPU is not mounted. |

# Checking the simple device communication status

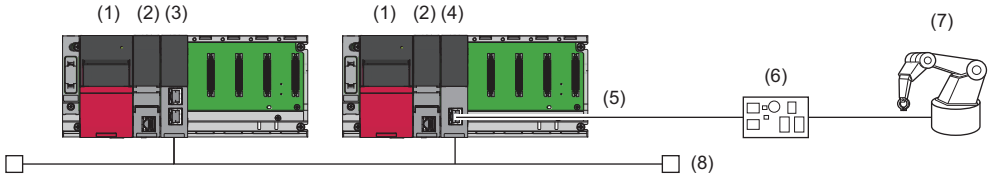
The simple device communication status can be checked with the buffer memory areas.

☞ Page 519 Simple device communication area

# Communication example

Use the simple device communication function and execute the protocol of RD: Read CPU operating status for the CR800-R to read the operating status of the CPU module.

## System configuration



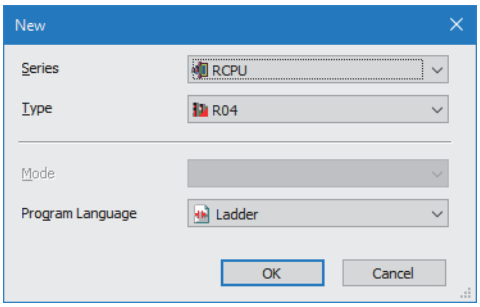

(1)Power supply module (R61P)
(2)CPU module (R04CPU)
(3)RJ71EN71 (IP address: 192.168.0.10)
(4)R16RTCPU (IP address: 192.168.0.20)
(5)SSCNET Ⅲ/H
(6)Robot controller
(7)Robot
(8)Ethernet

## Parameter settings

### ■Settings for request source

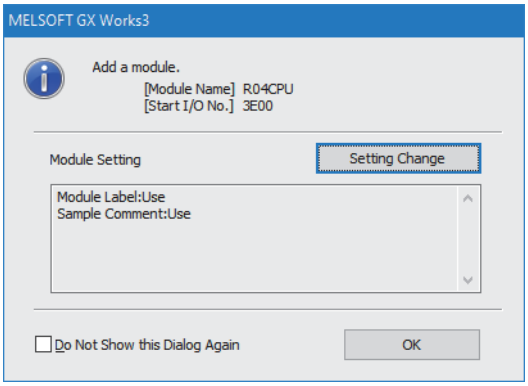
***1.*** Set the CPU module in the following window.

[Project] ⇨ [New]



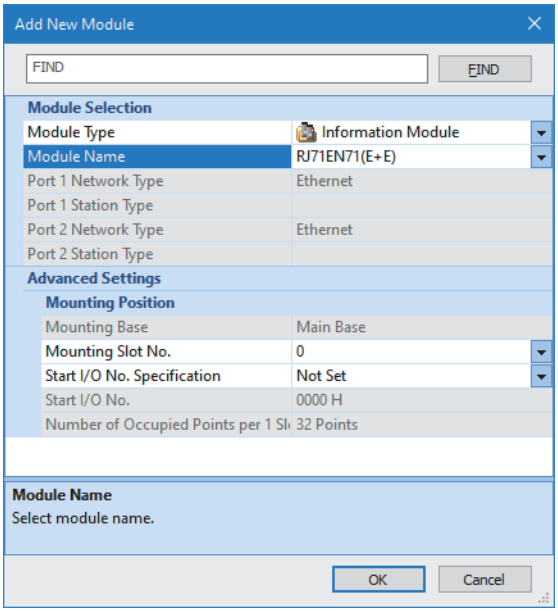

***2.*** Click the [Setting Change] button and set to use module labels.

***3.*** Click the [OK] button to add the module labels of the CPU module.

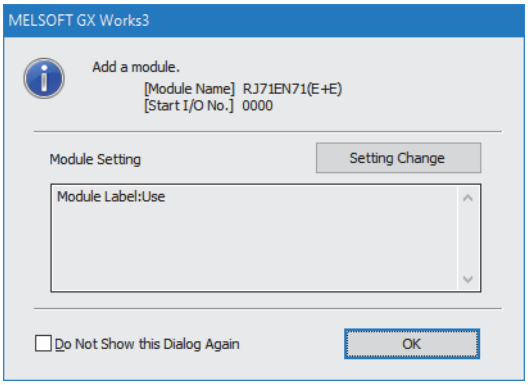


**4.** Set the RJ71EN71 as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]

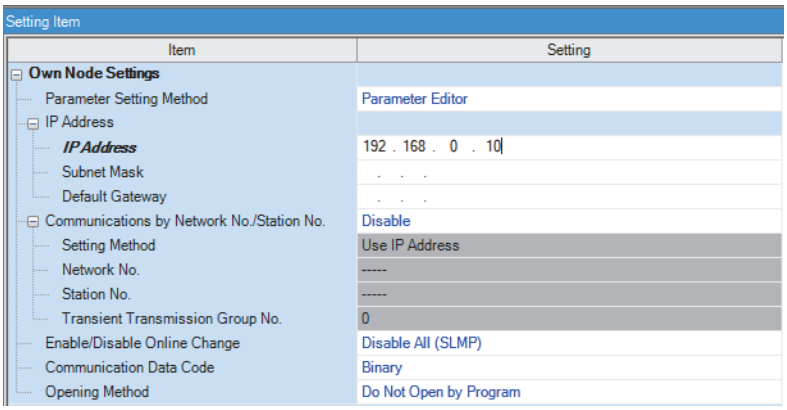



**5.** Click the [OK] button to add the module labels of the RJ71EN71.




**6.** Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [Own Node Settings]



**7.** Set the items in "Application Settings" as follows.

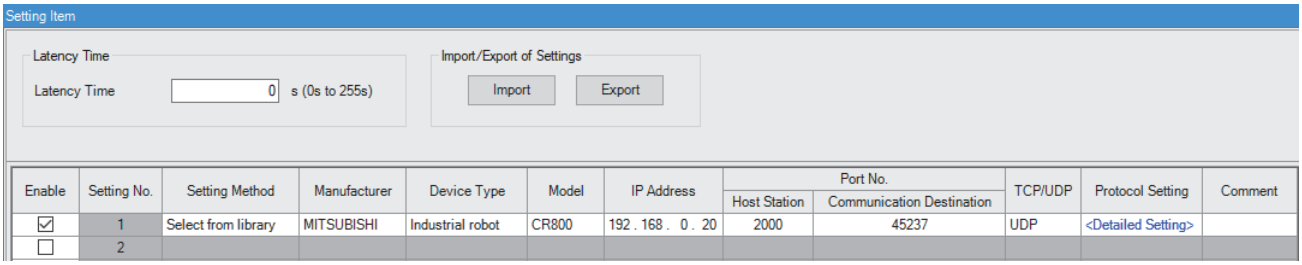
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple Device Communication Setting]

Setting Item

| Item | Setting |
|---|---|
| **Simple Device Communication Setting** | |
| To Use or Not to Use Simple Device Communication Setting | Use |
| Resource Setting | Normal |
| Instruction Initial Value to Start/Stop Function | Start |
| CPU Response Monitoring Timer | 5 |
| Unit | s |
| Communication Destination Setting | <Detailed Setting> |
| Device List | <Confirmation> |

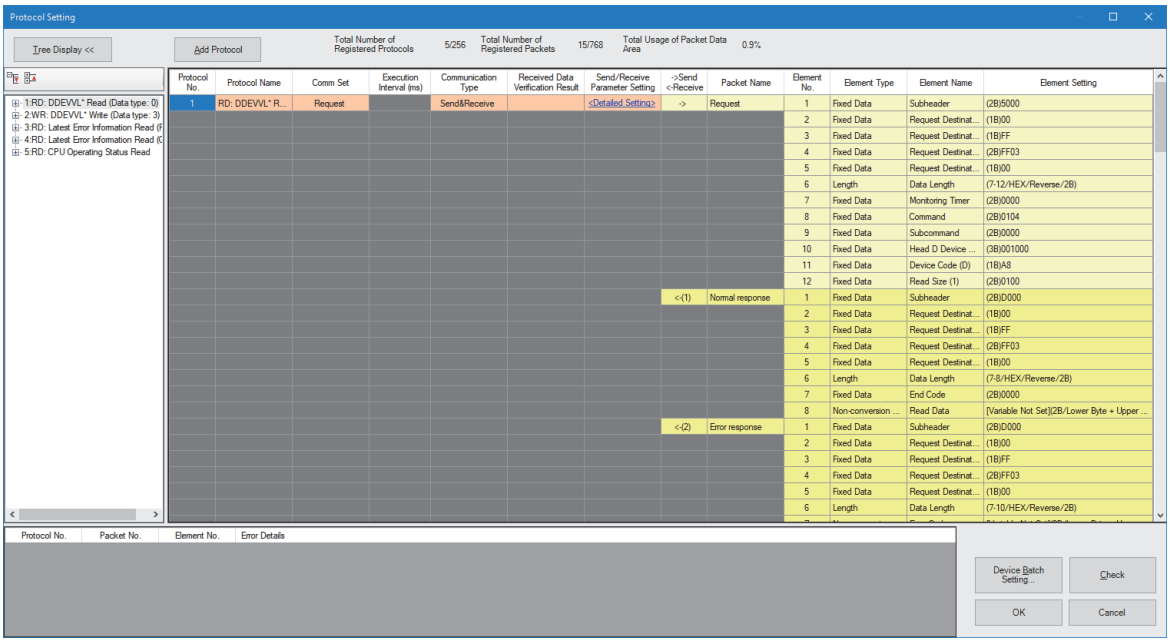
**8.** Set the items in "Communication Destination Setting" in the following window.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(E+E)] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Simple Device Communication Setting] ⇨ [Communication Destination Setting] ⇨ [Detailed Setting]



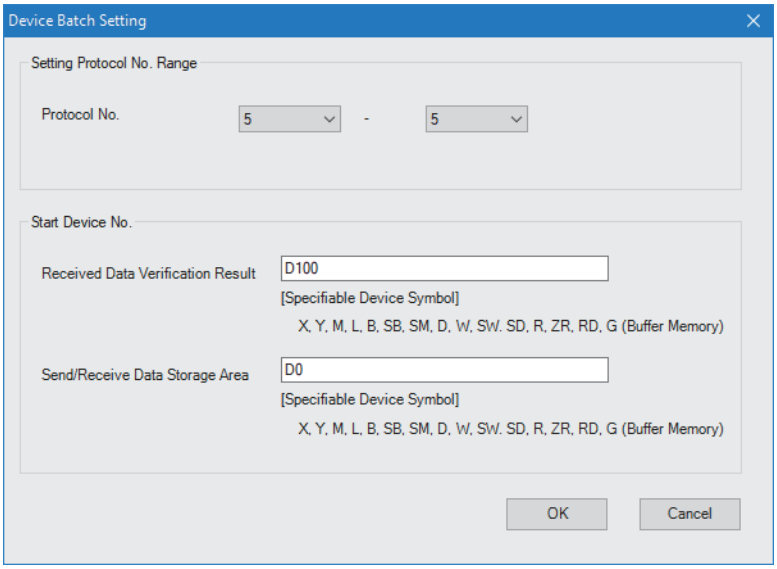

**9.** Click "Detailed Setting" on "Protocol Setting" to open the "Protocol Setting" window.

**10.** Delete protocols other than the protocol of RD: CPU Operating Status Read, and change "Comm Set" to "Request".



**11.** Click the [Device Batch Setting] button to open the "Device Batch Setting" window.

**12.** Set the items in "Device Batch Setting" in the following window.




**13.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this example, default values were used for parameters that are not shown above. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

## ■Settings for request destination

Set the parameters listed below for the communication destination.

| Parameter name | Description | Setting value |
|---|---|---|
| NETIP | IP address | 192.168.0.20 |
| NETMSK | Subnet mask | 255.255.255.0 |
| NETGW | Default gateway | 192.168.0.254 |
| SLMPPORT | SLMP server communication port number | 45237 |
| SLMPCP | SLMP server communication port protocol<br>• 0: TCP<br>• 1: UDP | 1 |
| SLMPNWNO | SLMP network number | 1 |
| SLMPNDID | SLMP station number | 1 |

**Point**

For steps to set the parameters, refer to the following.

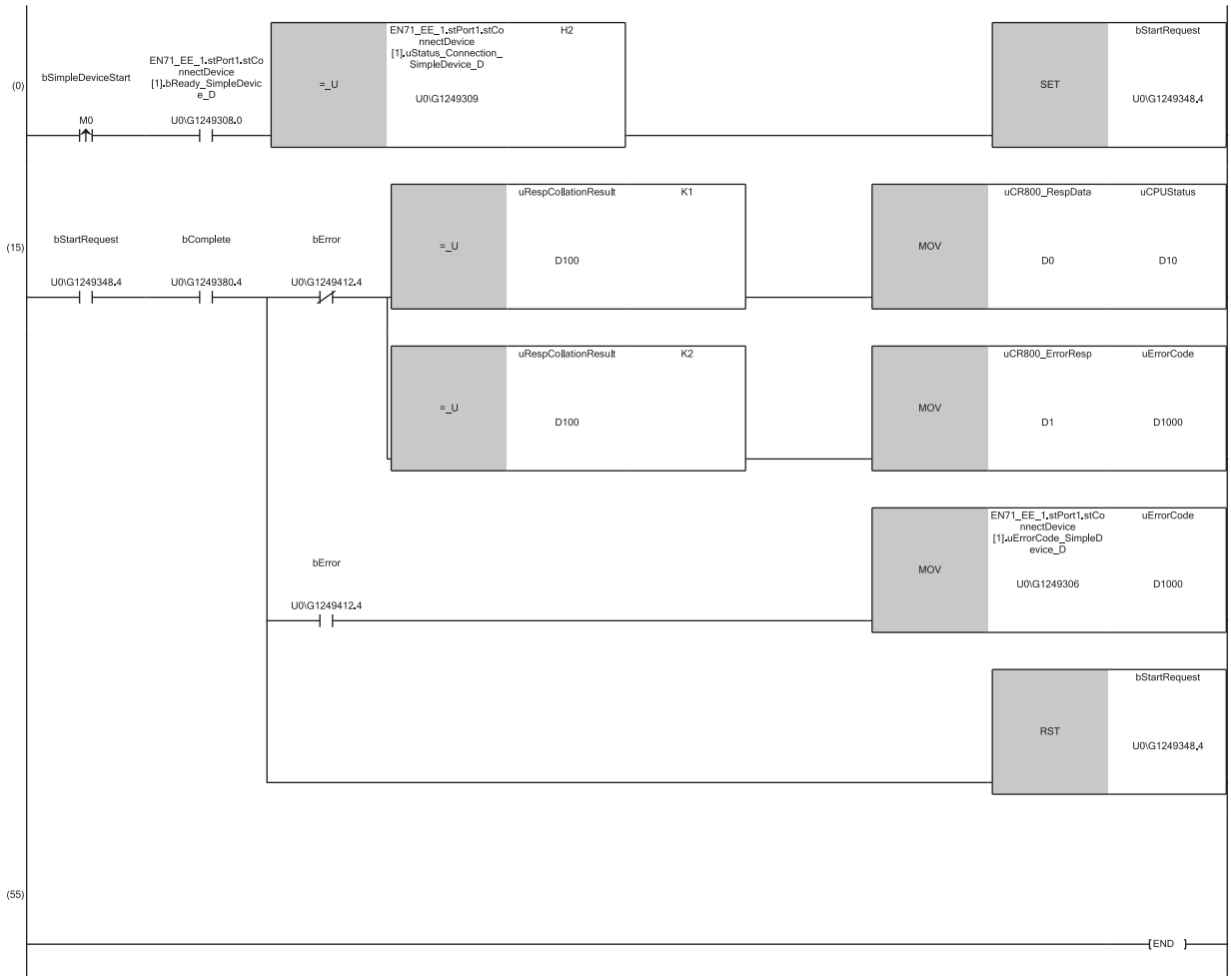
CR800 series controller, CR750/CR751 series controller Ethernet Function Instruction Manual (BFP-A3379)

## Program examples

Write the following program to the request source CPU module.

| Classification | Label name | Description | Device |
|---|---|---|---|
| Module label | EN71_EE_1.stPort1.stConnectDevice[1].bReady_SimpleDevice_D | Setting No.1 ready | Un\G1249308.0 |
| | EN71_EE_1.stPort1.stConnectDevice[1].uStatus_Connection_SimpleDevice_D | Setting No.1 connection status | Un\G1249309 |
| | EN71_EE_1.stPort1.stConnectDevice[1].uErrorCode_SimpleDevice_D | Simple device communication error code for setting No.1 | Un\G1249306 |
| Label to be defined | Define global labels as shown below: | | |

| | Label Name | Data Type | | Class | Assign |
|---|---|---|---|---|---|
| 1 | uCR800_RespData | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D0 |
| 2 | uCR800_ErrorResp | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D1 |
| 3 | uRespCollationResult | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D100 |
| 4 | bStartRequest | Bit | ... | VAR_GLOBAL | U0\G1249348.4 |
| 5 | bComplete | Bit | ... | VAR_GLOBAL | U0\G1249380.4 |
| 6 | bError | Bit | ... | VAR_GLOBAL | U0\G1249412.4 |
| 7 | bSimpleDeviceStart | Bit | ... | VAR_GLOBAL | M0 |
| 8 | uCPUStatus | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D10 |
| 9 | uErrorCode | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D1000 |




(0) When 'Ready' (Un\G1249308.0) is on and also 'Status of connection' (Un\G1249309) is 2H (waiting for the request) for setting No.1, turning on the start instruction for the simple device communication (M0) turns on 'Start request for each protocol' (Un\G1249348.4) of the protocol number 5.

(15)The existence of a response from the communication destination is judged from whether or not 'Execution completion for each protocol' (Un\G1249380.4) is turned on and also 'Error for each protocol' (Un\G1249412.4) is turned off for the protocol number 5. At this time, the response is judged as normal if the received data verification result (D100) for the protocol number 5 is 1H, and judged as error if the result is 2H.
On the other hand, if 'Error for each protocol' (Un\G1249412.4) of the protocol number 5 is also turned on, the status is judged as that an error has occurred in the simple device communication function.

# Operation

This function supports the communication methods listed below in combination of "Comm Set" and "Communication Type".

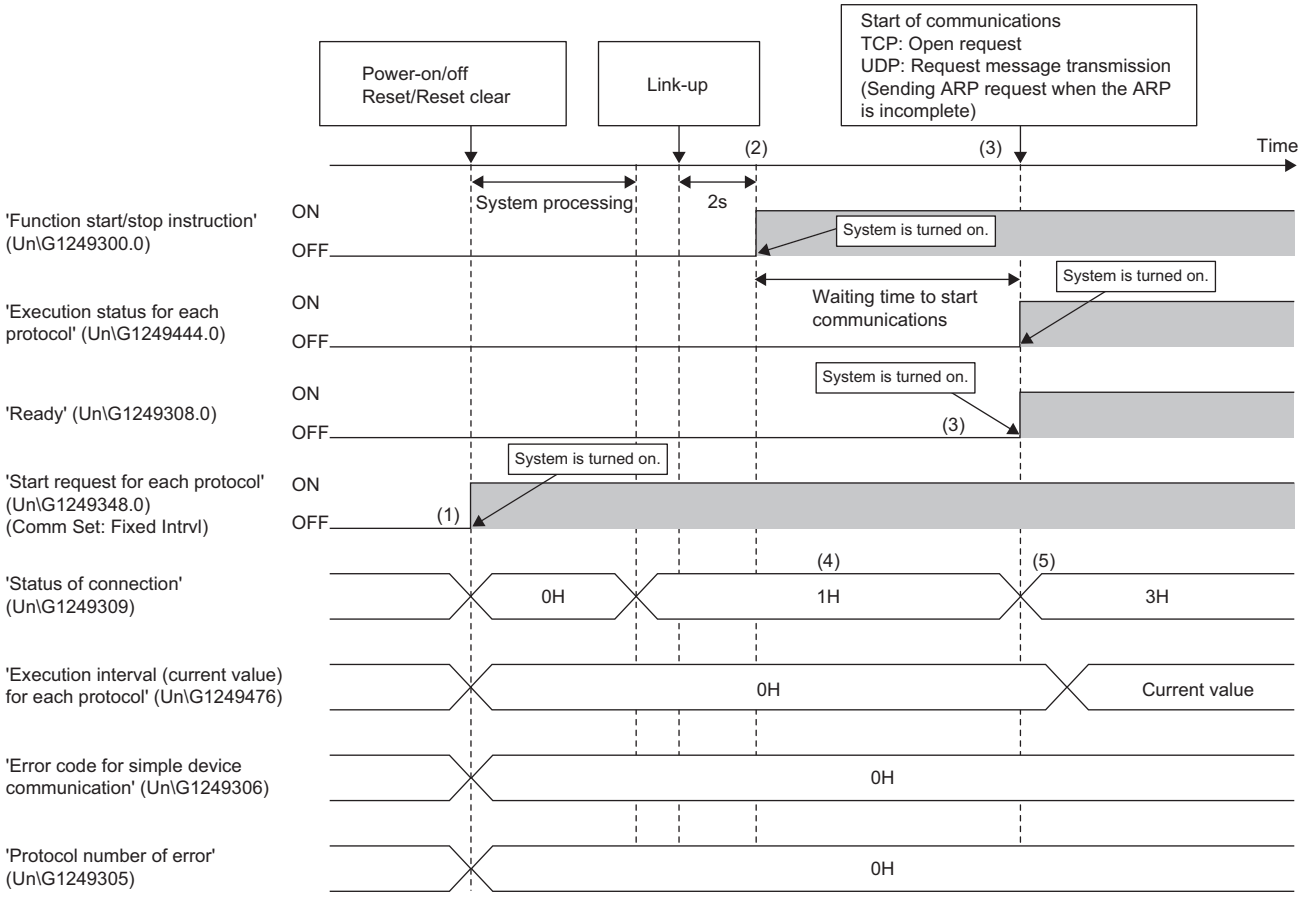
| Comm Set | Communication Type | Communication method |
|---|---|---|
| Fixed Intrvl | Send & Receive | The RJ71EN71 or RnENCPU (network part) becomes a client and communicates data with the communication destination (server) at the fixed interval. |
| Fixed Intrvl | Send Only | The RJ71EN71 or RnENCPU (network part) sends data to the communication destination at the fixed interval. |
| Fixed Intrvl | Receive Only | The RJ71EN71 or RnENCPU (network part) constantly receives send data from the communication destination. |
| Request | Send & Receive | The RJ71EN71 or RnENCPU (network part) becomes a client and communicates data with the communication destination (server) only once on a request. |
| Request | Send Only | The RJ71EN71 or RnENCPU (network part) sends data to the communication destination only once on a request. |
| Request | Receive Only | The RJ71EN71 or RnENCPU (network part) receives send data from the communication destination only once on a request. |

Unless otherwise specified, each timing chart is described using an example of the protocol number 1 of setting No.1 that is connected to P1.

Also, the timing charts whose "Communication Type" is not "Send Only" are described using an example of the case where the CPU module device is set only for "Device" of receive packets in the protocol setting.

## Operation of own station before communication starts

### ■Normal operation (when "Instruction Initial Value to Start/Stop Function" is set to "Start")




(1)The own station performs the system processing (preparation processing) to be performed after the system is powered off and on or after the system is reset and the reset state is cleared. In this case, the default value of 'Start request for each protocol' (Un\G1249348.0) is on (start) if the communication setting for each protocol is the fixed interval.

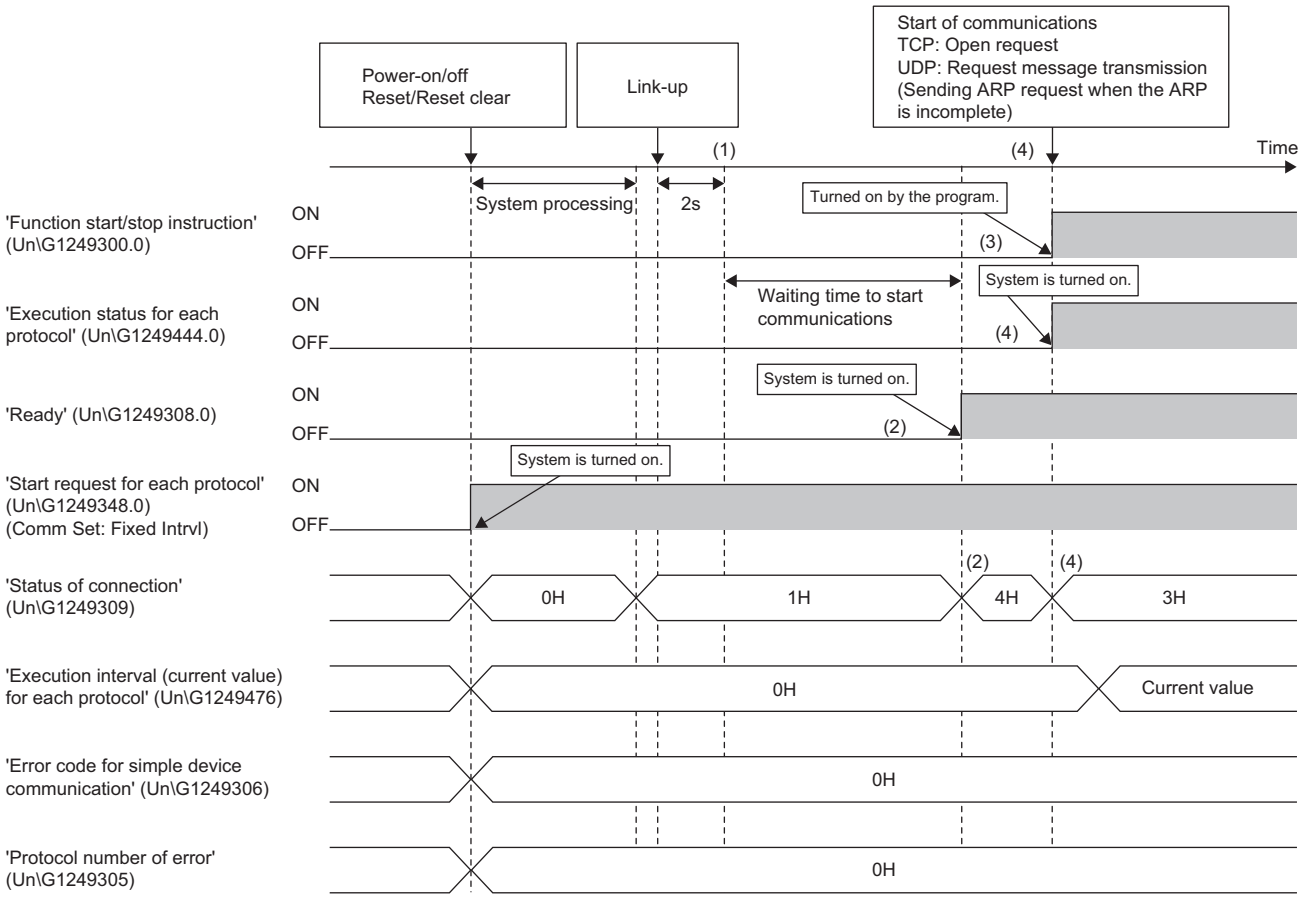
(2)When the system processing is completed, the link-up is detected. Two seconds later, the measurement of the latency time starts. When "Instruction Initial Value to Start/Stop Function" is set to "Start", 'Function start/stop instruction' (Un\G1249300.0) is turned on (enable).

(3)After the latency time has elapsed, 'Ready' (Un\G1249308.0) is turned on (ready), executing the data communication.

(4)When the system processing is completed, 'Status of connection' (Un\G1249309) becomes 1H (preparing).

(5)When the latency time has elapsed, 'Status of connection' (Un\G1249309) becomes 3H (communicating). The connection is opened at the time when 'Status of connection' (Un\G1249309) becomes 3H (communicating).

## ■Normal operation (when "Instruction Initial Value to Start/Stop Function" is set to "Stop")




(1)When the system processing to be performed after the system is powered off and on or after the system is reset and the reset state is cleared is completed, the link-up is detected. Two seconds later, the measurement of the latency time starts.
(2)After the latency time has elapsed, 'Ready' (Un\G1249308.0) is turned on (ready) and 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).
(3)At a desired timing, 'Function start/stop instruction' (Un\G1249300.0) is turned on (enable).
(4)'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed) and 'Status of connection' (Un\G1249309) becomes 3H (communicating), starting the communication with the communication destination. The connection is opened at the time when 'Status of connection' (Un\G1249309) becomes 3H (communicating).

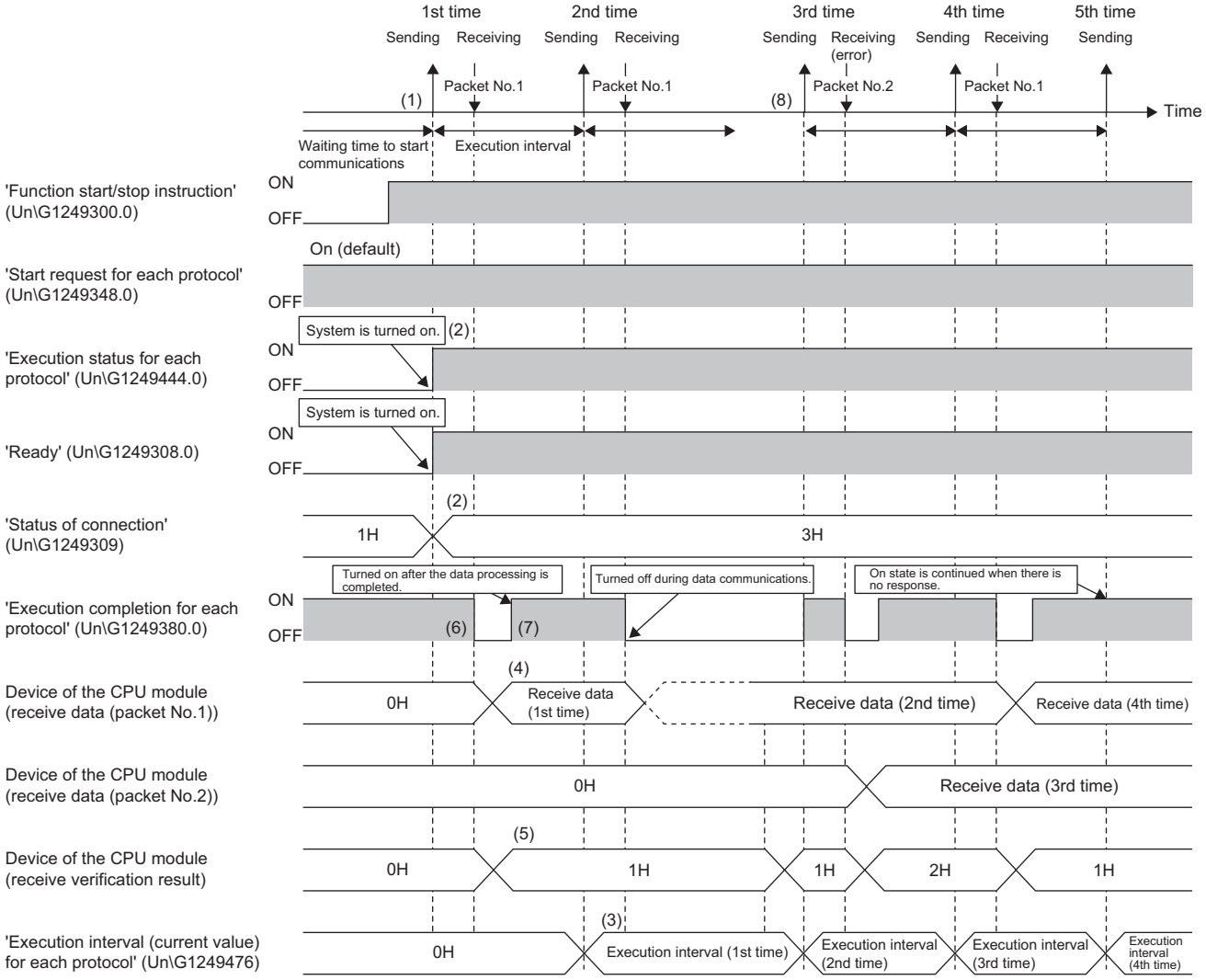
## ■Operation in error (Parameter error)

If a parameter error occurs when the module starts, the buffer memory area for simple device communications becomes 0 or is turned off.

## When "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Send & Receive"

### ■Normal operation

• When read command is used




(1)When the latency time has elapsed and if 'Function start/stop instruction' (Un\G1249300.0) is turned on (enable) and 'Start request for each protocol' (Un\G1249348.0) is turned on (start), the data communication is started.

(2)When the data communication is started, 'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed) and 'Status of connection' (Un\G1249309) becomes 3H (communicating).

(3)When the request is sent to the communication destination and "Execution Interval (ms)" has elapsed, and if the response data is received from the communication destination, the actual execution interval is stored in 'Execution interval (current value) for each protocol' (Un\G1249476) and the data is sent.

(4)For the read command (when the CPU module device is assigned in the response packet), the response data from the communication destination is received by the simple device communication processing of the own station and the CPU module device in the response packets that match the received response data is updated by the END processing.

(5)At the same time, the packet number for which the verification matches is registered in the CPU module device that is set in "Received Data Verification Result". (If the verification mismatches, 0 is registered.)

(6)'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed) after completion of the initial processing and turned off (not completed) during the update of the CPU module device.

(7)After completion of the update, 'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed). (If the CPU module device is assigned for each protocol, refer to the CPU module device using the on (completed) of 'Execution completion for each protocol' (Un\G1249380.0) as the interlock to prevent the receive data from being inconsistent.)

(8)If the processing until the completion of general data update takes time more than the execution interval that is set, the next sending is immediately executed.

• When write command is used

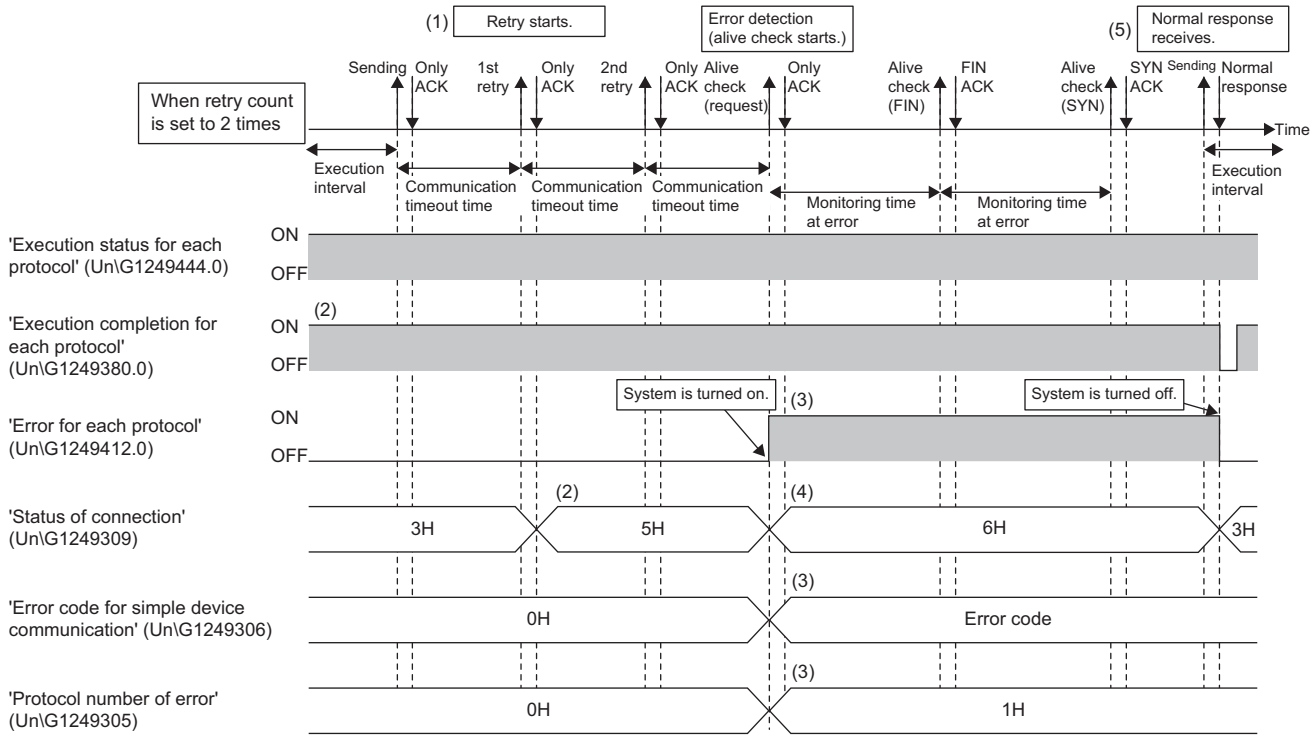
For the write command (when the CPU module device is assigned in the request packet), the END processing is performed to send (transfer) the CPU module device of the own station to the system area for simple device communications. Then, the simple device communication processing is performed to send data of the system area to the communication destination at every execution interval.

While the write data is obtained from the CPU module device, 'Execution completion for each protocol' (Un\G1249380.0) is turned off (not completed). In this case, changing the value of the CPU module device may cause inconsistency in the write data. Update the CPU module device when 'Execution completion for each protocol' (Un\G1249380.0) is on (completed).

**Point**

Since the simple device communication function cannot distinguish between normal responses and error responses, it continues to operate normally even if an error response is received.

## ■Operation in error (TCP)




(1)After data is sent to the communication destination, if only ACK is received without receiving any response data, the system makes retries (resends) when "Timeout Time" has elapsed.

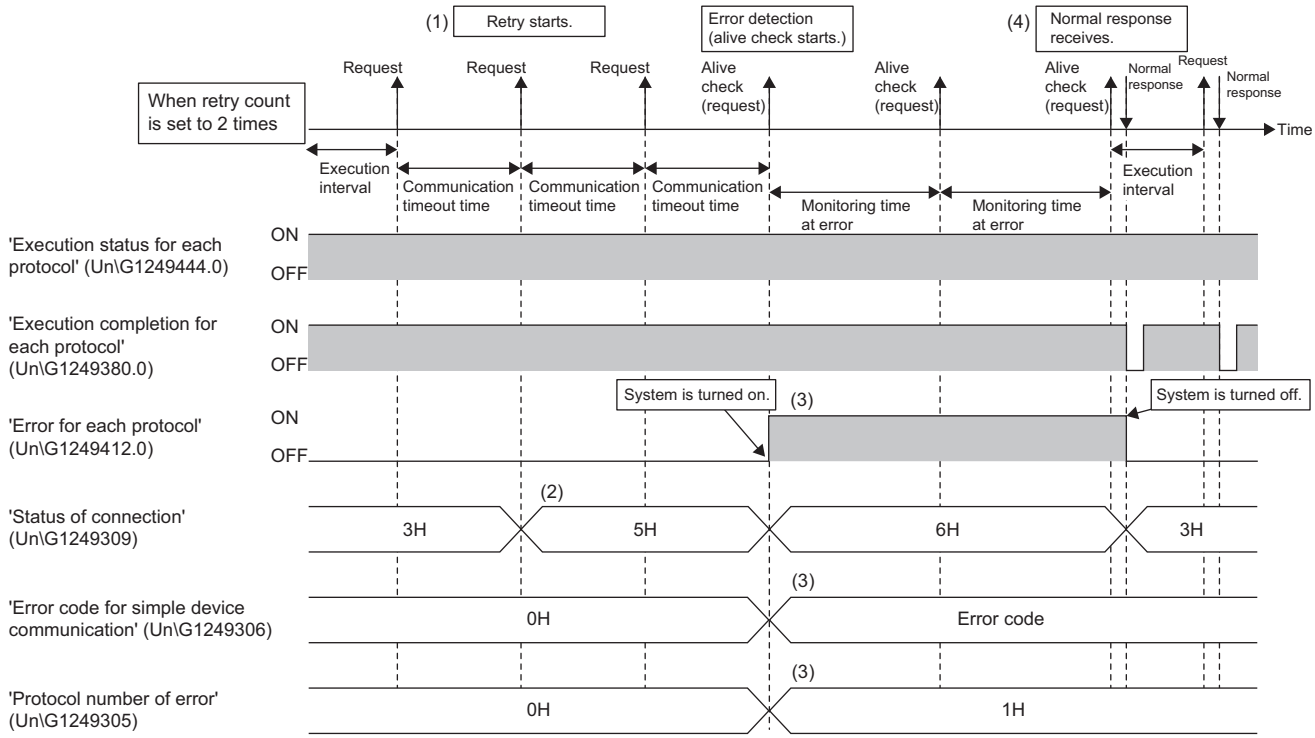
(2)During the retry, 'Status of connection' (Un\G1249309) becomes 5H (retry being executed). In this case, 'Execution interval (current value) for each protocol' (Un\G1249476) will not be updated. In addition, 'Execution completion for each protocol' (Un\G1249380.0) will remain on (completed). (Communications stop also for other protocols of the same communication destination, but communications with the other communication destinations continue.)

(3)Then, when the normal response cannot be received even if the data is sent for the number of "Timeout Retry Count Setting", "Timeout Time" elapses and then an error code is stored in 'Error code for simple device communication' (Un\G1249306) and the protocol number of error is stored in 'Protocol number of error' (Un\G1249305). Also, after 'Error for each protocol' (Un\G1249412.0) is turned on (error), the fixed interval communication is performed during the monitoring time at error. (The system shifts to the alive check mode to send close requests/open requests at the fixed interval during "Monitoring Time At Error" (low speed cycle).)

(4)During monitoring at error, 'Status of connection' (Un\G1249309) becomes 6H (monitoring at error).

(5)If any normal response from the communication destination is received during the monitoring at error (after the data is sent at the fixed interval in the low speed cycle), the system returns to the normal state (performs communications at "Execution Interval (ms)"). After the system returned to the normal state, 'Status of connection' (Un\G1249309) becomes 3H (communicating).

## ■Operation in error (UDP)

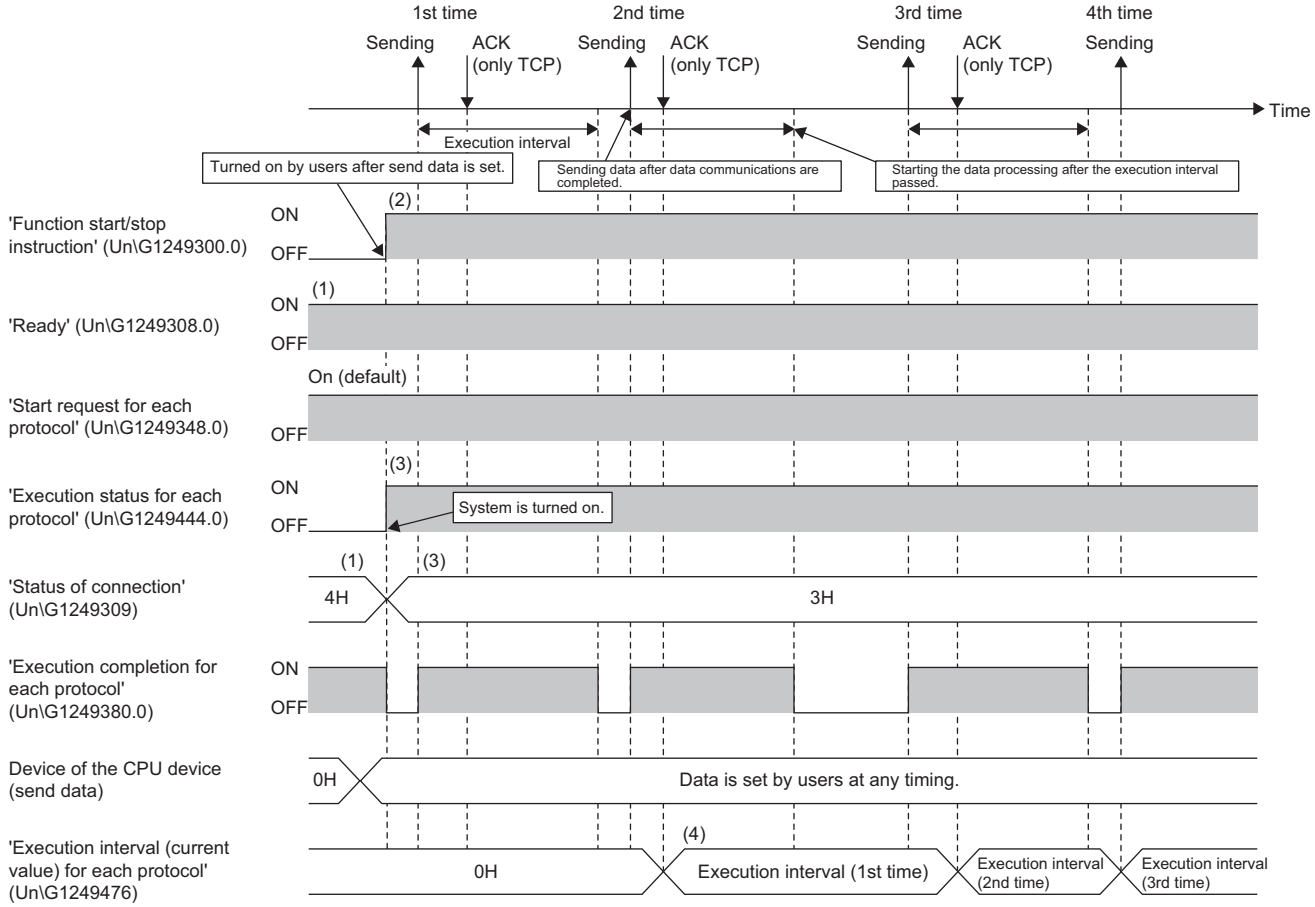



(1)After data is sent to the communication destination, if no response data is received, the system makes retries (resends) when "Timeout Time" has elapsed.
(2)During the retry, 'Status of connection' (Un\G1249309) becomes 5H (retry being executed). In this case, 'Execution interval (current value) for each protocol' (Un\G1249476) will not be updated. In addition, 'Execution completion for each protocol' (Un\G1249348.0) will remain on (completed). (Communications stop also for other protocols of the same communication destination, but communications with the other communication destinations continue.)
(3)Then, when the normal response cannot be received even if the data is sent for the number of "Timeout Retry Count Setting", "Timeout Time" elapses and then an error code is stored in 'Error code for simple device communication' (Un\G1249306) and the protocol number of error is stored in 'Protocol number of error' (Un\G1249305). Also, after 'Error for each protocol' (Un\G1249412.0) is turned on (error), the fixed interval communication is performed during the monitoring time at error.
(4)If any normal response from the communication destination is received during the monitoring at error (after the data is sent at the fixed interval in the low speed cycle), the system returns to the normal state (performs communications at "Execution Interval (ms)"). After the system returned to the normal state, 'Status of connection' (Un\G1249309) becomes 3H (communicating).

## When "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Send Only"

For the "Send Only" communication type, only the request packet can be set, and thus the operation for obtaining the value of the CPU module device before sending is shown.

### ■Normal operation




(1)When the system is powered off and on or the system is reset and the reset state is cleared and then the latency time has elapsed, 'Ready' (Un\G1249308.0) is turned on (ready) and 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).
(2)Set data to be sent to the CPU module device to turn on (enable) 'Function start/stop instruction' (Un\G1249300.0).
(3)When all 'Function start/stop instruction' (Un\G1249300.0), 'Ready' (Un\G1249308.0), and 'Start request for each protocol' (Un\G1249348.0) are turned on, 'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed) and 'Status of connection' (Un\G1249309) becomes 3H (communicating). When the data is sent successfully (for TCP, ACK is received), the first communication is completed.
(4)After "Execution Interval (ms)" has elapsed, the CPU module device is obtained and the data is sent. When the data is successfully sent, the actual execution interval is stored in 'Execution interval (current value) for each protocol' (Un\G1249476).

While the write data is obtained from the CPU module device, 'Execution completion for each protocol' (Un\G1249380.0) is turned off (not completed).

In this case, changing the value of the CPU module device may cause inconsistency in the write data.

Update the CPU module device when 'Execution completion for each protocol' (Un\G1249380.0) is on (completed).

### ■Operation in error (TCP)

The operation is the same as the case where "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Send & Receive".

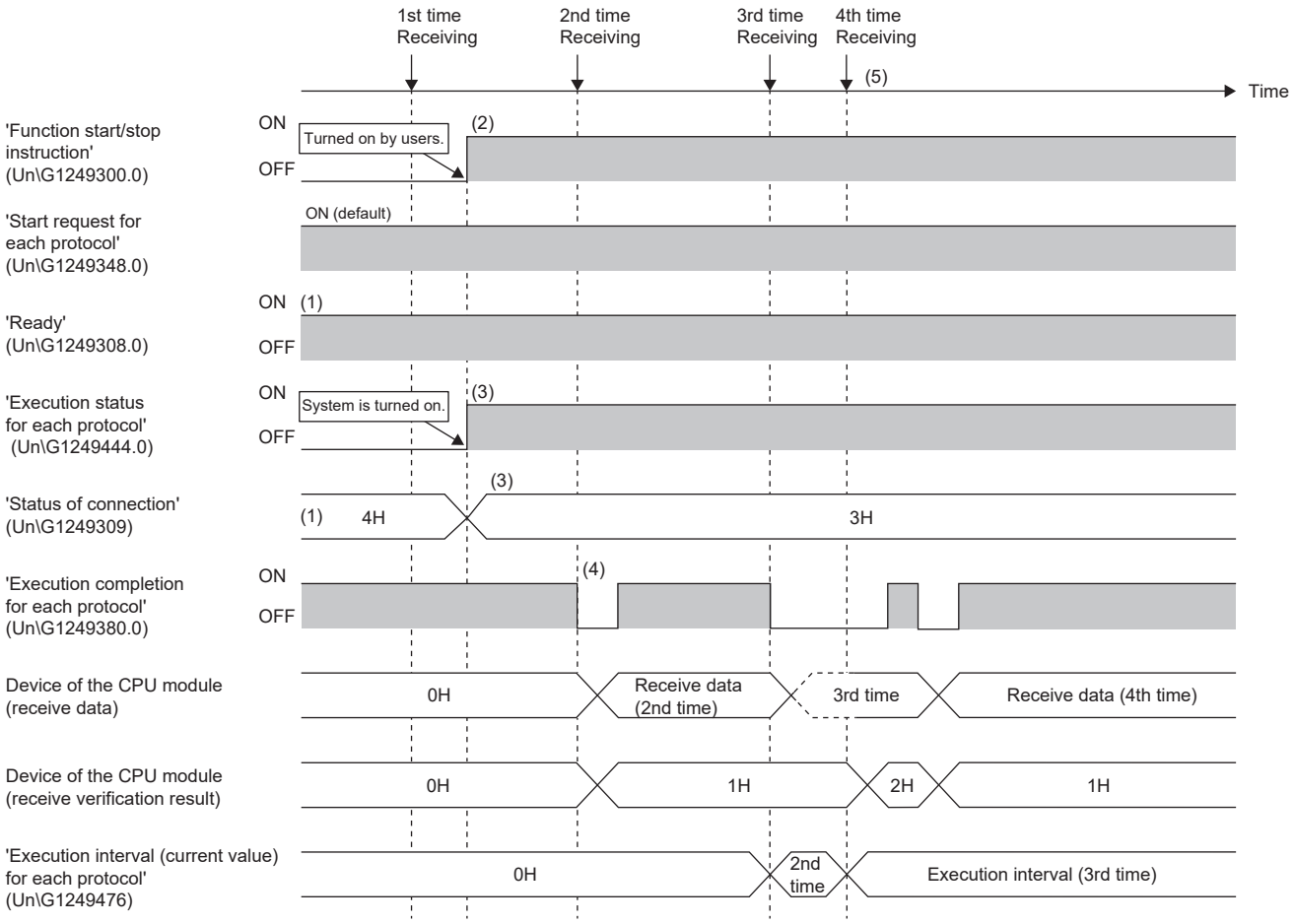
☞ Page 272 Operation in error (TCP)

## When "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Receive Only"

For the "Receive Only" communication type, only the response packet can be set, and thus the operation for updating the value of the CPU module device before receiving is shown.

### ■Normal operation

'Execution interval (current value) for each protocol' (Un\G1249476) in the following figure indicates the receiving cycle. The maximum value to be stored is 65535ms. When the value exceeds the limit, "65535ms" is stored.



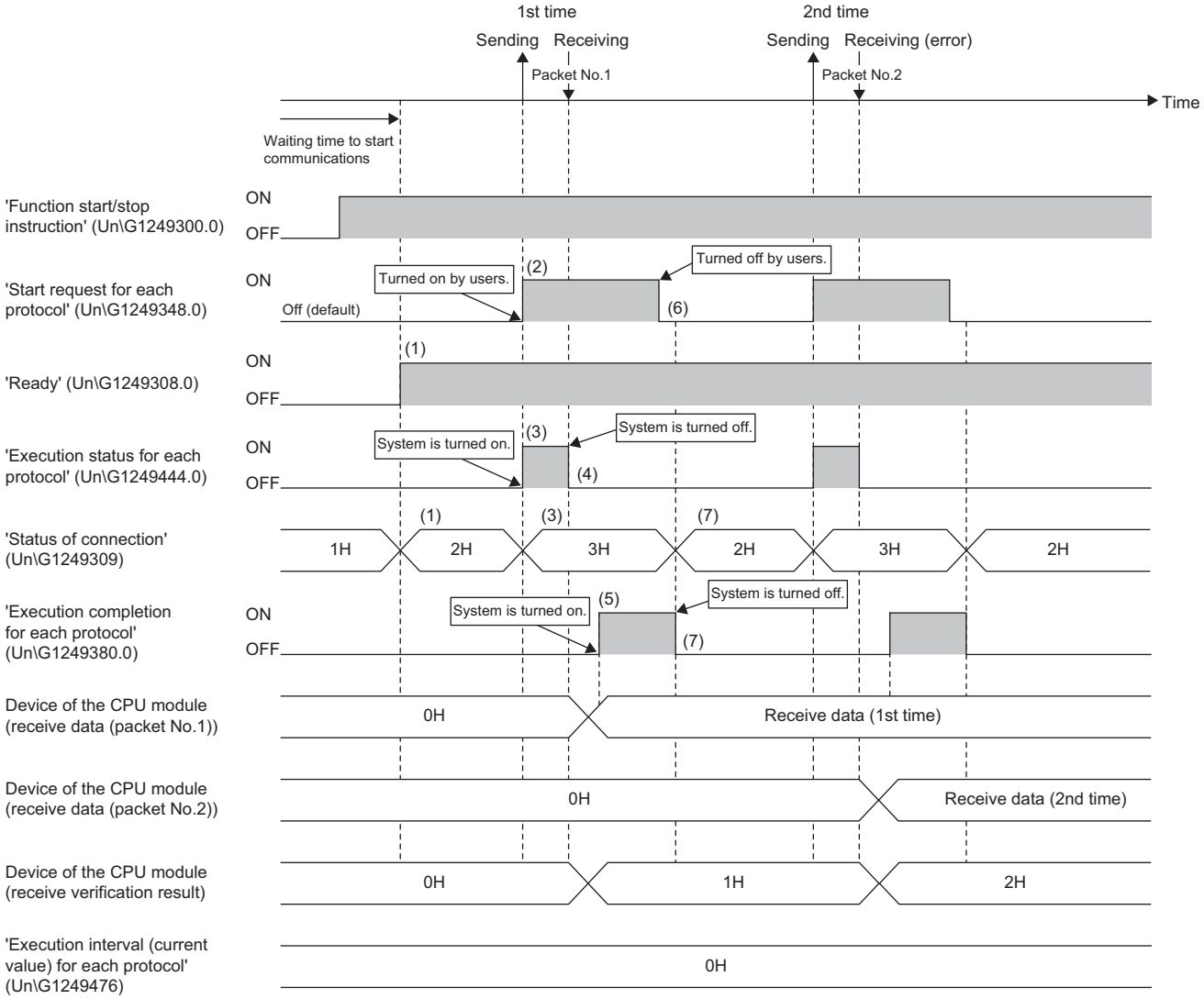

(1)When the system is powered off and on or the system is reset and the reset state is cleared and then the latency time has elapsed, 'Ready' (Un\G1249308.0) is turned on (ready) and 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).
(2)After receiving has become ready, turn on (enable) 'Function start/stop instruction' (Un\G1249300.0).
(3)When all 'Function start/stop instruction' (Un\G1249300.0), 'Execution completion for each protocol' (Un\G1249348.0), and 'Ready' (Un\ G1249308.0) are turned on, 'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed). At this point, 'Status of connection' (Un\G1249309) becomes 3H (communicating), then the module enters the receive data wait state. (The data that is received from the external device while 'Execution status for each protocol' (Un\G1249444.0) is off (communication stop) is discarded.)
(4)When data is received from the external device during the receive data wait state, 'Execution completion for each protocol' (Un\G1249380.0) is turned off (not completed), and the CPU module device is updated. After the update is completed, 'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed), and the module enters the receive data wait state again. (By making a reference to the CPU module device using 'Execution completion for each protocol' (Un\G1249380.0) as the interlock, inconsistency in the received data can be prevented.)
(5)The data that is received while 'Execution status for each protocol' (Un\G1249444.0) is on (communicating) and also 'Execution completion for each protocol' (Un\G1249380.0) is off (not completed) is stored in the internal receive buffer. Then, the data is processed immediately after 'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed).

### Precautions

The receive data wait continues even when "Timeout Time" has elapsed without receiving data from the external device. In addition, if a close is detected during receive data wait in TCP, alive check will be executed at the interval of the monitoring time at error × 2 after the number of times in "Timeout Time" multiplied by "Timeout Retry Count Setting" has elapsed. (This symptom is not treated as an error.)

## When "Comm Set" is "Request" and "Communication Type" is "Send & Receive"

### ■Normal operation

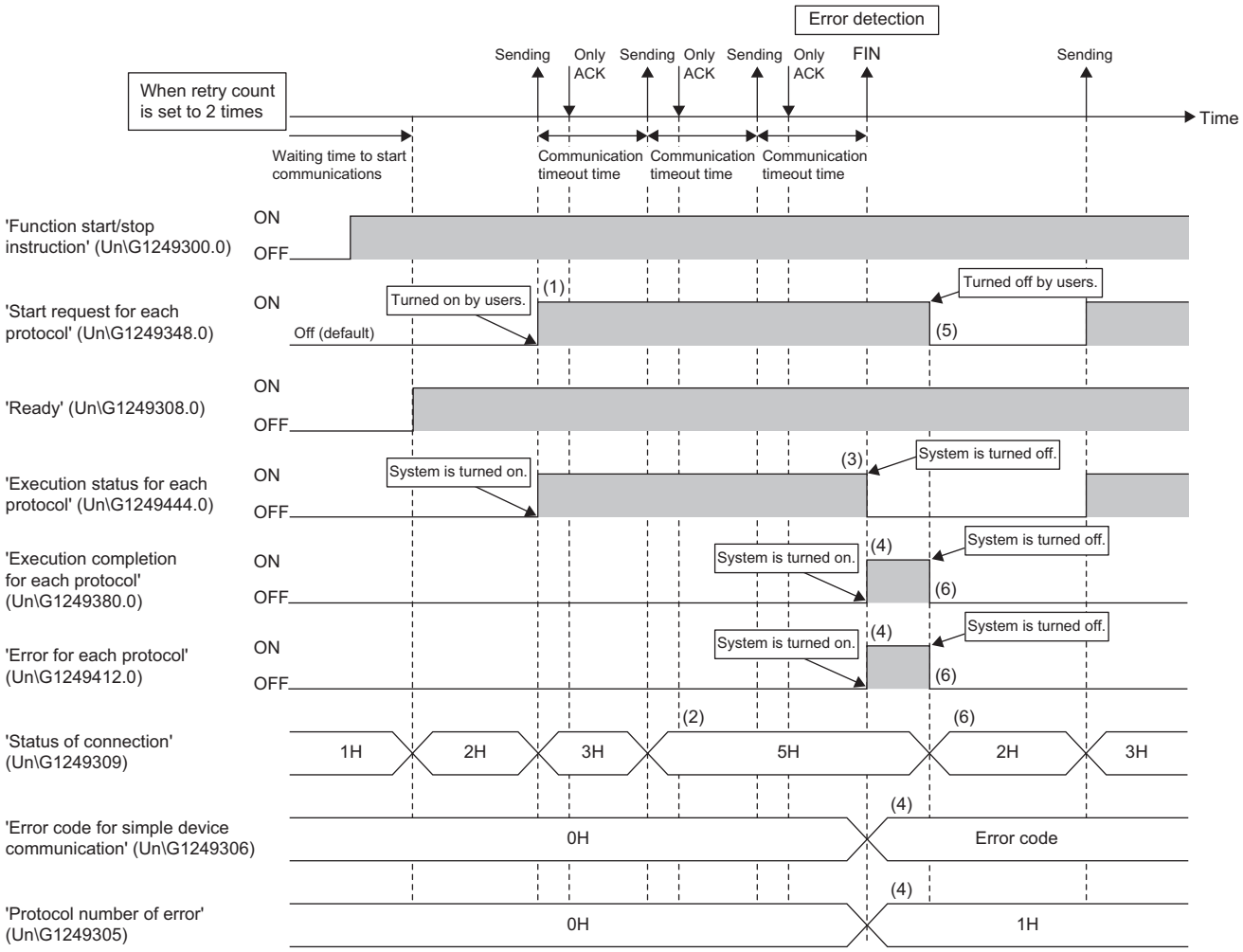



(1)When the system is powered off and on or the system is reset and the reset state is cleared and then the latency time has elapsed, 'Ready' (Un\G1249308.0) is turned on (ready) and 'Status of connection' (Un\G1249309) becomes 2H (waiting for the request). Also, the default of 'Start request for each protocol' (Un\G1249348.0) will be off (stop).
(2)Turn on (start) 'Start request for each protocol' (Un\G1249348.0) to start communications.
(3)'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed) and 'Status of connection' (Un\G1249309) becomes 3H (communicating).
(4)When any response data is received from the communication destination, 'Execution status for each protocol' (Un\G1249444.0) is turned off (communication stop).
(5)When the update of the CPU module device is completed, 'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed).
(6)On (completed) of 'Execution completion for each protocol' (Un\G1249380.0) acts as a trigger for obtaining the value of the CPU module device and turning off (stop) 'Start request for each protocol' (Un\G1249348.0).
(7)When 'Start request for each protocol' (Un\G1249348.0) is turned off (stop), 'Execution completion for each protocol' (Un\G1249380.0) is turned off (not completed) and 'Status of connection' (Un\G1249309) becomes 2H (waiting for the request). Also, 'Execution interval (current value) for each protocol' (Un\G1249476) is fixed to 0.

### Precautions

If the system turns off and on 'Start request for each protocol' (Un\G1249348.0) before turning on (completed) 'Execution completion for each protocol' (Un\G1249380.0), the request will be ignored. (This symptom is not treated as an error.)

## ■Operation in error (TCP)

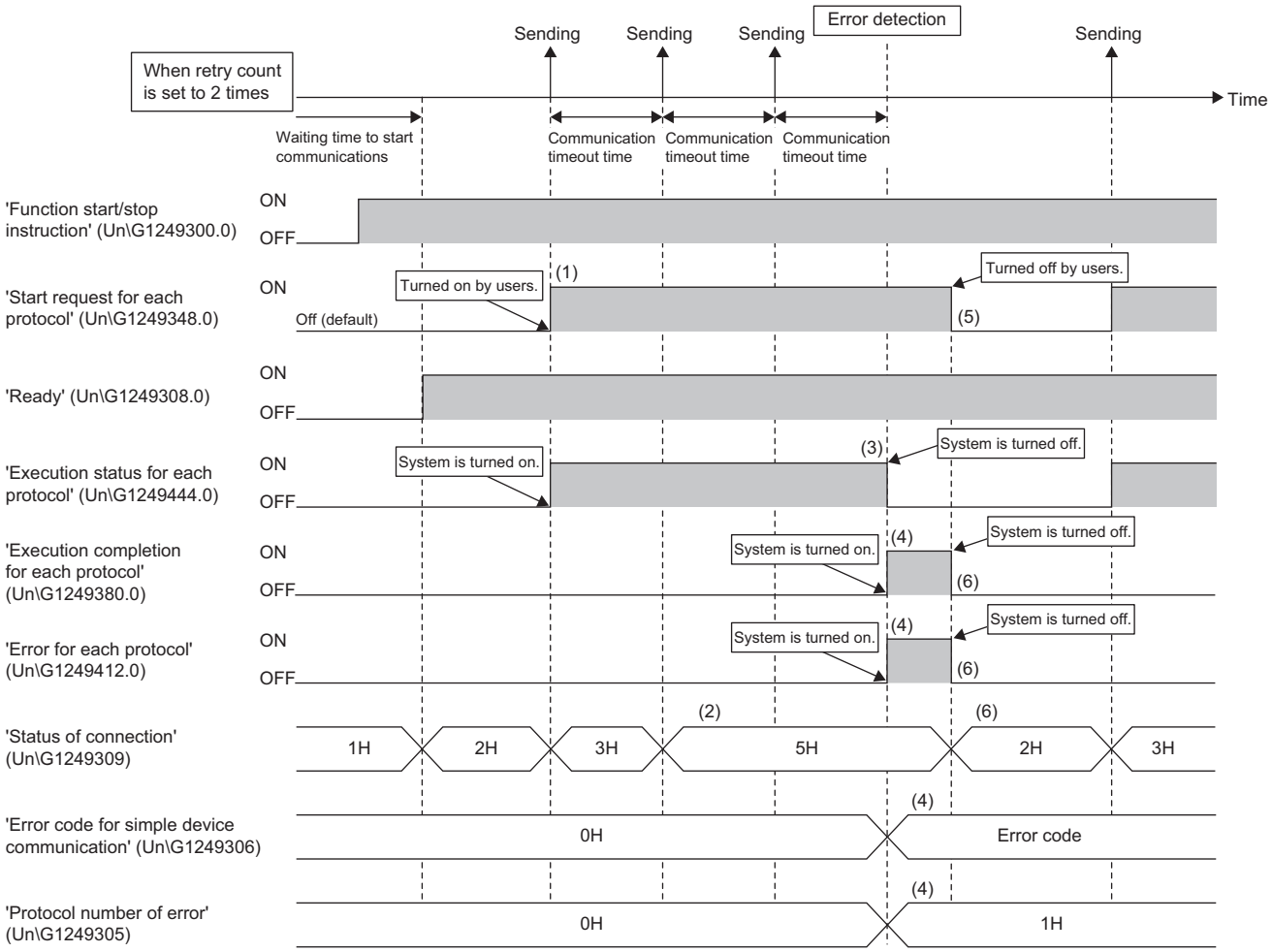



(1)Turn on (start) 'Start request for each protocol' (Un\G1249348.0).
(2)If no response is received even when "Timeout Time" has elapsed, 'Status of connection' (Un\G1249309) becomes 5H (retry being executed).
(3)Then, when the normal response cannot be received even if the data is sent for the number of "Timeout Retry Count Setting", 'Execution status for each protocol' (Un\G1249444.0) is turned off (communication stop).
(4)An error code is stored in 'Error code for simple device communication' (Un\G1249306) and the protocol number of error is stored in 'Protocol number of error' (Un\G1249305). Also, after 'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0) are turned on, the connection will be closed.
(5)On (completed) of 'Execution completion for each protocol' (Un\G1249380.0) is confirmed and then 'Start request for each protocol' (Un\G1249348.0) is turned off (stop).
(6)'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0) are turned off, and then 'Status of connection' (Un\G1249309) becomes 2H (waiting for the request).

### Precautions

If the system turns off 'Start request for each protocol' (Un\G1249348.0) before turning on (completed) 'Execution completion for each protocol' (Un\G1249380.0), the program may fail to detect on of 'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0).

## ■Operation in error (UDP)




(1)Turn on (start) 'Start request for each protocol' (Un\G1249348.0).
(2)If no response is received even when "Timeout Time" has elapsed, 'Status of connection' (Un\G1249309) becomes 5H (retry being executed).
(3)Then, when the normal response cannot be received even if the data is sent for the number of "Timeout Retry Count Setting", 'Execution status for each protocol' (Un\G1249444.0) is turned off (communication stop).
(4)An error code is stored in 'Error code for simple device communication' (Un\G1249306) and the protocol number of error is stored in 'Protocol number of error' (Un\G1249305). Also, 'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0) are turned on.
(5)On (completed) of 'Execution completion for each protocol' (Un\G1249380.0) is confirmed and then 'Start request for each protocol' (Un\G1249348.0) is turned off (stop).
(6)'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0) are turned off, and then 'Status of connection' (Un\G1249309) becomes 2H (waiting for the request).

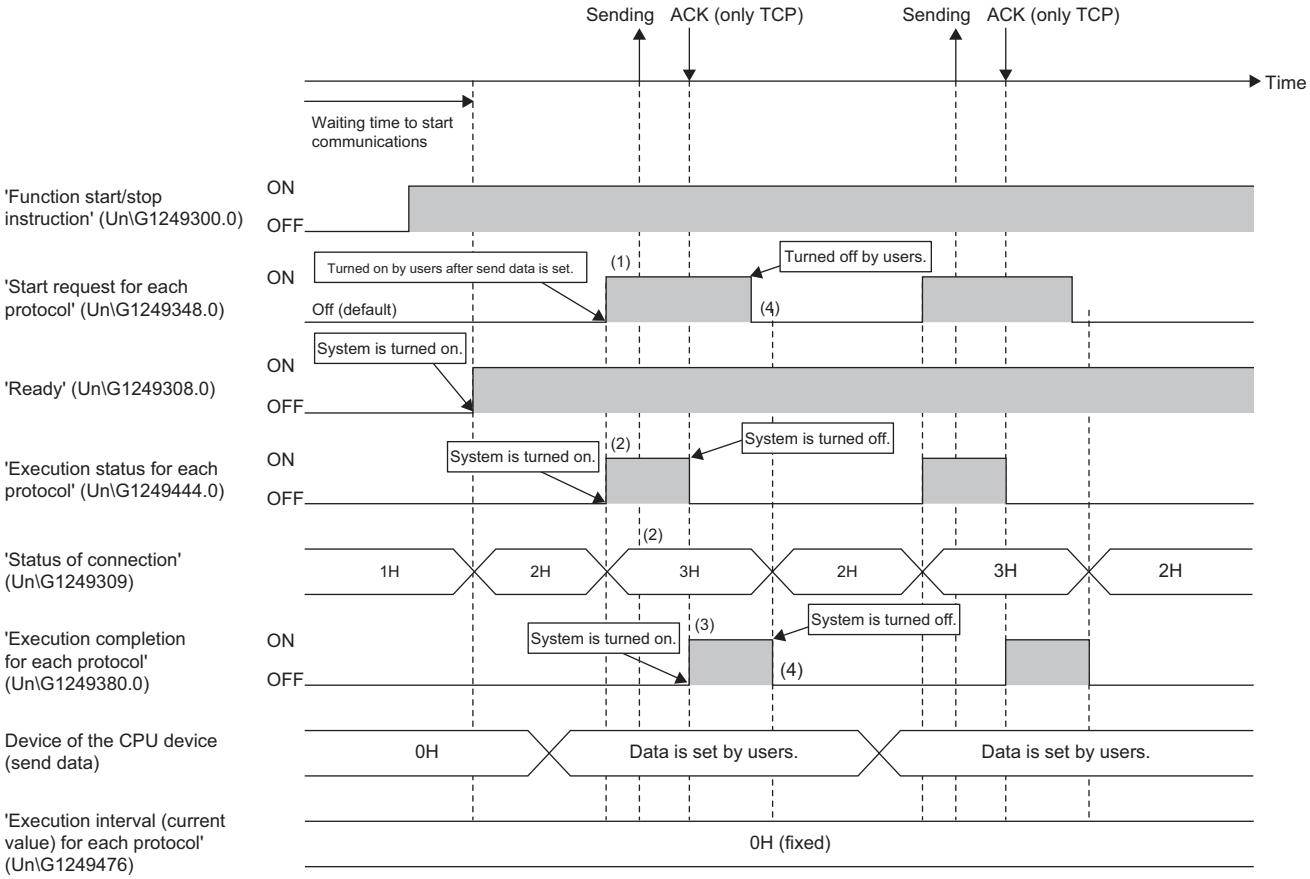
### Precautions

If the system turns off 'Start request for each protocol' (Un\G1249348.0) before turning on (completed) 'Execution completion for each protocol' (Un\G1249380.0), the program may fail to detect on of 'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0).

## When "Comm Set" is "Request" and "Communication Type" is "Send Only"

For the "Send Only" communication type, only the request packet can be set, and thus the operation for obtaining the value of the CPU module device before sending is shown.

### ■Normal operation




(1)Set data to be sent to the CPU module device and turn on (start) 'Start request for each protocol' (Un\G1249348.0).
(2)When 'Start request for each protocol' (Un\G1249348.0) is turned on (start), 'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed) and 'Status of connection' (Un\G1249309) becomes 3H (communicating).
(3)When the data is sent successfully (for TCP, ACK is received), 'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed).
(4)When the on (completed) of 'Execution completion for each protocol' (Un\G1249380.0) is confirmed and then 'Start request for each protocol' (Un\G1249348.0) is turned off (stop), 'Execution completion for each protocol' (Un\G1249380.0) is turned off (not completed) and the communication ends.

#### Precautions

If the system turns off and on 'Start request for each protocol' (Un\G1249348.0) again before turning on (completed) 'Execution completion for each protocol' (Un\G1249380.0), the request will be ignored. (This symptom is not treated as an error.)

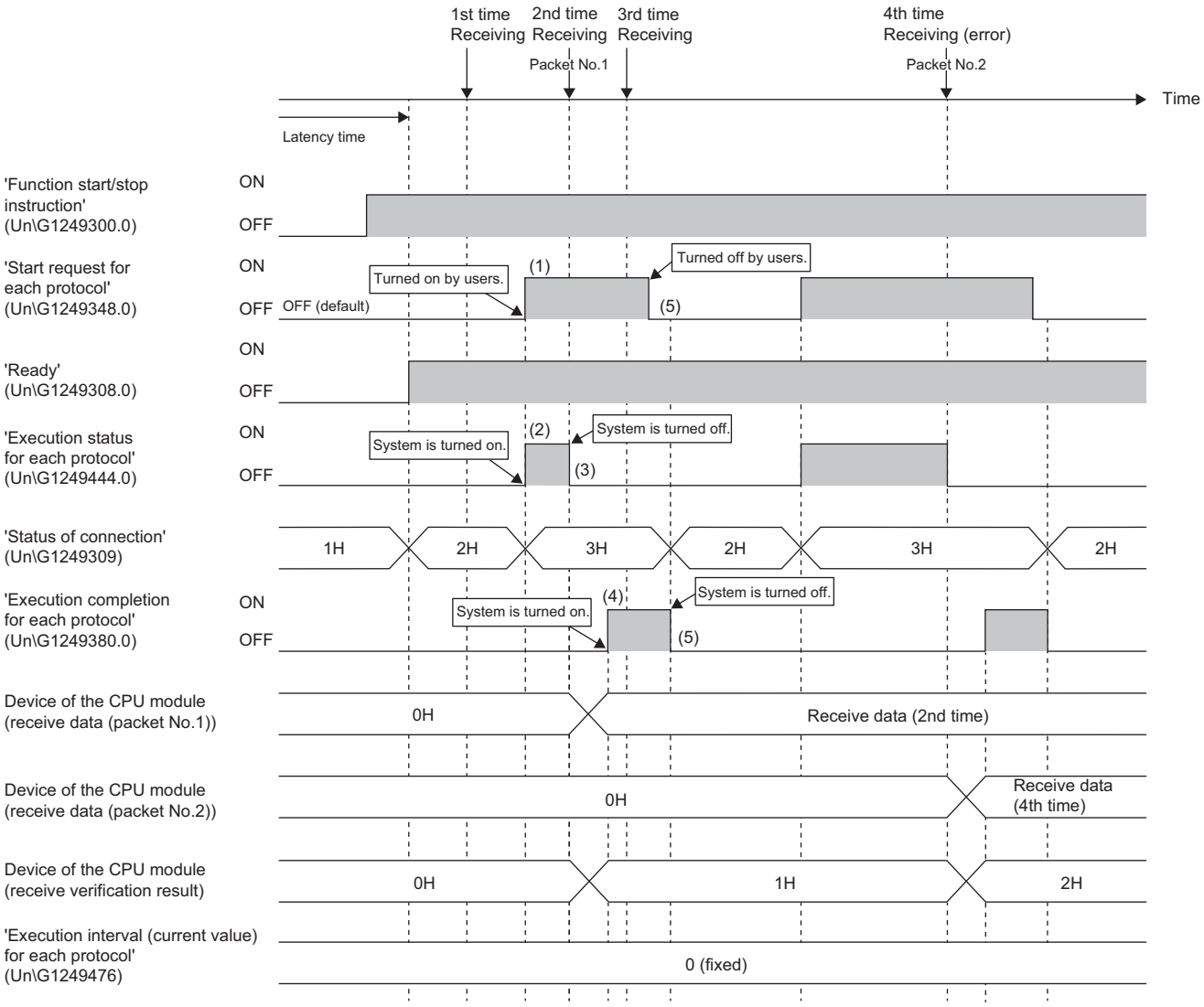
### ■Operation in error (TCP)

The operation is the same as the case where "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Send & Receive".
☞ Page 272 Operation in error (TCP)

## When "Comm Set" is "Request" and "Communication Type" is "Receive Only"

For the "Receive Only" communication type, only the response packet can be set, and thus the operation for updating the value of the CPU module device before receiving is shown.

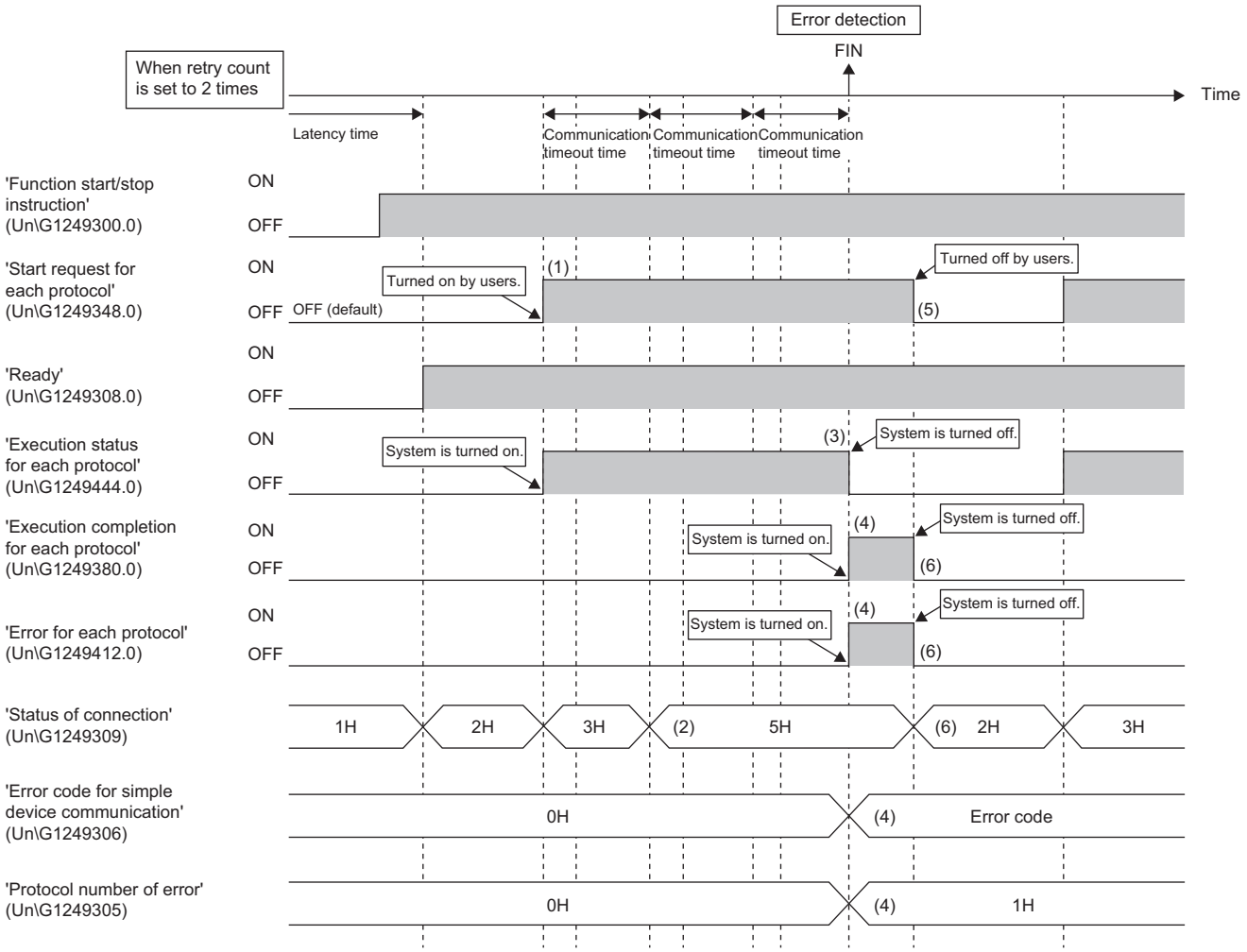
### ■Normal operation




(1)After receiving has become ready, turn on (start) 'Start request for each protocol' (Un\G1249348.0).
(2)When 'Start request for each protocol' (Un\G1249348.0) is turned on (start), 'Execution status for each protocol' (Un\G1249444.0) is turned on (being executed).
(3)When data is received from the external device, 'Execution status for each protocol' (Un\G1249444.0) is turned off (communication stop).
(4)When the update of the CPU module device is completed, 'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed).
(5)'Execution completion for each protocol' (Un\G1249380.0) is turned on (completed) and the value of the CPU module device is obtained. Then, when 'Start request for each protocol' (Un\G1249348.0) is turned off (stop), 'Execution completion for each protocol' (Un\G1249380.0) is turned off (not completed) and the communication ends.

Any of the data that is received from the external device while 'Execution status for each protocol' (Un\G1249444.0) is off (communication stop) is discarded.

## ■Operation in error (without data receiving)




(1)Turn on (start) 'Start request for each protocol' (Un\G1249348.0).
(2)If no data is received from the external device even when "Timeout Time" has elapsed, 'Status of connection' (Un\G1249309) becomes 5H (retry being executed).
(3)If no data is received from the external device even when the number of times in "Timeout Time" multiplied by "Retry Count Setting" has elapsed, 'Execution status for each protocol' (Un\G1249444.0) is turned off (communication stop).
(4)An error code is stored in 'Error code for simple device communication' (Un\G1249306) and the protocol number of error is stored in 'Protocol number of error' (Un\G1249305). Also, 'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0) are turned on, and the connection will be closed.
(5)Check that 'Execution completion for each protocol' (Un\G1249380.0) is on (completed), then turn off (stop) 'Start request for each protocol' (Un\G1249348.0).
(6)'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0) are turned off, and then 'Status of connection' (Un\G1249309) becomes 2H (waiting for the request).

### Precautions

If the system turns off 'Start request for each protocol' (Un\G1249348.0) before turning on (completed) 'Execution completion for each protocol' (Un\G1249380.0), the program may fail to detect on of 'Execution completion for each protocol' (Un\G1249380.0) and 'Error for each protocol' (Un\G1249412.0).

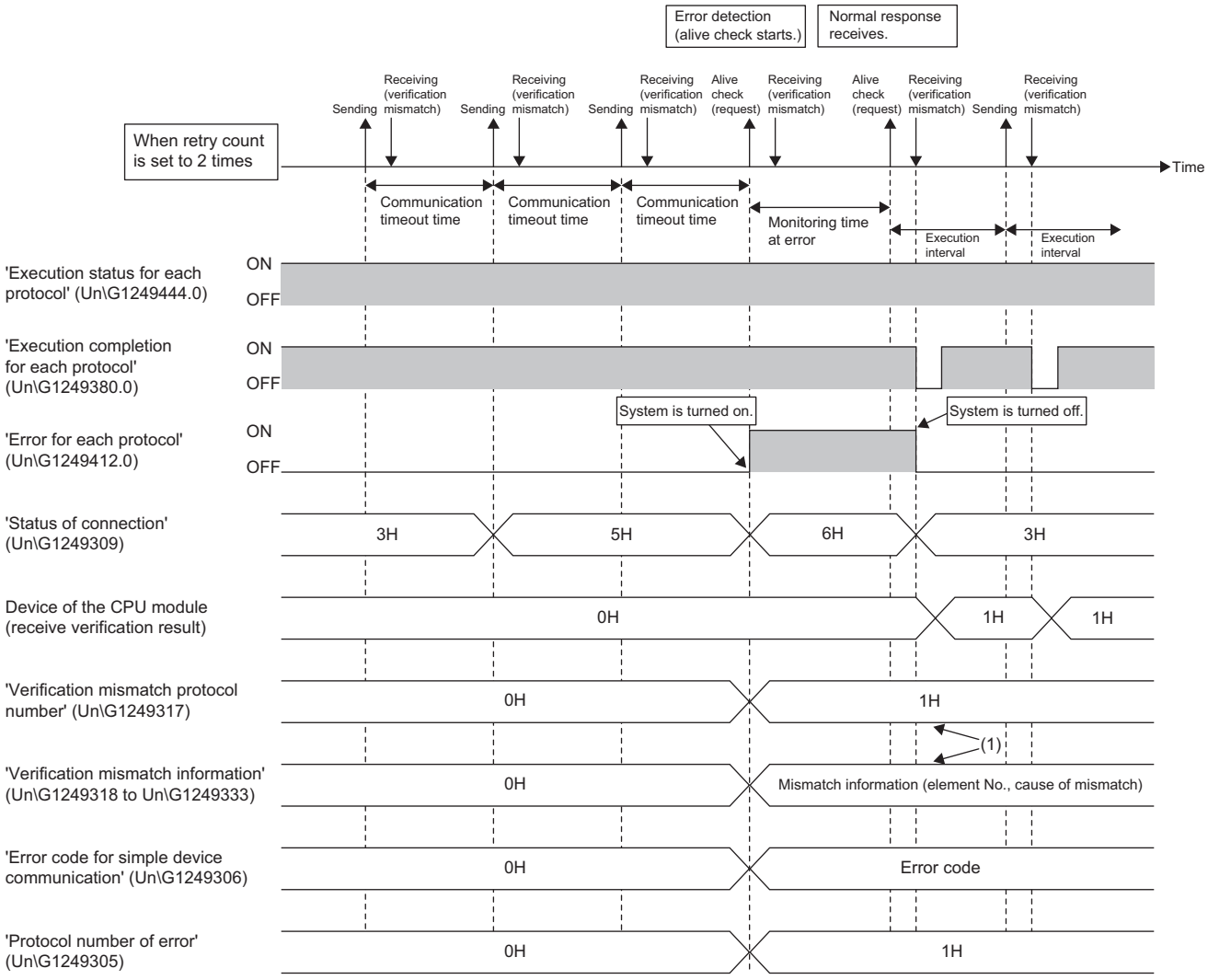
## For verification mismatch

When a packet for which the verification mismatches is received (the response data sent from the communication destination does not match the receive packet format), 0H[*1] is stored in the CPU module device (received data verification result), the protocol number of the verification mismatch in 'Verification mismatch protocol number' (Un\G1249317), and the mismatch information in 'Verification mismatch information' (Un\G1249318 to Un\G1249333).

Also, the following receive data will be abandoned.

- For TCP: Receive data that results in verification mismatch
- For UDP: UDP packet that results in verification mismatch

*1 The verification mismatch occurs also when the receive data is insufficient in TCP. However in such a case, 0H will not be stored and the previous value will be maintained.




(1)The error will not be cleared even when the verification mismatch has been resolved. To clear the error, use 'Error clear request' (Un\G1249307).

## Operations for stopping and restarting communications

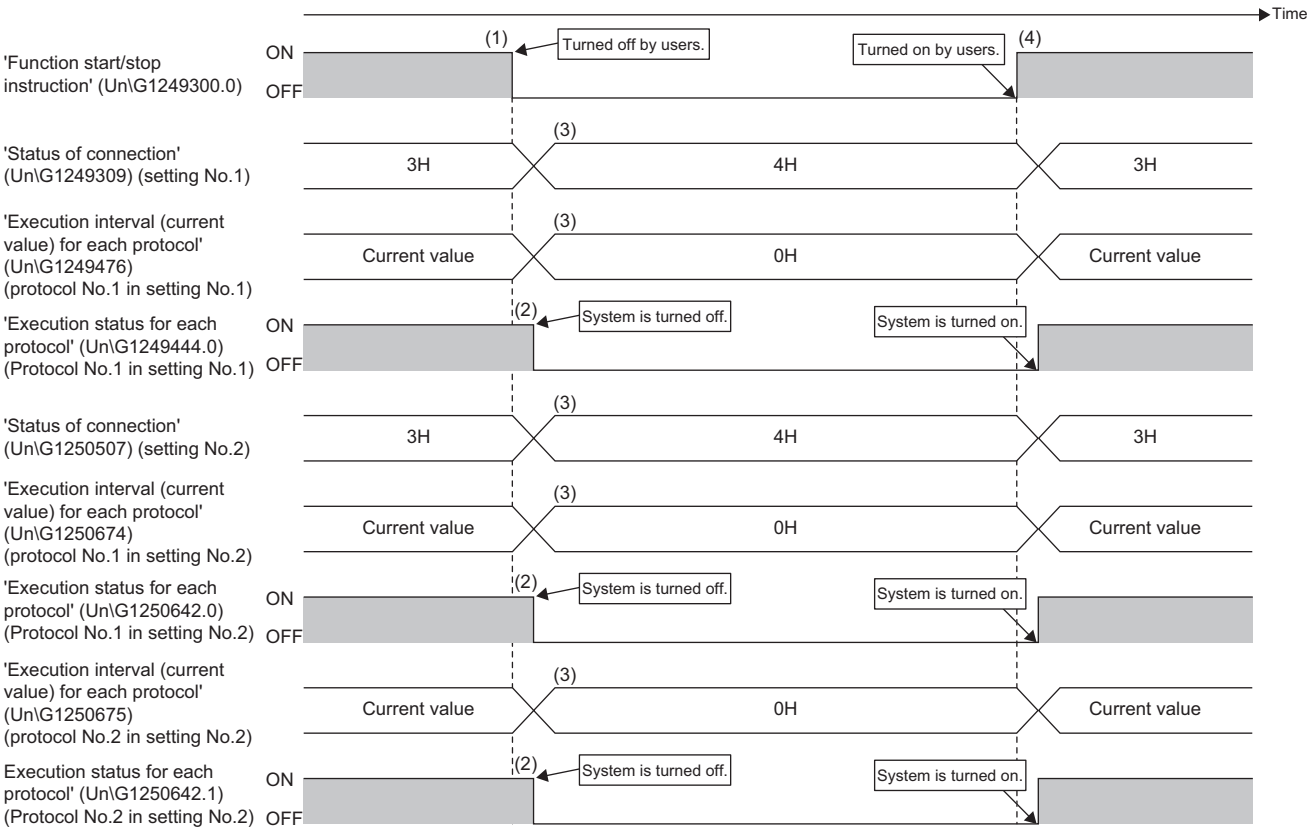
Communications can be stopped and restarted by operating the buffer memory areas in the following conditions:

- "Comm Set" is "Fixed Intrvl"
- "Comm Set" is "Request", "Communication Type" is "Receive Only", and "Timeout Time" is 0 (infinite wait)

If the communication destination does not respond with the stop request accepted, the communications will stop when "Timeout Time" has elapsed. Communications stop at the time when a stop request is accepted in either of the following conditions: "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Receive Only"; or "Comm Set" is "Request" and "Timeout Time" is 0 (infinite wait).

When the retry is being executed, the communications will stop after "Timeout Time" has elapsed without depending on "Timeout Retry Count Setting". When "Timeout Time" is 0 (infinite wait), communications stop after 1000ms, which is the default value of "Timeout Time", has passed.

### ■Stop and restart for each Ethernet port

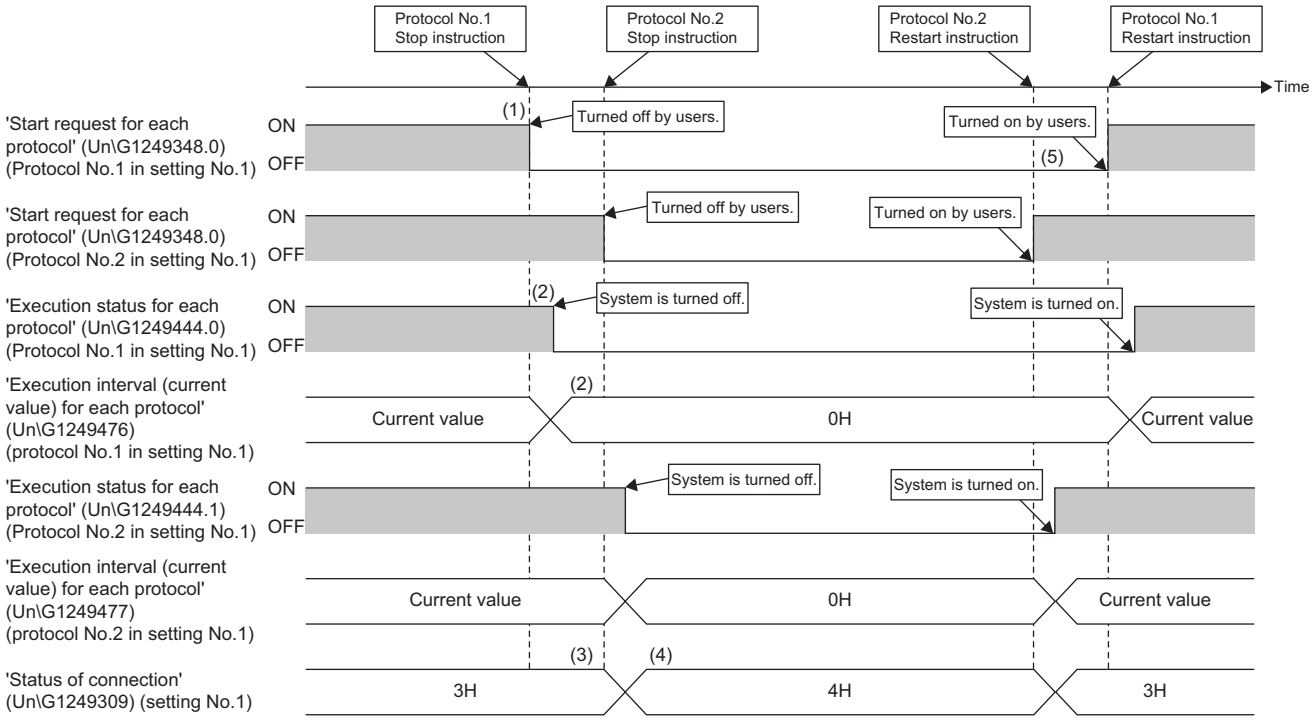



(1)Turn off (stop) 'Function start/stop instruction' (Un\G1249300.0) to stop communications with all the devices through the relevant Ethernet port.
(2)When the stop request is accepted, 'Execution status for each protocol' (Un\G1249444.0) (protocol No.1 in setting No.1), 'Execution status for each protocol' (Un\G1250642.0) (protocol No.1 in setting No.2), and 'Execution status for each protocol' (Un\G1250642.1) (protocol No.2 in setting No.2) are turned off (execution stop).
(3)'Status of connection' (Un\G1249309) becomes 4H (function being stopped). When "Comm Set" is "Fixed Intrvl", 0 is stored in the following: 'Execution interval (current value) for each protocol' (Un\G1249476) (protocol No.1 in setting No.1), 'Execution interval (current value) for each protocol' (Un\G1250674) (protocol No.1 in setting No.2), and 'Execution interval (current value) for each protocol' (Un\G1250675) (protocol No.2 in setting No.2).
(4)Turn on (enable) 'Function start/stop instruction' (Un\G1249300.0) to restart all the communications that are stopped. However, the protocols with 'Start request for each protocol' (Un\G1249348 to Un\G1249379) turned off (stop) will not restart the communications. When "Comm Set" is "Request", the communications restart under the following conditions: 'Start request for each protocol' (Un\G1249348 to Un\G1249379) is on (start) before 'Function start/stop instruction' (Un\G1249300.0) is turned off (stop). Then, 'Start request for each protocol' (Un\G1249348 to Un\G1249379) is turned off (stop) once and turned on (start) again.

The operations after the communications restart differ depending on 'Status of connection' (Un\G1249309) before the communications are stopped.

- For 1H (preparing): When the latency time has elapsed during the stop, 'Status of connection' (Un\G1249309) becomes 3H (communicating) and the communication starts. If the latency time has not elapsed, the status remains as 1H (preparing) and time measurement continues.
- For 2H (waiting for the request), 3H (communicating), 5H (retry being executed), and 6H (monitoring at error): The operation will restart at the time when 'Function start/stop instruction' (Un\G1249300.0) is turned on (enable), and 3H (communicating) is stored in 'Status of connection' (Un\G1249309) and on (being executed) is stored in 'Execution status for each protocol' (Un\G1249444.0). When "Comm Set" is "Fixed Intrvl", a value is stored in 'Execution interval (current value) for each protocol' (Un\G1249476).

## ■Stop and restart for each protocol




(1)Turn off (stop) 'Start request for each protocol' (Un\G1249348.0) of the protocol number 1 and 'Start request for each protocol' (Un\G1249348.1) of the protocol number 2 to stop the communications of the relevant protocols.
(2)When stop request is accepted, 'Execution status for each protocol' (Un\G1249444.0) is turned off (stop). When "Comm Set" is "Fixed Intrvl", 0 is stored in the following: 'Execution interval (current value) for each protocol' (Un\G1249476) of the protocol number 1 and in 'Execution interval (current value) for each protocol' (Un\G1249477) of the protocol number 2.
(3)When another protocol is being executed with the same communication destination, 'Status of connection' (Un\G1249309) maintains the present state.
(4)When all the protocols that communicate with the same communication destination are stopped, 'Status of connection (Un\G1249309) becomes 4H (function being stopped).
(5)Turn on (start) 'Start request for each protocol' (Un\G1249348.0) of the protocol number 1 and 'Start request for each protocol' (Un\G1249348.1) of the protocol number 2 to restart the communications that are stopped.

If all the protocols of the communication destination are stopped, the operation at the restart will be the same as the operation of the Ethernet port. (☞ Page 283 Stop and restart for each Ethernet port)
However, the communications will not restart if 'Function start/stop instruction' (Un\G1249300.0) is off (stop). After the restart instruction, the communications will be executed at the communication timing of the relevant protocol.

## Stored value for connection status

The following shows values to be stored in 'Status of connection' (Un\G1249309) in the following cases with the same setting number.

- When both "Fixed Intrvl" and "Request" are used for "Comm Set"
- When multiple protocols set to "Request" are used
- When a protocol with "Comm Set" set to "Fixed Intrvl" and "Communication Type" set to "Receive Only" is used with another protocol
- When multiple protocols, whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", are used

### ■When both "Fixed Intrvl" and "Request" are used for "Comm Set"

- While the protocol set to "Fixed Intrvl" is normally communicating, 'Status of connection' (Un\G1249309) becomes 3H (communicating) even if the protocol set to "Request" is waiting for the request.
- While any of the protocols set to "Fixed Intrvl" or "Request" becomes 5H (retry being executed) or 6H (monitoring at error), the value is stored in 'Status of connection' (Un\G1249309).
- When 'Start request for each protocol' (Un\G1249348.0) of the protocol set to "Fixed Intrvl" is off (stop), the value of the protocol set to "Request" is stored in 'Status of connection' (Un\G1249309).*1
- When 'Function start/stop instruction' (Un\G1249300.0) is off (stop), 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).

*1 The following specification also applies if multiple protocols are set to "Request".
☞ Page 286 When multiple protocols set to "Request" are used

### ■When multiple protocols set to "Request" are used

- While all the protocols are waiting for the request, 'Status of connection' (Un\G1249309) becomes 2H (waiting for the request).
- While any one of the protocols is 3H (communicating), 5H (retry being executed), or 6H (monitoring at error), the value is stored in 'Status of connection' (Un\G1249309).
- When 'Start request for each protocol' (Un\G1249348.0) of one protocol is off (stop), 'Status of connection' (Un\G1249309) is determined with the remaining protocols.
- When 'Function start/stop instruction' (Un\G1249300.0) is off (stop), 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).

### ■When a protocol, whose "Comm Set" is set to "Fixed Intrvl" and "Communication Type" to "Receive Only", is used with another protocol

- While any one of the protocols, other than those whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", is 3H (communicating), 5H (retry being executed), or 6H (monitoring at error), the values are stored in 'Status of connection' (Un\G1249309).
- While all the protocols, other than those whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", are either 2H (waiting for the request) or 4H (function being stopped), the processing is as follows: Among the protocols whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", except for the protocols which are 4H (function being stopped), the status of the smallest protocol number is stored in 'Status of connection' (Un\G1249309).
- When 'Function start/stop instruction' (Un\G1249300.0) is off (stop), 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).

### ■When multiple protocols, whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", are used

- When all the protocols are 4H (function being stopped), 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).
- When any one of the protocols is 3H (communicating), 5H (retry being executed), or 6H (monitoring at error), the status of the smallest protocol number is stored in 'Status of connection' (Un\G1249309), except for the protocol of 4H (function being stopped).
- When 'Function start/stop instruction' (Un\G1249300.0) is off (stop), 'Status of connection' (Un\G1249309) becomes 4H (function being stopped).

## When multiple protocols are used together

### ■When "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Receive Only"

The following describes the operation when a protocol, whose "Comm Set" is set to "Fixed Intrvl" and "Communication Type" to "Receive Only", is used in conjunction with another protocol that has the same setting number.

- A protocol whose "Comm Set" is set to "Fixed Intrvl" and "Communication Type" to "Receive Only" can be executed with another protocol concurrently.
- Receive data is verified first for a protocol with "Communication Type" set to "Send & Receive" or a protocol with "Comm Set" set to "Request" and "Communication Type" to "Receive Only" if the protocol and a protocol with "Communication Type" to "Receive Only" are in the receive data wait status.
- The receive data is verified in ascending order of the protocol number while multiple protocols, whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", are in the receive data wait state.
- Verification of the receive data finishes once a data match is verified.

Ex.

The following table shows the example verification order of when data are received after executing the following protocols: a protocol with "Comm Set" set to "Fixed Intrvl" and "Communication Type" to "Send & Receive", and multiple protocols with "Comm Set" set to "Fixed Intrvl" and "Communication Type" to "Receive Only".

The same verification order applies when the protocol with "Comm Set" set to "Fixed Intrvl" and "Communication Type" to "Send & Receive" is changed to a protocol with "Comm Set" set to "Request" and "Communication Type" to "Send & Receive" or "Receive Only".

| Verification order | Protocol number | Communication setting | Communication Type | Receive packet number | Verification result |
|---|---|---|---|---|---|
| 1 | 3 | Fixed Intrvl | Send & Receive | 1 | Mismatch |
| 2 | | | | 2 | Mismatch |
| 3 | 1 | Fixed Intrvl | Receive Only | 1 | Mismatch |
| 4 | | | | 2 | Mismatch |
| 5 | 2 | Fixed Intrvl | Receive Only | 1 | Mismatch |
| 6 | | | | 2 | Match |
| 7 | | | | 3 | Not to be verified |
| 8 | 4 | Fixed Intrvl | Receive Only | 1 | Not to be verified |

- While multiple protocols are in the receive data wait state, if all protocols are verified as mismatched, the following will be updated: 'Verification mismatch protocol number' (Un\G1249317), 'Verification mismatch information' (Un\G1249318 to Un\G1249333), and 'Number of error completion' (Un\G1249312 to Un\G1249313). At this time, each buffer memory area will store the latest verified protocol information.
- If the data received in TCP is insufficient, verification will be suspended. Once the next data is received, verification will resume from the protocol that had insufficient received data.
- Refer to Page 286 Stored value for connection status for the value to be stored in 'Status of connection' (Un\G1249309) under either of the following conditions: When a protocol, whose "Comm Set" is set to "Fixed Intrvl" and "Communication Type" to "Receive Only", is used in conjunction with another protocol; or when multiple protocols, whose "Comm Set" are set to "Fixed Intrvl" and "Communication Type" to "Receive Only", are used.
- When a protocol, whose "Comm Set" is set to "Fixed Intrvl" and "Communication Type" to "Receive Only", is being executed with another protocol, the latter protocol number will be stored in 'Protocol number currently executing' (Un\G1249316). Also, when multiple protocols with "Comm Set" set to "Fixed Intrvl" and "Communication Type" to "Receive Only" are being executed and the other protocols are not being executed, the smallest protocol number of the protocols being executed will be stored in 'Protocol number currently executing' (Un\G1249316).
- If a lot of protocols with "Comm Set" set to "Fixed Intrvl" and "Communication Type" set to "Receive Only" are executed, it takes longer to process the receive data. Turn off (stop) 'Start request for each protocol' (Un\G1249348.0) for unnecessary protocols.

## Protocol execution order

### ■When "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Receive Only"

For details on the protocol execution order, refer to the following.

☞ Page 287 When multiple protocols are used together

### ■Settings other than when "Comm Set" is "Fixed Intrvl" and "Communication Type" is "Receive Only"

The protocol execution order is as follows:

- Each protocol is executed independently for each setting number.
- When 'Start request for each protocol' (Un\G1249348.0) of multiple protocols are turned on (start) simultaneously, the protocols may not be executed in ascending order of the protocol number. For how to execute multiple protocols in a desired order when "Comm Set" is "Request", refer to Page 291 When executing multiple protocols in desired order with "Comm Set" set to "Request".
- When data communications start simultaneously, protocols are executed in ascending order based on the protocol number.
- When the communication setting of a protocol is "Fixed Intrvl" and the communication restarts after it is stopped, the relevant protocol is executed after the execution of the last protocol among protocols being executed or to be executed at the timing when 'Start request for each protocol' (Un\G1249348.0) turns on (start).
- When the communication setting of a protocol is "Request", the relevant protocol is executed after the execution of the last protocol among protocols being executed or to be executed at the timing when 'Start request for each protocol' (Un\G1249348.0) turns on (start).

# Precautions



This section provides precautions for the simple CPU communication function.

## Incomplete data reception and timeout

Because the load of Ethernet communication becomes heavy during the simple device communication, if another communication function (such as MELSOFT connection, SLMP communication, socket communication function, and simple CPU communication function) is simultaneously performed by using UDP, the data reception of UDP may not be completed, resulting in a timeout error or other errors. Therefore, to perform other communications during the simple CPU communication, the communication using TCP is recommended.

## Execution interval setting

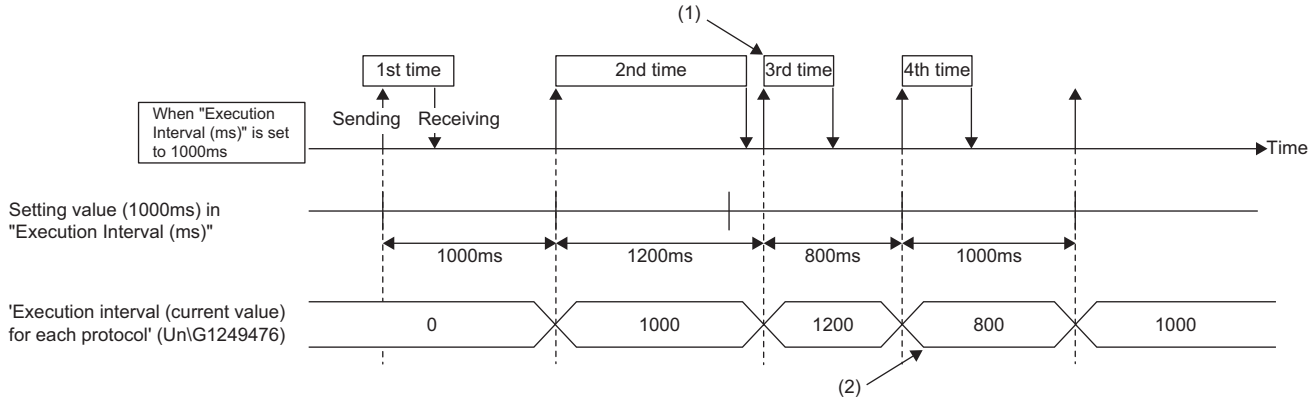
The execution interval of the simple device communication function and time required for completing the communications will differ according to the load rate of the Ethernet line, operating status of other communication functions (such as MELSOFT connection, SLMP communication, socket communication function, and simple CPU communication function), scan time of the communication destination, and system configuration. Thus, the set execution interval or time required for completing the communications may be increased.

### ■Cause of increased execution interval of the simple device communication function

- The scan time is long for the control CPU module of the RJ71EN71 and RnENCPU (network part) where the simple device communication function is set.
- When the RJ71EN71 and RnENCPU (network part) where the simple device communication function is set are communicating by using a function other than the simple device communication function, the communication processing using a function other than the simple device communication function takes time.
- The load on the Ethernet line is high.
- The number of communication destinations or protocols set in "Simple Device Communication Setting" under "Application Settings" is large.
- Response from the communication destination is delayed.
- More than one RJ71EN71 where the simple CPU communication function or simple device communication function is set are mounted on a single base unit (main base unit or extension base unit).

To perform communications at the set execution interval, set a sufficient execution interval. If the execution interval needs to be checked, actually perform communications to check the interval with 'Execution interval (current value) for each protocol' (Un\G1249476) and 'Execution interval (maximum value) for each protocol' (Un\G1249988).

## Stored value for 'Execution interval (current value) for each protocol' (Un\G1249476)




(1)If a response from the communication destination takes time more than "Execution Interval (ms)" that is set, the next sending is immediately executed.
(2)If the response time from the communication destination exceeds "Execution Interval (ms)" in the previous communication, the value to be stored in 'Execution interval (current value) for each protocol' (Un\G1249476) in the next communication will be smaller than the value set for "Execution Interval (ms)".

## Duplication of own station port numbers

The simple device communication cannot be performed in the following cases.

- When the own station port number set in the simple device communication and the own station port number of the programmable controller set in the external device configuration are the same
- When the own station port number set in the simple device communication and the one set in another function such as the socket communication are the same

## IP address and model of communication destination

Check the IP address and model of the communication destination before setting the parameters.
The simple device communication cannot be performed when no communication destination exists (an incorrect IP address is specified) or when an incorrect model is selected.

- When "Comm Set" is "Fixed Intrvl", the connection status of each setting number changes from 5H (retry being executed) to 6H (monitoring at error).
- When "Comm Set" is "Request", the connection status of each setting number becomes 5H (retry being executed).

## When the file register (R, ZR) is used

When using the file register (R, ZR) for the device on the own station, always assign the device within the setting range. If the devices are not assigned, an error occurs (error code: 4031), and the simple device communication of the "Setting No." which exceeds the range cannot be performed.

## Number of applicable settings

- Up to 16 settings are applicable per port when "Normal" is set for "Resource Setting" under "Application Settings". If, however, 49 or more connections have been used for other functions, the maximum number of applicable settings is the difference calculated by subtracting the number of connections used for other function from 64 connections.
- Up to 32 settings are applicable per port when "Extension 1" is set for "Resource Setting" under "Application Settings". If, however, 33 or more connections have been used for other functions, the maximum number of applicable settings is the difference calculated by subtracting the number of connections used for other function from 64 connections.
- Up to 64 settings are applicable per port when "Extension 2" is set for "Resource Setting" under "Application Settings". If, however, 1 or more connections have been used for other functions, the maximum number of applicable settings is the difference calculated by subtracting the number of connections used for other function from 64 connections.

## Measures against errors occurring during protocol execution

When an error occurs during protocol execution, the protocol execution may not be completed properly. Check the error details, take appropriate measures, and execute the protocol again.

## When using a protocol with "Communication Type" set to "Receive Only"

The precautions for when using a protocol with "Communication Type" set to "Receive Only" are as follows:

- If data is received during receive processing in a protocol with "Communication Type" set to "Receive Only", the received data may be discarded.
- With TCP and "Communication Type" set to "Receive Only", the connection is closed by turning off 'Function start/stop instruction' (Un\G1249300.0) or 'Start request for each protocol' (Un\G1249348.0) at the time when 'Status of connection' (Un\G1249309) makes the transition from values other than 4H (function being stopped) and 2H (waiting for the request) to 4H (function being stopped) or 2H (waiting for the request).
- When only a protocol with "Communication Type" set to "Receive Only" is used in TCP, the system cannot detect a communication disconnection with the external device, which disables reconnection. Therefore, it is recommended to use protocols in conjunction with the one whose "Communication Type" is set to "Send & Receive" or "Send Only". When attempting to reconnect using only a protocol with "Communication Type" set to "Receive Only", close the connection once before reconnecting through the method described above.
- If the protocol with TCP and "Communication Type" set to "Receive Only" has not opened a connection even after "Timeout Time" (calculated by "Retry Count Setting" + 1) has elapsed, a communication error will be detected and an error (error code: CF81H) will occur. When "Timeout Time" is set to 0 (infinite wait), "Timeout Time" is treated as the default value (1000ms).
- If a protocol with TCP and "Comm Set" set to "Fixed Intrvl" and "Communication Type" to "Receive Only" is used, the connection may close when 'Status of connection' (Un\G1249309) becomes 6H (monitoring at error).

## When executing multiple protocols in desired order with "Comm Set" set to "Request"

Ensure that 'Execution completion for each protocol' (Un\G1249380.0) for the preceding protocol has been turned on (complete), then turn on (start) 'Start request for each protocol' (Un\G1249348.0).

## Data received outside of protocol execution timing

- Data are not received unless the connection is open.
- When only protocols whose "Communication Type" are set to "Send & Receive" or "Send Only" are set, the data received outside of the receive data wait timing of the protocol will be held in the receive data area. The setting of "Clear Receive System Area" in "Send/Receive Parameter Setting" determines how the data held in the receive data area are to be treated.

| Setting in "Clear Receive System Area" | How data held in receive data area to be treated |
|---|---|
| Yes (default) | Discarded and not used for receive processing |
| No | Used for receive processing |

- When a protocol whose "Communication Type" is "Receive Only" is set, data will be discarded if received at the time when no protocols are in the receive data wait status.

## Restrictions applicable depending on versions

### ■Firmware version

The simple device communication function is available for the RJ71EN71 or RnENCPU (network part) whose firmware version is "56" or later and that uses the engineering tool with version 1.075D or later.

<table>
<tr><th rowspan="2">Engineering tool</th><th colspan="5">RJ71EN71 or RnENCPU (network part)</th></tr>
<tr><th>Firmware version of "56" or later</th><th>Firmware version of "59" or later</th><th>Firmware version of "71" or later</th><th>Firmware version of "75" or later</th><th>Firmware version of "77" or later</th></tr>
<tr><td>Version 1.075D or later</td><td colspan="5">• Number of settings per port: 16 ("Normal"), 32 ("Extension 1")<br>• Number of protocols per port: 256 ("Normal"), 512 ("Extension 1")<br>• Communication setting: "Fixed Intrvl" and "Request"<br>• Communication types: Send Only, Send & Receive</td></tr>
<tr><td>Version 1.080J or later</td><td>Same as the engineering tool with versions 1.075D and later</td><td colspan="4">Protocol setting editing function added</td></tr>
<tr><td>Version 1.095Z or later</td><td>Same as the engineering tool with versions 1.075D and later</td><td>Same as the engineering tool with versions 1.080J and later</td><td colspan="3">Following settings added<br>• Element "Error check code"<br>• Enable (Upper → Lower, two-word units), Enable (Upper → Lower, four-word units) for byte swap for non-conversion variables</td></tr>
<tr><td>Version 1.100E or later</td><td>Same as the engineering tool with versions 1.075D and later</td><td>Same as the engineering tool with versions 1.080J and later</td><td>Same as the engineering tool with versions 1.095Z and later</td><td colspan="2">Following settings added<br>• Element "Conversion Variable"<br>• "Data Unit" for the length</td></tr>
<tr><td>Version 1.105K or later</td><td>Same as the engineering tool with versions 1.075D and later</td><td>Same as the engineering tool with versions 1.080J and later</td><td>Same as the engineering tool with versions 1.095Z and later</td><td>Same as the engineering tool with versions 1.100E and later</td><td>Following settings added<br>Communication type: Receive Only<br>Resource setting: Extension 2*1<br>• Number of communication destinations: 64<br>• Number of protocols: 2048<br>• Number of packets: 6144<br>• Packet data area size: 491520 bytes<br>• Number of elements: 1 to 128<br>• Packet length: 1 to 10238 bytes</td></tr>
</table>

*1 Supported by the modules that have production information where third and fourth digits from the left are as follows:

| Model | The third and fourth digits from the left of the production information |
|---|---|
| RJ71EN71 | "25" or later |
| R04ENCPU | "56" or later |
| R08ENCPU | "54" or later |
| R16ENCPU | "51" or later |
| R32ENCPU | "52" or later |
| R120ENCPU | "44" or later |

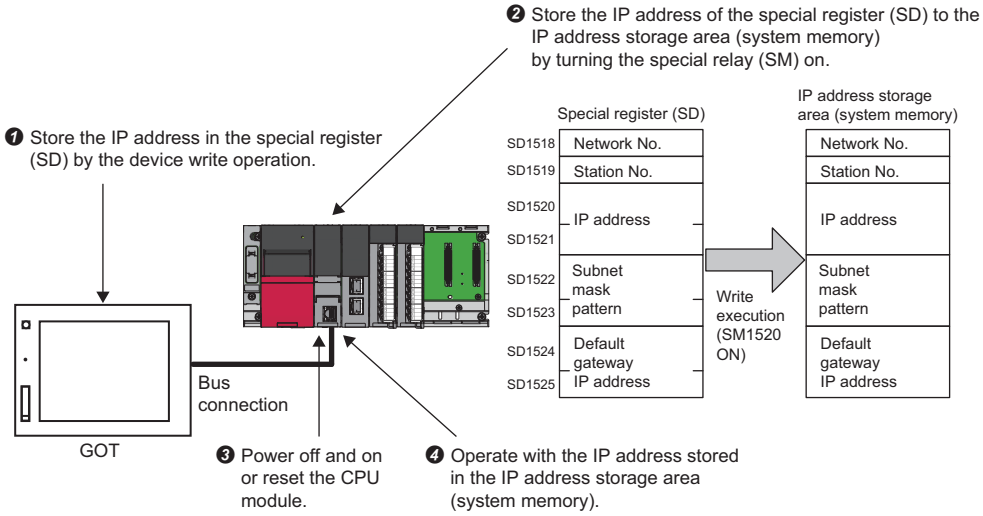
# 1.17 IP Address Change Function



This function can change the IP address of the Ethernet-equipped module or other item without changing the parameter settings.

Ex.

Case of the CPU module (built-in Ethernet port part)




**Point**

- When this function is used in the CPU module (built-in Ethernet port part), it can be used also by manipulating the special relay and special register values from the GOT. For details on the special relay and special register, refer to the manual for the CPU module used.
- When this function is used in the RJ71EN71 or RnENCPU (network part), it can be used also by manipulating the buffer memory values from the GOT. (☞ Page 486 Buffer Memory)

**Restriction**

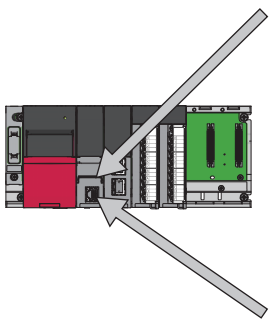
- When this function is used in the RJ71EN71 or RnENCPU (network part), check the firmware version of the module. The network number and station number of the CPU module (built-in Ethernet port part) can be changed in the CPU module with the firmware version supporting communications by network number/ station number. (☞ Page 586 Added and Enhanced Functions)
- The IP address change function is not available when "Q Compatible Ethernet" is selected for the network type in the RJ71EN71.

# IP address of the Ethernet-equipped module

The value set with the module parameters during initial processing is set as the IP address of the Ethernet-equipped module. When this function is used, the IP address set during initial processing is the value stored in the IP address storage area instead of the value set in the parameters.

Ex.

Case of the CPU module (built-in Ethernet port part)



[When using parameters]

CPU module

| | |
|---|---|
| IP address | 192.168.3.39 |
| Subnet mask pattern | 255.255.255.0 |
| Default gateway IP address | 192.168.3.254 |
| Network No. | 3 |
| Station No. | 39 |

Parameter file
Module parameter

| | |
|---|---|
| IP address | 192.168.3.39 |
| Subnet mask pattern | 255.255.255.0 |
| Default gateway IP address | 192.168.3.254 |
| Network No. | 3 |
| Station No. | 39 |

The IP address in the parameter file is set.

[When using the IP address change function]

CPU module

| | |
|---|---|
| IP address | 192.168.3.40 |
| Subnet mask pattern | 255.255.255.0 |
| Default gateway IP address | 192.168.3.254 |
| Network No. | 5 |
| Station No. | 20 |

Parameter file
Module parameter

| | |
|---|---|
| IP address | 192.168.3.39 |
| Subnet mask pattern | 255.255.255.0 |
| Default gateway IP address | 192.168.3.254 |
| Network No. | 3 |
| Station No. | 39 |

IP address storage area (system memory)

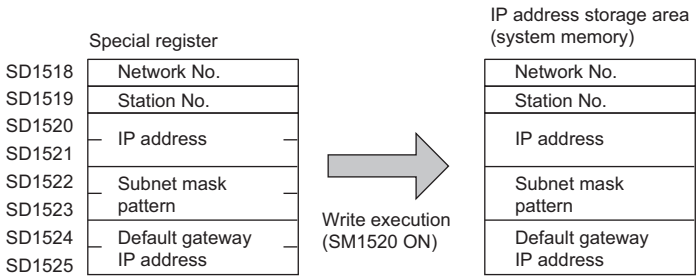
| | |
|---|---|
| IP address | 192.168.3.40 |
| Subnet mask pattern | 255.255.255.0 |
| Default gateway IP address | 192.168.3.254 |
| Network No. | 5 |
| Station No. | 20 |

The IP address in the IP address storage area (system memory) is set.

## Writing and clearing the IP address

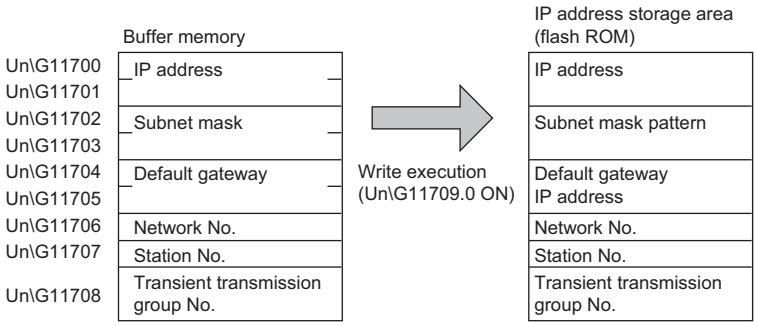
### ■Case of the CPU module (built-in Ethernet port part)

The IP address value is written into the IP address storage area (system memory). The write and clear operations are performed with the special relay and special register.



## ■Case of the RJ71EN71 or RnENCPU (network part)

The IP address value is written into the IP address storage area (flash ROM). The write and clear operations are performed with the buffer memory.



# Usage methods

## Write operation

### ■Case of the CPU module (built-in Ethernet port part)

Follow the steps below.

1. Store the value to be changed in 'IP address setting' (SD1518 to SD1525).
2. Turn off and on 'IP address storage area write request' (SM1520).
3. Check the write results with the following special relay and special register.

| Device | At normal completion | At abnormal completion |
|---|---|---|
| 'IP address storage area write request' (SM1520) | Off | Off |
| 'IP address storage area write error' (SM1521) | Off | On |
| 'IP address storage area writing error cause' (SD1526) | — | Stores the cause of the error |

4. Power off and on or set the CPU module.
5. If the IP address stored in the IP address storage area (system memory) is a valid value, the stored IP address is set as the IP address of the CPU module (built-in Ethernet port part). (If the value is invalid or not set, the value set in the module parameters is set as the IP address of the CPU module (built-in Ethernet port part).)
6. Check the IP address with the buffer memory. (☞ Page 496 CPU module (built-in Ethernet port part))

### ■Case of the RJ71EN71 or RnENCPU (network part)

Follow the steps below.

1. Store the values to be changed in 'IP address setting' (Un\G11700 to Un\G11705) and 'Communications by Network No./Station No.' (Un\G11706 to Un\G11708).
2. Turn off and on 'IP address storage area write request' (Un\G11709.0).
3. Check the write results with the following buffer memory areas.

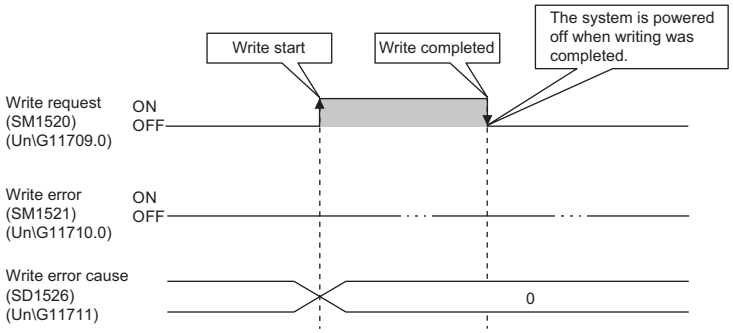
| Buffer memory | At normal completion | At abnormal completion |
|---|---|---|
| 'IP address storage area write request' (Un\G11709.0) | Off | Off |
| 'IP address storage area write error' (Un\G11710.0) | Off | On |
| 'IP address storage area writing error cause' (Un\G11711) | — | Stores the cause of the error |

4. Power off and on or set the CPU module.
5. If the IP address stored in the IP address storage area (flash ROM) is a valid value, the stored IP address is set as the IP address of the RJ71EN71 or the RnENCPU (network part). (If the value is invalid or not set, the value set in the module parameters is set as the IP address of the RJ71EN71 or the RnENCPU (network part).)
6. Check the IP address with the buffer memory. (☞ Page 486 RJ71EN71, RnENCPU (network part))

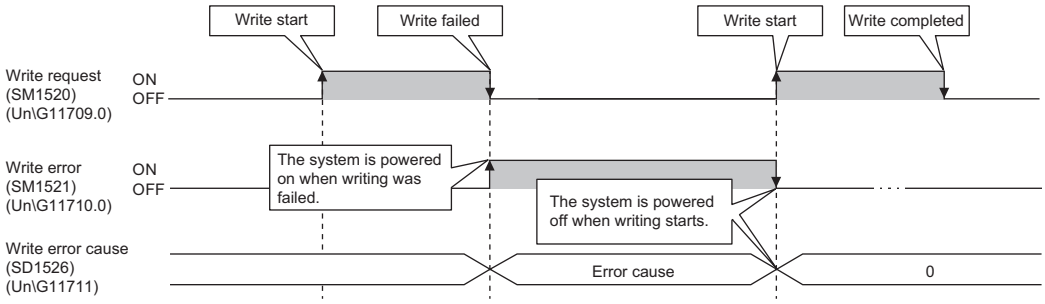
## ■Operation of the special relay and special register, or of the buffer memory

The following section shows the operation of the special relay and special register or of the buffer memory when writing to the IP address storage area.

- Operation at normal completion




- Operation at abnormal completion




## ■Cause of error at abnormal completion

- If the data is not written correctly into the IP address storage area (system memory) in the CPU module (built-in Ethernet port part), the cause of the error is stored in 'IP address storage area writing error cause' (SD1526).

| Stored value | Cause of error |
|---|---|
| 100H | The value for 'IP address setting' (SD1518 to SD1525) exceeds the setting range. |
| 200H | An error occurred during write. |
| 300H | Writing cannot be performed since the following functions are being executed.<br>• CPU module data backup/restoration function |
| 400H | Writing was started during the clear processing. |

- If the data is not written correctly into the IP address storage area (flash ROM) in the RJ71EN71 or the RnENCPU (network part), the cause of the error is stored in 'IP address storage area writing error cause' (Un\G11711).

| Stored value | Cause of error |
|---|---|
| 100H | The value for 'IP address' (Un\G11700 to Un\G11701) exceeds the setting range. |
| 101H | The value for 'Default gateway' (Un\G11704 to Un\G11705) exceeds the setting range. |
| 102H | The network address of 'Default gateway' (Un\G11704 to Un\G11705) or of the gateway IP address is different from that of the IP address of the own node. |
| 103H | The value for 'Network number' (Un\G11706) exceeds the setting range. |
| 104H | The value for 'Station number' (Un\G11707) exceeds the setting range. |
| 105H | The value for 'Transient transmission group number' (Un\G11708) exceeds the setting range. |
| 106H | The port 1 and port 2 IP addresses written in the IP address storage area are the same. |
| 107H | The port 1 and port 2 network numbers or station numbers written in the IP address storage area are the same. |
| 200H | An error occurred during write. |
| 400H | Writing was started during the clear processing. |

## Clearing operation

### ■Case of the CPU module (built-in Ethernet port part)

Execute clearing by turning off and on 'IP address storage area clear request' (SM1522).

1. Turn on 'IP address storage area clear request' (SM1522).
2. Check the write results with the following special relay and special register.

| Device | At normal completion | At abnormal completion |
|---|---|---|
| 'IP address storage area clear request' (SM1522) | Off | Off |
| 'IP address storage area clear error' (SM1523) | Off | On |
| 'IP address storage area clear error cause' (SD1527) | — | Stores the cause of the error |

3. Power off and on or set the CPU module.
4. Check the IP address with the buffer memory. (☞ Page 496 CPU module (built-in Ethernet port part))

### ■Case of the RJ71EN71 or RnENCPU (network part)

Execute clearing by turning off and on 'IP address storage area clear request' (Un\G11712.0).

1. Turn on 'IP address storage area clear request' (Un\G11712.0).
2. Check the write results with the following buffer memory areas.

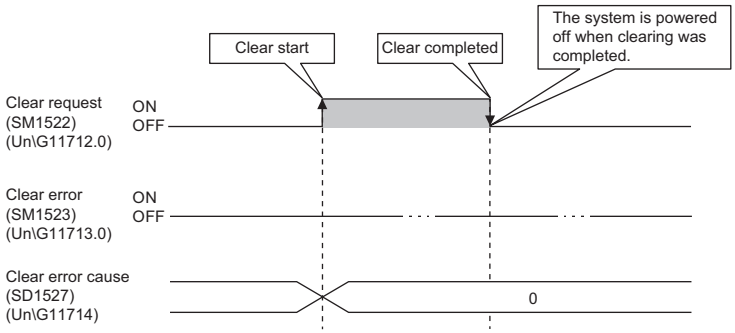
| Buffer memory | At normal completion | At abnormal completion |
|---|---|---|
| 'IP address storage area clear request' (Un\G11712.0) | Off | Off |
| 'IP address storage area clear error' (Un\G11713.0) | Off | On |
| 'IP address storage area clear error cause' (Un\G11714) | — | Stores the cause of the error |

3. Power off and on or set the CPU module.
4. Check the IP address with the buffer memory. (☞ Page 486 RJ71EN71, RnENCPU (network part))

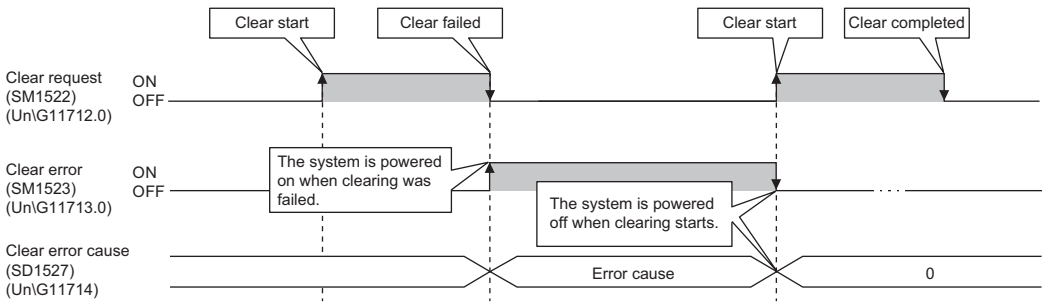
### ■Operation of the special relay and special register, or of the buffer memory

The following section shows the operation of the special relay and special register or of the buffer memory when clearing the IP address storage area.

• Operation at normal completion




• Operation at abnormal completion



### ■Cause of error at abnormal completion

If the IP address storage area is not cleared correctly, the cause of the error is stored as follows:

| Module | Storage location | Stored value | Cause of error |
|---|---|---|---|
| CPU module (built-in Ethernet port part) | 'IP address storage area clear error cause' (SD1527) | 200H | Error occurred during clearing. |
| | | 400H | Clearing was started during the write processing. |
| RJ71EN71, RnENCPU (network part) | 'IP address storage area clear error cause' (Un\G11714) | 200H | Error occurred during clearing. |
| | | 400H | Clearing was started during the write processing. |

# Checking the IP address

## Checking with the buffer memory

Check with the following buffer memory areas.

| Item | CPU module (built-in Ethernet port part) | RJ71EN71, RnENCPU (network part) |
|---|---|---|
| Own node IP address | Un\G50 to Un\G51 | Un\G4 to Un\G5 |
| Subnet mask | Un\G60 to Un\G61 | Un\G14 to Un\G15 |
| Default gateway IP address | Un\G64 to Un\G65 | Un\G18 to Un\G19 |
| Own node network number | Un\G77 | Un\G31 |
| Station number | Un\G78 | Un\G32 |
| Transient transmission group number | — | Un\G33 |

## Checking with module diagnostics

Check by using the [Module Information List] tab of the "Module Diagnostics" window. (☞ Page 423 Module Information List)

# Checking the operating status

For the RJ71EN71 or the RnENCPU (network part), when operation is performed based on the IP address of the IP address storage area, 'IP address change function operating status' (Un\G11715.0) turns on.

# Precautions

The following section lists the precautions for using the IP address.

## Power off and reset operation

Do not power off or reset when writing to or clearing the IP address storage area. The values may not be applied onto the IP address storage area. Power off or reset after checking the falling edge of the IP address storage area write request or IP address storage area clear request.

## Parameter IP address

For the IP address of the Ethernet-equipped module, the value in the IP address storage area has precedence over the parameter value. When using the IP address specified with the parameters, clear the IP address storage area.

## Write processing and clear processing execution timing

- Executing the write or clear processing to the IP address storage area may not be possible if an operation that turns off and on, or on and off an IP address storage area write request or IP address storage area clear request in a short time is performed.
- If the IP address storage area write request is turned off and on again while writing to the IP address storage area, the write processing that was executed first will complete normally, and the following write operation will be ignored. (This also applies to the clear operation.)
- If the IP address storage area clear request is turned off and on while writing to the IP address storage area, the clear operation will not be completed. (This applies to the write operation if writing is executed during the clear processing.)
- If both an IP address storage area write request and IP address storage area clear request are turned off and on, the write operation will be executed as top priority, and the clear operation will not be completed.
- If the write operation and clear operation are performed on the IP address storage area in the RJ71EN71 or the RnENCPU (network part), communication may be stopped. Execute requests while control is not being performed at P1 or P2.

## Communications by network number/station number

- For the network number, station number, and transient transmission group number (RJ71EN71 or RnENCPU (network part) only), the values in the IP address storage area have precedence over the module parameter values. To use the network number, station number, and transient transmission group number specified in the module parameters, clear the IP address storage area.
- When the IP address is stored in the IP address storage area, if the network number, station number, and transient transmission group number (RJ71EN71 or RnENCPU (network part) only) are not stored, communications using the network number and station number become impossible. For communications using the network number and station number, even if not changing the values, store the network number, station number, and transient transmission group number in the IP address storage area.

## Execution of the IP address change function during execution of another function

During execution of the CPU module data backup/restoration function, the IP address change function of the CPU module (built-in Ethernet port part) cannot be executed. Check that the CPU module data backup/restoration function is not being executed and then execute the IP address change function.

## Priority of IP addresses when re-initial processing is performed

While the IP address set with the IP address change function is being used for operation, if the IP address is changed through re-initial processing, the new IP address will be used thereafter.
After the CPU module is powered off and on or reset, the IP address set with the IP address change function is used again for operation.

# 1.18 Redundant System Function

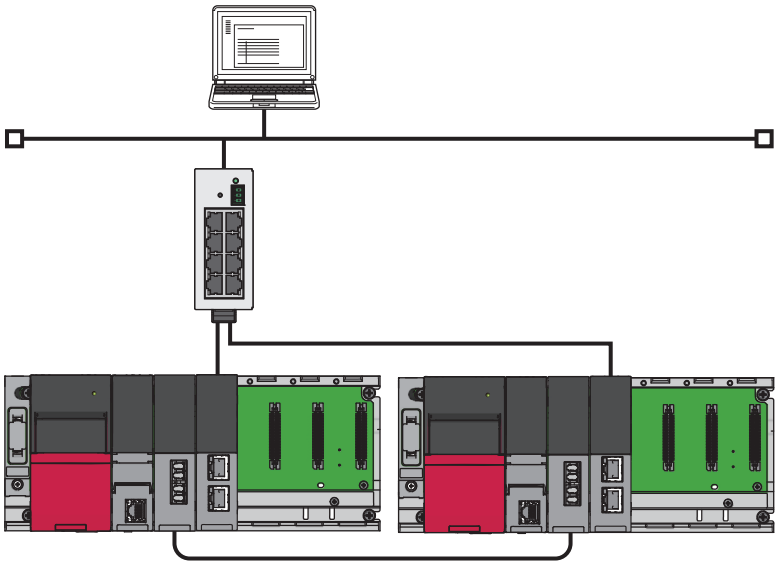
The system with a control system and a standby system is configured using the RJ71EN71. If an error occurs in the control system, the standby system is switched as the new control system to continue communication.
Before using function, check the firmware version of the RJ71EN71 and the CPU module used. (☞ Page 586 Added and Enhanced Functions)
When mounting the RJ71EN71 on the extension base unit in the redundant system with redundant extension base unit, refer to the following.
☞ Page 563 Using the Module in the Redundant System with Redundant Extension Base Unit

**Point**

To switch the system when a communication error or cable disconnection on Ethernet is detected, use the RJ71EN71.
The system is not switched when a communication error or cable disconnection in the CPU module (built-in Ethernet port part) occurs.

## System configuration

The Process CPU, redundant function module, and RJ71EN71 are mounted on the base units of both the system A and system B of a redundant system.
A redundant system is configured by connecting the redundant function modules of both systems using tracking cables and connecting the RJ71EN71s in both systems using Ethernet cables.



**Point**

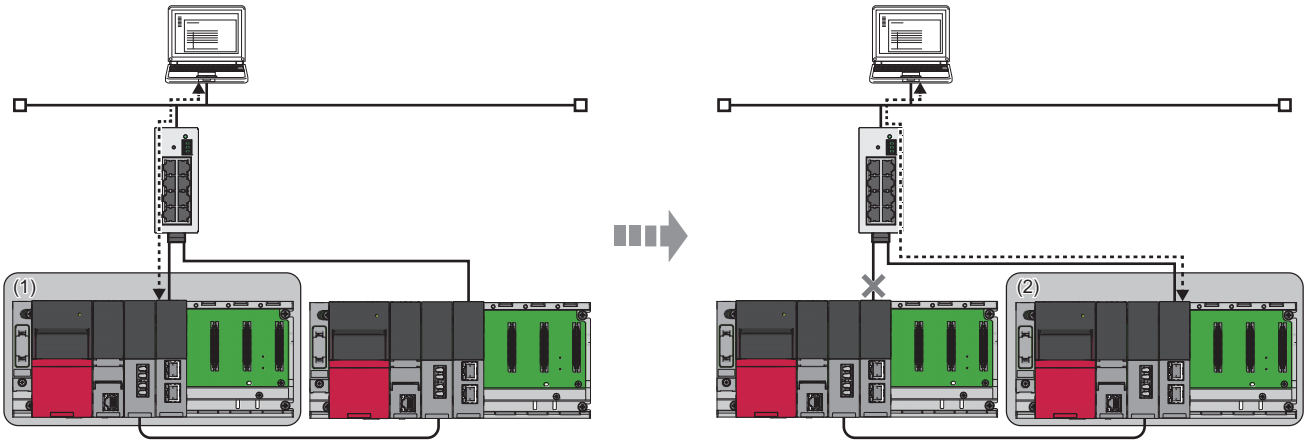
Set the network type for the RJ71EN71 to either of the following.
- "Port 1 Network Type" and "Port 2 Network Type" are set to "Ethernet".
- "Port 1 Network Type" is set to "Q Compatible Ethernet".

# System switching request

If an error of system switching cause occurs in the control system, the RJ71EN71 issues a system switching request to the control system CPU module to switch the system.

**Restriction**

The CPU module (built-in Ethernet port part) cannot issue a system switching request.




1. A system switching request occurs in the control system (1). (☞ Page 303 System switching request)
2. The RJ71EN71 of control system (1) issues a system switching request to the control system CPU module.
3. The system switches from the control system to the standby system.
4. Standby system (2) starts operating as the new control system to continue network control.

**Point**

For OPS connection, through establishing a TCP/IP connection with both systems, communication can be smoothly continued simply by changing the communication destination even when system switching occurs.

## Precautions

Depending on the operating status of the standby system, system switching is not executed.
For details on the operating status of the standby system and whether system switching is executable, refer to the following.
📖 MELSEC iQ-R Process CPU Module User's Manual
In the following cases, system switching is not executed even if the RJ71EN71 issues a system switching request.

- When one module in the redundant module group is operating normally (☞ Page 306 Redundant module group settings)
- When another port in the port group is operating normally (☞ Page 307 Port group setting)

# System switching cause

This section describes causes that make the RJ71EN71 issue a system switching request. Whether to issue a system switching request due to each cause is set in "Redundant System Settings" under "Application Settings". (☞ Page 416 Redundant System Settings)

**Point**

For details on system switching causes other than a system switching request from the RJ71EN71, refer to the following.

📖 MELSEC iQ-R Process CPU Module User's Manual

## Moderate/major error and hardware failure

When detecting an error where the module cannot continue the processing, the RJ71EN71 notifies the control system CPU module about the error and issues a system switching request.

## Communication error detection

Set connections targeted by communication error detection in "Redundant System Settings" under "Application Settings". (☞ Page 416 Redundant System Settings)

The following connections can be set as targets of communication error monitoring.

- User connections
- Auto-open UDP port
- FTP transmission port (TCP/IP)
- MELSOFT transmission port (TCP/IP)
- MELSOFT transmission port (UDP/IP)
- Simple CPU communication port

### ■Alive check error

When the RJ71EN71 has not communicated with the external device for a certain period of time while the connection is open, this function checks whether the external device is alive. (☞ Page 331 Existence Confirmation)

When no response is returned from the external device, the Ethernet-equipped module closes the relevant connection and issues a system switching request to the control system CPU module.

### ■ULP timeout

When the TCP/IP connection is open and when data is being sent, if no ACK response is returned from the external device for a period exceeding the TCP/ULP timer value set in the "Timer Settings for Data Communication" under "Application Settings", a send error occurs and a system switching request is issued to the control system CPU module. (☞ Page 401 Timer Settings for Data Communication)

### ■Precautions

- The auto-open UDP port and MELSOFT transmission port (UDP/IP) will be handled as targets of communication error monitoring only when the remote password setting is enabled and these ports are unlocked.
- Do not set connections with the following settings in module parameters as connections targeted by communication error detection. Otherwise, when a communication error occurs by cable disconnection detection or due to the power-off of the external device, system switching may continuously occur in a redundant system.

| Setting items | Description |
|---|---|
| "Own Node Settings" under "Basic Settings" | "Do Not Open by Program" is selected for "Opening Method". |
| "External Device Configuration" under "Basic Settings" | "UDP" is selected for "Protocol". |
| | "KeepAlive" or "UDP" is set for "Existence Confirmation". |

- Do not set connections with "MELSOFT Connection" set for "Communication Method" in "External Device Configuration" under "Basic Settings", as connections targeted by communication error detection. Otherwise, all MELSOFT products connected to the network will be targeted by communication error monitoring and external devices cannot be set.

## Disconnection detection

Monitor the connection status of the cable connected to the RJ71EN71 in the control system.
When the disconnection state continues even after the disconnection detection monitoring time has elapsed, a system switching request is issued to the control system CPU module. Disconnection detection occurs in any of the following cases.

- Disconnection between the RJ71EN71 and the hub
- Cable disconnection at the connector on the hub side
- Power-off of the hub
- Cable disconnection at the connector on the RJ71EN71 side

**Point**

- Cable connection status monitoring is started after link-up. Disconnection detection is not performed on a port that has never been linked up and a port unconnected to an Ethernet cable.
- Set a value in the range from several seconds to several tens of seconds for the disconnection detection monitoring time. If the disconnection detection monitoring time is set too short, system switching may occur due to such factors as noise.

# Redundant group setting

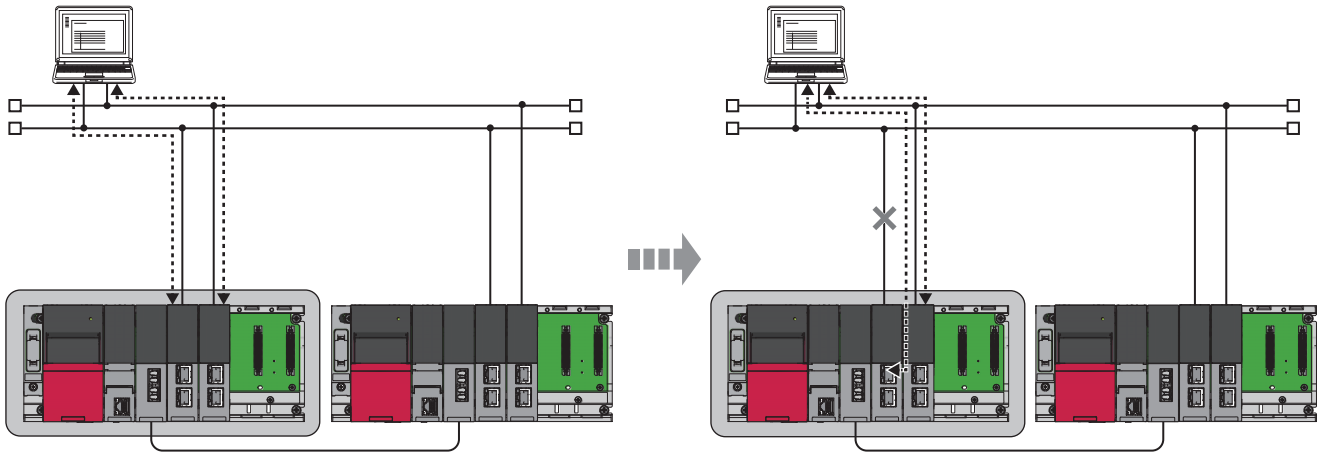
The redundant group setting is used to ensure that when the Ethernet network is duplicated in a redundant system, system switching is not executed for an error that occurs in one Ethernet network.
There are two types of redundant group setting: redundant module group setting and port group setting.

**Restriction**

The redundant group setting is not available for the CPU module (built-in Ethernet port part).

## Redundant module group settings

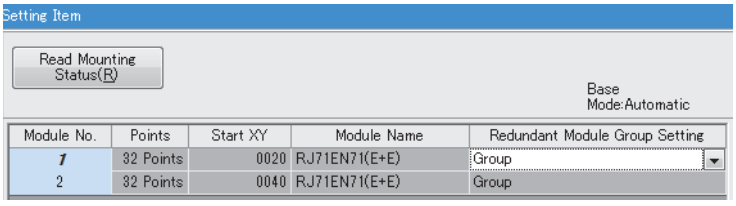
Two RJ71EN71s are mounted on each of the control system and standby system, so even if a communication error occurs in one of the RJ71EN71s, when communication is possible using the network of the other RJ71EN71, a system switching request is not issued and communication by the control system continues.
The redundant module group can be set only on modules that have adjacent start I/O numbers and are mounted on adjacent slots of the base unit.



When both grouped RJ71EN71s have a communication error, a system switching request is issued.

### ■Setting procedure

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment] tab ⇨ [Redundant Module Group Setting]




**Point**

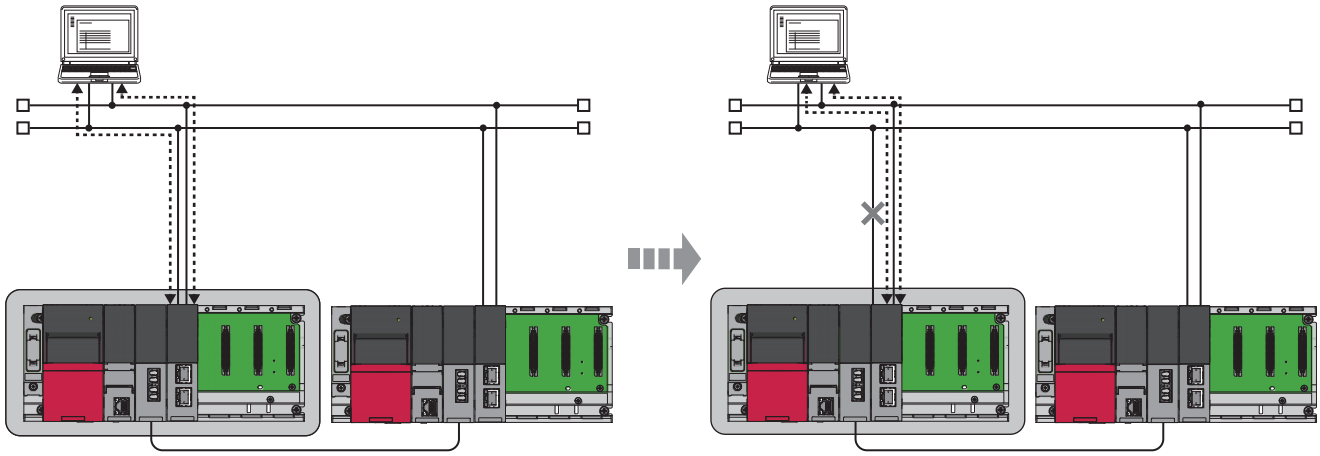
If the RJ71EN71s set as a redundant module group are not mounted adjacently to each other on the base unit, an error occurs. Check the mounting positions on the base unit and the start I/O number again.

## Port group setting

A cable is connected to each of the two Ethernet port of RJ71EN71, so even if a communication error occurs in one of the ports, when communication is possible on the other port, a system switching request is not issued and communication by the control system continues.

**Restriction**

This function cannot be used when "Q Compatible Ethernet" is selected in the network type.



### ■Setting procedure

Set the port group in "Redundant System Settings" under "Application Settings". (☞ Page 416 Redundant System Settings)

**Point**

- When "Switch Systems When a System Switching Error Occurs in Both of Port 1 and Port 2" is selected for "Port Group Setting", an event at the time of system switching is registered into the event history of the P1 side only.
- Set the same value for the port group setting on the P1 side and P2 side each.

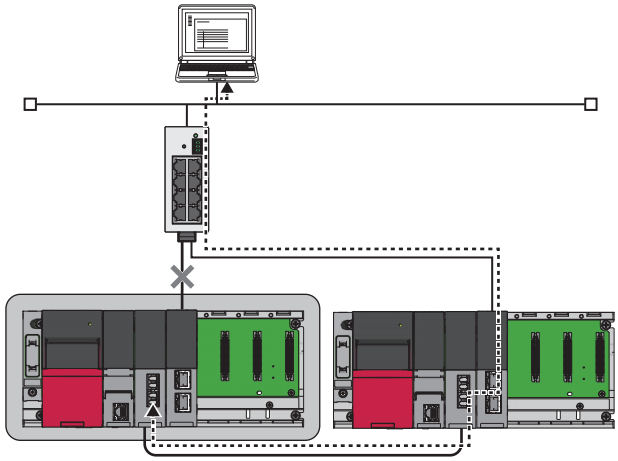
## Precautions

- The redundant group setting is enabled when the RJ71EN71 issues a system switching request. When a cause other than a moderate/major error or hardware failure in the RJ71EN71 occurs, or when a system switching request caused by other than the system switching cause from the RJ71EN71 occurs, systems are switched regardless of this setting.
- In "Redundant System Settings" under "Application Settings", set "Enable" for either "System Switching Request Issuing at Disconnection Detection" or "System Switching Request Issue at Communication Error". If "Disable" is set to the both settings, system does not switch because a system switching request is not issued to the CPU module.

# Communication path bypass function

When a communication error occurs between the application and Ethernet-equipped module supporting the following redundant system, the path on which the error occurs is bypassed so that communication can continue without changing the communication path setting.

- Applications operating on OPS
- GX Works3



**Restriction**

The communications path bypass function cannot be used for direct connection. Use for connection via a hub.

## Setting procedure

The bypass function is always enabled. The parameter setting is not required.

## Precautions

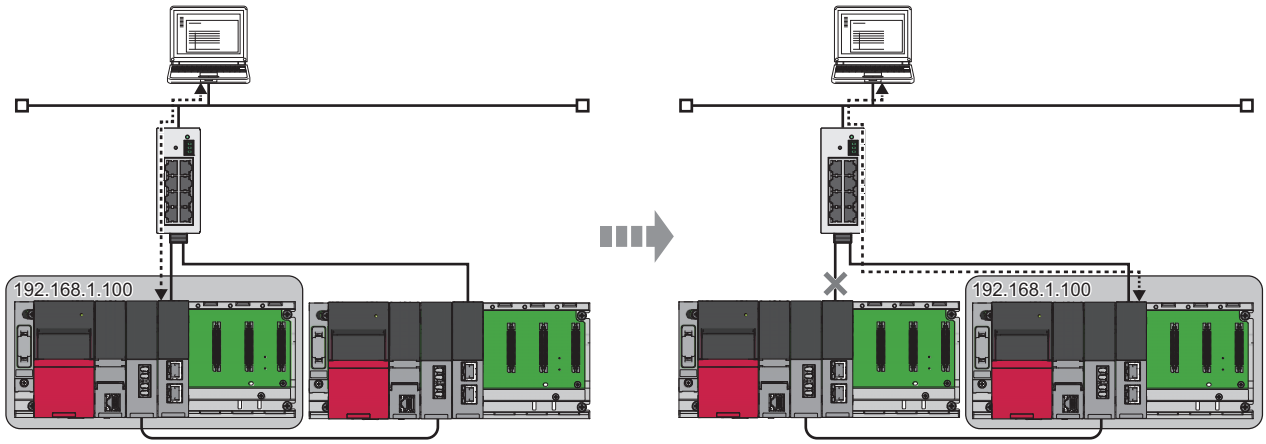
If an Ethernet-equipped module detects a disconnection or communication error, it does not release the relevant connection until the alive check function detects an error.

When a disconnection or communication error occurs continually, if the relevant connection has not been released, an Ethernet-equipped module bypasses the communication path by using a new connection.

For that reason, if a disconnection or communication error is detected over and over again in a short period of time, available connections run out and an error occurs in GX Works3.

# Both systems identical IP address setting function

When an IP address common to systems A and B (control system IP address) is used, even if system switching occurs, programs can perform communication without changing the connection destination.
Note that even when this setting is used, communication using the individual IP address of systems A and B each is possible.




**Restriction**

This control system IP address cannot be used when the network type for the RJ71EN71 is set to "Q Compatible Ethernet".

## Setting procedure

Set the control system IP address in "Redundant System Settings" under "Application Settings". (☞ Page 416 Redundant System Settings)

## Access range

When communicating using the control system IP address, access is only possible from an external device in the same network.
Since a network number and station number are used for access from another network, communication using the control system IP address is not possible.

## Availability of control system IP address for each function

The following table lists the availability of control system IP address for each function.

○: Available, △: Restrictions exist, ×: Not available

| Function | Availability | Precautions |
|---|---|---|
| Connection with MELSOFT products and a GOT | △ | • When accessing from MELSOFT products (excluding GX Works3), the control system IP address can be used. When accessing from GX Works3 and a GOT, use the IP address of system A or B.<br>• When system switching occurs during file access, file access is interrupted. When accessing files, use the IP address of system A or B.<br>• When system switching occurs due to a timeout or communication error in TCP/IP communications, the connection needs to be re-opened.<br>• When the control system IP address is used, neither the write operation nor the control operation can be performed. Use the IP address of system A or B. |
| SLMP communications | △ | • When system switching occurs during file access, file access is interrupted. When accessing files, use the IP address of system A or B.<br>• When system switching occurs due to a timeout or communication error in TCP/IP communications, the connection needs to be re-opened. |
| Communications using the predefined protocol | △ | • In TCP/IP communications, the control system IP address can be used only when the Ethernet-equipped module side is set as Passive open. When it is set as Active open, use the IP address of system A or B.<br>• In UDP/IP communications, the control system IP address can be used only when the Ethernet-equipped module receives data.<br>• When system switching occurs due to a timeout or communication error in TCP/IP communications, the connection needs to be re-opened.<br>• When using the protocol used for sending or sending/receiving data, use the IP address of system A or B. |
| Socket communications | △ | • In TCP/IP communications, the control system IP address can be used only when the Ethernet-equipped module side is set as Passive open. When it is set as Active open, use the IP address of system A or B.<br>• In UDP/IP communications, the control system IP address can be used only when the Ethernet-equipped module receives data.<br>• When system switching occurs due to a timeout or communication error in TCP/IP communications, the connection needs to be re-opened. |
| Communications using the fixed buffer | △ | |
| Communication using the random access buffer | △ | |
| Communications using a link dedicated instruction | ○ | For details on dedicated instructions, refer to the following. (☞ Page 310 How to use dedicated instructions) |
| File transfer function (FTP server) | △ | When system switching occurs during file access, file access is interrupted. When accessing files, use the IP address of system A or B. |
| Time setting function (SNTP client) | × | — |
| Security function | ○ | When using the remote password, after system switching occurs, the unlock processing needs to be performed again. |
| IP address change function | × | — |
| MODBUS/TCP communications | ○ | The device can be used when the firmware version of "42" or later is used. |
| Simple CPU Communication Function | △ | The device can be used when the firmware version of "42" or later is used.<br>When a control system IP address is set, it is used for the own station IP address; otherwise, the IP addresses of systems A and B are used. |

### ■How to use dedicated instructions

The following table lists how to use the control system IP address for dedicated instructions specifying an IP address.

| Dedicated instruction | Usage methods |
|---|---|
| GP.CONOPEN | Set the external device IP address in the control data for the control system IP address. |
| SP.SOCOPEN | |
| OPEN | |
| SP.SOCCINF | |
| SP.SOCCSET | |
| READ | Set the address specification method for the target station in the control data as the IP address, and set the control system IP address to the target station. |
| SREAD | |
| WRITE | |
| SWRITE | |

## Precautions

- When using the control system IP address, set connections separately for system A and B IP addresses. When the control system IP address and IP addresses of systems A and B are used together for the same connection, loss of data, data arrival order interchange, and others may be occur.
- Even when the control system IP address is used, the IP address on the programmable controller side shown in "External Device Configuration" under "Basic Settings" displays the IP address set in "Own Node Settings" under "Basic Settings".
- Do not set the control system IP address to the IP address in "PLC side I/F" of the "Specify Connection Destination Connection" window. Even when the control system IP address is set, if system switching occurs during connection, the new control system cannot be tracked.
- In TCP/IP communications, the control system IP address can be used only when the Ethernet-equipped module side is set as Passive open. When it is set as Active open, use the IP address of system A or B.
- In UDP/IP communications, the control system IP address can be used only when the Ethernet-equipped module receives data.
- When systems are switched, if communication using the control system IP address cannot be performed normally, perform the processing to update the ARP table of the external device.
- If the external device does not have a function to update the ARP table when receiving a GARP, it cannot track the new control system at the time of system switching, and communication becomes impossible until the ARP table of the external device is updated. The following table lists modules with function to update the ARP table when receiving a GARP.

| Applicable module | | Applicable version |
|---|---|---|
| MELSEC iQ-R series | Process CPU (redundant mode) | Firmware version of "04" or later |
| | RJ71EN71 | Firmware version of "12" or later |
| MELSEC-Q series | QJ71E71-100 | Serial number (first five digits) of "12062" or later |
| MELSEC-L series | LJ71E71-100 | — |

**Point**

For whether an external device not manufactured by Mitsubishi supports a function to update the ARP table when receiving a GARP, check the manual or specifications for the relevant external device.

# Functions restricted in a redundant system

Functions with restrictions when communicating with a redundant system over Ethernet are as follows.

## Re-initial processing

When performing re-initial processing using the UINI instruction in a program, specify "0H" for the reinitialization specification change target (S+2) in the control data to execute the instruction.

When performing re-initial processing, do not change the following parameter settings. Normal communication is not possible if the parameters are changed.

- "Own Node Settings" in "Basic Settings"
- "Redundant System Settings" in "Application Settings"

In addition, performing re-initial processing causes a temporary disconnection. If the value of "Disconnection Detection Monitoring Time" in "Redundant System Settings" is set shorter than the actual disconnection time, system switching will occur.

## Open/close processing

### ■Communications using TCP/IP

Set the Ethernet-equipped module side as Passive open and perform the open/close processing from the external device side. When Active open is performed from the Ethernet-equipped module side, if system switching occurs before execution of the close processing from the Ethernet-equipped module side, the close processing may become unexecutable.

### ■SLMP communications and communications using the random access buffer

When communicating with the standby system using user connection, set "Do Not Open by Program" for "Opening Method" in "Own Node Settings" under "Basic Settings".

### ■When the network type is "Q Compatible Ethernet"

For the standby system CPU module, the open/close processing by input/output signals cannot be performed.

## Direct connection with MELSOFT products and a GOT

When the network type is "Q Compatible Ethernet", execute direct connection with the engineering tool either using the CPU module (built-in Ethernet port part) or after setting "Disable" for "System Switching Request Issuing at Disconnection Detection" in "Redundant System Settings" under "Application Settings".

When "Enable" is set for "System Switching Request Issuing at Disconnection Detection", if a cable is disconnected after a direct connection with the engineering tool is made, system switching will occur.

## Searching the CPU module on the network

Even with the control system IP address set, when a CPU module search is performed, the control system IP address is not displayed. The IP addresses of systems A and B are displayed.

## SLMP communications

### ■Message format

For the request message, set 00H to the request destination network number and FFH to the request destination station number.

### ■Access to the control system, standby system, and systems A and B

For SLMP communications, using the command request destination module I/O number, the control system CPU, standby system CPU, system A CPU, or system B CPU can be specified as the access destination.

By specifying the access destination CPU module, even if system switching occurs due to a communication error, access to the specified CPU module is possible using the communications path bypass function. For details on the request destination module I/O number, refer to the following.

📖 SLMP Reference Manual

### ■When accessing other stations via a redundant system

When accessing other stations by dynamic routing via a redundant system on the same Ethernet, access via the control system. An error occurs if accessing via the standby system.

When accessing other stations via the standby system on the same Ethernet, set "Routing Setting" of "CPU Parameter".

### ■When the control system IP address is not used

When the control system IP address is not used, access the control system CPU module by using the following procedure.

1. Execute the open processing in the order system A → B.
2. Read 'Control system judgment flag' (SM1634) and 'Standby system judgment flag' (SM1635) to judge the control system.
3. Specify the control system CPU module for the request destination module I/O number and send a command message.
4. Check the response message.

To continue the communication, send command messages continuously.

5. Execute the close processing in the order the control system → the standby system.

## Communications using the predefined protocol

When the control system IP address is not used, establish the connection with the Ethernet-equipped modules of both systems and perform communications. When only sending data from the external device to the Ethernet-equipped module, send the data to the Ethernet-equipped modules of both the control and standby systems because the system switching timing is not able to be ascertained.

## Socket communications

### ■Receive processing by the standby system

If data is sent to the standby system Ethernet-equipped module, data reception processing will not be performed because the module reads and discards the received data.

### ■Receive processing with interrupt program

If system switching occurs before execution with the interrupt program, and the control system is switched to the standby system, the interrupt factor will not be transferred to the other system. The interrupt factor will be retained on the own system. Therefore, if system switching occurs again and the own system becomes the control system, the interrupt program will be executed by the retained interrupt factor.

### ■When data is sent from an external device

- When the control system IP address is not used, the system switching timing is not able to be ascertained. Send data to the Ethernet-equipped modules of both the control and standby systems.
- To use the control system IP address, check whether the Ethernet-equipped module is alive from the external device side.

## Communications using the fixed buffer

### ■Receive processing by the standby system

If data is sent to the standby system Ethernet-equipped module, data reception processing will not be performed because the module reads and discards the received data.

### ■Receive processing with interrupt program

If system switching occurs before execution with the interrupt program, and the control system is switched to the standby system, the interrupt factor will not be transferred to the other system. The interrupt factor will be retained on the own system. Therefore, if system switching occurs again and the own system becomes the control system, the interrupt program will be executed by the retained interrupt factor.

### ■When data is sent from an external device

The data sending method will differ according to whether the communication method and the control system IP address are used.

| Communication method | Control system IP address | Send method |
|---|---|---|
| Fixed Buffer (Procedure Exist) | Not Use | When system switching occurs because of a response timeout to the Ethernet-equipped module, send by changing the connection destination to the other system. |
| | Use | When system switching occurs because of a response timeout to the Ethernet-equipped module, open a new control system.<br>Also, check whether the Ethernet-equipped module is alive from the external device side. |
| Fixed Buffer (No Procedure) | — | System switching timing cannot be ascertained because of no setting for the response timeout. Send data to the Ethernet-equipped modules of both systems. |

**Point**

When performing the resend processing when system switching occurs, pay attention to the following.

- When communicating while synchronizing send and receive, system switching may occur in the middle of communication, and systems may be switched in a state where send and receive are not synchronized together. When system switching occurs, out of safety considerations, restart communication after initialization of synchronization.
- When using dedicated instructions, note that the completion of the execution of a write instruction needs to be executed is difficult to judge, and that the same instruction needs to be executed again. Note, however, that the same instruction may be executed twice.

## Communication using the random access buffer

Because tracking is not performed on the buffer memory of the Ethernet-equipped module, when writing data to the random access buffer, write the same data to the control and standby systems.

## Communications using a dedicated instruction

### ■System switching during dedicated instruction execution

When systems are switched during execution of a dedicated instruction, the dedicated instruction may not be completed. Execute the dedicated instruction again from the control system CPU module after system switching.

### ■When the control system IP address is used

When using dedicated instructions for the Ethernet-equipped modules of a redundant system, the same IP address can be set for both systems A and B by setting the control system IP address for the external device IP address in the control data. (☞ Page 309 Both systems identical IP address setting function)

### ■Specification of the target station CPU type in dedicated instructions

For a dedicated instruction in which the target station CPU type is specified as a control data, the control system CPU, standby system CPU, system A CPU, and system B CPU can be accessed by specifying the type. By specifying the access destination CPU module, even if system switching occurs due to a communication error, access to the specified CPU module is possible using the communications path bypass function. (📖 MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks, 📖 MELSEC iQ-R Programming Manual (Module Dedicated Instructions))

### ■Processing when a dedicated instruction abnormally ends

When the dedicated instruction is executed by specifying the access destination CPU module, it may be completed with an error if systems are switched in the target station. If the dedicated instruction was completed with an error, execute it again.

### ■SEND instruction

- When the target station is in a redundant system, the communication request source station must identify that the target station is the control system to execute the SEND instruction. When the target station is the standby system, the RECV instruction is not executed at the target station after data is sent by the SEND instruction, and the target station storage channel becomes occupied and unable to be used.
- When a redundant system exists in a network where broadcast communications are performed, the RECV instruction is not executed at the standby system, and the storage channel becomes occupied and unable to be used.

### ■RECV and RECVS instructions

- When the SEND instruction is executed for the control system, if the systems are switched before execution of the RECV instruction and interrupt program, the control system retains 'RECV instruction execution request' (Un\G5301) and the interrupt program interrupt factor (interrupt pointer). If system switching occurs again, and the standby system is switched to the control system, the RECV instruction and interrupt program will be executed by the retained 'RECV instruction execution request' (Un\G5301) and interrupt program interrupt factor.
- When the SEND instruction is executed for the standby system, and the standby system receives data from the sending station, the standby system retains 'RECV instruction execution request' (Un\G5301) and the interrupt program interrupt factor (interrupt pointer). Therefore, when the standby system is switched to the control system due to system switching, the RECV instruction and interrupt program will be executed by the retained 'RECV instruction execution request' (Un\G5301) and interrupt program interrupt factor.

### ■REQ instruction

When performing remote STOP or remote RUN on a redundant system, do as follows: Perform STOP on the standby system → STOP on the control system → RUN on the control system → RUN on the standby system.

## File transfer function (FTP server)

Perform file writing with the file transfer function (FTP server) as follows.

1. Log into the Ethernet-equipped modules of both systems.
2. Perform remote STOP on the CPU modules of both systems.*1
3. Write the file to the CPU modules of both systems.*2
4. Perform remote RUN on the CPU modules of both systems.
5. Log out from the Ethernet-equipped modules of both systems.
6. Clear the errors from the standby system CPU module.*3

*1 When the status of the control system CPU module is STOP state, a continuation error occurs in the standby system CPU module.

*2 When the parameter file has been written, the CPU modules must be reset.

*3 When the status of the control system CPU module changed from STOP to RUN state, check the error status of the standby system CPU module, and if an error has occurred, turn on 'Error reset' (SM50) to clear the error.

## Time setting function (SNTP client)

When the time is set by a program, an error may occur if system switching occurs in the timing in which the program is executed. If an error occurs, execute the program after system switching.

## IP filter

When using an IP filter, exclude the control system IP address, system A IP address, and system B IP address from the IP filter settings. Otherwise, operation may not be performed normally.

## IP address change function

Do not use the IP address change function to change IP addresses in a redundant system. Normal communication is not possible if an IP address has been changed. When the IP address is changed using the IP address change function, clear the IP address storage area.

## Communications with Different Networks

Precautions exist for communications with different networks via a redundant system. (☞ Page 555 When the networks contain a redundant system)

## MODBUS/TCP slave function

### ■When the control system IP address is not used

The processing to determine the present control system on the MODBUS/TCP master device side and to switch the communication destination (system A or system B) is required. Whether the communication destination is the control system or the standby system is determined by reading Control system judgment flag (SM1634) and Standby system judgment flag (SM1635) of the slave station CPU module.

### ■When the control system IP address is used

Once system switching is executed, TCP connection used before is disconnected.
Execute the re-connection processing from the MODBUS/TCP master device to restart TCP connection.

### ■When the data is written to the buffer memory (H)

The buffer memory of the Ethernet-equipped module is not tracked. When executing the MODBUS command for write to the MODBUS device assigned in the buffer memory (H), write the same data to the system A and system B.

## Simple CPU communications

### ■When system switching is executed

Communications stop temporarily due to execution of system switching, therefore communications may not execute within the "Execution Interval" specified in the "Simple CPU Communication Setting" under "Application Settings".

### ■When User setting area (Un\G722500 to Un\G1246787) is used

The buffer memory of the Ethernet-equipped module is not tracked. When writing the data to User setting area (Un\G722500 to Un\G1246787), write the same data to the system A and system B.

**Point**

Even if system switching is executed, the data of User setting area (Un\G722500 to Un\G1246787) is held in the system A and system B.

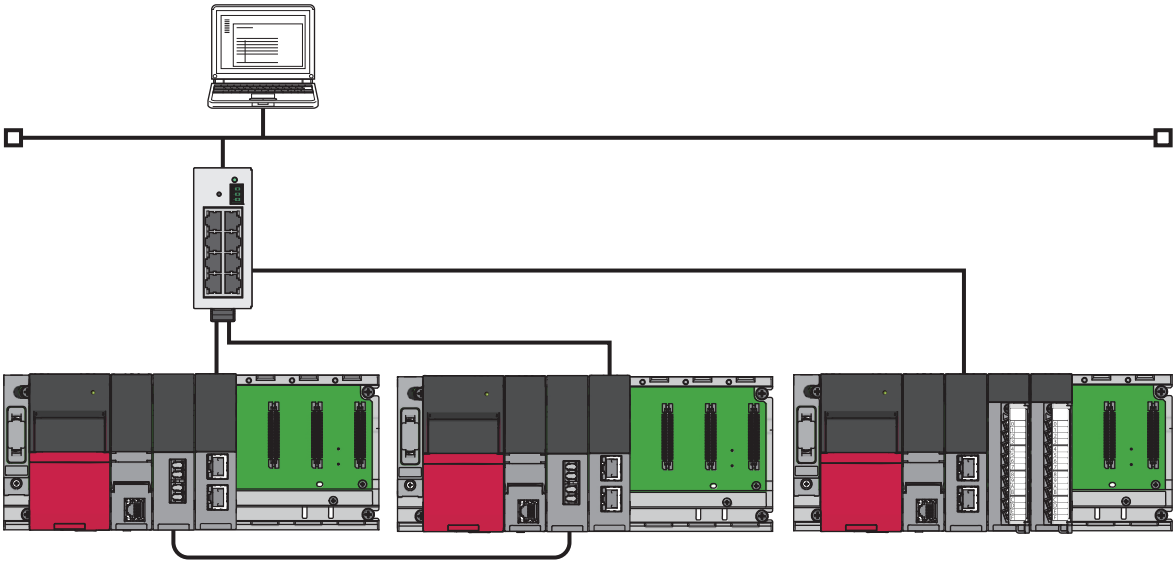
## Simple device communications

Since redundant systems are not supported, the operation is not guaranteed.

# Setting example

This section describes the setting details for socket communications in a redundant system.

## System configuration example



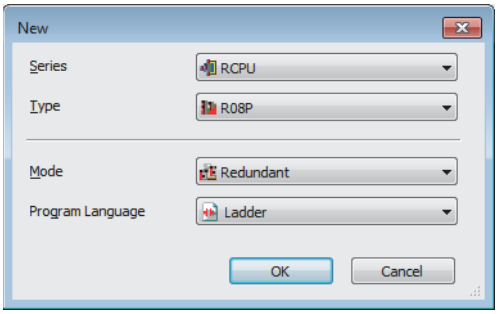
## Receiving side (redundant system) settings

Connect the engineering tool to the CPU module to be set as the control system and set the parameters.

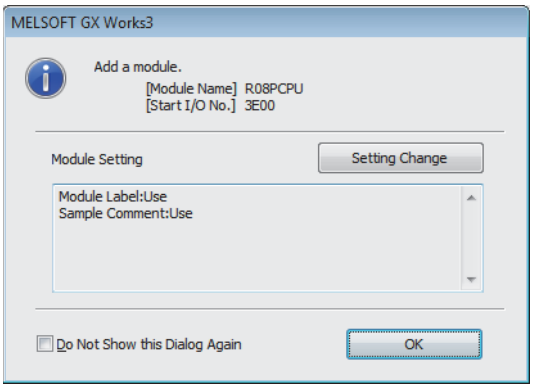
In a redundant system, the same program and parameter are written to both systems. A new project does not need to be created for the standby system.

***1.*** Set the CPU module as follows.

[Project] ⇨ [New]

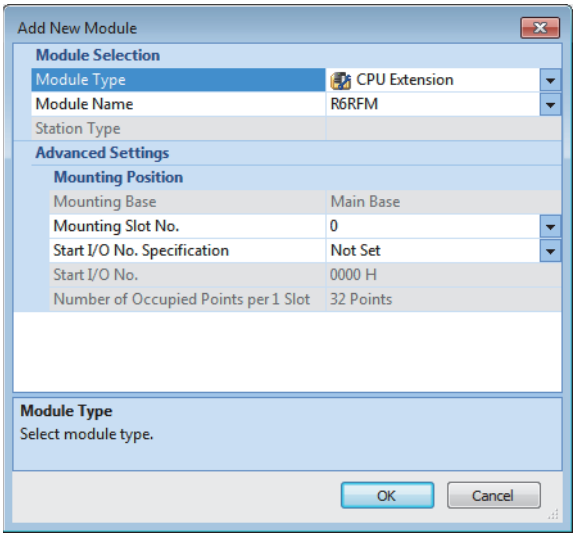



***2.*** Click the [OK] button to add the module labels of the CPU module.

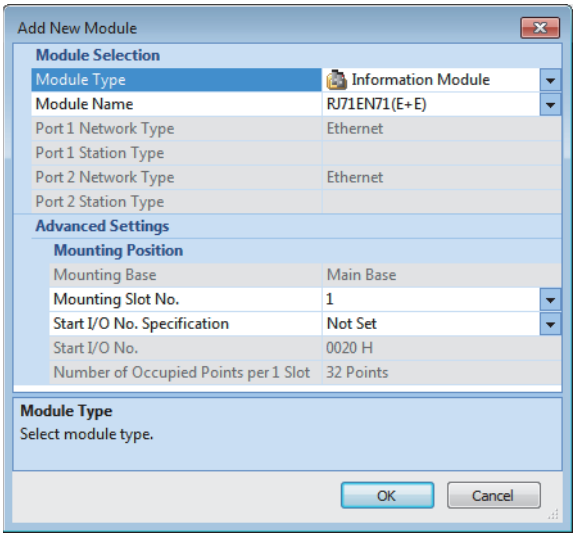


**3.** In the I/O assignment setting, set the redundant function module for slot No.0.

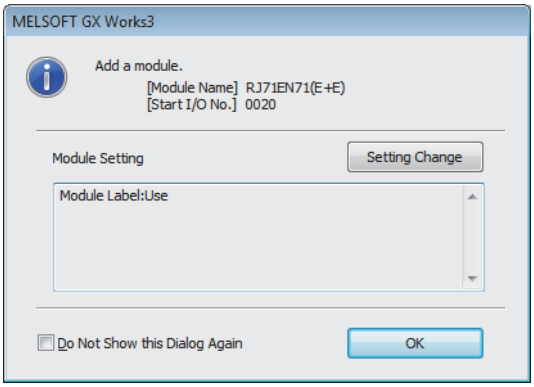
[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment] tab ⇨ [I/O Assignment Setting]




**4.** Set the RJ71EN71 for slot No.1




**5.** Click the [OK] button to add the module labels of the RJ71EN71.



**6.** Set the items in "Basic Settings" as follows.

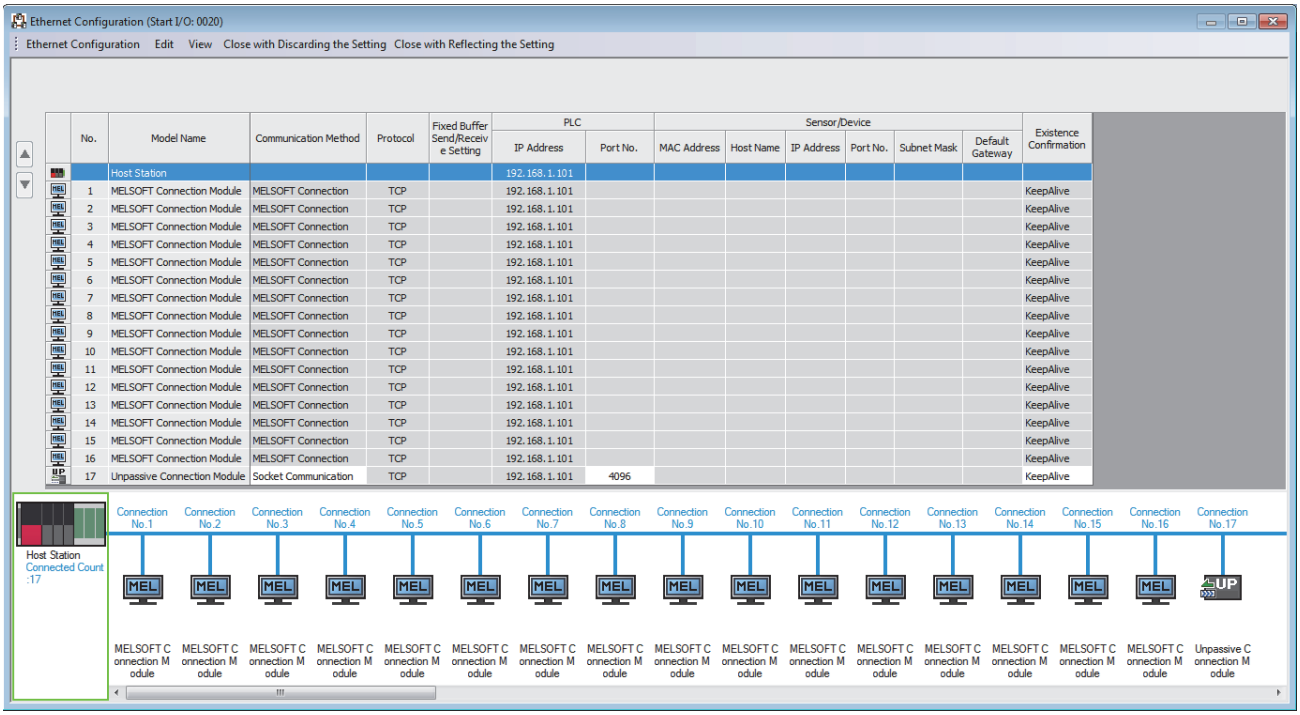
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings]

Setting Item

| Item | Setting |
|---|---|
| **Own Node Settings** | |
| Parameter Setting Method | Parameter Editor |
| IP Address | |
| IP Address | 192 . 168 . 1 . 101 |
| Subnet Mask | . . . |
| Default Gateway | . . . |
| Communications by Network No./Station No. | Disable |
| Setting Method | Use IP Address |
| Network Number | ----- |
| Station No. | ----- |
| Transient Transmission Group No. | 0 |
| Enable/Disable Online Change | Disable All (SLMP) |
| Communication Data Code | Binary |
| Opening Method | Do Not Open by Program |
| **External Device Configuration** | |
| External Device Configuration | <Detailed Setting> |

**7.** Set the network configuration as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [External Device Configuration]



**8.** Set the redundant settings in the following items.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Redundant System Settings]

Setting Item

| Item | Setting |
|---|---|
| **Redundant System Settings** | |
| Own Node (System A) IP Address | 192 . 168 . 1 . 101 |
| System B IP Address | 192 . 168 . 1 . 102 |
| Own Node (System A) Station Number | ----- |
| System B Station Number | ----- |
| Use of Control System IP Address | Use |
| Control System IP Address | 192 . 168 . 1 . 100 |
| Own Node (System A) Module Operation Mode | Online |
| System B Module Operation Mode | Online |
| Port Group Setting | No Setting |
| System Switching Request Issuing at Disconnection Detection | Enable |
| Disconnection Detection Monitoring Time | 2 |
| Unit | s |
| System Switching Request Issue at Communication Error | Enable |
| User Connection | <Detailed Setting> |
| System Connection | |
| Auto-open UDP Port | No System Switching Request at Communication Error |
| MELSOFT Transmission Port (UDP/IP) | No System Switching Request at Communication Error |
| MELSOFT Transmission Port (TCP/IP) | No System Switching Request at Communication Error |
| FTP Transmission Port (TCP/IP) | No System Switching Request at Communication Error |

**9.** Set whether to issue a system switching request for the user connection.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Application Settings] ⇨ [Redundant System Settings] ⇨ [System Switching Request Issue at Communication Error] ⇨ [User Connection]

User Connection

| Connection No. | System Switching Request | Communication Method | Protocol | Fixed Buffer Send/Receive Setting | Existence Confirmation | Comment |
|---|---|---|---|---|---|---|
| 1 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 2 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 3 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 4 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 5 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 6 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 7 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 8 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 9 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 10 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 11 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 12 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 13 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 14 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 15 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 16 | Not to Issue | MELSOFT Connection | TCP | | KeepAlive | |
| 17 | Issue | Socket Communication | TCP | | KeepAlive | |

OK Cancel

**10.** In the "System A/B Setting" window, set the redundant function module of own system to system A, and either reset the CPU modules of both systems or power off and on both systems.

[Online] ⇨ [Redundant PLC Operation] ⇨ [System A/B Setting]

System A/B Setting

Please set the connective system of redundant system.
Please set other system based on the setting of connective system.

System Status

Connective System: System Unknown

Other System: System Unknown

Set the Connective System

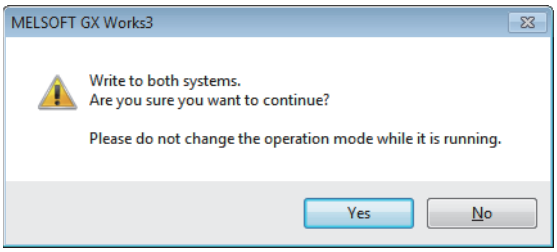
System A

Execute Close

**11.** Write the set parameters to the system A CPU module.

[Online] ⇨ [Write to PLC]

**12.** Click the [Yes] button to write to the CPU modules of both systems.




**13.** Either reset the CPU modules of both systems or power off and on both systems.

**Point**

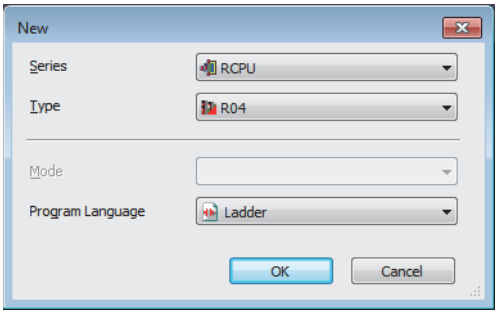
In this setting example, default settings are used for the parameters other than those described. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

## Sending side

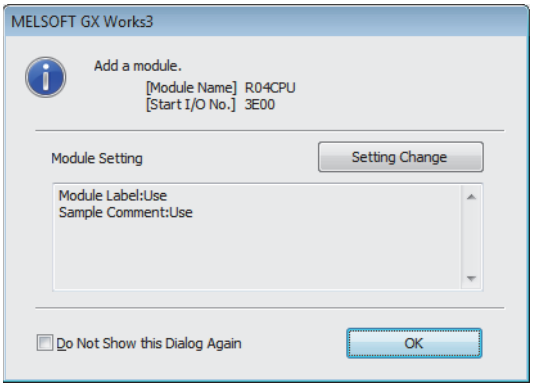
Connect the engineering tool to the CPU module on the sending side and set the parameters.

**1.** Set the CPU module as follows.

[Project] ⇨ [New]



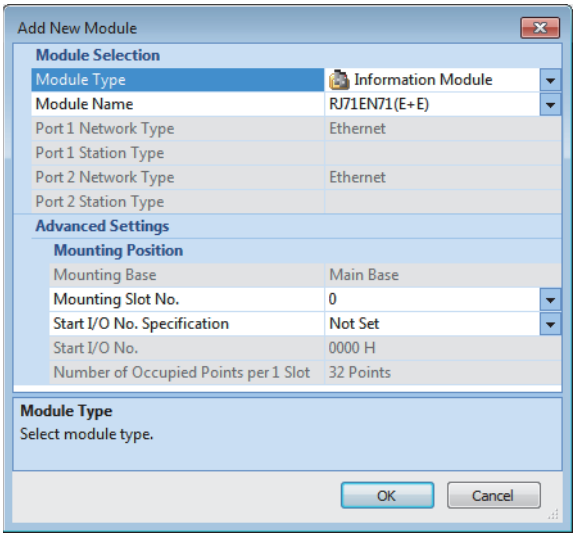

**2.** Click the [Setting Change] button in the following window and set the item to use module labels.



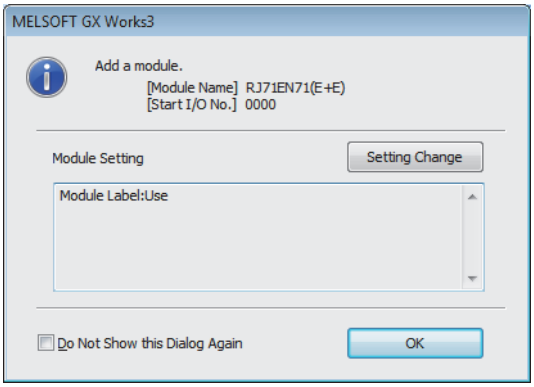

**3.** Click the [OK] button to add the module labels of the CPU module.

**4.** In the I/O assignment setting, set the RJ71EN71 for slot No.0.

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment] tab ⇨ [I/O Assignment Setting]

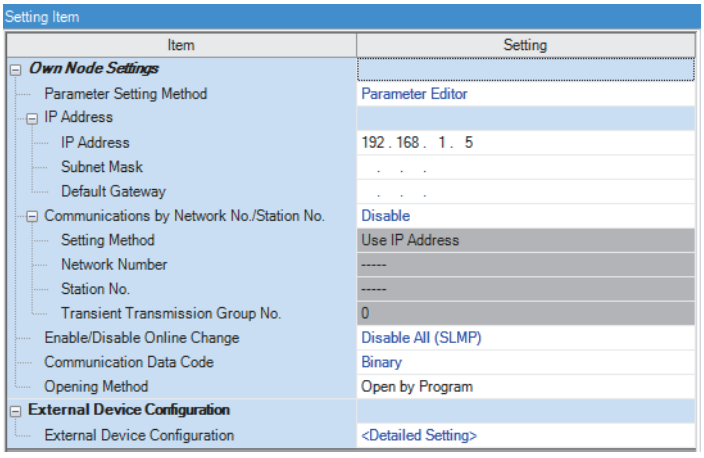



**5.** Click the [OK] button to add the module labels of the RJ71EN71.



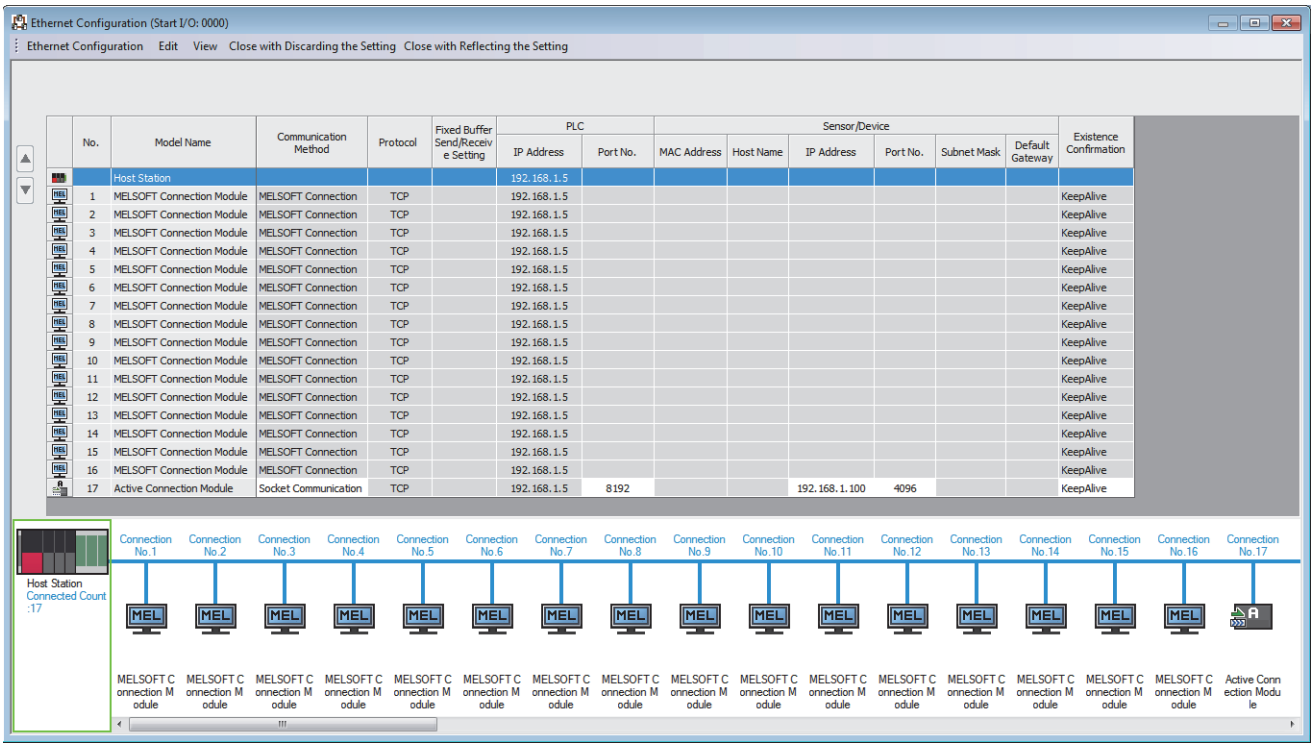

**6.** Set the items in "Basic Settings" as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings]



**7.** Set the network configuration as follows. The redundant system control system IP address is specified for the send destination IP address.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Port 1 Module Parameter (Ethernet)] ⇨ [Basic Settings] ⇨ [External Device Configuration]




**8.** Write the set parameters to the CPU module. Then reset the CPU module or power off and on the system.

[Online] ⇨ [Write to PLC]

**Point**

In this setting example, default settings are used for the parameters other than those described. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)

# 1.19 Discard Received Data at CPU STOP



The received data from the communication destination is discarded when the status of the CPU module is changed from RUN to STOP (PAUSE).

This function is available for below two communication methods:

- Communications using the fixed buffer
- Socket communications

## Setting procedure

This function can be enabled/disabled by directly changing the value in 'Discard received data at CPU STOP setting area' of buffer memory areas with a program.

### ■Discard received data at CPU STOP setting area

RJ71EN71, RnENCPU (network part): Un\G1950032 to Un\G1950039

CPU module (built-in Ethernet port part): Un\G275

# 2 PARAMETER SETTINGS

This chapter describes the parameter settings required for communications between the Ethernet-equipped module and external devices.

## 2.1 Setting Parameters

1. Add the RJ71EN71 in the engineering tool.*1

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ Right-click ⇨ [Add New Module]

2. The basic setting and application setting are included in the parameter settings. Select one of the settings from the tree on the window shown below.

[Navigation window] ⇨ [Parameter] ⇨ Target module ⇨ [Module Parameter]

3. After setting parameters, click the [Apply] button.
4. Write the settings to the CPU module using the engineering tool.

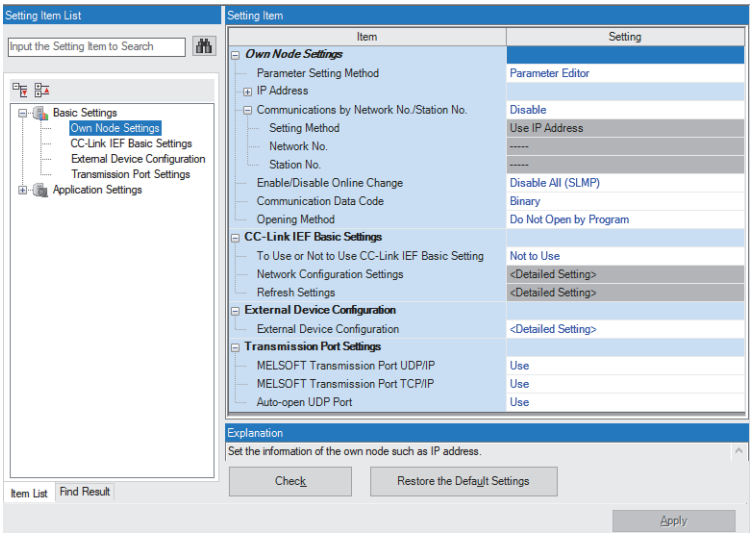
[Online] ⇨ [Write to PLC]

5. The settings are reflected by resetting the CPU module or powering off and on the system.

*1 Addition of the module is not required when using the Ethernet function of the CPU module (CPU part for the RnENCPU).

## 2.2 Basic Settings

Set the own node settings and external device configuration of the Ethernet-equipped module.




| Setting items | Description | Reference |
|---|---|---|
| Own Node Settings | Set the IP address and communication data code of the Ethernet-equipped module. | Page 327 Own Node Settings |
| CC-Link IEF Basic Settings*1 | Set the network configuration settings and refresh settings of CC-Link IE Field Network Basic. | CC-Link IE Field Network Basic Reference Manual |
| External Device Configuration | Set the method and protocol used for communicating with external devices. | Page 329 External Device Configuration |
| Transmission Port Settings*2 | Set the Ethernet ports used for communication and functions. | Page 332 Transmission Port Settings |

*1 This setting is displayed only for modules supporting CC-Link IE Field Network Basic.

*2 This setting is not available for the RJ71EN71.
This item is displayed in the settings for the RnENCPU (network part) or CPU module (built-in Ethernet port part).

# Own Node Settings

Set the IP address and communication data code of the Ethernet-equipped module.

| Item | | Description | Setting range |
|---|---|---|---|
| Parameter Setting Method | | Set the own station settings by parameters. | Parameter Editor (fixed) |
| IP Address | IP Address | Set the IP address of the own station. Set the IP address class within the range of classes A, B and C.<br>Ensure that the Ethernet-equipped module on the own station and the external device to be communicated with have the same class and subnet address. Consult with the network manager for the IP address setting.<br>When the parameters are written without the IP address set (blank), the following IP address is set.<br>• P1 connector of RJ71EN71: 192.168.3.40<br>• P2 connector of RJ71EN71: 192.168.4.40<br>• RnENCPU (network part): 192.168.3.40<br>• CPU module (built-in Ethernet port part): 192.168.3.39<br>When using the IP address in a redundant system, use as the system A IP address. | • Blank<br>• 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| | Subnet Mask | Set the subnet mask pattern of the default gateway when setting the IP address of the default gateway and communicating with an external device on another network via a router. All the devices on a subnetwork must have the same subnet mask. This setting is not required when communicating in single network. | • Blank<br>• 0.0.0.1 to 255.255.255.255<br>(Default: Blank) |
| | Default Gateway | Set the IP address of the default gateway (the device which the own node passes through to access a device of another network).<br>Set the value that satisfies the following conditions.<br>• The IP address class is any of A, B, and C.<br>• The subnet address of the default gateway is the same as that of the Ethernet-equipped module on the own station.<br>• The host address bits are not all "0" or all "1". | • Blank<br>• 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| Communications by Network No./Station No. | — | Select "Enable" to set the network number, station number, and transient transmission group number[*1].<br>This setting is not required when the following functions are not used.<br>• Connection with the MELSOFT products and the GOT (when connecting by specifying the network number and station number)<br>• Communications using the SLMP (when specifying the target station with its network number and station number)[*1]<br>• Communications using a link dedicated instruction[*1]<br>• Communications with different networks | • Disable<br>• Enable<br>(Default: Disable) |
| | Setting Method | Select the method for setting the network number and station number. (☞ Page 328 Setting Method) | • Use IP Address<br>• Not Use IP Address<br>(Default: Use IP Address) |
| | Network Number | Enter the network number of the own station when selecting "Not Use IP Address" in "Setting Method".<br>Set a unique network number. | 1 to 239<br>(Default: 1) |
| | Station No. | Enter the station number of the own station when selecting "Not Use IP Address" in "Setting Method". | 1 to 120<br>(Default: 1) |
| | Transient Transmission Group No.[*1] | Set the transient transmission group number of the own station. | 0 to 32<br>(Default: 0) |
| Enable/Disable Online Change | | Select whether to enable external devices to write data in SLMP communications while the CPU module is in RUN state. (☞ Page 328 Enable/Disable Online Change) | • Disable All (SLMP)<br>• Enable All (SLMP)<br>(Default: Disable All (SLMP)) |
| Communication Data Code | | Select the communication data code used for communications. | • Binary<br>• ASCII<br>(Default: Binary) |
| Opening Method | | Select how to open a connection when using UDP/IP communications or Passive open of TCP/IP communications.<br>• When "Do Not Open by Program" is selected, a connection is open when the system received the Active request. Program for open/close processing is not required.<br>• When "Open by Program" is selected, the open/close processing are performed by a program. The module cannot communicate when the CPU module is in STOP state. | • Do Not Open by Program<br>• Open by Program<br>(Default: Do Not Open by Program) |

*1 This setting is not available for the CPU module (built-in Ethernet port part).

## Details of items

### ■Setting Method

When "Use IP Address" is selected, network number and station number will be set from the third and fourth octet of the IP address.

For example, when IP address is set to 192.168.1.10, the network number is set to "1" and the station number is set to "10".

When the IP address is used, the value out of the range of the network number and station number cannot be set in the third octet and fourth octet

When "Not Use IP Address" is selected, set the network number and station number.

**Point**

When "Port 1 Network Type" and "Port 2 Network Type" are set to "Ethernet" for the RJ71EN71 and the same network number is set for both of P1 and P2, P1 is always used for relay to other networks.

### ■Enable/Disable Online Change

Select whether to enable SLMP data write command received by the Ethernet-equipped module received while the CPU module is in RUN state.

This setting is effective for not only the CPU module on the own station but also the ones on the other stations.

When "Disable All (SLMP)" is selected, data writing using SLMP is disabled while the CPU module is in RUN state.

When "Enable All (SLMP)" is selected, data writing using SLMP is enabled even when the CPU module is in RUN state.

To write data to the FTP server when the CPU module is in RUN state, set "Allow Online Change" under "FTP Server Settings" in "Application Settings". (☞ Page 338 FTP Server Settings)

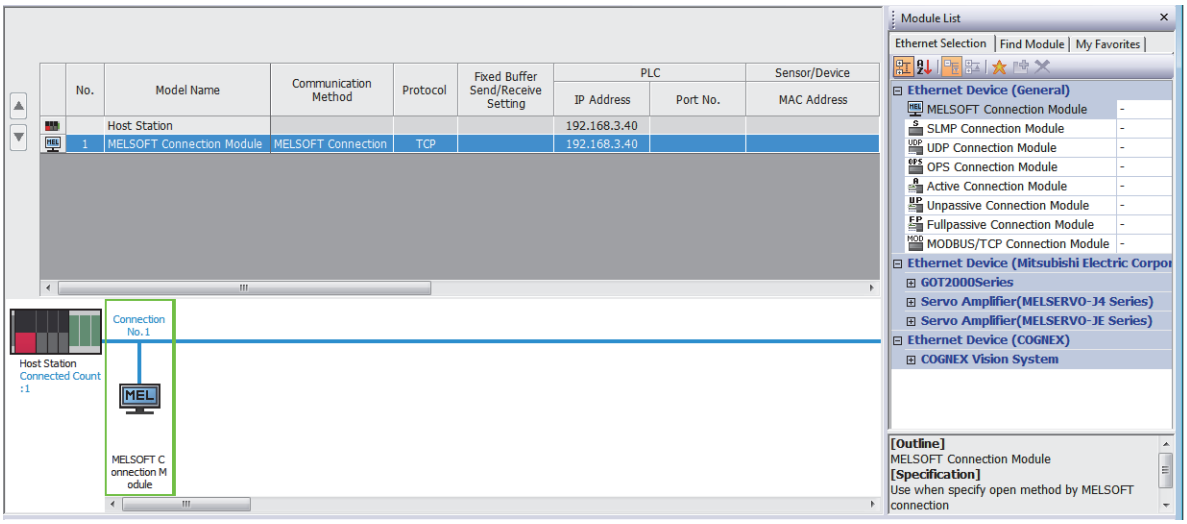
# External Device Configuration

Set the method and protocol used for communicating with external devices.

## Setting procedure

The procedure for setting the external device to be connected is shown below.

**1.** Select the external device to be connected in "Module List" and drag it to "List of devices" or "Device map area".




**2.** Set the required items. The required items vary depending on the selected external device.

**3.** Select [Close with Reflecting the Setting] to finish the external device configuration settings.

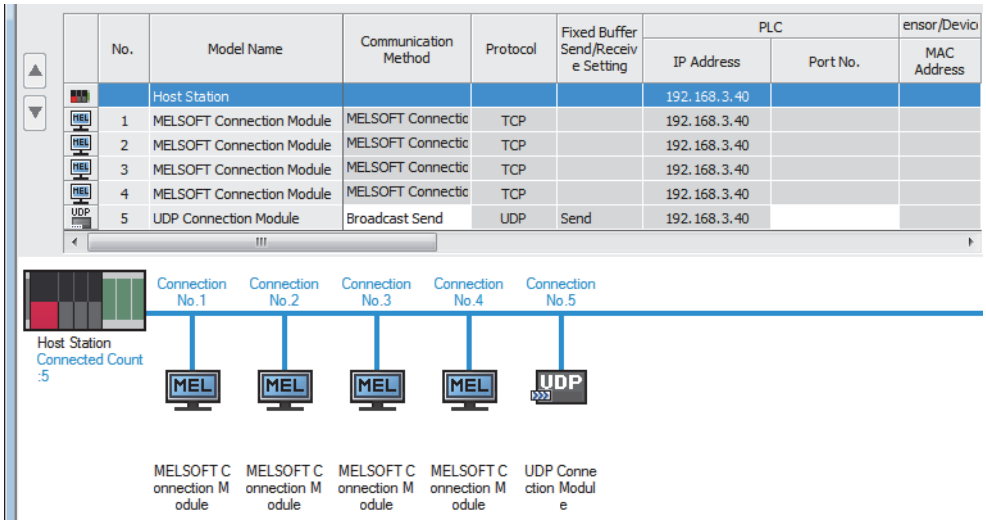
[Ethernet Configuration] ⇨ [Check] ⇨ [System Configuration]

**Restriction**

External devices must be set from the connection No.1

To use specific connection number, set "MELSOFT Connection Module" in the connection number not used.

• When only the connection No.5 is used.



## Setting items

| Item | | Description | Setting range |
|---|---|---|---|
| [Detect Now] button | | Execute the automatic detection of connected devices.<br>The [Detect Now] button is displayed only when the CPU module (built-in Ethernet port part) is used.<br>For details, refer to the following.<br>📖 iQ Sensor Solution Reference Manual | — |
| No. | | Connection number for distinguishing settings for each user connection. | The number is set in the following range starting with 1.<br>• RJ71EN71 and RnENCPU (network part): 1 to 64[*1]<br>• CPU module (built-in Ethernet port part): 1 to 16 |
| Model Name | | The name of the external device is displayed. | — |
| Communication Method | | Set the method for communication with the external device. | • Broadcast Send<br>• Broadcast Receive<br>• Fixed Buffer (Procedure Exist)[*2]<br>• Fixed Buffer (No Procedure)[*2]<br>• Random Access Buffer[*2]<br>• Predefined Protocol<br>• Socket Communication[*3]<br>• MELSOFT Connection<br>• SLMP<br>• OPS Connection<br>• MODBUS/TCP[*2] |
| Protocol | | Select the communication protocol for the external device. | • TCP[*4]<br>• UDP[*4] |
| Fixed Buffer Send/Receive Setting[*2] | | For communications using the fixed buffer, select whether to use the buffer for sending or for receiving in a connection to the external device. | • Send<br>• Receive<br>• Pairing (Receive)<br>• Pairing (Send) |
| PLC | IP Address | The IP address of the own node, which is set in "IP Address" under "Own Node Settings" of "Basic Settings", is displayed.<br>Or the IP address of the communication destination set in the device supporting iQSS is displayed when using the automatic detection of connected devices. | — |
| | Port No. | Set the port number for each connection of the Ethernet-equipped module.<br>When "Communication Method" is set to "MODBUS/TCP", set the same value to all port numbers. | 1 to 4999, 5010 to 65534<br>(Default: Blank)[*4] |
| Sensor/Device | MAC Address | The MAC address of the device supporting iQSS is displayed when using the automatic detection of connected devices. | — |
| | Host Name | Set the name for identifying the device supporting iQSS.<br>This setting is available only for the devices supporting iQSS which were detected by automatic detection of connected devices. | 63 characters maximum<br>The following one-byte characters can be used.<br>• Number (0 to 9)<br>• Alphabetical character (a to z, A to Z)<br>• Hyphen (-)<br>• Period (.)<br>• Colon (:)<br>• Underscore (_)<br>(Default: Blank)[*4] |
| | IP Address | Set the IP address of the external device. | 0.0.0.1 to 223.255.255.254, 255.255.255.255[*5]<br>(Default: Blank)[*4] |
| | Port No. | Set the port number of the external device.<br>Set "65535" to set all the port numbers as the target of data receive. | 1 to 65534, 65535<br>(Default: Blank)[*4] |
| | Subnet Mask | Set the subnet mask of the device supporting iQSS.<br>This setting is available only for the devices supporting iQSS which were detected by automatic detection of connected devices. | 192.0.0.0 to 255.255.255.252<br>(Default: Blank)[*4] |
| | Default Gateway | Set the default gateway of the device supporting iQSS.<br>This setting is available only for the devices supporting iQSS which were detected by automatic detection of connected devices. | 0.0.0.1 to 223.255.255.254<br>(Default: Blank)[*4] |

| Item | Description | Setting range |
|---|---|---|
| Existence Confirmation | Select the method of alive check which is performed when the Ethernet-equipped module has not communicated with the external device for a certain period of time. When the module cannot communicate with the external device, the connection will be closed. (☞ Page 331 Existence Confirmation) | • KeepAlive<br>• UDP<br>• Do not confirm existence |

*1 When "Q Compatible Ethernet" is selected in the network type, the setting range is 1 to 16.
*2 This setting is not available for the CPU module (built-in Ethernet port part).
*3 This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".
*4 When the automatic detection of the connected devices is executed, the values read from the connected devices will be the default.
*5 When "Communication Method" is set to "OPS Connection", 255.255.255.255 can be set.

**Point**

Comment can be set on the "Properties" window displayed by right-clicking the module in "List of devices" or "Device map area" and selecting "Properties". The following can be performed depending on the selected device.

- Changing the image
- Creating association with a file or application

## ■Existence Confirmation

When the Ethernet-equipped module has not communicated with the external device for a certain period of time while the connection is open, this function checks whether the external device is alive by sending an alive check message to the device and waiting for the response.

The following table lists the details on alive check.

| Item | Applicable protocol | Description |
|---|---|---|
| KeepAlive | TCP/IP | This method is used for a connection opened using TCP/IP. The Ethernet-equipped module performs an alive check by sending an alive check ACK message to the external device with which communications have not been performed for a certain period of time and waiting to see whether the response is received.<br>The connection will be automatically closed when the open state is not continued.*1 |
| UDP | UDP/IP | This method is used for a connection opened using UDP/IP. The Ethernet-equipped module performs an alive check by sending the PING command (ICMP echo request/response function) to the external device with which communications have not been performed for a certain period of time and waiting to see whether the response is received.*2 |
| Do not confirm existence | TCP/IP, UDP/IP | Alive check is not performed. |

*1 The connection may be disconnected if the external device does not support the TCP KeepAlive function (response to a KeepAlive ACK message).
*2 The Ethernet-equipped module automatically sends an echo response packet when it receives a PING echo request command. (It sends a response to the received PING command even if the connection used in the data communications with the external device is closed.)

If a response message cannot be received from the external device (or if an error has been detected) using the alive check function, the following are performed.

- The corresponding connection will be forcibly closed. (The line is disconnected.) Open the connection again using a user program.
- Open completion signal is turned off, and the error code is stored in the buffer memory areas.

# Transmission Port Settings

Set the Ethernet ports used for communication and functions.

**Restriction**

This setting is not available for the RJ71EN71.
This item is displayed in the settings for the RnENCPU (network part) or CPU module (built-in Ethernet port part).

| Item | Description | Setting range |
|---|---|---|
| MELSOFT Transmission Port UDP/IP | Select whether to use transmission port 5001 or 5006 for UDP/IP connection. | • Use<br>• Not Use<br>(Default: Use) |
| MELSOFT Transmission Port TCP/IP | Select whether to use transmission port 5007 for TCP/IP connection. | • Use<br>• Not Use<br>(Default: Use) |
| Auto-open UDP Port | Select whether to use transmission port 5005 for auto-open UDP. | • Use<br>• Not Use<br>(Default: Use) |

## Precautions

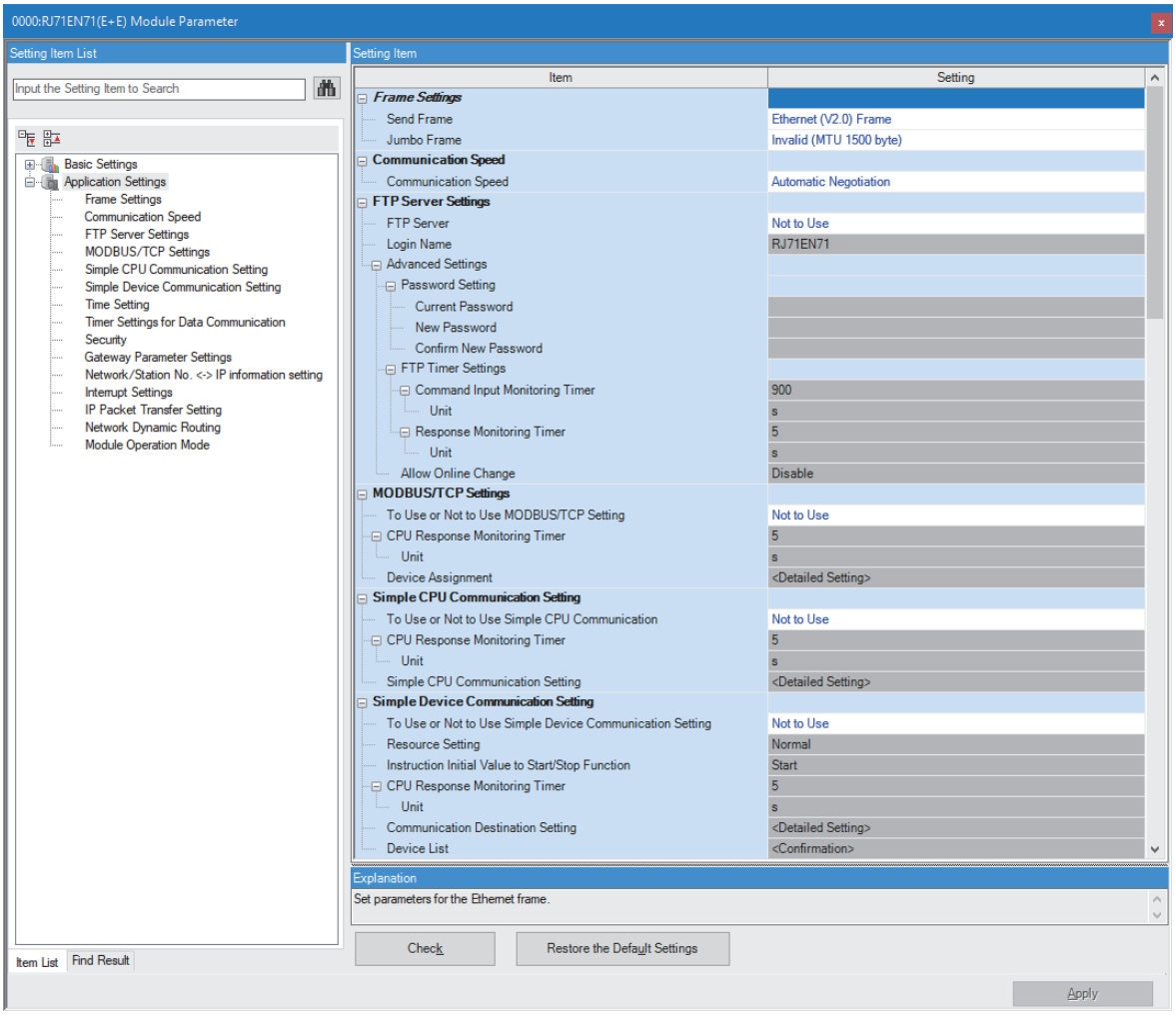
When all transmission port settings are set to "Not Use", communication with engineering tools by using Ethernet ports is not possible.
If communication using a port other than an Ethernet port (USB) or direct connection with MELSOFT is allowed, communicate with direct connection.

# 2.3 Application Settings

Set frame or communication speed of the Ethernet-equipped module.



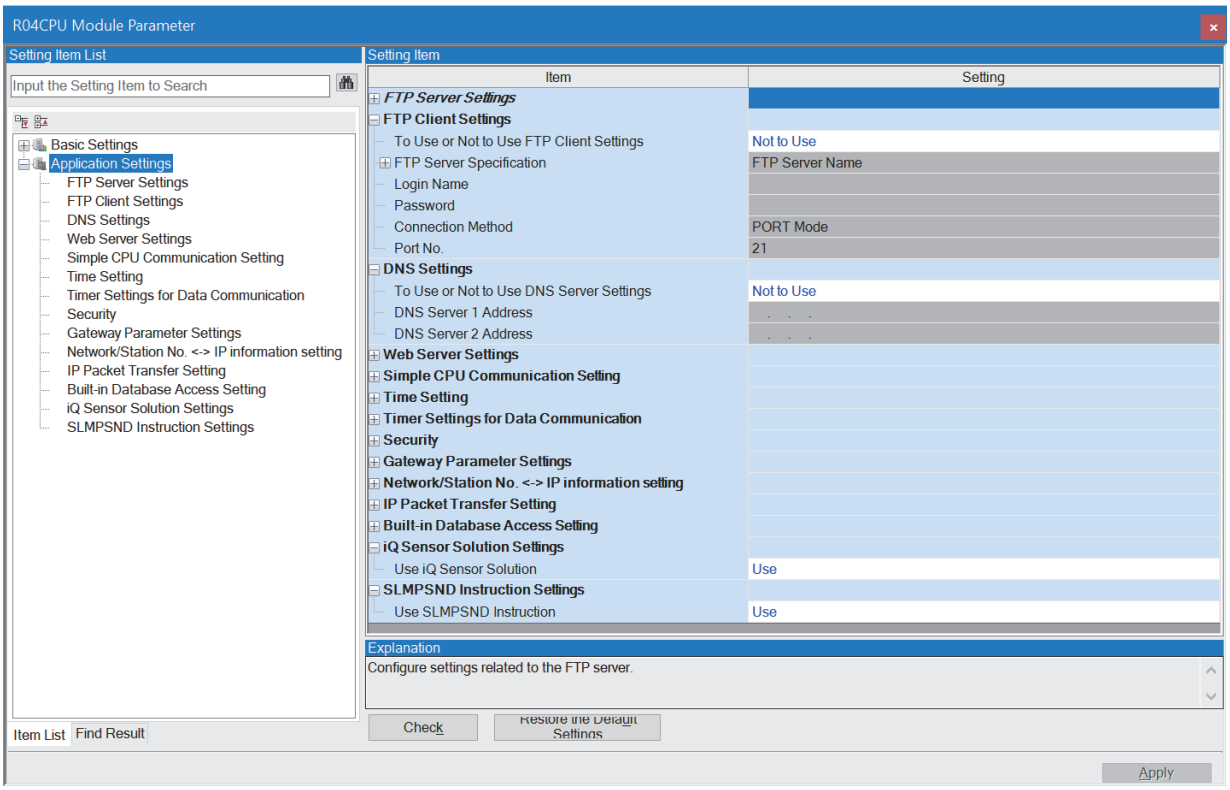
## RJ71EN71, RnENCPU (network part)



| Setting item | Description | Reference |
|---|---|---|
| Frame Settings | Set parameters for the Ethernet frame. | Page 336 Frame Settings |
| Communication Speed | Set the communication speed between the module and the external device. | Page 337 Communication Speed |
| FTP Server Settings | Set the file transfer function (FTP server). | Page 338 FTP Server Settings |
| MODBUS/TCP Settings*1 | Set MODBUS/TCP. | Page 340 MODBUS/TCP Settings |
| Simple CPU Communication Setting*1 | Set the simple CPU communication setting. | Page 342 Simple CPU communication setting (RJ71EN71, RnENCPU (network part)) |
| Simple Device Communication Setting*1 | Set the simple device communication setting. | Page 347 Simple Device Communication Setting |
| Time Setting*1 | Set the time setting function (SNTP client). | Page 400 Time Setting |
| Timer Settings for Data Communication | Set the timer used for the following communications.<br>• Connection with MELSOFT products and a GOT<br>• Communications using the SLMP<br>• Communications using the predefined protocol<br>• Socket communications/Communications using the fixed buffer<br>• Communication using the random access buffer<br>• File transfer function (FTP server)<br>• File Transfer Function (FTP Client) | Page 401 Timer Settings for Data Communication |
| Security*1 | Set the security measures for access to the Ethernet-equipped module. | Page 404 Security |
| Gateway Parameter Settings | Set to communicate with an external device on Ethernet via a router and gateway. | Page 404 Gateway Parameter Settings |
| Network/Station No. <-> IP information setting | Set to communicate with another network module or CPU module by network number and station number. | Page 407 Network/Station No. <-> IP information setting |
| Interrupt Settings | Set to start up an interrupt program. | Page 414 Interrupt Settings |
| IP Packet Transfer Setting*1 | Set the IP packet transfer function.<br>For details on the IP packet transfer function, refer to the following.<br>MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)<br>MELSEC iQ-R CC-Link IE Field Network User's Manual (Application) | Page 414 IP Packet Transfer Setting |
| Network Dynamic Routing*1 | Set the dynamic routing. | Page 415 Network Dynamic Routing |
| Module Operation Mode | Set the module operation mode. | Page 415 Module Operation Mode |
| Redundant system settings*2 | Set when using an RJ71EN71 in a redundant system. | Page 416 Redundant System Settings |

*1 This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".
*2 This item is displayed only when the module configuration for a redundant system is used.

## CPU module (built-in Ethernet port part)




| Setting item | Description | Reference |
|---|---|---|
| FTP Server Settings | Set the file transfer function (FTP server). | Page 338 FTP Server Settings |
| FTP Client Settings | Set the file transfer function (FTP client). | Page 339 FTP Client Settings |
| DNS Settings | Set the DNS setting. | Page 339 DNS Settings |
| Web Server Settings | Set the Web server function. (📖 MELSEC iQ-R/MELSEC iQ-F Web Server Function Guide Book) | — |
| Simple CPU Communication Setting | Set the simple CPU communication setting. | Page 345 Simple CPU communication setting (CPU module (built-in Ethernet port part)) |
| Time Setting | Set the time setting function (SNTP client). | Page 400 Time Setting |
| Timer Settings for Data Communication | Set the timer used for the following communications.<br>• Connection with MELSOFT products and a GOT<br>• Communications using the SLMP<br>• Communications using the predefined protocol<br>• Socket communications/Communications using the fixed buffer<br>• Communication using the random access buffer<br>• File transfer function (FTP server)<br>• File Transfer Function (FTP Client) | Page 401 Timer Settings for Data Communication |
| Security | Set the security measures for access to the Ethernet-equipped module. | Page 404 Security |
| Gateway Parameter Settings | Set to communicate with an external device on Ethernet via a router and gateway. | Page 404 Gateway Parameter Settings |
| Network/Station No. <-> IP information setting | Set to communicate with another network module or CPU module by network number and station number. | Page 407 Network/Station No. <-> IP information setting |
| IP Packet Transfer Setting | Set the IP packet transfer function.<br>For details on the IP packet transfer function, refer to the following.<br>📖 MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)<br>📖 MELSEC iQ-R CC-Link IE Field Network User's Manual (Application) | Page 414 IP Packet Transfer Setting |
| Redundant System Settings | Set the redundant system function.<br>Only functions that meet usage conditions with a redundant system can be set.<br>(☞ Page 302 Redundant System Function) | Page 416 Redundant System Settings |
| Built-in database access setting | Set the function of CPU module built-in database access from an external device.<br>📖 Manual for the CPU module used | — |
| iQ Sensor Solution Settings | Set the setting related to iQ Sensor Solution. | Page 419 iQ Sensor Solution Settings |
| SLMPSND Instruction Settings | Set the setting related to the SLMPSND instruction. | Page 419 SLMPSND Instruction Settings |

# Frame Settings

Set parameters for the Ethernet frame.

| Item | Description | Setting range |
|---|---|---|
| Send Frame | Set the frame of the Ethernet header for the data link layer to be sent by the RJ71EN71 or the RnENCPU (network part) to the Ethernet (V2.0) frame. | Ethernet (V2.0) Frame (fixed) |
| Jumbo Frame*1 | Set the frame size of the data to be sent over Ethernet.<br>Set when the data transmission speed is 1Gbps and communication is not efficient because the frame size is too small. | • Invalid (MTU 1500 byte)<br>• 2KB (MTU 2034 byte)<br>• 3KB (MTU 3058 byte)<br>• 4KB (MTU 4082 byte)<br>• 5KB (MTU 5106 byte)<br>• 6KB (MTU 6130 byte)<br>• 7KB (MTU 7154 byte)<br>• 8KB (MTU 8178 byte)<br>• 9KB (MTU 9004 byte)<br>(Default: Invalid (MTU 1500 byte)) |

*1 This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".

## Jumbo Frame

The following describes the precautions for using jumbo frames.

- In TCP/IP communications, the setting is enabled when the external device supports jumbo frames. If the external device does not support jumbo frames, the Ethernet-equipped module sends data in normal frame size even though the "Jumbo Frame" is enabled.
- In TCP/IP communications, data sending from the external device may fail when "Jumbo Frame" of the Ethernet-equipped module is set to "Invalid (MTU 1500 byte)" and that of the external device is enabled. When communicating in jumbo frames, enable the jumbo frame setting for both of the Ethernet-equipped module and external device.
- In UDP/IP communications, the Ethernet-equipped module sends data in jumbo frames according to the "Jumbo Frame" even though the external device does not support jumbo frames.
- When the hub on the communication path does not support jumbo frames, the Ethernet-equipped module cannot communicate with the external device because the hub discards the packets even if the module sends data in jumbo frame.
- Data can be sent in jumbo frames only when the transmission speed is 1Gbps. When "Automatic Negotiation" in "Communication Speed" is set to "Jumbo Frame" and the Ethernet-equipped module communicates at a speed less than 1Gbps, the module sends data in the standard frame size.
- When the item other than "Automatic Negotiation" or "1Gbps/Full-Duplex" is set in "Communication Speed" of "Application Settings", "Jumbo Frame" cannot be set to other than "Invalid (MTU 1500 byte)".

# Communication Speed

Select the communication speed and communication mode from the following. (Default: Automatic Negotiation)
When "Automatic Negotiation" is selected, communication speed will be automatically set to be the same as that of another device connected to the network.

- Automatic Negotiation
- 1Gbps/Full-Duplex*1
- 100Mbps/Half-Duplex
- 100Mbps/Full-Duplex
- 10Mbps/Half-Duplex
- 10Mbps/Full-Duplex

*1 This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".
Select "Automatic Negotiation" to use 1Gbps/full-duplex.

**Point**

- When the auto-negotiation function is used, there is a time lag before starting communication because it requires the time to determine the communication speed and communication mode by FLP burst.
- It is recommended to set the auto-negotiation also in the communication destination when using the auto-negotiation function. If auto-negotiation is not set in ether of the Ethernet-equipped module or partner external device, communication mode cannot be identified and it is automatically set to half-duplex mode, resulting in unstable communication.
- Set the lower communication speed when communication fails due to noise or other causes.

# FTP Server Settings

Set the file transfer function (FTP server).

| Item | | Description | Setting range |
|---|---|---|---|
| FTP Server | | Select whether to use the file transfer function (FTP server) of the Ethernet-equipped module. | • Not Use<br>• Use<br>(Default: Not Use) |
| Login Name | | Set the login name to be used for file transfer request (login) from the external device. | 12 characters maximum (one-byte alphanumeric character)<br>[Default]<br>• RJ71EN71 and RnENCPU (network part): RJ71EN71<br>• CPU module (built-in Ethernet port part): RCPU |
| Advanced Settings | Password Setting | Set the password to be used for file transfer request (login) from the external device to the Ethernet-equipped module. (☞ Page 338 Password Setting) | — |
| | FTP Timer Settings | Set the following timers used for the file transfer function (FTP server). (☞ Page 338 FTP Timer Settings)<br>• Command Input Monitoring Timer<br>• Response Monitoring Timer | — |
| | Allow Online Change | Select whether to enable data writing from the external device using the file transfer function (FTP server) while the CPU module is in RUN state. | • Disable<br>• Enable<br>(Default: Disable) |

## Password Setting

### ■Current Password

Enter the current password for login to the Ethernet-equipped module.

Default password (initial setting) is the following.

| Module | Password |
|---|---|
| RJ71EN71, RnENCPU (network part) | RJ71EN71 |
| CPU module (built-in Ethernet port part) | RCPU |

Although the default password can be used, it is recommended to change it to another password to prevent unauthorized access.

### ■New Password, Confirm New Password

Enter the new password in "New Password" and "Confirm New Password" when changing the password.

Set a password within 0 to 32 one-byte characters. Number, alphabet, special character (?,!&\%#*) can be used.

## FTP Timer Settings

### ■Command Input Monitoring Timer

Set the monitoring time for the Ethernet-equipped module to monitor the command input time from the FTP client

It is recommended to use the default value (900s) for this timer value as much as possible.

When changing the setting value, determine the command input monitoring timer value upon consulting with the administrator of the external device or system.

Set a value within the following range.

| Unit | Setting range |
|---|---|
| s | 1 to 16383 |
| ms*1 | 100 to 16383000 |

*1 Set in increments of 100ms.

The FTP connection is disconnected if there is no command input from the FTP client side within the time of the command input monitoring timer value after the FTP client login.

When restarting the file transfer, start over from the login operation again.

### ■Response Monitoring Timer

Set the monitoring time for a response from the CPU module after the Ethernet-equipped module receives the request data from the external device and requests the CPU module for read/write.

It is recommended to use the default value (5s) for this timer value as much as possible.

When changing the setting value, determine the response monitoring timer value upon consulting with the system administrator.

Set a value within the following range.

| Unit | Setting range |
|---|---|
| s | 1 to 16383 |
| ms | 100 to 16383000 (in increments of 100ms) |

## FTP Client Settings

Set the file transfer function (FTP client).

| Item | Description | Setting range |
|---|---|---|
| To Use or Not to Use FTP Client Settings | Set whether to use the file transfer function (FTP client) or not. | • Not Use<br>• Use<br>(Default: Not Use) |
| FTP Server Specification | Set the specification method of the FTP server. | • FTP Server Name<br>• FTP Server IP Address<br>(Default: FTP Server Name) |
| FTP Server Name | Set the FTP server name (domain name) of the connection destination. | 1 to 256 characters |
| FTP Server IP Address | Set the IP address of the FTP server of the connection destination in the decimal format. | 0.0.0.1 to 223.255.255.254 |
| Login Name | Set the login name for login to the FTP server. | 1 to 32 characters |
| Password | Set the password for login to the FTP server. | 0 to 32 characters |
| Connection Method | Set the connection method of the FTP server. | • PORT Mode<br>• PASV Mode<br>(Default: PORT Mode) |
| Port No. | Set the port number for control of the FTP server. | 1 to 65535<br>(Default: 21) |

## DNS Settings

Set the IP address of the DNS server.

Set the IP address in the following cases.

- When "FTP Server Specification" in "FTP Client Settings" is set to "FTP Server Name"
- When the FTP server is specified with the server name for the data logging file transfer function.
- When the FTP server name is specified with the control data of the file transfer function instruction (SP.FTPPUT/SP.FTPGET)

| Item | Description | Setting range |
|---|---|---|
| To Use or Not to Use DNS Server Settings | Set whether to use the DNS server or not. | • Not Use<br>• Use<br>(Default: Not Use) |
| DNS Server 1 Address | Set the IP address of the DNS server 1 in the decimal format. | 0.0.0.1 to 223.255.255.254 |
| DNS Server 2 Address | Set the IP address of the DNS server 2 in the decimal format. | 0.0.0.1 to 223.255.255.254 |

**Point**

When "To Use or Not to Use DNS Server Settings" is set to "Use", set either one or both "DNS Server 1 Address" and "DNS Server 2 Address".

# MODBUS/TCP Settings

Set MODBUS/TCP.

| Item | Description | Setting range |
|---|---|---|
| To Use or Not to Use MODBUS/TCP Setting | "Use" is displayed when a MODBUS/TCP connection device is added in "External Device Configuration" under "Basic Settings". ("To Use or Not to Use MODBUS/TCP Setting" cannot be edited.) | • Not Use<br>• Use<br>(Default: Not Use) |
| CPU Response Monitoring Timer | Set the time for monitoring the response from the CPU module.<br>(☞ Page 340 CPU Response Monitoring Timer) | — |
| Device Assignment | Set the parameters to associate the MODBUS devices with the device memory areas of the CPU module.<br>(☞ Page 341 Device Assignment) | — |

## CPU Response Monitoring Timer

Set the time for monitoring the response from the CPU module.

When the CPU module does not send a response within the set time after the reception of a request message from the MODBUS/TCP master device, the state of waiting for the response from the Ethernet-equipped module is canceled.

Set the CPU Response Monitoring Timer within the following range.

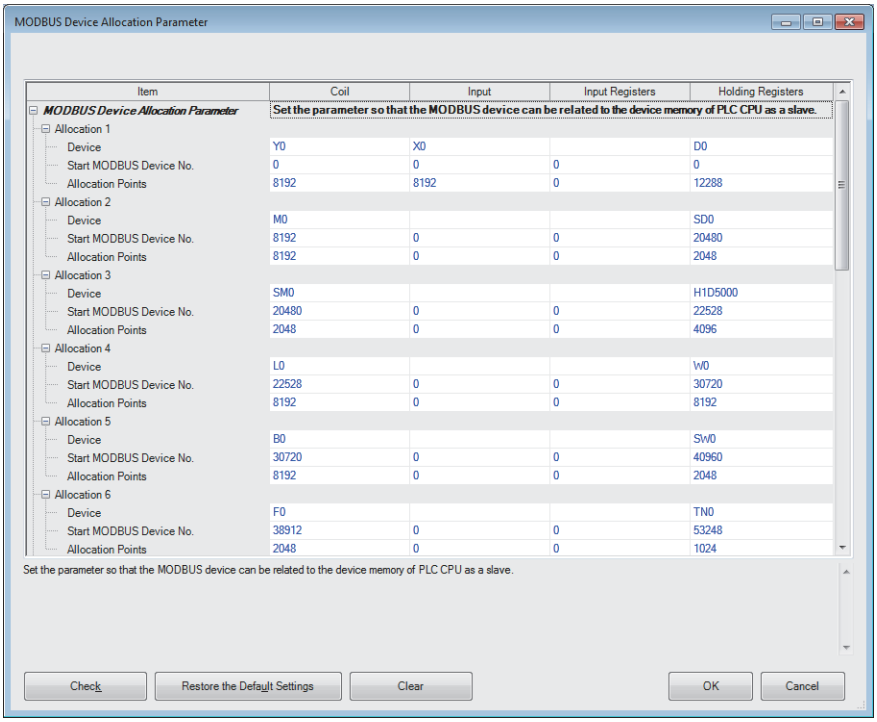
| Unit | Setting range |
|---|---|
| s | 1 to 16383 |
| ms | 100 to 16383000 (in increments of 100ms) |

## Device Assignment

Set the MODBUS device assignment parameters to use the MODBUS device assignment function.

Create the association between the following MODBUS devices and the device memory areas of the CPU module.

- Coil
- Input
- Input register
- Holding register




| Item | Description |
|---|---|
| Device | Set the devices of the CPU module to be assigned to the MODBUS devices.<br>Use H to specify a buffer memory area. (Example: H1D5000) |
| Start MODBUS Device No. | Set the start number of the devices of the CPU module to be assigned to the MODBUS devices.<br>The setting value for the start MODBUS device number can be obtained with the following formula.<br>• Start MODBUS device number = Last five digits of the target MODBUS device number - 1 |
| Allocation Points | Set the number of device points of the CPU module to be assigned to the MODBUS devices. |

**Point**

When "Device" is blank, values set in "Start MODBUS Device No." and "Allocation Points" are ignored.
In this case, data written into the programmable controller will not be assigned to the MODBUS devices.
(When the written data are read from the programmable controller, "Start MODBUS Device No." and "Allocation Points" are set to 0.)

# Simple CPU communication setting (RJ71EN71, RnENCPU (network part))

This setting is used for using the simple CPU communication function with the RJ71EN71 or RnENCPU (network part).
Set "Use" to "To Use or Not to Use Simple CPU Communication" before setting other items.

| Item | Description | Setting range |
|---|---|---|
| To Use or Not to Use Simple CPU Communication | Select whether to use the simple CPU communication or not. | • Not Use<br>• Use<br>(Default: Not Use) |
| CPU Response Monitoring Timer | Set the time for monitoring the response from the CPU module.<br>When the CPU module does not send a response within the set time, the state of waiting for the response from the Ethernet-equipped module is canceled. | • s: 1 to 16383<br>• ms: 100 to 16383000 (in increments of 100ms)<br>(Default: 5s) |
| Simple CPU Communication Setting | Set the items to use the simple CPU communication. (☞ Page 342 Simple CPU communication setting) | |
| "Communication Destination Setting" window | Click "Detailed Setting" to display the window. (☞ Page 344 "Communication Destination Setting" window) | |

## Details of items

### ■Simple CPU communication setting

Set the items to use the simple CPU communication.

| Item | | Description | Setting range |
|---|---|---|---|
| Latency Time | | Set the time required to start communications after completion of the CPU module startup.<br>This setting is common to all the setting numbers and cannot be changed for each setting number.<br>After system switching in a redundant system, communications are restarted regardless of the latency time. | 0 to 255s (in increments of 1s)<br>(Default: 0s) |
| Import/Export of Settings | [Import] button<br>[Export] button | Import/export the simple CPU communication setting parameters from/to a CSV file.<br>☞ Page 213 Import/Export of Settings | — |
| Setting No. | | Setting numbers 1 to 512 are available.<br>Any numbers can be used. (Numbers to use do not have be sequential.) | — |
| Communication Pattern | | Set whether to read or write data. | • Read<br>• Write<br>(Default: Blank) |
| Communication Setting: Execution Interval (ms) | Communication setting | Set the data communication timing. | • Fixed Interval<br>• On Request<br>(Default: Fixed Interval) |
| | Execution interval | Set the execution interval of communications when "Fixed Interval" is set for the communication setting. | 1 to 65535ms (in increments of 1ms)<br>(Default: 100ms) |
| Communication Destination (IP Address) | Source | Set the transmission source.<br>When "Write" is set to "Communication Pattern", the own station (IP address of the own station) is displayed. | ☞ Page 344 "Communication Destination Setting" window |
| | Destination | Set the transmission destination.<br>When "Read" is set to "Communication Pattern", the own station (IP address of the own station) is displayed. | |
| Target PLC No.*1 | | Specify the CPU number of the communication destination.<br>It will be the connected station when "Not Specified" is selected. | • Not Specified<br>• PLC No.1<br>• PLC No.2<br>• PLC No.3<br>• PLC No.4<br>• PLC No.5<br>• PLC No.6<br>• PLC No.7<br>• PLC No.8<br>(Default: Not Specified) |

| Item | | Description | Setting range |
|---|---|---|---|
| Bit Device | Points | Set the type, start number, and end number of the "Source" bit device and the type and start number of the "Destination" bit device. When these values are input, the number of points will be displayed automatically. | The setting range differs depending on the communication destination. (☞ Page 194 Devices that can be specified) |
| | Type | | |
| | Start | | |
| | End | | |
| Word Device | Points | Set the type, start number, and end number of the "Source" word device and the type and start number of the "Destination" word device. When these values are input, the number of points will be displayed automatically. | |
| | Type | | |
| | Start | | |
| | End | | |
| Communication Time-out Period (ms) | | Set the time period until retries are performed after an error response from the communication destination, the time period until the abnormal state is detected, and the time period to wait for a response from the communication destination. Configure the setting so that (communication time-out period) ≥ (execution interval). For details on the communication time-out period, refer to the following. ☞ Page 212 Communication Time-out Period/Communication Retry Count/ Monitoring Time At Error | 10 to 65535ms (in increments of 1ms) (Default: 1000ms) When the device type of the communication destination set to "SLMP Corresponding Device (QnA Compatible 3E Frame)", the value is fixed to 65535ms. |
| Communication Retry Count | | Set the number of communication retry to be performed within the communication time-out period when the communication destination returns an error response or does not respond. For details on the communication retry count, refer to the following. ☞ Page 212 Communication Time-out Period/Communication Retry Count/ Monitoring Time At Error | 0 to 255 times (Default: 3) When the device type of the communication destination set to "SLMP Corresponding Device (QnA Compatible 3E Frame)", the value is fixed to 0. |
| Monitoring Time At Error (s) | | Set the communication time after a communication error has occurred. For details on the monitoring time at error, refer to the following. ☞ Page 212 Communication Time-out Period/Communication Retry Count/ Monitoring Time At Error | 1 to 300s (Default: 30s) |
| Comment | | Sets a comment for the setting number, if required. | 32 characters maximum (Default: Blank) |

*1 The target PLC number differs depending on the communication destination. For details, refer to the following.
☞ Page 193 Target PLC No.

## ■"Communication Destination Setting" window

The communication destination of the simple CPU communication is specified on the window.

| Item | Description | Setting range |
|---|---|---|
| Device Type | Specify the device type of the communication destination. | • MELSEC iQ-R (Built-in Ethernet)<br>• MELSEC-Q (Built-in Ethernet)<br>• MELSEC-L (Built-in Ethernet)<br>• MELSEC iQ-F (Built-in Ethernet)<br>• MELSEC iQ-L (Built-in Ethernet)<br>• MELSEC iQ-R (Ethernet Module)<br>• MELSEC-Q (Ethernet Module)<br>• MELSEC-L (Ethernet Module)<br>• MELSEC-A/AnS (Ethernet Module)<br>• MELSEC-FX3 (Ethernet Block/Adapter)<br>• SLMP Corresponding Device (QnA Compatible 3E Frame)<br>• OMRON (CS/CJ series)<br>• OMRON (NJ/NX series)<br>• KEYENCE (KV series)<br>• Panasonic (FP2SH series)<br>• Panasonic (FP7 series)<br>• YASKAWA MP3000 series<br>• YASKAWA MP2000 series<br>• Yokogawa FA-M3 series<br>• MODBUS/TCP Corresponding Device<br>• SIEMENS S7 series<br>• SIEMENS S7 series (extension)<br>• Fuji Electric MICREX-SX series<br>• JTEKT TOYOPUC series<br>(Default: Blank) |
| UDP/TCP | Set the protocol used for the communication. | • UDP<br>• TCP<br>(Default: Blank) |
| IP Address | Enter the IP address of the communication destination. | 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| Port No. | Set the port number of the communication destination. | 1 to 65534<br>(Default: Blank) |
| Host Station Port No. | Set the port number of the own station. | 1 to 65534<br>(Default: Blank) |
| Option (hexadecimal) | Set the option setting value. | The value differs depending on the external devices.<br>(☞ Page 191 Option (hexadecimal)) |

# Simple CPU communication setting (CPU module (built-in Ethernet port part))

Set the following items when using the simple CPU communication function with the CPU module (built-in Ethernet port part).

| Item | | Description | Setting range |
|---|---|---|---|
| Latency Time | | Set the time required to start communications after completion of the CPU module startup.<br>This setting is common to all the setting numbers and cannot be changed for each setting number. | 0 to 255s (in increments of 1s)<br>(Default: 0s) |
| Initial Communication Setting | | Set the operation when the communication destination does not respond at the initial communication.<br>This setting is enabled when the communication setting is set to "Fixed Interval". In addition, this setting is common to all the setting numbers and cannot be changed for each setting number.<br>☞ Page 243 Initial Communication Setting | Communicate until the communication destination responds.<br>(Default: Without check for the Rn(EN)CPU, with check for the RnPCPU (fixed)) |
| Import/Export of Settings | [Import] button | Import/export the simple CPU communication setting parameters from/to a CSV file.<br>☞ Page 244 Import/Export of Settings | — |
| | [Export] button | | |
| Setting No. | | Displays setting No.1 to 64.<br>An arbitrary setting number can be set. (Numbers to use do not have be sequential.) | — |
| Communication Pattern | | Set whether to read or write data. | • Read<br>• Write<br>(Default: Blank) |
| Communication Setting: Execution Interval (ms) | Communication setting | Sets this item to communicate data periodically. | • Fixed Interval<br>• On Request*2<br>(Default: Fixed Interval) |
| | Execution interval | Set the execution interval of communications when "Fixed Interval" is set for the communication setting. | 10 to 65535ms (in increments of 1ms)<br>(Default: 100ms) |
| Communication Destination (IP Address) | Source | Set the transmission source.<br>When "Write" is set to "Communication Pattern", the own station (IP address of the own station) is displayed. | ☞ Page 346 Communication destination setting |
| | Destination | Set the transmission destination.<br>When "Read" is set to "Communication Pattern", the own station (IP address of the own station) is displayed. | |
| Target PLC No.*1 | | Specify the CPU number of the communication destination.<br>It will be the connected station when "Not Specified" is selected. | • Not Specified<br>• PLC No.1<br>• PLC No.2<br>• PLC No.3<br>• PLC No.4<br>(Default: Not Specified) |
| Bit Device | Points | Set the type, start number, and end number of the "Source" bit device and the type and start number of the "Destination" bit device. When these values are input, the number of points will be displayed automatically. | The setting range differs depending on the communication destination. (☞ Page 233 Devices that can be specified) |
| | Type | | |
| | Start | | |
| | End | | |
| Word Device | Points | Set the type, start number, and end number of the "Source" word device and the type and start number of the "Destination" word device. When these values are input, the number of points will be displayed automatically. | |
| | Type | | |
| | Start | | |
| | End | | |
| Communication Time-out Period (ms) | | Set the time period until retries are performed after an error response from the communication destination, the time period until the abnormal state is detected, and the time period to wait for a response from the communication destination.<br>Configure the setting so that (communication time-out period) ≥ (execution interval).<br>For details on the communication time-out period, refer to the following.<br>☞ Page 242 Communication Time-out Period/Communication Retry Count/ Monitoring Time At Error | 10 to 65535ms (in increments of 1ms)<br>(Default: 1000ms)<br>When the communication destination set to "SLMP Corresponding Device (QnA Compatible 3E Frame)", the value is fixed at 65535ms. |

| Item | Description | Setting range |
|---|---|---|
| Communication Retry Count | Set the number of communication retry to be performed within the communication time-out period when the communication destination returns an error response or does not respond.<br>For details on the communication retry count, refer to the following.<br>☞ Page 242 Communication Time-out Period/Communication Retry Count/ Monitoring Time At Error | 0 to 255 times<br>(Default: 3)<br>When the communication destination set to "SLMP Corresponding Device (QnA Compatible 3E Frame)", the value is fixed at 0. |
| Monitoring Time At Error (s) | Set the communication time after a communication error has occurred.<br>This setting is enabled when the communication setting is set to "Fixed Interval".<br>For details on the monitoring time at error, refer to the following.<br>☞ Page 242 Communication Time-out Period/Communication Retry Count/ Monitoring Time At Error | 1 to 300s<br>(Default: 30s) |
| Comment | Sets a comment for the setting number, if required. | 32 characters maximum<br>(Default: Blank) |

*1 Although the target PLC No.5 to 8 can be selected for the communication destination in the engineering tool, they cannot be used for the CPU module (built-in Ethernet port part).

*2 To set this value, check the versions of the CPU module and engineering tool. (☞ Page 586 Added and Enhanced Functions)

## Communication destination setting

The following table lists the communication destination setting specified for the simple CPU communication.

| Item | Description | Setting range |
|---|---|---|
| Device Type | Specify the device type of the communication destination. | • MELSEC iQ-R (Built-in Ethernet)<br>• MELSEC iQ-L (Built-in Ethernet)<br>• MELSEC iQ-F (Built-in Ethernet)<br>• MELSEC-Q (Built-in Ethernet)<br>• MELSEC-L (Built-in Ethernet)<br>• MELSEC-Q (Ethernet Module)*1<br>• MELSEC-L (Ethernet Module)*1<br>• MELSEC-A/AnS (Ethernet Module)<br>• MELSEC-FX3 (Ethernet Block/Adapter)<br>• SLMP Corresponding Device (QnA Compatible 3E Frame)<br>(Default: Blank) |
| IP Address | Enter the IP address of the communication destination. | 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| Port No. | Set the port number of the communication destination. | • MELSEC-A/AnS (Ethernet Module): 256 to 65534<br>• MELSEC-FX3 (Ethernet Block/Adapter): 1025 to 5548, 5552 to 65534<br>• SLMP Corresponding Device (QnA Compatible 3E Frame): 1 to 4999, 5010 to 65534<br>• Other than the above: Setting not required<br>(Default: Blank) |
| Host Station Port No. | Set the port number of the own station. | • MELSEC-A/AnS (Ethernet Module): 1 to 4999, 5010 to 65534<br>• MELSEC-FX3 (Ethernet Block/Adapter): 1 to 4999, 5010 to 65534<br>• SLMP Corresponding Device (QnA Compatible 3E Frame): 1 to 4999, 5010 to 65534<br>• Other than the above: Setting not required<br>(Default: Blank) |

*1 This cannot be selected for the RnPCPU.

# Simple Device Communication Setting

This setting is used for using the simple device communication function with the RJ71EN71 or RnENCPU (network part).
Set "To Use or Not to Use Simple Device Communication" to "Use" before setting other items.

| Item | Description | Setting range |
|---|---|---|
| To Use or Not to Use Simple Device Communication | Select whether to use the simple device communication or not. | • Not Use<br>• Use<br>(Default: Not Use) |
| Resource Setting | Set the maximum numbers of the communication destinations, protocols, and packets that can be set. (☞ Page 348 Resource Setting) | • Normal<br>• Extension 1*2<br>• Extension 2<br>(Default: Normal) |
| Instruction Initial Value to Start/Stop Function | Select whether to start the simple device communication with the function enabled or stopped. | • Start<br>• Stop<br>(Default: Start) |
| CPU Response Monitoring Timer | Set the time for monitoring the response from the CPU module.<br>When the CPU module does not send a response within the set time, the state of waiting for the response from the Ethernet-equipped module is canceled. | • s: 1 to 16383<br>• ms: 100 to 16383000 (in increments of 100ms)*1<br>(Default: 5) |
| Unit | Select the unit of the CPU response monitoring timer.<br>When "ms" is selected, set the value in increments of 100ms. | • s<br>• ms<br>(Default: s) |
| "Communication Destination Setting" window | Click "Detailed Setting" to display the window. (☞ Page 349 "Communication Destination Setting" window) | |
| "Device List" window | Click "Confirmation" to display the window. (☞ Page 399 "Device List" window) | |

*1 The setting range differs for each timer setting unit. In addition, the error of the timer is ±10ms.

*2 When RJ71EN71(E+E) is used for the module and does not support "Extension 2", set "To Use or Not to Use Simple Device Communication" of the other ports to "Not to Use". (☞ Page 292 Restrictions applicable depending on versions)

## Resource Setting

The module name (network type) determines the availability of the options.

When the value of "Resource Setting" is changed from "Extension 1" to "Normal", any communication destination information that falls outside the range for "Normal" will be deleted.

When the value of "Resource Setting" is changed from "Extension 2" to either "Extension 1" or "Normal", all communication destination information will be deleted.

| Resource Setting | Availability/Maximum number of settings | Model name (network type) | | |
|---|---|---|---|---|
| | | RJ71EN71(E+E) | RJ71EN71(E+CCIEC) _RJ71EN71(E+IEC) | RJ71EN71(E+CCIEF) _RJ71EN71(E+IEF) |
| Normal | Availability | Available | Available | Available |
| | No. of communication destinations | 16 destinations/port | | |
| | No. of protocols | 256/port | | |
| | No. of packets | 768/port | | |
| | Packet data area | 61440 bytes/port | | |
| | No. of elements | 1 to 32/packet | | |
| | Packet length (except for the header) | 4096 bytes/packet | | |
| Extension 1 | Availability | Available*1 | Available | Available |
| | No. of communication destinations | 32 destinations/port | | |
| | No. of protocols | 512/port | | |
| | No. of packets | 1536/port | | |
| | Packet data area | 122880 bytes/port | | |
| | No. of elements | 1 to 32/packet | | |
| | Packet length (except for the header) | 4096 bytes/packet | | |
| Extension 2 | Availability | Available | Available | Available |
| | No. of communication destinations | 64 destinations/port | | |
| | No. of protocols | 2048/port | | |
| | No. of packets | 6144/port | | |
| | Packet data area | 491520 bytes/port | | |
| | No. of elements | 1 to 128/packet | | |
| | Packet length (except for the header) | 10238 bytes/packet | | |

*1 Without "Extension 2" supported, this option cannot be set if "To Use or Not to Use Simple Device Communication" of other ports is set to "Use". (☞ Page 292 Restrictions applicable depending on versions)

*2 Up to 512 protocols can be set for each "Communication Destination Setting".

## Instruction Initial Value to Start/Stop Function

Set the initial value to be stored in 'Function start/stop instruction' (Un\G1249300.0) when the initialization of the simple device communication is completed.

- Start: On (1)
- Stop: Off (0)

**Point**

To start up with an instruction from the communication destination, set to "Stop". To perform rudderless communications, set to "Start".

For details on the operations, refer to the following pages.

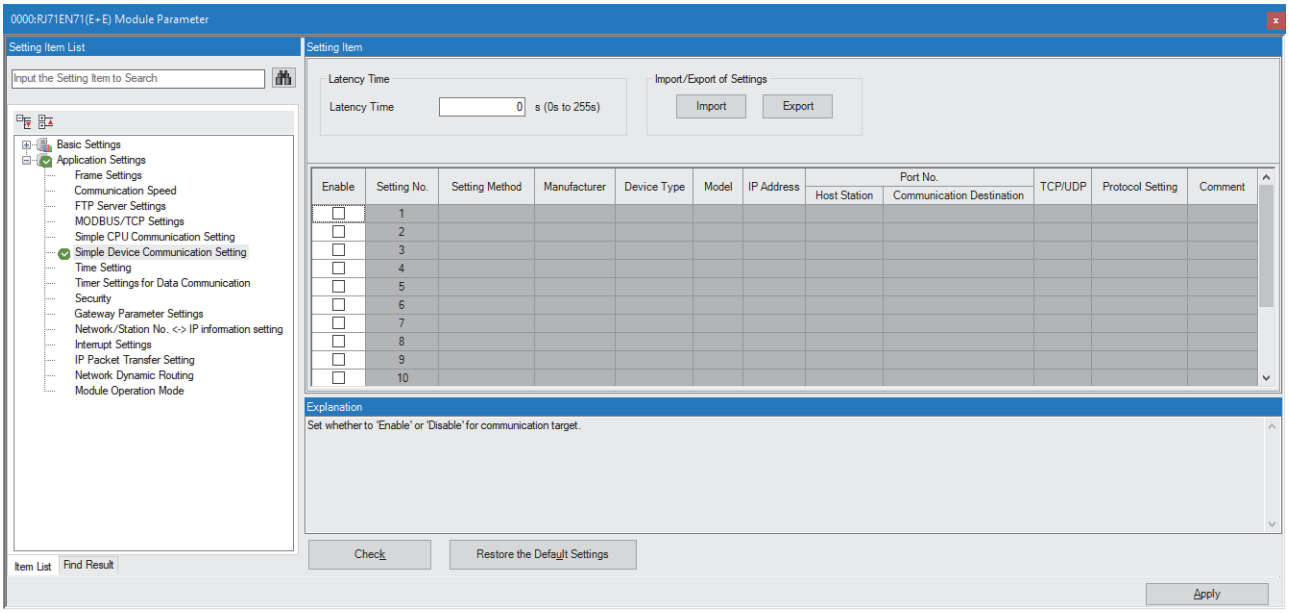
☞ Page 268 Normal operation (when "Instruction Initial Value to Start/Stop Function" is set to "Start")

☞ Page 269 Normal operation (when "Instruction Initial Value to Start/Stop Function" is set to "Stop")

## "Communication Destination Setting" window

The communication destination of the simple device communication is set on the window.

Select the checkbox (enable the communication destination setting) before setting other items.




| Item | | Description | Setting range |
|---|---|---|---|
| Checkbox (Enable the communication destination setting) | | Set "Enable" or "Disable" for the communication destination. | • Disable (Not select)<br>• Enable (Select)<br>(Default: Disable (Not select)) |
| Setting No.*1 | | Displays the number of communication destinations that can be set.<br>The communication destination can be set to a desired setting number. | • 1 to 16<br>• 1 to 32<br>• 1 to 64<br>(Default: 1 to 16, 1 to 32, or 1 to 64) |
| Setting Method | | Select the setting method for the communication destination. | • Select from library<br>• User Selection<br>(Default: Blank) |
| Manufacturer*2 | | Set the manufacturer name of the communication destination. | • When selected from library: Simple device communication library file<br>• When set by user: 0 to 32 characters<br>(Default: Blank) |
| Device Type*2 | | Set the device type of the communication destination. | • When selected from library: Simple device communication library file<br>• When set by user: 0 to 32 characters<br>(Default: Blank) |
| Model*2 | | Set the model of the communication destination. | • When selected from library: Simple device communication library file<br>• When set by user: 0 to 32 characters<br>(Default: Blank) |
| IP Address | | Enter the IP address of the communication destination. | 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| Port No. | Host Station*3 | Set the port number of the own station. | 1 to 4999, 5010 to 65534<br>(Default: Blank) |
| | Communication Destination | Set the port number of the communication destination. | 1 to 65534<br>(Default: Blank) |
| TCP/UDP | | Select the protocol used for the communication. | • TCP Active<br>• UDP<br>(Default: Blank) |
| "Protocol Setting" window | | Set the protocol(s) to be used for the communication. (☞ Page 351 "Protocol Setting" window) | |
| Comment*2 | | Set (and displays) any comment, if required. | Up to 64 characters<br>(Default: Blank) |

| Item | | Description | Setting range |
|---|---|---|---|
| Latency Time*4 | | Set the time required to start communications after completion of the CPU module startup.<br>This setting is common to "Setting No." and cannot be changed for each "Setting No.". | 0 to 255 (in increments of 1s)<br>(Default: 0) |
| Import/Export of Settings | [Import] button | Reads the settings from the simple device communication setting file and displays them in the window. | — |
| | [Export] button | Saves the current settings as a simple device communication setting file. | — |

*1 The default is 1 to 16 when "Resource Setting" is set for "Normal", 1 to 32 for "Extension 1", and 1 to 64 for "Extension 2".

*2 Because this item is saved as character string, the display language change of the engineering tool is not applicable. The content is always displayed in the language of the parameter settings.

*3 Use 1024 to 4999 or 5010 to 61439 for the port number of the own station. 1 to 1023 are typically reserved port numbers (WELL KNOWN PORT NUMBERS) and 61440 to 65534 are used by other communication functions.

*4 When the latency time is set, the system delays the communication start timing for the specified time before performing the simple device communication. To start communications after the communication destination is ready for the communications, intentionally change the start time to avoid errors in the simple device communications.

## ■Import/export of simple device communication setting file

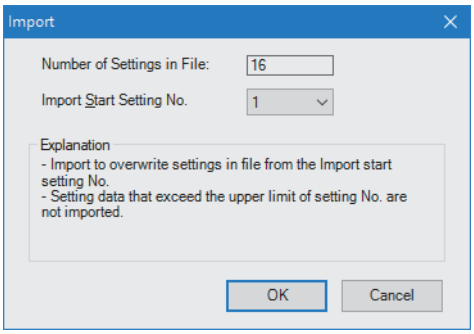
This function saves the protocol information, including the set communication destination information and device assignment information, to a file and restores the parameters from that file. Saving is performed for each communication destination, and one file can contain multiple instances of communication destination information.

**Point**

By using import/export, the communication destination settings, including the protocol settings, can be easily duplicated and reused for other systems and devices.

• Import procedure

1. Click the [Import] button.
2. Select the simple device communication setting file (*.gx3sc) to import, and click the [Open] button.
3. Set the first setting number of the setting reflection destination, and then click the [OK] button.

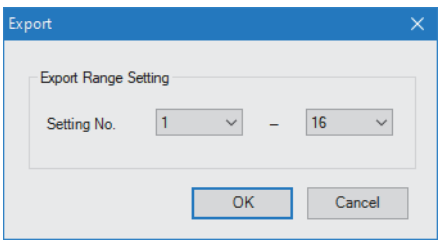



**Point**

Settings that exceed the upper limit of the setting number are not imported.

• Export procedure

1. Click the [Export] button.
2. Set the range of setting numbers to be saved, and then click the [OK] button.

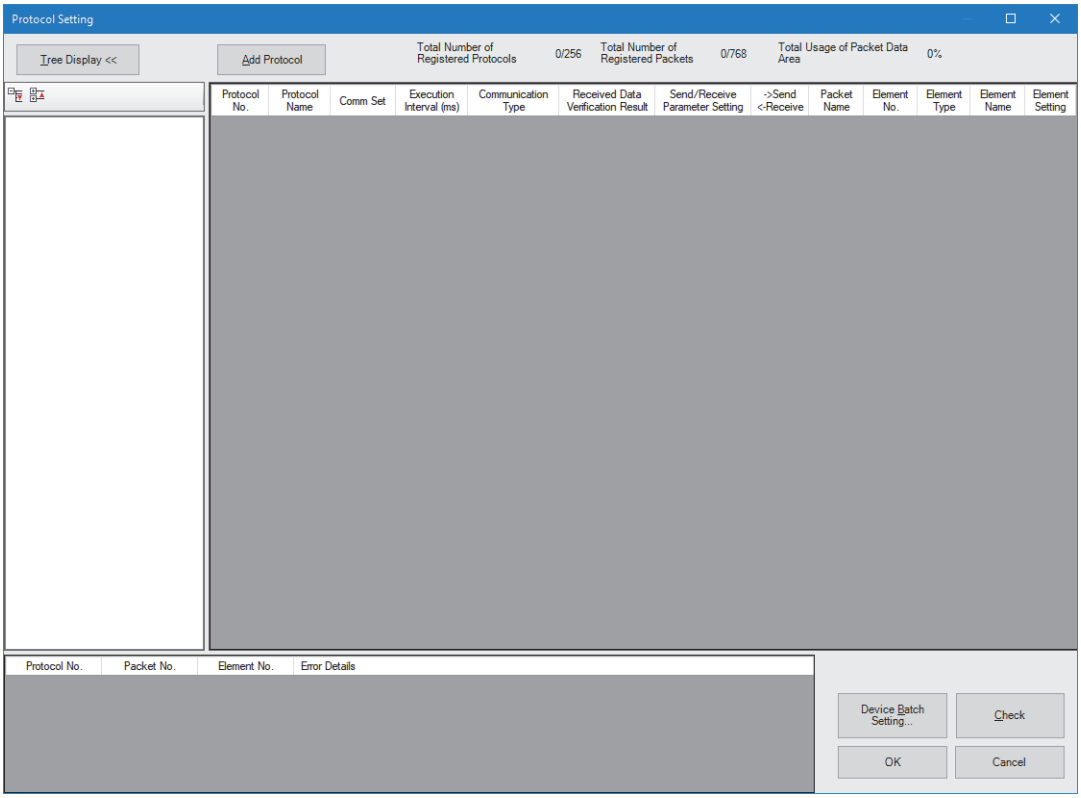



3. Select the save destination, and click the [Save] button.

## "Protocol Setting" window

Set the protocol for communicating with the communication destination.

Click the [Add Protocol] button before setting other items.




| Item | Description | Setting range |
|---|---|---|
| [Tree Display] button (protocol list display switching) | Shows/hides the protocol list. | — |
| Protocol list | Displays the registered protocols in tree format. | — |
| [Add Protocol] button | Adds a new protocol.<br>Defaults are entered in "Protocol Name" and "Packet Name".<br>When "Setting Method" on the "Communication Destination Setting" window is set to "Select from library", the protocol to be set (the protocol registered in the simple device communication library file) can be selected from the "Protocol Name" drop-down menu. | — |
| Total Number of Registered Protocols*1 | Displays the total number of protocols registered for all the communication destinations including the destinations for which the checkbox (enable the communication destination setting) is not selected on the "Communication Destination Setting" window. | • 0/256 to 256/256<br>• 0/512 to 512/512<br>• 0/2048 to 2048/2048 |
| Total Number of Registered Packets*2 | Displays the total number of packets registered for all the communication destinations including the destinations for the checkbox (enable the communication destination setting) is not selected on the "Communication Destination Setting" window. | • 0/768 to 768/768<br>• 0/1536 to 1536/1536<br>• 0/6144 to 6144/6144 |
| Total Usage of Packet Data Area | Displays the ratio of the currently registered packet data size to the maximum size that can be registered in the packet data area (flash ROM area in the module that stores packet data for communications with the communication destination). The currently registered packet data size includes the communication destinations for which the checkbox (enable the communication destination setting) is not selected on the "Communication Destination Setting" window. | 0% to 100.0% |
| Protocol No.*3 | Set (and displays) the protocol number. | • 1 to 256<br>• 1 to 512<br>(Default: Blank) |
| Protocol Name*5*6 | From the simple device communication library, select the protocol to be used.<br>The name of the selected protocol can also be edited. | • Select from the simple device communication library file.<br>• 1 to 48 characters<br>(Default: Blank) |

| Item | Description | Setting range |
|---|---|---|
| Comm Set | Select the communication method for the protocol.<br>Request: Data are communicated upon request only.<br>Fixed Intrvl: Data are communicated at a specified execution interval. | • Request<br>• Fixed Intrvl<br>(Default: Blank) |
| Execution Interval (ms) | Set the cycle (interval) for sending data to the communication destination.<br>This item cannot be set for protocols whose "Comm Set" is "Request".<br>This item cannot be set for protocols whose "Communication Type" is "Receive Only". | 1 to 65535ms (to be specified in increments of 1ms)<br>(Default: Blank) |
| Communication Type | Displays the protocol communication type.<br>• Send Only: The send packet is sent once.<br>• Receive Only: If there is a matching packet up to 16 registered and received packets, it is received.<br>• Send & Receive: After the send packet is sent, a packet is received if the packet matches any of receive packets registered up to 16. | • Send Only<br>• Receive Only<br>• Send & Receive<br>(Default: Blank) |
| Received Data Verification Result | Set (and displays) the device that stores the received packet number.<br>This item cannot be set if "Communication Type" is "Send Only". | Refer to devices that can be specified on the following page.<br>☞ Page 262 Devices that can be specified<br>(Default: Blank) |
| Send/Receive Parameter Setting | Configure the settings for communications with the protocol. (☞ Page 397 "Send/Receive Parameter Setting" window) | |
| → Send<br>← Receive | Displays the packet send direction.<br>For receive packets, this item also displays the packet number.<br>→: Indicates send cases.<br>←(n): Indicates receive cases. (n: Receive packet number (1 to 16)) | For send cases: →<br>For receive cases: ←(n)<br>(Default: Blank) |
| Packet Name*5 | Displays the packet name.<br>The name of the selected packet can also be edited. | 1 to 48 characters<br>(Default: Blank) |
| Element No. | Displays the packet element number. | • 1 to 32<br>• 1 to 128<br>(Default: Blank) |
| Element Type | Displays the packet element type. | • Static Data<br>☞ Page 360 Element type (Static data)<br>• Length<br>☞ Page 361 Element type (Length)<br>• Non-conversion Variable (Fixed-Length)<br>☞ Page 363 Element type (Non-conversion variable)<br>• Non-conversion Variable (Variable Length)<br>☞ Page 363 Element type (Non-conversion variable)<br>• Conversion Variable (Fixed-Length)<br>☞ Page 370 Element type (Conversion variable)<br>• Conversion Variable (Variable-Length)<br>☞ Page 370 Element type (Conversion variable)<br>• Non-verified Reception (Fixed-Length)<br>☞ Page 387 Element type (Non-verified reception)<br>• Non-verified Reception (Variable-Length)<br>☞ Page 387 Element type (Non-verified reception)<br>• Error check code<br>☞ Page 388 Element type (Error check code)<br>(Default: Blank) |
| Element Name*5 | Displays the packet element name. | 1 to 48 characters<br>(Default: Blank) |
| Element Setting | Set the packet elements. | Refer to the element types below.<br>☞ Page 354 Element setting<br>(Default: Blank) |
| [Device Batch Setting] button | Set devices used in the protocol by batch. (☞ Page 398 "Device Batch Setting" window) | |
| [Check] button | Checks the setting details of the protocol setting. | |
| Protocol No. | Displays the number of the protocol where an error occurred. | • 1 to 512<br>• —: Protocol-independent error |
| Packet No. | Displays the packet where an error occurred. (n: Receive packet number (1 to 16)) | • →<br>• ←(n)<br>• —: Packet-independent error |
| Element No. | Displays the number of the element where an error occurred. | • 1 to 32<br>• —: Element-independent error |
| Error Details | Displays the error details. | Character string of the error details |

*1 The number will be 0/256 to 256/256 when "Resource Setting" is set for "Normal", 0/512 to 512/512 for "Extension 1", and 0/2048 to 2048/2048 for "Extension 2".
*2 The number will be 0/768 to 768/768 when "Resource Setting" is set for "Normal", 0/1536 to 1536/1536 for "Extension 1", and 0/6144 to 6144/6144 for "Extension 2".
*3 The number will be 1 to 256 when "Resource Setting" is set for "Normal", and 1 to 512 for either "Extension 1" or "Extension 2".
*4 The number will be 1 to 32 when "Resource Setting" is set for either "Normal" or "Extension 1", and 1 to 128 for "Extension 2".
*5 Because this item is saved as character string, the display language change of the engineering tool is not applicable. The content is always displayed in the language of the parameter settings.
*6 If the protocol name entered by text input matches a protocol name obtained from the simple device communication library, the behavior is the same as when the protocol name is selected from the drop-down menu.



**Point**

- To delete a protocol, select any item from "Protocol No." to "Send/Receive Parameter Setting", and press the Delete key.
- To delete multiple protocols, right-click anywhere, select "Delete Multiple Protocols", set the "Protocol No." range to be deleted, and then click the [OK] button. Click the [Yes] button on the confirmation window to delete the specified protocols. (The protocols that are not deleted move forward.)
- If the "Protocol Setting" window is opened with no protocol set, a specific protocol is automatically added. For details, refer to the list of applicable protocols on the following page.

MELSEC iQ-R Simple Device Communication Library Reference Manual

## ■Protocol editing

Protocol editing is done with the right-click menu.

| Item | Description | Remarks |
|---|---|---|
| Add Protocol | Same as the [Add Protocol] button | — |
| Open Send/Receive Parameter Setting | Same as "Send/Receive Parameter Setting" | — |
| Add Receive Packet | Adds a receive packet to the end of the protocol on which the cursor is positioned. | Receive packets can be interchanged from the drop-down menu of the "→ Send ← Receive" column. |
| Add Element | Adds an element to the end of the packet on which the cursor is positioned. | After an element is added, the element can be interchanged by editing the number in the "Element No." column. |
| Delete | Deletes the protocol/packet/element on the cursor line. | Send packets cannot be deleted. |
| Copy | Copies the protocol/packet/element on the cursor line. | — |
| Paste Overwrite | Pastes the copied protocol/packet/element over the protocol/packet/element at the cursor position. | — |
| Paste New | Pastes the copied protocol/packet/element as a new protocol/packet/element. | — |
| Delete Multiple Protocols | Opens the "Delete Multiple Protocols" window. | — |
| Device Batch Setting | Same as the [Device Batch Setting] button | — |

## Element setting

The packet is created with a combination of packet elements.

The number of elements that can be set in one packet and the maximum data length per packet are as follows:

- "Resource Setting" set to either "Normal" or "Extension 1": Up to 32 elements can be set in one packet; The maximum data length per packet is 4096 bytes.
- "Resource Setting" set to "Extension 2": Up to 128 elements can be set in one packet; The maximum data length per packet is 10238 bytes.

The data length of one packet is calculated by adding up the data length of each element set in the packet. The following table shows the setting items used to calculate the data length of each element.

| Element type | Setting item used to calculate data length |
|---|---|
| Static Data | Byte size of data set in "Setting Value" |
| Length | "Data Length" |
| Non-conversion Variable | "Data Length/Maximum Data Length" |
| Conversion Variable | "Number of Data/Maximum Number of Data", "Digit", "Signed/Unsigned", "Number of Decimals", "Delimiter"<br>☞ Page 354 How to calculate the data length of a conversion variable |
| Non-verified Reception | "Data Length"*1 |
| Error check code | "Data Length"*2 |

*1 If the value is 0 (variable number of characters), it is treated as data length 0.

*2 If the processing method is 16-bit CRC (MODBUS specification), it is treated as data length 2.

**Point**

If the engineering tool version is 1.075D or earlier, the only element type for which settings can be made is static data.

### ■How to calculate the data length of a conversion variable

| Number of Data/Maximum Number of Data | Delimiter | Calculation method |
|---|---|---|
| 1 | — | Data length of a conversion variable = (data length per data piece × "Number of Data/Maximum Number of Data") |
| 2 or more | No Delimiter | |
| | Other than "No Delimiter" | Data length of a conversion variable = (data length per data piece × "Number of Data/Maximum Number of Data") - 1 |

The data length per data piece is calculated with the following formula: "Digit"*1 + "Signed/Unsigned"*2 + "Number of Decimals"*3 + "Delimiter"*4

*1 The calculation value varies depending on the following conditions.

| Digit | Conversion | Conversion data type | Calculation value |
|---|---|---|---|
| Variable Number of Digits | • HEX→ASCII Hexadecimal<br>• ASCII Hexadecimal→HEX | Word | 4 |
| | | Double Word | 8 |
| | • HEX→ASCII Decimal<br>• ASCII Decimal→HEX | Word | 5 |
| | | Double Word | 10 |
| | | FLOAT [Single Precision] | 7 |
| | | FLOAT [Double Precision] | 15 |
| Other than "Variable Number of Digits" | — | — | 1 to 16 |

*2 "Unsigned": 0, Other than "Unsigned": 1

*3 "No Decimal Point": 0, Other than "No Decimal Point": 1

*4 "No Delimiter": 0, Other than "No Delimiter": 1

Ex.

When the following settings are configured: "Conversion": "HEX→ASCII Decimal" or "ASCII Decimal→HEX", "Conversion Data Type": "Word", "Number of Data/Maximum Number of Data": 256, "Digit": 5, "Signed/Unsigned": Other than "Unsigned", "Number of Decimals": "Variable Point", "Delimiter": "Comma"

Data length of the conversion variable = ((5 + 1 + 1 + 1) × 256) - 1 = 2047 bytes

Ex.

When the following settings are configured: "Conversion": "HEX→ASCII Decimal" or "ASCII Decimal→HEX", "Conversion Data Type": "FLOAT [Double Precision]", "Number of Data/Maximum Number of Data": 227, "Digit": "Variable Number of Digits", "Signed/Unsigned": Other than "Unsigned", "Number of Decimals": "Variable Point", "Delimiter": "Comma"

Data length of the conversion variable = ((15 + 1 + 1 + 1) × 227) - 1 = 4085 bytes

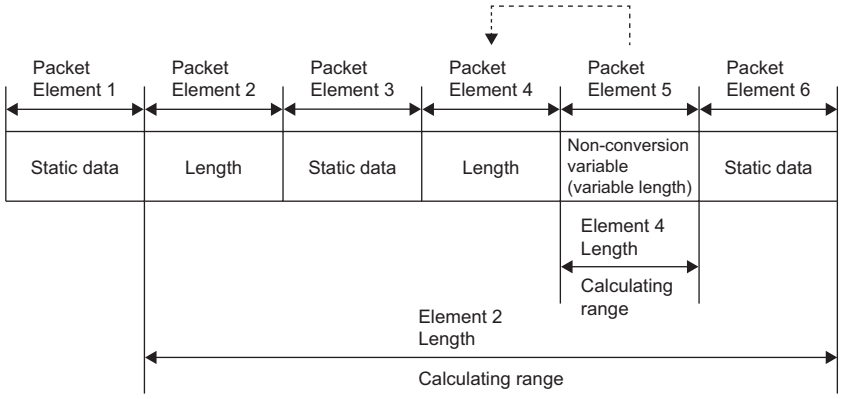
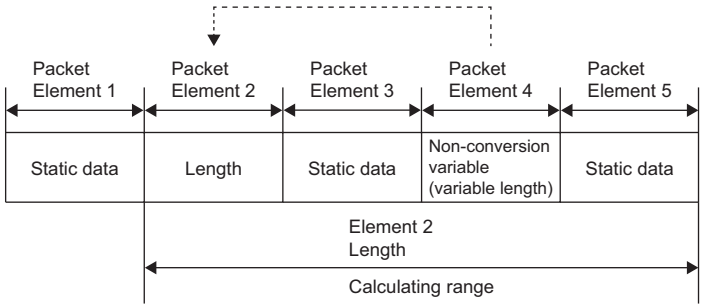
## ■Display format of each element type

| Element type | Setting item | Display format | Display example |
|---|---|---|---|
| Static Data | • Code Type<br>• Setting Value | (Size) "Setting Value"<br>(When "Code Type" is "ASCII String") | (4B)"1234" |
| | | (Size) Setting Value<br>(When "Code Type" is "HEX") | (2B)1234 |
| Length | • Code Type<br>• Data Length<br>• Data Flow<br>• Data Unit<br>• Calculation Range | (Calculation Range/Code Type/Data Flow/Data Length/Data Unit) | (7-12/HEX/Forward/2B/×1) |
| Non-conversion Variable (Fixed-Length) | • Data Length/Maximum Data Length<br>• Unit of Stored Data<br>• Byte Swap<br>• Send/Receive Data Storage Area | [Data storage area](Data Length/Unit of Stored Data/ Byte Swap) | [W0-W9](20B/Lower Byte + Upper Byte/No Swap) |
| Non-conversion Variable (Variable-Length) | • Data Length/Maximum Data Length<br>• Unit of Stored Data<br>• Byte Swap<br>• Send/Receive Data Length Storage Area | [Data length storage area][Data storage area](Data Length/Unit of Stored Data/ Byte Swap) | [W0][W1-W0A](20B/Lower Byte + Upper Byte/No Swap) |
| Conversion Variable (Fixed-Length) | • Conversion<br>• Fixed Number of Data/Variable Number of Data<br>• Number of Data/Maximum Number of Data<br>• Conversion Data Type<br>• Digit<br>• Blank-padded Character<br>• Signed/Unsigned<br>• Number of Decimals<br>• Delimiter<br>• Send/Receive Data Storage Area | [Data storage area] (Conversion/Number of Data/Conversion Data Type/Digit/Blank-padded Character/Singed/Unsigned/Number of Decimals/Delimiter) | [W0-W1] (→ 10/2/W/Digit (1)/Padding (0)/Unsigned/Decimal Point (none)/Delimiter (none)) |
| Conversion Variable (Variable-Length) | ■When "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data"<br>• Conversion<br>• Fixed Number of Data/Variable Number of Data<br>• Number of Data/Maximum Number of Data<br>• Conversion Data Type<br>• Digit<br>• Blank-padded Character<br>• Signed/Unsigned<br>• Number of Decimals<br>• Delimiter<br>• Send/Receive Data Storage Area<br>■When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data"<br>• Conversion<br>• Fixed Number of Data/Variable Number of Data<br>• Number of Data/Maximum Number of Data<br>• Conversion Data Type<br>• Digit<br>• Blank-padded Character<br>• Signed/Unsigned<br>• Number of Decimals<br>• Delimiter<br>• Send/Receive Data Count Storage Area | ■When "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data"<br>[Data storage area] (Conversion/Number of Data/Conversion Data Type/Digit/Blank-padded Character/Singed/Unsigned/Number of Decimals/Delimiter)<br>■When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data"<br>[Data count storage area] [Data storage area] (Conversion/Number of Data/Conversion Data Type/Digit/Blank-padded Character/Signed/Unsigned/Number of Decimals/Delimiter) | ■When "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data"<br>[W0-W1] (→ 10/2/W/Digit (variable)/Padding (0)/Unsigned/Decimal point (none)/Delimiter (none))<br>■When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data"<br>[W0] [W1-W2] (→ 10/2/W/Digit (1)/Padding (0)/Unsigned/Decimal point (none)/Delimiter (none)) |
| Non-verified Reception (Fixed-Length) | Data Length (1 to 4096) | (Data Length) | (20B) |
| Non-verified Reception (Variable-Length) | Data Length (0) | (Variable) | (Variable) |

| Element type | Setting item | Display format | Display example |
|---|---|---|---|
| Error check code | • Processing Method<br>Code Type<br>Data Length<br>Data Flow<br>Complement Calculation<br>Calculation Range | (Calculation Range/Processing Method/Code Type/Complement Calculation/Data Flow/Data Length) | (7-12/Prty/HEX/2C/Forward/2B) |



## ■Terms related to element settings

| Term | Description |
|---|---|
| Fixed length element | An element that determines the packet data length when setting parameters.<br>• Length<br>• Non-conversion Variable (Fixed-Length)<br>• Conversion Variable*1<br>• Non-verified Reception (Fixed-Length)<br>• Error check code |
| Variable length element | An element that does not determine the packet data length when setting parameters.<br>• Non-conversion Variable (Variable-Length)<br>• Conversion Variable*2<br>• Non-verified Reception (Variable-Length) |
| Variable length element that determines the data length by itself | A variable length element that determines the data length by itself regardless of the placement of other elements during verification when a packet is received.<br>• Conversion Variable*3 |
| Innermost length | • If an element is included in multiple length calculation ranges, the length with the innermost calculation range is the innermost length of that configuration. (If multiple lengths have the same calculation range, then the length placed earlier is the innermost length.)<br>• If an element is included in only a single length calculation range, that length is the innermost length.<br>• If an element is not included in any length calculation ranges, that element does not have an innermost length.<br>■Example 1: Innermost length of element 5 (Non-conversion Variable (Variable-Length)) → Element 4<br><br><br>■Example 2: Innermost length of element 4 (Non-conversion Variable (Variable-Length)) → Element 2<br><br> |

*1 When the following settings are configured: "Fixed Number of Data/Variable Number of Data": "Fixed Number of Data", "Digit": Other than "Variable Number of Digits", "Signed/Unsigned": Other than "Signed/Positive Sign Character (None)", "Number of Decimals": Other than "Variable Point"

*2 When the conditions of a fixed length element are not met

*3 When either one of the below is applicable
(1) "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data", "Digit" is not "Variable Number of Digits", and conditions of a fixed length element are not met.
(2) "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data", "Number of Data/Maximum Number of Data" is "1", "Digit" is "Variable Number of Digits", and also "Delimiter" is not "No Delimiter".

## ■Restrictions on placement of variable length elements in a receive packet

| No. | Restriction |
|---|---|
| 1 | One or more following conditions must be satisfied when a variable length element is used outside the length calculation range.<br>❶ There is static data outside the length calculation range after a variable length element. (A fixed length element outside the calculation range can be placed between the variable length element and static data)<br>❷ The final element of a packet is a variable length element or there is only a fixed length element outside the length calculation range after a variable length element.<br>❸ The variable length element determines the data length by itself. |
| 2 | When the variable length element that does not determine the data length by itself is used within the length calculation range, place the innermost length before the variable length element. |
| 3 | One or more following conditions must be satisfied when a variable length element is used within the length calculation range.<br>❶ Static data is placed after the variable length element, and the innermost length of both the variable length element and static data is the same. (If the innermost length is the same, a fixed length element can be placed between the variable length element and static data.)<br>❷ The variable length element is the final element within the innermost length calculation range or there is only a fixed length element with the same innermost length that is placed after the variable length element.<br>❸ The variable length element determines the data length by itself. |

Ex.

❶ in No.1: When there is static data outside the length calculation range after a variable length element

• Can be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | Element 6 | Element 7 | ... | Element 32 |
|---|---|---|---|---|---|---|---|---|
| Static data | Variable length element | Static data | Length | Variable length element | Static data | Static data | No setting | No setting |
| — | | | | Calculation range of element 4 | | | — | |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | Element 6 | Element 7 | ... | Element 32 |
|---|---|---|---|---|---|---|---|---|
| Static data | Variable length element | Static data | Length | Variable length element | Static data | Static data | No setting | No setting |
| — | | Calculation range of element 4 | | | | | — | |

Ex.

❷ of No.1: When a variable length element is the final element of a packet or when there is only a fixed length element outside the calculation range after a variable length element

• Can be set.

| Element 1 | Element 2 | Element 3 | ... | Element 32 |
|---|---|---|---|---|
| Static data | Variable length element | Non-verified reception (fixed length) | No setting | No setting |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | ... | Element 32 |
|---|---|---|---|---|
| Static data | Variable length element | Variable length element | No setting | No setting |

Ex.

❸ of No.1: When the variable length element determines the data length by itself

• Can be set.

| Element 1 | Element 2 | Element 3 | ... | Element 32 |
|---|---|---|---|---|
| Static data | Variable length element (Determines the data length by itself.) | Non-conversion variable (variable length) | No setting | No setting |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | ... | Element 32 |
|---|---|---|---|---|
| Static data | Variable length element (Does not determine the data length by itself.) | Non-conversion variable (variable length) | No setting | No setting |

Ex.

No.2: When the variable length element that does not determine the data length by itself is used within the length calculation range and the innermost length is placed before the variable length element

• Can be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | ... | Element 32 |
|---|---|---|---|---|---|---|
| Static data | Length | Length | Static data | Variable length element | No setting | No setting |
| — | Calculation range of element 2 | | | | — | |
| — | | Calculation range of element 3 | | | — | |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | ... | Element 32 |
|---|---|---|---|---|---|---|
| Static data | Length | Variable length element (The innermost length is before a variable length element.) | Length | Static data | No setting | No setting |
| — | Calculation range of element 2 | | | | — | |
| — | | Calculation range of element 4 | | | — | |

Ex.

❶ of No.3: When static data is placed after the variable length element and the innermost length of both the variable length element and static data is the same.

• Can be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | Element 6 | Element 7 | ... | Element 32 |
|---|---|---|---|---|---|---|---|---|
| Static data | Length | Variable length element | Non-verified reception (fixed length) | Static data | Length | Variable length element | No setting | No setting |
| — | | | | | Calculation range of element 6 | | — | |
| — | | Calculation range of element 2 | | | | | — | |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | Element 6 | Element 7 | ... | Element 32 |
|---|---|---|---|---|---|---|---|---|
| Static data | Length | Variable length element | Non-verified reception (fixed length) | Static data (Its innermost length is not the same.) | Length | Variable length element | No setting | No setting |
| — | | | | Calculation range of element 6 | | | — | |
| — | | Calculation range of element 2 | | | | | — | |

Ex.

❷ of No.3: When the variable length element is the final element within the innermost length calculation range or there is only a fixed length element with the same innermost length that is placed after the variable length element

• Can be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | Element 6 | ... | Element 32 |
|---|---|---|---|---|---|---|---|
| Static data | Length | Variable length element | Non-verified reception (fixed length) | Length | Non-verified reception (fixed length) | No setting | No setting |
| — | | Calculation range of element 2 | | Calculation range of element 5 | | — | |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | Element 4 | Element 5 | Element 6 | ... | Element 32 |
|---|---|---|---|---|---|---|---|
| Static data | Length | Variable length element | Non-verified reception (fixed length) | Length | Non-verified reception (fixed length) | No setting | No setting |
| — | | | | Calculation range of element 5 (Its innermost length is not the same.) | | — | |
| — | | Calculation range of element 2 | | | | — | |

Ex.

❸ of No.3: When the variable length element determines the data length by itself

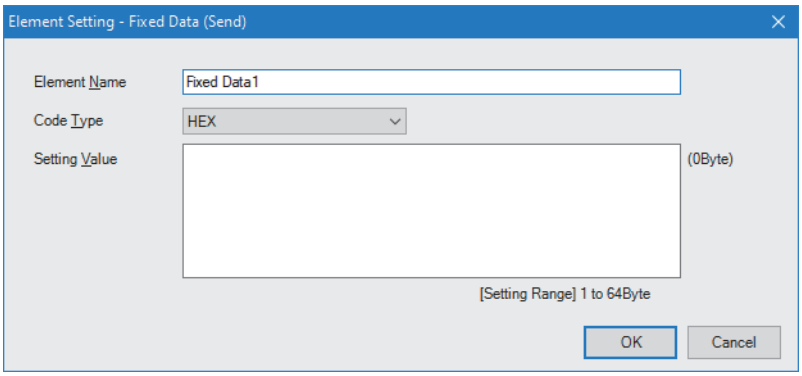
• Can be set.

| Element 1 | Element 2 | Element 3 | Element 4 | ... | Element 32 |
|---|---|---|---|---|---|
| Static data | Length | Variable length element (Determines the data length by itself.) | Non-conversion variable (variable length) | No setting | No setting |
| — | | Calculation range of element 2 | | — | |

• Cannot be set.

| Element 1 | Element 2 | Element 3 | Element 4 | ... | Element 32 |
|---|---|---|---|---|---|
| Static data | Length | Variable length element (Does not determine the data length by itself.) | Non-conversion variable (variable length) | No setting | No setting |
| — | | Calculation range of element 2 | | — | |

## ■Element type (Static data)




Use this type when specific codes such as commands or character strings are contained in the packet.

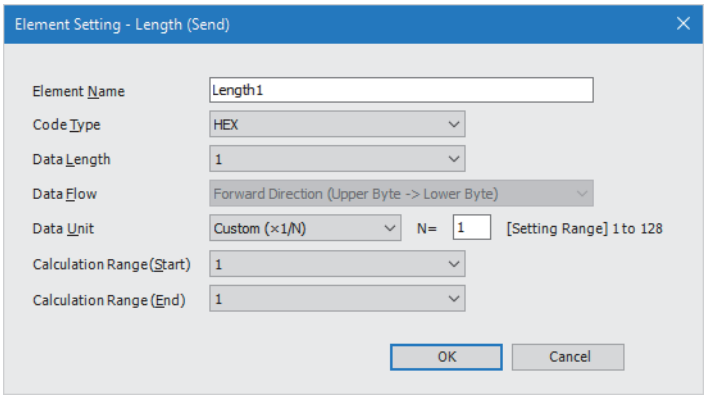
• When sending: The specified code or character string is sent.

• When receiving: The receive data is verified.

Up to 32 static data elements can be placed anywhere in the data part when "Resource Setting" is set to either "Normal" or "Extension 1", and up to 128 static data elements can be placed anywhere in the data part when "Resource Setting" is set to "Extension 2".

| Item | Description | Setting range | Display example in "Element Setting" |
|---|---|---|---|
| Element Name | Set the element name. | 48 characters<br>(Default: Fixed Data + Element No.) | — |
| Code Type | Select a data type of the setting value. | • ASCII String<br>• HEX<br>(Default: HEX) | — |
| Setting Value | Set data within 1 to 64 bytes. | • When "Code Type" is "ASCII String": 20H to 7EH<br>• When "Code Type" is "HEX": 00H to FFH<br>(Default: Blank) | • When "Code Type" is "ASCII String": "1234"<br>• When "Code Type" is "HEX": 1234 |

## ■Element type (Length)




The length code is used when there is an element that indicates the data length in the packet.

- When sending: Automatically calculates the data length in the specified range, and adds it to the packet.
- When receiving: From the received data, the data (value) corresponding to the length is verified as the specified range's data length.

Up to 31 length elements can be placed anywhere in the data part when "Resource Setting" is set to either "Normal" or "Extension 1", and up to 127 length elements can be placed anywhere in the data part when "Resource Setting" is set to "Extension 2".

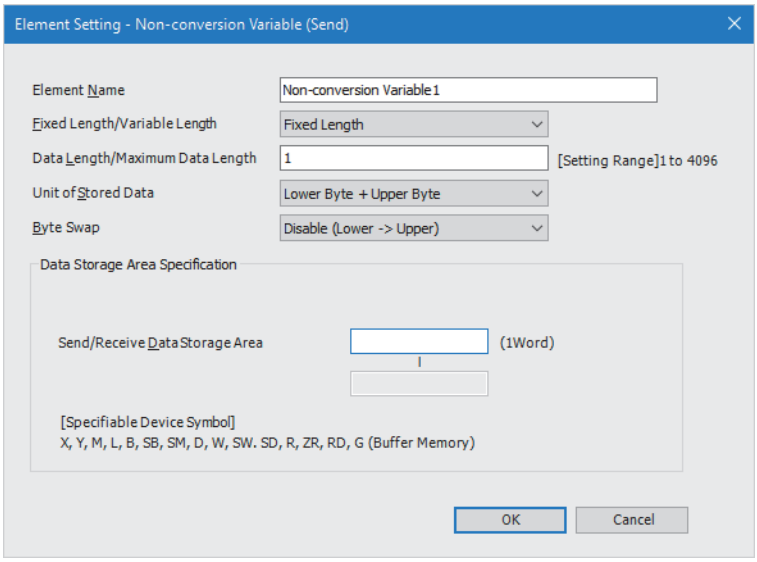
| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Element Name | Set the element name. | | 48 characters<br>(Default: Length + Element No.) | — |
| Code Type | Select the data length type. | | • ASCII HEX<br>• ASCII DEC<br>• HEX<br>(Default: HEX) | • Hexadecimal<br>• Decimal<br>• HEX |
| Data Length | Select the data length on the line. | | 1 to 4<br>(Default: 1) | 4B |
| Data Flow*1 | Forward Direction (Upper Byte → Lower Byte) | • When sending: Sends the calculated length in order from the upper byte.<br>• When receiving: Receives the data in order from the upper byte. | • Forward Direction (Upper Byte → Lower Byte)<br>• Reverse Direction (Lower Byte → Upper Byte)<br>• Byte Swap (by Word)<br>(Default: -) | Forward |
| | Reverse Direction (Lower Byte → Upper Byte) | • When sending: Sends the calculated length in order from the lower byte.<br>• When receiving: Receives the data in order from the lower byte. | | Reverse |
| | Byte Swap (by Word)*2 | • When sending: Interchanges the bytes in word units and sends the calculated length.<br>• When receiving: Interchanges the bytes in word units and receives the data. | | Byte |
| Data Unit | Select the data unit.*3 | | • Byte (×1)<br>• Bit (×8)<br>• Word (×1/2)<br>• Double Word (×1/4)<br>• Custom (×1/N)*4<br>(Default: Byte (×1)) | • ×1<br>• ×8<br>• ×1/2<br>• ×1/4<br>• ×1/N |
| Calculation Range (Start)*5 | Select the start packet element number for the range to be calculated. | | • 1 to 32<br>• 1 to 128<br>(Default: 1) | 1 to 128 |
| Calculation Range (End)*5 | Select the end packet element number for the range to be calculated. | | • 1 to 32<br>• 1 to 128<br>(Default: 1) | |

*1 This cannot be set if "Data Length" is 1 byte.
*2 This can be selected only when "Data Length" is set to 4 bytes.
*3 If the data length cannot be divided when sending, decimal places are rounded up.
*4 N is a value between 1 and 128.
*5 The number will be 1 to 32 when "Resource Setting" is set for either "Normal" or "Extension 1", and 1 to 128 for "Extension 2".

**Restriction**

- If there are no elements other than length, an element error occurs. (When using length, one or more elements other than length are required.)
- If the calculation result exceeds the number of digits set with "Data Length", the excessive digit value is discarded (invalidated). (For example, if the data length is 2 bytes and the data size calculation results are 123 bytes, the length value will be 23.)
- When "Code Type" is "ASCII HEX", a mismatch will occur if a character string other than "0" to "9", "A" to "F", and "a" to "f" is received.
- When "Code Type" is "ASCII DEC", a mismatch will occur if a character string other than "0" to "9" is received.
- When converting to ASCII characters during send, Ethernet-equipped modules use "0" to "9" and "A" to "F".
- When arranging multiple length elements, the length calculation ranges must not overlap even partially, and the previous length calculation range may not exceed the arranged length.
- A length element cannot be arranged at the final position of the packet elements.

## ■Element type (Non-conversion variable)




Use this type to send data of the CPU module device or buffer memory as part of the send packet, or to store part of the receive packet in the CPU module device or buffer memory.

Up to 32 non-conversion variables can be placed in one packet when "Resource Setting" is set to either "Normal" or "Extension 1", and up to 128 non-conversion variables can be placed in one packet when "Resource Setting" is set to "Extension 2".

- Use for send packets: For the data length storage area, specify the size of data to be sent in the unit of bytes.
- Use for receive packets: The Ethernet-equipped module stores the size of received data in the unit of bytes in the data length storage area. (When the receive data is an odd number of bytes, the last data is stored with one byte of 00H.)

| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Element Name | Set the element name. | | 48 characters<br>(Default: Non-conversion Variable + Element No.) | — |
| Fixed Length/ Variable Length | Fixed Length | The data whose length is fixed is sent and received. | • Fixed Length<br>• Variable Length<br>(Default: Fixed Length) | — |
| | Variable Length | • When sending: The data length is specified at the time of the protocol execution and the data is sent.<br>• When receiving: The data whose length is variable is received. | | |
| Data Length/ Maximum Data Length*3 | Set the data length of the send/receive data in byte units.<br>(When "Fixed Length/Variable Length" is "Variable Length", set the maximum data length that can be specified for the data length storage area.) | | • 1 to 4096<br>• 1 to 10238<br>(Default: 1) | 4096B |
| Unit of Stored Data | Lower Byte + Upper Byte | • When sending: Each one word (2 bytes) data in the data storage area is sent in the order of the lower byte to the upper byte.<br>• When receiving: The receive data is stored to the data storage area in the order of the lower byte to the upper byte. | • Lower Byte + Upper Byte<br>• Lower Bytes Only<br>(Default: Lower Byte + Upper Byte) | Lower/Upper |
| | Lower Bytes Only | • When sending: Each lower byte data in the data storage area is sent. The Ethernet-equipped module ignores the upper byte data.<br>• When receiving: The receive data is stored to each lower byte in the data storage area. The Ethernet-equipped module stores 00H in the upper byte. | | Lower |

| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Byte Swap | Disable (Lower → Upper) | Byte swap is not performed. | • Disable (Lower → Upper)<br>• Enable (Upper → Lower)<br>• Enable (Upper → Lower, two-word units)*2<br>• Enable (Upper → Lower, four-byte units)*2<br>(Default: Disable (Lower → Upper)) | No Swap |
| | Enable (Upper → Lower) | • When sending: Data in the upper byte and lower byte are swapped by one word (2 bytes) and sent. When "Unit of Stored Data" is "Lower Byte + Upper Byte" and "Data Length" is an odd number of bytes, the upper byte is sent at transmission of the last byte. When "Unit of Stored Data" is "Lower Bytes Only" and "Data Length" is an odd number of bytes, data without any byte swap is sent at transmission of the last byte.<br>• When receiving: Data in the upper byte and lower byte are swapped by word and sent. When "Unit of Stored Data" is "Lower Byte + Upper Byte" and "Data Length" is an odd number of bytes, the last byte is stored to the upper byte. When "Unit of Stored Data" is "Lower Bytes Only" and "Data Length" is an odd number of bytes, the last byte is stored without any byte swap. | | Swap |
| | Enable (Upper → Lower, two-word units)*1 | • When sending: Data in the upper byte and lower byte are swapped by two words (4 bytes) and sent. When "Data Length" is not a multiple of 4, the remainder of the data is sent without any byte swap.<br>• When receiving: Data in the upper word and lower word are swapped in two-word units and sent. When "Data Length" is not a multiple of 4, the remainder of the data is stored without any byte swap. | | Swap (2W) |
| | Enable (Upper → Lower, four-byte units)*1 | • When sending: Four-byte data is sent in order from the upper byte. When "Data Length" is not a multiple of 4, the remainder of the data is sent without any byte swap.<br>• When receiving: The receive data is received in four-byte units in order from the upper byte. When "Data Length" is not a multiple of 4, the remainder of the data is stored without any byte swap. | | Swap (4B) |
| Data Storage Area Specification | Specify the start device for storing the variable value. | | Refer to devices that can be specified on the following page.<br>☞ Page 262 Devices that can be specified<br>(Default: Blank) | • [W0-W9]<br>• [W0][W1-W0A] |

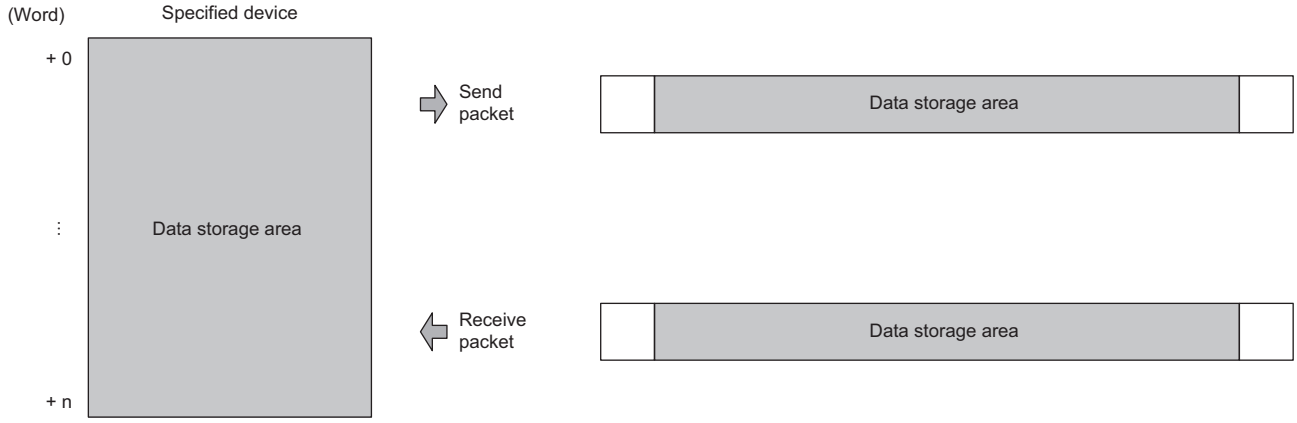
*1 This cannot be set if "Data Length" is 3 bytes or less. This can be selected only when "Data Length" is set to 4 bytes or more.

*2 The device with the firmware version of "71" or later is available.

*3 The number will be 1 to 4096 when "Resource Setting" is set for either "Normal" or "Extension 1", and 1 to 10238 for "Extension 2".

• When "Fixed Length/Variable Length" is "Fixed Length"

The area after the specified device number becomes the data storage area.




For send packet: Send data is stored by the program.

For receive packet: Receive data is stored by the Ethernet-equipped module.

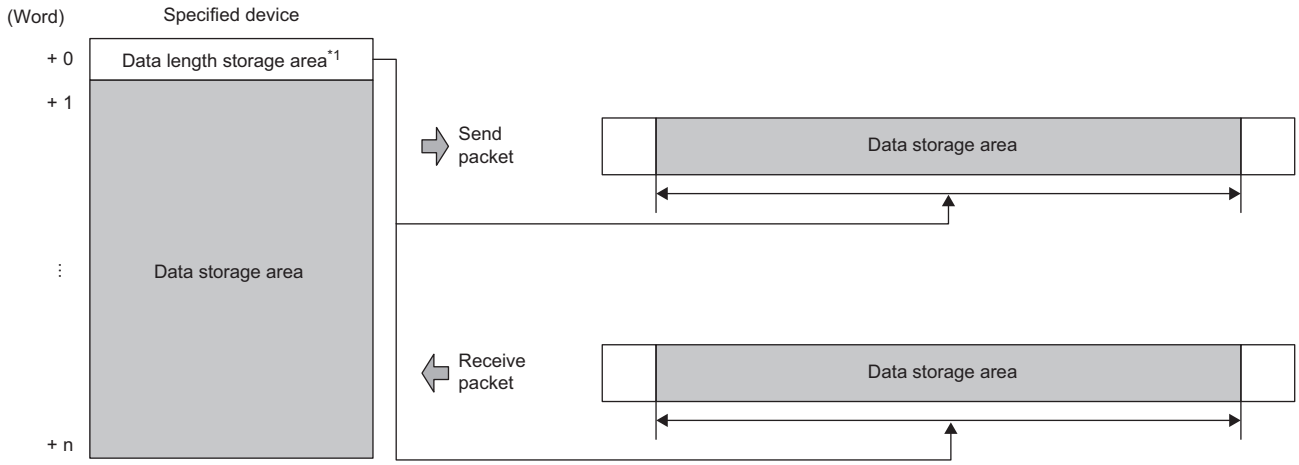
**Point**

The occupied data storage area differs according to the "Unit of Stored Data".

- When "Lower Byte + Upper Byte" is selected, the same size as the data length is occupied. (However, when the data length of a send packet is an odd number, the upper byte (lower byte for "Byte Swap") of the end device is not sent. When the data length of a receive packet is an odd number, the last data is stored with one byte of 00H.)
- When "Lower Bytes Only" is selected, a size double the data length is occupied.

• When "Fixed Length/Variable Length" is "Variable Length"

The first word of the specified device number becomes the data length storage area, and the area after the specified device number + 1 becomes the data storage area.




For send packet: Send data is stored by the program.

For receive packet: Receive data is stored by the Ethernet-equipped module.

*1 The data length unit is byte fixed.

**Point**

The occupied data storage area differs according to the "Unit of Stored Data".

- When "Lower Byte + Upper Byte" is selected, the same size as the data length + one word (for data length storage area) are occupied. (However, when the data length of a send packet is an odd number, the upper byte (lower byte for "Byte Swap") of the end device is not sent. When the data length of a receive packet is an odd number, the last data is stored with one byte of 00H.)
- When "Lower Bytes Only" is selected, a size double the data length + one word (for data length storage area) is occupied.

Restriction

For restrictions, refer to the following.

☞ Page 358 Restrictions on placement of variable length elements in a receive packet

## Precautions

When a non-conversion variable (variable length) is used in a TCP receive packet, the placed non-conversion variable (variable length) is not included in the length calculation range, and if no static data is placed in the subsequent elements, the system will continue to wait until the following number of bytes of response data are received, regardless of the value set in "Data Length/Maximum Data Length" of the non-conversion variable (variable length).

- When "Resource Setting" is set to either "Normal" or "Extension 1": 4096 bytes
- When "Resource Setting" is set to "Extension 2": 10238 bytes

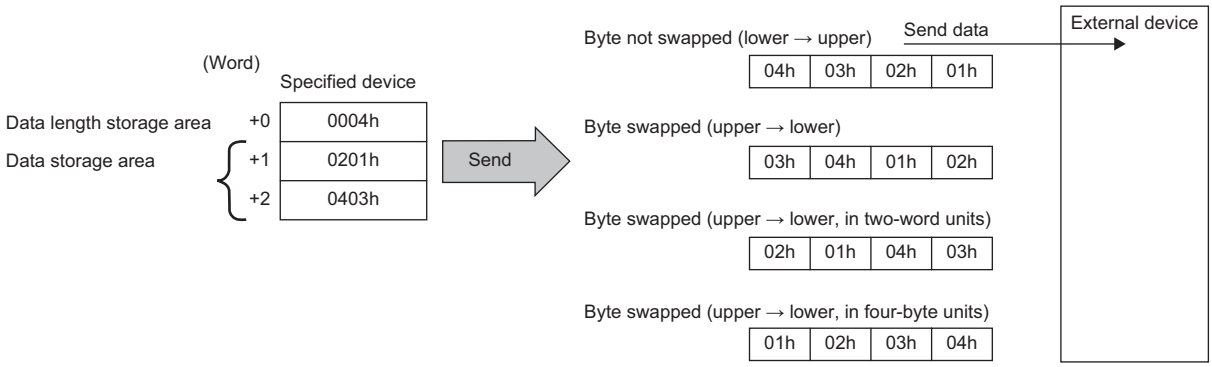
• Byte swap operation

Specific operation examples for each byte swap setting are shown below.

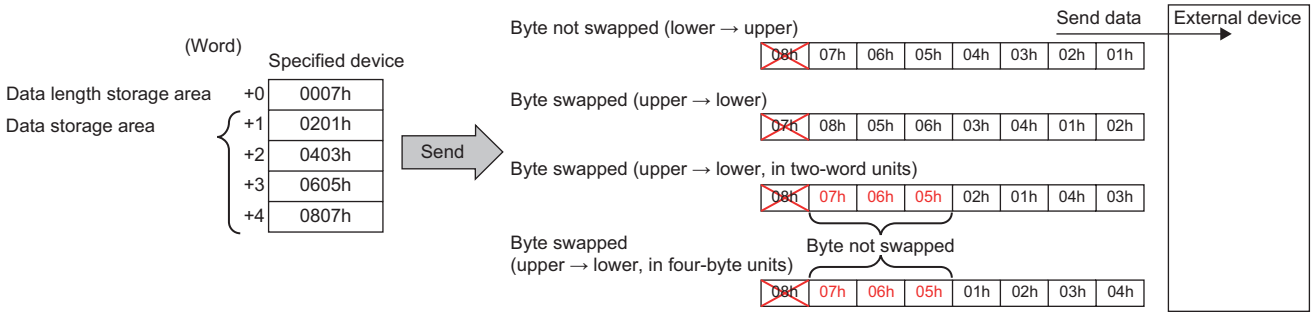
**Ex.**

Example of sending when "Unit of Stored Data" is set to "Lower Byte + Upper Byte"

• When "Data Length" is a multiple of 4



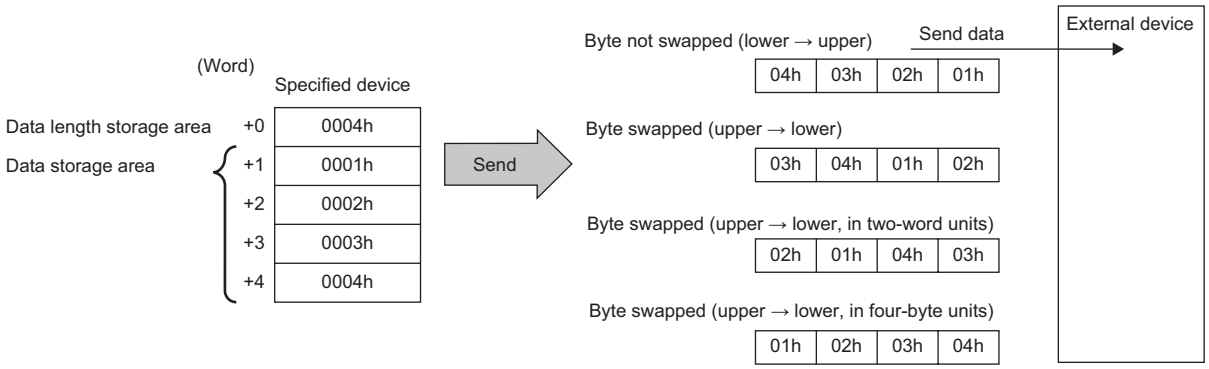

• When "Data Length" is not a multiple of 4



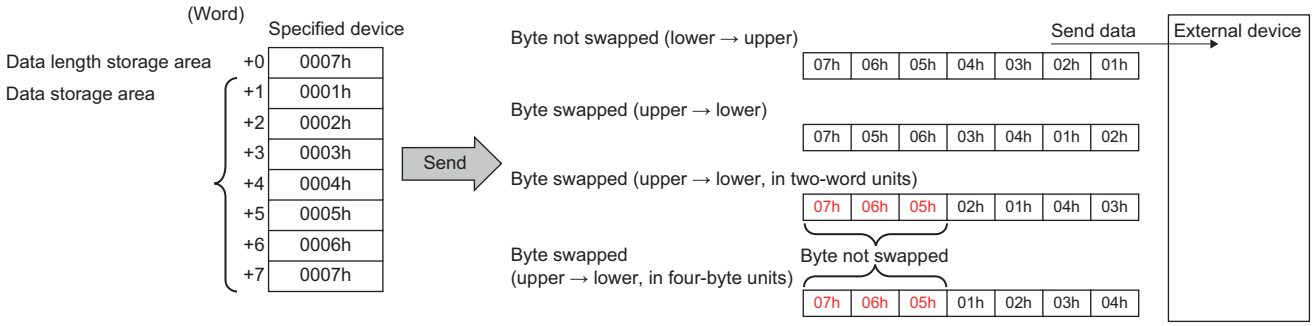

**Ex.**

Example of sending when "Unit of Stored Data" is set to "Lower Byte Only"

• When "Data Length" is a multiple of 4



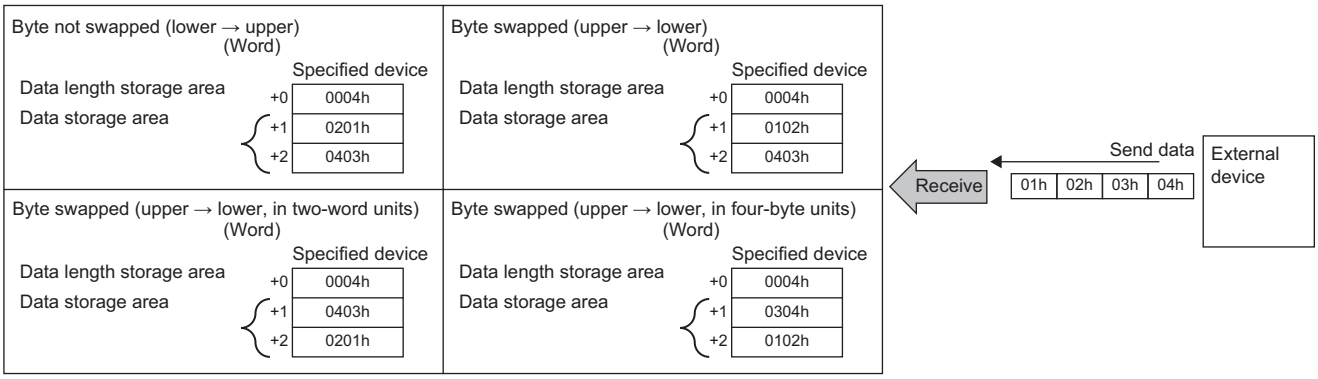

• When "Data Length" is not a multiple of 4



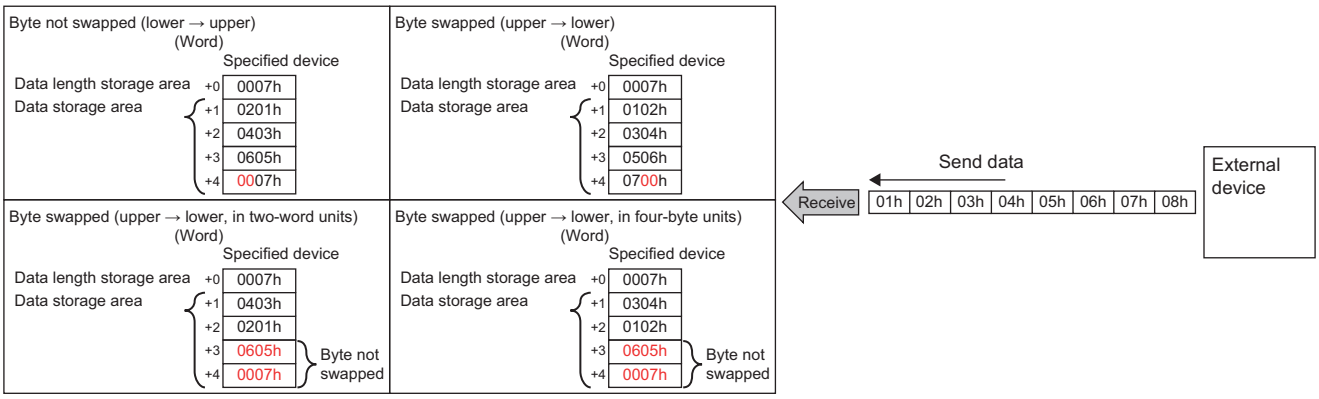
**Ex.**

Example of receiving when "Unit of Stored Data" is set to "Lower Byte + Upper Byte"

- When "Data Length" is a multiple of 4



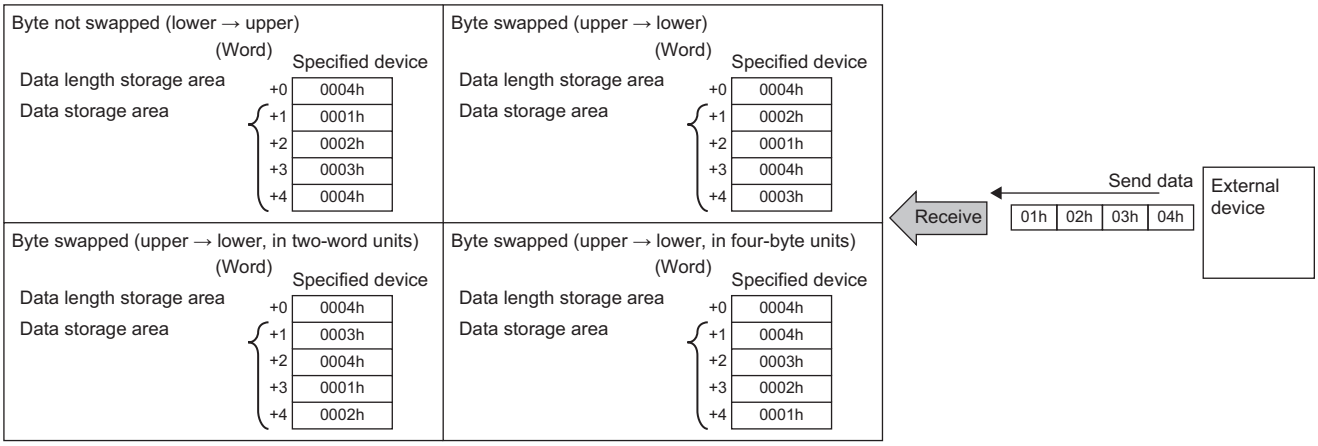

- When "Data Length" is not a multiple of 4



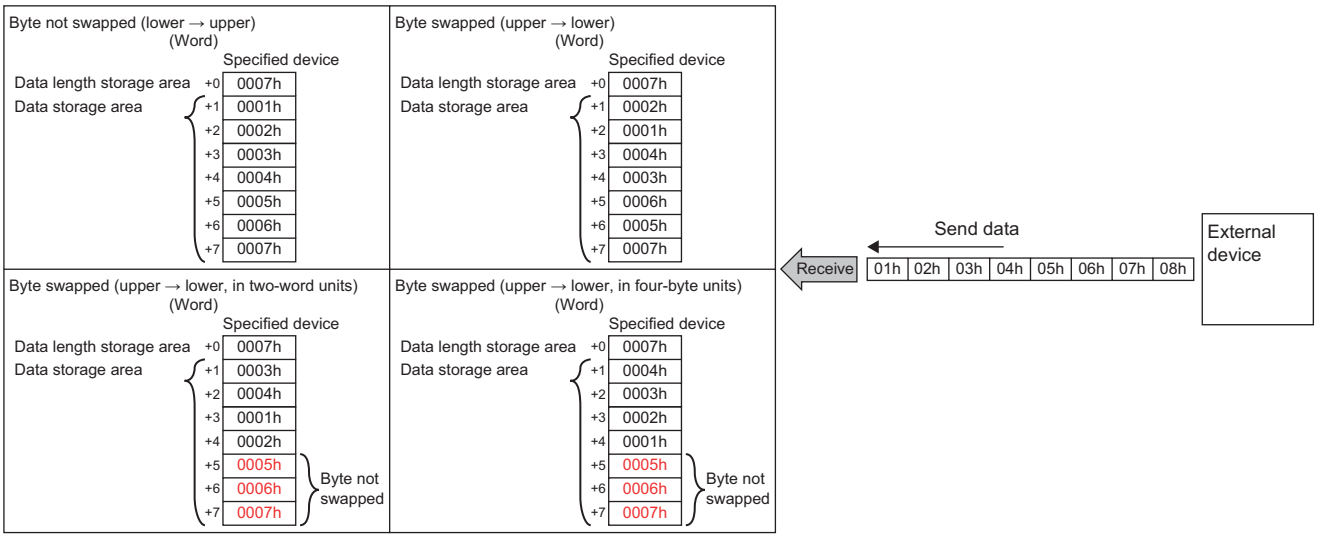

**Ex.**

Example of receiving when "Unit of Stored Data" is set to "Lower Byte Only"

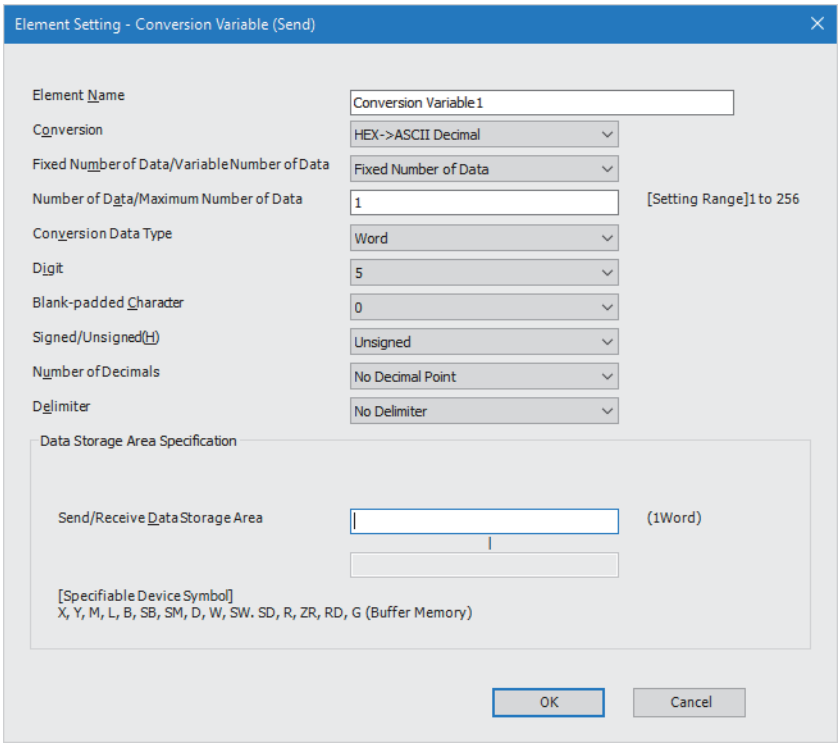
- When "Data Length" is a multiple of 4



• When "Data Length" is not a multiple of 4



## ■Element type (Conversion variable)




Use this type to send numerical value data of the CPU module device or buffer memory while converting it into an ASCII string or to store receive data (ASCII string) in the CPU module device or buffer memory while converting it into numerical value data.

Up to 32 conversion variables can be placed in one packet when "Resource Setting" is set to either "Normal" or "Extension 1", and up to 128 conversion variables can be placed in one packet when "Resource Setting" is set to "Extension 2".

| Item | Description | | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|---|
| Element Name | Set the element name. | | | 48 characters<br>(Default: Conversion Variable + Element No.) | — |
| Conversion | When sending | HEX→ASCII Hexadecimal | Numerical value data stored in the data storage area is converted into a hexadecimal ASCII string. | • HEX→ASCII Hexadecimal<br>• HEX→ASCII Decimal<br>(Default: HEX→ASCII Decimal) | →16<br>→10 |
| | | HEX→ASCII Decimal | Numerical value data stored in the data storage area is converted into a decimal ASCII string. | | |
| | When receiving | ASCII Hexadecimal→HEX | Receive data is handled as a hexadecimal ASCII string, converted into numerical value data, and stored in the data storage area. | • ASCII Hexadecimal→HEX<br>• ASCII Decimal→HEX<br>(Default: ASCII Decimal→HEX) | 16→<br>10→ |
| | | ASCII Decimal→HEX | Receive data is handled as a decimal ASCII string, converted into numerical value data, and stored in the data storage area. | | |
| Fixed Number of Data/Variable Number of Data | Fixed Number of Data | The quantity of send/receive data is fixed. | | • Fixed Number of Data<br>• Variable Number of Data<br>(Default: Fixed Number of Data) | — |
| | Variable Number of Data | • When sending: Data quantity to be sent is specified at the time of the protocol execution and the data is sent.<br>• When receiving: The data whose quantity is variable is received. (When "Variable Number of Digits" is set, a delimiter is required.) | | | |

| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Number of Data/ Maximum Number of Data | Set the quantity of send/receive data using the unit specified in "Conversion Data Type".<br>(When "Variable Number of Data" is set, set the maximum data quantity that can be specified for the data quantity storage area.) | | 1 to 256<br>(Default: 1) | 1 |
| Conversion Data Type | Select the number of words to be handled as a piece of data regarding data in the data storage area.<br>When "Word" is selected, a word is converted as a piece of data.<br>When "Double Word" is selected, two words are converted as a piece of data.<br>When "FLOAT [Single Precision]" is selected, two words are converted as a piece of data.<br>When "FLOAT [Double Precision]" is selected, four words are converted as a piece of data. | | • Word<br>• Double Word<br>• FLOAT [Single Precision]<br>• FLOAT [Double Precision]<br>(Default: Word) | • W<br>• D<br>• Single<br>• Double |
| Digit*2 | 1 to 15 | Select the number of digits per data of the send/receive data.<br>If the specified number of digits is not reached, the upper digits are padded with the character specified in "Blank-padded Character". | • 1 to 16<br>• Variable Number of Digits<br>(Default: 5) | • Digit (1)<br>• Digit (variable) |
| | Variable Number of Digits | • When sending: Only the data section converted into an ASCII string is sent in variable length.<br>• When receiving: An ASCII string of only the data section is received in variable length. When "Number of Data/ Maximum Number of Data" is 2 or more, a delimiter is required.<br>When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Single Precision]", the maximum number of digits is 7.<br>When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Double Precision]", the maximum number of digits is 15. | | |
| | 8 | When "Conversion" is "HEX→ASCII Hexadecimal" or "ASCII Hexadecimal→HEX" and "Conversion Data Type" is "FLOAT [Single Precision]", the number of digits is fixed to 8. | | |
| | 16 | When "Conversion" is "HEX→ASCII Hexadecimal" or "ASCII Hexadecimal→HEX" and "Conversion Data Type" is "FLOAT [Double Precision]", the number of digits is fixed to 16. | | |
| Blank-padded Character | Select the character for padding the upper digits to satisfy the specification of digits when the number of digits is not "Variable Number of Digits" and the send/receive data does not satisfy the specified number of digits. | | • 0<br>• Space<br>(Default: 0) | • Padding (0)<br>• Padding (S) |
| Signed/ Unsigned*1 | Unsigned | Signed characters of send/receive data are not added or specified. | • Unsigned<br>• Signed/Positive Sign Character (None)<br>• Signed/Positive Sign Character (+)<br>• Signed/Positive Sign Character (0)<br>• Signed/Positive Sign Character (Space)<br>(Default: Unsigned) | • Unsigned<br>• Signed (none)<br>• Signed (+)<br>• Signed (0)<br>• Signed (S) |
| | Signed/Positive Sign Character (None) | Signed characters are not added or specified when the send/receive data is a positive number.<br>Signed characters for negative numbers are fixed to "-". | | |
| | Signed/Positive Sign Character (+) | • When sending: If send data is a positive number, the set signed character is added. Signed characters for negative numbers are fixed to "-".<br>• When receiving: Specify the signed character for the positive number of receive data. Signed characters for negative numbers are fixed to "-". | | |
| | Signed/Positive Sign Character (0) | | | |
| | Signed/Positive Sign Character (Space) | | | |

| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Number of Decimals*1*3 | No Decimal Point | Decimal point is not added to send/receive data. | • No Decimal Point<br>• Variable Point<br>• 1 to 14<br>(Default: No Decimal Point) | • Decimal point (none)<br>• Decimal point (variable)<br>• Decimal point (1) |
| | Variable Point | When "Conversion Data Type" is "Word" or "Double Word", a decimal point is added to send/receive data based on the value stored at the decimal point position of the data storage area. When "Conversion Data Type" is "FLOAT [Single Precision]" or "FLOAT [Double Precision]", a decimal point is added to send/receive data based on the value stored in the exponent of the numerical value data. | | |
| | 1 to 14 | Specify the number of digits for the decimal part of send/receive data. | | |
| Delimiter | Select the delimiter for data to be entered after each data piece. When the data quantity is 2 or more, a delimiter is not placed after the final data.<br>When sending: A desired delimiter can be set.<br>When receiving: "No Delimiter" cannot be set in either of the following cases.<br>• When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data" and also "Digit" is "Variable Number of Digits"<br>• When "Number of Data/Maximum Number of Data" is 2 or more and also "Digit" is "Variable Number of Digits" | | • No Delimiter<br>• Comma<br>• Space<br>(Default: No Delimiter) | • Delimiter (none)<br>• Delimiter (C)<br>• Delimiter (S) |
| Data Storage Area Specification | Specify the start device for storing the variable value. | | Refer to devices that can be specified on the following page.<br>☞ Page 262 Devices that can be specified<br>(Default: Blank) | • [W0-W1]<br>• [W0][W1-W2] |

*1 This item can be set when "Conversion" is "HEX→ASCII Decimal" or "ASCII Decimal→HEX".

*2 The setting range differs according to the setting details of other setting items. For details, refer to the following.

| Conversion data type | Conversion details | Setting range for number of digits |
|---|---|---|
| Word | • HEX→ASCII Hexadecimal<br>• ASCII Hexadecimal→HEX | Variable Number of Digits, 1 to 4 |
| | • HEX→ASCII Decimal<br>• ASCII Decimal→HEX | Variable Number of Digits, 1 to 5 |
| Double Word | • HEX→ASCII Hexadecimal<br>• ASCII Hexadecimal→HEX | Variable Number of Digits, 1 to 8 |
| | • HEX→ASCII Decimal<br>• ASCII Decimal→HEX | Variable Number of Digits, 1 to 10 |
| FLOAT [Single Precision] | • HEX→ASCII Hexadecimal<br>• ASCII Hexadecimal→HEX | 8 |
| | • HEX→ASCII Decimal<br>• ASCII Decimal→HEX | Variable Number of Digits, 1 to 7 |
| FLOAT [Double Precision] | • HEX→ASCII Hexadecimal<br>• ASCII Hexadecimal→HEX | 16 |
| | • HEX→ASCII Decimal<br>• ASCII Decimal→HEX | Variable Number of Digits, 1 to 15 |

*3 The setting range differs according to the setting details of other setting items. For details, refer to the following.

| Conversion data type | Digit | Setting range for number of decimal places |
|---|---|---|
| Word | 1 | No Decimal Point |
| | 2 to 5 | • No Decimal Point<br>• Variable Point<br>• 1 to ("Digit" - 1) |
| | Variable Number of Digits | • No Decimal Point<br>• Variable Point<br>• 1 to 4 |

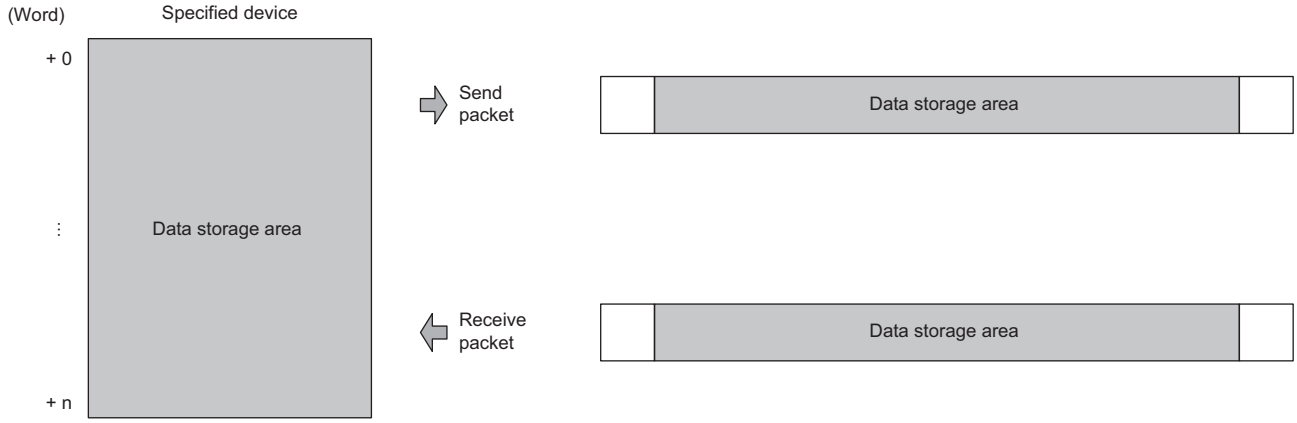
| Conversion data type | Digit | Setting range for number of decimal places |
|---|---|---|
| Double Word | 1 | No Decimal Point |
| | 2 to 10 | • No Decimal Point<br>• Variable Point<br>• 1 to ("Digit" - 1) |
| | Variable Number of Digits | • No Decimal Point<br>• Variable Point<br>• 1 to 9 |
| FLOAT [Single Precision] | 1 | No Decimal Point |
| | 2 to 7 | • No Decimal Point<br>• Variable Point<br>• 1 to ("Digit" - 1) |
| | Variable Number of Digits | • No Decimal Point<br>• Variable Point<br>• 1 to 6 |
| FLOAT [Double Precision] | 1 | No Decimal Point |
| | 2 to 15 | • No Decimal Point<br>• Variable Point<br>• 1 to ("Digit" - 1) |
| | Variable Number of Digits | • No Decimal Point<br>• Variable Point<br>• 1 to 14 |

• When "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data"

The area after the specified device number becomes the data storage area.

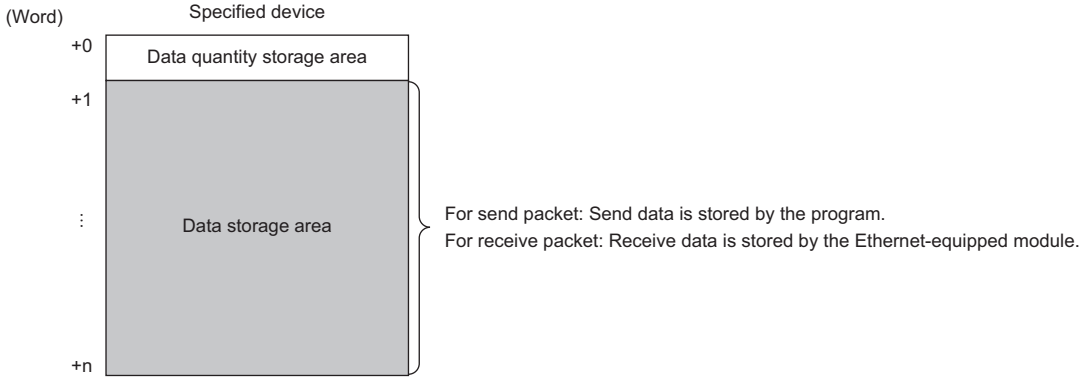
For send packet: Send data is stored by the program.

For receive packet: Receive data is stored by the Ethernet-equipped module.




• When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data"

The first word of the specified device number becomes the data quantity storage area, and the area after the specified device number + 1 becomes the data storage area.




• Occupied size of data storage area

The size occupying the data storage area differs according to the "Conversion Data Type" and "Number of Decimals" settings.

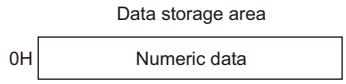
| Setting item | | Occupied size of data storage area per data |
|---|---|---|
| Conversion data type | Number of decimals | |
| Word | No Decimal Point, 1 to 4 | 1 word |
| | Variable Point | 2 words |
| Double Word | No Decimal Point, 1 to 9 | 2 words |
| | Variable Point | 4 words |
| FLOAT [Single Precision] | No Decimal Point, 1 to 6 | 2 words |
| | Variable Point | 2 words |
| FLOAT [Double Precision] | No Decimal Point, 1 to 14 | 4 words |
| | Variable Point | 4 words |

• Data storage area structure per data

This section describes data storage area structure per data.

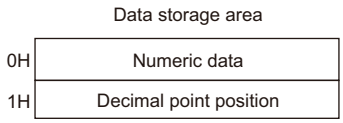
**Ex.**

"Conversion Data Type": "Word", "Number of Decimals": "No Decimal Point" or 1 to 4




**Ex.**

"Conversion Data Type": "Word", "Number of Decimals": "Variable Point"



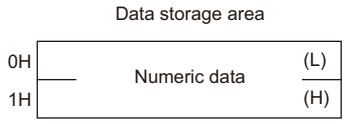

At the decimal point position, the decimal point position of send/receive data is stored as shown in the table below.

| Send/receive data (when the number of digits is 5) | 0H: Numerical value data | 1H: Decimal point position*1 |
|---|---|---|
| 12345 | 12345(3039H) | 1(1H) |
| 1234.5 | 12345(3039H) | 10(AH) |
| 123.45 | 12345(3039H) | 100(64H) |
| 12.345 | 12345(3039H) | 1000(3E8H) |
| 1.2345 | 12345(3039H) | 10000(2710H) |

*1 For send packets, an error occurs if a value other than the values shown in the table is stored at the decimal point position.

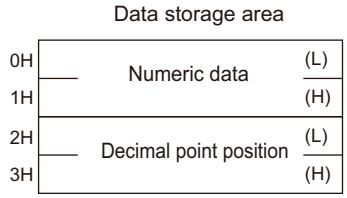
**Ex.**

"Conversion Data Type": "Double Word", "Number of Decimals": "No Decimal Point" or 1 to 9




**Ex.**

"Conversion Data Type": "Double Word", "Number of Decimals": "Variable Point"



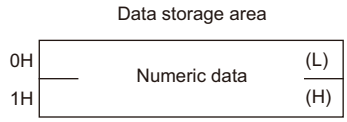

At the decimal point position, the decimal point position of send/receive data is stored as shown in the table below.

| Send/receive data (when the number of digits is 10) | 0H to 1H: Numerical value data | 2H to 3H: Decimal point position*1 |
|---|---|---|
| 1234567890 | 1234567890(499602D2H) | 1(1H) |
| 123456789.0 | 1234567890(499602D2H) | 10(AH) |
| 12345678.90 | 1234567890(499602D2H) | 100(64H) |
| 1234567.890 | 1234567890(499602D2H) | 1000(3E8H) |
| : | : | : |
| 1.234567890 | 1234567890(499602D2H) | 1000000000(3B9ACA00H) |

*1 For send packets, an error occurs if a value other than the values shown in the table is stored at the decimal point position.

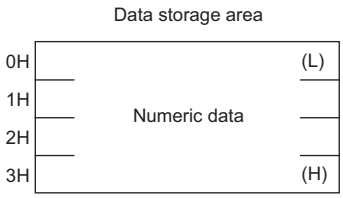
Ex.

"Conversion Data Type": "FLOAT [Single Precision]"




Ex.

"Conversion Data Type": "FLOAT [Double Precision]"

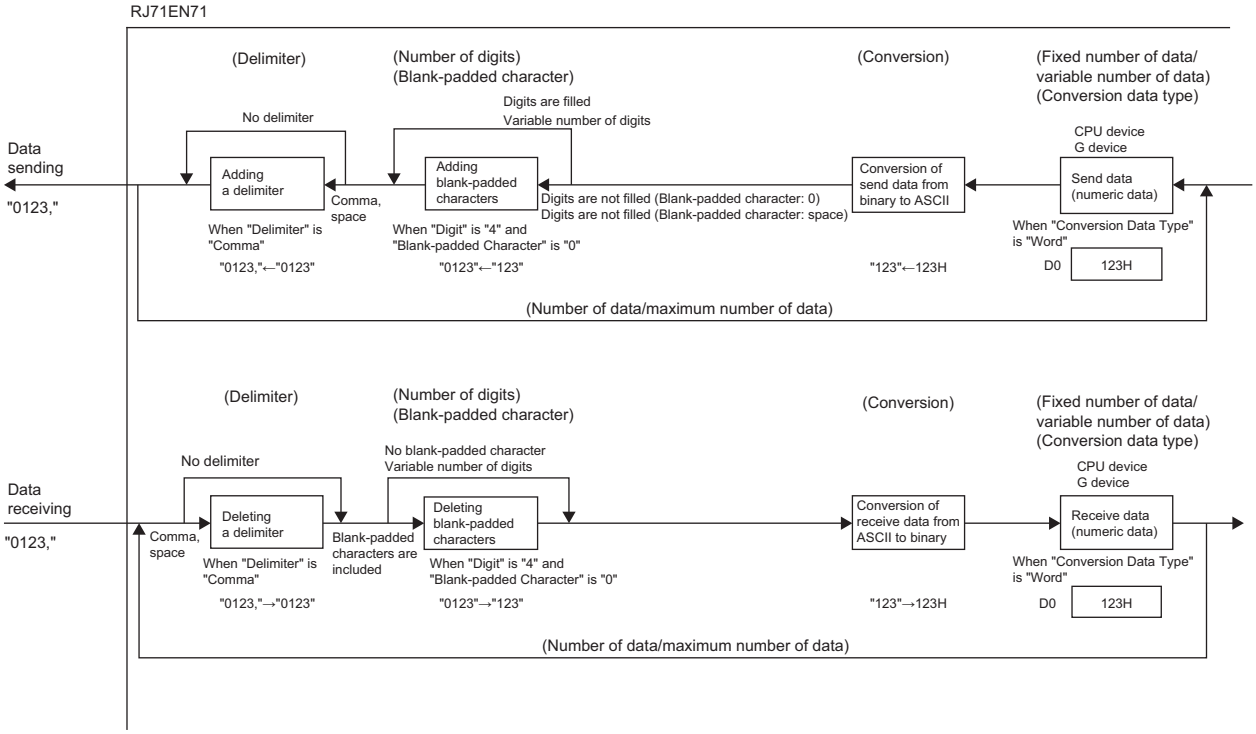



• Range of numerical values that can be handled in the data storage area

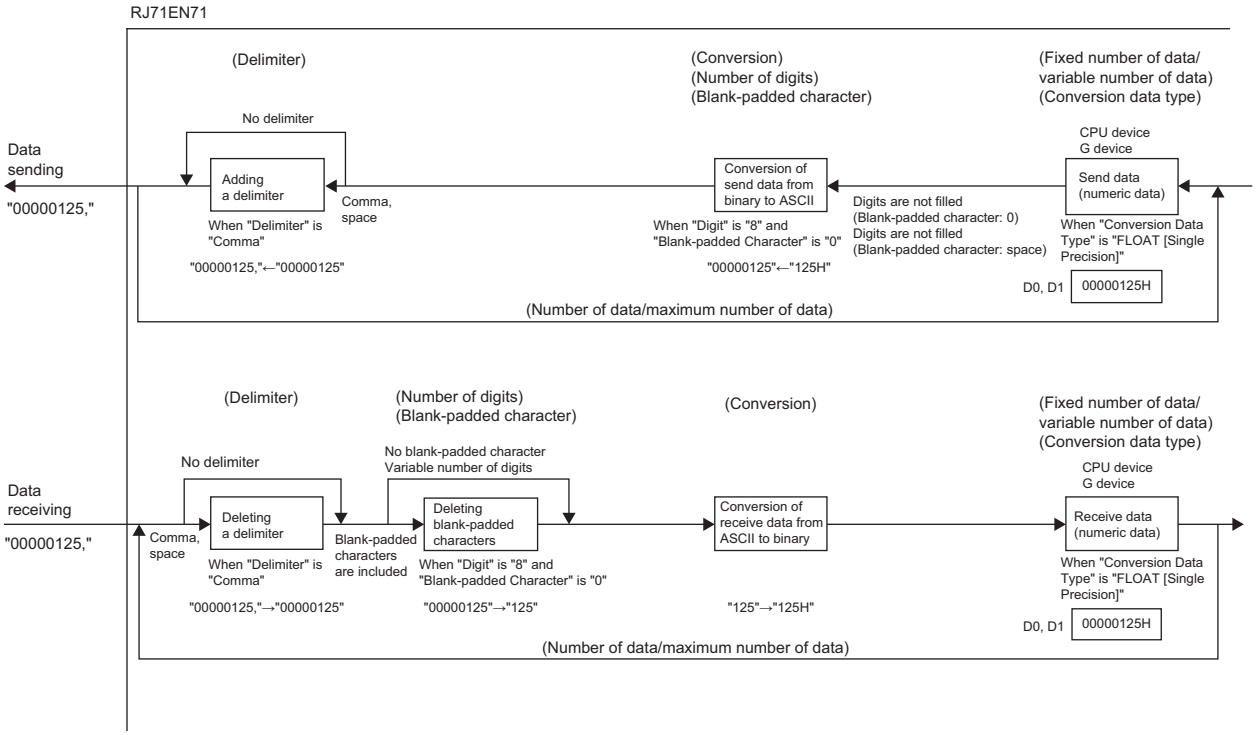
The range of numerical values that can be handled in the data storage area is as shown below.

| Conversion | Signed/unsigned | Conversion data type | Numerical value range |
|---|---|---|---|
| • HEX→ASCII Decimal<br>• ASCII Decimal→HEX | Unsigned | Word | 0 to 65535 (0H to FFFFH) |
| | | Double Word | 0 to 4294967295 (0H to FFFFFFFFH) |
| | | FLOAT [Single Precision] | 0 ≤ [Single-precision real number data] ≤ 9999999 |
| | | FLOAT [Double Precision] | 0 ≤ [Double-precision real number data] ≤ 999999999999999 |
| | Other than unsigned | Word | -32768 to 32767 (8000H to FFFFH, 0H to 7FFFH) |
| | | Double Word | -2147483648 to 2147483647 (80000000H to FFFFFFFFH, 0H to 7FFFFFFFH) |
| | | FLOAT [Single Precision] | -9999999 ≤ [Single-precision real number data] ≤ 9999999 |
| | | FLOAT [Double Precision] | -999999999999999 ≤ [Double-precision real number data] ≤ 999999999999999 |
| • HEX→ASCII Hexadecimal<br>• ASCII Hexadecimal→HEX | — | Word | 0H to FFFFH |
| | | Double Word | 0H to FFFFFFFFH |
| | | FLOAT [Single Precision] | 0H to FFFFFFFFH |
| | | FLOAT [Double Precision] | 0H to FFFFFFFFFFFFFFFFH |

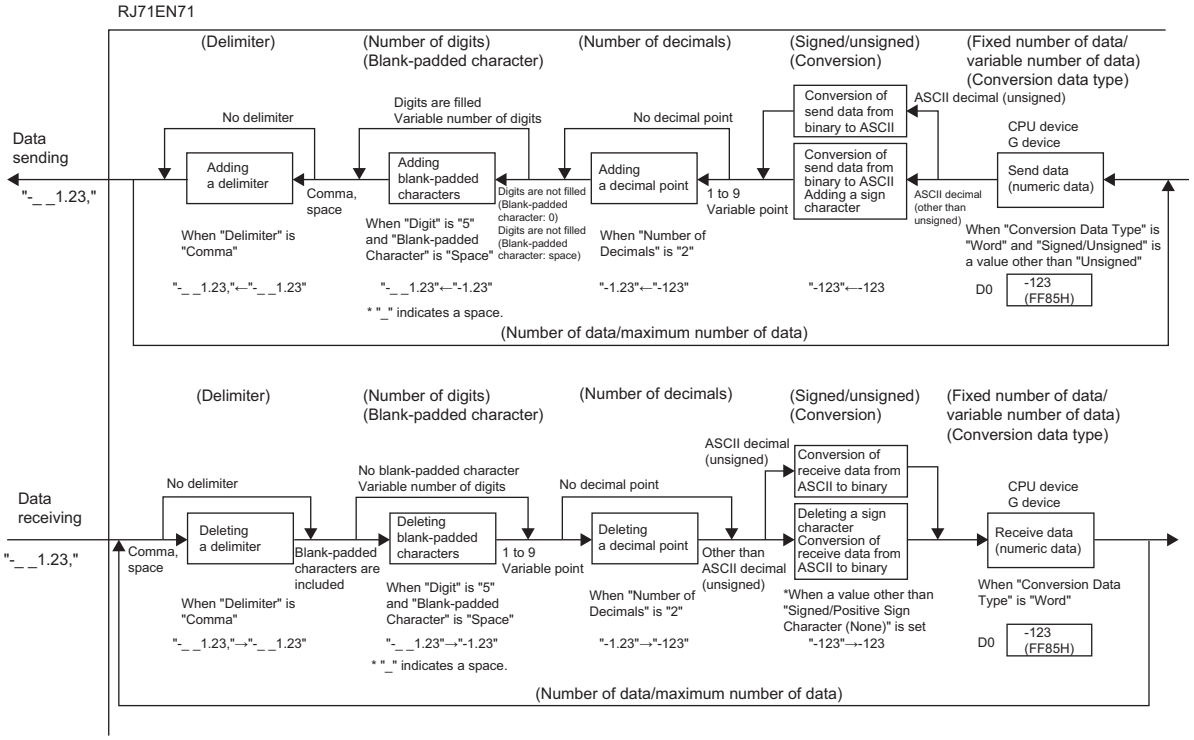
- Operation when "Conversion" is "HEX→ASCII Hexadecimal" or "ASCII Hexadecimal→HEX" and "Conversion Data Type" is "Word" or "Double Word"



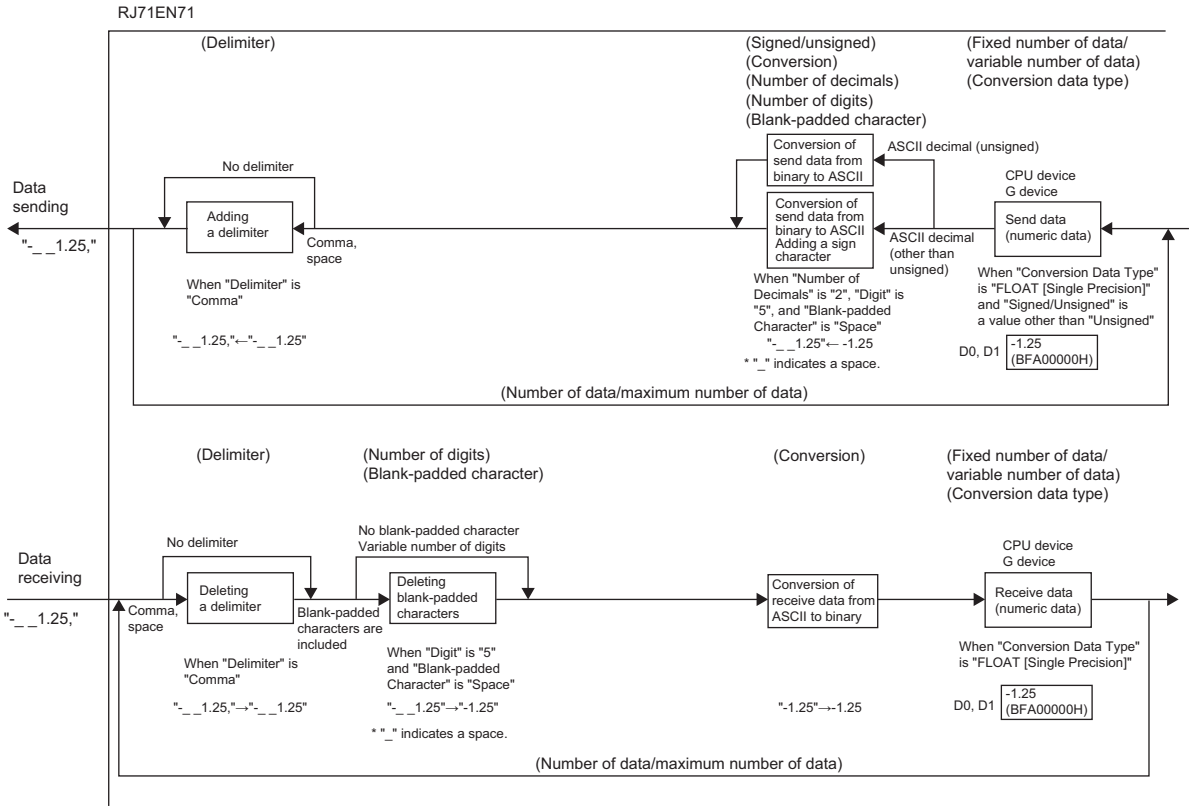

- Operation when "Conversion" is "HEX→ASCII Hexadecimal" or "ASCII Hexadecimal→HEX" and "Conversion Data Type" is "FLOAT [Single Precision]" or "FLOAT [Double Precision]"



- Operation when "Conversion" is "HEX→ASCII Decimal" or "ASCII Decimal→HEX" and "Conversion Data Type" is "Word" or "Double Word"




- Operation when "Conversion" is "HEX→ASCII Decimal" or "ASCII Decimal→HEX" and "Conversion Data Type" is "FLOAT [Single Precision]" or "FLOAT [Double Precision]"



• Character for padding blank digits

If the number of digits specified in "Digit" is not reached when sending, the upper digits are padded with the character specified in "Blank-padded Character".

When receiving, both "0" and "_ (space)" are processed as the character for padding blank digits regardless of the specification in "Blank-padded Character".

When six digits are specified in "Digit", data is handled as shown below. ("_" in the table indicates a space.)

| Receive data | Operation |
|---|---|
| 000120 | The first three digits are handled as the character for padding blank digits. |
| _ _ 0120 | The first three digits are handled as the character for padding blank digits. |
| 0 _ 0120 | The first three digits are handled as the character for padding blank digits. |
| _ _ _ 120 | The first three digits are handled as the character for padding blank digits. |
| _.00120 | The first three digits are handled as the character for padding blank digits. |
| 00012 _ | The data is handled as a verification mismatch. |
| _ _ _ 12 _ | The data is handled as a verification mismatch. |
| 0001 _ 0 | The data is handled as a verification mismatch. |
| 0._1200 | The data is handled as a verification mismatch. |
| _ (space only) | The data is handled as a verification mismatch. |

Ex.

The following table shows send data when a packet is formed in the structure of [Static data], [Conversion variable], [Static data], and the data stored in the data storage area is as follows: D0 = 837 (0345H), D1 = 18 (0012H). (120345H = 1180485 in decimal)

| Item | | Description | | |
|---|---|---|---|---|
| Example 1 | Fixed Number of Data/Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 1 | 1 | 1 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | Word | Word | Word |
| | Conversion | HEX→ASCII Decimal | HEX→ASCII Decimal | HEX→ASCII Decimal |
| | Digit | 5 | 5 | Variable Number of Digits |
| | Blank-padded Character | 0 | Space | — |
| | Signed/Unsigned | Signed/Positive Sign Character (None) | Signed/Positive Sign Character (+) | Signed/Positive Sign Character (+) |
| | Number of Decimals | No Decimal Point | 2 | No Decimal Point |
| | Delimiter | No Delimiter | Comma | Comma |
| | Send data*1 | [Static data]00837[Static data] | [Static data]+_ _8.37,[Static data] | [Static data]+837,[Static data] |
| Example 2 | Fixed Number of Data/Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 1 | 2 | 2 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | Double Word | Word | Word |
| | Conversion | HEX→ASCII Decimal | HEX→ASCII Decimal | HEX→ASCII Decimal |
| | Digit | 10 | 5 | 5 |
| | Blank-padded Character | 0 | Space | 0 |
| | Signed/Unsigned | Signed/Positive Sign Character (+) | Signed/Positive Sign Character (None) | Signed/Positive Sign Character (+) |
| | Number of Decimals | 8 | No Decimal Point | 2 |
| | Delimiter | No Delimiter | No Delimiter | Comma |
| | Send data*1 | [Static data]+00.01180485[Static data] | [Static data]_ _ 837 _ _ _18[Static data] | [Static data]+008.37,+000.18[Static data] |

*1 "_" indicates a one-byte space.

Ex.

The following table shows send data when a packet is formed in the structure of [Static data], [Conversion variable], [Static data], and the data stored in the data storage area is as follows: D0 = 34790 (87E6H), D1 = 16709 (4145H), D2 = 31458 (7AE2H), D3 = 16628 (40F4H), D4 = 0 (0000H), D5 = 17334 (43B6H), D6 = 65262 (FEEEH), D7 = 17095 (42C7H). (414587E6H, 40F47AE2H, 43B60000H, and 42C7FEEEH are 12.345678329467773, 7.640000343322754, 364, and 99.99790954589844 in single-precision real numbers.)

When numerical values do not fit in the decimal part after conversion due to the stored data and setting details in "Digit" and "Number of Decimals", the uppermost digit of the decimal part that does not fit is rounded off.

| Item | | Description*1 | | |
|---|---|---|---|---|
| Example 1 | Fixed Number of Data/Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 4 | 4 | 4 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | FLOAT [Single Precision] | FLOAT [Single Precision] | FLOAT [Single Precision] |
| | Conversion | HEX→ASCII Decimal | HEX→ASCII Decimal | HEX→ASCII Decimal |
| | Digit | Variable Number of Digits | Variable Number of Digits | 4 |
| | Blank-padded Character | — | — | 0 |
| | Signed/Unsigned | Unsigned | Signed/Positive Sign Character (None) | Signed/Positive Sign Character (+) |
| | Number of Decimals | Variable Point | 1 | Variable Point |
| | Delimiter | No Delimiter | Comma | Space |
| | Send data*2 | [Static data]12.345687.6436499.99791[Static data] | [Static data]12.3,7.6,364.0,100.0[Static data] | [Static data]+12.35_+07.64_+0364_+0100[Static data] |
| Example 2 | Fixed Number of Data/Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 4 | 4 | 4 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | FLOAT [Single Precision] | FLOAT [Single Precision] | FLOAT [Single Precision] |
| | Conversion | HEX→ASCII Decimal | HEX→ASCII Decimal | HEX→ASCII Decimal |
| | Digit | 5 | 6 | 7 |
| | Blank-padded Character | Space | 0 | Space |
| | Signed/Unsigned | Signed/Positive Sign Character (0) | Signed/Positive Sign Character (Space) | Signed/Positive Sign Character (+) |
| | Number of Decimals | No Decimal Point | 3 | Variable Point |
| | Delimiter | No Delimiter | Comma | Space |
| | Send data*2 | [Static data]0_ _ _120_ _ _ _80_ _3640_ _100[Static data] | [Static data]_012.346,_007.640,_364.000,_099.998[Static data] | [Static data]+12.34568_+_ _ _ _7.64_+_ _ _ _364_+99.99791[Static data] |

*1 When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Single Precision]", the maximum number of digits is 7 if "Digit" is "Variable Number of Digits".

*2 "_" indicates a one-byte space.

Ex.

The following table shows send data when a packet is formed in the structure of [Static data], [Conversion variable], [Static data], and the data stored in the data storage area is as follows: D0 = 28790 (7076H), D1 = 54063 (D32FH), D2 = 45308 (B0FCH), D3 = 16424 (4028H), D4 = 49812 (C294H), D5 = 10485 (28F5H), D6 = 36700 (8F5CH), D7 = 16414 (401EH), D8 = 0 (0000H), D9 = 0 (0000H), D10 = 49152 (C000H), D11 = 16502 (4076H), D12 = 21142 (5296H), D13 = 55872 (DA40H), D14 = 65535 (FFFFH), D15 = 16472 (4058H). (4028B0FCD32F7076H, 401E8F5C28F5C294H, 4076C00000000000H, and 4058FFFFDA405296H are 12.34567890123456, 7.640000000000004, 364, and 99.99999099999908 in double-precision real numbers.)

When numerical values do not fit in the decimal part after conversion due to the stored data and setting details in "Digit" and "Number of Decimals", the uppermost digit of the decimal part that does not fit is rounded off.

| Item | | Description*1 | | |
|---|---|---|---|---|
| Example 1 | Fixed Number of Data/Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 4 | 4 | 4 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | FLOAT [Double Precision] | FLOAT [Double Precision] | FLOAT [Double Precision] |
| | Conversion | HEX→ASCII Decimal | HEX→ASCII Decimal | HEX→ASCII Decimal |
| | Digit | Variable Number of Digits | Variable Number of Digits | 4 |
| | Blank-padded Character | — | — | 0 |
| | Signed/Unsigned | Unsigned | Signed/Positive Sign Character (None) | Signed/Positive Sign Character (+) |
| | Number of Decimals | Variable Point | 1 | Variable Point |
| | Delimiter | No Delimiter | Comma | Space |
| | Send data*2 | [Static data]12.34567890123467.6436499.9999909999991[Static data] | [Static data]12.3,7.6,364.0,100.0[Static data] | [Static data]+12.35_+07.64_+0364_+0100[Static data] |
| Example 2 | Fixed Number of Data/Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 4 | 4 | 4 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | FLOAT [Double Precision] | FLOAT [Double Precision] | FLOAT [Double Precision] |
| | Conversion | HEX→ASCII Decimal | HEX→ASCII Decimal | HEX→ASCII Decimal |
| | Digit | 8 | 9 | 15 |
| | Blank-padded Character | Space | 0 | Space |
| | Signed/Unsigned | Signed/Positive Sign Character (0) | Signed/Positive Sign Character (Space) | Signed/Positive Sign Character (+) |
| | Number of Decimals | No Decimal Point | 6 | Variable Point |
| | Delimiter | No Delimiter | Comma | Space |
| | Send data*2 | [Static data]0_ _ _ _ _ _120_ _ _ _ _ _80_ _ _ _ _3640_ _ _ _ _100[Static data] | [Static data]_012.345679,_007.640000,_364.000000,_099.999991[Static data] | [Static data]+12.3456789012346_+_ _ _ _ _ _ _ _ _ _7.64_+_ _ _ _ _ _ _ _ _ _364_+99.9999909999991[Static data] |

*1 When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Double Precision]", the maximum number of digits is 15 if "Digit" is "Variable Number of Digits".

*2 "_" indicates a one-byte space.

Ex.

The following table shows send data when a packet is formed in the structure of [Static data], [Conversion variable], [Static data], and data stored in the data storage area is as follows: D0 = 837 (0345H), D1 = 18 (0012H), D2 = 43913 (AB89H), D3 = 26607 (67EFH).

| Item | | Description | | |
|---|---|---|---|---|
| Example 1 | Fixed Number of Data/ Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 1 | 1 | 1 |
| | Send/Receive Data Storage Area | D0 | D0 | D0 |
| | Conversion Data Type | Word | Double Word | FLOAT [Single Precision] |
| | Conversion | HEX→ASCII Hexadecimal | HEX→ASCII Hexadecimal | HEX→ASCII Hexadecimal |
| | Digit | 3 | 7 | 8 |
| | Blank-padded Character | 0 | Space | 0 |
| | Signed/Unsigned | — | — | — |
| | Number of Decimals | — | — | — |
| | Delimiter | No Delimiter | Comma | Space |
| | Send data*1 | [Static data]345[Static data] | [Static data]_120345,[Static data] | [Static data]00120345_[Static data] |
| Example 2 | Fixed Number of Data/ Variable Number of Data | Fixed Number of Data | Fixed Number of Data | Fixed Number of Data |
| | Number of Data/Maximum Number of Data | 1 | 2 | 3 |
| | Send/Receive Data Storage Area | D0 | D0 | D1 |
| | Conversion Data Type | FLOAT [Double Precision] | Double Word | Word |
| | Conversion | HEX→ASCII Hexadecimal | HEX→ASCII Hexadecimal | HEX→ASCII Hexadecimal |
| | Digit | 16 | Variable Number of Digits | 4 |
| | Blank-padded Character | Space | — | 0 |
| | Signed/Unsigned | — | — | — |
| | Number of Decimals | — | — | — |
| | Delimiter | No Delimiter | Space | Comma |
| | Send data*1 | [Static data]67EFAB8900120345[Static data] | [Static data]120345_67EFAB89[Static data] | [Staticdata]0012,AB89,67EF[Static data] |

*1 "_" indicates a one-byte space.

## Precautions

### ■Error in a single-precision real number and a double-precision real number

When "Conversion Data Type" is "FLOAT [Single Precision]" or "FLOAT [Double Precision]", an error may occur during the "HEX→ASCII Decimal" conversion or "ASCII Decimal→HEX" conversion.

In this case, the conversion result may not match the conversion result of using the ESTR instruction or EVAL instruction of the CPU module or the value that can be monitored using the engineering tool. To obtain matching data, use "Word" or "Double Word".

### ■Restrictions by placement

Refer to the following.

☞Page 358 Restrictions on placement of variable length elements in a receive packet

**Point**

When a conversion variable is used in a TCP receive packet, if the conversion variable shown below is not included in the length calculation range and if no static data is placed in the subsequent elements, the system will continue to wait until 4096 bytes of response data are received with "Resource Setting" set to either "Normal" or "Extension 1". Or, the system will continue to wait until 10238 bytes of response data are received with "Resource Setting" set to "Extension 2".

- "Fixed Number of Data/Variable Number of "Data" is "Variable Number of Data".
- "Fixed Number of Data/Variable Number of "Data" is "Fixed Number of Data" and "Digit" is "Variable Number of Digits" (except when "Number of Data/Maximum Number of Data" is 1 and "Delimiter" is not "No Delimiter").



## ■Restrictions when sending

- When "Conversion Data Type" is "Word" or "Double Word", "Digit" is not "Variable Number of Digits", and "Number of Decimals" is "Variable Point", an error occurs if the decimal point position is larger than the numerical value data.
- When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Single Precision]" or "FLOAT [Double Precision]", an error occurs if data outside the range that can be handled in the send data storage area is set. The following shows setting examples where an error occurs.

| Conversion data type | Signed/ unsigned | Data stored in send data storage area | Error details |
|---|---|---|---|
| FLOAT [Single Precision] | Unsigned | • D0 = 34790 (87E6H)<br>• D1 = 49477 (C145H) | C14587E6H is -12.345678329467773 in a single-precision real number. |
| FLOAT [Single Precision] | Other than "Unsigned" | • D0 = 38528 (9680H)<br>• D1 = 19224 (4B18H) | 4B189680H is 10000000 in a single-precision real number. |
| FLOAT [Double Precision] | Unsigned | • D0 = 0 (0000H)<br>• D1 = 9780 (2634H)<br>• D2 = 27637 (6BF5H)<br>• D3 = 17164 (430CH) | 430C6BF526340000H is 1000000000000000 in a double-precision real number. |
| FLOAT [Double Precision] | Other than "Unsigned" | • D0 = 65530 (FFFAH)<br>• D1 = 9779 (2633H)<br>• D2 = 27637 (6BF5H)<br>• D3 = 49932 (C30CH) | C30C6BF52633FFFAH is -999999999999999.2 in a double-precision real number. |

- An error occurs if the integer part is omitted after conversion due to setting details in "Digit" and "Number of Decimals". The following shows setting examples where an error occurs.

| Conversion | Conversion data type | Number of digits | Number of decimal places | Data stored in send data storage area | Error details |
|---|---|---|---|---|---|
| HEX→ASCII Decimal | Word | "3" | — | D0 = 1234 (4D2H) | The integer part "1" is omitted as "1234" is converted into "234" due to the "Digit" setting value. |
| HEX→ASCII Decimal | Double Word | "6" | — | • D0 = 24910 (614EH)<br>• D1 = 188 (00BCH) | The integer part "12" is omitted as "12345678" is converted into "345678" due to the "Digit" setting value. |
| HEX→ASCII Decimal | FLOAT [Single Precision] | Variable Number of Digits | "6" | • D0 = 34790 (87E6H)<br>• D1 = 16709 (4145H) | 414587E6H is 12.345678329467773 in a single-precision real number. The integer part "1" is omitted as "12.345678329467773" is converted into "2.345678" when the seventh decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values.<br>(When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Single Precision]", the maximum number of digits is 7 if "Digit" is "Variable Number of Digits".) |
| HEX→ASCII Decimal | FLOAT [Single Precision] | "2" | Variable Point | • D0 = 59872 (E9E0H)<br>• D1 = 17142 (42F6H) | 42F6E9E0H is 123.456787109375 in a single-precision real number. The integer part "1" is omitted as "123.456787109375" is converted into "23" when the first decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |
| HEX→ASCII Decimal | FLOAT [Single Precision] | "3" | Variable Point | • D0 = 65534 (FFFEH)<br>• D1 = 17529 (4479H) | 4479FFFEH is 999.9998779296875 in a single-precision real number. The integer part "1" is omitted as "999.9998779296875" is converted into "000" when the first decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |

| Conversion | Conversion data type | Number of digits | Number of decimal places | Data stored in send data storage area | Error details |
|---|---|---|---|---|---|
| HEX→ASCII Decimal | FLOAT [Single Precision] | "5" | "4" | • D0 = 34790 (87E6H)<br>• D1 = 16709 (4145H) | 414587E6H is 12.345678329467773 in a single-precision real number. The integer part "1" is omitted as "12.345678329467773" is converted into "2.3457" when the fifth decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |
| HEX→ASCII Decimal | FLOAT [Single Precision] | "6" | "3" | • D0 = 65534 (FFFEH)<br>• D1 = 17529 (4479H) | 4479FFFEH is 999.9998779296875 in a single-precision real number. The integer part "1" is omitted as "999.9998779296875" is converted into "000.000" when the fourth decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |
| HEX→ASCII Decimal | FLOAT [Double Precision] | Variable Number of Digits | "14" | • D0 = 28794 (707AH)<br>• D1 = 54063 (D32FH)<br>• D2 = 45308 (B0FCH)<br>• D3 = 16424 (4028H) | 4028B0FCD32F707AH is 12.345678901234567 in a double-precision real number. The integer part "1" is omitted as "12.345678901234567" is converted into "2.34567890123457" when the 15th decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values.<br>(When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Double Precision]", the maximum number of digits is 15 if "Digit" is "Variable Number of Digits".) |
| HEX→ASCII Decimal | FLOAT [Double Precision] | Variable Number of Digits | "7" | • D0 = 65533 (FFFDH)<br>• D1 = 65535 (FFFFH)<br>• D2 = 55171 (D783H)<br>• D3 = 16791 (4197H) | 4197D783FFFFFFFDH is 99999999.99999996 in a double-precision real number. The integer part "1" is omitted as "99999999.99999996" is converted into "00000000.0000000" when the eighth decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values.<br>(When "Conversion" is "HEX→ASCII Decimal" and "Conversion Data Type" is "FLOAT [Double Precision]", the maximum number of digits is 15 if "Digit" is "Variable Number of Digits".) |
| HEX→ASCII Decimal | FLOAT [Double Precision] | "2" | Variable Point | • D0 = 19609 (4C99H)<br>• D1 = 2043 (07FBH)<br>• D2 = 56636 (DD3CH)<br>• D3 = 16478 (405EH) | 405EDD3C07FB4C99H is 123.45678901234568 in a double-precision real number. The integer part "1" is omitted as "123.45678901234568" is converted into "23" when the first decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |
| HEX→ASCII Decimal | FLOAT [Double Precision] | "8" | Variable Point | • D0 = 65533 (FFFDH)<br>• D1 = 65535 (FFFFH)<br>• D2 = 55171 (D783H)<br>• D3 = 16791 (4197H) | 4197D783FFFFFFFDH is 99999999.99999996 in a double-precision real number. The integer part "1" is omitted as "99999999.99999996" is converted into "00000000" when the first decimal place of send data is rounded off due to the "Digit" and "Variable Number of Digits" setting values. |
| HEX→ASCII Decimal | FLOAT [Double Precision] | "9" | "8" | • D0 = 28794 (707AH)<br>• D1 = 54063 (D32FH)<br>• D2 = 45308 (B0FCH)<br>• D3 = 16424 (4028H) | 4028B0FCD32F707AH is 12.345678901234567 in a double-precision real number. The integer part "1" is omitted as "12.345678901234567" is converted into "2.34567890" when the ninth decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |
| HEX→ASCII Decimal | FLOAT [Double Precision] | "14" | "6" | • D0 = 65533 (FFFDH)<br>• D1 = 65535 (FFFFH)<br>• D2 = 55171 (D783H)<br>• D3 = 16791 (4197H) | 4197D783FFFFFFFDH is 99999999.99999996 in a double-precision real number. The integer part "1" is omitted as "99999999.99999996" is converted into "00000000.000000" when the first decimal place of send data is rounded off due to the "Digit" and "Number of Decimals" setting values. |
| HEX→ASCII Hexadecimal | Word | "3" | — | D0 = 4660 (1234H) | The integer part "1" is omitted as "1234" is converted into "234" due to the "Digit" setting value. |
| HEX→ASCII Hexadecimal | Double Word | "5" | — | • D0 = 17767 (4567H)<br>• D1 = 291 (0123H) | The integer part "12" is omitted as "1234567" is converted into "34567" due to the "Digit" setting value. |

- When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data", an error occurs if the value stored in the data quantity storage area is larger than the value set in "Number of data/Maximum Number of Data".

## ■Restrictions when receiving

- When "Conversion" is "ASCII Hexadecimal→HEX", a verification mismatch occurs if a character string other than "0" to "9", "A" to "F", or "a" to "f" is received.
- When "Conversion" is "\ASCII Decimal→HEX", a verification mismatch occurs if a character string other than "0" to "9" is received. However, a verification mismatch does not occur in the following cases.

| Item | Operation |
|---|---|
| Signed/Unsigned | When an item other than "Unsigned" is set, signed characters can be received.<br>However, when a signed character is received at a position other than the start of a piece of data, a verification mismatch occurs. |
| Number of Decimals | When an item other than "No Decimal Point" is set, a period (.) can be received.<br>However, when a period (.) is received at a digit other than the set digit or when a period (.) is received at the start or end of a piece of data, a verification mismatch occurs. |
| Delimiter | When an item other than "No Delimiter" is set, a delimiter can be received.<br>However, when a delimiter is received at a position other than a delimiting point of data or when data not including a delimiter is received when an item other than "No Delimiter" is set, a verification mismatch occurs. |

- When "Digit" is "Variable Number of Digits", a verification mismatch occurs if the number of digits of received data exceeds the following upper limit.

| Conversion data type | Conversion | Upper limit of receive data quantity |
|---|---|---|
| Word | ASCII Decimal→HEX | 5 digits maximum |
| | ASCII Hexadecimal→HEX | 4 digits maximum |
| Double Word | ASCII Decimal→HEX | 10 digits maximum |
| | ASCII Hexadecimal→HEX | 8 digits maximum |
| FLOAT [Single Precision] | ASCII Decimal→HEX | 7 digits maximum |
| | ASCII Hexadecimal→HEX | Cannot be set. |
| FLOAT [Double Precision] | ASCII Decimal→HEX | 15 digits maximum |
| | ASCII Hexadecimal→HEX | Cannot be set. |

- When "Digit" is "Variable Number of Digits", if received data only contains a delimiter or blank character string, it is handled as if 0 was received and the corresponding receive data storage area is overwritten with 0.

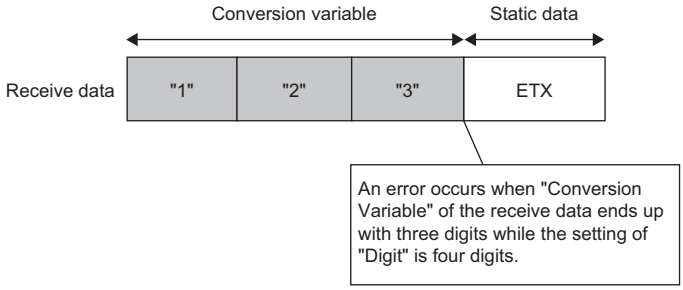
Ex.

When "Number of data/Maximum Number of Data" is 3 and received data is ",,", it is handled as if "0,0,0" was received.

- When "Fixed Number of Data/Variable Number of Data" is "Variable Number of Data", a verification mismatch may occur if data whose number of digits is less than the number of digits specified in "Digit" is received. However, when "Digit" is "Variable Number of Digits" and "Delimiter" is set, if received data contains parts where delimiters repeat (no data), it is handled as if 0 was received and the corresponding receive data storage area is overwritten with 0.

**Ex.**

When "Digit" is set to 4 digits



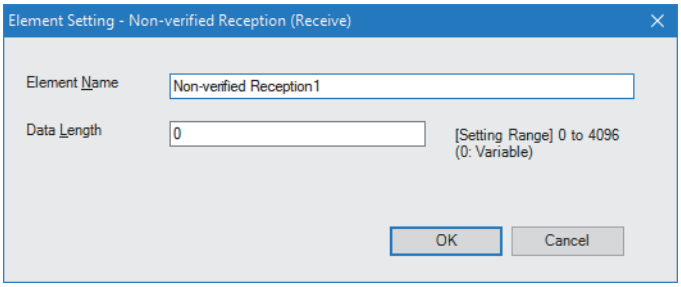

- When a conversion variable is a variable length element and a data quantity more than "Number of data/Maximum Number of Data" is received, only data that meets "Number of data/Maximum Number of Data" is stored and the remaining receive data is discarded. (The protocol is completed successfully. However, a verification mismatch occurs in the following cases.)

· "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data", "Digit" is "Variable Number of Data", and "Number of data/Maximum Number of Data" is 2 or more.

· "Fixed Number of Data/Variable Number of Data" is "Fixed Number of Data", "Digit" is "Variable Number of Data", "Number of data/Maximum Number of Data" is 1, and "Delimiter" is "No Delimiter".

- Among packet data received from an external device, data corresponding to a variable must be distinguished from static data immediately after a conversion variable. If it cannot be distinguished, receive processing may not be performed normally. (For example, if the value of static data immediately after a conversion variable is used for a conversion variable, verification and receive processing are performed while recognizing the value as static data right after a conversion variable.)
- When "Conversion Data Type" is "FLOAT [Single Precision]" or "FLOAT [Double Precision]", receiving data outside the range causes a verification mismatch.
- When "Number of Decimals" is 1 to 15, if the number of decimal places set in the parameter does not match the number of decimal places of received data, a verification mismatch occurs. (For example, when "Digit" is "Variable Number of Digits, "Number of Decimals" is 1, and received data is 1.23, a verification mismatch occurs as the number of decimal places set in the parameter is 1 while the received data has two decimal digits.)

## ■Element type (Non-verified reception)




Use this when receive data includes unnecessary data.
If the receive packet contains non-verified reception, Ethernet-equipped module skims over the specified number of characters.
Up to 32 non-verified reception elements can be placed in one packet when "Resource Setting" is set to either "Normal" or "Extension 1", and up to 128 non-verified reception elements can be placed in one packet when "Resource Setting" is set to "Extension 2".

| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Element Name | Set the element name. | | 48 characters<br>(Default: Non-verified Reception + Element No.) | — |
| Data Length | 0 (variable number of characters) | Set when the number of characters that are not verified differs between each communication session. | • 0 to 4096<br>• 0 to 10238<br>(Default: 0) | (Variable) |
| | • 1 to 4096<br>• 1 to 10238<br>(Specified number of characters) | Set the number of characters that are not verified.*1 | | (4096B) |

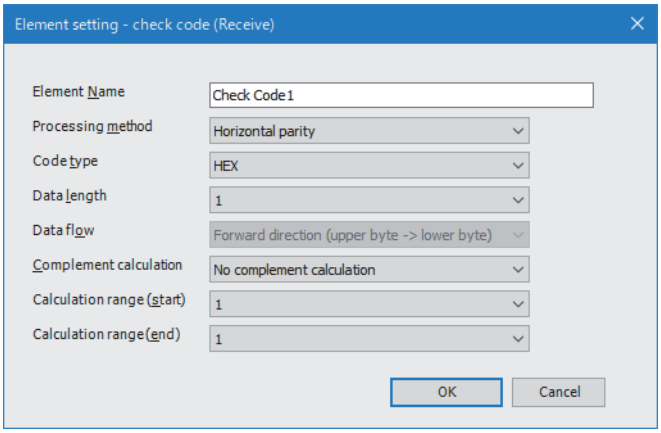
*1 The number will be 1 to 4096 when "Resource Setting" is set for either "Normal" or "Extension 1", and 1 to 10238 for "Extension 2".

**Restriction**

When setting non-verified reception (variable number of characters), one of the following conditions must be met.

- Place the static data after the non-verified reception. However, placing fixed-length elements between non-verified reception and static data is not a problem.
- Set the length before non-verified reception (ensuring that the calculation range includes the non-verified reception). However, when placing multiple non-verified receptions within the calculation range of one length, the above condition must be satisfied.
- Place the non-verified receptions at the end of the packet elements. However, placing fixed-length elements after non-verified reception is not a problem.

## ■Element type (Error check code)




Use this when there is a packet element that indicates the error check code data.
This element automatically calculates the error check code specified during transmission and reception, adds the code to the send packet, and detects errors in the receive packet.

| Item | Description | | Setting range | Display example in "Element Setting" |
|---|---|---|---|---|
| Element Name | Set the element name. | | 48 characters<br>(Default: Element Type + Element No.) | — |
| Processing method | Select the processing method. | | • Horizontal parity<br>• Sum check<br>• 16-bit CRC (MODBUS specification)<br>(Default: Horizontal parity) | • Prty<br>• Sum<br>• C(M) |
| Code type | Select the error check code format.<br>This item cannot be set when the processing method is "16-bit CRC (MODBUS specification)". | | • ASCII HEX<br>• ASCII DEC<br>• HEX<br>(Default: HEX) | • Hexadecimal<br>• Decimal<br>• HEX |
| Data length | Select the data length on the line.<br>This item cannot be set when the processing method is "16-bit CRC (MODBUS specification)". | | 1 to 4<br>(Default: 1) | 1B |
| Data flow*1 | Forward direction (upper byte → lower byte) | • When sending: The calculated error check code is sent in order from the upper byte.<br>• When receiving: The data is treated as error check code in order from the upper byte. | • Forward direction (upper byte → lower byte)<br>• Reverse direction (lower byte → upper byte)<br>• Byte swap (by word)<br>(Default: -) | Forward |
| | Reverse direction (lower byte → upper byte) | • When sending: The calculated error check code is sent in order from the lower byte.<br>• When receiving: The data is treated as error check code in order from the lower byte. | | Reverse |
| | Byte swap (by word) | • When sending: The bytes are swapped in word units and the calculated error check code is sent.<br>• When receiving: The bytes are swapped in word units and the data is treated as error check code. | | Byte |
| Complement calculation | Select the complement calculation.<br>This item cannot be set when the processing method is "16-bit CRC (MODBUS specification)". | | • No complement calculation<br>• One's complement calculation<br>• Two's complement calculation<br>(Default: No complement calculation) | • NoC<br>• 1C<br>• 2C |
| Calculation Range (Start)*2 | Select the start packet element number for the range to be calculated. | | • 1 to 31<br>• 1 to 127<br>(Default: 1) | 1-1 |
| Calculation Range (End)*2 | Select the end packet element number for the range to be calculated. | | | |

*1 This cannot be set if "Data length" is 1 byte. "Byte swap (by word)" can be selected only when "Data length" is set to 4 bytes. This cannot be set when "Processing method" is 16-bit CRC (MODBUS specification).
*2 The number will be 1 to 31 when "Resource Setting" is set for either "Normal" or "Extension 1", and 1 to 127 for "Extension 2".

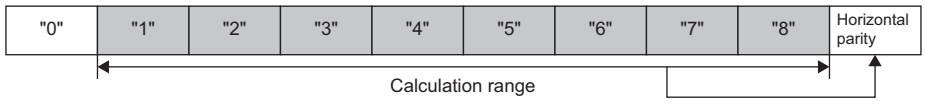
• Calculation of error check code

The following shows procedures for calculating the error check code.

Ex.

Calculation of horizontal parity

The following shows how to calculate horizontal parity using the following data as an example.




"1" (31H) 0011 0001
XOR
"2" (32H) 0011 0010 = 0000 0011
XOR
"3" (33H) 0011 0011 = 0011 0000
XOR
"4" (34H) 0011 0100 = 0000 0100
XOR
"5" (35H) 0011 0101 = 0011 0001
XOR
"6" (36H) 0011 0110 = 0000 0111
XOR
"7" (37H) 0011 0111 = 0011 0000
XOR
"8" (38H) 0011 1000 = **0000 1000** (Binary)

Horizontal parity = **0 8** (Hexadecimal)

• Data flow: Forward direction, "No complement calculation" specified (08H in hexadecimal is 8 in decimal)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | "8"<br>(38H) | "08"<br>(30H 38H) | "008"<br>(30H 30H 38H) | "0008"<br>(30H 30H 30H 38H) |
| ASCII DEC | "8"<br>(38H) | "08"<br>(30H 38H) | "008"<br>(30H 30H 38H) | "0008"<br>(30H 30H 30H 38H) |
| HEX | 08H | 0008H | 000008H | 00000008H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Forward direction, "One's complement calculation" specified (One's complement of 0000 0008H is FFFF FFF7H.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FFF7H in hexadecimal is 65527 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | "7"<br>(37H) | "F7"<br>(46H 37H) | "FF7"<br>(46H 46H 37H) | "FFF7"<br>(46H 46H 46H 37H) |
| ASCII DEC | "7"<br>(37H) | "27"<br>(32H 37H) | "527"<br>(35H 32H 37H) | "5527"<br>(35H 35H 32H 37H) |
| HEX | F7H | FFF7H | FFFFF7H | FFFFFFF7H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Forward direction, "Two's complement calculation" specified (Two's complement of 0000 0008H is FFFF FFF8H.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FFF8H in hexadecimal is 65528 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | "8"<br>(38H) | "F8"<br>(46H 38H) | "FF8"<br>(46H 46H 38H) | "FFF8" (46H 46H 46H 38H) |
| ASCII DEC | "8"<br>(38H) | "28" (32H 38H) | "528" (35H 32H 38H) | "5528" (35H 35H 32H 38H) |
| HEX | F8H | FFF8H | FFFFF8H | FFFFFFF8H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Reverse direction, "No complement calculation" specified (08H in hexadecimal is 8 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | "80"<br>(38H 30H) | "800"<br>(38H 30H 30H) | "8000"<br>(38H 30H 30H 30H) |
| ASCII DEC | — | "80"<br>(38H 30H) | "800"<br>(38H 30H 30H) | "8000"<br>(38H 30H 30H 30H) |
| HEX | — | 0800H | 080000H | 08000000H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Reverse direction, "One's complement calculation" specified (One's complement of 0000 0008H is FFFF FFF7H.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FFF7H in hexadecimal is 65527 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | "7F"<br>(37H 46H) | "7FF"<br>(37H 46H 46H) | "7FFF"<br>(37H 46H 46H 46H) |
| ASCII DEC | — | "72"<br>(37H 32H) | "725"<br>(37H 32H 35H) | "7255"<br>(37H 32H 35H 35H) |
| HEX | — | F7FFH | F7FFFFH | F7FFFFFFH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Reverse direction, "Two's complement calculation" specified (Two's complement of 0000 0008H is FFFF FFF8H.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FFF8H in hexadecimal is 65528 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | "8F"<br>(38H 46H) | "8FF"<br>(38H 46H 46H) | "8FFF"<br>(38H 46H 46H 46H) |
| ASCII DEC | — | "82"<br>(38H 32H) | "825"<br>(38H 32H 35H) | "8255"<br>(38H 32H 35H 35H) |
| HEX | — | F8FFH | F8FFFFH | F8FFFFFFH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Byte swap, "No complement calculation" specified (08H in hexadecimal is 8 in decimal)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | — | — | "0800"<br>(30H 38H 30H 30H) |
| ASCII DEC | — | — | — | "0800"<br>(30H 38H 30H 30H) |
| HEX | — | — | — | 00080000H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Byte swap, "One's complement calculation" specified (One's complement of 0000 0008H is FFFF FFF7H.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FFF7H in hexadecimal is 65527 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | — | — | "F7FF"<br>(46H 37H 46H 46H) |
| ASCII DEC | — | — | — | "2755"<br>(32H 37H 35H 35H) |
| HEX | — | — | — | FFF7FFFFH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Byte Swap, "Two's complement calculation" specified (Two's complement of 0000 0008H is FFFF FFF8H.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FFF8H in hexadecimal is 65528 in decimal.)

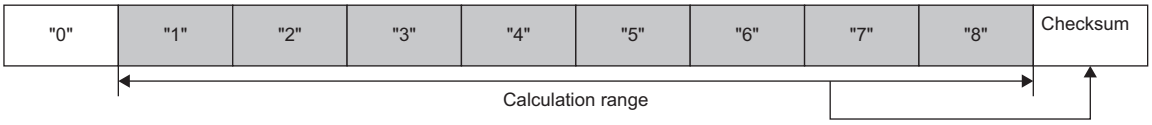
| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | — | — | "F8FF"<br>(46H 38H 46H 46H) |
| ASCII DEC | — | — | — | "2855"<br>(32H 38H 35H 35H) |
| HEX | — | — | — | FFF8FFFFH |

*1 The values in parentheses are the ASCII codes.

**Ex.**

Calculation of sum check

The following shows how to calculate sum check codes using the following data as an example.




(For above packet)
Checksum value = 31H + 32H + 33H + 34H + 35H + 36H + 37H + 38H = 1A4H

• Data flow: Forward direction, "No complement calculation" specified (1A4H in hexadecimal is 420 in decimal.)

| Code Type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | "4"<br>(34H) | "A4"<br>(41H 34H) | "1A4"<br>(31H 41H 34H) | "01A4"<br>(30H 31H 41H 34H) |
| ASCII DEC | "0"<br>(30H) | "20"<br>(32H 30H) | "420"<br>(34H 32H 30H) | "0420"<br>(30H 34H 32H 30H) |
| HEX | A4H | 01A4H | 0001A4H | 000001A4H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Forward direction, "One's complement calculation" specified (One's complement of 0000 01A4H is FFFF FE5BH.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FE5BH in hexadecimal is 65115 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | "B"<br>(42H) | "5B"<br>(35H 42H) | "E5B"<br>(45H 35H 42H) | "FE5B"<br>(46H 45H 35H 42H) |
| ASCII DEC | "5"<br>(35H) | "15"<br>(31H 35H) | "115"<br>(31H 31H 35H) | "5115"<br>(35H 31H 31H 35H) |
| HEX | 5BH | FE5BH | FFFE5BH | FFFFFE5BH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Forward direction, "Two's complement calculation" specified (Two's complement of 0000 01A4H is FFFF FE5CH.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FE5CH in hexadecimal is 65116 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | "C"<br>(43H) | "5C"<br>(35H 43H) | "E5C" (45H 35H 43H) | "FE5C"<br>(46H 45H 35H 43H) |
| ASCII DEC | "6"<br>(36H) | "16"<br>(31H 36H) | "116"<br>(31H 31H 36H) | "5116"<br>(35H 31H 31H 36H) |
| HEX | 5CH | FE5CH | FFFE5CH | FFFFFE5CH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Reverse direction, "No complement calculation" specified (1A4H in hexadecimal is 420 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | "4A"<br>(34H 41H) | "4A1"<br>(34H 41H 31H) | "4A10"<br>(34H 41H 31H 30H) |
| ASCII DEC | — | "02"<br>(30H 32H) | "024"<br>(30H 32H 34H) | "0240"<br>(30H 32H 34H 30H) |
| HEX | — | A401H | A40100H | A4010000H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Reverse direction, "One's complement calculation" specified (One's complement of 0000 01A4H is FFFF FE5BH.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FE5BH in hexadecimal is 65115 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | "B5"<br>(42H 35H) | "B5E"<br>(42H 35H 45H) | "B5EF"<br>(42H 35H 45H 46H) |
| ASCII DEC | — | "51"<br>(35H 31H) | "511"<br>(35H 31H 31H) | "5115"<br>(35H 31H 31H 35H) |
| HEX | — | 5BFEH | 5BFEFFH | 5BFEFFFFH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Reverse direction, "Two's complement calculation" specified (Two's complement of 0000 01A4H is FFFF FE5CH.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FE5CH in hexadecimal is 65116 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | "C5"<br>(43H 35H) | "C5E"<br>(43H 35H 45H) | "C5EF"<br>(43H 35H 45H 46H) |
| ASCII DEC | — | "61"<br>(36H 31H) | "611"<br>(36H 31H 31H) | "6115"<br>(36H 31H 31H 35H) |
| HEX | — | 5CFEH | 5CFEFFH | 5CFEFFFFH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Byte swap, "No complement calculation" specified (1A4H in hexadecimal is 420 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | — | — | "A401"<br>(41H 34H 30H 31H) |
| ASCII DEC | — | — | — | "2004"<br>(32H 30H 30H 34H) |
| HEX | — | — | — | 01A40000H |

*1 The values in parentheses are the ASCII codes.

• Data flow: Byte Swap, "One's complement calculation" specified (One's complement of 0000 01A4H is FFFF FE5BH.)

When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FE5BH in hexadecimal is 65115 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | — | — | "5BFE"<br>(35H 42H 47H 46H) |
| ASCII DEC | — | — | — | "1551"<br>(31F 35H 35H 31H) |
| HEX | — | — | — | FE5BFFFFH |

*1 The values in parentheses are the ASCII codes.

• Data flow: Byte Swap, "Two's complement calculation" specified (Two's complement of 0000 01A4H is FFFF FE5CH.)
When "Code type" is "ASCII DEC", the last one word is extracted and converted from hexadecimal to decimal. (FE5CH in hexadecimal is 65116 in decimal.)

| Code type | Data length*1 | | | |
|---|---|---|---|---|
| | 1 byte | 2 bytes | 3 bytes | 4 bytes |
| ASCII HEX | — | — | — | "5CFE"<br>(35H 43H 47H 46H) |
| ASCII DEC | — | — | — | "1651"<br>(31F 36H 35H 31H) |
| HEX | — | — | — | FE5CFFFFH |

*1 The values in parentheses are the ASCII codes.

**Ex.**

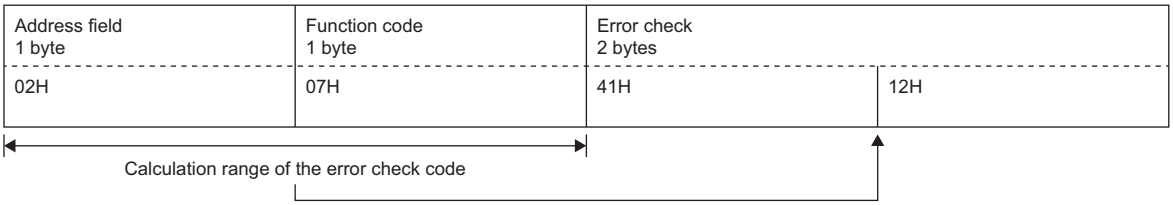
16-bit CRC (MODBUS specification) calculation procedure

This error check method is used when data are sent/received with the RTU mode of the MODBUS protocol. The data length of CRC is fixed at 2 bytes (16 bits), and the CRC is calculated every 1 byte (8 bits) from the start of the calculation range according to the following procedure.

1. Load a 16-bit register whose bits are all "1".
2. Calculate the exclusive OR (XOR) of the first 1 byte (8 bits) in the calculation range and the bits in Step 1 above.
3. Shift the result of Step 2 to the right by 1 bit.
4. If the least significant bit of Step 2 is "1", calculate the exclusive OR (XOR) of the result of Step 3 and the generator polynomial (A001H). If the least significant bit is "0", shift the result to the right by 1 bit without calculating the exclusive OR (XOR).
5. Repeat Steps 3 and 4 eight times.
6. Calculate the exclusive OR (XOR) of the result of Step 5 and the next 1 byte (8 bits).
7. Repeat Steps 3 through 6 until all bytes in the calculation range have been processed. The resulting value is the calculated CRC value.
8. The CRC value is stored in a packet in the order of lower 8 bits to upper 8 bits.

For specific calculation examples, refer to the following.

A calculation example for adding an error check (16-bit CRC (MODBUS specification)) to a MODBUS RTU mode frame is shown below.




The calculation process is shown below.

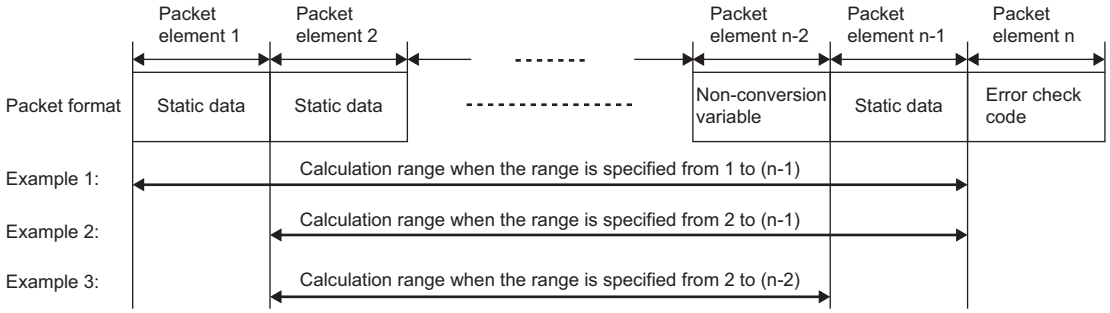
| CRC error check procedure | 16-bit register (MSB) | | | Flag*1 |
|---|---|---|---|---|
| 1 | Load a 16-bit register whose bits are all "1". | 1111 1111 | 1111 1111 | — |
| 2 | Address field (02H) | — | 0000 0010 | |
| | Exclusive OR (XOR) | 1111 1111 | 1111 1101 | |
| 3 | Shift 1 | 0111 1111 | 1111 1110 | 1 |
| 4 | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1101 1111 | 1111 1111 | |
| 5 | Shift 2 | 0110 1111 | 1111 1111 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1100 1111 | 1111 1110 | |
| | Shift 3 | 0110 0111 | 1111 1111 | 0 |
| | Shift 4 | 0011 0011 | 1111 1111 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1001 0011 | 1111 1110 | |
| | Shift 5 | 0100 1001 | 1111 1111 | 0 |
| | Shift 6 | 0010 0100 | 1111 1111 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1000 0100 | 1111 1110 | |
| | Shift 7 | 0100 0010 | 0111 1111 | 0 |
| | Shift 8 | 0010 0001 | 0011 1111 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1000 0001 | 0011 1110 | |
| 6 | Function code (07H) | — | 0000 0111 | — |
| | Exclusive OR (XOR) | 1000 0001 | 0011 1001 | |
| 7 | Shift 1 | 0100 0000 | 1001 1100 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1110 0000 | 1001 1101 | |
| | Shift 2 | 0111 0000 | 0100 1110 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1101 0000 | 0100 1111 | |
| | Shift 3 | 0110 1000 | 0010 0111 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1100 1000 | 0010 1110 | |
| | Shift 4 | 0110 0100 | 0001 0011 | 0 |
| | Shift 5 | 0011 0010 | 0000 1001 | 1 |
| | Generator polynomial (A001H) | 1010 0000 | 0000 0001 | |
| | Exclusive OR (XOR) | 1001 0010 | 0000 1000 | |
| | Shift 6 | 0100 1001 | 0000 0100 | 0 |
| | Shift 7 | 0010 0100 | 1000 0010 | 0 |
| | Shift 8 | 0001 0010 | 0100 0001 | 0 |
| 8 | CRC value | 12H | 41H | — |

*1 Least significant bit of previous exclusive OR (XOR)

Ex.

Calculation range of error check code

The following shows an example of specifying the calculation range of the error check code.

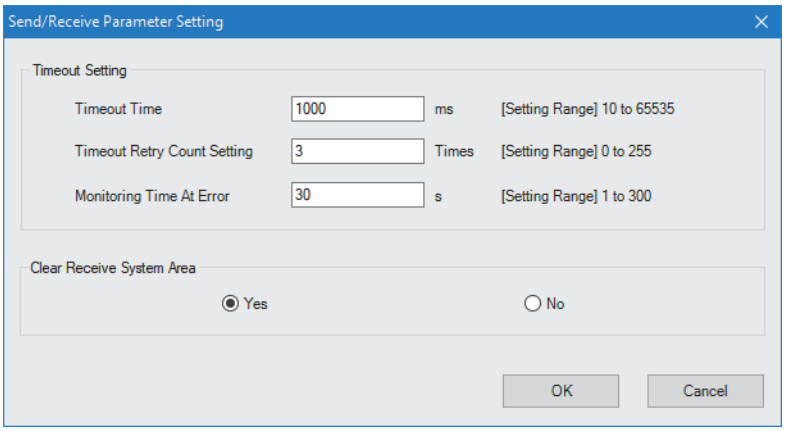



**Restriction**

- When the "Code type" setting is "ASCII HEX", a verification mismatch is judged to have occurred if a character string other than "0" to "9", "A" to "F", or "a" to "f" is received.
- When the "Code type" setting is "ASCII DEC", a verification mismatch is judged to have occurred if a character string other than "0" to "9" is received.
- When converting to ASCII characters during send, Ethernet-equipped modules use "0" to "9" and "A" to "F".
- Only one "Error check code" can be set in one packet.
- Packet elements after "Error check code" cannot be included in the calculation range.
- "Error check code" cannot be included in the calculation range.
- Set at least one element before "Error check code".
- If the calculated "Error check code" and the received "Error check code" do not match, a verification mismatch is judged to have occurred.

## "Send/Receive Parameter Setting" window

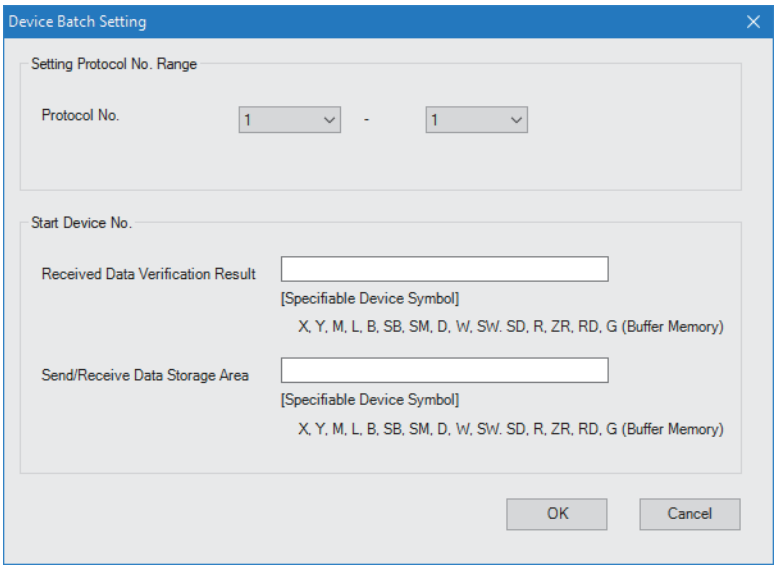
The data area at reception and the timeout are set for each protocol.




| Item | Description | Setting range |
|---|---|---|
| Timeout Time | Set the time to wait for the response from the communication destination.<br>Set "Timeout Time" that equals to or is longer than the execution interval.<br>0 (infinite wait) can be set only when "Comm Set" is set to "Request" and "Communication Type" is "Receive Only". | • 10 to 65535 (to be specified in increments of 1ms)<br>• 0 (infinite wait)<br>(Default: 1000) |
| Timeout Retry Count Setting | Set the number of retries to wait the communication time-out period when the communication destination does not respond. | 0 to 255<br>(Default: 3) |
| Monitoring Time At Error | Set the time to monitor when a communication error occurs.<br>Set "Monitoring Time At Error" that is longer than "Timeout Time".<br>This item cannot be set when "Comm Set" is "Request". | 1 to 300 (to be specified in increments of 1s)<br>(Default: 30) |
| Clear Receive System Area | Select whether to clear the system area (receive data area) of the module before execution of the protocol.<br>When "No" is selected, data received before the module executes the protocol is also included in the reception target for the protocol.<br>This item cannot be set if "Communication Type" is "Receive Only". | • Yes<br>• No<br>(Default: Yes) |

## "Device Batch Setting" window

Select the protocol range and setting items to be set to configure the settings for desired devices by batch.

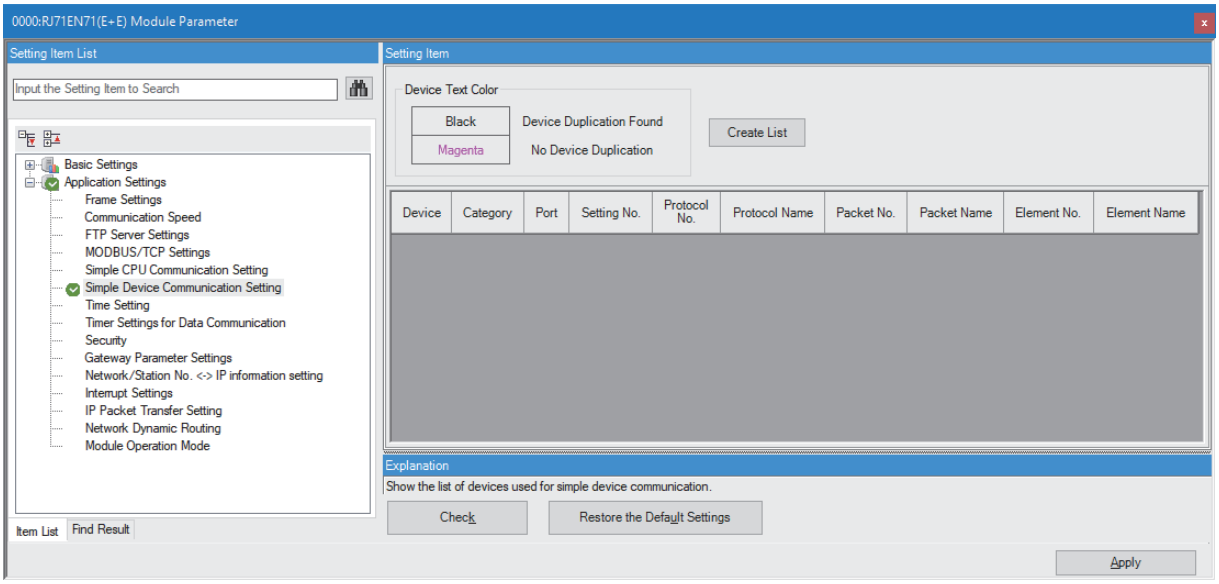



| Item | Description | Setting range |
|---|---|---|
| Setting Protocol No. Range*1 | Select the start number and end number of the protocols whose settings are configured by batch. | • 1 to 256<br>• 1 to 512<br>(Default: 1) |
| Start device No. | Set the start number of devices whose settings are configured by batch.<br>Items for which the device is set will be set by batch, and items with blank will not be set by batch. | Refer to devices that can be specified on the following page.<br>☞ Page 262 Devices that can be specified<br>(Default: Blank) |

*1 The number will be 1 to 256 when "Resource Setting" is set for "Normal", and 1 to 512 for either "Extension 1" or "Extension 2".

## "Device List" window

Displays devices that are being used for the simple device communications to check if any duplicate devices exist.
When a blank is shown, click the [Create List] button.




| Item | Description |
|---|---|
| [Create List] button | Creates the list of set devices. |
| Device | Displays devices that are being used for the simple device communications. |
| Category | Displays the classification for which "Device" is set. |
| Port | Displays the Ethernet port for which "Device" is set. |
| Setting No. | Displays the setting number of the protocol for which "Device" is set. |
| Protocol No. | Displays the protocol number of the protocol for which "Device" is set. |
| Protocol Name | Displays the protocol name of the protocol for which "Device" is set. |
| Packet No. | Displays the packet number of the packet for which "Device" is set.<br>This item is not displayed for devices whose "Category" is not set to "Variable".<br>• Send: Indicates the send packets.<br>• Receive(n): Indicates the receive packets. (n: Receive packet number (1 to 16)) |
| Packet Name | Displays the packet name of the packet for which "Device" is set.<br>This item is not displayed for devices whose "Category" is not set to "Variable". |
| Element No. | Displays the element number of the element for which "Device" is set.<br>This item is not displayed for devices whose "Category" is not set to "Variable". |
| Element Name | Displays the element name of the element for which "Device" is set.<br>This item is not displayed for devices whose "Category" is not set to "Variable". |

**Point**

Enable the float display on the "Device List" window to edit the program while referring to the list of set devices.

### Precautions

The device list is cleared if the following operations are performed.

- When "Resource Setting" is changed
- When any cell of "Communication Destination Setting" is edited
- When the [OK] button is clicked on the "Protocol Setting" window
- When any parameter of other ports is set and the [Apply] button is clicked (RJ71EN71(E+E) only)

# Time Setting

Set the time setting function (SNTP client).

| Item | | Description | Setting range |
|---|---|---|---|
| Time Setting (SNTP Client) | | Select whether to use the time setting function (SNTP client). | • Not Use<br>• Use<br>(Default: Not Use) |
| SNTP Server IP Address | | Set the IP address of the SNTP server. | 0.0.0.1 to 223.255.255.254<br>(Default: 0.0.0.1) |
| Timer Setting After Power-on and Reset | | Select whether to execute the time setting function upon power-on or reset. | • Disable<br>• Enable<br>(Default: Disable) |
| Execution Timing | — | Select the execution timing of the time setting. | • Fixed Time<br>• Fixed Scan Interval<br>(Default: Fixed Time) |
| | Time Intervals | When "Fixed Scan Interval" is selected, set the time interval (minute) of time setting executions. | 1 to 1440<br>(Default: 1 Minute) |
| | Specified Time (Hour, Minute, Day of Week) | When "Fixed Time" is selected, set the day of the week and the clock time (hour and minute) when the time setting function is executed.<br>(☞ Page 400 Specified Time (Hour, Minute, Day of Week)) | — |

**Point**

The SNTP server must be only one in a network. The time to be output is the same even though multiple modules in the same system retrieve time from the same SNTP server.

## Execution Timing

### ■Specified Time (Hour, Minute, Day of Week)

Set an execution time value within the following range in "Clock Time (Hour, Minute)".

| Item | Setting range |
|---|---|
| Hour | 0 to 23<br>(Default: 12) |
| Minute | 0 to 59<br>(Default: 0) |

To specify the day of the week when time setting is executed, set "Not Set" in the day when time setting is not executed under "Day of Week". (Time setting is set to be executed every day (all the days are set to "Set") by default.)

When specifying the day of the week, set at least one day of the week to "Set". An error occurs when all the days are set to "Not Set".

# Timer Settings for Data Communication

Set the timer used for the following communications.

- Connection with MELSOFT products and a GOT
- Communications using the SLMP
- Communications using the predefined protocol
- Socket communications/Communications using the fixed buffer
- Communication using the random access buffer
- File transfer function (FTP server)
- File Transfer Function (FTP Client)

When changing the timer settings for data communication, refer to the precautions before setting. (☞ Page 402 Precautions for settings)

| Item | | Description | Setting range |
|---|---|---|---|
| Change/Set Timer Value | | Select whether to change timer values from the default.<br>The timer operates with its default value when "No" is selected. | • No<br>• Yes<br>(Default: No) |
| TCP Resend Timer | | Set the waiting time to resend ACK if it is not returned when a connection is opened or data is sent in TCP/IP.<br>This timer is also used as the minimum set time for arrival monitoring for data link instructions. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 10s) |
| Destination Alive Check Start Interval Timer | | Set the time interval between the reception of the last message from the external device and the start of alive check. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 600s) |
| Destination Alive Check Interval Timer | | Set the time interval for performing alive check again when no response is returned from the external device of alive check target. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 10s) |
| Destination Alive Check Resend Count | | Set the number of times to perform alive check when no response is returned from the external device of alive check target. | 1 to 99999<br>(Default: 3) |
| Advanced Settings | Response Monitoring Timer | Set the following time.<br>• The time to wait for a response after sending command in communications with a fixed buffer (procedure exists)<br>• The time interval between the first message and last message when receiving the divided messages. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 30s) |
| | TCP ULP Timer | Set the time-to-live of the send packet in TCP/IP communications.<br>For example, when the TCP resend timer value is set to 10 seconds and the TCP ULP timer value is set to 30 seconds, data will be resend every 10 seconds if no response is returned from the external device after data sending, and timeout error occurs if no response is returned within 30 seconds. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 30s) |
| | TCP End Timer | When closing the TCP/IP connection by the own station, set the monitoring time for waiting for a FIN from the external device after the own station sends a FIN and an ACK is returned from the external device. If a FIN is not received from the external device within the time specified by the TCP end timer value, an RST is sent to the external device and the connection is forcibly closed. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 20s) |
| | TCP Zero Window Timer | The window means the receive buffer on the receive side. If the receive buffer on the receive side has no free space (window size = 0), data sending is waited until the receive side has free receive buffer space. At this time, the sending side sends packets for checking the send window to the receive side according to the TCP zero window timer value to check the receiving status. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 10s) |
| | IP Assembly Timer | The communication data may be divided at the IP level before being sent due to the buffer limitation of the sending station or the receiving station. Set the waiting time for the divided data in such a case. | Unit [s]: 1 to 16383<br>Unit [ms]: 100 to 16383000*1<br>(Default: 5s) |

*1 Set in increments of 100ms.

## Precautions for settings

### ■Setting value for each timer of the Ethernet-equipped module side.

Set the timer values so that the following formula is met.

(Response monitoring timer value) ≥ (TCP ULP timer value) ≥ (TCP end timer value) ≥ (TCP resend timer value) > (IP assembly timer value)

(TCP resend timer value) = (TCP zero window timer value)

When connecting Mitsubishi products to the line, configure the same settings for both modules.

### ■Setting value for each timer of the external device side.

Set the timer values so that the following formula is met. The frequency of a communication error, such as a transmission timeout, may be higher if the timer values do not meet the formula.

(TCP ULP timer value on the external device) > (TCP resend timer value on the Ethernet-equipped module)

(Monitoring timer value of the application software on the external device) > {(TCP ULP timer value on the Ethernet-equipped module) × n*1}

*1 "n" is the number of TCP segment transmission and is calculated by the following formula.

$$n = \text{A value that } \left( \frac{\text{Message size sent by the Ethernet-equipped module}}{\text{Maximum Segment size}} \right) \text{ is rounded up to the nearest integer}$$

**Ex.**

The number of TCP segment transmission when communications are performed on the same line

The Maximum Segment Size is 1460 bytes on the same line (without a router) and the number of TCP segment transmissions is as follows.

- n = 1 when the size of the message sent by the Ethernet-equipped module is 1460 bytes or less
- n = 2 when the size of the message sent by the Ethernet-equipped module is greater than 1460 bytes

**Ex.**

The number of TCP segment transmission when communications are performed on a different line

The Maximum Segment Size is at least 536 bytes on another line (via a dial-up router or other communication device) and the number of TCP segment transmissions is as follows.

- n = 1 when the size of the message sent by the Ethernet-equipped module is 536 bytes or less
- n = 2 when the size of the message sent by the Ethernet-equipped module is greater than 536 bytes and no more than 1072 bytes
- n = 3 when the size of the message sent by the Ethernet-equipped module is greater than 1072 bytes and no more than 1608 bytes

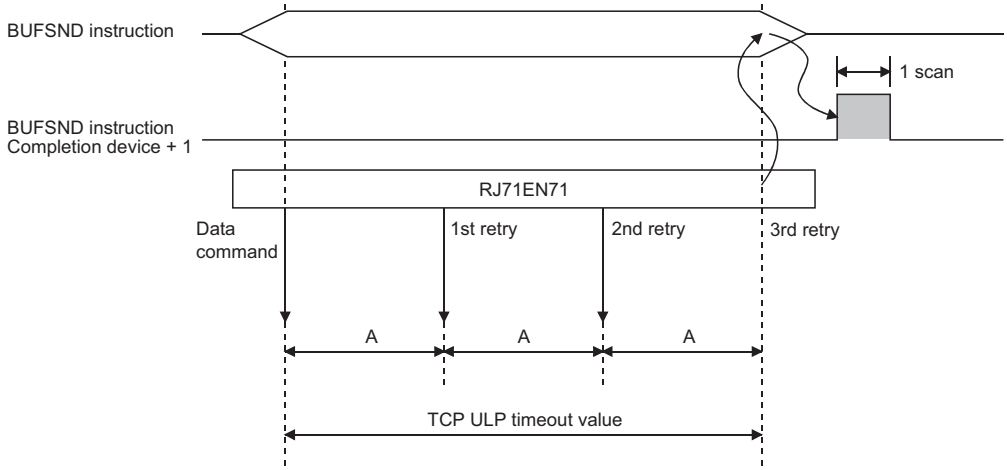
## ■Number of retries

When a communication failure occurs due to a problem such as noise, change the value so that the number of retries may increase. The number of retries is obtained using the following formula. (When using the default values, 3 = 30 ÷ 10)

Number of retries = TCP ULP timer value ÷ TCP resend timer value

**Ex.**

When the number of retries is three and data sending fails, a data sending error occurs at the timing shown in the figure below (in communications using the fixed buffer).




A: TCP resend timer value (The time at which data are sent when ACK is not returned after data transmission)

When not performing the above retry process (when setting 0 time), configure the setting so that the following formula is met.

TCP ULP timer value = TCP end timer value = TCP resend timer value

(Set the same value for the timer values.)

# Security

Set the security measures for access to the Ethernet-equipped module.

| Item | | Description | Setting range |
|---|---|---|---|
| IP Filter Settings | IP Filter | Set whether to enable the IP filter. | • Not Use<br>• Use<br>(Default: Not Use) |
| | IP Filter Settings | Set the IP address to be allowed or denied. (☞ Page 404 IP Filter Settings) | — |
| Disable Direct Connection with MELSOFT | | Set whether enable or disable direct connection to the engineering tool. | • Disable<br>• Enable<br>(Default: Enable) |
| Do Not Respond to CPU Module Search | | Select whether to respond to the CPU module search. | • Do Not Respond<br>• Respond<br>(Default: Respond) |

## IP Filter Settings

Up to 32 IP addresses can be set as an IP address to be allowed or denied by the IP filter.

Range specification and specification of the IP addresses to be excluded from the set range are also possible.

| Item | Description | Setting range |
|---|---|---|
| Access from IP address below | Select whether to allow or deny the access from the specified IP addresses. | • Allow<br>• Deny<br>(Default: Allow) |
| Range Setting | Select this item when specifying the IP addresses by range. | (Default: Clear) |
| IP Address | Set the IP address to be allowed or denied.<br>When selecting "Range Setting", enter the start IP address (left field) and end IP address (right field) of the range. | 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| IP Address Excluded from Range | When selecting "Range Setting", set the IP address to be excluded from the set range.<br>Up to 32 IP addresses can be set. | 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |

# Gateway Parameter Settings

With gateway parameter settings, the Ethernet-equipped module can communicate with external devices on other Ethernet networks via a router and gateway. One default router and up to eight routers can be set.

| Item | | Description | Setting range |
|---|---|---|---|
| Gateway Other Than Default Gateway | | Set to communicate with an external device on the other Ethernet via a router. | • Use<br>• Not Use<br>(Default: Not Use) |
| Gateway Information | No.1 to No.8 | Set the information of the gateway other than the default gateway. (☞ Page 405 Gateway Information) | — |

**Point**

Set the default gateway when communicating via the default gateway. (☞ Page 327 Own Node Settings)

## Gateway Information

### ■Gateway IP Address

When communicating with an external device on another Ethernet network through a gateway other than the default gateway, set the IP address of the gateway. (Setting range: 0.0.0.1 to 223.255.255.254)

Set a value that satisfies the following conditions.

- The IP address class is any of A, B, and C.
- The subnet address of the gateway is the same as that of the Ethernet-equipped module on the own station.
- The host address bits are not all "0" or all "1".

**Point**

- When the Ethernet-equipped module communicates with an external device on another Ethernet network by Passive open, communication is possible without gateway parameter settings.
- In a system where the Proxy router is used, the gateway parameter settings are not required.

### ■Subnet Address

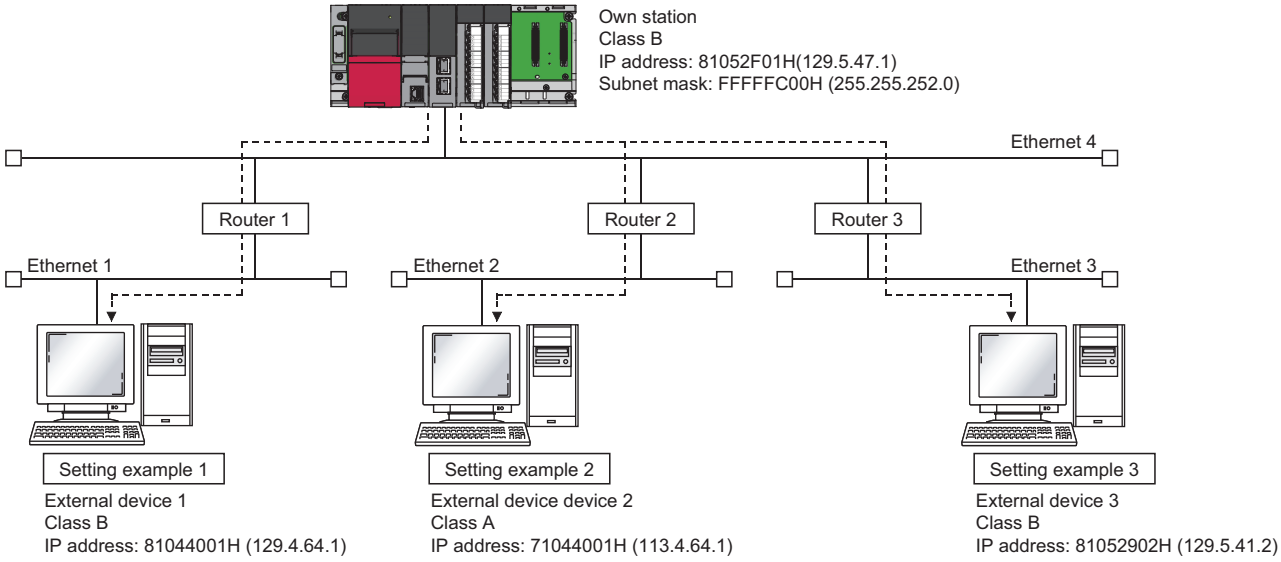
When communicating with an external device on another Ethernet network through a gateway other than the default gateway, set the network address*1 or subnet address*2 of the external device. (Setting range: 0.0.0.1 to 255.255.255.254)

Set a value that satisfies the following conditions.

- The IP address class is any of A, B, and C.
- The host address bits are all "0".

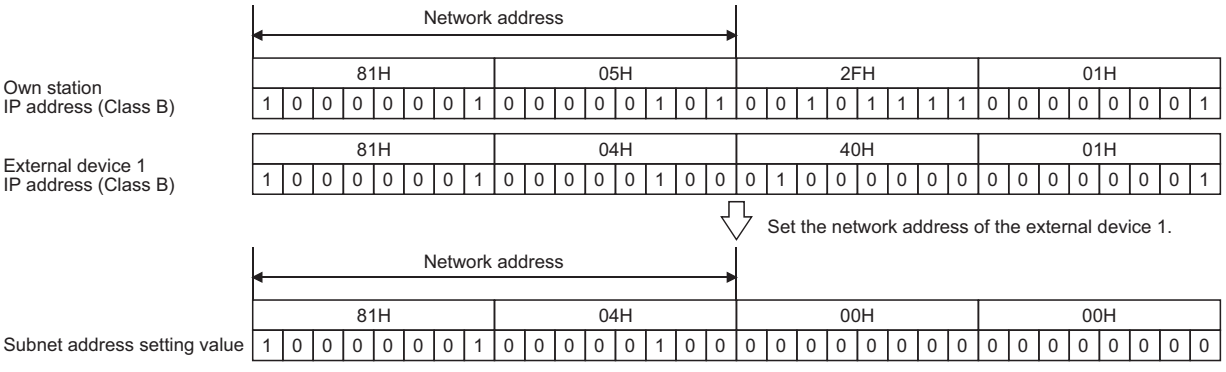
*1 Set the network address of the external device when its class (network address) is different from that of the Ethernet-equipped module on the own station.

*2 Set the network address of the external device when its class (network address) is the same as that of the Ethernet-equipped module on the own station.



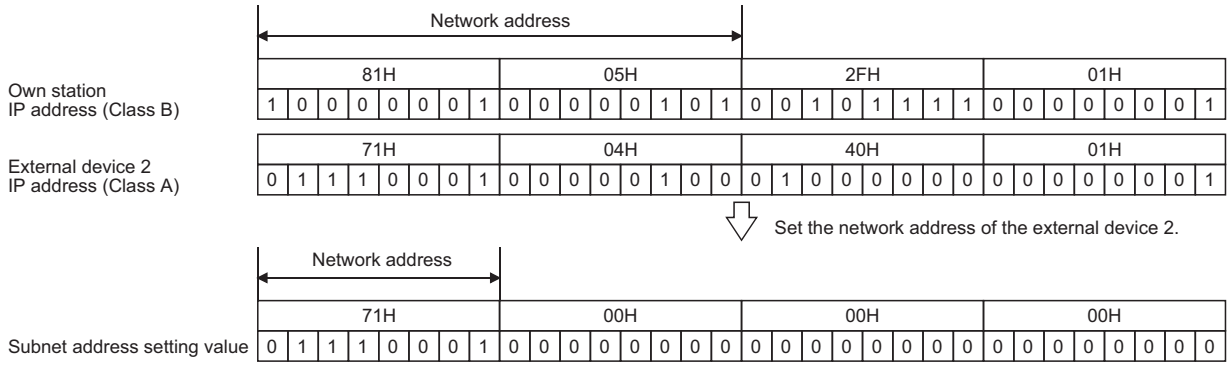

Ex.

When the network addresses differ between the Ethernet-equipped module on the own station and the external device



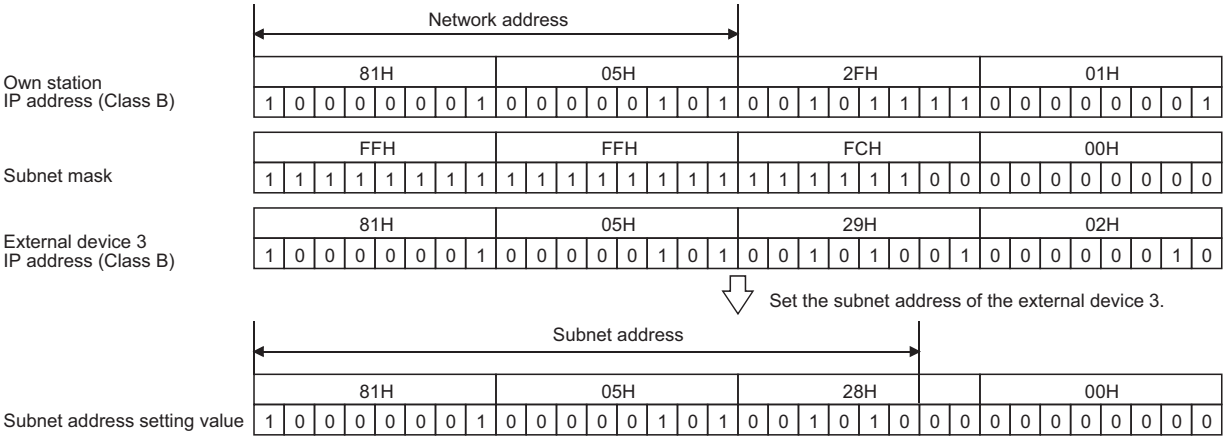
**Ex.**

When the classes differ between the Ethernet-equipped module on the own station and the external device




**Ex.**

When the network address of the Ethernet-equipped module on the own station is the same as that of the external device



# Network/Station No. <-> IP information setting

Set to communicate with another network module or CPU module by network number and station number.
Communication in Ethernet is performed according to the IP address and UDP port number, and access to another network is performed according to the network number and station number.
Therefore, when the Ethernet-equipped module accesses another network, the IP address and UDP port number must be converted into the network number and station number of the network.



| Item | Description | Setting range |
|---|---|---|
| Setting System | Select the system to obtain the IP address and port number of the external device from its network number and station number. (☞ Page 407 Setting system) | • Automatic Response System<br>• IP Address Calculation System<br>• Table Conversion System<br>• Combination System<br>(Default: Automatic Response System) |
| Subnet Mask Pattern | Set the subnet mask pattern. (☞ Page 413 Subnet Mask Pattern) | • Blank<br>• 0.0.0.1 to 255.255.255.255<br>(Default: Blank) |
| Conversion Settings | Set the information of the external device. (☞ Page 413 Conversion Settings) | — |

**Point**

- When transferring messages which are sent to other stations specifying network number and station number, to another network, "Gateway Parameter Settings" in "Application Settings" is required. (☞ Page 404 Gateway Parameter Settings)
- "Network/Station No. <-> IP information setting" is required when other stations are accessed via other Ethernet network systems and must be registered in all the Ethernet-equipped module which has the network number and station number.
- The setting can be made in the CPU module (built-in Ethernet port part) if it has the firmware version supporting communications by network number/station number. (☞ Page 586 Added and Enhanced Functions)

## Setting system

Select the system to obtain the IP address and port number of the external device from its network number and station number.
There are four methods as shown below. The items required to be set ("Subnet Mask Pattern" and "Conversion Settings") differ depending on the method.
○: Setting is required, ×: Setting is not required

| Setting system | Subnet mask pattern | Conversion settings |
|---|---|---|
| Automatic Response System | × | × |
| IP Address Calculation System | ○ | × |
| Table Conversion System | × | ○ |
| Combination System | ○ | ○ |

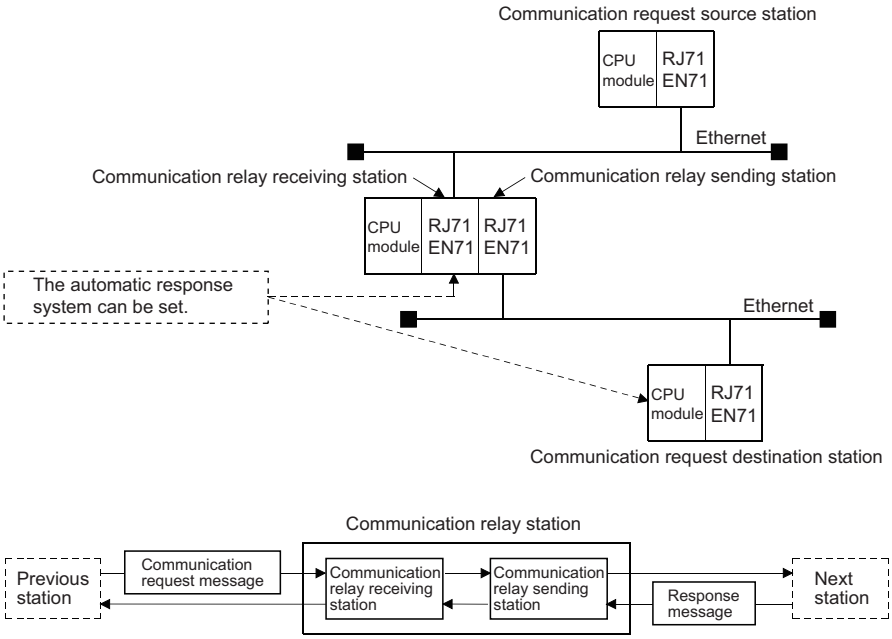
**Point**

- In MELSOFT connection using UDP/IP, if the external device is not registered in the conversion table of table conversion system, the Ethernet-equipped module communicates with the device using the automatic response system. Therefore, even when both of communications with different networks using the link dedicated instruction and MELSOFT connection are performed simultaneously, the Ethernet-equipped module can communicate with the external device in MELSOFT connection regardless of the registered data in the conversion table.
- If the IP address and port number cannot be retrieved when IP address calculation system, table conversion system, or combination system is set, the Ethernet-equipped module will retrieve them using the automatic response system and send a response to the external device.

## ■Automatic Response System

This system differs from the other conversion method in that specification can only be made when the own station is the communication request destination station or communication relay receiving station.

Relay communications via other networks can be performed easily because the IP address and port number of the destination station are not required to be set.







Conversion processing of the automatic response system is as below.

- When receiving the communication request message (command frame), the Ethernet-equipped module internally stores the send source network number, IP address, and UDP port number in the request message, as the station information. The response message (response) for the communication request message is returned to the destination IP address and UDP port number calculated from the stored network number and station number. Therefore, by receiving the other station access communication request message first, the communication destination stations can be maintained.
- The maximum number of station information that can be set is 120. When more than 120 other station access communication request messages are received, the Ethernet-equipped module starts deleting from the oldest data to store the station information in the newly received message communication request message. However, it is ignored if the station information is the same as the station information already stored. (Information from the same station is not stored twice.)

## ■IP Address Calculation System

This system obtains the IP address of the destination station by calculating the network number and station number with the formula described below. The UDP port number predefined for the Ethernet-equipped module is used as the UDP port number of the destination.

IP address of the partner station = (IP address of the own station) Logical product (Net mask pattern for CC-Link IE Controller Network, CC-Link IE Field Network, MELSECNET/H, MELSECNET/10 routing) Logical sum (Network number and station number of the destination)

Conversion processing of the IP address calculation system is as below.

- When a communication request message (command frame) to access other station is received, the IP address is calculated from the network number and station number of the destination in the communication request message, which are stored in the Ethernet-equipped module, and the communication request message is sent to the next station (relay station or destination station).
- The response message (response) for the communication request message is returned based on the return IP address and the stored data above.

**Ex.**

IP address calculation example

The logical product and logical sum are calculated differently depending on the class of the own station IP address. An IP address is calculated as follows.

(For class A)

- When the own station IP address is 79238102H
- When the net mask pattern for routing other networks is FF000000H
- When the destination network number is 03H, and the station number is 05H

| | | |
|---|---|---|
| Own station IP Address | | 7 9 . 2 3 . 8 1 . 0 2 |
| Net Mask Pattern | Logical product | F F . 0 0 . 0 0 . 0 0 |
| Logical product value | | 7 9 . 0 0 . 0 0 . 0 0 |
| Network No. and station No. | Logical sum | 0 3 . 0 5 |
| Partner station IP Address | | 7 9 . 0 0 . 0 3 . 0 5 |

(For class B)

- When the own station IP address is 8438FA0AH
- When the net mask pattern for routing other networks is FFFF0000H
- When the destination network number is 03H, and the station number is 05H

| | | |
|---|---|---|
| Own station IP Address | | 8 4 . 3 8 . F A . 0 A |
| Net Mask Pattern | Logical product | F F . F F . 0 0 . 0 0 |
| Logical product value | | 8 4 . 3 8 . 0 0 . 0 0 |
| Network No. and station No. | Logical sum | 0 3 . 0 5 |
| Partner station IP Address | | 8 4 . 3 8 . 0 3 . 0 5 |

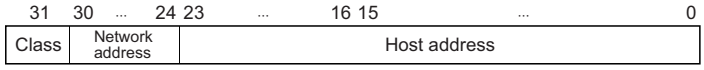
(For class C)

- When the own station IP address is CA65300AH
- When the net mask pattern for routing other networks is FFFFFF00H
- When the destination station number is 02H (The network number is not used.)

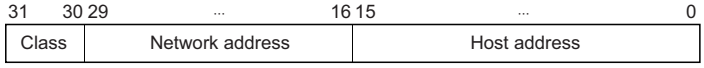
| | | |
|---|---|---|
| Own station IP Address | | C A . 6 5 . 3 0 . 0 A |
| Net Mask Pattern | Logical product | F F . F F . F F . 0 0 |
| Logical product value | | C A . 6 5 . 3 0 . 0 0 |
| Network No. and station No. | Logical sum | 0 2 |
| Partner station IP Address | | C A . 6 5 . 3 0 . 0 2 |

**Point**

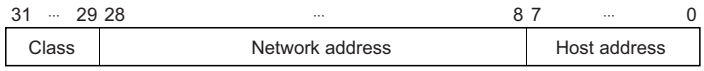
- IP address configuration of class A




- IP address configuration of class B




- IP address configuration of class C



## ■Table Conversion System

This system uses the network number, station number, IP address set in the conversion table of the Network/Station number <-> IP information. The UDP port number predefined for the Ethernet-equipped module is used as the UDP port number of the destination.

If duplicate network and station numbers are set in the conversion table, the data set with the younger registration number is used.

If no data is registered in the conversion table, communication may not be performed successfully.

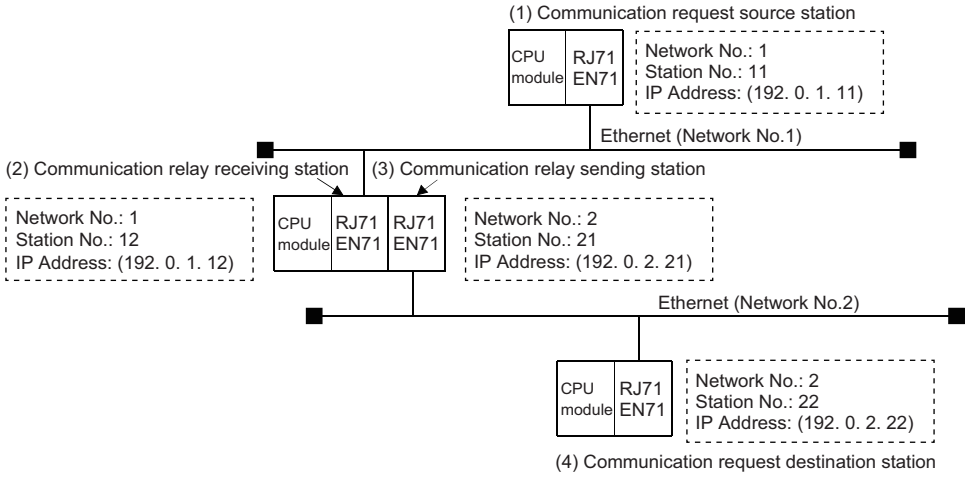
Conversion processing of the table conversion system is as below.

- When a communication request message (command frame) to access other station is received, the same network number and station number are searched in the conversion table of the Network/Station No. <-> IP information, and the communication request message is sent to the destination station with the corresponding IP address.
- The response message (response) for the communication request message is returned based on the return IP address and the data in the conversion table above.

Ex.

Conversion table registration example

The following shows the example of conversion table registration based on the network map.




(a): For communication request message transmission, b): For response message transmission

| Item | | Setting details of each Ethernet-equipped module when accessing from (1) to (2) | | | |
|---|---|---|---|---|---|
| | | (1) Communication request source station (a) | (2) | (3) | (4) |
| Setting value (decimal) | Network No., Station No. | 1, 12 | Setting not required | Setting not required | Setting not required |
| | IP Address | 192.0.1.12 | | | |

| Item | | Setting details of each Ethernet-equipped module when accessing from (1) to (4) | | | |
|---|---|---|---|---|---|
| | | (1) Communication request source station (a) | (2) Communication relay receiving station (b) | (3) Communication relay sending station (a) | (4) Communication request destination station (b) |
| Setting value (decimal) | Network No., Station No. | 1, 12 | 1, 11 | 2, 22 | 2, 21 |
| | IP Address | 192.0.1.12 | 192.0.1.11 | 192.0.2.22 | 192.0.2.21 |

| Item | | Setting details of each Ethernet-equipped module when accessing from (2) to (1) | | | |
|---|---|---|---|---|---|
| | | (1) | (2) Communication request destination station (a) | (3) | (4) |
| Setting value (decimal) | Network No., Station No. | Setting not required | 1, 11 | Setting not required | Setting not required |
| | IP Address | | 192.0.1.11 | | |

| Item | | Setting details of each Ethernet-equipped module when accessing from (3) to (4) | | | |
|---|---|---|---|---|---|
| | | (1) | (2) | (3) Communication relay sending station (a) | (4) |
| Setting value (decimal) | Network No., Station No. | Setting not required | Setting not required | 2, 22 | Setting not required |
| | IP Address | | | 192.0.2.22 | |

| Item | | Setting details of each Ethernet-equipped module when accessing from (4) to (1) | | | |
|---|---|---|---|---|---|
| | | (1) Communication request source station (b) | (2) Communication relay sending station (a) | (3) Communication relay receiving station (b) | (4) Communication request destination station (a) |
| Setting value (decimal) | Network No., Station No. | 1, 12 | 1, 11 | 2, 22 | 2, 21 |
| | IP Address | 192.0.1.12 | 192.0.1.11 | 192.0.2.22 | 192.0.2.21 |

| Item | | Setting details of each Ethernet-equipped module when accessing from (4) to (3) | | | |
|---|---|---|---|---|---|
| | | (1) | (2) | (3) | (4) Communication request destination station (a) |
| Setting value (decimal) | Network No., Station No. | Setting not required | Setting not required | Setting not required | 2, 21 |
| | IP Address | | | | 192.0.2.21 |

## ■Combination System

This system combines the IP address calculation system and the table conversion system.

Select this system when accessing other stations with the same network number, accessing other stations in other networks or Ethernet with different network number, or relaying from other networks to the Ethernet network system.

The set target Ethernet-equipped module can be installed either at a communication request source station, communication relay station, or communication request destination station.

Conversion processing of the combination system is as below.

- When a communication request message (command frame) to access other station is received, the communication request message is sent to the next station obtained by the table conversion system at first.
- If the IP address of the external device cannot be retrieved by the table conversion system, the IP address is retrieved by the IP address calculation system to send the communication request message.
- The response message (response) to the communication request message is returned based on the return IP address and data in the conversion table, or the stored data.

The relationship between the conversion method and the definition of the Ethernet-equipped module mounted station is as follows.

○: Can be set, ×: Cannot be set

| Conversion method | Ethernet-equipped module mounted station definition | | | |
|---|---|---|---|---|
| | Communication request source station | Communication relay station | | Communication request destination station |
| | | Relay receiving station | Relay sending station | |
| Automatic Response System | ×*1 | ○ | ×*1 | ○ |
| IP Address Calculation System | ○ | ○ | ○ | ○ |
| Table Conversion System | ○ | ○ | ○ | ○ |
| Combination System | ○ | ○ | ○ | ○ |

*1 For the external device that completed the communication, the station can be a communication request source station or a relay sending station.

## Subnet Mask Pattern

Specify the mask value based on the guidelines given below. This pattern is used in a logical sum with the own station IP address when calculating the IP address of the external device using the IP address calculation system.
When setting the subnet mask, specify the target settings of the IP address class, network address, and subnet address so that all bits are "1" in the mask pattern.
Specify the mask pattern with a decimal/hexadecimal value obtained by dividing the 32-bit mask value into 8-bit segments.
When the subnet mask is not specified, the mask pattern setting is not necessary. The following mask value is used as the mask pattern according to the own station IP Address class.

| Class | Mask value used |
|---|---|
| Class A | FF.00.00.00H |
| Class B | FF.FF.00.00H |
| Class C | FF.FF.FF.00H |

## Conversion Settings

The conversion setting is required to obtain the IP address from the network number and station number using the table conversion system.
If the own station is an Ethernet-equipped module and the next station to which the communication request/response message is passed to access other station is also an Ethernet-equipped module, specify the network number, station number, and IP address of the Ethernet-equipped module of the next station.

# Interrupt Settings

Set the parameters required for starting up an interrupt program.

| Item | Description | Setting range |
|---|---|---|
| Interrupt Occurrence Factor | Set the interrupt cause.*2 | • SEND Instruction Data Reception*3<br>• Reception Connection<br>(Default: Blank) |
| Channel No./Connection No. | Set the start condition of an interrupt program. When "Interrupt Occurrence Factor" is set to "SEND Instruction Data Reception", set the number of the channel which receives send data.<br>When "Interrupt Occurrence Factor" is set to "Reception Connection", set the number of the connection which receives send data. | • SEND Instruction Data Reception: 1 to 8<br>• Reception Connection: 1 to 64*1<br>(Default: Blank) |
| Detection Method | The detection method for the interrupt condition is displayed when "Interrupt Occurrence Factor" is set. | Edge<br>(Default: Blank) |
| Interrupt Pointer | Set an interrupt pointer (device (I)) to be used for a program. | Device: I0 to I15, I50 to I1023<br>(Default: Blank) |
| Comment | Set a comment for the interrupt pointer used, if required. | 256 one-byte characters maximum<br>(Default: Blank) |

*1 When "Q Compatible Ethernet" is selected in the network type, the setting range is 1 to 16.
*2 An interrupt program is started when the receive data is stored in the channel or connection specified in "Channel No./Connection No.".
*3 When selecting "SEND Instruction Data Reception", set "Communications by Network No./Station No." under "Own Node Settings" of "Basic Settings" to "Enable", and set the network number and station number.

**Point**

Maximum number of interrupt settings can be set to one module (sum of the settings for the P1 connector and the P2 connector) is 16.
Setting number of the interrupt settings is shared in P1 connector and P2 connector.
For example, when SI00 is set on the P1 connector side, SI00 on the P2 connector side cannot be set.

# IP Packet Transfer Setting

Set when using the IP packet transfer function.
The IP packet transfer function enables the Ethernet device (such as computers) connected to the Ethernet-equipped module to communicate with the device on other networks that has an IP address via CC-Link IE Field Network or CC-Link IE Controller Network.
To use this function, set "IP Packet Transfer Function" to "Use". (Default: Not Use)
For details on the IP packet transfer function, refer to the following.

- MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)
- MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

**Point**

If the application timeout of the request source device occurs due to a heavy communication load on the path that an IP packet takes, measure the response time using the PING command from the request source device and adjust the application timeout time. Data size must be within 1460 bytes when the PING command is used.

# Network Dynamic Routing

Set when using the dynamic routing.

Setting "Network Dynamic Routing" is possible only when values are set in "Network No." and "Station No." under "Own Node Settings" of "Basic Settings". (☞ Page 328 Details of items)

For details, refer to the following.

☞ Page 553 Communications with Different Networks

**Point**

- When the communication path is set in "Routing Setting" of "CPU Parameter", setting of "Routing Setting" has a priority.
- When enabling dynamic routing, do not connect the CPU module (built-in Ethernet port part) and the Ethernet-equipped module on the same Ethernet using a hub or other means.

# Module Operation Mode

Set the module operation mode.

Set the same operation mode for both of the P1 connector and P2 connector.

For a redundant system, select the module operation mode of system A.

For details on module communication test mode, refer to the following.

☞ Page 424 Module communication test

| Item | Description | Setting range |
|---|---|---|
| Module Operation Mode | Online<br>• Normal operation mode<br>Offline<br>• Communication with other stations and dedicated instruction cannot be performed in this mode. Select this mode to debug the program while the network is not yet connected.<br>Module Communication Test<br>• A mode for operating the module communication test. Select this mode to check the operation when the module is in abnormal condition. | • Online<br>• Offline<br>• Module Communication Test<br>(Default: Online) |

When different operation modes are set for the P1 connector and P2 connector, the Ethernet-equipped module operates with the following operation modes.

| Module Operation Mode | | Operation mode of Ethernet-equipped module |
|---|---|---|
| P1 connector | P2 connector | |
| Online | Offline | Offline |
| | Module Communication Test | Module Communication Test |
| Offline | Online | Offline |
| | Module Communication Test | Module Communication Test |
| Module Communication Test | Online | Module Communication Test |
| | Offline | |

# Redundant System Settings

Set when using an Ethernet-equipped module in a redundant system.

| Item | Description | Setting range |
|---|---|---|
| To Use or Not to Use Redundant System Settings | Select whether to use the redundant function or not.<br>Set the same setting for port 1 and port 2.<br>Select "Use" when installing the RJ71EN71 to the main base unit.<br>Select "Not Use" when installing the RJ71EN71 to the extension base unit. | • Use<br>• Not Use<br>(Default: Use) |
| Own Node (System A) IP Address | The IP address set in "Own Node Settings" under "Basic Settings" is displayed.<br>Not displayed when it has not been set. | — |
| System B IP Address | Set the IP address of system B.<br>Ensure that this address has the same class and subnet address as "Own Node (System A) IP Address".<br>When the parameters are written without the IP address set (blank), the following IP address is set.<br>• P1 connector of RJ71EN71: 192.168.3.42<br>• P2 connector of RJ71EN71: 192.168.4.42<br>• CPU module (built-in Ethernet port part): 192.168.3.41 | • Blank<br>• 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| Own Node (System A) Station Number | The station number set in "Own Node Settings" under "Basic Settings" is displayed.<br>Not displayed when it has not been set. | — |
| System B Station Number[*1] | Set the system B station number.<br>Set when the station number is set in "Own Node Settings" under "Basic Settings". | 1 to 120<br>(Default: 2) |
| Use of Control System IP Address[*2] | Set whether to use the control system IP address or not.<br>• When "Use" is selected, set the control system IP address. When the control system IP address is used, access using the control IP address is possible even if system switching occurs.<br>• When "Not Use" is selected, access is made using the own node (system A) or system B IP address.<br>When the RJ71EN71 is used, pay attention to the following.<br>• When using port group settings, the setting for "Use of Control System IP Address" should be the same for port 1 and port 2. | • Not Use<br>• Use<br>(Default: Not Use) |
| Control System IP Address[*2] | Set the control system IP address. Set the address after setting "Use" to "Use of Control System IP Address".<br>Set so that the IP address has the same class and subnet address as the IP addresses used in the system. Consult with the network manager for the IP address setting.<br>Set the IP address not to overlap the one used in the system.<br>Set the IP address class within the range of classes A, B and C.<br>Set the same control system IP address for system A P1 and system B P1, or for system A P2 and system B P2. | • Blank<br>• 0.0.0.1 to 223.255.255.254<br>(Default: Blank) |
| Own Node (System A) Module Operation Mode | The module operation mode set for "Module Operation Mode" under "Application Settings" is displayed. | — |
| System B Module Operation Mode[*1] | Set the module operation mode of system B.<br>For details on the module operation mode, refer to the following.<br>☞ Page 415 Module Operation Mode | • Online<br>• Offline<br>• Module Communication Test<br>(Default: Online) |
| Port Group Setting[*1][*2] | Set the port group.<br>Set the same value for port 1 and port 2.<br>If "No Setting" is selected, when a system switching error occurs on either port, performs system switching.<br>This setting is not available when "Q Compatible Ethernet" is set for the RJ71EN71 network type. | • No Setting<br>• Switch Systems When a System Switching Error Occurs in Port 1<br>• Switch Systems When a System Switching Error Occurs in Port 2<br>• Switch Systems When a System Switching Error Occurs in Both of Port 1 and Port 2<br>(Default: No Setting) |
| System Switching Request Issuing at Disconnection Detection[*1] | Set whether to issue a system switching request when a cable disconnection is detected. | • Enable<br>• Disable<br>(Default: Enable) |
| Disconnection Detection Monitoring Time[*1] | Set the time between cable disconnection detection after communication starts normally and issuance of a system switching request to the control system CPU module. | Unit [s]: 0.0 to 30.0<br>Unit [ms]: 0 to 30000 (in increments of 100ms)<br>(Default: 2s) |

| Item | Description | Setting range |
|---|---|---|
| System Switching Request Issue at Communication Error*1 | Set whether to issue a system switching request when a communication error occurs. | • Enable<br>• Disable<br>(Default: Disable) |
| User Connection*1 | Set the connection for issuance of a system switching request when a communication error occurs, from the connections set in "External Device Configuration" under "Basic Settings".<br>By setting the comment for the connection device in "External Device Configuration" under "Basic Settings", the connection can be easily identified.<br>(☞ Page 329 External Device Configuration) | • Not to Issue<br>• Issue<br>(Default: Not to Issue) |
| System Connection*1 | Set whether to issue a system switching request when a communication error occurs for each port. Set this item after setting "Enable" to "System Switching Request Issue at Communication Error". ☞ Page 417 System connection | |

*1 This setting is not available for the CPU module (built-in Ethernet port part).

*2 This setting is not available when the RJ71EN71 network type is set to "Q Compatible Ethernet".

## System connection

Set "Enable" to "System Switching Request Issue at Communication Error" before setting the following items.

| Item | Description | Setting range |
|---|---|---|
| Auto-open UDP Port | Set whether to issue a system switching request using the auto open UDP port when a communication error occurs. This port is used for SLMP communications.<br>This setting is enabled when the following conditions are met.<br>• The remote password setting is enabled.<br>• The remote password is unlocked.<br>When the above conditions are not met, even if a communication error occurs in the target connection, a system switching request will not be issued. | • No System Switching Request at Communication Error<br>• System Switching Request at Communication Error<br>(Default: No System Switching Request at Communication Error) |
| MELSOFT Transmission Port (UDP/IP) | Set whether to issue a system switching request using the MELSOFT transmission port (UDP/IP) when a communication error occurs. This port is used for UDP/IP communications with MELSOFT products.<br>This setting is enabled when the following conditions are met.<br>• The remote password setting is enabled.<br>• The remote password is unlocked.<br>When the above conditions are not met, even if a communication error occurs in the target connection, a system switching request will not be issued.<br>When "System Switching Request at Communication Error" is selected, a system switching request is issued when a communication error occurs in the following ports.<br>• Port used for communications using a dedicated instruction<br>• Relay transmission port with other networks<br>• MELSOFT direct connection port<br>• Direct connection port with the engineering tool<br>When "No System Switching Request at Communication Error" is selected, even if a communication error occurs in any of the above ports, a system switching request will not be issued. | • No System Switching Request at Communication Error<br>• System Switching Request at Communication Error<br>(Default: No System Switching Request at Communication Error) |
| MELSOFT Transmission Port (TCP/IP) | Set whether to issue a system switching request using the MELSOFT transmission port (TCP/IP) when a communication error occurs. This port is used for TCP/IP communications with MELSOFT products. | • No System Switching Request at Communication Error<br>• System Switching Request at Communication Error<br>(Default: No System Switching Request at Communication Error) |
| FTP Transmission Port (TCP/IP) | Set whether to issue a system switching request using the FTP transmission port (TCP/IP) when a communication error occurs. This port is used for the file transfer function (FTP server). | • No System Switching Request at Communication Error<br>• System Switching Request at Communication Error<br>(Default: No System Switching Request at Communication Error) |
| Simple CPU communication port | Set whether to issue a system switching request using the simple CPU communication port when a communication error occurs. This port is used for the simple CPU communication function. | • No System Switching Request at Communication Error<br>• System Switching Request at Communication Error<br>(Default: No System Switching Request at Communication Error) |

## Precautions on simple CPU communication

Before using the simple CPU communication in a redundant system, check the following.

### ■Redundant system settings

To execute system switching when a communication error (alive check error and ULP timeout) is detected, set the following items in "Redundant System Settings" under "Application Settings". (☞ Page 304 Communication error detection)

- "System Switching Request Issue at Communication Error": Enable
- "Simple CPU Communication Port under System connection": System Switching Request at Communication Error (☞ Page 417 System connection)

System switching is executed only when a communication error occurs. When the following errors occur, system switching is not executed.

- CFB4H: An abnormal response was received from the external device.
- CFB5H: The frame received from the external device is incorrect.

When the simple CPU communication have never been executed, system switching is not executed even if a communication error occurs.

For other redundant settings, such as "System B IP address", "Control system IP address", and "Port group settings", configure them as necessary.

When a control system IP address is set, it is used for the own station IP address; otherwise, the IP addresses of systems A and B are used.

### ■Operation during system switching

Depending on "Communication Setting: Execution Interval (ms)" in "Simple CPU Communication Setting" under "Application Settings", system switching operates in the following manner.

| Communication setting | Operation |
|---|---|
| Fixed interval | Communications stop temporarily during system switching and restart automatically from the new control system. |
| On Request | In the same manner as in the normal operation start, when Request to start communication at request (Un\G721896 to Un\G721899, Un\G1247300 to Un\G1247327) of each setting numbers is turned on under the following status after system switching is completed, communications restart.<br>• Preparation completion (Un\G721912 to Un\G721915, Un\G1247412 to Un\G1247439) for each setting number is turned on: Ready<br>• Simple CPU communication status (Un\G721936 to Un\G721999, Un\G1247460 to Un\G1247907) for each setting number is 2: Waiting for request |

## iQ Sensor Solution Settings

Set the setting related to iQ Sensor Solution.

| Item | Description | Setting range |
|---|---|---|
| Use iQ Sensor Solution | Select whether to use iQ Sensor Solution compatible functions. | • Use<br>• Not to Use<br>(Default: Use) |

## SLMPSND Instruction Settings

Set the setting related to the SLMPSND instruction.

| Item | Description | Setting range |
|---|---|---|
| Use SLMPSND Instruction | Select whether to use the SLMPSND instruction. | • Use<br>• Not to Use<br>(Default: Use) |

# 3 TROUBLESHOOTING

This chapter describes troubleshooting of when the Ethernet function is used.

## 3.1 Checking with LED

This section describes troubleshooting using LED.

Error status can be determined by status of the RUN LED and the ERR LED.

| RUN LED | ERR LED | Error status*1 | Description |
|---|---|---|---|
| Off | On, flashing | Major error | An error such as hardware failure or memory failure. The module stops operating. |
| On | Flashing | Moderate error | An error, such as parameter error, which affect module operation. The module stops operating. |
| On | On | Minor error | An error such as communication failure. The module continues operating. |

*1 When multiple errors occur, the error status is displayed in the order of major, moderate, and minor.

**Point**

For the RJ71EN71 or RnENCPU, whether the error occurs in the P1 or P2 can be checked with P ERR LED.

### When the RUN LED or READY LED turns off

When the READY LED of the CPU module turns off after power-on, refer to the troubleshooting of the CPU module. (Manual for the CPU module used)

When the RUN LED of the RJ71EN71 or RnENCPU turns off, check the following.

| Check item | Action |
|---|---|
| Is the Ethernet-equipped module mounted correctly? | Securely mount the Ethernet-equipped module on the base unit. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (Page 424 Module communication test)

### When the ERROR LED or ERR LED turns on or is flashing

When the ERROR LED of the CPU module turns on or is flashing, perform the module diagnostics of the CPU module.

[Diagnostics] ⇨ [Module Diagnostics (CPU Diagnostics)]

When the ERR LED of the RJ71EN71 or RnENCPU turns on or is flashing, identify the error cause using the engineering tool. (Page 422 Checking the Module Status)

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (Page 424 Module communication test)

## When the SD/RD LED does not turn on at data sending

When the SD/RD LED does not turn on at data sending, check the following items.

| Check item | Action |
|---|---|
| Is the ERROR LED or ERR LED on or flashing? | Identify the error cause using the engineering tool. (☞ Page 422 Checking the Module Status, ☞ Page 426 Checking the Network Status). |
| Are the cables properly connected? | Correct the cable properly. |
| | Perform the following tests to check for the status of the cable connection and line.<br>• PING test (☞ Page 431 PING Test)<br>• Communication status test (☞ Page 432 Communication Status Test) |
| Is the program correct? | Check and correct the send program of the Ethernet-equipped module. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*1

*1 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

## When data cannot be received with the SD/RD LED off

When data cannot be received with the SD/RD LED off, check the following.

| Check item | Action |
|---|---|
| Is the ERROR LED or ERR LED on or flashing? | Identify the error cause using the engineering tool. (☞ Page 422 Checking the Module Status, ☞ Page 426 Checking the Network Status). |
| Are the cables properly connected? | Correct the cable properly. |
| | Perform the following tests to check for the status of the cable connection and line.<br>• PING test (☞ Page 431 PING Test)<br>• Communication status test (☞ Page 432 Communication Status Test) |
| Are the parameter settings correct? | Check the module parameters of the Ethernet-equipped module. Correct the value for the following setting if it is wrongly set.<br>• "IP Address" under "Own Node Settings" of "Basic Settings"<br>• "Gateway Parameter Settings" of "Application Settings"<br>☞ Page 326 PARAMETER SETTINGS |
| Is the program correct? | Check and correct the send program of the external device. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*1

*1 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

## When the L ER LED turns on

When the L ER LED turns on, check the following.

| Check item | Action |
|---|---|
| Are the Ethernet cables used normally? | • Check if the Ethernet cable which conforms the standard is used. (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup))<br>• Check if the station-to-station distance is set within range. (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup))<br>• Check if the Ethernet cables are not disconnected. |
| Does the cabling condition (bending radius) meet the specifications? | Refer to the manual for the Ethernet cable, and correct the bending radius. |
| Is the hub used operating normally? | • Check if the hub which conforms the standard is used. (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup))<br>• Power off and on the hub. |
| Is there any source of noise near the module or cables? | Change the location of the module or cables. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)

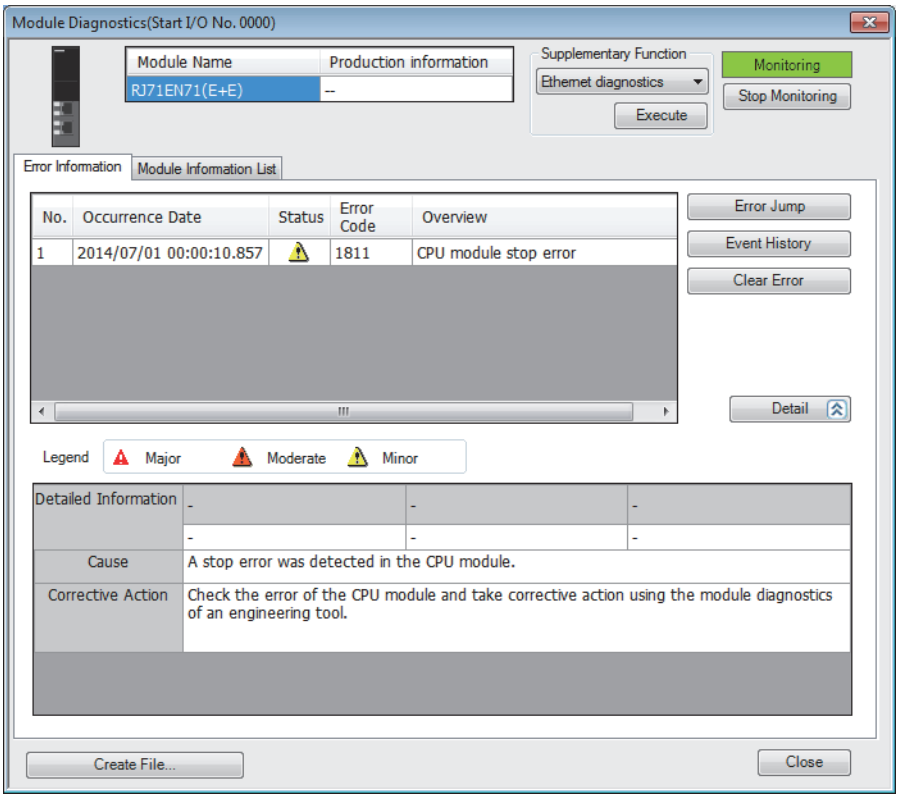
# 3.2 Checking the Module Status

The following table lists the functions which can be used in the "Module Diagnostics" window of the Ethernet-equipped module.

| Functions | | Applications |
|---|---|---|
| Error Information | | Displays the details of the errors currently occurring and the actions to remove them.<br>"-" may be displayed for "Occurrence Date" of an error that occurred immediately after power-on. To check the occurrence date, click the [Event History] button to refer to the event history. |
| Module information list | | Displays various status information of the Ethernet-equipped module |
| Supplementary Function | Ethernet Diagnostics | Enables checking the cause to resolve the problem when an error occurs in the Ethernet system. (☞ Page 426 Checking the Network Status) |

## Error Information

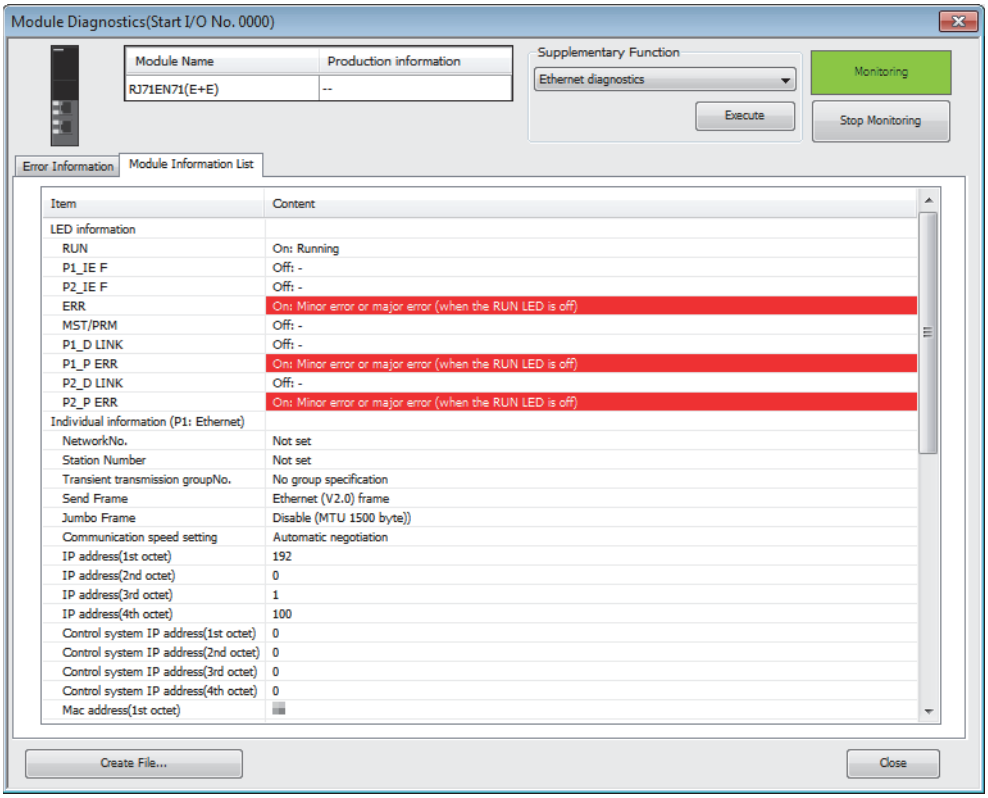
Check the details of the error currently occurring and action to remove the error.




| Item | Description |
|---|---|
| Status | Major: An error such as hardware failure or memory failure. The module stops operating. |
| | Moderate: An error, such as parameter error, which affect module operation. The module stops operating. |
| | Minor: An error such as communication failure. The module continues operating. |
| Detailed Information | Displays detailed information about each error (maximum of 3 pieces). |
| Cause | Displays the detailed error causes. |
| Corrective Action | Displays the actions to eliminate the error causes. |

## Module Information List

Switch to the [Module Information List] tab to check various status information of the Ethernet-equipped module. (The displayed items vary depending on the module.)




| Item | | Description |
|---|---|---|
| LED information | | Displays the LED status of the Ethernet-equipped module. |
| Individual information (P1: Ethernet)*1 | Network Number | Displays the network number set for the selected module. |
| | Station Number | Displays the station number set for the selected module. |
| | Transient transmission group No. | Displays the transient transmission group number set for the selected module. |
| | Send Frame | Displays the send frame setting set for the selected module. |
| | Jumbo Frame | Displays the jumbo frame setting set for the selected module. |
| | Communication speed setting | Displays the communication mode set for the selected module. |
| | IP address (1st octet) to IP address (4th octet) | Displays the IP address set for the selected module. |
| | Control system IP address (1st octet) to Control system IP address (4th octet) | Displays the control system IP address set for the selected module. |
| | MAC address (1st octet) to MAC address (6th octet) | Displays the MAC address of the selected module. |
| Individual information (P2: Ethernet)*1 | | (Same as the P1 connector) |

*1 This item is displayed when the network type is set to "Ethernet".
For when the network type is set to "CC-Link IE Control", refer to the following.
MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)
For when the network type is set to "CC-Link IE Field", refer to the following.
MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

# Module communication test

The module communication test checks the hardware of the RJ71EN71 or the RnENCPU (network part). When the communication using the RJ71EN71 or the RnENCPU (network part) is unstable, whether a hardware failure occurs or not can be checked.

The following table lists the tests performed.

| Test item | Description |
|---|---|
| Internal self-loopback test | Checks whether the communication function of the module can be performed normally. |
| External self-loopback test | Checks whether the communication can be performed normally with the cable connected between two connectors. |

## Procedure

### ■Systems other than the redundant system

1. Set the module operation mode for P1 and P2 each to module communication test mode in the following item.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Module Parameter] ⇨ [Application Settings] ⇨ [Module Operation Mode]

2. Connect the P1 and P2 of the RJ71EN71 or the RnENCPU (network part) with an Ethernet cable.
3. Write the module parameters to the CPU module.
4. Reset or power off and on the CPU module to start the module communication test.

**Point**

Do not perform a module communication test while connected to another station. The operation of another station may failed.

### ■Redundant system

To perform a module communication test for a redundant system, set the operation mode to separate mode. (Manual for the CPU module used)

Before execution, perform system switching in advance so that the module communication test target station operates as the standby system.

The following describes a procedure for executing a module communication test for the RJ71EN71 of system B (standby system).

1. Set the module operation mode of system A to online mode in the following item.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Module Parameter] ⇨ [Application Settings] ⇨ [Module Operation Mode]

2. Set the module operation mode of system B to module communication test mode in the following item.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71] ⇨ [Module Parameter] ⇨ [Application Settings] ⇨ [Redundant System Settings] ⇨ [System B Module Operation Mode]

3. Connect the P1 and P2 of the RJ71EN71 of system B with an Ethernet cable.
4. Set the system B CPU module to the STOP state.
5. Write the module parameters to the system B CPU module.
6. Reset or power off and on the system B CPU module to start a module communication test.

**Point**

- To perform a module communication test on system A, set the module operation mode of system A to module communication test mode and the module operation mode of system B to online mode.
- When returning the module to a normal operation state after the test, reconnect the Ethernet cable as before the test, and then execute the CPU module memory copy to transfer the parameters of the control system to the standby system. (Manual for the CPU module used)
- Do not perform a module communication test while connected to another station. The operation of another station may failed.



## Check of status and result

The test status and result can be checked with the dot matrix LED of the module.

| Test status | LED display |
|---|---|
| Test in progress | The dot matrix LED displays "UCT". |
| Normal completion | The dot matrix LED displays "OK". |
| Abnormal end | The ERR LED turns on and the dot matrix LED indicates "ERR" and error number alternately at intervals of 1 second. |

### ■Error number when the test abnormally ended.

The dot matrix LED indicates the error number with the form of "Target Ethernet port Error number".

For example, "1 3" is displayed when error No.3 occurs in P1.

If the module communication test fails, take the following actions.

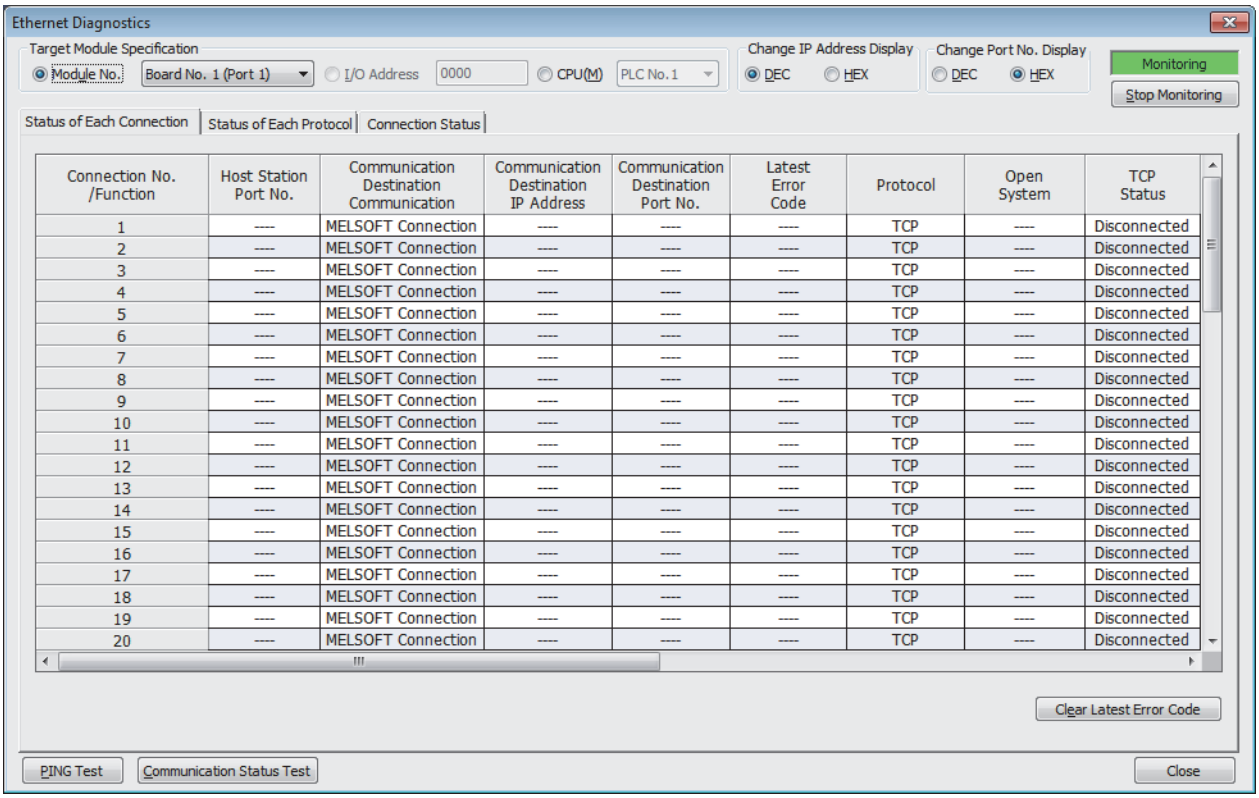
| Error number | Description | Action |
|---|---|---|
| 1 | Internal self-loopback test failure | Please consult your local Mitsubishi representative. |
| 2 | External self-loopback test connection error | Check the Ethernet cable connection or replace the Ethernet cable, and perform the test again. If the test fails again, Please consult your local Mitsubishi representative. |
| 3 | External self-loopback test communication error | Replace the Ethernet cable and perform the test again. If the test fails again, Please consult your local Mitsubishi representative. |

# 3.3 Checking the Network Status

## Ethernet diagnostics

The communication status of the Ethernet-equipped module and external device can be checked with Ethernet diagnostics.

[Diagnostics] ⇨ [Ethernet Diagnostics]




| Connection No. /Function | Host Station Port No. | Communication Destination Communication | Communication Destination IP Address | Communication Destination Port No. | Latest Error Code | Protocol | Open System | TCP Status |
|---|---|---|---|---|---|---|---|---|
| 1 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 2 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 3 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 4 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 5 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 6 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 7 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 8 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 9 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 10 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 11 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 12 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 13 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 14 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 15 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 16 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 17 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 18 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 19 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |
| 20 | ---- | MELSOFT Connection | ---- | ---- | ---- | TCP | ---- | Disconnected |

Set the Ethernet-equipped module to be diagnosed in "Target Module Specification".

**Point**

When starting the Ethernet diagnostics with a setting other than "No Specification" in "Other Station Setting" in the "Specify Connection Destination Connection" window, use the engineering tool version 1.035M or later.
The Ethernet diagnostics can be started for stations of up to eight networks ahead (number of relay stations: 7) including the network to which the station directly connected with the engineering tool belongs.
When specifying a setting other than "No Specification", specify an Ethernet-equipped module to be diagnosed. If another Ethernet-equipped module is specified, the Ethernet diagnostics can be started only for the specified Ethernet-equipped module, but not for the non-specified Ethernet-equipped module even if it is on the same base unit.

## Precautions

### ■Communication paths through which Ethernet diagnostics cannot be started

If the communication path includes the following items, the Ethernet diagnostics cannot be started.

- MELSECNET/H
- Multidrop connection with a serial communication module
- Interface board for a personal computer
- GOT (when the transparent function is used)
- MELSEC-Q/L series network module

If any of the following conditions apply in the "Specify Connection Destination Connection" window, Ethernet diagnostics cannot be started.

- "Other Station (Single Network)" is specified in "Other Station Setting" and "CC-Link" or "C24" is specified in "Network Communication Route".
- "Other Station (Co-existence Network)" is specified in "Other Station Setting" and "CC-Link" or "C24" is specified in "Co-existence Network Route".

### ■Measures for redundant system configuration

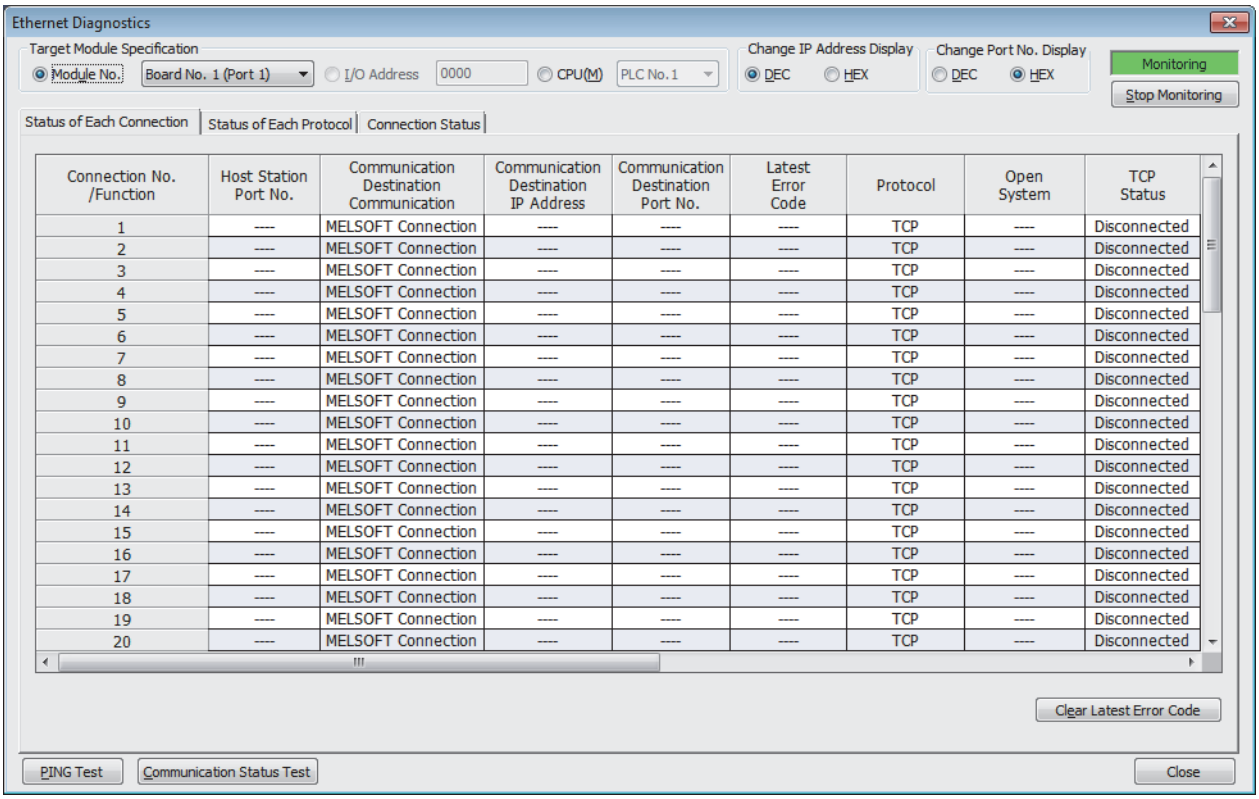
In a redundant system, the Ethernet diagnostics cannot be started when a system has been specified for "Specify Redundant CPU" in "Target System" in the "Specify Connection Destination Connection" window. Directly connect a station to be diagnosed to the engineering tool and specify the module connected to the engineering tool ("Specify Redundant CPU" is set to "Not Specified") to start the Ethernet diagnostics.

In addition, the Ethernet diagnostics cannot be started in the module diagnostics window when the system is switched between A and B in "Main Base Information" of "System Monitor".

In a redundant system configuration of the remote head module, the Ethernet diagnostics cannot be started when the engineering tool is connected to the remote head module of the standby system. Connect it to the remote head module of the control system, and start the diagnostics.

## Status of Each Connection

The status of each connection of the Ethernet-equipped module selected.




The following table lists the displayed items in "Status of Each Connection" tab.

| Item | Description |
|---|---|
| Connection No./Function | Displays the connection number and functions (FTP server, FTP client*3, MELSOFT direct connection). |
| Host Station Port No. | Displays the own station port number used. |
| Communication Destination Communication Method*2 | Displays the communication method. |
| Communication Destination IP Address | Displays the IP address of the sensor/device to be connected, which is set in the parameter settings. |
| Communication Destination Port No. | Displays the port number of the sensor/device to be connected, which is set in the parameter settings. |
| Latest Error Code | Displays the error code that indicates the definition of latest error occurred. |
| Protocol | Displays the protocol (TCP/IP or UDP/IP) |
| Open System | Displays the open method (Active, Unpassive, or Fullpassive) when the protocol of the connection is TCP/IP. |
| TCP Status | Displays the status (open status) of connection with the sensor/device when the protocol of the connection status is TCP/IP. |
| Pairing Open*1 | Displays the setting status of pairing open when the communication method of the connection is the fixed buffer. |
| Existence Confirmation*1 | Displays the alive check method set in the parameter settings. |
| Remote Password Status | Displays the remote password setting status. |
| Consecutive failed unlock attempts | Displays the total number of continuous failure of remote password unlock. |
| Force Deactivation Status | Displays the status of forced invalidation specification. |

*1 This item is displayed only for the RJ71EN71 and the RnENCPU (network part).

*2 This item is displayed only when the network type of the RJ71EN71 is set to "Q Compatible Ethernet" and "MELSOFT Connection Module" is set as an external device in "External Device Configuration" of "Basic Settings".

*3 This item is displayed only for the CPU module (built-in Ethernet port part).

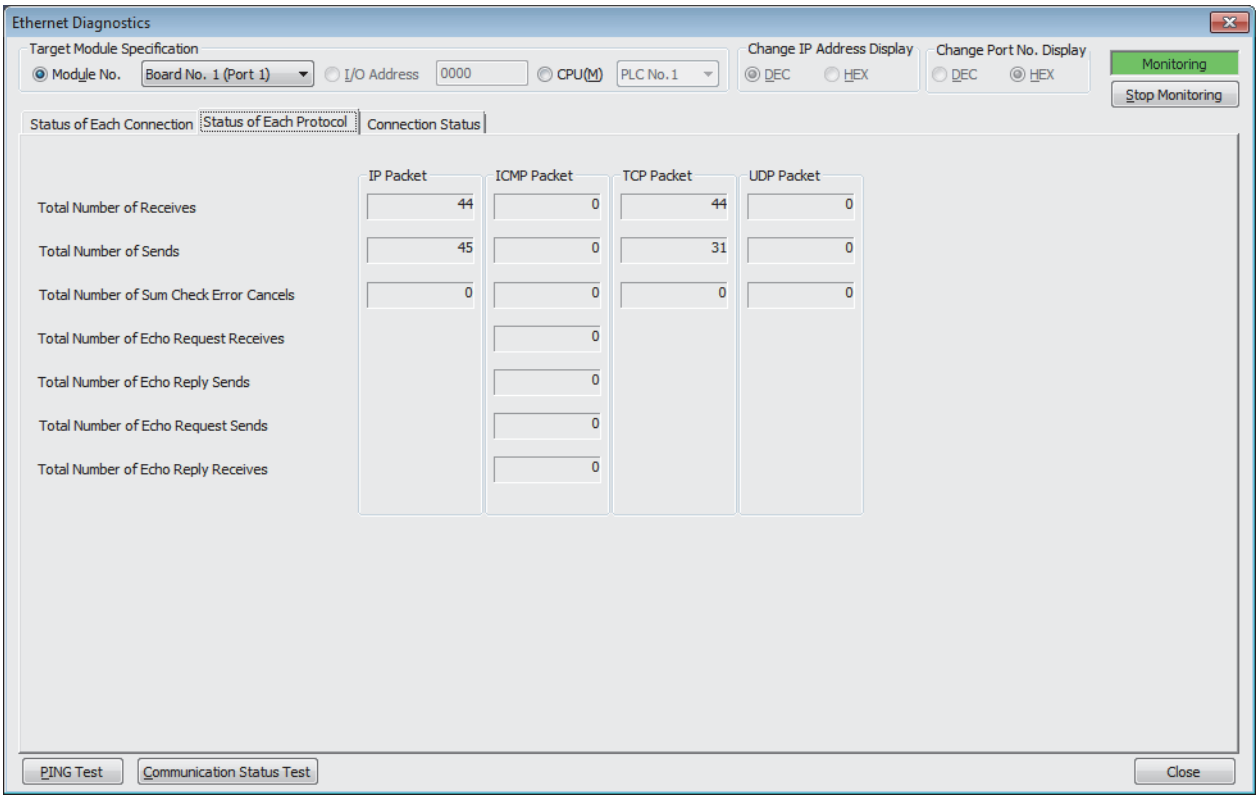
Click the [Clear Latest Error Code] button to clear all the errors displayed in "Latest Error Code" of each connection.

**Point**

Information about connection No.17 to 64, FTP server, MELSOFT direct connection are not displayed when "Q Compatible Ethernet" is set in the network type of the RJ71EN71.

## Status of Each Protocol

The total number of packets sent/received by each protocol of the selected Ethernet-equipped module can be checked.

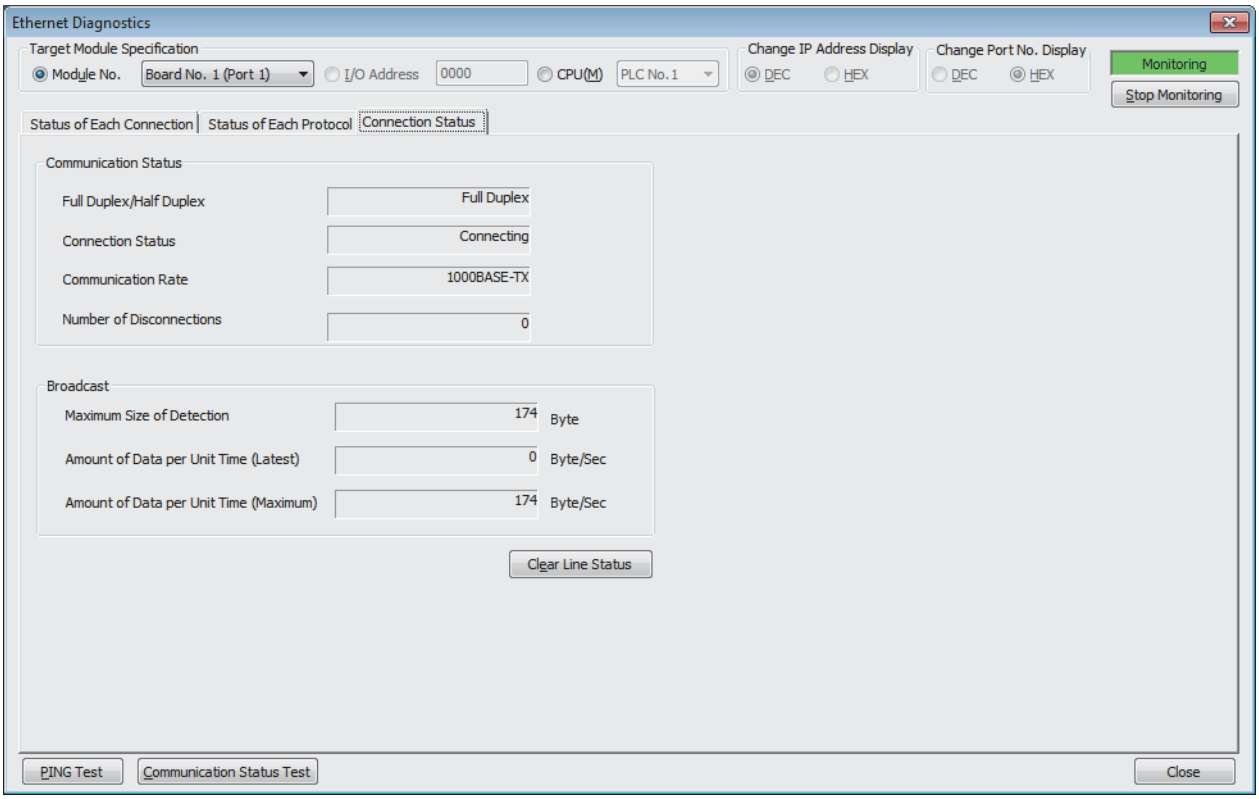



The following table lists the displayed items in "Status of Each Protocol" tab.

| Item | Description | Display range |
|---|---|---|
| Total Number of Receives | Displays the total number of received packets. | 0 to 4294967295 |
| Total Number of Sends | Displays the total number of sent packets. | 0 to 4294967295 |
| Total Number of Sum Check Error Cancels | Displays the number of times the received packet was discarded due to checksum error. | 0 to 4294967295 |
| Total Number of Echo Request Receives | Displays the total number of received ICMP echo request packets. | 0 to 4294967295 |
| Total Number of Echo Reply Sends | Displays the total number of sent ICMP echo reply packets. | 0 to 4294967295 |
| Total Number of Echo Request Sends | Displays the total number of sent ICMP echo request packets. | 0 to 4294967295 |
| Total Number of Echo Reply Receives | Displays the total number of received ICMP echo reply packets. | 0 to 4294967295 |

## Connection Status

The communication status of the Ethernet-equipped module.




The following table lists the displayed items in "Connection Status" tab.

| Item | | Description | Display range |
|---|---|---|---|
| Communication Status | Full Duplex/Half Duplex | Displays whether the line is full-duplex or half-duplex. | — |
| | Connection Status | Displays the cable connection status. | — |
| | Communication Rate | Displays the communication speed. | — |
| | Number of Disconnections | Displays the number of times the line entered a state where communication cannot be performed. | 0 to 65535 |
| Broadcast | Maximum Size of Detection | Displays the maximum size of discarded broadcast messages. | 0 to 65535 |
| | Amount of Data per Unit Time (Latest) | Displays the size (latest value) per second of discarded broadcast messages. | 0 to 4294967295 |
| | Amount of Data per Unit Time (Maximum) | Displays the size (maximum value) per second of discarded broadcast messages. | 0 to 4294967295 |

Click the [Clear Line Status] button to clear all the data in "Broadcast".

**Point**

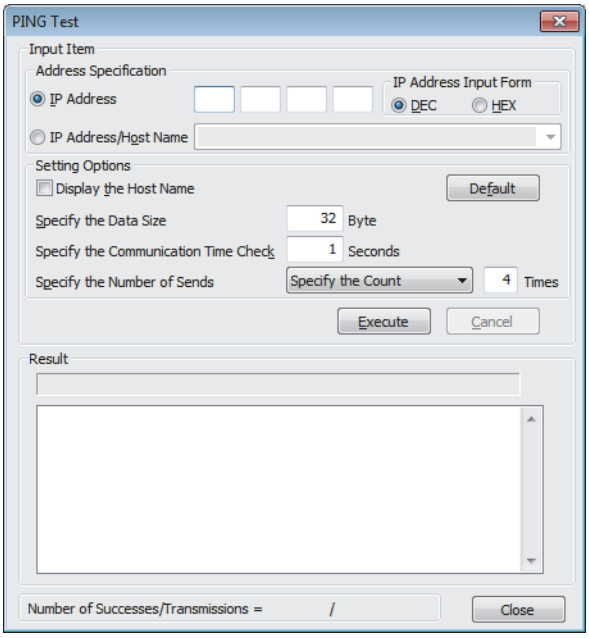
Information in "Broadcast" is not displayed when "Q Compatible Ethernet" is selected for the network type of the RJ71EN71.

## PING Test

The PING test checks existence of an Ethernet device on the same Ethernet network.

This test is performed on the network of stations connected to the engineering tool by sending packets for check. If a response returns, the communication can be performed.

"Ethernet Diagnostics" window ⇨ [PING Test] button




### ■Procedure

Set the required items in "Input Item" and click the [Execute] button to execute the PING test. The test results are displayed in the "Result" box.

**Point**

- When executing the PING test from the RJ71EN71 and the RnENCPU (network part) by connecting the engineering tool and CPU module with a USB cable, network number and station number setting is required to specify the PING executing station.
- When the Ethernet diagnostics are started with a setting other than "No Specification" in "Other Station Setting" on the "Specify Connection Destination Connection" window, the PING test cannot be used.

### ■Action for abnormal end

If the test fails, check the following and perform the test again.

- Whether the Ethernet-equipped module is properly mounted on the base unit
- Connection to the Ethernet network
- Parameter settings written in the CPU module
- Operating status of the CPU module (whether or not an error has occurred)
- IP addresses set in the engineering tool and the PING test target station
- Whether the external device has been reset after the Ethernet-equipped module was replaced

## Communication Status Test

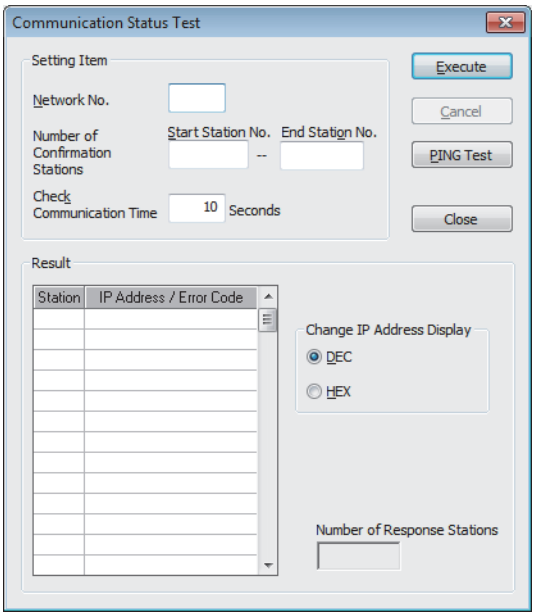
The communication status test checks whether the communication between the own station and the external device on the same Ethernet is operated normally.

The following external devices are the target of communication status test.

- RJ71EN71 (when the Ethernet function is used)
- RnENCPU (network part) (when the Ethernet function is used)
- MELSEC-Q series Ethernet interface module (function version B or later)
- MELSEC-L series Ethernet interface module

This test is performed on the network of stations connected to the engineering tool, by sending the test messages sequentially to specified network and station number range. If the module that received the message returns a response, communication is normal.

"Ethernet Diagnostics" window ⇨ [Communication Status Test] button




**Restriction**

The communication status test cannot be used in the following cases.

- When the Ethernet function of the CPU module (CPU part for the RnENCPU) is used
- When the Ethernet diagnostics are started with a setting other than "No Specification" in "Other Station Setting" on the "Specify Connection Destination Connection" window

### ■Procedure

Set the required items in "Setting Item" and click the [Execute] button to execute the communication status test. The test results are displayed in the "Result" box.

## ■Action for abnormal end

If the communication status test fails, "No Response" or an error code is displayed in the "IP Address / Error Code" field in "Result".

| Displayed test result | Status of the external device | Cause | Action |
|---|---|---|---|
| No Response | No error | The initial process for the Ethernet-equipped module has not been completed normally. | Correct the module parameters. |
| | | There is an error in the line connection to the Ethernet-equipped module. (Cable disconnection, line disconnection, or others) | • Check the cable.<br>• Check the hub and gateway. |
| | | The IP address of the Ethernet-equipped module is incorrect. (The class or subnet address differs from that of the Ethernet-equipped module.) | Correct the module parameters. |
| | | The same IP address has been set to multiple Ethernet-equipped modules. | |
| | | The same network number or station number has been set to multiple Ethernet-equipped modules. | |
| | No error/Error exists | The Ethernet line is heavily loaded. | Perform the test again when the Ethernet line is not heavily loaded. |
| | Error exists | The routing settings are not configured. | Correct the set value for "Routing Setting" of "CPU Parameter". |
| Error code | No error | The "MELSOFT Transmission Port (UDP/IP)" for the Ethernet-equipped module is locked with the remote password. | Disable the remote password setting and write the parameters to the CPU module. |
| | | The target module does not support the communication status test. | Check the module name and function version. |
| | No error/Error exists | The Ethernet line is heavily loaded. | Perform the test again when the Ethernet line is not heavily loaded. |

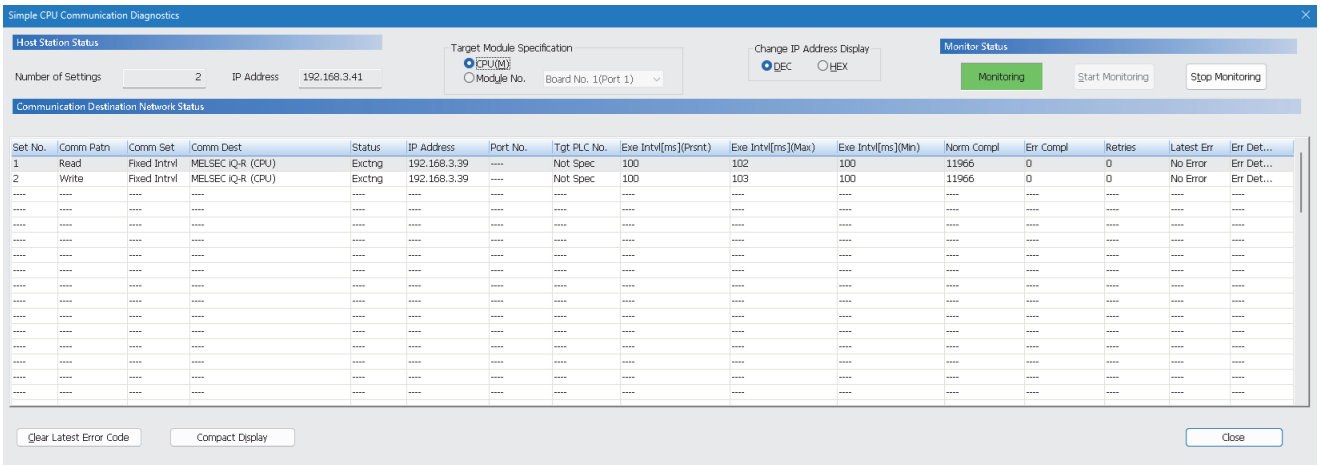
## ■Precautions

- The communication status test cannot be executed when "MELSOFT Transmission Port (UDP/IP)" for the target Ethernet-equipped module is locked with the remote password.
- When executing the communication status test, set "Gateway Other Than Default Gateway" under "Gateway Parameter Settings" in "Application Settings" to "Not Use".

# Simple CPU communication diagnostics

The communication status of the simple CPU communication function can be checked with "Simple CPU Communication Diagnostics".

[Diagnostics] ⇨ [Simple CPU Communication Diagnostics]




| Item | Description |
|---|---|
| Set No. | Displays the setting number of the simple CPU communication. |
| Comm Patn | Displays the communication pattern (read/write) set with the parameters. |
| Comm Set | Displays the communication setting (Fixed Interval/On Request) set with the parameters. |
| Tgt PLC No. | Displays the communication destination set with the parameters.<br>The destination is displayed when the simple CPU communication is executed with the RJ71EN71 or RnENCPU (network part). |
| Status | Displays the communication status ("Not Set", "Prepg", "Rqst Wtng", "Exctng", "Stpng", "Rtryng", "Mntrng", or "Comm Dis").<br>During system switching in a redundant system, "Prepg" is displayed. |
| IP Address | Displays the IP address of the communication destination set with the parameters. |
| Port No. | Displays the port number of the communication destination set with the parameters. |
| Tgt PLC No. | Displays the target PLC number set with the parameters. |
| Exe Intv [ms] (Prsnt)<br>Exe Intv [ms] (Max)<br>Exe Intv [ms] (Min) | Displays the current value, the maximum value, and the minimum value of the execution interval in increments of ms.<br>The values will not be updated during retry or monitoring. During communication stop, 0 is displayed on "Exe Intv [ms] (Prsnt)".<br>During system switching in a redundant system, 0 is displayed.<br>Displays "-----" when "On Request" is set to "Comm Set" in "Simple CPU Communication Settings" under "Application Setting". |
| Norm Compl<br>Err Compl<br>Retries | Displays the accumulated count of communications that have been completed successfully, completed with an error, and retried.<br>• 0 to 4294967295: Accumulated count<br>When the count exceeds 4294967295, counting is continued from 1 again. |
| Latest Err | Displays the error code when an error has occurred. Displays "No Error" when no error has occurred.<br>The error code of the last error remains displayed even after the communication status has changed to the state in which the communication can be performed normally.*1<br>When another error occurs, it will be overwritten with a new one. |
| Err Det... | Displays the descriptions and corrective actions for the error occurring in the selected setting number. |
| [Clear Latest Error Code] button | Clears the error code.<br>This operation also clears the error codes (Un\G1194 to Un\G1257) of the corresponding setting number in the CPU buffer memory. |
| (Own station) IP address | The IP address of the connected own station is displayed.<br>• The IP address set in "Own Node Settings" under "Basic Settings" in the RJ71EN71 or RnENCPU (network part) is displayed.<br>• When the RJ71EN71 is used in a redundant system, the control system IP address is displayed.<br>• For the Process CPU (redundant mode), the IP address of the connected system A or system B is displayed. |

*1 If the communication setting of the CPU module (built-in Ethernet port part) is set to "On Request", turn on Request to start communication at request. When the communication becomes normal, clear the last error code.

**Point**

If the engineering tool with an earlier version is used, the diagnostics results may not display correctly. For example, the execution interval may display as "0" when "Comm Set" is set to "On Request". To diagnose correctly, update the engineering tool version.

The simple CPU communication diagnostics can be started for stations of up to eight networks ahead (number of relay stations: 7) including the network to which the station directly connected with the engineering tool belongs.

In addition, when specifying a setting other than "No Specification" for "Other Station Setting" in the "Specify Connection Destination Connection" window, specify an Ethernet-equipped module to be diagnosed.

Unless the desired Ethernet-equipped module is specified, even if a different Ethernet-equipped module on the same base unit is specified, the simple CPU communication diagnostics cannot be started for the desired module.



## Precautions

### ■Communication paths through which simple CPU communication diagnostics cannot be started

If the communication path includes the following items, the simple CPU communication diagnostics cannot be started.

- MELSECNET/H
- Multidrop connection with a serial communication module
- Interface board for a personal computer
- GOT (when the transparent function is used)
- MELSEC-Q/L series network module

If any of the following conditions apply in the "Specify Connection Destination Connection" window, simple CPU communication diagnostics cannot be started.

- "Other Station (Single Network)" is specified in "Other Station Setting" and "CC-Link" or "C24" is specified in "Network Communication Route".
- "Other Station (Co-existence Network)" is specified in "Other Station Setting" and "CC-Link" or "C24" is specified in "Co-existence Network Route".

### ■Measures for redundant system configuration

In a redundant system, to specify the RJ71EN71 as the target module in the simple CPU communication diagnostics, directly connect the station to be diagnosed to the engineering tool and specify the module connected to the engineering tool ("Specify Redundant CPU" is set to "Not Specified") to start simple CPU communication diagnostics.

In addition, the simple CPU communication diagnostics cannot be started in the module diagnostics window when the system is switched between A and B in "Main Base Information" of "System Monitor".

In a redundant system of the remote head module, the simple CPU communication diagnostics cannot be started when the engineering tool is connected to the remote head module of the standby system. Connect it to the remote head module of the control system, and start the diagnostics.

### ■Display contents in redundant system configuration

Even if system switching is executed, "Norm Compl", "Err Compl", "Retries", "Latest Err", and "Err Det..." are not tracked and the data is stored in the system A and system B. Therefore, "Norm Compl", "Err Compl", and "Retries" is accumulated per system and "Latest Err" and "Err Det..." are different between the system A and system B.

# Error log area

Errors that occurred in each processing operation for data communications between the RJ71EN71 and RnENCPU (network part) and the external device can be checked with the buffer memory. For details, refer to the following.

☞ Page 506 Error log area

# 3.4 Troubleshooting by Symptom

This section describes troubleshooting method by symptom. If an error has occurred in the Ethernet-equipped module, identify the error cause using the engineering tool. (☞ Page 422 Checking the Module Status)

## Communications cannot be performed with the external device.

The following table lists how to troubleshoot the problem when the Ethernet-equipped module cannot communicate with the external device.

| Check item | Action |
|---|---|
| Is the READY LED of the CPU module off? | If the READY LED of the CPU module is off, perform the troubleshooting of the CPU module. (📖 Manual for the CPU module used) |
| Is the RUN LED of the RJ71EN71 or the RnENCPU (network part) off? | If the RUN LED of the RJ71EN71 or the RnENCPU (network part) is off, reset the CPU module. If the RUN LED is still off even after resetting the CPU module, the possible cause is a hardware failure. Replace the RJ71EN71 or RnENCPU and restart all the external devices that were performing communications with the own station.*1 |
| Is the Ethernet cable securely connected? | If the Ethernet cable is not securely connected, take following actions.<br>• Securely lock the cable.<br>• Check the wiring. (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup)) |
| Does the external device conform to the Ethernet standard? | If the device does not conform to the Ethernet standard, replace it with one conforming to the standard. (📖 MELSEC iQ-R Ethernet/CC-Link IE User's Manual (Startup)) |
| Is the external device powered on? | If the external device is not powered on, power it on. |
| Does any error occur in the external device, hub, router, or other devices? | If an error occurs, refer to the manual for each device and take action. |
| Does the protocol (TCP/IP or UDP/IP) match with that of the external device? | Correct the parameters so that the protocol matches with that of the external device. (☞ Page 329 External Device Configuration) |
| Is the access from the external device set to be denied in "IP Filter Settings" under "Security" of "Application Settings"? | If the access from the external device is denied, change the setting so that the access is allowed. (☞ Page 404 IP Filter Settings) |
| Was a device on the line (such as an Ethernet-equipped module, external device, hub, and router) replaced with the one with the same IP address? | If a device was replaced, take any of the following actions.<br>• Wait for update of the ARP cache (waiting time varies depending on the device)<br>• Restart all the devices on the line.*1 |
| Is the IP address of the external device specified correctly? | Execute a PING test to the IP address of the external device.<br>If the existence of the external device cannot be checked, network configuration is incorrect. Consult with the system or network manager. |
| Is "Module Operation Mode" of "Application Settings" set to a mode other than "Online"?*4 | Set "Module Operation Mode" of "Application Settings" to "Online". (☞ Page 415 Module Operation Mode) |
| Has the initial processing been completed normally? (Check 'Initial status' (Un\G1900024.0).)*3 | If the initial processing has not been completed successfully, correct the module parameters. (☞ Page 326 PARAMETER SETTINGS) |
| Does the communication data code (ASCII or binary) setting match with that of the external device? | If the communication data code settings do not match, data cannot be sent or received because the external device cannot normally decode commands.*2<br>Set the same data code as that of the external device in "Communication Data Code" under "Own Node Settings" of "Basic Settings". (☞ Page 327 Own Node Settings) |
| Is the target connection in forcibly invalidated state in the "Force Deactivation Status" column in the "Status of Each Connection" tab of the "Ethernet Diagnostics" window? | Turn off the area corresponding to the target connection in 'Forced connection invalidation setting area' (Un\G5646 to Un\G5650) to cancel the forced invalidation. |
| Is the access from the Ethernet-equipped module denied by the security setting of the external device? | Correct the security setting (such as firewall) of the external device. |
| For the RJ71EN71 or the RnENCPU (network part), is 'IP address change function operating status' (Un\G11715.0) on? | When 'IP address change function operating status' (Un\G11715.0) is on, the module is operating according to the settings of the IP address change function. Either clear the IP address storage area, or correct the settings. (☞ Page 293 IP Address Change Function) |
| Does a timeout occur when connecting to the engineering tool? | Increase the value for "Check at Communication Time" and "Retry Times" in the following windows in the "Specify Connection Destination Connection" window and connect again.<br>• "Host Station Detailed Setting" window of "No Specification"<br>• "Other Station Detailed Setting" window of "Other Station (Single Network)" or "Other Station (Co-existence Network)" |

| Check item | Action |
|---|---|
| When connecting to an engineering tool, use [Ethernet Diagnostics] ⇨ [TCP Status] to check if the connection to be established is "Connected". | Another user may be connected, so wait for the connection to be disconnected or try to connect using UDP/IP. If no connection is familiar, turn on the target connection area in the forced connection invalidation setting area, and execute forced invalidation. (☞ Page 531 Forced connection invalidation setting area) |

When the problem cannot be solved with above actions, check the error cause according to the communication processing and take action.

- Communications using the SLMP (☞ Page 440 Communications using the SLMP cannot be performed.)
- Communications using the predefined protocol (☞ Page 441 Communications using the predefined protocol cannot be performed.)
- Socket communications/Communications using the fixed buffer (☞ Page 443 Socket communications/communications using the fixed buffer cannot be performed.)
- Communications using the random access buffer (☞ Page 442 Communications using the random access buffer cannot be performed.)
- Communications using link dedicated instructions (☞ Page 445 Communications using a link dedicated instruction cannot be performed.)
- File transfer function (FTP server) (☞ Page 445 Access using the file transfer function (FTP server) cannot be performed.)

*1 A device on Ethernet has a table of IP addresses and their corresponding MAC address, called "ARP cache". When a device on the line is replaced by the one having the same IP address, the MAC address in the ARP cache is different from that of the replaced device; therefore, communications may not be normally performed. The "ARP cache" is updated by resetting the device or after a while. The time varies depending on the device.

*2 If the communication data code setting is different between the Ethernet-equipped module and the external device, error codes that are not found in the error code list may be returned to the external device. When receiving data with different data codes, the Ethernet-equipped module cannot decode commands normally. The Ethernet-equipped module returns an error response according to the communication data code setting.

*3 The buffer memory address is for when the network type of the RJ71EN71 or the RnENCPU (network part) is set to "Ethernet" and the P1 connector is used.
When using the P2 connector or the CPU module (built-in Ethernet port part), or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*4 Check this item only for the modules that have the relevant setting in the module parameters.

# A message sent from the external device cannot be received frequently.

When the Ethernet-equipped module frequently fails to receive a message sent from the external device, check the following items.

| Check item | Action |
|---|---|
| Are errors frequently detected in 'Simultaneous transmission error detection count (receive buffer full count)' (Un\G5022 to Un\G5023)?*1 | If many errors are detected or a system error is stored, there may be a heavy load on the Ethernet line due to data sending and receiving between connected devices.<br>• Taking action such as separating networks or decreasing the number of data sending reduces the load on the Ethernet line.<br>• Consult with the network manager and reduce the load on the Ethernet line. |
| Has a system error been stored in 'Latest error code' (Un\G100 to Un\G163 (Connection No.1 to No.64 latest error code))?*1 | |
| Has 1H been stored in 'State of receive buffer' (Un\G1900030)?*3 | • Taking action such as separating networks or decreasing the number of data sending reduces the load on the Ethernet line.<br>• When communications using the fixed buffer is used, check whether the BUFRCV instruction is executed. (☞ Page 77 Receive procedure)<br>• To receive data at shorter intervals than the scan time of the CPU module using the fixed buffer, add the normally closed contact for the completion device of the BUFRCV instruction to the execution conditions of the BUFRCV instruction. (☞ Page 95 Labels) |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*2

*1 The buffer memory address is for when the network type of the RJ71EN71 or the RnENCPU (network part) is set to "Ethernet" and the P1 connector is used.
When using the P2 connector or the CPU module (built-in Ethernet port part), or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*2 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

*3 Availability for the buffer memory depends on the firmware version. (☞ Page 586 Added and Enhanced Functions)

# Direct connection to the engineering tool is not possible.

When the direct connection to the engineering tool is not possible, check the following items.

| Check item | Action |
|---|---|
| Is the Ethernet-equipped module directly connected to the engineering tool with a single cable? | Directly connect the Ethernet-equipped module to the engineering tool with a single cable. |
| Is "Disable Direct Connection with MELSOFT" under "Security" of "Application Settings" is set to "Disable"? | If "Disable Direct Connection with MELSOFT" is set to "Disable", change the setting to "Enable". |
| Is the transfer setup setting of the engineering tool correct? | If the setting is incorrect, check the following items.<br>• Check that "Ethernet Board" is selected in "PC side I/F".<br>• Check that "Ethernet Port Direct Connection" is selected in the "PLC side I/F Detailed Setting of PLC Module" window or "PLC side I/F Detailed Setting of Ethernet Module" window of "PLC side I/F".<br>• When "PLC side I/F" is "Ethernet Module", check that "Other Station (Single Network)" is selected for "Other Station Setting" and "Access to Ethernet module set on PLC side I/F" is selected in the "Network Communication Route Detailed Setting of Ethernet" window of "Network Communication Route". |
| Does a timeout occur when connecting to the engineering tool? | Increase the value for "Check at Communication Time" and "Retry Times" in the "Other Station Detailed Settings" window of "Other Station Setting" of the "Specify Connection Destination Connection" window and connect again. |
| Does the device (personal computer) to be connected support the direct connection? | If the device does not support the direct connection, connect a hub. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*1

*1 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

# CPU module search on the network cannot be performed.

When the CPU module search on the network cannot be performed, check the following items.

| Check item | Action |
|---|---|
| Is "Do Not Respond to CPU Module Search" under "Security" of "Application Settings" is set to "Do Not Respond"? | Set "Do Not Respond to CPU Module Search" under "Security" of "Application Settings" to "Respond". |
| Is the CPU module cannot be searched connected to a router? | The CPU module connected via a router cannot be searched.<br>Modify the connection to search the CPU module. |
| Is the CPU module cannot be searched connected with wireless LAN? | Perform the search again. (The packet may be lost.) |
| Is the service processing (such as file read) load of the search target CPU module high? | • Increase the response waiting time and perform the search again.<br>• Reduce the service processing load of the CPU module. |

# Communications using the SLMP cannot be performed.

When communications using the SLMP cannot be performed, check the following items.

| Check item | Action |
|---|---|
| Has the connection with the external device been opened? (Check 'Open completion signal' (Un\G1900000 to Un\G1900007).)*1*2 | • If the connection with the external device is not opened, perform the open processing.<br>• If an error occurs, check and eliminate the error cause. |
| Is the correct command format used for the command type, device, address, and others? | Correct the command format. (📖 SLMP Reference Manual) |
| Is "Enable/Disable Online Change" under "Own Node Settings" in "Basic Settings" set to "Disable All (SLMP)"? | If "Enable/Disable Online Change" is set to "Disable All (SLMP)", change the setting to "Enable All (SLMP)". (☞ Page 327 Own Node Settings) |
| Did the external device send a command? | If the device did not send a command, send a command to the Ethernet-equipped module. |
| Was a response returned to the device that had sent the command? | • If no response was returned, check if the correct IP address was specified in the command. If not, correct the IP address and send the command again.<br>• If a response was returned, check the end and error codes to correct the error. |
| Is the communication method of the connection used for communication with the external device is set to an item other than "SLMP" in the "External Device Configuration" window of "Basic Settings"? | Correct the communication method setting if an item other than "SLMP" is set. (☞ Page 329 External Device Configuration) |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*3

*1 The buffer memory address is for when the network type of the RJ71EN71 or the RnENCPU (network part) is set to "Ethernet" and the P1 connector is used.
When using the P2 connector or the CPU module (built-in Ethernet port part), or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*2 If the connection of only the external device is closed due to cable disconnection, personal computer restart, or other reasons, reopen the connection using the same port used before the error occurred. The E71 does not close a connection if it receives an Active open request again from the external device with a different IP address or a port number.

*3 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

# Communications using the predefined protocol cannot be performed.

When communications using the predefined protocol cannot be performed, check the following items and take action.

| Check item | Action |
|---|---|
| Is the IP address setting of the external device correct? | Check and correct the IP address of the external device. |
| Has the connection with the external device been opened? (Check 'Open completion signal' (Un\G1900000 to Un\G1900007).)*1*2 | • If the connection with the external device is not opened, perform the open processing.<br>• If an error occurs, check and eliminate the error cause. |
| Has the send/receive processing (dedicated instruction) for the communications using the predefined protocol been executed properly? | • If the send/receive processing has not been executed, check and correct the execution condition of the send/receive processing.<br>• If send/receive processing has completed with an error, check the error code in the completion status and eliminate the error cause. |
| Is the appropriate protocol used for the external device in the predefined protocol settings and program? | Correct the predefined protocol settings and program If inappropriate protocol is used.<br>For appropriate protocol for the external device, refer to the manual for the device used. |
| Is the communication method of the connection used for communication with the external device is set to an item other than "Predefined Protocol" in the "External Device Configuration" window of "Basic Settings"? | Correct the communication method setting if an item other than "Predefined Protocol" is set. (☞ Page 329 External Device Configuration) |
| Is 'Predefined protocol ready' (Un\G1901002.0) on?*1 | If 'Predefined protocol ready' (Un\G1901002.0) is off, write the protocol setting data to the module. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*3

*1 The buffer memory address is for when the network type of the RJ71EN71 or the RnENCPU (network part) is set to "Ethernet" and the P1 connector is used.
When using the P2 connector or the CPU module (built-in Ethernet port part), or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*2 If the connection of only the external device is closed due to cable disconnection, personal computer restart, or other reasons, reopen the connection using the same port used before the error occurred. The E71 does not close a connection if it receives an Active open request again from the external device with a different IP address or a port number.

*3 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

# Communications using the random access buffer cannot be performed.

When communications using the random access buffer cannot be performed, check the following items and take action.

| Check item | Action |
|---|---|
| Has the connection with the external device been opened? (Check 'Open completion signal' (Un\G1900000 to Un\G1900007).)*1*2 | • If the connection with the external device is not opened, perform the open processing.<br>• If an error occurs, check and eliminate the error cause. |
| Did the external device send a command? | Send a command to the Ethernet-equipped module from the external device. |
| Is the correct buffer memory address specified for the command? | If the specified buffer memory address is incorrect, correct it and send the command again. |
| Has data been set in the specified address of the random access buffer? | If any data has not been set, write the data. |
| Has the write data been set in the external device? | If any write data has not been set, set the data. |
| Was a response returned to the device that had sent the command? | • If no response was returned, check if the correct IP address was specified in the command. If not, correct the IP address and send the command again.<br>• If a response was returned, check the end and error codes to correct the error. |
| Is the communication method of the connection used for communication with the external device is set to an item other than "Random Access Buffer" in the "External Device Configuration" window of "Basic Settings"? | Correct the communication method setting if an item other than "Random Access Buffer" is set. (☞ Page 329 External Device Configuration) |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)

*1 The buffer memory address is for when the network type of the RJ71EN71 or the RnENCPU (network part) is set to "Ethernet" and the P1 connector is used.
When using the P2 connector or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*2 If the connection of only the external device is closed due to cable disconnection, personal computer restart, or other reasons, reopen the connection using the same port used before the error occurred. The E71 does not close a connection if it receives an Active open request again from the external device with a different IP address or a port number.

# Socket communications/communications using the fixed buffer cannot be performed.

When socket communications or communications using the fixed buffer cannot be performed, check the following items and take action.

| Check item | Action |
|---|---|
| Has the connection with the external device been opened? (Check 'Open completion signal' (Un\G1900000 to Un\G1900007).)*1*2 | • If the connection with the external device is not opened, perform the open processing.<br>• If an error occurs, check and eliminate the error cause. |
| Has the send/receive processing been executed properly? | • If the send/receive processing has not been executed, check and correct the execution condition of the send/receive processing.<br>• If send/receive processing has completed with an error, check the error code in the completion status and eliminate the error cause. |
| Does 'Socket/fixed buffer reception status signal' (Un\G1900016 to Un\G1900023) turn on when receive processing is executed? | • Send data from the external device.<br>• If data has been sent, check that the destination address (IP address and port number) is correctly set. |
| Is the communication method of the connection used for communication with the external device is set to an item other than the following in the "External Device Configuration" window of "Basic Settings"?<br>• Socket communications: "Socket Communication"<br>• Communications using the fixed buffer: "Fixed Buffer (Procedure Exist)" or "Fixed Buffer (No Procedure)" | Correct the communication method setting if an item other than the one described in the left is set. (☞ Page 329 External Device Configuration) |
| Is the communication method of the connection used for communication with the external device is set to an item other than "Broadcast Send" or "Broadcast Receive" in the "External Device Configuration" window of "Basic Settings" when broadcast communications is performed? | Correct the communication method setting if an item other than the one described in the left is set. (☞ Page 329 External Device Configuration) |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*3

*1 The buffer memory address is for when the network type of the RJ71EN71 or the RnENCPU (network part) is set to "Ethernet" and the P1 connector is used.
When using the P2 connector or the CPU module (built-in Ethernet port part), or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*2 If the connection of only the external device is closed due to cable disconnection, personal computer restart, or other reasons, reopen the connection using the same port used before the error occurred. The E71 does not close a connection if it receives an Active open request again from the external device with a different IP address or a port number.

*3 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

# Communications using MODBUS/TCP cannot be performed.

## Communication fails at MODBUS/TCP connection.

When communication is not possible at MODBUS/TCP connection, check the following items and take action.

For troubleshooting of the MODBUS/TCP master function, refer to Page 448 When the simple CPU communication function is used.

| Check item | Action |
|---|---|
| Does the external device support MODBUS/TCP? | If the device does not support MODBUS/TCP, replace it with the one supporting the MODBUS/TCP standard. |
| Is the external device a MODBUS/TCP master device? | The external device must be a MODBUS/TCP master device. |
| Was a new connection attempted when 64 external devices have already been connected? | Reduce the number of external devices connected to the Ethernet-equipped module.<br>If the number of connections with external devices exceeds 64, the connection is disconnected automatically from the device for which the longest time has elapsed from the last communication. |
| Is the Ethernet-equipped module supporting the MODBUS/TCP function used? | Check the firmware version of the Ethernet-equipped module and ensure that the module supports the MODBUS/TCP function.<br>When the Ethernet-equipped module being used does not support the MODBUS/TCP function, replace it with the module that supports the function. |

## Messages sent from the MODBUS/TCP master device cannot be received.

When messages sent from the MODBUS/TCP master device cannot be received completely or frequently, check the following items and take action.

| Check item | Action |
|---|---|
| Are there times when 'State of receive buffer' (Un\G1900030) becomes 1 (Receive buffer full)? | The load on the Ethernet line may high due to the data communication between the nodes. Take the following actions.<br>• Reduce the load on the Ethernet line by separating networks, decreasing the number of data sending, or other method.<br>• Consult with the network manager and reduce the load on the Ethernet line. |

## A response message cannot be sent to the MODBUS/TCP master device.

When a response message to the request message sent from the MODBUS/TCP master device or a supposed response message cannot be sent, check the following items and take action.

| Check item | Action |
|---|---|
| Is the registration of the MODBUS device assignment parameters correct? | Check the settings of the MODBUS device assignment parameters. |
| Was an abnormal response code sent? | Check the abnormal response code and correct any possible problems. |
| Is there any error code stored in 'Error log area for MODBUS/TCP function' (Un\G720896 to Un\G721153)? | Take an appropriate action in accordance with the stored error code.<br>(☞ Page 453 List of Error Codes) |
| Is the request message received from the MODBUS/TCP master device correct? | Correct the request message that the MODBUS/TCP master device sends. |
| Is the message length of the application data area of the Ethernet frame six bytes or longer (from communication ID to message length of the MBAP header)? | |
| Does the message length of the MBAP header + 6 bytes match with the message length of the application data area of the Ethernet frame? | |

# A dedicated instruction is not completed.

If the dedicated instruction is not completed when using the RJ71EN71 or the RnENCPU (network part), check if an option other than "Online" is set in "Module Operation Mode" under "Application Settings".
Set "Online".

# Communications using a link dedicated instruction cannot be performed.

When communications using a link dedicated instruction cannot be performed, check the following items and take action.

| Check item | Action |
|---|---|
| Are the network number and station number set in the Ethernet-equipped module? | Set the network number and station number. |
| Is the number of resends set upon every instruction execution? | Modify the program to set the number of resends upon every instruction execution. |
| Are station numbers unique? | Change the duplicated station number. |
| Is the network number duplicated on the network? | Correct the parameters so than the network number is not overlapped. |
| Is a communication path by the dynamic routing function is determined? | • Power on the system and start transient transmission after a while.<br>• When "Dynamic Routing" under "Application Settings" of the station on the communication path is set to "Disable", change it to "Enable". |
| Does the relay station to be passed support the dynamic routing function? | If the relay station to be passed does not support the dynamic routing function, set all the stations on the communication path in "Routing Setting" of "CPU Parameter". |
| Are multiple link dedicated instructions with same channel setting executed simultaneously? | • Set different channel to each instructions.<br>• Shift the execution timing of the link dedicated instructions. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)

# Access using the file transfer function (FTP server) cannot be performed.

When the Ethernet-equipped module cannot be accessed from the external device (FTP client) using the file transfer function (FTP server), check the following items and take action.

| Check item | Action |
|---|---|
| Is the file transfer function (FTP server) for the Ethernet-equipped module enabled? | Set "FTP Server" under "FTP Server Settings" of "Application Settings" to "Use". (☞ Page 338 FTP Server Settings) |
| Are the login name and password correct? | Check the login name and password set in the parameters, and log in again. (☞ Page 338 FTP Server Settings) |
| Is the IP address entered at login correct? | Check the IP address of the Ethernet-equipped module set in the parameters, and log in again. (☞ Page 327 Own Node Settings) |
| Is executed command used properly? | Check the instructions of the command and use the command properly. (☞ Page 122 FTP command) |
| In multiple CPU system, is the access destination CPU module correct? | Execute the cpuchg command and change the access destination CPU module. |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*1

*1 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

# Access using the file transfer function (FTP client) cannot be performed.

When the CPU module cannot access to the external device (FTP server) during execution of the file transfer function (FTP client), check the following items and take action.

| Check item | Action |
|---|---|
| Is a value other than 0000H stored in (s1)+1 (Completion status) of the SP.FTPPUT or SP.FTPGET instruction? | Take action for the error code stored in (s1)+1 (Completion status), and execute the instruction again. (☞ Page 453 List of Error Codes) |

# The time setting function (SNTP client) cannot be performed normally.

When the time setting function (SNTP client) cannot be performed normally, check the following items and take action.

| Check item | Action |
|---|---|
| Is the time setting function (SNTP client) is enabled for the Ethernet-equipped module? | Set "Time Setting (SNTP Client)" under "Time Setting" of "Application Settings" to "Use". (☞ Page 400 Time Setting) |
| Is "SNTP Server IP Address" set correctly? | Change the "SNTP Server IP Address" under "Time Setting" in "Application Settings" to the correct IP address. (☞ Page 400 Time Setting) |

If the above action does not solve the problem, perform the module communication test to check for hardware failure. (☞ Page 424 Module communication test)*1

*1 The module communication test cannot be performed for the CPU module (built-in Ethernet port part). Perform the troubleshooting of the CPU module and check for hardware failure. (📖 Manual for the CPU module used)

# The access cannot be allowed/denied correctly by the IP filter.

When the access cannot be allowed/denied correctly by the IP filter, check the following items and take action.

| Check item | Action |
|---|---|
| Is "Access from IP address below" correctly set in "IP Filter Settings" under "Security" of "Application Settings"? | Correct the setting of "Access from IP address below". |
| Are the IP addresses set in "IP Address" in "IP Filter Settings" under "Security" of "Application Settings"? | Correct the IP addresses.<br>When the IP addresses are specified by range, check also "IP Address Excluded from Range". |
| Is the IP address of the proxy server set to be allowed? | Set the IP address of the proxy server to be denied. |

# Remote password does not work.

When the remote password does not work, check the following items and take action.

| Check item | Action |
|---|---|
| Is the remote password setting of the target connection disabled in the "Remote Password Detail Setting" window? | If the remote password is disabled, change the setting so that it is enabled. (☞ Page 178 Remote password) |

# Communications using the built-in Ethernet port are slow.

When communications using the built-in Ethernet port are slow, check the following items and take action.

| Check item | Action |
|---|---|
| Is the communication request load from the external device high? | Check the buffer memory Un\G100 (receive packet count (current) per unit time (one second)) of the CPU module (built-in Ethernet port part), and if the value is significantly large compared to the normal communications, the following actions may improve the processing speed.<br>• Review the communication request from the engineering tool and GOT.<br>• Review the communication request from the transfer destination (execution source) of the simple CPU communications.<br>• Review the file transfer function (FTP server).<br>• Review the communications using the SLMP. |
| Is the communication request load from the CPU module high? | Check the buffer memory Un\G102 (send packet count (current) per unit time (one second)) of the CPU module (built-in Ethernet port part), and if the value is significantly large compared to the normal communications, the following actions may improve the processing speed.<br>• Review the execution frequency of the instructions for the built-in Ethernet function (socket communications instruction/predefined protocol support function instruction/SLMP frame send instruction/file transfer function instruction)<br>• Review the settings (such as the number of points) of CC-Link IE Field Network Basic.<br>• Review the settings (such as the execution interval and the number of points) of the simple CPU communications.<br>• Review the settings (such as automatic detection and backup execution) of iQSS (built-in Ethernet).<br>• Review the settings (such as the number of executions) of data logging file transfer. |

# When the simple CPU communication function is used

## Communications cannot be performed with the communication destination.

When communications with the external device cannot be performed using the simple CPU communication function, check the following items and take action.

| Check item | Action |
|---|---|
| Does any error occur in the simple CPU communication diagnostics or simple CPU communication error code in the buffer memory? | Eliminate the error cause. When communications can be performed by increasing the communication retry count and communication time-out period, review the communication retry count and communication time-out period, or check the line status of Ethernet communications with the external device, such as a cable, hub, and router, to check if the line is busy. |
| Is the communication status of the simple CPU communication diagnostics or the simple CPU communication status of the buffer memory "preparing"? | Check the cable between the Ethernet-equipped module and hub, and check if the hub is operating normally. |
| Is the module supporting the simple CPU communication function used? | Refer to the following page and check the firmware version of the module that can use a desired device as a communication destination. (☞ Page 226 Restrictions applicable depending on versions)<br>In "Product Information List" of "System Monitor", check the firmware version of the target module, and ensure that the module supports the simple CPU communication.<br>When the module used does not support the simple CPU communication, update the firmware to the version supporting the function or replace the module with a module that supports the function. |
| Is the own station port number set in the external device configuration within the range of 61696 to 65534? | When the communication destination of the simple CPU communication is a Mitsubishi Electric programmable controller, the own station port number is dynamically set within the range of 61696 to 65534.<br>When the own station port number is set within the range of 61696 to 65534 in the external device configuration, the port numbers overlap, resulting in a communication failure. Set the own station port number to a number other than 61696 to 65534 in the external device configuration. |
| Has the number of connections reached the upper limit in the external device? | Reduce the number of used connections by sharing a TCP connection with multiple communication settings, or through other means.<br>Setting the own station port numbers to the same value allows for the sharing of one TCP connection. |

If communications cannot be performed despite the check for the above items, the hardware failure may have been occurred. Please consult your local Mitsubishi representative.

## Data cannot be read or written in the specified way.

If data cannot be read or written in the specified way by using the simple CPU communication function, check the following items and take action.

| Check item | Action |
|---|---|
| Is the set module type applicable to the module used? | Refer to the following page and check the firmware version of the module that can use the set module type. (☞ Page 226 Restrictions applicable depending on versions)<br>In "Product Information List" of "System Monitor", check the firmware version of the target module, and ensure that the module supports the simple CPU communication.<br>When the module used does not support the simple CPU communication, update the firmware to the version supporting the function or replace the module with a module that supports the function. |
| Are the settings of the communication destination correct? | Check that the module type and the range set in "Simple CPU Communication Setting" match with the settings of the communication destination. |

## Communications with the communication destination are unstable.

When communications with the communication destination using the simple CPU communication is unstable, check the following items and take action.

| Check item | Action |
|---|---|
| Does any error occur in the simple CPU communication diagnostics or simple CPU communication error code in the CPU buffer memory? | Eliminate the error cause. |
| Is the retry of communications performed? | Check the line status of Ethernet communications with the external device, such as a cable, hub, and router, to check if the line is busy. |

## Communications cannot be performed at the set execution interval.

When communications cannot be performed at the set execution interval in the simple CPU communication function, check the following items and take action.

| Check item | Action |
|---|---|
| Is another Ethernet function executed simultaneously in the CPU module where the simple CPU communication function is set? | Review the Ethernet function executed simultaneously in the CPU module where the simple CPU communication function is set. |
| Is the service processing load of the CPU module of the communication destination high? | • When the communication destination is the model for which service processing setting is available, increase the number of times and the time of service processing of the communication destination.<br>• If the communication destination supports the COM(P) instruction and CCOM instruction, incorporate the COM(P) instruction and CCOM instruction into the sequence program of the communication destination.<br>• Review the execution interval. |
| Is the Ethernet communication load high? Or, is the communication quality low? | Check the line status of Ethernet communication with the external device, such as a cable, hub, and router, to reduce the load and improve the communication quality. (Check the buffer memory Un\G100/Un\G101 (receive packet count per unit time (one second)) or Un\G102/Un\G103 (send packet count per unit time (one second)) of the CPU module (built-in Ethernet port part) and reduce the load and improve the communication quality.) |
| Is the response from the SLMP-compatible device delayed? (When the SLMP-compatible device (QnA-compatible 3E frame) is used) | • Check the function execution status of the communication destination (SLMP-compatible device).<br>• Review the execution interval. |

## The update by reading and writing the device data is slow.

When the update by reading and writing the device data is slow in the simple CPU communications, check the following items and take action.

| Check item | Action |
|---|---|
| The scan time of the CPU module where the simple CPU communication function is set is long. | Incorporate the COM(P) instruction and CCOM instruction into the sequence program of the CPU module where the simple CPU communication function is set. |

# When the simple device communication function is used

## Unable to communicate with the communication destination.

When communications with the communication destination cannot be performed using the simple device communication function, check the following items and take action.

| Check item | Action |
|---|---|
| Is the module supporting the simple device communication function used? | In "Product Information List" of "System Monitor", check the firmware version of the target module, and ensure that the module supports the simple device communication.<br>When the module being used does not support the simple device communication, update the firmware that supports the function or replace the module with a module that supports the function. |
| Is the IP address setting of the communication destination correct? | Check the IP address of the communication destination, and correct it if incorrect. |
| Is 'Status of connection' (Un\G1249309) 3H (communicating)? | • If the status is 1H (preparing), wait until the latency time has elapsed.<br>• If the status is 2H (waiting for the request) or 4H (communication stop), turn on 'Function start/stop instruction' (Un\G1249300.0) and 'Start request for each protocol' (Un\G1249348 to Un\G1249379).<br>• If the status is 5H (retry being executed) or 6H (monitoring at error), wait until "Timeout Time" or "Monitoring Time At Error" has elapsed and then check the status again. |
| Is the Ethernet cable securely connected? | Connect the Ethernet cable correctly.*1 |
| Is the appropriate protocol used for the communication destination in the protocol settings? | If the protocol for the communication destination is inappropriate, correct it to the appropriate protocol.<br>For protocols, refer to the manual for the communication destination. |

*1 When 'Status of connection' (Un\G1249309) is 5H (retry being executed) at reconnection of the Ethernet cable, the time until the communication returns to the normal state may results in ((Retry Count + 1) × "Timeout Time") + ("Monitoring Time At Error" × 2) at the maximum.
When 'Status of connection' (Un\G1249309) is 6H (monitoring at error), the time until the communication returns to the normal state may results in ("Monitoring Time At Error" × 2) at the maximum.

## Verification mismatch occurs even when the response is received.

When the verification mismatch occurs even when the response from the communication destination is received using the simple device communication function, check the following items and take action.

| Check item | Action |
|---|---|
| Check 'Verification mismatch protocol number' (Un\G1249317) and 'Verification mismatch information' (Un\G1249318 to Un\G1249333). | Check the element number with which the verification mismatch has occurred and the cause of the verification mismatch, and correct the setting details of the element that causes the mismatch. |
| Check the setting details of "Clear Receive System Area" of "Send/Receive Parameter Setting". | If "No" is set, change the setting to "Yes". |

# Data communications cannot be performed with the device supporting iQ Sensor Solution.

When communications with the device supporting iQ Sensor Solution cannot be performed, check the following items and take action.

| Check item | Action |
|---|---|
| In "iQ Sensor Solution Settings" under "Application Settings", is "Use" set for "iQ Sensor Solution Settings" under "Application Settings"? | If "Not to Use" is set, change the setting to "Use". |

# When the redundant system function is used

## Communications cannot be performed with the external device.

When communications with the external device cannot be performed using the redundant system function, check the following items and take action.

| Check item | Action |
|---|---|
| Are the control system IP address and the IP addresses of systems A and B used together for the same connection? | Set the different connections for the each of control system IP address and IP addresses of systems A and B. |

## System switching cannot be performed normally.

When system switching cannot be performed normally by the redundant system function, check the following items and take action.

| Check item | Action |
|---|---|
| Is the power supply module, CPU module, or redundant function module of the standby system operating normally? | Perform troubleshooting on the power supply module, CPU module, or redundant function module of the standby system. (📖 Manual for the CPU module used) |
| Has a system switching request been issued to the CPU module? (Check 'Module number in the system switching request issued from the own system network module' (SD1645) of the control system CPU module.) | Check an error on the Ethernet-equipped module mounted on the control system CPU module, and correct the error. |
| In "Redundant System Settings" under "Application Settings", is "Enable" selected for "System Switching Request Issue at Communication Error"? | In "Redundant System Settings" under "Application Settings", select "Enable" for "System Switching Request Issue at Communication Error". |
| For "User Connection" or "System Connection" of "System Switching Request Issue at Communication Error" in "Redundant System Settings" under "Application Settings", is it specified that a system switching request is issued in the target connection? | For "User Connection" or "System Connection" of "System Switching Request Issue at Communication Error" in "Redundant System Settings" under "Application Settings", specify that a system switching request is issued in the target connection. |
| Do the settings of the engineering tool and the buffer memory details match? | Write the set parameters to the CPU module, and reset the CPU module or power off and on the system. |
| Has a cable disconnection been detected? | Check that there is no error in the external device. |
| In "Redundant System Settings" under "Application Settings", is "Enable" selected for "System Switching Request Issuing at Disconnection Detection"? | In "Redundant System Settings" under "Application Settings", select "Enable" for "System Switching Request Issuing at Disconnection Detection". |
| In "Redundant System Settings" under "Application Settings", is the value set for "Disconnection Detection Monitoring Time" correct? | In "Redundant System Settings" under "Application Settings", correct the value set for "Disconnection Detection Monitoring Time". |
| In "Timer Settings for Data Communication" under "Application Settings", is the value set for "TCP ULP Timer" correct? | In "Timer Settings for Data Communication" under "Application Settings", correct the value set for "TCP ULP Timer". |
| In "External Device Configuration" under "Basic Settings", is "Existence Confirmation" set to "KeepAlive"? | In "External Device Configuration" under "Basic Settings", set "Existence Confirmation" to "KeepAlive". |
| In "Timer Settings for Data Communication" under "Application Settings", is the value set for "Destination Alive Check Start Interval Timer" correct? | In "Timer Settings for Data Communication" under "Application Settings", correct the value set for "Destination Alive Check Start Interval Timer". |
| In "Timer Settings for Data Communication" under "Application Settings", is the value set for "Destination Alive Check Interval Timer" correct? | In "Timer Settings for Data Communication" under "Application Settings", correct the value set for "Destination Alive Check Interval Timer". |
| In "External Device Configuration" under "Basic Settings", is "Communication Method" set to "Broadcast Send"? | In "External Device Configuration" under "Basic Settings", set "Communication Method" to an option other than "Broadcast Send", and set "IP Address" in "Sensor/Device". |
| Has the connection with the external device been opened? (Check 'Open completion signal' (Un\G1900000 to Un\G1900007).)*1*2 | • If the connection with the external device is not opened, perform the open processing.<br>• If an error occurs, check and eliminate the error cause. |
| For configuration using the redundant module group setting, are settings made for "Redundant Module Group Setting" in the [I/O Assignment] tab of the "System Parameter" window? | Correct the settings for "Redundant Module Group Setting" in the [I/O Assignment] tab of the "System Parameter" window. |
| For configuration using the port group setting, are settings made for "Port Group Setting" in "Redundant System Settings" under "Application Settings"? | In "Redundant System Settings" under "Application Settings", correct the settings for "Port Group Setting". |

*1 The buffer memory address is for when the network type of the RJ71EN71 is set to "Ethernet" and the P1 connector is used. When using the P2 connector or the CPU module (built-in Ethernet port part), or setting the network type to "Q Compatible Ethernet", refer to the following.
☞ Page 486 Buffer Memory

*2 If the connection of only the external device is closed due to cable disconnection, personal computer restart, or other reasons, reopen the connection using the same port used before the error occurred. The E71 does not close a connection if it receives an Active open request again from the external device with a different IP address or a port number.

## Communication fails at OPS connection.

When communication using the redundant system function is not possible at the time of an OPS connection, check the following items and take action.

| Check item | Action |
|---|---|
| In "External Device Configuration" under "Basic Settings", is "OPS Connection Module" set for the external device? | Correct the settings for "External Device Configuration" under "Basic Settings". |
| In "External Device Configuration" under "Basic Settings", is the port number setting correct? | |
| In "External Device Configuration" under "Basic Settings", is the external device IP address setting correct? | |

# 3.5 List of Error Codes

This section lists the error codes, error details and causes, and action for the errors occur in the processing for data communication between the Ethernet-equipped module and slave stations or caused by processing requests from the CPU module on the own station.

Error codes are classified into major error, moderate error, and minor error, and can be checked in the [Error Information] tab in the "Module Diagnostics" window of an Ethernet-equipped module. (☞ Page 422 Error Information)



| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| 1080H | The number of writes to the flash ROM has exceeded 100000. | Replace the module. | — |
| 1120H | Clock setting has failed when the system is powered on or the CPU module is reset. | • Check if the time settings are correctly set in parameter.<br>• Check if the specified SNTP server is operating normally and there is no failure on the network accessing to the SNTP server computer. | — |
| 1124H | • The default gateway is not set correctly.<br>• The gateway IP address is not set correctly.<br>• The default gateway/gateway IP address (network address after the subnet mask) is different from that of the IP address of the own node. | • Correct the default gateway IP address.<br>• Set the same network address as that of the IP address. | Parameter information<br>• Parameter type<br>• I/O No.<br>• Parameter No.<br>• Network No.<br>• Station No. |
| 1128H | The port number is incorrect. | Correct the port number. | — |
| 1129H | The port number of the external device is not set correctly. | Correct the port number of the external device. | — |
| 112CH | The request using all stations specification has failed. | Try the request using current station specification. | — |
| 112DH | The data was sent to the external device while the IP address setting of the device set in "External Device Configuration" under "Basic Settings" was incorrect. | • Correct the IP address of the external device in "External Device Configuration" under "Basic Settings".<br>• Check that the IP address class of the external device is set to A/B/C in "External Device Configuration" under "Basic Settings". | — |
| 112EH | A connection could not be established in the open processing. | • Check the operation of the external device.<br>• Check if the open processing has been performed in the external device.<br>• Correct the port number of the Ethernet-equipped module, IP address/port number of the external device, and opening method.<br>• When the firewall is set in the external device, check if the access is permitted.<br>• Check if the Ethernet cable is disconnected. | — |
| 1133H | The response send failed during socket communications or communications using the fixed buffer. | • Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| 1134H | A TCP ULP timeout error has occurred in the TCP/IP communication. (The external device does not send an ACK response.) | • Check the operation of the external device.<br>• Correct the TCP ULP timeout value of the Ethernet-equipped module.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| 1152H | • The IP address is not set correctly.<br>• The same IP address has been set to port 1 and port 2 of the Ethernet-equipped module. | • Correct the IP addresses.<br>• Set different IP addresses for port 1 and port 2. | Parameter information<br>• Parameter type<br>• I/O No.<br>• Parameter No.<br>• Network No.<br>• Station No. |
| 1155H | • The specified connection was already closed in TCP/IP communications.<br>• Open processing is not performed. | • Perform the open processing for the specified connection.<br>• Check if the open processing has been performed in the external device. | — |
| 1157H | • The specified connection was already closed in UDP/IP communications.<br>• Open processing is not performed. | • Perform the open processing for the specified connection.<br>• Check if the open processing has been performed in the external device. | — |
| 1158H | • The receive buffer or send buffer is not sufficient.<br>• The window size of the external device is not sufficient. | • Check the operation of the external device or switching hub.<br>• When communications using a fixed buffer or socket communications is used, increase the execution frequency of the BUFRCV/SOCRCV instruction.<br>• When the value of 'State of receive buffer' (Un\G1900030) is 0001H, reduce the frequency of data receive from the external device. | — |
| 1165H | Data was not sent correctly with UDP/IP. | • Check the settings for connection with the external device.<br>• Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the PING test and communication status test, and if the test was completed with an error, take the corrective action. | — |
| 1166H | Data was not sent correctly with TCP/IP. | • Check the settings for connection with the external device.<br>• Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the PING test and communication status test, and if the test was completed with an error, take the corrective action.<br>• The close request may have been received from the external device when the data was sent. Check whether "TCP Status" of the corresponding connection number is set to "Disconnected" with the Ethernet diagnostics of the engineering tool. | — |
| 1167H | Unsent data found, but could not be sent. | • Check the settings for connection with the external device.<br>• Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the PING test and communication status test, and if the test was completed with an error, take the corrective action. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| 1180H | • The IP address is not set correctly.<br>• The same IP address has been set to port 1 and port 2 of the Ethernet-equipped module.<br>• The same IP address has been set as the system A IP address, system B IP address, and/or control system IP address.<br>• Network addresses of the system A IP address, system B IP address, and control system IP address are different. | • Correct the IP addresses.<br>• Set different IP addresses for port 1 and port 2.<br>• Set different IP addresses for the system A IP address, system B IP address, and control system IP address.<br>• Set the same network address for the system A IP address, system B IP address, and control system IP address. | Parameter information<br>• Parameter type<br>• I/O No.<br>• Parameter No.<br>• Network No.<br>• Station No. |
| 1801H | IP address of the external device could not be acquired. | • Correct the IP address in "Network/Station No. <-> IP information setting" under "Application Settings".<br>• Check if the network or station number of the external device is correctly specified by using control data of the dedicated instruction. | Parameter information<br>• Parameter type<br>• I/O No.<br>• Parameter No.<br>• Network No.<br>• Station No. |
| 1811H | An error was detected in the CPU module. | Check the error of the CPU module and take action using the module diagnostics of the engineering tool. | — |
| 1830H | Number of reception requests of transient transmission (link dedicated instruction) exceeded upper limit of simultaneously processable requests. | Lower the transient transmission usage frequency, and then perform again. | — |
| 1845H | Too many processings of transient transmission (link dedicated instruction) and cannot perform transient transmission. | Correct the transient transmission execution count. | — |
| 1F07H | An error occurs in the protocol data set in "Simple Device Communication Setting". | • From "Communication Destination Setting" under "Simple Device Communication Setting", correct the setting details of "Protocol Setting".<br>• If the error occurs again even after taking the above actions, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | Packet error information<br>Setting No.<br>Protocol No.<br>Packet type<br>Element No. |
| 20E0H | The module cannot communicate with the CPU module. | The hardware failure of the CPU module may have been occurred. Please consult your local Mitsubishi representative. | — |
| 2160H | Overlapping IP addresses were detected. | Check and correct the IP addresses. | — |
| 2220H | • A network module having the firmware version not supporting the simple CPU communication function is used.<br>• The number of simple CPU communication settings is 65 or more.<br>• The parameter setting is corrupted. | • Use a network module having the firmware version that supports 512 simple CPU communication settings.<br>• Set the simple CPU communication setting within the range from No.1 to No.64.<br>• Check the detailed information of the error by executing module diagnostics using the engineering tool, and write the displayed parameter. If the error occurs again even after taking the above, the possible cause is a hardware failure of the module. Please consult your local Mitsubishi representative. | Parameter information<br>• Parameter type |
| 2221H | The set value is out of the range. | Check the detailed information (parameter information) of the error by executing module diagnostics using the engineering tool, and correct the parameter setting corresponding to the displayed number (parameter number). If the same error code is displayed again, the possible cause is a hardware failure of the data memory of the CPU module, a memory card, the I/O module, or intelligent function module. Please consult your local Mitsubishi representative. | Parameter information<br>• Parameter type<br>• I/O No.<br>• Parameter No.<br>• Network No.<br>• Station No. |
| 2250H | The protocol setting data stored in the CPU module is not for the Ethernet-equipped module. | Write the protocol setting data for the Ethernet-equipped module to the CPU module. | Parameter information<br>• Parameter type |
| 24C0H to 24C3H | An error was detected on the system bus. | • Take measures to reduce noise.<br>• Reset the CPU module, and run it again. If the error occurs again even after taking the above, the possible cause is a hardware failure of the module, base unit, or extension cable. Please consult your local Mitsubishi representative. | System configuration information<br>• I/O No.<br>• Base No.<br>• Slot No.<br>• CPU No. |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| 24C6H | An error was detected on the system bus. | • Take measures to reduce noise.<br>• Reset the CPU module, and run it again. If the error occurs again even after taking the above, the possible cause is a hardware failure of the module, base unit, or extension cable. Please consult your local Mitsubishi representative. | — |
| 3008H | In the redundant system with redundant extension base unit, a module name other than "RJ71EN71(E+E)" and "RJ71EN71(Q)" is set on the extension base. | When mounting a module on the extension base unit in the redundant system with redundant extension base unit, select "RJ71EN71(E+E)" or "RJ71EN71(Q)" for the module name in the "Add New Module" window. | Parameter information<br>• Parameter type<br>• I/O No.<br>• Parameter No.<br>• Network No.<br>• Station No. |
| 3019H | • When mounting a module on the main base unit in the redundant system, "Not Use" is set to "To Use or Not to Use Redundant System Settings" under "Application Settings".<br>• When mounting a module on the extension base unit in the redundant system with redundant extension base unit, "Use" is set to "To Use or Not to Use Redundant System Settings" under "Application Settings". | • When mounting a module on the main base unit in the redundant system, set "Use" to "To Use or Not to Use Redundant System Settings" under "Application Settings".<br>• When installing a module to the extension base unit in the redundant system with redundant extension base unit, set "Not Use" to "To Use or Not to Use Redundant System Settings" under "Application Settings". | Parameter information<br>• Parameter type |
| 301AH | When mounting a module on the extension base unit in the redundant system with redundant extension base unit, "Enable" is set to "Dynamic Routing" under "Application Settings". | When mounting a module on the extension base unit in the redundant system with redundant extension base unit, Set "Disable" to "Dynamic Routing" under "Application Settings". | Parameter information<br>• Parameter type |
| 301BH | When mounting a module on the extension base unit in the redundant system with redundant extension base unit, "Use" is set to "IP Packet Transfer Function" under "Application Settings". | When mounting a module on the extension base unit in the redundant system with redundant extension base unit, set "Not Use" to "IP Packet Transfer Function" under "Application Settings". | Parameter information<br>• Parameter type |
| 301CH | • Incorrect protocol data is set in "Simple Device Communication Setting".<br>• A packet used is not supported by the current firmware version. | • From "Communication Destination Setting" under "Simple Device Communication Setting", correct the setting details of "Protocol Setting".<br>• Update the firmware.<br>• If the error occurs again even after taking the above actions, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | ■Detailed information 1<br>Parameter information<br>• Parameter type<br>■Detailed information 2<br>Packet error information<br>• Setting No.<br>• Protocol No.<br>• Packet type<br>• Element No. |
| 301DH | A function used is not supported by the current firmware version. | Update the firmware. | ■Detailed information 1<br>Parameter information<br>• Parameter type<br>■Detailed information 2<br>Packet error information<br>• Setting No.<br>• Protocol No.<br>• Packet type |
| 301EH | The "Resource Setting" is incorrectly set in "Simple Device Communication Setting" of the port 1 or port 2. | Ensure that the module supports "Extension 2" of "Resource Setting" by checking the third and fourth digits from the left in the production information of the Ethernet-equipped module. (☞ Page 292 Restrictions applicable depending on versions)<br>■For the Ethernet-equipped module not supporting "Extension 2"<br>• To select "Extension 1" in "Resource Setting", set "To Use or Not Use Simple Device Communication" of the other ports to "Not to Use".<br>• To select "Normal" in "Resource Setting", set "Resource Setting" of the other ports to "Normal" or set "To Use or Not Use Simple Device Communication" to "Not to Use".<br>■For the Ethernet-equipped module supporting "Extension 2"<br>Update the firmware. | Parameter information<br>• Parameter type |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| 3020H | A value of the port number is out of range. | • Check the each system port number registered in the buffer memory.<br>• If the error occurs again even after taking the above, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | — |
| 3022H | The total number of external devices and communication destinations set using parameters exceeds 64. | Set the total number of external devices set in "External Device Configuration", communication destinations set in "Simple CPU Communication Setting", and communication destinations set in "Simple Device Communication Setting" within a range that does not exceed 64 per port. | — |
| 3023H | "Port No. Host Station" set in "Simple Device Communication Setting" overlaps a value set for another function. | • Change "Port No. Host Station" set in "Simple Device Communication Setting" to a value different from "PLC Port No." of "External Device Configuration".<br>• Change "Port No. Host Station" set in "Simple Device Communication Setting" to a value different from "Host Station Port No." of "Simple CPU Communication Setting".<br>• If "Port No. Host Station" of "Simple Device Communication Setting" is set within a range of 61440 to 65534, set within a range of 1024 to 4999 or 5010 to 61439. | Packet error information<br>• Setting No. |
| 3026H | The module does not support "Extension 2", which is set in "Resource Setting" under "Simple Device Communication Setting". | Verify if the Ethernet-equipped module supports "Extension 2" by checking the third and fourth digits from the left in its production information. If the Ethernet-equipped module does not support "Extension 2", change the setting to "Normal" or "Extension 1". (☞ Page 292 Restrictions applicable depending on versions) | — |
| 3040H | Response data of the dedicated instruction cannot be created. | • Increase the request interval.<br>• Decrease the number of request nodes.<br>• Wait for a response to the previous request before sending the next request.<br>• Correct the timeout value. | — |
| 3060H | The send/receive data size exceeds the allowable range. | • Check and change the send data size of the Ethernet-equipped module or the external device.<br>• If the same error code is displayed again, the possible cause is a hardware failure of the error module or CPU module. Please consult your local Mitsubishi representative. | — |
| 3C00H to 3C03H | • A hardware failure has been detected.<br>• In a redundant system configuration, the control system was powered off or reset without the tracking cable connected. | • Take measures to reduce noise.<br>• Reset the CPU module, and run it again. If the same error code is displayed again, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative.<br>• In a redundant system configuration, fix the connection of the tracking cable, then reset the CPU modules of both systems. | — |
| 3C0FH to 3C11H | A hardware failure has been detected. | • Take measures to reduce noise.<br>• Reset the CPU module, and run it again. If the same error code is displayed again, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | — |
| 3C13H | A hardware failure has been detected. | Reset the CPU module, and run it again. If the same error code is displayed again, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | — |
| 3C14H | A hardware failure has been detected. | Reset the CPU module, and run it again. If the same error code is displayed again, the possible cause is a hardware failure of the error module or CPU module. Please consult your local Mitsubishi representative. | — |
| 3C2FH | An error was detected in the memory. | Reset the CPU module, and run it again. If the same error code is displayed again, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| 3E01H | Network type of the own station is unexpected setting. | Rewrite the module parameter using the engineering tool. If the same error code is displayed again, the possible cause is a hardware failure of the error module. Please consult your local Mitsubishi representative. | — |
| 4000H to 4FFFH | Errors detected by the CPU module (📖 Manual for the CPU module used) | | |
| C001H | • The IP address setting value of the E71 for the initial processing is incorrect.<br>• The setting value of the subnet mask field for the router relay function is incorrect. | • Correct the IP address setting value for the initial processing.<br>• Check if the class of the IP address is set to A/B/C.<br>• Correct the subnet mask setting value for the initial processing. | — |
| C012H | The port number used in a connection already opened is set. (For TCP/IP) | Correct the port numbers of the Ethernet-equipped module and the external device. | — |
| C013H | The port number used in a connection already opened is set. (For UDP/IP) | Correct the port numbers of the Ethernet-equipped module and the external device. | — |
| C015H | • The specified IP address of the external device for the open processing is incorrect.<br>• The specified IP address of the external device of the dedicated instruction is incorrect. | • Correct the specified IP address of the external device of the CONOPEN/OPEN instruction. Set the class to A/B/C.<br>• Execute the dedicated instruction again after correcting the specified IP address of the external device. | — |
| C016H | The open processing of the connection specified for pairing open has been already completed. | • Check that none of the connections targeted for pairing open has been opened.<br>• Correct the combination of modules set for pairing open. | — |
| C018H | The specified IP address of the external device is incorrect. | Correct the specified IP address of the external device. | — |
| C020H | The send/receive data length exceeds the allowable range. | • Correct the data length to be sent.<br>• When the amount of data to be sent exceeds the limit, divide the data into smaller chunks to send it. | — |
| C021H | An abnormal end response was received for communications using the fixed buffer and random access buffer. | • Check the command/response type of the subheader.<br>• Check the data length setting to be sent.<br>• Check if the communication data code setting of the Ethernet-equipped module meets the binary/ASCII of the message to be sent. | — |
| C022H | • A response could not be received within the response monitoring timer value.<br>• The connection with the external device was closed while waiting for a response. | • Check the operation of the external device.<br>• Correct the response monitoring timer value of the Ethernet-equipped module.<br>• Check the open status of the connection with the external device. | — |
| C024H | • Communications using the fixed buffer or communications using a random access buffer were executed when communication method is set to the "Predefined Protocol" connection.<br>• Predefined protocol was executed when communication method is set to "Fixed Buffer (Procedure Exist)" or "Fixed Buffer (No Procedure)" connection. | • Check that there is no error in the connection number specification of the dedicated instruction.<br>• Correct the communication method of the connection with the external device. | — |
| C025H | There is an error in the usage setting area when starting the open processing by the CONOPEN/OPEN instruction or I/O signals. | • When starting the open by using the CONOPEN/OPEN instruction, correct the usage setting area of the control data.<br>• When starting the open by I/O signals, correct the connection usage setting area of the buffer memory. | — |
| C026H | An error has occurred when reading/writing/verifying the predefined protocol setting data. | • Check that connection cable with the engineering tool is not disconnected and read/write/verify the predefined protocol setting data again.<br>• Do not write data simultaneously when writing protocol setting data from multiple engineering tools. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C027H | Message send of the socket communications has failed. | • Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module.<br>• Check the IP address specified as the destination. | — |
| C028H | Message send of the fixed buffer has failed. | • Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module.<br>• Check the IP address specified as the destination. | — |
| C029H | • Description of control data is not correct.<br>• Open instruction was executed through open settings parameter even though parameters are not set. | • Correct the descriptions of the control data.<br>• Set the open settings parameters. Execute the OPEN instruction through control data. | — |
| C035H | The existence of the external device could not be checked within the response monitoring timer value. | • Check the operation of the external device.<br>• Correct the settings in "Timer Settings for Data Communication" under "Application Settings" of the Ethernet-equipped module.<br>• Check if the connection cable is disconnected. | — |
| C040H | • Not all the data could be received within the response monitoring timer value.<br>• Sufficient data for the data length could not be received.<br>• The remaining part of the message divided at the TCP/IP level could not be received within the response monitoring timer value. | • Correct the data length of the communication data.<br>• The packets may be crowded in the line, so send the data again from the external device after a random time has passed. | — |
| C050H | When "ASCII" has been selected in the communication data code setting of the Ethernet-equipped module, ASCII code data which cannot be converted into binary code data has been received. | Check if ASCII code data which cannot be converted into binary code data has been sent from the external device. | — |
| C051H | • The number of read/write points from/to the device of SLMP message is out of the allowable range in the CPU module (in units of words).<br>• The number of write points for the long counter of SLMP message is not in two-word units. | Correct the number of read/write points and send the SLMP message to the Ethernet-equipped module again. | — |
| C052H | The number of read/write points from/to the device of SLMP message is out of the allowable range in the CPU module (in units of bits). | Correct the number of read/write points and send the SLMP message to the Ethernet-equipped module again. | — |
| C053H | The number of read/write points from/to the random device of SLMP message is out of the allowable range in the CPU module (in units of bits). | Correct the number of read/write points and send the SLMP message to the Ethernet-equipped module again. | — |
| C054H | The number of read/write points from/to the random device of SLMP message is out of the allowable range in the CPU module (in units of words, double words). | Correct the number of read/write points and send the SLMP message to the Ethernet-equipped module again. | — |
| C055H | The read/write size from/to the file data of SLMP message is out of the allowable range. | Correct the read/write size and send the SLMP message to the Ethernet-equipped module again. | — |
| C056H | The read/write request exceeds the largest address. | Correct the start address or the number of read/write points so that the request does not exceed the largest address and send the data to the Ethernet-equipped module again. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C057H | The request data length of the SLMP message does not match with the number of data in the character (a part of text). | Check and correct the text or request data length, and send the SLMP message to the Ethernet-equipped module again. | — |
| C058H | The request data length of the SLMP message after the ASCII/binary conversion does not match with the number of data in the character (a part of text). | Check and correct the text or request data length, and send the SLMP message to the Ethernet-equipped module again. | — |
| C059H | • The specified command and subcommand of the SLMP message are incorrect<br>• The function which is not supported by the target device was executed. | • Check that there is no error in the specification of the command and subcommand of the SLMP message.<br>• Check whether the function executed is supported by the target device.<br>• Check the version of the target device. | — |
| C05AH | The Ethernet-equipped module cannot read/write data from/to the device specified by the SLMP message. | Correct the specification of the device to be read/written and send the SLMP message to the Ethernet-equipped module again. | — |
| C05BH | The Ethernet-equipped module cannot read/write data from/to the device specified by the SLMP message. | Correct the specification of the device to be read/written and send the SLMP message to the Ethernet-equipped module again. | — |
| C05CH | • The received request data of the SLMP message is incorrect.<br>• The setting value of the communication setting when the iQSS function is executed is out of range.<br>• When the iQSS function is executed, the items of communication setting which cannot be set on the target device are set.<br>• When the iQSS function is executed, the required setting items have not been set to the target device. | • Correct the request data and send the SLMP message to the Ethernet-equipped module again.<br>• Correct the setting details of when the iQSS function is executed, and retry the operation. | — |
| C05DH | The "Monitor Request" command is received before the monitor registration is performed by "Monitor Registration/Clear" command of the SLMP message. | Register the monitoring data using "Monitor Registration/Clear" command and perform monitoring. | — |
| C05EH | • The time between received the SLMP message from the Ethernet-equipped module and returned response from the access destination exceeded the monitoring timer value set in the SLMP command.<br>• An SLMP request message to which a command without a response message is specified is send to a module with the other network number as an access destination. | • Increase the monitoring timer value.<br>• Check if the access destination is operating normally.<br>• Correct the network number or request destination station number.<br>• When the access destination is a module with a different network number, check if "Routing Setting" of "CPU Parameter" is correctly set.<br>• If the access destination is a module with a different network number, check if the network number is not in use.<br>• If an SLMP request message to which a command without a response message is specified is send to a module with the other network number as an access destination, this error code is responded, which does not mean failure. | — |
| C05FH | This request cannot be executed to the access destination specified by the SLMP message. | Correct the access destination. | — |
| C060H | The request details for bit devices of the SLMP message is incorrect. | Correct the request details and send the SLMP message to the Ethernet-equipped module again. | — |
| C061H | • The request data length of the SLMP message does not match with the number of data in the character (a part of text).<br>• The write data length specified by the label write command is not even byte.<br>• When the iQSS function is executed, incorrect frame is received. | • Check and correct the text or request data length, and send the SLMP message to the Ethernet-equipped module again.<br>• Add the dummy data for one byte, and specify the length by even byte.<br>• Check the operating status and connection status of the target device of when the iQSS function is executed.<br>• Check the connection of an Ethernet cable and a hub of when the iQSS function is executed.<br>• Check the line status of Ethernet of when the iQSS function is executed.<br>• Reset the CPU module and device to be targets of the iQSS function, and retry the operation.<br>For the error occurred when the iQSS function is executed, contact the manufacturer of the target device if the above actions do not solve the problem. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C070H | The device memory cannot be extended for the access destination specified by the SLMP message. | • Correct the SLMP message to read/write data without the device memory set for extension.<br>• Specify the extension of the device memory only for an Ethernet-equipped module mounted station and an R/Q/QnACPU via CC-Link IE Controller Network, MELSECNET/H, or MELSECNET/10. | — |
| C071H | The number of device points for data read/write set for modules other than an R/Q/QnACPU is out of the range. | Correct the number of read/write points and send the SLMP message to the Ethernet-equipped module again. | — |
| C072H | The request details of the SLMP message is incorrect. (For example, a request for data read/write in bit units has been issued to a word device.) | • Check if the data can be requested to the access destination.<br>• Correct the request details and send the SLMP message to the Ethernet-equipped module again. | — |
| C073H | The access destination of the SLMP message cannot issue this request. (For example, the number of double word access points cannot be specified for modules other than an R/Q/QnACPU.) | Correct the request details of the SLMP message. | — |
| C075H | The request data length for the label access is out of range. | • Correct the number of array points and the number of read/write points, and send the SLMP message to the Ethernet-equipped module again.<br>• Correct the number of abbreviated label points and send the SLMP message to the Ethernet-equipped module again.<br>• Correct the label name length and send the SLMP message to the Ethernet-equipped module again.<br>• Correct the label to shorten the label name and send the SLMP message to the Ethernet-equipped module again.<br>• Correct the read/write data length and send the SLMP message to the Ethernet-equipped module again. | — |
| C081H | The termination processing for the Ethernet-equipped module that is involved with the reinitialization processing is being performed, and arrival of link dedicated instructions cannot be checked. | Finish all the communications to perform the reinitialization processing of the Ethernet-equipped module. | — |
| C083H | The communication processing was abnormally ended in the link dedicated instruction communications | • Check if the relay station and external station are operating normally.<br>• Check if there is an error with the cable connection between the own station and the external station.<br>• If the line is heavily loaded, reduce the load, and retry the operation.<br>• Increase the number of resends for the link dedicated instructions and execute the operation. | — |
| C084H | The communication processing was abnormally ended in the link dedicated instruction communications | • Check if the own station/relay station/external station is operating normally.<br>• Check if there is an error with the cable connection between the own station and the external station.<br>• Increase the TCP resend timer value of the Ethernet-equipped module. | — |
| C085H | The target station's channel specified by the link dedicated instruction SEND is currently in use. | Correct the target station's channel of the SEND instruction. | — |
| C0B2H | There is no sufficient space in the receive buffer or the send buffer of the relay station or external station for the MELSOFT connection, link dedicated instructions, or SLMP. (Send · receive buffer full error) | • Increase the request interval (execution interval) and execute the operation.<br>• Do not access through one station using the MELSOFT connection, link dedicated instruction, or SLMP.<br>• Wait for a response to the previous request before sending the next request.<br>• Correct the set value for "Timer Settings for Data Communication" under "Application Settings" of the Ethernet-equipped module. | — |
| C0B3H | A request that cannot be processed was issued from the CPU module. | • Correct the request details.<br>• Correct the network number or request destination station number. | — |
| C0B6H | The channel specified by the dedicated instruction is out of the range. | Correct the channel to a value within the allowable range of each dedicated instruction. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C0BAH | Since the close processing is in execution using the CONCLOSE/CLOSE instruction, a send request cannot be accepted. | Do no perform the send processing during the close processing. | — |
| C0C4H | The UINI instruction has been executed during communications. | Execute the UINI instruction after closing all connections. | — |
| C0D0H | The specified data length of the link dedicated instruction is incorrect. | Correct the data length. | — |
| C0D1H | The number of resends of the link dedicated instruction is incorrect. | Correct the number of resends. | — |
| C0D3H | The number of relay stations to communicate with other networks exceeds the allowable range. | • Check if the specification (network number/station number) for the communication destination is correct.<br>• Check that number of relay stations accessing to the communication destination is 7 or less.<br>• Correct the settings in the Network station number <-> IP information for the stations between the own station and the communication destination. | — |
| C0D4H | The number of relay stations to communicate with other networks exceeds the allowable range. | • Check if the specification (network number/station number) for the communication destination is correct.<br>• Check that number of relay stations accessing to the communication destination is 7 or less.<br>• Correct the settings in the Network station number <-> IP information for the stations between the own station and the communication destination. | — |
| C0D5H | The number of retries of the link dedicated instruction is incorrect. | Correct the number of retries. | — |
| C0D6H | The network number or station number of the link dedicated instruction is incorrect. | Check if the specification (network number/station number) for the communication destination is correct. | — |
| C0D7H | Data were sent without the initial processing completed. | After normal completion of the initial processing, perform the communications with the external device. | — |
| C0D8H | The number of specified blocks exceeded the range. | Correct the number of blocks. | — |
| C0D9H | The specified subcommand of the SLMP message is incorrect. | Correct the subcommand. | — |
| C0DAH | A response to the PING test could not be received within the time of the communication time check. | • Correct the IP address and host name of the target module where the PING test is executed.<br>• Check that the initial processing of the Ethernet-equipped module where the PING test is executed is completed successfully. | — |
| C0DBH | The IP address and host name of the target module where the PING test is execute are incorrect. | Correct the IP address and host name of the target module where the PING test is executed. | — |
| C0DEH | Data could not be received within the specified arrival monitoring time. | • Correct the specified arrival monitoring time.<br>• Correct the channel of the link dedicated instruction.<br>• Check if the sending station and relay station are operating normally. | — |
| C101H | A response could not be received from the DNS server. | • Check the address of the DNS server.<br>• Check that the data communications with the DNS server is possible by using the Ping command.<br>• Check that the own station IP address and the DNS server IP address are in the same class. (If not, check the router setting.) | — |
| C1A2H | • A response to the request could not be received.<br>• In transient transmission, the number of relay to other networks exceeded seven. | • Check and correct the response waiting time.<br>• Change the system configuration so that the number of relay may be seven or less.<br>• For the RECV instruction, execute again after correcting the channels used by own station in the control data.<br>• For the RECV instruction, check that 'RECV execution request' (Un\G5301 b0 to b7) are on.<br>• Check the operation of the external device.<br>• Check if the external device supports the executed function.<br>• If the request destination is on another network, check if the CPU module working as the relay station supports the routing setting, and take action. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C1A4H | • Any of the specified command, subcommand, or request destination module I/O number of the SLMP message is incorrect.<br>• The specified clear function set by the ERRCLEAR instruction is incorrect.<br>• The specified information to be read set by the ERRRD instruction is incorrect.<br>• The Ethernet diagnostics, CC-Link IE Field Network diagnostics, or CC-Link IE Controller Network diagnostics was tried to be used when the engineering tool is directly connected to the Ethernet port of the RJ71EN71.<br>• The function which is not supported by the target device was executed. | • Check that all of the specified command, subcommand, and request destination module I/O number of the SLMP message is correct.<br>• Correct the specified value for the clear function set by the ERRCLEAR instruction.<br>• Correct the specified value for the information to be read set by the ERRRD instruction.<br>• Connect it directly to the CPU module (built-in Ethernet port part) or use a USB cable to use the Ethernet diagnostics, CC-Link IE Field Network diagnostics, or CC-Link IE Controller Network diagnostics. | — |
| C1A5H | The specified target station or clear target is incorrect. | Correct the specified the target station or clear target. | — |
| C1A6H | The specified connection number is incorrect. | Correct the setting value of the connection number. | — |
| C1A7H | The specified network number is incorrect. | Correct the specified network number. | — |
| C1A8H | The specified station number is incorrect. | Correct the specified station number. | — |
| C1A9H | The specified device number is incorrect. | Correct the specified device number. | — |
| C1AAH | The specified device name is incorrect. | Correct the specified device name. | — |
| C1ACH | The specified number of resends is incorrect. | Correct the number of resends. | — |
| C1ADH | The specified data length is incorrect. | Correct the specified data length. | — |
| C1AFH | The specified port number is incorrect. | Correct the specified port number | — |
| C1B0H | The open processing of the specified connection has been already completed. | • Do not perform the open processing to a connection already opened.<br>• When communications with the external device cannot be performed, perform the close processing before the open processing. | — |
| C1B1H | The open processing of the specified connection has not been completed. | After completion of the open processing, perform the communication. | — |
| C1B2H | The open or close processing using CONOPEN/CONCLOSE/OPEN/CLOSE instruction is being executed in the specified connection. | Execute again after the CONOPEN/CONCLOSE/OPEN/CLOSE instruction is completed. | — |
| C1B3H | Another send or receive instruction is being executed in the specified channel. | • Change the channel number.<br>• Execute again after the send or receive instruction is completed. | — |
| C1B4H | The specified arrival monitoring time is incorrect. | Set the arrival monitoring time to a value within the allowable range. | — |
| C1B8H | The RECV instruction was executed for the channel that had not received data. | • Correct the execution condition of the RECV instruction for the channel that had received data.<br>• Correct the specified channel of the RECV instruction. | — |
| C1B9H | The CONOPEN/OPEN instruction cannot be executed for the specified connection. | Correct the specified connection. | — |
| C1BAH | The dedicated instruction was executed with the initialization not completed. | Execute the dedicated instruction after the initial processing is completed. | — |
| C1BBH | The target station CPU type of the link dedicated instruction is incorrect. | Correct the specified target station CPU type. | — |
| C1BCH | The target network number of the link dedicated instruction is incorrect. | • Execute the link dedicated instruction again after correcting the network number.<br>• When specifying another network station, check if the settings in "Routing Setting" of "CPU Parameter" are made correctly. | — |
| C1BDH | The target station number of the link dedicated instruction is incorrect. | • Execute the link dedicated instruction again after correcting the target station number.<br>• When specifying another network station, check if the settings in "Routing Setting" of "CPU Parameter" are made correctly. | — |
| C1BEH | The command code of the dedicated instruction is incorrect. | • Execute again after correcting the command code at the request source.<br>• If the request source is on another network, check if the routing parameters are set correctly, and take action. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C1BFH | The channel used in the dedicated instruction is incorrect. | • Execute again after correcting the used channel within the allowable range at the request source.<br>• If the request source is on another network, check if the routing parameters are set correctly, and take action. | — |
| C1C0H | The transient data is incorrect. | • Correct the transient data at the request source, and retry the operation.<br>• If the error occurs again even after taking the above, please consult your local Mitsubishi representative. | — |
| C1C1H | The transient data is incorrect. | • Correct the transient data at the request source, and retry the operation.<br>• If the error occurs again even after taking the above, please consult your local Mitsubishi representative. | — |
| C1C2H | When the dedicated instruction was executed, data was received twice. | • Check the network status and take corrective action using the Ethernet diagnostics of the engineering tool.<br>• Check if the switching hub and the cables at the request source are connected properly.<br>• If the request source is on another network, check if the routing parameters are set correctly, and take action. | — |
| C1C4H | The arrival check of the link dedicated instruction was completed with an error. | • Execute link dedicated instruction again after changing the execution type in the control data to "No arrival acknowledgment".<br>• For the REQ instruction, execute again after correcting request type. | — |
| C1C5H | A dedicated instruction which the target station does not support was executed. | • Change the target station at the station that executed the SEND instruction.<br>• If the error occurs again even after taking the above, please consult your local Mitsubishi representative. | — |
| C1C6H | The execution or error completion type of the dedicated instruction is incorrect. | • Execute again after correcting the execution/error completion type in the control data.<br>• If the problem cannot be resolved with the above actions, the possible cause is a hardware failure of the module. Please consult your local Mitsubishi representative. | — |
| C1C7H | The request type of the REQ instruction is incorrect. | • Execute again after correcting the request type of the REQ instruction.<br>• If the problem cannot be resolved with the above actions, the possible cause is a hardware failure of the module. Please consult your local Mitsubishi representative. | — |
| C1C8H | The channel specified in the dedicated instruction is in use. | Change the channels used by own station or the target station's channel in the control data. | — |
| C1C9H | The device specification for the ZNRD/ZNWR instruction is not correct. | • Execute again after correcting the device specification for the ZNRD/ZNWR instruction.<br>• If the problem cannot be resolved with the above actions, the possible cause is a hardware failure of the module. Please consult your local Mitsubishi representative. | — |
| C1CAH | The device specification for the ZNRD/ZNWR instruction is not correct. | • Execute again after correcting the device specification for the ZNRD/ZNWR instruction.<br>• If the problem cannot be resolved with the above actions, the possible cause is a hardware failure of the module. Please consult your local Mitsubishi representative. | — |
| C1CBH | The transient data is incorrect. | • Correct the transient data at the request source, and retry the operation.<br>• If the error occurs again even after taking the above, please consult your local Mitsubishi representative. | — |
| C1CCH | A response of the data length that exceeds the allowable range was received by the SLMPSND instruction. | • Execute again after correcting the request data to be within the range.<br>• If the error occurs again even after taking the above, please consult your local Mitsubishi representative. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C1CDH | Message send of the SLMPSND instruction has failed. | • Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module.<br>• Check the IP address specified as the destination.<br>• Change "Use" in "SLMPSND Instruction Settings" under "Application settings" of the module for which the SLMPSND instruction was executed, and retry the operation. | — |
| C1D0H | The requested module I/O No. of the dedicated instruction is incorrect. | • Execute again after correcting the requested module I/O No. at the request source of the dedicated instruction.<br>• If the error occurs again even after taking the above, please consult your local Mitsubishi representative. | — |
| C1D2H | The target station IP address of the link dedicated instruction is incorrect. | Execute the link dedicated instruction again after correcting the IP address. | — |
| C1D3H | The dedicated instruction not supported by the communication method of the connection was executed. | • Check that the dedicated instruction can be executed by the specified communication method. Correct the program when the instruction cannot be executed.<br>• Check that there is no error in the connection specification of the dedicated instruction. | — |
| C200H | The remote password is incorrect. | Correct the remote password, and unlock/lock the remote password again. | — |
| C201H | The remote password status of the port used for communications is in the lock status. | After unlocking the remote password, perform communications. | — |
| C202H | When another station was accessed, the remote password could not be unlocked. | When accessing another station, do not set the remote password on the relay station or access station, or do not execute the remote password check on them. | — |
| C203H | An error has occurred by checking the remote password. | Correct the remote password, and unlock/lock the remote password again. | — |
| C204H | The device is different from the one requesting the remote password unlock processing. | Request the lock processing of the remote password from the external device that requested the unlock processing of the remote password. | — |
| C205H | When another station was accessed, the remote password could not be unlocked. | When accessing another station, do not set the remote password on the relay station or access station, or do not execute the remote password check on them. | — |
| C207H | The file name has too many characters. | Name the file with 255 characters or less. | — |
| C208H | The password length is out of range. | Set the password within 6 to 32 characters. | — |
| C400H | The ECPRTCL instruction was executed when Predefined protocol ready is not completed. | • Execute the ECPRTCL instruction after Predefined protocol ready has turned on.<br>• Execute the ECPRTCL instruction after rewriting the protocol setting data to the Ethernet-equipped module.<br>• If the error occurs again even after taking the above, the possible cause is a hardware failure of the module. Please consult your local Mitsubishi representative. | — |
| C401H | The protocol number specified by the ECPRTCL instruction is not registered in the Ethernet-equipped module. | • Correct the specified protocol number and execute the instruction again.<br>• Register the protocol specified protocol number to the Ethernet-equipped module. | — |
| C402H | An error has occurred in the protocol setting data registered in the Ethernet-equipped module and the ECPRTCL instruction cannot be executed. | Correct the protocol setting data and register it again. | — |
| C403H | Multiple dedicated instructions was executed simultaneously. | • Do not execute the dedicated instructions which do not support simultaneous execution.<br>• Correct the specified connection number and execute the dedicated instruction again. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C404H | The protocol being executed by the ECPRTCL instruction was canceled. | Check the canceled protocol in the control data of the ECPRTCL instruction (execution count result) and eliminate the cause of the cancellation. | — |
| C405H | The protocol number specified by the ECPRTCL instruction is incorrect. | Correct the specified protocol number. | — |
| C406H | The continuous protocol execution count of the ECPRTCL instruction is incorrect. | Correct the continuous protocol execution count. | — |
| C407H | The connection number specified by the ECPRTCL instruction is incorrect. | • Correct the specified connection number and execute the protocol again.<br>• Correct the specified connection number in "External Device Configuration" under "Basic Settings" of the Ethernet-equipped module and execute the protocol again. | — |
| C408H | An error has occurred when the send processing of the predefined protocol using the ECPRTCL instruction was performed. | • Check the operation of the external device or switching hub.<br>• Since there may be congestion of packets on the line, send data after a certain period of time.<br>• Check if the connection cable is disconnected.<br>• Check that there is no connection failure with the switching hub.<br>• Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C410H | Receive waiting time of the ECPRTCL instruction timed out. | • Check if the cable is disconnected.<br>• Correct the specified connection number in "External Device Configuration" under "Basic Settings" of the Ethernet-equipped module and execute the protocol again.<br>• Check that there is no error in the external device.<br>• Check that the sending from the external device is not interrupted.<br>• Check that there is no data lost due to a receive error.<br>• Check that there is no error in the data (packet) sent by the external device. | — |
| C412H | The data which cannot be converted from ASCII to binary code was received. | • Check that there is no data lost due to a receive error.<br>• Check that there is no error in the data (packet) sent by the external device. | — |
| C413H | The number of digits of the received data using the predefined protocol is not sufficient. | • Check that there is no data lost due to a receive error.<br>• Check that there is no error in the data (packet) sent by the external device. | — |
| C414H | The number of digits of the received data using the predefined protocol is incorrect. | • Check that there is no data lost due to a receive error.<br>• Check that there is no error in the data (packet) sent by the external device. | — |
| C417H | The data length or data quantity of the received data using the predefined protocol is out of range. | • Check the maximum allowable data length and specify the maximum length or less in the data length storage area.<br>• Check the maximum allowable data quantity, and specify the maximum quantity or less in the data quantity storage area. | — |
| C420H | Protocol setting data write has failed. | • Write the data again.<br>• If the error occurs again even after taking the above, the possible cause is a hardware failure of the specified module. Please consult your local Mitsubishi representative. | — |
| C421H | Writing was requested to the module whose flash ROM write count had exceeded the limit. | Replace the module because the number of writes exceeded the limit. | — |
| C430H | Protocol setting data was written during the ECPRTCL instruction execution. | Do not write the protocol setting data during the ECPRTCL instruction execution. | — |
| C431H | Close processing of the connection was performed during the ECPRTCL instruction execution. | • Check the operation of the external device.<br>• Check the connection open status with the external device.<br>• Open the connection with the external device again and execute the instruction. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C440H to C44FH | A communication error has occurred with an engineering tool when executing the Ethernet diagnostics. | • Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C610H to C613H | The module processing was completed with an error. | • Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C614H | The module processing was completed with an error. | • Correct the setting value of "Response Monitoring Timer" under "FTP Server Settings" because writing files may require a longer time.<br>• Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C615H | The module processing was completed with an error. | • Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C616H | Connection of the control port to the FTP server failed. | • Correct the IP address setting of the Ethernet-equipped module.<br>• Correct the FTP server setting.<br>• Check connection with the FTP server.<br>• Disconnect the user session on the FTP server.<br>• Data communications may not be ready. Wait for a while and perform the operation again.<br>• Correct the TCP ULP timeout value because connecting to the FTP server may require a longer time. | — |
| C617H | Disconnection of the control port to the FTP server failed. | • Correct the FTP server setting.<br>• Check connection with the FTP server. | — |
| C618H | Login to the FTP server failed. | • Correct the FTP server setting (login user name and login password).<br>• Check the FTP server software settings (login user name and login password).<br>• Check the data communication history of the FTP server software. | — |
| C619H | Execution of the FTP command to the FTP server failed. | • Correct the FTP server setting (folder path and connection method).<br>• Check that the user has a right to access (read/write) the FTP server or the specified file.<br>• Check that the specified folder path exists in the FTP server.<br>• Check that the specified file exists in the FTP server.<br>• Correct the FTP server software settings.<br>• Check the data communication history of the FTP server software.<br>• Check that the file access is being performed in the FTP server. | — |
| C620H | Connection of the data transfer port to the FTP server failed. | • Check connection with the FTP server.<br>• Correct the FTP server setting (connection method).<br>• If a firewall is active or the proxy server is on the connection path, consult a network administrator about the settings. | — |
| C621H | Disconnection of the data transfer port to the FTP server failed. | • Check connection with the FTP server.<br>• Correct the FTP server setting (connection method).<br>• If a firewall is active or the proxy server is on the connection path, consult a network administrator about the settings. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C622H | An error has occurred during file transfer to the FTP server. | • Delete unnecessary files on the FTP server to increase free space.<br>• Check connection with the FTP server.<br>• The specified file may be used in the other process. Wait for a while and perform the operation again.<br>• The Ethernet line may be congested. Wait for a while and perform the operation again. | — |
| C623H | A response could not be received from the FTP server. | • Check that the FTP server name is registered in the DNS.<br>• Change the FTP server name to the IP address, and check the operation.<br>• Check that the data communications with the FTP server is possible by using the Ping command. | — |
| C700H | The module processing was completed with an error. | • Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C701H | The IP address (network number) setting is incorrect in communications using the IP packet transfer function. | Check the IP address (network number). | — |
| C702H | The IP address (station number) setting is incorrect in communications using the IP packet transfer function. | Check the IP addresses (station number). | — |
| C703H | The destination IP address (upper level) setting is incorrect in communications using the IP packet transfer function. | Check the destination IP address. | — |
| C704H | The destination IP address (lower level) setting is incorrect in communications using the IP packet transfer function. | Check the destination IP address. | — |
| C705H to C707H | The module processing was completed with an error. | • Execute the communication status test, and if the test was completed with an error, take the corrective action.<br>• Execute the module communication test, and check that there is no failure in the module. | — |
| C708H | When communicating with the IP packet transfer function, "IP Packet Transfer Function" is set as "Not Use" in "IP Packet Transfer Setting" under "Application Settings" of the Ethernet-equipped module connected with the Ethernet devices. | When communicating with the IP packet transfer function, set "IP Packet Transfer Function" as "Not Use" in "IP Packet Transfer Setting" under "Application Settings". | — |
| C709H | A communication error has occurred with MELSOFT direct connection. | • Do no execute the specification of the direct connection when direct connection is not used.<br>• Do not power off the system or reset the CPU module, or remove the cable during the communications when direct connection is used. | — |
| C810H | Remote password authentication has failed when required. | Set a correct password and perform password authentication again. | — |
| C811H | Remote password authentication has failed when required. | Set a correct password and perform password authentication again one minute later. | — |
| C812H | Remote password authentication has failed when required. | Set a correct password and perform password authentication again 5 minutes later. | — |
| C813H | Remote password authentication has failed when required. | Set a correct password and perform password authentication again 15 minutes later. | — |
| C814H | Remote password authentication has failed when required. | Set a correct password and perform password authentication again 60 minutes later. | — |
| C815H | Remote password authentication has failed when required. | Set a correct password and perform password authentication again 60 minutes later. | — |
| C816H | The security function was activated and remote password authentication cannot be performed. | Set a correct password and perform password authentication again after a certain period of time. | — |
| C840H | Number of transient request exceeded the upper limit of simultaneously processable requests. | • Pause the transient transmission temporarily, and retry the operation.<br>• Lower the transient transmission usage frequency, and then perform again. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C842H | The routing setting is not set to reach to the destination network number. | • Execute the link dedicated instruction again after correcting the target network number and station number.<br>• When the dynamic routing is used, check that communication path to the destination network number is set.<br>• When the dynamic routing is not used, or a module of the series other than MELSEC iQ-R is included, retry the operation after correcting the settings in "Routing Setting" of "CPU Parameter". | — |
| C843H | Link dedicated instruction that cannot be executed on the network type were executed. | Check the network type of the Ethernet-equipped module. | — |
| C844H | Incorrect frame is received.<br>• Unsupported pre-conversion protocol<br>• Unsupported frame type<br>• Application header variable part<br>• Application header HDS<br>• Application header RTP<br>• Read command not requiring response | Correct the request data at the transient request source, and retry the operation. | — |
| C860H | The CPU response monitoring timer issued a timeout during MODBUS/TCP communications. | When the CPU module has an error, eliminate it. | — |
| C861H | A request message containing an unsupported function code was received during MODBUS/TCP communications. | Check the supported function code, and correct the request message issued on the master side. | — |
| C862H | A request message containing an unsupported sub-code was received during MODBUS/TCP communications. | Check the supported sub-code and correct the request message issued on the master side. | — |
| C863H | The MODBUS device assignment parameters have not been set for the MODBUS device specified in the received request message in MODBUS/TCP communications. | Set the MODBUS device assignment parameters for the MODBUS device specified in the request message. | — |
| C864H | The range of the MODBUS devices specified in the received request message exceeded the setting range of the MODBUS device assignment parameters in MODBUS/TCP communications. | Set the MODBUS device assignment parameters so that they include the range of the MODBUS devices specified in the request message. | — |
| C865H | The range of the MODBUS devices specified in the received request message exceeded the upper limit for the MODBUS devices in MODBUS/TCP communications. (The upper limit of extended file register is 10000 and the upper limit of the MODBUS device is 65536.) | Check the MODBUS devices specified on the master side that sent the request message. | — |
| C866H | The start address and the number of access points of the MODBUS device specified in the received request message are incorrect in MODBUS/TCP communications. | • Check if the specified value exceeds the maximum number of access points for the function.<br>• Ensure that the number of access points is not 0.<br>• When accessing a MODBUS device to which a long device is assigned in the MODBUS device assignment parameters, check that the start address and the number of access points are specified in units of two points.<br>• When accessing a MODBUS device to which LTN or LSTN is assigned in the MODBUS device assignment parameters, check that the start address and the number of access points are specified in units of four points. | — |
| C867H | The number of write points specified for the received request message does not match with the specified number of bytes in MODBUS/TCP communications. | Check the number of write points and the number of bytes specified on the master side that issued the request message. | — |
| C868H | The received write data size does not match with the specified number of bytes in MODBUS/TCP communications. | Check the write data size and the number of bytes specified on the master side that issued the request message. | — |
| C869H | The value of the reference type specified in the received request message (FC20, FC21) is incorrect in MODBUS/TCP communications. | Check the reference type specified on the master side that issued the request message. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| C86AH | The content of the data part of the received request message is incorrect in MODBUS/TCP communications. The size of the received request message is smaller than the minimum size required for the function code, or larger than the maximum size required for the function code. | Check the data part of the request message on the master side that issued the message. | — |
| C86BH | The content of MBAP header of the received request message is incorrect in MODBUS/TCP communications. | Check the content of the request message on the master side that issued it. | — |
| C86CH | The number of received request messages exceeded the number that can be received simultaneously. | Wait for a while and retry sending on the master side that issued it. (The number of request messages that can be received simultaneously is 64 for each of PORT1 and PORT2.) | — |
| CEE0H | The devices supporting iQSS which were detected by the other peripheral device, or other iQSS functions were executed while the automatic detection of connected devices is in process. | Execute the other function after the automatic detection of connected devices is completed. | — |
| CEE1H | Incorrect frame is received. | • Check the operating status and connection status of the target device.<br>• Check the connection of an Ethernet cable and a hub.<br>• Check the line status of Ethernet.<br>• Reset the CPU module and target device, and retry the operation.<br>If the above actions do not solve the problem, contact the manufacturer of the target device. | — |
| CEE2H | Incorrect frame is received. | • Check the operating status and connection status of the target device.<br>• Check the connection of an Ethernet cable and a hub.<br>• Check the line status of Ethernet.<br>• Reset the CPU module and target device, and retry the operation.<br>If the above actions do not solve the problem, contact the manufacturer of the target device. | — |
| CF10H | Incorrect frame is received. | • Check the operating status and connection status of the target device.<br>• Check the connection of an Ethernet cable and a hub.<br>• Check the line status of Ethernet.<br>• Reset the CPU module and target device, and retry the operation.<br>If the above actions do not solve the problem, contact the manufacturer of the target device. | — |
| CF20H | • The setting value of the communication setting is out of range.<br>• The items of communication setting which cannot be set on the target device are set.<br>• The required setting items have not been set to the target device. | Correct the setting details, and retry the operation. | — |
| CF30H | The parameter which is not supported by the target device was specified. | Check the version of the target device. | — |
| CF31H | Incorrect frame is received. | • Check the operating status and connection status of the target device.<br>• Check the connection of an Ethernet cable and a hub.<br>• Check the line status of Ethernet.<br>• Reset the CPU module and target device, and retry the operation.<br>If the above actions do not solve the problem, contact the manufacturer of the target device. | — |
| CF70H | An error occurred on the Ethernet communication path. | • Check the operation of the target device.<br>• Check if the connection cable is disconnected. | — |
| CF71H | A timeout error has occurred. | • Check the operation of the target device. Since there may be congestion of packets on the line, perform the operation after a while.<br>• Correct the setting details of when the iQSS function is executed, and retry the operation.<br>• Check the connection of an Ethernet cable and a hub. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| CF80H | During the simple device communication, communications with the communication destination cannot be executed. Or the external device is disconnected. | • Check if the connection cable is disconnected.<br>• Check the line status with the communication destination such as a cable, hub, and router.<br>• Correct the IP address and Ethernet address of the communication destination.<br>• Check if the communication destination has the ARP function, and communicate with a communication destination that has the ARP function. | — |
| CF81H | During the simple device communication, the communication has failed due to a communication timeout. | • Check the operation of the communication destination.<br>• Check the line status with the communication destination such as a cable, hub, and router.<br>• Correct the communication timeout period.<br>• Correct the setting of the relevant setting number for the Ethernet-equipped module, and execute the protocol again.<br>• Check if the communication destination has the ARP function, and communicate with a communication destination that has the ARP function. | — |
| CF82H | During the simple device communication, the "Send/Receive Data Length Storage Area" or "Send/Receive Data Count Storage Area" value in "Non-conversion Variable (variable length)" or "Conversion Variable (variable length)" of "Variable Number of Data" set for the send packet exceeds the range that can be set. | Check that the value set in "Send/Receive Data Length Storage Area" or "Send/Receive Data Count Storage Area" is equal to or below "Data Length/Maximum Data Length" or "Number of Data/Maximum Number of Data" set in "Element Setting". | — |
| CF83H | During the simple device communication, the data size of send data has been 0. | • Correct the setting of the element for the send packet.<br>• When "Non-conversion Variable (variable length)" or "Conversion Variable (variable length)" of "Variable Number of Data" is used for the element, check that the value in "Send/Receive Data Length Storage Area" or "Send/Receive Data Count Storage Area" is not 0. | — |
| CF84H | During the simple device communication, sending the send packet has failed. | • When the communication setting is "Request", execute the protocol again after a while.<br>• When the communication setting is "Fixed Intrvl", extend the execution interval. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| CF85H | Data outside the range was set during simple device communication. | When "Conversion Variable" is used for the element, check the following items.<br>• If the conversion data type is "Word" or "Double Word", the number of digits is not "Variable Number of Digits", and also the number of decimal places is "Variable Point", check the numerical value data as well as the decimal point position value and configure settings so that the decimal point position does not exceed the numerical value data.<br>• Check that the value in the send data storage area is not outside the range of numerical values that can be used. The range of numerical values that can be used is shown below.<br>■When conversion is "HEX→ASCII Decimal"<br>[Signed/Unsigned: "Unsigned"]<br><Single-precision real number><br>0 ≤ [Single-precision real number data] ≤ 9999999<br><Double-precision real number><br>0 ≤ [Double-precision real number data] ≤ 999999999999999<br>[Signed/Unsigned: Other than "Unsigned"]<br><Single-precision real number><br>-9999999 ≤ [Single-precision real number data] ≤ 9999999<br><Double-precision real number><br>-999999999999999 ≤ [Double-precision real number data] ≤ 999999999999999<br>■When conversion is "HEX→ASCII Hexadecimal"<br><Single-precision real number><br>0H to FFFFFFFFH<br><Double-precision real number><br>0H to FFFFFFFFFFFFFFFFH<br>• Check the send data storage area value, number of digits, as well as the setting for the number of decimal places and set them so that the integer part remains after conversion.<br>• When the conversion data type is "Word" or "Double Word" and the number of decimal places is "Variable Point", check whether the correct value is stored at the decimal point position of the send/receive data storage area. | — |
| CF8AH | During the simple device communication, the request to the CPU module has been failed. | • Check and correct the monitoring time of the CPU Response Monitoring Timer.<br>• Correct "Device/Label Access Service Processing Setting" under "Service Processing Setting" of "CPU Parameter". | — |
| CF8CH | During the simple device communication, the receive data does not match with any of the receive packets set using the parameter. | Check 'Verification mismatch protocol number' (Un\G1249317) and 'Verification mismatch information' (Un\G1249318 to Un\G1249333) and correct the setting details of the element. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| CFB0H | Sending data failed due to a resend timeout while the simple CPU communication. | • Check the operation of the communication destination.<br>• Check the line status with the communication destination such as a cable, hub, and router.<br>• Check and correct the latency time.<br>• Correct the IP address and Ethernet address of the communication destination.<br>• Check if the communication destination has the ARP function, and communicate with a communication destination that has the ARP function.<br>• When the communication destination is a MODBUS/TCP-compatible device and "Write" is set to "Communication Pattern", do not specify Input or Input Register for the transmission destination.<br>• If the number of connections exceeds the upper limit in the external device, reduce the number of used connections by sharing a TCP connection with multiple communication settings, or through other means. Setting the own station port numbers to the same value allows for the sharing of one TCP connection.<br>• The type of the communication destination depends on the firmware version of the network module. Check the firmware version. (☞ Page 226 Restrictions applicable depending on versions) | — |
| CFB1H | Communications with the communication destination do not executed while the simple CPU communication. Or the external device is disconnected. | • Check if the connection cable is disconnected.<br>• Check the line status with the communication destination such as a cable, hub, and router.<br>• Correct the IP address and Ethernet address of the communication destination.<br>• Check if the communication destination has the ARP function, and communicate with a communication destination that has the ARP function.<br>• The type of the communication destination depends on the firmware version of the network module. Check the firmware version. (☞ Page 226 Restrictions applicable depending on versions) | — |
| CFB2H | The same specified own station port number is already used for the simple CPU communication. | • Correct the specified port number so that it is not duplicate.<br>• It depends on the firmware version whether the network module can make multiple settings for the same communication destination. Check the firmware version of the network module. | — |
| CFB3H | The request to the CPU module has failed while the simple CPU communication. | • Check and correct the monitoring time of the CPU Response Monitoring Timer.<br>• Correct "Device/Label Access Service Processing Setting" under "Service Processing Setting" of "CPU Parameter". | — |
| CFB4H | An abnormal response was received from the communication destination while the simple CPU communication. | Check the abnormal response code in the buffer memory. | — |
| CFB5H | The frame received from the communication destination is incorrect while the simple CPU communication. | • Check the operation of the communication destination.<br>• Check the line status with the communication destination such as a cable, hub, and router.<br>• When the communication destination is SIEMENS S7 series or SIEMENS S7 series (extension), check "Simple CPU Communication Setting" to see whether "Type", "Start", "End" of the specified device match with the assignment of the communication destination. | — |
| CFBDH | The device specified as the communication destination is out of specification range for the simple CPU communication. | Correct the address of "Type", "Start", "End" of the device specified as the communication destination in "Simple CPU Communication Setting" under "Application settings". | — |
| CFBEH | A communication error has occurred with an engineering tool during the simple CPU communication diagnostics. | • Execute the communication status test, and if the test was completed with an error, take the corrective action. Execute the module communication test, and check that there is no failure in the module. | — |

| Error code (Hex) | Error details and causes | Action | Detailed information |
|---|---|---|---|
| CFBFH | The simple CPU communication cannot be executed. | • When the communication destination is the MELSEC-Q (Ethernet module) or MELSEC-L (Ethernet module), set the network number and station number in "Own Node Settings" under "Basic Settings".<br>• Take measures to reduce noise.<br>• It depends on the firmware version of the network module whether the on-request communications is possible or not. Check the firmware version of the network module.<br>• The type of the communication destination depends on the firmware version of the network module. Check the firmware version. (☞ Page 226 Restrictions applicable depending on versions)<br>• If the same error code is displayed even after executing the simple CPU communication again, please consult your local Mitsubishi representative. | — |

# 3.6 List of Parameter Numbers

This section lists the parameter numbers displayed in the module diagnostics.

## System Parameter

| Item | Parameter No. |
|---|---|
| Redundant Module Group Setting | 0400H |

## Basic Settings

| Item | | | Parameter No. |
|---|---|---|---|
| Own Node Settings | Parameter Setting Method | | 7100H |
| | IP Address | IP Address | A012H |
| | | Subnet Mask | A012H |
| | | Default Gateway | A013H |
| | Communications by Network No./Station No. (RJ71EN71 and RnENCPU (network part)) | | 7100H |
| | Communications by Network No./Station No. (CPU module (built-in Ethernet port part)) | | 7A20H |
| | Setting Method | | 7100H |
| | Network Number | | 7100H |
| | Station No. | | 7100H |
| | Transient Transmission Group No. | | A010H |
| | Enable/Disable Online Change | | A030H |
| | Communication Data Code | | A031H |
| | Opening Method | | A031H |
| CC-Link IEF Basic Setting | Network Configuration Settings | | 7A00H |
| | Refresh Settings | | 7420H |
| External Device Configuration | | | A031H |
| Transmission Port Settings (RnENCPU (network part)) | MELSOFT Transmission Port UDP/IP | | A0E4H |
| | MELSOFT Transmission Port TCP/IP | | |
| | Auto-open UDP Port | | |
| Transmission Port Settings (CPU module (built-in Ethernet port part)) | MELSOFT Transmission Port UDP/IP | | 7A30H |
| | MELSOFT Transmission Port TCP/IP | | |
| | Auto-open UDP Port | | |

## Application Settings

| Item | | Parameter No. |
|---|---|---|
| Frame Settings | | A036H |
| Communication Speed | | 7100H |
| FTP Server Settings | | A037H |
| FTP Client Settings | | A03DH |
| DNS Settings | | A03EH |
| Web Server Settings | | A035H |
| MODBUS/TCP Settings | To Use or Not to Use MODBUS/TCP Setting | A0B0H |
| | CPU Response Monitoring Timer | A0B0H |
| | Device Assignment | A0B2H |
| Simple CPU Communication Setting (RJ71EN71 and RnENCPU (network part)) | | A0E1H |
| Simple CPU Communication Setting (CPU Module (built-in Ethernet port part)) | | 7A10H |
| Simple Device Communication Setting | | A0E2H |
| Time Setting | | A039H |
| Timer Settings for Data Communication | | A038H |
| Security | IP Filter Settings | A03AH |
| | Disable Direct Connection with MELSOFT | A034H |
| | Do Not Respond to CPU Module Search | A034H |

| Item | | Parameter No. |
|---|---|---|
| Gateway Parameter Settings | | A013H |
| Network/Station No. <-> IP information setting | | A033H |
| Interrupt Settings | Interrupt Occurrence Factor | A014H |
| | Channel No./Connection No. | A014H |
| | Detection Method | A014H |
| | Interrupt Pointer | A014H |
| | Comment | A015H |
| IP Packet Transfer Setting | | A030H |
| Network Dynamic Routing | | A030H |
| Module Operation Mode | | 7100H |
| Redundant System Settings | System B IP Address | A03CH |
| | System B Station Number | 7110H |
| | Use of Control System IP Address | A03CH |
| | Control System IP Address | A03CH |
| | System B Module Operation Mode | 7110H |
| | Port Group Setting | A03CH |
| | System Switching Request Issuing at Disconnection Detection | A03CH |
| | Disconnection Detection Monitoring Time | A03CH |
| | System Switching Request Issue at Communication Error | A03CH |
| | User Connection | A03CH |
| | System Connection | A03CH |
| Built-in database access setting | | A0E0H |
| iQ Sensor Solution Settings | | 7A31H |
| SLMPSND Instruction Settings | | |

# 3.7 Event List

This section lists the events occurs in Ethernet.

| Event code (Hex) | Event type | Description | Cause |
|---|---|---|---|
| 00100H | System | Link-up | Link-up has occurred when the network cable connected to the external device was connected. |
| 00110H | | TCP connection communication start/end | Communication using the TCP connection with the external device was started or ended. |
| 00120H | | FTP connection start/stop | FTP connection was started from the external device or disconnected. |
| 00130H | | Receive frame error | • Receive frame error was detected.<br>• The Ethernet-equipped module was connected to the CC-Link IE Controller Network or CC-Link IE Field Network. |
| 00140H | | Time synchronization failure | No response was returned from the time information server (SNTP server) and time synchronization has failed. |
| 00400H | | Restart by a program | The module was restarted by the program. |
| 00800H | | Link-down | Link-down has occurred when network cable connected to the external device was disconnected. |
| 00901H | | Data length error | The send/receive data length exceeds the allowable range. |
| 00902H | | Initial processing error | Socket communications or communications using the fixed buffer were executed before the initial processing was completed. |
| 00903H | | Specification IP address error | Broadcast address was specified to the IP address of the external device other than Unpassive. |
| 00904H | | Socket communications transmission failed | Message send of the socket communications has failed. |
| 00905H | | Not opened | • The open processing of the external device has not been completed.<br>• The connection with the external device is closed. |
| 00906H | | Alive check error | The existence of the external device could not be checked within the response monitoring timer value. |
| 00907H | | Divided messages receive timeout error | • Not all the data could be received within the response monitoring timer value.<br>• Sufficient data for the data length could not be received.<br>• The remaining part of the message divided at the TCP/IP level could not be received within the response monitoring timer value. |
| 00908H | | IP assembly timeout error | An IP assembly timeout error has occurred. (The remaining part of the divided data could not be received and a timeout has occurred.) |
| 00909H | | TCP specification port number error | The port number used in a connection already opened is set. (For TCP/IP) |
| 0090AH | | UDP specification port number error | The port number used in a connection already opened is set. (For UDP/IP) |
| 00A00H | | System switching request issued | A system switching request occurred. |
| 00C02H | | Access abnormal response of another station | • Abnormal response was returned from another station when accessing another station.<br>• Abnormal response was returned to another station when accessed from another station. |
| 00D00H | | Fixed buffer response send failed | Response send of the communications using the fixed buffer has failed. |
| 00D01H | | Pairing open error | The open processing of the connection specified for pairing open has been already completed. |
| 00D02H | | End code error | An abnormal end response was received for communications using the fixed buffer and random access buffer. |
| 00D03H | | Response monitoring timeout error | • A response could not be received within the response monitoring timer value.<br>• The connection with the external device was closed while waiting for a response. |
| 00D04H | | Fixed buffer send failed | Message send of the fixed buffer has failed. |
| 00D05H | | Random access communication error | The parameter (start address and data word) of the data received by the random access communication is faulty. |

| Event code (Hex) | Event type | Description | Cause |
|---|---|---|---|
| 10200H | Security | Remote password lock | The lock processing of the remote password was performed. |
| 10201H | | Remote password unlock successful | The unlock processing of the remote password was succeeded. |
| 10202H | | Remote password unlock failed | The unlock processing of the remote password has failed. |
| 10300H | | Access from IP restricted with IP filter setting | Accessed from IP address restricted with the IP filter setting. |
| 10500H | | Forced invalidation setting | Forced invalidation setting was executed. |
| 10501H | | Forced invalidation cancel | Forced invalidation setting was canceled. |
| 24E00H | Operation | ERR LED off execution | The ERR LED has turned off using the output signal Y17 of the module. |
| 24E10H | Operation | Re-initial processing by buffer memory | The Ethernet port was reinitialized based on a command from the buffer memory. |

# 3.8 End Codes Returned to an External Device During Data Communications

The following table lists the error codes stored in the end code appended to a response during data communications.

| End code (Hex) | Error details and causes | Action |
|---|---|---|
| 00H | Normal completion | When each communication is normally completed, the error code 00H is stored. |
| 02H | The device range destination of devices to be read/written from/to is incorrect. | Check and correct the specified start device and the number of points. |
| 50H | • Codes for the command/response type of a subheader are not within the specifications.<br>• In communications using the fixed buffer, if the data length setting is smaller than the actual data amount, the remaining data is processed as the second data. In this case, a subheader undefined command type error may occur. | • Check and correct the command/response type set for the external device. (Because the Ethernet-equipped module automatically adds the command/response type, the user setting is not required.)<br>• Check and correct the data length. |
| 51H | In communications using the random access buffer, the start address specified by an external device has been set outside the range from 0 to 6143. | Check and correct the specified start address. |
| 52H | • In communications using the random access buffer, the start address and the number of data words (depending on the setting when reading data) specified by an external device exceeds the range from 0 to 6143.<br>• Data by the number of words specified (text) cannot be sent in one frame. (The data length value or the amount of the text sent/received are not in the allowable range.) | • Check and correct the start address and the number of data words.<br>• Correct the number of read/write points. |
| 54H | When "ASCII" has been selected for "Communication Data Code" in "Own Node Settings" under "Basic Settings" of the Ethernet-equipped module, ASCII code data which cannot be converted into binary code data has been sent from the external device. | Check and correct the data sent from the external device. |
| 55H | • When "Disable All (SLMP)" has been selected for "Enable/Disable Online Change" in "Own Node Settings" under "Basic Settings" of the Ethernet-equipped module, the external device requested a data write while the CPU module was running.<br>• While the CPU module was running, the external device requested for writing a parameter or program. | • Set "Enable All (SLMP)" for "Enable/Disable Online Change" in "Own Node Settings" under "Basic Settings" of the Ethernet-equipped module, and write data. (However, a parameter, program, or microcomputer program cannot be written while the CPU module is running.)<br>• Set the switch on the CPU module to STOP and write the data. |
| 56H | The external device specified an incorrect device. | Correct the device specification. |
| 57H | • The number of points for a command specified by an external device exceeds the maximum number of processing points (number of processes that can be executed per communication) for each process.<br>• Addresses from the start address (start device number and start step number) to the specified number of points exceed the largest addresses (device number and step number) for each process.<br>• The byte length of a command is not within the range defined by the specifications.<br>• When data is written, the set number of write data points is different from the specified number of points.<br>• A monitor request was issued even though monitoring data is not registered. | • Correct the specified points or the start address (device number and step number).<br>• Check the data length of the command and set the data again.<br>• Register the monitoring data.<br>• Data cannot be read from/written to an area with an address after the last address. Correct the specified address.<br>• Correct the block number.<br>• In the block number specification of the extension file register, do not specify a block number exceeding the range of the corresponding memory cassette size. |
| 58H | • The start address (start device number and start step number) of a command specified by an external device has been set outside the range that can be specified.<br>• For data read/write in a microcomputer program or file register (R), values exceeding the CPU module parameter setting range has been specified.<br>• A block number specified for an extension file register does not exist.<br>• A file register cannot be specified.<br>• A word device has been specified for the command for bit devices.<br>• The start number of bit devices has been specified using a value other than a multiple of 16 in the command for word devices. | • Correct the value to the one within the range that can be specified for each process.<br>• Correct the block number.<br>• Correct the device specification.<br>• Correct the command or specified device. |
| 59H | An extension file register cannot be specified. | Correct the device specification. |
| 60H | The communication time between the CPU module and the Ethernet-equipped module exceeded the response monitoring timer value. | Increase the response monitoring timer value. |

| End code (Hex) | Error details and causes | Action |
|---|---|---|
| 63H | In communications using the fixed buffer, the remote password of the port for the destination Ethernet module is in the lock status. | • After unlocking the remote password using the SLMP, perform communications using the fixed buffer.<br>• Exclude ports used for communications using the fixed buffer from the targets of the remote password check. |
| A0H | The request cannot be specified for the target connection. | • Correct the request details.<br>• Correct the open settings. |
| A1H | The request details cannot be analyzed because the text part length or the request data length is short. | Correct the text part length or the header request data length, and send the request to the Ethernet-equipped module again. |
| A2H | The request cannot be processed. | Correct the request details and command. |

# APPENDICES

# Appendix 1 Module Label

The I/O signals and buffer memory of the Ethernet-equipped module can be set using module label.

## Structure of the module label

The module label name is defined with the following structure.

"Instance name"_"Module number"."Port number"."Label name"

"Instance name"_"Module number"."Label name"

"Instance name"_"Module number"."Port number"."Label name"_D

"Instance name"_"Module number"."Label name"_D

Ex.

EN71_EE_1.stPort1.wnVal_IP_Address_D[0]

### ■Instance name

The following are the instance names of the RJ71EN71 and the RnENCPU (network part) corresponding to each network type setting.

| Model | | Instance name |
|---|---|---|
| RJ71EN71 | RnENCPU (network part) | |
| RJ71EN71 (E+E) | — | EN71_EE |
| RJ71EN71 (CCIEC) | _RJ71EN71(CCIEC) | EN71_C |
| RJ71EN71 (E+CCIEC) | _RJ71EN71(E+IEC) | EN71_EC |
| RJ71EN71 (CCIEF) | _RJ71EN71(CCIEF) | EN71_F |
| RJ71EN71 (E+CCIEF) | _RJ71EN71(E+IEF) | EN71_EF |

### ■Module number

A sequential number starting with "1" for identifying a module from the one with the same instance name.

### ■Port number

The port number is added to the module labels corresponding the buffer memory areas other than P1/P2 common area.

The following lists the characters corresponding to the port numbers.

| Port number | Added characters |
|---|---|
| P1 | stPort1 |
| P2 | stPort2 |

### ■Label name

A label name unique to the module.

### ■_D

This symbol indicates that the module label is for direct access. The label without "_D" is for refresh. The following are the differences between refresh and direct access.

| Type | Description | Access timing |
|---|---|---|
| Refresh | The values read/written from/to the module labels are reflected to the module at refresh. The execution time of the program can be shortened. | At refresh |
| Direct access | The values read/written from/to the module labels are reflected to the module immediately. Although the execution time of the program is longer than the one at refresh, the responsiveness is improved. | At writing to or reading from the module label |

A

# Appendix 2 I/O Signals

This section describes the I/O signals of the RJ71EN71 and the RnENCPU (network part) for the CPU module. The I/O signal assignment of when the start I/O number of the RJ71EN71 or the RnENCPU (network part) is "0" is listed below.

## List of I/O signals

The following table lists I/O signals. The device X is an input signal from the RJ71EN71 and the RnENCPU (network part) to the CPU module. The device Y is an output signal from the CPU module to the RJ71EN71 and the RnENCPU (network part). The I/O signals differ when the network type is Ethernet, and Q-compatible Ethernet.

### When network type is "Ethernet"

#### ■Input signals

| Device number | Signal name |
|---|---|
| X0*1 | Module failure |
| X1 to XE | Use prohibited |
| XF | Module READY |
| X10 to X19 | Use prohibited |
| X1A*2 | P ERR LED (P1) lit confirmation |
| X1B*2 | P ERR LED (P2) lit confirmation |
| X1C*2 | ERR LED lit confirmation |
| X1D to X1F | Use prohibited |

*1 This signal can be used when the P2 connector is set for CC-Link IE Field Network. For details on operation of the signal, refer to the following.
MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

*2 This signal can be used when the RJ71EN71 or RnENCPU (network part) with firmware version "90" or later is used and the P1 connector is set for Ethernet and the P2 connector is also set for Ethernet.

#### ■Output signals

| Device number | Signal name |
|---|---|
| Y0 to Y16 | Use prohibited |
| Y17*1 | ERR LED off request |
| Y18 to Y1F | Use prohibited |

*1 This signal can be used when the RJ71EN71 or RnENCPU (network part) with firmware version "90" or later is used and the P1 connector is set for Ethernet and the P2 connector is also set for Ethernet.

**Point**

- Do not use (turn on) any "use prohibited" signals as an input or output signal to the CPU module. Doing so may cause malfunction of the programmable controller system.
- For the I/O signals of the P2 connector when the P2 connector is set for CC-Link IE Controller Network, refer to the following.

MELSEC iQ-R CC-Link IE Controller Network User's Manual (Application)

- For the I/O signals of the P2 connector when the P2 connector is set for CC-Link IE Field Network, refer to the following.

MELSEC iQ-R CC-Link IE Field Network User's Manual (Application)

## When the network type is "Q Compatible Ethernet"

With the Q-compatible Ethernet, the same I/O signals as the MELSEC-Q series Ethernet interface module can be used except in some cases.

The following table lists the differences of the RJ71EN71 and MELSEC-Q series Ethernet interface modules.

| Device number | RJ71EN71 | QJ71E71-100 |
|---|---|---|
| X1C | ERR LED lit confirmation*1<br>(On: Lit, Off: —) | COM.ERR. LED lit confirmation<br>(On: Lit, Off: —) |
| Y17 | ERR LED off request*2<br>(On: At off request, Off: —) | COM.ERR. LED off request<br>(On: At off request, Off: —) |
| Y19 | Use prohibited | Initial request signal<br>(On: At request, Off: —) |

*1 This signal turns on when the ERR LED or the P ERR LED of P1 is on or flashing.

*2 The details are listed below.

· The ERR LED and the P ERR LED of P1 turn off.

· An off request continues to be issued while this signal is on. (The ERR LED and the P ERR LED of P1 do not light while on.)

· 24E00H (ERR LED off execution) is registered in event history when off process is executed.

· This signal cannot be turned off during major error.

· If the error occurs again after turning on and off, the ERR LED or P ERR LED will turn on or flash according to the details of the error.

· Currently occurring error and error history information are not cleared from the module diagnostics window and the error log areas of the buffer memory.

For details on the other I/O signals, refer to the following.

Q Corresponding Ethernet Interface Module User's Manual (Basic)

**Point**

Do not use (turn on) any "use prohibited" signals as an input or output signal to the CPU module. Doing so may cause malfunction of the programmable controller system.

# Details of I/O signals

The following shows details of I/O signals when the network type is Ethernet.

## Module ready (XF)

This signal is used to check the status of module operation preparation.

- Off: Not available for module operation
- On: Available for module operation

## P ERR LED (P1) lit confirmation (X1A)

This signal turns on when the P ERR LED (P1) on the RJ71EN71 is on or flashing. (Always on during flashing)

- Off: LED off
- On: LED on

## P ERR LED (P2) lit confirmation (X1B)

This signal turns on when the P ERR LED (P2) on the RJ71EN71 is on or flashing. (Always on during flashing)

- Off: LED off
- On: LED on

## ERR LED lit confirmation (X1C)

This signal turns on when the ERR LED on the RJ71EN71 is on or flashing. (Always on during flashing)

- Off: LED off
- On: LED on

## ERR LED off request (Y17)

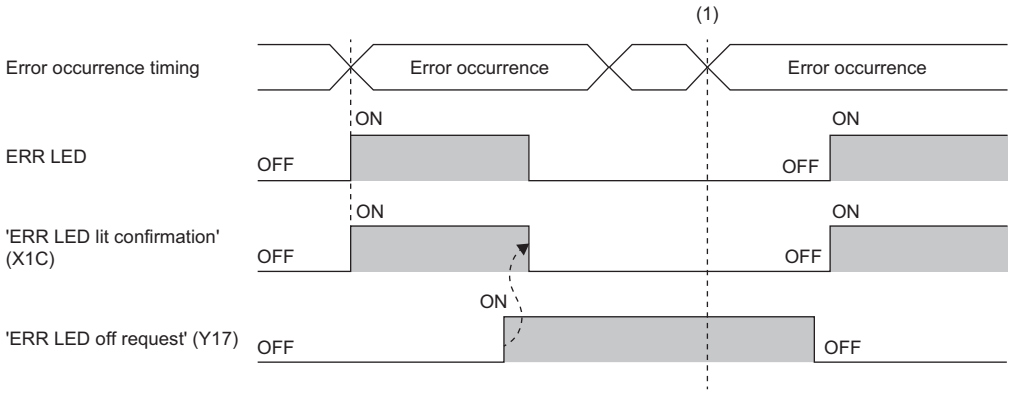
By turning on 'ERR LED off request' (Y17), the P ERR LED (P1), P ERR LED (P2), and ERR LED on the RJ71EN71 turn off.

The P ERR LED (P1), P ERR LED (P2), and ERR LED do not turn on while 'ERR LED off request' (Y17) is on.

- Off: Off request not issued
- On: Off request being issued

### ■Timing chart

The following figure shows the timing when the ERR LED turns off by 'ERR LED off request' (Y17).




(1) Since 'ERR LED off request' (Y17) is on, the ERR LED does not turn on even if an error occurs.

**Point**

- 24E00H (ERR LED off execution) is registered in event history when off process is executed.
- It cannot be turned off during major error.
- If the error occurs again after turning off 'ERR LED off request' (Y17), the ERR LED or P ERR LED will turn on or flash according to the highest-priority error, in the following order of severity: major > moderate > minor.
- Currently occurring error and error history information are not cleared from the module diagnostics window and the error log areas of the buffer memory.

# Appendix 3 Buffer Memory

The buffer memory is used for the following applications.

| Module | Application |
|---|---|
| RJ71EN71, RnENCPU (network part) | Exchanges data with the CPU module. |
| CPU module (built-in Ethernet port part) | Stores data such as the setting values for the Ethernet function and monitor values, and data used for exchanging data with the multiple CPU function. |

Buffer memory values are reset to default when the CPU module is reset or the system is powered off.

## List of buffer memory addresses

### RJ71EN71, RnENCPU (network part)

The following table lists the buffer memory addresses of the RJ71EN71 and the RnENCPU (network part) when the network type is set to "Ethernet".

For the addresses of the RJ71EN71 when the network type is set to "Q Compatible Ethernet", refer to the following.

(☞ Page 499 When using the RJ71EN71 (network type: Q-compatible Ethernet))

| P1 address (P2 address[*1]) | | Application | Name |
|---|---|---|---|
| Decimal | Hexadecimal | | |
| 0 to 3 (2000000 to 2000003) | 0H to 3H (1E8480H to 1E8483H) | System area | |
| 4 to 5 (2000004 to 2000005) | 4H to 5H (1E8484H to 1E8485H) | Own node setting status storage area | Own node IP address |
| 6 to 13 (2000006 to 2000013) | 6H to DH (1E8486H to 1E848DH) | | System area |
| 14 to 15 (2000014 to 2000015) | EH to FH (1E848EH to 1E848FH) | | Subnet mask |
| 16 to 17 (2000016 to 2000017) | 10H to 11H (1E8490H to 1E8491H) | | System area |
| 18 to 19 (2000018 to 2000019) | 12H to 13H (1E8492H to 1E8493H) | | Default gateway IP address |
| 20 to 27 (2000020 to 2000027) | 14H to 1BH (1E8494H to 1E849BH) | | System area |
| 28 to 30 (2000028 to 2000030) | 1CH to 1EH (1E849CH to 1E849EH) | | Own node MAC address |
| 31 (2000031) | 1FH (1E849FH) | | Own node network number |
| 32 (2000032) | 20H (1E84A0H) | | Station number |
| 33 (2000033) | 21H (1E84A1H) | | Transient transmission group number |
| 34 (2000034) | 22H (1E84A2H) | | Send frame setting |
| 35 (2000035) | 23H (1E84A3H) | | Jumbo frame setting |
| 36 (2000036) | 24H (1E84A4H) | | Communication speed setting |
| 37 (2000037) | 25H (1E84A5H) | | Auto-open UDP port number |
| 38 (2000038) | 26H (1E84A6H) | | MELSOFT connection TCP port number |
| 39 (2000039) | 27H (1E84A7H) | | MELSOFT connection UDP port number |
| 40 (2000040) | 28H (1E84A8H) | | MELSOFT direct connection port number |
| 41 to 99 (2000041 to 2000099) | 29H to 63H (1E84A9H to 1E84E3H) | System area | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 100 to 163<br>(2000100 to 2000163) | 64H to A3H (1E84E4H to 1E8523H) | Connection status storage area | Connection No.1 latest error code to Connection No.64 latest error code | |
| 164 to 226<br>(2000164 to 2000226) | A4H to E2H<br>(1E8524H to 1E8562H) | System area | | |
| 227<br>(2000227) | E3H<br>(1E8563H) | Error log area | Number of errors | |
| 228<br>(2000228) | E4H<br>(1E8564H) | | Error log write pointer | |
| 229<br>(2000229) | E5H<br>(1E8565H) | | Error log block 1 | Error code/end code |
| 230<br>(2000230) | E6H<br>(1E8566H) | | | Subheader |
| 231<br>(2000231) | E7H<br>(1E8567H) | | | Command code |
| 232<br>(2000232) | E8H<br>(1E8568H) | | | Connection No. |
| 233<br>(2000233) | E9H<br>(1E8569H) | | | Own station port number |
| 234, 235<br>(2000234, 2000235) | EAH, EBH<br>(1E856AH, 1E856BH) | | | Communication destination IP address |
| 236<br>(2000236) | ECH<br>(1E856CH) | | | Communication destination port No. |
| 237<br>(2000237) | EDH<br>(1E856DH) | | | System area |
| 238 to 372<br>(2000238 to 2000372) | EEH to 174H<br>(1E856EH to 1E85F4H) | | Error log block 2 to 16 | (Same as Error log block 1) |
| 373 to 999<br>(2000373 to 2000999) | 175H to 3E7H<br>(1E85F5H to 1E8867H) | System area | | |
| 1000<br>(2001000) | 3E8H<br>(1E8868H) | System port latest error code storage area | FTP server latest error code | |
| 1001<br>(2001001) | 3E9H<br>(1E8869H) | | MELSOFT direct connection latest error code | |
| 1002 to 4999<br>(2001002 to 2004999) | 3EAH to 1387H<br>(1E886AH to 1E9807H) | System area | | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 5000 to 5001 (2005000 to 2005001) | 1388H to 1389H (1E9808H to 1E9809H) | Status for each protocol | IP packet | Received packet total count |
| 5002 to 5003 (2005002 to 2005003) | 138AH to 138BH (1E980AH to 1E980BH) | | | Received packet checksum error discard count |
| 5004 to 5005 (2005004 to 2005005) | 138CH to 138DH (1E980CH to 1E980DH) | | | Sent packet total count |
| 5006 to 5021 (2005006 to 2005021) | 138EH to 139DH (1E980EH to 1E981DH) | | | System area |
| 5022 to 5023 (2005022 to 2005023) | 139EH to 139FH (1E981EH to 1E981FH) | | | Simultaneous transmission error detection count (receive buffer full count) |
| 5024 to 5028 (2005024 to 2005028) | 13A0H to 13A4H (1E9820H to 1E9824H) | | | System area |
| 5029 (2005029) | 13A5H (1E9825H) | | | Receive abort count |
| 5030 to 5039 (2005030 to 2005039) | 13A6H to 13AFH (1E9826H to 1E982FH) | | System area | |
| 5040 to 5041 (2005040 to 2005041) | 13B0H to 13B1H (1E9830H to 1E9831H) | | ICMP packet | Received packet total count |
| 5042 to 5043 (2005042 to 2005043) | 13B2H to 13B3H (1E9832H to 1E9833H) | | | Received packet checksum error discard count |
| 5044 to 5045 (2005044 to 2005045) | 13B4H to 13B5H (1E9834H to 1E9835H) | | | Sent packet total count |
| 5046 to 5047 (2005046 to 2005047) | 13B6H to 13B7H (1E9836H to 1E9837H) | | | Received echo request total count |
| 5048 to 5049 (2005048 to 2005049) | 13B8H to 13B9H (1E9838H to 1E9839H) | | | Sent echo reply total count |
| 5050 to 5051 (2005050 to 2005051) | 13BAH to 13BBH (1E983AH to 1E983BH) | | | Sent echo request total count |
| 5052 to 5053 (2005052 to 2005053) | 13BCH to 13BDH (1E983CH to 1E983DH) | | | Received echo reply total count |
| 5054 to 5079 (2005054 to 2005079) | 13BEH to 13D7H (1E983EH to 1E9857H) | | System area | |
| 5080 to 5081 (2005080 to 2005081) | 13D8H to 13D9H (1E9858H to 1E9859H) | | TCP packet | Received packet total count |
| 5082 to 5083 (2005082 to 2005083) | 13DAH to 13DBH (1E985AH to 1E985BH) | | | Received packet checksum error discard count |
| 5084 to 5085 (2005084 to 2005085) | 13DCH to 13DDH (1E985CH to 1E985DH) | | | Sent packet total count |
| 5086 to 5119 (2005086 to 2005119) | 13DEH to 13FFH (1E985EH to 1E987FH) | | System area | |
| 5120 to 5121 (2005120 to 2005121) | 1400H to 1401H (1E9880H to 1E9881H) | | UDP packet | Received packet total count |
| 5122 to 5123 (2005122 to 2005123) | 1402H to 1403H (1E9882H to 1E9883H) | | | Received packet checksum error discard count |
| 5124 to 5125 (2005124 to 2005125) | 1404H to 1405H (1E9884H to 1E9885H) | | | Sent packet total count |
| 5126 to 5159 (2005126 to 2005159) | 1406H to 1427H (1E9886H to 1E98A7H) | | System area | |
| 5160 to 5161 (2005160 to 2005161) | 1428H to 1429H (1E98A8H to 1E98A9H) | Status for each protocol | Receiving error | Framing error count |
| 5162 to 5163 (2005162 to 2005163) | 142AH to 142BH (1E98AAH to 1E98ABH) | | | Receive FIFO overflow count |
| 5164 to 5165 (2005164 to 2005165) | 142CH to 142DH (1E98ACH to 1E98ADH) | | | CRC error count |
| 5166 to 5188 (2005166 to 2005188) | 142EH to 1444H (1E98AEH to 1E98C4H) | | System area | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 5189<br>(2005189) | 1445H<br>(1E98C5H) | Own node operation status storage area | LED status | |
| 5190<br>(2005190) | 1446H<br>(1E98C6H) | | System area | |
| 5191<br>(2005191) | 1447H<br>(1E98C7H) | | Hub connection status area | Communication mode |
| 5192<br>(2005192) | 1448H<br>(1E98C8H) | | | Connection status |
| 5193<br>(2005193) | 1449H<br>(1E98C9H) | | | Communication speed |
| 5194<br>(2005194) | 144AH<br>(1E98CAH) | | | Disconnection count |
| 5195 to 5199<br>(2005195 to 2005199) | 144BH to 144FH<br>(1E98CBH to 1E98CFH) | | System area | |
| 5200<br>(2005200) | 1450H<br>(1E98D0H) | | IP address duplication status storage area | Same IP address detection flag |
| 5201 to 5203<br>(2005201 to 2005203) | 1451H to 1453H<br>(1E98D1H to 1E98D3H) | | | MAC address of the station already connected to the network |
| 5204 to 5206<br>(2005204 to 2005206) | 1454H to 1456H<br>(1E98D4H to 1E98D6H) | | | MAC address of the station with the IP address already used |
| 5207 to 5300<br>(2005207 to 2005300) | 1457H to 14B4H<br>(1E98D7H to 1E9934H) | System area | | |
| 5301<br>(2005301) | 14B5H<br>(1E9935H) | Area for sending/ receiving instructions | RECV instruction execution request | |
| 5302 to 5322<br>(2005302 to 2005322) | 14B6H to 14CAH<br>(1E9936H to 1E994AH) | | System area | |
| 5323<br>(2005323) | 14CBH<br>(1E994BH) | | Link dedicated instruction | ZNRD instruction execution result |
| 5324<br>(2005324) | 14CCH<br>(1E994CH) | | | System area |
| 5325<br>(2005325) | 14CDH<br>(1E994DH) | | | ZNWR instruction execution result |
| 5326 to 5625<br>(2005326 to 2005625) | 14CEH to 15F9H<br>(1E994EH to 1E9A79H) | System area | | |
| 5626<br>(2005626) | 15FAH<br>(1E9A7AH) | Remote password lock status storage area | Remote password lock status (connection No.1 to 16) | |
| 5627<br>(2005627) | 15FBH<br>(1E9A7BH) | | Remote password lock status (connection No.17 to 32) | |
| 5628<br>(2005628) | 15FCH<br>(1E9A7CH) | | Remote password lock status (connection No.33 to 48) | |
| 5629<br>(2005629) | 15FDH<br>(1E9A7DH) | | Remote password lock status (connection No.49 to 64) | |
| 5630<br>(2005630) | 15FEH<br>(1E9A7EH) | | Remote password lock status system port | |
| 5631 to 5645<br>(2005631 to 2005645) | 15FFH to 160DH<br>(1E9A7FH to 1E9A8DH) | System area | | |
| 5646<br>(2005646) | 160EH<br>(1E9A8EH) | Forced connection invalidation setting area | Forced connection invalidation (connection No.1 to 16) | |
| 5647<br>(2005647) | 160FH<br>(1E9A8FH) | | Forced connection invalidation (connection No.17 to 32) | |
| 5648<br>(2005648) | 1610H<br>(1E9A90H) | | Forced connection invalidation (connection No.33 to 48) | |
| 5649<br>(2005649) | 1611H<br>(1E9A91H) | | Forced connection invalidation (connection No.49 to 64) | |
| 5650<br>(2005650) | 1612H<br>(1E9A92H) | | Forced connection invalidation system port | |
| 5651 to 8299<br>(2005651 to 2008299) | 1613H to 206BH<br>(1E9A93H to 1EA4EBH) | System area | | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 8300 (2008300) | 206CH (1EA4ECH) | Predefined protocol support function execution status check area | Connection No.1 | Protocol execution status |
| 8301 (2008301) | 206DH (1EA4EDH) | | | System area |
| 8302 to 8317 (2008302 to 2008317) | 206EH to 207DH (1EA4EEH to 1EA4FDH) | | | Received data verification result (receive packet No.1 to 16) |
| 8318 (2008318) | 207EH (1EA4FEH) | | | Protocol execution count |
| 8319 (2008319) | 207FH (1EA4FFH) | | | Protocol cancellation specification |
| 8320 to 8335 (2008320 to 2008335) | 2080H to 208FH (1EA500H to 1EA50FH) | | | System area |
| 8336 to 8875 (2008336 to 2008875) | 2090H to 22ABH (1EA510H to 1EA72BH) | | Connection No.2 to Connection No.16 | |
| 8876 to 10999 (2008876 to 2010999) | 22ACH to 2AF7H (1EA72CH to 1EAF77H) | System area | | |
| 11000 (2011000) | 2AF8H (1EAF78H) | Time setting function (SNTP) area | Time setting function operation result | |
| 11001 (2011001) | 2AF9H (1EAF79H) | | Time setting function execution time | (Year) |
| 11002 (2011002) | 2AFAH (1EAF7AH) | | | (Month) |
| 11003 (2011003) | 2AFBH (1EAF7BH) | | | (Day) |
| 11004 (2011004) | 2AFCH (1EAF7CH) | | | (Hour) |
| 11005 (2011005) | 2AFDH (1EAF7DH) | | | (Minute) |
| 11006 (2011006) | 2AFEH (1EAF7EH) | | | (Second) |
| 11007 (2011007) | 2AFFH (1EAF7FH) | | | (Day of the week) |
| 11008 (2011008) | 2B00H (1EAF80H) | | Time setting function required response time | |
| 11009 (2011009) | 2B01H (1EAF81H) | | Time setting function (SNTP client) execution | |
| 11010 to 11049 (2011010 to 2011049) | 2B02H to 2B29H (1EAF82H to 1EAFA9H) | System area | | |
| 11050 to 11051 (2011050 to 2011051) | 2B2AH to 2B2BH (1EAFAAH to 1EAFABH) | IP packet transfer function area | IP packet transfer latest data volume | |
| 11052 to 11053 (2011052 to 2011053) | 2B2CH to 2B2DH (1EAFACH to 1EAFADH) | | IP packet transfer maximum data volume | |
| 11054 to 11499 (2011054 to 2011499) | 2B2EH to 2CEBH (1EAFAEH to 1EB16BH) | System area | | |
| 11500 to 11514 (2011500 to 2011514) | 2CECH to 2CFAH (1EB16CH to 1EB17AH) | Dynamic routing function area | Communication path determination status | |
| 11515 to 11599 (2011515 to 2011599) | 2CFBH to 2D4FH (1EB17BH to 1EB1CFH) | System area | | |
| 11600 to 11663 (2011600 to 2011663) | 2D50H to 2D8FH (1EB1D0H to 1EB20FH) | Remote password function monitoring area | Connection No.1 continuous unlock failure count to Connection No.64 continuous unlock failure count | |
| 11664 (2011664) | 2D90H (1EB210H) | | Auto-open UDP port continuous unlock failure count | |
| 11665 (2011665) | 2D91H (1EB211H) | | MELSOFT transmission port (UDP/IP) continuous unlock failure count | |
| 11666 (2011666) | 2D92H (1EB212H) | | MELSOFT transmission port (TCP/IP) continuous unlock failure count | |
| 11667 (2011667) | 2D93H (1EB213H) | | FTP transmission port (TCP/IP) continuous unlock failure count | |
| 11668 (2011668) | 2D94H (1EB214H) | | MELSOFT direct connection continuous unlock failure count | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 11669<br>(2011669) | 2D95H<br>(1EB215H) | System area | | |
| 11670 to 11671<br>(2011670 to 2011671) | 2D96H to 2D97H<br>(1EB216H to 1EB217H) | Area for both systems identical IP address setting function | Control system IP address | |
| 11672 to 11699<br>(2011672 to 2011699) | 2D98H to 2DB3H<br>(1EB218H to 1EB233H) | System area | | |
| 11700 to 11701<br>(2011700 to 2011701) | 2DB4H to 2DB5H<br>(1EB234H to 1EB235H) | IP address change function area | IP Address | IP address |
| 11702 to 11703<br>(2011702 to 2011703) | 2DB6H to 2DB7H<br>(1EB236H to 1EB237H) | | | Subnet mask |
| 11704 to 11705<br>(2011704 to 2011705) | 2DB8H to 2DB9H<br>(1EB238H to 1EB239H) | | | Default Gateway |
| 11706<br>(2011706) | 2DBAH<br>(1EB23AH) | | Communications by Network No./ Station No. | Network number |
| 11707<br>(2011707) | 2DBBH<br>(1EB23BH) | | | Station number |
| 11708<br>(2011708) | 2DBCH<br>(1EB23CH) | | | Transient transmission group number |
| 11709<br>(2011709) | 2DBDH<br>(1EB23DH) | | IP address storage area write request | |
| 11710<br>(2011710) | 2DBEH<br>(1EB23EH) | | IP address storage area write error | |
| 11711<br>(2011711) | 2DBFH<br>(1EB23FH) | | IP address storage area writing error cause | |
| 11712<br>(2011712) | 2DC0H<br>(1EB240H) | | IP address storage area clear request | |
| 11713<br>(2011713) | 2DC1H<br>(1EB241H) | | IP address storage area clear error | |
| 11714<br>(2011714) | 2DC2H<br>(1EB242H) | | IP address storage area clear error cause | |
| 11715<br>(2011715) | 2DC3H<br>(1EB243H) | | IP address change function operating status | |
| 11716 to 19999<br>(2011716 to 2019999) | 2DC4H to 4E1FH<br>(1EB244H to 1ED29FH) | System area | | |
| 20000 to 26143<br>(2020000 to 2026143) | 4E20H to 661FH<br>(1ED2A0H to 1EEA9FH) | Random access buffer area | Random access buffer | |
| 26144 to 65534<br>(2026144 to 2065534) | 6620H to FFFEH<br>(1EEAA0H to 1F847EH) | System area | | |
| 65535<br>(2065535) | FFFFH<br>(1F847FH) | Network type information area | Network type information | |
| 65536<br>(2065536) | 10000H<br>(1F8480H) | Area for communication using a fixed buffer | Fixed buffer No.1 | Data length |
| 65537 to 70655<br>(2065537 to 2070655) | 10001H to 113FFH<br>(1F8481H to 1F987FH) | | | Fixed buffer data |
| 70656 to 147455<br>(2070656 to 2147455) | 11400H to 23FFFH<br>(1F9880H to 20C47FH) | | Fixed buffer No.2 to Fixed buffer No.16 | (The bit configuration is the same as Fixed buffer No.1) |
| 147456 to 720895<br>(2147456 to 2720895) | 24000H to AFFFFH<br>(20C480H to 29847FH) | System area | | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 720896 (2720896) | B0000H (298480H) | Error log area for MODBUS/TCP function | Number of errors | |
| 720897 (2720897) | B0001H (298481H) | | Error log write pointer | |
| 720898 (2720898) | B0002H (298482H) | | Error log 1 | Detail error code |
| 720899 (2720899) | B0003H (298483H) | | | Abnormal response code |
| 720900 (2720900) | B0004H (298484H) | | | Function code |
| 720901 (2720901) | B0005H (298485H) | | | Own station port number |
| 720902 to 720903 (2720902 to 2720903) | B0006H to B0007H (298486H to 298487H) | | | Destination IP address |
| 720904 (2720904) | B0008H (298488H) | | | Destination port number |
| 720905 (2720905) | B0009H (298489H) | | | System area |
| 720906 to 721153 (2720906 to 2721153) | B000AH to B0101H (29848AH to 298581H) | | Error log 2 to Error log 32 | (Same as Error log 1) |
| 721154 to 721895 (2721154 to 2721895) | B0102H to B03E7H (298582H to 298867H) | System area | | |
| 721896 to 721899 (2721896 to 2721899) | B03E8H to B03EBH (298868H to 29886BH) | Simple CPU communication area | Request to start communication at request for each setting number | |
| 721900 to 721903 (2721900 to 2721903) | B03ECH to B03EFH (29886CH to 29886FH) | | Request to stop fixed interval communication for each setting number | |
| 721904 to 721907 (2721904 to 2721907) | B03F0H to B03F3H (298870H to 298873H) | | Request to restart fixed interval communication for each setting number | |
| 721908 to 721911 (2721908 to 2721911) | B03F4H to B03F7H (298874H to 298877H) | | Execution status flag for each setting number | |
| 721912 to 721915 (2721912 to 2721915) | B03F8H to B03FBH (298878H to 29887BH) | | Preparation completion for each setting number | |
| 721916 to 721935 (2721916 to 2721935) | B03FCH to B040FH (29887CH to 29888FH) | | System area | |
| 721936 to 721999 (2721936 to 2721999) | B0410H to B044FH (298890H to 2988CFH) | | Simple CPU communication status for each setting number | |
| 722000 to 722063 (2722000 to 2722063) | B0450H to B048FH (2988D0H to 29890FH) | | Simple CPU communication error code for each setting number | |
| 722064 to 722127 (2722064 to 2722127) | B0490H to B04CFH (298910H to 29894FH) | | Abnormal response code for each setting number | |
| 722128 to 722191 (2722128 to 2722191) | B04D0H to B050FH (298950H to 29898FH) | | Execution interval (current value) for each setting number | |
| 722192 to 722499 (2722192 to 2722499) | B0510H to B0643H (298990H to 298AC3H) | | System area | |
| 722500 to 1246787 (2722500 to 3246787) | B0644H to 130643H (298AC4H to 318AC3H) | Simple CPU communication area/ Simple device communication area | User setting area | |

| P1 address (P2 address[*1]) | | Application | Name |
|---|---|---|---|
| Decimal | Hexadecimal | | |
| 1246788 to 1247299 (3246788 to 3247299) | 130644H to 130843H (318AC4H to 318CC3H) | Simple CPU communication area | System area |
| 1247300 to 1247327 (3247300 to 3247327) | 130844H to 130859H (318CC4H to 318CDFH) | | Request to start communication at request for each setting number |
| 1247328 to 1247355 (3247328 to 3247355) | 130860H to 13087BH (318CE0H to 318CFBH) | | Request to stop fixed interval communication for each setting number |
| 1247356 to 1247383 (3247356 to 3247383) | 13087CH to 130897H (318CFCH to 318D17H) | | Request to restart fixed interval communication for each setting number |
| 1247384 to 1247411 (3247384 to 3247411) | 130898H to 1308B3H (318D18H to 318D33H) | | Execution status flag for each setting number |
| 1247412 to 1247439 (3247412 to 3247439) | 1308B4H to 1308CFH (318D34H to 318D4FH) | | Preparation completion for each setting number |
| 1247440 to 1247459 (3247440 to 3247459) | 1308D0H to 1308E3H (318D50H to 318D63H) | | System area |
| 1247460 to 1247907 (3247460 to 3247907) | 1308E4H to 130AA3H (318D64H to 318F23H) | | Simple CPU communication status for each setting number |
| 1247908 to 1248355 (3247908 to 3248355) | 130AA4H to 130C63H (318F24H to 3190E3H) | | Simple CPU communication error code for each setting number |
| 1248356 to 1248803 (3248356 to 3248803) | 130C64H to 130E23H (3190E4H to 3192A3H) | | Abnormal response code for each setting number |
| 1248804 to 1249251 (3248804 to 3249251) | 130E24H to 130FE3H (3192A4H to 319463H) | | Execution interval (current value) for each setting number |
| 1249252 to 1249299 (3249252 to 3249299) | 130FE4H to 131013H (319464H to 319493H) | System area | |

| P1 address (P2 address[*1]) | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 1249300<br>(3249300) | 131014H<br>(319494H) | Simple device communication area | Function start/stop instruction | |
| 1249301<br>(3249301) | 131015H<br>(319495H) | | System area | |
| 1249302 to 1249303<br>(3249302 to 3249303) | 131016H to 131017H<br>(319496H to 319497H) | | Setting No.1 | IP address |
| 1249304<br>(3249304) | 131018H<br>(319498H) | | | Port number |
| 1249305<br>(3249305) | 131019H<br>(319499H) | | | Protocol number of error |
| 1249306<br>(3249306) | 13101AH<br>(31949AH) | | | Error code for simple device communication |
| 1249307<br>(3249307) | 13101BH<br>(31949BH) | | | Error clear request |
| 1249308<br>(3249308) | 13101CH<br>(31949CH) | | | Ready |
| 1249309<br>(3249309) | 13101DH<br>(31949DH) | | | Status of connection |
| 1249310 to 1249311<br>(3249310 to 3249311) | 13101EH to 13101FH<br>(31949EH to 31949FH) | | | Number of successful completion |
| 1249312 to 1249313<br>(3249312 to 3249313) | 131020H to 131021H<br>(3194A0H to 3194A1H) | | | Number of error completion |
| 1249314 to 1249315<br>(3249314 to 3249315) | 131022H to 131023H<br>(3194A2H to 3194A3H) | | | Retry count |
| 1249316<br>(3249316) | 131024H<br>(3194A4H) | | | Protocol number currently executing |
| 1249317<br>(3249317) | 131025H<br>(3194A5H) | | | Verification mismatch protocol number |
| 1249318 to 1249333<br>(3249318 to 3249333) | 131026H to 131035H<br>(3194A6H to 3194B5H) | | | Verification mismatch information (receive packet numbers 1 to 16) |
| 1249334 to 1249347<br>(3249334 to 3249347) | 131036H to 131043H<br>(3194B6H to 3194C3H) | | | System area |
| 1249348 to 1249379<br>(3249348 to 3249379) | 131044H to 131063H<br>(3194C4H to 3194E3H) | | | Start request for each protocol |
| 1249380 to 1249411<br>(3249380 to 3249411) | 131064H to 131083H<br>(3194E4H to 319503H) | | | Execution completion for each protocol |
| 1249412 to 1249443<br>(3249412 to 3249443) | 131084H to 1310A3H<br>(319504H to 319523H) | | | Error for each protocol |
| 1249444 to 1249475<br>(3249444 to 3249475) | 1310A4H to 1310C3H<br>(319524H to 319543H) | | | Execution status for each protocol |
| 1249476 to 1249987<br>(3249476 to 3249987) | 1310C4H to 1312C3H<br>(319544H to 319743H) | | | Execution interval (current value) for each protocol |
| 1249988 to 1250499<br>(3249988 to 3250499) | 1312C4H to 1314C3H<br>(319744H to 319943H) | | | Execution interval (maximum value) for each protocol |
| 1250500 to 1325973<br>(3250500 to 3325973) | 1314C4H to 143B95H<br>(319944H to 32C015H) | | Setting No.2 to 64 | (Same as setting No.1) |
| 1325974 to 18999999<br>(3325974 to 3899999) | 143B96H to 1CFDDFH<br>(32C016H to 3B825FGH) | System area | | |

*1 The P2 address cannot be used for the RnENCPU (network part).

## ■P1/P2 common area

| Address | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 1900000 to 1900007 | 1CFDE0H to 1CFDE7H | Ethernet PORT1/2 common information | Open completion signal | |
| 1900008 to 1900015 | 1CFDE8H to 1CFDEFH | | Open request signal | |
| 1900016 to 1900023 | 1CFDF0H to 1CFDF7H | | Socket/fixed buffer reception status signal | |
| 1900024 | 1CFDF8H | | Initial status | |
| 1900025 | 1CFDF9H | | Initial error code | |
| 1900026 to 1900029 | 1CFDFAH to 1CFDFDH | System area | | |
| 1900030*1 | 1CFDFEH | Receive buffer status storage area | State of receive buffer | |
| 1900031 to 1901001 | 1CFDFFH to 1D01C9H | System area | | |
| 1901002 | 1D01CAH | Predefined protocol support function check area | Predefined protocol ready | |
| 1901003 to 1901019 | 1D01CBH to 1D01DBH | System area | | |
| 1901020 | 1D01DCH | Predefined protocol setting data check area | Predefined protocol setting data error information | Protocol number |
| 1901021 | 1D01DDH | | | Setting type |
| 1901022 | 1D01DEH | | | Packet number |
| 1901023 | 1D01DFH | | | Element number |
| 1901024 | 1D01E0H | | Number of registered predefined protocols | |
| 1901025 to 1901031 | 1D01E1H to 1D01E7H | | System area | |
| 1901032 to 1901047 | 1D01E8H to 1D01F7H | | Predefined protocol registration | |
| 1901048 to 1901999 | 1D01F8H to 1D05AFH | System area | | |
| 1902000 to 1904047 | 1D05B0H to 1D0DAFH | Send/receive area for predefined protocol support function | Send/receive area for predefined protocol support function | |
| 1904048 to 1921023 | 1D0DB0H to 1D4FFFH | System area | | |
| 1921024 to 1929215 | 1D5000H to 1D6FFFH | User setting area for MODBUS/TCP function | | |
| 1929216 to 1950031 | 1D7000H to 1DC14FH | System area | | |
| 1950032 to 1950039 | 1DC150H to 1DC157H | Discard received data at CPU STOP setting area | | |
| 1950040 | 1DC158H | Setting area for re-initial processing | Reinitialization specification | |
| 1950041 to 1950042 | 1DC159H to 1DC15AH | | Reinitialization specification error factor | |
| 1950043 to 1950044 | 1DC15BH to 1DC15CH | | Reinitialization specification change target | |
| 1950045 to 1950048 | 1DC15DH to 1DC160H | | Reinitialization Own IP Address | |
| 1950049 to 1950050 | 1DC161H to 1DC162H | | Communication condition setting (Ethernet Operation Setting) area | |
| 1950051 to 1999999 | 1DC163H to 1E847FH | System area | | |

*1 Availability for the buffer memory depends on the firmware version. (☞ Page 586 Added and Enhanced Functions)

**Point**

- Do not write any data to the system area. Doing so may cause malfunction of the programmable controller system.
- If the value in an area of one word in size becomes equal to or higher than 65536, the count stops at 65535 (FFFFH).

A

## CPU module (built-in Ethernet port part)

The following table lists the buffer memory addresses related to the Ethernet function of the CPU module (built-in Ethernet port part).

| Address | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 50 to 51 | 32H to 33H | Own node setting status storage area | Own node IP address | |
| 60 to 61 | 3CH to 3DH | | Subnet mask | |
| 64 to 65 | 40H to 41H | | Default gateway IP address | |
| 74 to 76 | 4AH to 4CH | | Own node MAC address | |
| 77 | 4DH | | Own node network number | |
| 78 | 4EH | | Station number | |
| 80 | 50H | | Send frame setting | |
| 82 | 52H | | Communication speed setting | |
| 83 | 53H | | Auto-open UDP port number | |
| 84 | 54H | | MELSOFT connection TCP port number | |
| 85 | 55H | | MELSOFT connection UDP port number | |
| 86 | 56H | | MELSOFT direct connection port number | |
| 87 | 57H | | Network number/Station number setting status | |
| 100 | 64H | Communication load status of the built-in Ethernet port | Receive packet count per unit time (one second) | Receive packet count (current) per unit time (one second) |
| 101 | 65H | | | Receive packet count (maximum) per unit time (one second) |
| 102 | 66H | | Send packet count per unit time (one second) | Send packet count (current) per unit time (one second) |
| 103 | 67H | | | Send packet count (maximum) per unit time (one second) |
| 130 to 145 | 82H to 91H | Connection status storage area | Connection No.1 latest error code to Connection No.16 latest error code | |
| 146 | 92H | System port latest error code storage area | FTP server latest error code | |
| 147 | 93H | | MELSOFT direct connection latest error code | |
| 148 | 94H | | FTP client latest error code | |
| 150 to 151 | 96H to 97H | Status for each protocol | IP packet | Received packet total count |
| 152 to 153 | 98H to 99H | | | Received packet checksum error discard count |
| 154 to 155 | 9AH to 9BH | | | Sent packet total count |
| 172 to 173 | ACH to ADH | | | Simultaneous transmission error detection count<br>(Receive buffer full count) |
| 179 | B3H | | | Receive abort count |
| 182 to 183 | B6H to B7H | | ICMP packet | Received packet total count |
| 184 to 185 | B8H to B9H | | | Received packet checksum error discard count |
| 186 to 187 | BAH to BBH | | | Sent packet total count |
| 188 to 189 | BCH to BDH | | | Received echo request total count |
| 190 to 191 | BEH to BFH | | | Sent echo reply total count |
| 192 to 193 | C0H to C1H | | | Sent echo request total count |
| 194 to 195 | C2H to C3H | | | Received echo reply total count |
| 196 to 197 | C4H to C5H | | TCP packet | Received packet total count |
| 198 to 199 | C6H to C7H | | | Received packet checksum error discard count |
| 200 to 201 | C8H to C9H | | | Sent packet total count |
| 204 to 205 | CCH to CDH | | UDP packet | Received packet total count |
| 206 to 207 | CEH to CFH | | | Received packet checksum error discard count |
| 208 to 209 | D0H to D1H | | | Sent packet total count |
| 224 to 225 | E0H to E1H | | Receiving error | Framing error count |
| 226 to 227 | E2H to E3H | | | Receive FIFO overflow count |
| 228 to 229 | E4H to E5H | | | CRC error count |

| Address | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 242 | F2H | Own node operation status storage area | Hub connection status area | Communication mode |
| 243 | F3H | | | Connection status |
| 244 | F4H | | | Communication speed |
| 245 | F5H | | | Disconnection count |
| 251 | FBH | | IP address duplication status storage area | Same IP address detection flag |
| 252 to 254 | FCH to FEH | | | MAC address of the station already connected to the network |
| 255 to 257 | FFH to 101H | | | MAC address of the station with the IP address already used |
| 270 | 10EH | Remote password lock status storage area | Remote password lock status connection No.1 to 16 | |
| 271 | 10FH | | Remote password lock status system port | |
| 275 | 113H | Discard received data at CPU STOP setting area | Discard received data connection No.1 to 16 | |
| 280 | 118H | Forced connection invalidation setting area | Forced connection invalidation connection No.1 to 16 | |
| 281 | 119H | | Forced connection invalidation system port | |
| 290 | 122H | Time setting function (SNTP) area | Time setting function operation result | |
| 291 | 123H | | Time setting function execution time | (Year) |
| 292 | 124H | | | (Month) |
| 293 | 125H | | | (Day) |
| 294 | 126H | | | (Hour) |
| 295 | 127H | | | (Minute) |
| 296 | 128H | | | (Second) |
| 297 | 129H | | | (Day of the week) |
| 298 | 12AH | | Time setting function required response time | |
| 299 | 12BH | | Time setting function (SNTP client) execution | |
| 310 to 311 | 136H to 137H | IP packet transfer function area | IP packet transfer latest data volume | |
| 312 to 313 | 138H to 139H | | IP packet transfer maximum data volume | |
| 320 to 335 | 140H to 14FH | Remote password function monitoring area | Connection No.1 continuous unlock failure count to Connection No.16 continuous unlock failure count | |
| 336 | 150H | | Auto-open UDP port continuous unlock failure count | |
| 337 | 151H | | MELSOFT transmission port (UDP/IP) continuous unlock failure count | |
| 338 | 152H | | MELSOFT transmission port (TCP/IP) continuous unlock failure count | |
| 339 | 153H | | FTP transmission port (TCP/IP) continuous unlock failure count | |
| 340 | 154H | | MELSOFT direct connection continuous unlock failure count | |
| 350 | 15EH | Predefined protocol support function execution status check area | Connection No.1 | Protocol execution status |
| 351 | 15FH | | | System area |
| 352 to 367 | 160H to 16FH | | | Received data verification result (receive packet No.1 to 16) |
| 368 | 170H | | | Protocol execution count |
| 369 | 171H | | | Protocol cancellation specification |
| 370 to 669 | 172H to 29DH | | Connection No.2 to Connection No.16 | |
| 680 | 2A8H | Open completion signal | Open completion signal | |
| 681 | 2A9H | Open request signal | Open request signal | |
| 682 | 2AAH | Socket communications reception status signal | Socket communications reception status signal | |
| 683 | 2ABH | Initial status | Initial status | |
| 684 | 2ACH | Initial error code | Initial error code | |
| 692 | 2B4H | Predefined protocol support function check area | Predefined protocol ready | |

A

| Address | | Application | Name | |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 710 | 2C6H | Predefined protocol setting data check area | Predefined protocol setting data error information | Protocol number |
| 711 | 2C7H | | | Setting type |
| 712 | 2C8H | | | Packet number |
| 713 | 2C9H | | | Element number |
| 714 | 2CAH | | Number of registered predefined protocols | |
| 722 to 729 | 2D2H to 2D9H | | Predefined protocol registration | |
| 740 to 741 | 2E4H to 2E5H | Area for both systems identical IP address setting function | Control system IP address | |
| 760 | 2F8 | Socket communications (TCP/IP) | Connection close wait time setting area for socket communications (TCP/IP) | Enabling/disabling connection close wait time setting |
| 761 | 2F9 | | | Connection close wait time setting timer value |
| 770 to 785 | 302H to 311H | Connection status storage area | Connection No.1 latest error code (communication error) to Connection No.16 latest error code (communication error) | |
| 786 | 312H | | FTP server latest error code (communication error) | |
| 787 | 313H | | MELSOFT direct connection latest error code (communication error) | |
| 788 | 314H | | FTP client latest error code (communication error) | |
| 789 | 315H | | Auto-open UDP port latest error code (communication error) | |
| 790 | 316H | | MELSOFT connection auto-open UDP port latest error code (communication error) | |
| 791 | 317H | | MELSOFT connection auto-open TCP port latest error code (communication error) | |
| 792 | 318H | | Insufficient send buffer or memory count | |
| 1090 to 1093 | 442H to 445H | Simple CPU communication function | Request to start communication at request for each setting number | |
| 1094 | 446H | | Request to stop fixed interval communication for each setting number | Stop request setting No.1 to 16 |
| 1095 | 447H | | | Stop request setting No.17 to 32 |
| 1096 | 448H | | | Stop request setting No.33 to 48 |
| 1097 | 449H | | | Stop request setting No.49 to 64 |
| 1098 | 44AH | | Request to restart fixed interval communication for each setting number | Restart request setting No.1 to 16 |
| 1099 | 44BH | | | Restart request setting No.17 to 32 |
| 1100 | 44CH | | | Restart request setting No.33 to 48 |
| 1101 | 44DH | | | Restart request setting No.49 to 64 |
| 1102 | 44EH | | Execution status flag for each setting number | Execution status flag setting No.1 to 16 |
| 1103 | 44FH | | | Execution status flag setting No.17 to 32 |
| 1104 | 450H | | | Execution status flag setting No.33 to 48 |
| 1105 | 451H | | | Execution status flag setting No.49 to 64 |
| 1106 | 452H | | Preparation completion for each setting number | Preparation completion setting No.1 to 16 |
| 1107 | 453H | | | Preparation completion setting No.17 to 32 |
| 1108 | 454H | | | Preparation completion setting No.33 to 48 |
| 1109 | 455H | | | Preparation completion setting No.49 to 64 |
| 1130 to 1193 | 46AH to 4A9H | | Simple CPU communication status for each setting number | Simple CPU communication status setting No.1 to Simple CPU communication status setting No.64 |
| 1194 to 1257 | 4AAH to 4E9H | | Simple CPU communication error code for each setting number | Simple CPU communication error code setting No.1 to Simple CPU communication error code setting No.64 |
| 1258 to 1321 | 4EAH to 529H | | Execution interval (current value) for each setting number | Execution interval (current value) setting No.1 to Execution interval (current value) setting No.64 |
| 1322 to 1385 | 52AH to 569H | | Abnormal response code for each setting number | Abnormal response code setting No.1 to Abnormal response code setting No.64 |

**Point**

- Do not write any data to the system area. Doing so may cause malfunction of the programmable controller system.
- If the value in an area of one word in size becomes equal to or higher than 65536, the count stops at 65535 (FFFFH).

## When using the RJ71EN71 (network type: Q-compatible Ethernet)

With the Q-compatible Ethernet, the same buffer memory addresses as the MELSEC-Q series Ethernet interface module can be used, excluding some addresses.

### ■Buffer memory addresses usable only with RJ71EN71

The following table lists the buffer memory addresses that can be used only with RJ71EN71.

| Address | | Application | Name |
|---|---|---|---|
| Decimal | Hexadecimal | | |
| 20700 to 20701 | 50DCH to 50DDH | Forced connection invalidation | Forced connection invalidation specification<br>20700 (b0 to 15): Connection No.1 to 16<br>20701 (b0): Auto-open UDP port<br>20701 (b1): MELSOFT transmission port (UDP/IP)<br>20701 (b2): MELSOFT transmission port (TCP/IP)<br>20701 (b3): FTP transmission port |
| 65535 | FFFFH | Network type information | Network type information |

### ■Buffer memory addresses differing from MELSEC-Q series Ethernet interface modules

The following table lists the buffer memory addresses having specifications that differ from the ones of the MELSEC-Q series Ethernet interface module.

| Address | | Name | |
|---|---|---|---|
| Decimal | Hexadecimal | MELSEC-Q series Ethernet interface module | RJ71EN71 |
| 0 to 1 | 0H to 1H | Own station E71 IP address<br>Initial value: C00001FEH | Own station IP address<br>Initial value: C0A80328H |
| 4 | 4H | Special function settings<br>Initial value: 100H | Special function settings<br>Initial value: 10H |
| 31 | 1FH | Communication condition setting (Ethernet Operation Setting) area<br>Initial value: 0H<br>Communication data code setting (b1)<br>0: Communications in a binary code<br>1: Communications in an ASCII code<br>TCP existence confirmation setting (b4)<br>0: Use the Ping<br>1: Use the KeepAlive<br>Send frame setting (b5)<br>0: Ethernet<br>1: IEEE 802.3<br>Setting of write enable/disable at RUN time (b6)<br>0: Disable<br>1: Enable<br>Initial timing setting (b8)<br>0: Do not wait for OPEN (Communications impossible at STOP time)<br>1: Always wait for OPEN (Communication possible at STOP time)<br>Reinitialization specification (b15)<br>0: Reinitialization process complete (reset by the system)<br>1: Reinitialization process request (set by the user) | Communication condition setting area<br>Initial value: 100H<br>Communication data code (b1)<br>0: Binary<br>1: ASCII<br>Enable/disable online change (b6)<br>0: Disable all (SLMP)<br>1: Enable all (SLMP)<br>Opening method (b8)<br>0: Open by program<br>1: Do not open by program<br>Reinitialization specification (b15)<br>0: Reinitialization process complete (reset by the system)<br>1: Reinitialization process request (set by the user) |

| Address | | Name | | | |
|---|---|---|---|---|---|
| Decimal | Hexadecimal | MELSEC-Q series Ethernet interface module | | RJ71EN71 | |
| 40 | 28H | Communication address setting area Connection No.1 | Own station port No. Initial value: 0H | Communication address setting area Connection No.1 | Own station port No. Initial value: 0H |
| 41 to 42 | 29H to 2AH | | Destination IP address Initial value: 0H | | Destination IP address Initial value: 0H |
| 43 | 2BH | | Destination Port No. Initial value: 0H | | Destination Port No. Initial value: 0H |
| 44 to 46 | 2CH to 2EH | | Destination MAC address Initial value: FFFFFFFFFFFFH | | System area |
| 47 to 95 | 2FH to 5FH | Communication address setting area Connection No.2 to 8 (Same as connection No.1) | | Communication address setting area Connection No.2 to 8 (Same as connection No.1) | |
| 106 to 107 | 6AH to 6BH | Own station IP address Initial value: 0H | | Own station IP address Initial value: C0A80328H | |
| 108 to 110 | 6CH to 6EH | Own station MAC address Initial value: 0H | | Own station MAC address Initial value: MAC address of the RJ71EN71 | |
| 116 | 74H | Auto-open UDP port number Initial value: 0H | | Auto-open UDP port number Initial value: 1388H | |
| 200 | C8H | LED status<br>INIT. LED (b0)<br>0: Off<br>1: On (initial processing completed)<br>OPEN LED (b1)<br>0: Off<br>1: On (connection open processing completed)<br>ERR. LED (b3)<br>0: Off<br>1: On (setting error)<br>COM.ERR. LED (b4)<br>0: Off<br>1: On (communication error) | | LED status<br>P ERR LED (b0)<br>0: Off<br>1: On or flashing | |
| 201 | C9H | Hub connection status area<br>Communication mode (b9)<br>0: Half-duplex<br>1: Full-duplex<br>Hub connection status (b10)<br>0: Hub not connected/disconnected<br>1: Hub connected<br>Data transmission speed (b15, b14)<br>00: Operating at 10BASE-T<br>01: Operating at 100BASE-TX | | Hub connection status area<br>Communication mode (b9)<br>0: Half-duplex<br>1: Full-duplex<br>Hub connection status (b10)<br>0: Hub not connected/disconnected<br>1: Hub connected<br>Data transmission speed (b15, b14)<br>00: Operating at 10BASE-T<br>01: Operating at 100BASE-TX<br>10: Operating at 1000BASE-T | |
| 202 | CAH | Switch status (Operational mode setting)<br>0: Online<br>1: Offline<br>2: Self-loopback test<br>3: Hardware test | | Module operation mode setting<br>0: Online mode<br>1: Offline mode<br>4: Module communication test mode | |

| Address | | Name | |
|---|---|---|---|
| Decimal | Hexadecimal | MELSEC-Q series Ethernet interface module | RJ71EN71 |
| 203 | CBH | Status of settings with a programming tool<br>Initial value: 0H<br>Communication data code setting (b1)<br>0: Communications in a binary code<br>1: Communications in an ASCII code<br>Initial/open method setting (b2)<br>0: No parameter setting (startup according to the sequence program)<br>1: Parameter setting (startup according to the parameters)<br>TCP existence confirmation setting (b4)<br>0: Use the Ping<br>1: Use the KeepAlive<br>Send frame setting (b5)<br>0: Ethernet<br>1: IEEE 802.3<br>Setting of write enable/disable at RUN time (b6)<br>0: Disable<br>1: Enable<br>Initial timing setting (b8)<br>0: Do not wait for OPEN (Communications impossible at STOP time)<br>1: Always wait for OPEN (Communication possible at STOP time) | Status of settings with a programming tool<br>Initial value: 100H<br>Communication data code setting (b1)<br>0: Communications in a binary code<br>1: Communications in an ASCII code<br>Send frame setting (b5)<br>0: Ethernet (fixed)<br>Setting of write enable/disable at RUN time (b6)<br>0: Disable<br>1: Enable<br>Opening method (b8)<br>0: Open by program<br>1: Do not open by program |
| 398 to 399 | 18EH to 18FH | Simultaneous transmission error detection count | Receive buffer full count |
| 20486 | 5006H | Remote password status<br>Initial value: 0H | Remote password lock/unlocked status (user connection)<br>Initial value: Depending on the remote password setting |
| 20487 | 5007H | Remote password status<br>Initial value: 0H | Remote password lock/unlocked status (system connection)<br>Initial value: Depending on the remote password setting |
| 20595 | 5073H | Connection No.1 Accumulated count of unlock process abnormal end | Connection No.1 Remote password mismatch count |
| 20600 | 5078H | Connection No.2 Accumulated count of unlock process abnormal end | Connection No.2 Remote password mismatch count |
| 20605 | 507DH | Connection No.3 Accumulated count of unlock process abnormal end | Connection No.3 Remote password mismatch count |
| 20610 | 5082H | Connection No.4 Accumulated count of unlock process abnormal end | Connection No.4 Remote password mismatch count |
| 20615 | 5087H | Connection No.5 Accumulated count of unlock process abnormal end | Connection No.5 Remote password mismatch count |
| 20620 | 508CH | Connection No.6 Accumulated count of unlock process abnormal end | Connection No.6 Remote password mismatch count |
| 20625 | 5091H | Connection No.7 Accumulated count of unlock process abnormal end | Connection No.7 Remote password mismatch count |
| 20630 | 5096H | Connection No.8 Accumulated count of unlock process abnormal end | Connection No.8 Remote password mismatch count |
| 20635 | 509BH | Connection No.9 Accumulated count of unlock process abnormal end | Connection No.9 Remote password mismatch count |
| 20640 | 50A0H | Connection No.10 Accumulated count of unlock process abnormal end | Connection No.10 Remote password mismatch count |
| 20645 | 50A5H | Connection No.11 Accumulated count of unlock process abnormal end | Connection No.11 Remote password mismatch count |
| 20650 | 50AAH | Connection No.12 Accumulated count of unlock process abnormal end | Connection No.12 Remote password mismatch count |
| 20655 | 50AFH | Connection No.13 Accumulated count of unlock process abnormal end | Connection No.13 Remote password mismatch count |
| 20660 | 50B4H | Connection No.14 Accumulated count of unlock process abnormal end | Connection No.14 Remote password mismatch count |
| 20665 | 50B9H | Connection No.15 Accumulated count of unlock process abnormal end | Connection No.15 Remote password mismatch count |

| Address | | Name | |
|---|---|---|---|
| Decimal | Hexadecimal | MELSEC-Q series Ethernet interface module | RJ71EN71 |
| 20670 | 50BEH | Connection No.16 Accumulated count of unlock process abnormal end | Connection No.16 Remote password mismatch count |
| 20675 | 50C3H | Auto-open UDP port<br>Accumulated count of unlock process abnormal end | Auto-open UDP port<br>Remote password mismatch count |
| 20680 | 50C8H | MELSOFT application transmission port (UDP)<br>Accumulated count of unlock process abnormal end | MELSOFT transmission port (UDP/IP)<br>Remote password mismatch count |
| 20685 | 50CDH | MELSOFT application transmission port (TCP)<br>Accumulated count of unlock process abnormal end | MELSOFT transmission port (TCP/IP)<br>Remote password mismatch count |
| 20690 | 50D2H | FTP transmission port<br>Accumulated count of unlock process abnormal end | FTP transmission port<br>Remote password mismatch count |
| 21056 | 5240H | Receive buffer full detection signal | State of receive buffer |

## ■Buffer memory addresses not supported by the RJ71EN71

The following table lists the buffer memory addresses of the MELSEC-Q series Ethernet interface modules that are not supported by the RJ71EN71.

| Address | | Application | Name | | |
|---|---|---|---|---|---|
| Decimal | Hexadecimal | | | | |
| 944 to 949 | 3B0H to 3B5H | FTP setting area | FTP login name | | |
| 950 to 953 | 3B6H to 3B9H | | Password | | |
| 20488 | 5008H | System port information area | System port use prohibited designation area | System port use prohibited designation | |
| 20592 | 5070H | Monitoring area | Remote password function monitoring area | Remote password mismatch notification accumulated count designation (user connection) | |
| 20593 | 5071H | | | Remote password mismatch notification accumulated count designation (system connection) | |
| 20594 | 5072H | | | Connection No.1 | Accumulated count of unlock process normal completion |
| 20596 | 5074H | | | | Accumulated count of lock process normal completion |
| 20597 | 5075H | | | | Accumulated count of lock process abnormal end |
| 20598 | 5076H | | | | Accumulated count of lock process based on close |
| 20599 | 5077H | | | Connection No.2 | Accumulated count of unlock process normal completion |
| 20601 | 5079H | | | | Accumulated count of lock process normal completion |
| 20602 | 507AH | | | | Accumulated count of lock process abnormal end |
| 20603 | 507BH | | | | Accumulated count of lock process based on close |
| 20604 | 507CH | | | Connection No.3 | Accumulated count of unlock process normal completion |
| 20606 | 507EH | | | | Accumulated count of lock process normal completion |
| 20607 | 507FH | | | | Accumulated count of lock process abnormal end |
| 20608 | 5080H | | | | Accumulated count of lock process based on close |
| 20609 | 5081H | | | Connection No.4 | Accumulated count of unlock process normal completion |
| 20611 | 5083H | | | | Accumulated count of lock process normal completion |
| 20612 | 5084H | | | | Accumulated count of lock process abnormal end |
| 20613 | 5085H | | | | Accumulated count of lock process based on close |
| 20614 | 5086H | | | Connection No.5 | Accumulated count of unlock process normal completion |
| 20616 | 5088H | | | | Accumulated count of lock process normal completion |
| 20617 | 5089H | | | | Accumulated count of lock process abnormal end |
| 20618 | 508AH | | | | Accumulated count of lock process based on close |
| 20619 | 508BH | | | Connection No.6 | Accumulated count of unlock process normal completion |
| 20621 | 508DH | | | | Accumulated count of lock process normal completion |
| 20622 | 508EH | | | | Accumulated count of lock process abnormal end |
| 20623 | 508FH | | | | Accumulated count of lock process based on close |
| 20624 | 5090H | | | Connection No.7 | Accumulated count of unlock process normal completion |
| 20626 | 5092H | | | | Accumulated count of lock process normal completion |
| 20627 | 5093H | | | | Accumulated count of lock process abnormal end |
| 20628 | 5094H | | | | Accumulated count of lock process based on close |
| 20629 | 5095H | | | Connection No.8 | Accumulated count of unlock process normal completion |
| 20631 | 5097H | | | | Accumulated count of lock process normal completion |
| 20632 | 5098H | | | | Accumulated count of lock process abnormal end |
| 20633 | 5099H | | | | Accumulated count of lock process based on close |
| 20634 | 509AH | | | Connection No.9 | Accumulated count of unlock process normal completion |
| 20636 | 509CH | | | | Accumulated count of lock process normal completion |
| 20637 | 509DH | | | | Accumulated count of lock process abnormal end |
| 20638 | 509EH | | | | Accumulated count of lock process based on close |
| 20639 | 509FH | | | Connection No.10 | Accumulated count of unlock process normal completion |
| 20641 | 50A1H | | | | Accumulated count of lock process normal completion |
| 20642 | 50A2H | | | | Accumulated count of lock process abnormal end |
| 20643 | 50A3H | | | | Accumulated count of lock process based on close |

| Address | | Application | Name | | |
|---|---|---|---|---|---|
| Decimal | Hexadecimal | | | | |
| 20644 | 50A4H | Monitoring area | Remote password function monitoring area | Connection No.11 | Accumulated count of unlock process normal completion |
| 20646 | 50A6H | | | | Accumulated count of lock process normal completion |
| 20647 | 50A7H | | | | Accumulated count of lock process abnormal end |
| 20648 | 50A8H | | | | Accumulated count of lock process based on close |
| 20649 | 50A9H | | | Connection No.12 | Accumulated count of unlock process normal completion |
| 20651 | 50ABH | | | | Accumulated count of lock process normal completion |
| 20652 | 50ACH | | | | Accumulated count of lock process abnormal end |
| 20653 | 50ADH | | | | Accumulated count of lock process based on close |
| 20654 | 50AEH | | | Connection No.13 | Accumulated count of unlock process normal completion |
| 20656 | 50B0H | | | | Accumulated count of lock process normal completion |
| 20657 | 50B1H | | | | Accumulated count of lock process abnormal end |
| 20658 | 50B2H | | | | Accumulated count of lock process based on close |
| 20659 | 50B3H | | | Connection No.14 | Accumulated count of unlock process normal completion |
| 20661 | 50B5H | | | | Accumulated count of lock process normal completion |
| 20662 | 50B6H | | | | Accumulated count of lock process abnormal end |
| 20663 | 50B7H | | | | Accumulated count of lock process based on close |
| 20664 | 50B8H | | | Connection No.15 | Accumulated count of unlock process normal completion |
| 20666 | 50BAH | | | | Accumulated count of lock process normal completion |
| 20667 | 50BBH | | | | Accumulated count of lock process abnormal end |
| 20668 | 50BCH | | | | Accumulated count of lock process based on close |
| 20669 | 50BDH | | | Connection No.16 | Accumulated count of unlock process normal completion |
| 20671 | 50BFH | | | | Accumulated count of lock process normal completion |
| 20672 | 50C0H | | | | Accumulated count of lock process abnormal end |
| 20673 | 50C1H | | | | Accumulated count of lock process based on close |
| 20674 | 50C2H | | | Auto-open UDP port | Accumulated count of unlock process normal completion |
| 20676 | 50C4H | | | | Accumulated count of lock process normal completion |
| 20677 | 50C5H | | | | Accumulated count of lock process abnormal end |
| 20678 | 50C6H | | | | Accumulated count of lock process based on close |
| 20679 | 50C7H | | | MELSOFT transmission port (UDP/IP) | Accumulated count of unlock process normal completion |
| 20681 | 50C9H | | | | Accumulated count of lock process normal completion |
| 20682 | 50CAH | | | | Accumulated count of lock process abnormal end |
| 20683 | 50CBH | | | | Accumulated count of lock process based on close |
| 20684 | 50CCH | | | MELSOFT transmission port (TCP/IP) | Accumulated count of unlock process normal completion |
| 20686 | 50CEH | | | | Accumulated count of lock process normal completion |
| 20687 | 50CFH | | | | Accumulated count of lock process abnormal end |
| 20688 | 50D0H | | | | Accumulated count of lock process based on close |
| 20689 | 50D1H | | | FTP transmission port | Accumulated count of unlock process normal completion |
| 20691 | 50D3H | | | | Accumulated count of lock process normal completion |
| 20692 | 50D4H | | | | Accumulated count of lock process abnormal end |
| 20693 | 50D5H | | | | Accumulated count of lock process based on close |
| 20737 to 20855 | 5101H to 5177H | HTTP status storage area | | | |
| 20992 | 5200H | "Issue system switching request at disconnection detection" status storage area | "Issue system switching request at disconnection detection" | | |
| 20993 | 5201H | | Disconnection detection monitoring time | | |
| 21008 to 21009 | 5210H to 5211H | "System switching settings when communication error occurs" status storage area | | | |
| 22640 to 24575 | 5870H to 5FFFH | E-mail status storage area | | | |

For details on other buffer memory addresses that can be used in the RJ71EN71, refer to the following.

Q Corresponding Ethernet Interface Module User's Manual (Basic)

# Details of buffer memory addresses (RJ71EN71, RnENCPU (network part))

This section describes the details of buffer memory addresses of the RJ71EN71 and RnENCPU (network part).
The buffer memory addresses are for when the P1 connector is used.

## Own node setting status storage area

### ■Own node IP address (Un\G4 to Un\G5)

The IP address set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G4 | Third octet, fourth octet |
| Un\G5 | First octet, second octet |

### ■Subnet mask (Un\G14 to Un\G15)

The subnet mask set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G14 | Third octet, fourth octet |
| Un\G15 | First octet, second octet |

### ■Default gateway IP address (Un\G18 to Un\G19)

The default gateway IP address set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G18 | Third octet, fourth octet |
| Un\G19 | First octet, second octet |

### ■Own node MAC address (Un\G28 to Un\G30)

The MAC address of the Ethernet-equipped module is stored.

| Address | Description |
|---|---|
| Un\G28 | 5th byte, 6th byte of the MAC address |
| Un\G29 | 3rd byte, 4th byte of the MAC address |
| Un\G30 | 1st byte, 2nd byte of the MAC address |

### ■Jumbo frame setting (Un\G35)

| Address | Description |
|---|---|
| Un\G35 | Stores the jumbo frame set with the module parameter.<br>0: Disable (MTU 1500 byte)<br>2: 2KB (MTU 2034 byte)<br>3: 3KB (MTU 3058 byte)<br>4: 4KB (MTU 4082 byte)<br>5: 5KB (MTU 5106 byte)<br>6: 6KB (MTU 6130 byte)<br>7: 7KB (MTU 7154 byte)<br>8: 8KB (MTU 8178 byte)<br>9: 9KB (MTU 9004 byte) |

### ■Communication speed setting (Un\G36)

| Address | Description |
|---|---|
| Un\G36 | Stores the communication speed set with the module parameter.<br>0: Automatic negotiation<br>1: 10Mbps/half-duplex<br>2: 10Mbps/full-duplex<br>3: 100Mbps/half-duplex<br>4: 100Mbps/full-duplex<br>6: 1Gbps/full-duplex |

## Connection status storage area

### ■Connection No.1 latest error code to connection No.64 latest error code (Un\G100 to Un\G163)

The latest error code for each connection is stored.

| Address | Name |
|---|---|
| Un\G100 | Connection No.1 latest error code |
| Un\G101 | Connection No.2 latest error code |
| ⋮ | |
| Un\G162 | Connection No.63 latest error code |
| Un\G163 | Connection No.64 latest error code |

## Error log area

### ■Number of errors (Un\G227)

| Address | Description |
|---|---|
| Un\G227 | The registered error count is stored in the error log block area.<br>If 65536 or more errors occur, the count stops at FFFFH (65535).<br>Even if the count in the storage area for the number of errors stops, the process of storing error information in the following areas continues.<br>• Error log write pointer (Un\G228)<br>• Error log block area (Un\G229 to Un\G372) |

### ■Error log write pointer (Un\G228)

| Address | Description |
|---|---|
| Un\G228 | This area is used for storing the error log block number at which the latest error log was registered.<br>0: No error (no error log registered)<br>1 or more: Error log block number at which the latest error log was registered<br>When the pointer value is 16, this indicates that the latest error log is registered in the area of error log block 16. When 17 or more errors occur, the registration of error logs starts again from the area of error log block 1. |

### ■Error log block 1 (Un\G229 to Un\G237)

| Address | Name | Description |
|---|---|---|
| Un\G229 | Error code/end code | The error code that shows the error details and the end code that is appended to a response for data communications are stored.<br>☞ Page 453 List of Error Codes<br>☞ Page 479 End Codes Returned to an External Device During Data Communications |
| Un\G230 | Subheader | The subheader code of the message that became an error is stored in b0 to b7. (0 is stored in b8 to b15.)<br>0: Not used, subheader code unknown |
| Un\G231 | Command code | The command code of a message that became an error or the values for the lower bytes of the request type and subrequest type of a data link instruction are stored.<br>■Command code<br>• b15 to b0: Command code<br>■Request type and subrequest type<br>• b15 to b8: Subrequest type<br>• b7 to b0: Request type<br>0: Not used, command code unknown |
| Un\G232 | Connection No. | The connection number in which an error occurred is stored in b0 to b7. (0 is stored in b8 to b15.)<br>0: Not used, connection number unknown<br>1 to 128: User connections<br>129 or more: System connections |
| Un\G233 | Own station port No. | The own station port number when the error occurred is stored.<br>0: Not used, own station port number unknown |
| Un\G234, Un\G235 | Communication destination IP address | The IP address of the communication destination when the error occurred is stored.<br>0: Not used, communication destination IP address unknown |
| Un\G236 | Communication destination port No. | The port number of the communication destination when the error occurred is stored.<br>0: Not used, communication destination port number unknown |
| Un\G237 | System area | |

**Point**

The values stored in the error log areas are cleared when the Ethernet module mounted station is powered on, reset is operated, or the ERRCLEAR instruction is executed.
(However, when the network type is "Q Compatible Ethernet", the values are not cleared by execution of the UINI instruction and re-initial processing by 'Communication condition setting area' (Un\G31) of the buffer memory.)
For details on the ERRCLEAR instruction, refer to the following.
📖 MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

### ■Error log block 2 to 16 (Un\G238 to Un\G372)

These areas are used for storing the information of error log block 2 to 16 in the same order as Error log block 1.

## Status for each protocol

### ■IP packet (Un\G5000 to Un\G5029)

The IP status is counted in the range of 0 to 4294967295 (FFFFFFFH). (The number of receive aborts is counted in the range of 0 to 65535 (FFFFH).)

| Address | Name |
|---|---|
| Un\G5000 to Un\G5001 | Received packet total count |
| Un\G5002 to Un\G5003 | Received packet checksum error discard count |
| Un\G5004 to Un\G5005 | Sent packet total count |
| Un\G5022 to Un\G5023 | Simultaneous transmission error detection count (receive buffer full count) |
| Un\G5029 | Receive abort count |

### ■ICMP packet (Un\G5040 to Un\G5053)

The ICMP status is counted in the range of 0 to 4294967295 (FFFFFFFH).

| Address | Name |
|---|---|
| Un\G5040 to Un\G5041 | Received packet total count |
| Un\G5042 to Un\G5043 | Received packet checksum error discard count |
| Un\G5044 to Un\G5045 | Sent packet total count |
| Un\G5046 to Un\G5047 | Received echo request total count |
| Un\G5048 to Un\G5049 | Sent echo reply total count |
| Un\G5050 to Un\G5051 | Sent echo request total count |
| Un\G5052 to Un\G5053 | Received echo reply total count |

### ■TCP packet (Un\G5080 to Un\G5085)

The TCP status is counted in the range of 0 to 4294967295 (FFFFFFFH).

| Address | Name |
|---|---|
| Un\G5080 to Un\G5081 | Received packet total count |
| Un\G5082 to Un\G5083 | Received packet checksum error discard count |
| Un\G5084 to Un\G5085 | Sent packet total count |

### ■UDP packet (Un\G5120 to Un\G5125)

The UDP status is counted in the range of 0 to 4294967295 (FFFFFFFH).

| Address | Name |
|---|---|
| Un\G5120 to Un\G5121 | Received packet total count |
| Un\G5122 to Un\G5123 | Received packet checksum error discard count |
| Un\G5124 to Un\G5125 | Sent packet total count |

### ■Receiving error (Un\G5160 to Un\G5165)

The receive error status is counted in the range of 0 to 4294967295 (FFFFFFFH).

| Address | Description |
|---|---|
| Un\G5160 to Un\G5161 | Framing error count |
| Un\G5162 to Un\G5163 | Receive FIFO overflow count |
| Un\G5164 to Un\G5165 | CRC error count |

## Own node operation status storage area

### ■LED status (Un\G5189)

| Address | Description |
|---|---|
| Un\G5189 | Stores the on/off status of the P ERR LED.<br>0: Off<br>1: On or flashing |

### ■Hub connection status area (Un\G5191 to Un\G5194)

The hub connection status of the Ethernet-equipped module is stored.

| Address | Name | Description |
|---|---|---|
| Un\G5191 | Communication mode | Stores the communication mode.<br>0: Half-duplex<br>1: Full-duplex |
| Un\G5192 | Connection status | Stores the connection status.<br>0: Hub not connected or disconnected<br>1: Hub connected |
| Un\G5193 | Communication speed | Stores the communication speed.<br>0: Operating at 10BASE-T<br>1: Operating at 100BASE-TX<br>2: Operating at 1000BASE-T |
| Un\G5194 | Disconnection count | Stores the number of times the cable was disconnected. |

### ■IP address duplication status storage area (Un\G5200 to Un\G5206)

Information when IP address is duplicated is stored.

| Address | Name | Description |
|---|---|---|
| Un\G5200 | Same IP address detection flag | Stores IP address duplication status.<br>0: IP address not duplicated<br>1: IP address duplicated |
| Un\G5201 to Un\G5203 | MAC address of the station already connected to the network | Stores the MAC address of the station that has been already connected to the network in the station with duplicated IP address.<br>Un\G5201: 5th and 6th byte of the MAC address<br>Un\G5202: 3rd and 4th byte of the MAC address<br>Un\G5203: 1st and 2nd byte of the MAC address<br>"FFFFFFFFFFFFH" is stored in the station that has been already connected to the network. |
| Un\G5204 to Un\G5206 | MAC address of the station with the IP address already used | Stores the MAC address of the station with duplicated IP address in the station that has been already connected to the network.<br>Un\G5204: 5th and 6th byte of the MAC address<br>Un\G5205: 3rd and 4th byte of the MAC address<br>Un\G5206: 1st and 2nd byte of the MAC address<br>"FFFFFFFFFFFFH" is stored in the station with the duplicated IP address. |

## Area for sending/receiving instructions

### ■RECV instruction execution request (Un\G5301)

| Address | Description |
|---|---|
| Un\G5301 | Stores the RECV instruction execution request status of each channel in b0 to b7 (channel 1 to 8).<br>On: Requesting<br>Off: No request |

### ■Link dedicated instruction (Un\G5323 to Un\G5325)

The execution result of the link dedicated instruction is stored.

| Address | Name | Description |
|---|---|---|
| Un\G5323 | ZNRD instruction execution result | Stores the execution result of the ZNRD instruction.<br>0: Completed successfully<br>Other than 0: Completed with an error (An error code is stored.) |
| Un\G5325 | ZNWR instruction execution result | Stores the execution result of the ZNWR instruction.<br>0: Completed successfully<br>Other than 0: Completed with an error (An error code is stored.) |

## Remote password lock status storage area

The remote password lock status of each connection is stored.

- 0: Unlocked or remote password not set
- 1: Lock status

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G5626 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G5629 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

The numbers in the table indicate connection numbers.

### ■Remote password lock status system port (Un\G5630)

| Address | Description |
|---|---|
| Un\G5630 | Stores the remote password lock status of the system port in b0 to b4.<br>0: Unlocked or remote password not set<br>1: Lock status<br>The bits corresponding to each system port are shown below.<br>b0: Auto-open UDP port<br>b1: MELSOFT transmission port (UDP/IP)<br>b2: MELSOFT transmission port (TCP/IP)<br>b3: FTP transmission port<br>b4: MELSOFT direct connection |

## Forced connection invalidation setting area

Set the connection to be forcibly invalidated.

- 0: Use allowed
- 1: Use prohibited

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G5646 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G5649 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

The numbers in the table indicate connection numbers.

### ■Forced connection invalidation system port (Un\G5650)

| Address | Description |
|---|---|
| Un\G5650 | Set the system port to be forcibly invalidated.<br>0: Use allowed<br>1: Use prohibited<br>The bits corresponding to each system port are shown below.<br>b0: Auto-open UDP port<br>b1: MELSOFT transmission port (UDP/IP)<br>b2: MELSOFT transmission port (TCP/IP)<br>b3: FTP transmission port<br>b4: MELSOFT direct connection |

## Predefined protocol support function execution status check area

### ■Connection No.1 (Un\G8300 to Un\G8335)

The execution status of the predefined protocol support function is stored.

For addresses corresponding to connections after the connection No.2, refer to the following.

☞ Page 486 List of buffer memory addresses

| Address | Name | Description |
|---|---|---|
| Un\G8300 | Protocol execution status | Stores the status of the protocol being executed at connection No.1.<br>0: Unexecuted<br>1: Waiting for transmission<br>2: Sending<br>3: Waiting for data reception<br>4: Receiving<br>5: Execution completed |
| Un\G8301 | System area | — |
| Un\G8302 | Received data verification result (receive packet No.1) | Stores the verification results of receive packet No.1.<br>• Element No. where the verification result did not match (b0 to b7)<br>0: Verification matched<br>1 to 32: Element No. where the verification result did not match<br>FFH: Verification not performed<br>• The cause of mismatch (verification result code) (b8 to b15)*1 |
| Un\G8303 to Un\G8317 | Received data verification result (receive packet No.2 to 16) | The configuration is the same as receive packet No.1. |
| Un\G8318 | Protocol execution count | Stores the number of protocol executions in Connection No.1.<br>0: Protocol not executed<br>1 to 65535: Number of executions (When the number exceeds 65535, the value remains 65535.) |
| Un\G8319 | Protocol cancellation specification | Cancels the protocol executed in Connection No.1.<br>0: No cancellation instruction<br>1: Cancellation request (set by user)<br>2: Cancellation completed (set by system) |
| Un\G8320 to Un\G8335 | System area | — |

*1 The following table lists the verification result codes and causes of verification mismatch.

| Stored value | Description | Cause |
|---|---|---|
| 00H | Normal | — |
| 01H | Insufficient receive data | The total packet size of receive data is smaller than that set in protocol data. |
| 10H | Data not matched | The receive data do not match the value set in protocol data. |
| 11H | ASCII-binary conversion error | When the code type setting is "ASCII Hexadecimal", data not in ASCII code are received. |
| 12H | Data length error | The received length value exceeded 2046 bytes. |
| 30H | Data length size error | The Length value received from the external device does not match the actual length. |
| FFH | Verification not performed | — |

## Time setting function (SNTP) area

### ■Time setting function operation result (Un\G11000)

| Address | Description |
|---|---|
| Un\G11000 | Stores the operation result of the time setting function.<br>0: Unexecuted<br>1: Success<br>FFFFH: Failure |

### ■Time setting function execution time (Un\G11001 to Un\G11007)

| Address | Name | Description |
|---|---|---|
| Un\G11001 | Year | Stores the year that the time setting function was executed. |
| Un\G11002 | Month | Stores the month that the time setting function was executed. |
| Un\G11003 | Day | Stores the date that the time setting function was executed. |
| Un\G11004 | Hour | Stores the time (hour) that the time setting function was executed. |
| Un\G11005 | Minute | Stores the time (minute) that the time setting function was executed. |
| Un\G11006 | Second | Stores the time (second) that the time setting function was executed. |
| Un\G11007 | Day of the week | Stores the day of the week that the time setting function was executed.<br>0: Sunday<br>1: Monday<br>2: Tuesday<br>3: Wednesday<br>4: Thursday<br>5: Friday<br>6: Saturday |

### ■Time setting function required response time (Un\G11008)

| Address | Description |
|---|---|
| Un\G11008 | Stores the time required for the module to set the time after sending to the SNTP server.<br>Range: 0 to FFFEH (Unit: ms) |

### ■Time setting function (SNTP client) execution (Un\G11009)

| Address | Description |
|---|---|
| Un\G11009 | Executes the time setting function when b0 is turned on.<br>The time setting function can be executed at a random time from a program.<br>The function is not executed if b0 is turned on during execution of the time setting function.<br>This is valid only when "Use" is set for "Time Setting (SNTP Client)" under "Time Setting" in "Application Settings". |

## IP packet transfer function area

### ■IP packet transfer latest data volume (Un\G11050 to Un\G11051)

| Address | Description |
|---|---|
| Un\G11050 to Un\G11051 | Stores the latest value (byte/s) of the total size (byte) per unit time (one second) of the transferred IP packet.<br>(The low-order 16-bit value is stored in the 1st word, and the high-order 16-bit value is stored in the 2nd word.)<br>Range: 0 to 4294967295 (FFFFFFFFH)<br>The value is 4294967295 (FFFFFFFFH) if the above value is exceeded. |

### ■IP packet transfer maximum data volume (Un\G11052 to Un\G11053)

| Address | Description |
|---|---|
| Un\G11052 to Un\G11053 | Stores the maximum value (byte/s) of the total size (byte) per unit time (one second) of the transferred IP packet.<br>(The low-order 16-bit value is stored in the 1st word, and the high-order 16-bit value is stored in the 2nd word.)<br>Range: 0 to 4294967295 (FFFFFFFFH)<br>The value is 4294967295 (FFFFFFFFH) if the above value is exceeded. |

## Dynamic routing function area

### ■Communication path determination status (Un\G11500 to Un\G11514)

The determination status of the communication path for each network number of the destination station is stored.

- 0: Path undetermined
- 1: Path determined

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G11500 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G11514 | Empty | 239 | 238 | 237 | 236 | 235 | 234 | 233 | 232 | 231 | 230 | 229 | 228 | 227 | 226 | 225 |

The numbers in the table indicate network numbers.

## Remote password function monitoring area

The number of times each connection and system port remote password unlocking failed continuously is counted in the range of 0 to 65535.

| Address | Name |
|---|---|
| Un\G11600 | Connection No.1 continuous unlock failure count |
| Un\G11601 | Connection No.2 continuous unlock failure count |
| ⋮ | |
| Un\G11662 | Connection No.63 continuous unlock failure count |
| Un\G11663 | Connection No.64 continuous unlock failure count |
| Un\G11664 | Auto-open UDP port continuous unlock failure count |
| Un\G11665 | MELSOFT transmission port (UDP/IP) continuous unlock failure count |
| Un\G11666 | MELSOFT transmission port (TCP/IP) continuous unlock failure count |
| Un\G11667 | FTP transmission port (TCP/IP) continuous unlock failure count |
| Un\G11668 | MELSOFT direct connection continuous unlock failure count |

## Area for both systems identical IP address setting function

### ■Control system IP address (Un\G11670 to Un\G11671)

The control system IP address set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G11670 | Third octet, fourth octet |
| Un\G11671 | First octet, second octet |

## IP address change function area

### ■IP address setting (Un\G11700 to Un\G11705)

Set the IP address, subnet mask, and default gateway to be stored in the IP address storage area (flash ROM).

| Address | Name | Description |
|---|---|---|
| Un\G11700 | IP address | Third octet, fourth octet |
| Un\G11701 | | First octet, second octet |
| Un\G11702 | Subnet mask | Third octet, fourth octet |
| Un\G11703 | | First octet, second octet |
| Un\G11704 | Default Gateway | Third octet, fourth octet |
| Un\G11705 | | First octet, second octet |

### ■Communications by network number/station number (Un\G11706 to Un\G11708)

Set the network number, station number, and transient transmission group number to be stored in the IP address storage area (flash ROM).

| Address | Name |
|---|---|
| Un\G11706 | Network number |
| Un\G11707 | Station number |
| Un\G11708 | Transient transmission group number |

### ■IP address storage area write request (Un\G11709)

| Address | Description |
|---|---|
| Un\G11709 | • When b0 is turned on, writes the values stored in 'IP address setting' (Un\G11700 to Un\G11705) and 'Communications by Network No./Station No.' (Un\G11706 to Un\G11708) to the IP address storage area (flash ROM).<br>• When the data is completely written to the IP address storage area (flash ROM), b0 turns off regardless of the write results. |

### ■IP address storage area write error (Un\G11710)

| Address | Description |
|---|---|
| Un\G11710 | When writing to the IP address storage area (flash ROM) ends abnormally, b0 turns on. When writing completes successfully, b0 turns off. |

### ■IP address storage area writing error cause (Un\G11711)

| Address | Description |
|---|---|
| Un\G11711 | Stores the cause of the abnormal end of writing to the IP address storage area (flash ROM).<br>0H: No error<br>100H:The value for 'IP address' (Un\G11700 to Un\G11701) exceeds the setting range.<br>101H:The value for 'Default gateway' (Un\G11704 to Un\G11705) exceeds the setting range.<br>102H:The network address of 'Default gateway' (Un\G11704 to Un\G11705) or of the gateway IP address is different from that of the IP address of the own node.<br>103H:The value for 'Network number' (Un\G11706) exceeds the setting range.<br>104H:The value for 'Station number' (Un\G11707) exceeds the setting range.<br>105H:The value for 'Transient transmission group number' (Un\G11708) exceeds the setting range.<br>106H:The port 1 and port 2 IP addresses written in the IP address storage area are the same.<br>107H:The port 1 and port 2 network numbers or station numbers written in the IP address storage area are the same.<br>200H:An error occurred during write.<br>400H:Writing was started during the clear processing. |

### ■IP address storage area clear request (Un\G11712)

| Address | Description |
|---|---|
| Un\G11712 | • When b0 is turned on, clears the IP address storage area (flash ROM).<br>• When the IP address storage area (flash ROM) is cleared completely, b0 turns off regardless of the clear results. |

### ■IP address storage area clear error (Un\G11713)

| Address | Description |
|---|---|
| Un\G11713 | When clearing the IP address storage area (flash ROM) ends abnormally, b0 turns on. When clearing completes successfully, b0 turns off. |

### ■IP address storage area clear error cause (Un\G11714)

| Address | Description |
|---|---|
| Un\G11714 | Stores the cause of the abnormal end of clearing the IP address storage area (flash ROM).<br>0H: No error<br>200H:Error occurred during clearing.<br>400H:Clearing was started during the write processing. |

### ■IP address change function operating status (Un\G11715)

| Address | Description |
|---|---|
| Un\G11715 | Stores the IP address change function operating status.<br>0: Operating according to the module parameter settings<br>1: Operating according to the IP address storage area (flash ROM) settings |

## Random access buffer area

### ■Random access buffer (Un\G20000 to Un\G26143)

This area is used for reading or writing the data when exchanging data with the random access buffer.

## Network type information area

### ■Network type information (Un\G65535)

| Address | Description |
|---|---|
| Un\G65535 | Stores the network type.<br>0: Ethernet<br>1: CC-Link IE Field Network<br>2: CC-Link IE Controller Network<br>3: Ethernet (Q series-compatible) |

## Area for communication using a fixed buffer

### ■Fixed buffer No.1 (Un\G65536 to Un\G70655)

This area is used for sending and receiving data during fixed buffer communication.

For addresses corresponding to connections after the connection No.2, refer to the following.

☞ Page 486 List of buffer memory addresses

| Address | Name | Description |
|---|---|---|
| Un\G65536 | Data length | Stores the data length of the fixed buffer data. |
| Un\G65537 to Un\G70655 | Fixed buffer data | Stores the fixed buffer data. |

## Error log area for MODBUS/TCP function

### ■Number of errors (Un\G720896)

This area is used for storing the number of errors that occurred during communications using MODBUS/TCP.

### ■Error log write pointer (Un\G720897)

This area is used for storing the error log number at which the latest error log was registered.

### ■Error log 1 (Un\G720898 to Un\G720905)

| Address | Name | Description |
|---|---|---|
| Un\G720898 | Detail error code | Stores an error code. (☞ Page 453 List of Error Codes) |
| Un\G720899 | Abnormal response code | Stores a MODBUS abnormal response code. (📖 MELSEC iQ-R MODBUS and MODBUS/TCP Reference Manual) |
| Un\G720900 | Function code | Stores a function code that caused the error. |
| Un\G720901 | Own station port number | Stores the own station port number when the error occurred. |
| Un\G720902 to Un\G720903 | Destination IP address | Stores the IP address of the destination station when the error occurred. |
| Un\G720904 | Destination port number | Stores the port number of the destination station when the error occurred. |
| Un\G720905 | System area | — |

### ■Error log 2 to 32 (Un\G720906 to Un\G721153)

These areas are used for storing the information of the 2nd to 32nd errors in the same order as Error log 1.

## Simple CPU communication area

### ■Request to start communication at request for each setting number (Un\G721896 to Un\G721899, Un\G1247300 to Un\G1247327)

This request contact starts the data send when the communication setting is "On Request" and is stored using the following bit pattern for each setting number.

- Off → On: Requested (start request)
- On → Off: Completed (start completion)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G721896 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G721899 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |
| **Address** | **b15** | **b14** | **b13** | **b12** | **b11** | **b10** | **b9** | **b8** | **b7** | **b6** | **b5** | **b4** | **b3** | **b2** | **b1** | **b0** |
| Un\G1247300 | 80 | 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1247327 | 512 | 511 | 510 | 509 | 508 | 507 | 506 | 505 | 504 | 503 | 502 | 501 | 500 | 509 | 508 | 507 |

### ■Request to stop fixed interval communication for each setting number (Un\G721900 to Un\G721903, Un\G1247328 to Un\G1247355)

This request contact stops the data send when the communication setting is "Fixed Interval" and is stored using the following bit pattern for each setting number.

- Off → On: Requested (stop request)
- On → Off: Completed (stop completion)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G721900 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G721903 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |
| **Address** | **b15** | **b14** | **b13** | **b12** | **b11** | **b10** | **b9** | **b8** | **b7** | **b6** | **b5** | **b4** | **b3** | **b2** | **b1** | **b0** |
| Un\G1247328 | 80 | 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1247355 | 512 | 511 | 510 | 509 | 508 | 507 | 506 | 505 | 504 | 503 | 502 | 501 | 500 | 509 | 508 | 507 |

### ■Request to restart fixed interval communication for each setting number (Un\G721904 to Un\G721907, Un\G1247356 to Un\G1247383)

This request contact restarts the data send when the communication setting is "Fixed Interval" and is stored using the following bit pattern for each setting number.

- Off → On: Requested (restart request)
- On → Off: Completed (restart completion)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G721904 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G721907 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |
| **Address** | **b15** | **b14** | **b13** | **b12** | **b11** | **b10** | **b9** | **b8** | **b7** | **b6** | **b5** | **b4** | **b3** | **b2** | **b1** | **b0** |
| Un\G1247356 | 80 | 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1247383 | 512 | 511 | 510 | 509 | 508 | 507 | 506 | 505 | 504 | 503 | 502 | 501 | 500 | 509 | 508 | 507 |

### ■Execution status flag for each setting number (Un\G721908 to Un\G721911, Un\G1247384 to Un\G1247411)

The status of data communication is stored using the following bit pattern for each setting number.

- Off: Communication stop (function not used)
- On: Communicating

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G721908 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G721911 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1247384 | 80 | 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1247411 | 512 | 511 | 510 | 509 | 508 | 507 | 506 | 505 | 504 | 503 | 502 | 501 | 500 | 509 | 508 | 507 |

### ■Preparation completion for each setting number (Un\G721912 to Un\G721915, Un\G1247412 to Un\G1247439)

The preparation completion status of the simple CPU communication is stored using the following bit pattern for each setting number.

- Off: Not ready (function not used)
- On: Ready

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G721912 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G721915 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1247412 | 80 | 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1247439 | 512 | 511 | 510 | 509 | 508 | 507 | 506 | 505 | 504 | 503 | 502 | 501 | 500 | 509 | 508 | 507 |

### ■Simple CPU communication status for each setting number (Un\G721936 to Un\G721999, Un\G1247460 to Un\G1247907)

| Address | Name | Description |
|---|---|---|
| Un\G721936 | Simple CPU communication status setting No.1 | The simple CPU communication status is stored.<br>0H: Unset (function not used)<br>1H: Preparing<br>2H: Waiting for the request<br>3H: Communicating<br>4H: Communication stop<br>5H: Retry being executed<br>6H: Monitoring<br>AH: Communications impossible |
| Un\G721937 | Simple CPU communication status setting No.2 | |
| ⋮ | | |
| Un\G721998 | Simple CPU communication status setting No.63 | |
| Un\G721999 | Simple CPU communication status setting No.64 | |

| Address | Name | Description |
|---|---|---|
| Un\G1247460 | Simple CPU communication status setting No.65 | The simple CPU communication status is stored.<br>0H: Unset (function not used)<br>1H: Preparing<br>2H: Waiting for the request<br>3H: Communicating<br>4H: Communication stop<br>5H: Retry being executed<br>6H: Monitoring<br>AH: Communications impossible |
| Un\G1247461 | Simple CPU communication status setting No.66 | |
| ⋮ | | |
| Un\G1247906 | Simple CPU communication status setting No.511 | |
| Un\G1247907 | Simple CPU communication status setting No.512 | |

## ■Simple CPU communication error code for each setting number (Un\G722000 to Un\G722063, Un\G1247908 to Un\G1248355)

| Address | Name | Description |
|---|---|---|
| Un\G722000 | Simple CPU communication error code setting No.1 | The cause of the error detected in the simple CPU communication is stored.<br>0: No error (function not used)<br>Other than 0: Error code (☞ Page 453 List of Error Codes)<br>The value is cleared to 0 with a clear request from the engineering tool. |
| Un\G722001 | Simple CPU communication error code setting No.2 | |
| ⋮ | | |
| Un\G722062 | Simple CPU communication error code setting No.63 | |
| Un\G722063 | Simple CPU communication error code setting No.64 | |

| Address | Name | Description |
|---|---|---|
| Un\G1247908 | Simple CPU communication error code setting No.65 | The cause of the error detected in the simple CPU communication is stored.<br>0: No error (function not used)<br>Other than 0: Error code (☞ Page 453 List of Error Codes)<br>The value is cleared to 0 with a clear request from the engineering tool. |
| Un\G1247909 | Simple CPU communication error code setting No.66 | |
| ⋮ | | |
| Un\G1248354 | Simple CPU communication error code setting No.511 | |
| Un\G1248355 | Simple CPU communication error code setting No.512 | |

## ■Abnormal response code for each setting number (Un\G722064 to Un\G722127, Un\G1248356 to Un\G1248803)

| Address | Name | Description |
|---|---|---|
| Un\G722064 | Abnormal response code setting No.1 | The abnormal response code detected in the simple CPU communication is stored.<br>0: No error (function not used)<br>Other than 0: Abnormal response code<br>For abnormal response codes, refer to the manual for the external device.<br>This function is available when CFB4H is stored in "Simple CPU communication error code for each setting number" (Un\G722000 to Un\G722063). |
| Un\G722065 | Abnormal response code setting No.2 | |
| ⋮ | | |
| Un\G722126 | Abnormal response code setting No.63 | |
| Un\G722127 | Abnormal response code setting No.64 | |

| Address | Name | Description |
|---|---|---|
| Un\G1248356 | Abnormal response code setting No.65 | The abnormal response code detected in the simple CPU communication is stored.<br>0: No error (function not used)<br>Other than 0: Abnormal response code<br>For abnormal response codes, refer to the manual for the external device.<br>This function is available when CFB4H is stored in "Simple CPU communication error code for each setting number" (Un\G1247908 to Un\G1248355). |
| Un\G1248357 | Abnormal response code setting No.66 | |
| ⋮ | | |
| Un\G1248802 | Abnormal response code setting No.511 | |
| Un\G1248803 | Abnormal response code setting No.512 | |

## ■Execution interval (current value) for each setting number (Un\G722128 to Un\G722191, Un\G1248804 to Un\G1249251)

| Address | Name | Description |
|---|---|---|
| Un\G722128 | Execution interval (current value) setting No.1 | If "Fixed Interval" is set for communication setting, the current value of the execution interval is stored.<br>0: Unset (function not used), communications impossible<br>Other than 0: Execution interval (unit: ms) |
| Un\G722129 | Execution interval (current value) setting No.2 | |
| ⋮ | | |
| Un\G722190 | Execution interval (current value) setting No.63 | |
| Un\G722191 | Execution interval (current value) setting No.64 | |

| Address | Name | Description |
|---|---|---|
| Un\G1248804 | Execution interval (current value) setting No.65 | If "Fixed Interval" is set for communication setting, the current value of the execution interval is stored.<br>0: Unset (function not used), communications impossible<br>Other than 0: Execution interval (unit: ms) |
| Un\G1248805 | Execution interval (current value) setting No.66 | |
| ⋮ | | |
| Un\G1249250 | Execution interval (current value) setting No.511 | |
| Un\G1249251 | Execution interval (current value) setting No.512 | |

## ■User setting area (Un\G722500 to Un\G1246787)

This area can be used freely during simple CPU communication and simple device communication.

## Simple device communication area

### ■Function start/stop instruction (Un\G1249300.0)

This instruction is the contact to start the simple device communication function. The initial value can be changed with "Instruction Initial Value to Start/Stop Function".

☞ Page 348 Instruction Initial Value to Start/Stop Function

- 0: Stop
- 1: Start (Initial value)

### ■Setting No.1 (Un\G1249302 to Un\G1250499)

| Address | Name | Description |
|---|---|---|
| Un\G1249302 | IP address | Third octet, fourth octet |
| Un\G1249303 | | First octet, second octet |
| Un\G1249304 | Port No. | Stores the port number set for setting No.1. |
| Un\G1249305 | Protocol number of error | Stores the protocol number of an error detected in setting No.1. |
| Un\G1249306 | Error code for simple device communication | Stores the latest error code of an error detected in setting No.1. |
| Un\G1249307 | Error clear request | Stores the contact to clear the error code detected in setting No.1 in the program.<br>• 0H: Error clear not instructed<br>• 1H: Error clear request (set by users)<br>• 2H: Error clear completed (set by system) |
| Un\G1249308.0 | Ready | Stores the ready status of setting No.1.<br>• 0: Not ready<br>• 1: Ready |
| Un\G1249309 | Status of connection | Stores the connection status of setting No.1.<br>• 0H: Unset<br>• 1H: Preparing<br>• 2H: Waiting for the request<br>• 3H: Communicating<br>• 4H: Function being stopped<br>• 5H: Retry being executed<br>• 6H: Monitoring at error |
| Un\G1249310 to Un\G1249311 | Number of successful completion | Stores the number of times that protocols executed in setting No.1 are successfully completed.<br>• 0: Protocol not executed<br>• 1 to 4294967295: Number of successful completion (When the number reaches or exceeds 4294967295, the value does not change.) |
| Un\G1249312 to Un\G1249313 | Number of error completion | Stores the number of times that protocols executed in setting No.1 are completed with error.<br>• 0: Protocol not executed<br>• 1 to 4294967295: Number of error completion (When the number reaches or exceeds 4294967295, the value does not change.) |
| Un\G1249314 to Un\G1249315 | Retry count | Stores the number of protocol retries executed in setting No.1.<br>• 0: Protocol not executed<br>• 1 to 4294967295: Number of retries (When the number reaches or exceeds 4294967295, the value does not change.) |
| Un\G1249316 | Protocol number currently executing | Stores the protocol number currently executed in setting No.1. (Stores the value at a timing of execution of the protocol and holds the value until the next protocol execution.)<br>• 0: The protocol has never been executed.<br>• Other than 0: Protocol number currently executed |
| Un\G1249317 | Verification mismatch protocol number | Stores the latest protocol number that results in the verification mismatch in setting No.1.<br>This area will not be cleared even when the verification mismatch has been resolved. To clear the area, use 'Error clear request' (Un\G1249307). |

| Address | Name | Description |
|---|---|---|
| Un\G1249318 to Un\G1249333 | Verification mismatch information (receive packet numbers 1 to 16) | Displays the verification mismatch information of the protocol number stored in 'Verification mismatch protocol number' (Un\G1249317) for each receive packet.<br>bit 0 to 7: Element number where the verification mismatch occurred<br>• 0H: Verification matched<br>• 1 to 32: Element number where the verification mismatch occurred<br>• FEH: Whole packets<br>• FFH: Verification not performed<br>bit 8 to F: Cause of the verification mismatch*1<br>This area will not be cleared even when the verification mismatch has been resolved. To clear the area, use 'Error clear request' (Un\G1249307). |
| Un\G1249334 to Un\G1249347 | System area | — |
| Un\G1249348 to Un\G1249379 | Start request for each protocol | Stores the request contact to start the execution of each protocol in setting No.1.<br>• Un\G1249348.0: Protocol 1<br>• Un\G1249348.1: Protocol 2<br>⋮<br>• Un\G1249379.15: Protocol 512<br>The default differs depending on the communication setting of the protocol settings.<br>• Default when the communication setting is "Fixed Intrvl": On (Start)<br>• Default when the communication setting is "Request": Off (Stop) |
| Un\G1249380 to Un\G1249411 | Execution completion for each protocol | Stores the execution completion status for each protocol of setting No.1.<br>• Un\G1249380.0: Protocol 1<br>• Un\G1249380.1: Protocol 2<br>⋮<br>• Un\G1249411.15: Protocol 512<br>Default: Off (Not completed) |
| Un\G1249412 to Un\G1249443 | Error for each protocol | Stores the error status for each protocol of setting No.1.<br>• Un\G1249412.0: Protocol 1<br>• Un\G1249412.1: Protocol 2<br>⋮<br>• Un\G1249443.15: Protocol 512<br>Default: Off (Not occurred) |
| Un\G1249444 to Un\G1249475 | Execution status for each protocol | Stores the data communication status for each protocol of setting No.1.<br>• Un\G1249444.0: Protocol 1<br>• Un\G1249444.1: Protocol 2<br>⋮<br>• Un\G1249475.15: Protocol 512<br>Default: Off (Communication stop) |
| Un\G1249476 to Un\G1249987 | Execution interval (current value) for each protocol | Stores the current value of the execution interval for each protocol whose communication setting is "Fixed Intrvl" in setting No.1.<br>• Un\G1249476: Protocol 1<br>• Un\G1249477: Protocol 2<br>⋮<br>• Un\G1249987: Protocol 512<br>Current value is 0: Unset (function not used), communications impossible<br>Current value is other than 0: Execution interval (ms) |
| Un\G1249988 to Un\G1250499 | Execution interval (maximum value) for each protocol | Stores the maximum value of the execution interval for each protocol whose communication setting is "Fixed Intrvl" in setting No.1.<br>• Un\G1249988: Protocol 1<br>• Un\G1249989: Protocol 2<br>⋮<br>• Un\G1250499: Protocol 512<br>Maximum value is 0: Unset (function not used), communications impossible<br>Maximum value is other than 0: Execution interval (ms) |

*1 The following table lists the verification result codes and causes of verification mismatch.

| Stored value | Description | Cause |
|---|---|---|
| 0H | No mismatch information | — |
| 1H | Data not matched | The static data did not match the setting value. |
| 2H | Insufficient receive data | The receive data is shorter than the length specified in the packet configuration. |
| 3H | Packet configuration not matched | Static data that indicates the end was not received after variable length data. |
| 4H | Packet length exceeded | • According to verification results, the packet length exceeded the maximum value.<br>• The receive data is longer than the length specified in the packet configuration. |
| 10H | Length value range error | The length value exceeded the maximum value for packet length. |
| 11H | Length value not matched | The length value did not match the received packet. |
| 20H | ASCII-binary conversion error | • The format of the length (ASCII) or conversion variable is different from the specified format.<br>• Analysis result of data received as a conversion variable was completed with an error. |
| 21H | Insufficient number of digits | • The number of digits received as a conversion variable (ASCII HEX → HEX) was an odd number.<br>• Data with a number of digits that is less than the number of digits specified by the conversion variable was received. |
| 22H | Number of digits exceeded | The number of digits of a conversion variable (the number of digits is variable) exceeded the range that can be stored in the data storage area.<br>As a conversion variable (data quantity is fixed and the number of digits is variable), data quantity more than the data quantity set in "Number of Data/Maximum Number of Data" was received. |
| 23H | Numerical value outside the range | A value of a conversion variable after conversion was outside the range that can be stored in the data storage area. |
| 30H | Error check code error | The calculated error check code and the received error check code did not match. |
| FEH | Configuration error of simple device communication packet | Incorrect protocol data is set in "Simple Device Communication Setting". |

The following formulas can be used for calculating the buffer memory addresses and bit positions for Start request for each protocol, Execution completion for each protocol, Error for each protocol, and Execution status for each protocol.

- Buffer memory address[*1]: Offset value + (1198 × ([Setting No.] - 1)) + (([Protocol number] - 1) / 16)
- Bit position: Remainder of ([Protocol number] - 1) / 16

In the calculation, assign the values listed below to the offset value.

| Name | Offset value |
|---|---|
| Start request for each protocol | 1249348 |
| Execution completion for each protocol | 1249380 |
| Error for each protocol | 1249412 |
| Execution status for each protocol | 1249444 |

*2 Round down after decimal places

**Ex.**

To obtain the buffer memory address of Start request for each protocol (setting No.32, protocol number 512)

Address: 1249348 + (1198 × (32 - 1)) + ((512 - 1) / 16) = 1286517.9375

Bit position: (512 - 1) / 16 = 31 with a remainder of 15 (FH)

The buffer memory address and bit position are Un\G1286517.F.

### ■Setting No.2 to 64 (Un\G1250500 to Un\G1325973)

These areas are used for storing the information of setting No.2 to 64 in the same order as setting No.1.

### ■User setting area (Un\G722500 to Un\G1246787)

This area can be used freely during simple CPU communication and simple device communication.

## Ethernet PORT1/2 common information

### ■Open completion signal (Un\G1900000 to Un\G1900007)

The open status of each connection is stored.

- 0: Closed or not open
- 1: Open completed

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1900000 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1900007 | 128 | 127 | 126 | 125 | 124 | 123 | 122 | 121 | 120 | 119 | 118 | 117 | 116 | 115 | 114 | 113 |

The numbers in the table indicate connection numbers.

### ■Open request signal (Un\G1900008 to Un\G1900015)

The open processing status of each connection is stored.

- 0: No open request
- 1: Requesting open

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1900008 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1900015 | 128 | 127 | 126 | 125 | 124 | 123 | 122 | 121 | 120 | 119 | 118 | 117 | 116 | 115 | 114 | 113 |

The numbers in the table indicate connection numbers.

### ■Socket/fixed buffer reception status signal (Un\G1900016 to Un\G1900023)

The reception status of each connection is stored.

- 0: Data not received
- 1: Data reception completed

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1900016 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1900023 | 128 | 127 | 126 | 125 | 124 | 123 | 122 | 121 | 120 | 119 | 118 | 117 | 116 | 115 | 114 | 113 |

The numbers in the table indicate connection numbers.

### ■Initial status (Un\G1900024)

| Address | Description |
|---|---|
| Un\G1900024 | Stores the status of the initial processing of the RJ71EN71 or the RnENCPU (network part).<br>Initial normal completion status (b0)<br>0: —<br>1: Initialization normal completion<br>Initial abnormal completion status (b1)<br>0: —<br>1: Initialization abnormal completion<br>b2 to b15: Use prohibited |

### ■Initial error code (Un\G1900025)

| Address | Description |
|---|---|
| Un\G1900025 | Stores the information when the initial processing of the RJ71EN71 or the RnENCPU (network part) is completed abnormally.<br>0: In initial processing or initial normal completion<br>Other than 0: Initial processing error code (An error code is stored.) |

## Receive buffer status storage area

### ■State of receive buffer (Un\G1900030)

| Address | Description |
|---|---|
| Un\G1900030 | Stores the receive buffer status.<br>0: Receive buffer not full<br>1: Receive buffer full |

## Predefined protocol support function check area

### ■Predefined protocol ready (Un\G1901002)

| Address | Description |
|---|---|
| Un\G1901002 | Stores the ready status of the protocol setting data.<br>0: —<br>1: Ready |

## Predefined protocol setting data check area

### ■Predefined protocol setting data error information (Un\G1901020 to Un\G1901023)

| Address | Name | Description |
|---|---|---|
| Un\G1901020 | Protocol number | When a protocol setting data error is detected, stores the protocol number where the error was detected.<br>Protocol is checked in order from smallest protocol number. The protocol number where an error was detected first is stored.<br>0: No error<br>1 to 128: Protocol number<br>65535: Cannot identify*1 |
| Un\G1901021 | Setting type | 0 is stored if an error is detected in the packet setting or element setting.<br>1 is stored if an error is detected in the protocol detailed setting. (Valid when protocol number value is 1 to 128)<br>0: Packet setting or element setting<br>1: Protocol preferences<br>65535: Cannot identify*1 |
| Un\G1901022 | Packet number | When an error is detected in the protocol setting data, stores the packet number that detected the error.<br>The packets are checked in order of send packets and then receive packets (expected packets) from smallest number. The packet number where an error was detected first is stored. (Valid when setting type value is 0)<br>0: Send packet<br>1 to 16: Receive packet number<br>65535: Cannot identify*1 |
| Un\G1901023 | Element number | When an error is detected in the protocol setting data, stores the element number where the error was detected.<br>The elements are checked in order of smallest element number. The element number where an error was detected first is stored. (Valid when setting type value is 0)<br>1 to 32: Element number<br>65535: Cannot identify*1 |

*1 The setting value may be unidentifiable (65535) in the following cases.
· When a setting that cannot be detected by the current Ethernet-equipped module version is written in
· When protocol setting data is broken (hardware failure)

### ■Number of registered predefined protocols (Un\G1901024)

| Address | Description |
|---|---|
| Un\G1901024 | Stores the protocol number of the registered protocol setting data.<br>0 is stored if the protocol setting data check result is abnormal.<br>0: No registration<br>1 to 128: Number of registrations |

### ■Predefined protocol registration (Un\G1901032 to Un\G1901047)

Whether protocol setting data is registered or not is stored.

All bits are set to 0 if the protocol setting data check result is abnormal.

- 0: No registration
- 1: Registered

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1901032 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1901039 | 128 | 127 | 126 | 125 | 124 | 123 | 122 | 121 | 120 | 119 | 118 | 117 | 116 | 115 | 114 | 113 |
| Un\G1901040 to Un\G1901047 | Empty | | | | | | | | | | | | | | | |

The numbers in the table indicate protocol numbers.

## Send/receive area for predefined protocol support function

### ■Send/receive area for predefined protocol support function (Un\G1902000 to Un\G1904047)

This area is used for sending and receiving data during communication with predefined protocol.

## User setting area for MODBUS/TCP function

### ■User setting area for MODBUS/TCP function (Un\G1921024 to Un\G1929215)

This area is used for sending and receiving data during communications using MODBUS/TCP.

## Discard received data at CPU STOP setting area

### ■Discard received data at CPU STOP setting area (Un\G1950032 to Un\G1950039)

Stores one of the following bit patterns for each connection number.

- 0: Disable (The receive data is not discarded during CPU STOP.)
- 1: Enable (The receive data is discarded during CPU STOP.)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1950032 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1950039 | 128 | 127 | 126 | 125 | 124 | 123 | 122 | 121 | 120 | 119 | 118 | 117 | 116 | 115 | 114 | 113 |

The numbers in the table indicate connection numbers.

## Setting area for re-initial processing

### ■Reinitialization specification (Un\G1950040)

Request re-initial processing.

- 0: Reinitialization process complete (reset by the system)
- 1: Reinitialization process request (set by the user)

### ■Reinitialization specification error factor (Un\G1950041 to Un\G1950042)

When re-initial processing is performed, if the value specified in the setting area is out of range, the corresponding bits of this area will turn on.

| Address | Name | Description |
|---|---|---|
| Un\G1950041 | Reinitialization specification error factor (Port1) | b0: Transmission speed/communication mode change setting out of range<br>b1: Own station IP address setting out of range<br>b2: Re-initial processing execution error (Simple device communication function)<br>b3: Re-initial processing execution error (Redundant function)<br>b4 to b15: Use prohibited |
| Un\G1950042 | Reinitialization specification error factor (Port2) | |

### ■Reinitialization specification change target (Un\G1950043 to Un\G1950044)

When changing the following parameters, specify the modification target.

- Own station IP address
- Ethernet operation setting
- Transmission speed
- Communication mode

| Address | Name | Description |
|---|---|---|
| Un\G1950043 | Reinitialization specification change target (Port1) | ■b0: Modification specification for the own station IP address<br>Specify whether to change the own station IP address.<br>(When changing the IP address, specify it using 'Reinitialization Own IP Address' (Un\G1950045 to Un\G1950048).)<br>• 0: Do not change<br>• 1: Change<br>■b1: Modification specification for the Ethernet operation setting<br>Specify whether to change the Ethernet operation setting.<br>(When changing the setting, specify it using 'Communication condition setting (Ethernet Operation Setting) area' (Un\G1950049 to Un\G1950050).)<br>• 0: Do not change<br>• 1: Change<br>■b2 to b11: Use prohibited<br>■b12 to b15: Modification specification for the transmission speed/communication mode<br>Specify the transmission speed and communication mode.<br>• 0: Do not change<br>• 1: Automatic negotiation<br>• 2: 100Mbps/full-duplex communication<br>• 3: 100Mbps/half-duplex communication<br>• 4: 10Mbps/full-duplex communication<br>• 5: 10Mbps/half-duplex communication<br>• 6: 1Gbps/full-duplex communication |
| Un\G1950044 | Reinitialization specification change target (Port2) | |

### ■Reinitialization Own IP Address (Un\G1950045 to Un\G1950048)

Specify the IP address of the own station. (Setting range: 00000001H to DFFFFFFEH)

- Port 1: Un\G1950045 to Un\G1950046
- Port 2: Un\G1950047 to Un\G1950048

Ex.

If setting the IP address of port 1 to 192.168.3.50, write the following values.

Un\G1950045: 0332H

Un\G1950046: C0A8H

## ■Communication condition setting (Ethernet Operation Setting) area (Un\G1950049 to Un\G1950050)

Set the Ethernet operation setting.

| Address | Name | Description |
|---|---|---|
| Un\G1950049 | Communication condition setting (Ethernet Operation Setting) area (Port 1) | b0: Use prohibited<br>b1: Communication data code setting<br>• 0: Communications in a binary code<br>• 1: Communications in an ASCII code<br>b2 to b5: Use prohibited<br>b6: Enable/disable online change<br>• 0: Disable all (SLMP)<br>• 1: Enable all (SLMP)<br>b7: Use prohibited<br>b8: Opening method<br>• 0: Open by program<br>• 1: Do not open by program<br>b9 to b15: Use prohibited |
| Un\G1950050 | Communication condition setting (Ethernet Operation Setting) area (Port 2) | |

# Details of buffer memory addresses (CPU module (built-in Ethernet port part))

This section descries the details of buffer memory addresses of the CPU module (built-in Ethernet port part).

## Own node setting status storage area

### ■Own node IP address (Un\G50, Un\G51)

The IP address set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G50 | Third octet, fourth octet |
| Un\G51 | First octet, second octet |

### ■Subnet mask (Un\G60, Un\G61)

The subnet mask set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G60 | Third octet, fourth octet |
| Un\G61 | First octet, second octet |

### ■Default gateway IP address (Un\G64, Un\G65)

The default gateway IP address set with the module parameter is stored.

| Address | Description |
|---|---|
| Un\G64 | Third octet, fourth octet |
| Un\G65 | First octet, second octet |

### ■Own node MAC address (Un\G74 to Un\G76)

The MAC address of the Ethernet-equipped module is stored.

| Address | Description |
|---|---|
| Un\G74 | 5th byte, 6th byte of the MAC address |
| Un\G75 | 3rd byte, 4th byte of the MAC address |
| Un\G76 | 1st byte, 2nd byte of the MAC address |

### ■Network number/Station number setting status (Un\G87)

This area is used for storing the setting status of the network number/station number.

| Address | Name | Description |
|---|---|---|
| Un\G87 | Network number/Station number setting status | Stores the setting status of the network number/station number that has been set with parameters or by the IP address change function.<br>b0: Network number/station number validity flag (0: Disabled, 1: Enabled) |

## Communication load status

### ■Receive packet count per unit time (one second) (Un\G100, Un\G101)

The number of receive packets per unit time (one second) of the built-in Ethernet port communications is stored.

| Address | Name | Description |
|---|---|---|
| Un\G100 | Receive packet count (current) per unit time (one second) | ■The number of receive packets per unit time (one second) of the built-in Ethernet port is stored.<br>■When the processing speed of the built-in Ethernet communications is slow and if this value is large, the following actions may improve the processing speed.<br>• Review the communication request from the engineering tool and GOT.<br>• Review the communication request from the transfer destination (execution source) of the simple CPU communications.<br>• Review the file transfer function (FTP server).<br>• Review the communications using the SLMP. |
| Un\G101 | Receive packet count (maximum) per unit time (one second) | The maximum value of the number of receive packets per unit time (one second) of the built-in Ethernet port is stored. |

### ■Send packet count per unit time (one second) (Un\G102, Un\G103)

The number of send packets per unit time (one second) of the built-in Ethernet port communications is stored.

| Address | Name | Description |
|---|---|---|
| Un\G102 | Send packet count (current) per unit time (one second) | ■The number of send packets per unit time (one second) of the built-in Ethernet port is stored.<br>■When the processing speed of the built-in Ethernet communications is slow and if this value is large, the following actions may improve the processing speed.<br>• Review the execution frequency of the instructions for the built-in Ethernet function (socket communications instruction/predefined protocol support function instruction/ SLMP frame send instruction/file transfer function instruction)<br>• Review the settings (such as the number of points) of CC-Link IE Field Network Basic.<br>• Review the settings (such as the execution interval and the number of points) of the simple CPU communications.<br>• Review the settings (such as automatic detection and backup execution) of iQSS (built-in Ethernet).<br>• Review the settings (such as the number of executions) of data logging file transfer. |
| Un\G103 | Send packet count (maximum) per unit time (one second) | The maximum value of the number of send packets per unit time (one second) of the built-in Ethernet port is stored. |

## Connection status storage area

### ■Connection No.1 latest error code to connection No.16 latest error code (Un\G130 to Un\G145)

The latest error code for each connection is stored.

| Address | Name |
|---|---|
| Un\G130 | Connection No.1 latest error code |
| Un\G131 | Connection No.2 latest error code |
| ⋮ | |
| Un\G144 | Connection No.15 latest error code |
| Un\G145 | Connection No.16 latest error code |

## Own node operation status storage area

### ■Hub connection status area (Un\G242 to Un\G245)

The hub connection status of the Ethernet-equipped module is stored.

| Address | Name | Description |
|---|---|---|
| Un\G242 | Communication mode | Stores the communication mode.<br>0: Half-duplex<br>1: Full-duplex |
| Un\G243 | Connection status | Stores the connection status.<br>0: Hub not connected or disconnected<br>1: Hub connected |
| Un\G244 | Communication speed | Stores the communication speed.<br>0: Operating at 10BASE-T<br>1: Operating at 100BASE-TX<br>2: Operating at 1000BASE-T |
| Un\G245 | Disconnection count | Stores the number of times the cable was disconnected. |

### ■IP address duplication status storage area (Un\G251 to Un\G257)

Information when IP address is duplicated is stored.

| Address | Name | Description |
|---|---|---|
| Un\G251 | Same IP address detection flag | Stores IP address duplication status.<br>0: IP address not duplicated<br>1: IP address duplicated |
| Un\G252 to Un\G254 | MAC address of the station already connected to the network | Stores the MAC address of the station that has been already connected to the network in the station with duplicated IP address.<br>Un\G252: 5th and 6th byte of the MAC address<br>Un\G253: 3rd and 4th byte of the MAC address<br>Un\G254: 1st and 2nd byte of the MAC address<br>"FFFFFFFFFFFFH" is stored in the station that has been already connected to the network. |
| Un\G255 to Un\G257 | MAC address of the station with the IP address already used | Stores the MAC address of the station with duplicated IP address in the station that has been already connected to the network.<br>Un\G255: 5th and 6th byte of the MAC address<br>Un\G256: 3rd and 4th byte of the MAC address<br>Un\G257: 1st and 2nd byte of the MAC address<br>"FFFFFFFFFFFFH" is stored in the station with the duplicated IP address. |

## Remote password lock status storage area

### ■Remote password lock status system port (Un\G270)

The remote password lock status of each connection is stored.

- 0: Unlocked or remote password not set
- 1: Lock status

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G270 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

The numbers in the table indicate connection numbers.

### ■Remote password lock status system port (Un\G271)

| Address | Description |
|---|---|
| Un\G271 | Stores the remote password lock status of the system port in b0 to b4.<br>0: Unlocked or remote password not set<br>1: Lock status<br>The bits corresponding to each system port are shown below.<br>b0: Auto-open UDP port<br>b1: MELSOFT transmission port (UDP/IP)<br>b2: MELSOFT transmission port (TCP/IP)<br>b3: FTP transmission port<br>b4: MELSOFT direct connection |

## Discard received data at CPU STOP setting area

### ■Discard received data at CPU STOP setting area (Un\G275)

Stores one of the following bit patterns for each connection number.

- 0: Disable (The receive data is not discarded during CPU STOP.)
- 1: Enable (The receive data is discarded during CPU STOP.)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G275 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

The numbers in the table indicate connection numbers.

## Forced connection invalidation setting area

If the connection to be established is already unintentionally "connected", the connection may have been done deliberately in bad faith. Turn on the area that corresponds to the target connection and execute forced invalidation.

### ■Forced connection invalidation setting area (Un\G280)

Set the connection to be forcibly invalidated.

- 0: Use allowed
- 1: Use prohibited

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G280 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

The numbers in the table indicate connection numbers.

### ■Forced connection invalidation system port (Un\G281)

| Address | Description |
|---|---|
| Un\G281 | Set the system port to be forcibly invalidated.<br>0: Use allowed<br>1: Use prohibited<br>The bits corresponding to each system port are shown below.<br>b0: Auto-open UDP port<br>b1: MELSOFT transmission port (UDP/IP)<br>b2: MELSOFT transmission port (TCP/IP)<br>b3: FTP transmission port<br>b4: MELSOFT direct connection |

## Time setting function (SNTP) area

### ■Time setting function operation result (Un\G290)

| Address | Description |
|---|---|
| Un\G290 | Stores the operation result of the time setting function.<br>0: Unexecuted<br>1: Success<br>FFFFH: Failure |

### ■Time setting function execution time (Un\G291 to Un\G297)

| Address | Name | Description |
|---|---|---|
| Un\G291 | Year | Stores the year that the time setting function was executed. |
| Un\G292 | Month | Stores the month that the time setting function was executed. |
| Un\G293 | Day | Stores the date that the time setting function was executed. |
| Un\G294 | Hour | Stores the time (hour) that the time setting function was executed. |
| Un\G295 | Minute | Stores the time (minute) that the time setting function was executed. |
| Un\G296 | Second | Stores the time (second) that the time setting function was executed. |
| Un\G297 | Day of the week | Stores the day of the week that the time setting function was executed.<br>0: Sunday<br>1: Monday<br>2: Tuesday<br>3: Wednesday<br>4: Thursday<br>5: Friday<br>6: Saturday |

### ■Time setting function required response time (Un\G298)

| Address | Description |
|---|---|
| Un\G298 | Stores the time required for the module to set the time after sending to the SNTP server.<br>Range: 0 to FFFEH (Unit: ms) |

### ■Time setting function (SNTP client) execution (Un\G229)

| Address | Description |
|---|---|
| Un\G299 | Executes the time setting function when b0 is turned on.<br>The time setting function can be executed at a random time from a program.<br>The function is not executed if b0 is turned on during execution of the time setting function.<br>This is valid only when "Use" is set for "Time Setting (SNTP Client)" under "Time Setting" in "Application Settings". |

## Connection close wait time setting area for socket communications (TCP/IP)

### ■Enabling/disabling connection close wait time setting (Un\G760)

Set enabling/disabling connection close wait time setting for each connection (Nos. 1 to 16) using the bit patterns listed below.

- 0: Disable
- 1: Enable*1

*1 To close the connection and then reopen it from the external device with this setting enabled, the external device needs to reopen it after the close request is sent and the timer value of the latency time setting has elapsed. Reopening the connection before the time has elapsed may fail the open processing.

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G760 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |

The numbers in the table indicate connection numbers.

This setting is applicable only to socket communications (TCP/IP). Other communication methods are excluded.

When the communication is disconnected from the external device, the setting of enable/disable is reflected. However, this does not apply when the own CPU module executes SP.SOCCLOSE to disconnect the communication.

### ■Connection close wait time setting timer value (Un\G761)

| Address | Description |
|---|---|
| Un\G761 | Set the timer value (ms) of the connection close wait time setting.<br>Range: 0 to 5000*1<br>When Un\G760 is "Enable" at the disconnection, the disconnection timing of the own CPU module is delayed by the timer setting value of Un\G761. |

*1 When a value outside the range (5001 to 65535) is set, the latency time is 5000ms.

## Connection status storage area

### ■Latest error code connection No.1 (communication error) to latest error code connection No.16 (communication error) (Un\G770 to Un\G785)

The latest error code (communication error) for each connection is stored.

| Address | Name |
|---|---|
| Un\G770 | Latest error code connection No.1 (communication error) |
| Un\G771 | Latest error code connection No.2 (communication error) |
| ⋮ | |
| Un\G784 | Latest error code connection No.15 (communication error) |
| Un\G785 | Latest error code connection No.16 (communication error) |

### ■FTP server latest error code (communication error) (Un\G786)

The latest error code (communication error) for the FTP server is stored.

| Address | Name |
|---|---|
| Un\G786 | FTP server latest error code (communication error) |

### ■MELSOFT direct connection latest error code (communication error) (Un\G787)

The latest error code (communication error) for MELSOFT direct connection is stored.

| Address | Name |
|---|---|
| Un\G787 | MELSOFT direct connection latest error code (communication error) |

### ■FTP client latest error code (communication error) (Un\G788)

The latest error code (communication error) for the FTP client is stored.

| Address | Name |
|---|---|
| Un\G788 | FTP client latest error code (communication error) |

### ■Auto-open UDP port latest error code (communication error) (Un\G789)

The latest error code (communication error) for the auto-open UDP port is stored.

| Address | Name |
|---|---|
| Un\G789 | Auto-open UDP port latest error code (communication error) |

### ■MELSOFT connection auto-open UDP port latest error code (communication error) (Un\G790)

The latest error code (communication error) for the MELSOFT connection auto-open UDP port is stored.

| Address | Name |
|---|---|
| Un\G790 | MELSOFT connection auto-open UDP port latest error code (communication error) |

### ■MELSOFT connection auto-open TCP port latest error code (communication error) (Un\G791)

The latest error code (communication error) for the MELSOFT connection auto-open TCP port is stored.

| Address | Name |
|---|---|
| Un\G791 | MELSOFT connection auto-open TCP port latest error code (communication error) |

### ■Send buffer memory number of exhaustion (Un\G792)

The insufficient send buffer or memory count is stored.

| Address | Name |
|---|---|
| Un\G792 | Send buffer memory number of exhaustion |

## Simple CPU communication function

### ■Request to start communication at request for each setting number (Un\G1090 to Un\G1093)

This request contact sends and receives the data when the communication setting is "On Request" and is stored using the following bit pattern for each setting number.

- Off → On: Requested (start request)
- On → Off: Completed (start completion)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1090 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1093 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

### ■Request to stop fixed interval communication for each setting number (Un\G1094 to Un\G1097)

This request contact stops the data send when the communication setting is "Fixed Interval" and is stored using the following bit pattern for each setting number.

- Off → On: Requested (stop request)
- On → Off: Completed (stop completion)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1094 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1097 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

### ■Request to restart fixed interval communication for each setting number (Un\G1098 to Un\G1101)

This request contact restarts the data send when the communication setting is "Fixed Interval" and is stored using the following bit pattern for each setting number.

- Off → On: Requested (restart request)
- On → Off: Completed (restart completion)

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1098 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1101 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

### ■Execution status flag for each setting number (Un\G1102 to Un\G1105)

The status of data communication is stored using the following bit pattern for each setting number.

- Off: Communication stop (function not used)
- On: Communicating

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1102 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1105 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

### ■Preparation completion for each setting number (Un\G1106 to Un\G1109)

The preparation completion status of the simple CPU communication is stored using the following bit pattern for each setting number.

- Off: Not ready (function not used)
- On: Ready

| Address | b15 | b14 | b13 | b12 | b11 | b10 | b9 | b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | b0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Un\G1106 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| ⋮ | | | | | | | | | | | | | | | | |
| Un\G1109 | 64 | 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 |

### ■Simple CPU communication status for each setting number (Un\G1130 to Un\G1193)

| Address | Name | Description |
|---|---|---|
| Un\G1130 | Simple PLC status setting No.1 | The simple CPU communication status is stored.<br>0H: Unset (function not used)<br>1H: Preparing<br>2H: Waiting for the request<br>3H: Communicating<br>4H: Communication stop<br>5H: Retry being executed<br>6H: Monitoring at error<br>AH: Communications impossible |
| Un\G1131 | Simple PLC status setting No.2 | |
| ⋮ | | |
| Un\G1192 | Simple PLC status setting No.63 | |
| Un\G1193 | Simple PLC status setting No.64 | |

### ■Simple CPU communication error code for each setting number (Un\G1194 to Un\G1257)

| Address | Name | Description |
|---|---|---|
| Un\G1194 | Simple PLC error code setting No.1 | The cause of the error detected in the simple CPU communication is stored.<br>0: No error (function not used)<br>Other than 0: Error code (☞ Page 453 List of Error Codes)<br>The value is cleared to 0 with a clear request from the engineering tool. |
| Un\G1195 | Simple PLC error code setting No.2 | |
| ⋮ | | |
| Un\G1256 | Simple PLC error code setting No.63 | |
| Un\G1257 | Simple PLC error code setting No.64 | |

### ■Execution interval (current value) for each setting number (Un\G1258 to Un\G1321)

| Address | Name | Description |
|---|---|---|
| Un\G1258 | Execution interval (current value) setting No.1 | If "Fixed Interval" is set for communication setting, the current value of the execution interval is stored.<br>0: Unset (function not used), communications impossible, "Communication Setting" is "On Request"<br>Other than 0: Execution interval (unit: ms) |
| Un\G1259 | Execution interval (current value) setting No.2 | |
| ⋮ | | |
| Un\G1320 | Execution interval (current value) setting No.63 | |
| Un\G1321 | Execution interval (current value) setting No.64 | |

### ■Abnormal response code for each setting number (Un\G1322 to Un\G1385)

| Address | Name | Description |
|---|---|---|
| Un\G1322 | Abnormal response code setting No.1 | The abnormal response code detected in the simple CPU communication is stored.<br>0: No error (function not used)<br>Other than 0: Abnormal response code<br>For abnormal response codes, refer to the manual for the external device. |
| Un\G1323 | Abnormal response code setting No.2 | |
| ⋮ | | |
| Un\G1384 | Abnormal response code setting No.63 | |
| Un\G1385 | Abnormal response code setting No.64 | |

# Appendix 4 Dedicated Instruction

This section describes the dedicated instruction that can be used in Ethernet.
The following shows the meaning of the marks in the tables of this chapter.
○: Available, ×: Not available

**Point**

For details on dedicated instructions used in the RJ71EN71 and RnENCPU (network part), refer to the following.
📖 MELSEC iQ-R Programming Manual (Module Dedicated Instructions)
For details on dedicated instructions used in the CPU module (built-in Ethernet port part), refer to the following.
📖 MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks)

## Precautions

### ■Data change

Do not change any data specified (such as control data) until execution of the dedicated instruction is completed.

### ■When the dedicated instruction is not completed

Check whether "Module Operation Mode" in "Application Settings" of the RJ71EN71 and the RnENCPU (network part) is "Online".
A dedicated instruction cannot be executed when the mode is "Offline" or "Module Communication Test".

### ■Information including dedicated instructions that perform open processing

When performing open processing using the following dedicated instructions and module FB, start transmission and reception only after the dedicated instructions and module FB currently being executed for the same connection have completed.

- OPEN instruction
- CONOPEN instruction
- SOCOPEN instruction
- M+Model name_ConnectionOpen (Module FB that performs open processing internally)

### ■When dedicated instructions or others are simultaneously executed for the same connection

The RnENCPU (CPU part) cannot simultaneously execute dedicated instructions and module FBs for the same connection. *1
When the RJ71EN71 or RnENCPU (network part) simultaneously executes dedicated instructions and module FBs for the same connection, the operation is as described below:

- Communications using the fixed buffer, communications using the predefined protocol: Dedicated instructions and module FBs cannot be simultaneously executed. *1
- Socket communications: Regarding the following dedicated instructions and module FBs, simultaneous transmission and reception are possible by specifying the same connection number. *2

○: Simultaneous execution possible, ×: Simultaneous execution not possible

| Dedicated instruction/ module FB | SOCSND instruction | SOCRCV instruction | SOCRCVS instruction | M+Model name_Send_Socket | M+Model name_Recv_Socket |
|---|---|---|---|---|---|
| SOCSND instruction | × | ○ | ○ | × | ○ |
| SOCRCV instruction | ○ | × | × | ○ | × |
| SOCRCVS instruction | ○ | × | × | ○ | × |
| M+Model name_Send_Socket | × | ○ | ○ | × | ○ |
| M+Model name_Recv_Socket | ○ | × | × | ○ | × |

*1 Even if dedicated instructions and module FBs are simultaneously executed, those which will be executed later are not processed. Execute again after the dedicated instruction or module FB in execution is completed.
*2 The simultaneous execution of dedicated instructions and module FBs depends on the firmware version of the module. (☞ Page 586 Added and Enhanced Functions)

## Open/close processing instructions

The following table lists the Ethernet instructions to connect or disconnect the connection with the communication destination.

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| GP.OPEN | Establishes a connection. (Q series compatible instruction) | ○ | × |
| ZP.OPEN | | ○ | × |
| GP.CONOPEN | Establishes a connection. | ○ | × |
| SP.SOCOPEN | | × | ○ |
| GP.CLOSE | Closes the connection. (Q series compatible instruction) | ○ | × |
| ZP.CLOSE | | ○ | × |
| GP.CONCLOSE | Closes the connection. | ○ | × |
| SP.SOCCLOSE | | × | ○ |

## Instruction for SLMP communications

The following table lists the built-in Ethernet instructions used for communications using the SLMP.

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| SP.SLMPSND | Sends SLMP message to the SLMP-compatible device. | × | ○ |

## Instructions for predefined protocol communications

The following table lists the Ethernet instructions and built-in Ethernet instructions used for communications using the communication protocol.

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| GP.ECPRTCL | Executes the protocol registered with the engineering tool's communication protocol support function. | ○ | × |
| SP.ECPRTCL | | × | ○ |
| SP.SOCCINF*1 | Reads connection information | × | ○*2 |
| SP.SOCCSET*1 | Changes the communication target | × | ○*2 |

*1 These instructions are the same as the instruction for the socket communications (SP.SOCCINF/SP.SOCCSET). (MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks))

*2 Only the following models can be used. The supporting firmware version differs depending on the model.
· R00CPU, R01CPU, R02CPU: There are no restrictions on the version.
· Programmable controller CPUs other than the above: "29" or later

A

# Socket communications instructions

The following table lists the Ethernet instructions and built-in Ethernet instructions used for socket communications.

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| GP.SOCRCV | Reads the receive data from the external device. | ○ | × |
| SP.SOCRCV | | × | ○ |
| G.SOCRCVS | | ○ | × |
| S.SOCRCVS | | × | ○ |
| GP.SOCSND | Sends data to the external device. | ○ | × |
| SP.SOCSND | | × | ○ |
| SP.SOCCINF | Reads connection information | × | ○ |
| SP.SOCCSET | Changes the communication target | × | ○ |
| SP.SOCRMODE | Changes the connection receive mode. | × | ○ |
| S.SOCRDATA | Reads the specified size of data from the socket communication receive data area. | × | ○ |
| SP.SOCRDATA | | × | ○ |

**Point**

If the instruction has a completion device, do not change the various data (such as control data and request data) specified with the executed instruction until execution of the instruction is completed.

# Instructions for communications using the fixed buffer

The following table lists the Ethernet instructions used for communications using the fixed buffer.

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| GP.BUFRCV | Reads the receive data from the external device. | ○ | × |
| ZP.BUFRCV | | ○ | × |
| G.BUFRCVS | Reads the receive data with an interrupt program. | ○ | × |
| Z.BUFRCVS | | ○ | × |
| GP.BUFSND | Sends data to the external device. | ○ | × |
| ZP.BUFSND | | ○ | × |

# Link dedicated instructions

The following table lists the instructions common to Ethernet and CC-Link IE used for transient transmission with programmable controllers on other stations.
A station on a network other than Ethernet can also be accessed. (☞ Page 553 Communications with Different Networks)

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| JP.READ | Reads data from the word device of another station. | ○ | × |
| GP.READ | | ○ | × |
| JP.SREAD | Reads data from the word device of another station. (with completion device) | ○ | × |
| GP.SREAD | | ○ | × |
| JP.WRITE | Writes data in the word device of another station. | ○ | × |
| GP.WRITE | | ○ | × |
| JP.SWRITE | Writes data in the word device of another station. (with completion device) | ○ | × |
| GP.SWRITE | | ○ | × |
| JP.SEND | Sends data to another station. | ○ | × |
| GP.SEND | | ○ | × |
| JP.RECV | Reads the receive data from another station. (for main program) | ○ | × |
| GP.RECV | | ○ | × |
| G.RECVS | Reads the receive data from another station. (for interrupt program) | ○ | × |
| Z.RECVS | | ○ | × |
| J.ZNRD | Reads data from the word device in another station (ACPU). | ○ | × |
| JP.ZNRD | | ○ | × |
| J.ZNWR | Writes data in the word device in another station (ACPU). | ○ | × |
| JP.ZNWR | | ○ | × |
| J.REQ | Requests the remote RUN/STOP to the CPU module on another station. | ○ | × |
| JP.REQ | | ○ | × |
| G.REQ | | ○ | × |
| GP.REQ | | ○ | × |
| J.REQ | Reads/Writes the clock data from/to the CPU module of other stations. | ○ | × |
| JP.REQ | | ○ | × |
| G.REQ | | ○ | × |
| GP.REQ | | ○ | × |

### Precautions

When executing multiple link dedicated instructions simultaneously, check that the channels for the instructions are not duplicated. Link dedicated instructions with the same channel cannot be executed simultaneously. To use the same channel for multiple link dedicated instructions, configure an interlock so that an instruction is executed after completion of another.

# File transfer function instruction

The following table lists the built-in Ethernet instructions used for the file transfer function (FTP client).

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| SP.FTPPUT | This instruction sends files in the CPU module (FTP client) to the folder path of the specified FTP server. | × | ○ |
| SP.FTPGET | This instruction retrieves files on the FTP server to the folder path of the specified CPU module (FTP client). | × | ○ |

# Other dedicated instructions

The following table lists the other Ethernet instructions.

| Instruction symbol | Description | RJ71EN71, RnENCPU (network part) | CPU module (built-in Ethernet port part) |
|---|---|---|---|
| GP.ERRCLEAR | Turns off the LED and clears error information.*1 | ○ | × |
| ZP.ERRCLEAR | | ○ | × |
| GP.ERRRD | Reads error information. | ○ | × |
| ZP.ERRRD | | ○ | × |
| G.UINI | Performs re-initial processing. | ○ | × |
| GP.UINI | | ○ | × |
| Z.UINI | | ○ | × |
| ZP.UINI | | ○ | × |

*1 Availability for turning off the LED depends on the firmware version of the RJ71EN71. (☞ Page 586 Added and Enhanced Functions)

# Appendix 5 Initial Processing

This section describes information including operation of initial processing.

## Initial processing

Initial processing is the process of applying the data communication parameters to an Ethernet-equipped module to enable communication with an external device.
The initial processing is performed automatically, so no program is required to perform the processing.

### Procedure

The following describes a procedure for performing initial processing.

1. Set the parameters required for communication via Ethernet.
2. Write the set parameters to the CPU module.
3. Power off and on the CPU module or reset it.
4. The initial processing for the Ethernet-equipped module will be performed.

If no parameters are set, initial processing will be performed using default parameters.

### Checking the result of initial processing

The result of initial processing can be checked using the buffer memory of the Ethernet-equipped module.

#### ■For the RJ71EN71 or RnENCPU (network part)

| Item | Description |
|---|---|
| At normal completion | Bit 0 of 'Initial status' (Un\G1900024) turns on. |
| At abnormal completion | Bit 1 of 'Initial status' (Un\G1900024) turns on.<br>In addition, an error code is stored in 'Initial error code' (Un\G1900025), allowing error information to be checked. |

#### ■For the CPU module (built-in Ethernet port part)

| Item | Description |
|---|---|
| At normal completion | Bit 0 of 'Initial status' (Un\G683) turns on. |
| At abnormal completion | Bit 1 of 'Initial status' (Un\G683) turns on.<br>In addition, an error code is stored in 'Initial error code' (Un\G684), allowing error information to be checked. |

A

# Re-initial processing

Re-initial processing is the process of returning the Ethernet-equipped module to its startup state without powering off and on the programmable controller or resetting it.
In addition, communication with the external device can also be resumed by changing some of the Ethernet function parameters.
Re-initial processing is performed only on ports with the network type set for Ethernet. Even if communication is performed on other ports using CC-Link IE Field Network or CC-Link IE Controller Network, communication continues during re-initial processing.

**Restriction**

Re-initial processing cannot be performed if the following functions are used.
- Simple device communication function
- Redundant function (excluding Q compatible mode)

**Restriction**

Before performing re-initial processing, check the versions of the module and engineering tool to be used.
(☞ Page 586 RJ71EN71, RnENCPU (network part))

## Description

The following information is updated when re-initial processing is performed.
- Address information (IP address and MAC address) of the external device held by the Ethernet-equipped module
- Ethernet-equipped module parameters

## Parameters that can be changed in re-initial processing

The following parameters can be changed in re-initial processing.
The parameters changed during re-initial processing are restored to their previous values when the CPU module is powered on and off or is reset.

| Item | | Description |
|---|---|---|
| Basic Settings | Own Node Settings | IP Address*1 |
| | | Enable/Disable Online Change*2 |
| | | Communication Data Code*2 |
| | | Opening Method (When TCP (Passive) or UDP is set)*2 |
| Application Settings | Communication Speed | Communication Speed |

*1 When "Use IP Address" is selected for "Setting Method" under "Basic Settings", changing the IP address does not change the network number and station number. The values based on the IP address configured in the parameter remain as they are.
*2 Change the parameters for all connections at once, rather than changing them for each connection individually.
When changing the external device configuration settings for each connection individually, execute the CONOPEN instruction or M+Model name_ConnectionOpen (module FB that performs open processing internally).

## Execution method

The following methods are available for performing re-initial processing.

- Execute the UINI instruction (dedicated instruction). (This method is available only for ports with the network type set for Q-compatible Ethernet.)
- Use the 'Reinitialization process request' (Un\G1950040) of the buffer memory.

For the UINI instruction, refer to the following.

MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

**Point**

- Before performing re-initial processing, terminate all ongoing data communication with the external device and wait until close processing is complete.
- If the IP address of the Ethernet-equipped module is changed, reset the external device as well. If the external device has the MAC address of the communication destination, the change to the IP address may not be reflected, preventing ongoing communication from being continued.

## Procedure

The following describes a procedure for performing re-initial processing by buffer memory.

**1.** Check the normal completion of the initial processing.

Check whether bit 0 of 'Initial status' (Un\G1900024) is on (Initialization normal completion).

**2.** Terminate data communication with the external device currently in progress using the following method.

| Communication method | Termination method |
|---|---|
| Communication using the fixed buffer | Terminate all data communication with the external device currently in progress and perform close processing for all open connections. |
| Socket communication | |
| Communication using predefined protocol (including the MODBUS/TCP master function) | |
| Communication using the random access buffer | |
| SLMP communication | |
| Simple CPU communication (including the MODBUS/TCP master function) | For the setting numbers whose current communication setting is "Fixed Interval", turn on the corresponding bits of 'Request to stop fixed interval communication for each setting number' (Un\G721900 to Un\G721903, Un\G1247328 to Un\G1247355). |
| MELSOFT connection | Terminate all data communication with the external device currently in progress and close the connection from the external device. |
| MODBUS/TCP slave function | |
| FTP | |
| SNTP | No operation is required because the system performs close processing. |

**3.** When changing parameters, set the parameters using the following buffer memory.

- 'Reinitialization specification change target' (Un\G1950043 to Un\G1950044)
- 'Reinitialization Own IP Address' (Un\G1950045 to Un\G1950048)
- 'Communication condition setting (Ethernet Operation Setting) area' (Un\G1950049 to Un\G1950050)

**4.** Perform re-initial processing.

Write 1 (Reinitialization process request) to 'Reinitialization specification' (Un\G1950040).

If any parameter set in step 3 is outside the setting range, the following conditions occur, and re-initial processing is not performed.

- The corresponding bits of 'Reinitialization specification error factor' (Un\G1950041 to Un\G1950042) turn on for the error cause. (☞ Page 526 Reinitialization specification error factor (Un\G1950041 to Un\G1950042))
- 'Reinitialization specification' (Un\G1950040) becomes 0 (Reinitialization process complete).

**5.** When the system starts re-initial processing, bit 0 of 'Initial status' (Un\G1900024) turns off.

**6.** When the re-initial processing is all completed, 'Reinitialization specification' (Un\G1950040) becomes 0 (Reinitialization process complete), and bit 0 of 'Initial status' (Un\G1900024) turns on (Initialization normal completion).

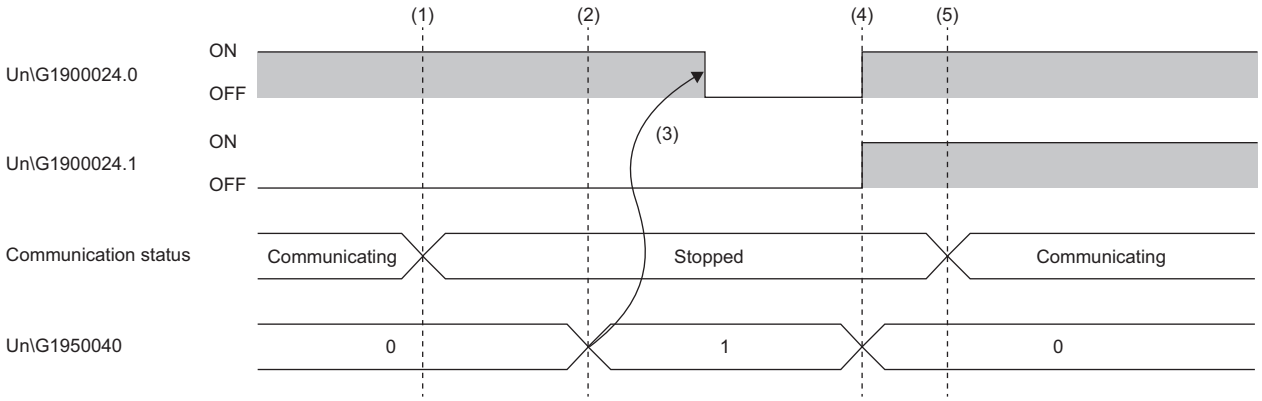
If the processing is completed with an error, bit 1 of 'Initial status' (Un\G1900024) turns on (Initialization abnormal completion), and the error code is stored in 'Initial error code' (Un\G1900025).

**7.** After the re-initial processing is completed, resume data communication with the external device using the following method.

| Communication method | Resumption method |
|---|---|
| Communication using the fixed buffer | After the re-initial processing is completed, open the connection again. |
| Socket communication | |
| Communication using predefined protocol (including the MODBUS/TCP master function) | |
| Communication using the random access buffer | |
| SLMP communication | |
| Simple CPU communication (including the MODBUS/TCP master function) | Turn on the corresponding bits of 'Request to restart fixed interval communication for each setting number' (Un\G721904 to Un\G721907, Un\G1247356 to Un\G1247383). |
| MELSOFT connection | The external device reopens the connection to resume data communication. |
| MODBUS/TCP slave function | |
| FTP | |
| SNTP | No operation is required because the system performs open processing. |

## Timing chart

The following figure shows the timing of re-initial processing.




(1) All data communication with the external device is terminated.
(2) 1 (Reinitialization process request) is written in 'Reinitialization specification' (Un\G1950040) to perform re-initial processing.
(3) When re-initial processing starts, bit 0 of 'Initial status' (Un\G1900024) turns off.
(4) When the re-initial processing is completed, 'Reinitialization specification' (Un\G1950040) becomes 0 (Reinitialization process complete), and bit 0 of 'Initial status' (Un\G1900024) turns on (Initialization normal completion). When the re-initial processing is completed with an error, bit 1 of 'Initial status' (Un\G1900024) turns on (Initialization abnormal completion).
(5) Data communication with the external device is resumed.

## Precautions

- If data communication with the external device is performed using SNTP, the re-initial processing may temporarily interrupt data communication, causing a communication error.
- Re-initial processing cannot be performed while a major error is occurring.

A

# Appendix 6 TCP/IP Communications, UDP/IP Communications

This chapter describes the communication flow and procedure of TCP/IP communications and UDP/IP communications.

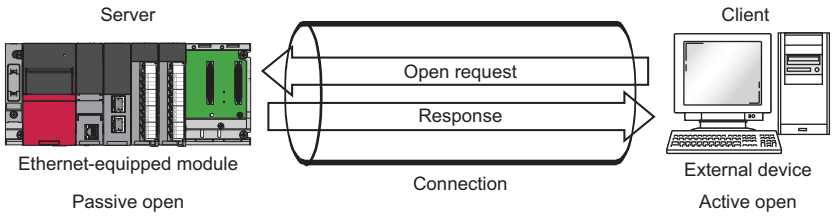
## TCP/IP communications

This section describes TCP/IP communications.

### Establishing a connection

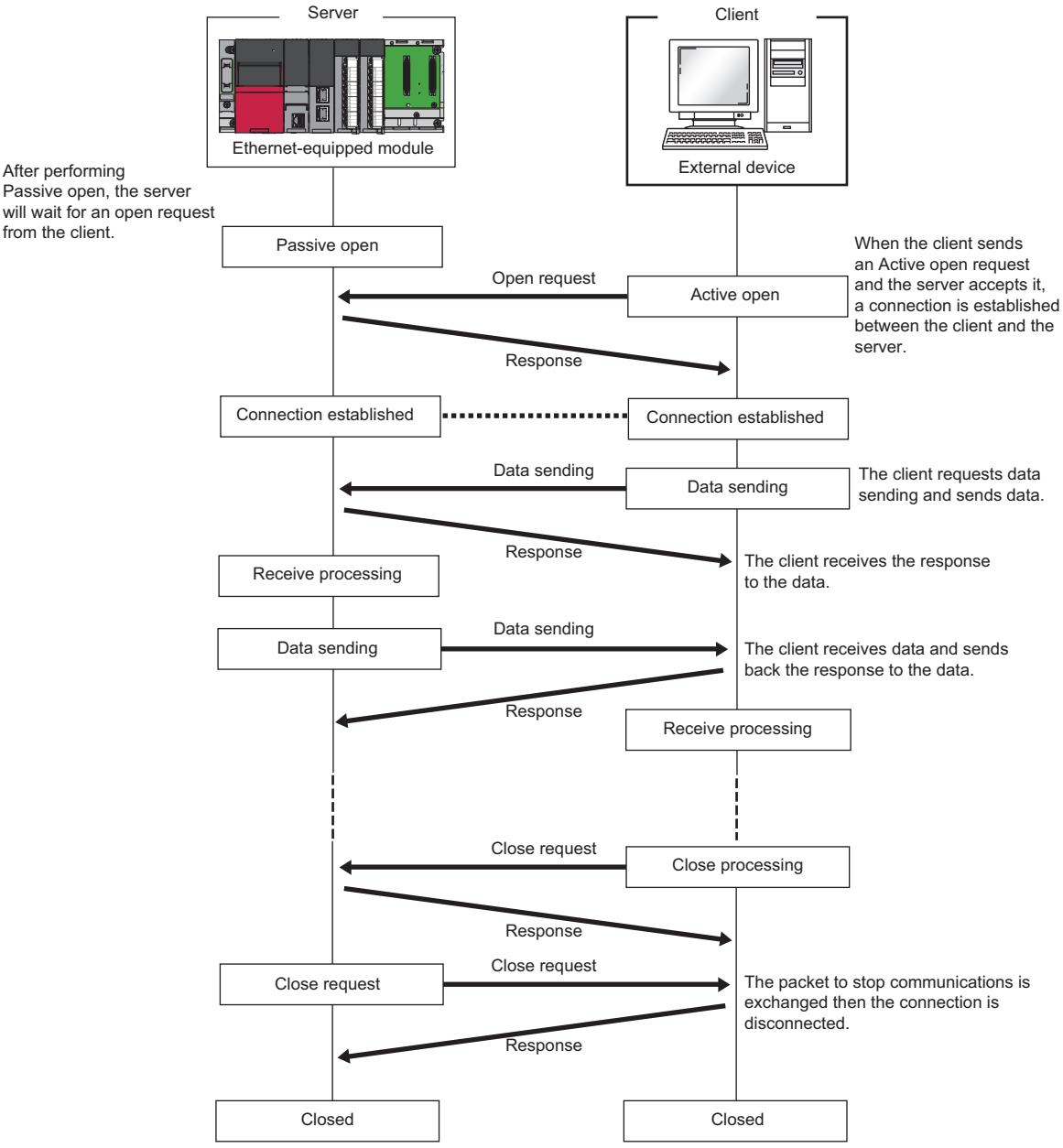
With TCP/IP communications, a connection must be established between the communicating devices. If the server side device has executed the Passive open processing and is in the standby state, the client side device makes an open request (Active open processing) to the server. When a response is returned, the connection is established.
With TCP/IP communications, a connection is established during communication. Since data is exchanged while checking that the data has correctly reached the communication destination, the data reliability can be ensured. Note that the line load is larger than UDP/IP communications.

Ex.
When the Ethernet-equipped module is Passive open



## Communication flow

This section describes the flow from the establishment of connection to end of communication.




**Point**

Wait at least 500ms or more before executing the open processing again after the close request is sent from the external device to the Ethernet-equipped module.

## Active open procedure

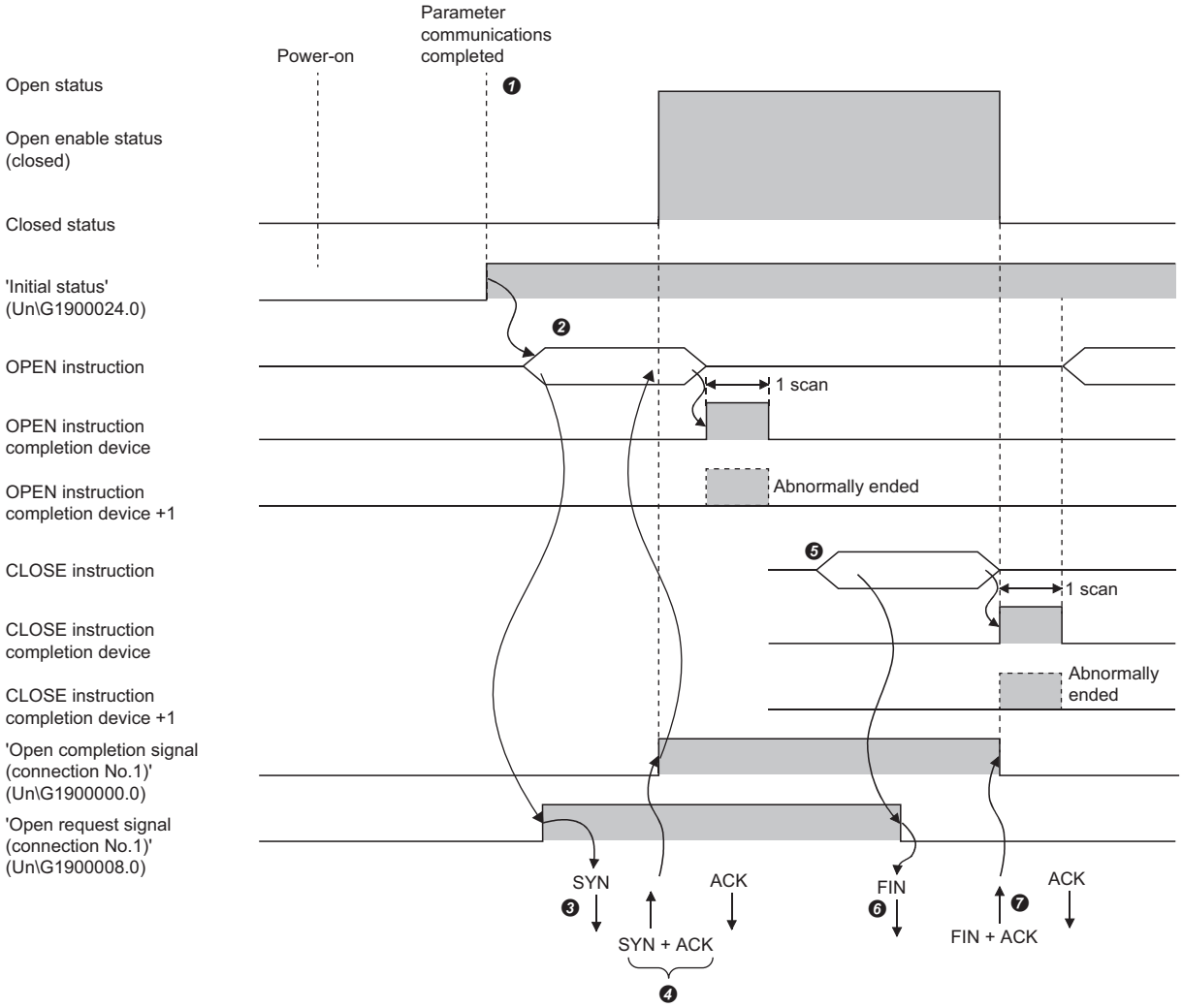
Active open is a connection method that performs an active open processing in respect to an external device (Passive open) that is in a passive open standby state for a connection. The following figure shows the process for the Ethernet-equipped module to Active open.

For OPEN/CLOSE instruction, refer to the following.

MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

**Ex.**

Open/close processing for connection No.1




❶ After the module parameters are set, check that the initial processing for the Ethernet-equipped module has completed normally. ('Initial status' (Un\G1900024.0): On)

❷ Start the open processing using the OPEN instruction.*3 ('Open request signal (connection No.1)' (Un\G1900008.0): On)

❸ The Ethernet-equipped module executes the open processing. (The module sends open request (SYN) to the external device.)

❹ Data can be exchanged after the open processing completes normally.*1

❺ Start the close processing using the CLOSE instruction. ('Open request signal (connection No.1)' (Un\G1900008.0): Off)

❻ The Ethernet-equipped module executes the close processing. (The module sends close request (FIN) to external device.)

❼ Data communication ends when close processing completes normally.*2

*1 If RST is returned from the external device after SYN is sent from the Ethernet-equipped module, open abnormal completion occurs immediately, and the open processing ends.

*2 If ACK or FIN is not returned even after the TCP end timer time, the Ethernet-equipped module forcibly cuts off the connection (sends RST). (Close abnormal completion)

*3 If the open processing target port has not been linked up, the OPEN instruction will complete with an error. Execute the open processing again after link-up, or check that 'Connection status' (Un\G5192) is set to 1 before starting the open processing. If auto-negotiation fails, the open processing will complete with an error. Retry the open processing after a while.
If the open processing completes with an error again, check the Ethernet cable connection or the operation of the external device and switching hub.

## Passive open procedure

The following two types of connection methods can be used to Passive open the Ethernet-equipped module.

| Connection method | Description |
|---|---|
| Unpassive | This connection method executes a passive open processing for the connection to all devices connected to the network without restriction to the IP address or port number of the communication destination. |
| Fullpassive | When the IP address and port number of the communication destination are specified, this connection method executes a passive open processing for the connection of the specific external device. |

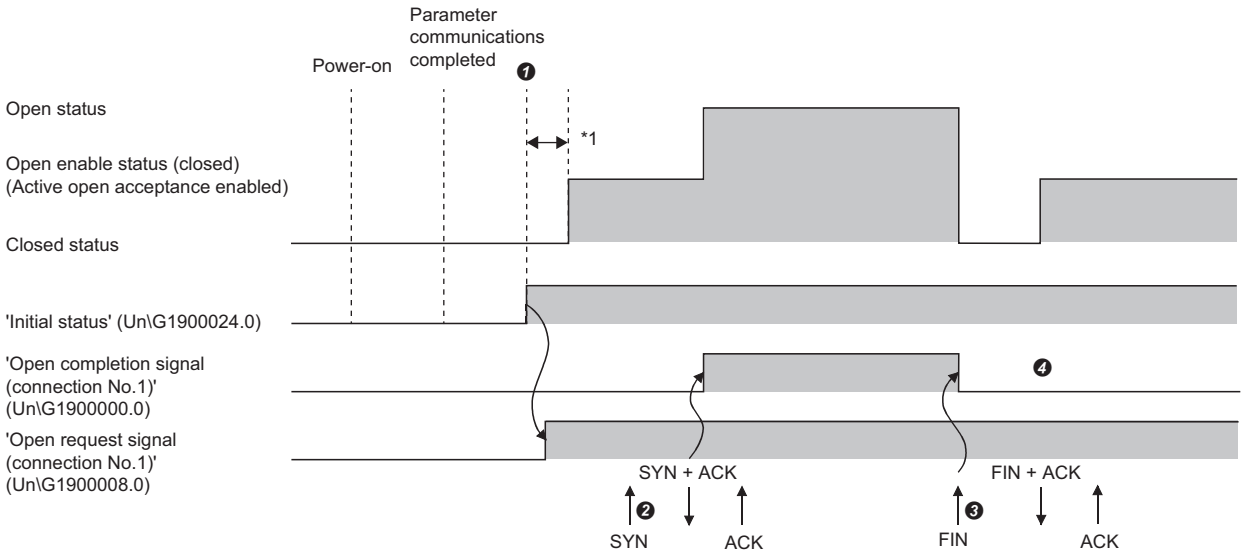
The open/close processing procedure for Passive open follows the setting of "Opening Method" under "Own Node Settings" in "Basic Settings". (☞ Page 327 Own Node Settings)

### ■When "Do Not Open by Program" is set

The Ethernet-equipped module is constantly in the open standby state, so the connection is established when Active open is initiated by the external device. This eliminates the need for an open/close processing program on the Ethernet-equipped module side.

Ex.

Open/close processing for connection No.1




❶ After the module parameters are set, check that the initial processing for the Ethernet-equipped module has completed normally. ('Initial status' (Un\G1900024.0): On) When the initial processing completes normally, the connection enters the open enable state, and the module waits for the open request from the external device.

❷ The Ethernet-equipped module executes the open processing when an open request (SYN) is received from the external device. When the open processing ends normally, 'Open completion signal (connection No.1)' (Un\G1900000.0) turns on and data communication is enabled.

❸ The Ethernet-equipped module executes the close processing when the close request (FIN) is received from the external device. When the close processing completes normally, the open completion signal turns off and data communication is disabled.

❹ After the internal processing in the Ethernet-equipped module completes, the connection stands by for the open request again.

*1 The open request (SYN) received between the initial processing normal completion to the open request standby state is handled as an error, and the Ethernet-equipped module sends a connection forced close (RST) (to the external device that sent the open request (SYN)).

**Point**

When the open/close processing is executed with a dedicated instruction from the Ethernet-equipped module, even if "Do Not Open by Program" is set in "Opening Method" under "Own Node Settings" in "Basic Settings", the connection will not return to the open request standby state after the close processing completes.

## ■When "Open by Program" is set

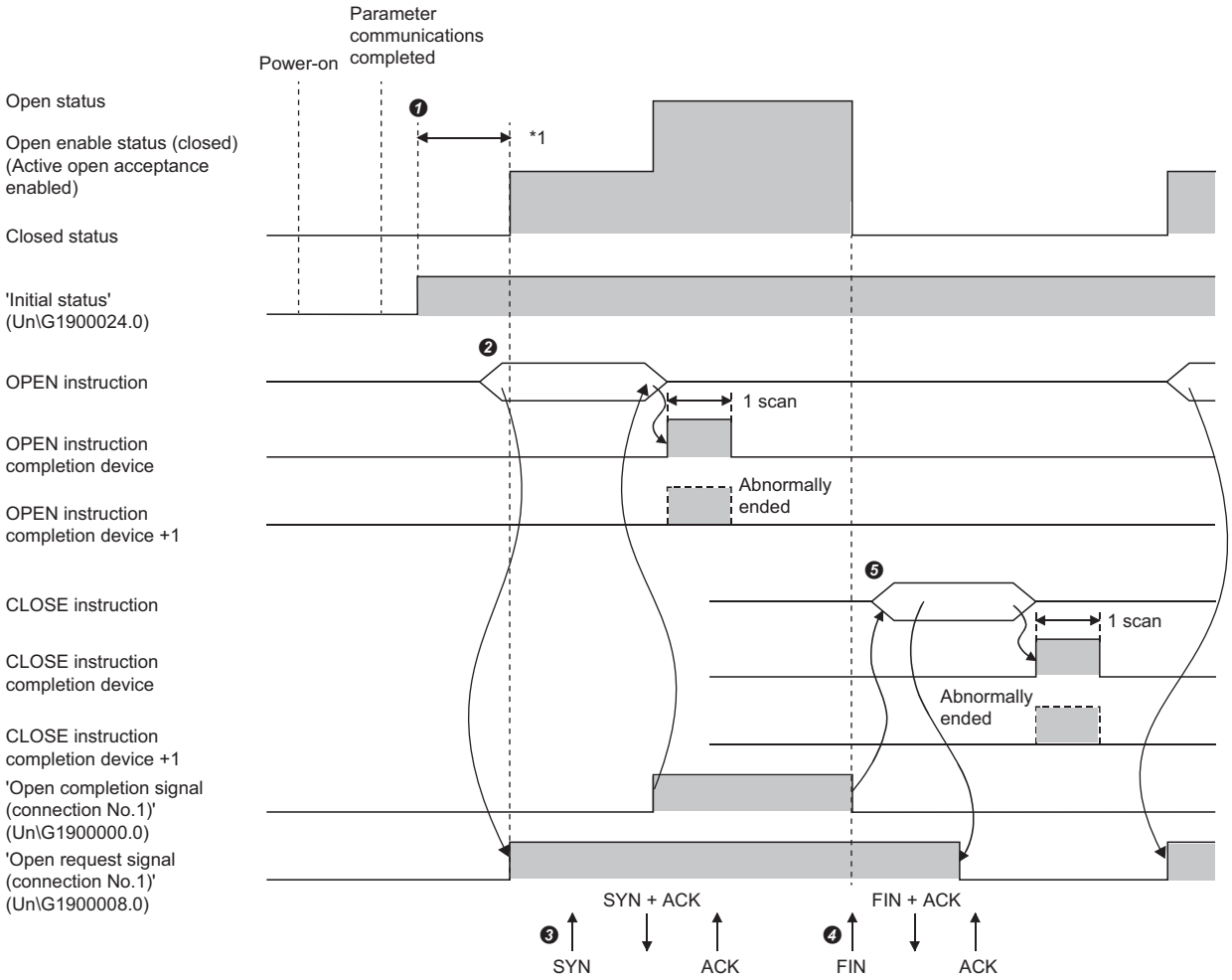
Before the open/close request is received from the external device, the Ethernet-equipped module must execute the OPEN/CLOSE instruction and enter the open/close standby state. Data can be sent and received after the open processing completes normally.

For OPEN/CLOSE instruction, refer to the following.

MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

Ex.

Open/close processing for connection No.1




❶ After the module parameters are set, check that the initial processing for the Ethernet-equipped module has completed normally. ('Initial status' (Un\G1900024.0): On)

❷ Start the open processing using the OPEN instruction. ('Open request signal (connection No.1)' (Un\G1900008.0): On)

❸ The Ethernet-equipped module executes the open processing when an open request (SYN) is received from the external device. When the open processing ends normally, 'Open completion signal (connection No.1)' (Un\G1900000.0) turns on and data communication is enabled.

❹ The Ethernet-equipped module executes the close processing when the close request (FIN) is received from the external device. When the close processing completes normally, the open completion signal turns off and data communication is disabled.

*1 The open request (SYN) received between the initial processing normal completion to the open request standby state is handled as an error, and the Ethernet-equipped module sends a connection forced close (RST) (to the external device that sent the open request (SYN)).

**Point**

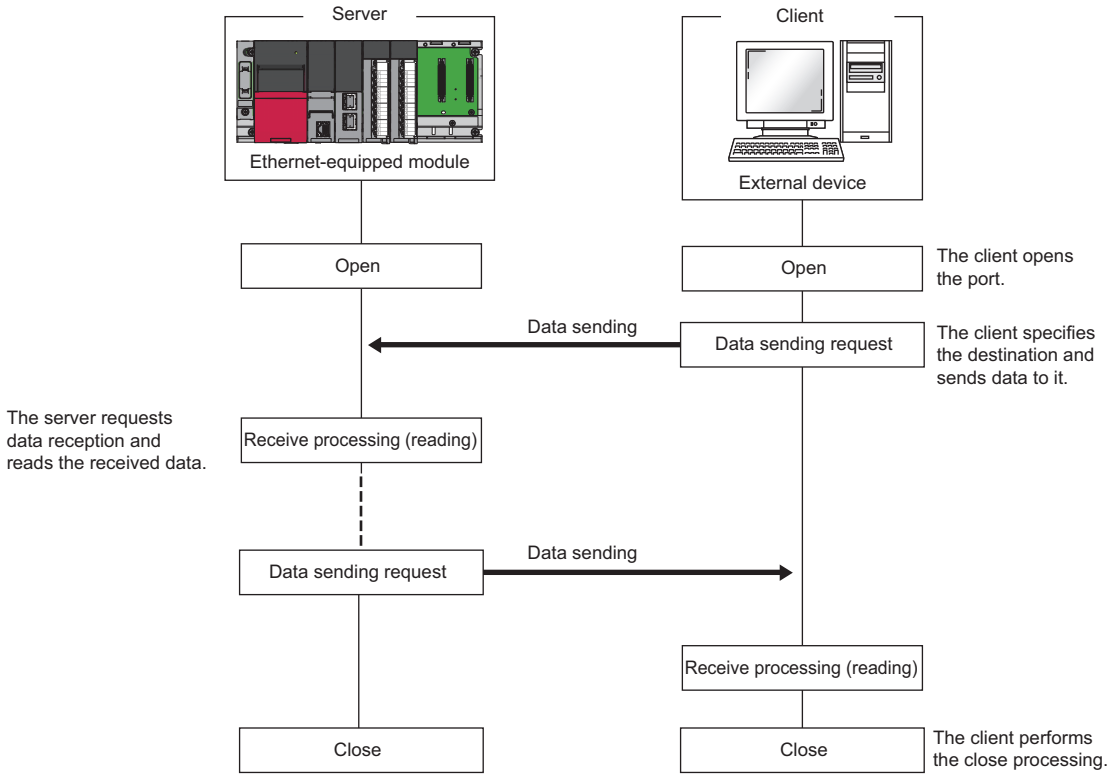
- Change the connection setting before executing the OPEN instruction.
- After the open processing is executed, the open request cannot be canceled until the open processing completes. Execute the close processing (CLOSE instruction) after open completes.

# UDP/IP communications

This section describes the UDP/IP communications. Since UDP/IP communications does not establish a connection during communication and does not check that the communication destination has correctly received the data, the line load is lower. Note that the data reliability is lower than TCP/IP communications.

## Communication flow

UDP/IP communications does not require a process to establish a connection with the external device as is required with TCP/IP communications.




**Point**

Wait at least 500ms or more before executing the open processing again after the close request is sent from the external device to the Ethernet-equipped module.

## Open procedure

The open/close processing procedure is as follows, according to the setting of "Opening Method" under "Own Node Settings" in "Basic Settings". (☞ Page 327 Own Node Settings)

### ■When "Do Not Open by Program" is set

After the Ethernet-equipped module mounted station starts up, the UDP/IP communications setting connection automatically opens, and data send/receive is enabled. Program for open/close processing is not required.

**Point**

When the open/close processing is executed with a dedicated instruction from the Ethernet-equipped module, even if "Do Not Open by Program" is set in "Opening Method" under "Own Node Settings" in "Basic Settings", the open/close processing after a connection with the external device is established must be executed by the program.

## ■When "Open by Program" is set

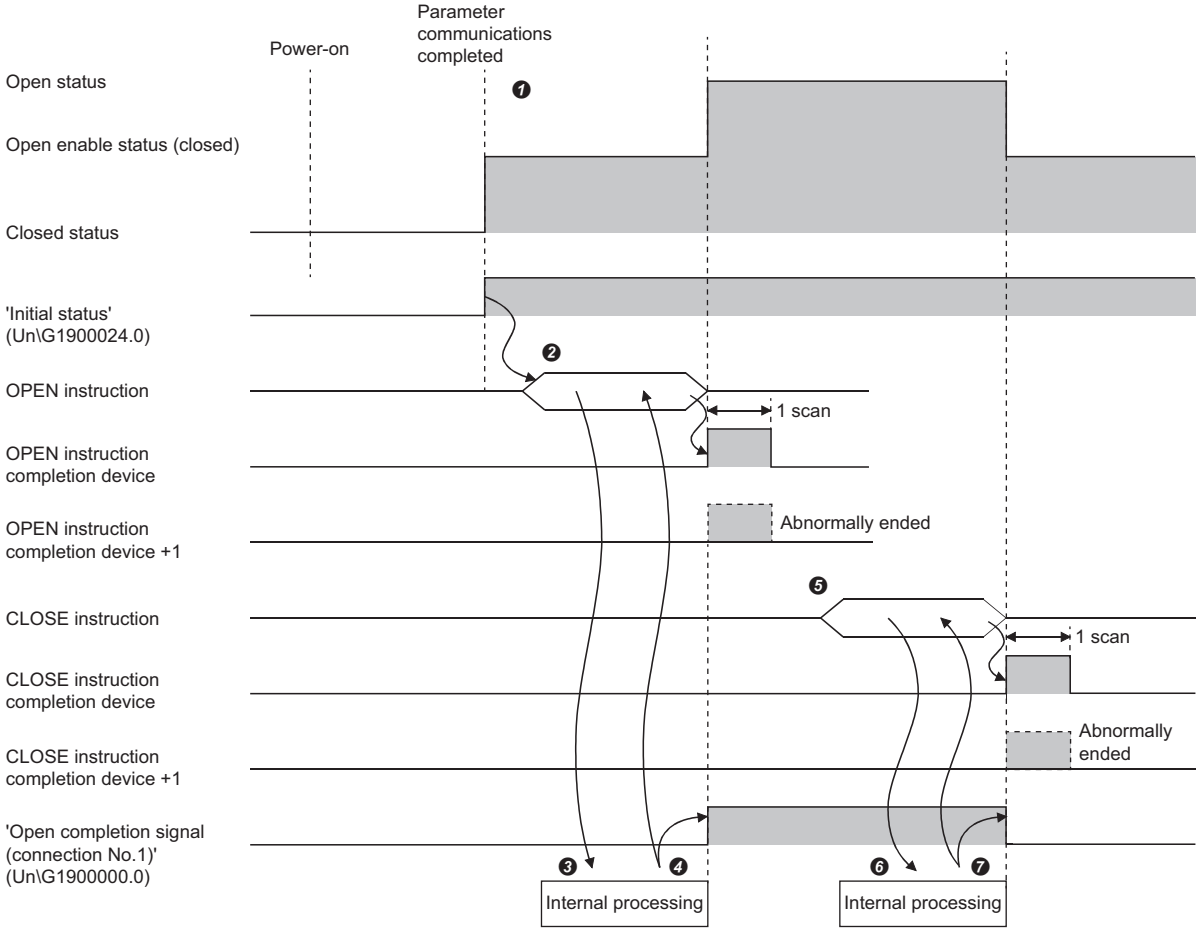
Before the open/close request is received from the external device, the Ethernet-equipped module must execute the OPEN/CLOSE instruction and enter the open/close standby state. Data can be sent and received after the open processing completes normally.

For OPEN/CLOSE instruction, refer to the following.

MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

Ex.

Open/close processing for connection No.1




❶ After the module parameters are set, check that the initial processing for the Ethernet-equipped module has completed normally. ('Initial status' (Un\G1900024.0): On)

❷ Start the open processing using the OPEN instruction. ('Open request signal (connection No.1)' (Un\G1900008.0): On)

❸ The Ethernet-equipped module executes the open processing. (only internal processing)

❹ Data can be exchanged after the open processing completes normally.

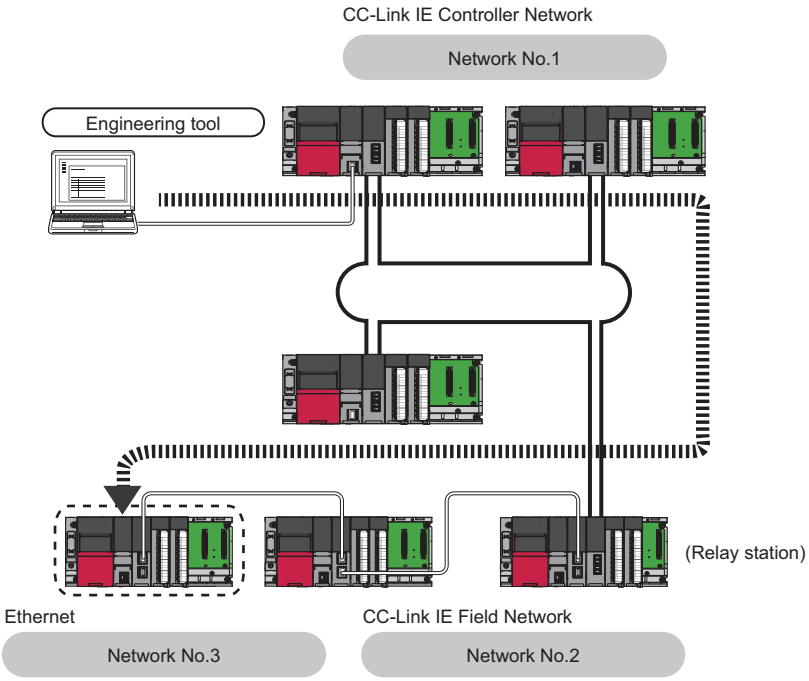
❺ Start the close processing using the CLOSE instruction. ('Open request signal (connection No.1)' (Un\G1900008.0): Off)

❻ The Ethernet-equipped module executes the close processing. (only internal processing)

❼ Data communication ends when close processing completes normally.

# Appendix 7 Communications with Different Networks

Different network stations can be accessed with dedicated instructions or the engineering tool, allowing seamless communication.




The following functions are used to communicate with different networks.

| Functions capable of communication | Communication request source | Reference source |
|---|---|---|
| Communications using the SLMP | External device | ☞ Page 31 SLMP Communications<br>🕮 SLMP Reference Manual |
| Communications using a link dedicated instruction | CPU module on another station | ☞ Page 118 Link Dedicated Instruction Communication |
| Communications with other stations using the engineering tool | Engineering tool | 🕮 GX Works3 Operating Manual |

**Point**

- UDP/IP communications is used to communicate with other networks. Data is always exchanged as binary codes.
- Communications can be made with stations up to eight networks apart (number of relay stations: 7).
- When the network type for the RJ71EN71 is set to "Q Compatible Ethernet", refer to the following. (☞ Page 554 When the networks consist of MELSEC iQ-R series and other series)

A

## When the networks consist of only MELSEC iQ-R series

Communication paths are automatically set for communication with the following networks of MELSEC iQ-R series.

- Ethernet
- CC-Link IE TSN
- CC-Link IE Controller Network
- CC-Link IE Field Network

### ■Setting procedure

Use the following procedure.

1. Set the network number, station number, and transient transmission group number in the "Own Node Settings" under "Basic Settings". (☞ Page 328 Details of items)
2. Set the "Network Dynamic Routing" under "Application Settings" to "Enable".

**Point**

- Communication paths are automatically set, but they can also be manually set. To set communication paths manually, refer to the following. (☞ Page 554 When the networks consist of MELSEC iQ-R series and other series)
- The communication path cannot be set automatically for Ethernet-equipped modules connected via a router. Set the communication path manually. (☞ Page 554 When the networks consist of MELSEC iQ-R series and other series)
- After all the Ethernet-equipped module mounted stations on the path are powered on or reset, determination of the communication path may take the amount of time set in "Timer Settings for Data Communication" under "Application Settings" of the Ethernet-equipped modules on the path.

## When the networks consist of MELSEC iQ-R series and other series

Setting communication paths allows communication with the following networks configured with modules other than MELSEC iQ-R series.

- Ethernet
- CC-Link IE Controller Network
- CC-Link IE Field Network
- MELSECNET/H
- MELSECNET/10

### ■Setting procedure

Use the following procedure.

1. Set the network number, station number, and transient transmission group number in the "Own Node Settings" under "Basic Settings". (☞ Page 328 Details of items)
2. Set the send destination station information in "Network/Station No. <-> IP information setting" under "Application Settings". (☞ Page 407 Network/Station No. <-> IP information setting)
3. Set communication paths in "Routing Setting" of "CPU Parameter". (📖 Manual for the CPU module used)

## When the networks contain a redundant system

- When setting a communication path automatically, after the time specified by the destination alive check timer (time calculated from destination alive check start interval timer value, destination alive check interval timer value, and destination alive check resend count), the communication path setting will be changed. Set as small a value as possible for the "Destination Alive Check Start Interval Timer", "Destination Alive Check Interval Timer", and "Destination Alive Check Resend Count".
- When the communication paths are automatically set, the paths are updated if system switching occurs in a redundant system on the paths. During update of the communication paths, communications may be interrupted or data may be lost because the paths are temporarily undetermined. If an error occurs in communications, resend data from the request source.
- When the request source is a redundant system and the communication paths are automatically set, perform the communications from the control system. When performing the communications from the standby system, set "Routing Setting" of "CPU Parameter"
- When the request destination is a redundant system and the communication paths are automatically set, specify the control system station as the destination. When specifying the standby system as the request destination, set "Routing Setting" of "CPU Parameter".
- When relaying a redundant system and setting the communication paths manually, set the control system station as a relay station in "Routing Setting" of "CPU Parameter". If system switching occurs, the relay station needs to be changed to a station in the new control system using the RTWRITE instruction. (MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks))

# Appendix 8 Processing Time

Calculate the minimum processing time for each function with the following formula. Note that the processing time may be even longer due to the network load rate (line congestion), each connected device's window size, number of connections being used simultaneously, and the system configuration. The value calculated with the following formula is a guide for the processing time when communicating with only one connection.

## Minimum processing time of fixed buffer communications (between the RJ71EN71s*1)

*1 The processing time will be the same even if the RnENCPU (network part) is mixed.

### ■Procedure exists

Tfs = St + Ke + (Kdf × Df) + Sr

- Tfs: Time from start to end of send (unit: ms)
- St: Sending station scan time
- Ke, Kdf: Constant (refer to following table)
- Df: Number of send data words
- Sr: Receiving station scan time

| Item | RJ71EN71, RnENCPU | | | |
|---|---|---|---|---|
| | During TCP/IP communications | | During UDP/IP communications | |
| | Ke | Kdf | Ke | Kdf |
| When communicating with binary code data | 6 | 0.0030 | 3 | 0.0020 |
| When communicating with ASCII code data | 6 | 0.0100 | 4 | 0.0015 |

### ■No procedure

Tfs = St + Ke + (Kdf × Df)

- Tfs: Time from start to end of send (unit: ms)
- St: Sending station scan time
- Ke, Kdf: Constant (refer to following table)
- Df: Number of send data bytes

| Item | RJ71EN71, RnENCPU | | | |
|---|---|---|---|---|
| | During TCP/IP communications | | During UDP/IP communications | |
| | Ke | Kdf | Ke | Kdf |
| When communicating with binary code data | 4 | 0.0010 | 3 | 0.0007 |

## Minimum processing time of communications using the random access buffer

Trs = Kr + (Kdr × Df) + external device ACK processing time (added only during TCP/IP communications)

- Trs: Time from reception of personal computer request data to complete of process by RJ71EN71 (unit: ms)
- Kr, Kdr: Constant (refer to following table)
- Df: Number of request data words
- External device ACK process time: Time until external device returns ACK upon completion of random access buffer read/write

| Item | | RJ71EN71, RnENCPU | | | |
|---|---|---|---|---|---|
| | | During TCP/IP communications | | During UDP/IP communications | |
| | | Kr | Kdr | Kr | Kdr |
| During read | When communicating with binary code data | 2.5 | 0.0020 | 1.8 | 0.0025 |
| | When communicating with ASCII code data | 2.5 | 0.0060 | 1.9 | 0.0065 |
| During write | When communicating with binary code data | 2.5 | 0.0025 | 1.8 | 0.0025 |
| | When communicating with ASCII code data | 2.6 | 0.0070 | 1.9 | 0.0060 |

## Processing time of communications using MODBUS/TCP

The following shows the time required by the Ethernet-equipped module when it receives a request message from the MODBUS/TCP master device, completes the processing, and sends a response message.

### ■Calculation formula

Tsl = Ks1 + (Ks2 × n1) + (St × n2) [ms]

<table>
<tr><th>Variable</th><th>Meaning</th><th colspan="4">Processing time, constant</th></tr>
<tr><td>Tsl</td><td>Request message processing time (performance of the automatic response function) (unit: ms)</td><td colspan="4">—</td></tr>
<tr><td>Ks1</td><td>Internal processing time of the automatic response function (unit: ms)</td><td colspan="4">Fixed to 2.5ms</td></tr>
<tr><td>Ks2</td><td>Communication time between the Ethernet-equipped module and CPU module (unit: ms)</td><td colspan="4">Fixed to 4.5ms</td></tr>
<tr><td>St</td><td>Own station scan time (unit: ms)</td><td colspan="4">—</td></tr>
<tr><td rowspan="13">n1</td><td rowspan="13">Function code for the processing target and setting details of the memory assignment</td><td>Function code</td><td colspan="2">CPU module device assignment</td><td>Buffer memory assignment</td></tr>
<tr><td>01</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>02</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>03</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>04</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>05</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>06</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>15</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>16</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>20</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>21</td><td colspan="2">1</td><td>0</td></tr>
<tr><td>22</td><td colspan="2">2</td><td>0</td></tr>
<tr><td>23</td><td colspan="2">2</td><td>0</td></tr>
<tr><td rowspan="14">n2</td><td rowspan="14">Function code for the processing target and setting details of the memory assignment</td><td rowspan="2">Function code</td><td colspan="2">CPU module device assignment</td><td rowspan="2">Buffer memory assignment</td></tr>
<tr><td>Normal</td><td>Worst case</td></tr>
<tr><td>01</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>02</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>03</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>04</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>05</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>06</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>15</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>16</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>20</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>21</td><td>1</td><td>2</td><td>0</td></tr>
<tr><td>22</td><td>2</td><td>4</td><td>0</td></tr>
<tr><td>23</td><td>2</td><td>4</td><td>0</td></tr>
</table>

## System switching time when a redundant system is used

The following shows the system switching time when the Ethernet-equipped module mounted on the control system in the redundant system detects a communication error or cable disconnection and sends a system switching request to the control system CPU module. System switching time is the time from communication error or cable disconnection detection to when the control system CPU module is switched over.

### ■Calculation formula

The following is the formula to calculate the system switching time.*1

| System switching cause | System switching time [ms] |
|---|---|
| Communication error detection (alive check error) | Tsi + (Ti × (Tr + 1)) + St + Tsw |
| Communication error detection (ULP timeout occurrence) | Ttu + St + Tsw |
| Disconnection detection | Td + St + Tsw |

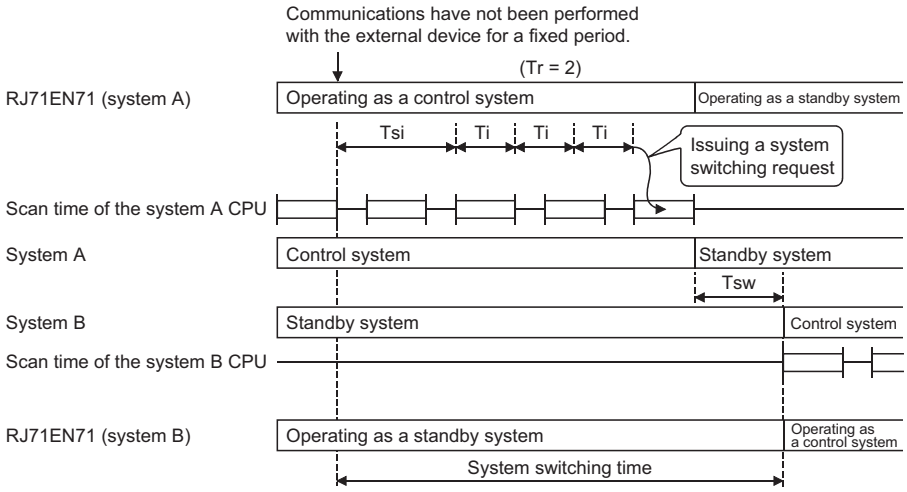
*1 The meanings of the variables in the equation are as follows.

| Variable | Meaning |
|---|---|
| Tsi | Destination alive check start interval timer [ms] (☞ Page 401 Timer Settings for Data Communication) |
| Ti | Destination alive check interval timer [ms] (☞ Page 401 Timer Settings for Data Communication) |
| Tr | Destination alive check resend count (☞ Page 401 Timer Settings for Data Communication) |
| St | 1 scan time [ms] (📖 Manual for the CPU module used) |
| Tsw | CPU module system switching time [ms] (📖 Manual for the CPU module used) |
| Ttu | TCP ULP timer value [ms] (☞ Page 401 Timer Settings for Data Communication) |
| Td | Disconnection detection monitoring time [ms] (☞ Page 416 Redundant System Settings) |

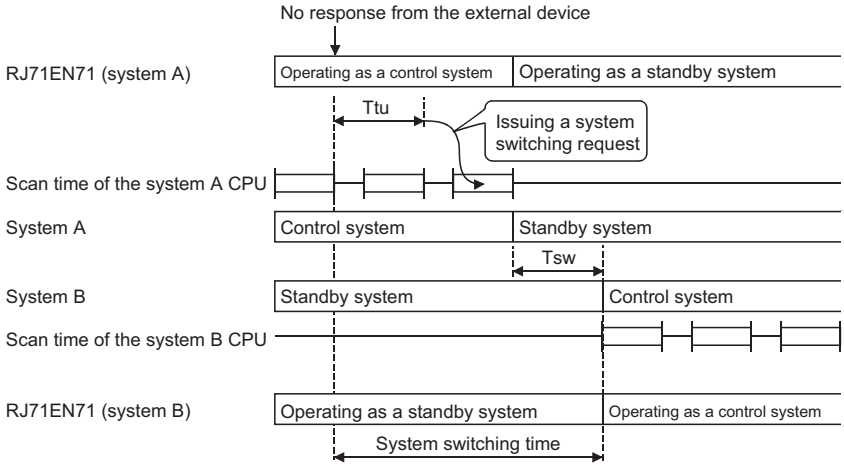
## ■Timing chart

This section describes a timing chart for each system switching cause.

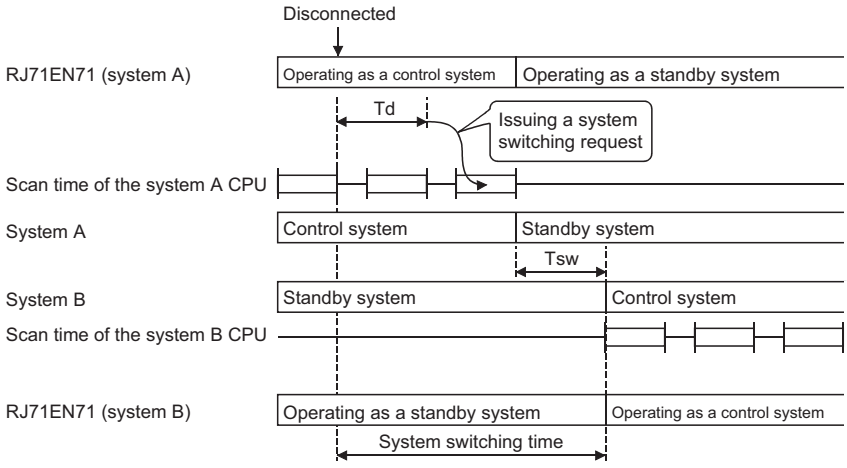
- Communication error detection (alive check error)




- Communication error detection (ULP timeout occurrence)




- Disconnection detection




A

# Performance list of simple CPU communication function

## Execution interval

The following shows the performance of the execution interval on the simple CPU communication function (CPU module (built-in Ethernet port part)). The execution interval varies depending on the number of settings, the number of communication points, and the scan time of the CPU module. The interval is also affected by the execution conditions of other functions or Ethernet communication condition.

### ■When "Communication Setting" is "Fixed Interval"

- Communication setting: "Fixed Interval"
- Sequence scan time (own station): 1ms (scan time before the execution of the simple CPU communication function)
- Communication destination: MELSEC iQ-R (built-in Ethernet)*1
- Device data: Bit device = M, Word device = D
- Assume the number of settings is as shown in the table and each setting is for an individual communication destination (not for the same destination). (Duplicated communication destinations may increase the execution interval. The more duplicates, the larger the delay in the execution interval, which is approximately eight times longer at maximum.)
- The retry of communication is not performed.

*1 The sequence scan time for the communication destination is 1ms (for any model).

| Communication pattern | Number of communication points for one setting | Processing time by the number of settings (unit: ms) | | | | |
|---|---|---|---|---|---|---|
| | | 1 | 8 | 16 | 32 | 64 |
| Read | 32 words for each device*2 (Total 64 words) | 10 | 11 | 17 | 34 | 76 |
| | 64 words for each device*2 (Total 128 words) | 10 | 11 | 17 | 35 | 77 |
| | 256 words for each device*2 (Total 512 words) | 10 | 13 | 20 | —*3 | —*3 |
| Write | 32 words for each device*2 (Total 64 words) | 10 | 11 | 18 | 35 | 77 |
| | 64 words for each device*2 (Total 128 words) | 10 | 11 | 18 | 35 | 80 |
| | 256 words for each device*2 (Total 512 words) | 10 | 13 | 20 | —*3 | —*3 |

*2 Number of points for each device (bit device and word device)

*3 Maximum number of points exceeded (maximum 8192 words)

## ■When "Communication Setting" is "On Request"

- Communication setting: "On Request"
- Sequence scan time (own station): 1ms (scan time before the execution of the simple CPU communication function)
- Communication destination: MELSEC iQ-R (built-in Ethernet)*1
- Device data: Bit device = M, Word device = D
- Assume the number of settings is as shown in the table and each setting is for an individual communication destination (not for the same destination). (Duplicated communication destinations may increase the execution interval. The more duplicates, the larger the delay in the execution interval, which is approximately eight times longer at maximum.)
- The retry of communication is not performed.

*1 The sequence scan time for the communication destination is 1ms (for any model).

| Communication pattern | Number of communication points for one setting | Processing time by the number of settings (unit: ms) | | | | | |
|---|---|---|---|---|---|---|---|
| | | 1 | | 4 | | 8 | |
| | | 1st time*3 | 2nd time or later*4 | 1st time*3 | 2nd time or later*4 | 1st time*3 | 2nd time or later*4 |
| Read | 32 words for each device*2 (Total 64 words) | 6 | 13 | 8 | 18 | 13 | 25 |
| | 64 words for each device*2 (Total 128 words) | 6 | 13 | 8 | 18 | 13 | 25 |
| | 256 words for each device*2 (Total 512 words) | 6 | 13 | 8 | 18 | 13 | 25 |
| Write | 32 words for each device*2 (Total 64 words) | 6 | 13 | 8 | 18 | 13 | 25 |
| | 64 words for each device*2 (Total 128 words) | 6 | 13 | 8 | 18 | 13 | 25 |
| | 256 words for each device*2 (Total 512 words) | 6 | 13 | 8 | 18 | 13 | 25 |

*2 Number of points for each device (bit device and word device)

*3 Applicable when the identifying conditions shown below are not satisfied

*4 Applicable when the identifying conditions shown below are satisfied

[Identifying conditions]

· The communication of the same setting number has been executed for the second time or more, and the previous communication has been completed successfully without an error.

· Among the settings for the same communication destination, any of the communications is set to "Fixed Interval", and it has been executed without an error. (Except for those in a stop state.)

· Among the settings for the same communication destination, any of the communications is set to "On Request" while communications set to "Fixed Interval" do not exist or have stopped. Also, the previous execution of another setting number where the communication setting is "On Request" has been completed successfully without an error.

# Appendix 9 Mounting the Module with a Remote Head Module

This chapter describes restrictions and precautions for using the RJ71EN71 that is mounted with a remote head module.

## Restrictions on functions and specifications

### Functions

| Function | Restriction |
|---|---|
| Communications using the SLMP | • The command that can be used differs from that of the CPU module. (📖 SLMP Reference Manual)<br>• The file that can be operated differs from that of the CPU module. (📖 MELSEC iQ-R CC-Link IE Field Network Remote Head Module User's Manual (Application)) |
| Communications using the predefined protocol | This function cannot be used. |
| Socket communications | This function cannot be used. |
| Communications using the fixed buffer | This function cannot be used. |
| Communications using MODBUS/TCP | • When the network type is "Q Compatible Ethernet", this function cannot be used.<br>• The default values set for the MODBUS device assignment parameters cannot be used for creating a project for the engineering tool using the remote head module. Clear the values first before starting the setting. |
| Communications using a link dedicated instruction | Although the own station can be specified as the relay station or target station, the link dedicated instruction cannot be executed from the own station. |
| File transfer function (FTP server) | • The file that can be operated differs from that of the CPU module. (📖 MELSEC iQ-R CC-Link IE Field Network Remote Head Module User's Manual (Application))<br>• The number of files that can be written to the data memory differs from that of the CPU module. (📖 MELSEC iQ-R CC-Link IE Field Network Remote Head Module User's Manual (Startup)) |
| Time setting function (SNTP client) | This function cannot be used. |
| IP filter function | When the network type is "Q Compatible Ethernet", this function cannot be used. |
| Simple CPU communication function | When the network type is "Q Compatible Ethernet", this function cannot be used. |
| Simple device communication function | This function cannot be used. |
| IP address change function | When the network type is "Q Compatible Ethernet", this function cannot be used. |
| Redundant system function | Although the RJ71EN71 can be connected to a redundant system of a remote head module, redundant system of the RJ71EN71 cannot be configured. |

### Module parameters

There are differences between when the RJ71EN71 is connected to the CPU module and when the RJ71EN71 is connected to the remote head module. For details, refer to the following.

📖 MELSEC iQ-R CC-Link IE Field Network Remote Head Module User's Manual (Application)

This manual describes the restrictions specific to the RJ71EN71.

| Parameter name | | Restrictions |
|---|---|---|
| Basic Settings | Own Node Settings | Select "Do Not Open by Program" for "Opening Method". |
| | External Device Configuration | When performing TCP/IP communications, select "Unpassive Connection Module" or "Fullpassive Connection Module" for the external device. |

### Open/close processing

Perform the open/close processing of the connection from the external device.

# Appendix 10 Using the Module in the Redundant System with Redundant Extension Base Unit

This chapter describes restrictions and precautions for using the RJ71EN71 that is mounted on the extension base unit in the redundant system.

## Restrictions on functions and specifications

### Applicable network combination

| Network type | Setting in GX Works3 | Restrictions |
|---|---|---|
| Ethernet | RJ71EN71(E+E) | The network can be used for a module with firmware version of "50" or later.*1 |
| CC-Link IE Controller Network | RJ71EN71(CCIEC) | This network type cannot be used. |
| CC-Link IE Field Network | RJ71EN71(CCIEF) | This network type cannot be used. |
| Ethernet + CC-Link IE Controller Network | RJ71EN71(E+CCIEC) | This network type cannot be used. |
| Ethernet + CC-Link IE Field Network | RJ71EN71(E+CCIEF) | This network type cannot be used. |
| Q Compatible Ethernet | RJ71EN71(Q) | The network can be used for a module with firmware version of "50" or later.*1 |

*1 When using the network, check the versions of the CPU module compatible with the extension base unit and the engineering tool.

### Functions

| Function | Restriction |
|---|---|
| Connection with MELSOFT products and a GOT | When "Not Specified" is set, a CPU module of the control system is accessed.<br>When the CPU module in the system A is in the control system, enable communication by specifying "System A", "Control System", or "Not Specified" for "Redundant CPU Setting" in "Specify Connection Destination Connection".<br>When the CPU module in the system A is in the standby system, enable communication by specifying "System A" or "Standby System" for "Redundant CPU Setting" in "Specify Connection Destination Connection". |
| Communications using the predefined protocol | This function cannot be used. |
| Socket communications | This function cannot be used. |
| Communications using the fixed buffer | When the network type is only "Q Compatible Ethernet", this function can be used. (☞ Page 565 Example of communications using the fixed buffer) |
| MODBUS/TCP communications | When the network type is "Q Compatible Ethernet", this function cannot be used. |
| Communications using a link dedicated instruction | Although the own station can be specified as the relay station or target station, the link dedicated instruction cannot be executed from the own station. |
| File transfer function (FTP server) | This function cannot be used. |
| IP filter | When the network type is "Q Compatible Ethernet", this function cannot be used. |
| Simple CPU communication function | When the network type is "Q Compatible Ethernet", this function cannot be used. |
| Simple device communication function | This function cannot be used. |
| IP address change function | When the network type is "Q Compatible Ethernet", this function cannot be used. |

## Dedicated instructions

Any dedicated instructions of the RJ71EN71 cannot be used.

When using the dedicated instructions for the RJ71EN71, install it to the main base unit.

**Point**

Regarding turning off the ERR LED using the ERRCLEAR instruction, when the network type of the RJ71EN71 is one of those listed below, 'ERR LED off request' (Y17) can also be used instead. (☞ Page 482 I/O Signals)

- Ethernet*1
- Ethernet + CC-Link IE Controller Network*1
- Ethernet + CC-Link IE Field Network*1
- Q-compatible Ethernet

*1 'ERR LED off request' (Y17) is available for the RJ71EN71 with firmware version "90" or later.

## Module FBs

Any module FBs of the RJ71EN71 cannot be used.

## Module parameters

### ■Application Settings

- Set "Not Use" to "To Use or Not to Use Redundant System Settings" in "Redundant System Settings".
- Do not specify Interrupt settings.
- Set "Disable" to "Dynamic Routing Setting" under "Network Dynamic Routing".
- Set "Not Use" to "IP Packet Transfer Function" in "IP Packet Transfer Setting".

## Open/close processing

The open/close processing by input/output signals cannot be performed.

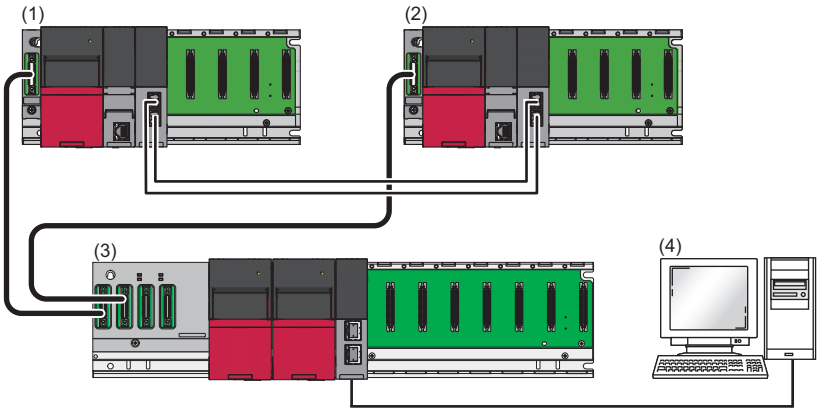
# Precautions

- When system switching occurs during communications using MELSOFT connection, SLMP communication from other stations, or communications using dedicated instructions of other stations, communication timeout error may occur.
- When system switching occurs during simple CPU communication, a timeout error may occur in the CPU response monitoring timer. If required, try the CPU response monitoring timer again.
- When system switching occurs during MODBUS/TCP communications, a timeout error may occur in the CPU response monitoring timer. If required, try the CPU response monitoring timer again.
- For replacement from the MELSEC-Q series, set "Q Compatible Ethernet" to the RJ71EN71 network type

# Example of communications using the fixed buffer

An example of communications using a fixed buffer when the RJ71EN71 is installed to the extension base unit is shown below.

## System configuration



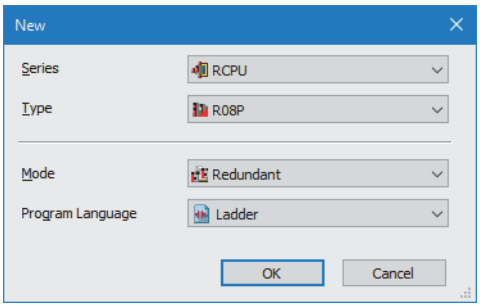

(1) Control system (R35B, R61P, R08PCPU, R6RFM)
(2) Standby system (R35B, R61P, R08PCPU, R6RFM)
(3) First level (R68WRB, R63RP, R63RP, RJ71EN71)
(4) Communication destination device

## Parameter setting

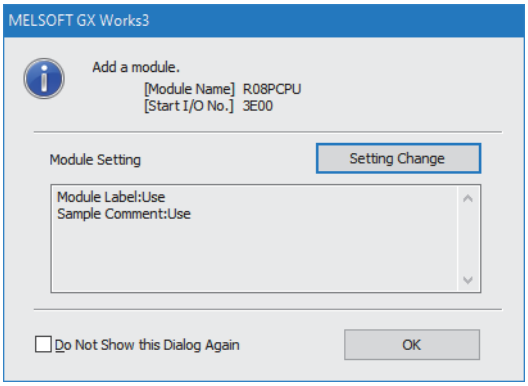
Connect the engineering tool to the CPU module to be set as the control system and set the parameters.
In a redundant system, the same program and parameter are written to both systems. A new project does not need to be created for the standby system.

**1.** Set the CPU module as follows.

[Project] ⇨ [New]



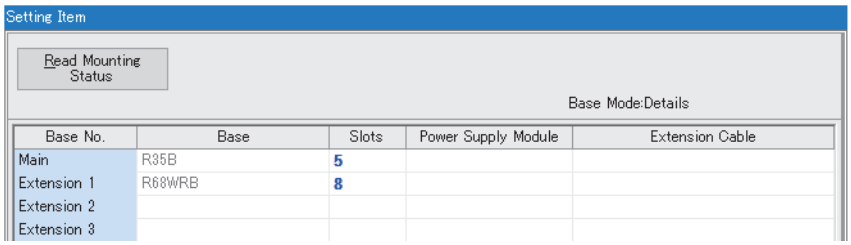

**2.** Click the [OK] button to add the module labels of the CPU module.




A

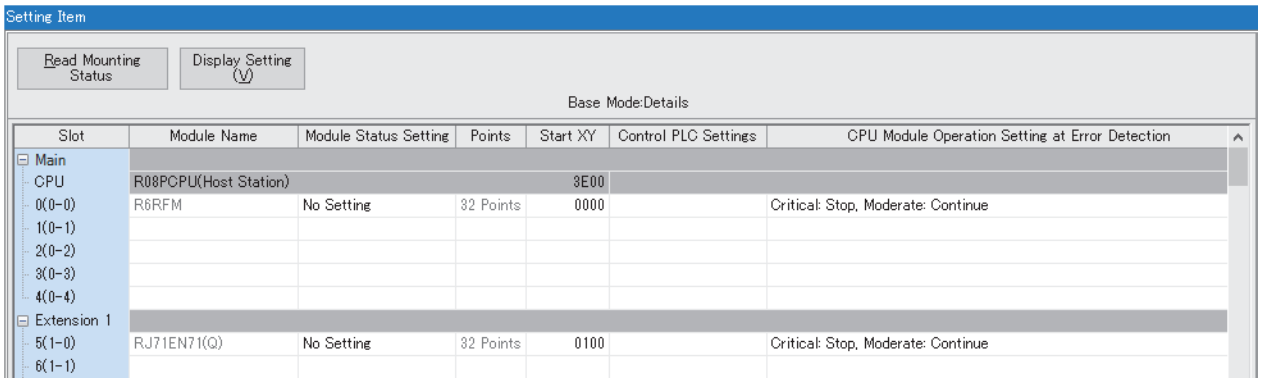
**3.** Set the base unit information as follows.

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [Base/Power/Extension Cable Setting]



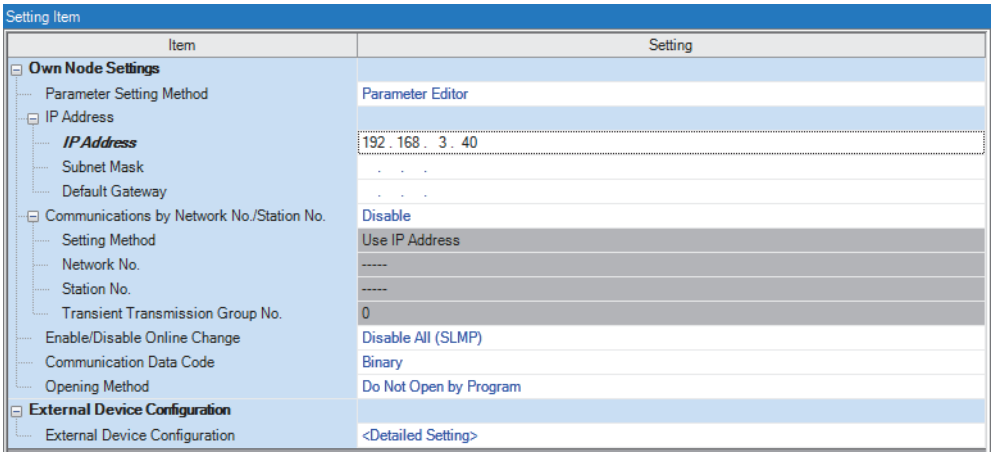

**4.** Specify the information for the redundant function module and RJ71EN71 as follows.

[Navigation window] ⇨ [Parameter] ⇨ [System Parameter] ⇨ [I/O Assignment Setting]



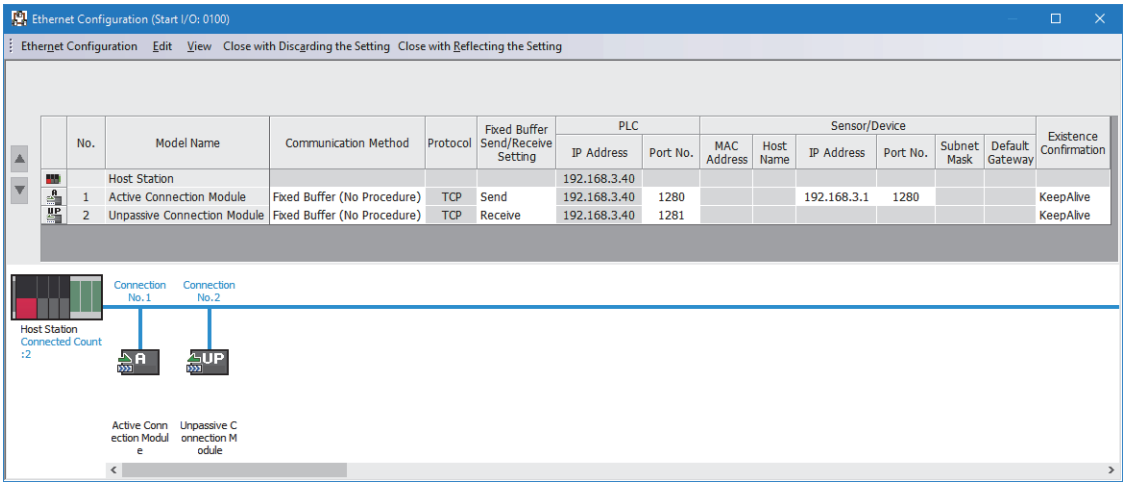

**5.** Set the IP address as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(Q)] ⇨ [Basic Settings] ⇨ [Own Node Settings]



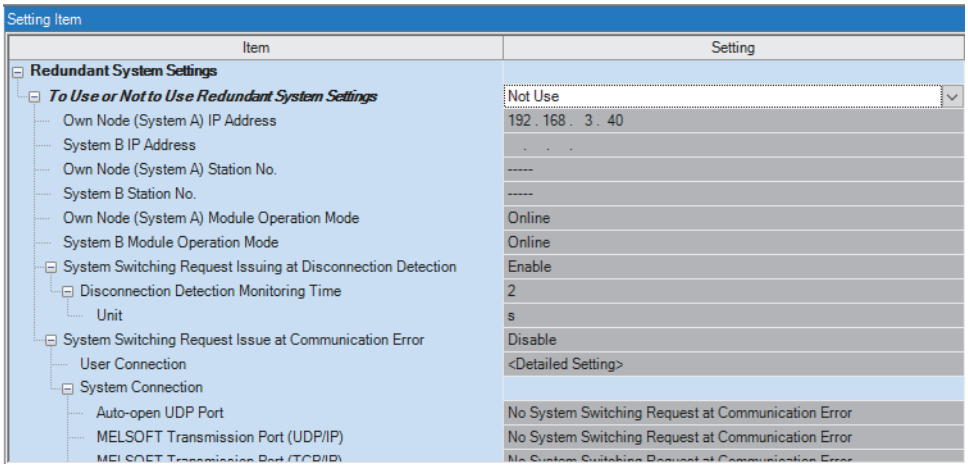
**6.** Set the network configuration as follows.

[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(Q)] ⇨ [Basic Settings] ⇨ [External Device Configuration]




**7.** Set "Not Use" to "To Use or Not to Use Redundant System Settings".

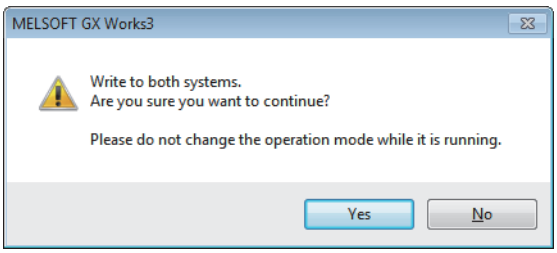
[Navigation window] ⇨ [Parameter] ⇨ [Module Information] ⇨ [RJ71EN71(Q)] ⇨ [Application Settings] ⇨ [Redundant System Settings]




**8.** Write the set parameters to the control system CPU module.

[Online] ⇨ [Write to PLC]

**9.** Click the [Yes] button to write to the CPU modules of both systems.




**10.** Either reset the CPU modules of both systems or power off and on both systems.

**Point**

- In this setting example, default settings other than above are used for the parameters. For the parameter setting, refer to the chapter explaining the parameters in this manual. (☞ Page 326 PARAMETER SETTINGS)
- Since PLS instructions and other instructions in the program, check that "Transfer" has been set to "Signal Flow Memory Tracking Setting" in "Tracking Setting" under "Redundant Settings" in "CPU Parameter".

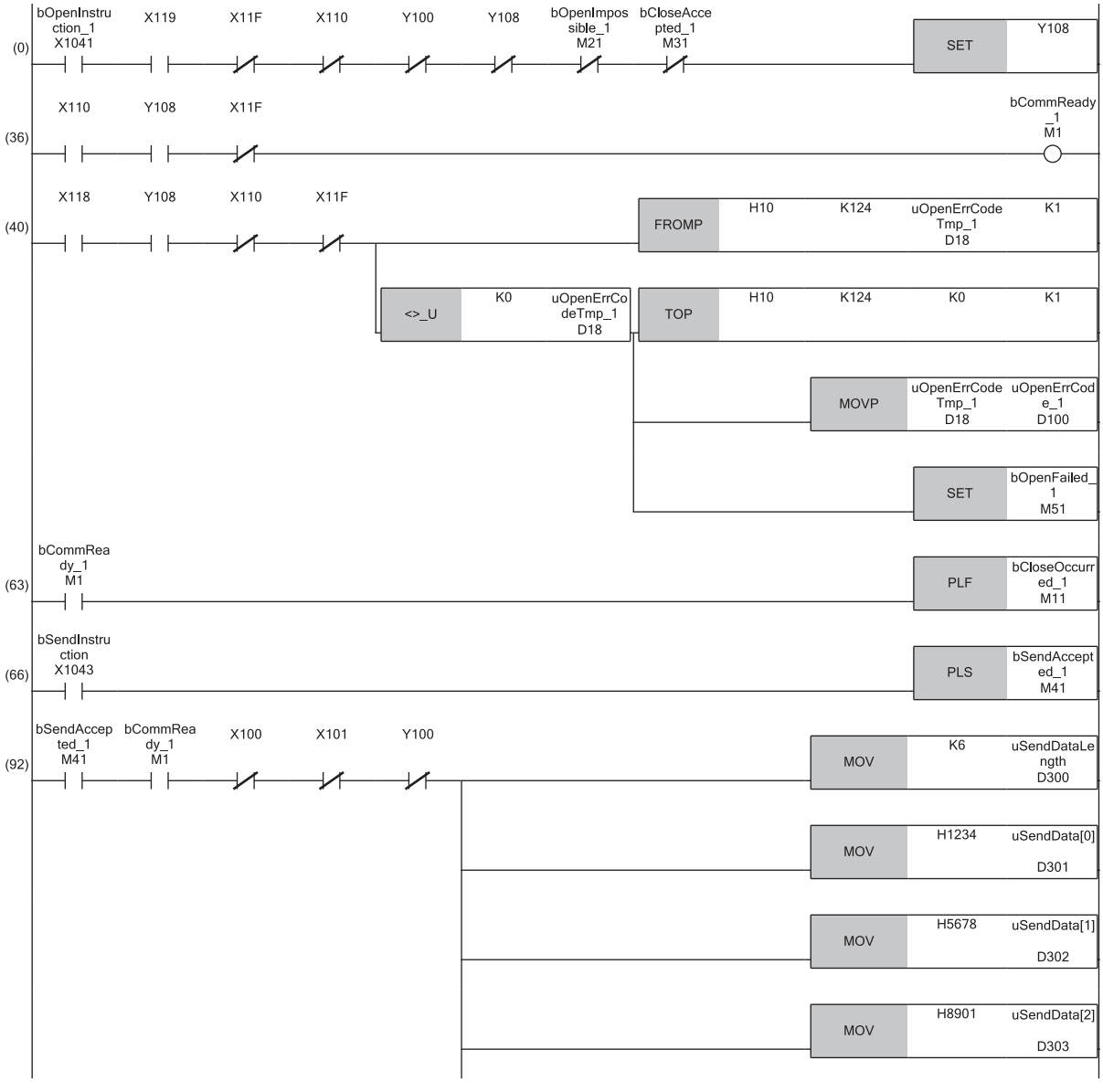
## Program examples

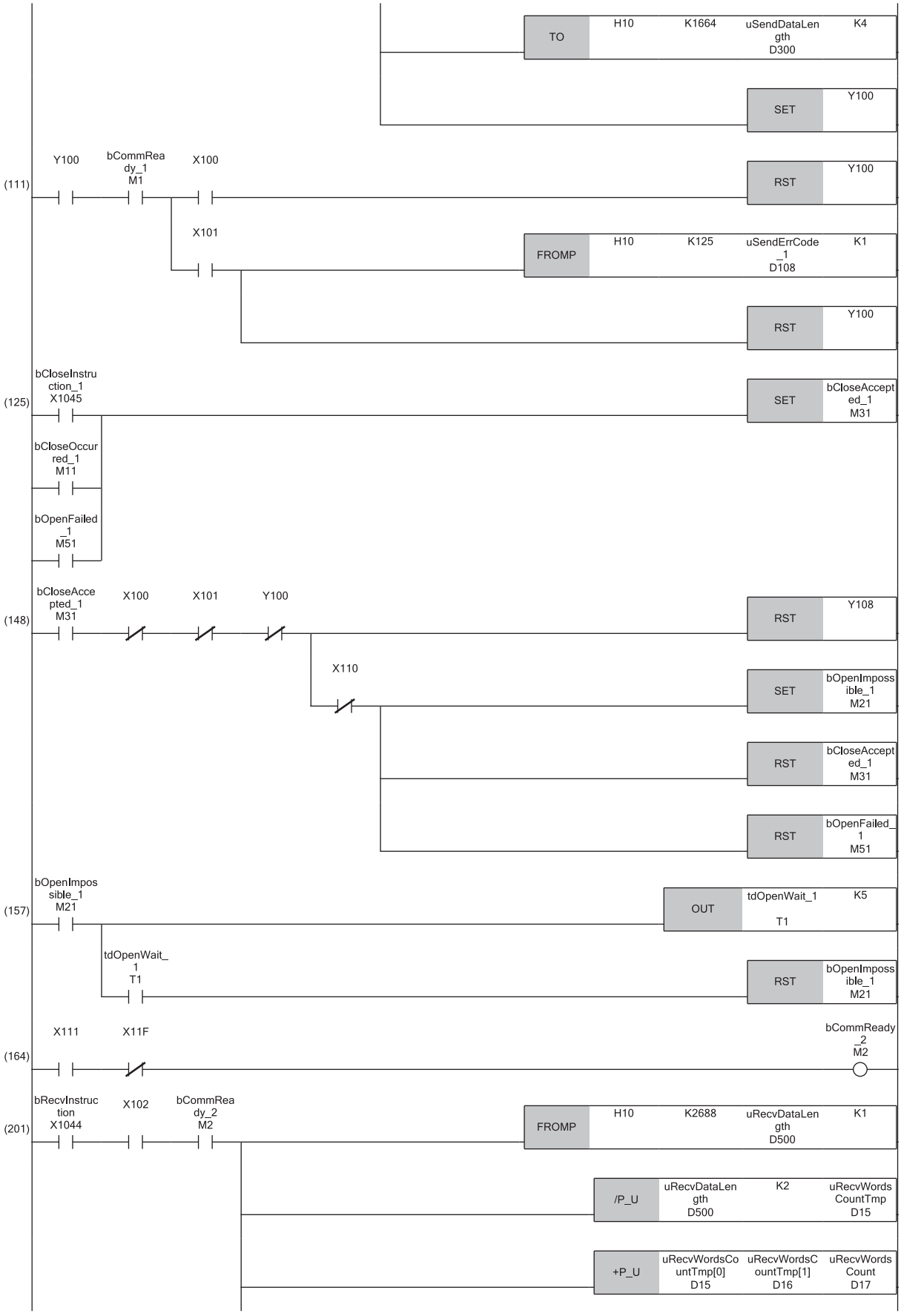
In this example, connection No.1 is opened and data is sent to an external device.

Also, the data from an external device is received using connection No.2.

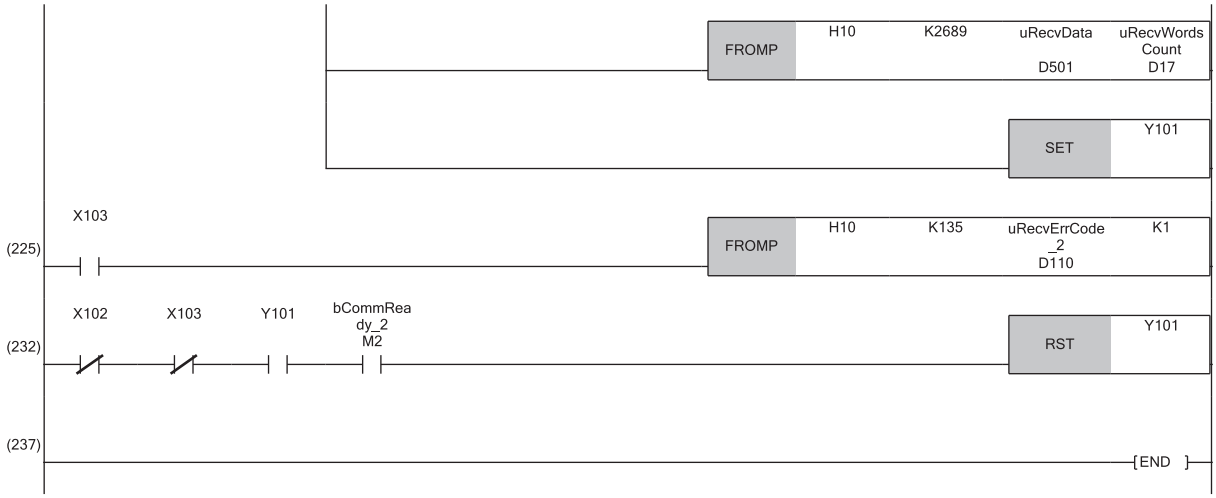
| Classification | Label name |
|---|---|
| Label to be defined | Define global labels as shown below: |

| Label Name | Data Type | | Class | Assign (Device/Label) |
|---|---|---|---|---|
| bOpenInstruction_1 | Bit | ... | VAR_GLOBAL | X1041 |
| bSendInstruction | Bit | ... | VAR_GLOBAL | X1043 |
| bRecvInstruction | Bit | ... | VAR_GLOBAL | X1044 |
| bCloseInstruction_1 | Bit | ... | VAR_GLOBAL | X1045 |
| uRecvWordsCountTmp | Word [Unsigned]/Bit String [16-bit](0..1) | ... | VAR_GLOBAL | D15 |
| uRecvWordsCount | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D17 |
| uOpenErrCodeTmp_1 | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D18 |
| uOpenErrCode_1 | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D100 |
| uSendErrCode_1 | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D108 |
| uRecvErrCode_2 | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D110 |
| uSendDataLength | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D300 |
| uSendData | Word [Unsigned]/Bit String [16-bit](0..2) | ... | VAR_GLOBAL | D301 |
| uRecvDataLength | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D500 |
| uRecvData | Word [Unsigned]/Bit String [16-bit] | ... | VAR_GLOBAL | D501 |
| bCommReady_1 | Bit | ... | VAR_GLOBAL | M1 |
| bCommReady_2 | Bit | ... | VAR_GLOBAL | M2 |
| bCloseOccurred_1 | Bit | ... | VAR_GLOBAL | M11 |
| bOpenImpossible_1 | Bit | ... | VAR_GLOBAL | M21 |
| bCloseAccepted_1 | Bit | ... | VAR_GLOBAL | M31 |
| bSendAccepted_1 | Bit | ... | VAR_GLOBAL | M41 |
| bOpenFailed_1 | Bit | ... | VAR_GLOBAL | M51 |
| tdOpenWait_1 | Timer | ... | VAR_GLOBAL | T1 |




TO
H10
K1664
uSendDataLength
D300
K4
SET
Y100
(111)
Y100
bCommReady_1
M1
X100
RST
Y100
X101
FROMP
H10
K125
uSendErrCode_1
D108
K1
RST
Y100
(125)
bCloseInstruction_1
X1045
SET
bCloseAccepted_1
M31
bCloseOccurred_1
M11
bOpenFailed_1
M51
(148)
bCloseAccepted_1
M31
X100
X101
Y100
RST
Y108
X110
SET
bOpenImpossible_1
M21
RST
bCloseAccepted_1
M31
RST
bOpenFailed_1
M51
(157)
bOpenImpossible_1
M21
OUT
tdOpenWait_1
T1
K5
tdOpenWait_1
T1
RST
bOpenImpossible_1
M21
(164)
X111
X11F
bCommReady_2
M2
(201)
bRecvInstruction
X1044
X102
bCommReady_2
M2
FROMP
H10
K2688
uRecvDataLength
D500
K1
/P_U
uRecvDataLength
D500
K2
uRecvWordsCountTmp
D15
+P_U
uRecvWordsCountTmp[0]
D15
uRecvWordsCountTmp[1]
D16
uRecvWordsCount
D17

A




(0) When bOpenInstruction_1 is turned on, open processing of connection No.1 is performed.
(66)When bSendInstruction is turned on, data is sent to an external device using connection No.1.
(125)When bCloseInstruction_1 is turned on, close processing of connection No.1 is performed.
(201)When bRecvInstruction is turned on, data is sent to an external device using connection No.2.

# Appendix 11 Port Numbers Used by Ethernet-equipped Module

The following port numbers are used by the system and cannot be specified.

## RJ71EN71, network part for the RnENCPU

| Port No. | | Applications | Protocol | Default port status |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 20 | 14H | For FTP server function (data port) | TCP/IP | Close |
| 21 | 15H | For FTP server function (control port) | TCP/IP | Close |
| 123 | 7BH | For clock setting function (SNTP client) | TCP/IP | Close |
| 5000 | 1388H | Auto-open UDP port (default value) | UDP/IP | Open |
| 5001 | 1389H | MELSOFT transmission port (UDP/IP) | UDP/IP | Open |
| 5002 | 138AH | MELSOFT transmission port (TCP/IP) | TCP/IP | Open |
| 5003 | 138BH | MELSOFT direct connection port | UDP/IP | Open*1 |
| 5009 | 1391H | For searching modules on the network | UDP/IP | Open*1 |
| 61696 to 61999 | F100H to F22FH | For simple CPU communication function | UDP/IP | Close |

*1 This setting can be changed in "Application Settings" of the module parameter. (☞ Page 333 Application Settings)

## RnCPU, CPU part of the RnENCPU

| Port No. | | Applications | Protocol | Default port status |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 20 | 14H | For FTP server function (data port) | TCP/IP | Close |
| 21 | 15H | For FTP server function (control port) | TCP/IP | Close |
| 80 | 50H | For Web server function | TCP/IP | Close |
| 123 | 7BH | For clock setting function (SNTP client) | TCP/IP | Close |
| 5001 | 1389H | MELSOFT transmission port (UDP/IP) | UDP/IP | Open*1 |
| 5005 | 138DH | Auto-open UDP port (default value) | UDP/IP | Open*1 |
| 5006 | 138EH | MELSOFT transmission port (UDP/IP) | UDP/IP | Open*1 |
| 5007 | 138FH | MELSOFT transmission port (TCP/IP) | TCP/IP | Open*1 |
| 5008 | 1390H | For MELSOFT direct connection | UDP/IP | Open*2 |
| 5009 | 1391H | For searching modules on the network | UDP/IP | Open*2 |
| 61440 | F000H | For sending data for device supporting iQ Sensor Solution | UDP/IP | Open*2 |
| 61441 | F001H | For SLMPSND instruction | UDP/IP | Open*2 |
| 61442 | F002H | For data backup/restoration for device supporting iQ Sensor Solution | UDP/IP | Open*2 |
| 61448 | F008H | For DNS client | UDP/IP | Close |
| 61450 | F00AH | For CC-Link IE Field Network Basic function | UDP/IP | Close |
| 61461 | F015H | For database access function | UDP/IP | Close |
| 61696 to 61999 | F100H to F22FH | For simple CPU communication function | UDP/IP | Close |
| 62000 to 65534 | F230H to FFFEH | For file transfer function (FTP client) | TCP/IP | Close |

*1 This setting can be changed in "Basic Settings" of the module parameter. (☞ Page 326 Basic Settings)
*2 This setting can be changed in "Application Settings" of the module parameter. (☞ Page 333 Application Settings)

## RnPCPU

| Port No. | | Applications | Protocol | Default port status |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 20 | 14H | For FTP server function (data port) | TCP/IP | Close |
| 21 | 15H | For FTP server function (control port) | TCP/IP | Close |
| 80 | 50H | For Web server function | TCP/IP | Close |
| 123 | 7BH | For clock setting function (SNTP client) | TCP/IP | Close |
| 5005 | 138DH | Auto-open UDP port (default value) | UDP/IP | Open*1 |
| 5006 | 138EH | MELSOFT transmission port (UDP/IP) | UDP/IP | Open*1 |
| 5007 | 138FH | MELSOFT transmission port (TCP/IP) | TCP/IP | Open*1 |
| 5008 | 1390H | For MELSOFT direct connection | UDP/IP | Open*2 |
| 5009 | 1391H | For searching modules on the network | UDP/IP | Open*2 |
| 61441 | F001H | For SLMPSND instruction | UDP/IP | Open*2 |
| 61448 | F008H | For DNS client | UDP/IP | Close |
| 61696 to 61999 | F100H to F22FH | For simple CPU communication function | UDP/IP | Close |
| 62000 to 65534 | F230H to FFFEH | For file transfer function (FTP client) | TCP/IP | Close |

*1 This setting can be changed in "Basic Settings" of the module parameter. (☞ Page 326 Basic Settings)
*2 This setting can be changed in "Application Settings" of the module parameter. (☞ Page 333 Application Settings)

## RnSFCPU

| Port No. | | Applications | Protocol | Default port status |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 20 | 14H | For FTP server function (data port) | TCP/IP | Close |
| 21 | 15H | For FTP server function (control port) | TCP/IP | Close |
| 123 | 7BH | For clock setting function (SNTP client) | TCP/IP | Close |
| 5001 | 1389H | MELSOFT transmission port (UDP/IP) | UDP/IP | Open*1 |
| 5005 | 138DH | Auto-open UDP port (default value) | UDP/IP | Open*1 |
| 5006 | 138EH | MELSOFT transmission port (UDP/IP) | UDP/IP | Open*1 |
| 5007 | 138FH | MELSOFT transmission port (TCP/IP) | TCP/IP | Open*1 |
| 5008 | 1390H | For MELSOFT direct connection | UDP/IP | Open*2 |
| 5009 | 1391H | For searching modules on the network | UDP/IP | Open*2 |
| 61696 to 61999 | F100H to F22FH | For simple CPU communication function | UDP/IP | Close |
| 62000 to 65534 | F230H to FFFEH | For file transfer function (FTP client) | TCP/IP | Close |

*1 This setting can be changed in "Basic Settings" of the module parameter. (☞ Page 326 Basic Settings)
*2 This setting can be changed in "Application Settings" of the module parameter. (☞ Page 333 Application Settings)

## RnPSFCPU

| Port No. | | Applications | Protocol | Default port status |
|---|---|---|---|---|
| Decimal | Hexadecimal | | | |
| 20 | 14H | For FTP server function (data port) | TCP/IP | Close |
| 21 | 15H | For FTP server function (control port) | TCP/IP | Close |
| 123 | 7BH | For clock setting function (SNTP client) | TCP/IP | Close |
| 5005 | 138DH | Auto-open UDP port (default value) | UDP/IP | Open |
| 5006 | 138EH | MELSOFT transmission port (UDP/IP) | UDP/IP | Open |
| 5007 | 138FH | MELSOFT transmission port (TCP/IP) | TCP/IP | Open |
| 5008 | 1390H | For MELSOFT direct connection | UDP/IP | Open*1 |
| 5009 | 1391H | For searching modules on the network | UDP/IP | Open*1 |

*1 This setting can be changed in "Application Settings" of the module parameter. (☞ Page 333 Application Settings)

# Appendix 12 Operation Image and Data Structure of Predefined Protocol

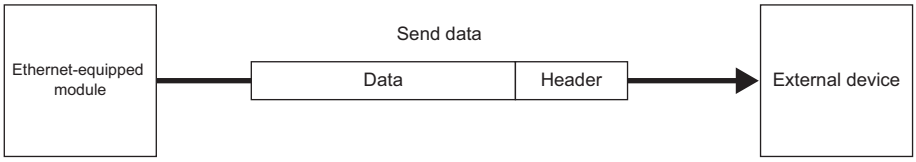
## Operation image of each communication type of protocol

With the predefined protocol support function, data is communicated with the external device using the "Send Only", "Receive Only", and "Send & Receive" communication types.

This section describes the operation images of each communication type.

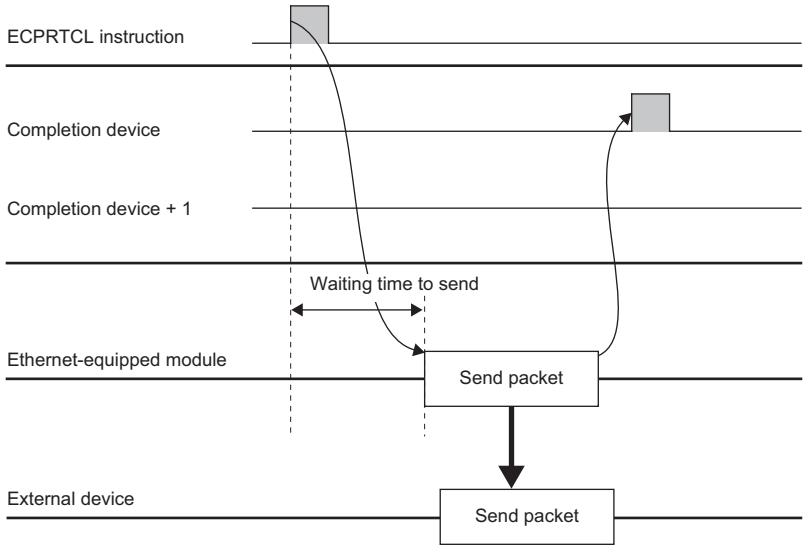
### For "Send Only" communication type

The specified packet is sent once.

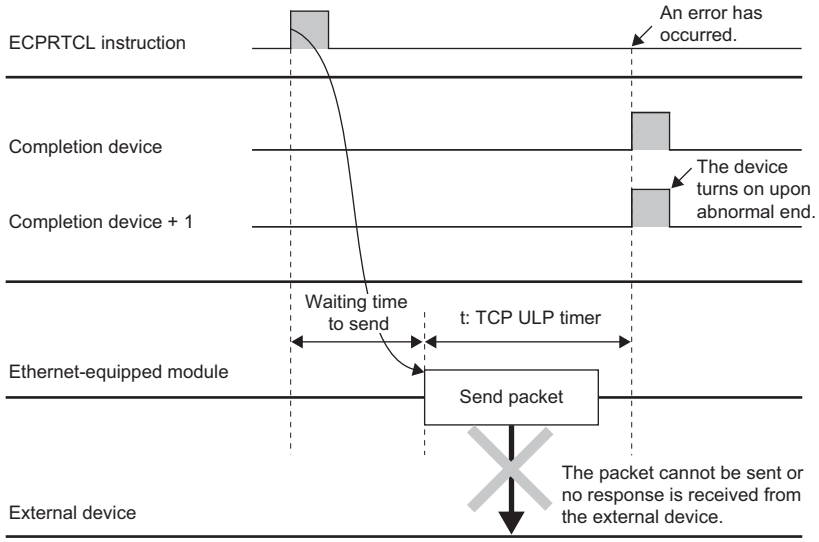



The operation image of "Send Only" is shown below.

#### ■When the instruction completed successfully




#### ■When the instruction completed with an error at TCP/IP (timeout error)




A

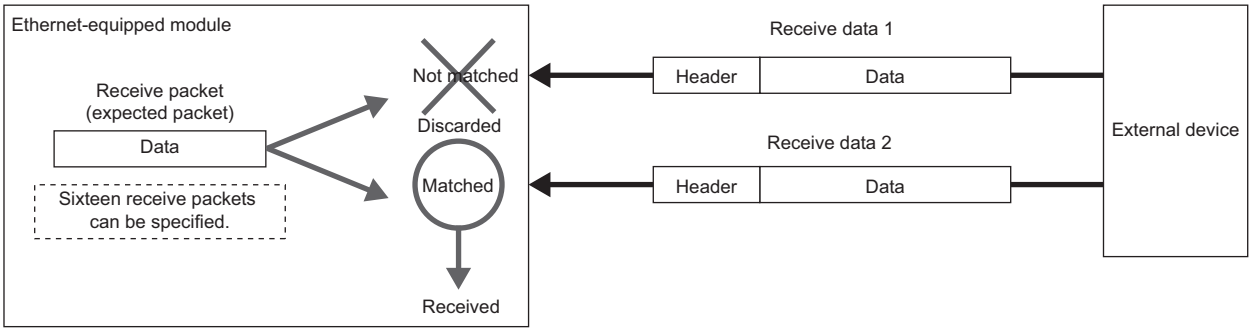
**Point**

For details on errors that occur when the communication completed abnormally, refer to the error code stored in the buffer memory.

(☞ Page 453 List of Error Codes)

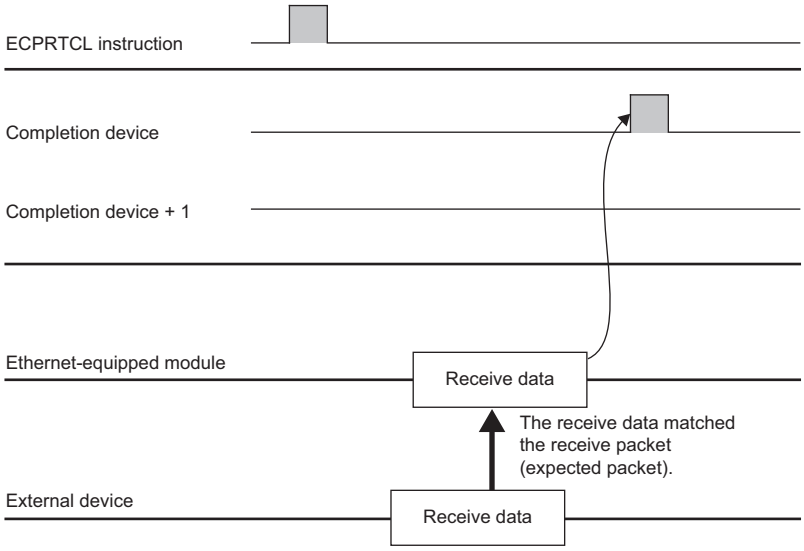
## For "Receive Only" communication type

When data is received from the external device, it is verified with the receive packet (expected packet). If it matches, the receive processing ends. If the verification does not match, the received data is discarded.

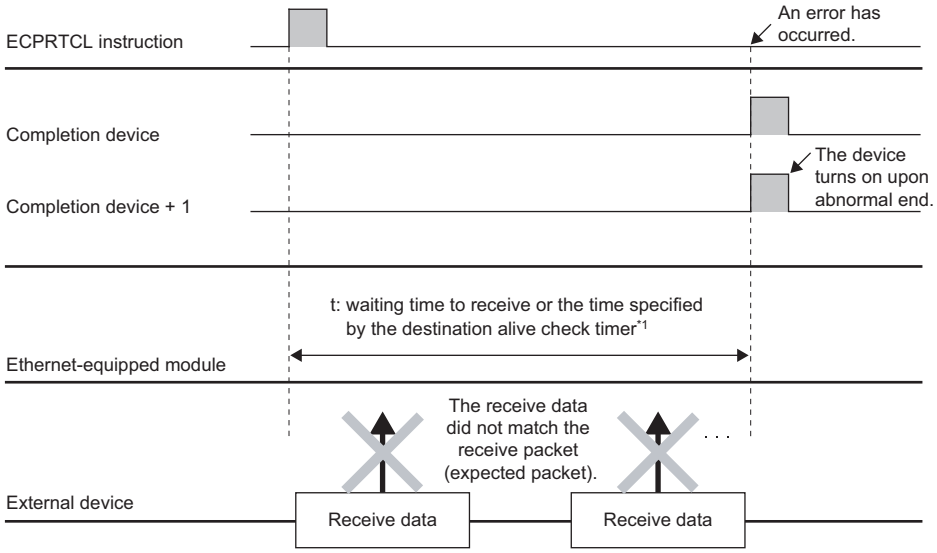



The operation image of "Receive Only" is shown below.

### ■When the instruction completed successfully



## ■When the instruction completed with an error (timeout error)




*1 When the receive packet (expected packet) verification does not match before the reception wait time, or when other device existence cannot be checked within the destination alive check time (time calculated from destination alive check start interval timer value, destination alive check interval timer value, and destination alive check resend count).

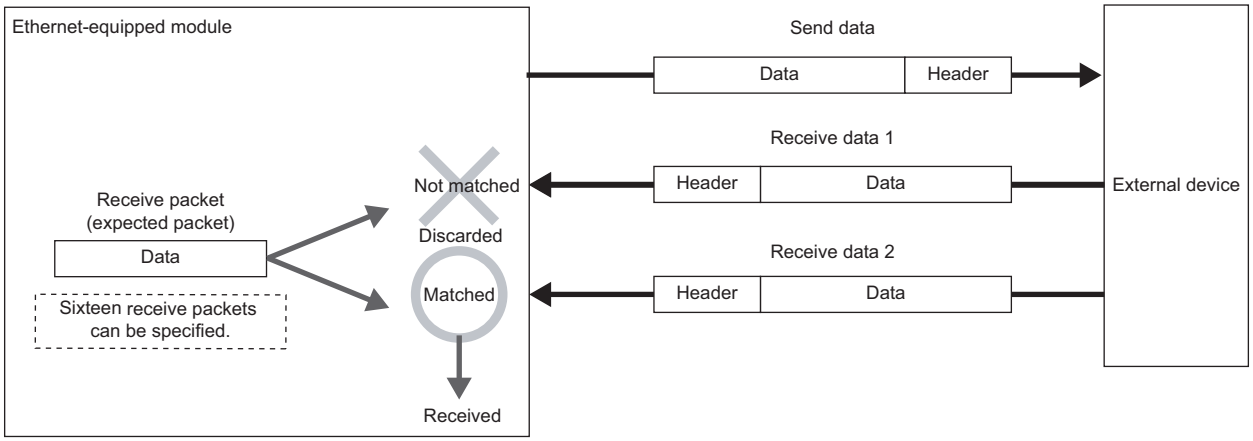
**Point**

- When variables are included in receive packet (expected packet) elements, variable data are not verified.
- Up to 16 receive packets (expected packets) can be specified.
- When multiple packets are specified, the received data is verified with the registered receive packet (expected packet) in the order of registration. The receive processing is completed when a matching receive packet (expected packet) is found, and the subsequent verification is canceled.
- The number of the matching received packet is stored in the control data of the ECPRTCL instruction and the buffer memory.
- For details on errors that occur when the communication completed abnormally, refer to the error code stored in the buffer memory. (☞ Page 453 List of Error Codes)

A

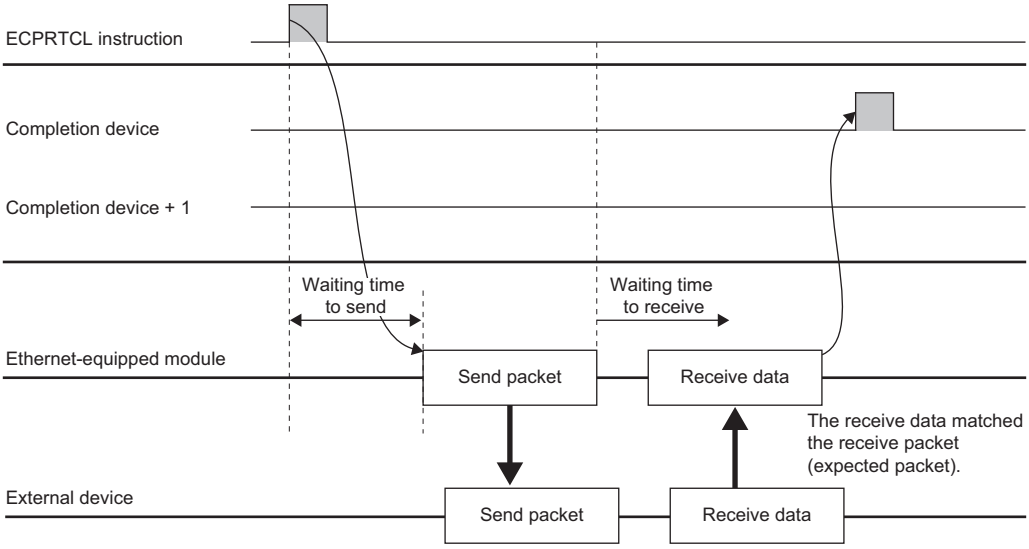
## For "Send & Receive" communication type

The packet is sent once. If the send completed successfully, the module shifts to the receiving standby state. When data is received from the external device, it is verified with the receive packet (expected packet). If it matches, the receive processing ends.

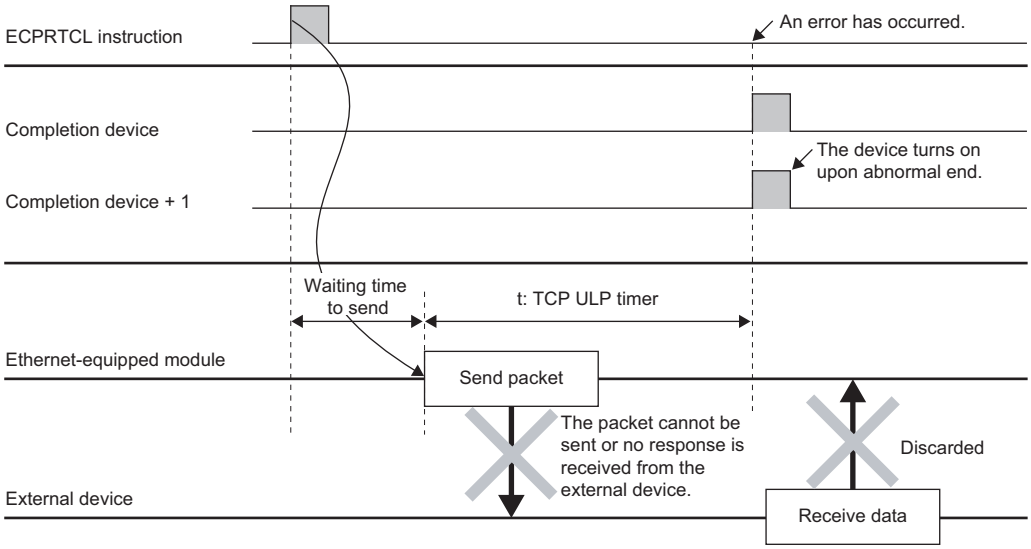



The operation image of "Send & Receive" is shown below.

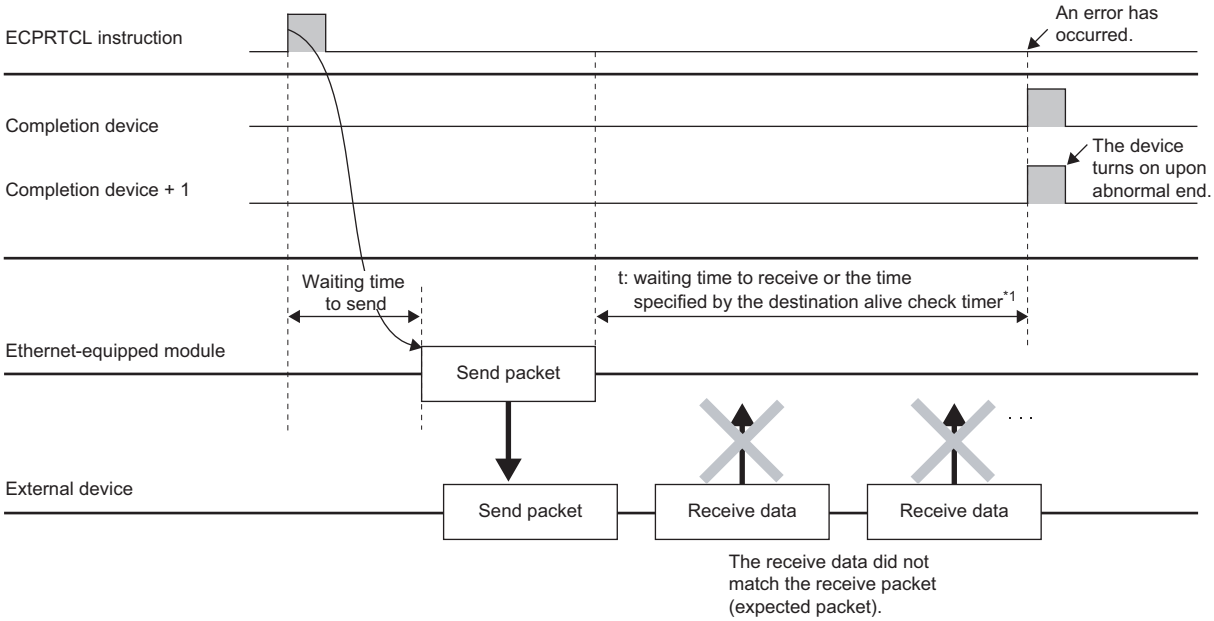
### ■When the instruction completed successfully



## ■When the instruction completed with an error (timeout error at sending)




## ■When the instruction completed with an error (timeout error of waiting time to receive)




*1 When the receive packet (expected packet) verification does not match before the reception wait time, or when other device existence cannot be checked within the destination alive check time (time calculated from destination alive check start interval timer value, destination alive check interval timer value, and destination alive check resend count).

**Point**

For details on errors that occur when the communication completed abnormally, refer to the error code stored in the buffer memory.

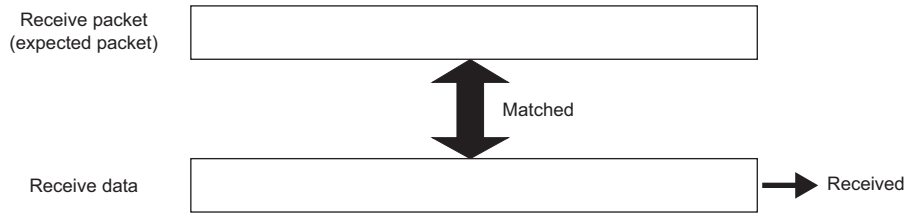
(☞ Page 453 List of Error Codes)

# Verification operation of receive packet

This section describes the receive packet (expected packet) verification operation for communication with an external device when the protocol communication type contains reception.

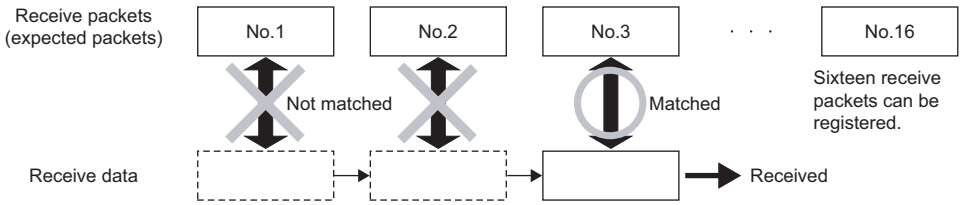
## When received data are matched with a receive packet (expected packet)

The received data is compared against the receive packet (expected packet), and the receive processing completes when the verification matches.



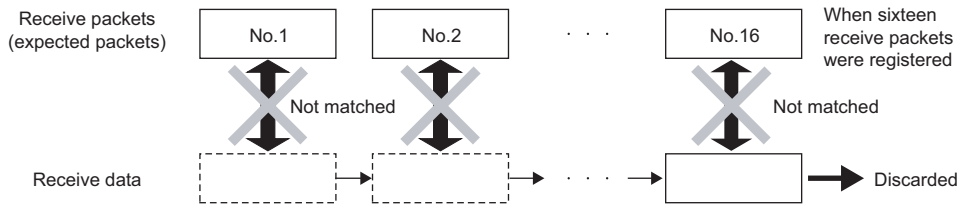

## When multiple receive packets (expected packets) are specified

Up to 16 receive packets (expected packets) can be registered with the predefined protocol support function. When the data are received, the registered receive packet (expected packet) is verified in the order of registration. The receive processing completes as soon as a matching receive packet (expected packet) is found.




## When received data does not match with all receive packet (expected packet)

If the received data does not match any of the registered receive packets (expected packets), the received data is discarded.



# Example of packet element data

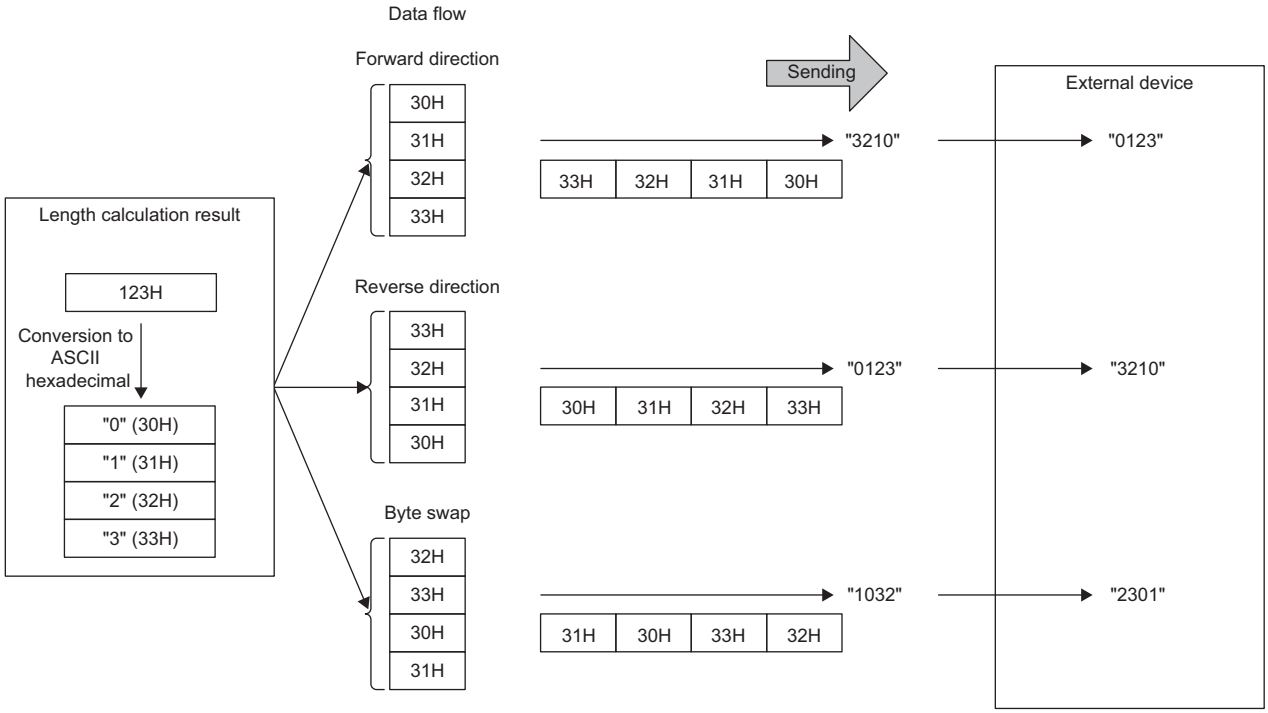
The element processing procedures that can be set for the packets, the actual data examples, and others are shown below.

## Length

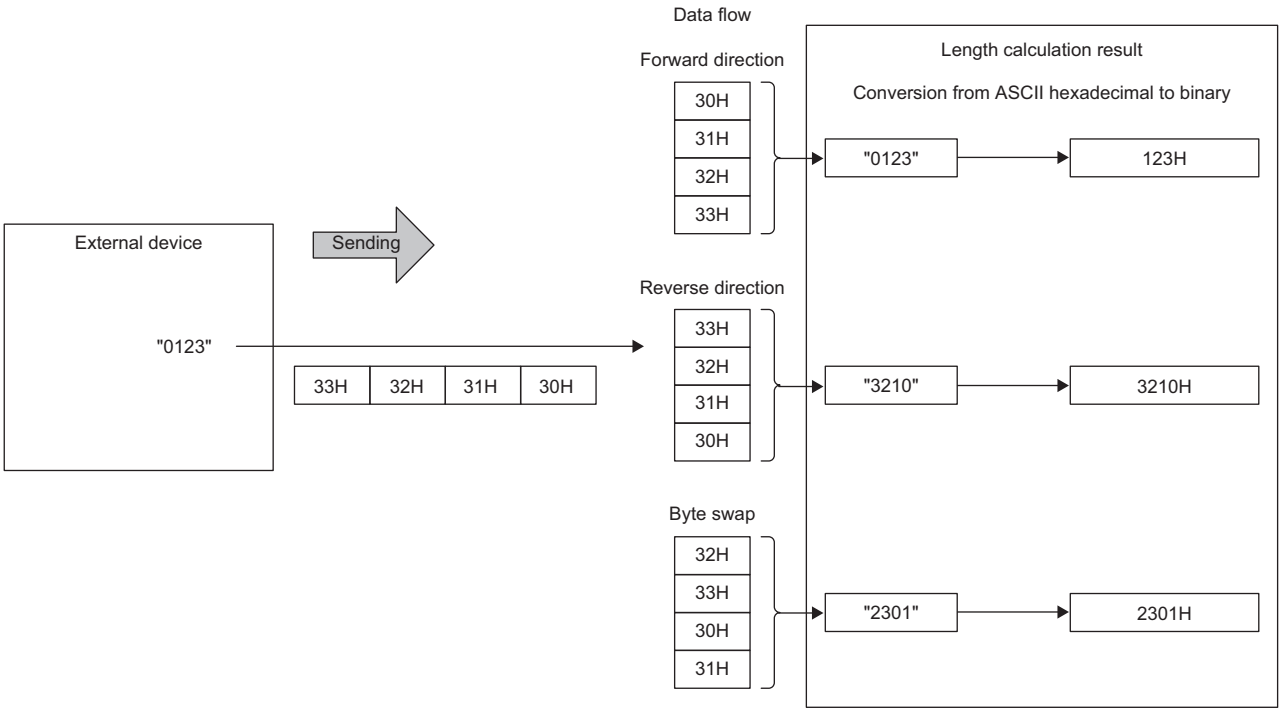
### ■Process procedure

The Ethernet-equipped module processes the length with the following procedure.

[During send]




[During receive]




A

### ■Data flow

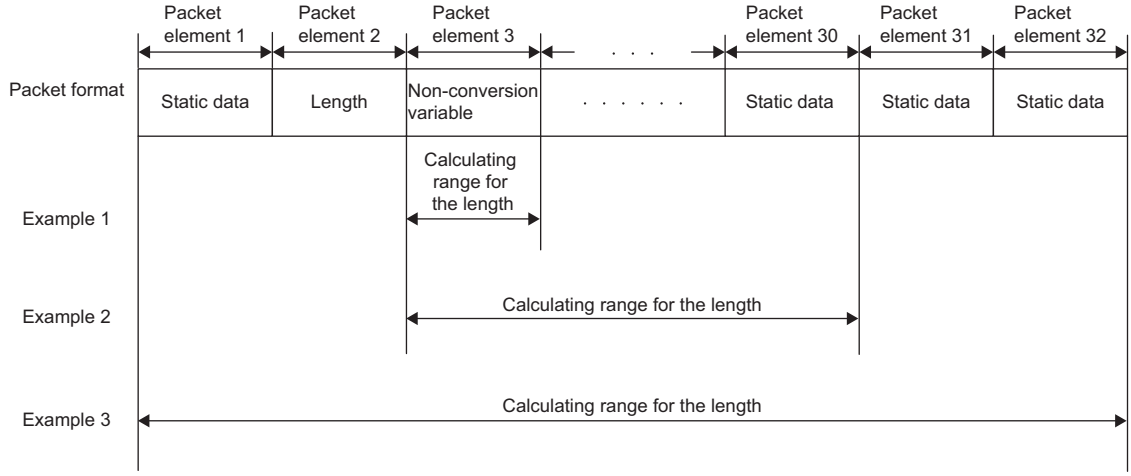
Data flow is used to specify the order of send data and receive data.

The data flow can be specified in forward direction (upper byte → lower byte), reverse direction (lower byte → upper byte), and byte swap (by word).

- Forward direction, reverse direction: Available when the data length is 2 bytes or more.
- Byte swap: Available only when the data length is 4 bytes.

### ■Calculating range

Examples of specifying the length calculating range are shown below.




Example 1:Calculating range when start of range is set as 3 and end is set as 3
Example 2:Calculating range when start of range is set as 3 and end is set as 30
Example 3:Calculating range when start of range is set as 1 and end is set as 32

## Non-conversion Variable

### ■Process procedure

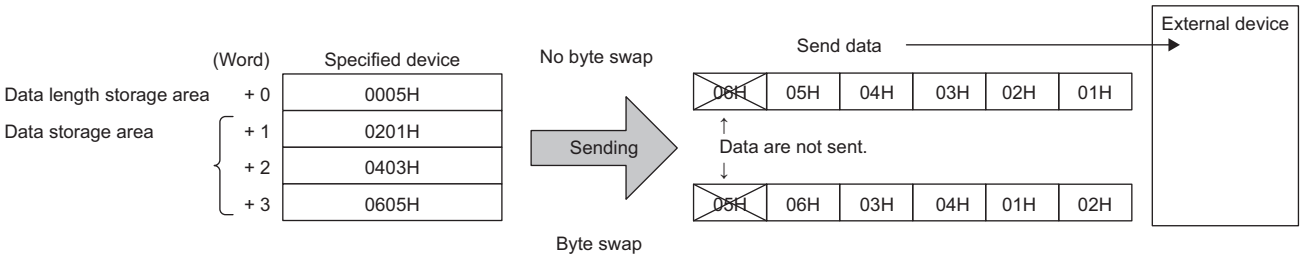
The Ethernet-equipped module processes the non-conversion variable element with the following procedure.

When unit of stored data is "Lower Byte + Upper Byte"

- When the data length of a send packet is an odd number, the upper byte (lower byte when bytes are swapped) of the end device is not sent.
- When the data length of a receive packet is an odd number, the last data is stored with one byte of 00H.

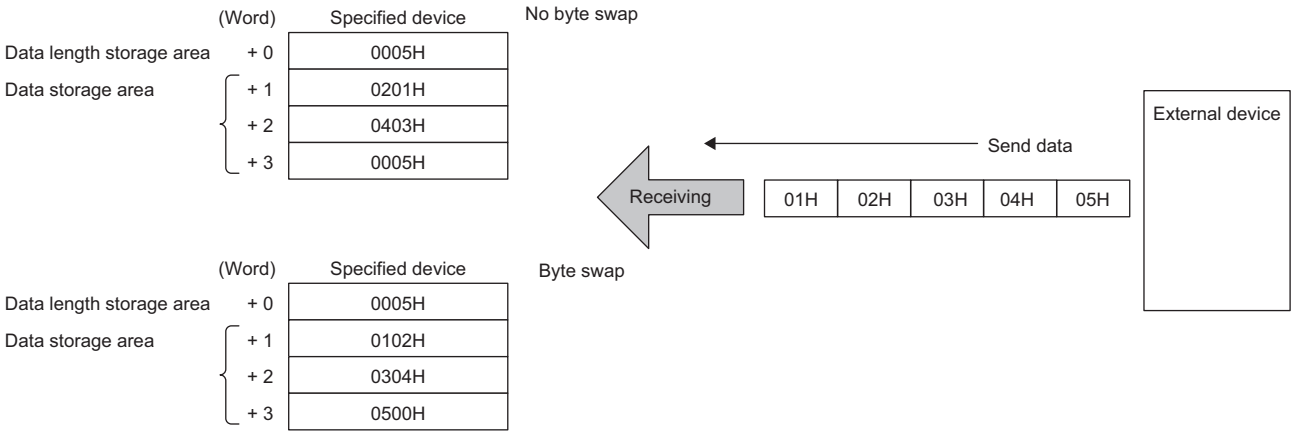
**Ex.**

When sending data whose length is an odd number



Ex.

When receiving data whose length is an odd number




When unit of stored data is "Lower Byte Only"

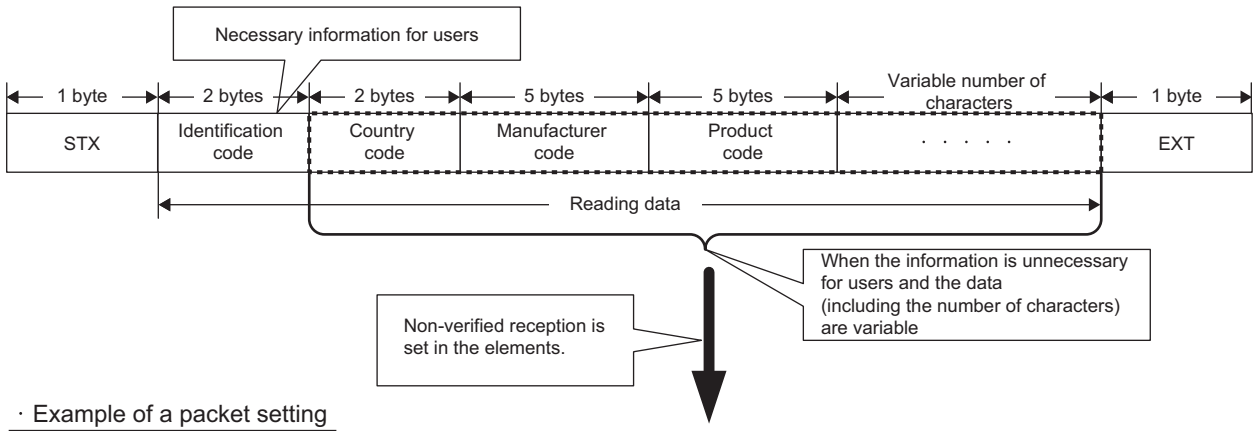
- A size double the data length is occupied. For the upper data, the Ethernet-equipped module ignores the data at sending and adds 00H to the data at receiving.

## Non-verified reception

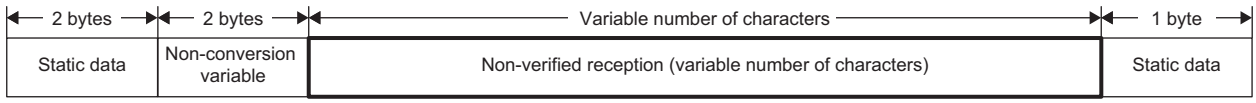
The usage examples for a non-verified reception element is shown below.

Ex.

· Example of a packet format of the external device




· Example of a packet setting




In the above type of packet format, the following operation can be executed by setting reception without verification.

- It is possible to store only the required information in the CPU module's device or buffer memory.
- Even if the received packet contains data that changes with each communication, it can be handled with one protocol (packet).

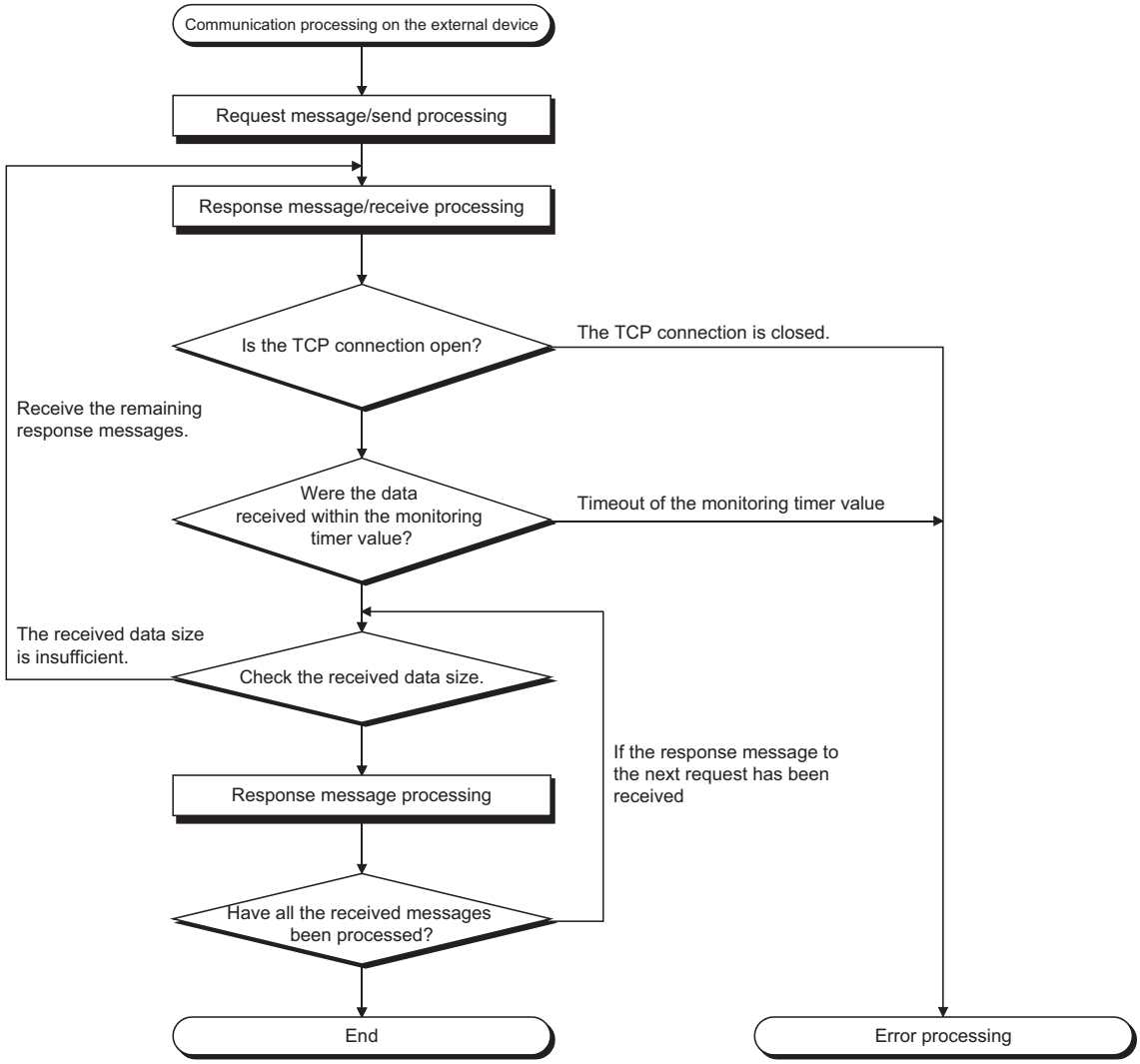
A

# Appendix 13 Example of External Device Program

An example of the program in the external device connected to the same Ethernet as the Ethernet-equipped module is shown below. Each program is the minimum required program for the communication. Change the IP address or port number according to the system. When including process for errors, add it separately.

## External device receiving process

An example of the external device receiving process is shown below.




**Point**

When communicating with Ethernet, the TCP socket functions (socket functions) are used in the personal computer. There is no concept of boundary with these functions. When the send side calls the send function once and sends data, the receiving side must call the recv function once, twice, or depending on the circumstance more times to read that data. (send and recv do not correspond one-on-one.) Thus, the external device's program process must execute the receiving process as shown above.

# Appendix 14 How to Turn Off ERR LED

If an error occurs in communications with the external device, the ERR LED turns on. Even though the error cause is eliminated, the ERR LED does not turn off. This section describes how to turn off the ERR LED.

## Method using a dedicated instruction

The ERR LED can be turned off by the ERRCLEAR instruction. Availability for turning off the LED depends on the firmware version of the RJ71EN71. (☞ Page 586 Added and Enhanced Functions)

For details on the ERRCLEAR instruction, refer to the following.

MELSEC iQ-R Programming Manual (Module Dedicated Instructions)

## Method using I/O signals

When the network type is either of the following, 'ERR LED off request' (Y17) can be used to turn off the ERR LED. (☞ Page 482 I/O Signals)

- Ethernet (Only when the P1 connector is set for Ethernet and the P2 connector is also set for Ethernet)
- Q-compatible Ethernet

# Appendix 15 Software Licenses and Copyrights

This section describes the licenses and copyrights of the software used in this product.

## MD5 Message-Digest Algorithm

This product includes code that was developed by RSA Data Security, Inc.
Copyright © 1991-2, RSA Data Security, Inc.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

# MEMO

# Appendix 16 Added and Enhanced Functions

## RJ71EN71, RnENCPU (network part)

The following table lists the added and enhanced functions in the RJ71EN71, RnENCPU (network part).

○: Available (No version restriction), ×: Not available

| Added and enhanced function | Firmware version | |
|---|---|---|
| | RJ71EN71 | RnENCPU (network part) |
| Turning off the LED with the ERRCLEAR instruction | "05" or later | ○ |
| 'State of receive buffer' (Un\G1900030)*1 | "05" or later | ○ |
| IP address change function | "10" or later | "10" or later |
| Redundant system function | "12" or later | × |
| Remote head module redundant system | "12" or later | × |
| Accessing to another station via the port 1 and port 2 using SLMP communications | "18" or later | × |
| Simple CPU communication function | "35" or later | "35" or later |
| MODBUS/TCP slave function | "35" or later | "35" or later |
| Simple CPU communication function for MELSEC iQ-F series | "35" or later | "35" or later |
| Simple CPU communication function<br>• Setting number of up to 512<br>• On-request communications<br>• MELSEC-A/AnS (Ethernet module) and MELSEC-FX3 (Ethernet block/adapter) as communication destination | "39" or later | "39" or later |
| MODBUS/TCP slave function for redundant system | "42" or later | "42" or later |
| Simple CPU communication function<br>• Redundant system<br>• MELSEC iQ-L series, SIEMENS S7 series, Fuji Electric MICREX-SX series, and JTEKT TOYOPUC series as communication destination | "42" or later | "42" or later |
| Discard received data at CPU STOP | "42" or later | "42" or later |
| Optional function to specify the module ID when using MODBUS/TCP in the simple CPU communication function | "47" or later | "47" or later |
| Extension base unit in a redundant system with redundant extension base unit | "50" or later | × |
| Simple device communication function | "56" or later | "56" or later |
| Protocol setting editing function with simple device communication function*2 | "59" or later | "59" or later |
| Simple CPU communication function*3<br>• SIEMENS S7 series (extension) as communication destination<br>• Additional devices that can be specified | "62" or later | "62" or later |
| 'Error log area' (Un\G227 to Un\G372) | "66" or later | "66" or later |
| Time setting function (SNTP client) for daylight saving time function | "68" or later | "68" or later |
| Simple device communication function*2<br>• Element "Error check code"<br>• Enable (Upper → Lower, two-word units), Enable (Upper → Lower, four-word units) for byte swap for non-conversion variables | "71" or later | "71" or later |
| Simple device communication function<br>• Element "Conversion Variable"<br>• "Data Unit" for the length | "75" or later | "75" or later |
| Simple device communication function*2<br>• "Receive Only" in "Communication Type"<br>• "Extension 2" in "Resource Setting" | "77" or later | "77" or later |
| Simple CPU communication function*3<br>• OMRON (NJ/NX series) as communication destination<br>• Option settings to enable FINS node addresses of the own station | "88" or later | "88" or later |
| Re-initial processing | "90" or later | "90" or later |
| Turning off the ERR LED using I/O signals | "90" or later | "90" or later |
| Simultaneous execution of the following dedicated instructions and module FBs<br>• SOCSND instruction<br>• SOCRCV instruction/SOCRCVS instruction<br>• M+Model name_Send_Socket<br>• M+Model name_Recv_Socket | "90" or later | "90" or later |

*1 'Receive buffer full detection signal' (Un\G21056) can be used when the network type is set to "Q Compatible Ethernet".
*2 For the engineering tool version restrictions, refer to the following.
☞ Page 292 Restrictions applicable depending on versions
*3 For the engineering tool version restrictions, refer to the following.
☞ Page 226 Restrictions applicable depending on versions

# CPU module (built-in Ethernet port part)

The following table lists the added and enhanced functions in the CPU module (built-in Ethernet port part).
R00/R01/R02: R00CPU, R01CPU, R02CPU, Rn: RnCPU (except R00CPU, R01CPU, and R02CPU), RnEN: RnENCPU, RnP(P): Process CPU (process mode), RnP(R): Process CPU (redundant mode), RnPSF: SIL2 Process CPU, RnSF: Safety CPU
○: Available (No version restriction), ×: Not available

| Added and enhanced function | Firmware version | | | | | | |
|---|---|---|---|---|---|---|---|
| | R00/R01/R02 | Rn | RnEN | RnP(P) | RnP(R) | RnPSF | RnSF |
| Redundant system function | × | × | × | × | "04" or later | ○ | × |
| File transfer function (FTP client) | ○ | "22" or later | "22" or later | "13" or later | "13" or later | × | × |
| High-speed write mode (File transfer function (FTP server)) | ○ | "23" or later | "23" or later | × | × | × | × |
| Simple CPU communication function | "05" or later | "35" or later | "35" or later | "34" or later | "34" or later | × | "30" or later |
| Communications by network number/station number | "08" or later | "40" or later | "40" or later | × | × | × | "36" or later |
| Simple CPU communication function for MELSEC iQ-F series | "08" or later | "40" or later | "40" or later | "34" or later | "34" or later | × | "30" or later |
| CPU module built-in database access function from external device | × | "28" or later | "28" or later | × | × | × | × |
| Simple CPU communication function<br>• MELSEC iQ-L series as a communication destination | "14" or later | "46" or later | "46" or later | "34" or later | "34" or later | × | "30" or later |
| Discard received data at CPU STOP | "15" or later | "47" or later | "47" or later | "25" or later | "25" or later | × | × |
| Simple CPU communication function<br>• MELSEC-Q (Ethernet module) and MELSEC-L (Ethernet module) as communication destination | "17" or later | "49" or later | "49" or later | × | × | × | × |
| Simple CPU communication function<br>• MELSEC-A/AnS (Ethernet module) and MELSEC-FX3 (Ethernet block/adapter) as communication destination | "17" or later | "49" or later | "49" or later | "34" or later | "34" or later | × | "30" or later |
| iQ Sensor Solution | 📖 iQ Sensor Solution Reference Manual | | | | | | |
| FTP server function file path name switching | "19" or later | "51" or later | "51" or later | × | × | × | × |
| Simple CPU communication function<br>• Initial communication setting | "21" or later | "53" or later | "53" or later | —*2 | —*2 | × | "30" or later |
| Connection close wait time setting | "23" or later | "56" or later | "56" or later | × | × | × | × |
| User Web page file of the Web server function as the files that can be transferred with the file transfer function (FTP server) | "30" or later | "63" or later | "63" or later | "37" or later | "37" or later | × | × |
| Simple CPU communication function<br>• "On Request" in "Communication Setting" | "38" or later*1 | "71" or later*1 | "71" or later*1 | × | × | × | × |
| Simple CPU communication function<br>• F, T, ST, and C devices that can be specified | "38" or later*1 | "71" or later*1 | "71" or later*1 | × | × | × | × |
| Simple CPU communication function<br>• MELSEC MX Controller (MX-R Model) (Built-in Ethernet), MELSEC MX Controller (MX-F Model) (Built-in Ethernet), and MELSEC iQ-R (Ethernet module) as communication destinations | "42" or later*3 | "81" or later*3 | "81" or later*3 | × | × | × | × |

*1 The version is available for the engineering tool with version "1.105K" or later.
*2 The initial communication setting is already set (enable) by default. The setting cannot be changed because "Continue Communicating Until Receiving Response" in the setting item is selected and grayed out. (☞ Page 245 Operations when the communication setting is "Fixed Interval")
*3 The version is available for the engineering tool with version "1.130L" or later.

# INDEX


## A

Active open . . . . . 548
Application data . . . . . 83,101
Area for sending/receiving instructions . . . . . 509
Automatic negotiation . . . . . 337
Automatic response system . . . . . 408

## C

Combination system . . . . . 412
Command input monitoring timer . . . . . 338
Communication data code . . . . . 327
Communication method . . . . . 330
Communication path determination status . . . . . 513
Communication status test . . . . . 432
Communications by network number/station number . . . . . 327
Connection close wait time setting . . . . . 67
Connection close wait time setting area for socket communications (TCP/IP) . . . . . 533
Connection status . . . . . 430
Connection status storage area . . . . . 506,529,533
Creating the protocol setting data . . . . . 37

## D

Drive name (drive No.) . . . . . 124

## E

Enable/disable online change . . . . . 327
End codes . . . . . 479
Error information . . . . . 422
Ethernet diagnostics . . . . . 426
Existence confirmation . . . . . 331

## F

Fixed buffer send/receive setting . . . . . 330
Forced connection invalidation setting area . . . . . 510,531
Fullpassive . . . . . 549

## G

Gateway IP address . . . . . 405

## H

Header . . . . . 83,100

## I

Initial error code . . . . . 523
Initial status . . . . . 523
IP address . . . . . 327
IP address calculation system . . . . . 409
IP address duplication status storage area . . 509,530
IP filter function . . . . . 177
IP filter settings . . . . . 404

## J

Jumbo frame . . . . . 336

## L

Lock processing . . . . . 180
Logical address . . . . . 100
Login name . . . . . 338

## M

Module communication test . . . . . 415,424
Module diagnostics . . . . . 422
Module information list . . . . . 423

## N

Network number . . . . . 327
No procedure . . . . . 72
Number of retries . . . . . 403

## O

Offline . . . . . 415
Online . . . . . 415
Open completion signal . . . . . 523
Open request signal . . . . . 523
Opening method . . . . . 327
Own node operation status storage area . . . 508,530
Own node setting status storage area . . . . . 505

## P

Pairing open . . . . . 81
Passive open . . . . . 549
Password setting . . . . . 338
Physical address . . . . . 100
PING test . . . . . 431
Predefined protocol library . . . . . 39
Procedure exist . . . . . 72

## R

Remote head module . . . . . 562
Remote password check operation . . . . . 181
Remote password lock status storage area . . . . . 510,530
Response monitoring timer . . . . . 339

## S

Searching modules on the network . . . . . 25
Send frame . . . . . 336
SLMP commands . . . . . 34
Socket/fixed buffer reception status signal . . . . . 523
State of receive buffer . . . . . 524
Station number . . . . . 327
Status for each protocol . . . . . 429
Status of each connection . . . . . 428
Status of each protocol . . . . . 507
Structure of the module label . . . . . 481

Subnet address . . . . . 405
Subnet mask pattern . . . . . 413

## T

Table conversion system . . . . . 411
Transient transmission group number . . . . . 327

## U

Unlock processing. . . . . 180
Unpassive . . . . . 549

I

# REVISIONS

*The manual number is given on the bottom left of the back cover.

| Revision date | *Manual number | Description |
|---|---|---|
| June 2014 | SH(NA)-081257ENG-A | First edition |
| July 2014 | SH(NA)-081257ENG-B | Error correction |
| November 2014 | SH(NA)-081257ENG-C | ■Added or modified parts<br>Section 2.2, 3.1, 3.2, 3.5, 3.7, Appendix 1, 2 |
| April 2015 | SH(NA)-081257ENG-D | ■Added models<br>R08PCPU, R16PCPU, R32PCPU, R120PCPU<br>■Added or modified parts<br>Section 1.3, 1.5, 2.3, 3.5, Appendix 2, 4, 11 |
| August 2015 | SH(NA)-081257ENG-E | ■Added models<br>R04ENCPU, R08ENCPU, R08SFCPU, R16ENCPU, R16SFCPU, R32ENCPU, R32SFCPU, R120ENCPU, R120SFCPU<br>■Added functions<br>Automatic detection of connected device, communication setting reflection, and sensor parameter read/write of the CPU module<br>■Added or modified parts<br>INTRODUCTION, RELEVANT MANUALS, TERMS, Section 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.10, 1.11, 2.1, 2.2, 2.3, 3.1, 3.2, 3.3, 3.4, 3.5, Appendix 1, 2, 3, 4, 7, 8 |
| January 2016 | SH(NA)-081257ENG-F | ■Added or modified parts<br>RELEVANT MANUALS, TERMS, Section 1.1, 1.3, 1.5, 1.8, 1.10, 3.4, 3.5, Appendix 3, 4, 8, 11 |
| May 2016 | SH(NA)-081257ENG-G | ■Added functions<br>File transfer function (FTP client), redundant system function<br>■Added or modified parts<br>INTRODUCTION, RELEVANT MANUALS, TERMS, Section 1.1, 1.3, 1.5, 1.9, 1.11, 1.12, 1.13, 2.2, 2.3, 3.2, 3.4, 3.5, 3.6, 3.7, Appendix 3, 4, 6, 7, 8, 9, 12 |
| October 2016 | SH(NA)-081257ENG-H | ■Added function<br>High-speed write mode (File transfer function (FTP server))<br>■Added or modified parts<br>Section 1.8, 2.2 |
| April 2017 | SH(NA)-081257ENG-I | ■Added function<br>Improvements on Ethernet diagnostics<br>■Added or modified parts<br>RELEVANT MANUALS, TERMS, Section 1.2, 1.9, 2.3, 3.3, Appendix 12 |
| October 2017 | SH(NA)-081257ENG-J | ■Added models<br>R00CPU, R01CPU, R02CPU, R08PSFCPU, R16PSFCPU, R32PSFCPU, R120PSFCPU<br>■Added or modified parts<br>SAFETY PRECAUTIONS, TERMS, Section 1.3, 1.9, 2.3, Appendix 4, 12 |
| April 2018 | SH(NA)-081257ENG-K | ■Added functions<br>Web server function and simple CPU communication function (CPU module (built-in Ethernet port part))<br>■Added or modified parts<br>Section 1.11, 1.13, 2.3, Appendix 3, 12 |
| October 2018 | SH(NA)-081257ENG-L | ■Added functions<br>MODBUS/TCP slave function, simple CPU communication function (RJ71EN71, RnENCPU (network part)), and connection by specifying the network number and station number (CPU module (built-in Ethernet port part))<br>■Added or modified parts<br>Section 1.1, 1.7, 1.14, 1.15, 1.16, 2.2, 2.3, Appendix 3, 9 |
| April 2019 | SH(NA)-081257ENG-M | ■Added function<br>Enhancements of the simple CPU communication function (RJ71EN71, RnENCPU (Network Part))<br>■Added or modified parts<br>TERMS, GENERIC TERMS AND ABBREVIATIONS, Section 1.15, 2.3, 3.4, 3.5, Appendix 3, 13 |
| August 2019 | SH(NA)-081257ENG-N | Error correction |
| October 2019 | SH(NA)-081257ENG-O | ■Added functions<br>Redundant system supported (RJ71EN71) of the MODBUS/TCP slave function, redundant system supported (RJ71EN71) of the simple CPU communication function and external device enhancement (RJ71EN71, RnENCPU (network part)), discard received data at CPU STOP (RJ71EN71, RnENCPU (network part))<br>■Added or modified parts<br>TERMS, GENERIC TERMS AND ABBREVIATIONS, Section 1.7, 1.14, 1.15, 1.17, 1.18, 2.3, Appendix 13 |

| Revision date | *Manual number | Description |
|---|---|---|
| February 2020 | SH(NA)-081257ENG-P | ■Added function<br>Discard received data at CPU STOP (built-in Ethernet port)<br>■Added or modified parts<br>Appendix 3, 13 |
| May 2020 | SH(NA)-081257ENG-Q | ■Added functions<br>Extension of the simple CPU communication function (built-in Ethernet port), optional function for specifying the module ID when using MODBUS/TCP in the simple CPU communication function<br>■Added or modified parts<br>Section 1.1, 1.14, 1.15, 2.2, 2.3, Appendix 13 |
| July 2020 | SH(NA)-081257ENG-R | ■Added or modified parts<br>SAFETY PRECAUTIONS, CONDITIONS OF USE FOR THE PRODUCT |
| October 2020 | SH(NA)-081257ENG-S | ■Added functions<br>Directory delimiter switching setting (FTP server), extension base unit in a redundant system with redundant extension base unit<br>■Added or modified parts<br>SAFETY PRECAUTIONS, RELEVANT MANUALS, TERMS, Section 1.1, 1.5, 1.9, 1.14, 1.17, 2.3, 3.5, Appendix 9, 13 |
| April 2021 | SH(NA)-081257ENG-T | ■Added function<br>Simple device communication function<br>■Added or modified parts<br>TERMS, Section 1.14, 1.15, 1.16, 1.18, 2.3, 3.4, Appendix 3, 8, 9, 14, 15 |
| August 2021 | SH(NA)-081257ENG-U | ■Added or modified part<br>Section 1.4 |
| October 2021 | SH(NA)-081257ENG-V | ■Added function<br>Extension for simple device communication function<br>■Added or modified parts<br>Section 1.3, 1.4, 1.13, 1.16, 2.2, 2.3, 3.4, 3.6, Appendix 3, 6, 10, 11, 15, 16 |
| June 2022 | SH(NA)-081257ENG-W | ■Added function<br>Enhancements of the simple CPU communication function<br>■Added or modified parts<br>SAFETY PRECAUTIONS, RELEVANT MANUALS, Section 1.9, 1.12, 1.13, 1.14, 1.15, 1.16, 2.3, 3.4, 3.5, Appendix 10, 14, 15 |
| July 2022 | SH(NA)-081257ENG-X | ■Added or modified part<br>Appendix 10 |
| December 2022 | SH(NA)-081257ENG-Y | ■Added or modified parts<br>Section 1.15, 2.3, 3.3, 3.8, Appendix 3, 15 |
| April 2023 | SH(NA)-081257ENG-Z | ■Added function<br>Extension for simple device communication function<br>■Added or modified parts<br>RELEVANT MANUALS, Section 1.1, 1.3, 1.9, 1.10, 1.11, 1.12, 1.14, 1.16, 1.18, 2.3, Appendix 3, 15 |
| June 2023 | SH(NA)-081257ENG-AA | ■Added or modified part<br>Appendix 15 |
| November 2023 | SH(NA)-081257ENG-AB | ■Added function<br>Extension for simple device communication function<br>■Added or modified parts<br>Section 1.16, 2.3, 3.5, Appendix 3, 10, 15 |
| May 2024 | SH(NA)-081257ENG-AC | ■Added function<br>Extension for simple device communication function<br>■Added or modified parts<br>Section 1.2, 1.4, 1.7, 1.10, 1.11, 1.14, 1.15, 1.16, 2.3, 3.3, 3.4, 3.5, Appendix 3, 7, 15 |
| October 2024 | SH(NA)-081257ENG-AD | ■Added or modified parts<br>RELEVANT MANUALS, Section 1.10, 1.14, 1.15, 1.16, 2.3, 3.3, Appendix 15 |
| October 2025 | SH(NA)-081257ENG-AE | ■Added or modified parts<br>RELEVANT MANUALS, Section 1.2, Appendix 10, 15 |
| April 2026 | SH(NA)-081257ENG-AF | ■Added or modified parts<br>SAFETY PRECAUTIONS, INTRODUCTION, RELEVANT MANUALS, Section 1.3, 1.7, 1.9, 1.10, 1.12, 1.14, 1.17, 2.2, 2.3, 3.1, 3.2, 3.4, 3.5, 3.6, 3.7, 3.8, Appendix 6, 7, 15 |

| Revision date | *Manual number | Description |
|---|---|---|
| October 2026 | SH(NA)-081257ENG-AG | ■Added or modified parts<br>Section 1.3, 1.4, 1.5, 1.14, 1.15, 1.17, 1.18, 2.2, 3.3, Appendix 2, 3, 4, 5, 16<br>■Renumbered<br>Appendix 5 to 15 → Appendix 6 to 16 |

Japanese manual number: SH-081253-AJ

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

© 2014 MITSUBISHI ELECTRIC CORPORATION

# WARRANTY

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

## 1. Gratis Warranty Term and Gratis Warranty Range

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the sales representative or Mitsubishi Service Company.

However, if repairs are required onsite at domestic or overseas location, expenses to send an engineer will be solely at the customer's discretion. Mitsubishi shall not be held responsible for any re-commissioning, maintenance, or testing on-site that involves replacement of the failed module.

[Gratis Warranty Term]

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

[Gratis Warranty Range]

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

(2) Even within the gratis warranty term, repairs shall be charged for in the following cases.

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

## 2. Onerous repair term after discontinuation of production

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

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

## 3. Overseas service

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

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

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

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

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

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

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

## 5. Changes in product specifications

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

# INFORMATION AND SERVICES

For further information and services, please contact your local Mitsubishi Electric sales office or representative.
Visit our website to find our locations worldwide.

MITSUBISHI ELECTRIC Factory Automation Global Website
Locations Worldwide
https://www.MitsubishiElectric.com/fa/about-us/overseas/

# TRADEMARKS

Microsoft and Windows are trademarks of the Microsoft group of companies.
Oracle and JavaScript are registered trademarks of Oracle and/or its affiliates.
Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.
The company names, system names and product names mentioned in this manual are either registered trademarks or trademarks of their respective companies.
In some cases, trademark symbols such as '™' or '®' are not specified in this manual.


SH(NA)-081257ENG-AG

TRANSLATION OF THE ORIGINAL INSTRUCTIONS

SH(NA)-081257ENG-AG(2610)MEE
MODEL: R-ETHER-U-OU-E

**MITSUBISHI ELECTRIC CORPORATION**

HEAD OFFICE: TOKYO BLDG., 2-7-3, MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
NAGOYA WORKS: 1-14, YADA-MINAMI 5-CHOME, HIGASHI-KU, NAGOYA 461-8670, JAPAN

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

Specifications subject to change without notice.