MITSUBISHI ELECTRIC
Changes for the Better

PLC
Ethernet

This course is for participants who will use a MELSEC-Q series Ethernet module for the first time.

* Ethernet is a registered trademark of Xerox Corp.

Introduction Purpose of This Course

This course is designed to provide basic knowledge regarding Ethernet modules for those who use the MELSEC-Q series Ethernet modules for the first time.

This course should give a participant a better understanding of the Ethernet module's data exchange formats, specifications, settings, and start-up procedure.

This course requires the basic knowledge of FA networks, MELSEC-Q series programmable controllers, sequence programs, and GX Works2.
Taking the following courses is recommended before starting this course.

1. MELSEC-Q Series Basics Course
2. GX Works2 Basics Course
3. Intelligent Function Module Course

Introduction Course Structure

The contents of this course are as follows.
We recommend that you start from Chapter 1.

Chapter 1 - Ethernet Overview

Explains the Ethernet data communication basics.

Chapter 2 - Example System Confirmation and System Configuration

Explains the network configuration for Ethernet, and Ethernet module specifications and settings.

Chapter 3 - Initial Configuration

Explains the Ethernet module operation procedures from the start-up to the operation test, using an example system.

Chapter 4 - Troubleshooting

Explains the network diagnosis procedure for the case of a failure.

Final Test

Passing grade: 60% and higher.

Introduction How to Use This e-Learning Tool

Go to the next page		Go to the next page.
Back to the previous page		Back to the previous page.
Move to the desired page		"Table of Contents" will be displayed, enabling you to navigate to the desired page.
Exit the learning		Exit the learning. Window such as "Contents" screen and the learning will be closed.

Introduction Cautions for Use

Safety precautions

When you learn by using actual products, please carefully read the safety precautions in the corresponding manuals.

Precautions in this course

- The displayed screens of the software version that you use may differ from those in this course.

  This course uses the following software version:

  - GX Works2 Version 1.493P

Chapter 1 Ethernet Overview

Chapter 1 provides an overview of the Ethernet data communication.

Ethernet is essential for daily information communications which occur via the factory's LAN, etc.

This course explains how an Ethernet module can exchange information with the CPU module and other Ethernet compatible devices.

To learn more about the data used for system control, please take the following courses:
CC-Link IE Controller Network, CC-Link IE Field Network, and CC-Link Network courses

To learn more about RS-232 and RS-422 serial interfaces used for electronic scales, temperature controllers, and bar code readers, etc., please take the following course:
Serial Communication Course

1.1 Ethernet in the FA Environment

There are two main network types in an FA environment: an "information network" and a "control network".

Information network

In information network, computers are usually used to send and collect information.
Typically, a large amount of information is transmitted by taking relatively a long time ranging from several minutes to several hours.
Information network is used to send production instructions to a production site and to receive production status reports from a production site.

Network example: Ethernet

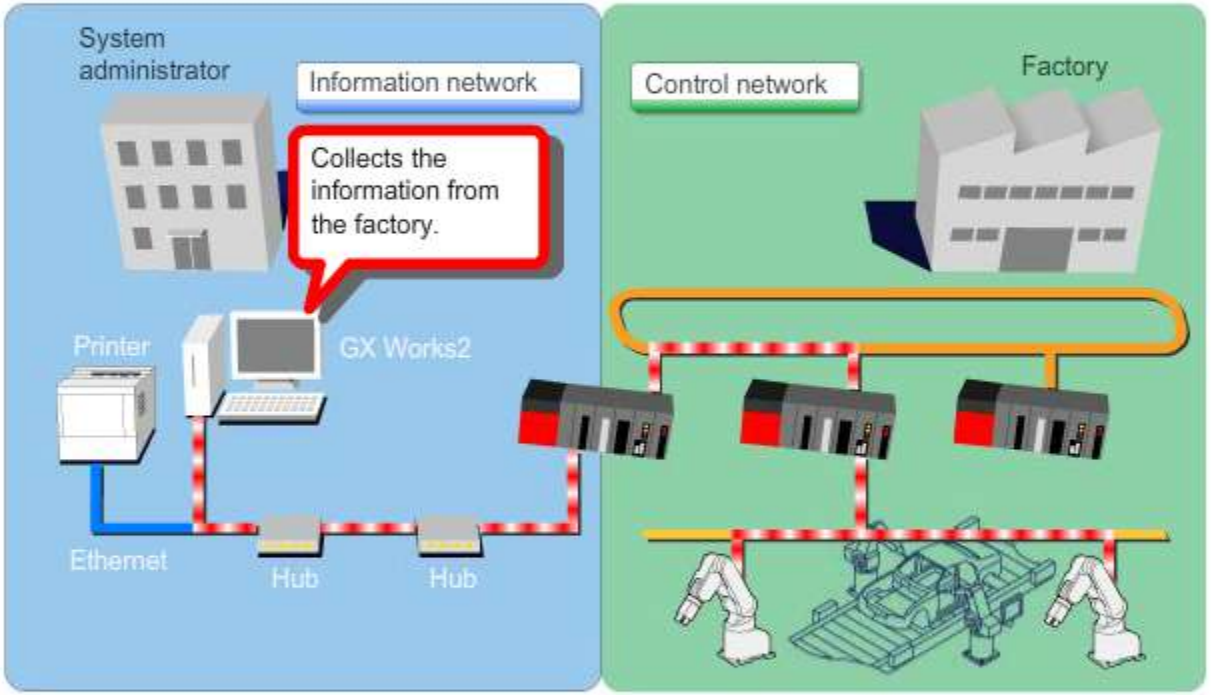
Control network

In control network, programmable controllers are usually used to send and collect information at bit and word format.
Typically, synchronization between information and an assembly line operation is required, therefore relatively a small amount of information is sent in a reliable manner in an interval of milliseconds.
Control network is used to transmit on/off statuses of sensors and actuators, workpiece position information, and the rotation speed of motors, etc.

Network examples: CC-Link IE Controller Network, CC-Link IE Field Network, CC-Link Network

1.1 Ethernet in the FA Environment

Ethernet is one of the information network standards.
With the increasing need for information links between factories and offices in recent years, Ethernet is gaining popularity as a network standard for sending instructions to the factory floor, and for receiving production status reports.

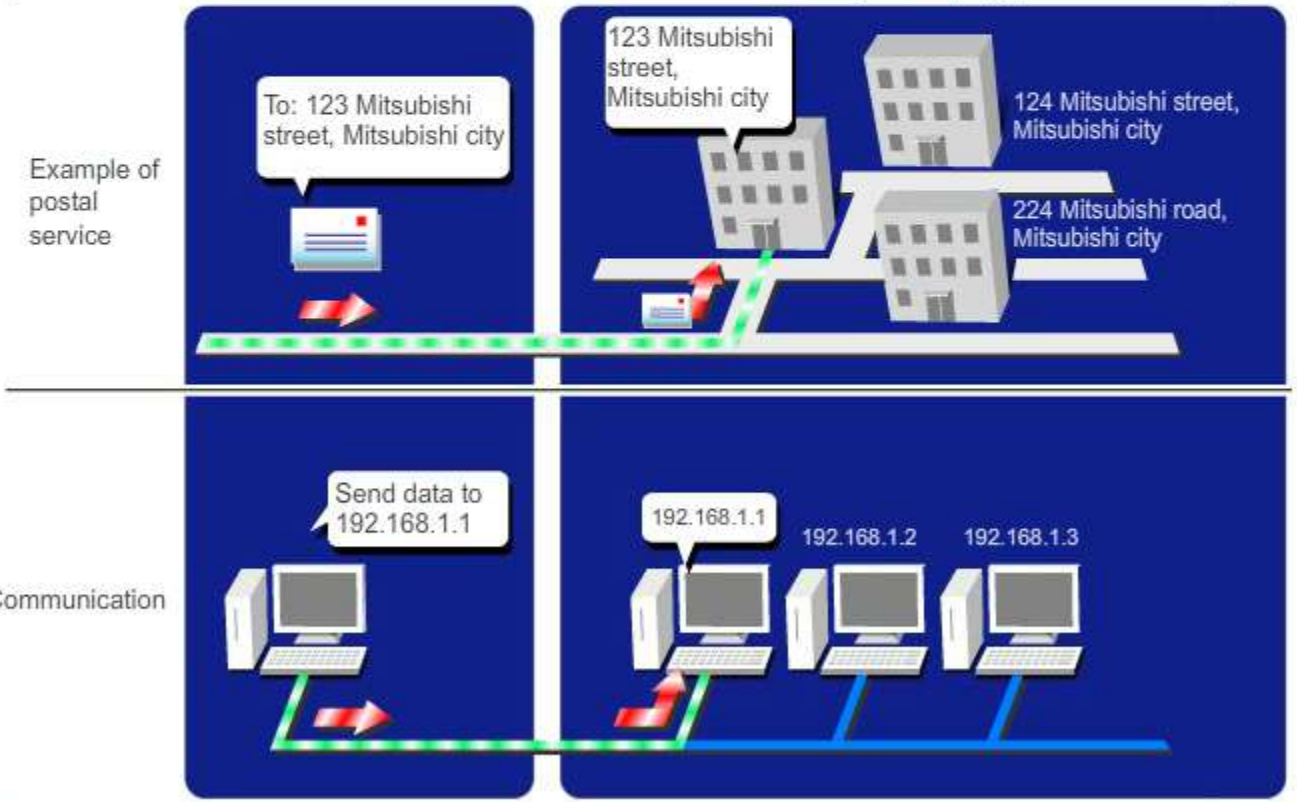


1.2 Ethernet Basics

This section explains the TCP/IP, which are the protocols widely deployed by Ethernet.
For devices to communicate, both the communication source and destination devices must be defined.
As shown in the animation below, these are similar to the sender's address and the receiver's address on an envelope.

1.2.1 IP address

IP communication is the foundation of TCP/IP communication. In IP communication, each communication device is identified by its IP address (Internet Protocol address). Normally, these addresses are expressed in decimal and are divided into four 8-bit sections by dots (e.g., "192.168.1.1").

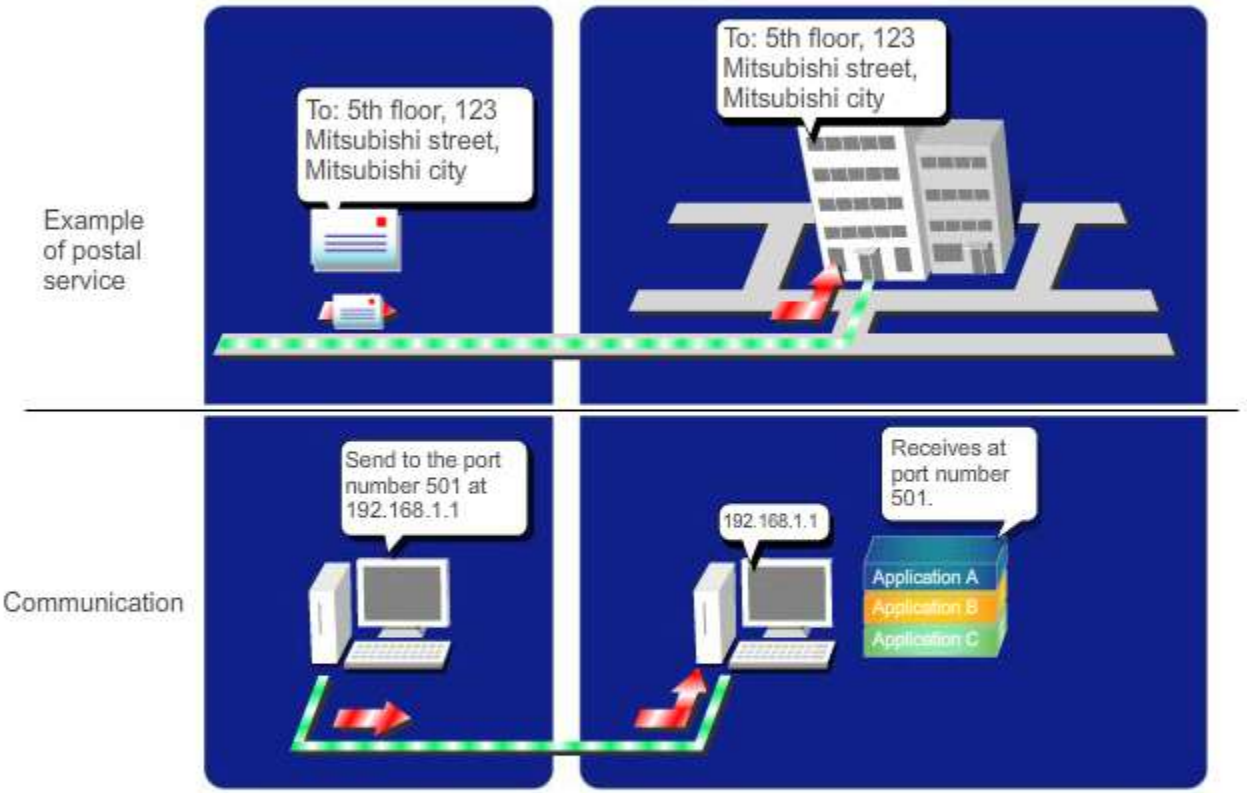


Note:

An IP address is not an arbitrary address. When connecting a device to an existing network, please consult the network administrator to assign an IP address.

1.2.2 Port number

The actual communication occurs between the applications running on the devices and computers.
In IP communication, the communicating applications are identified by their port numbers.
In the earlier postal service example, an IP address is the "street address", and a port number is the "floor number".



The port number ranges from 0 to 65535 (0 to FFFF). Of this, 0 to 1023 (0 to 3FF) are called "Well Known Port Numbers" and unique to each application program. (For example, the email recipient port number is 25, the homepage reference port number is 80, and the file transfer port numbers are 20 and 21, etc.).
For the communication between programmable controllers that are not associated with application programs, the port numbers 1025 to 65534 (401 to FFFE) are used.
* Port numbers are expressed in decimal in this section. The values shown in parentheses are in hexadecimal.

1.2.3 Communication methods

There are two main Internet protocol types: Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). Data which is sent by TCP can only be received at a TCP port. The features of these two protocols are described below.

Protocol name	Description
TCP	A highly reliable 1:1 communication format. Before sending any data, the connection with the other device is established. This protocol is suitable for applications in which reliable data transmissions are required.
UDP	Data from an application is simply sent to the specified destination. Transmissions occur in high speed because of its simple protocol. This protocol is suitable for applications such as a real-time monitor of a personal computer.

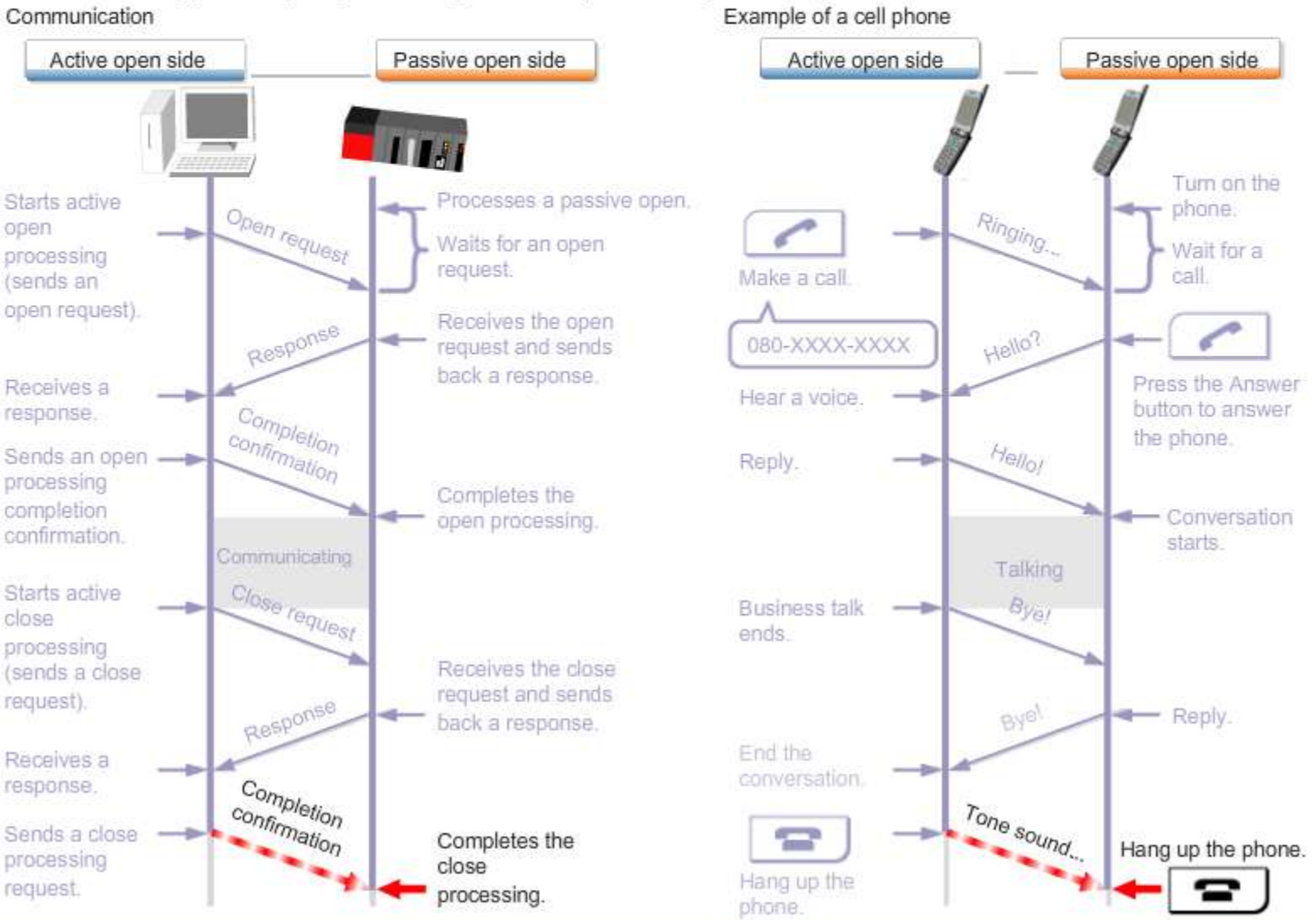
Item	TCP	UDP
Reliability	High	Low
Processing speed	Slow	Fast
Connection with the other device(s)	1:1	1:1 or 1:n
Data reception assurance	Yes	No
Operation at transmission error	Re-transmits automatically (according to the setting)	No retransmission (packet discarded)
Connection establishment *1	Required	Not required
Flow control	Yes	No
Congestion control (retransmission control) *2	Yes	No

*1: "Connection establishment" is explained in the "open/close processing" section.
*2: **"Congestion"** refers to a traffic jam of communication packets in the network.

All the examples given in this course are based on the **TCP** protocol.

1.2.4 Open/close processing

In TCP/IP communication, a dedicated connection (logical line) is established between the own device and its communicating device (other device).
Opening (establishing) this line is referred to as "open processing", and disconnecting the line is referred to as "close processing".
There are two types in open processing: "active open" and "passive open".



1.2.4 Open/close processing

The active/passive open type is determined depending on which device has the open authority.
For example, if a personal computer's program has an open processing program for an Ethernet module, the Ethernet module performs a passive open.

Open processing

- **Active open**

  An active open is requested to the other device which is in a passive open (unpassive/fullpassive) condition. In an example of a cell phone, this is equivalent to making a call to a recipient.

- **Passive open**

  In the passive open condition, the own device waits and receives an open request.
  In an example of a cell phone, this is equivalent to the standby mode being able to receive a call. There are two types of passive open: fullpassive open and unpassive open.

Fullpassive open	The own device accepts an active open request only from a specific network-connected device. In an example of a cell phone, the phone accepts incoming calls only from the party registered in its telephone directory.
Unpassive open	The own device accepts an active open request from any network-connected devices. In an example of a cell phone, the phone accepts any incoming calls including anonymous calls.

1.2.4 Open/close processing

Close processing

Close processing is an operation of disconnecting the connection (logical line), which has been established by open processing, with the other device. After completing the close processing, that connection line becomes available for another device.
In an example of a cell phone, "close processing" is the equivalent to hanging up a call following a conversation.

Open/close processing summary

If the Ethernet module has been set as the active open device, its communicating device (other device) will be set as the passive open device.
If the other device's specification is fixed, the settings of the Ethernet module must be adjusted as shown in the table below.

<table>
<tr><th>Communication protocol</th><th colspan="2">Own device</th><th colspan="2">Other device</th></tr>
<tr><td rowspan="3">TCP</td><td colspan="2" rowspan="2">Active open</td><td rowspan="2">Passive open</td><td>Fullpassive open</td></tr>
<tr><td>Unpassive open</td></tr>
<tr><td rowspan="2">Passive open</td><td>Fullpassive open</td><td colspan="2" rowspan="2">Active open</td></tr>
<tr><td>Unpassive open</td></tr>
<tr><td>UDP</td><td colspan="2">None</td><td colspan="2">None</td></tr>
</table>

1.3 Summary

In this chapter, you have learned:

- Ethernet in the FA environment
- Ethernet basics

Important points

Ethernet in FA environment	Ethernet is an information network for transmitting a large volume of data by taking relatively a long time.
Ethernet communication protocols	TCP and UDP are two main protocols (rules) used for communication between devices. • TCP is suitable for applications of which data must be transmitted in a highly reliable manner. • UDP is suitable for real-time monitoring applications, etc.
Open/close processing by TCP/IP	• TCP's virtual dedicated line is called a "connection", and opening this connection is called "open processing". • UDP does not require open processing. • Two types of open processing are active open and passive open. • The open processing types must be set correctly in order for the devices to establish a connection.

Chapter 2 Example System Confirmation and System Configuration

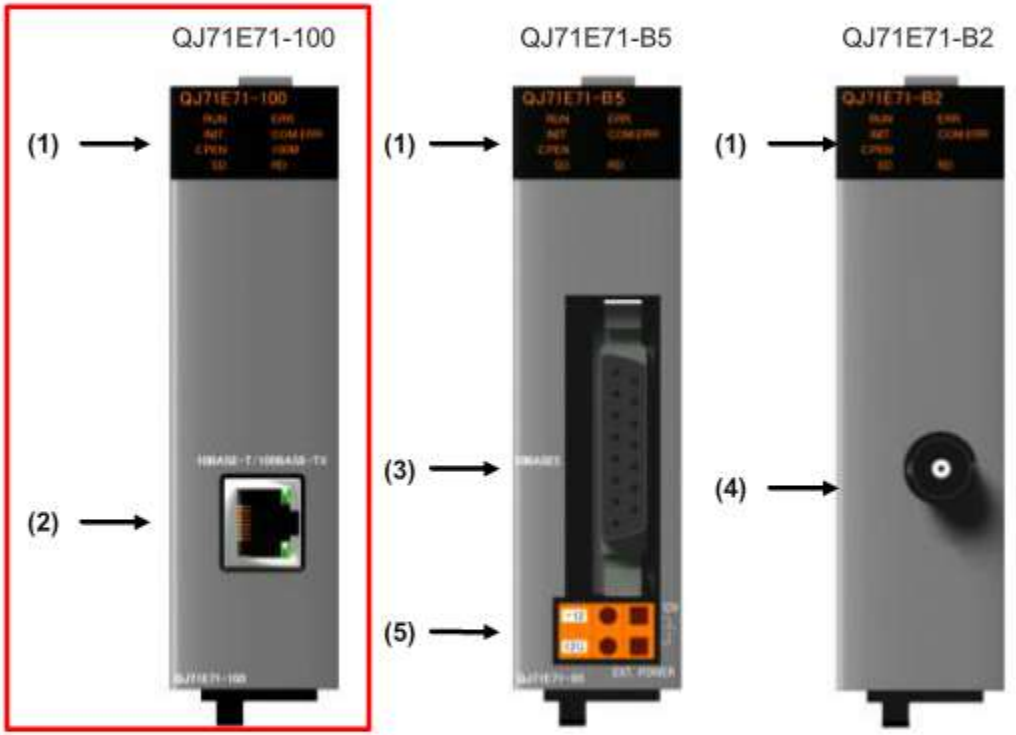
Chapter 2 explains an Ethernet network configuration, and the Ethernet module specifications and settings.

To configure an Ethernet network with programmable controllers, an Ethernet module must be used.
The previous chapter explained the TCP/IP on which communications are based.
This chapter explains the TCP/IP-based data communication procedure for programmable controllers.

2.1 Module Types and Component Names

Depending on the communication cables (media) used, an appropriate Ethernet module must be selected.

Component names and functions



There are two main cable types: twisted-pair and coaxial. The twisted-pair cable (LAN cable), with fast transmission speed and easily installable features, is more popular in recent years. For the twisted-pair cable, only the QJ71E71-100 Ethernet module is compatible. This course uses the QJ71E71-100 as an example.

Although the QJ71E71-B5 and QJ71E71-B2 modules have different hardware, their parameter settings and programming, etc., are the same with those of the QJ71E71-100 module.

No.	Name	Function
(1)	LED indicator	Indicates the module statuses.
(2)	10BASE-T / 100BASE-TX connector	Connector which connects the Ethernet module to the 10BASE-T / 100BASE-TX.
(3)	10BASE5 connector	Connector for the 10BASE5 AUI cable (transceiver cable).
(4)	10BASE2 connector	Connector which connects to 10BASE2 (coaxial cable).
(5)	External power supply terminal	Power supply terminal for supplying power to the transceiver (13.28V to 15.75V).

2.2 Communication Methods

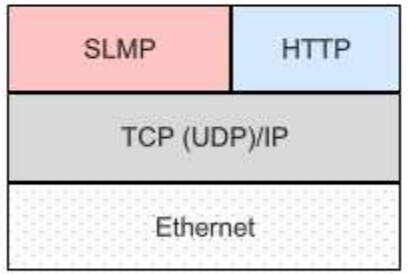
Data communication methods

There are three main communication methods available to an Ethernet module: "predefined protocol", "communication by fixed buffer", and "communication by random access buffer".

Although the Ethernet module has other communication functions such as E-mail function and web function, this course will focus on SLMP and predefined protocol support function.

Predefined protocol *1	SLMP	A type of communication protocol that allows an SLMP compatible external device to access an Ethernet module, etc.
	The send/receive messages to/from a SLMP compatible device can be created using the predefined protocol support function of the Ethernet module.	
Fixed buffer (passive)	Sequence program and personal computer programs that are saved in the pre-set area are sent to or received from a pre-set area of the other device.	
Random access buffer (passive)	Programmable controllers and other devices, such as a personal computer, access a common area to deposit or retrieve data.	

*1: The content which has been explained so far can be represented by the hierarchy shown on the right. As shown, the communication protocols exist above TCP/IP.
An example of communication protocols is an HTTP (HyperText Transfer Protocol), which is used to view web pages. The SLMP (SeamLess Message Protocol), which is used to access programmable controllers, is on the same level with HTTP.



SLMP: SeamLess Message Protocol. Using the messaging procedure established by CLPA (CC-Link Partner Association), data requests and response messages are transmitted seamlessly across different networks.
Active: A device that sends requests. In an IT system, this is a client computer, which requests information to a server computer.
Passive: A device that waits for requests. In an IT system, this is a server computer, which waits requests from a client computer.

2.3 Example System Operations

This section explains the example system used in this course.

The example system consists of "System A", which controls the factory's manufacturing line, and "System B", which manages the production system in the head office. The two systems are connected to each other by Ethernet.

The daily production target is saved in data register "D1000" in the head office's System B. Every day, at the factory production start (System A commencement), System A accesses System B at the head office and retrieves the production target set for the day.

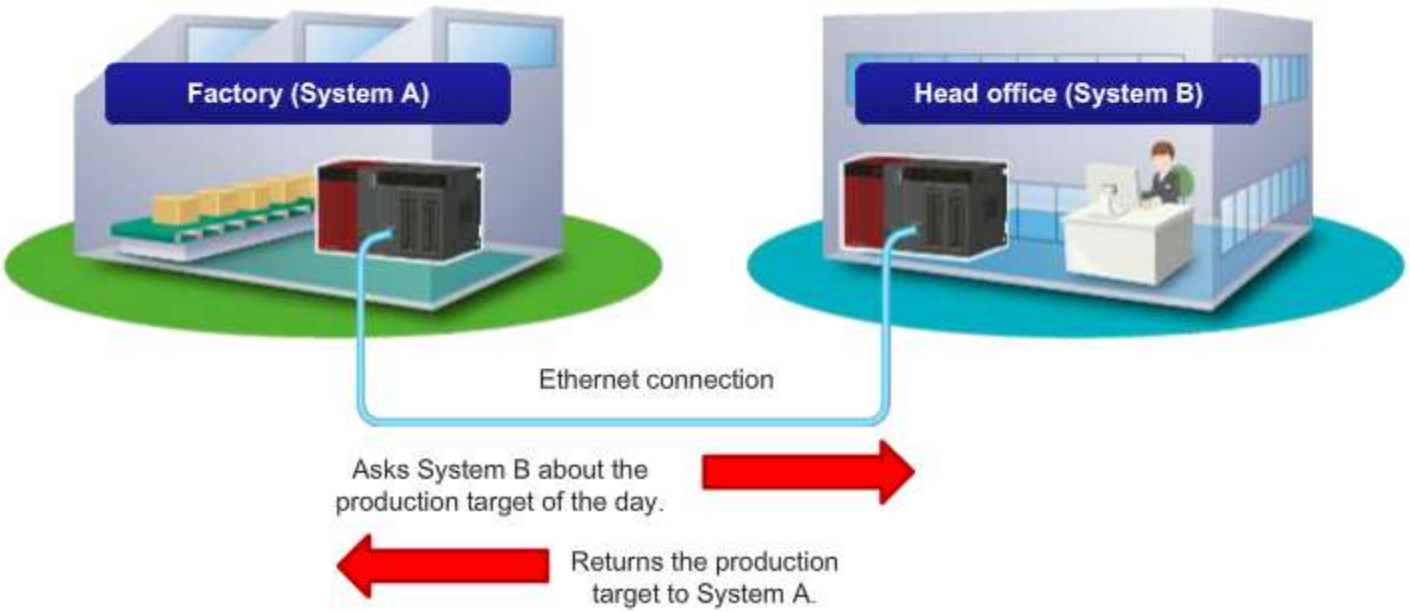
The communication protocol "SLMP" is used for data communication between System A and System B.

SLMP request side

- Active operation (Active open)
- Station No.: 1
- IP address: 192.168.0.2

SLMP response side

- Passive operation (Passive: Fullpassive open)
- Station No.: 2
- IP address: 192.168.0.3



2.4 Communication by SLMP

When devices are communicating by SLMP, the data requesting side and responding side communicate with each as shown below.

Connection establishment
Waits for an open request from the requesting side (passive open).

Connection establishment
Sends an open request to the responding side (active open).

Connection establishment
Receives the open request from the requesting side.

Message transmission
Once the connection is established, sends out data to the responding side.

Message execution
Executes processing such as reading/writing, etc., in accordance with the data sent from the requesting side.

Repeat if necessary

Response message reception
Receives the response from the responding side, and checks the execution result.

Response message return
After completing the execution, reruns a response to the requesting side.

Connection end
Sends a close request to the responding side, and ends the communication operation at the requesting side.

Connection end
Receives the close request from the requesting side, and ends communication operation at the responding side.

2.4.1 SLMP request and response messages

In SLMP, message units called "frames" are used. As shown below, an SLMP frame consists of several packets assembled in a specific format.

SLMP request

This is the format for sending a request from the requesting side to the (SLMP compatible) responding side.

*** In this course, "Request destination" in tables below denotes the SLMP responding side.**

Header	Subheader	Network number	Station number	Request destination* module I/O number	---	Request data length	Monitor timer	Request data

More details will be explained on the next page.

SLMP response

This is the format for returning a response from the (SLMP compatible) responding side to the requesting side.
There are two response types: One in which the responding side operation ended normally, and one in which the operation ended in error.
If the operation ended in error, an error code is saved at the "End code".

Normal end

Header	Subheader	Network number	Station number	Request destination module I/O number	---	Response data length	End code	Response data

Error end

Header	Subheader	Network number (Access station)	Station number (Access station)	Request destination module I/O number	---	Response data length	

	End code	Network number (response station)	Station number (response station)	Request destination module I/O number	---	Command	Sub command

2.4.1 SLMP request and response messages

The table below lists frame elements that require settings by the user.
For these elements, the "devices to read data" and the "devices to store data" must be set.
For details regarding the device assignment, please refer to Section 3.4.3.

Element		Packet type	Description
Header		Send/receive	Ethernet, TCP/IP, UDP/IP headers are automatically saved.
Subheader	Serial number	Send/receive	Set a serial number to relate a request with its corresponding response. (Optional)
Network number		Send/receive	Set the responding side's network number.
Station number		Send/receive	Set the responding side's station number.
Request destination module I/O number		Send/receive	Set the I/O numbers of the responding side CPU module.
Monitor timer		Send	Set the wait time for completing reading/writing at the responding side.
Request data *	Start device number	Send	Set the start device number of the responding side's device area where reading/writing is executed.
	Device code	Send	Set the type of the responding side device (X, Y, M, D, etc.) where reading/writing is to be executed.
	Number of device points	Send	Set the "number of device points" of the other device where reading/writing is to be executed.
Response data		Receive	Set the area to save the response received from the responding device.
Request data	Write data	Send	Set the area to save the write data to be sent to the responding side.
End code		Receive (error receive)	Set the area to save the error code received from the responding side.

* "Request data" includes the following elements: command, sub command, start device number, device code, number of device points, and the write data. The details of "command" and "sub command" are explained on the following page.

2.4.2 SLMP commands

A frame contains an SLMP command that specifies an operation to be performed at the (SLMP compatible) responding side.

The table below lists SLMP command examples.
The examples include a command for reading data from the responding side CPU module device, and a command for writing data in a device.

Item		Command	Sub Command	Description
Type	Operation			
Device	Read	0401	00□1	Reads values from the specified bit device in 1-bit units.
			00□0	• Reads values from the specified bit device in 16-bit units. • Reads values from the specified word device in 1-word units.
	Write	1401	00□1	Writes values to the specified bit device in 1-bit units.
			00□0	• Writes values from the specified bit device in 16-bit units. • Writes values from the specified word device in 1-word units.
Clear Error		1617	0000	Turns off the Ethernet module's "COM.ERR." LED indicator.

The □ part of the sub command varies according to the device being specified.

2.5 Summary

In this chapter, you have learned:

- Module types and component names
- Communication methods
- Example system operations
- Communication by SLMP

Important points

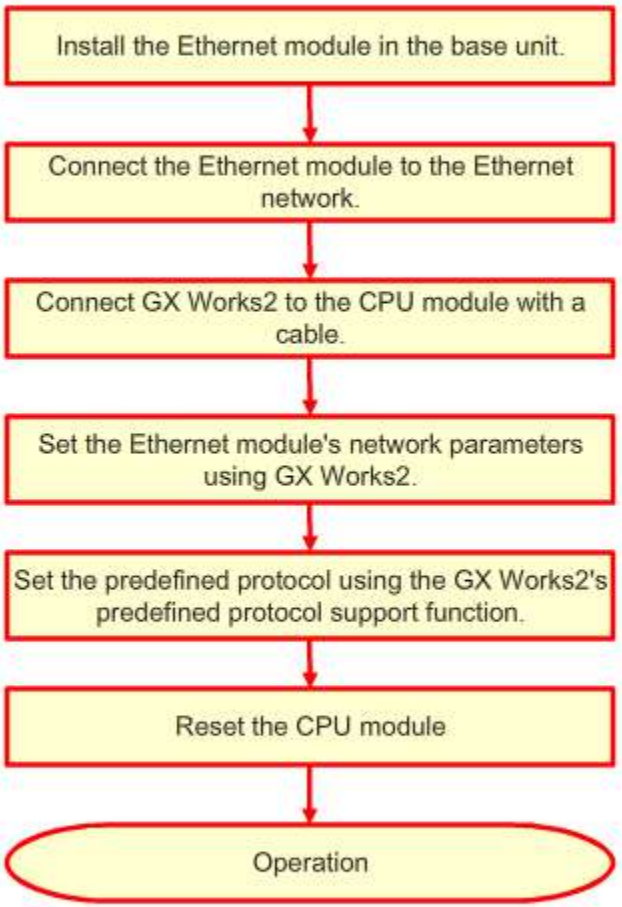
Data communication methods	"Predefined protocol", "fixed buffer communication", "random access buffer communication" are the main data communication methods.
SLMP	SLMP communication procedure, and the message frames and commands.

Chapter 3 Initial Configuration

Chapter 3 explains how to set up an Ethernet module for its first operation, especially the programing method using dedicated instructions.
By learning the system configuration, connection methods, and various setting operations for an Ethernet module, a participant will obtain the required knowledge to operate the module.

3.1 Pre-operation Settings and Setting Procedure

The settings and procedure which are performed prior to actual Ethernet module operation are shown below.

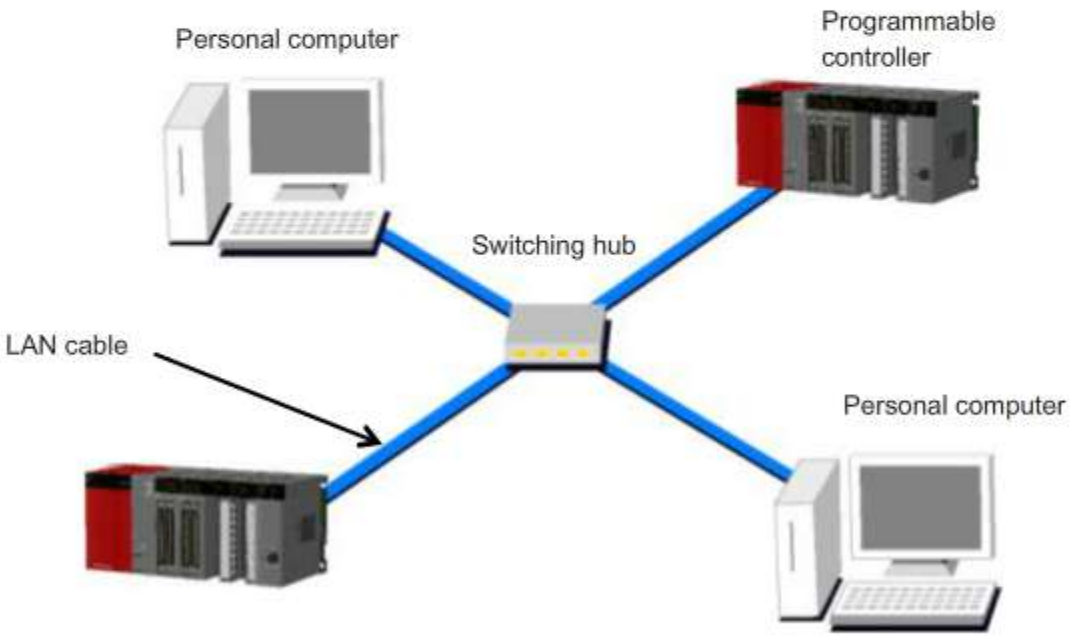


3.2 Connection Method

This section explains a connection example using the QJ71E71-100 Ethernet module.

3.2.1 Connecting the QJ71E71-100 Ethernet module

The connection example shown here is based on the QJ71E71-100 Ethernet module, which is the most popular Ethernet module. The connection configuration shown in the figure below is called a star type.
In this configuration, a switching hub is used to amplify signals and to control signal traffic.
In this configuration method, a failure in a device is unlikely to spread to the others.
Moreover, the required LAN cables are readily available.



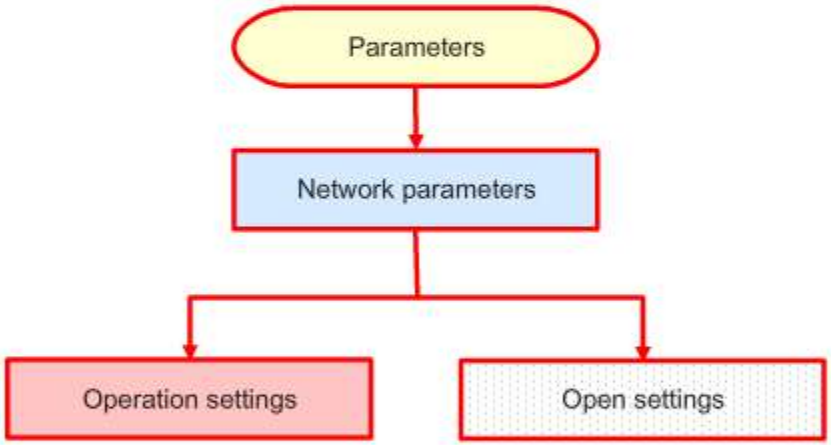
3.3 Parameter Settings

Parameters can be set using GX Works2.

Setting on GX Works2

The parameter setting function of GX Works2 allows the communication protocols to be set without any sequence program.
By simply setting parameters and writing them to the CPU module, a set of operations (for example, Ethernet module initial processing, open processing with the other device) can be performed automatically.

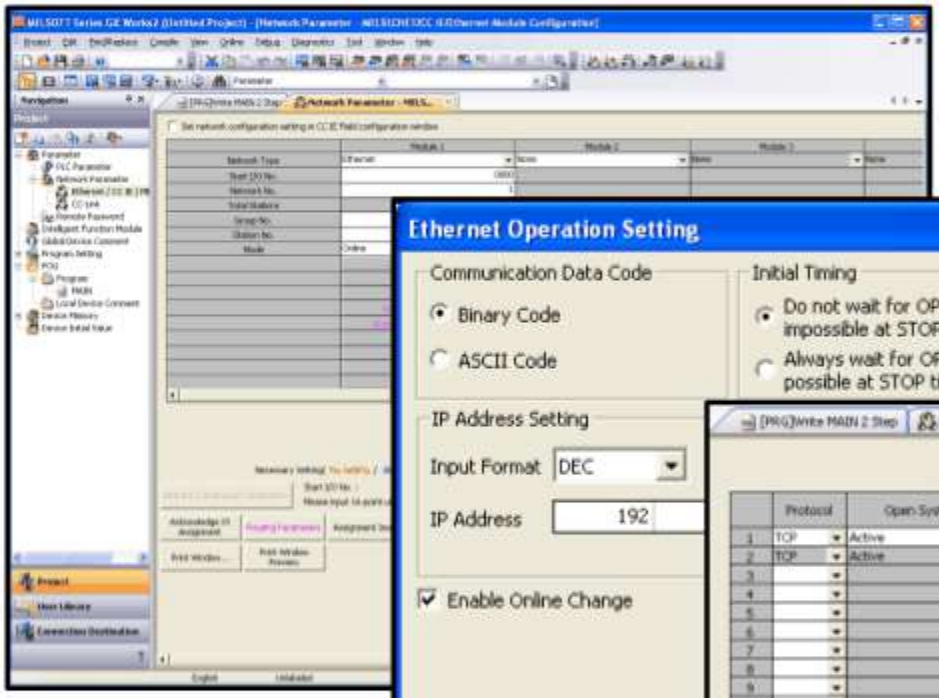
The diagram below shows the network parameter structure.



3.3.1 Network parameter settings

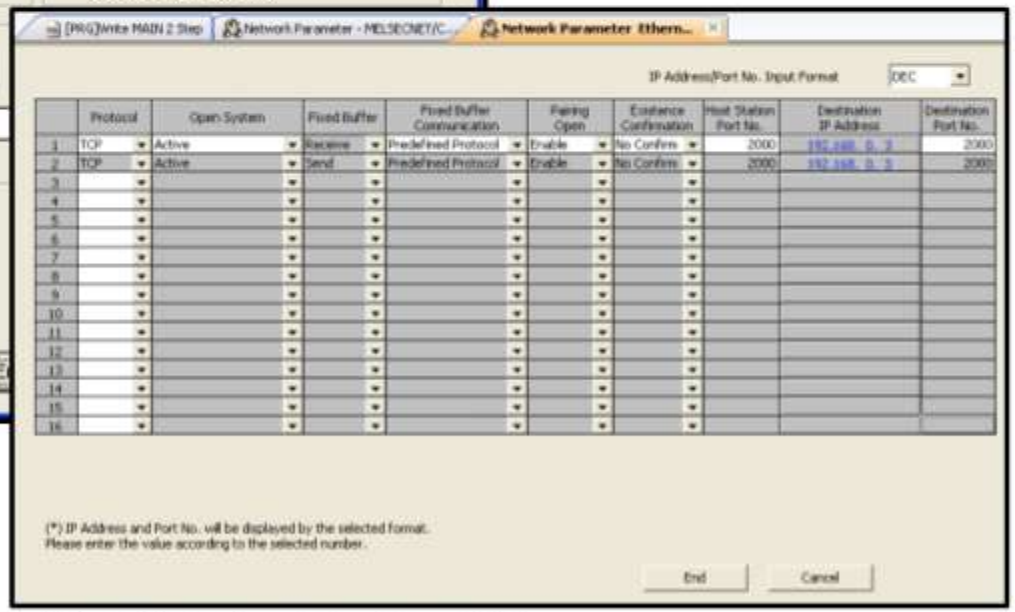
Network parameters

The setting windows are shown below.



Network Parameter window

Ethernet Operation Setting window



Network Parameter window (open setting)

3.3.1 Setting the network parameters

To set network parameters in GX Works2, open a Project, select [Network Parameter] – [Ethernet / CC IE / MELSECNET].

SLMP requesting side setting example (station No.1)

	Module 1
Network Type	Ethernet
Start I/O No.	0000
Network No.	1
Total Stations	
Group No.	0
Station No.	1
Mode	Online
	Operation Setting
	Initial Setting
	Open Setting

Network parameter setting area

Select "Ethernet".

If other networks (CC-Link IE Controller Network, CC-Link IE Field Network) exist, be sure to set a number different from their numbers.

Details are given in Section 3.3.2 Operation settings.

Details are given in Section 3.3.3 Open processing setting.

SLMP responding side setting example (station No.2)

	Module 1
Network Type	Ethernet
Start I/O No.	0000
Network No.	1
Total Stations	
Group No.	0
Station No.	2
Mode	Online

Network parameter setting area

This setting must be the same as that for station No.1.

3.3.2 Operation settings

The table below shows the settings required for an Ethernet module.

Bold fonts denote the default settings.

Item		Detail	Setting range / selections
Communication Data Code		Select the communication data code.	• **Binary code** • ASCII code
Initial Timing		Settings related to open timing.	• **Without open wait** • With open wait
IP Address Setting	Input Format	Select the IP address input format.	• **Decimal** • Hexadecimal
	IP Address	Set the own-station IP address.	– (default: "192.0.1.254")
Send Frame Setting		Select the send frame format.	• **Ethernet (V2.0)** • IEEE802.3
Enable Online Change		Permit/prohibit writing to the CPU module while the CPU module is running.	• Selected (permitted) • **Not selected (prohibited)**
TCP Existence Confirmation Setting		Select an alive check method in TCP communication.	• **Use the KeepAlive** • Use the Ping

For the example system of this course, the following settings are made.

Item		Setting value	
		SLMP requesting side	SLMP responding side
Communication Data Code		Binary code communication	
Initial Timing		Always wait for OPEN (communication possible at STOP time)	
IP Address Setting	Input Format	Decimal	
	IP Address	192.168.0.2	192.168.0.3
Enable Online Change		Selected	

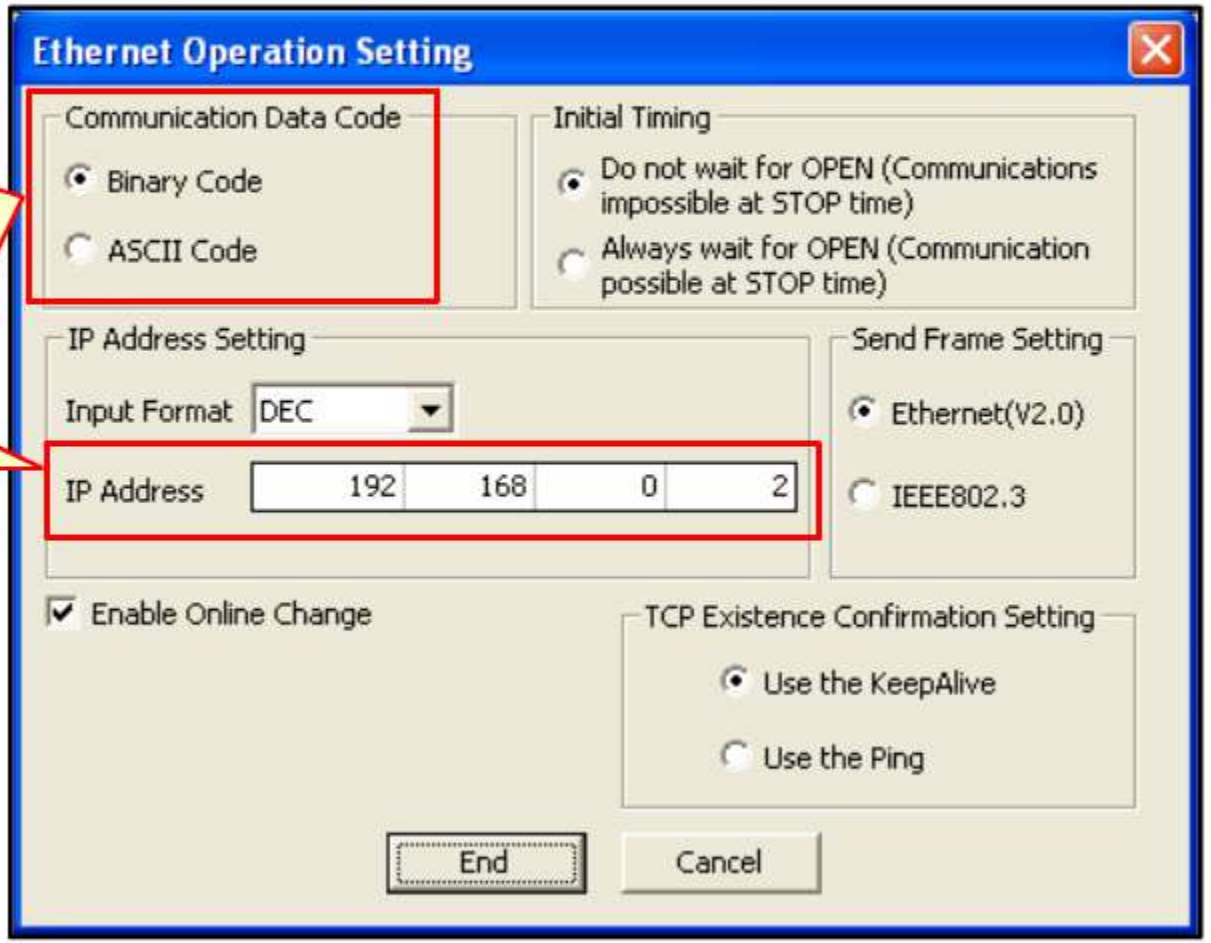
3.3.2 Operation settings

The operation setting window is shown below.

SLMP requesting side setting example

Select the communication data code for the communication with the other device.
The transmission/reception data amount with "Binary Code" is half of that with "ASCII Code". Selecting the latter reduces the load applied to the communication line.

Set the IP address of the requesting side.

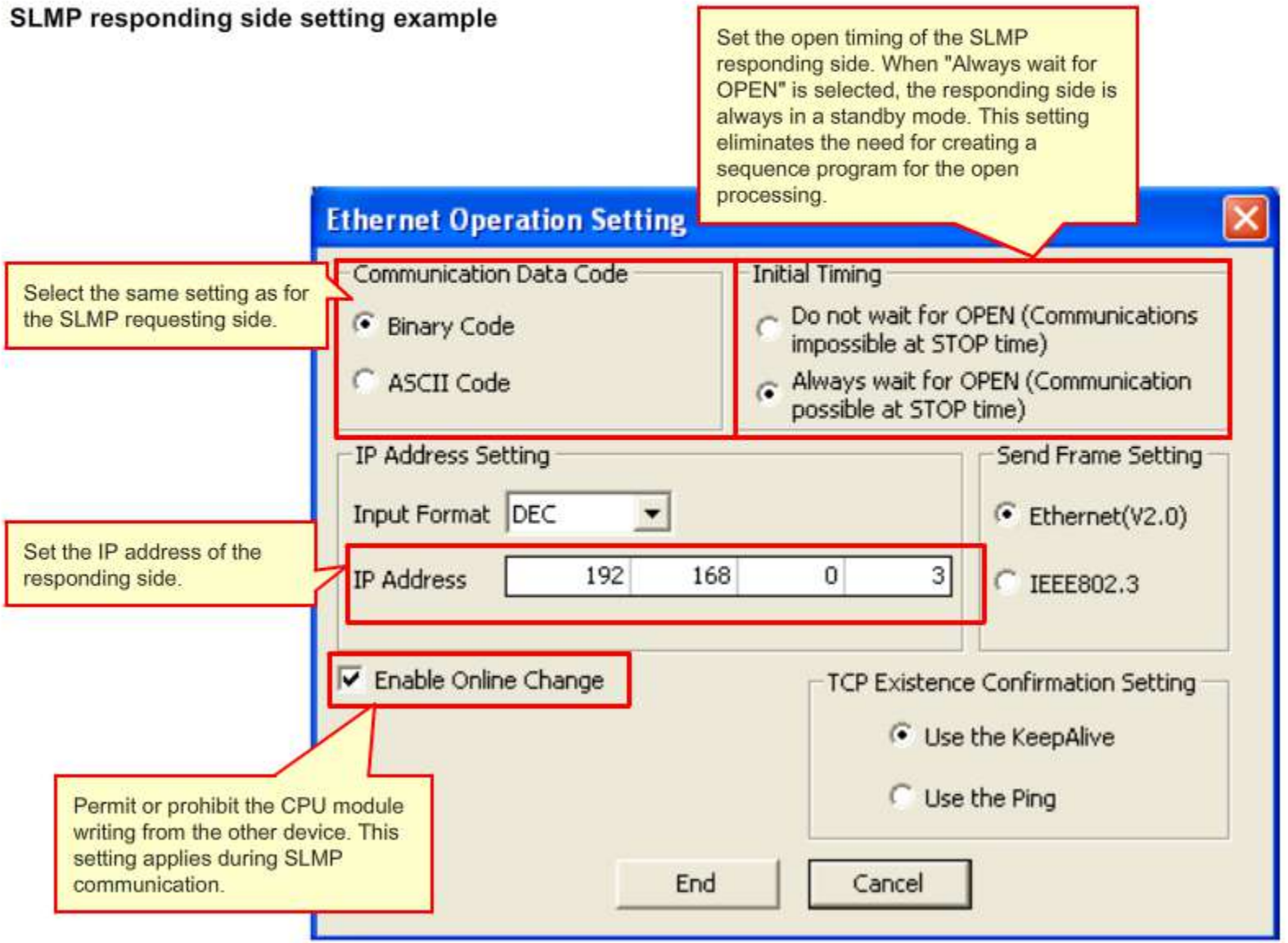


Ethernet Operation Setting window

3.3.2 Operation settings

The operation setting window is shown below.

SLMP responding side setting example



Ethernet Operation Setting window

3.3.3 Open processing settings

This section explains open processing settings required to exchange data with the communicating device.

SLMP requesting side setting example

OPEN Setting area

	Protocol	Open System	Fixed Buffer	Fixed Buffer Communication	Pairing Open	Existence Confirmation	Host Station Port No.	Destination IP Address	Destination Port No.
1	TCP	Active	Receive	Predefined Protocol	Enable	No Confirm	2000	192.168. 0. 3	2000
2	TCP	Active	Send	Predefined Protocol	Enable	No Confirm	2000	192.168. 0. 3	2000

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

SLMP responding side setting example

	Protocol	Open System	Fixed Buffer	Fixed Buffer Communication	Pairing Open	Existence Confirmation	Host Station Port No.	Destination IP Address	Destination Port No.
1	TCP	Fullpassive	Send	Procedure Exist	Disable	No Confirm	2000	192.168. 0. 2	2000
2									

(1)

OPEN Setting area

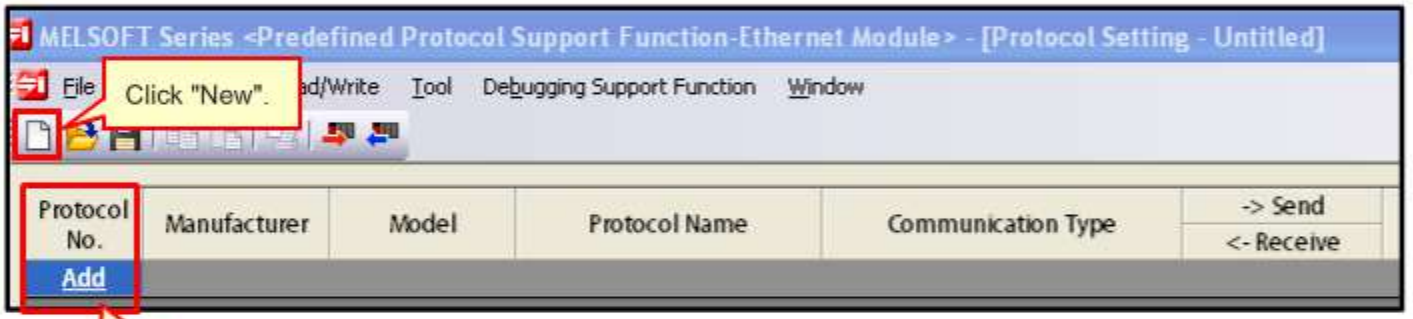
* In this example, the IP address and port number are specified in decimal.

No.	Item	Description
(1)	Protocol	Set the same protocol for the communicating device and for the own device.
(2)	Open System	Set this when "TCP" is selected at "Protocol". For the example system, the SLMP requesting side is set to "Active", and the SLMP responding side is set to "FullPassive".
(3)	Fixed Buffer	Select for which operation the fixed buffer is used for, "Send" or "Receive". For the SLMP responding side, "Send" is selected.
(4)	Fixed Buffer Communication	Select the communication method for the fixed buffer communications. For the SLMP responding side, "Procedure Exist" is selected.
(5)	Pairing Open	Select whether to use the pairing open for the fixed buffer communications. The receiving communication link and the transmitting communication link are handled as a pair, and the own station and the other station use a common port. This setting is made at the SLMP requesting side.
(6)	Existence Confirmation	Select whether to use the alive check function. Alive check is a function that sends a message to the other device to check if it is alive if no communication occurs for a preset time interval.
(7)	Host Station Port No.	Set the port number for the connection links. In this example, all are set to "2000".
(8)	Destination IP Address	Set the IP address of the other device.
(9)	Destination Port No.	Set the other device's port number. In this example, all are set to "2000".

3.4 Predefined Protocol Support Function

This function assists creating transmission/reception messages that are used with an SLMP compatible device. This section explains how to register a predefined protocol using the predefined protocol support function.

On the GX Works2 menu, select [Tools] – [Predefined protocol support function] – [Ethernet module] to open the predefined protocol support function.

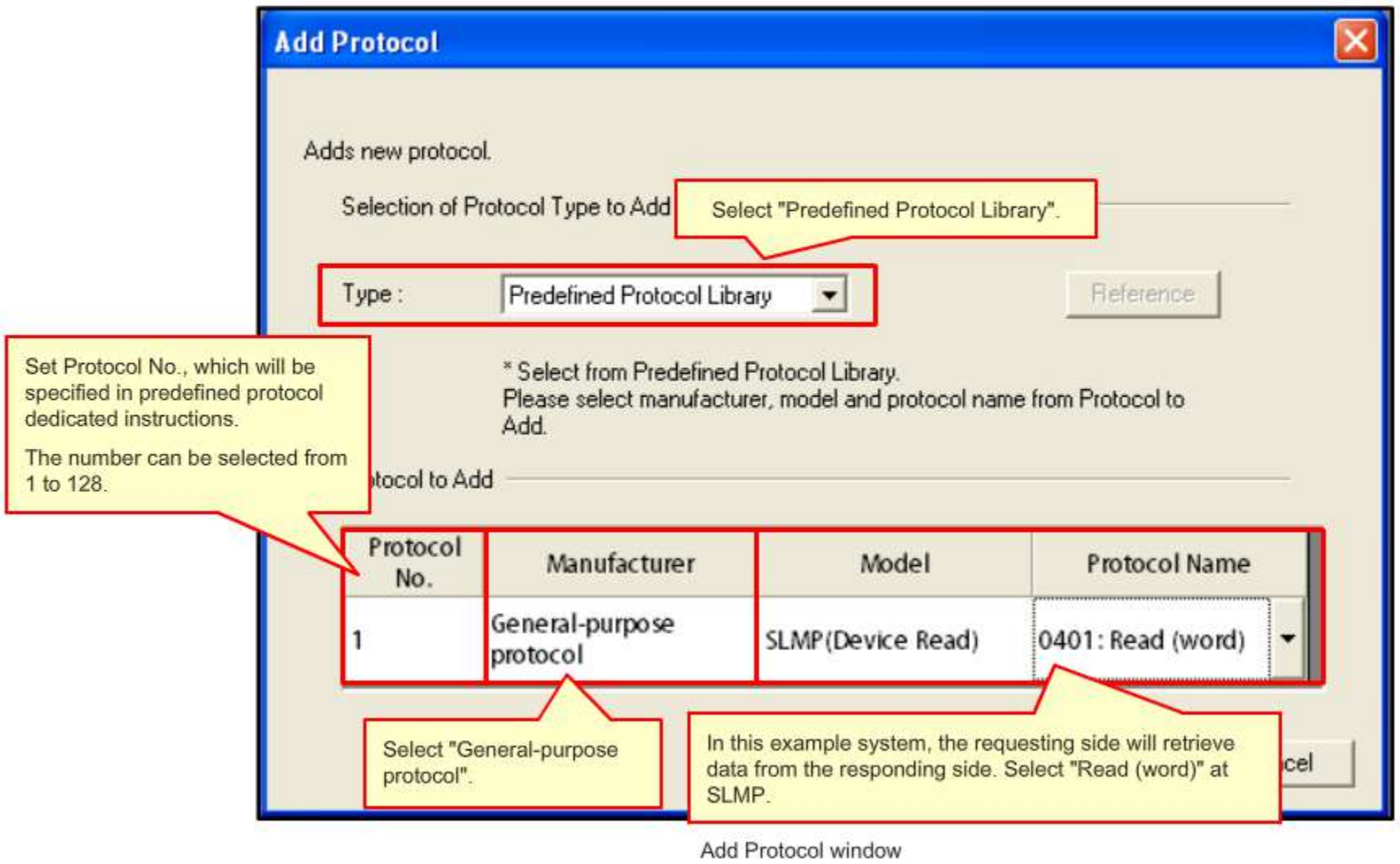


Protocol Setting window

Click "Add" to open the Add Protocol window.
Details are explained in Section 3.4.1.

3.4.1 Adding a protocol

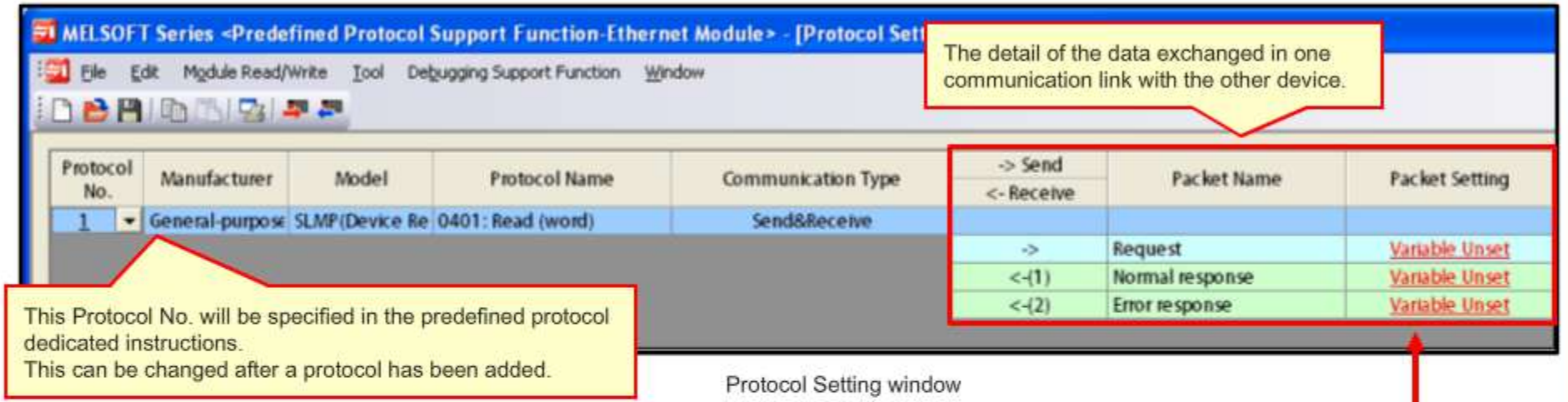
The "Add Protocol" setting window is shown below.



Add Protocol window

3.4.2 Protocol settings

The transmission/reception data details can be specified in the Protocol Setting window.



Protocol Setting window

The example system uses the "Device Read (word)" protocol, which is one of the selectable SLMP.
This protocol consists of the following three packets:

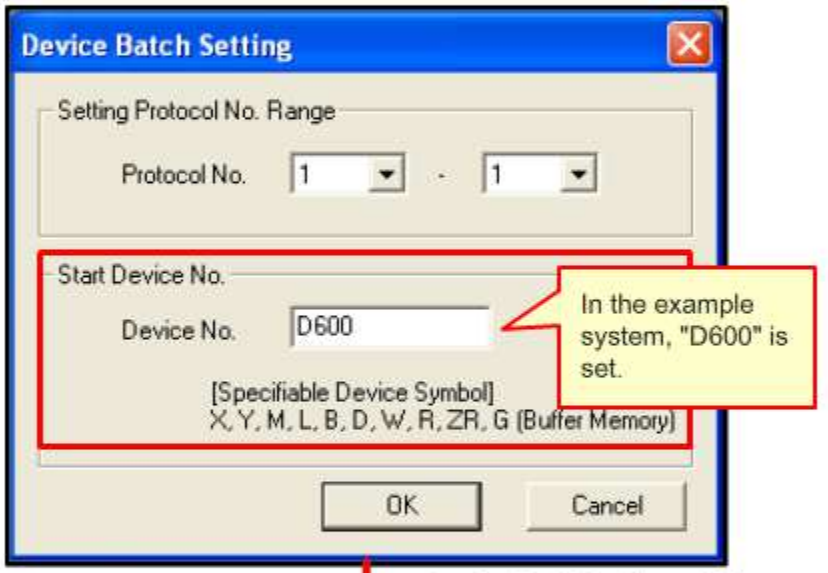
- Request
- Normal response
- Error response

For the packet which has not been set, "Variables Unset" is displayed in red. The details regarding the packet setting method are given on the following page.

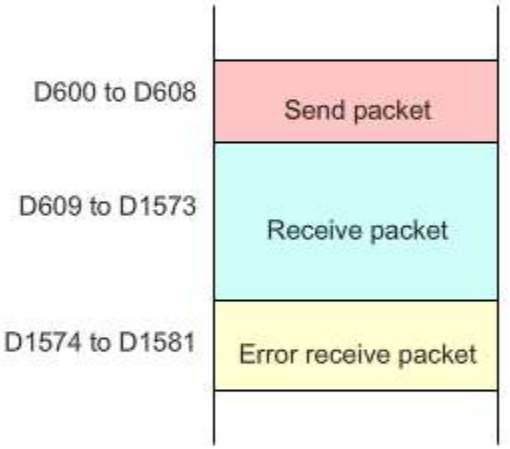
3.4.3 Packet settings

In packet setting, "device to read data" and the "device to store data" are set so that those settings can be used in programs.
"Device batch setting" of the predefined protocol support function enables batch setting of multiple devices.

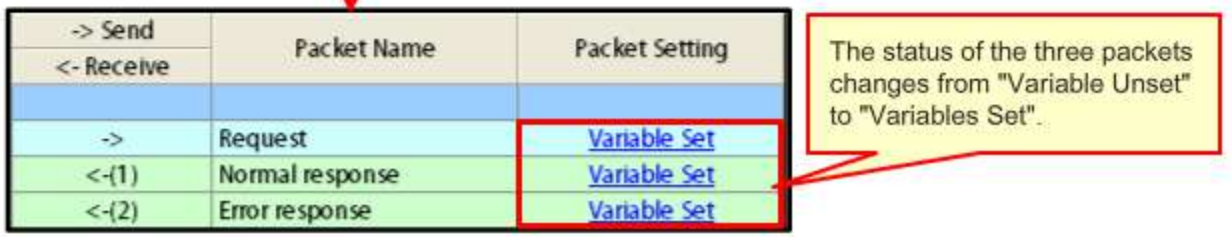
Select [Edit] – [Device Batch Setting] on the Predefined Protocol Support Function window, then enter the start device number.



Device Batch Settings window



Device assignment



Protocol Setting window

3.4.3 Packet settings

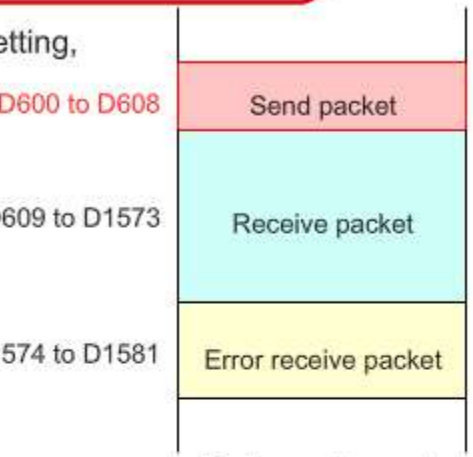
This section explains how the devices are automatically set as the result of the device batch setting, using the example system.

(1) Send packet

Packet Name	Packet Setting
Request	Variable Set
Normal response	Variable Set
Error response	Variable Set

Protocol Settings window

Click "Variable Set" at Request.



Protocol No. 1 Protocol Name 0401: Read (word)

Packet Type Send Packet Packet Name Request

D600 to D608, which are the send packet's data storage area, are automatically set

Element List

Element No.	Element Type	Element Name	Element Setting
1	Static Data	(Fixed data)	5400(2Byte)
2	Non-conversion Variable	Serial No.	[D600-D600](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
3	Static Data	(Fixed data)	0000(2Byte)
4	Non-conversion Variable	Network No.	[D601-D601](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
5	Non-conversion Variable	Station No.	[D602-D602](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
6	Non-conversion Variable	Requested module I/O No.	[D603-D603](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
7	Static Data	For future expansion	00(1Byte)
8	Length	Request data length	(Object element9-14/HEX/Reverse/2Byte)
9	Non-conversion Variable	Monitoring timer	[D604-D604](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
10	Static Data	Command	0104(2Byte)
11	Static Data	Subcommand	0000(2Byte)
12	Non-conversion Variable	Head device No.	[D605-D606](Fixed Length/3Byte/Lower/Upper Byte/No Swap)
13	Non-conversion Variable	Device code	[D607-D607](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
14	Non-conversion Variable	Number of device points	[D608-D608](Fixed Length/2Byte/Lower/Upper Byte/No Swap)

Packet Settings window

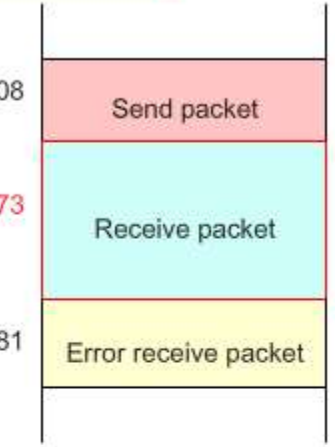
3.4.3 Packet settings

(2) Receive packet

Packet Name	Packet Setting
Request	Variable Set
Normal response	Variable Set
Error response	Variable Set

Protocol Settings window

Click "Variable Set" at Normal response



Device assignment

Protocol No. 1 Protocol Name 0401: Read (word)

Packet Type Receive Packet Packet Name Normal response

Packet No. 1

Element List

D609 to D1573, which are the receive packet's data storage area, are automatically set.

Element No.	Element Type	Element Name	Element Setting
1	Static Data	(Fixed data)	D400(2Byte)
2	Non-conversion Variable	Serial No.	[D609-D609](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
3	Static Data	(Fixed data)	0000(2Byte)
4	Non-conversion Variable	Network No.	[D610-D610](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
5	Non-conversion Variable	Station No.	[D611-D611](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
6	Non-conversion Variable	Requested module I/O No.	[D612-D612](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
7	Static Data	For future expansion	00(1Byte)
8	Length	Response data length	(Object element9-10/HEX/Reverse/2Byte)
9	Static Data	End code	0000(2Byte)
10	Non-conversion Variable	Response data	[D613][D614-D1573](Variable Length/1920Byte/Lower/Upper Byte/No Swap)

Packet Settings window

3.4.3 Packet settings

(3) Error receive packet

Packet Name	Packet Setting
Request	Variable Set
Normal response	Variable Set
Error response	Variable Set

Protocol Settings window

Click "Variable Set" at Error response.

D600 to D608 — Send packet

D609 to D1573 — Receive packet

D1574 to D1581 — Error receive packet

Device assignment

Protocol No.	1	Protocol Name	0401: Read (word)
Packet Type	Receive Packet	Packet Name	Error response
Packet No.	2		

Element List

Element No.	Element Type	Element Name	Element Setting
1	Static Data	(Fixed data)	D400(2Byte)
2	Non-conversion Variable	Serial No.	[D1574-D1574](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
3	Static Data	(Fixed data)	0000(2Byte)
4	Non-conversion Variable	Network No.	[D1575-D1575](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
5	Non-conversion Variable	Station No.	[D1576-D1576](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
6	Non-conversion Variable	Requested module I/O No.	[D1577-D1577](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
7	Static Data	For future expansion	00(1Byte)
8	Length	Response data length	(Object element9-15/HEX/Reverse/2Byte)
9	Non-conversion Variable	End code	[D1578-D1578](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
10	Non-conversion Variable	Network No.	[D1579-D1579](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
11	Non-conversion Variable	Station No.	[D1580-D1580](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
12	Non-conversion Variable	Requested module I/O No.	[D1581-D1581](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
13	Static Data	For future expansion	00(1Byte)
14	Static Data	Command	0104(2Byte)
15	Static Data	Subcommand	0000(2Byte)

D1574 to D1581, which are the error receive packet's data storage area, are automatically set.

Packet Settings window

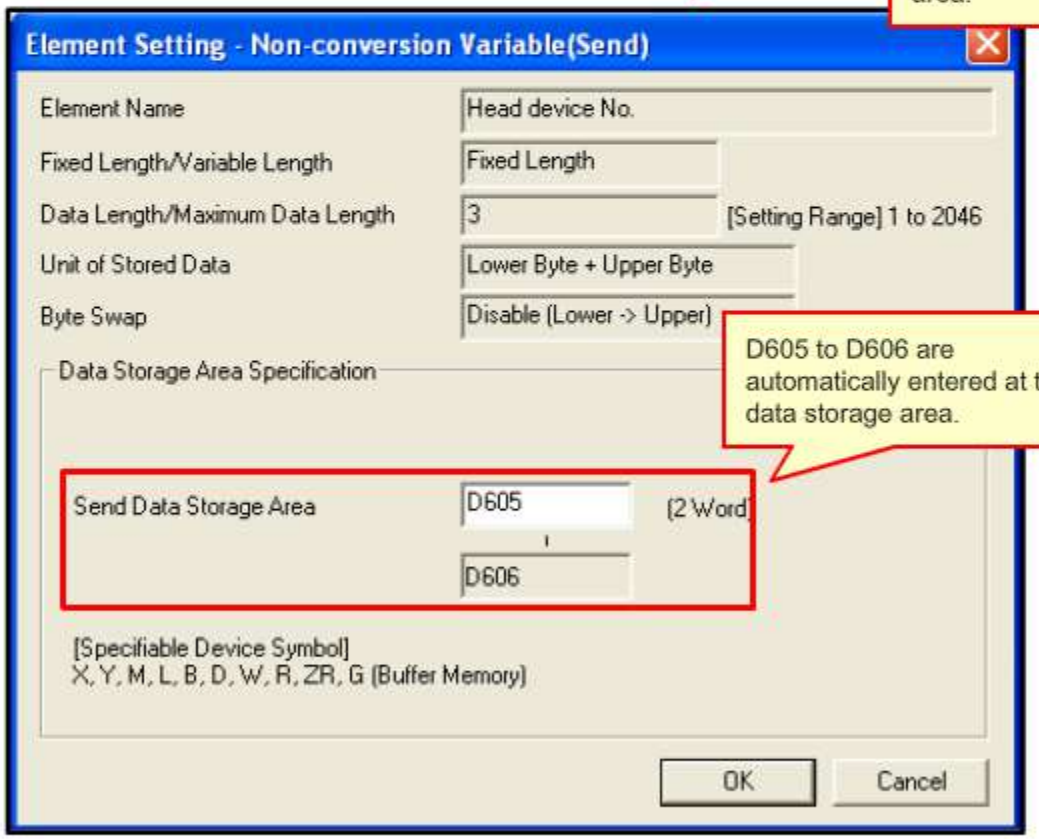
3.4.4 Element settings

The setting detail for each element can be checked and changed.

12	Non-conversion Variable	Head device No.	[D605-D606](Fixed Length/3Byte/Lower/Upper Byte/No Swap)
13	Non-conversion Variable	Device code	[D607-D607](Fixed Length/1Byte/Lower/Upper Byte/No Swap)
14	Non-conversion Variable	Number of device points	[D608-D608](Fixed Length/2Byte/Lower/Upper Byte/No Swap)

Packet Settings window

Click on the element's blue font area.



D605 to D606 are automatically entered at the data storage area.

Element Setting window

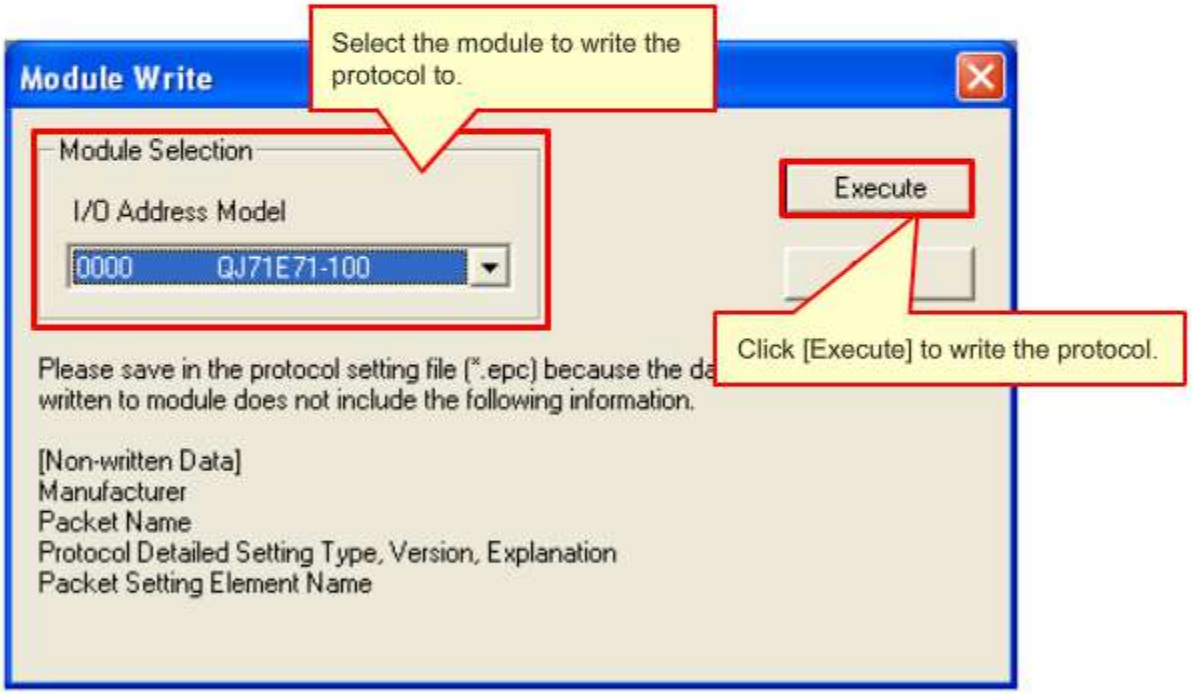
3.5 Saving a Created Protocol, and Writing It To a PLC

Saving a protocol

A created protocol can be saved to a personal computer as a protocol setting file.
From the predefined protocol support function's menu, select [File] – [Save As].

Writing a protocol to a PLC

The procedure for writing a created protocol to the Ethernet module is given below.
From the predefined protocol support function's menu, select [Online] – [Write to Module].



Module Write window

3.6 CPU Module Reset

After parameters or predefined protocols are written, the programmable controller CPU module must be reset. The CPU module can be reset by the following procedure.

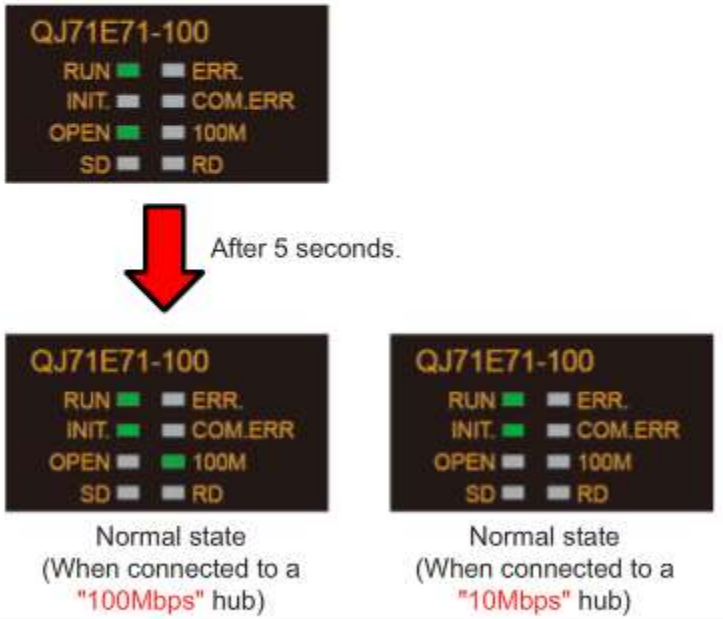
Reset method for QCPU universal model:

(1) Open the CPU module's front cover and set the [RUN/STOP/REST] switch to "RESET".

(2) After the ERR.LED blinks several times and then goes off, return the switch to the "STOP" position.

The Ethernet module's initial processing is completed if its "RUN", "INIT.", and "100M" LEDs turn on.

* The "100M" LED stays off if the QJ71E71-100 is connected to a 10Mbps hub.



Normal state (When connected to a "100Mbps" hub)

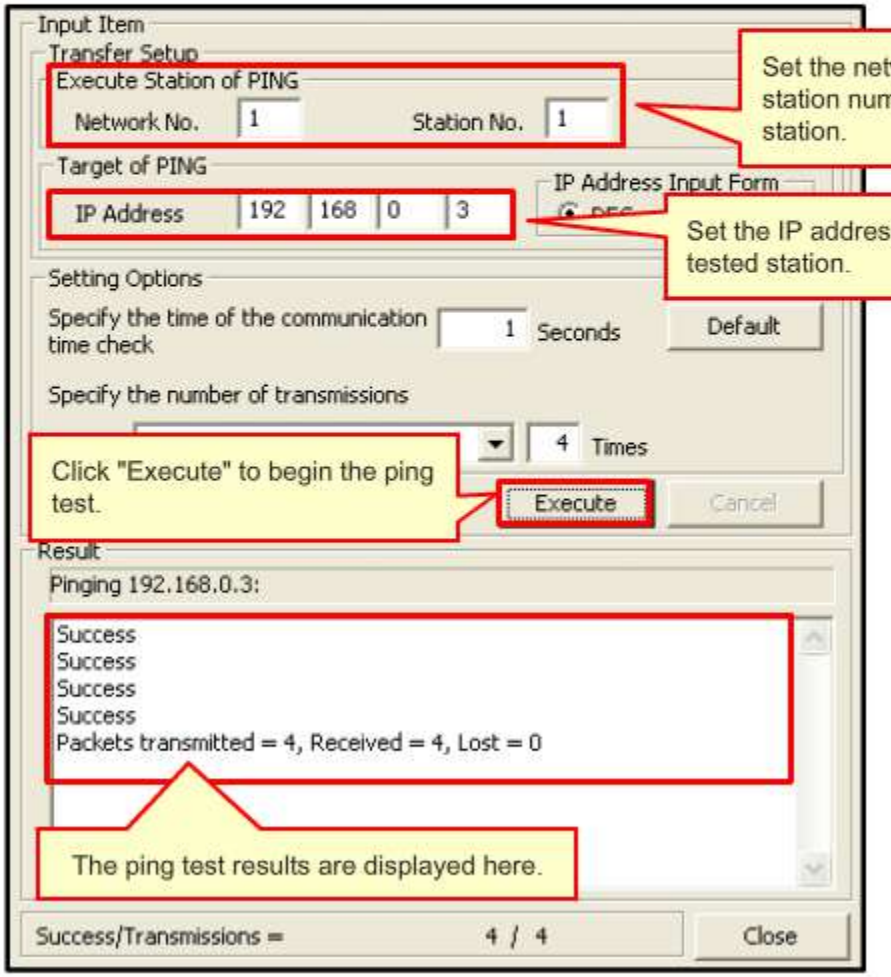
Normal state (When connected to a "10Mbps" hub)

3.7 Communication Check

A "ping test" can be performed to verify normal communication of an Ethernet module.

Ping test check method

(1) From the GX Works2 menu, select [Diagnosis] – [Ethernet Diagnosis] to open the Ethernet Diagnosis window.
(2) Click the "PING Test" button to open the PING test window.



Ping Test window

The Ethernet module's LED indicators can also be checked.

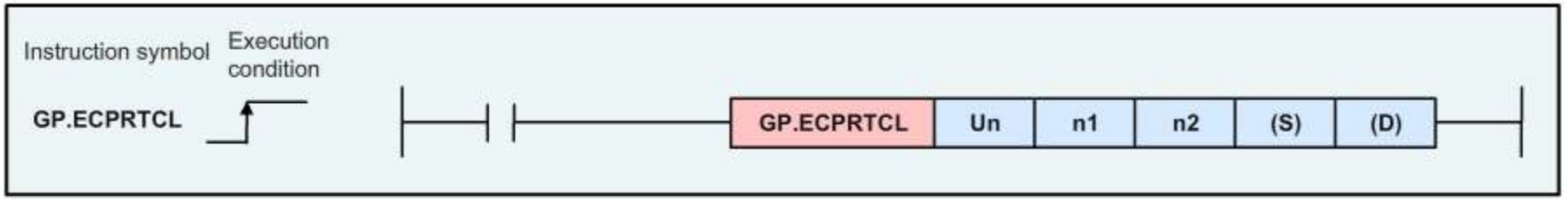
Ethernet module's LED indicator status when operation is normal

This LED may be off depending on the communication speed.

3.8 Dedicated Instructions

A dedicated instruction can be used to execute a protocol saved in a flash ROM.

Dedicated instruction



Setting data

Setting data	Details	Setting range	Setting by	Value for the example system
Un	Ethernet module's first I/O number (00 to FEH: First two digits of the three-digit I/O signal)	User	BIN 16 bits	Select the module installation slot 0.
n1	Connection No. (1 to 16)	User	BIN 16 bits device name	Set "1" because the protocol is saved as No.1.
n2	Number of protocol setting data to be continuously executed (1 to 8)	User	BIN 16 bits device name	Set "1" to execute single protocol.
(S)	Start number of the device in which control data is stored.	User, system	Device name	Set "D500".
(D)	Device number of the bit device that will be turned on when execution is completed. At an error completion, (D) + 1 is also turned on.	System	Bit	"M1000"

3.8 Dedicated Instructions

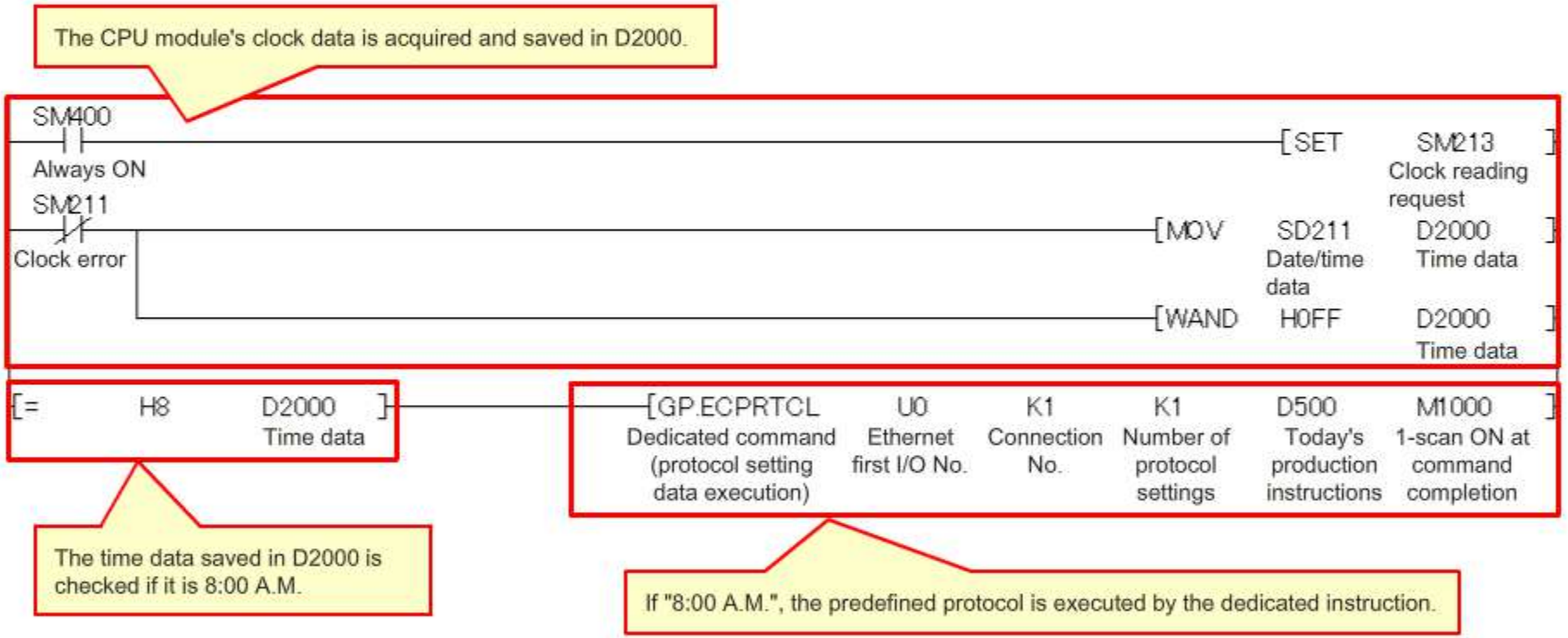
Control data

Control data is the data area storing the parameters to be executed by the GP.ECPRTCL instruction.
The execution results are also saved here.

Device	Name	Details	Setting by	Data type	Value for the example system
(S)+0=D500	Execution count result	• The number of predefined protocols executed by the ECPRTCL instruction is saved. • The number includes executed protocols in which an error has occurred. • "0" is saved if the setting data or control data is incorrectly set.	0, 1 to 8	System	The system automatically writes "1" for a normal response.
(S)+1=D501	Completion status	• The status at completion is saved. • When multiple predefined protocols are executed, the execution result of the last executed predefined protocol is stored. 0000H: Normal completion Other than 0000H (error code): Error completion	-	System	The system automatically writes "0" for a normal response, or an error code for an error.
(S)+2=D502	Protocol No. to be executed	The protocol No. to be executed first.	1 to 128	User	Write "1" in D502 because only the protocol No.1 is used.
↓		↓	0, 1 to 128		
(S)+9=D509		The protocol number to be executed at the 8th order.			

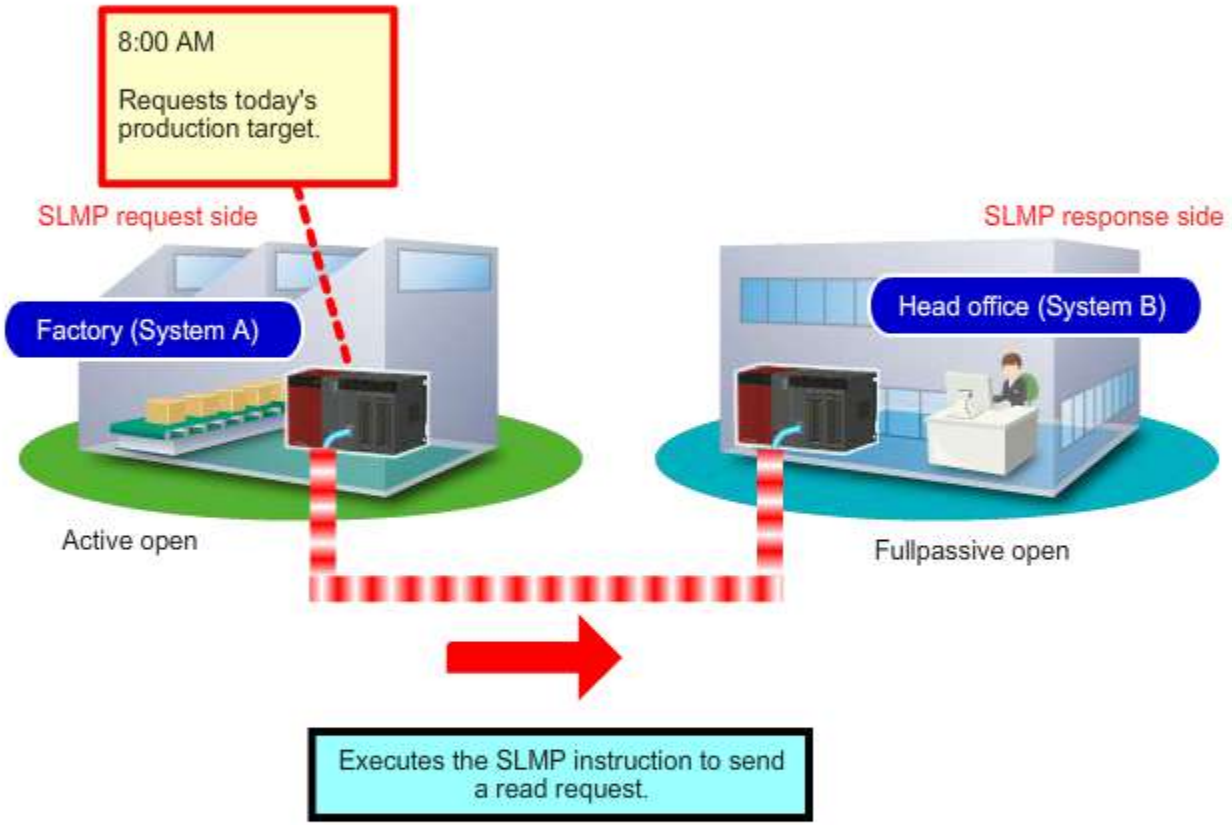
3.9 Sequence Program Example

The following example shows an SLMP responding side's sequence program, which uses a dedicated instruction.
Please remember the example system introduced in Section 2.3. In the example, System A at factory floor accesses System B at the head office at 8 A.M. every morning to retrieve the production target of the day. In this example, the number of executed predefined protocols is "1".



3.10 Example System Operation

Please check the example system operation with the animation below.



3.11 Summary

In this chapter, you have learned:

- Pre-operation settings and setting procedure
- Connection method
- Parameter settings
- Predefined protocol support function
- Saving a created protocol, and writing it to a PLC
- CPU module reset
- Communication check
- Dedicated instructions
- Sequence program example
- Example system operation

Important points

Pre-operation settings and setting procedure	The installation procedure should be checked before using an Ethernet module.
Network parameter settings	GX Works2 is used to configure the network parameter settings. GX Works2 is also used to configure necessary settings to the programmable controllers to which the Ethernet module is connected.
Parameter writing	The parameters required for Ethernet module operation are written to the CPU module.
Communication check	A ping test is used to check the normal communication.

Chapter 4 Troubleshooting

Chapter 4 describes network diagnosis for problems.

4.1 Troubleshooting

This section explains the errors that can occur in data communication between an Ethernet module and its communicating device, and corrective actions for such errors.
When a problem occurs, check the LED indicator status first, then take the appropriate measure for that status.
Errors such as the COM.ERR cannot be diagnosed by the LED status alone. Use GX Works2 to check the error details.

4.1.1 Checking errors by the LED indicator status

The following section lists error conditions that can be checked from the Ethernet module's LED indicators.

4.1.1 Checking errors by the LED indicator status

LED	Normal	Error	Possible cause	Corrective action
RUN	ON (Green)	OFF	Watchdog timer error	Reset the CPU module, and check if the LED is still on. If the RUN LED is still on, the Ethernet module may be faulty. Repair or replace the module.
			Poor Ethernet module installation	Check if the power supply module's power supply capacity (5 VDC) is sufficient. Turn the power off, and re-install the module.
ERR.	OFF	ON (RED)	Module parameter setting error	Use GX Works2 to check/correct the Ethernet module's parameter settings.
			CPU module error	If the CPU module's "RUN" LED is off or blinking, or if the ERR. LED is on, verify the error content and remove the cause. Verify that the Ethernet module is installed in a Q-mode CPU module.
			Ethernet module error (H/W error)	Replace the Ethernet module.
COM.ERR	OFF	ON (RED)	Identify the error detail by checking the error code, then correct the error cause. For the COM error, use GX Works2's Ethernet diagnosis function to check the error code. For error code details, please refer to the corresponding manual of the Ethernet module.	
SD	ON (Green) during data transmission	OFF (data cannot be send)	"ERR." or "COM.ERR" LED is ON.	Remove the cause for "ERR." or "COM.ERR".
			Incorrect cable connection	Check the cable connection.
			Incorrect program	Revise the send sequence program.
RD	ON (Green) during data reception	OFF (data cannot be received)	"ERR." or "COM.ERR" LED is ON	Remove the cause for "ERR." or "COM.ERR".
			Incorrect cable connection	Check the cable connection.
			Own-station IP address setting error	If the cable is connected correctly, use GX Works2 to change the own-station IP address, the router and the subnet mask settings.
			Incorrect program	Revise the other device's send program.

Some of the common problems are listed on the following page.

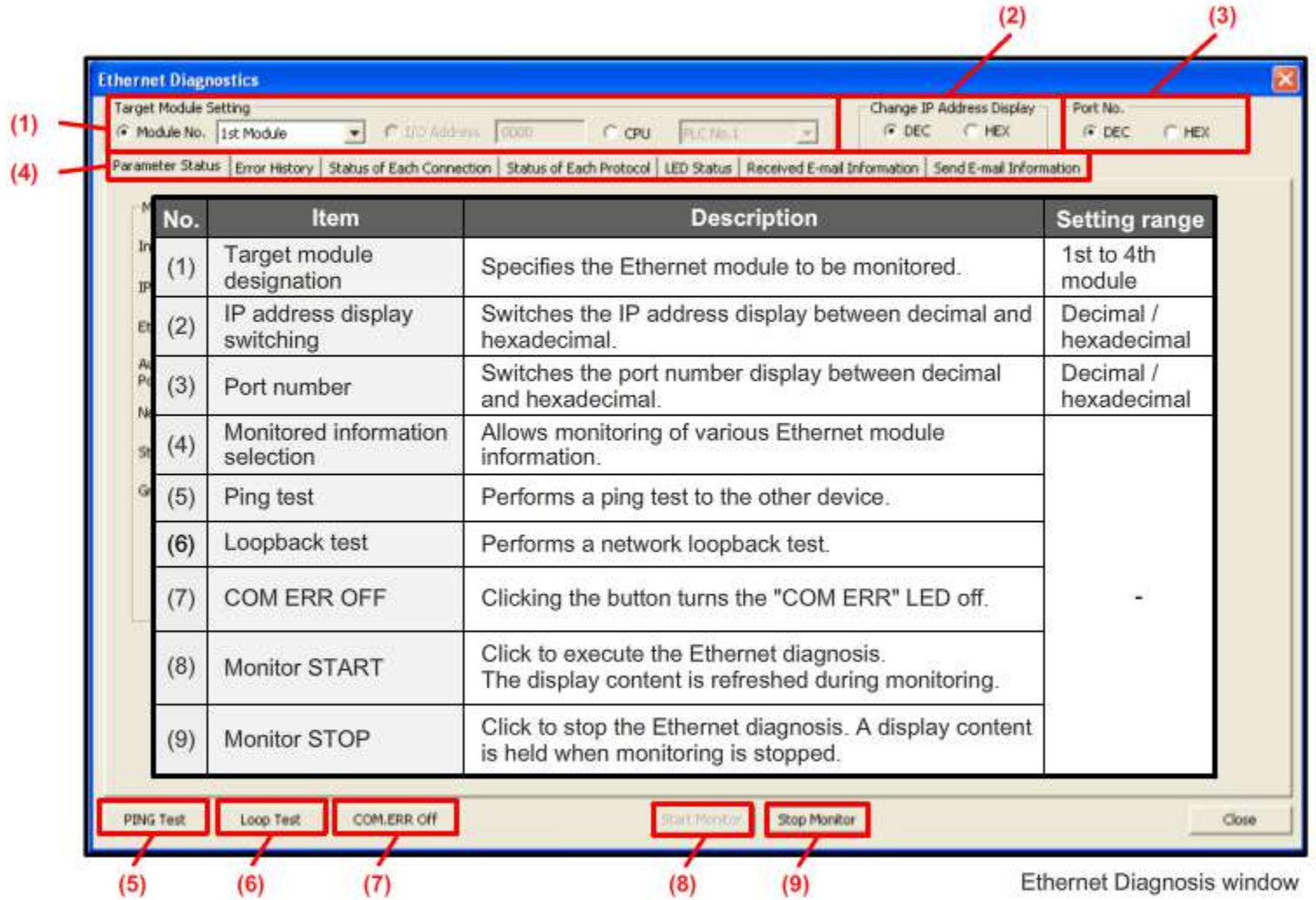
4.1.2 List of common problems

The table below lists some of the common problems. A user should check this first when a problem occurs.

Item	Problem	Possible cause	Corrective action
Problems that occur at start-up	An open processing is performed by SLMP from a personal computer, but that processing cannot be completed.	An incorrect port number is set at the personal computer or Ethernet module. (Note that the personal computer port number is usually set in decimal, but the Ethernet module port number is set in hexadecimal.)	Return to the open setting, and recheck the port numbers.
	An open processing from a personal computer has been competed, but no communication occurs.	Binary/ASCII is set incorrectly at the communication data code.	Return to the operation setting, and recheck the communication data code setting.
Problems that occur during operation	An Ethernet module fails to communicate.	• The hub power is off. • The cable is cut off or not connected properly.	• Check the hub power. • Check the cable connection.

4.1.3 Check by Ethernet diagnosis function

The GX Works2's "Ethernet diagnosis" function can be used to check the error codes and details for errors occurring at the Ethernet module.



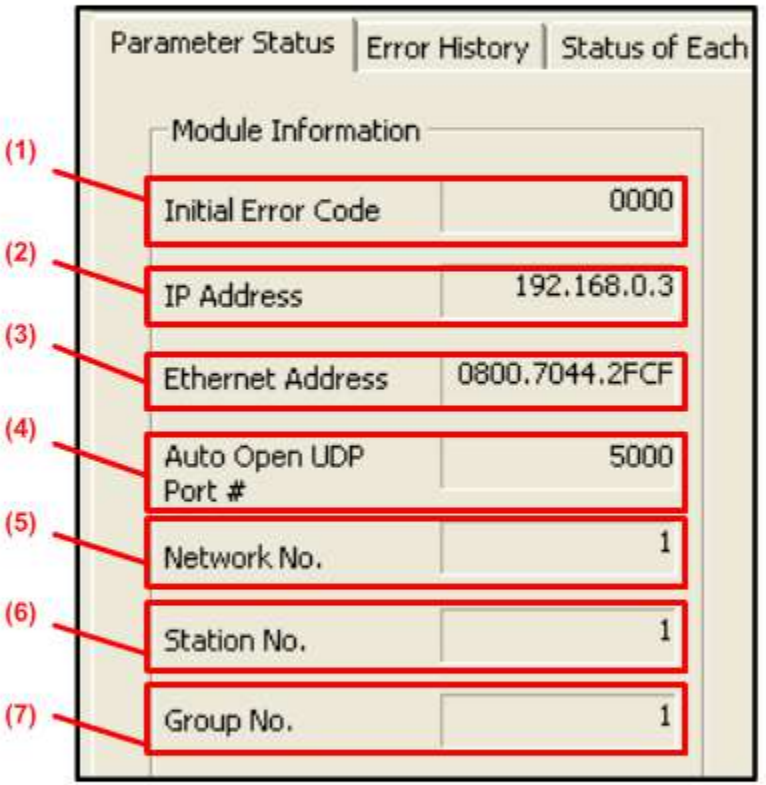
No.	Item	Description	Setting range
(1)	Target module designation	Specifies the Ethernet module to be monitored.	1st to 4th module
(2)	IP address display switching	Switches the IP address display between decimal and hexadecimal.	Decimal / hexadecimal
(3)	Port number	Switches the port number display between decimal and hexadecimal.	Decimal / hexadecimal
(4)	Monitored information selection	Allows monitoring of various Ethernet module information.	-
(5)	Ping test	Performs a ping test to the other device.	
(6)	Loopback test	Performs a network loopback test.	
(7)	COM ERR OFF	Clicking the button turns the "COM ERR" LED off.	
(8)	Monitor START	Click to execute the Ethernet diagnosis. The display content is refreshed during monitoring.	
(9)	Monitor STOP	Click to stop the Ethernet diagnosis. A display content is held when monitoring is stopped.	

Ethernet Diagnosis window

4.1.3 Check by Ethernet diagnosis function

Parameters status

When the Ethernet module's initial processing is executed, the following values are automatically set. Check that the set values are consistent with the designed values.



Ethernet Diagnosis window (Parameters Status)

Example of "ERR." indicator LED

No.	Item	Description
(1)	Initial Error Code	An error code is displayed if a connection error occurs. (Normal status: "0000")
(2)	IP Address	The IP address of the Ethernet module is displayed.
(3)	Ethernet Address	The Ethernet address of the Ethernet module is displayed.
(4)	Auto Open UDP Port #	The port number for the initial processing is displayed.
(5)	Network No.	The network number of the Ethernet module is displayed.
(6)	Station No.	The station number of the Ethernet module is displayed.
(7)	Group No.	The group No. of the Ethernet module is displayed.

4.1.3 Check by Ethernet diagnosis function

Error history

The COM.ERR LED indicates an error occurring during data communication between the Ethernet module and the other device, or an error requested from the CPU module. Use the Ethernet diagnosis function to check the error log to identify the error code, then take the appropriate corrective action.

* For error code details, please refer to the corresponding manual of the Ethernet module.

Example of "COM.ERR" indicator ON status

Parameter Status | Error History | Status of Each Connection | Status of Each Protocol | LED Status | Received E-mail Information | Send E-mail Information

Number of Error Occurrences: 2

Displays the number of error occurrences.

No.	Error End Code	Sub Header	Command Code	Connection No.	Host Station Port No.	Destination IP Address	Destination Port No.
Latest	C061	0054	0401	0001	4096	192.168.0.2	8192
2	C061	0054	0401	0001	4096	192.168.0.2	8192
3							
4							
5							
6							
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							

Displays the details of errors which have occurred.

Clear History

Clears the error log.

Ethernet Diagnosis window (Error Log)

4.1.3 Check by Ethernet diagnosis function

Status of each connection

The status of each connection is indicated by a connection number.

Example of "OPEN" indicator ON status

Parameter Status | Error History | Status of Each Connection | Status of Each Protocol | LED Status | Received E-mail Information | Send E-mail Information

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

No.	Host Station Port No.	Destination IP Address	Destination Port No.	Open Error Code	Fixed Buffer Send/Receive Error Code	Connection End Code	Protocol
1	2000	192.168.0.2	2000	0000	0000	0000	TCP
2							

mail Information | Send E-mail Information

(9) (10) (11)

Open System	Pairing Open	Existence Confirmation
Fullpassive	No Pairs	No Confirm

Ethernet Diagnosis window (status of each connection)

No.	Item	Description
(1)	No.	Connection No. (corresponds to the open setting No.)
(2)	Host Station Port No.	Port number used by the Ethernet module.
(3)	Destination IP Address	IP address of the other device where the connection has been established.
(4)	Destination Port No.	Port number of the other device where the connection has been established.
(5)	Open Error Code	Saves the open processing result for the relevant connection.
(6)	Fixed Buffer Send/Receive Error Code	During the relevant connection's fixed buffer communication, saves the error code of an error which occurs during a data transmission to the other device.
(7)	Connection End code	During the relevant connection's fixed buffer communication, saves the response code from other device.
(8)	Protocol	Protocol used by the relevant connection.
(9)	Open System	Open format used by the relevant connection.
(10)	Pairing Open	Pairing open enabled/disabled status.
(11)	Existence Confirmation	Alive check enabled/disabled status.

4.2 Summary

In this chapter, you have learned:

- Troubleshooting

Important points

Checking for errors by the LED indicator status	The method for checking the LED indicator statuses to identify errors was explained.
Ethernet diagnosis	The method for using the GX Works2 Ethernet diagnosis function to check error details was explained.

Test Final Test

Now that you have completed all of the lessons of the PLC Ethernet Course, you are ready to take the final test. If you are unclear on any of the topics covered, please take this opportunity to review those topics.
There are a total of 10 questions (41 items) in this Final Test.
You can take the final test as many times as you like.

How to score the test

After selecting the answer, make sure to click the **Answer** button. Your answer will be lost if you proceed without clicking the Answer button. (Regarded as unanswered question.)

Score results

The number of correct answers, the number of questions, the percentage of correct answers, and the pass/fail result will appear on the score page.

Correct Answers : 2

Total Questions : 9

Percentage : 22%

To pass the test, you have to answer **60%** of the questions correct.

Proceed | Review | Retry

- Click the **Proceed** button to exit the test.
- Click the **Review** button to review the test. (Correct answer check)
- Click the **Retry** button to retake the test again.

Test Final Test 1

Forward

Ethernet communication protocol

The table below lists characteristics of TCP and UDP.
Please select the correct terms to complete the table.

Item	TCP	UDP
Reliability		
Processing speed		
Connection with other device(s)	--Select--	--Select--
Data reception assurance		
Operation at transmission error	--Select--	--Select--
Connection establishment	--Select--	--Select--
Flow control	Yes	No
Congestion control (retransmission control)	Yes	No
Communicating device change during open connection	Not possible	Possible

Answer Back

Test Final Test 2

Open/close processing in TCP/IP communication

The following sentences are descriptions about the open processing.
Please select the correct term for each description.

Term	Description
--Select--	Sends an active open request to the other device that is in a passive open state.
--Select--	Waits for an open request from the other device that requests an active open.
--Select--	Accepts an active open request only from a specific network-connected device.
--Select--	Accepts an active open request from any network-connected device.

Answer Back

Test Final Test 3

IP address

The following sentences are descriptions about IP address.
Please select the correct terms to complete the sentences.

Description
IP address (Internet Protocol address) is an identification number that is assigned to a device/computer connected to an IP network, such as Internet and intranet. An IP address is a set of numbers expressed in --Select-- and is usually divided into four ---------- sections by dots (e.g., "192.168.1.1").

Answer Back

Test Final Test 4

Ethernet port number

The following sentences are descriptions about a port number.
Please select the correct term for each description.

Description
The actual communication occurs between the applications running on the devices and computers. In TCP and UDP, a port number is used to identify which application is communicating. Port numbers that are unique to each application. : --Select-- (Well Known Port Numbers) * For example, the email recipient port number is 25, the homepage reference port number is 80, and the file transfer port number is 20. Port numbers that can be freely set for an Ethernet module : --Select--

Answer Back

Test Final Test 5

Data code

The following sentences are descriptions about communication data codes.
Please select the correct term for each description.

Term	Description
▼	For sending/receiving 1-byte data as it is.
▼	For sending/receiving 1-byte data as two ASCII code characters.

Answer Back

Test Final Test 6

Communication protocol

The following sentences are descriptions about Ethernet communication protocols.
Please select the correct term for each description.

Term	Description
--Select--	A type of communication protocol that allows an SLMP compatible external device to access an Ethernet module, etc.
--Select--	Communication with the CPU module or a personal computer, etc. is performed using the fixed buffer in an Ethernet module memory.
--Select--	Communication with the CPU module or a personal computer, etc. is performed using the random access buffer in an Ethernet module memory.

Answer Back

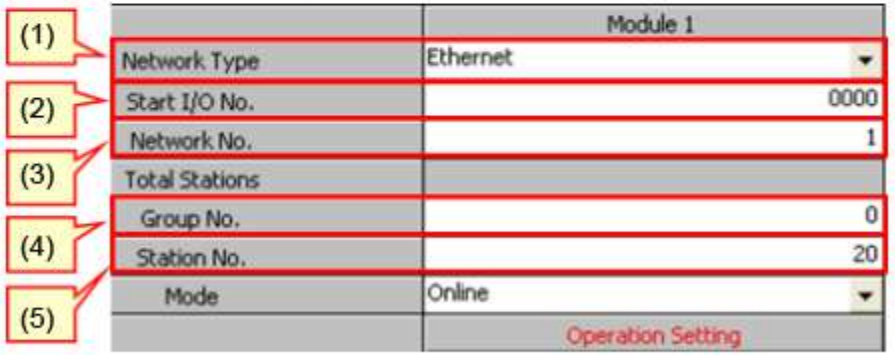
PLC_Ethernet_ENG

Test Final Test 7

Network parameter setting

The following sentences are descriptions about Network Parameter window.
Please select the correct section for each description.

Number	Description
[▼]	The start I/O No. of the Ethernet module is set in the units of 16 points (hexadecimal).
[▼]	When the installed module is selected here, the corresponding items will be selectable.
[▼]	The station number of the Ethernet module is selected. (Setting range: 1 to 64)
[▼]	The group number of the Ethernet module is selected. (Setting range: 1 to 32)
[▼]	The network number of the Ethernet module is selected. (Setting range: 1 to 239)



Answer Back

PLC_Ethernet_ENG

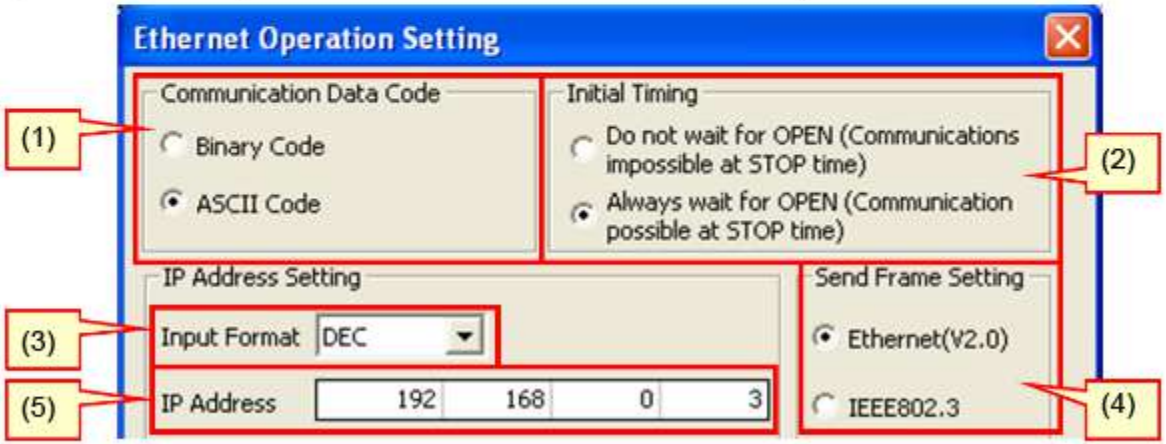
Test Final Test 8

Forward

Network parameter setting

The following sentences are descriptions about Ethernet Operation Setting window.
Please select the correct section for each description.

Number	Description
[▼]	Select the IP address input format.
[▼]	This is a setting about the open processing.
[▼]	Select the communication data code.
[▼]	Set the own-station's IP address.
[▼]	Select the send frame setting.



Answer Back

Test Final Test 9

Troubleshooting

The following sentences are descriptions about troubles common to an Ethernet module.
Please select the correct corrective action for each description.

Term	Symptom	Possible cause	Corrective action
Problems that occur at start-up	An open processing is performed by SLMP from a personal computer, but that processing cannot be completed.	An incorrect port number is set at the personal computer or Ethernet module. (Note that the personal computer port number is usually set in decimal, but the Ethernet module port number is set in hexadecimal.)	
	An open processing from a personal computer has been competed, but no communication occurs.	Binary/ASCII is set incorrectly at the communication data code.	
Problems that occur during operation	An Ethernet module fails to communicate.	The hub power is off, or the cable is cut off or not connected properly.	

(1): Check the hub power, and check the cable connection.
(2): Return to the open setting, and recheck the port numbers.
(3): Return to the operation setting, and recheck the communication data code setting.

Answer Back

Test Final Test 10

Checks by Ethernet diagnostics function

The following sentences are descriptions about Ethernet Diagnostics window tabs.
Please select the correct tab for each description.

Term	Description
--Select--	After executing the initial processing of the Ethernet module, the saved parameter values should be checked.
--Select--	The LEDs indicate an error occurred during processing of data communications between the Ethernet module and other devices, or an error in the requests from the CPU module.
--Select--	After connection is established by open processing, the connection status is displayed for each device.

Answer Back

Test Score

You have completed the Final Test. You results area as follows.
To end the Final Test, proceed to the next page.

Correct answers : 0

Total questions : 10

Percentage : 0%

Proceed | Review | Retry

You failed the test.

PLC_Ethernet_ENG

You have completed the PLC Ethernet Course.

Thank you for taking this course.

We hope you enjoyed the lessons and the information you acquired in this course will be useful in the future.

You can review the course as many times as you want.

Review Close