OMRON

Programmable Controller CJ-series

General Ethernet (TCP/IP) Connection Guide

OMRON Corporation
Industrial Handheld DPM Reader
V460-H-series

Network

Connection

Guide

About Copyrights and Trademarks

Microsoft product screen shots used with permission from Microsoft.

Windows is a registered trademark of Microsoft Corporation in the USA and other countries.

ODVA, EtherNet/IP are trademarks of ODVA.

Sysmac and SYSMAC are trademarks or registered trademarks of OMRON Corporation in Japan and other countries for OMRON factory automation products.

Company names and product names in this document are the trademarks or registered trademarks of their respective companies.

Contents

1. Related Manuals

The following manuals are related to this document.

To ensure system safety, make sure to always read and follow the information provided in all *Safety Precautions* and *Precautions for Safe Use* in the manuals for each device which is used in the system.

Cat. No.	Model	Manual name
W420	CS1W-ETN21 CJ1W-ETN21	CJ-series Ethernet Units Operation Manual Construction of Networks
W421	CS1W-ETN21 CJ1W-ETN21	CJ-series Ethernet Units Operation Manual Construction of Applications
W446	—	CX-Programmer Operation Manual
W474	CJ2□-CPU□□	CJ Series Instructions Reference Manual
Z461	V460-H Series	V460-H Industrial Handheld DPM Reader User Manual
Z462	V460-H Series	V460-H Industrial Handheld DPM Reader Communication Manual

2. Terms and Definitions

Below is a list of terms used in this manual and their definitions.

| Term	Description/Definition
IP Address	Ethernet uses IP addresses to achieve communications. Each IP address (specifically, Internet Protocol address) identifies a specific node (host computer, controller, etc.) on an Ethernet network, IP addresses must be set and managed so that they are not duplicated.
Socket	A socket is an interface that allows you to directly use TCP or UDP functions from the user program. CJ Series Programmable Controllers support socket services in the following ways. • Manipulating dedicated control bits in the CPU Bus Unit Area in the CIO Area • Sending FINS commands (CMND instructions) to the Ethernet Unit To use socket services, you need to establish a connection with a remote node and disconnect it after use. In this document, processing for establishing a connection is referred to as "socket open" or "TCP open" and for disconnecting it as "socket close" or "close". You can use the socket services to send and receive arbitrary data to and from the remote node.
Active and Passive	When you open a TCP socket connection with nodes, open processing is executed for each node. The method to open a connection differs depending on whether the node is to serve as a client or server. In this document, processing to open a connection as a server is referred to as "passive open" and as a client is referred to as "active open" or "active open processing"
keep-alive Function	When a remote node (server or client) does not respond for a set period of time or longer in TCP/IP socket services, the keep-alive function sends a communications frame to the node to check the connection status. If the node does not respond to it, the function performs this check at a certain interval, and closes the connection if it does not respond to all check frames.
linger function	This is a TCP socket option that sends RST data when the TCP socket is closed. This enables immediate open processing using the same port number, without waiting for the port to be opened. If the linger option is not specified, the controller issues FIN data when the TCP socket is closed and, after that, performs end control such as a send data arrival check with the remote node for approximately 1 minute. Therefore, TCP sockets with the same port number may not be used immediately.

3. Restrictions and Precautions

(1) Before building a system, understand the specifications of devices which are used in the system. Allow some margin for ratings and performance, and provide safety measures such as installing a safety circuit in order to minimize the risk in case of failure.

(2) To ensure system safety, make sure to read and follow the information provided in all *Safety Precautions* and *Precautions for Safe Use* in the manuals for each device which is used in the system.

(3) The user is encouraged to confirm the standards and regulations that the system must conform to.

(4) It is prohibited to copy, to reproduce, and to distribute a part or the whole of this document without the permission of OMRON Corporation.

(5) The information contained in this document is current as of November 2023.
It is subject to change for improvement without notice.

The following notations are used in this document.

⚠ WARNING	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or may result in serious injury or death. Additionally, there may be severe property damage.
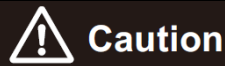
⚠ Caution	Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

Precautions for Safe Use

Precautions on what to do and what not to do to ensure safe usage of the product.

Precautions for Correct Use

Precautions on what to do and what not to do to ensure proper operation and performance.

Note

Additional information to read as required.
This information is provided to increase understanding or make operation easier.

Symbols

The filled circle symbol indicates operations that you must do.
The specific operation is shown in the circle and explained in text.
This example shows a general precaution for something that you must do.

4. Overview

This document describes the procedures for connecting the OMRON Industrial Handheld DPM Reader products (V460-H Series) to a CJ Series Programmable Controller (hereinafter referred to as the controller) via Ethernet and for checking their connections.

You can establish an Ethernet communication connection by understanding the setting procedures and key points of setup through the Ethernet communications settings in the "CX-Programmer Project File" prepared in advance.

In this project file, the Ethernet connection is checked by sending a read trigger command to the code reader and receiving the read data from it.

Obtain the latest version of the CX-Programmer Project File from OMRON in advance.

Name	Filename	Version
CX-Programmer Project File (Extension: cxp)	OMRON_V460_CJ_ETN(TCP)_V100.cxp	Ver. 1.00

⚠ **Caution**

The purpose of this document is to describe the wiring methods, communication settings, and setting procedures required to establish a connection for communications with applicable devices. In addition, the program used in this document is designed to check that the connection has been correctly performed (connection check). Since the program is not intended for permanent use on-site, full consideration is not given to functionality and performance. When configuring an actual system, please refer to the wiring methods, communication settings, and setting procedures described in this document to design and create a program that meets your purpose.

5. Applicable Products and Device Configuration

5.1. Applicable Products

The applicable devices that are required to ensure a connection are as follows:

Manufacturer	Name	Model	Version
OMRON	Ethernet Unit	CJ1W-ETN21	Same or later version as indicated in section 5.2.
OMRON	CJ2 Series CPU Unit	CJ2□-CPU□□	
OMRON	Code reader	V460-H0PX	

Note

From among the above applicable devices, this document uses the devices listed in section 5.2 for the connection check. When using devices that are not described in section 5.2, check the connection according to this document.

Note

This document describes the procedures for establishing the communication connection of the device, and does not describe the operation, installation and wiring method of the device. For details on the above products (other than communication connection procedures), please refer to the instruction manual for the product or contact OMRON.

Precautions for Correct Use

The connection and connection check procedures described in this document use the devices listed in section 5.2, from among the above applicable devices.
You cannot use devices with versions earlier than the versions listed in section 5.2.
To use models that are not listed in section 5.2. or versions that are later than those listed in section 5.2., check the differences in the specifications according to their instruction manuals before operating the devices.

5.2. Device Configuration

The system components required for reproducing the connection procedures described in this document are as follows.
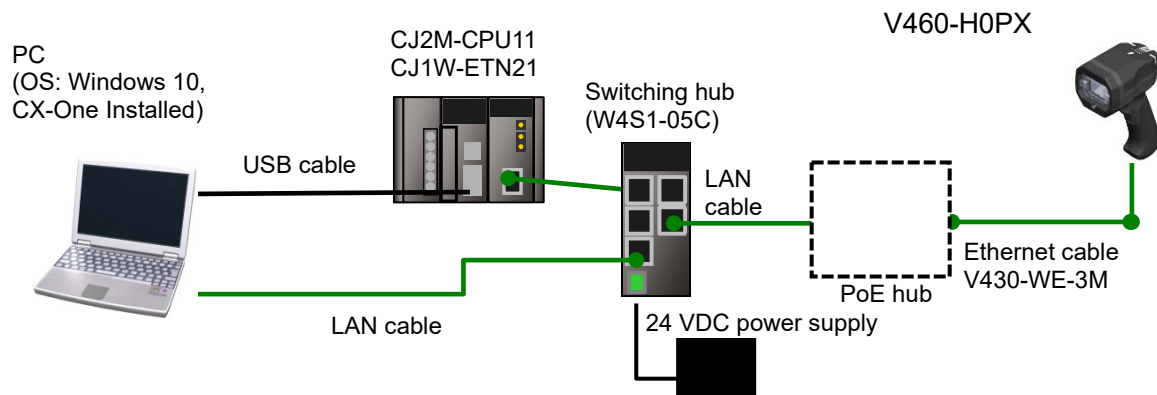
Manufacturer	Name	Model	Version
OMRON	CPU Unit	CJ2M-CPU11	Ver. 2.0
OMRON	Ethernet Unit	CJ1W-ETN21	Ver. 1.5
OMRON	Power Supply Unit	CJ1W-PA202	---
OMRON	CX-One	CXONE-AL□□C-V4 /AL□□D-V4	Ver. 4.□□
OMRON	Switching hub	W4S1-05C	Ver. 1.00
---	24 VDC power supply (for switching hub)	---	---
OMRON	CX-Programmer	(Included with CX-One)	Ver. 9.72
OMRON	CX-Protocol	(Included with CX-One)	Ver. 2.03
OMRON	CX-Protocol Project File (Ladder Program)	OMRON_V460_CJ_ETN(TCP)_V1_00.cxp	Ver. 1.00
---	PC (OS: Windows 10)	---	---
---	USB cable (USB 2.0-compliant B-type connector)	---	---
OMRON	Ethernet cable	V430-WE-3M	---
---	LAN cable (Standard Ethernet cable)	---	---
OMRON	Code reader	V460-H0PX	Ver. 1.0
---	Single port PoE injector	98-9000311-01	---

Precautions for Correct Use

Obtain the latest version of the above CX-Programmer Project File from OMRON in advance. (Contact OMRON for information on how to obtain this file.)

Precautions for Correct Use

This document assumes that the USB is used to connect the PLC. For information on how to install the USB driver, refer to the *CX-Programmer Operation Manual* (Cat. No. W446).

Note

Refer to the *Industrial Switching Hub W4S1 Series User Manual* (0969584-7) for power supply specifications that can be used for 24 VDC power supply (for the switching hub).

6. Ethernet Settings

This section shows the specifications of the communication parameter settings, variable names and other information provided in this document.

📝 **Note**

This document and the project file only cover the operations that you can perform using the settings and commands described in this section. To use communication settings that are not described here, you need to modify the project file.

6.1. Ethernet Communication Settings

The settings required to perform Ethernet communications are as follows.

6.1.1. Communications Settings for Setting PC and Code Reader

This document assumes that you use the settings below to set the code reader using a setting PC.

Parameter name	Setting PC	Code reader
IP address	192.168.188.100	192.168.188.2 (default)
Subnet mask	255.255.0.0	255.255.0.0 (default)
Gateway	Blank (default)	0.0.0.0 (default)

* For the use cases in this document, setting the gateway is unnecessary because the devices are connected within the same segment of the network.

6.1.2. Communication Settings for Ethernet Unit and Code Reader

It is assumed that you use the settings below to connect the Ethernet Unit and the code reader.

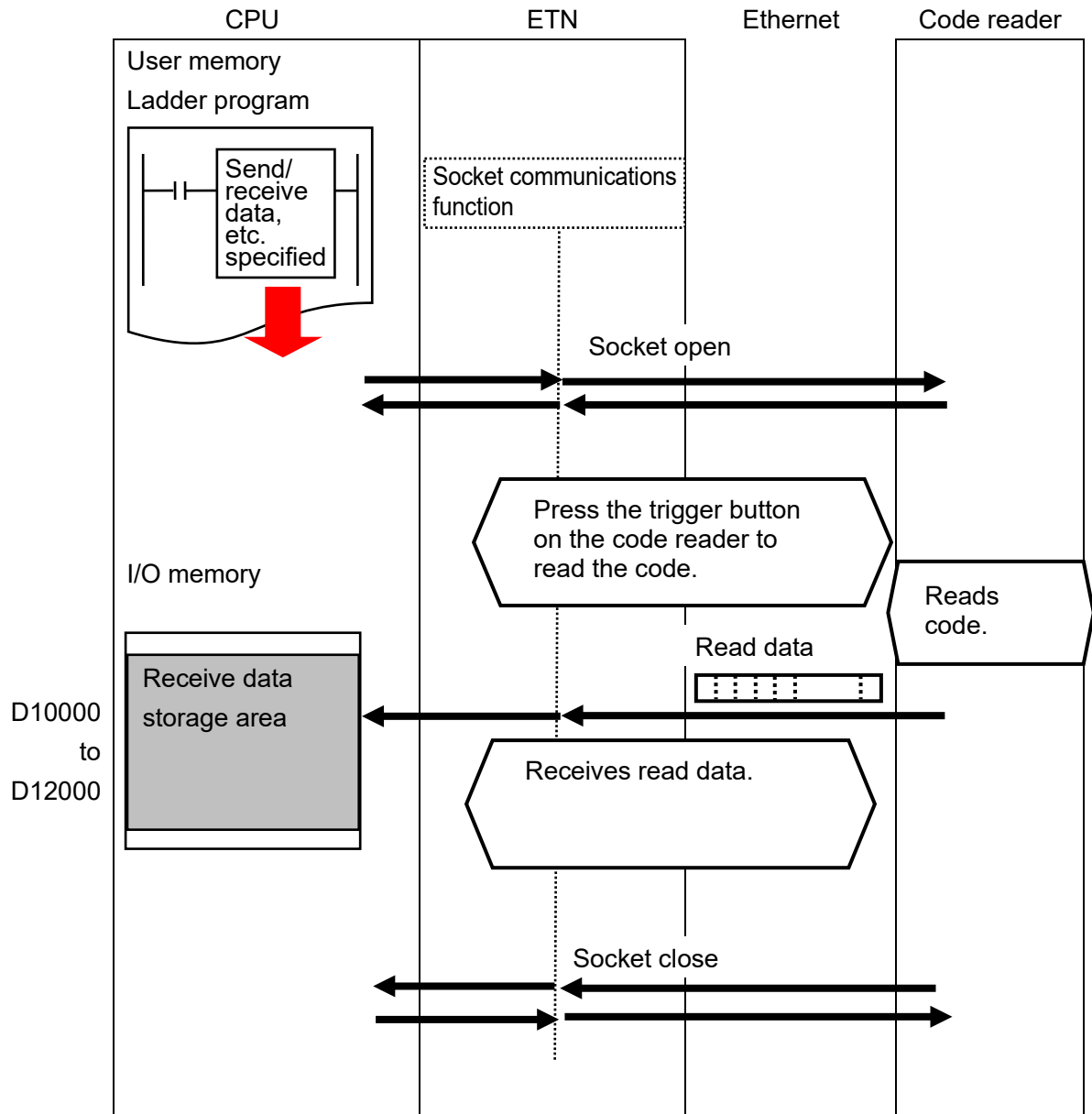|	CJ1W-ETN21	Code reader
Unit number	0 (default)	---
Node address	01 (default)	---
Automatic setting	---	OFF
IP address	192.168.188.1	192.168.188.2
Subnet mask	255.255.0.0	255.255.0.0 (default)
Port number	(set by software part)	2001 (fixed)

* For the use cases in this document, setting the gateway is unnecessary because the devices are connected within the same segment of the network.

6.2. Example of Connection Check for Communications

This document assumes that you use a ladder program (also referred to as the software part) to execute "socket open", "send and receive", and "socket close" from the PLC to the code reader (V460 Series).

The controller sends a "read trigger" command to the code reader. The code reader sends the read data back to the controller.

An overview of the operation is shown below.

CPU	ETN	Ethernet	Code reader

User memory

Ladder program

| Send/ receive data, etc. specified |

Socket communications function

Socket open

Press the trigger button on the code reader to read the code.

Reads code.

I/O memory

Read data

Receive data storage area
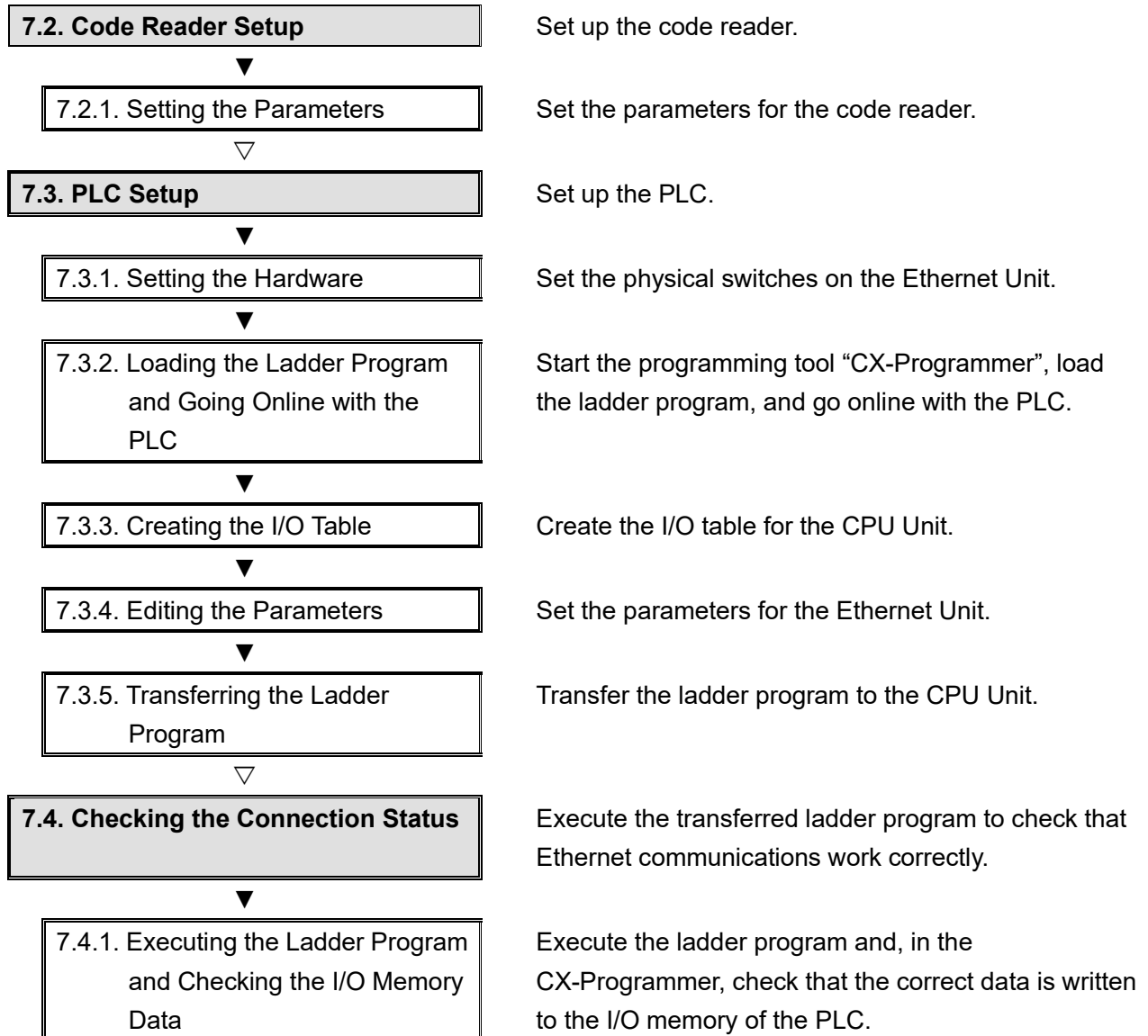
D10000 to D12000

Receives read data.

Socket close

7. Connection Procedure

This section describes the procedures for connecting the controller to an Ethernet network. In this document, it is assumed that the controller and the code reader use the factory default settings. For how to initialize the devices, refer to *Section 8. Initializing the System*.

7.1. Operation Flow

The procedures for connecting and setting up the Ethernet Unit are as follows.

7.2. Code Reader Setup	Set up the code reader.
▼	
7.2.1. Setting the Parameters	Set the parameters for the code reader.
▽	
7.3. PLC Setup	Set up the PLC.
▼	
7.3.1. Setting the Hardware	Set the physical switches on the Ethernet Unit.
▼	
7.3.2. Loading the Ladder Program and Going Online with the PLC	Start the programming tool "CX-Programmer", load the ladder program, and go online with the PLC.
▼	
7.3.3. Creating the I/O Table	Create the I/O table for the CPU Unit.
▼	
7.3.4. Editing the Parameters	Set the parameters for the Ethernet Unit.
▼	
7.3.5. Transferring the Ladder Program	Transfer the ladder program to the CPU Unit.
▽	
7.4. Checking the Connection Status	Execute the transferred ladder program to check that Ethernet communications work correctly.
▼	
7.4.1. Executing the Ladder Program and Checking the I/O Memory Data	Execute the ladder program and, in the CX-Programmer, check that the correct data is written to the I/O memory of the PLC.

7.2. Code Reader Setup

Set up the code reader.

Precautions for Correct Use

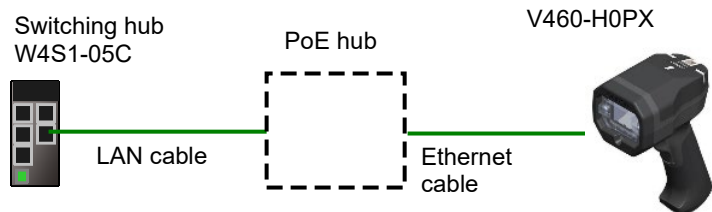Use a PC (personal computer) to set the parameters for the code reader.

Note that you may need to change the PC settings depending on the condition of your PC.

7.2.1. Setting the Parameters

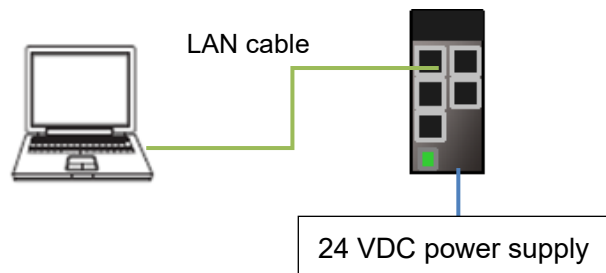Set the parameters for the code reader.

Set the IP address of your PC to *192.168.188.100* and its subnet mask to *255.255.0.0*.

1	Connect the cord reader and the switching hub to the PoE injector with cables.	
2	Connect the PC to the switching hub with a LAN cable. Connect 24 VDC power supply (for the switching hub) to the switching hub.	
3	Set the IP Address of the PC. For the IP address, enter *192.168.188.100*. For the subnet mask, enter *255.255.0.0*. For the procedure to open the screen on the right, please refer to step 4.	

| 4	**Static connection (Setting the fixed IP address)** (1) Click the **Start** Button at the lower left corner of the screen, and then click **Settings**. (2) Select **Windows Settings – Network & Internet**. (3) Under **Change your network settings**, click **Change adapter options**. (4) In **Network Connections**, right-click on the **Ethernet** icon and select **Properties**. (5) In the **Ethernet Properties** Dialog Box, select **Internet Protocol Version 4 (TCP/IPv4)**, and click the **Properties** Button. Set the IP Address of the PC to *192.168.188.100*. (6) Click the **OK** Button.
5	Start your browser and enter *http://192.168.188.2*. "Google Chrome" is the recommended browser.
6	When the WebLinkHH startup screen is displayed, go to step 8. If you cannot access by WebLinkHH, go to step 7.
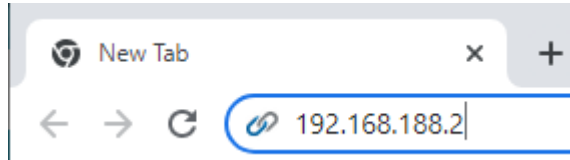
7	If the WebLinkHH startup screen does not appear, it means that communications are not
	established between the code reader and the PC. Please check the following.
	• The code reader and the PC have a proper physical (cable) connection.
	    → Refer to steps 1 and 2 for checking the connection.
	• The IP Addresses of the PC and code reader are set correctly.
	    → Refer to step 4 for setting the IP address of the PC.
	For other measures that can be taken, please refer to *What should I do if I can't connect to*
	WebLinkHH? in *Q&A – Section 8* of the *V460-H Industrial Handheld DPM Reader User Manual*
	(Cat. No. Z461).

8	The WebLinkHH screen appears.	
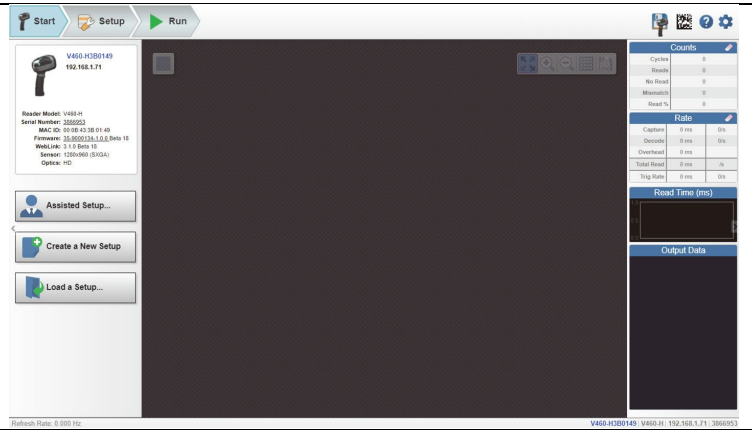
9	Click on the gear icon on the upper right of the screen and select **Advanced**.	
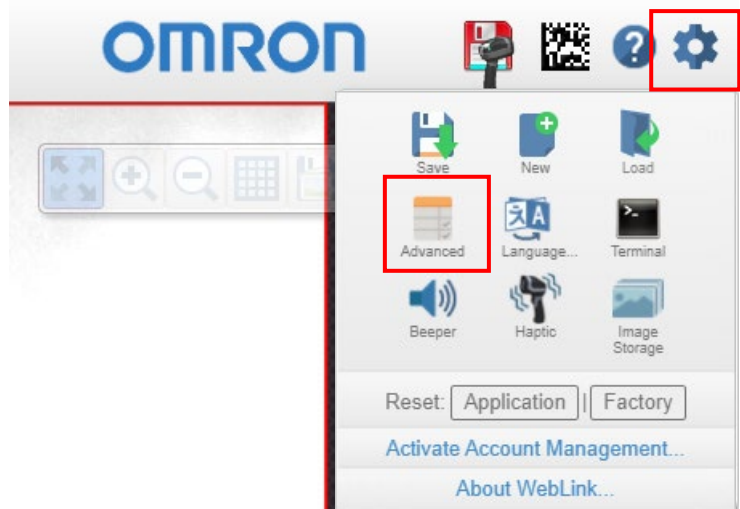
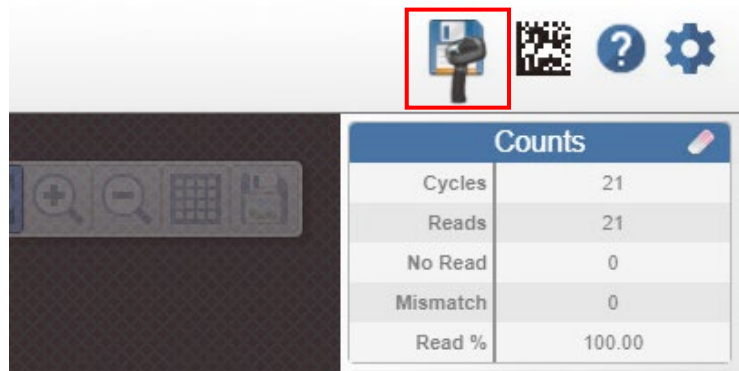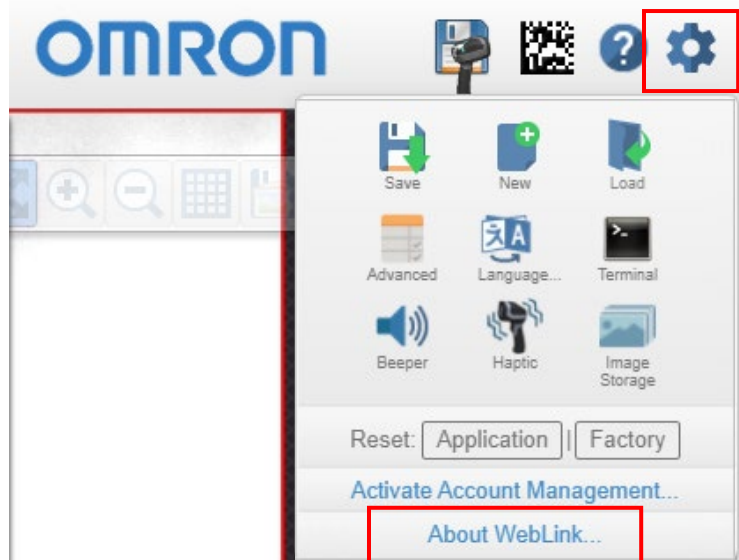|		
10	The Advanced Settings Screen appears. Select the **Communications** Tab and check the settings shown in the red frame for **Ethernet**. To use the defaults, you do not need to change the settings. If you need to change the IP address, for example when connecting multiple code readers, change the IP Address setting as necessary.	
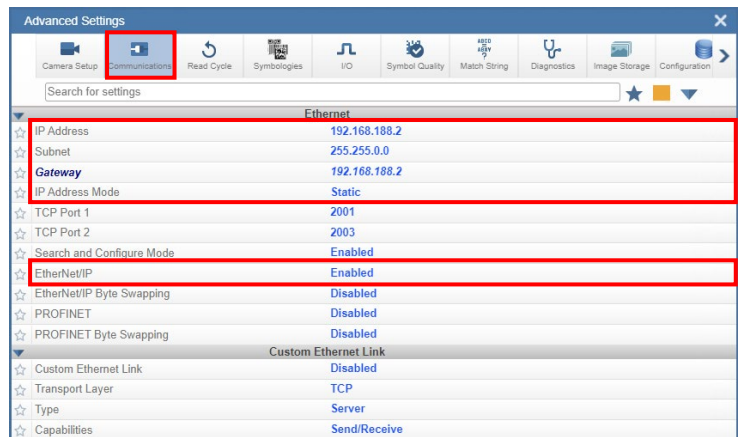
11	Click on the icon shown in the red frame to save the settings to the code reader.	
12	Finally, check the version number of the code reader. Click on the gear icon on the upper right of the screen and select **About WebLink**.	
13	You can check the current version of the code reader in **About WebLink**HH. Update the code reader to the latest version as needed.	

7.3. PLC Setup

Set up the PLC.

7.3.1. Setting the Hardware

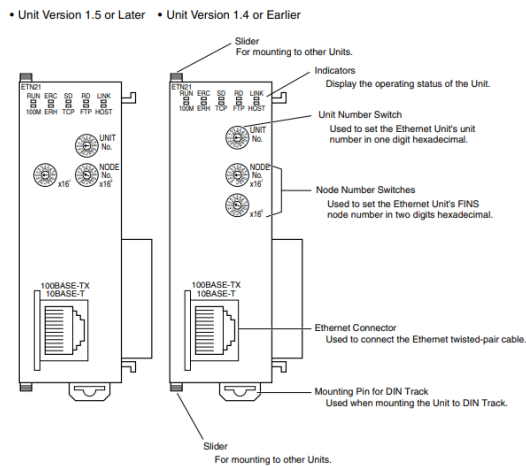Set the physical switches on the Ethernet Unit.

Precautions for Correct Use

Turn OFF the power supply before setting the hardware.

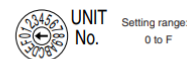|	
1	Confirm that the power supply to the PLC is OFF. * If the power supply is ON, you may not be able to proceed with the subsequent step of the procedure.	
2	Check the position of the physical switches on the front of the Ethernet Unit as shown in the figure on the right.	
3	Set the Unit No. switch to *0*.	
4	Set the Node address switches to the default values as follows. NODE No. x16^1: *0* NODE No. x16^0: *1* * Set the IP Address to *192.168.188.1*. * By default, the first three octets are fixed to "192.168.188". The value set by the Node address switches determines the fourth octet of the node's IP address.	

15

7.3.2. Loading the Ladder Program and Going Online with the PLC

Start the programming tool "CX-Programmer", load the ladder program, and go online with the PLC.

Install the Tool Software and USB driver on the PC beforehand.

Obtain the "ladder program" to use from the OMRON website.

1	Connect the PC and the PLC with a USB cable, and turn ON the power supply to the PLC.	
2	Start the CX-Programmer.	
3	Select **Open** from the **File** Menu.	
4	Select the CX-Programmer Project File of the version specified in *5.2. Device Configuration* (**OMRON_V460_CJ_ETN (TCP)_V100.cxp**) and click **Open**. * Please consult your OMRON representative and obtain the CX-Programmer Project File (Ladder Program) to use.	
5	After the loading of the ladder program is completed, select **Programs** in the project workspace.	 (Project Workspace)
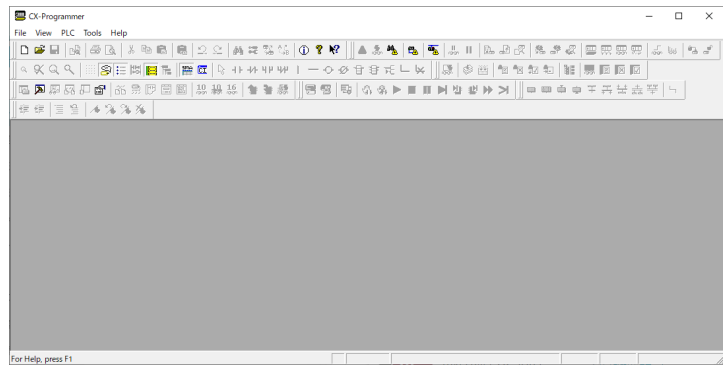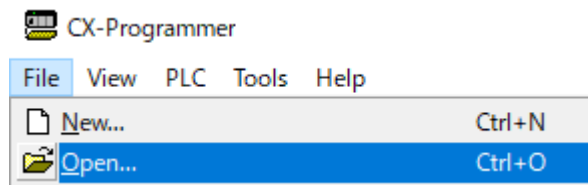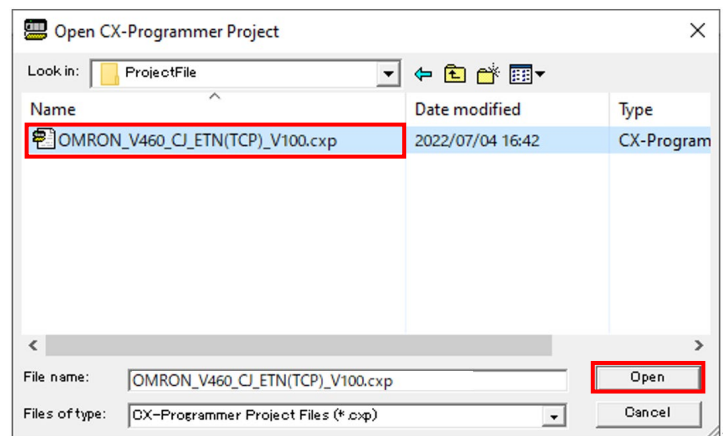
16
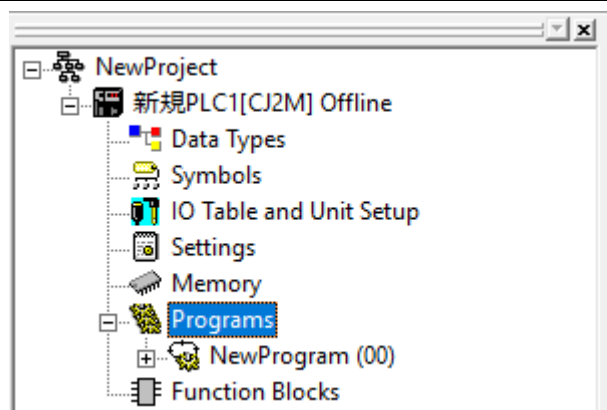
6	Select **Change Model** from the **PLC** Menu.	
7	The **Change PLC** Dialog Box is displayed. Select the PLC type (**CJ2M** in the figure on the right) from the **Device Type** pull-down menu, and click **Settings**.	
8	The **Device Type Settings** Dialog Box is displayed. Select the CPU type (**CPU11** in the figure on the right) from the **CPU Type** pull-down menu, and click **OK**.	
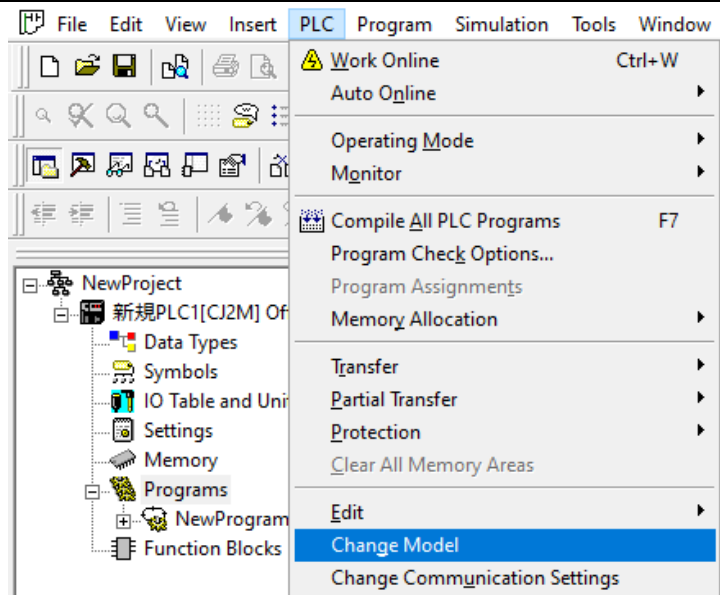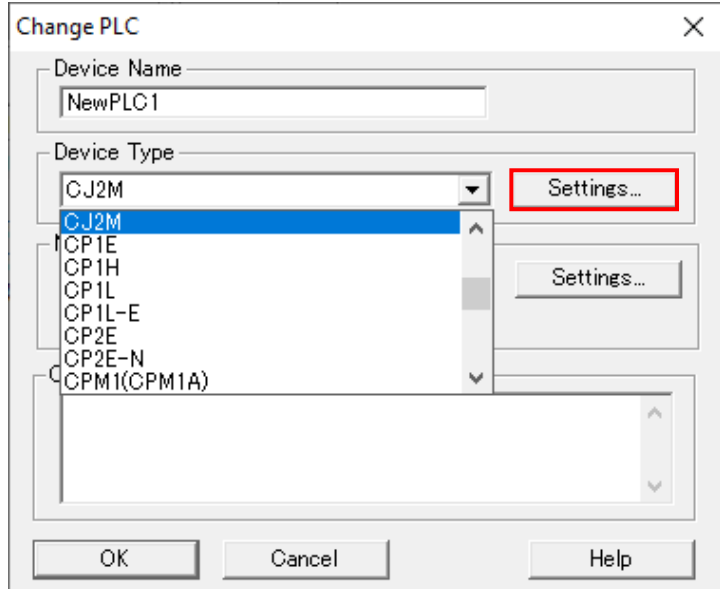
17
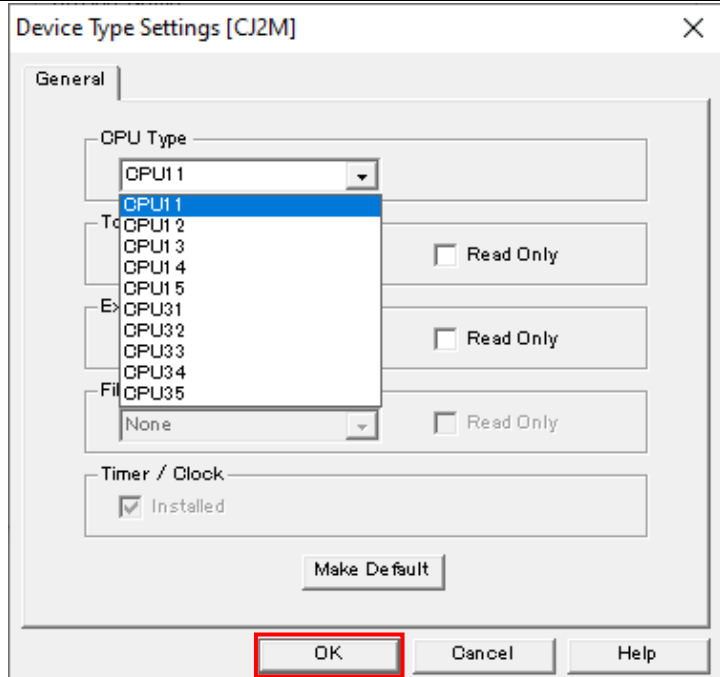
9	In the **Change PLC** Dialog Box, select the network type (**USB** in the figure on the right) from the **Network Type** pull-down menu, and click **OK**.	
	* If you change the PLC type in step 7 or CPU type in step. 8, a confirmation dialog box as shown in the figure on the right will appear. Click **Yes**. Then, confirm that the program has been correctly converted. (Although the figure on the right shows **Duplicated output** warnings, there is no particular problem.)
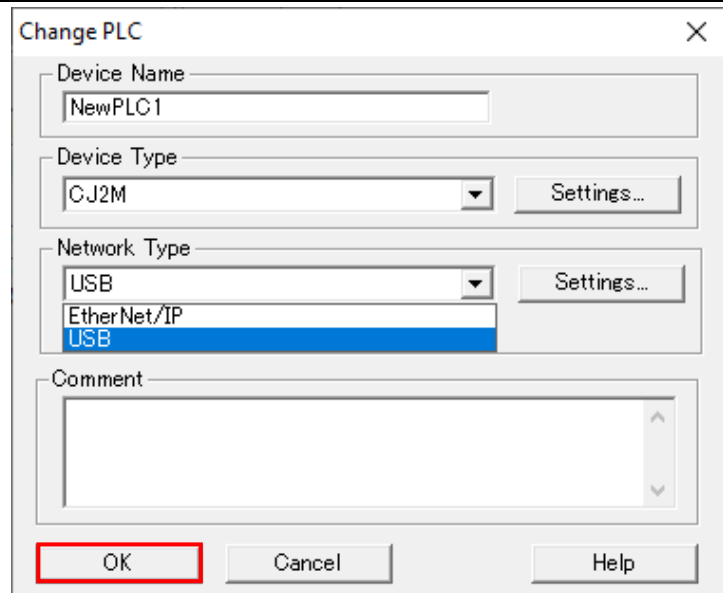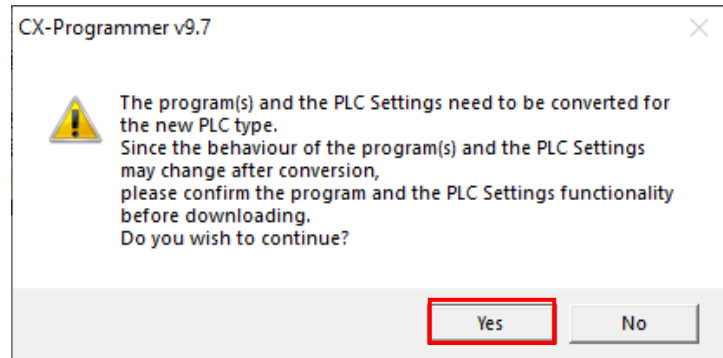
* If you change the device type to **CJ1**, a timer-related dialog box appears. Make a selection according to your operating environment. | |
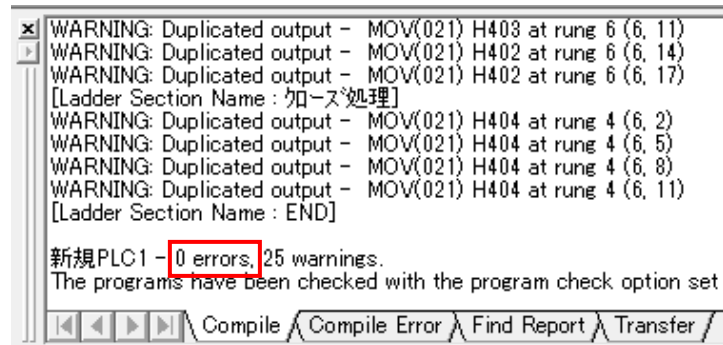
10	Select **Programs** in the project workspace, and select **Work Online** from the **PLC** Menu.	
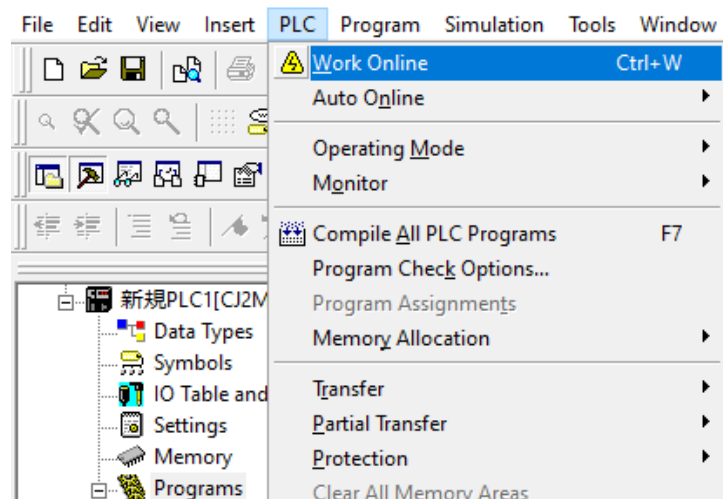
11	A dialog box as shown in the figure on the right appears. Click **Yes**.	
12	Confirm that the CX-Programmer is online with the PLC. * The CX-Programmer is online if the ⚠ icon appears to be depressed.	
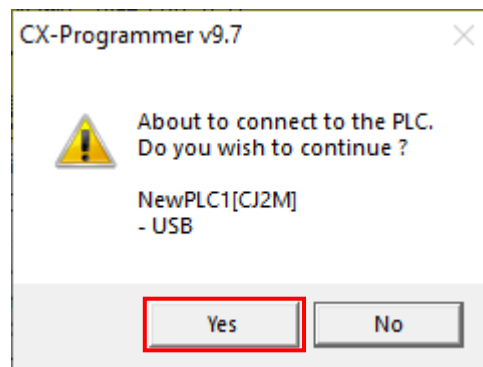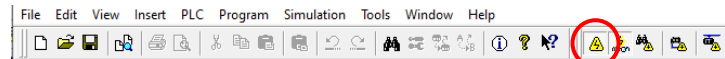
📝 **Note**

If you cannot go online with the PLC, check the physical cable connections, etc. If the physical cable connections are correct, return to step 6 and check the device type and other settings in steps 7 to 9.

For details, refer to the *CX-Programmer Operation Manual* (Cat. No. W446).

📝 **Note**

Some of the dialog boxes shown in this document may not be displayed depending on the environment settings of the CX-Programmer. For details on the environment settings of the CX-Programmer, refer to the *CX-Programmer Operation Manual* (Cat. No. W446).

This document assumes that the check box for **Confirm all operations affecting the PLC** is selected.

7.3.3. Creating the I/O Table

Create the I/O table for the PLC.

1	If the **PLC Operating Mode** is set to either **Run** or **Monitor**, follow steps (1) to (3) to change it to Program. (1) In the CX-Programmer, select **Operating Mode – Program** from the **PLC** Menu. (2) A dialog box as shown in the figure on the right appears. Click **Yes**. * For information on how the dialog box is displayed, refer to *Note* on the previous page. (3) Confirm that **Program Mode** is displayed to the right of the PLC model in the CX-Programmer's project tree. (See the figure on the right.)	
2	In the CX-Programmer, select **Edit – I/O Table and Unit Setup** from the **PLC** Menu.	
	The **PLC IO Table** Window is displayed.	
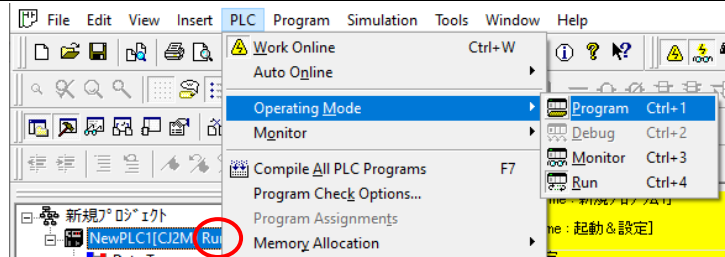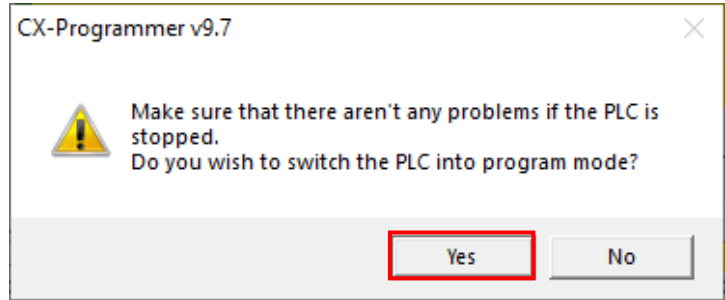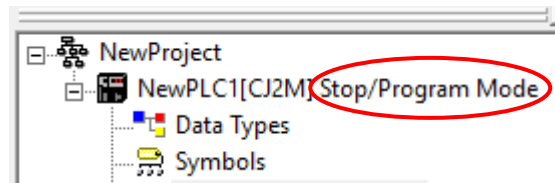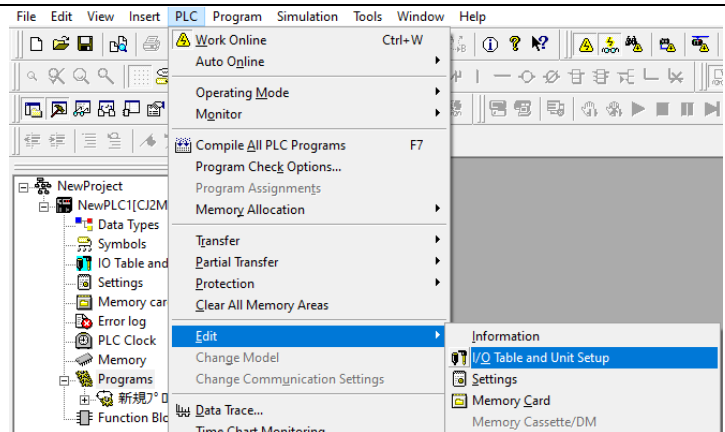
20
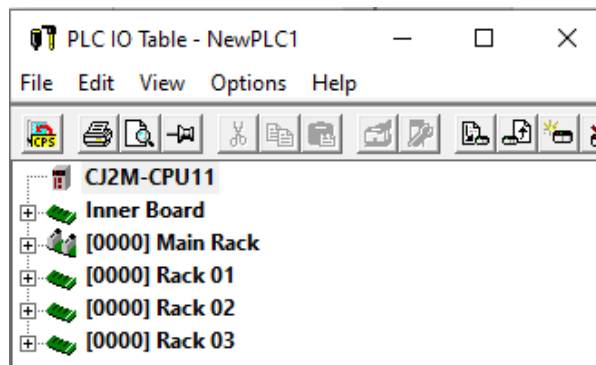
3	In the **PLC IO Table** Window, select **Create** from the **Options** Menu.	
	A dialog box as shown in the figure on the right appears. Click **Yes**.	
	A dialog box as shown in the figure on the right appears. Click **Yes**.	
4	The **Transfer from PLC** Dialog Box is displayed. Select the check boxes for **IO Table** and **SIO Unit Parameters** and click **Transfer**.	
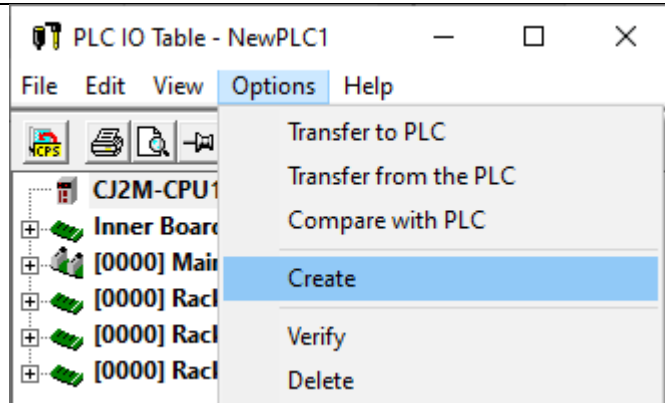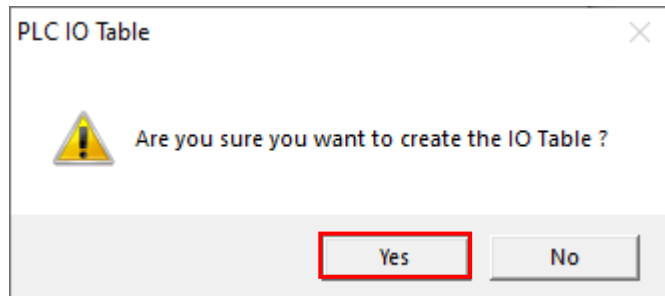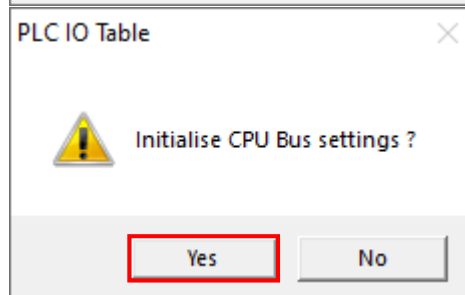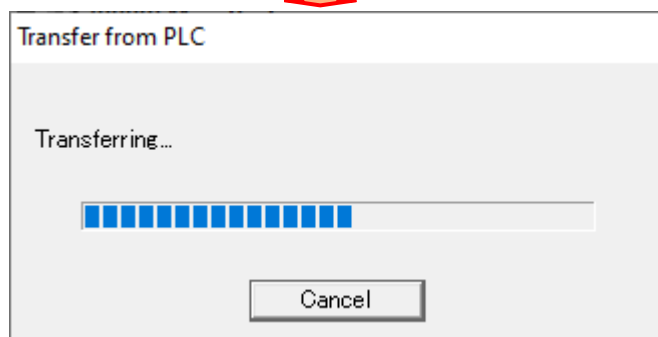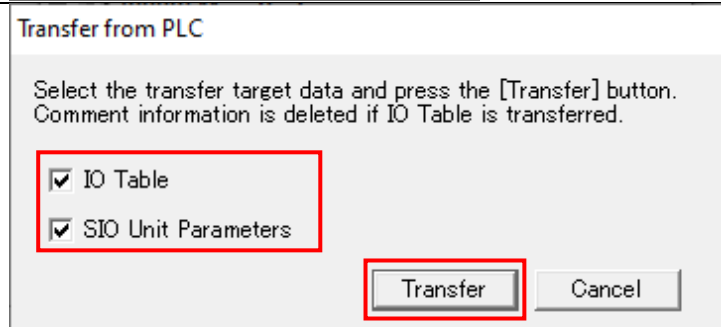
	When the transfer is complete, the **Transfer Results** Dialog Box appears. Check the messages in this dialog box to confirm that the transfer process is successfully completed. The figure on the right shows **Transfer Success: 1 Unit** and **Transfer Unsuccessful: 0 Unit**, which means I/O table creation is successfully completed. Click **OK**.	Transfer Results <IO Table> Transfer Success <Special Units settings> CPU Bus Unit00: Transfer Success Transfer Success:1 Unit Transfer Unsuccessful:0 Unit OK

7.3.4. Editing the Parameters

Set the parameters for the Ethernet Unit.

1	In the **PLC IO Table** Window, double-click **[0000] Main Rack** to open the tree.	PLC IO Table - NewPLC1 — □ ×
		File Edit View Options Help
		
		CJ2M-CPU11
		Inner Board
		[0000] Main Rack
		    00 [1500] CJ1W-ETN21(ETN21Mode)(Ethernet Unit)
		    01 [0000] Empty Slot
		    02 [0000] Empty Slot

2	Right-click on **00 [1500] CJ1W-ETN21(ETN21Mode)** and select **Unit Setup** from the menu.	CJ2M-CPU11
		Inner Board
		[0000] Main Rack
		    00 [1500] CJ1W-ETN21(ETN21Mode)(Ethernet Unit)
		    01 [0000] Add Unit
		    02 [0000] Change / Confirm Units
		    03 [0000] Change Unit No
		    04 [0000] Unit Comment
		    05 [0000] SYSMAC BUS Master
		    06 [0000]
		    07 [0000] Unit Setup
		    08 [0000] Save Parameters
		    09 [0000]

3	The **Edit Parameters** Dialog Box is displayed. Select the **Setting** Tab Page, and enter *192.168.188.1* in the **IP Address** field and *255.255.0.0* in the **Sub-net Mask** field. After entering the above values, click **Transfer [PC to Unit]**.	
4	A confirmation dialog box as shown in the figure on the right appears. Confirm that there is no problem and click **Yes**.	
	Confirm that **Transfer successful** is displayed and click **Close**.	
5	A confirmation dialog box as shown in the figure on the right appears. Read the information and click **Yes**.	
	After you restart the unit, another dialog box as shown in the figure on the right appears. Read the information and click **OK**.	

23

6	Click **Compare** to confirm that the IP address is correctly changed.	
7	Confirm that **Compare successful** is displayed and click **Close**.	
8	Click **OK** in the **Edit Parameters** Dialog Box.	

24

7.3.5. Transferring the Ladder Program

Transfer the ladder program to the CPU Unit.

1	In the CX-Programmer, select **Programs** in the project workspace, and select **Transfer – To PLC** from the **PLC** Menu.	
2	Select the check boxes for **Program(s)**, **Comments**, and **Program index**, and click the **OK** Button. * Do not select **IO table** and **Special Unit Setup** since you have set these data in 7.3.3. and 7.3.4. * **Comments** and **Program index** may not be shown depending on the PLC model. If so, transfer the project data with only **Program(s)** selected.	
3	A dialog box as shown in the figure on the right appears. Click **Yes**.	

4	If the PLC is not in PROGRAM Mode, a dialog box as shown in the figure on the right appears. Click **Yes**.	CX-Programmer v9.7 × ⚠ Make sure that there aren't any problems if the PLC is stopped. Do you wish to switch the PLC into program mode? [Yes] [No]
5	Confirm that the download is successfully completed (**Download successful** is displayed) as shown on the right, and then click **OK**.	Download × Program Download to PLC NewPLC1 Download successful ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ [OK]
6	Select **Programs** in the project workspace, and select **Transfer – Compare with PLC** from the **PLC** Menu.	File Edit View Insert PLC Program Simulation Tools Window Help Work Online Ctrl+W Auto Online Operating Mode Monitor Compile All PLC Programs F7 Program Check Options... Program Assignments Memory Allocation Transfer ▸ To PLC... Ctrl+T Partial Transfer ▸ From PLC... Ctrl+Shift+T Protection ▸ Compare with PLC... Clear All Memory Areas To File... Edit ▸ From File... Change Model Change Communication Settings NewProject NewPLC1[CJ2M Data Types Symbols IO Table and Settings Memory car Error log PLC Clock Memory Programs
7	Select the check box for **Program(s)**, and click the **OK** Button.	Compare Options × PLC: NewPLC1 Include: ☑ Program(s) [OK] [Cancel]
8	Confirm that **Compare successful** is displayed as shown on the right, and click the **OK** Button.	CX-Programmer v9.7 × ⓘ Compare successful [OK]

7.4. Checking the Connection Status

Execute the transferred ladder program to check that Ethernet communications work correctly.

Precautions for Correct Use

Before performing the following steps, confirm that the LAN cable is connected securely.
If it is not connected, first turn OFF the power supply to the device and then connect the LAN cable.

7.4.1. Executing the Ladder Program and Checking the I/O Memory Data

Execute the ladder program and, in the CX-Programmer, check that the correct data is written to the I/O memory of the PLC.

Precautions for Safe Use

Confirm the system safety before you execute the ladder program.
The connected devices may malfunction regardless of the operating mode of the unit, resulting in injury.

1	This document uses the 2D code shown in the right figure as an example of reading. Set the code reader to the position where it can read the 2D code in the right figure.	
2	In the CX-Programmer, open the tree structure under **Programs** in the project workspace, and double-click **Start and Setup**. In the Ladder Window, the ladder diagram for **Start and Setup** is displayed.	
3	Select **Operating Mode – Monitor** from the **PLC** Menu.	
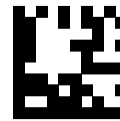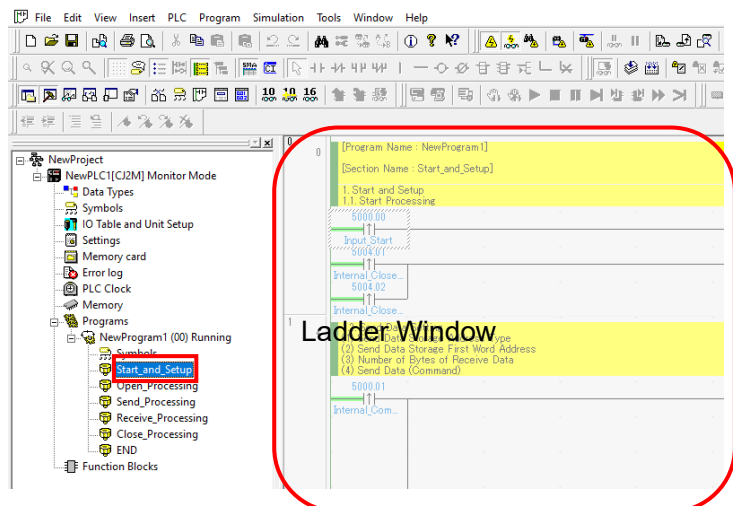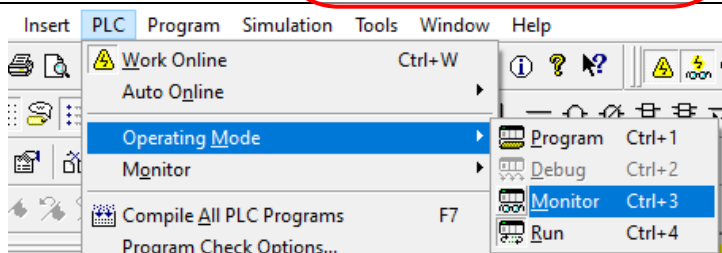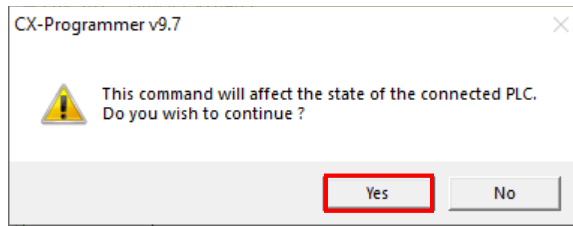
4	A dialog box as shown in the figure on the right appears. Click **Yes**.	
5	Confirm that the operating mode indication has changed to **Monitor Mode**.	
6	In the Ladder Window, right-click **Input_Start** in Block 0 and select **Set – On** from the menu.	
7	Confirm that the *Input_Start* contact is ON.	
8	Press the trigger button on the code reader within 15 seconds after the *Input_Start* is turned ON to read the code. This program will end with an error after a certain period of time because the communication processing branches to the error end.	
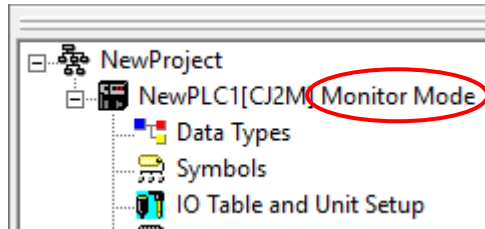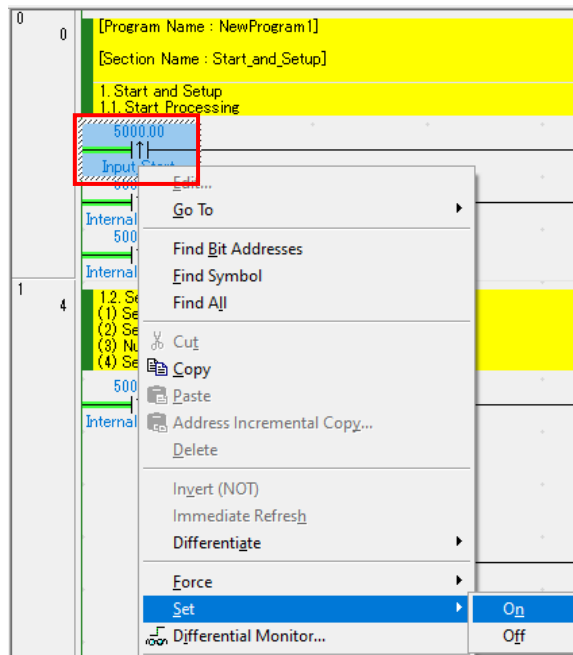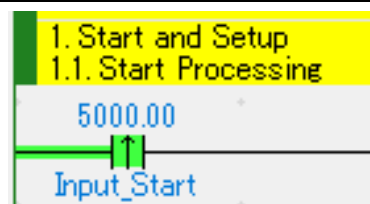
9	Select **Edit – Memory** from the **PLC** Menu.	
10	In the PLC Memory Window displayed, double-click **D** from the list in the left pane of the PLC Memory Window.	 PLC Memory Window
11	In the **D** Window displayed, enter *10000* in the **Start Address** field. Confirm that the first word is changed to *D10000*.	
12	Select **Monitor** from the **Online** Menu.	
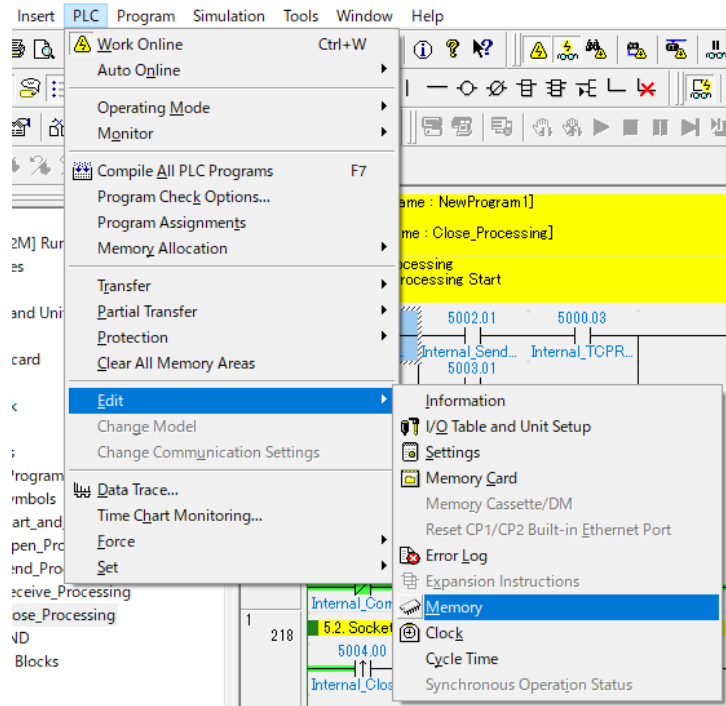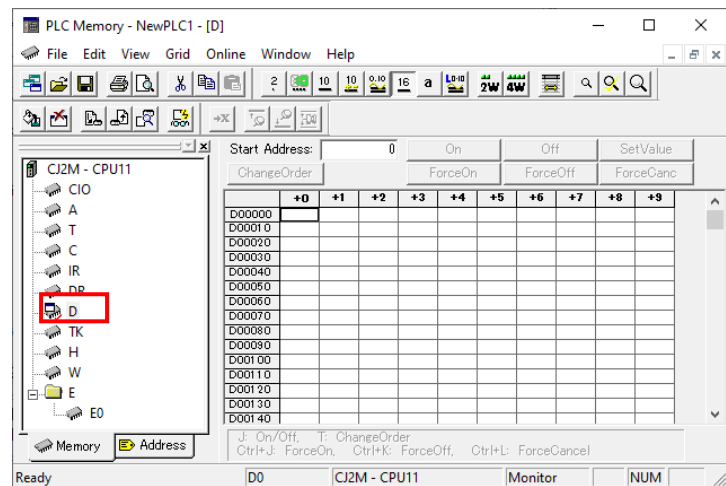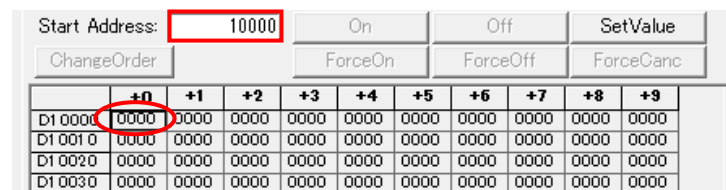
29
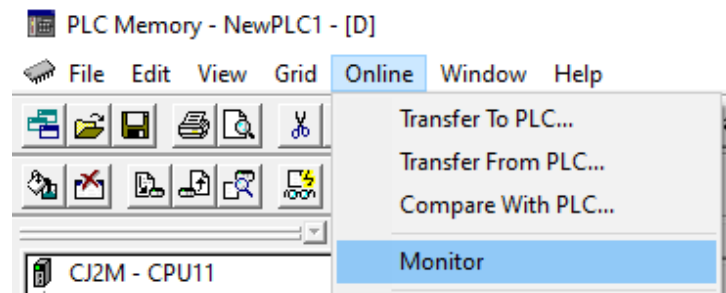
13	The **Monitor Memory Areas** Dialog Box is displayed. Select the check box next to **D** and click **Monitor**.	
14	Check the received data in **D** Window shown in the figure on the right. * In word **D10000**, the number of bytes of receive data is stored. It is *0013* in hex (*19* in decimal), which is equivalent to 9.5 words. This means that the received data is stored in words **D10001** to **D10009** and the upper bytes of word **D10010**. Select **Display** – **Text** from the **View** Menu. The receive data is now displayed in ASCII text as shown in the figure on the right, indicating that the communications have ended normally.	 ASCII text display
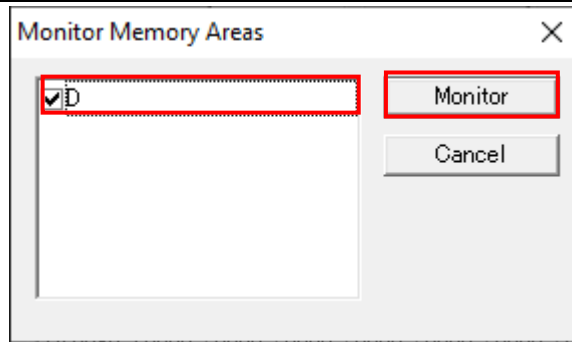
8. Initializing the System

In this document, it is assumed that the Ethernet Unit and the code reader use the factory default settings.
If you change their settings from the defaults, you may not be able to perform various setting procedures as described.

8.1. Ethernet Unit

To initialize the Ethernet Unit, open the **PLC I/O Table** Window in the CX-Programmer. Then, right-click on the Ethernet Unit **00 [1500] CJ1W-ETN21** and select **Unit Setup** from the menu as described in step 2 of the procedure in *7.3.4. Editing Parameters*.
The **Edit Parameters** Dialog Box is displayed. Click **Set Defaults** to proceed.

8.2. Code Reader

For information on initializing the code reader, please refer to *How to initialize the settings?* in *8-3 Q&A* of the *V460-H Industrial Handheld DPM Reader User Manual* (Cat. No. Z461).

9. Software Part

9.1. Overview

This section describes the specifications and functions of the software part used for connecting a code reader (V460-H Series) to a PLC (Ethernet Unit).

The software part refers to a ladder program for the PLC.
The software part performs TCP socket communications by manipulating dedicated control bits (using the socket service function of the Ethernet Unit) to get the software version from the code reader and judge whether the TCP socket communications have ended normally or abnormally.
In the software part, "normal end" means that TCP socket communications have ended normally.
"Error end" means that TCP socket communications have ended abnormally, or that an error has occurred in the code reader (judged by response data from the code reader).

In the software part, timers are executed in BCD mode. It is assumed they operate with the default settings of OMRON CJ1 Series PLCs (CPU Units).

The software part does not use the keep-alive and linger functions (TCP socket options) since their use is determined individually at the time of building the system.

In this section, if it is necessary to distinguish between decimal data and hexadecimal data, add "&" to the beginning of decimal data and "#" to the beginning of hexadecimal data.
(Example: "&1000" for decimal data, "#03E8" for hexadecimal data)

📓 **Note**

We have verified in our test configuration that the software part enables communications for the product versions and product lot used for evaluation.
However, we do not guarantee its operations where there are electrical noise or other disturbances, or variations in the performance of the devices themselves.

📓 **Note**

Contact OMRON for information on how to obtain the software part.

9.1.1. Communications Data Flow

This is the flow from issuing a TCP socket communications command from the PLC (Ethernet Unit) to the code reader and receiving response data from the code reader. The software part executes a processing sequence of TCP open to TCP close in a continuous manner. If response data is divided and arrives as multiple pieces of receive data, receive processing will be repeated. Also, there are cases where there is no response data depending on the code reader or send command. In such as case, receive processing can be skipped by setting the receive processing required/not required setting to "receive processing not required" in advance.

* If the receive processing required/not required setting is set to "receive processing required", in send processing, the program waits for the arrival of receive data before going to receive processing.
If the setting is "receive processing not required", the program goes to close processing immediately after send processing.

1.	TCP Open Processing	The Ethernet Unit issues a TCP open request to the code reader to establish a TCP connection.
	▼	
2.	Command Send Processing * If "receive processing required" is set, the Ethernet Unit waits for response data to arrive.	The Ethernet Unit issues a send message that is set in the ladder program to the code reader and receives response data.
	▼	
3.	Response Receive Processing * If response data is divided and arrives in pieces, the Ethernet Unit repeats receive processing. * If "receive processing not required" is set, the Ethernet Unit skips receive processing.	The Ethernet Unit stores the response data received from the code reader in the internal memory of the specified CPU Unit.
	▼	
4.	Close Processing	The Ethernet Unit issues a close request to the code reader to close the TCP connection.

9.1.2. TCP Socket Communications by Manipulating Dedicated Control Bits

This section provides an overview of TCP socket communications by manipulating dedicated control bits and the general movement of send and receive messages.

📝 **Note**

For details, refer to *Section 6 Socket Services* in the *CS/CJ-series Ethernet Units Operation Manual Construction of Applications* (Cat. No. W421).

● Socket Services by Manipulating Dedicated Control Bits

To use socket services by manipulating dedicated control bits, store the required parameters in Socket Service Parameter Areas in the CPU Bus Unit Area and then turn ON socket service request switches.

Socket Service Parameter Areas (Allocated DM Area)

The socket service parameters used for socket service requests are allocated in the DM Area as shown below. (The first word m is calculated as m = D30000 + (100 x Unit number).)
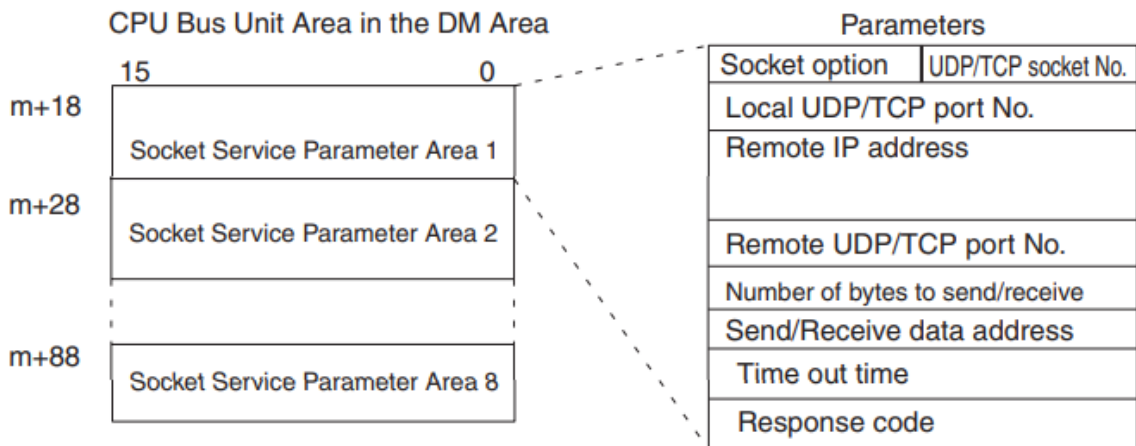
The following description uses Socket Service Parameter Area 1 as an example.

(1) m+18: Socket option and TCP socket number
- To use the keep-alive function, set bit 08 to *1* (ON).
- To use the linger function, set bit 09 to *1* (ON).
- Specify the TCP Socket number to use as *&1* to *&8* in bits 00 to 07.
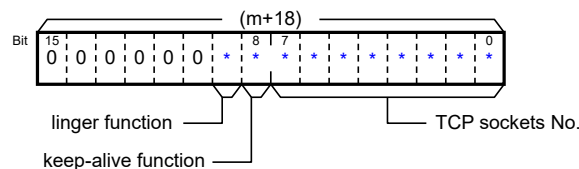
(2) m+19: Local TCP port number
- Specify the TCP port number used by the socket to send and receive data. Normally, specify *1024* or higher. When *0* is specified, an unused TCP port number is automatically assigned.

(3) m+20 and m+21: Remote IP address

- Specify the IP address of the remote node.

  The first and second octets of the remote IP address are stored in m+20. The third and fourth octets of the remote IP address are stored in m+21.
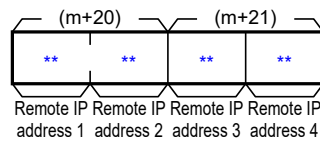
(4) m+22: Remote TCP port number

- Specify the TCP port number of the remote node.

(5) m+23: Number of bytes of send/receive data

- Specify the number of bytes of send data or receive data for a send request or receive request.

(6) m+24 and m+25: Send/Receive data address

- Specify the first word of the source that sends send data for a send request, or the first word of the destination that receives receive data for a reception request.
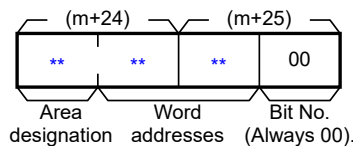
(7) m+26: Timeout

- Specify the time limit in increments of 0.1 s within which the Socket Service Request Switch is turned ON and then OFF (reception is complete) for a receive request.

  When *0* is specified, timeout monitoring for the receive request will not be performed.

(8) m+27: Response code

- The execution result of an open request, send request, receive request, or close request will be stored as a response code.

Socket Service Request Switches (CIO Area)

To issue socket service requests by manipulating dedicated control bits, manipulate Socket Service Request Switches. Socket Service Request Switches are allocated to the CIO Area words for each socket number as shown below.

(The first word n is calculated as n = 1500 + (25 x Unit number).)
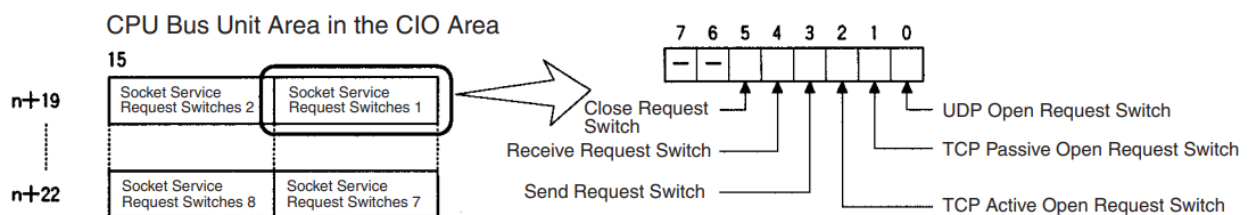
- Send and Receive Messages
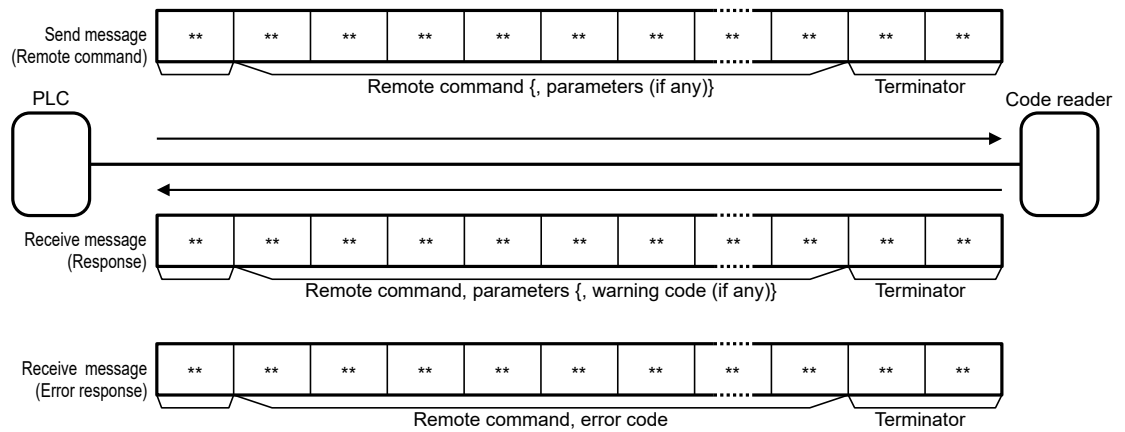
- Communications Sequence

  The figure below shows the processing flow of TCP communications between the code reader (server) and the PLC (client).
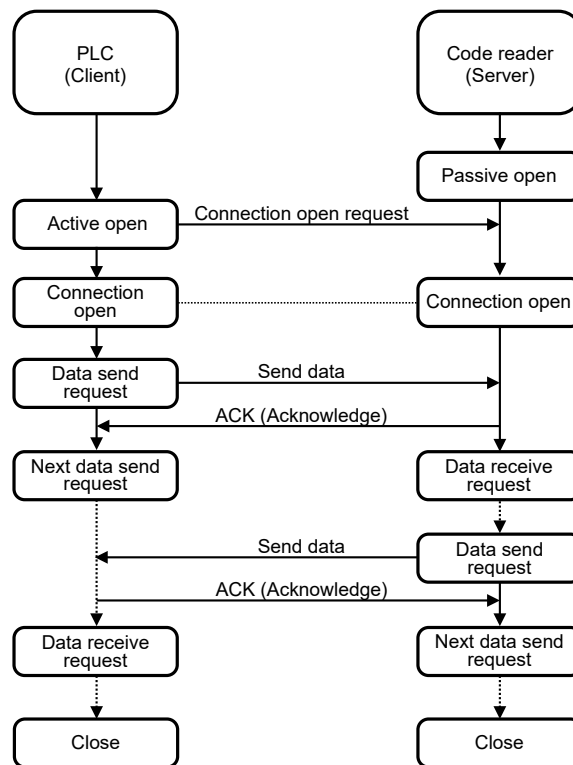
9.2. Error Judgment Processing

This section describes error judgment processing by the software part.

9.2.1. Error Judgment by the Software Part

This software part performs error judgment processing for the following four types of errors (1) to (4). Refer to *9.6. Error Code List* for information on error codes.
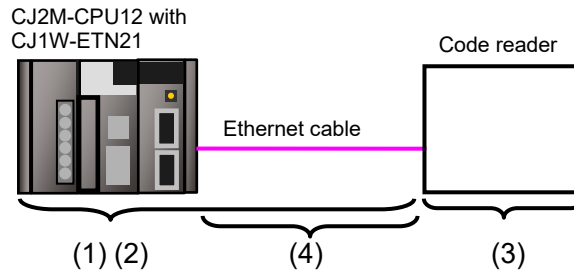
(1) Communications error during TCP socket communications by manipulating dedicated control bits

An error that occurred in TCP socket communications, such as main unit error, command format error, or parameter error, is judged as a "communications error". The judgment is made based on the response codes in the allocated DM Area for TCP socket communications by manipulating dedicated control bits.

(2) Timeout error during communications with the code reader

An error that occurred due to abnormal open, send, receive, or close processing that failed to complete within the monitoring time is judged as a timeout error. This judgment is made based on timer monitoring in the software part. Refer to *9.2.2. Time Monitoring Function* for information on time monitoring using the internal timers of the software part.

(3) Code reader error

An error such as a command error, parameter error, or inexecutable error on the code scanner is judged as a "code reader error". The judgment is made based on the response data returned from the code reader.

* This error judgment is not supported by V460-H because it does not return a response in the event of an error.

(4) TCP connection status error at end of processing

The software part uses a procedure in which the overall processing ends after the last close processing is done, regardless of whether the open to receive processing steps have ended normally or abnormally. Therefore, judgment of whether close processing has ended normally is made according to the TCP Connection Status words in the allocated DM Area. If there is an error in close processing, the next open processing may not be performed correctly. Refer to *9.2.3. TCP Connection Status Error Situation and Correction* for information on how to correct a TCP connection status error.

9.2.2. Time Monitoring Function

This section describes the time monitoring function of the software part.

● Time Monitoring Using Internal Timers of the Software Part
Assuming that processing has the executing status and does not end due to an error, the software part uses its internal timers to interrupted the processing (i.e., timeout). The timeout is set to *15* s for each processing phase from open to close.

Time Monitoring Using Internal Timers of the Software Part

Processing	Monitoring description	Timeout
Open processing	Time from start to end of open processing	After 15 s
Send processing	Time from start to end of send processing * When "receive processing required" is set, the software part checks the arrival of the first receive data to judge that the end of processing.	After 15 s
Receive processing	Time from start to end of receive processing * If receive processing is repeated, the software part monitors the time for each repetition of receive processing.	After 15 s
Close processing	Time from start to end of close processing * The software part checks that the TCP connection status is normal after close processing to judge the end of the processing.	After 15 s

● Time Monitoring Using the Ethernet Unit (Socket Services)
The Ethernet Unit has the time monitoring function as a socket service, which monitors the arrival of receive data. This is set as a socket service parameter for receive processing. In the software part, the parameter is named **Receive Standby Time**, which is set to *300* ms (default). If receive data does not arrive from the code reader within this time, it judges that the receive processing has ended.

📝 **Note**

For information on time monitoring using socket services, refer to *6-7 Using Socket Services by Manipulating Dedicated Control Bits* in the *CS/CJ-series Ethernet Units Operation Manual Construction of Applications* (Cat. No. W421).

● Resending and Time Monitoring Using the Ethernet Unit (TCP/IP)
If a communications error occurs, TCP/IP automatically resends data and monitors the processing time if there is no problem with the Ethernet Unit. If processing abnormally ends in the middle of it, the software part stops the resending and time monitoring via TCP/IP in the close processing. However, if the close processing shows a TCP connection status error, the resending and time monitoring via TCP/IP may continue to be active in the Ethernet Unit. Refer to *9.2.3. TCP Connection Status Error Situation and Correction* for information on the error situation and correction.

Resending and Time Monitoring via TCP/IP
* The number of seconds indicates the time elapsed since the first request after the occurrence of the error.

Processing	First resend	Times of resend	Last resend	Last timeout
Open request (TCP active)	After approx. 5 s	3	After approx. 41 s	After approx. 75 s
Send request	Within 1 s	12	After approx. 446 s	After approx. 510 s
Receive request	Resending and time monitoring via TCP/IP not supported			
Close request	Within 1 s	12	After approx. 446 s	After approx. 510 s

9.2.3. TCP Connection Status Error Situation and Correction

This section describes the situation and corrections if a TCP connection status error occurs.

● Effect of a TCP Connection Status Error

If, after the occurrence of a TCP connection status error, you execute the software part again without taking any corrective action or without noticing the error, a state of "code reader not passive open" (hereinafter referred to as "open processing error") can occur. This is considered as the effect of the TCP connection status error at the end of the previous communications processing. You can determine what error occurred in the Error Code Storage Area.

Error Code Storage Area

Address: Stored content	Error code: Error description
H400: Code indicating the end status of open processing	004A: Code reader not passive open
H404: Code indicating the end status of close processing	F402: TCP connection status error

● Situation When a TCP Connection Status Error Occurs

Both a TCP connection status error after close processing and an open processing error in the next communications processing due to the effect of the TCP connection status error can occur because the close processing has not completed in the code reader. In this situation, despite that the Ethernet Unit has ended all processing steps (up to close processing) in the software part, it has not received the close completion notification from the code reader (i.e., the completion of the close processing in the code reader is not confirmed).

● Correction

Check whether the communications port of the code reader is closed since the close processing may not be completed in the code reader. As a result, if the communications port of the code reader is not closed or its state cannot be confirmed, the communications port must be reset. To reset the communications port of the code reader, you can use software restart or turn OFF and then ON the power supply. For details, refer to the manual for the code reader.

Precautions for Correct Use

Reset the communication port of the code reader after confirming that it is not connected to another device.

● Situation When a TCP Connection Status Error Occurs in the PLC (Ethernet Unit)

When a TCP connection status error occurs, the software part has ended its processing, but resending and time monitoring by the Ethernet Unit (TCP/IP function) may be active, as described in *9.2.2. Time Monitoring Function*. However, this resending will stop under the following situations, so there is no particular need to consciously stop it.
• The software part is started and an open processing request is issued again.
• A communications problem such as cable disconnection is resolved during resending.
• Resend processing is ended by the TCP/IP time monitoring (timeout) function.
• The Ethernet Unit is restarted or turned OFF.

9.3. Memory Map

This section describes the memory map of the software part.

9.3.1. List of Used Bits

The tables below show bits, words, and timers that are required to execute the software part.

You can change the following allocations to any addresses.

Precautions for Correct Use

When you change the address allocations, be careful not to cause address duplication.

● Input Bits

The following bits are used to operate the software part.

Address	Data type	Variable name	Description
5000.00	BOOL	Input_Start	Turns from OFF to ON to start the software part.
5010	UINT _BCD	Input_OpenMonitorTime _BCD	Sets the monitoring time for open processing in increments of 10 ms. (This is set to #500 (5 s).)
5011	UINT _BCD	Input_SendMonitorTime _BCD	Sets the monitoring time for send processing in increments of 10 ms. (This is set to #500 (5 s).)
5012	UINT _BCD	Input_ReceiveMonitorTime _BCD	Sets the monitoring time for receive processing in increments of 10 ms. (This is set to #500 (5 s).)
5013	UINT _BCD	Input_CloseMonitorTime _BCD	Sets the monitoring time for close processing in increments of 10 ms. (This is set to #500 (5 s).)
5014	UINT	Input_ReceiveStandbyTime _BIN	Sets the arrival standby time for receive data in increments of 100 ms. (This is set to &3 (300 ms).)
5020	UINT	Input_ReceiveProcessing Required/NotRequired	Sets whether or not receive processing is required, taking into account whether the code reader returns a response to the command sent from the PLC. * If receive processing is not required: Set *&0*. The program will skip receive processing and go to close processing without waiting for receive data in send processing. Specify this value when response data is not sent back to the command sent. * If receive processing is required: Set *&1*. The program will wait for the arrival of receive data in send processing. The program will go to receive processing after checking the arrival of receive data. Specify this value when response data is sent back to the command sent.
D9000	UINT	Input_NumberOfSendDataBytes	Sets the number of bytes of send data.
D9001 D9002 to D9128	WORD [128]	Input_SendData[0] Input_SendData[1] to Input_SendData[127]	Send data storage area to set the send command. (An area of 128 words is secured.)

● Output Bits

The following bits reflect the execution results of the software part.

Address	Data type	Variable name	Description
D10000	INT	Output_NumberOfReceiveDataBytes	Stores the number of bytes of receive data.
D10001 D10002 to D12000	WORD [2000]	Output_ReceiveData[0] Output_ReceiveData[1] to Output_ReceiveData[1999]	Stores receive data (response). (An area of 2000 words is secured.)
H400	WORD	Output_OpenErrorCode	Stores the error code for a communications error or timeout error detected in open processing. *#0000* is stored when the processing ends normally.
H401	WORD	Output_SendErrorCode	Stores the error code for a communications error or timeout error detected in send processing. *#0000* is stored when the processing ends normally.
H402	WORD	Output_ReceiveErrorCode	Stores the error code for a communications error or timeout error detected in receive processing. *#0000* is stored when the processing ends normally.
H403	WORD	Output_CodeReaderErrorCode	Stores the error code of a code reader error detected as a result of receive processing. *#0000* is stored when the processing ends normally.
H404	WORD	Output_CloseErrorCode	Stores the error code for a communications error, timeout error, or TCP connection status error detected in close processing. *#0000* is stored when the processing ends normally.

● Work Bits

The following bits are used only for the purpose of calculation by the software part.

Address	Data type	Variable name	Description
5000.01	BOOL	Internal_CommunicationsExecuting	Turns ON when the software part is running and turns OFF when not running.
5000.02	BOOL	Internal_TCPReceiveProcessingRequired	Executes receive processing after send processing when ON.
5000.03	BOOL	Internal_TCPReceiveProcessingNotRequired	Executes close processing by skipping receive processing after send processing when ON.
5000.04	BOOL	Internal_InitialSetupEnd	Turns ON when the initial setup ends.
5001.00	BOOL	Internal_OpenExecuting	Turns ON when open processing is in executing state.
5001.01	BOOL	Internal_OpenNormalEnd	Turns ON when open processing ends normally.
5001.02	BOOL	Internal_OpenErrorEnd	Turns ON when open processing ends abnormally.
5001.03	BOOL	Internal_OpenResponseCodeError	Turns ON when an error is set in the response code in the Socket Service Parameter Area (allocated DM Area) as a result of open processing.
5001.04	BOOL	Internal_OpenTimeout	Turns ON when open processing times out.
5002.00	BOOL	Internal_SendExecuting	Turns ON when send processing is in executing state.
5002.01	BOOL	Internal_SendNormalEnd	Turns ON when send processing ends normally.
5002.02	BOOL	Internal_SendErrorEnd	Turns ON when send processing ends abnormally.

Address	Data type	Variable name	Description
5002.03	BOOL	Internal_SendResponse CodeError	Turns ON when an error is set in the response code in the Socket Service Parameter Area (allocated DM Area) as a result send processing.
5002.04	BOOL	Internal_SendTimeout	Turns ON when send processing times out.
5003.00	BOOL	Internal_ReceiveExecuting	Turns ON when receive processing is in executing state.
5003.01	BOOL	Internal_ReceiveNormalEnd	Turns ON when receive processing ends normally.
5003.02	BOOL	Internal_ReceiveErrorEnd	Turns ON when receive processing ends abnormally.
5003.03	BOOL	Internal_ReceiveResponse CodeError	Turns ON when an error is set in the response code in the Socket Service Parameter Area (allocated DM Area) as a result receive processing.
5003.04	BOOL	Internal_ReceiveTimeout	Turns ON when receive processing times out.
5003.05	BOOL	Internal_ReceiveCode ReaderError	Turns ON when a code reader error is detected as a result of receive processing.
5003.06	BOOL	Internal_ReceiveRepetition ON	Turns ON when repeated receive processing is required.
5003.07	BOOL	Internal_ReceiveRequestON	Used for manipulating dedicated control bits for a receive request. Turns ON and OFF repeatedly when repeated receive processing is required.
5004.00	BOOL	Internal_CloseExecuting	Turns ON when close processing is in executing state.
5004.01	BOOL	Internal_CloseNormalEnd	Turns ON when close processing ends normally.
5004.02	BOOL	Internal_CloseErrorEnd	Turns ON when close processing ends abnormally.
5004.03	BOOL	Internal_CloseResponse CodeError	Turns ON when an error is set in the response code in the Socket Service Parameter Area (allocated DM Area) as a result of lose processing.
5004.04	BOOL	Internal_CloseTimeout	Turns ON when close processing times out.
5004.05	BOOL	Internal_CloseStatusError	Turns ON when a TCP connection status error is set as a result of close processing.
5005	UINT	Internal_TCPConnection Status	Extracts and sets the lower 4 bits of the TCP Connection Status in the allocated DM Area to determine the TCP connection status as a result of close processing.
5030	UINT	Internal_SendDataAddress Type	Sets the address type of the send data storage area. (This is set to #82 (DM memory).)
5031	UINT	Internal_SendDataFirstWord	Sets the first word of the send data storage area. (This is set in D9001 together with &9001 (Address type).)
5110	UINT	Internal_ReceiveData AddressType	Sets the address type stored in receive data storage area. (This is set to #82 (DM memory).)
5111	INT	Internal_ReceiveDataFirst Word	Sets the first word stored in receive data storage area. (This is set in D10001 together with &10001 (Address type).)
5112	INT	Internal_ReceiveDataFirst WordIncrement	Stores the offset value (increment) for the storage address when there are more than one set of receive data.
5113	INT	Internal_ReceiveDataFirst WordIncrementRemaining	Stores the adjustment value (increment) for calculating the receive data storage address if the previous data has an odd number of bytes.
5114	UINT	Internal_ReceiveData JudgmentArea_UINT	This area is used for converting the data type of receive data from WORD to UINT for comparison and judgment.

● Timers

The following timers are used for the software part.

Address	Data type	Variable name	Description
T1000	BOOL	Internal_OpenMonitorTimer ON	Measures the monitoring time for open processing.
T1001	BOOL	Internal_SendMonitorTimer ON	Measures the monitoring time for send processing.
T1002	BOOL	Internal_ReceiveMonitor TimerON	Measures the monitoring time for receive processing.
T1003	BOOL	Internal_CloseMonitorTimer ON	Measures the monitoring time for close processing.

9.3.2.   Fixed Allocated Bits

The tables below show bits that are required to execute the software part.

The following address allocations cannot be arbitrarily changed since they are fixed by the unit address (unit number) and used socket number settings for the Ethernet Unit.

This software part uses Socket No. 0, TCP Socket No. 8, and Socket Service Parameter Area 8.

● CIO Area

Address	Data type	Variable name
1516.13	BOOL	ETN_DataReceived
1522.10	BOOL	ETN_OpenRequest
1522.11	BOOL	ETN_SendRequest
1522.12	BOOL	ETN_ReceiveRequest
1522.13	BOOL	ETN_CloseRequest

● Allocated DM Area

Address	Data type	Variable name
D30008	UINT	ETN_NumberOfTCPReceiveDataBytes
D30016	UINT	ETN_TCPConnectionStatus
D30088	UINT	ETN_SocketNo8
D30089	UINT	ETN_LocalPortNo
D30090	UINT	ETN_RemotePortIPAddress_1
D30091	UINT	ETN_RemotePortIPAddress_2
D30092	UINT	ETN_RemotePortNo
D30093	INT	ETN_NumberOfSend/ReceiveDataBytes
D30094	UINT	ETN_Send/ReceiveDataAddress_1
D30095	UINT	ETN_Send/ReceiveDataAddress_2
D30096	UINT	ETN_TimeoutValue
D30097	UINT	ETN_ResponseCode

🗒 **Note**

For information on the CIO Area and allocated DM Area, refer to *Section 6 Socket Services* in the *CS/CJ-series Ethernet Units Operation Manual Construction of Applications* (Cat. No. W421).

9.4. Ladder Program

9.4.1. Functional Components of Ladder Program

The functional components of the software part are as follows.

Category	Subcategory	Description
1. Start and Setup	1.1. Start Processing 1.2. Send Data Setting 1.3. Control Data Setting 1.4. Common Parameter Setting 1.5. Receive Data Storage Area Setting 1.6. Receive Processing Required/Not Required Flag Setting 1.7. Error Code Storage Area Initialization 1.8. Initialization End Processing	Starts communications processing and initializes command settings, parameter settings, and error code storage area.
2. Open Processing	2.1. Open Processing Start 2.2. Socket Service Parameter Area Setting 2.3. Open Request Switch ON 2.4. Normal/Error Judgment Processing 2.5. Error Code Storage Area Setting	Executes TCP open (active) processing. Processing starts after communications processing is started and initial setup is done.
3. Send Processing	3.1. Send Processing Start 3.2. Socket Service Parameter Area Setting 3.3. Send Request Switch ON 3.4. Normal/Error Judgment Processing 3.5. Error Code Storage Area Setting	Executes send processing. Processing starts if open processing ends normally.
4. Receive Processing	4.1. Receive Processing Start 4.2. Socket Service Parameter Area Setting 4.3. Receive Request ON 4.4. Receive Request Switch ON 4.5. Normal/Error Judgment Processing 4.6. Receive Processing Repetition Information Calculation 4.7. Error Code Storage Area Setting	Starts processing if the Receive Processing Required Flag is set to *Required* and send processing has ended normally. If send data is divided and arrives as multiple pieces, receive processing is repeated.
5. Close Processing	5.1. Close Processing Start 5.2. Socket Service Parameter Area Setting 5.3. Close Request Switch ON 5.4. Normal/Error Judgment Processing 5.5. Error Code Storage Area Setting	Executes close processing. Processing starts in the following cases. • Receive Processing Required Flag is set to *Not required* and send processing has ended normally. • Receive processing ends normally. • Open processing ends abnormally. • Send processing ends abnormally. • Receive processing ends abnormally.

9.4.2. Detailed Description of Functional Components

The software part is shown on the following pages.
To change the communications settings or send data (command) of the code reader, modify the data enclosed in the red frames.

● 1. Start and Setup

```
                                    MOV(021)    Move
                                     #0000      Source word

                                     D9003       Destination
                                   Input_SendD...  Send data


                                    MOV(021)    Move
                                     #0000      Source word

                                     D9004       Destination
                                   Input_SendD...  Send data


                                    MOV(021)    Move
                                     #0000      Source word

                                     D9005       Destination
                                   Input_SendD...  Send data
```
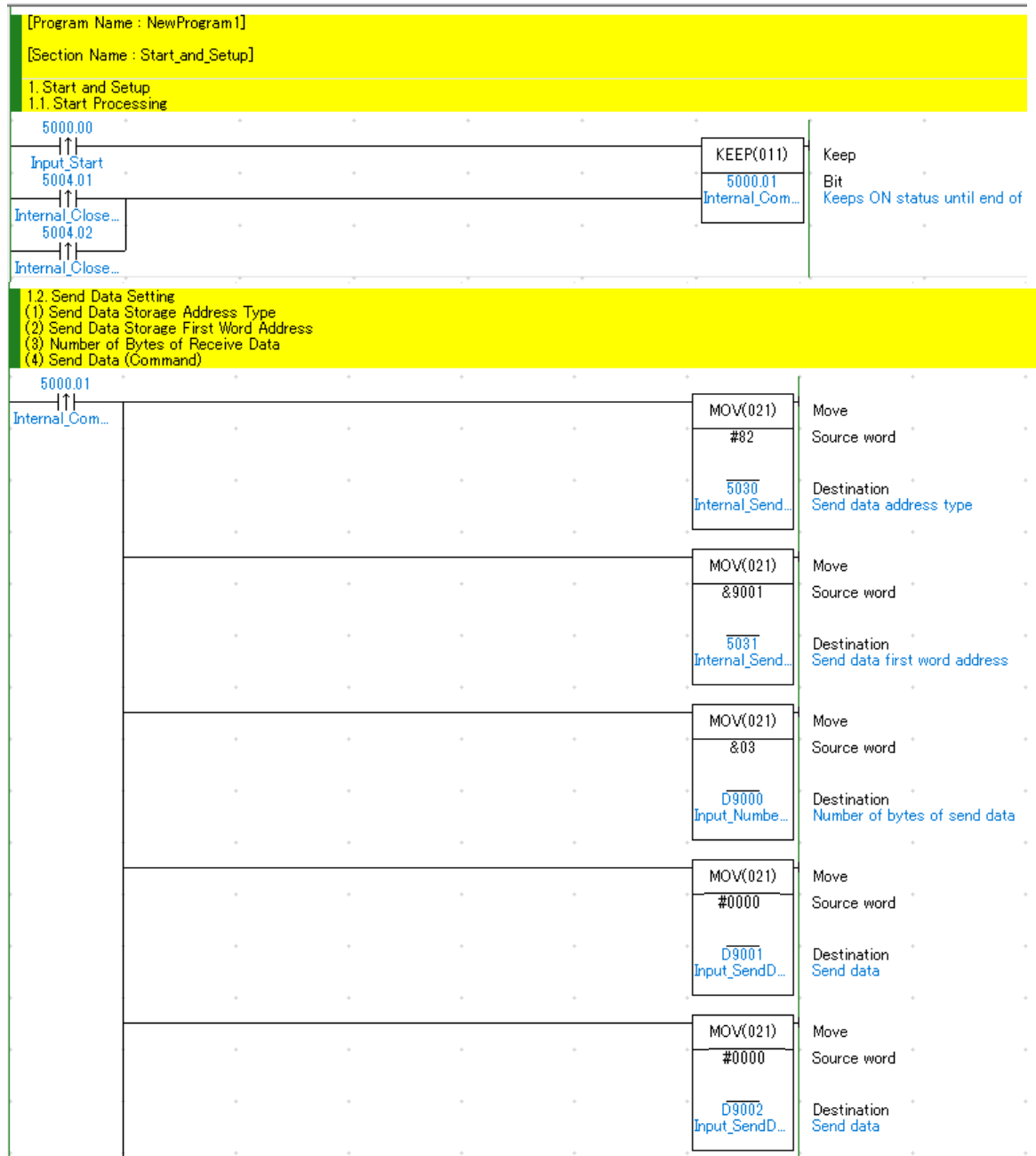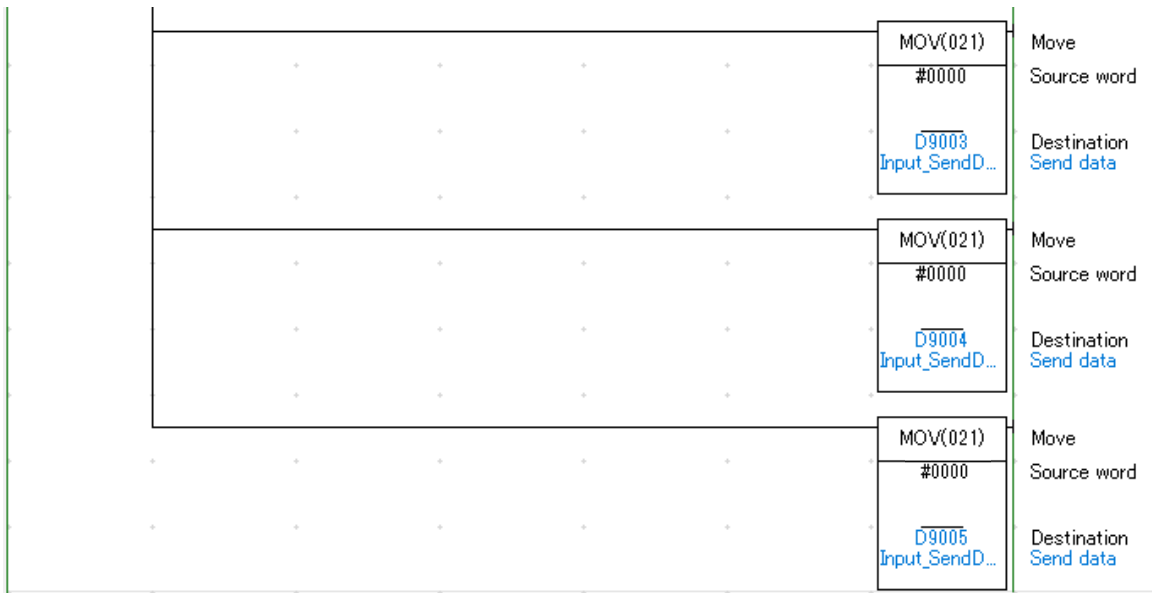
No.	Overview	Description
1.1.	Start Processing	Turns ON the INPUT_START switch to start communications processing. The communications processing ends after the end of close processing.
1.2.	Send Data Setting	Sets the number of bytes of send data and send data (command).

1.3. Control Data Setting
(1) Open Processing Max. Standby Time (Unit: 10ms_BCD)
(2) Send Processing Max. Standby Time (Unit: 10ms_BCD)
(3) Receive Processing Max. Standby Time (Unit: 10ms_BCD)
(4) Close Processing Max. Standby Time (Unit: 10ms_BCD)
(5) Receive Data Arrival Standby Time (Unit: 100ms_BIN)
(6) Receive Processing Required/Not Required Setting for Command Sent (&0: Not Required, &1: Required)

```
 5000.01
  ─┤↑├─
Internal_Com...                      MOV(021)    Move
                                      #1500      Source word

                                      5010       Destination
                                   Input_OpenM...  Open monitoring time (BCD)


                                     MOV(021)    Move
                                      #1500      Source word

                                      5011       Destination
                                   Input_SendM...  Send monitoring time (BCD)


                                     MOV(021)    Move
                                      #1500      Source word

                                      5012       Destination
                                   Input_Receiv...  Receive monitoring time (BCD)


                                     MOV(021)    Move
                                      #1500      Source word

                                      5013       Destination
                                   Input_CloseM...  Close monitoring time (BCD)
```

46

MOV(021)	Move
&3	Source word
5014 Input_Receiv...	Destination Receive Standby Time (BIN)

MOV(021)	Move
&1	Source word
5020 Input_Receiv...	Destination Sets whether or not receive processing is required for send processing.

```
1.4. Common Parameter Setting
(1) Use Socket No. 8
(2) Use Socket Service Parameter Area 8
(3) Local Port No. Automatic Allocation: &0
(4) Remote Node IP Address: 192.168.250.2
(5) Remote Port No.: According to Code Reader Specifications
```

5000.01
Internal_Com...

MOV(021)	Move
&8	Source word
D30088 ETN_Socket...	Destination Socket option for ETN Unit/Socket No. 8

MOV(021)	Move
&0	Source word
D30089 ETN_LocalPo...	Destination Local port number for ETN Unit

MOVD(083)	Move Digit
&192	Source word
#210	Control word
D30090 ETN_Remote...	Destination word Remote IP address for ETN Unit

MOVD(083)	Move Digit
&168	Source word
#10	Control word
D30090 ETN_Remote...	Destination word Remote IP address for ETN Unit

MOVD(083)	Move Digit
&188	Source word
#210	Control word
D30091 ETN_Remote...	Destination word Remote IP address for ETN Unit

MOVD(083)	Move Digit
&2	Source word
#10	Control word
D30091 ETN_Remote...	Destination word Remote IP address for ETN Unit

MOV(021)	Move
&2001	Source word
D30092 ETN_Remote...	Destination Remote port number for ETN Unit

No.	Overview	Description
1.3.	Control Data Setting	Sets the monitoring time for each processing phase. Set whether or not receive processing is required.
1.4.	Common Parameter Setting	Sets the common parameters for TCP socket communications.
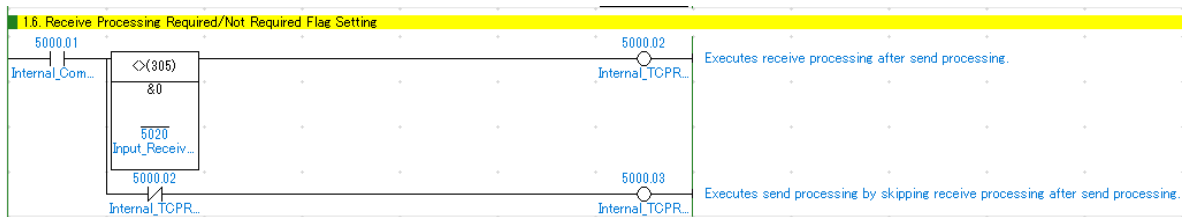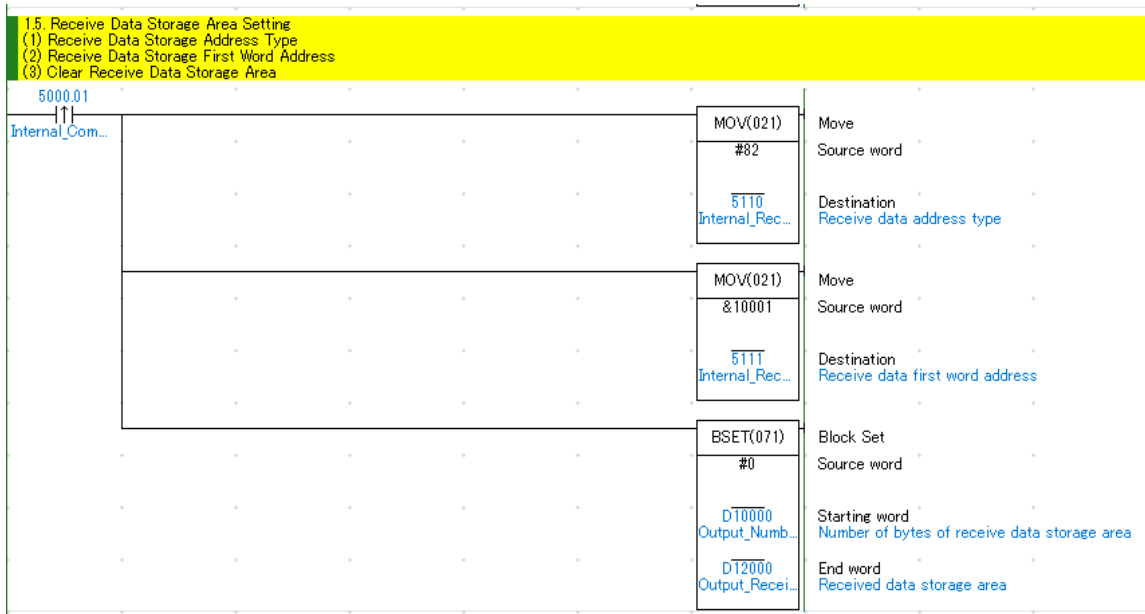
No.	Overview	Description
1.5.	Receive Data Storage Area Setting	Clears the receive area.
1.6.	Receive Processing Required/Not Required Flag Setting	Sets the flag to reflect the receive processing required/not required setting.

Precautions for Safe Use

Check the customer specifications of the program before changing the receive data storage area. An unexpected memory area may be overwritten.

1.7. Error Code Storage Area Initialization

```
5000.01
─┤↑├─                                    ┌─ MOV(021) ─┐  Move
Internal_Com…                            │   #FFFF    │  Source word
                                         │            │
                                         │   H400     │  Destination
                                         │ Output_Open…│  Open processing response code/timeout code
                                         └────────────┘

                                         ┌─ MOV(021) ─┐  Move
                                         │   #FFFF    │  Source word
                                         │            │
                                         │   H401     │  Destination
                                         │ Output_Send…│  Send processing response code/timeout code
                                         └────────────┘

                                         ┌─ MOV(021) ─┐  Move
                                         │   #FFFF    │  Source word
                                         │            │
                                         │   H402     │  Destination
                                         │ Output_Recei…│ Receive processing response code/timeout code
                                         └────────────┘

                                         ┌─ MOV(021) ─┐  Move
                                         │   #FFFF    │  Source word
                                         │            │
                                         │   H403     │  Destination
                                         │ Output_Code…│ Code reader end code/response code
                                         └────────────┘

                                         ┌─ MOV(021) ─┐  Move
                                         │   #FFFF    │  Source word
                                         │            │
                                         │   H404     │  Destination
                                         │ Output_Close…│ Close processing response code/timeout code
                                         └────────────┘
```

1.8. Initial Setup End Processing

```
5000.01                                            5000.04
─┤ ├─────────────────────────────────────────────────( )──  Ends initial setup. → To open processing
Internal_Com…                                  Internal_Initial…
```

No.	Overview	Description
1.7.	Error Code Storage Area Initialization	Initializes the error code storage area in the event of an error.
1.8.	Initial Setup End Processing	Turns ON the Initial Setup End Flag.

● 2. Open Processing
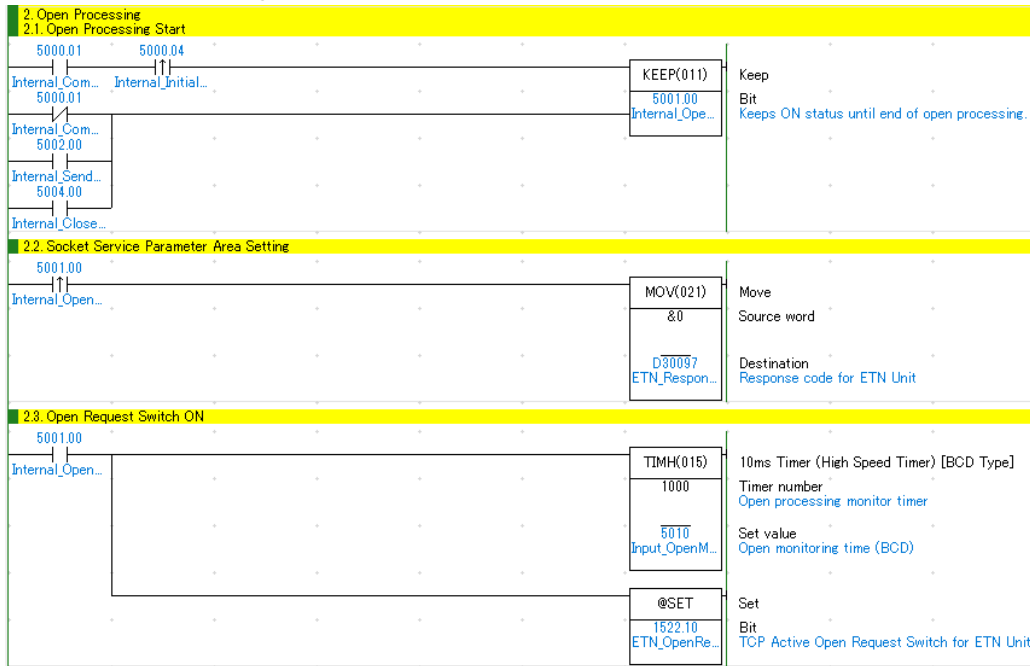
No.	Overview	Description
2.1.	Open Processing Start	Starts open processing. The open processing ends by moving to send or close processing.
2.2.	Socket Service Parameter Area Setting	Sets parameters required for open processing. • Clears the response code storage area.
2.3.	Open Request Switch ON	Starts the open processing monitor timer and turns ON the dedicated control bit for open processing request.

No.	Overview	Description
2.4.	Normal/Error Judgment Processing	Makes judgment of "normal/error end" or "timeout error" as a result of open processing. The processing goes to send processing if normal end or to close processing if error end.
2.5.	Error Code Storage Area Setting	Sets #0 in the error code storage area if the judgment in *2.4. Normal/Error Judgment Processing* is "normal end". Sets the following values in the error code storage area if judgment in *2.4. Normal/Error Judgment Processing* is "error end". • Response code error: Response code • Timeout: #F101 * Refer to *9.6. Error Code List* for details on error response codes.

- 3. Send Processing

3. Send Processing
3.1. Send Processing Start

5000.01	5001.01					KEEP(011)	Keep
Internal_Com...	Internal_Open...						
5000.01						5002.00	Bit
Internal_Com...						Internal_Send...	Keeps ON status until end of send processing.
5003.00							
Internal_Recei...							
5004.00							
Internal_Close...							

3.2. Socket Service Parameter Area Setting

5002.00					MOV(021)	Move
Internal_Send...					D9000	Source word
					Input_Numbe...	Number of bytes of send data
					D30093	Destination
					ETN_Number...	Number of byes of send/receive data for ETN Unit

					MOVD(083)	Move Digit
					5030	Source word
					Internal_Send...	Send data address type
					#210	Control word
					D30094	Destination word
					ETN_Send_R...	Send/receive data address for ETN Unit

					MOVD(083)	Move Digit
					5031	Source word
					Internal_Send...	Send data first word address
					#12	Control word
					D30094	Destination word
					ETN_Send_R...	Send/receive data address for ETN Unit

					MOVD(083)	Move Digit
					5031	Source word
					Internal_Send...	Send data first word address
					#210	Control word
					D30095	Destination word
					ETN_Send_R...	Send/receive data address for ETN Unit

					MOVD(083)	Move Digit
					#0	Source word
					#10	Control word
					D30095	Destination word
					ETN_Send_R...	Send/receive data address for ETN Unit

					MOV(021)	Move
					&0	Source word
					D30097	Destination
					ETN_Respon...	Response code for ETN Unit

3.3. Send Request Switch ON

5002.00					TIMH(015)	10ms Timer (High Speed Timer) [BCD Type]
Internal_Send...					1001	Timer number
						Send processing monitor timer
					5011	Set value
					Input_SendM...	Send monitoring time (BCD)

					@SET	Set
					1522.11	Bit
					ETN_SendRe...	Send Request Switch for ETN Unit

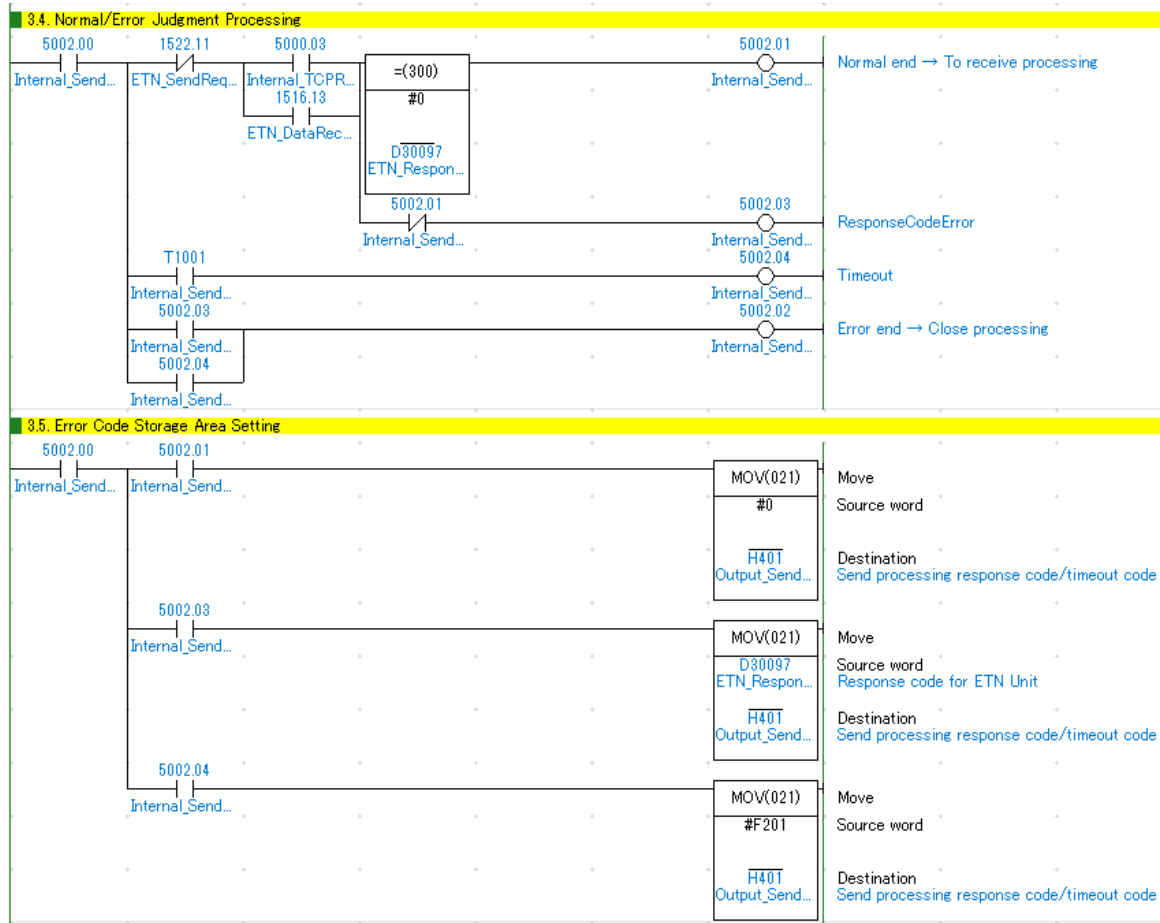| No.	Overview	Description
3.1.	Send Processing Start	Starts send processing. The send processing ends by moving to receive or close processing.
3.2.	Socket Service Parameter Area Setting	Sets parameters required for send processing. • Sets the number of bytes of send data. • Sets the address type of the send data storage area. • Sets the first word of the send data storage area. • Clears the response code storage area.
3.3.	Send Request Switch ON	Starts the send processing monitor timer and turns ON the dedicated control bit for send processing request.

No.	Overview	Description
3.4.	Normal/Error Judgment Processing	Makes judgment of "normal/error end" or "timeout error" as a result of send processing. The processing goes to receive processing if normal end or to close processing if error end.
3.5.	Error Code Storage Area Setting	Sets #0 in the error code storage area if the judgment in *3.4. Normal/Error Judgment Processing* is "normal end". Sets the following values in the error code storage area if judgment in *3.4. Normal/Error Judgment Processing* is "error end". • Response code error: Response code • Timeout: #F201 * Refer to *9.6. Error Code List* for details on error response codes.

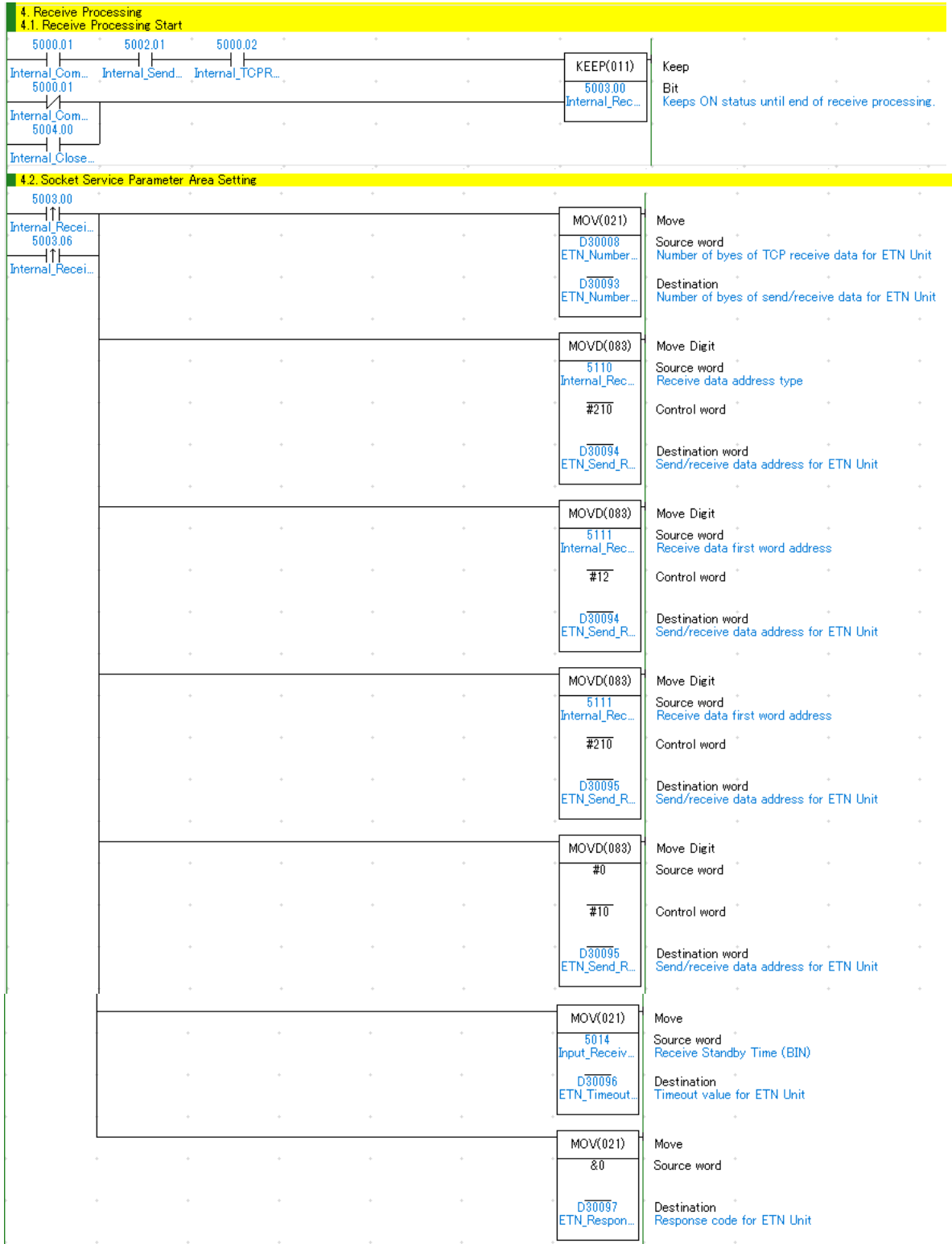- 4. Receive Processing

No.	Overview	Description
4.1.	Receive Processing Start	Starts receive processing when "Receiving processing required" is set.
		Receiving process is skipped when "Receiving process not required" is set.
		The receive processing ends by moving to close processing.
4.2.	Socket Service Parameter Area Setting	Sets parameters required for receive processing.
		• Sets the number of bytes of receive data.
		• Sets the address type of the receive data storage area.
		• Sets the first word of the receive data storage area.
		• Sets the Receive Standby Time for receive data.
		• Clears the response code storage area.

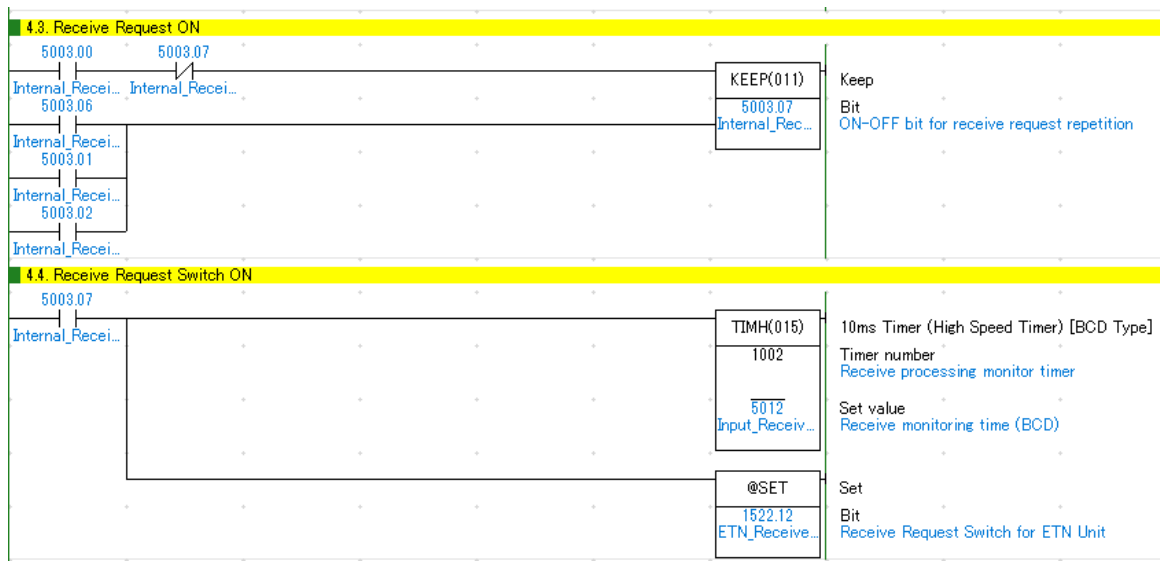| No.	Overview	Description
4.3.	Receive Request ON	Turns ON the Receive Request Switch by manipulating "Receive Request ON". (Turns ON and OFF "Receive Request ON" to repeat receive processing.)
4.4.	Receive Request Switch ON	Starts the receive processing monitor timer and turns ON the dedicated control bit for receive processing request.

4.5. Normal/Error Judgment Processing
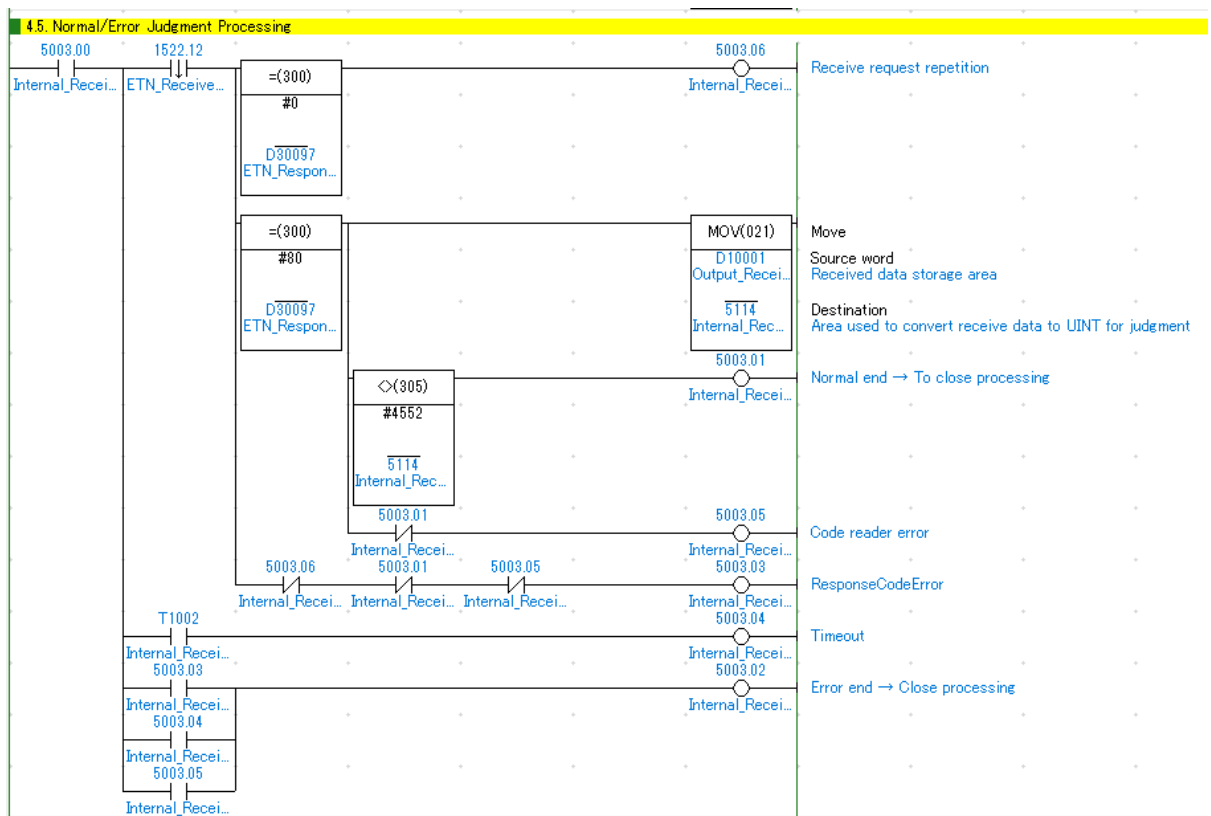
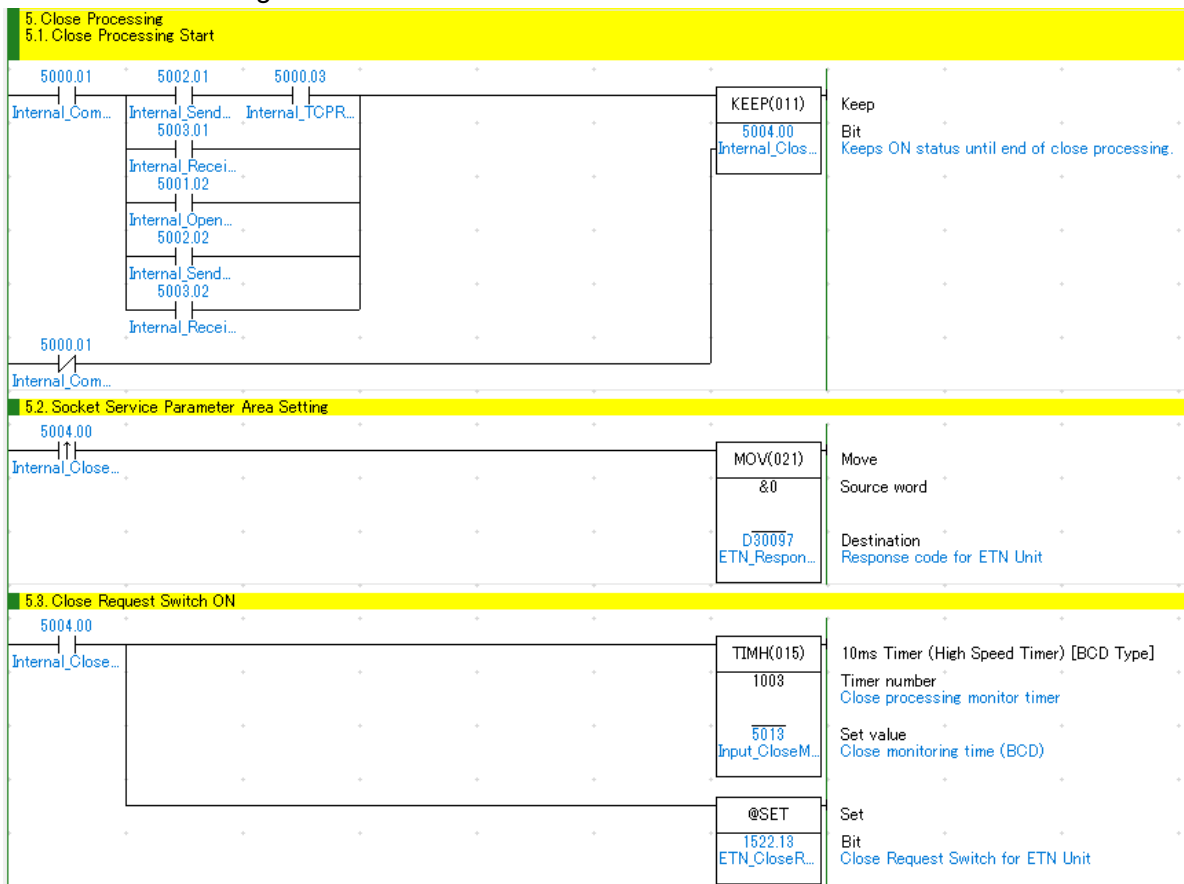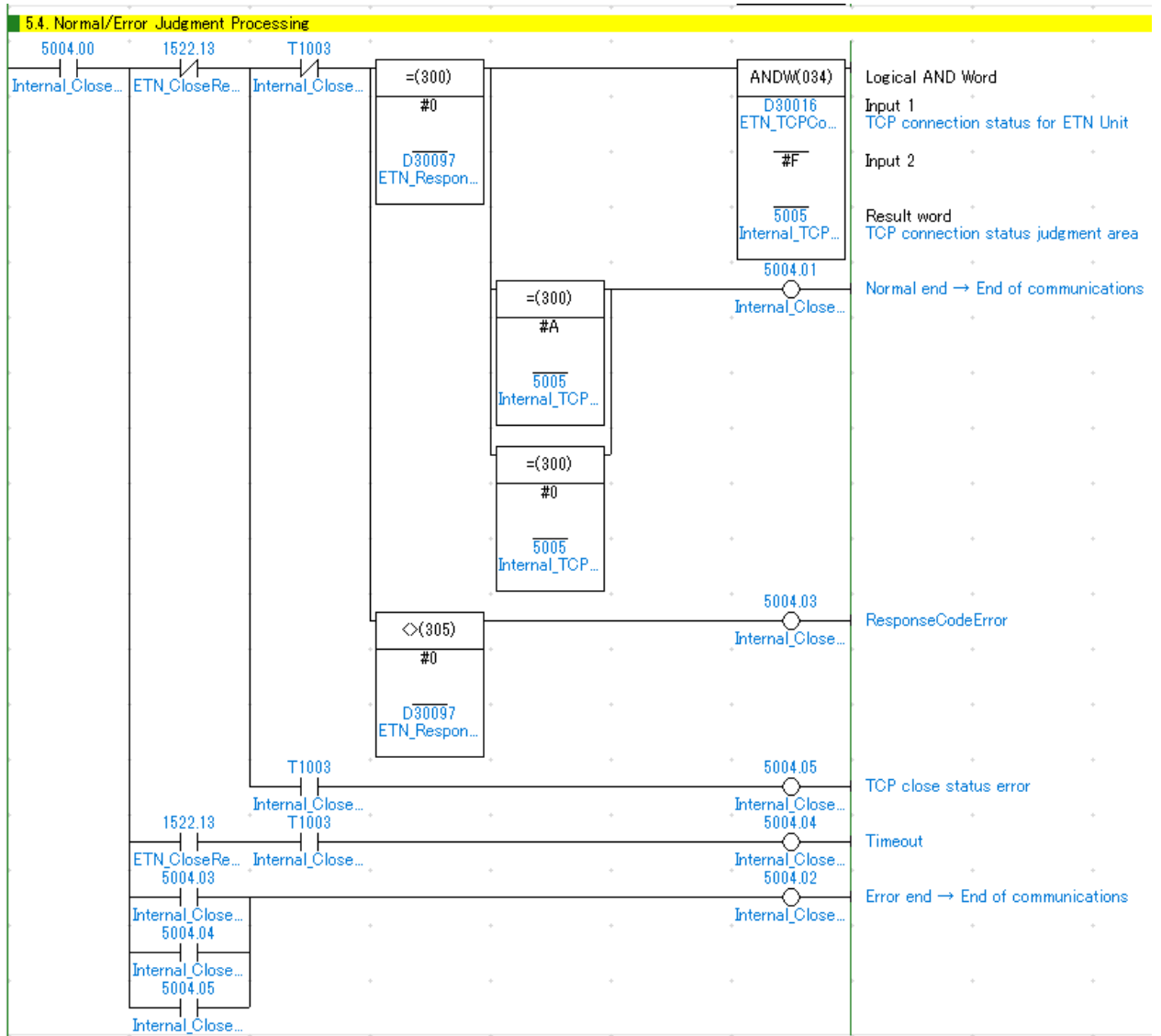| No.	Overview	Description
4.5.	Normal/Error Judgment Processing	Makes judgment of "receive processing repetition", "normal/error end, timeout error", or "code reader error" as a result of receive processing. The processing goes to close processing after the end of receive processing.

4.6. Receive Processing Repetition Information Calculation

5003.06
Internal_Recei...

+(400) — Signed Binary Add Without Carry
D30093 ETN_Number... — Augend word — Number of byes of send/receive data for ETN Unit
D10000 Output_Numb... — Addend word — Number of bytes of receive data storage area
D10000 Output_Numb... — Result word — Number of bytes of receive data storage area

/(430) — Signed Binary Divide
D30093 ETN_Number... — Dividend word — Number of byes of send/receive data for ETN Unit
&2 — Divisor word
5112 Internal_Rec... — Result word — Receive data first word increment

+(400) — Signed Binary Add Without Carry
5112 Internal_Rec... — Augend word — Receive data first word increment
5113 Internal_Rec... — Addend word — Receive data first word increment remaining
5112 Internal_Rec... — Result word — Receive data first word increment

+(400) — Signed Binary Add Without Carry
5111 Internal_Rec... — Augend word — Receive data first word address
5112 Internal_Rec... — Addend word — Receive data first word increment
5111 Internal_Rec... — Result word — Receive data first word address

4.7. Error Code Storage Area Setting

5003.00 Internal_Recei...   5003.01 Internal_Recei...

MOV(021) — Move
#0 — Source word
H402 Output_Recei... — Destination — Receive processing response code/timeout code

MOV(021) — Move
#0 — Source word
H403 Output_Code... — Destination — Code reader end code/response code

5003.05 Internal_Recei...

MOV(021) — Move
#0 — Source word
H402 Output_Recei... — Destination — Receive processing response code/timeout code

MOV(021) — Move
#F302 — Source word
H403 Output_Code... — Destination — Code reader end code/response code

5003.03 Internal_Recei...

MOV(021) — Move
D30097 ETN_Respon... — Source word — Response code for ETN Unit
H402 Output_Recei... — Destination — Receive processing response code/timeout code

5003.04 Internal_Recei...

MOV(021) — Move
#F301 — Source word
H402 Output_Recei... — Destination — Receive processing response code/timeout code

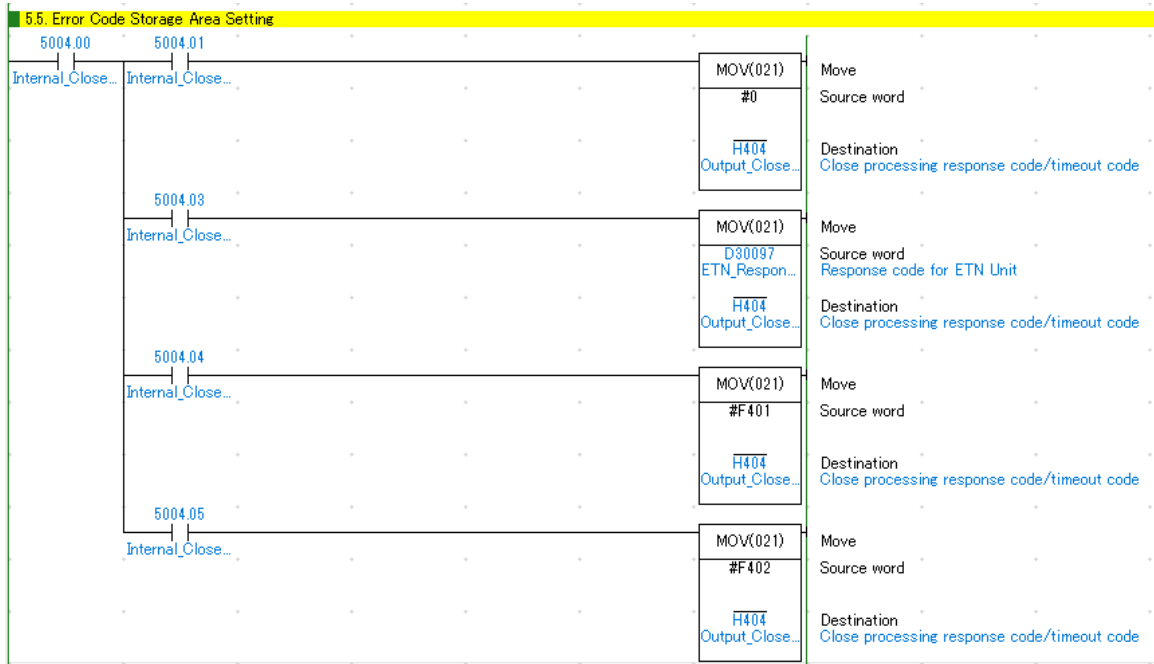| No.	Overview	Description
4.6.	Receive Processing Repetition Information Calculation	Performs the processing below if judgment in *4.5. Normal/Error Judgment Processing* is "receive processing repetition". • Adds the number of bytes of the receive data to the total number of bytes of receive data. • Calculates the first word of the receive data for storage of the next received data.
4.7.	Error Code Storage Area Setting	Sets #0 in the error code storage area if the judgment in *4.5. Normal/Error Judgment Processing* is "normal end". Sets the following values in the error code storage area if judgment in *4.5. Normal/Error Judgment Processing* is "error end". • Response code error: Response code • Timeout: #F301 • Code reader error: #F302, or code reader end code converted to hex * Refer to *9.6. Error Code List* for details on error response codes.

● 5. Close Processing

No.	Overview	Description
5.1.	Close Processing Start	Starts close processing. Close processing ends when the communications processing ends (changes to non-executing state).
5.2.	Socket Service Parameter Area Setting	Sets parameters required for close processing. • Clears the response code storage area.
5.3.	Close Request Switch ON	Starts the close processing monitor timer and turns ON the dedicated control bit for close processing request.

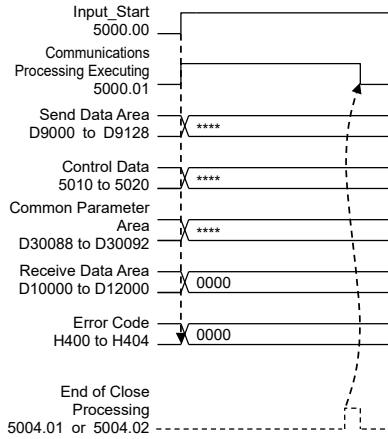| No.	Overview	Description
5.4.	Normal/Error Judgment Processing	Makes judgment of "normal/error end", "timeout error", or "close status error" as a result of close processing. The communications processing ends after the end of close processing.

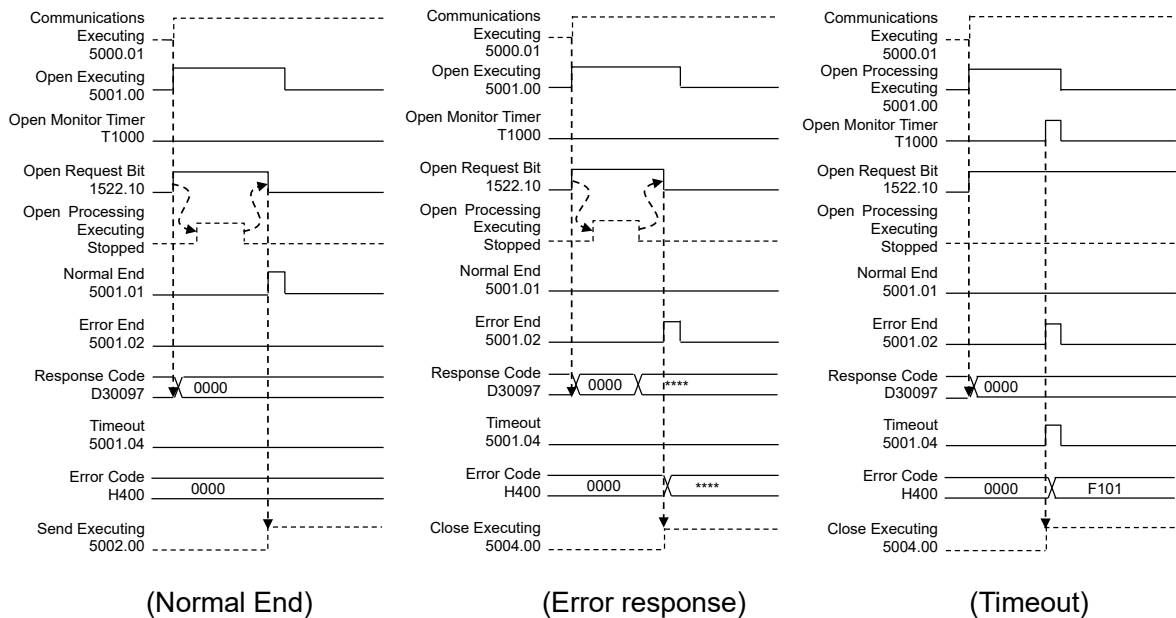| No.	Overview	Description
5.5.	Error Code Storage Area Setting	Sets #0 in the error code storage area if the judgment in *5.4. Normal/Error Judgment Processing* is "normal end".
		Sets the following values in the error code storage area if judgment in *5.4. Normal/Error Judgment Processing* is "error end".
		• Response code error: Response code
		• Timeout: #F401
		• Close processing status error: #F402
		* Refer to *9.6. Error Code List* for details on error response codes.

60

9.5. Timing Chart

The timing chart for the ladder program is shown below.
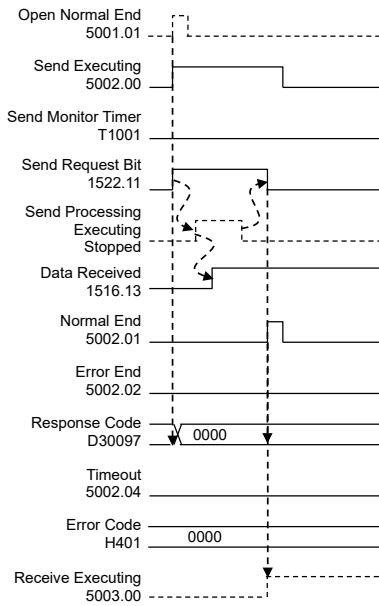
● Start and Setup

● Open Processing
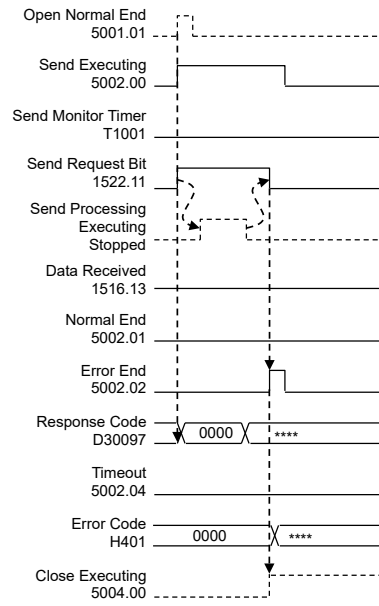
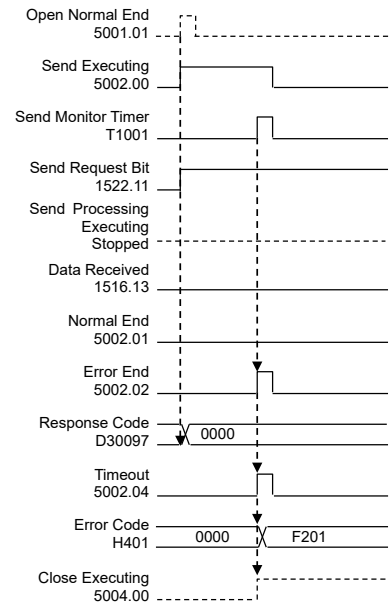

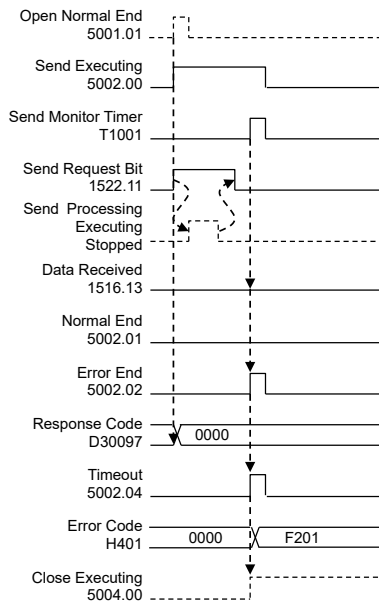(Normal End)   (Error response)   (Timeout)

● Send Processing

(Normal End)

(Error response)
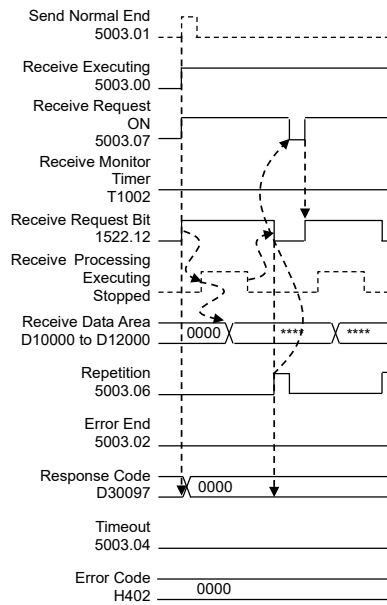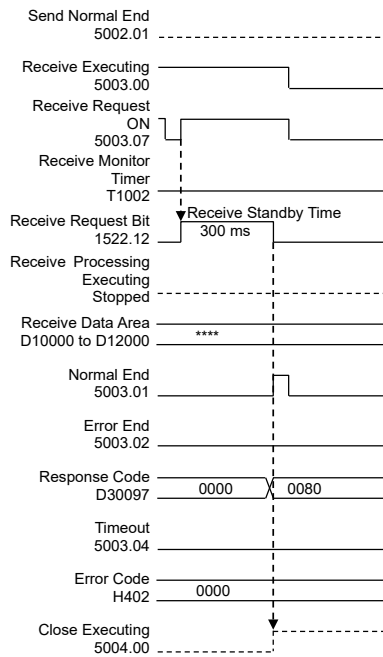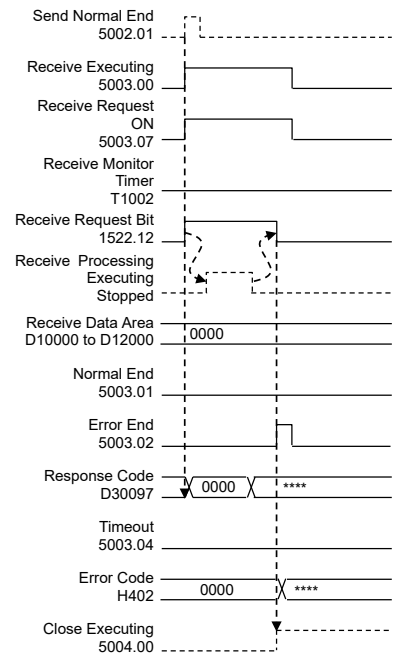

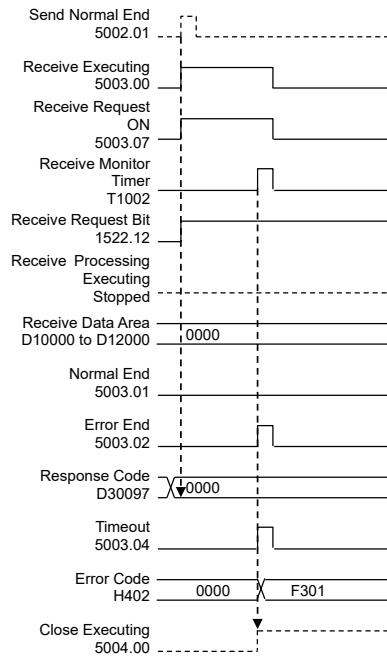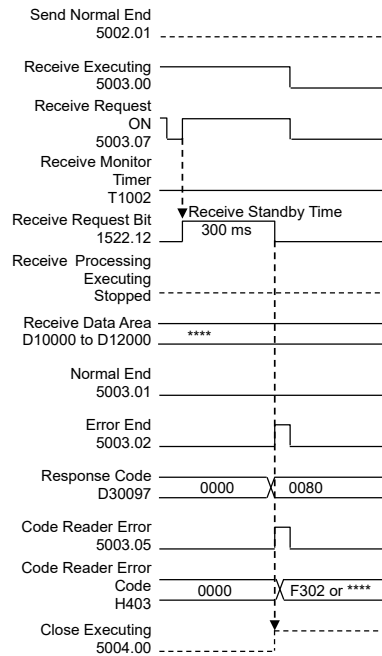(Timeout)

(Timeout: No receive data)

● Receive Processing

Signal	(Repetition)
Send Normal End 5003.01	
Receive Executing 5003.00	
Receive Request ON 5003.07	
Receive Monitor Timer T1002	
Receive Request Bit 1522.12	
Receive Processing Executing Stopped	
Receive Data Area D10000 to D12000	0000 **** ****
Repetition 5003.06	
Error End 5003.02	
Response Code D30097	0000
Timeout 5003.04	
Error Code H402	0000

(Repetition)

(Normal End)

Send Normal End 5002.01
Receive Executing 5003.00
Receive Request ON 5003.07
Receive Monitor Timer T1002
Receive Request Bit 1522.12 — Receive Standby Time 300 ms
Receive Processing Executing Stopped
Receive Data Area D10000 to D12000 — ****
Normal End 5003.01
Error End 5003.02
Response Code D30097 — 0000   0080
Timeout 5003.04
Error Code H402 — 0000
Close Executing 5004.00

(Error End)

Send Normal End 5002.01
Receive Executing 5003.00
Receive Request ON 5003.07
Receive Monitor Timer T1002
Receive Request Bit 1522.12
Receive Processing Executing Stopped
Receive Data Area D10000 to D12000 — 0000
Normal End 5003.01
Error End 5003.02
Response Code D30097 — 0000   ****
Timeout 5003.04
Error Code H402 — 0000   ****
Close Executing 5004.00

(Timeout)

Send Normal End 5002.01
Receive Executing 5003.00
Receive Request ON 5003.07
Receive Monitor Timer T1002
Receive Request Bit 1522.12
Receive Processing Executing Stopped
Receive Data Area D10000 to D12000 — 0000
Normal End 5003.01
Error End 5003.02
Response Code D30097 — 0000
Timeout 5003.04
Error Code H402 — 0000   F301
Close Executing 5004.00

(Code Reader Error)

Send Normal End 5002.01
Receive Executing 5003.00
Receive Request ON 5003.07
Receive Monitor Timer T1002
Receive Request Bit 1522.12 — Receive Standby Time 300 ms
Receive Processing Executing Stopped
Receive Data Area D10000 to D12000 — ****
Normal End 5003.01
Error End 5003.02
Response Code D30097 — 0000   0080
Code Reader Error 5003.05
Code Reader Error Code H403 — 0000   F302 or ****
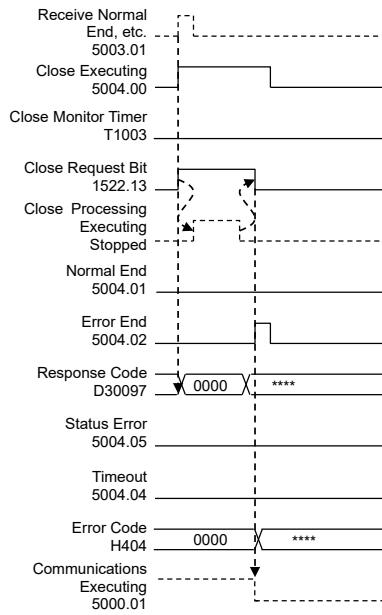Close Executing 5004.00

63
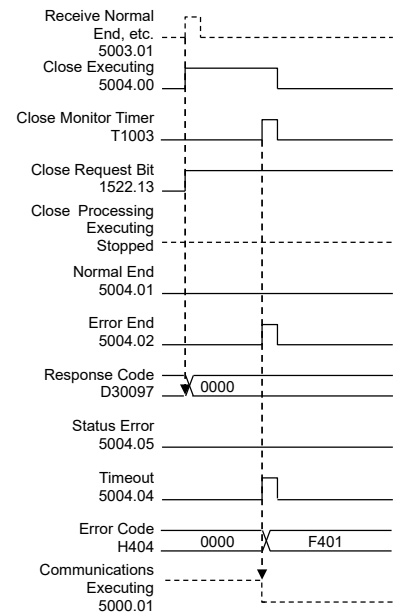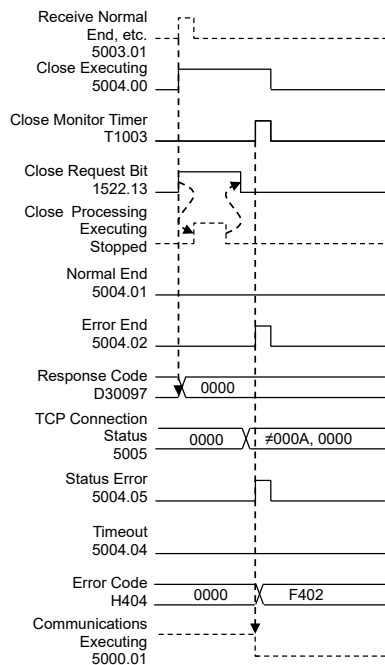
● Close Processing

(Normal End)   (Error response)   (Timeout)

(Status Error)

9.6. Error Code List

- Response Codes

  The response codes for open processing, send processing, receive processing, and close processing are set in H400, H401, H402, and H404, respectively.

  The table below shows the main response codes.

(O: Open processing, S: Send processing, R: Receive processing, C: Close processing, ○: Applicable processing)

Response code	O	S	R	C	Description
0000	○	○	○	○	Normal end
0105	○				Local IP address setting error
0302	○	○	○	○	CPU Unit error
1100	○	○	○		TCP socket number not 1 to 8, remote IP address 0 in open processing Number of bytes of send data out of allowable range in send processing Number of bytes of receive data out of allowable range in receive processing
1101		○	○		Variable type for send/ receive data address out of allowable range
1103		○	○		Bit address of send/receive data not 0
110C	○	○	○		Request switch turned ON in another processing
220F	○	○	○		Specified socket already open or in open processing executing state in open processing Specified socket in send processing executing state in send processing Specified socket in receive processing executing state in receive processing
2210		○	○	○	Connection not established for specified socket
2211	○	○	○	○	Service not executable because Unit is busy
2606	○				Unable to open specified socket via TCP because already it is opened by UDP
2607	○	○	○	○	Specified Socket Service Parameter Area in use by another socket
000D	○				Remote IP address parameter error
0020		○			Connection with remote socket lost in send processing
003E	○	○	○		Unable to secure internal buffer because receive load is high
0045	○				Local socket closed
0049	○				Port number duplicated
004A	○	○			Error or remoter device not in passive open state in open processing Communications error with remote node in send processing
004B		○			Communications error with remote node
004C	○				Remote IP address parameter error, incorrect parameter specification Active open request made to local TCP port of local node
0053	○		○		Communications error with remote node, remote node not existing
0066			○		Service not executable because memory for internal processing cannot be secured
0080			○		A timeout occurred during receiving.
0081	○	○	○		Socked closed in open processing Specified socket closed in send processing Specified socket closed in receive processing
FFFF	○	○	○	○	Processing skipped for some reason

📓 **Note**

For details, refer to *6-7-6 Response Codes* in *Section 6 Socket Services* in the *CS/CJ-series Ethernet Units Operation Manual Construction of Applications* (Cat. No. W421).

📓 **Note**

For details on the Ethernet Unit error and correction, refer to *8-4 troubleshooting Procedures* in *Section 8 Troubleshooting* in the *CS/CJ-series Ethernet Units Operation Manual Construction of Networks* (Cat. No. W420).

● Timeout Error/TCP Connection Status Error

The timeout error codes for open processing, send processing, and receive processing are set in H400, H401, and H402, respectively. The timeout error code/TCP connection status error code for close processing is set in H404.

(O: Open processing, S: Send processing, R: Receive processing, C: Close processing, ○: Applicable processing)

Error code	O	S	R	C	Description
0000	○	○	○	○	Normal end
F101	○				Open processing not completed within specified time
F201		○			Send processing not completed within specified time (This includes cases where response to be received was not received.)
F301			○		Receive processing not completed within specified time
F401				○	Close processing not completed within specified time
F402				○	Normal TCP connection status not shown within specified time after close processing
FFFF	○	○	○	○	Processing skipped for some reason

● Code Reader Error Code

The error codes of code reader errors detected in receive processing are set in H403.

(O: Open processing, S: Send processing, R: Receive processing, C: Close processing, ○: Applicable processing)

Error code	O	S	R	C	Description
0000			○		Normal end
F302			○		Response from code reader is "ER"
FFFF			○		Judgment of code reader error not made due to no response from code reader for some reason

* "ER" is the only error response supported for communications errors. For details, check the monitor information on the code reader.

10. Revision History

Revision Code	Revision Date	Revised Reason and Page
01	November 2023	First Publication