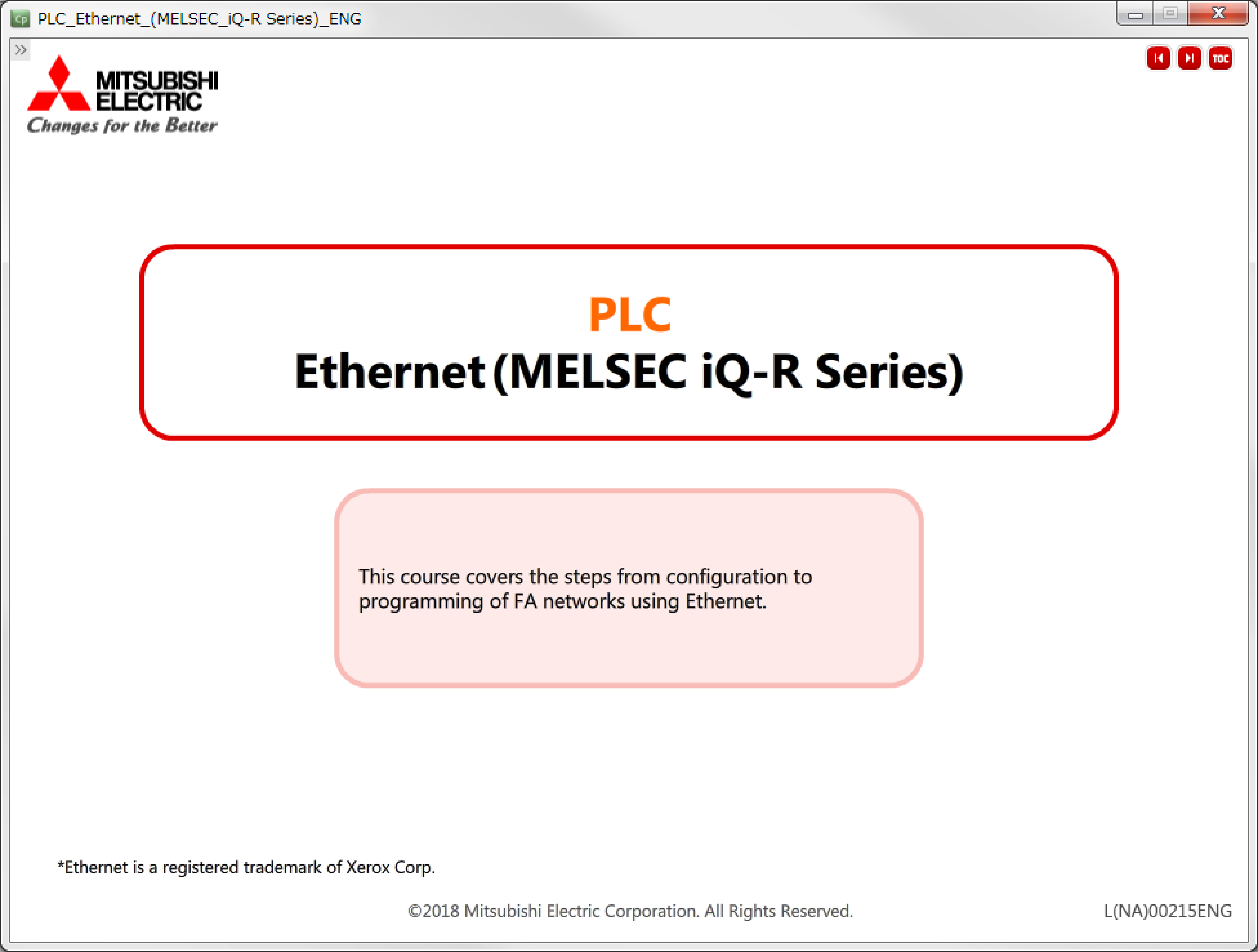
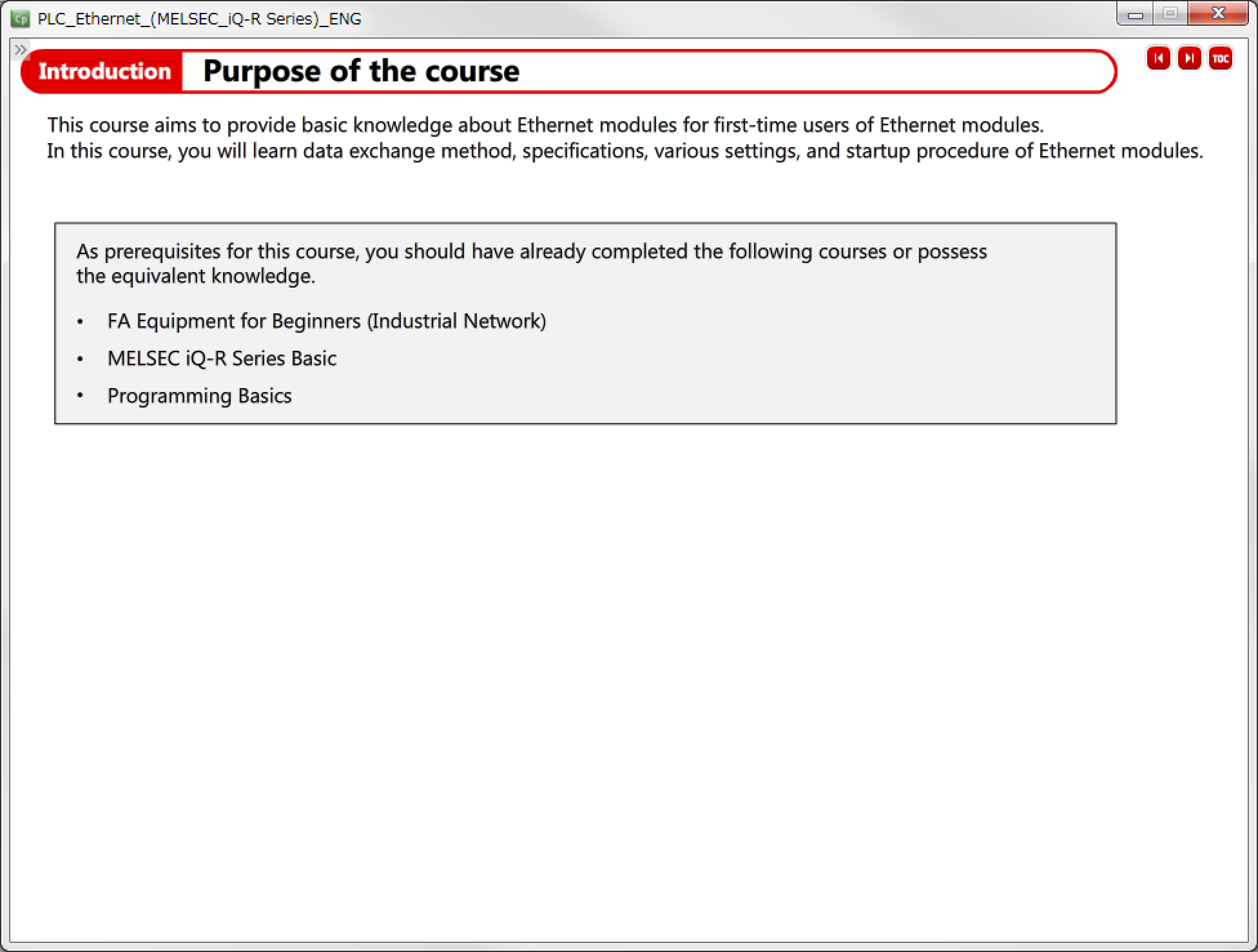


PLC
Ethernet (MELSEC iQ-R Series)

This course covers the steps from configuration to programming of FA networks using Ethernet.

*Ethernet is a registered trademark of Xerox Corp.

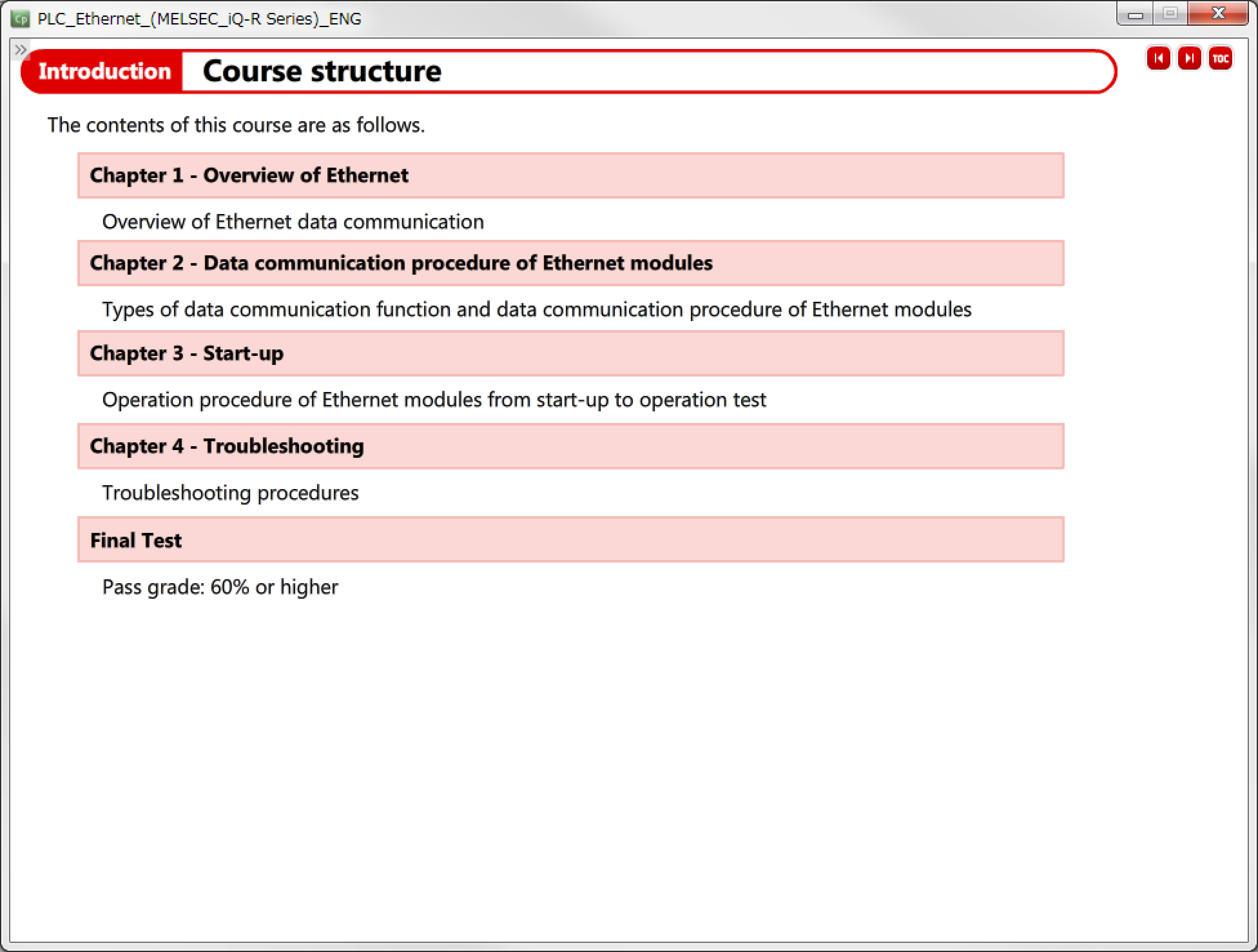


Introduction Purpose of the course

This course aims to provide basic knowledge about Ethernet modules for first-time users of Ethernet modules.
In this course, you will learn data exchange method, specifications, various settings, and startup procedure of Ethernet modules.

As prerequisites for this course, you should have already completed the following courses or possess the equivalent knowledge.

- FA Equipment for Beginners (Industrial Network)
- MELSEC iQ-R Series Basic
- Programming Basics



Introduction Course structure

The contents of this course are as follows.

Chapter 1 - Overview of Ethernet

Overview of Ethernet data communication

Chapter 2 - Data communication procedure of Ethernet modules

Types of data communication function and data communication procedure of Ethernet modules

Chapter 3 - Start-up

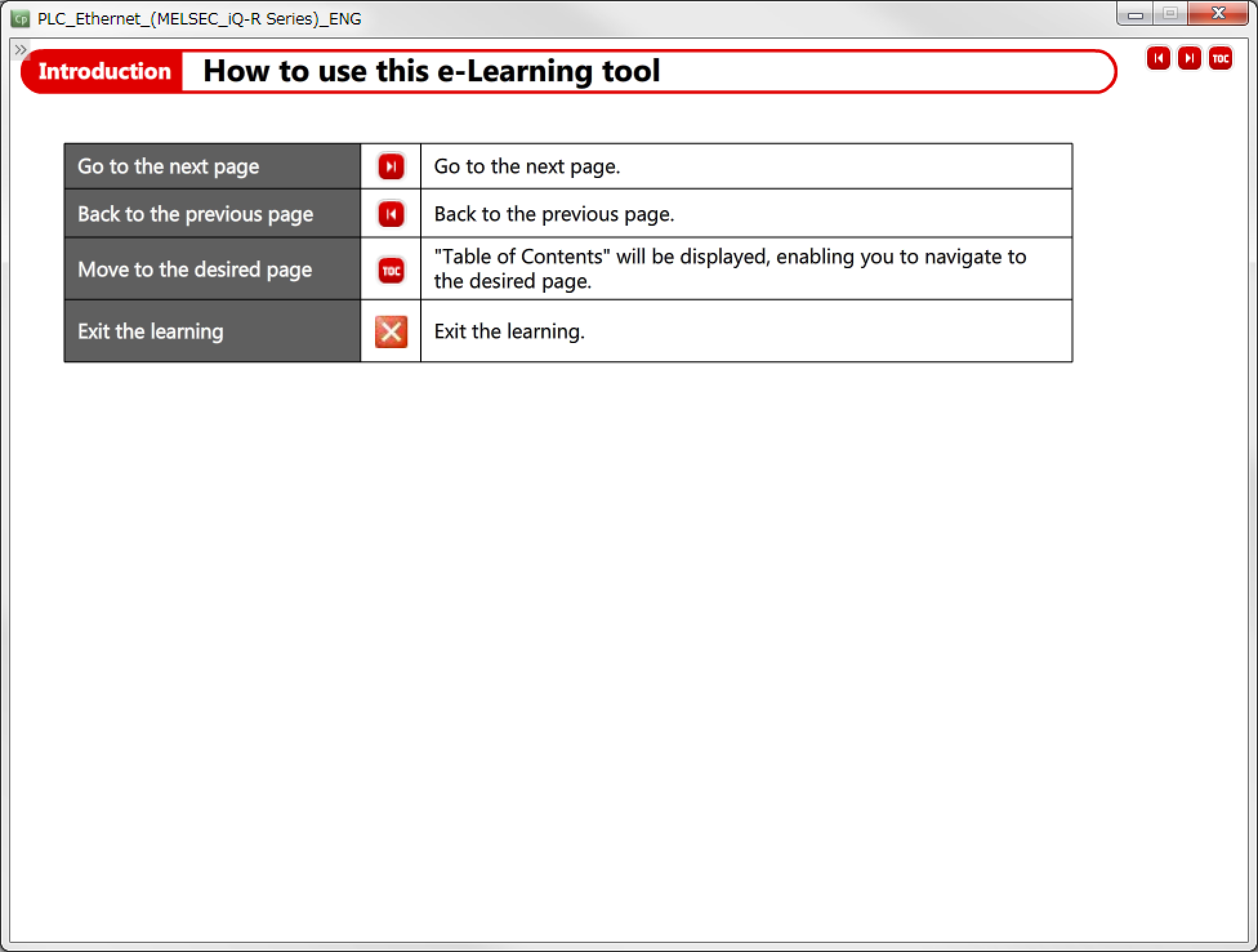
Operation procedure of Ethernet modules from start-up to operation test

Chapter 4 - Troubleshooting

Troubleshooting procedures

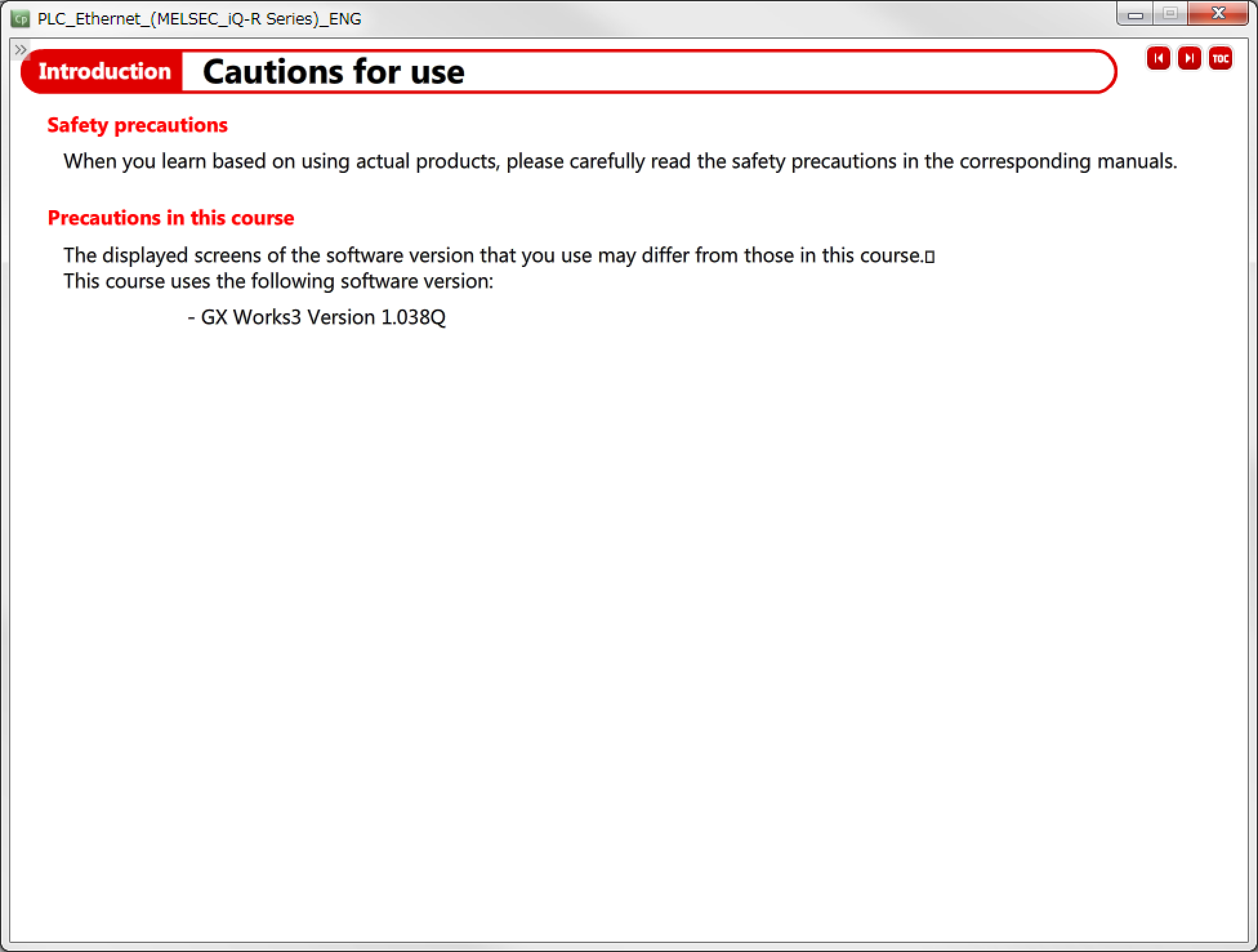
Final Test

Pass grade: 60% or 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	TOC	"Table of Contents" will be displayed, enabling you to navigate to the desired page.
Exit the learning	✕	Exit the learning.



Introduction Cautions for use

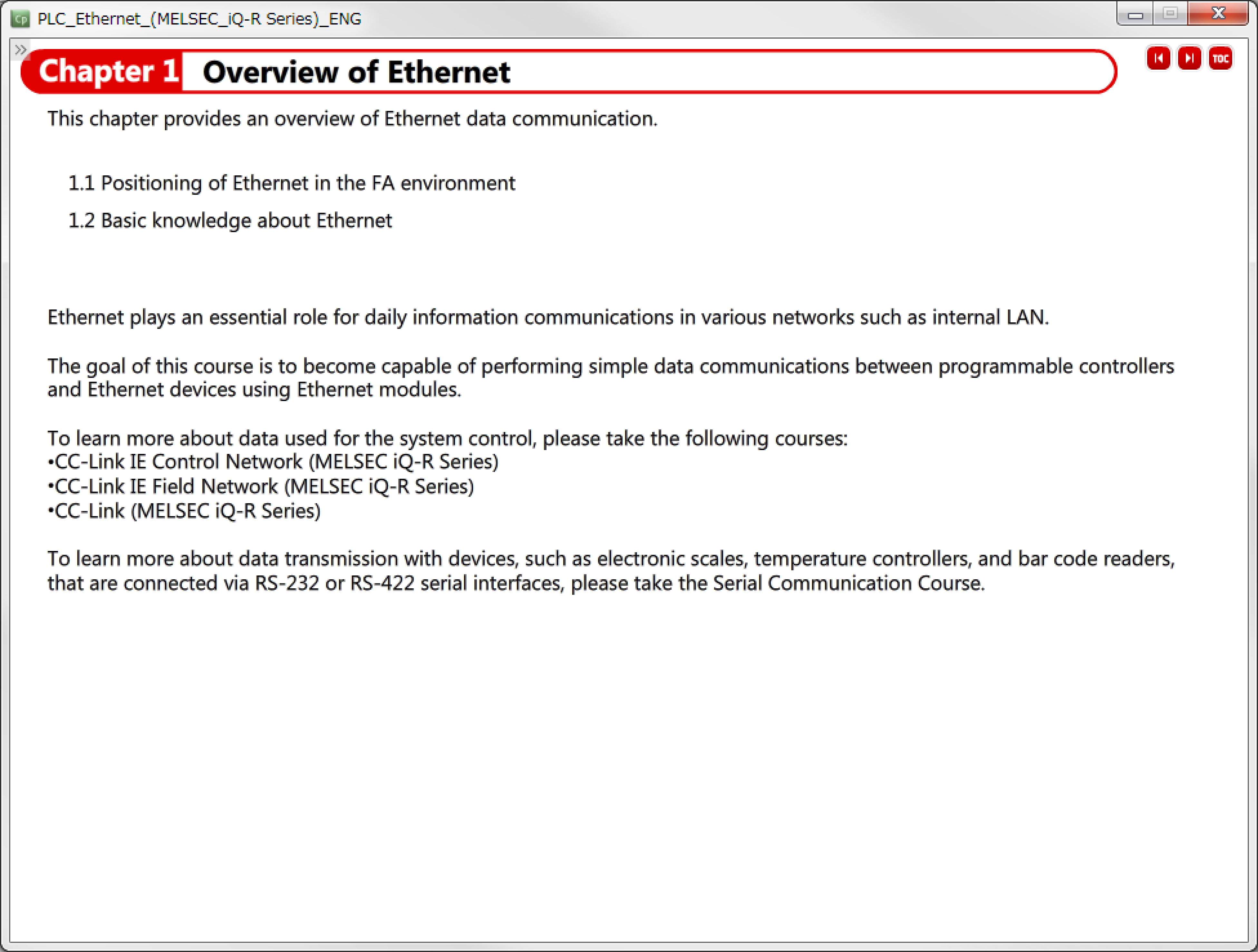
Safety precautions

When you learn based on 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 Works3 Version 1.038Q



Chapter 1 Overview of Ethernet

This chapter provides an overview of Ethernet data communication.

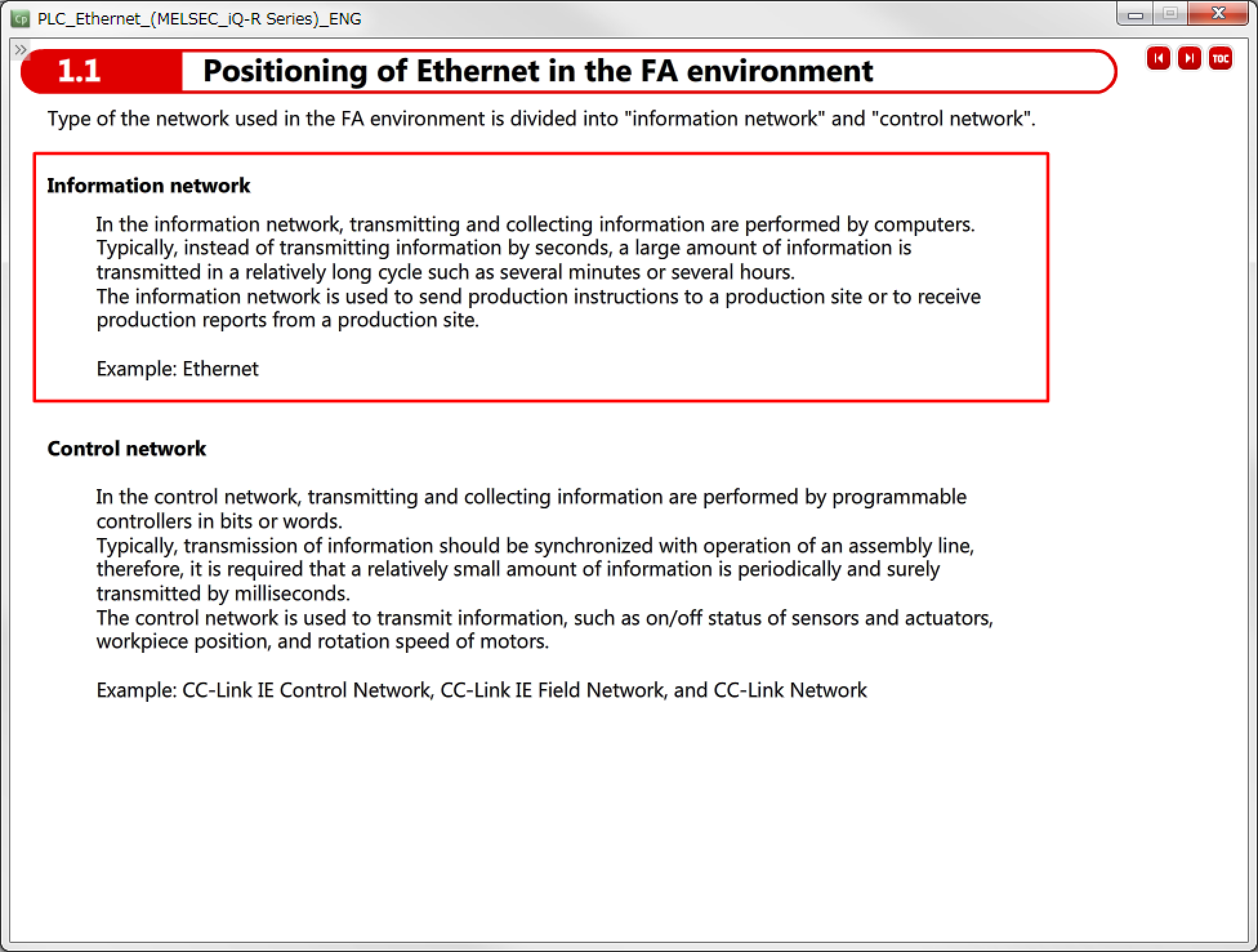
Ethernet plays an essential role for daily information communications in various networks such as internal LAN.

The goal of this course is to become capable of performing simple data communications between programmable controllers and Ethernet devices using Ethernet modules.

To learn more about data used for the system control, please take the following courses:

- CC-Link IE Control Network (MELSEC iQ-R Series)
- CC-Link IE Field Network (MELSEC iQ-R Series)
- CC-Link (MELSEC iQ-R Series)

To learn more about data transmission with devices, such as electronic scales, temperature controllers, and bar code readers, that are connected via RS-232 or RS-422 serial interfaces, please take the Serial Communication Course.



1.1 Positioning of Ethernet in the FA environment

Type of the network used in the FA environment is divided into "information network" and "control network".

Information network

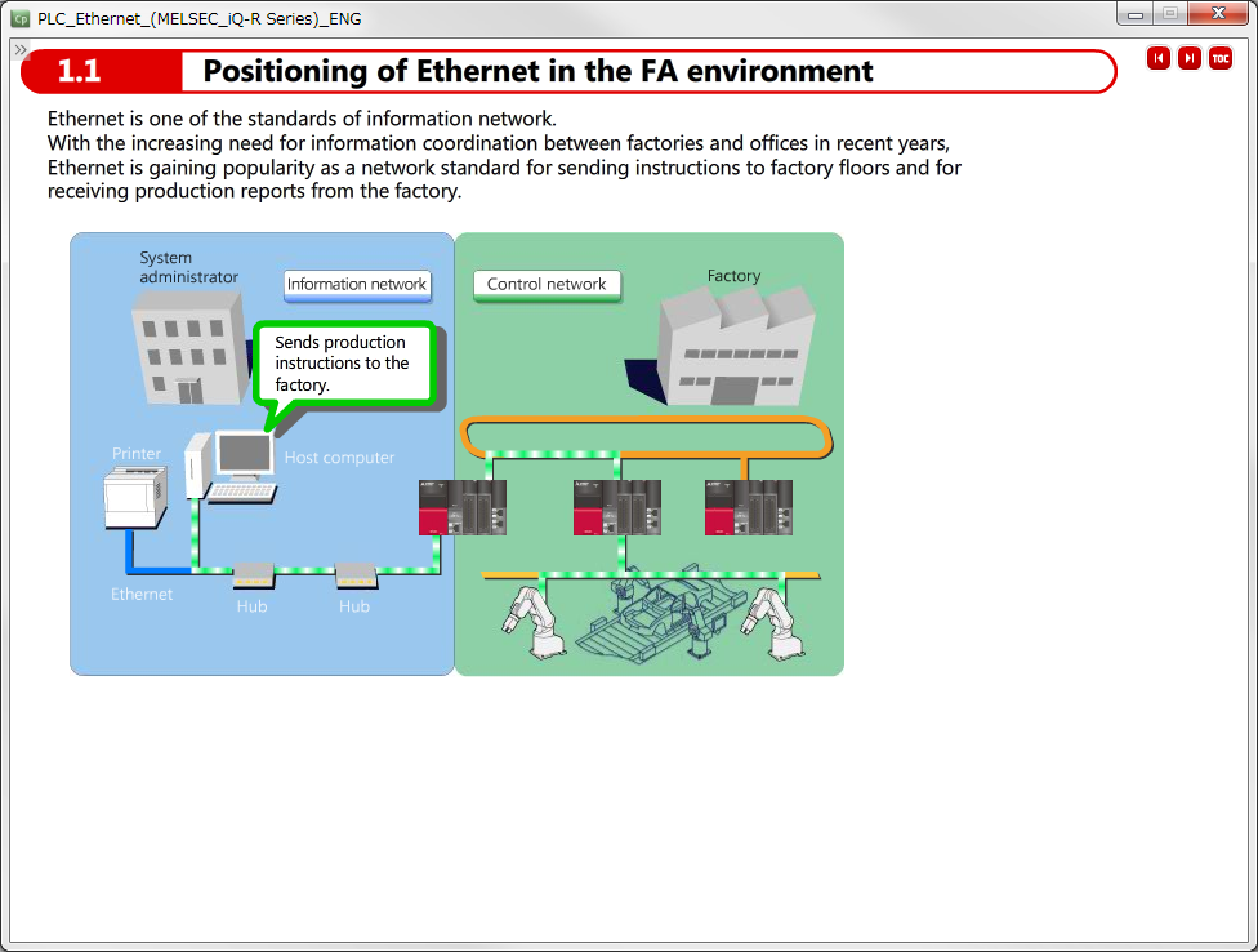
In the information network, transmitting and collecting information are performed by computers. Typically, instead of transmitting information by seconds, a large amount of information is transmitted in a relatively long cycle such as several minutes or several hours.
The information network is used to send production instructions to a production site or to receive production reports from a production site.

Example: Ethernet

Control network

In the control network, transmitting and collecting information are performed by programmable controllers in bits or words.
Typically, transmission of information should be synchronized with operation of an assembly line, therefore, it is required that a relatively small amount of information is periodically and surely transmitted by milliseconds.
The control network is used to transmit information, such as on/off status of sensors and actuators, workpiece position, and rotation speed of motors.

Example: CC-Link IE Control Network, CC-Link IE Field Network, and CC-Link Network



1.1 Positioning of Ethernet in the FA environment

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

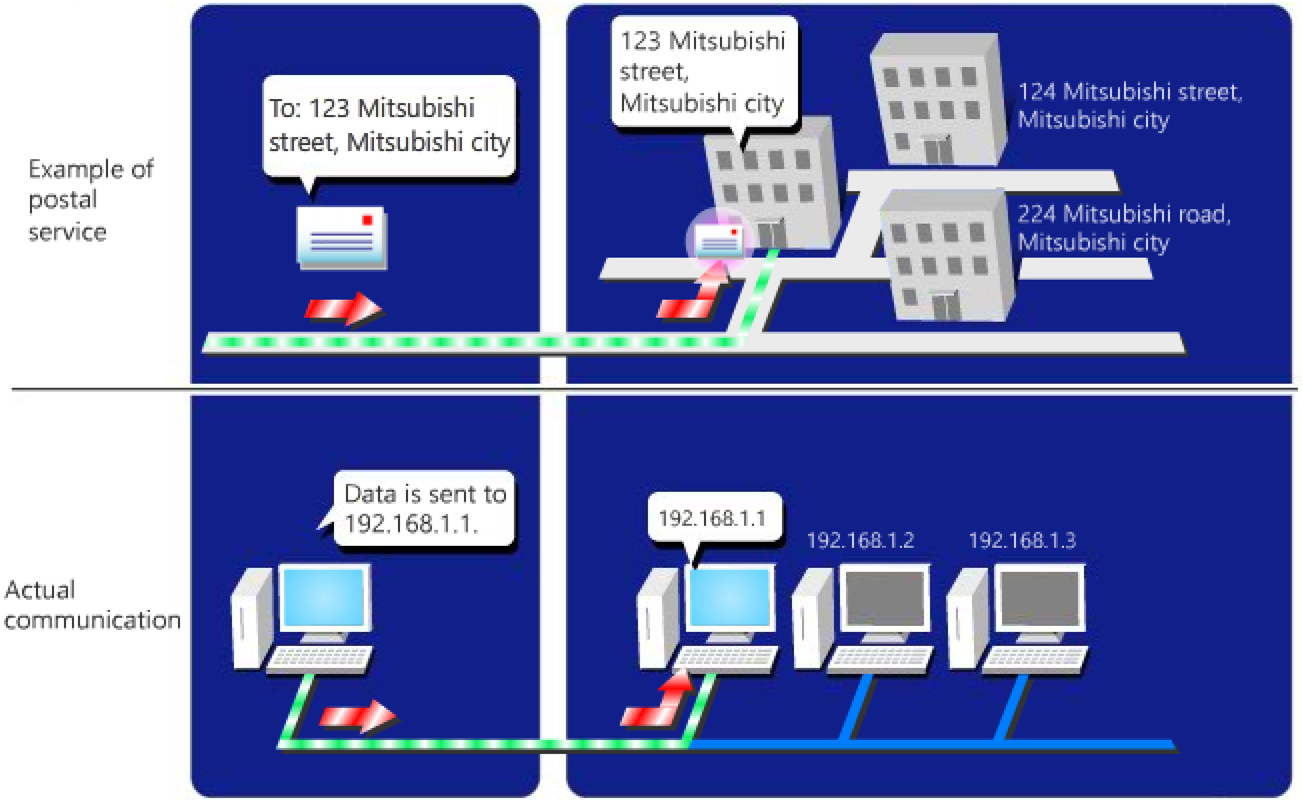
1.2 Basic knowledge about Ethernet

This section describes TCP/IP, which is one of the most commonly used protocols for Ethernet.

1.2.1 IP address

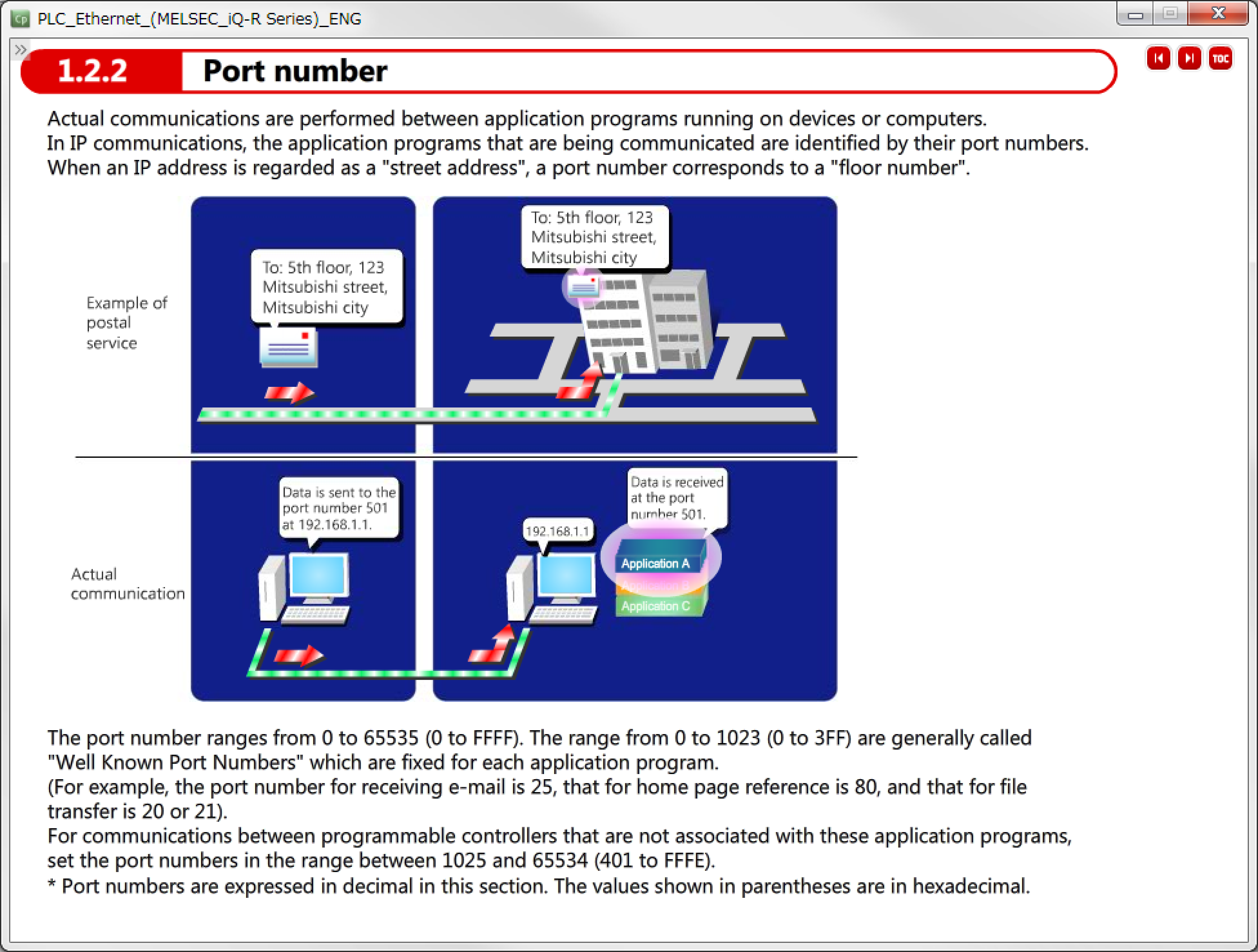
To perform communications between devices, both of the communication source and destination devices must be defined. As shown in the figure below, these are similar to a sender's address and a receiver's address on an envelope.

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


Example of postal service
To: 123 Mitsubishi street, Mitsubishi city
123 Mitsubishi street, Mitsubishi city
124 Mitsubishi street, Mitsubishi city
224 Mitsubishi road, Mitsubishi city
Actual communication
Data is sent to 192.168.1.1.
192.168.1.1
192.168.1.2
192.168.1.3

Note:

An IP address cannot be set arbitrarily. Before you connect a device to an existing network, it is necessary to consult the network administrator for assigning an IP address.


PLC_Ethernet_(MELSEC_iQ-R Series)_ENG
TOC

1.2.2 Port number

Actual communications are performed between application programs running on devices or computers.
In IP communications, the application programs that are being communicated are identified by their port numbers.
When an IP address is regarded as a "street address", a port number corresponds to a "floor number".

Example of postal service
To: 5th floor, 123 Mitsubishi street, Mitsubishi city
To: 5th floor, 123 Mitsubishi street, Mitsubishi city
Actual communication
Data is sent to the port number 501 at 192.168.1.1.
192.168.1.1
Data is received at the port number 501.
Application A
Application B
Application C

The port number ranges from 0 to 65535 (0 to FFFF). The range from 0 to 1023 (0 to 3FF) are generally called "Well Known Port Numbers" which are fixed for each application program.
(For example, the port number for receiving e-mail is 25, that for home page reference is 80, and that for file transfer is 20 or 21).
For communications between programmable controllers that are not associated with these application programs, set the port numbers in the range between 1025 and 65534 (401 to FFFE).
* Port numbers are expressed in decimal in this section. The values shown in parentheses are in hexadecimal.

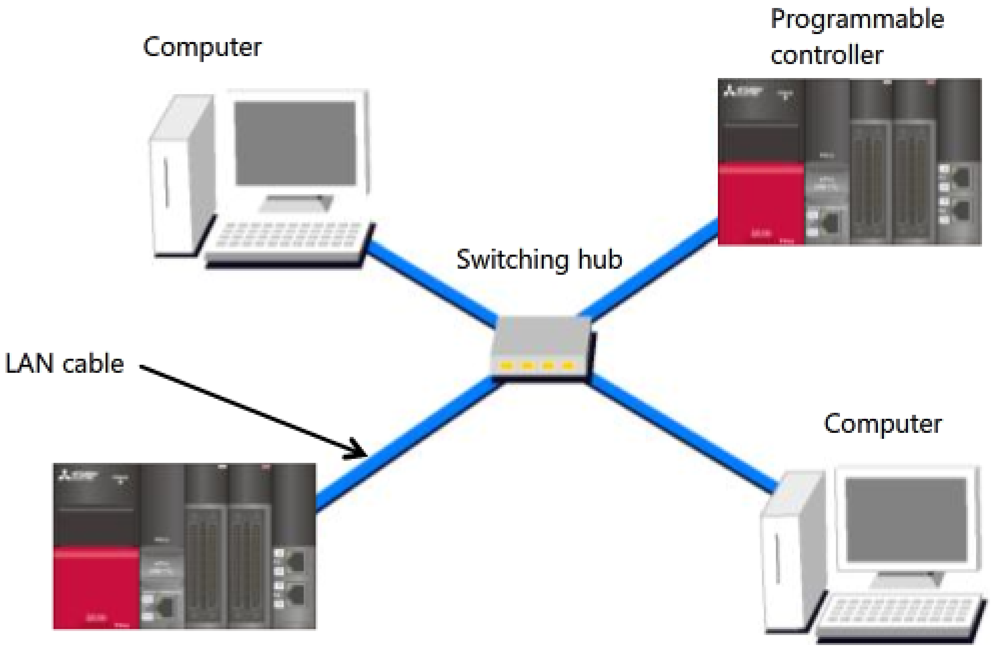
1.2.3 Wiring

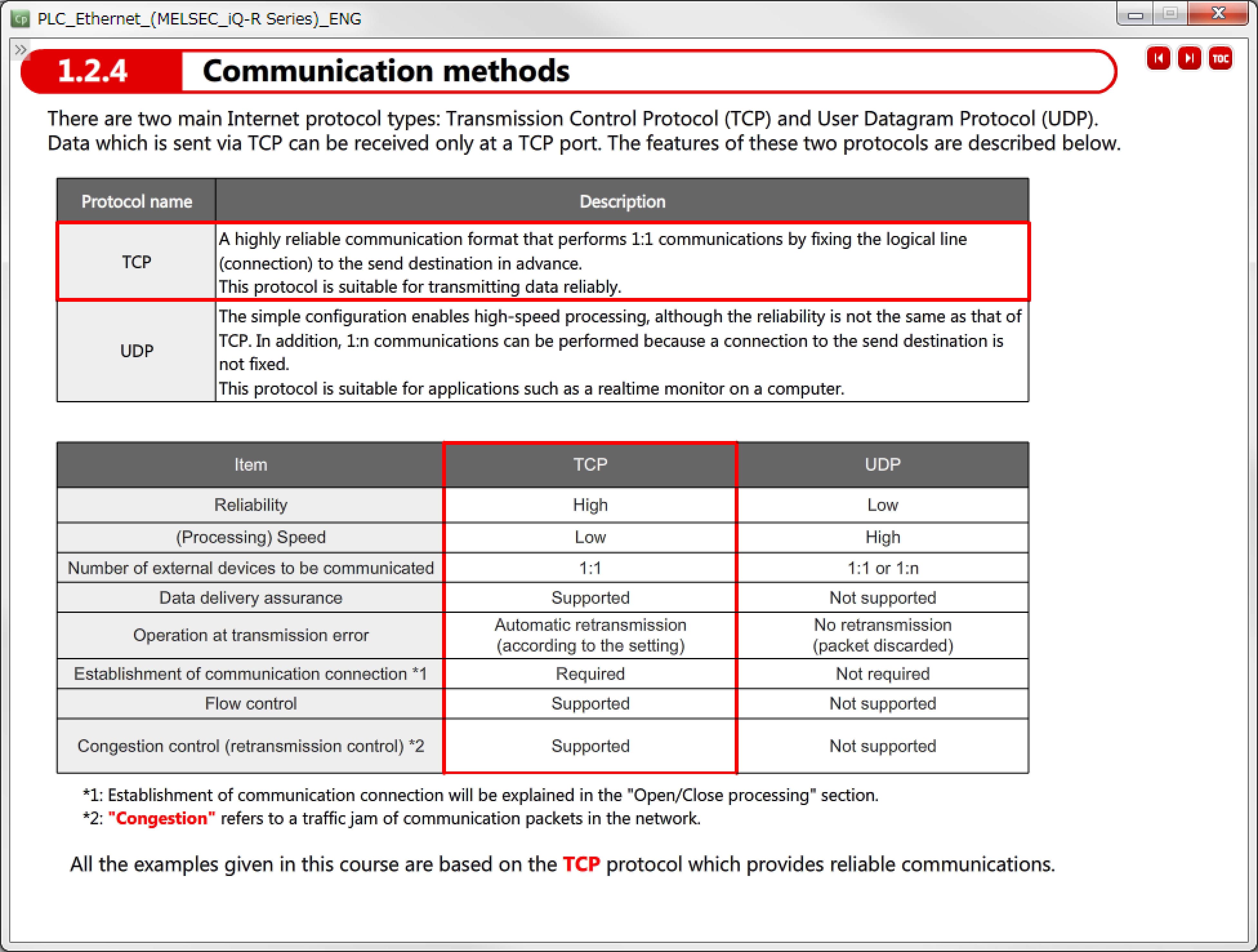
This section describes the most common example of the Ethernet connection.

The type that connection lines radiate in all directions as shown in the figure below is called a star topology.
In this type, a switching hub is used to shape, amplify, and control signals.

In this type, a failure occurred in a device is hard to affect the entire network.

Moreover, the required LAN cables are readily available.





1.2.4 Communication methods

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

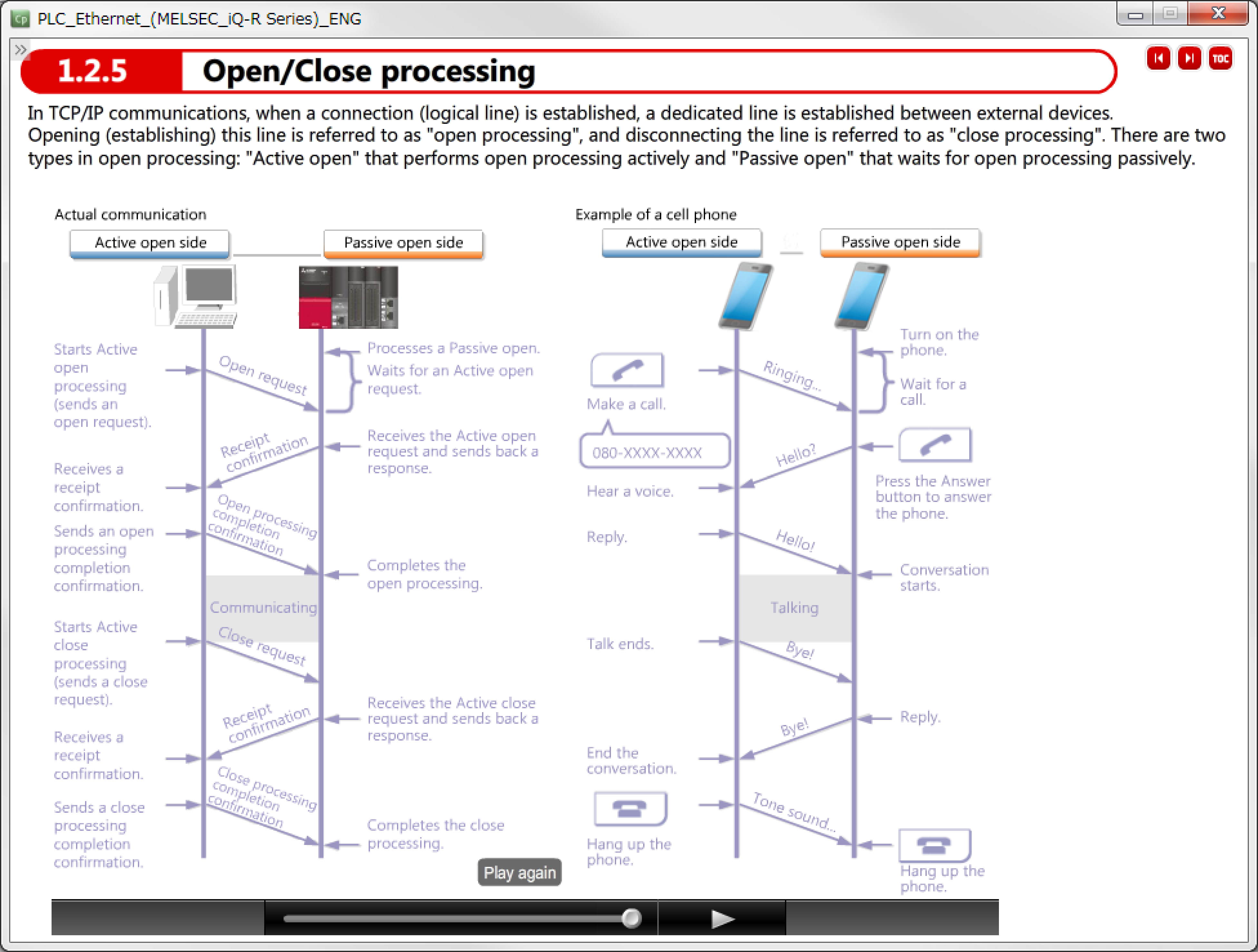
Protocol name	Description
TCP	A highly reliable communication format that performs 1:1 communications by fixing the logical line (connection) to the send destination in advance. This protocol is suitable for transmitting data reliably.
UDP	The simple configuration enables high-speed processing, although the reliability is not the same as that of TCP. In addition, 1:n communications can be performed because a connection to the send destination is not fixed. This protocol is suitable for applications such as a realtime monitor on a computer.

Item	TCP	UDP
Reliability	High	Low
(Processing) Speed	Low	High
Number of external devices to be communicated	1:1	1:1 or 1:n
Data delivery assurance	Supported	Not supported
Operation at transmission error	Automatic retransmission (according to the setting)	No retransmission (packet discarded)
Establishment of communication connection *1	Required	Not required
Flow control	Supported	Not supported
Congestion control (retransmission control) *2	Supported	Not supported

*1: Establishment of communication connection will be 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 which provides reliable communications.



1.2.5 Open/Close processing

In TCP/IP communications, when a connection (logical line) is established, a dedicated line is established between external devices. 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" that performs open processing actively and "Passive open" that waits for open processing passively.

Actual communication

Active open side

Passive open side

Active open side: Starts Active open processing (sends an open request).

Passive open side: Processes a Passive open. Waits for an Active open request.

Passive open side: Receives the Active open request and sends back a response.

Active open side: Receives a receipt confirmation.

Active open side: Sends an open processing completion confirmation.

Passive open side: Completes the open processing.

Active open side: Starts Active close processing (sends a close request).

Passive open side: Receives the Active close request and sends back a response.

Active open side: Receives a receipt confirmation.

Active open side: Sends a close processing completion confirmation.

Passive open side: Completes the close processing.

Example of a cell phone

Active open side

Passive open side

Passive open side: Turn on the phone. Wait for a call.

Active open side: Make a call.

Passive open side: Press the Answer button to answer the phone.

Active open side: Hear a voice.

Active open side: Reply.

Passive open side: Conversation starts.

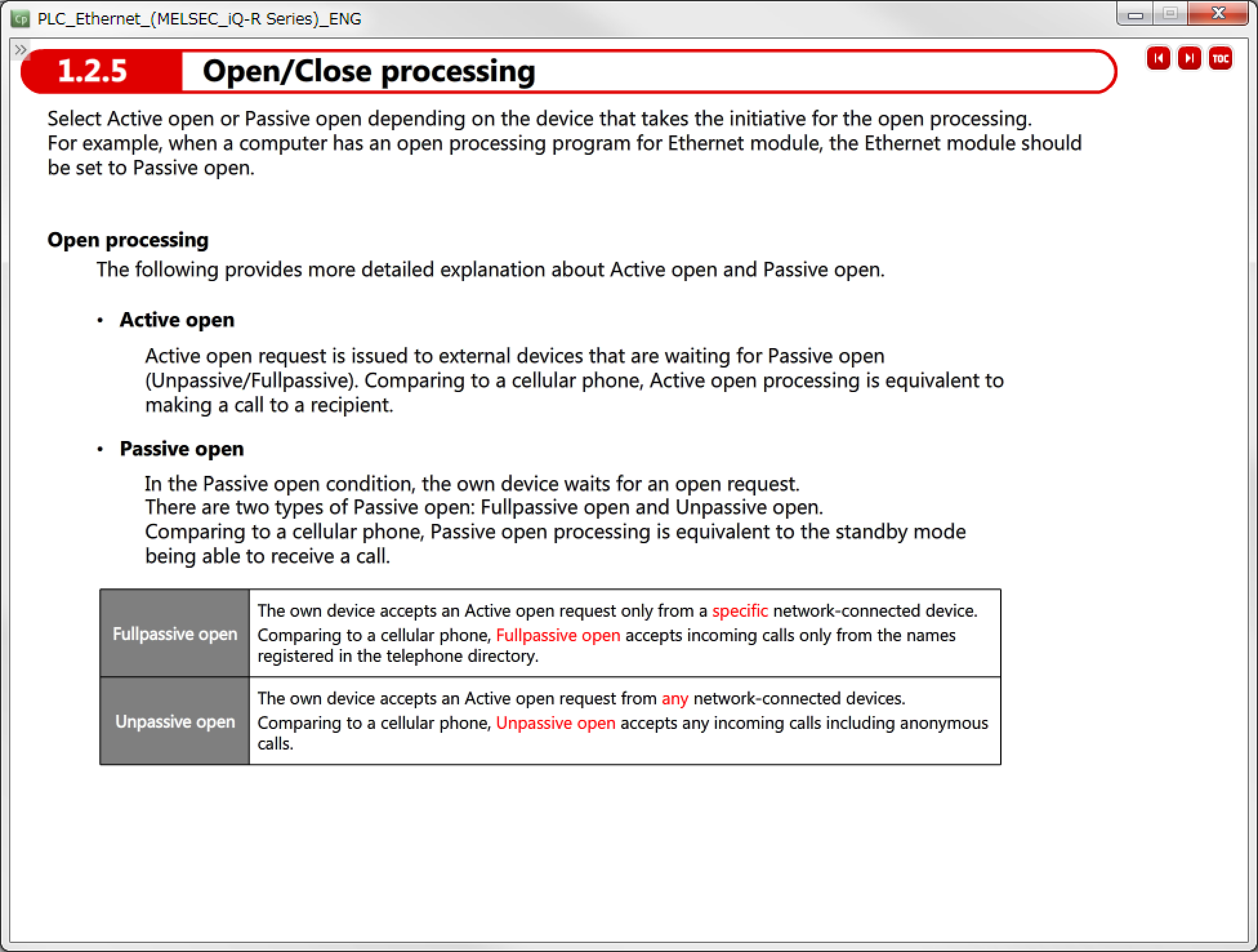
Active open side: Talk ends.

Passive open side: Reply.

Active open side: End the conversation.

Active open side: Hang up the phone.

Passive open side: Hang up the phone.



1.2.5 Open/Close processing

Select Active open or Passive open depending on the device that takes the initiative for the open processing.
For example, when a computer has an open processing program for Ethernet module, the Ethernet module should be set to Passive open.

Open processing

The following provides more detailed explanation about Active open and Passive open.

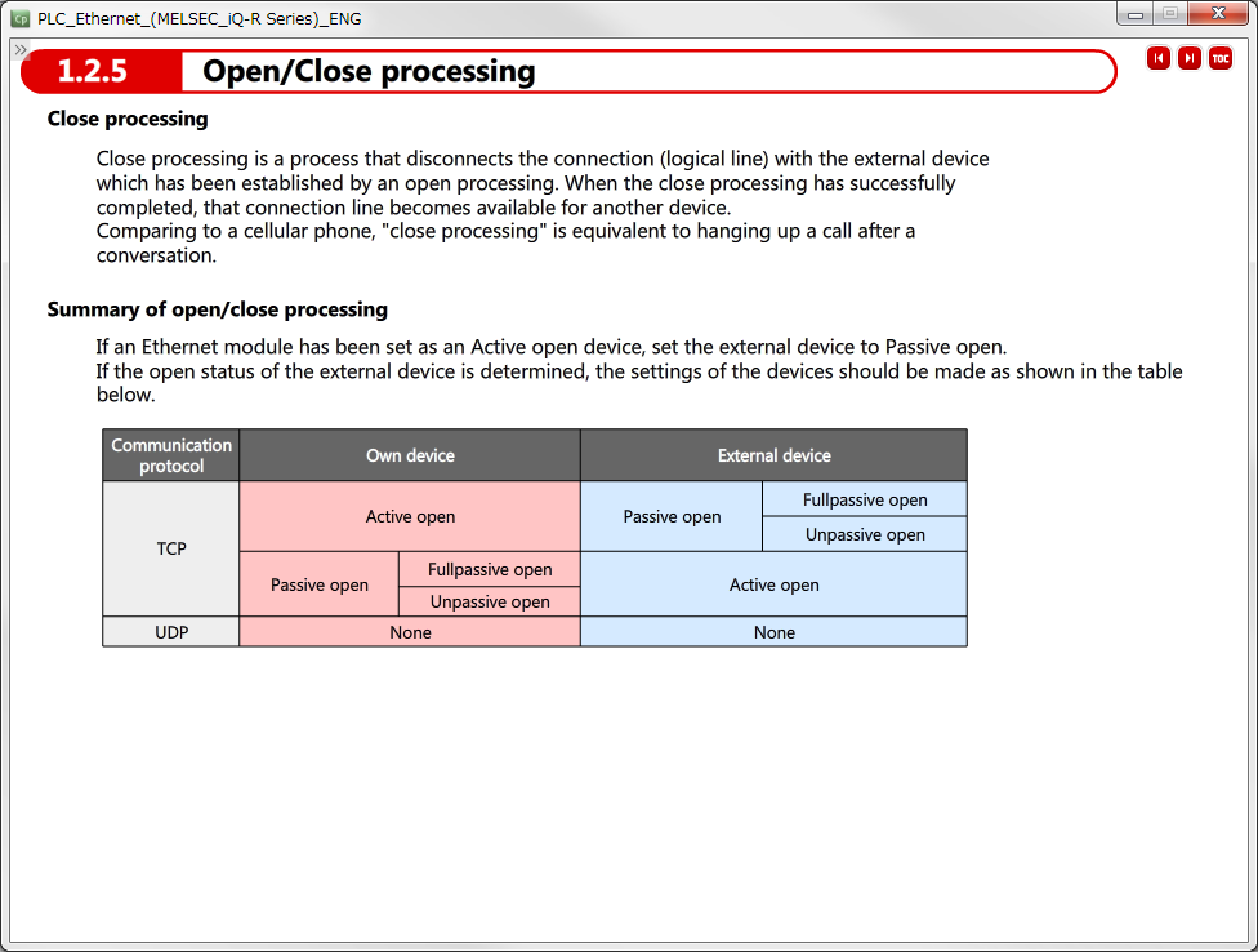
- **Active open**

  Active open request is issued to external devices that are waiting for Passive open (Unpassive/Fullpassive). Comparing to a cellular phone, Active open processing is equivalent to making a call to a recipient.

- **Passive open**

  In the Passive open condition, the own device waits for an open request.
  There are two types of Passive open: Fullpassive open and Unpassive open.
  Comparing to a cellular phone, Passive open processing is equivalent to the standby mode being able to receive a call.

Fullpassive open	The own device accepts an Active open request only from a specific network-connected device. Comparing to a cellular phone, Fullpassive open accepts incoming calls only from the names registered in the telephone directory.
Unpassive open	The own device accepts an Active open request from any network-connected devices. Comparing to a cellular phone, Unpassive open accepts any incoming calls including anonymous calls.



1.2.5 Open/Close processing

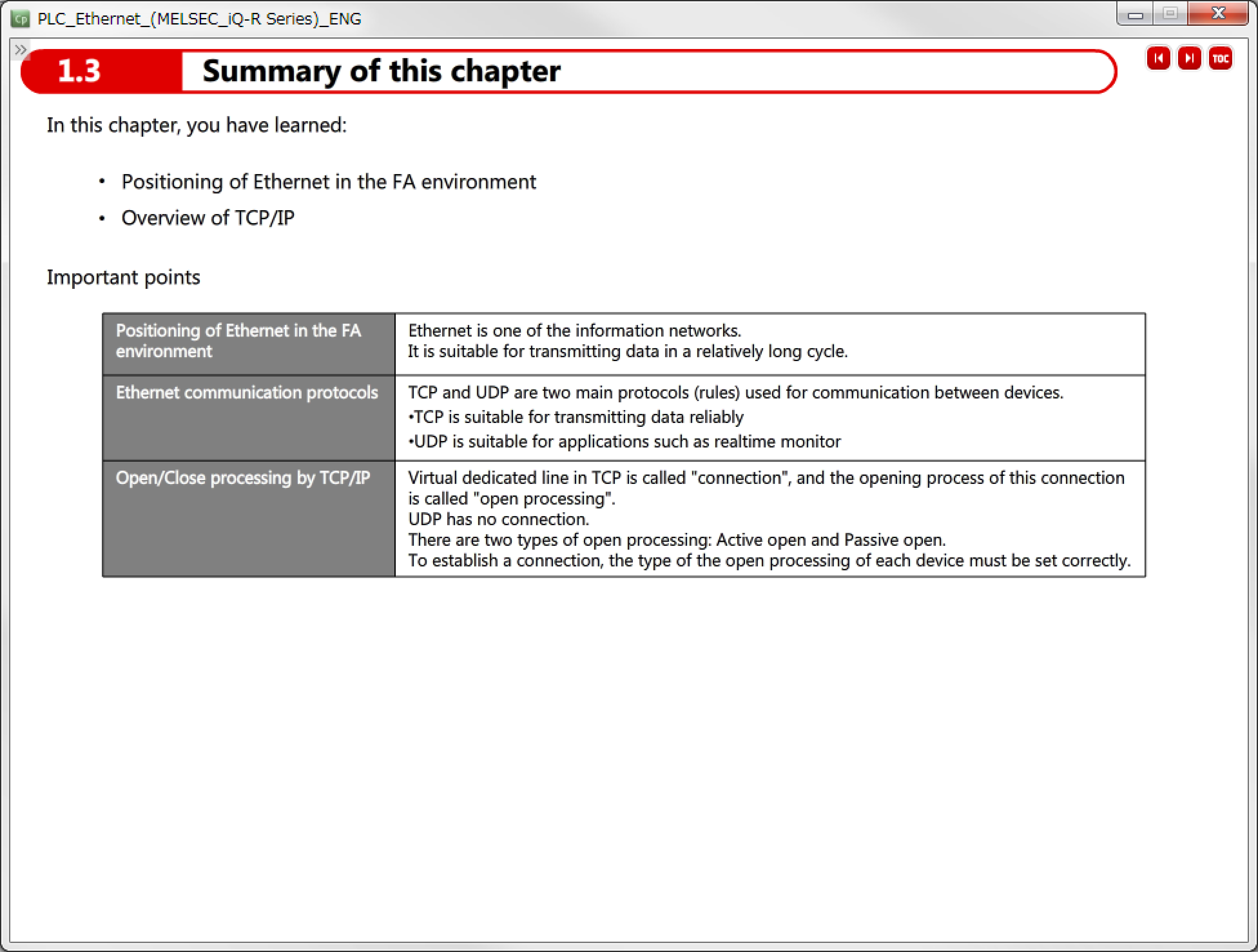
Close processing

Close processing is a process that disconnects the connection (logical line) with the external device which has been established by an open processing. When the close processing has successfully completed, that connection line becomes available for another device.
Comparing to a cellular phone, "close processing" is equivalent to hanging up a call after a conversation.

Summary of open/close processing

If an Ethernet module has been set as an Active open device, set the external device to Passive open.
If the open status of the external device is determined, the settings of the devices should be made as shown in the table below.

Communication protocol	Own device		External device	
TCP	Active open		Passive open	Fullpassive open
				Unpassive open
	Passive open	Fullpassive open	Active open	
		Unpassive open		
UDP	None		None	



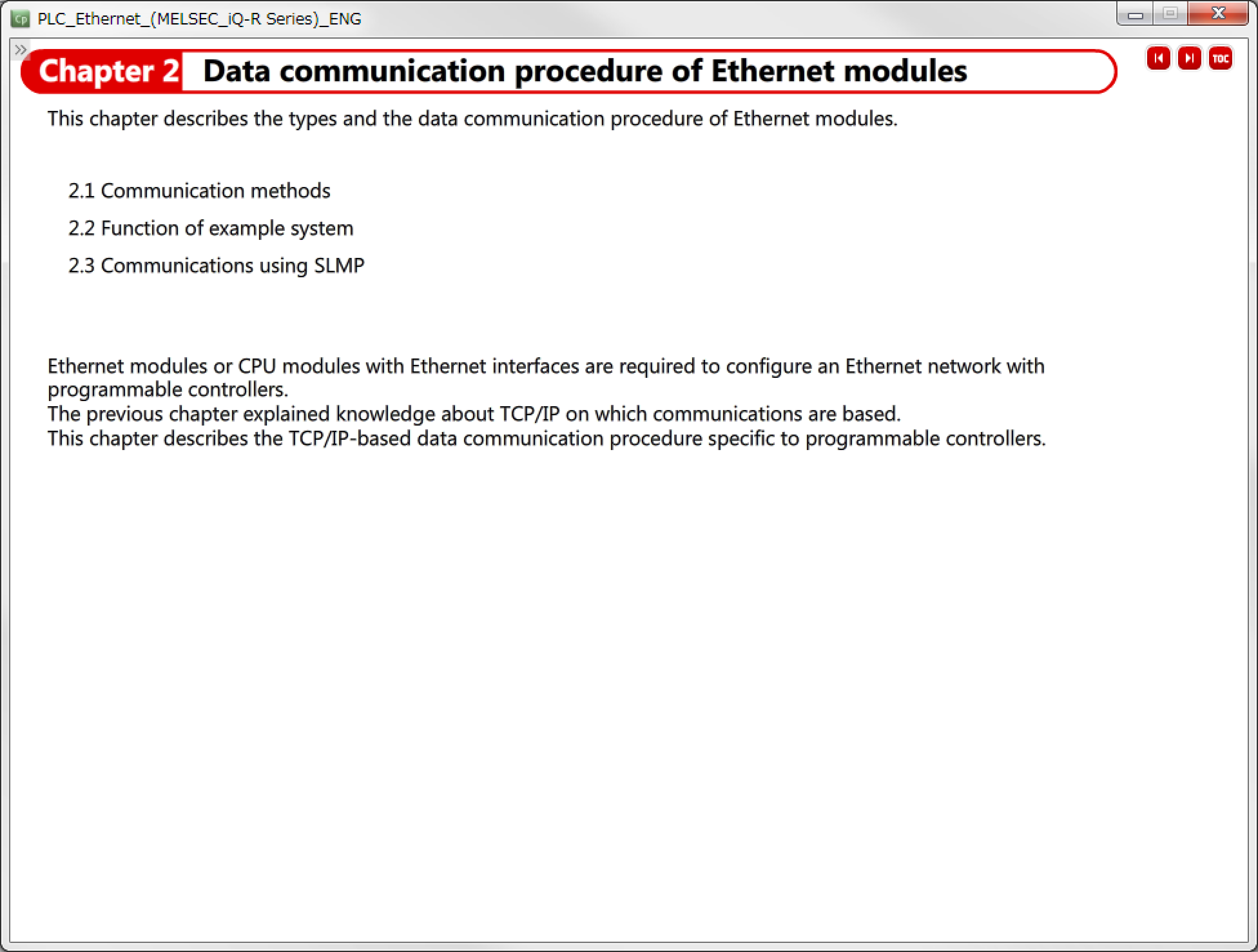
1.3 Summary of this chapter

In this chapter, you have learned:

- Positioning of Ethernet in the FA environment
- Overview of TCP/IP

Important points

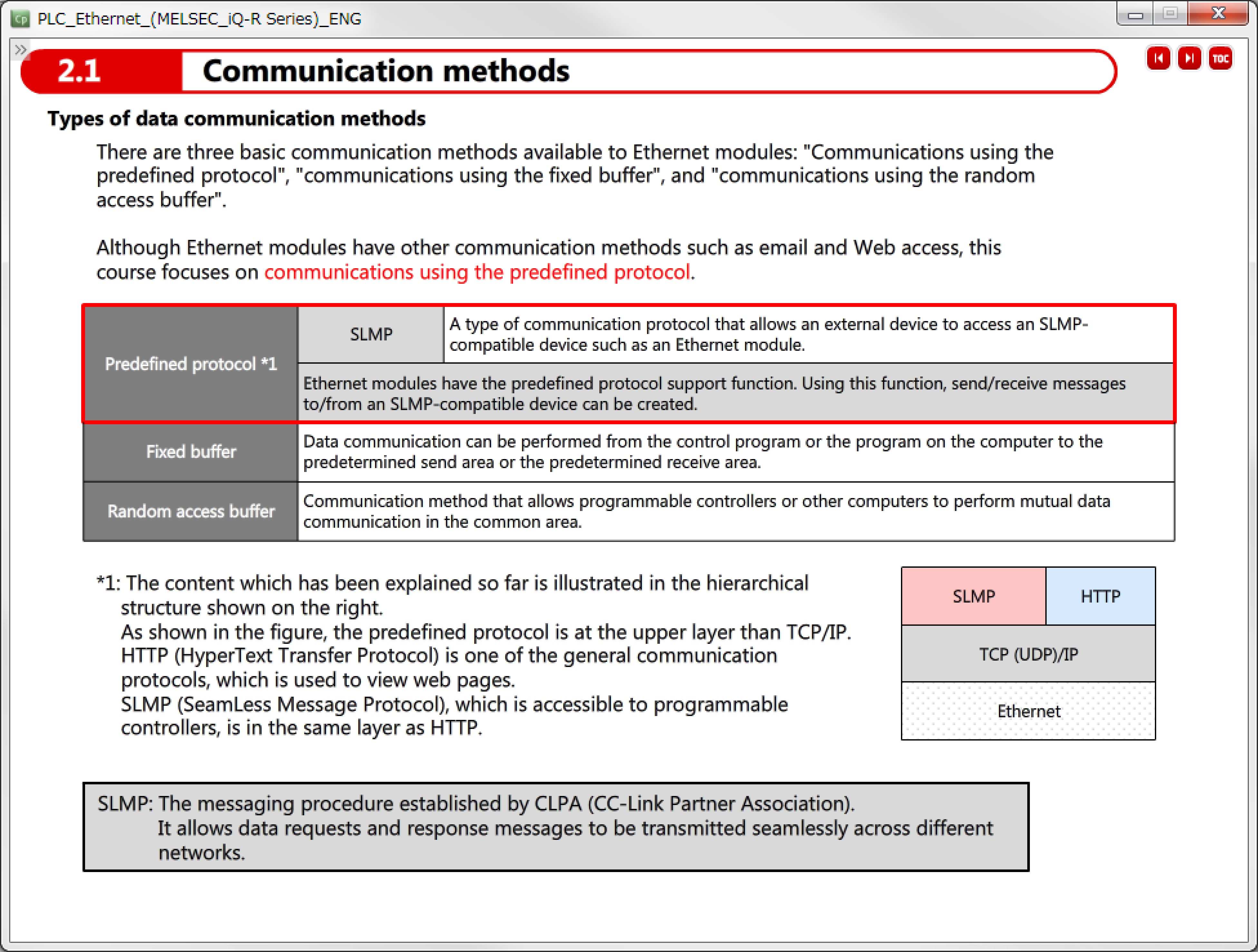
Positioning of Ethernet in the FA environment	Ethernet is one of the information networks. It is suitable for transmitting data in a relatively long cycle.
Ethernet communication protocols	TCP and UDP are two main protocols (rules) used for communication between devices. •TCP is suitable for transmitting data reliably •UDP is suitable for applications such as realtime monitor
Open/Close processing by TCP/IP	Virtual dedicated line in TCP is called "connection", and the opening process of this connection is called "open processing". UDP has no connection. There are two types of open processing: Active open and Passive open. To establish a connection, the type of the open processing of each device must be set correctly.



Chapter 2 Data communication procedure of Ethernet modules

This chapter describes the types and the data communication procedure of Ethernet modules.

Ethernet modules or CPU modules with Ethernet interfaces are required to configure an Ethernet network with programmable controllers.
The previous chapter explained knowledge about TCP/IP on which communications are based.
This chapter describes the TCP/IP-based data communication procedure specific to programmable controllers.



2.1 Communication methods

Types of data communication methods

There are three basic communication methods available to Ethernet modules: "Communications using the predefined protocol", "communications using the fixed buffer", and "communications using the random access buffer".

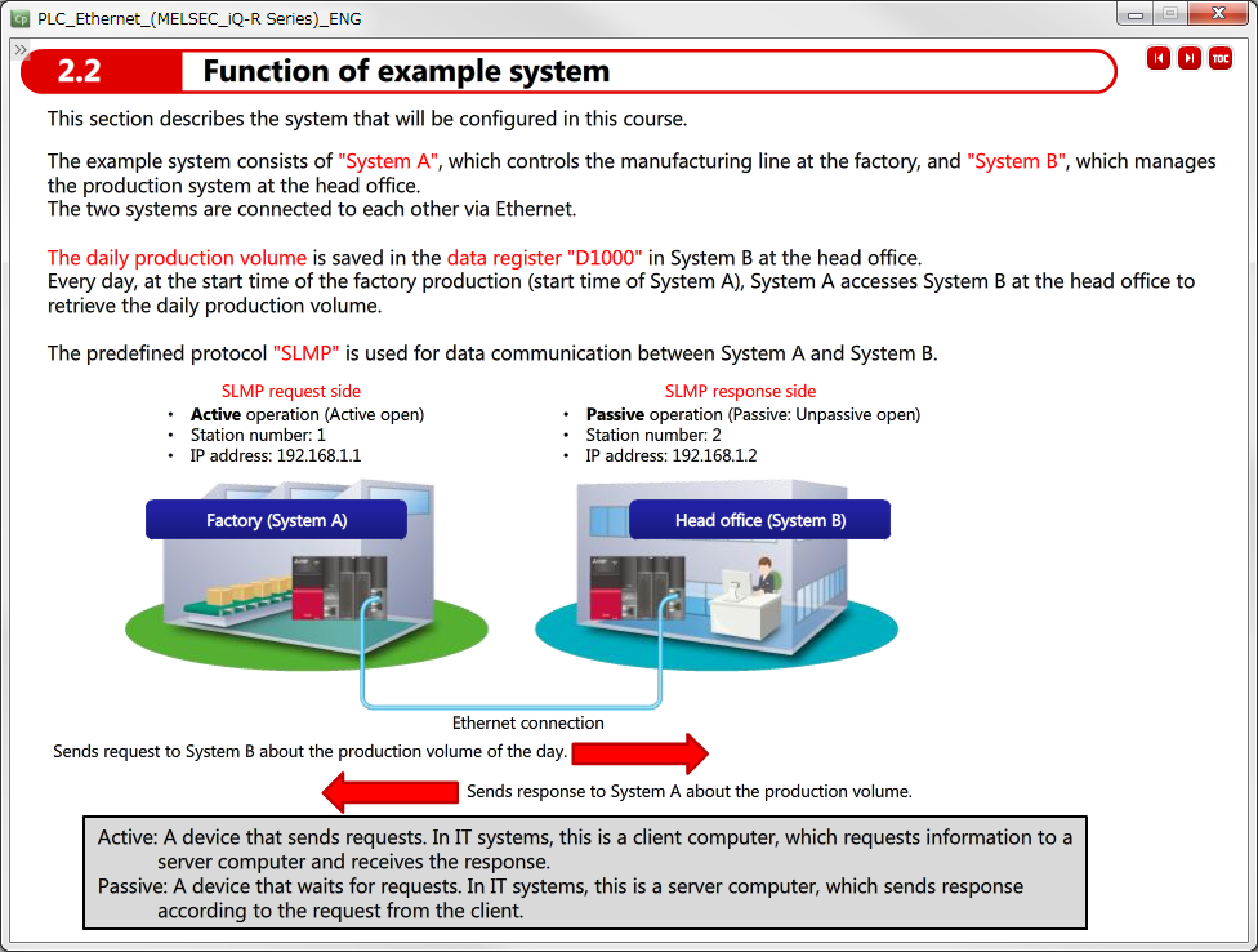
Although Ethernet modules have other communication methods such as email and Web access, this course focuses on communications using the predefined protocol.

Predefined protocol *1	SLMP	A type of communication protocol that allows an external device to access an SLMP-compatible device such as an Ethernet module.
	Ethernet modules have the predefined protocol support function. Using this function, send/receive messages to/from an SLMP-compatible device can be created.	
Fixed buffer	Data communication can be performed from the control program or the program on the computer to the predetermined send area or the predetermined receive area.	
Random access buffer	Communication method that allows programmable controllers or other computers to perform mutual data communication in the common area.	

*1: The content which has been explained so far is illustrated in the hierarchical structure shown on the right.
As shown in the figure, the predefined protocol is at the upper layer than TCP/IP.
HTTP (HyperText Transfer Protocol) is one of the general communication protocols, which is used to view web pages.
SLMP (SeamLess Message Protocol), which is accessible to programmable controllers, is in the same layer as HTTP.

SLMP	HTTP
TCP (UDP)/IP	
Ethernet	

SLMP: The messaging procedure established by CLPA (CC-Link Partner Association).
It allows data requests and response messages to be transmitted seamlessly across different networks.



2.2 Function of example system

This section describes the system that will be configured in this course.

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

The daily production volume is saved in the data register "D1000" in System B at the head office.
Every day, at the start time of the factory production (start time of System A), System A accesses System B at the head office to retrieve the daily production volume.

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

SLMP request side

- **Active** operation (Active open)
- Station number: 1
- IP address: 192.168.1.1

SLMP response side

- **Passive** operation (Passive: Unpassive open)
- Station number: 2
- IP address: 192.168.1.2

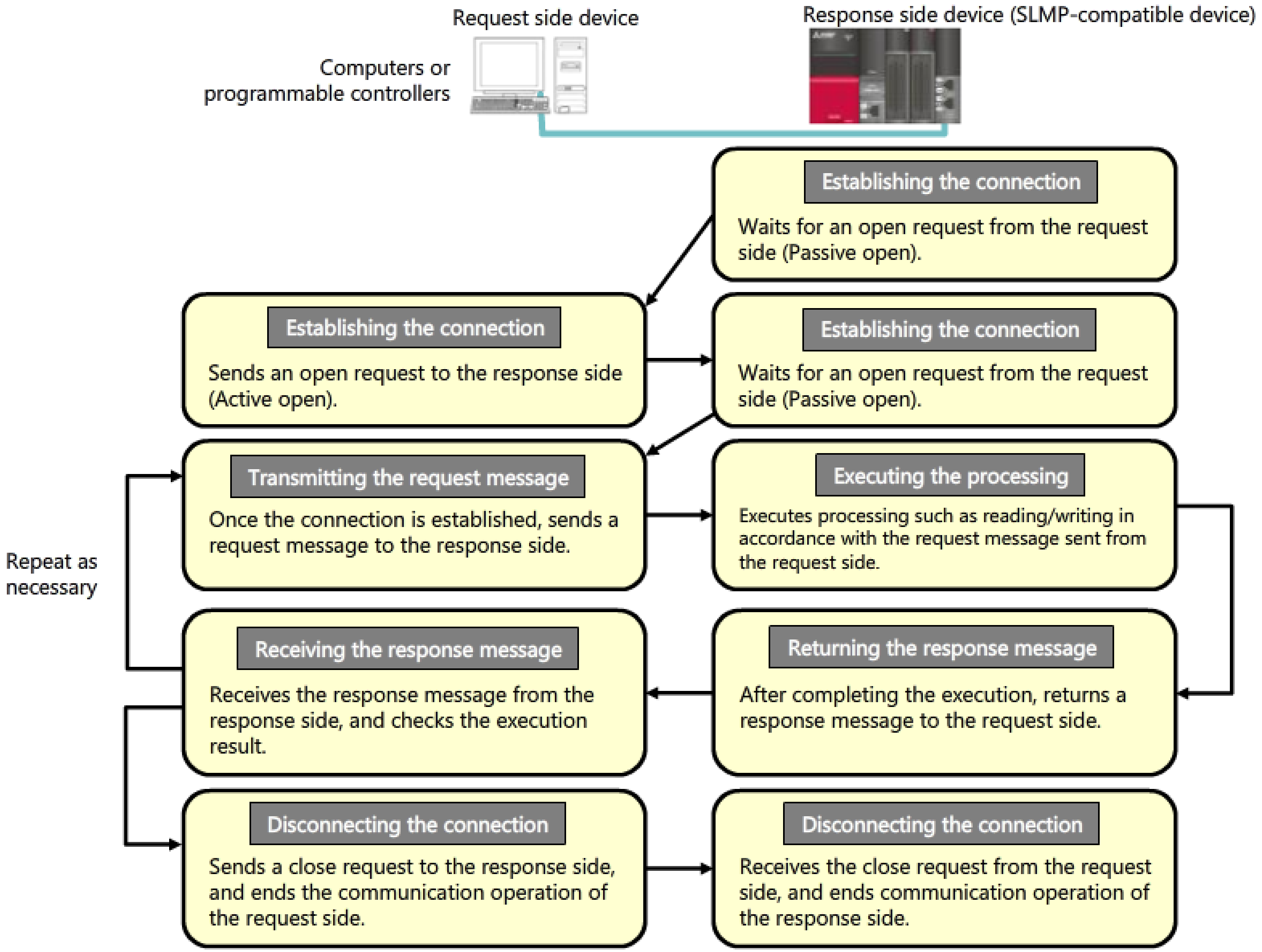
Sends request to System B about the production volume of the day.

Sends response to System A about the production volume.

Active: A device that sends requests. In IT systems, this is a client computer, which requests information to a server computer and receives the response.
Passive: A device that waits for requests. In IT systems, this is a server computer, which sends response according to the request from the client.

2.3 Communications using SLMP

When devices are communicating using SLMP, the data request side and the response side communicate with each other as shown below.



Request side device

Computers or programmable controllers

Response side device (SLMP-compatible device)

Establishing the connection

Waits for an open request from the request side (Passive open).

Establishing the connection

Sends an open request to the response side (Active open).

Establishing the connection

Waits for an open request from the request side (Passive open).

Repeat as necessary

Transmitting the request message

Once the connection is established, sends a request message to the response side.

Executing the processing

Executes processing such as reading/writing in accordance with the request message sent from the request side.

Receiving the response message

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

Returning the response message

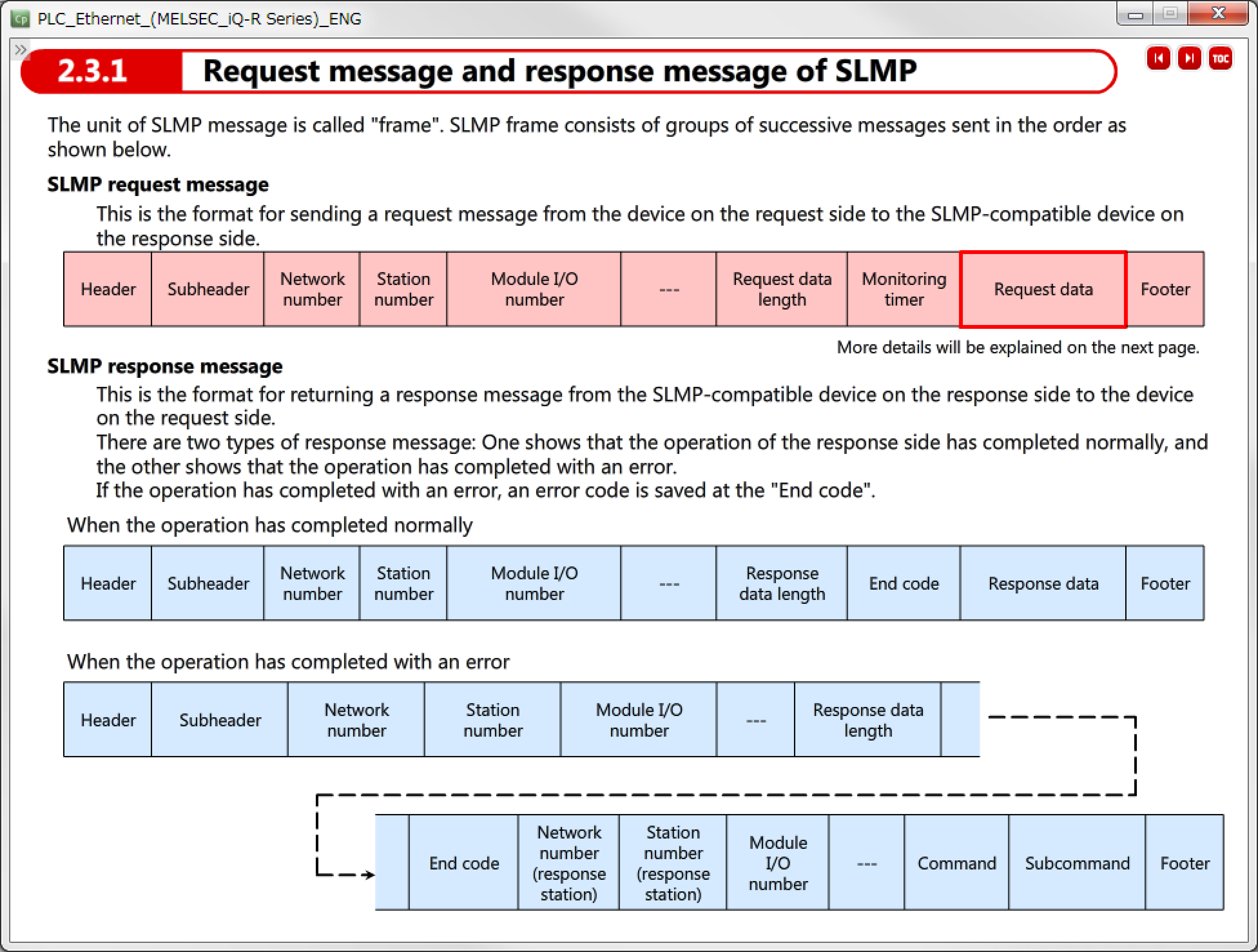
After completing the execution, returns a response message to the request side.

Disconnecting the connection

Sends a close request to the response side, and ends the communication operation of the request side.

Disconnecting the connection

Receives the close request from the request side, and ends communication operation of the response side.



2.3.1 Request message and response message of SLMP

The unit of SLMP message is called "frame". SLMP frame consists of groups of successive messages sent in the order as shown below.

SLMP request message

This is the format for sending a request message from the device on the request side to the SLMP-compatible device on the response side.

Header	Subheader	Network number	Station number	Module I/O number	---	Request data length	Monitoring timer	Request data	Footer

More details will be explained on the next page.

SLMP response message

This is the format for returning a response message from the SLMP-compatible device on the response side to the device on the request side.

There are two types of response message: One shows that the operation of the response side has completed normally, and the other shows that the operation has completed with an error.

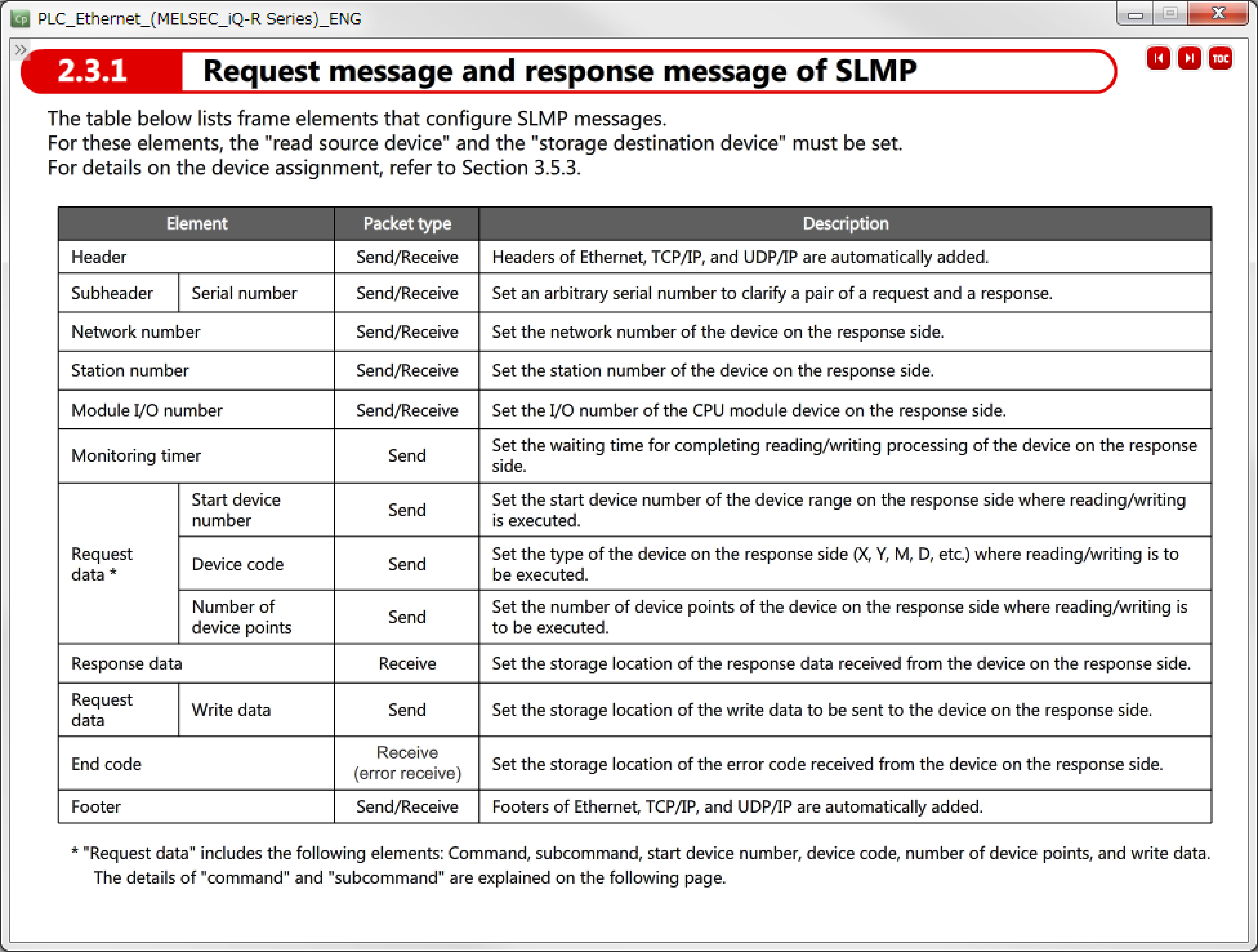
If the operation has completed with an error, an error code is saved at the "End code".

When the operation has completed normally

Header	Subheader	Network number	Station number	Module I/O number	---	Response data length	End code	Response data	Footer

When the operation has completed with an error

Header	Subheader	Network number	Station number	Module I/O number	---	Response data length		End code	Network number (response station)	Station number (response station)	Module I/O number	---	Command	Subcommand	Footer

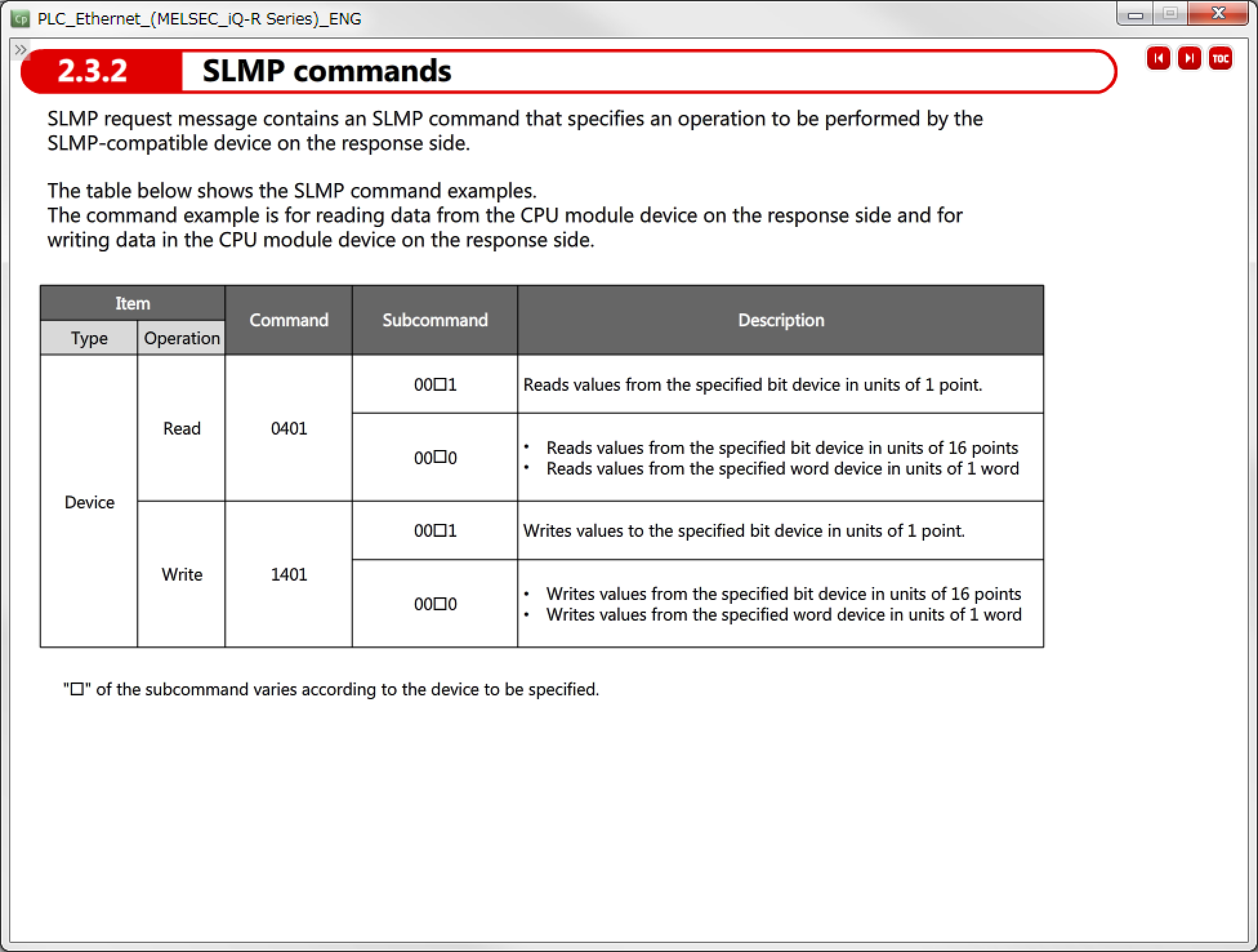


2.3.1 Request message and response message of SLMP

The table below lists frame elements that configure SLMP messages.
For these elements, the "read source device" and the "storage destination device" must be set.
For details on the device assignment, refer to Section 3.5.3.

Element		Packet type	Description
Header		Send/Receive	Headers of Ethernet, TCP/IP, and UDP/IP are automatically added.
Subheader	Serial number	Send/Receive	Set an arbitrary serial number to clarify a pair of a request and a response.
Network number		Send/Receive	Set the network number of the device on the response side.
Station number		Send/Receive	Set the station number of the device on the response side.
Module I/O number		Send/Receive	Set the I/O number of the CPU module device on the response side.
Monitoring timer		Send	Set the waiting time for completing reading/writing processing of the device on the response side.
Request data *	Start device number	Send	Set the start device number of the device range on the response side where reading/writing is executed.
	Device code	Send	Set the type of the device on the response side (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 device on the response side where reading/writing is to be executed.
Response data		Receive	Set the storage location of the response data received from the device on the response side.
Request data	Write data	Send	Set the storage location of the write data to be sent to the device on the response side.
End code		Receive (error receive)	Set the storage location of the error code received from the device on the response side.
Footer		Send/Receive	Footers of Ethernet, TCP/IP, and UDP/IP are automatically added.

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



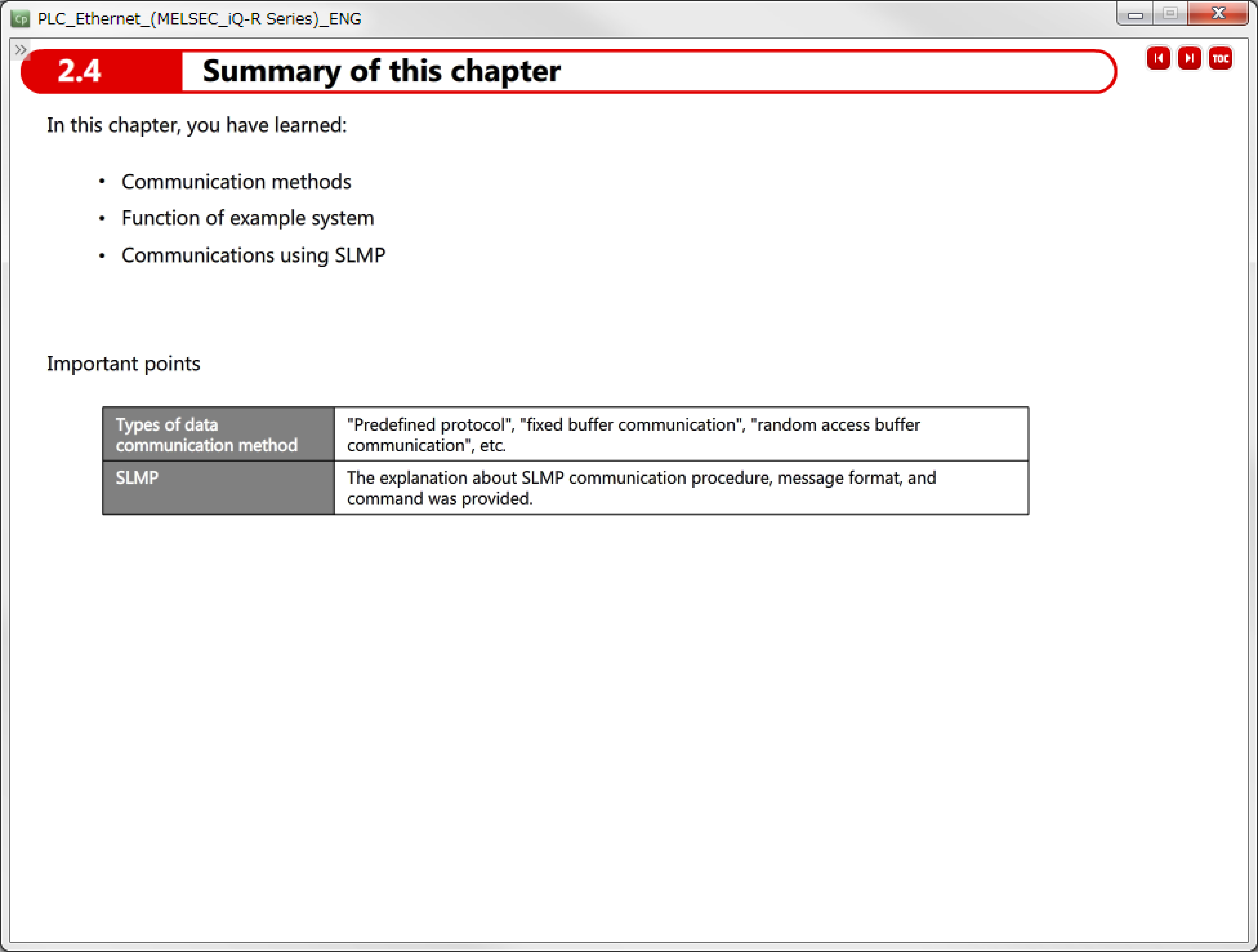
2.3.2 SLMP commands

SLMP request message contains an SLMP command that specifies an operation to be performed by the SLMP-compatible device on the response side.

The table below shows the SLMP command examples.
The command example is for reading data from the CPU module device on the response side and for writing data in the CPU module device on the response side.

Item		Command	Subcommand	Description
Type	Operation			
Device	Read	0401	00□1	Reads values from the specified bit device in units of 1 point.
			00□0	• Reads values from the specified bit device in units of 16 points • Reads values from the specified word device in units of 1 word
	Write	1401	00□1	Writes values to the specified bit device in units of 1 point.
			00□0	• Writes values from the specified bit device in units of 16 points • Writes values from the specified word device in units of 1 word

"□" of the subcommand varies according to the device to be specified.



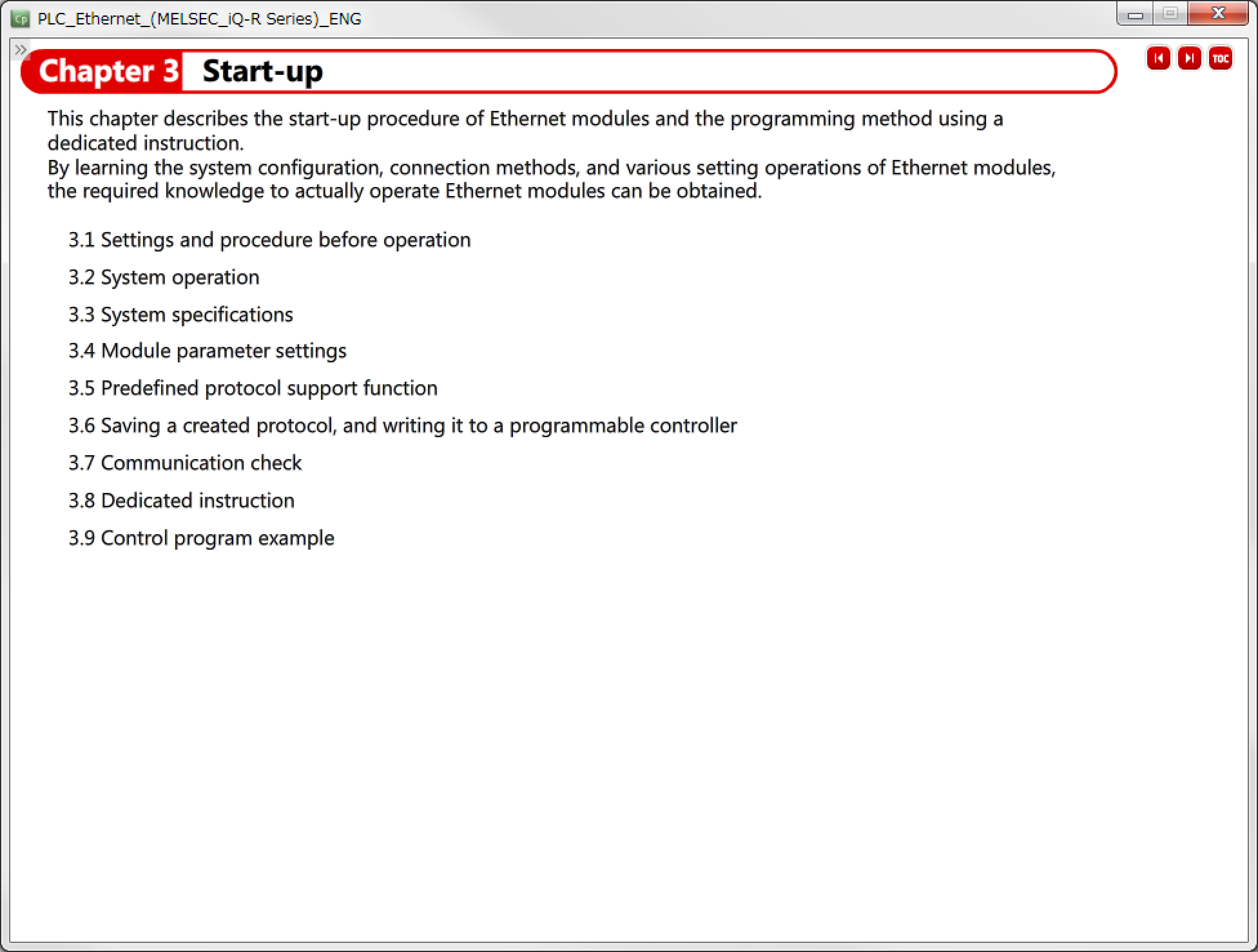
2.4 Summary of this chapter

In this chapter, you have learned:

- Communication methods
- Function of example system
- Communications using SLMP

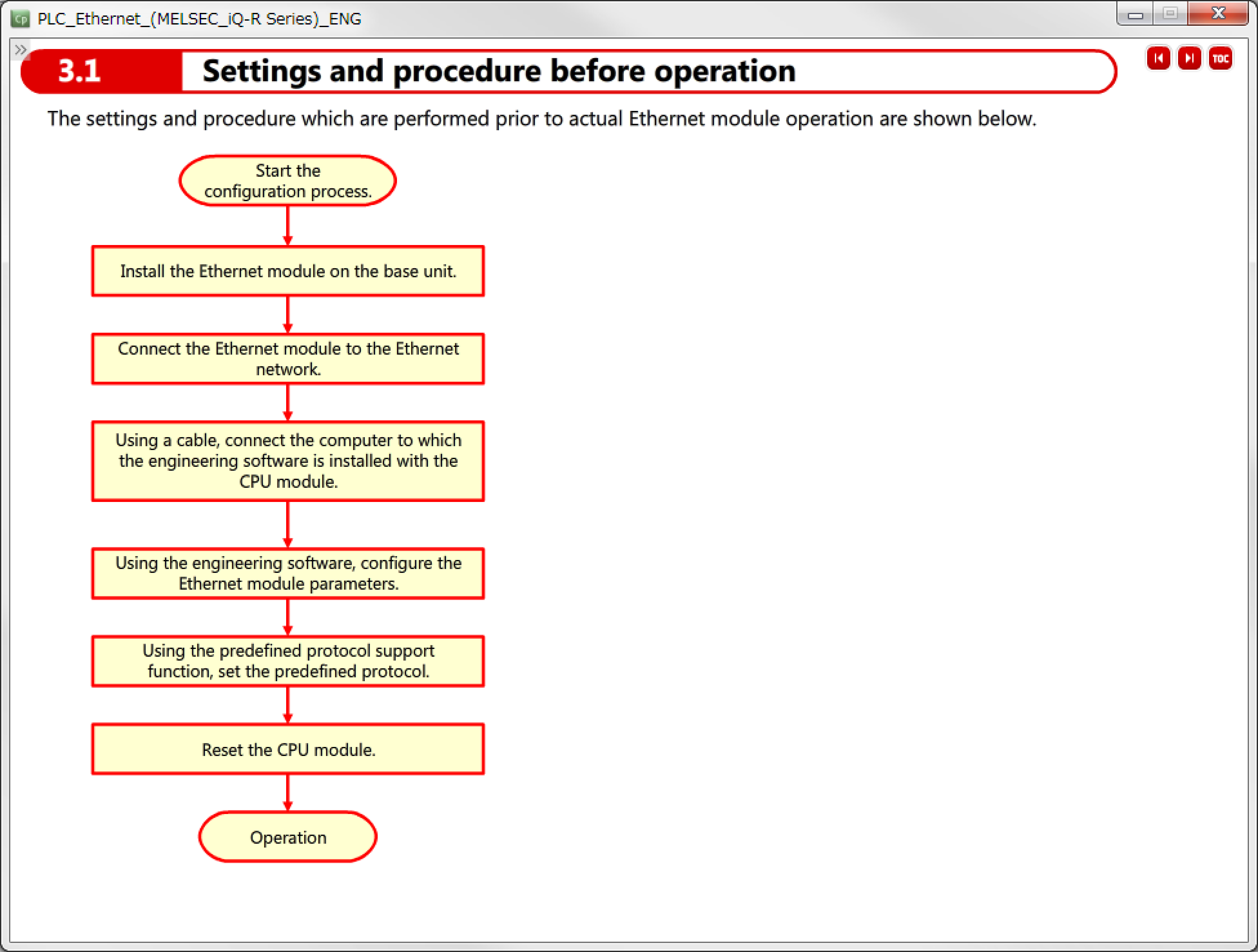
Important points

Types of data communication method	"Predefined protocol", "fixed buffer communication", "random access buffer communication", etc.
SLMP	The explanation about SLMP communication procedure, message format, and command was provided.



Chapter 3 Start-up

This chapter describes the start-up procedure of Ethernet modules and the programming method using a dedicated instruction.
By learning the system configuration, connection methods, and various setting operations of Ethernet modules, the required knowledge to actually operate Ethernet modules can be obtained.



3.1 Settings and procedure before operation

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

Start the configuration process.

Install the Ethernet module on the base unit.

Connect the Ethernet module to the Ethernet network.

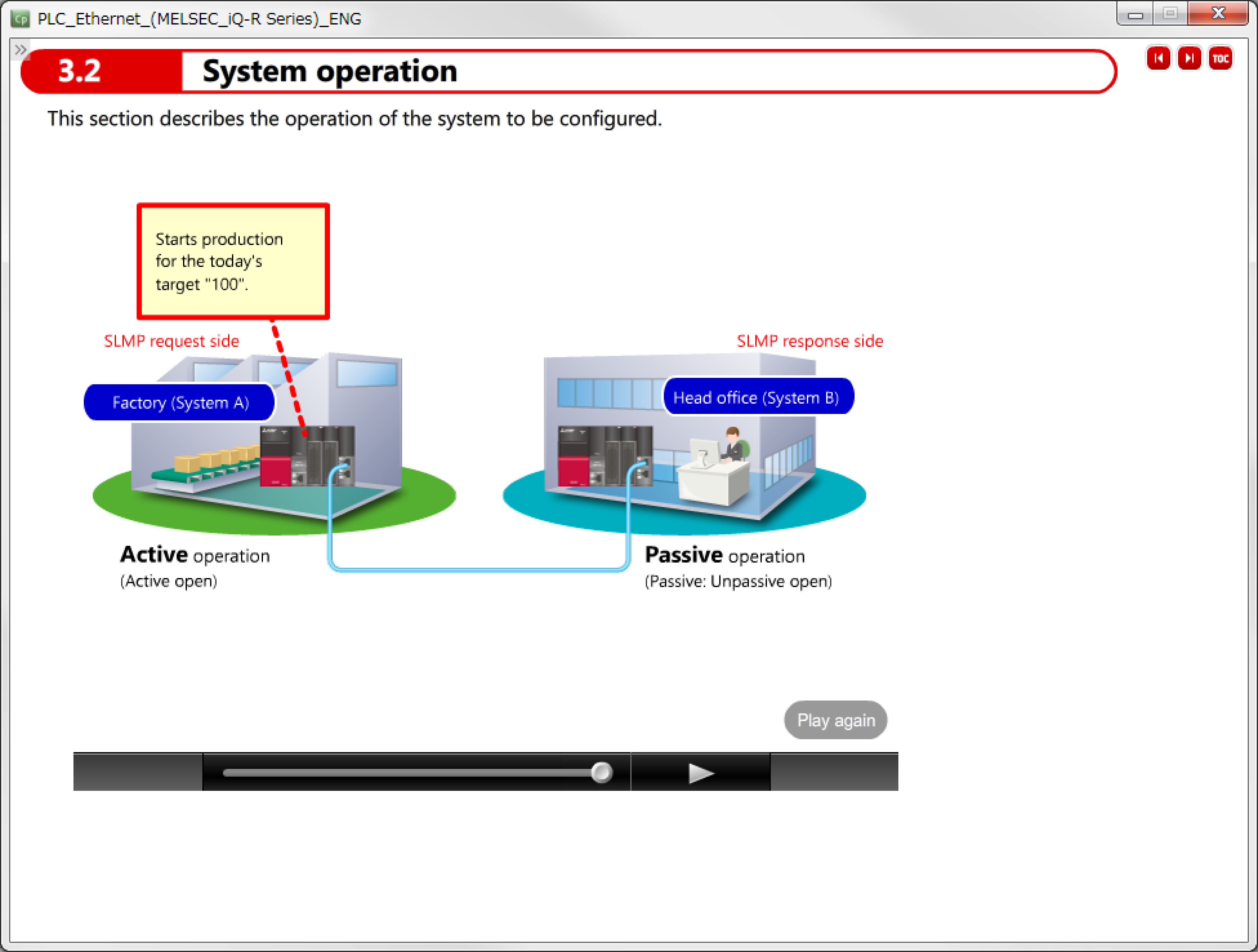
Using a cable, connect the computer to which the engineering software is installed with the CPU module.

Using the engineering software, configure the Ethernet module parameters.

Using the predefined protocol support function, set the predefined protocol.

Reset the CPU module.

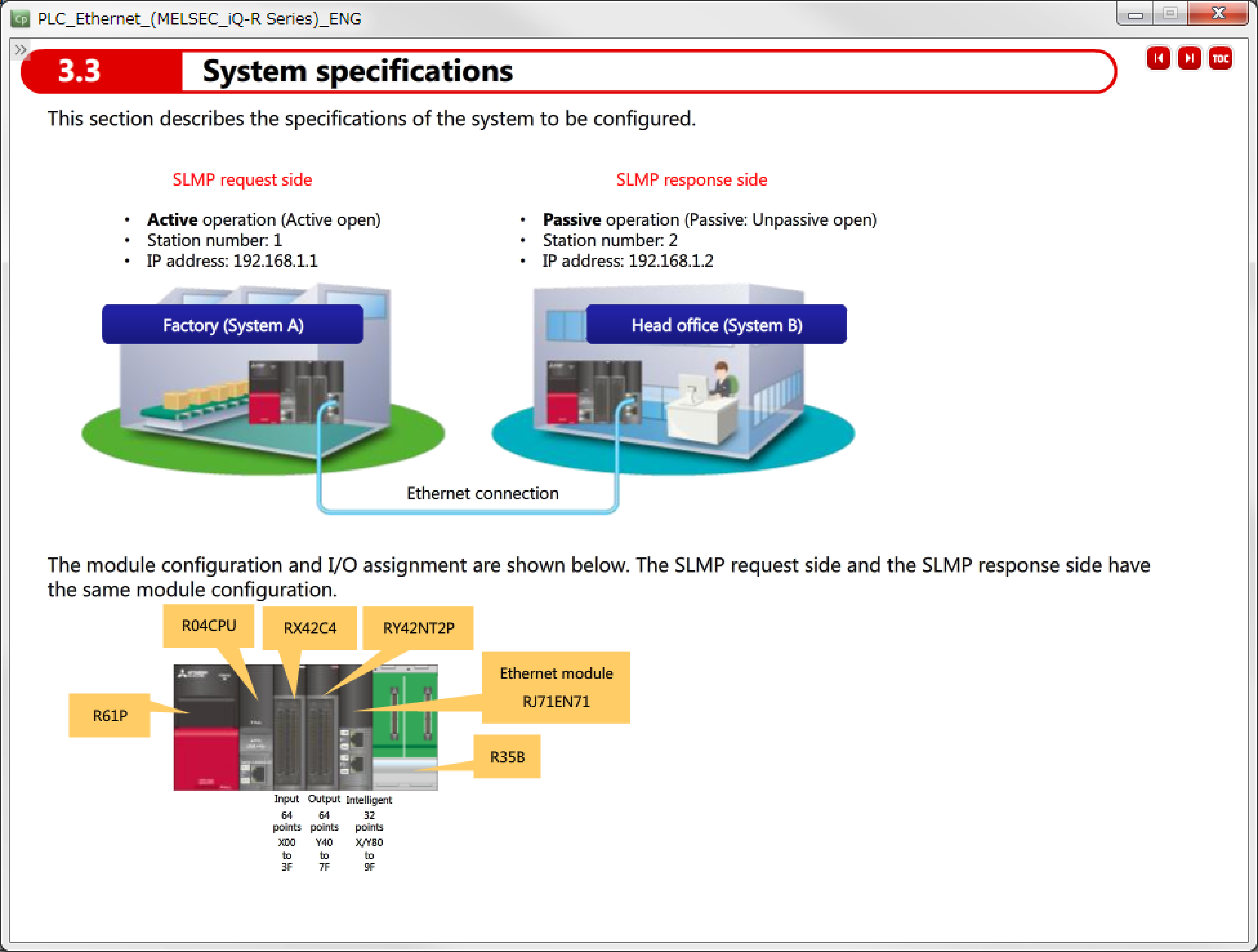
Operation



3.2 System operation

This section describes the operation of the system to be configured.

Starts production for the today's target "100".



3.3 System specifications

This section describes the specifications of the system to be configured.

SLMP request side

- **Active** operation (Active open)
- Station number: 1
- IP address: 192.168.1.1

SLMP response side

- **Passive** operation (Passive: Unpassive open)
- Station number: 2
- IP address: 192.168.1.2

The module configuration and I/O assignment are shown below. The SLMP request side and the SLMP response side have the same module configuration.

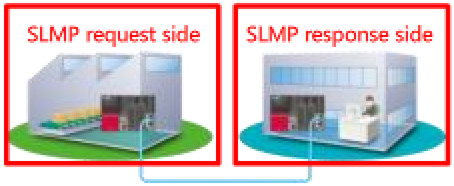
3.4 Module parameter settings

MELSOFT GX Works3 engineering software is used to configure module parameters.
Module parameters should be configured both on the SLMP request side and on the SLMP response side.

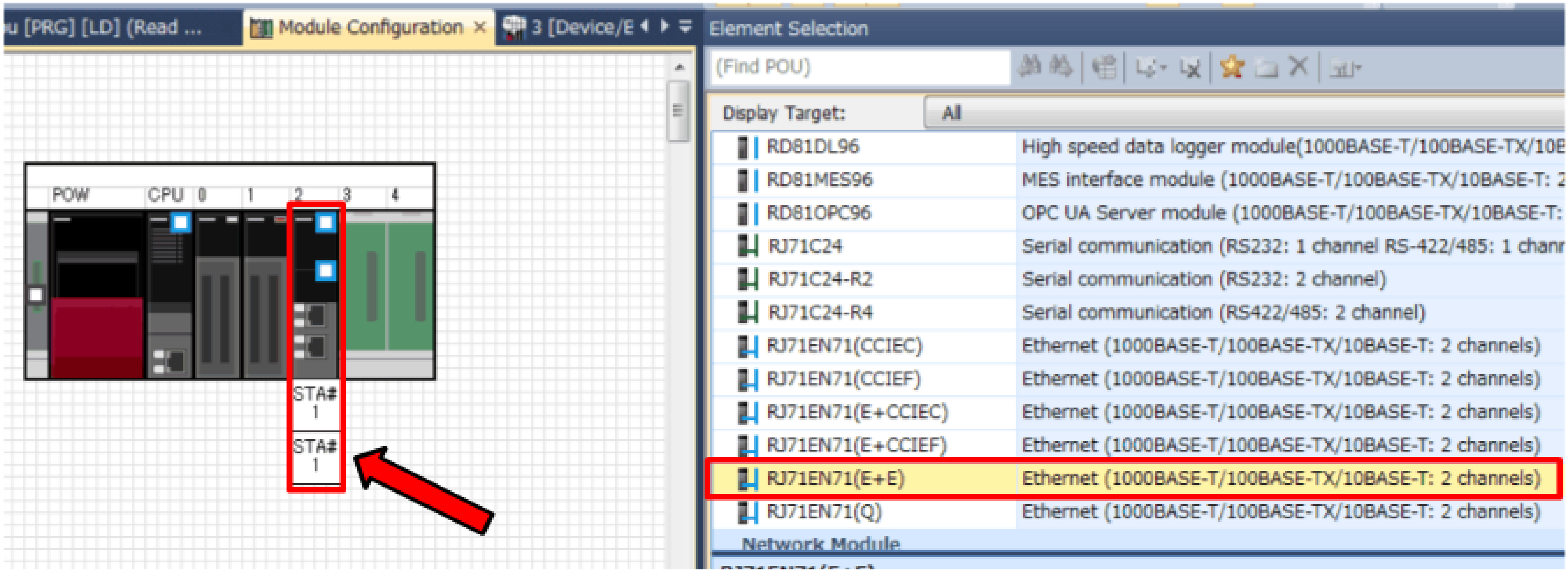
Configuring module parameters enables communication with external devices without using a control program.

3.4.1 Network module arrangement

Arrange the program elements of modules in the module configuration diagram in accordance with the network type.



The information within the parentheses in network module model names, such as "RJ71EN71(*****)", indicates the network type.
In the system to be configured in this course, select the Ethernet "**RJ71EN71(E+E)**" for both of the port 1 and port 2.

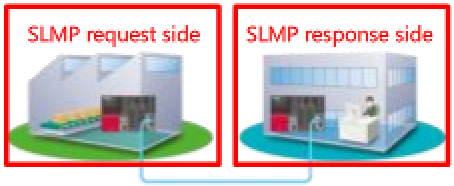


RD81DL96	High speed data logger module(1000BASE-T/100BASE-TX/10B
RD81MES96	MES interface module (1000BASE-T/100BASE-TX/10BASE-T: 2
RD81OPC96	OPC UA Server module (1000BASE-T/100BASE-TX/10BASE-T:
RJ71C24	Serial communication (RS232: 1 channel RS-422/485: 1 chanr
RJ71C24-R2	Serial communication (RS232: 2 channel)
RJ71C24-R4	Serial communication (RS422/485: 2 channel)
RJ71EN71(CCIEC)	Ethernet (1000BASE-T/100BASE-TX/10BASE-T: 2 channels)
RJ71EN71(CCIEF)	Ethernet (1000BASE-T/100BASE-TX/10BASE-T: 2 channels)
RJ71EN71(E+CCIEC)	Ethernet (1000BASE-T/100BASE-TX/10BASE-T: 2 channels)
RJ71EN71(E+CCIEF)	Ethernet (1000BASE-T/100BASE-TX/10BASE-T: 2 channels)
RJ71EN71(E+E)	Ethernet (1000BASE-T/100BASE-TX/10BASE-T: 2 channels)
RJ71EN71(Q)	Ethernet (1000BASE-T/100BASE-TX/10BASE-T: 2 channels)

3.4.2 Basic settings of network modules

This section describes the basic settings of network modules (Ethernet modules) such as IP address and communication data code.

Open [Basic Settings] from the module parameter setting window.



Specify the IP address of the own station.

When "Enable" is selected, the network number and the station number are configured according to the third and fourth octets of the IP address.

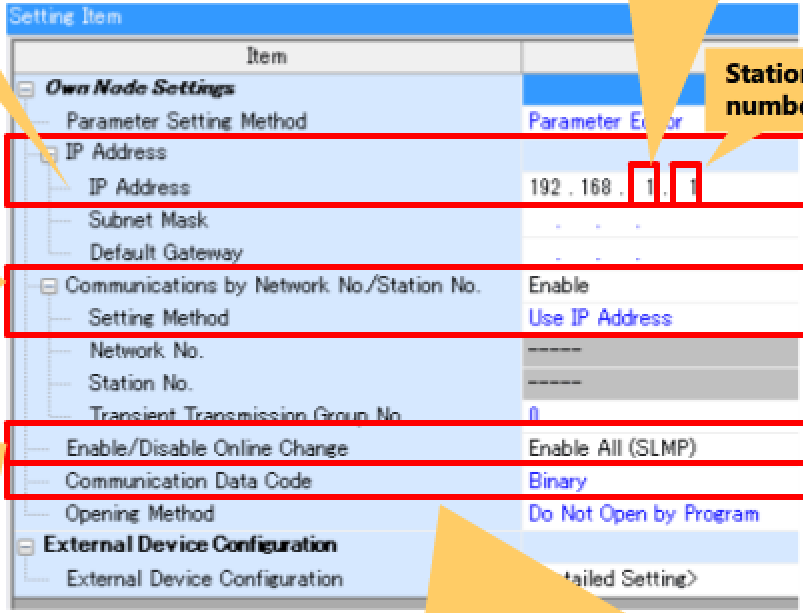
Enable or disable the external device to write data to the CPU module that is in the RUN state in SLMP communication.

Network number
If other networks such as CC-Link IE Control Network and CC-Link IE Field Network exist in the system, be sure to set a number different from their network numbers.

Station number

SLMP request side

- Active operation (Active open)
- Station number: 1
- IP address: 192.168.1.1

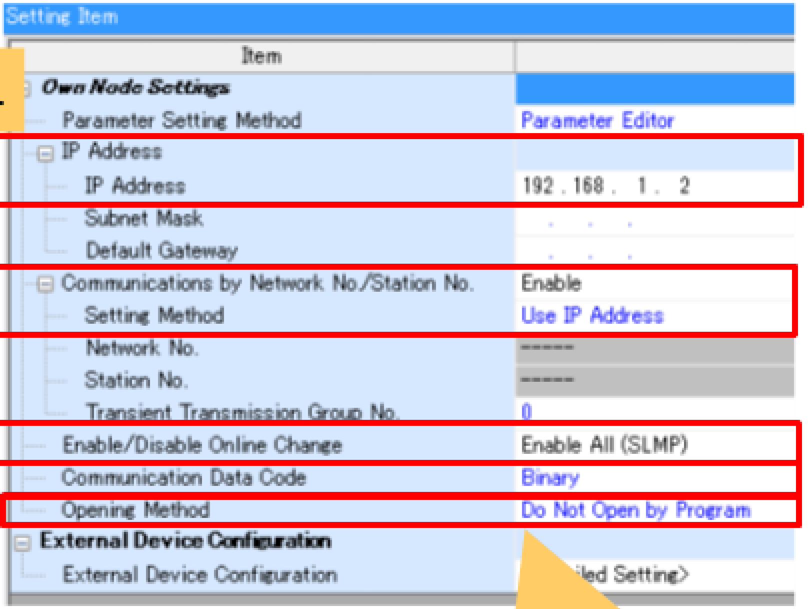


Setting Item

Item	
Own Node Settings	
Parameter Setting Method	Parameter Editor
IP Address	
IP Address	192 . 168 . 1 . 1
Subnet Mask	. . .
Default Gateway	. . .
Communications by Network No./Station No.	Enable
Setting Method	Use IP Address
Network No.	------
Station No.	------
Transient Transmission Group No.	0
Enable/Disable Online Change	Enable All (SLMP)
Communication Data Code	Binary
Opening Method	Do Not Open by Program
External Device Configuration	
External Device Configuration	ailed Setting>

SLMP response side

- Passive operation (Passive: Unpassive open)
- Station number: 2
- IP address: 192.168.1.2



Setting Item

Item	
Own Node Settings	
Parameter Setting Method	Parameter Editor
IP Address	
IP Address	192 . 168 . 1 . 2
Subnet Mask	. . .
Default Gateway	. . .
Communications by Network No./Station No.	Enable
Setting Method	Use IP Address
Network No.	------
Station No.	------
Transient Transmission Group No.	0
Enable/Disable Online Change	Enable All (SLMP)
Communication Data Code	Binary
Opening Method	Do Not Open by Program
External Device Configuration	
External Device Configuration	iled Setting>

Select the communication data code in accordance with the specifications of the external device.
• Binary: 1-byte data is sent/received as it is
• ASCII: 1-byte data is sent/received as two ASCII code characters

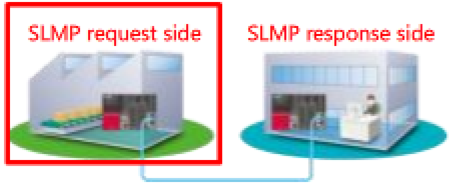
The data amount to be transmitted/received by the binary is half of that by the ASCII. Selecting the binary reduces the load applied to the communication line.

Select the open method for connections when communication is performed over the TCP protocol in Passive open mode or over the UDP protocol.

If "Do Not Open by Program" is selected, connections are opened when the system receives an Active request.

3.4.3 External device connection settings - SLMP request side

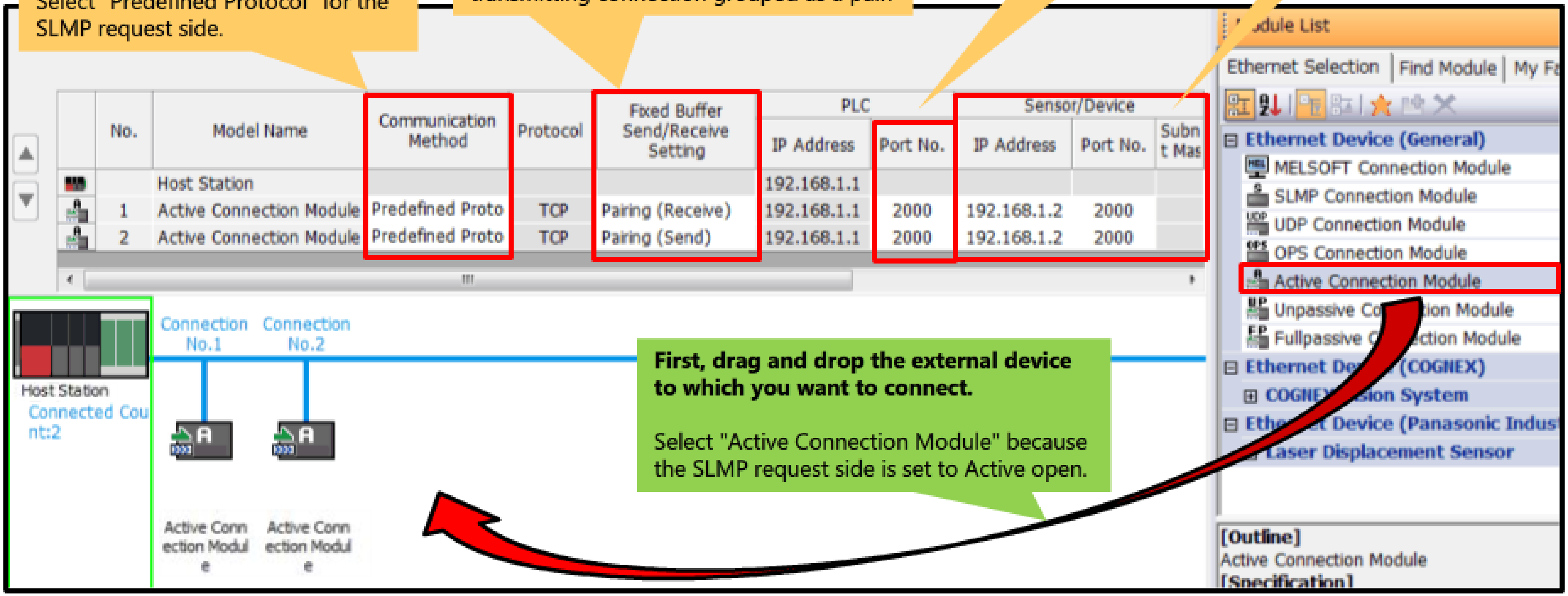
This section describes the external device connection settings configured on the SLMP request side.



From the module parameter setting window, open [Basic Settings] - [External Device Configuration].
First, select the external device to which you want to establish a connection from the module list, and place it in the diagram.

SLMP request side

- Active operation (Active open)
- IP address: 192.168.1.1



Select the communication method used with the external device.

Select "Predefined Protocol" for the SLMP request side.

Select the communication method for communications using a fixed buffer.

Select the pairing option to establish connections using one port each for the own station and the external device with the receiving connection and the transmitting connection grouped as a pair.

Set the port number for each connection link.

Set all ports for the system in this course to "2000".

Enter the IP address and the port number of the external device (SLMP response side).

	No.	Model Name	Communication Method	Protocol	Fixed Buffer Send/Receive Setting	PLC		Sensor/Device		
						IP Address	Port No.	IP Address	Port No.	Subn t Mas
		Host Station				192.168.1.1				
	1	Active Connection Module	Predefined Proto	TCP	Pairing (Receive)	192.168.1.1	2000	192.168.1.2	2000	
	2	Active Connection Module	Predefined Proto	TCP	Pairing (Send)	192.168.1.1	2000	192.168.1.2	2000	

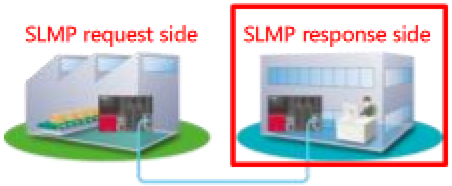
[Outline]
Active Connection Module
[Specification]

First, drag and drop the external device to which you want to connect.

Select "Active Connection Module" because the SLMP request side is set to Active open.

3.4.3 External device connection settings - SLMP response side

This section describes the settings for the SLMP response side.

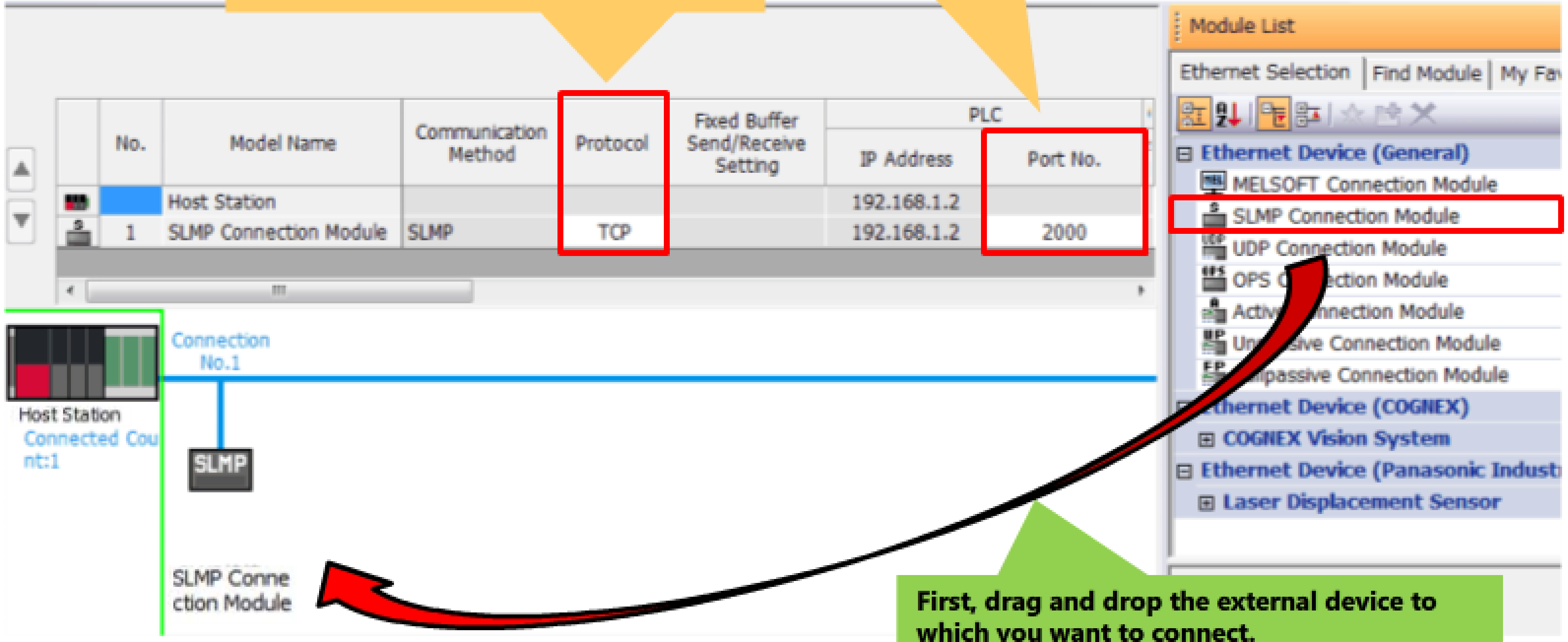


SLMP response side

- Passive operation (Passive: Unpassive open)
- IP address: 192.168.1.2

Used to specify the communications protocol used with the external device. Select "TCP".

Set "2000" to match the setting on the SLMP request side.



	No.	Model Name	Communication Method	Protocol	Fixed Buffer Send/Receive Setting	PLC IP Address	PLC Port No.
		Host Station				192.168.1.2	
	1	SLMP Connection Module	SLMP	TCP		192.168.1.2	2000

First, drag and drop the external device to which you want to connect.

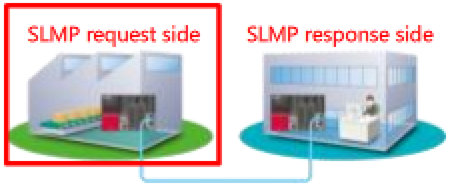
For the SLMP response side, select "SLMP Connection Module".

Now that the module parameter settings have been configured, next perform the parameter error check, apply the parameters, convert all, and write the settings to the CPU module.

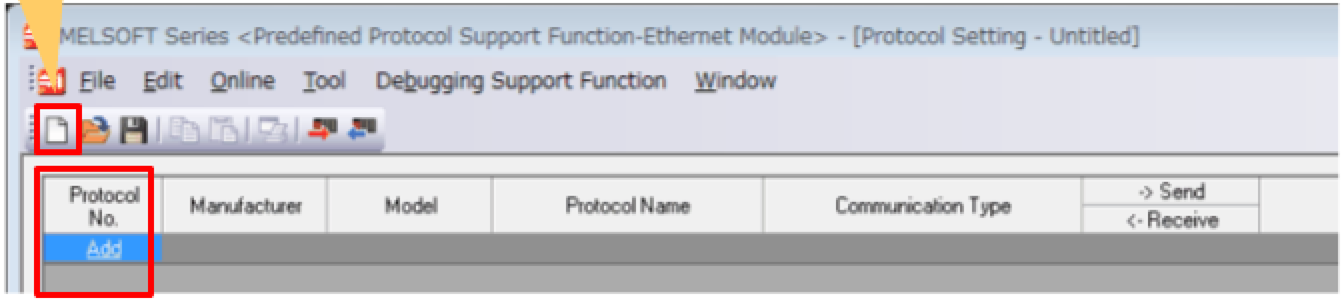
3.5 Predefined protocol support function

The predefined protocol support function helps create the send/receive messages necessary for communication with external devices.
This section describes how to register a predefined protocol using the predefined protocol support function.
Register the predefined protocol on the SLMP request side.

On the GX Works3 menu, select [Tool] - [Predefined Protocol Support Function] -[Ethernet Module] to open the predefined protocol support function.



Click [New].


MELSOFT Series <Predefined Protocol Support Function-Ethernet Module> - [Protocol Setting - Untitled]
File
Edit
Online
Tool
Debugging Support Function
Window

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive
Add					

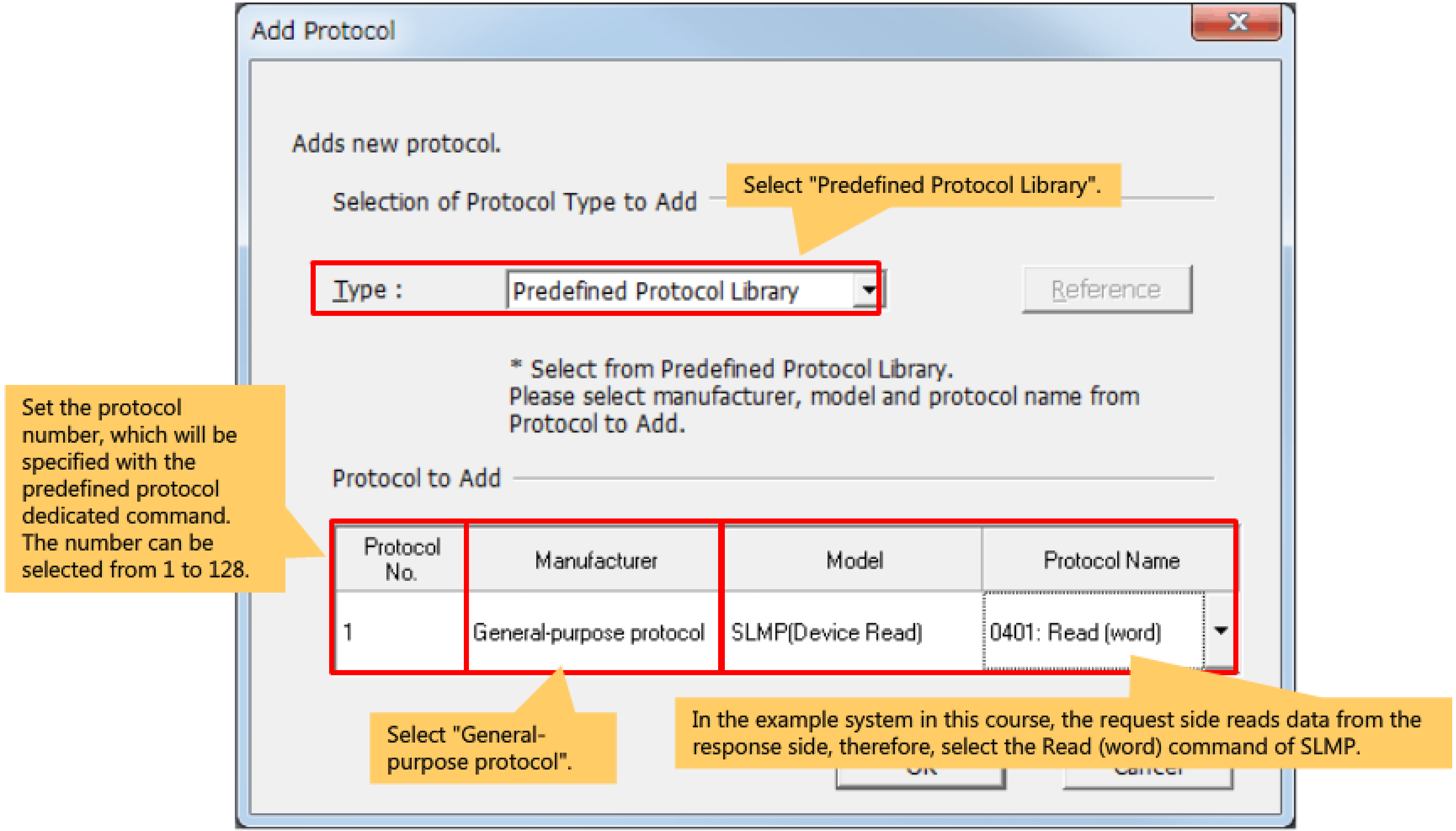
Protocol setting window

Click [Add] to open the "Add Protocol" window.

Details are explained in Section 3.5.1 on the next page.

3.5.1 Adding a protocol

The "Add Protocol" window is shown below.



Add Protocol

Adds new protocol.

Selection of Protocol Type to Add

Type : Predefined Protocol Library

* Select from Predefined Protocol Library.
Please select manufacturer, model and protocol name from Protocol to Add.

Protocol to Add

Protocol No.	Manufacturer	Model	Protocol Name
1	General-purpose protocol	SLMP(Device Read)	0401: Read (word)

Select "Predefined Protocol Library".

Set the protocol number, which will be specified with the predefined protocol dedicated command. The number can be selected from 1 to 128.

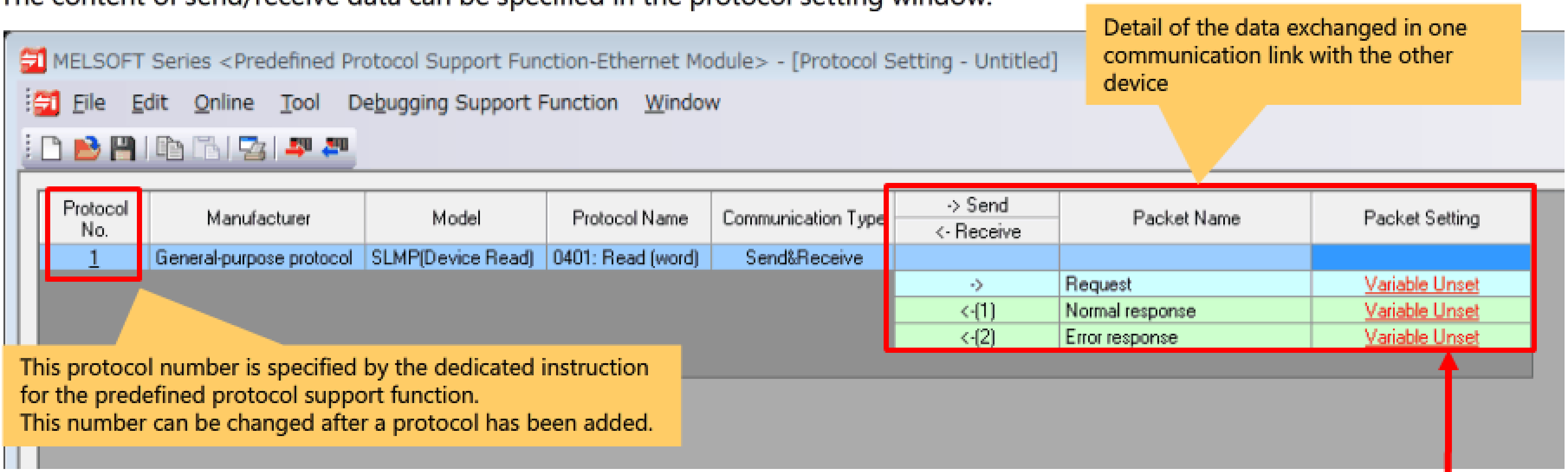
Select "General-purpose protocol".

In the example system in this course, the request side reads data from the response side, therefore, select the Read (word) command of SLMP.

Add Protocol window

3.5.2 Protocol settings

The content of send/receive data can be specified in the protocol setting window.


MELSOFT Series <Predefined Protocol Support Function-Ethernet Module> - [Protocol Setting - Untitled]
File Edit Online Tool Debugging Support Function Window

Protocol No.	Manufacturer	Model	Protocol Name	Communication Type	-> Send <- Receive	Packet Name	Packet Setting
1	General-purpose protocol	SLMP(Device Read)	0401: Read (word)	Send&Receive			
					->	Request	Variable Unset
					<-(1)	Normal response	Variable Unset
					<-(2)	Error response	Variable Unset

Detail of the data exchanged in one communication link with the other device

This protocol number is specified by the dedicated instruction for the predefined protocol support function.
This number can be changed after a protocol has been added.

Protocol setting window

In the example system in this course, the Device Read (word) protocol of the SLMP command is used.
This protocol consists of the following three packets:

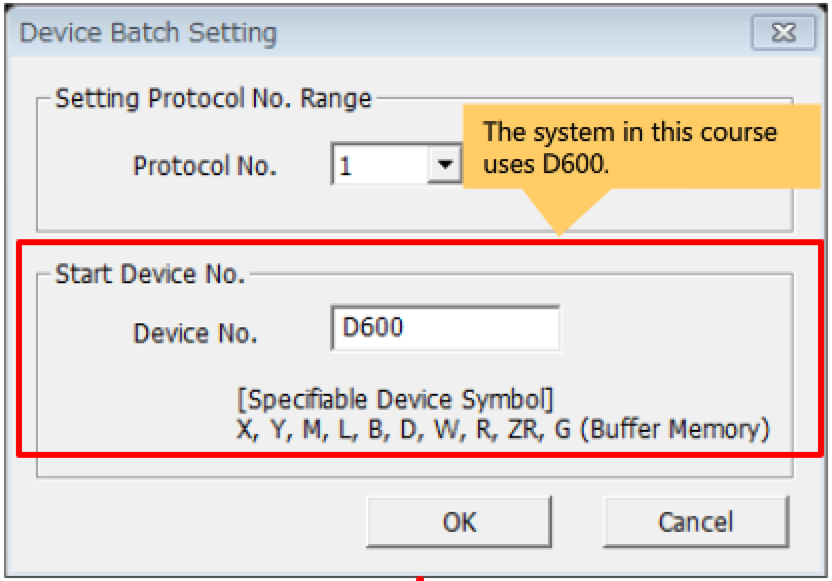
- Request
- Normal response
- Error response

If the packet has not been set, "Variable Unset" is displayed in red.
Details regarding the packet setting procedure are given on the following page.

3.5.3 Packet settings

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

Select [Edit] - [Device Batch Setting] of the predefined protocol support function, and enter the start device number.



Setting Protocol No. Range

Protocol No. 1

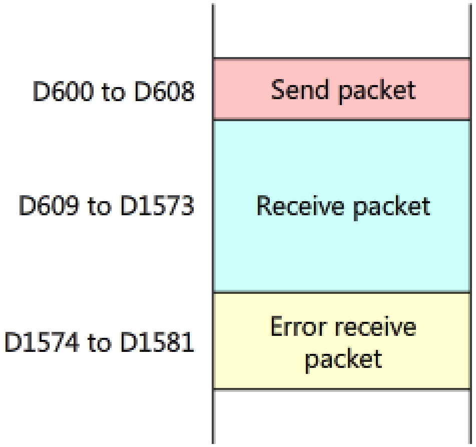
The system in this course uses D600.

Start Device No.

Device No. D600

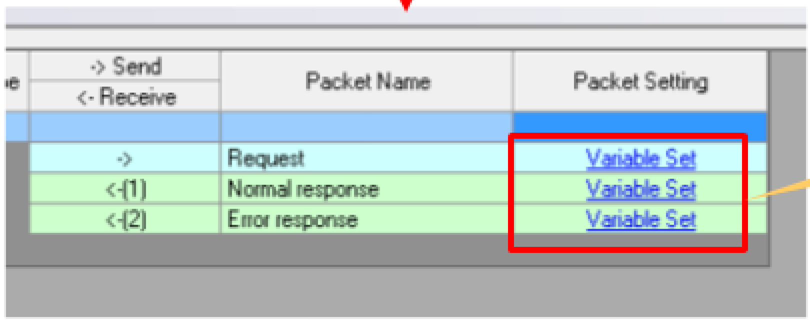
[Specifiable Device Symbol]
X, Y, M, L, B, D, W, R, ZR, G (Buffer Memory)

Device batch setting window



D600 to D608	Send packet
D609 to D1573	Receive packet
D1574 to D1581	Error receive packet

Device assignment



e	-> Send <- Receive	Packet Name	Packet Setting
	->	Request	Variable Set
	<-(1)	Normal response	Variable Set
	<-(2)	Error response	Variable Set

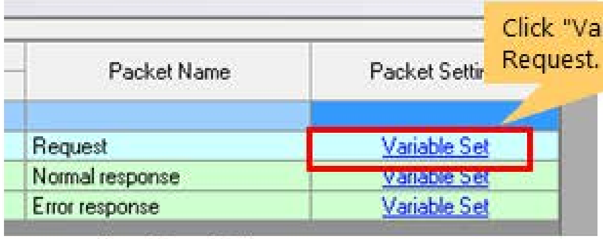
The status of the three packets changes from "Variable Unset" to "Variable Set".

Protocol setting window

3.5.3 Packet settings

This section describes how the devices are automatically set as the result of the device batch setting.

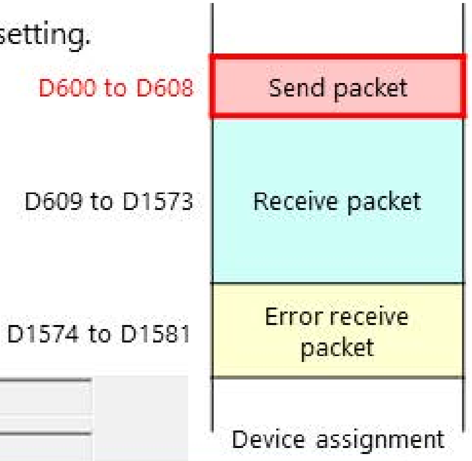
(1) Send packet



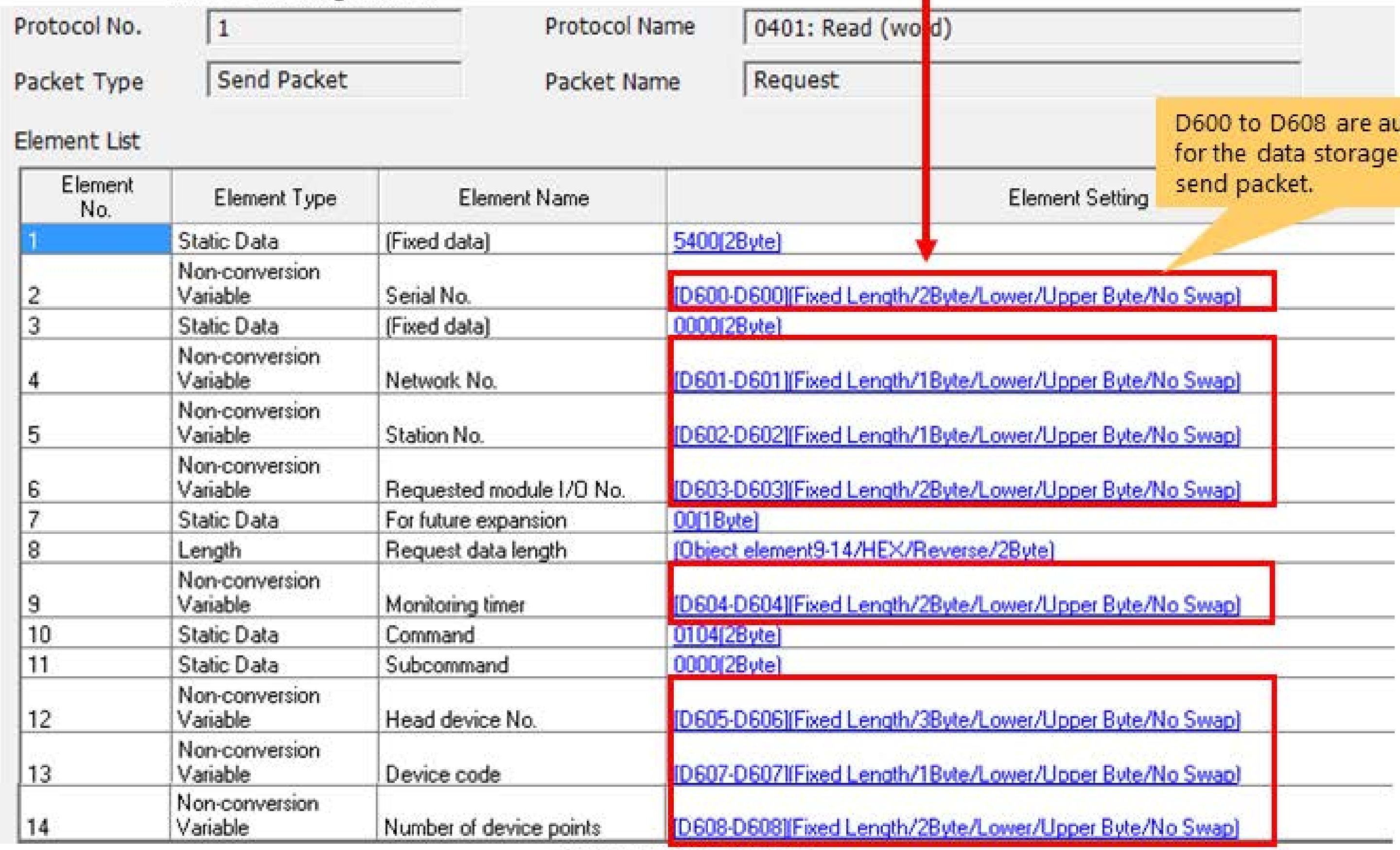
Packet Name	Packet Settir
Request	Variable Set
Normal response	Variable Set
Error response	Variable Set

Click "Variable Set" of Request.

Protocol setting window



Device assignment



Protocol No. 1

Protocol Name 0401: Read (word)

Packet Type Send Packet

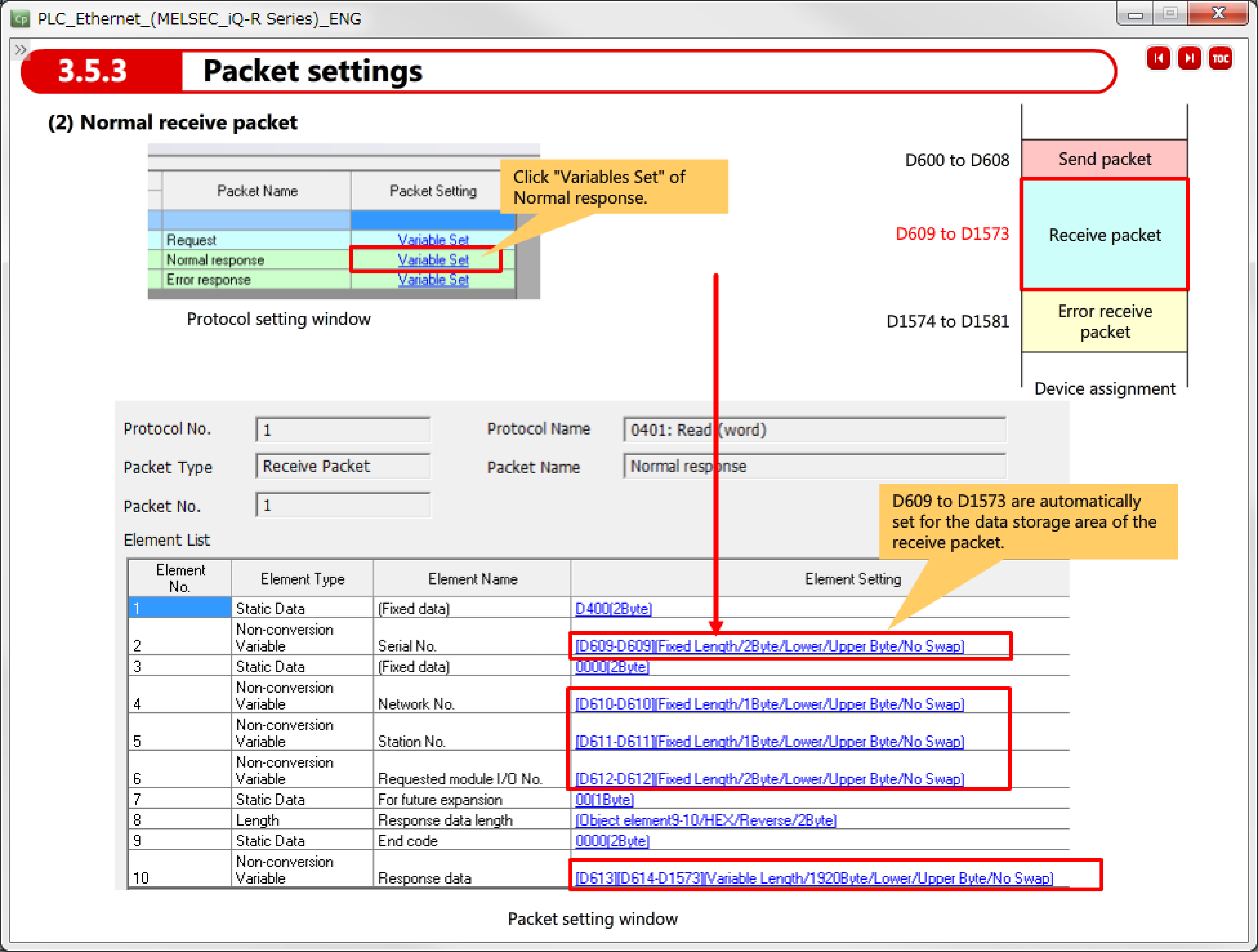
Packet Name Request

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)

D600 to D608 are automatically set for the data storage area of the send packet.

Packet setting window



3.5.3 Packet settings

(2) Normal receive packet

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

Protocol setting window

Click "Variables Set" of Normal 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 Normal response

Packet No. 1

Element List

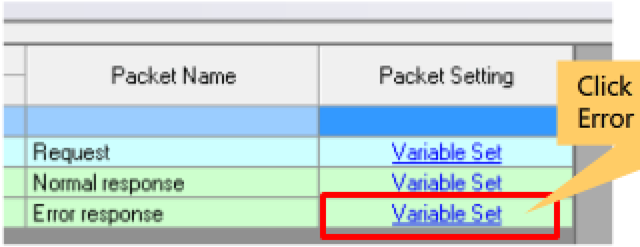
Element No.	Element Type	Element Name	Element Setting
1	Static Data	(Fixed data)	D400(2Byte)
2	Non-conversion Variable	Serial No.	[D609-D609][Fixed Length/2Byte/Lower/Upper Byte/No Swap]
3	Static Data	(Fixed data)	0000(2Byte)
4	Non-conversion Variable	Network No.	[D610-D610][Fixed Length/1Byte/Lower/Upper 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]

D609 to D1573 are automatically set for the data storage area of the receive packet.

Packet setting window

3.5.3 Packet settings

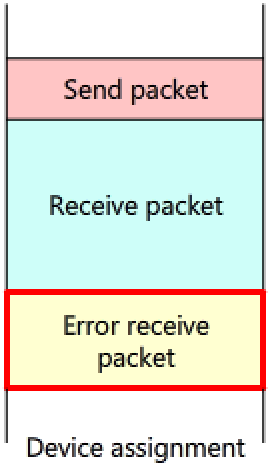
(3) Error receive packet



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

Click "Variables Set" of Error response.

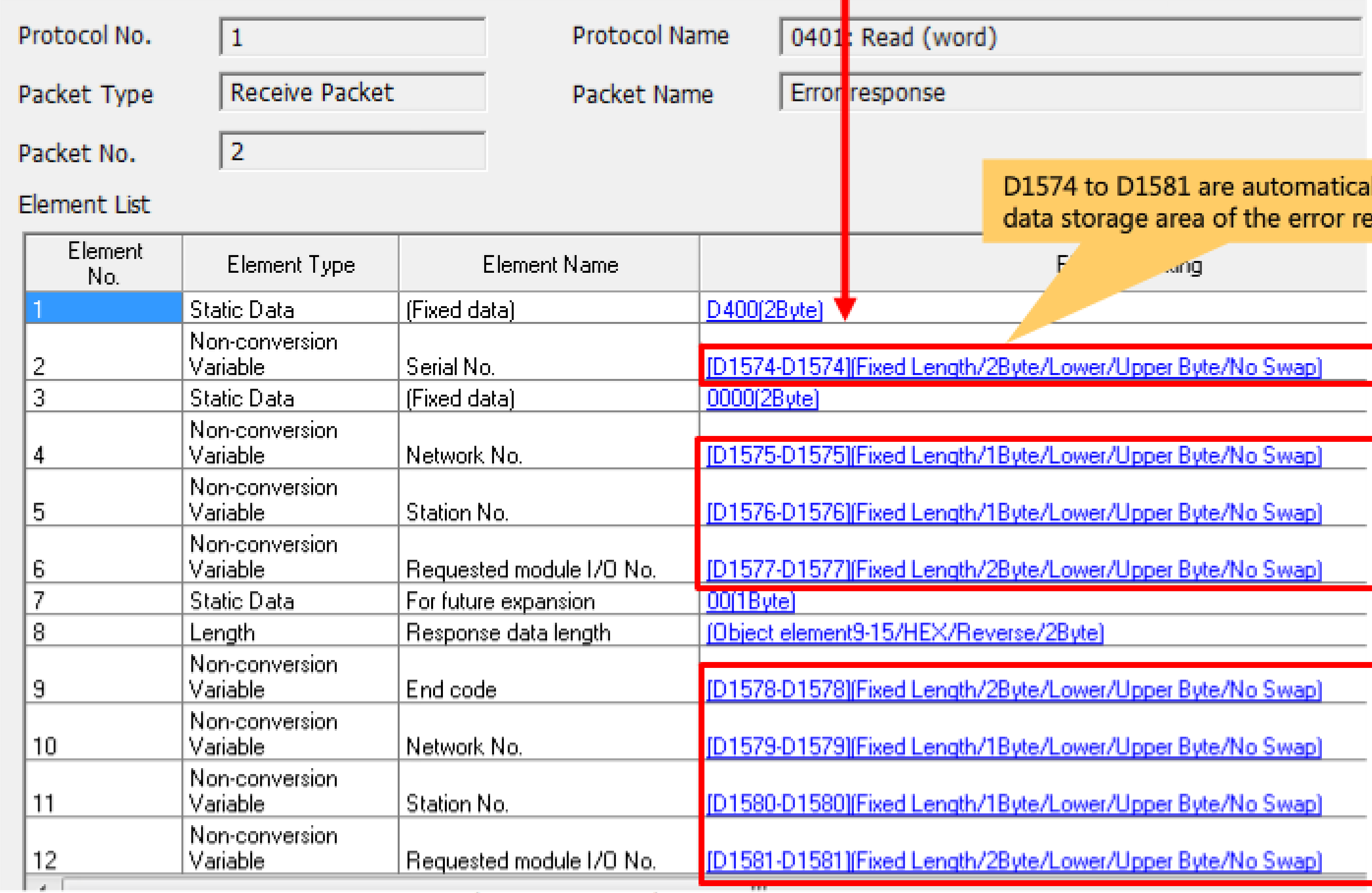
Protocol setting window



D600 to D608

D609 to D1573

D1574 to D1581



Protocol No. 1

Protocol Name 0401: Read (word)

Packet Type Receive Packet

Packet Name Error response

Packet No. 2

Element List

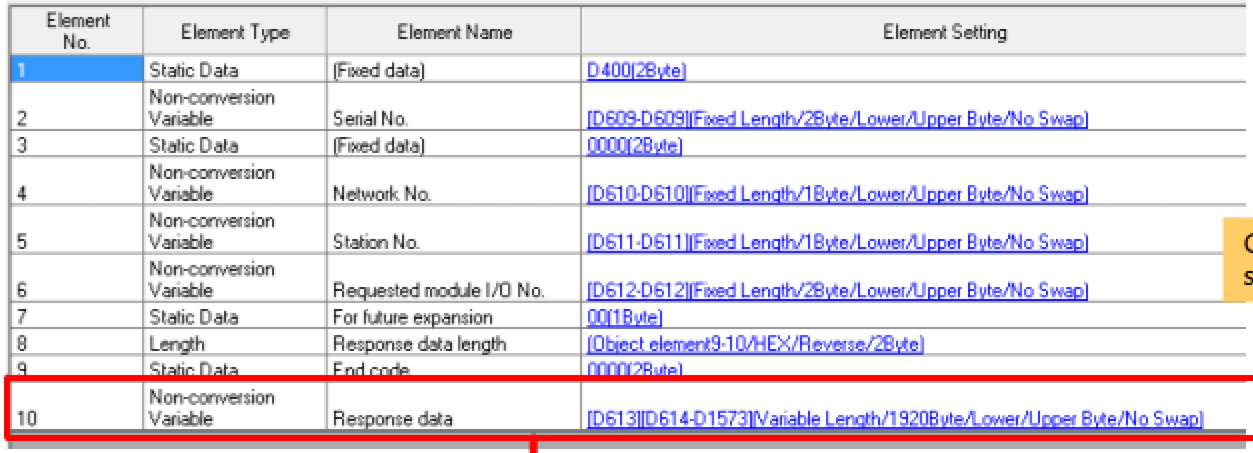
Element No.	Element Type	Element Name	F[illegible]ing
1	Static Data	(Fixed data)	D400(2Byte)
2	Non-conversion Variable	Serial No.	[D1574-D1574](Fixed Length/2Byte/Lower/Upper Byte/No Swap)
3	Static Data	(Fixed data)	0000(2Byte)
4	Non-conversion Variable	Network No.	[D1575-D1575](Fixed Length/1Byte/Lower/Upper 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)

D1574 to D1581 are automatically set for the data storage area of the error receive packet.

Packet setting window

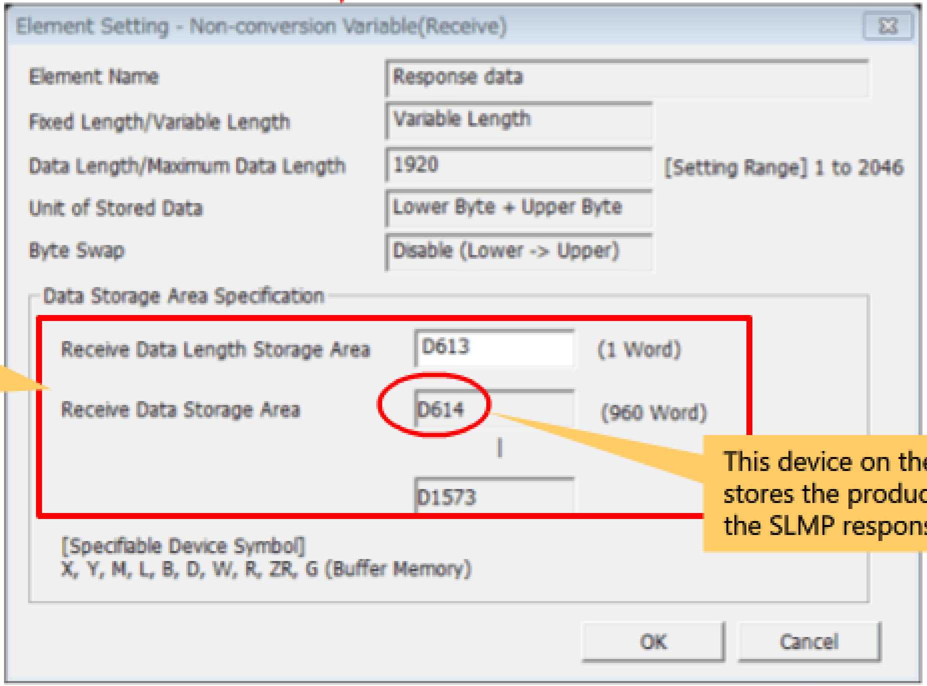
3.5.4 Element settings

The setting details for each element can be checked and changed.
The following figure shows setting details of the normal receive packet.



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]

Click on the element shown in blue.



Element Setting - Non-conversion Variable(Receive)

Element Name	Response data
Fixed Length/Variable Length	Variable Length
Data Length/Maximum Data Length	1920 [Setting Range] 1 to 2046
Unit of Stored Data	Lower Byte + Upper Byte
Byte Swap	Disable (Lower -> Upper)

Data Storage Area Specification

Receive Data Length Storage Area	D613 (1 Word)
Receive Data Storage Area	D614 (960 Word) | D1573

[Specifiable Device Symbol]
X, Y, M, L, B, D, W, R, ZR, G (Buffer Memory)

D613 to D1573 are automatically entered for the data storage area.

This device on the SLMP request side reads and stores the production instruction (D1000) from the SLMP response side.

Element setting window

3.6 Saving a created protocol, and writing it to a programmable controller

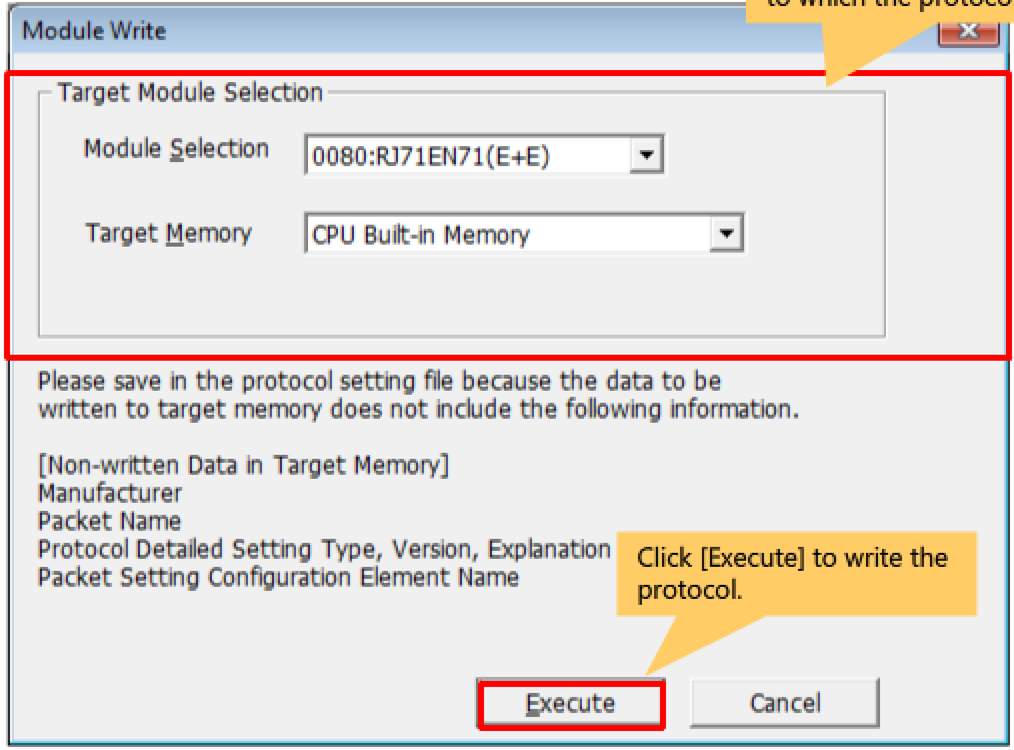
Saving a protocol

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

Writing a protocol to a programmable controller

The procedure for writing a created protocol to the Ethernet module is given below.
From the menu of the predefined protocol support function, select [Online] - [Write to Module], and then reset the CPU module.

Select the module and the memory to which the protocol is written.



Module Write

Target Module Selection

Module Selection: 0080:RJ71EN71(E+E)

Target Memory: CPU Built-in Memory

Please save in the protocol setting file because the data to be written to target memory does not include the following information.

[Non-written Data in Target Memory]
Manufacturer
Packet Name
Protocol Detailed Setting Type, Version, Explanation
Packet Setting Configuration Element Name

Click [Execute] to write the protocol.

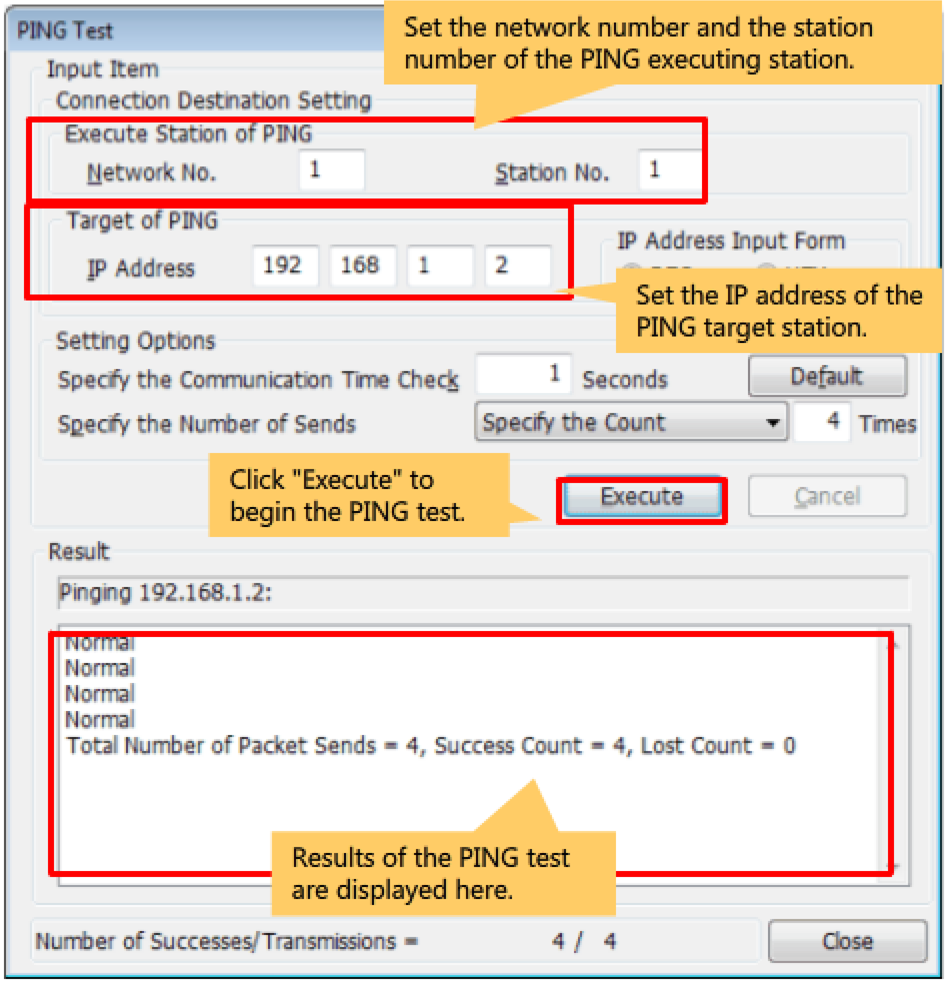
Module Write window

3.7 Communication check

The "PING test" can be performed to verify normal communication of an Ethernet module.

Procedure of PING test

(1) From the GX Works3 menu, select [Diagnostics] - [Ethernet Diagnostics] to open the Ethernet diagnostics window.
(2) While selecting "Board No.1 (Port 1)" of the target module, select the "Module No." checkbox.
(3) Click the "PING Test" button to open the PING test window.



PING Test

Input Item

Connection Destination Setting

Execute Station of PING

Network No. 1 Station No. 1

Set the network number and the station number of the PING executing station.

Target of PING

IP Address 192 168 1 2

IP Address Input Form

Set the IP address of the PING target station.

Setting Options

Specify the Communication Time Check 1 Seconds

Specify the Number of Sends: Specify the Count 4 Times

Click "Execute" to begin the PING test.

Result

```
Pinging 192.168.1.2:
Normal
Normal
Normal
Normal
Total Number of Packet Sends = 4, Success Count = 4, Lost Count = 0
```

Results of the PING test are displayed here.

Number of Successes/Transmissions = 4 / 4

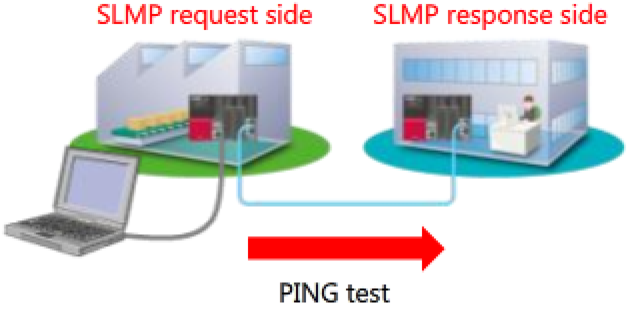
PING test window

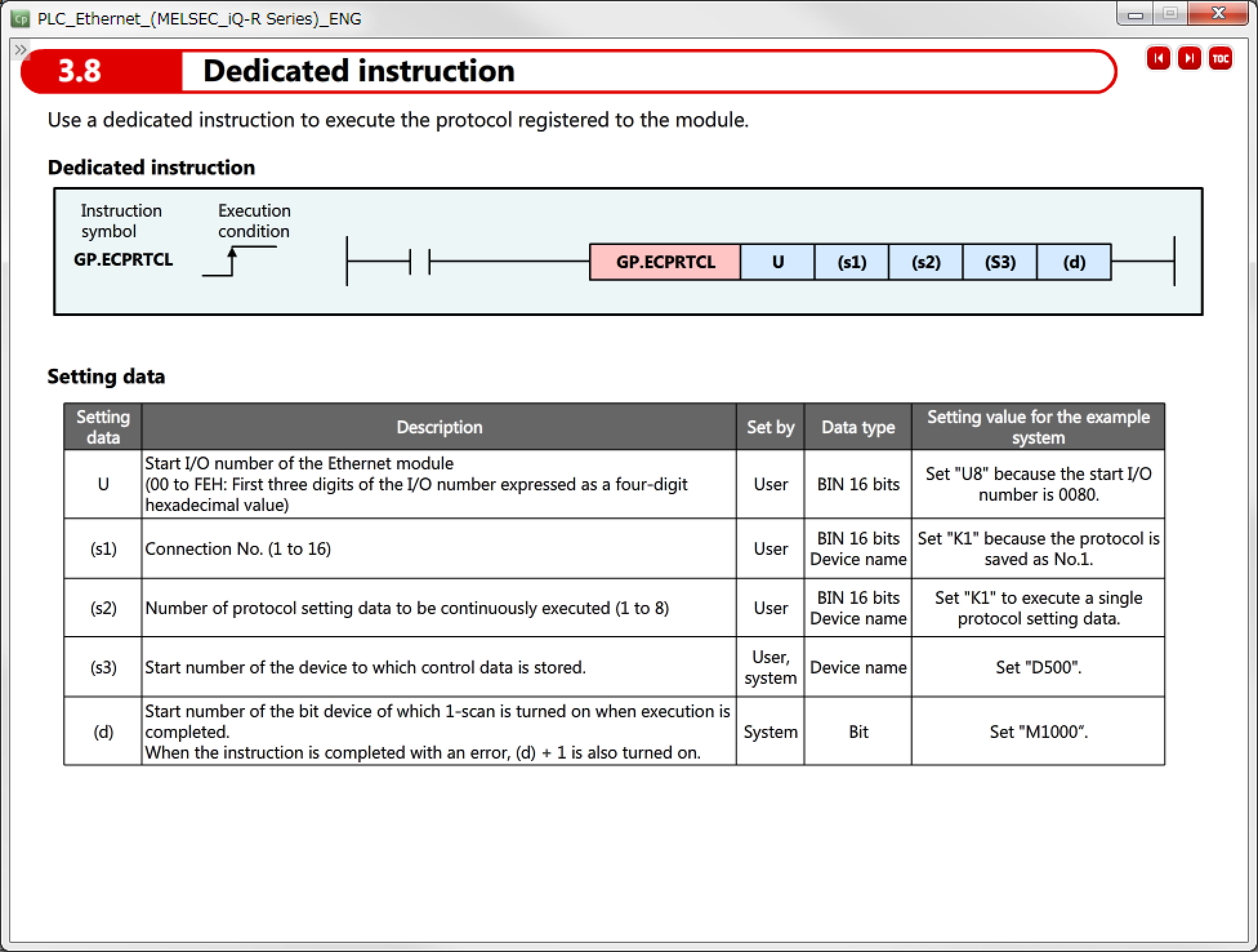
PING executing station

- Network No.1
- Station number: 1
- IP address: 192.168.1.1

PING target station

- Network No.1
- Station number: 2
- IP address: 192.168.1.2





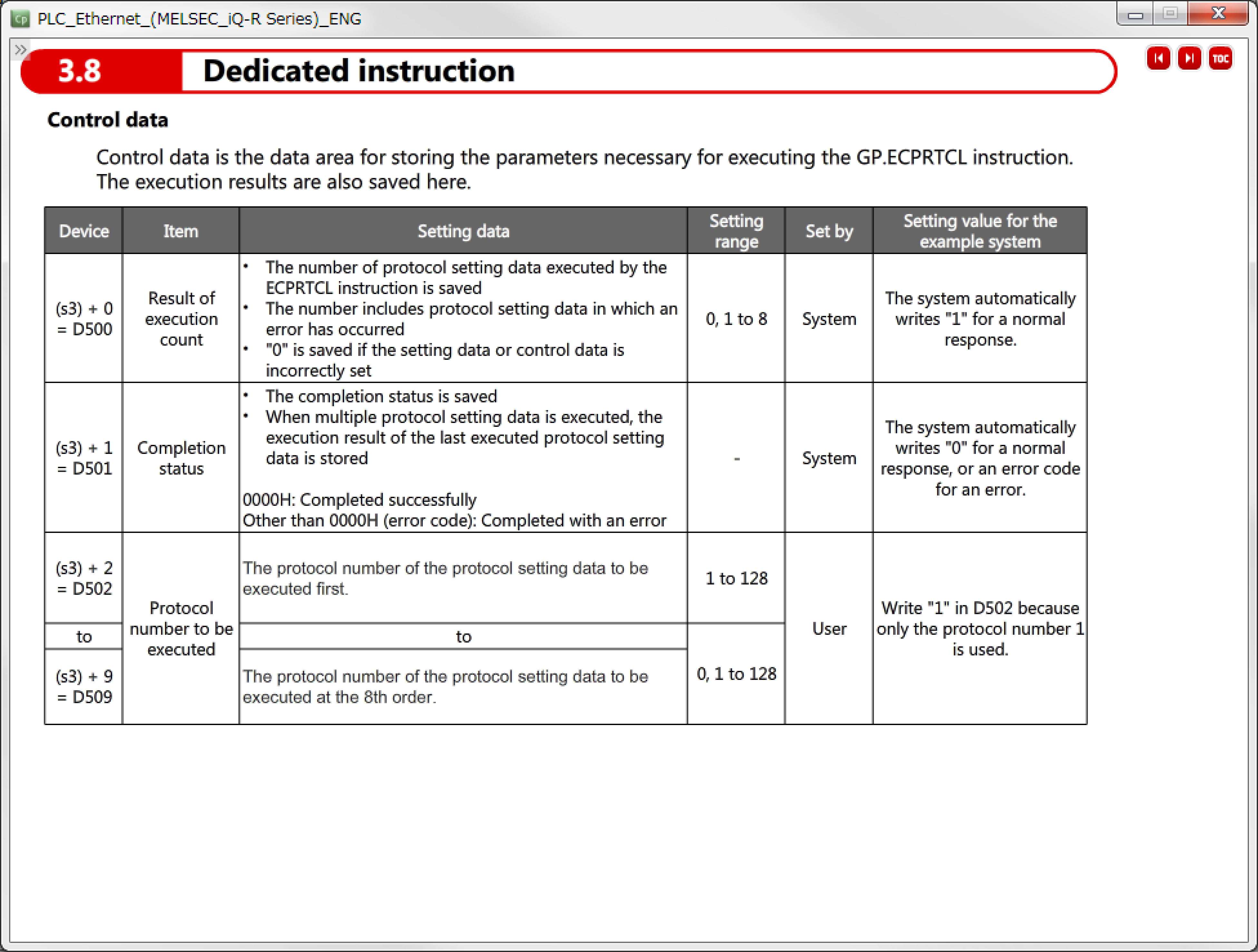
3.8 Dedicated instruction

Use a dedicated instruction to execute the protocol registered to the module.

Dedicated instruction

Setting data

Setting data	Description	Set by	Data type	Setting value for the example system
U	Start I/O number of the Ethernet module (00 to FEH: First three digits of the I/O number expressed as a four-digit hexadecimal value)	User	BIN 16 bits	Set "U8" because the start I/O number is 0080.
(s1)	Connection No. (1 to 16)	User	BIN 16 bits Device name	Set "K1" because the protocol is saved as No.1.
(s2)	Number of protocol setting data to be continuously executed (1 to 8)	User	BIN 16 bits Device name	Set "K1" to execute a single protocol setting data.
(s3)	Start number of the device to which control data is stored.	User, system	Device name	Set "D500".
(d)	Start number of the bit device of which 1-scan is turned on when execution is completed. When the instruction is completed with an error, (d) + 1 is also turned on.	System	Bit	Set "M1000".

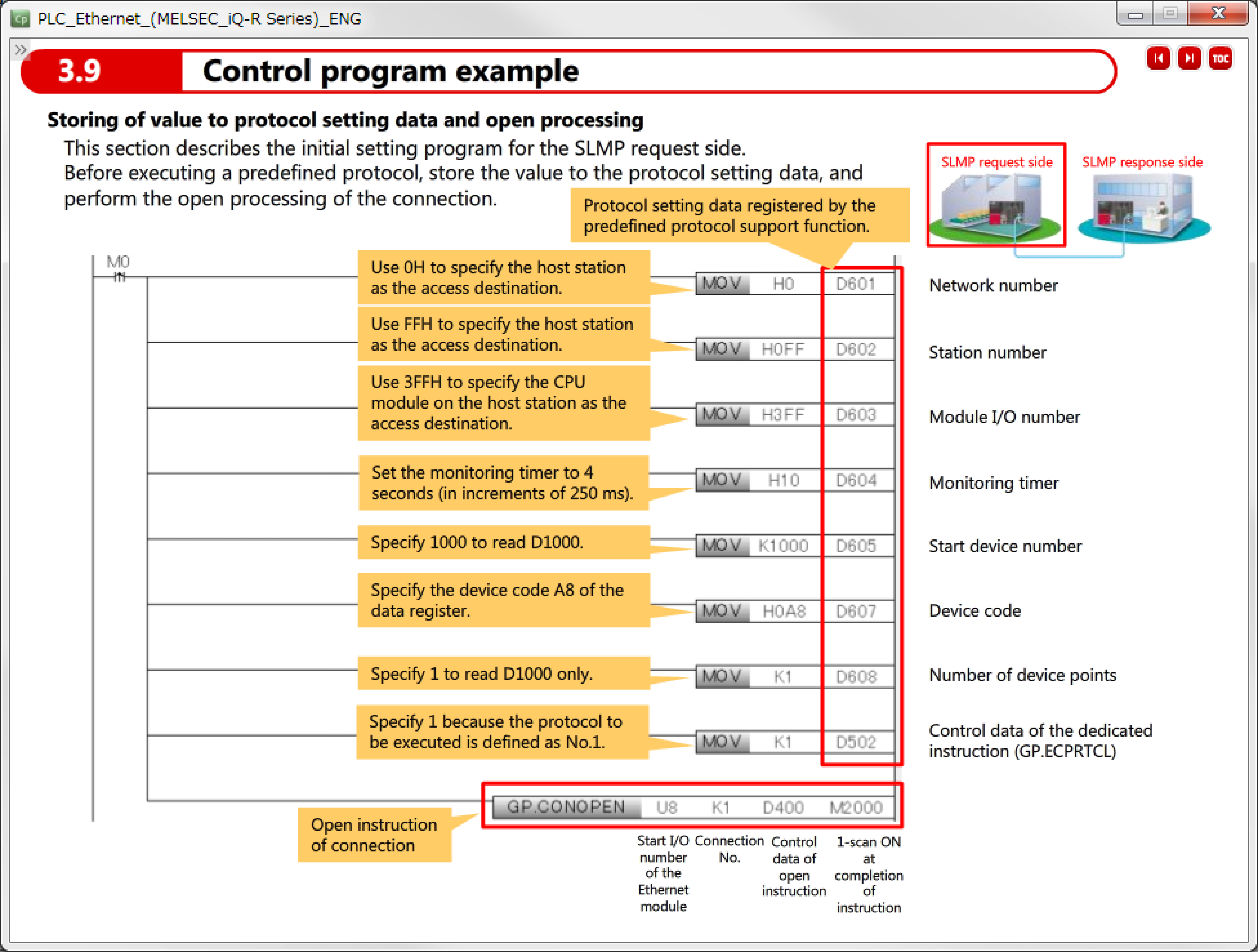


3.8 Dedicated instruction

Control data

Control data is the data area for storing the parameters necessary for executing the GP.ECPRTCL instruction. The execution results are also saved here.

Device	Item	Setting data	Setting range	Set by	Setting value for the example system
(s3) + 0 = D500	Result of execution count	• The number of protocol setting data executed by the ECPRTCL instruction is saved • The number includes protocol setting data 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.
(s3) + 1 = D501	Completion status	• The completion status is saved • When multiple protocol setting data is executed, the execution result of the last executed protocol setting data is stored 0000H: Completed successfully Other than 0000H (error code): Completed with an error	-	System	The system automatically writes "0" for a normal response, or an error code for an error.
(s3) + 2 = D502	Protocol number to be executed	The protocol number of the protocol setting data to be executed first.	1 to 128	User	Write "1" in D502 because only the protocol number 1 is used.
to		to	0, 1 to 128		
(s3) + 9 = D509		The protocol number of the protocol setting data to be executed at the 8th order.			



3.9 Control program example

Storing of value to protocol setting data and open processing

This section describes the initial setting program for the SLMP request side.
Before executing a predefined protocol, store the value to the protocol setting data, and perform the open processing of the connection.

Protocol setting data registered by the predefined protocol support function.

```
M0
-|↑|-+-[MOV  H0     D601 ]   Network number
     +-[MOV  H0FF   D602 ]   Station number
     +-[MOV  H3FF   D603 ]   Module I/O number
     +-[MOV  H10    D604 ]   Monitoring timer
     +-[MOV  K1000  D605 ]   Start device number
     +-[MOV  H0A8   D607 ]   Device code
     +-[MOV  K1     D608 ]   Number of device points
     +-[MOV  K1     D502 ]   Control data of the dedicated instruction (GP.ECPRTCL)
     +-[GP.CONOPEN  U8  K1  D400  M2000]
```

Use 0H to specify the host station as the access destination.

Use FFH to specify the host station as the access destination.

Use 3FFH to specify the CPU module on the host station as the access destination.

Set the monitoring timer to 4 seconds (in increments of 250 ms).

Specify 1000 to read D1000.

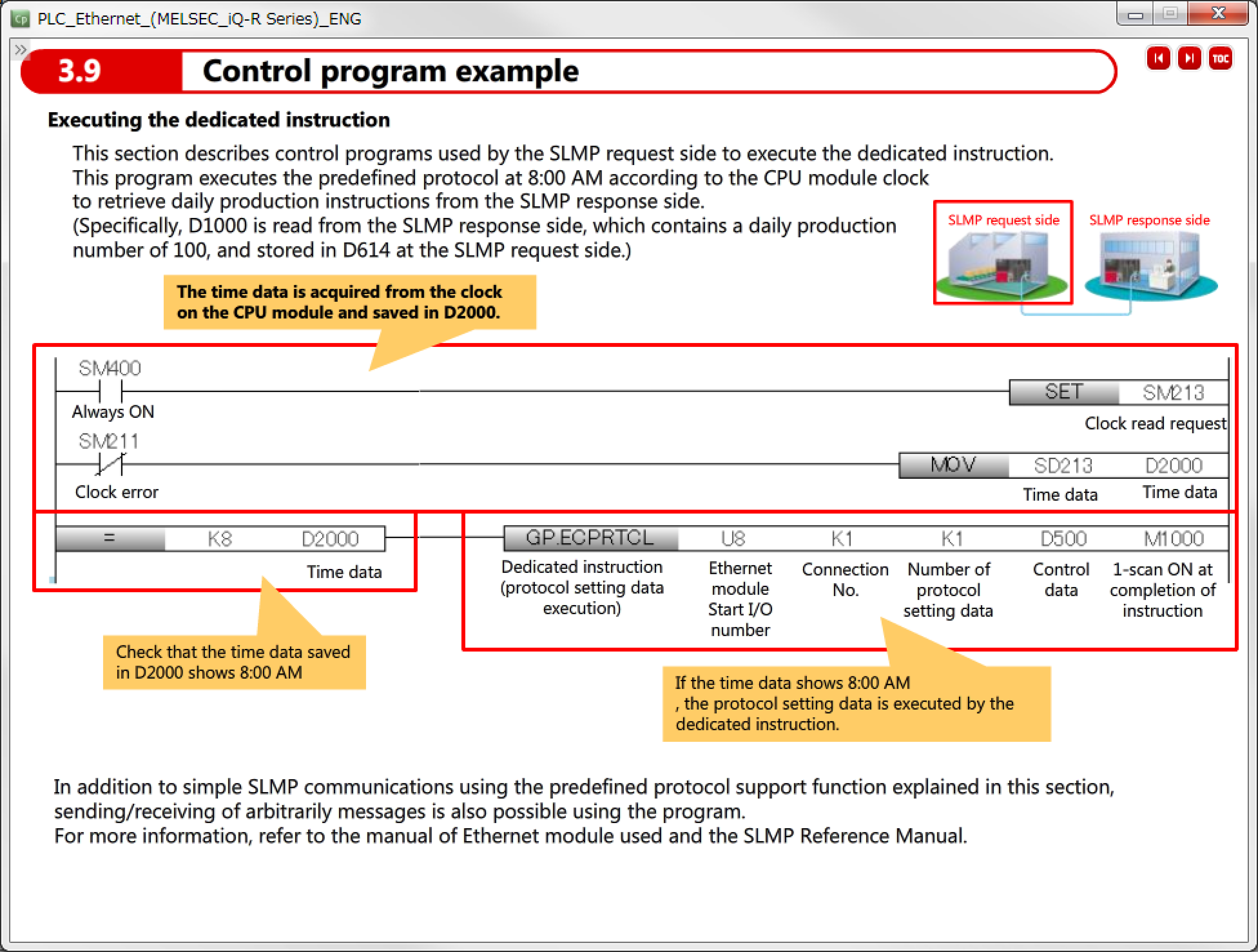
Specify the device code A8 of the data register.

Specify 1 to read D1000 only.

Specify 1 because the protocol to be executed is defined as No.1.

Open instruction of connection

GP.CONOPEN	U8	K1	D400	M2000
	Start I/O number of the Ethernet module	Connection No.	Control data of open instruction	1-scan ON at completion of instruction



3.9 Control program example

Executing the dedicated instruction

This section describes control programs used by the SLMP request side to execute the dedicated instruction.
This program executes the predefined protocol at 8:00 AM according to the CPU module clock to retrieve daily production instructions from the SLMP response side.
(Specifically, D1000 is read from the SLMP response side, which contains a daily production number of 100, and stored in D614 at the SLMP request side.)

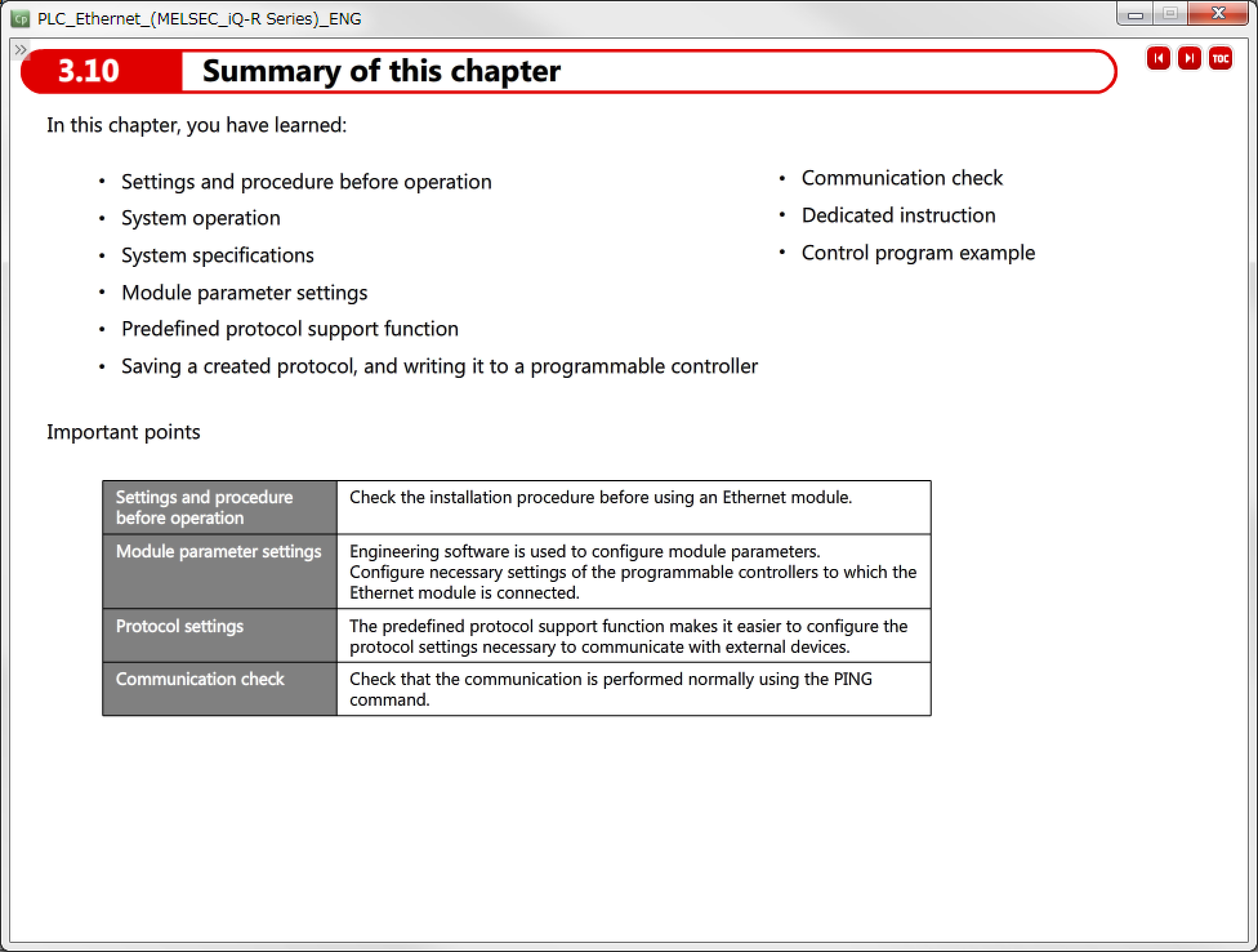
The time data is acquired from the clock on the CPU module and saved in D2000.

SM400
Always ON
SET SM213
Clock read request
SM211
Clock error
MOV SD213 D2000
Time data
Time data
= K8 D2000
Time data
GP.ECPRTCL U8 K1 K1 D500 M1000
Dedicated instruction (protocol setting data execution)
Ethernet module Start I/O number
Connection No.
Number of protocol setting data
Control data
1-scan ON at completion of instruction

Check that the time data saved in D2000 shows 8:00 AM

If the time data shows 8:00 AM
, the protocol setting data is executed by the dedicated instruction.

In addition to simple SLMP communications using the predefined protocol support function explained in this section, sending/receiving of arbitrarily messages is also possible using the program.
For more information, refer to the manual of Ethernet module used and the SLMP Reference Manual.



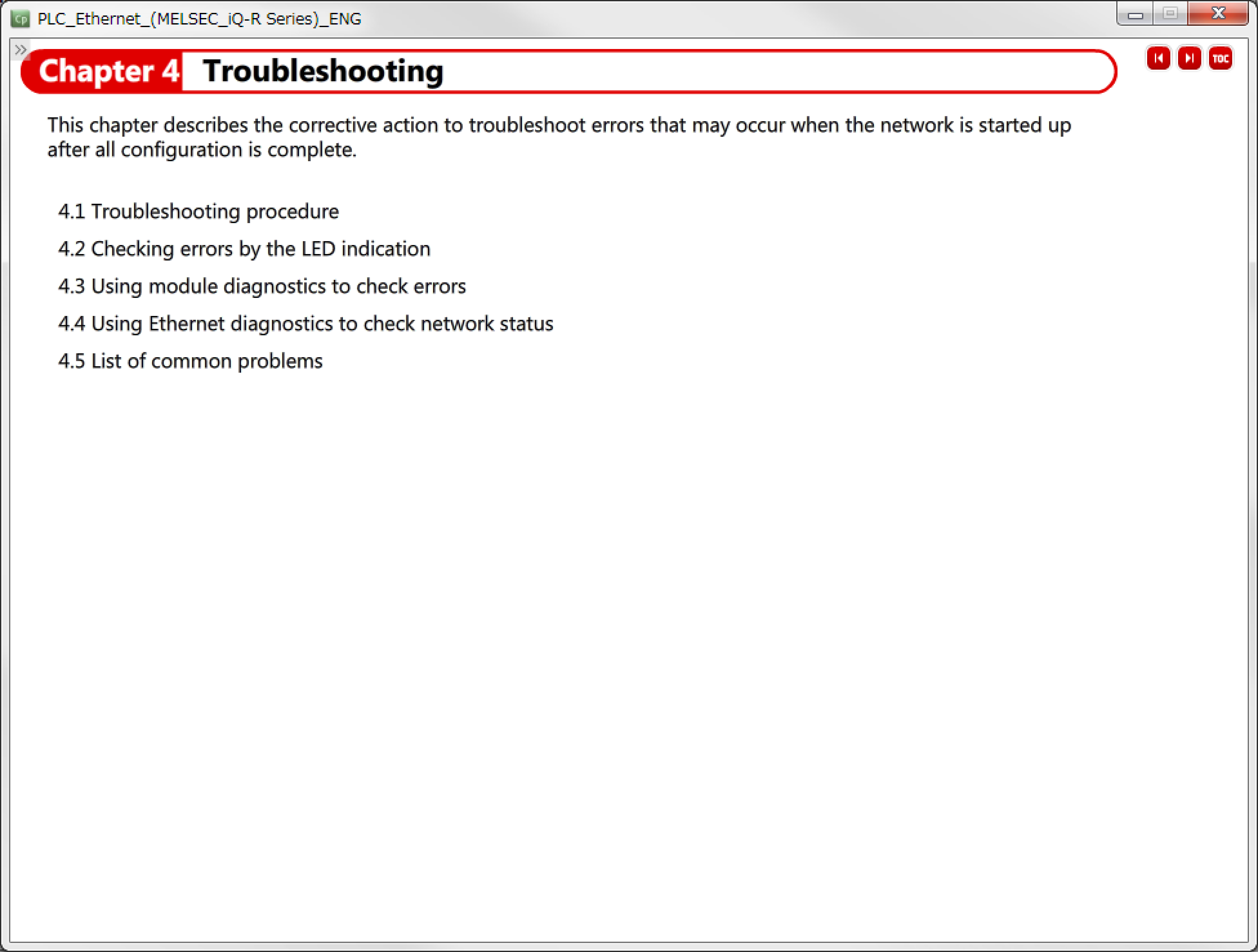
3.10 Summary of this chapter

In this chapter, you have learned:

- Settings and procedure before operation
- System operation
- System specifications
- Module parameter settings
- Predefined protocol support function
- Saving a created protocol, and writing it to a programmable controller
- Communication check
- Dedicated instruction
- Control program example

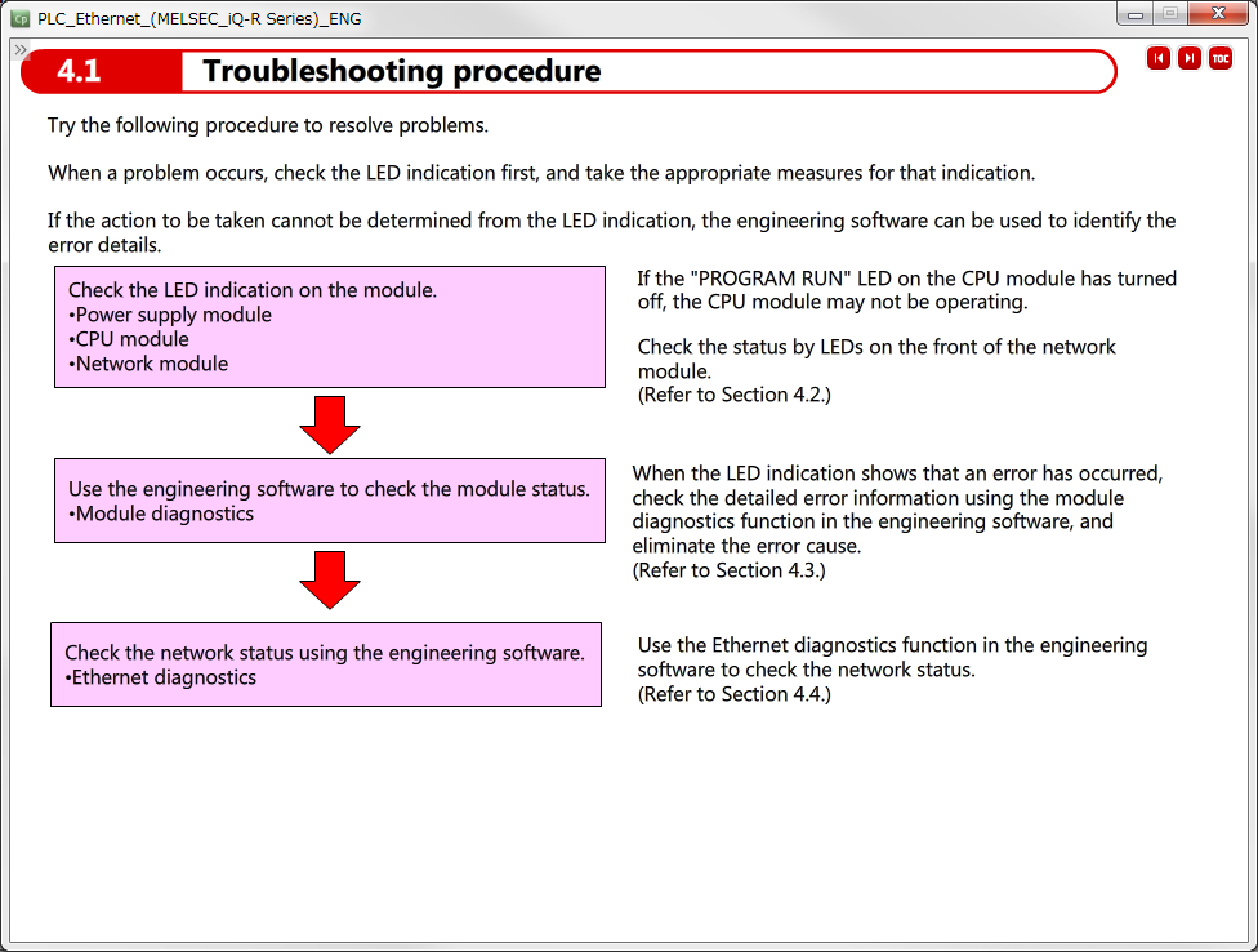
Important points

Settings and procedure before operation	Check the installation procedure before using an Ethernet module.
Module parameter settings	Engineering software is used to configure module parameters. Configure necessary settings of the programmable controllers to which the Ethernet module is connected.
Protocol settings	The predefined protocol support function makes it easier to configure the protocol settings necessary to communicate with external devices.
Communication check	Check that the communication is performed normally using the PING command.



Chapter 4 Troubleshooting

This chapter describes the corrective action to troubleshoot errors that may occur when the network is started up after all configuration is complete.



4.1 Troubleshooting procedure

Try the following procedure to resolve problems.

When a problem occurs, check the LED indication first, and take the appropriate measures for that indication.

If the action to be taken cannot be determined from the LED indication, the engineering software can be used to identify the error details.

Check the LED indication on the module.
- Power supply module
- CPU module
- Network module

If the "PROGRAM RUN" LED on the CPU module has turned off, the CPU module may not be operating.

Check the status by LEDs on the front of the network module.
(Refer to Section 4.2.)

Use the engineering software to check the module status.
- Module diagnostics

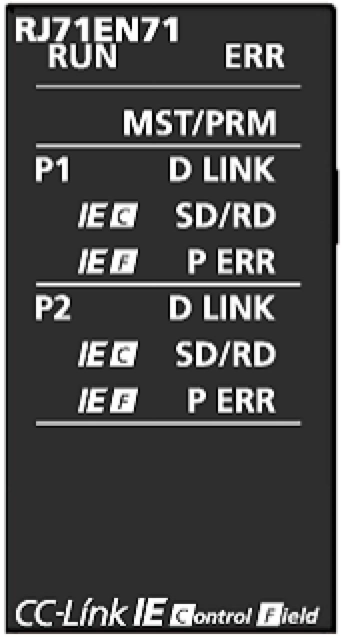
When the LED indication shows that an error has occurred, check the detailed error information using the module diagnostics function in the engineering software, and eliminate the error cause.
(Refer to Section 4.3.)

Check the network status using the engineering software.
- Ethernet diagnostics

Use the Ethernet diagnostics function in the engineering software to check the network status.
(Refer to Section 4.4.)

4.2 Checking for errors by the LED indication

If the network does not seem to be operating normally, check the network status using the LEDs on the front of modules without having to access the engineering software.

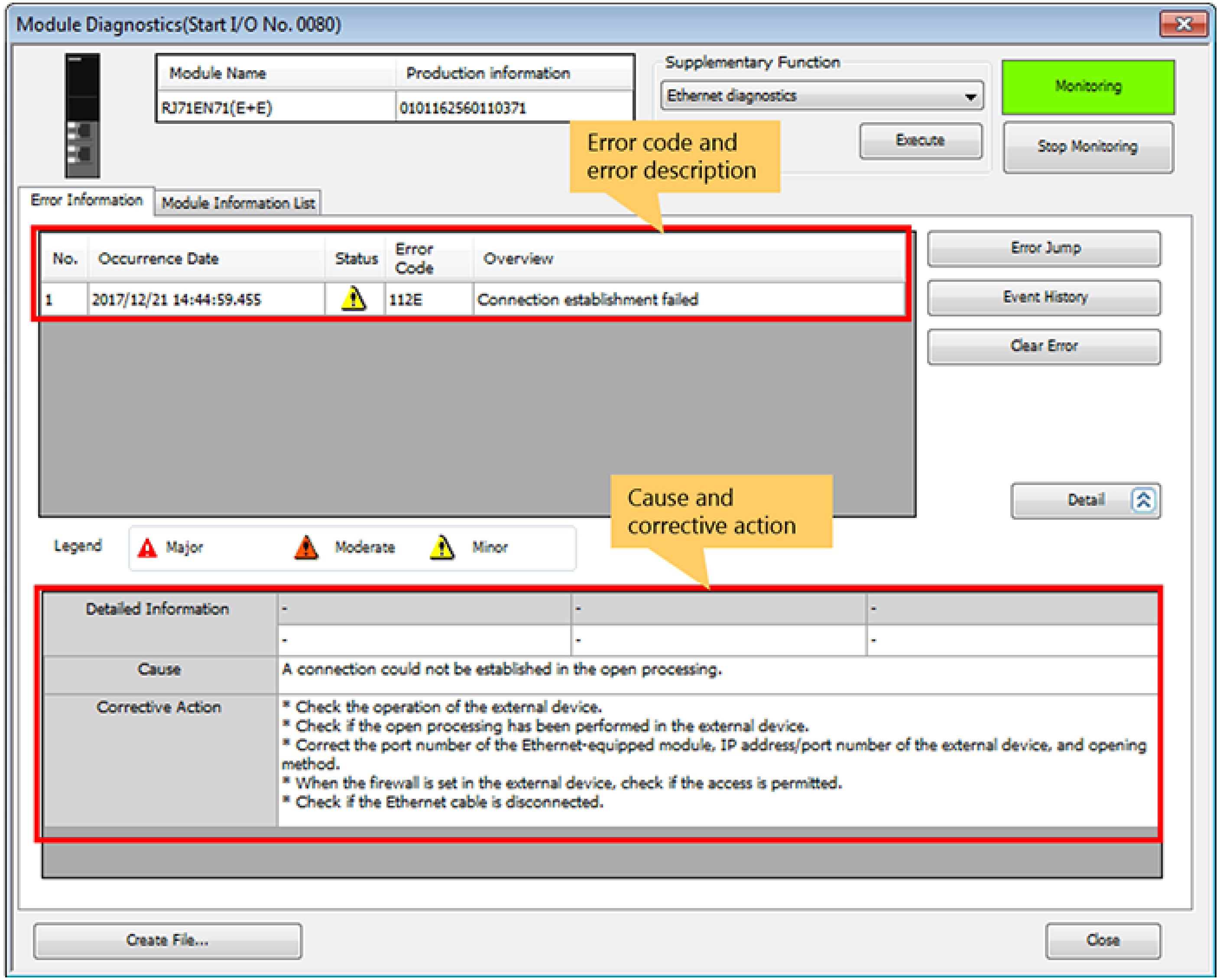


LED indicator on the Ethernet module

LED	Description	Indication		Troubleshooting procedure
		Normal	Error	
RUN	Operating status	On	Off	• Check that the Ethernet module is correctly installed
ERR	Error status	Off	On or flashing	• Check the details using the module diagnostics function of the engineering software
SD/RD	Data communication status	On	Off	• Check the cable connections, module parameters, and control programs for any issues or errors
P ERR	P1 or P2 error status	On	On or flashing	-

4.3 Checking for errors by the module diagnostics

If the network does not seem to be operating normally, use the engineering software to check the details. Execute [Module Diagnostics] from the system monitor on the [Diagnostics] menu. The details and corrective action for the error appear.



Module Name	Production information
RJ71EN71(E+E)	0101162560110371

Supplementary Function: Ethernet diagnostics

Error code and error description

No.	Occurrence Date	Status	Error Code	Overview
1	2017/12/21 14:44:59.455	⚠	112E	Connection establishment failed

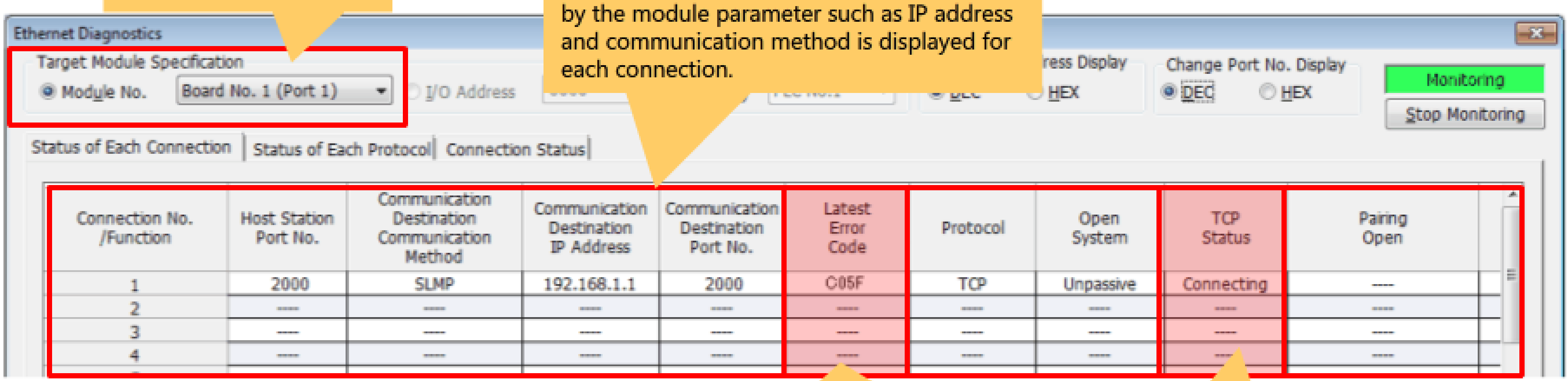
Legend ▲ Major ▲ Moderate ⚠ Minor

Cause and corrective action

Detailed Information	-	-	-
	-	-	-
Cause	A connection could not be established in the open processing.		
Corrective Action	* Check the operation of the external device. * Check if the open processing has been performed in the external device. * Correct the port number of the Ethernet-equipped module, IP address/port number of the external device, and opening method. * When the firewall is set in the external device, check if the access is permitted. * Check if the Ethernet cable is disconnected.		

4.4 Using Ethernet diagnostics to check network status

From the [Diagnostics] menu of the engineering software, execute [Ethernet Diagnostics] to check the communication status between the Ethernet module and external devices.



Ethernet Diagnostics

Target Module Specification: ● Module No. Board No. 1 (Port 1) ○ I/O Address

Change Port No. Display: ● DEC ○ HEX

Status of Each Connection | Status of Each Protocol | Connection Status

Connection No. /Function	Host Station Port No.	Communication Destination Communication Method	Communication Destination IP Address	Communication Destination Port No.	Latest Error Code	Protocol	Open System	TCP Status	Pairing Open
1	2000	SLMP	192.168.1.1	2000	C05F	TCP	Unpassive	Connecting	----
2	----	----	----	----	----	----	----	----	----
3	----	----	----	----	----	----	----	----	----
4	----	----	----	----	----	----	----	----	----

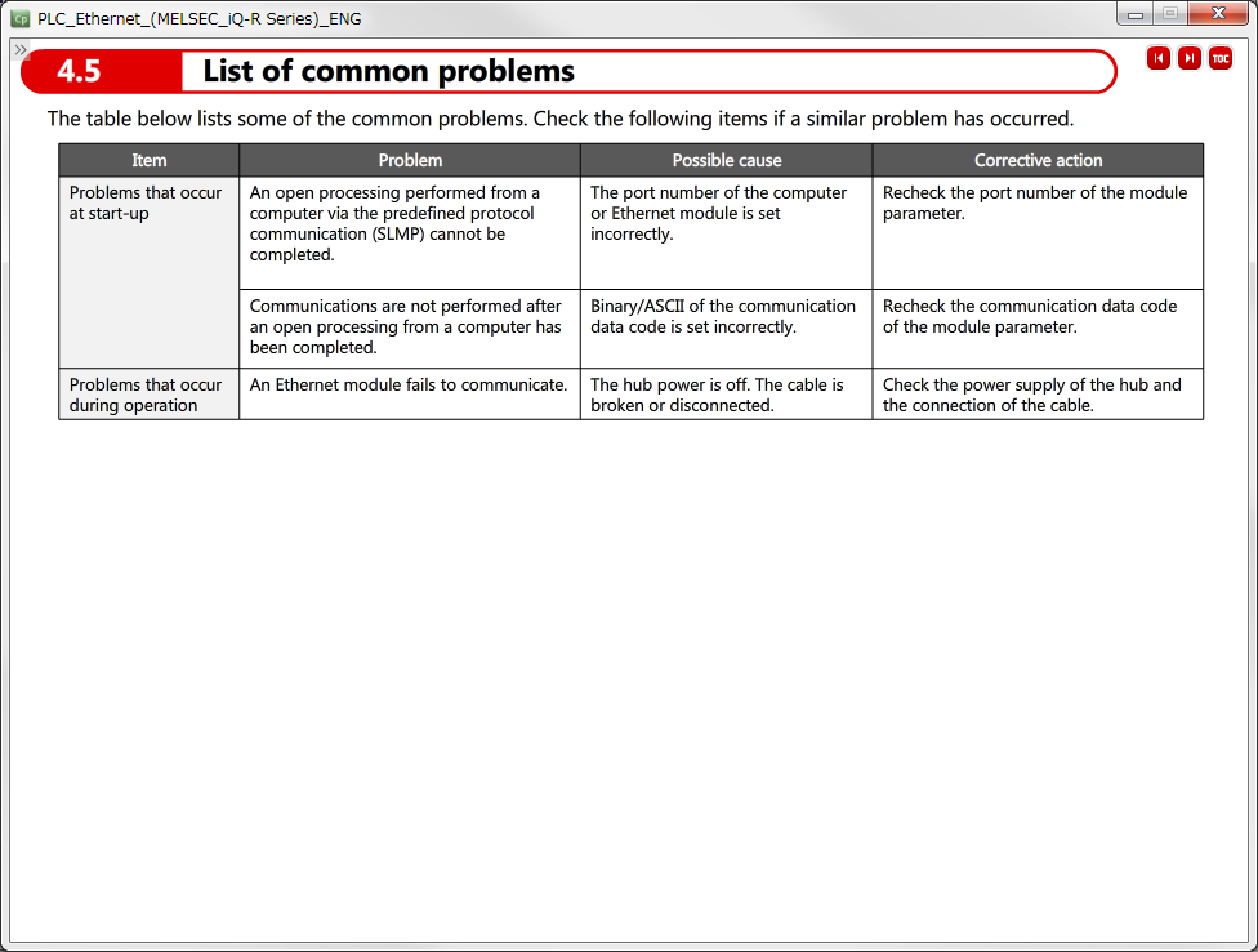
Specify the Ethernet module to be checked.

The communication setting status configured by the module parameter such as IP address and communication method is displayed for each connection.

Error codes representing the error definition appear here when an error occurs.

For details on specific error codes, refer to the manual for the Ethernet module used.

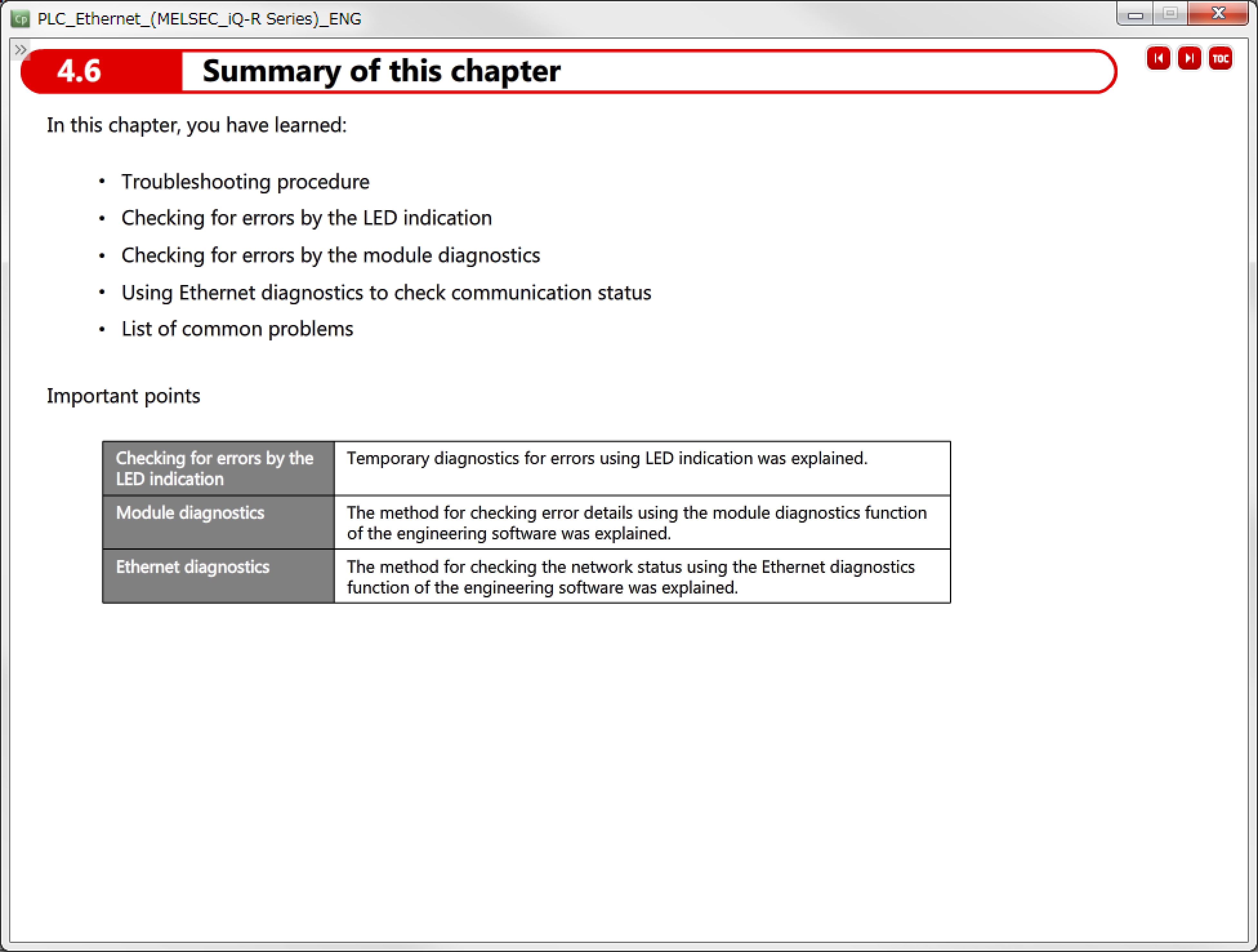
TCP protocol connection status appears here ("Connecting" or "Disconnected").



4.5 List of common problems

The table below lists some of the common problems. Check the following items if a similar problem has occurred.

Item	Problem	Possible cause	Corrective action
Problems that occur at start-up	An open processing performed from a computer via the predefined protocol communication (SLMP) cannot be completed.	The port number of the computer or Ethernet module is set incorrectly.	Recheck the port number of the module parameter.
	Communications are not performed after an open processing from a computer has been completed.	Binary/ASCII of the communication data code is set incorrectly.	Recheck the communication data code of the module parameter.
Problems that occur during operation	An Ethernet module fails to communicate.	The hub power is off. The cable is broken or disconnected.	Check the power supply of the hub and the connection of the cable.



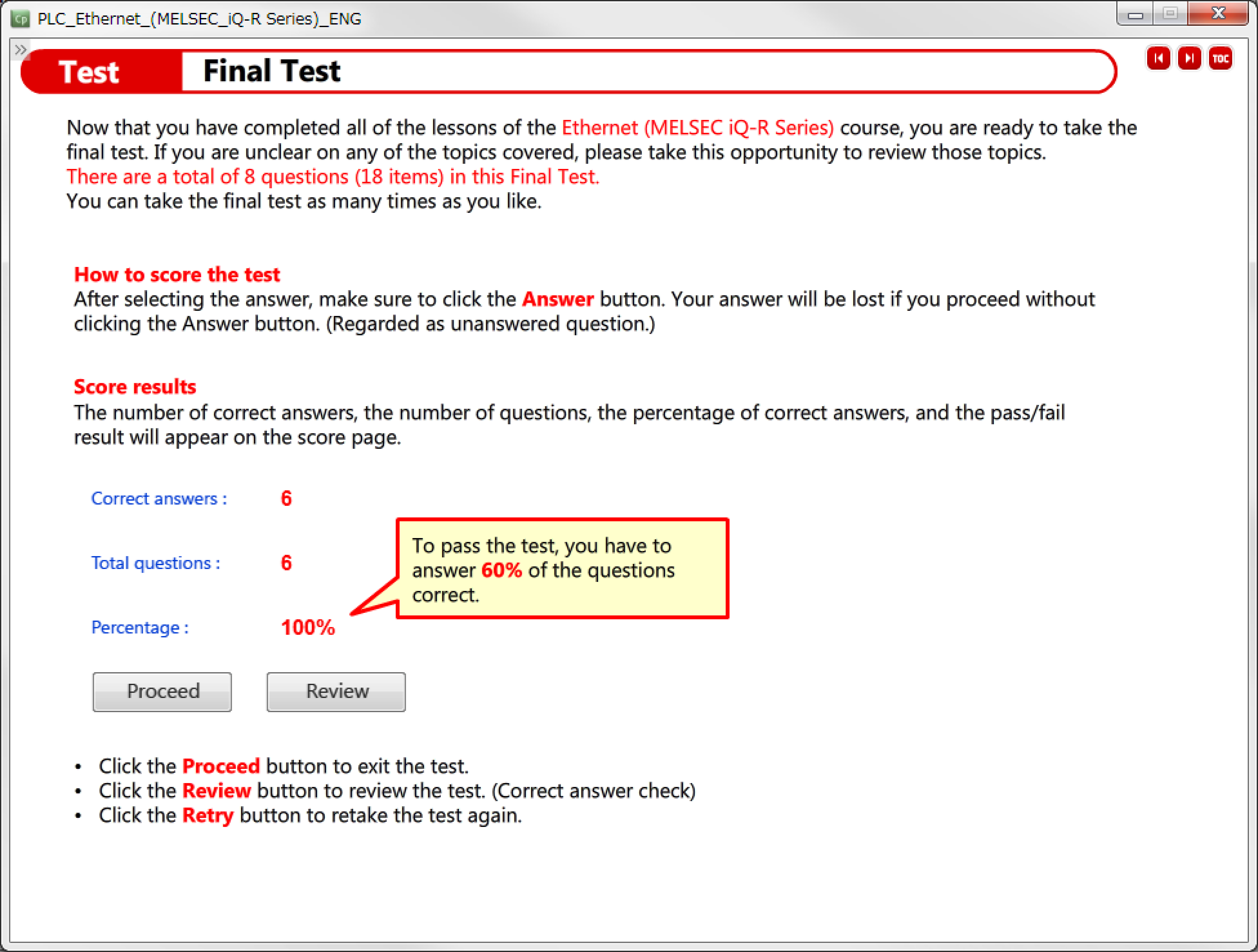
4.6 Summary of this chapter

In this chapter, you have learned:

- Troubleshooting procedure
- Checking for errors by the LED indication
- Checking for errors by the module diagnostics
- Using Ethernet diagnostics to check communication status
- List of common problems

Important points

Checking for errors by the LED indication	Temporary diagnostics for errors using LED indication was explained.
Module diagnostics	The method for checking error details using the module diagnostics function of the engineering software was explained.
Ethernet diagnostics	The method for checking the network status using the Ethernet diagnostics function of the engineering software was explained.



Test

Final Test

Now that you have completed all of the lessons of the Ethernet (MELSEC iQ-R Series) 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 8 questions (18 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.)

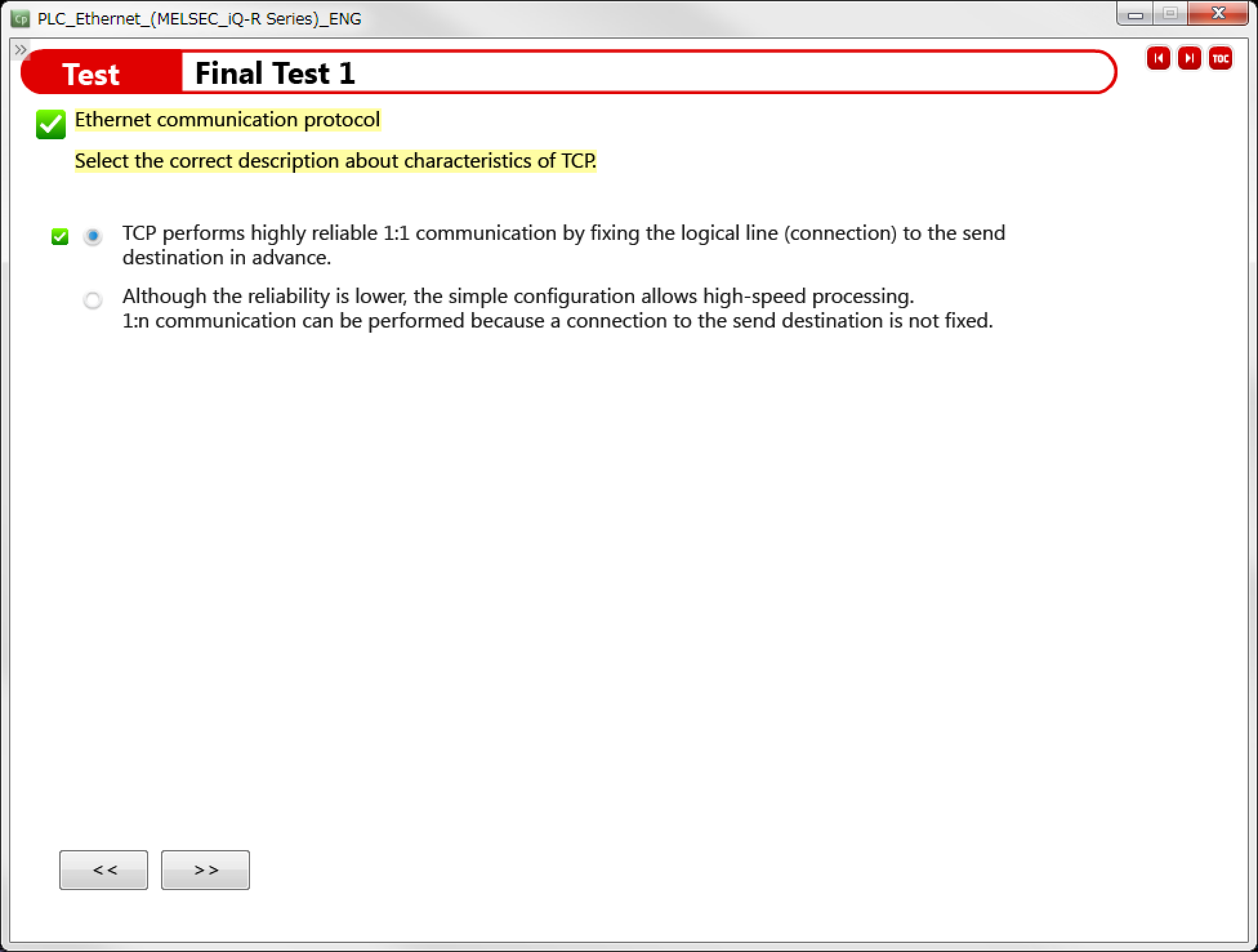
Score results

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 :	6
Total questions :	6
Percentage :	100%

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

- 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.



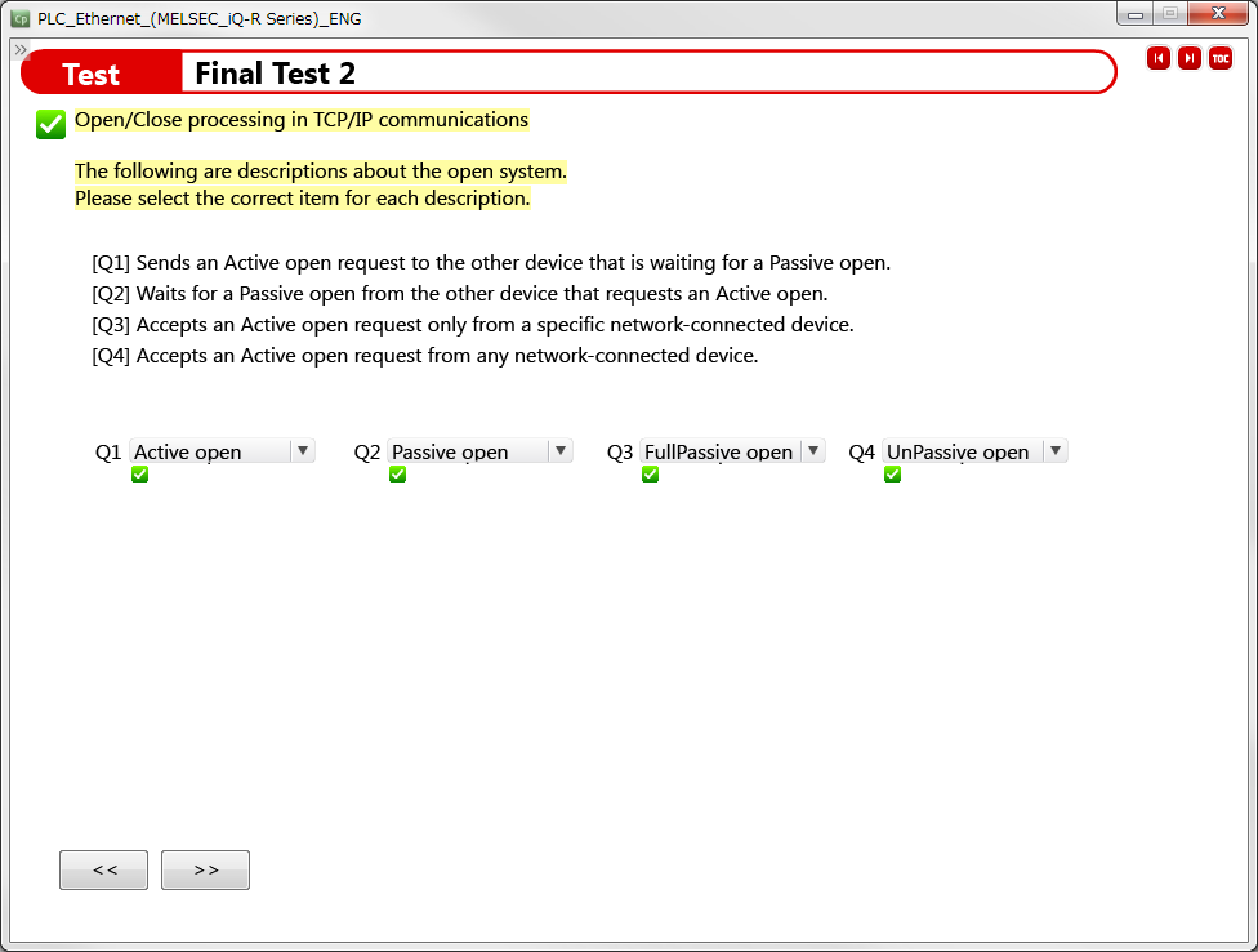
Test **Final Test 1**

✓ Ethernet communication protocol

Select the correct description about characteristics of TCP.

✓ ● TCP performs highly reliable 1:1 communication by fixing the logical line (connection) to the send destination in advance.

○ Although the reliability is lower, the simple configuration allows high-speed processing.
1:n communication can be performed because a connection to the send destination is not fixed.



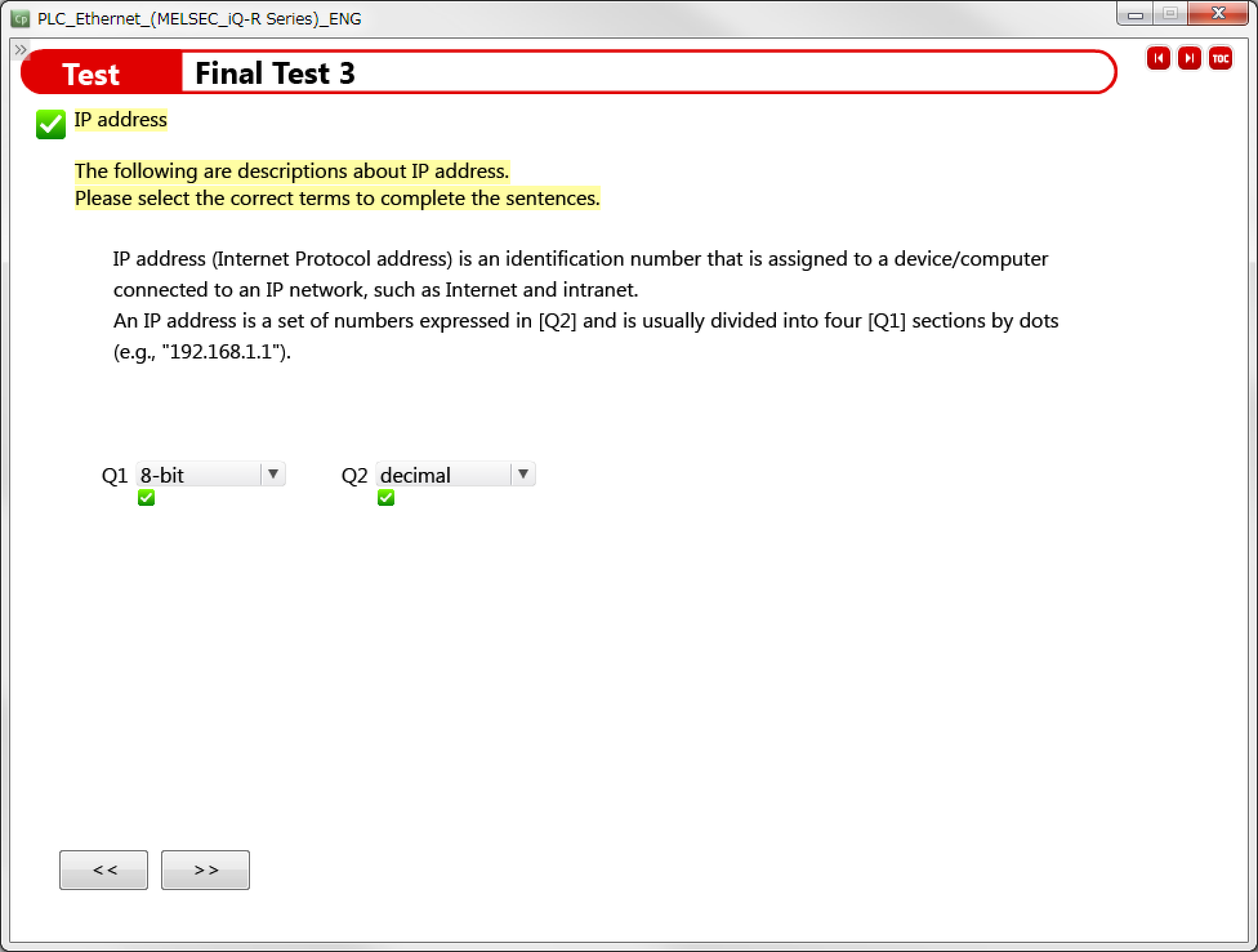
Test | Final Test 2

✓ Open/Close processing in TCP/IP communications

The following are descriptions about the open system.
Please select the correct item for each description.

[Q1] Sends an Active open request to the other device that is waiting for a Passive open.
[Q2] Waits for a Passive open from the other device that requests an Active open.
[Q3] Accepts an Active open request only from a specific network-connected device.
[Q4] Accepts an Active open request from any network-connected device.

Q1 Active open ▼ ✓
Q2 Passive open ▼ ✓
Q3 FullPassive open ▼ ✓
Q4 UnPassive open ▼ ✓



Test Final Test 3

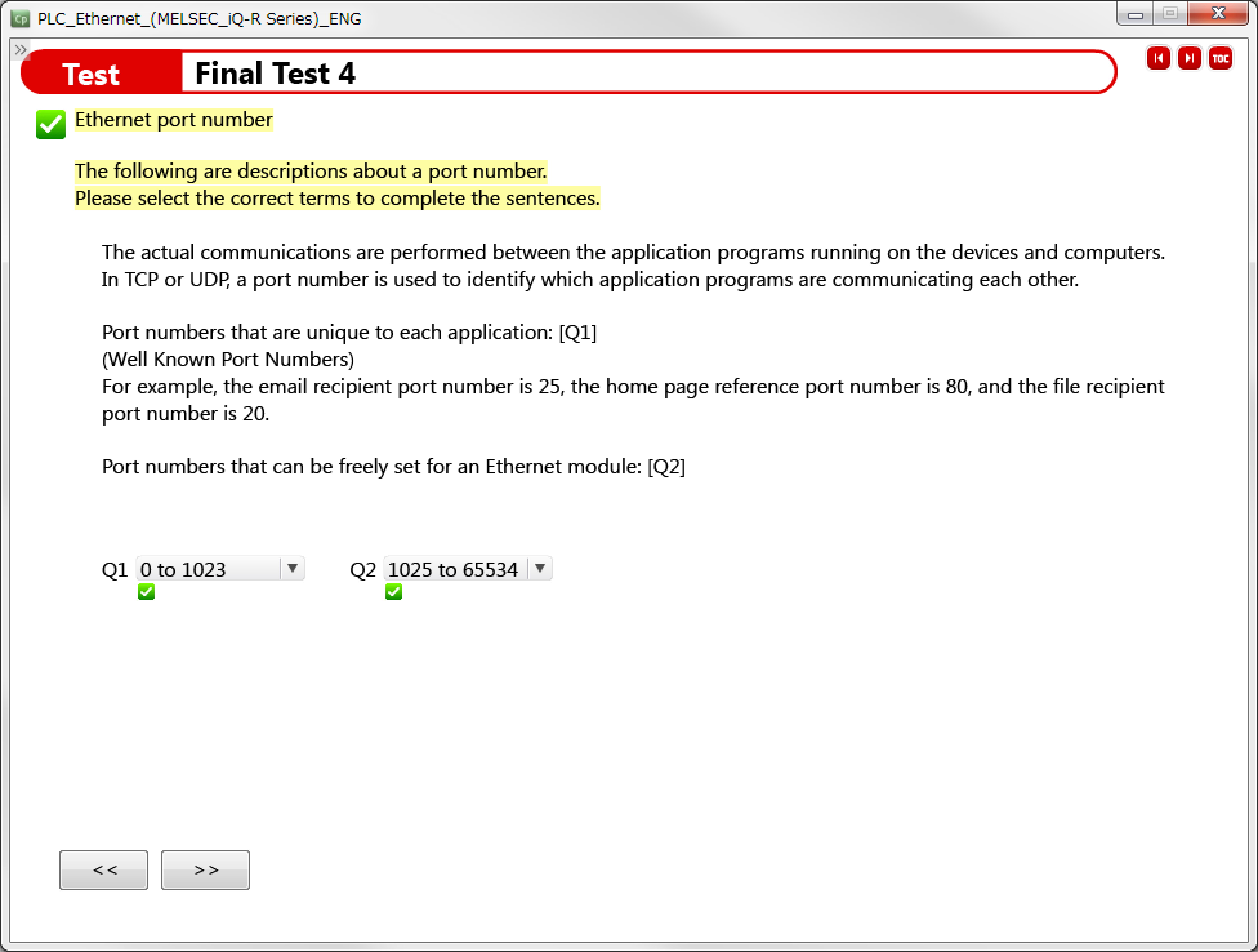
✓ IP address

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

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 [Q2] and is usually divided into four [Q1] sections by dots (e.g., "192.168.1.1").

Q1 8-bit ▼ ✓

Q2 decimal ▼ ✓



Test Final Test 4

✓ Ethernet port number

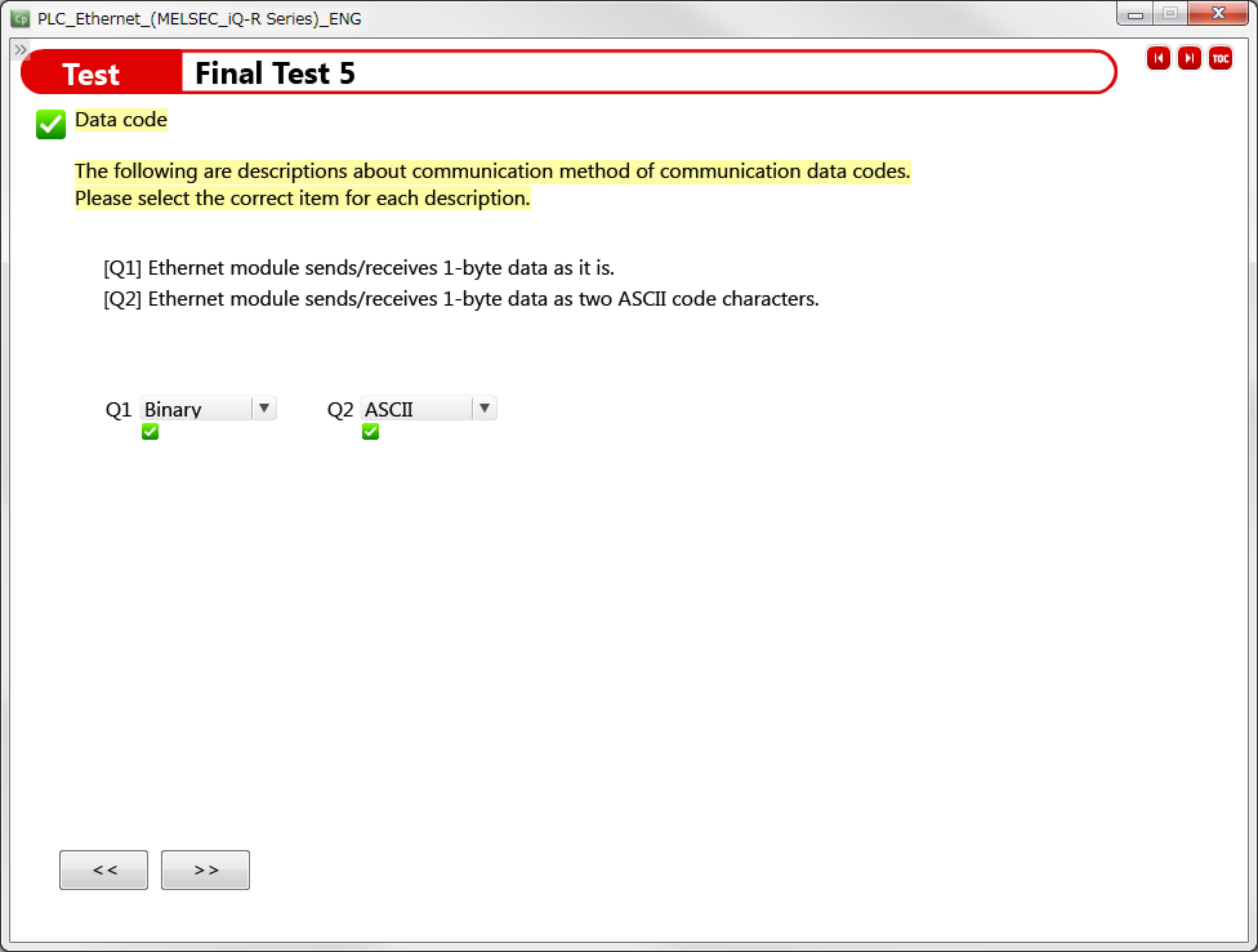
The following are descriptions about a port number.
Please select the correct terms to complete the sentences.

The actual communications are performed between the application programs running on the devices and computers. In TCP or UDP, a port number is used to identify which application programs are communicating each other.

Port numbers that are unique to each application: [Q1]
(Well Known Port Numbers)
For example, the email recipient port number is 25, the home page reference port number is 80, and the file recipient port number is 20.

Port numbers that can be freely set for an Ethernet module: [Q2]

Q1 0 to 1023 ▼ ✓ Q2 1025 to 65534 ▼ ✓



Test

Final Test 5

✓ Data code

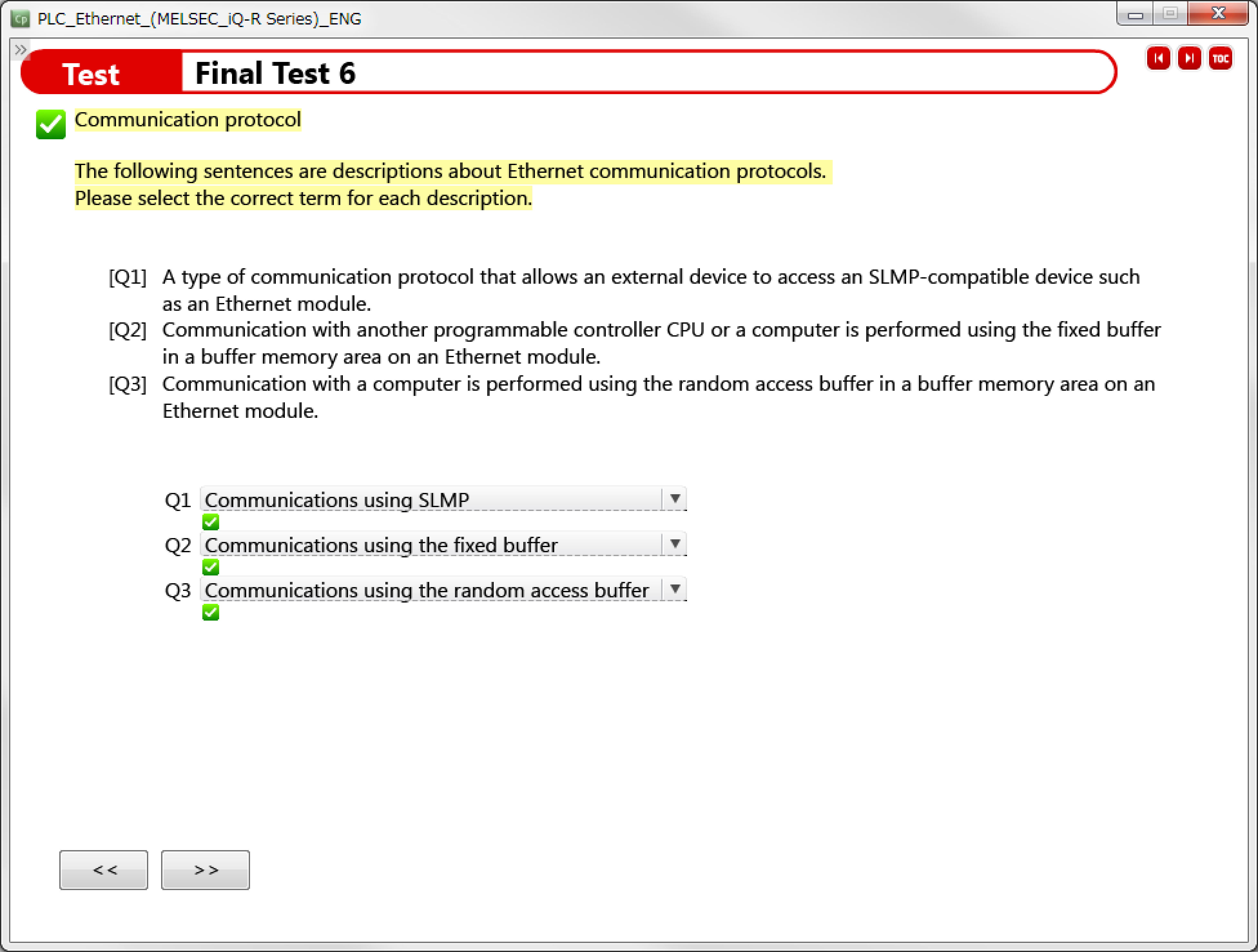
The following are descriptions about communication method of communication data codes.
Please select the correct item for each description.

[Q1] Ethernet module sends/receives 1-byte data as it is.

[Q2] Ethernet module sends/receives 1-byte data as two ASCII code characters.

Q1 Binary ▼ ✓

Q2 ASCII ▼ ✓



Test Final Test 6

✓ Communication protocol

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

[Q1] A type of communication protocol that allows an external device to access an SLMP-compatible device such as an Ethernet module.

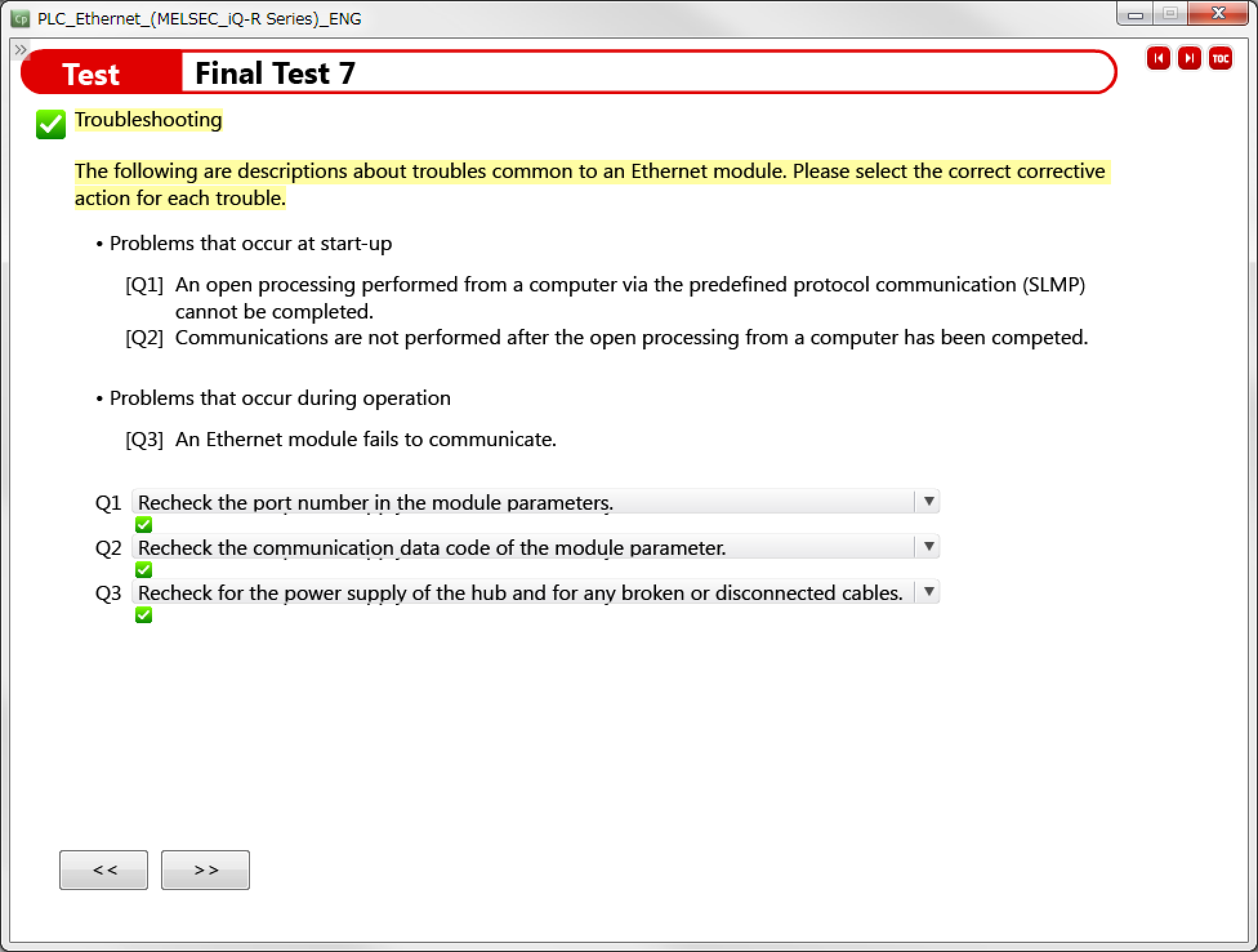
[Q2] Communication with another programmable controller CPU or a computer is performed using the fixed buffer in a buffer memory area on an Ethernet module.

[Q3] Communication with a computer is performed using the random access buffer in a buffer memory area on an Ethernet module.

Q1 Communications using SLMP ▼
✓

Q2 Communications using the fixed buffer ▼
✓

Q3 Communications using the random access buffer ▼
✓



Test Final Test 7

✓ Troubleshooting

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

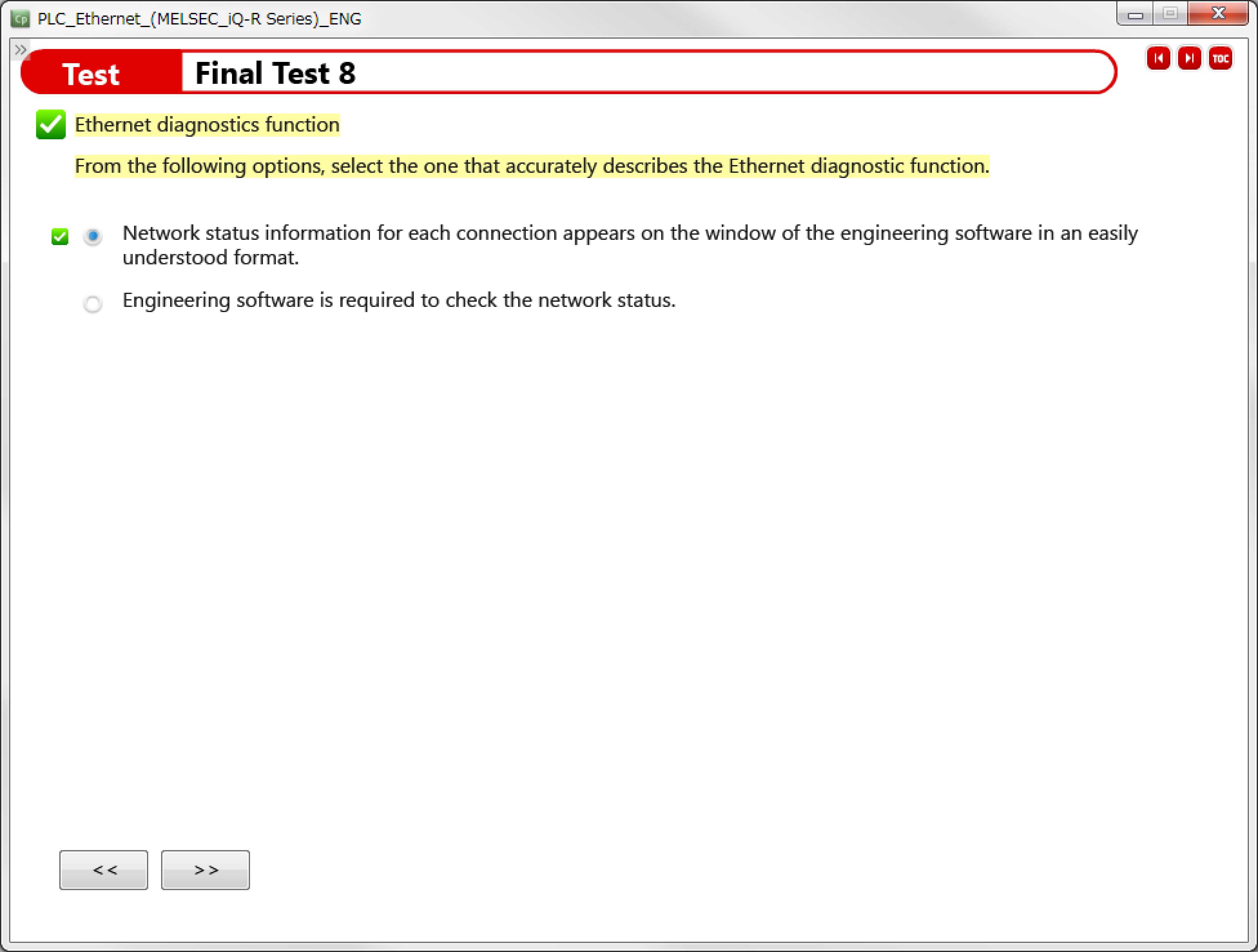
- Problems that occur at start-up

  [Q1] An open processing performed from a computer via the predefined protocol communication (SLMP) cannot be completed.
  [Q2] Communications are not performed after the open processing from a computer has been competed.

- Problems that occur during operation

  [Q3] An Ethernet module fails to communicate.

Q1 Recheck the port number in the module parameters. ▼
✓
Q2 Recheck the communication data code of the module parameter. ▼
✓
Q3 Recheck for the power supply of the hub and for any broken or disconnected cables. ▼
✓



Test

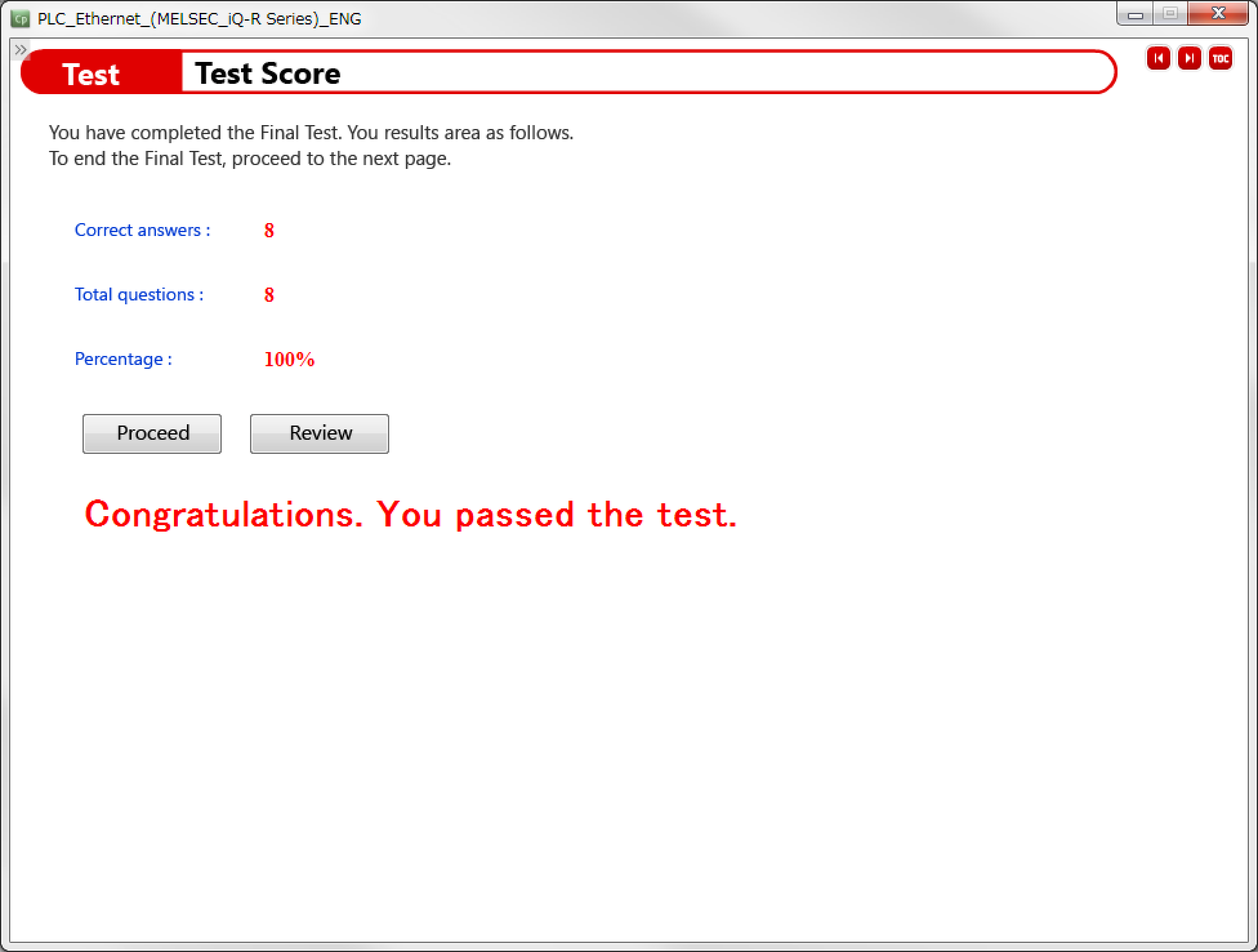
Final Test 8

✓ Ethernet diagnostics function

From the following options, select the one that accurately describes the Ethernet diagnostic function.

✓ ● Network status information for each connection appears on the window of the engineering software in an easily understood format.

○ Engineering software is required to check the network status.

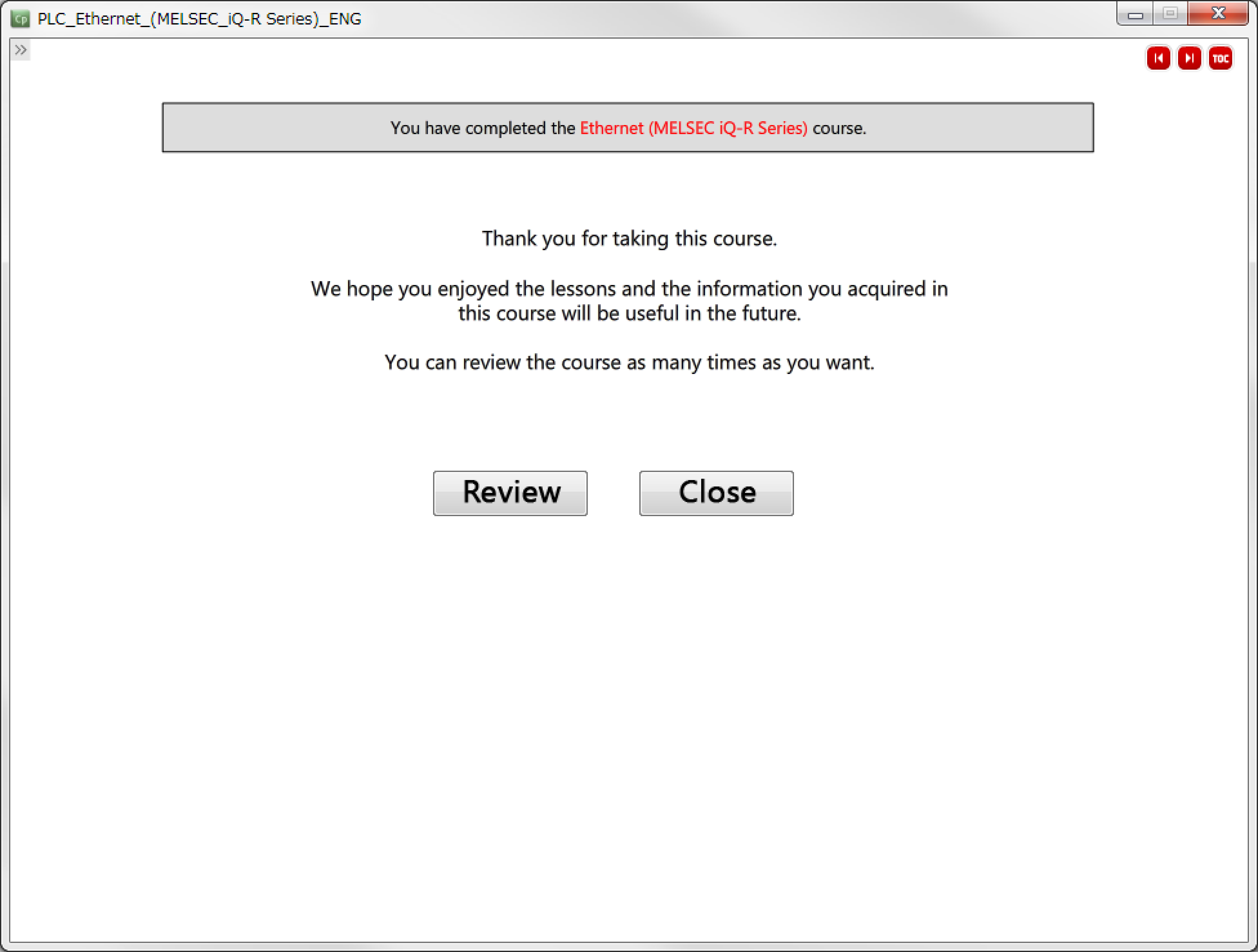


Test Test Score

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

Correct answers :	8
Total questions :	8
Percentage :	100%

Congratulations. You passed the test.



You have completed the Ethernet (MELSEC iQ-R Series) 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.