Machine Automation Controller NJ-series

Serial (RS-232C) Communications Connection Guide

OMRON Corporation
Auto Focus Multi Code Reader
V320-F / V420-F / V430-F-series

Network

Connection

Guide

Z430-E1-01

About Copyrights and Trademarks

Contents

1. Related Manuals

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.

The following OMRON Corporation (hereinafter referred to as “OMRON”) manuals are related to this document:

Cat. No.	Model	Manual name
W500	NJ Series	NJ-series CPU Unit Hardware User’s Manual
W501	NJ/NX Series	NJ/NX-series CPU Unit Software User’s Manual
W336	CJ Series	CJ-series Serial Communications Boards/Units Operation Manual
W504	SYSMAC-SE2□□□	Sysmac Studio Version 1 Operation Manual
W502	NJ/NX Series	Machine Automation Controller Instructions Reference Manual
Z432	V320-F/V330-F/V420-F/V430-F Series	MicroHAWK V320-F/V330-F/V420-F/V430-F Series Barcode Reader User Manual
Z407	V320-F/V330-F/V420-F/V430-F Series	Autofocus Multicode Reader MicroHAWK V320-F/V330-F/V420-F/V430-F Series User Manual for Communication Settings

2. Terms and Definitions

Term	Description/Definition
Protocol Macro	A function that stores the procedure to send/receive data (protocol) to/from a general-purpose external device in the Serial Communications Board or Unit and executes the PMCR instruction in the CPU Unit to enable data communications with general-purpose external devices.
Protocol	A unit that compiles independent communications processing steps for a specific general-purpose external device into a data communications procedure. A “protocol” consists of two or more sequences.
Sequence	A unit of independent communications processing steps that can be executed by the PMCR instruction in a ladder program. When a sequence is started, the steps in the sequence are executed sequentially.
Step	A unit to execute processing, which is send processing, receive processing, or send & receive processing of a message, or receive buffer clear or step standby. Up to 15 steps can be set in a sequence.
Send Message	A communications frame (command) sent to the destination general-purpose external device. It is called from a step in a sequence and sent to the general-purpose external device.
Receive Message	A communication frame (response) sent from the destination general-purpose external device. It is called from a step in a sequence and compared with data received from the general-purpose external device.
Reception Matrix	A reception matrix is used when it is impossible to uniquely identify communication frames (responses) sent from a general-purpose external device. In a reception matrix, more than one communication frame can be registered.
Case	A unit for registering multiple communication frames (responses) in a reception matrix. One communication frame is registered as one case. Up to 15 cases can be registered in a reception matrix.

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 March 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 code reader products (V320-F/V420-F/V430-F) to an NJ Series Machine Automation Controller with a Serial Communications Unit (hereinafter referred to as “controller”) via serial communications and for checking their connections.
Refer to *Section 6. Serial Communication Settings* and *Section 7. Connection Procedure* to understand the setting procedures and key points of setup to establish a serial communications connection.
Using the user program in the “Sysmac Studio Project File” and the protocol data in the “CX-Protocol Project File” prepared in advance, use the “Read trigger” command for the code reader to check the serial communications connection.

Obtain the latest version of the Project Files in advance. Contact OMRON for information on how to obtain this file.

Name	Filename	Version
Sysmac Studio Project File (Extension: smc2)	OMRON_V320_V420_V430_NJ_ PMCR232C_V100.smc2	Ver. 1.00
CX-Protocol Project File (Extension: psw)	OMRON_V320_V420_V430_NJ_ PMCR232C_V100.psw	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 Support Tools

5.1. Applicable Products

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

Manufacturer	Name	Model	Version
OMRON	NJ Series CPU Unit	NJ501-1500 NJ501-1400 NJ501-1300 NJ301-□□□□	Same or later version as indicated in section 5.2.
OMRON	Serial Communications Unit	CJ1W-SCU□1-V1 CJ1W-SCU□2	
OMRON	Code reader	V320-F□□□□□□□-□□□ V420-F□□□□□□□-□□□ V430-F□□□□□□□-□□□	

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.

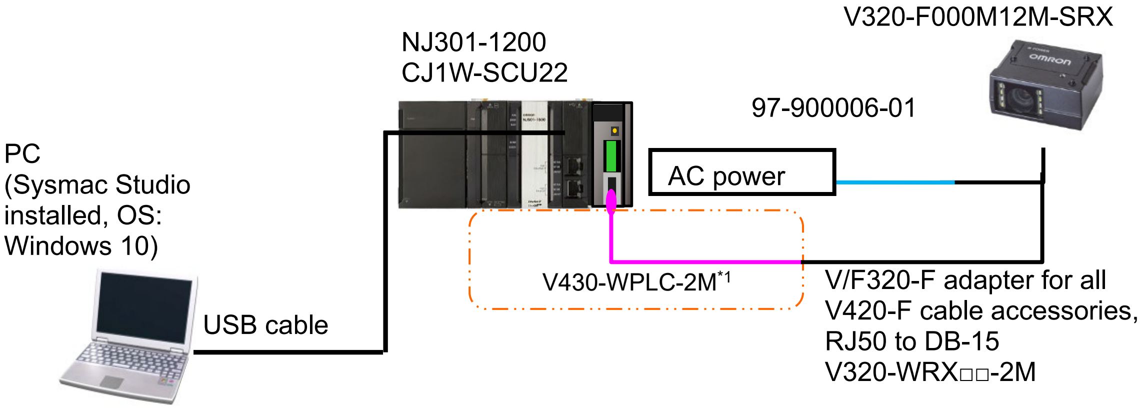
Precautions for Correct Use

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.

- Configuration with V320-F

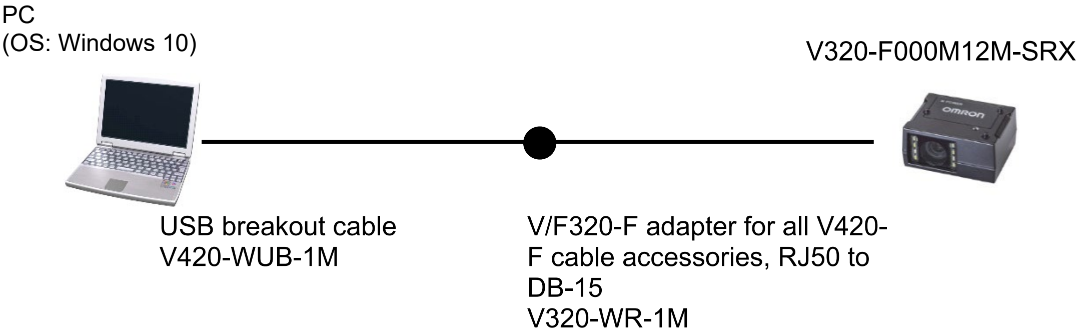

V320-F000M12M-SRX
NJ301-1200
CJ1W-SCU22
97-900006-01
PC
(Sysmac Studio installed, OS: Windows 10)
AC power
V430-WPLC-2M*1
V/F320-F adapter for all V420-F cable accessories, RJ50 to DB-15
V320-WRX□□-2M
USB cable

Manufacturer	Name	Model	Version
OMRON	NJ Series CPU Unit (Built-in EtherNet/IP Port)	NJ301-1200	Ver. 1.19
OMRON	Power Supply Unit	NJ-PA3001	
OMRON	Serial Communications Unit	CJ1W-SCU22	Ver. 2.1
OMRON	Sysmac Studio	SYSMAC-SE2□□□	Ver. 1.28
OMRON	Sysmac Studio Project File	OMRON_NJ_V320_V420_V430_PMCR232C_V100.smc2	Ver. 1.00
OMRON	CX-Protocol Project File	OMRON_NJ_V320_V420_V430_PMCR_V100.psw	Ver. 1.00
---	PC (OS: Windows 10)	---	
---	USB cable (USB 2.0-compliant B-type connector)	---	
OMRON	Code reader	V320-F000M12M-SRX	Ver. 2.1.0
OMRON	V/F320-F adapter for all V420-F cable accessories, RJ50 to DB-15	V320-WRX□□-2M	
OMRON	OMRON Programmable Controller (CS/CJ/NJ) RS-232C cable	V430-WPLC-2M*1	
OMRON	Power supply	97-900006-01	
	AC power supply		

*1. Refer to *6.2. Cable Wiring Diagram* for how to prepare the RS-232C cable yourself.

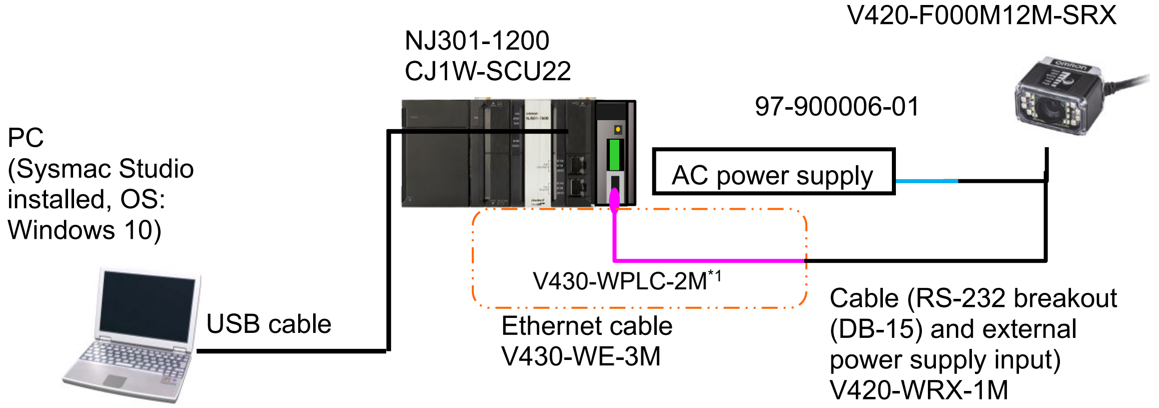
WebLink cannot be used for RS-232C connection.

To use WebLink, the following system components for USB connection are required.



Manufacturer	Name	Model	Version
	PC (OS: Windows 10)		
OMRON	Code reader	V320-F000M12M-SRX	Ver. 2.1.0
OMRON	V/F320-F adapter for all V420-F cable accessories, RJ50 to DB-15	V320-WR-1M	
OMRON	USB breakout cable	V420-WUB-1M	

- Configuration with V420-F

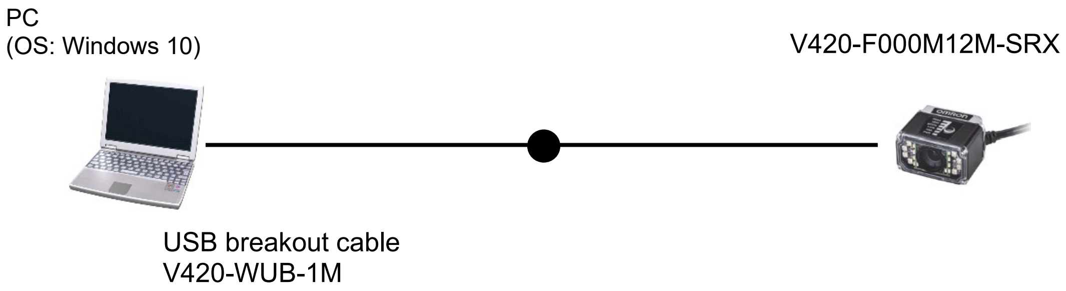


Manufacturer	Name	Model	Version
OMRON	NJ Series CPU Unit (Built-in EtherNet/IP Port)	NJ301-1200	Ver. 1.19
OMRON	Power Supply Unit	NJ-PA3001	
OMRON	Serial Communications Unit	CJ1W-SCU22	Ver. 2.1
OMRON	Sysmac Studio	SYSMAC-SE2□□□	Ver. 1.28
OMRON	Sysmac Studio Project File	OMRON_NJ_V320_ V420_V430_PMCR232C _V100.smc2	Ver. 1.00
OMRON	CX-Protocol Project File	OMRON_NJ_V320_ V420_V430_PMCR_ V100.psw	Ver. 1.00
---	PC (OS: Windows 10)	---	
---	USB cable (USB 2.0-compliant B-type connector)	---	
OMRON	Code reader	V420-F000M12M-SRX	Ver. 2.1.0
OMRON	Cable (RS-232 breakout (DB-15) and external power supply input)	V420-WRX-1M	
OMRON	OMRON Programmable Controller (CS/CJ/NJ) RS-232C cable	V430-WPLC-2M*1	
OMRON	Power supply	97-900006-01	
	AC power supply		

*1. Refer to *6.2. Cable Wiring Diagram* for how to prepare the RS-232C cable yourself.

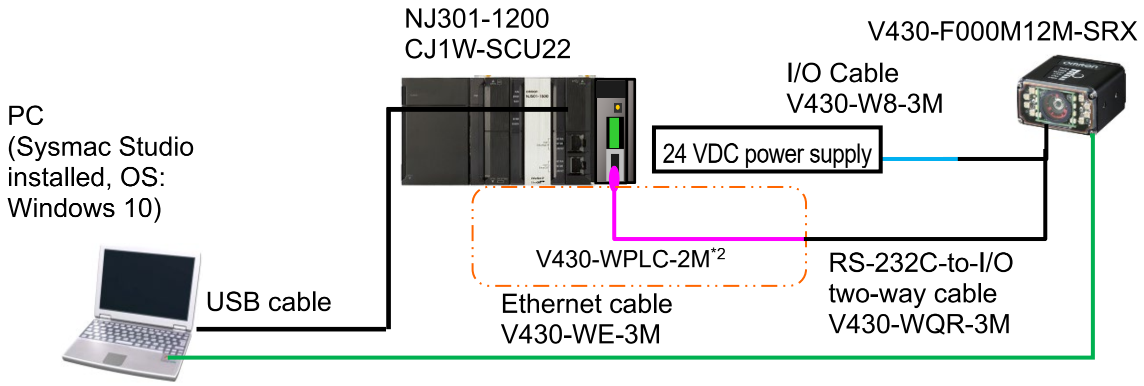
WebLink cannot be used for RS-232C connection.

To use WebLink, the following system components for USB connection are required.



Manufacturer	Name	Model	Version
	PC (OS: Windows 10)		
OMRON	Code reader	V420-F000M12M-SRX	Ver. 2.1.0
OMRON	USB breakout cable	V420-WUB-1M	

- Configuration with V430-F



Manufacturer	Name	Model	Version
OMRON	NJ Series CPU Unit (Built-in EtherNet/IP Port)	NJ301-1200	Ver. 1.19
OMRON	Power Supply Unit	NJ-PA3001	
OMRON	Serial Communications Unit	CJ1W-SCU22	Ver. 2.1
OMRON	Sysmac Studio	SYSMAC-SE2□□□	Ver. 1.28
OMRON	Sysmac Studio Project File	OMRON_NJ_V320_ V420_V430_PMCR232C_ V100.smc2	Ver. 1.00
OMRON	CX-Protocol Project File	OMRON_NJ_V320_V420_ V430_PMCR_V100.psw	Ver. 1.00
---	PC (OS: Windows 10)	---	
---	USB cable (USB 2.0-compliant B-type connector)	---	
OMRON	Code reader	V430-F000M12M-SRX	Ver. 2.1.0
OMRON	RS-232C-to-I/O two-way cable	V430-WQR-3M	
OMRON	OMRON Programmable Controller (CS/CJ/NJ) RS-232C cable	V430-WPLC-2M*2	
OMRON	I/O Cable	V430-W8-3M	
OMRON	Ethernet Cable*1	V430-WE-3M	
	24 VDC power supply		

*1: The Ethernet cable is used for WebLink connection.

*2. Refer to *6.2. Cable Wiring Diagram* for how to prepare the RS-232C cable yourself.

Precautions for Correct Use

Obtain the latest version of the Sysmac Studio Project File from OMRON in advance. (Contact OMRON for information on how to obtain this file.)

Precautions for Correct Use

Ensure that the Sysmac Studio and CX-Protocol are updated to the versions specified in this section or higher. If you use versions other than the versions specified in this section, there may be differences in the procedures in Section 7 and later. In that case, refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) or *CX-Protocol Operation Manual* (Cat. No. W344) to perform the equivalent procedures.

Note

The configuration may not be reproduced if the system component models or versions differ. Check your configuration and, if there is any difference in the models or versions, contact OMRON.

Note

This document assumes that the USB is used to connect the controller. For information on how to install the USB driver, refer to *A-1 Driver Installation for Direct USB Cable Connection* in *Appendices* of the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504).

Note

Refer to the *MicroHAWK V320-F/V330-F/V420-F/V430-F Series Barcode Reader User Manual* (Cat. No. Z432) for the power supply specifications that can be used for 24 VDC power supply (for the code reader).

6. Serial Communications Settings

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

Note

You need to modify the program to use communication settings that are not described in this section. For more information on the program, please refer to *Section 9. Program*.

6.1. Serial Communications Settings

The serial communications settings are as follows.

Parameter name	SCU unit*	Code reader
Unit number	0	---
Communications (connection) port	Port 2 (RS-232C)	---
Serial communications mode	Protocol macro	---
Data length	8 bits	8 bits (default)
Stop bit	1 bit	1 bit (default)
Parity	None	None (default)
Baud rate	9,600 bps (default)	9,600 bps
Protocol macro transmission method	Full-duplex	---
Header	---	None (default)
Footer	---	<CR+LF> (default)
Host port protocol	---	Point-to-point (default)

* SCU unit: Serial Communications Unit

Precautions for Correct Use

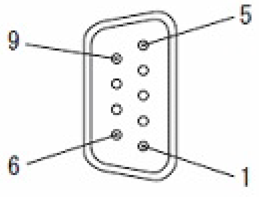
It is assumed that the CJ1W-SCU22 Serial Communications Unit is used with the unit number *0* and the communications (connection) port *Port 2*. If using other conditions to connect the Unit, refer to *Section 9. Program* and create a program by modifying the control data in the CIO Area and PMCR Instruction.

6.2. Cable Wiring Diagram

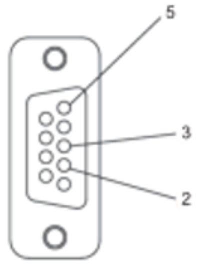
For details on cable wiring, refer to *Section 3 Installation and Connection* in the *CJ-series Serial Communications Boards/Units Operation Manual* (Cat. No. W336).
Check the connector shape and signal lines (pin assignment) before you prepare the cable.

■ Connector Shape and Signal Lines (Pin Assignment)
Applicable connector for CJ1W-SCU22: D-SUB 9-pin (Connector shape on CJ1W-SCU22 side: Female)

Pin No.	Abbreviation	Signal name	I/O
1	FG	Shield	---
2	SD	Send data	Output
3	RD	Receive data	Input
4	RS	Request to send	Output
5	CS	Clear to send	Input
6	5V	Power supply	---
7	DR	Data set ready	Input
8	ER	Data terminal ready	Output
9	SG	Signal ground	---
Shell	FG	Shield	---

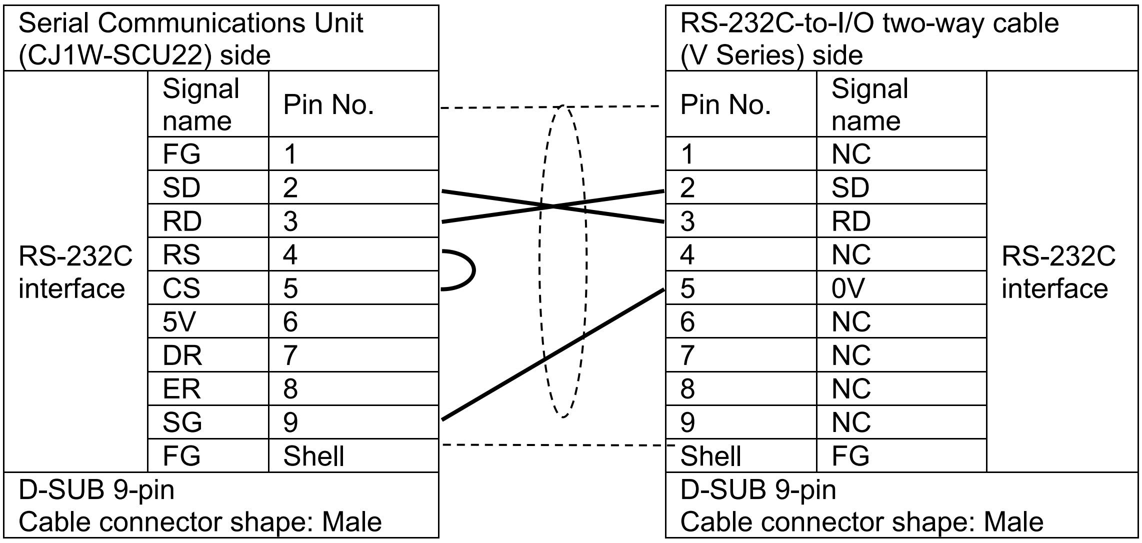


Applicable connector for V Series RS-232C cable: D-SUB 9-pin (Connector on V Series side: Female)

Pin No.	Signal name	Pin assignment
1	-	
2	SD	
3	RD	
4	-	
5	0V	
6	-	
7	-	
8	-	
9	-	

■ RS-232C Cable Pin Assignment

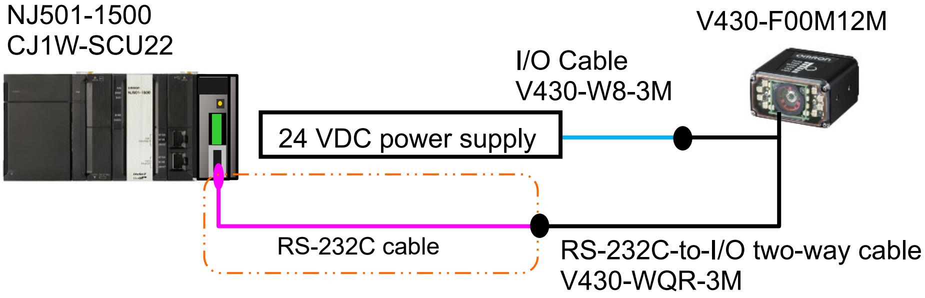
Prepare the RS-232C cable with reference to the following wiring diagram.



Serial Communications Unit (CJ1W-SCU22) side		
	Signal name	Pin No.
RS-232C interface	FG	1
	SD	2
	RD	3
	RS	4
	CS	5
	5V	6
	DR	7
	ER	8
	SG	9
	FG	Shell
D-SUB 9-pin Cable connector shape: Male		

RS-232C-to-I/O two-way cable (V Series) side		
Pin No.	Signal name	
1	NC	RS-232C interface
2	SD	
3	RD	
4	NC	
5	0V	
6	NC	
7	NC	
8	NC	
9	NC	
Shell	FG	
D-SUB 9-pin Cable connector shape: Male		

For V430-F



6.3. Example of Connection Check for Communications

This document assumes that you use a program in structured text (hereinafter, ST) language and protocol macro data to send and receive messages from the controller to the code reader. Between the controller and the code reader, the "Read trigger (Sequence No. 900)" message is sent and received. An overview of the sequence operation is shown below.

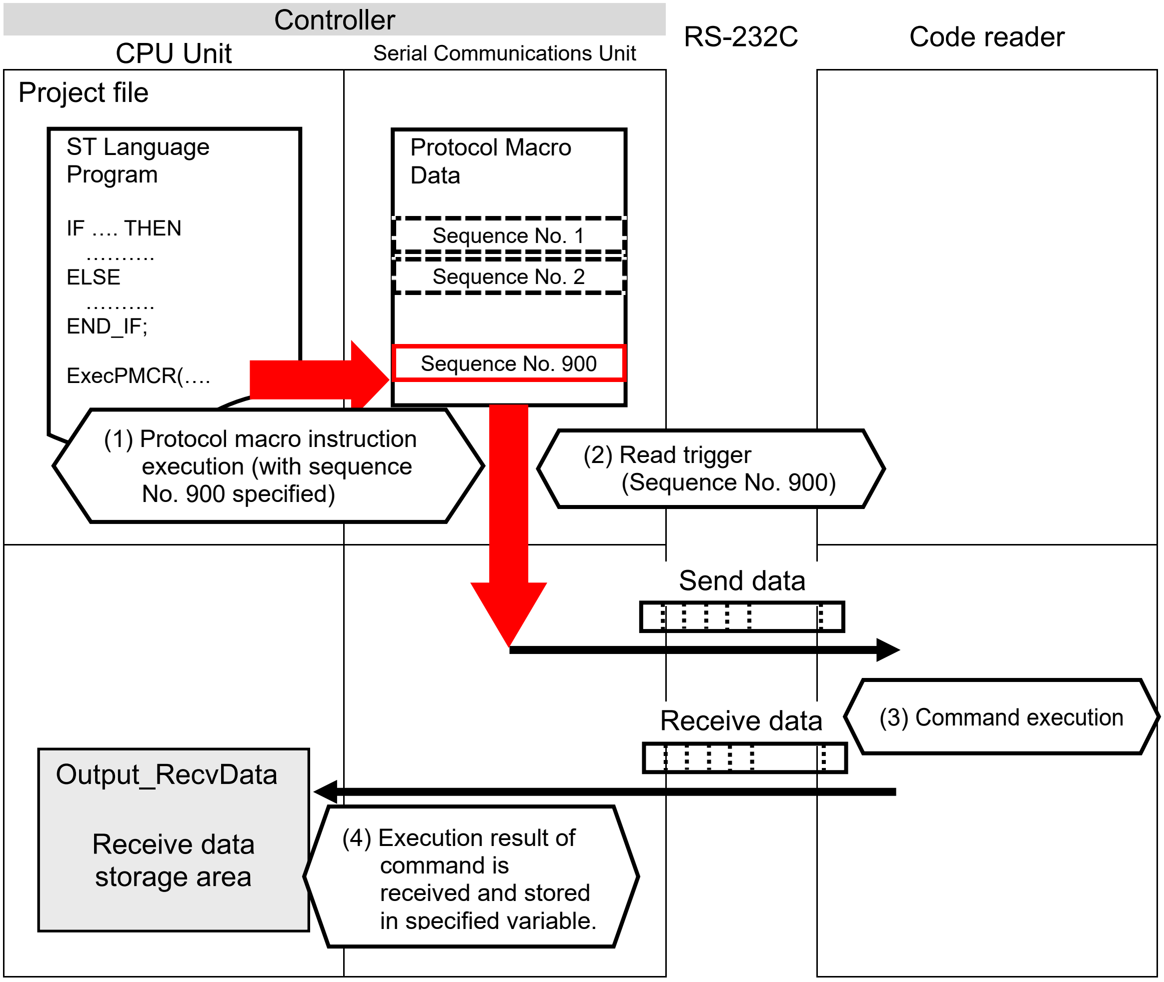
● Sequence Operation Overview

(1) In the ST language program, the protocol macro instruction (ExecPMCR instruction) is executed with sequence No. 900 of the specified protocol macro data.

(2) "Read trigger" command is selected from sequence No. 900 of the protocol macro data and sent to the code reader.

(3) The code reader executes the command according to the data sent from the controller.

(4) The controller receives the execution result of the command from the code reader and stores it in the specified memory address.


Controller
CPU Unit
Serial Communications Unit
RS-232C
Code reader
Project file
ST Language Program
IF …. THEN
……….
ELSE
……….
END_IF;
ExecPMCR(….
Protocol Macro Data
Sequence No. 1
Sequence No. 2
Sequence No. 900
(1) Protocol macro instruction execution (with sequence No. 900 specified)
(2) Read trigger (Sequence No. 900)
Send data
(3) Command execution
Receive data
Output_RecvData
Receive data storage area
(4) Execution result of command is received and stored in specified variable.

7. Connection Procedure

This section describes the procedures for connecting the code reader to the controller for serial communications.

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

Use the following procedures to set up the controller for serial communications connection.

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. Controller Setup	Set up the controller.
7.3.1. Setting the Hardware	Set the hardware switch on the Serial Communications Unit.
7.3.2. Starting the Sysmac Studio and Loading the Project File	Start the Sysmac Studio and load the project file.
7.3.3. Checking Parameters and Executing builds	Check the setting parameters. Then, perform program checks and builds on project data.
7.3.4. Going Online and Transferring the Project Data	Place the Sysmac Studio online and transfer the project data to the controller.
7.3.5. Transferring Unit Settings	Transfer the setting data for the Serial Communications Unit.
7.3.6. Starting the CX-Protocol and Loading the Protocol Macro Data	Start the CX-Protocol and load the protocol macro data.
7.3.7. Going Online and Transferring the Protocol Macro Data	Go online with the CX-Protocol and transfer the protocol macro data to the Serial Communications Unit.
7.4. Checking That Serial Communications Work	Execute the program to check that serial communications work correctly.
7.4.1. Starting a Trace	Start a trace in the CX-Protocol.
7.4.2. Executing the Program	Execute the program in the Sysmac Studio.
7.4.3. Checking the Trace Data	Using the trace data of the CX-Protocol, check that the correct data is sent and received.

7.4.4. Checking the Receive Data

In the Sysmac Studio, check that correct data is written to the controller variables.

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 [Using V320-F/V420-F]

Establish a USB connection between the V320/V420 and the PC according to the configuration shown on page 7 or page 9.
Once the connection is established, start the procedure from step 4.

* To use V320/V420, you need to install the driver. For how to install the driver, refer to *2-2-3 Connect to WebLink* in the *MicroHAWK V320-F/V330-F/V420-F/V430-F Series Barcode Reader User Manual* (Cat. No. Z432).

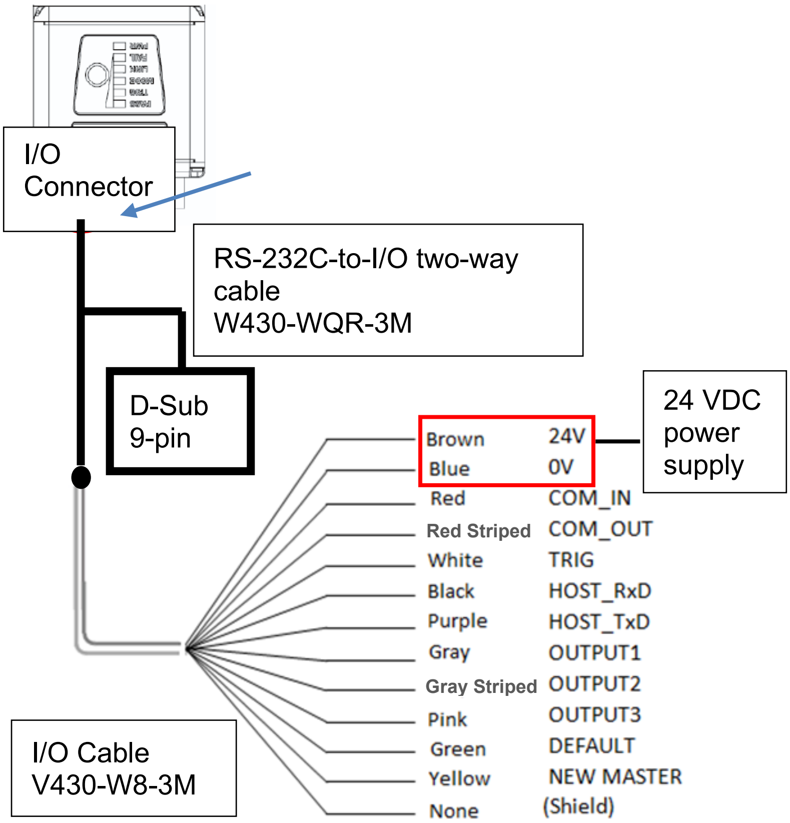
[Using V430-F]

Connect the RS-232C-to-I/O two-way cable V430-WQR-3M to the I/O connector.
Then, connect the I/O connector of V430-WQR-3M to the I/O cable V430-W8-3M.

Connect the 24V and 0V wires of W430-W8-3M to a 24 VDC power supply.

* In this document, only the power supply wires of the I/O cable are connected and checked. Be careful not to short-circuit any other wires.

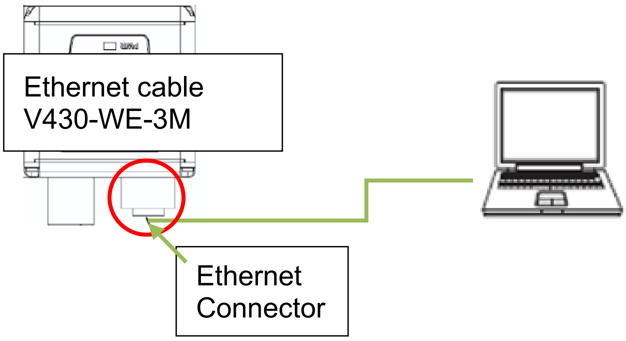
* Ground the shield wire as needed. For more information on grounding, please refer to *Grounding* in *Appendices* of the *MicroHAWK V320-F/V330-F/V420-F/V430-F Series Barcode Reader User Manual* (Cat. No. Z432).



2 [V430-F Series Only]

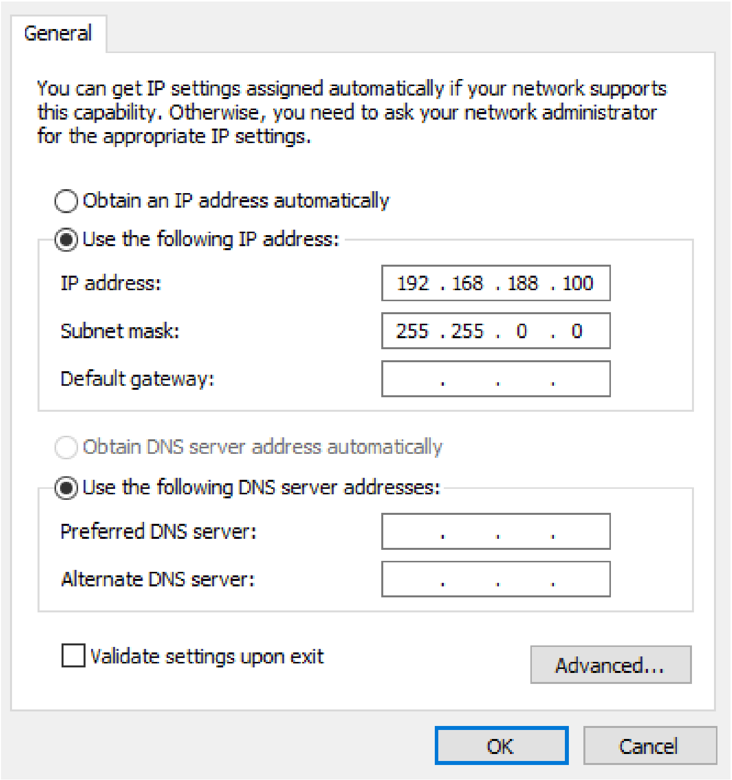
Connect the Ethernet connector on the code reader to the PC using the Ethernet cable V430-WE-3M.

Turn ON the 24 VDC power supply.



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 how to open the screen shown on the right in Windows 7, please refer to step 4.



General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 188 . 100

Subnet mask: 255 . 255 . 0 . 0

Default gateway: . . .

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

□ Validate settings upon exit

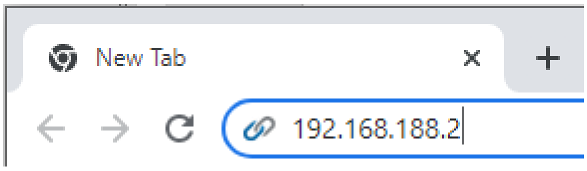
4 (1) From the Windows **Start** Menu, select **Control Panel – Network and Internet – Network and Sharing Center**.

(2) Click on **Local Area Connection**. The **Local Area Connection Status** Dialog Box is displayed. Click **Properties**.

(3) In the **Local Area Connection Properties** Dialog Box, select *Internet Protocol Version 4 (TCP / IPv4)*, and click the **Properties** Button.

(4) Click the **OK** Button.

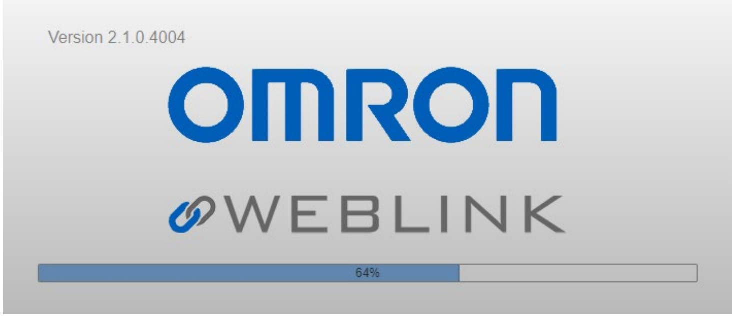
5 Start your browser and enter *http://192.168.188.2*.
“Google Chrome” is the recommended browser.



192.168.188.2

6 When the WebLink startup screen is displayed, go to step 8.

If you cannot access by WebLink, go to step 7.

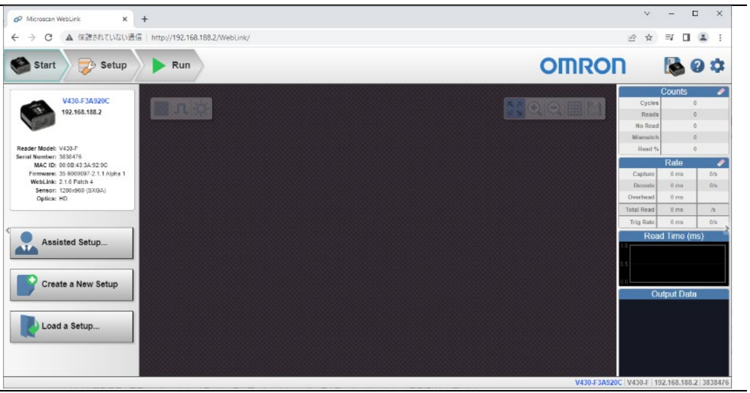


7 If the WebLink startup screen does not appear, it means that communications are not established between the code reader and the PC. Please check the following.

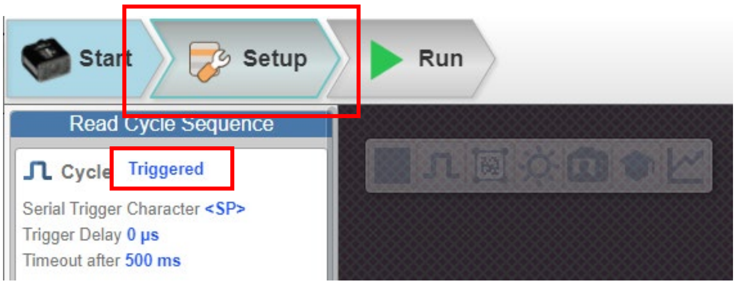
- The code reader and the PC have a proper physical (cable) connection.
  → Refer to steps 1 and 2 for checking the connection.
- The IP Addresses of the PC and code reader are set correctly.
  → Refer to step 4 for setting the IP address of the PC.

For other measures that can be taken, please refer to *When unable to access by WebLink* in *Q&A* in *Appendices* of the *MicroHAWK V320-F/V330-F/V420-F/V430-F Series Barcode Reader User Manual* (Cat. No. Z432).

8 The WebLink screen appears.



9 Click on the **Setup** Tab and, in **Read Cycle Sequence**, set **Cycle** to *Triggered*.



Read Cycle Sequence

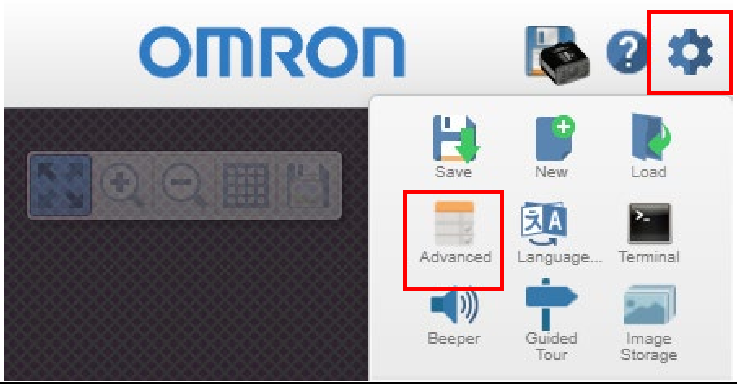
Cycle Triggered

Serial Trigger Character <SP>

Trigger Delay 0 µs

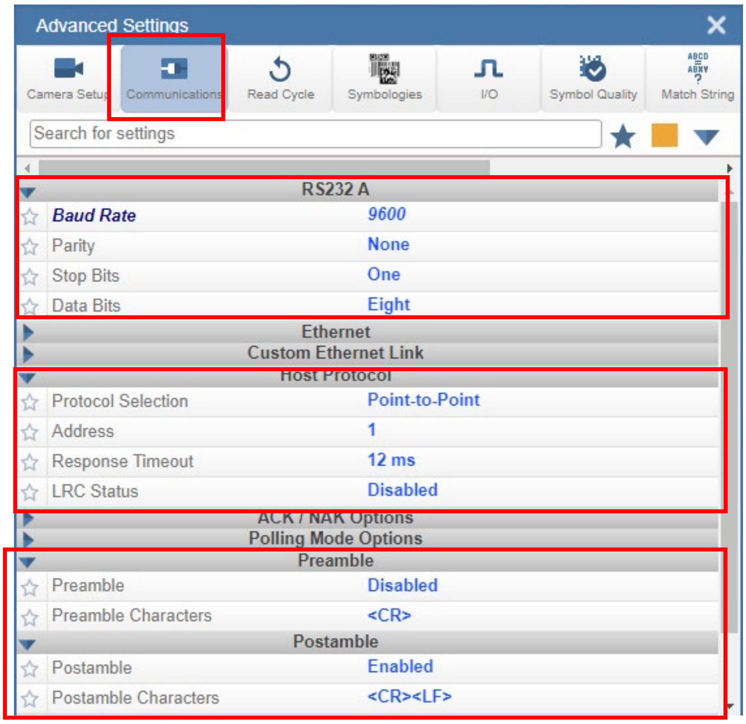
Timeout after 500 ms

10 Click on the gear icon on the upper right of the screen and select **Advanced**.



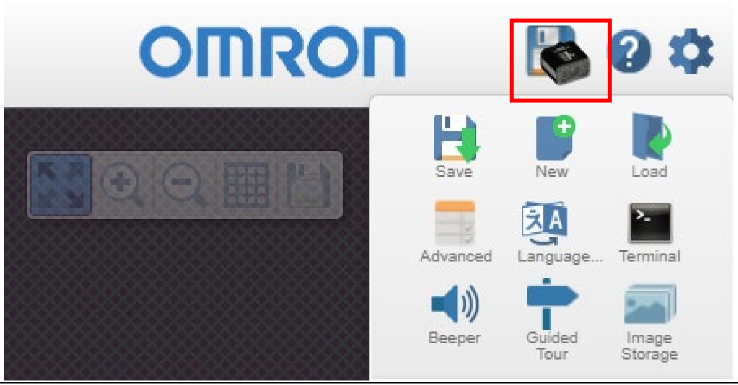
11 The Advanced Settings Screen appears.
Select the **Communications** Tab and check that the settings are as follows.

[RS232A]
Baud Rate: 9600
Parity: None
Stop Bits: One
Data Bits: Eight
[Host Protocol]
Protocol Selection: Point-to-Point
[Preamble]
Preamble: Disabled
[Postamble]
Postamble: Enabled
Postamble Characters: <CR><LF>

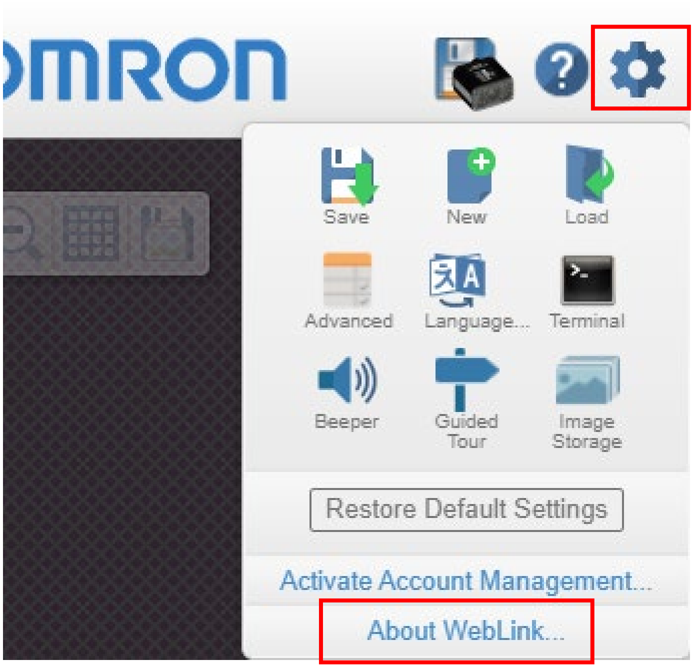


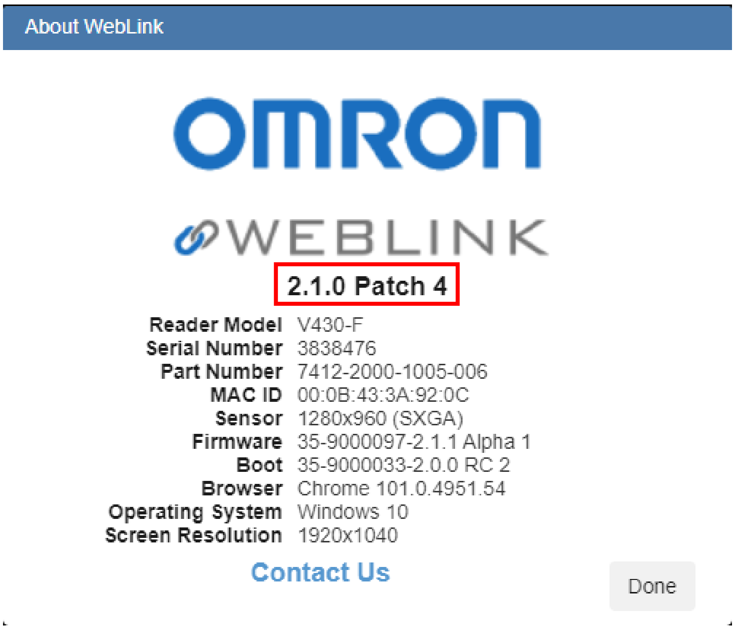
RS232 A	
Baud Rate	9600
Parity	None
Stop Bits	One
Data Bits	Eight
Ethernet	
Custom Ethernet Link	
Host Protocol	
Protocol Selection	Point-to-Point
Address	1
Response Timeout	12 ms
LRC Status	Disabled
ACK / NAK Options	
Polling Mode Options	
Preamble	
Preamble	Disabled
Preamble Characters	<CR>
Postamble	
Postamble	Enabled
Postamble Characters	<CR><LF>

12 Click on the icon shown in the red frame to save the settings to the code reader.



13 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**.



14	**About WebLink** is displayed, so you can check the current version of the code reader. Please update the code reader to the latest version if necessary.	

About WebLink

2.1.0 Patch 4

```
     Reader Model  V430-F
    Serial Number  3838476
      Part Number  7412-2000-1005-006
           MAC ID  00:0B:43:3A:92:0C
           Sensor  1280x960 (SXGA)
         Firmware  35-9000097-2.1.1 Alpha 1
             Boot  35-9000033-2.0.0 RC 2
          Browser  Chrome 101.0.4951.54
 Operating System  Windows 10
Screen Resolution  1920x1040
```

7.3. Controller Setup

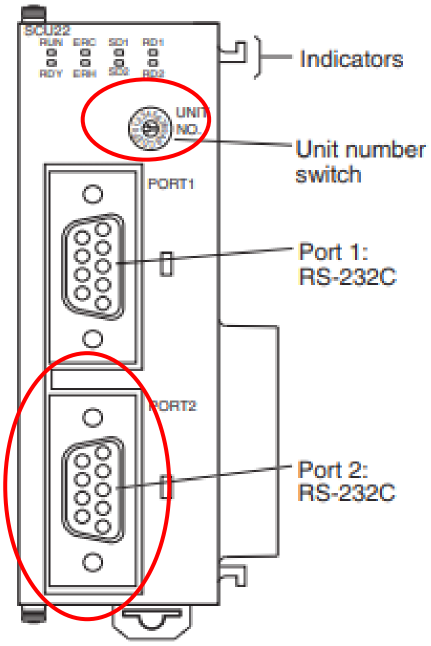
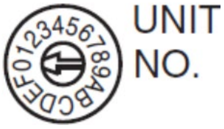
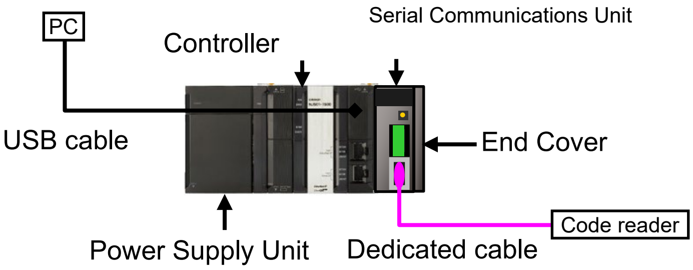
Set up the controller.

7.3.1. Setting the Hardware

Set the hardware switch on the Serial Communications Unit.

Precautions for Correct Use

Turn OFF the power supply before setting the hardware.

1	Confirm that the power supply to the controller is OFF. * If the power supply is ON, you may not be able to proceed with the subsequent steps of the procedure. Check the position of the hardware switch on the front of the Serial Communications Unit as shown in the figure on the right. Check the position of the “Port 2” (RS-232C) connector as shown in the figure on the right. * It is assumed that “Port 2” of the Serial Communications Unit is used.	 SCU22 RUN ERC SD1 RD1 RDY ERH SD2 RD2 UNIT NO. PORT1 PORT2 Indicators Unit number switch Port 1: RS-232C Port 2: RS-232C
2	Set the Unit No. switch to *0*. (The factory default unit number is *0*.)	
3	Connect the Serial Communications Unit to the controller, as shown in the figure on the right. Connect the dedicated cable of the code reader cable to the Serial Communications Unit. Connect the controller to the PC with a USB cable.	 PC Controller Serial Communications Unit USB cable End Cover Code reader Power Supply Unit Dedicated cable

7.3.2. Starting the Sysmac Studio and Loading the Project File

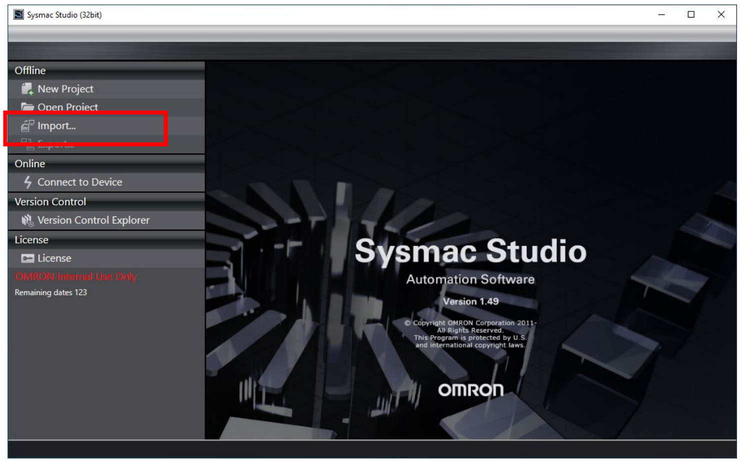
Start the Sysmac Studio and load the project file.

Install the Sysmac Studio and USB driver on the PC beforehand.

1 Make sure that the PC and controller are connected with the USB cable, and turn ON the power supply to the controller.

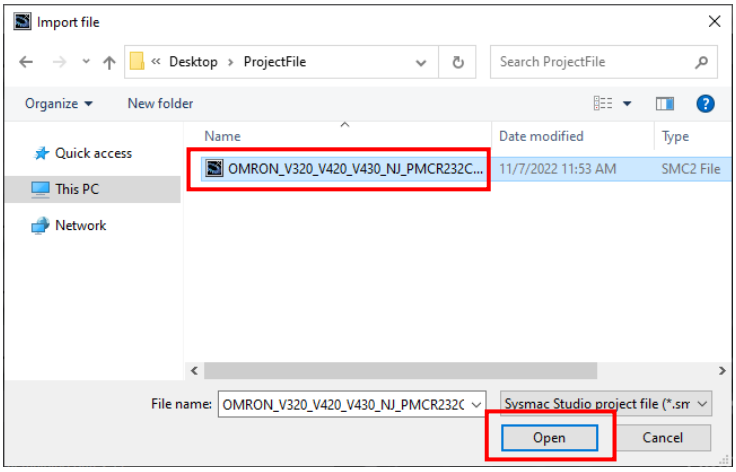
Start the Sysmac Studio and click **Import**.

* If a user account control dialog box is displayed at startup, select the option to start.



2 The **Import file** Dialog Box is displayed. Select **OMRON_V320_V420_V430_NJ_PMCR232C_V100.smc2** and click **Open**.

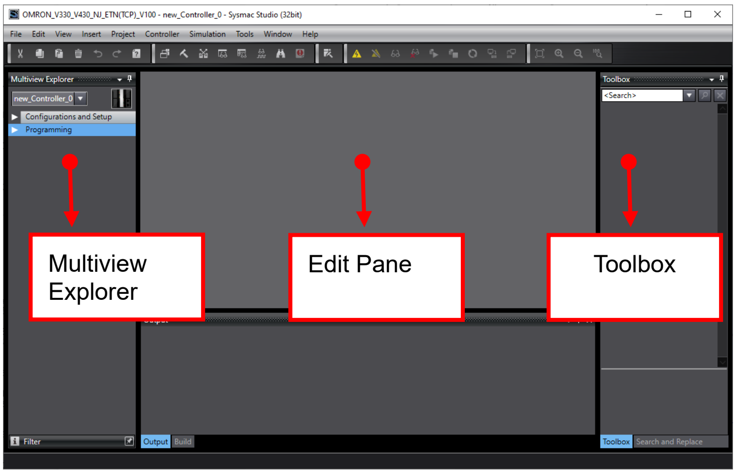
* Obtain the latest version of the project file from the OMRON website.



3 The OMRON_V320_V420_V430_NJ_PMCR232C_V100.smc2 Project Window is displayed.

The window consists of three panes: “Multiview Explorer” on the left side, “Edit Pane” in the center, and “Toolbox” on the right side.

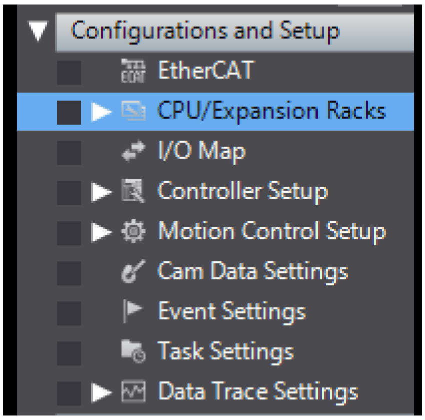
* If the error message *Cannot compare the repositories, the version on the target different from the source.* is displayed, change the Sysmac Studio version to the version indicated in *5.2. Device Configuration* or higher.



7.3.3. Checking Parameters and Executing builds

Check the setting parameters. Then, perform program checks and builds on project data.

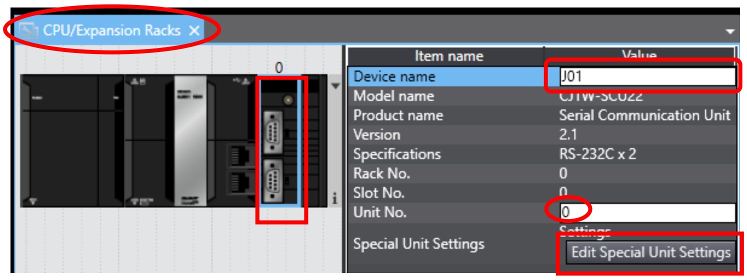
1 Double-click **CPU/Expansion Racks** under **Configurations and Setup** in the Multiview Explorer.



2 The **CPU and Expansion Racks** Tab Page is displayed in the Edit Pane.
Select the Serial Communications Unit in the tab page, as shown in the figure on the right.
Confirm that the settings are as follows: **Model name** is *CJ1W-SCU22*, **Device name** is *J01*, and **Unit No.** is *0*.

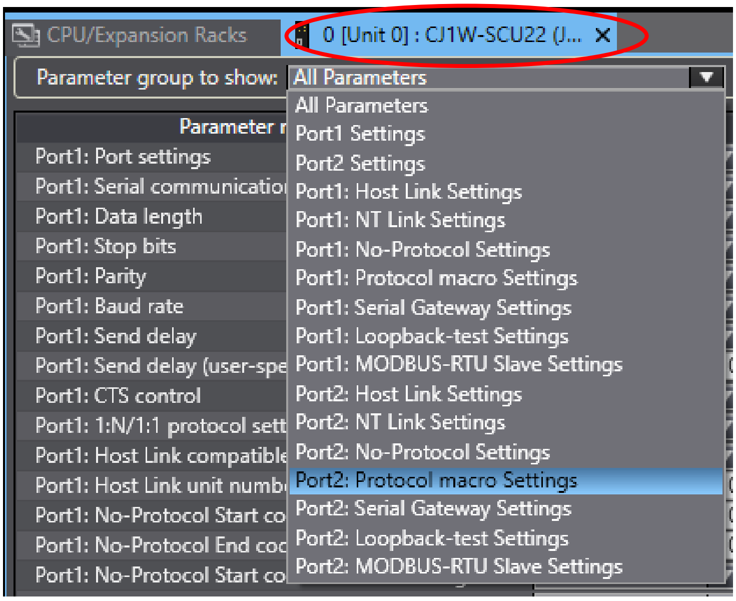
* If the settings are different, change them to be the same as shown.

Click **Edit Special Unit Settings**.



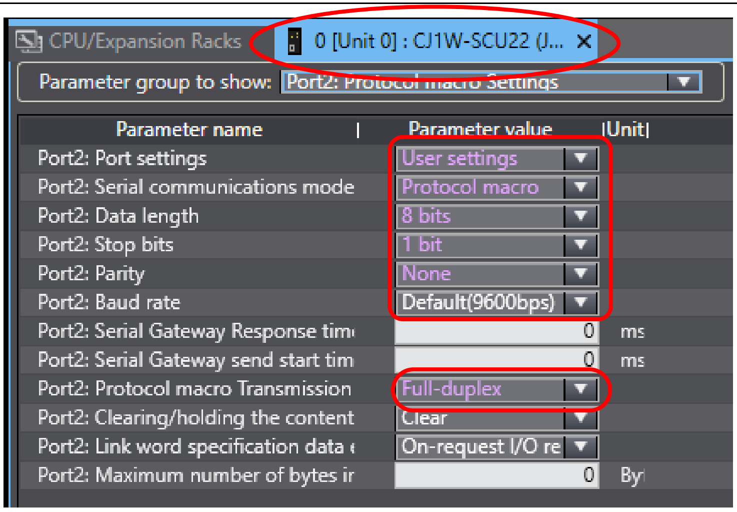
Item name	Value
Device name	J01
Model name	CJ1W-SCU22
Product name	Serial Communication Unit
Version	2.1
Specifications	RS-232C x 2
Rack No.	0
Slot No.	0
Unit No.	0
Special Unit Settings	Settings

3 The **0 [Unit 0]:** Tab Page is displayed.
Select **Port 2: Protocol macro Settings** from the **Parameter group to show** pull-down menu.


CPU/Expansion Racks
0 [Unit 0] : CJ1W-SCU22 (J...
Parameter group to show:
All Parameters
All Parameters
Port1 Settings
Port2 Settings
Port1: Host Link Settings
Port1: NT Link Settings
Port1: No-Protocol Settings
Port1: Protocol macro Settings
Port1: Serial Gateway Settings
Port1: Loopback-test Settings
Port1: MODBUS-RTU Slave Settings
Port2: Host Link Settings
Port2: NT Link Settings
Port2: No-Protocol Settings
Port2: Protocol macro Settings
Port2: Serial Gateway Settings
Port2: Loopback-test Settings
Port2: MODBUS-RTU Slave Settings
Parameter r
Port1: Port settings
Port1: Serial communicatioi
Port1: Data length
Port1: Stop bits
Port1: Parity
Port1: Baud rate
Port1: Send delay
Port1: Send delay (user-spe
Port1: CTS control
Port1: 1:N/1:1 protocol sett
Port1: Host Link compatible
Port1: Host Link unit numb
Port1: No-Protocol Start co
Port1: No-Protocol End coc
Port1: No-Protocol Start co

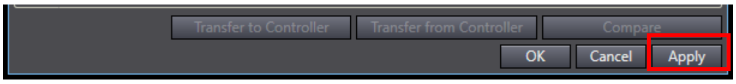
4 **Parameter group to show** is set to **Port 2: Protocol macro Settings**.
The settings of **Port2: Protocol macro Settings** are displayed. Confirm that **Port 2: Port settings** is set to *User settings* and other settings are set to be the same as those listed in section 6.1.

* If the settings are different, change them to be the same by selecting from the pull-down menu. If you change the values, click **Apply**.

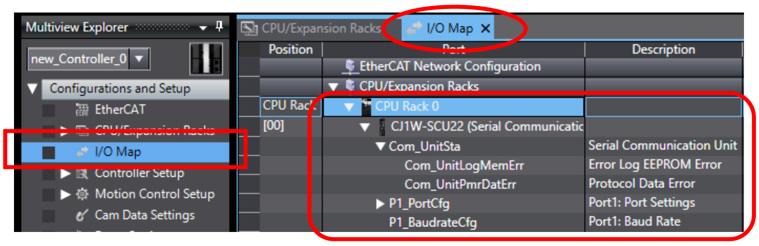


Parameter group to show: Port2: Protocol macro Settings

Parameter name	Parameter value	Unit
Port2: Port settings	User settings	
Port2: Serial communications mode	Protocol macro	
Port2: Data length	8 bits	
Port2: Stop bits	1 bit	
Port2: Parity	None	
Port2: Baud rate	Default(9600bps)	
Port2: Serial Gateway Response tim	0	ms
Port2: Serial Gateway send start tim	0	ms
Port2: Protocol macro Transmission	Full-duplex	
Port2: Clearing/holding the content	Clear	
Port2: Link word specification data	On-request I/O re	
Port2: Maximum number of bytes ir	0	By



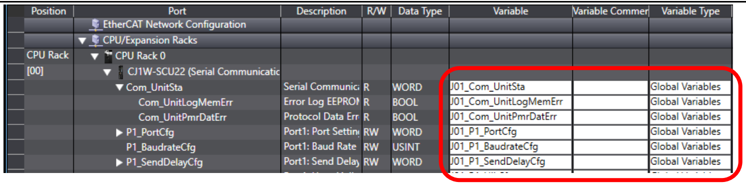
5 Double-click **I/O Map** under **Configurations and Setup** in the Multiview Explorer.
The **I/O Map** Tab Page showing the unit parameters is displayed.



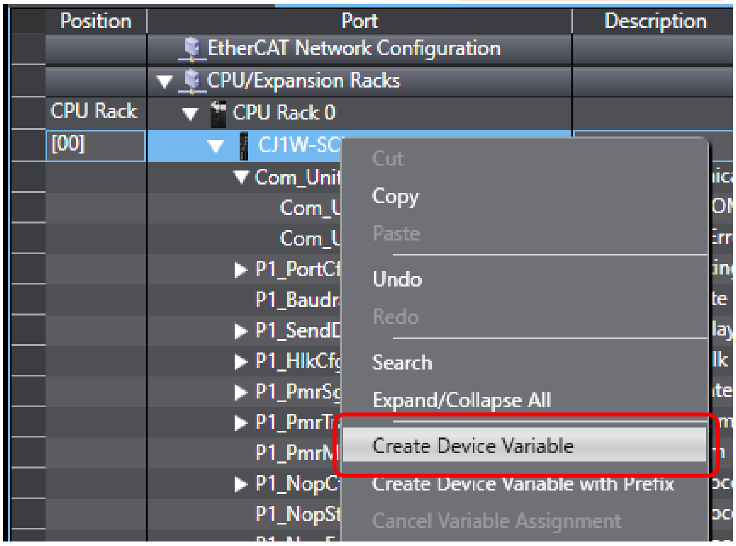
Position	Port	Description
	EtherCAT Network Configuration	
	CPU/Expansion Racks	
CPU Rack	CPU Rack 0	
[00]	CJ1W-SCU22 (Serial Communicatic	
	Com_UnitSta	Serial Communication Unit
	Com_UnitLogMemErr	Error Log EEPROM Error
	Com_UnitPmrDatErr	Protocol Data Error
	P1_PortCfg	Port1: Port Settings
	P1_BaudrateCfg	Port1: Baud Rate

6 In the **I/O Map** Tab Page, confirm that the **Variable** column shows variable names beginning with *J01* and the **Variable Type** column shows *Global Variables*.

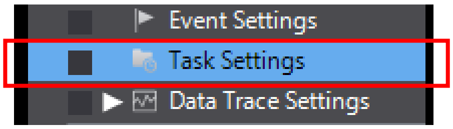
* If the settings are different, right-click on **CJ1W-SCU22** and select **Create Device Variable** from the menu, as shown in the figure on the right.



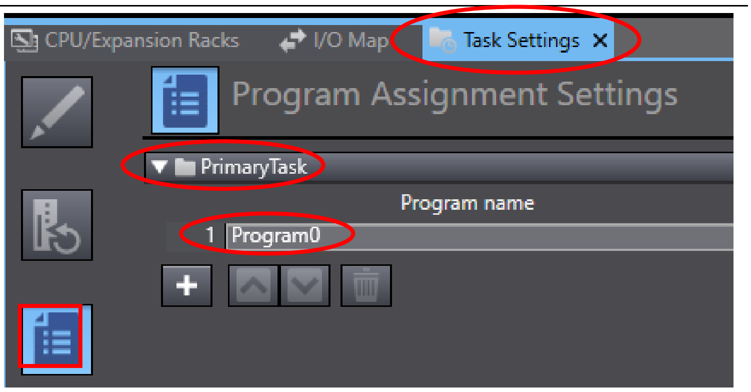
Position	Port	Description	R/W	Data Type	Variable	Variable Commen	Variable Type
	EtherCAT Network Configuration						
	CPU/Expansion Racks						
CPU Rack	CPU Rack 0						
[00]	CJ1W-SCU22 (Serial Communicatic						
	Com_UnitSta	Serial Communic	R	WORD	J01_Com_UnitSta		Global Variables
	Com_UnitLogMemErr	Error Log EEPROM	R	BOOL	J01_Com_UnitLogMemErr		Global Variables
	Com_UnitPmrDatErr	Protocol Data Err	R	BOOL	J01_Com_UnitPmrDatErr		Global Variables
	P1_PortCfg	Port1: Port Settin	RW	WORD	J01_P1_PortCfg		Global Variables
	P1_BaudrateCfg	Port1: Baud Rate	RW	USINT	J01_P1_BaudrateCfg		Global Variables
	P1_SendDelayCfg	Port1: Send Delay	RW	WORD	J01_P1_SendDelayCfg		Global Variables



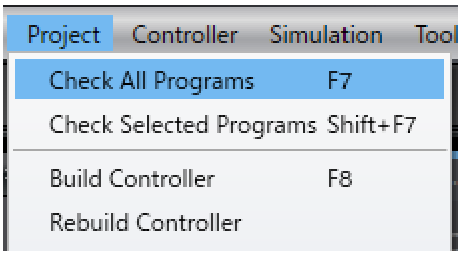
7 Double-click **Task Settings** under **Configurations and Setup** in the Multiview Explorer.



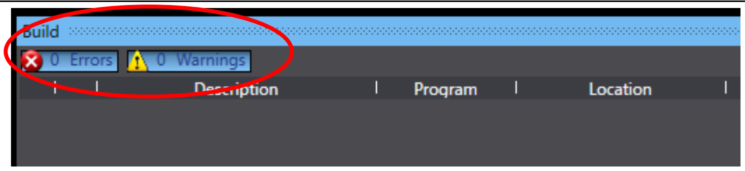
8 The Task Settings Tab Page is displayed in Edit Pane. Select **Program Assignment Settings** and confirm that *Program0* is set in **PrimaryTask**.



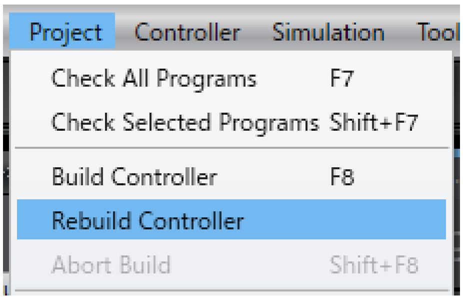
9 Select **Check All Programs** from the **Project** Menu.



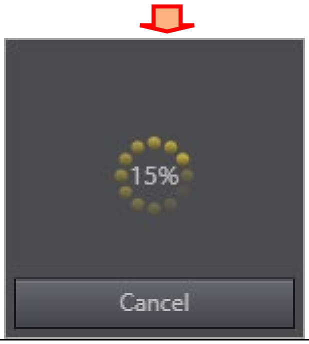
10 The Build Tab Page is displayed under the Edit Pane. Confirm that *0* is shown for both **Errors** and **Warnings**.



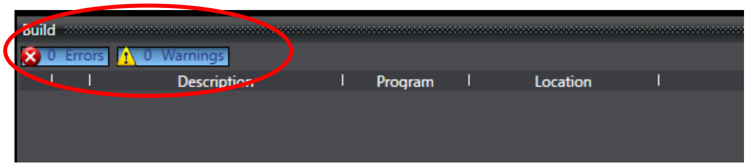
11 Select **Rebuild Controller** from the **Project** Menu.



A dialog box showing the progress of conversion appears.



12 In the **Build** Tab Page, confirm that *0* is shown for both **Errors** and **Warnings**.



7.3.4. Going Online and Transferring the Project Data

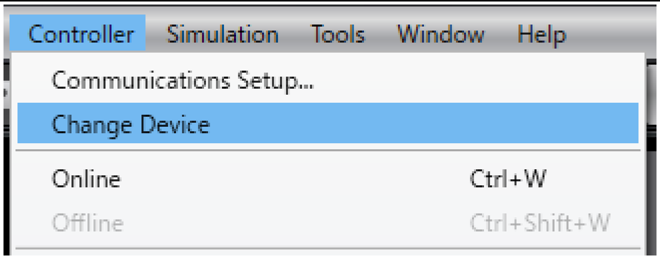
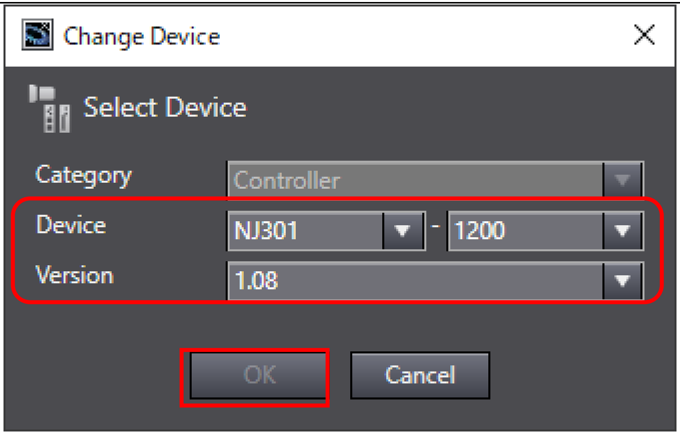
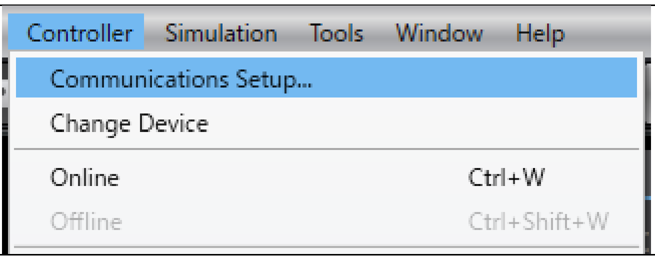
Place the Sysmac Studio online and transfer the project data to the controller.

⚠ **WARNING**

Always confirm safety at the destination node before you transfer a user program, configuration and setup data, device variables, or values in memory used for CJ-series Units from the Sysmac Studio .
The devices or machines may perform unexpected operation regardless of the operating mode of the CPU Unit.

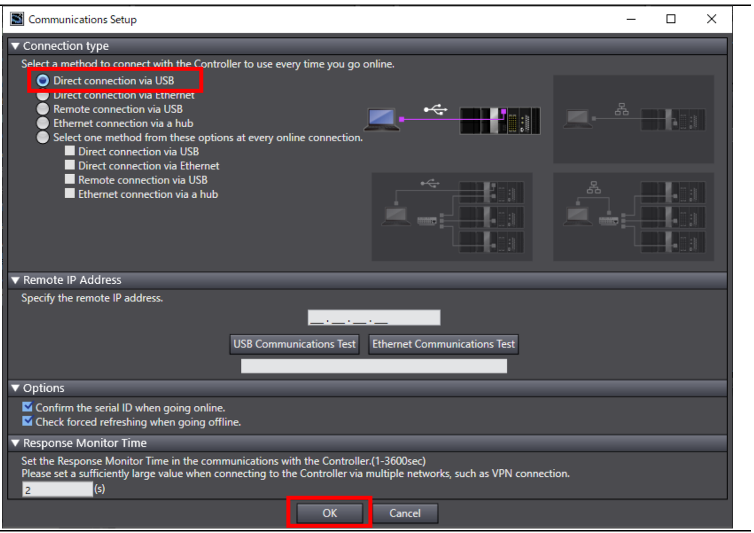
⚠ **Caution**

Be sure to confirm safety before resetting the controller or unit.

1	Select **Change Device** from the **Controller** Menu.	 Controller Simulation Tools Window Help Communications Setup... Change Device Online Ctrl+W Offline Ctrl+Shift+W
2	The **Change Device** Dialog Box is displayed. Confirm that **Device** and **Version** are set to use the controller as shown in the figure on the right, and click **OK**. * If the settings are different, set them to be the same by selecting from the pull-down menu.	 Category: Controller Device: NJ301 - 1200 Version: 1.08
3	Select **Communications Setup** from the **Controller** Menu.	 Controller Simulation Tools Window Help Communications Setup... Change Device Online Ctrl+W Offline Ctrl+Shift+W

4 The Communications Setup Dialog Box is displayed.
In **Connection type**, select the **Direct connection via USB** Option.

Click **OK**.



Communications Setup

▼ Connection type

Select a method to connect with the Controller to use every time you go online.

● Direct connection via USB
○ Direct connection via Ethernet
○ Remote connection via USB
○ Ethernet connection via a hub
○ Select one method from these options at every online connection.
□ Direct connection via USB
□ Direct connection via Ethernet
□ Remote connection via USB
□ Ethernet connection via a hub

▼ Remote IP Address

Specify the remote IP address.

__.__.__.__

▼ Options

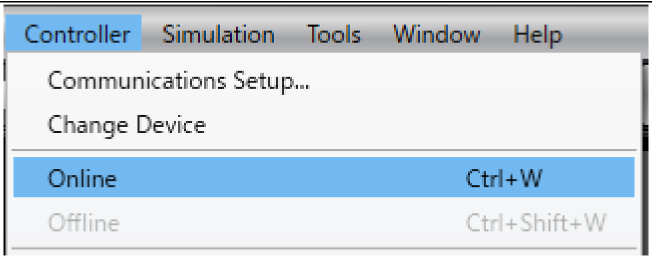
☑ Confirm the serial ID when going online.
☑ Check forced refreshing when going offline.

▼ Response Monitor Time

Set the Response Monitor Time in the communications with the Controller.(1-3600sec)
Please set a sufficiently large value when connecting to the Controller via multiple networks, such as VPN connection.

2 (s)

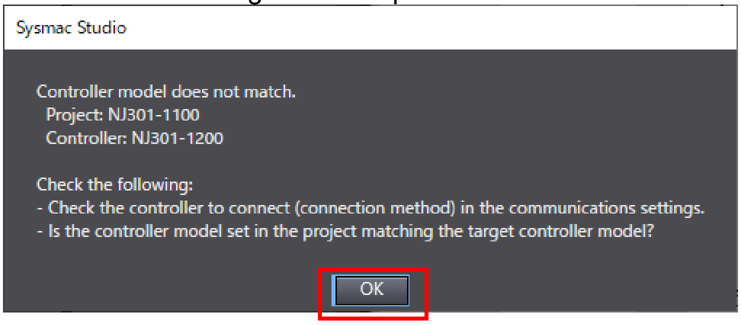
5 Select **Online** from the **Controller** Menu.



* If a confirmation dialog box as shown on the right is displayed, the controller model or version is different from the device settings in the project file. Review the controller model and version, and the device settings in the project file, return to step 1, and execute the procedure in this section again.
Click **OK** to close the dialog box.

* The model and version displayed in the confirmation dialog box differ depending on the controller used and the device settings in the project file.

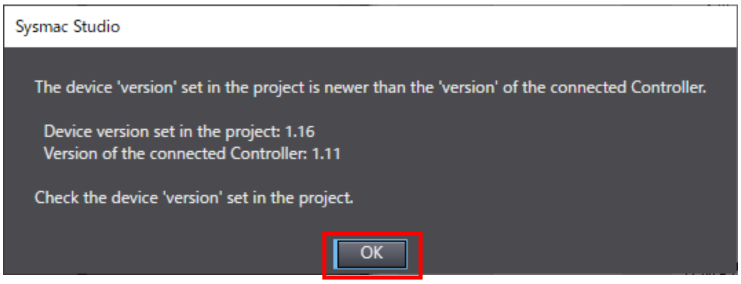
* Confirmation dialog box examples



Sysmac Studio

Controller model does not match.
Project: NJ301-1100
Controller: NJ301-1200

Check the following:
- Check the controller to connect (connection method) in the communications settings.
- Is the controller model set in the project matching the target controller model?



Sysmac Studio

The device 'version' set in the project is newer than the 'version' of the connected Controller.

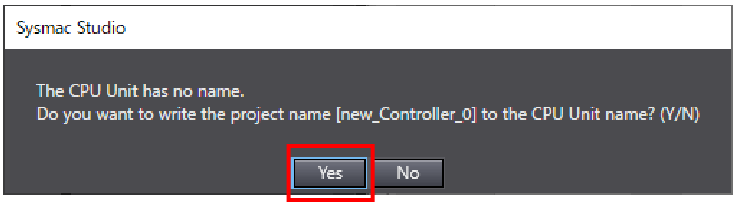
Device version set in the project: 1.16
Version of the connected Controller: 1.11

Check the device 'version' set in the project.

6 A confirmation dialog box as shown in the figure on the right appears. Click **Yes**.

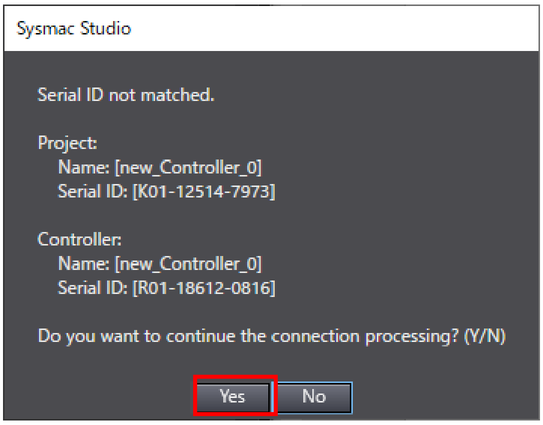
* The dialog box displayed differs depending on the status of the controller being used. Select **Yes** to proceed with the operation.

* The serial IDs displayed vary depending on the device.



Sysmac Studio

The CPU Unit has no name.
Do you want to write the project name [new_Controller_0] to the CPU Unit name? (Y/N)



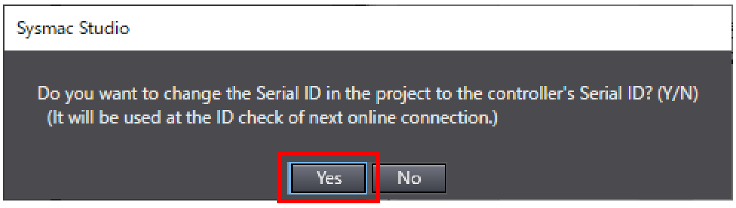
Sysmac Studio

Serial ID not matched.

Project:
Name: [new_Controller_0]
Serial ID: [K01-12514-7973]

Controller:
Name: [new_Controller_0]
Serial ID: [R01-18612-0816]

Do you want to continue the connection processing? (Y/N)



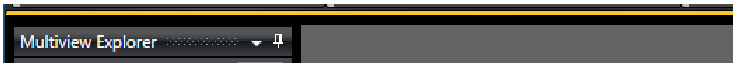
Sysmac Studio

Do you want to change the Serial ID in the project to the controller's Serial ID? (Y/N)
(It will be used at the ID check of next online connection.)

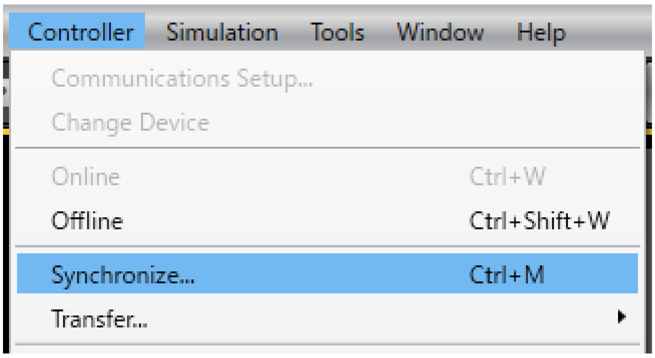
Note

Refer to *Section 5 Online Connections to a Controller* in the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details on online connection to the controller.

7 When you are online, a yellow border appears in the upper part of the Edit Pane.

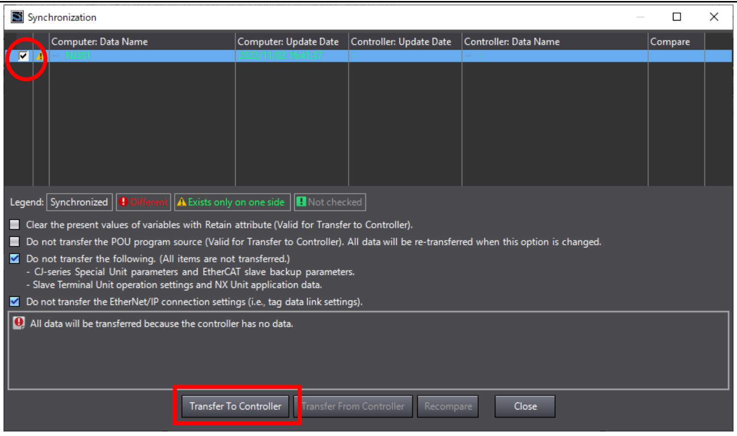


8 Select **Synchronize** from the **Controller** Menu.



9 The **Synchronization** Dialog Box is displayed.
Confirm that the check box for the data to transfer (i.e., **NJ501** on the figure on the right) is selected, and click **Transfer to Controller**.

* Executing **Transfer to Controller** transfers the project data from the Sysmac Studio to the controller for data comparison.



Synchronization

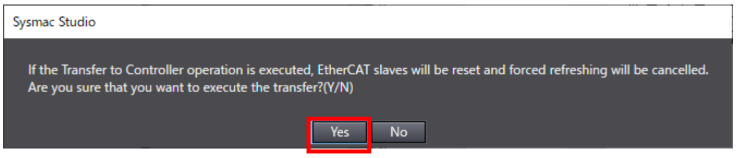
	Computer: Data Name	Computer: Update Date	Controller: Update Date	Controller: Data Name	Compare
☑	NJ301	2022/11/02 16:41:37	-	-	

Legend: Synchronized | Different | Exists only on one side | Not checked

☐ Clear the present values of variables with Retain attribute (Valid for Transfer to Controller).
☐ Do not transfer the POU program source (Valid for Transfer to Controller). All data will be re-transferred when this option is changed.
☑ Do not transfer the following. (All items are not transferred.)
- CJ-series Special Unit parameters and EtherCAT slave backup parameters.
- Slave Terminal Unit operation settings and NX Unit application data.
☑ Do not transfer the EtherNet/IP connection settings (i.e., tag data link settings).

All data will be transferred because the controller has no data.

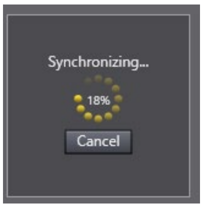
10 A confirmation dialog box appears. Click **Yes**.



Sysmac Studio

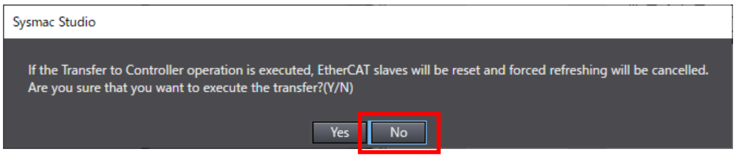
If the Transfer to Controller operation is executed, EtherCAT slaves will be reset and forced refreshing will be cancelled.
Are you sure that you want to execute the transfer?(Y/N)

The Synchronizing Dialog Box appears.



A confirmation dialog box appears. Click **No**.

* Select **No** here since you will perform the next operation in PROGRAM mode.

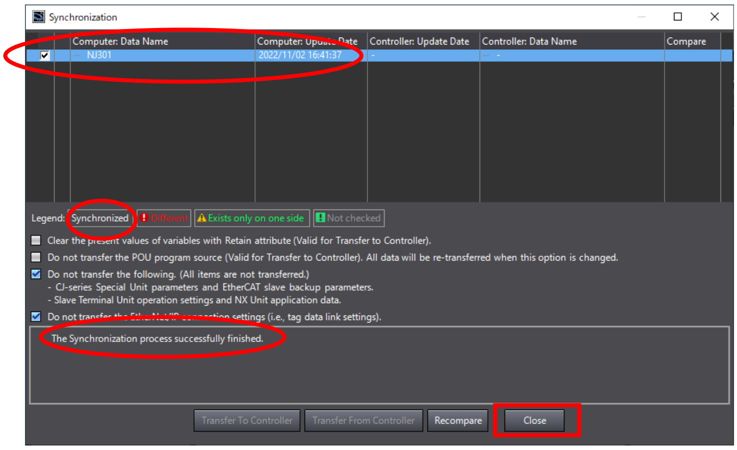


Sysmac Studio

If the Transfer to Controller operation is executed, EtherCAT slaves will be reset and forced refreshing will be cancelled.
Are you sure that you want to execute the transfer?(Y/N)

11 Confirm that the synchronized data is now shown in the text color of **Synchronized** and the following message is displayed:
The Synchronization process successfully finished.
If there is no problem, click **Close**.

* If synchronization fails, check the physical connections and redo the procedure.



Synchronization

	Computer: Data Name	Computer: Update Date	Controller: Update Date	Controller: Data Name	Compare
☑	NJ301	2022/11/02 16:41:37	-	-	

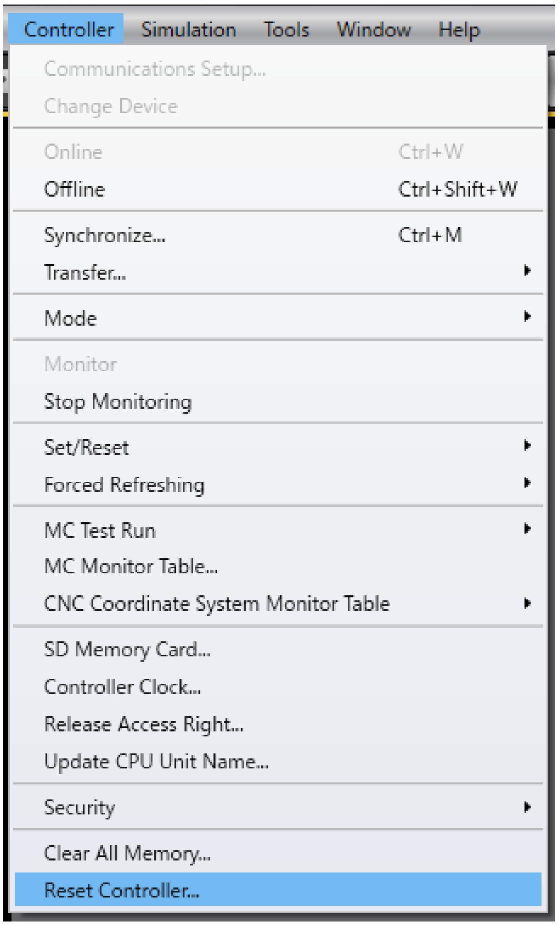
Legend: Synchronized | Different | Exists only on one side | Not checked

☐ Clear the present values of variables with Retain attribute (Valid for Transfer to Controller).
☐ Do not transfer the POU program source (Valid for Transfer to Controller). All data will be re-transferred when this option is changed.
☑ Do not transfer the following. (All items are not transferred.)
- CJ-series Special Unit parameters and EtherCAT slave backup parameters.
- Slave Terminal Unit operation settings and NX Unit application data.
☑ Do not transfer the EtherNet/IP connection settings (i.e., tag data link settings).

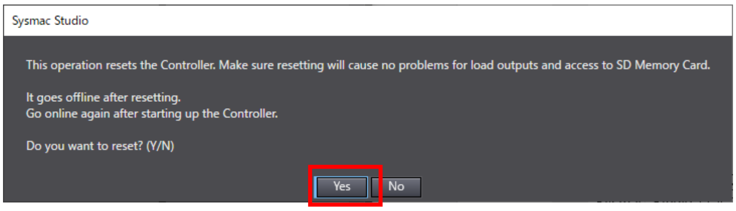
The Synchronization process successfully finished.

12 Select **Reset Controller** from the **Controller** Menu.

* If **Reset Controller** is grayed out, the operating mode is RUN mode. Select **Mode – PROGRAM** from the **Controller** Menu to change to PROGRAM mode, and then perform the procedure in this section.



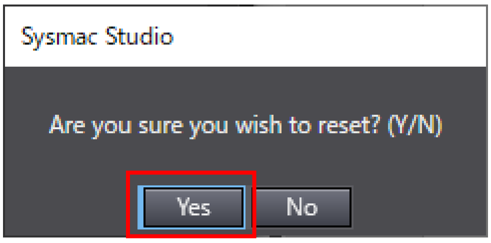
13 A series of confirmation dialog boxes appear. Click **Yes** to proceed.


Sysmac Studio

This operation resets the Controller. Make sure resetting will cause no problems for load outputs and access to SD Memory Card.

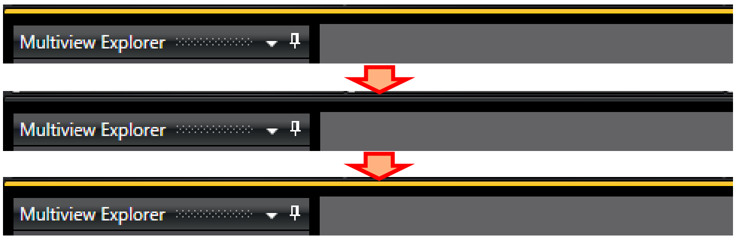
It goes offline after resetting.
Go online again after starting up the Controller.

Do you want to reset? (Y/N)


Sysmac Studio

Are you sure you wish to reset? (Y/N)

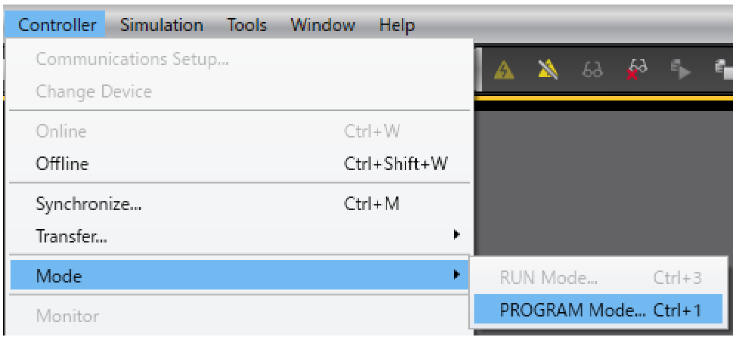
14 The controller is reset. The Sysmac Studio goes offline and the yellow border in the upper part of the Edit Pane disappears.
Perform steps 6 to 8 to go online again.



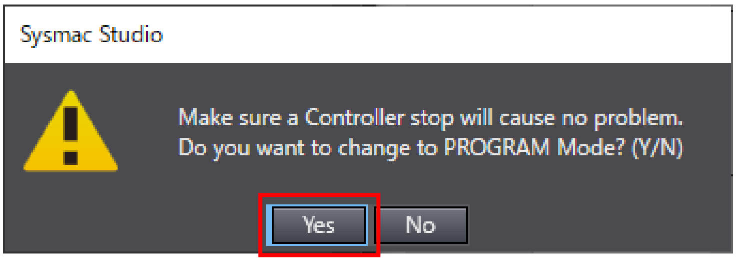
7.3.5. Transferring Unit Settings

Transfer the setting data for the Serial Communications Unit.

1 Select **Mode – PROGRAM Mode** from the **Controller** Menu.

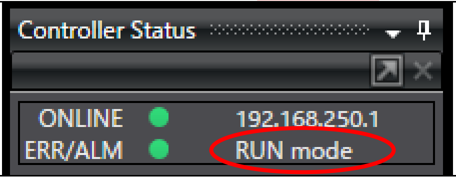


2 A confirmation dialog box appears. Click **Yes**.


Sysmac Studio

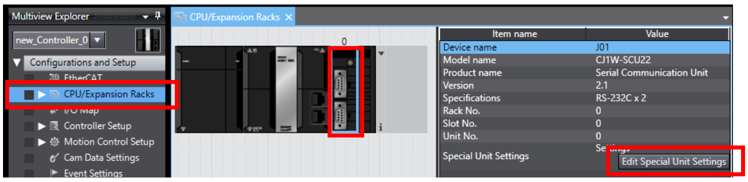
Make sure a Controller stop will cause no problem.
Do you want to change to PROGRAM Mode? (Y/N)

3 In **Controller Status**, **PROGRAM mode** is shown.


Controller Status

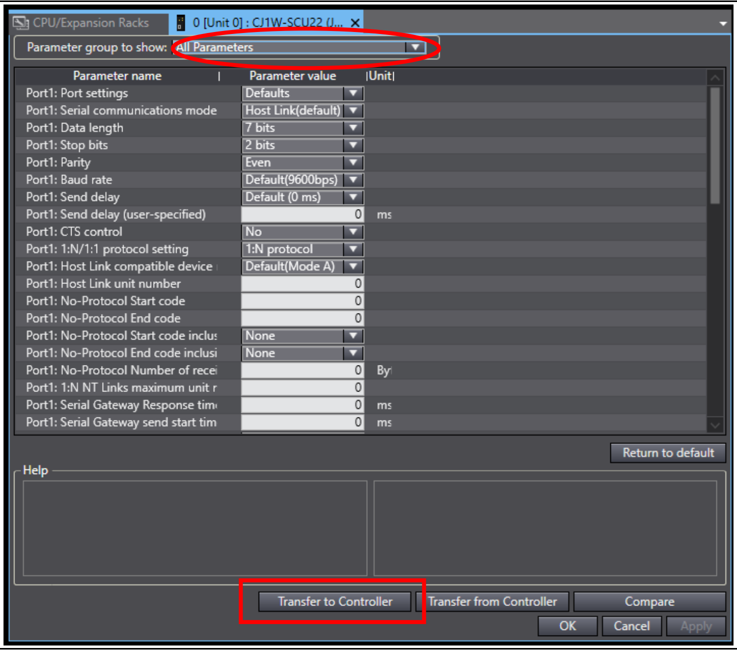
ONLINE ● 192.168.250.1
ERR/ALM ● RUN mode

4 Double-click **CPU/Expansion Racks** under **Configurations and Setup** in the Multiview Explorer.
Select the Serial Communications Unit in the tab page.
Click **Edit Special Unit Settings**.



Item name	Value
Device name	J01
Model name	CJ1W-SCU22
Product name	Serial Communication Unit
Version	2.1
Specifications	RS-232C x 2
Rack No.	0
Slot No.	0
Unit No.	0
Special Unit Settings	Settings

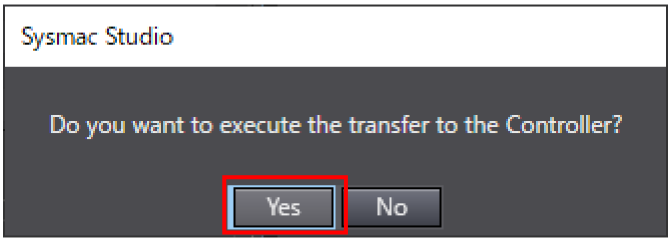
5 The **0 [Unit 0]:** Tab Page is displayed.
Click the **Transfer to Controller** Button.



Parameter group to show: All Parameters

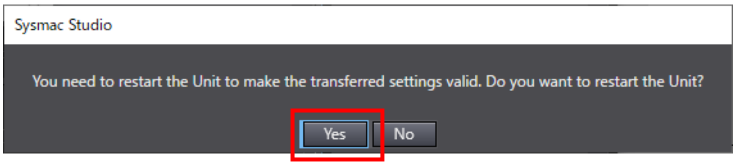
Parameter name	Parameter value	Unit
Port1: Port settings	Defaults	
Port1: Serial communications mode	Host Link(default)	
Port1: Data length	7 bits	
Port1: Stop bits	2 bits	
Port1: Parity	Even	
Port1: Baud rate	Default(9600bps)	
Port1: Send delay	Default (0 ms)	
Port1: Send delay (user-specified)	0	ms
Port1: CTS control	No	
Port1: 1:N/1:1 protocol setting	1:N protocol	
Port1: Host Link compatible device	Default(Mode A)	
Port1: Host Link unit number	0	
Port1: No-Protocol Start code	0	
Port1: No-Protocol End code	0	
Port1: No-Protocol Start code inclus	None	
Port1: No-Protocol End code inclusi	None	
Port1: No-Protocol Number of recei	0	By
Port1: 1:N NT Links maximum unit r	0	
Port1: Serial Gateway Response tim	0	ms
Port1: Serial Gateway send start tim	0	ms

6 A confirmation dialog box is displayed.
Click **Yes**.


Sysmac Studio

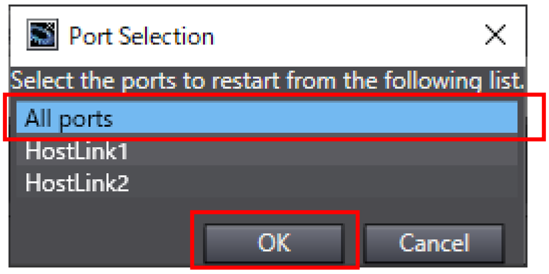
Do you want to execute the transfer to the Controller?

After the Transferring Dialog Box, a confirmation dialog is displayed.
Click **Yes**.


Sysmac Studio

You need to restart the Unit to make the transferred settings valid. Do you want to restart the Unit?

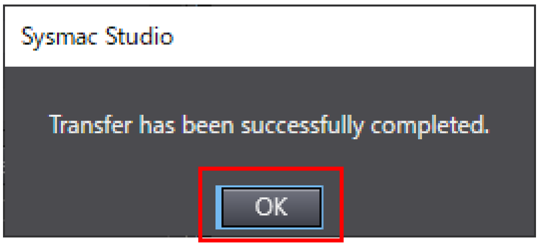
7 The Port Selection Dialog Box is displayed.
Select **All ports** and click **OK**.


Port Selection

Select the ports to restart from the following list.

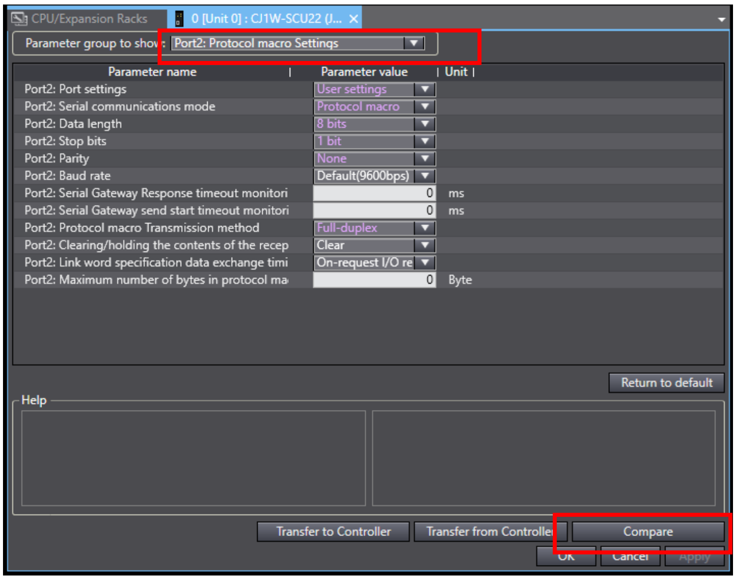
- All ports
- HostLink1
- HostLink2

8 A confirmation dialog box is displayed.
Click **OK**.


Sysmac Studio

Transfer has been successfully completed.

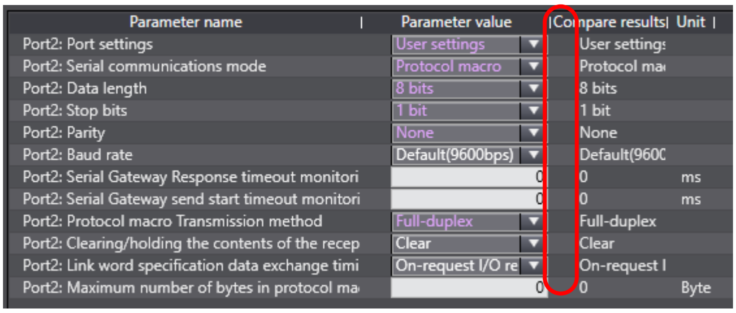
9 Open the **Parameter group to show** pull-down menu and select **Port 2: Protocol macro Settings**.
Click **Compare**.



Parameter group to show: Port2: Protocol macro Settings

Parameter name	Parameter value	Unit
Port2: Port settings	User settings	
Port2: Serial communications mode	Protocol macro	
Port2: Data length	8 bits	
Port2: Stop bits	1 bit	
Port2: Parity	None	
Port2: Baud rate	Default(9600bps)	
Port2: Serial Gateway Response timeout monitori	0	ms
Port2: Serial Gateway send start timeout monitori	0	ms
Port2: Protocol macro Transmission method	Full-duplex	
Port2: Clearing/holding the contents of the recep	Clear	
Port2: Link word specification data exchange timi	On-request I/O re	
Port2: Maximum number of bytes in protocol ma	0	Byte

10 Check that ≠ (mismatch) is not shown in the red frame in the figure on the right.

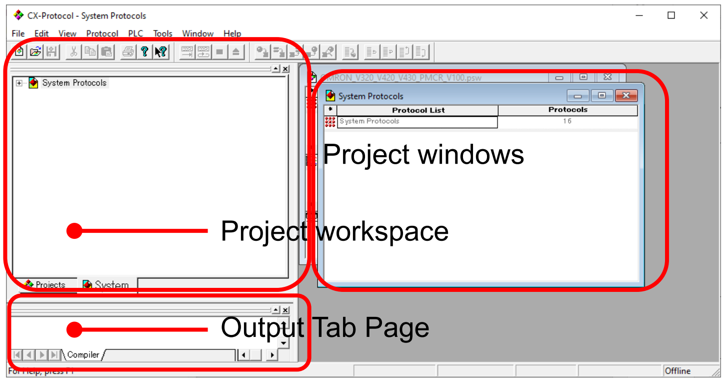


Parameter name	Parameter value	Compare results		Unit
Port2: Port settings	User settings ▼		User setting:	
Port2: Serial communications mode	Protocol macro ▼		Protocol ma	
Port2: Data length	8 bits ▼		8 bits	
Port2: Stop bits	1 bit ▼		1 bit	
Port2: Parity	None ▼		None	
Port2: Baud rate	Default(9600bps) ▼		Default(960C	
Port2: Serial Gateway Response timeout monitori	0		0	ms
Port2: Serial Gateway send start timeout monitori	0		0	ms
Port2: Protocol macro Transmission method	Full-duplex ▼		Full-duplex	
Port2: Clearing/holding the contents of the recep	Clear ▼		Clear	
Port2: Link word specification data exchange timi	On-request I/O re ▼		On-request I	
Port2: Maximum number of bytes in protocol ma	0		0	Byte

7.3.6. Starting the CX-Protocol and Loading the Protocol Macro Data

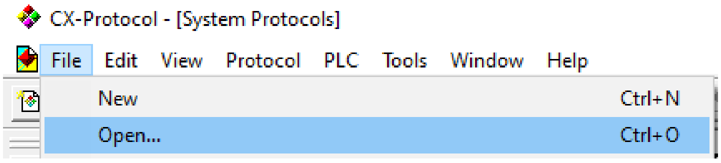
Start the CX-Protocol and load the protocol macro data.

1 Start the CX-Protocol.



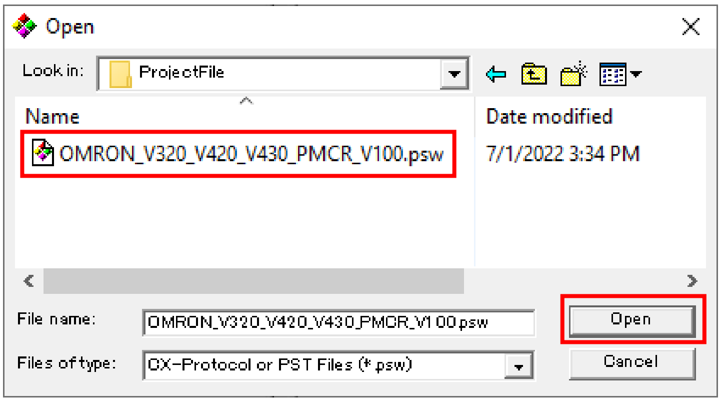
(CX-Protocol)

2 Select **Open** from the **File** Menu.

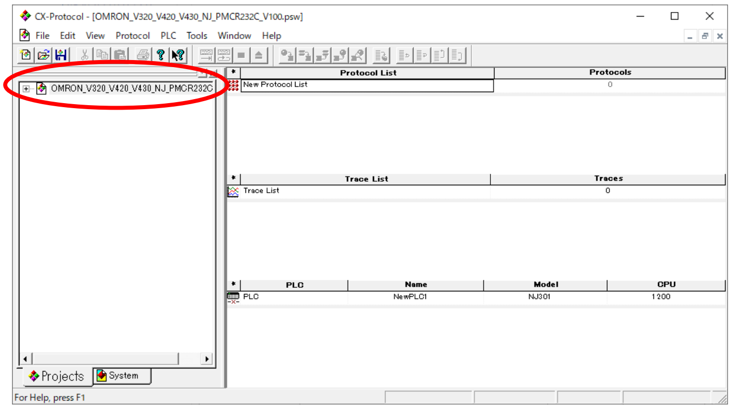

CX-Protocol - [System Protocols]
File Edit View Protocol PLC Tools Window Help
New Ctrl+N
Open... Ctrl+O

3 The Open Dialog Box is displayed. Select **OMRON_V320_V420_V430_PMCR_V100.psw** and click **Open**.

* Obtain the latest version of the protocol macro data from the OMRON website.


Open
Look in:
ProjectFile
Name
Date modified
OMRON_V320_V420_V430_PMCR_V100.psw
7/1/2022 3:34 PM
File name:
OMRON_V320_V420_V430_PMCR_V100.psw
Files of type:
CX-Protocol or PST Files (*.psw)
Open
Cancel

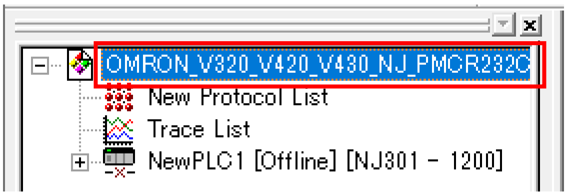
4 The protocol macro data is loaded and displayed in the project workspace and project windows.


CX-Protocol - [OMRON_V320_V420_V430_NJ_PMCR232C_V100.psw]
File Edit View Protocol PLC Tools Window Help
OMRON_V320_V420_V430_NJ_PMCR232C
Protocol List
Protocols
New Protocol List
0
Trace List
Traces
Trace List
0
PLC
Name
Model
CPU
PLC
NewPLC1
NJ301
1200
Projects
System
For Help, press F1

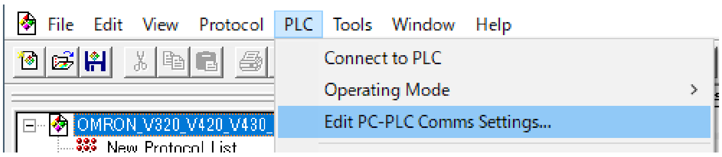
7.3.7. Going Online and Transferring the Protocol Macro Data

Go online with the CX-Protocol and transfer the protocol macro data to the Serial Communications Unit.

1 Double-click **OMRON_V320_V420_V430_PMCR_V100** in the project workspace to open the tree.

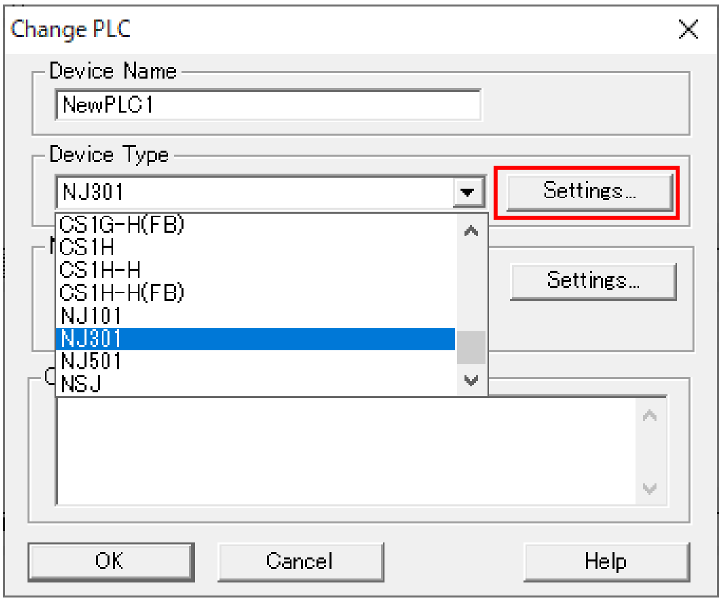

OMRON_V320_V420_V430_NJ_PMCR232C
New Protocol List
Trace List
NewPLC1 [Offline] [NJ301 - 1200]

2 Select **Edit PC-PLC Comms Settings** from the **PLC** Menu.



3 The **Change PLC** Dialog Box is displayed. Select the controller model from the **Device Type** pull-down menu, and click **Settings**.

* In this document, *NJ301* is used.

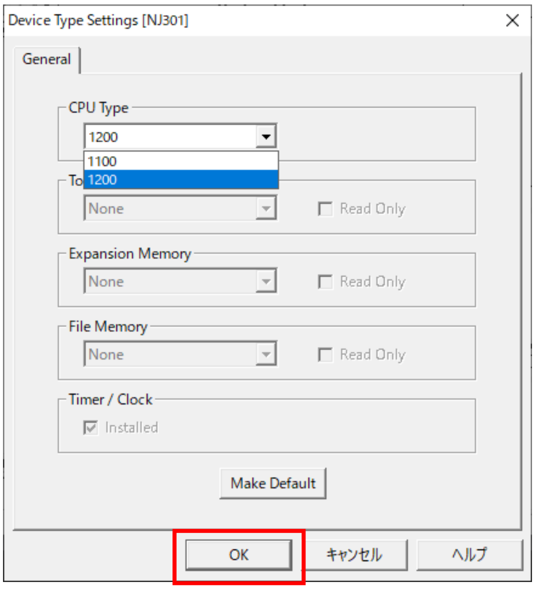

Change PLC

Device Name: NewPLC1

Device Type: NJ301

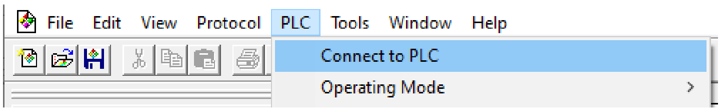
4 The **Device Type Settings** Dialog Box is displayed. Select the CPU type from the **CPU Type** pull-down menu, and click **OK**.

* In this document, *1200* is used.


Device Type Settings [NJ301]

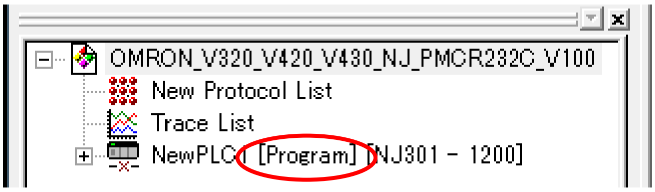
CPU Type: 1200

5 Select **Connect to PLC** from the **PLC** Menu.

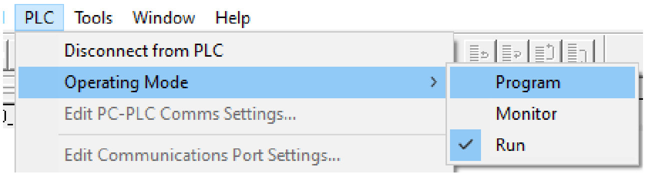


6 Confirm that the operating mode indication of the PLC icon in the project workspace has changed from **Offline** to **Program**, which means that the CX-Protocol is online with the controller.

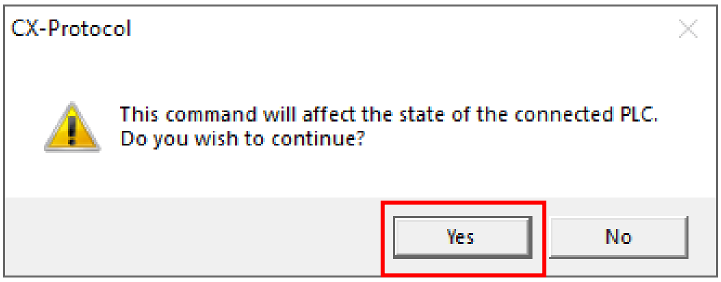
* If the indication is **Monitor** or **Run**, perform steps 8 to 9 below to change to **Program**.


OMRON_V320_V420_V430_NJ_PMCR232C_V100
New Protocol List
Trace List
NewPLC1 [Program] [NJ301 - 1200]

7 If the operating mode indication of the controller is **Run**, select **Operating Mode – Program** from the **PLC** Menu.

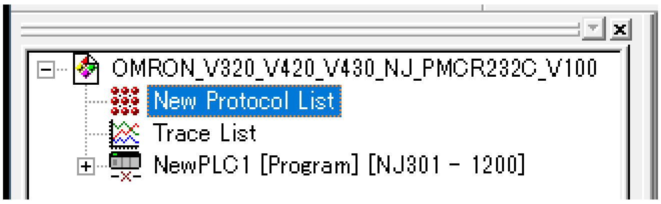


8 A dialog box as shown in the figure on the right appears. Click **Yes**.
Confirm that the mode indication has changed to **Program**, as described in step 7.


CX-Protocol

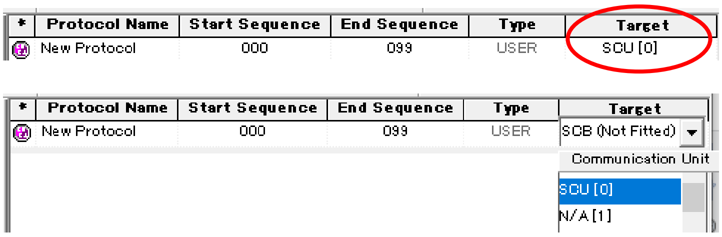
This command will affect the state of the connected PLC.
Do you wish to continue?

9 Double-click **New Protocol List** in the project workspace to open the tree.


OMRON_V320_V420_V430_NJ_PMCR232C_V100
New Protocol List
Trace List
NewPLC1 [Program] [NJ301 - 1200]

10 A project window as shown on the right is displayed. Confirm that *SCU[0]* is entered in **Target**.

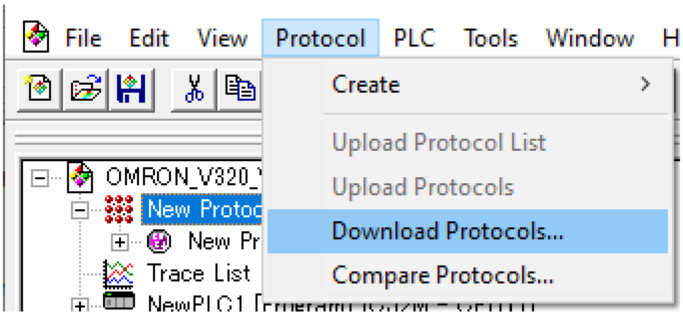
* If *SCU[0]* is not entered, select **SCU[0]** as shown in the figure on the right.



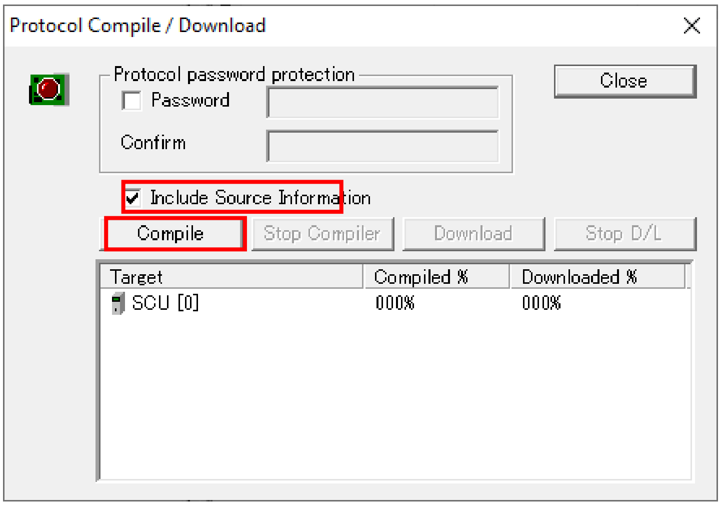
*	Protocol Name	Start Sequence	End Sequence	Type	Target
	New Protocol	000	099	USER	SCU [0]

*	Protocol Name	Start Sequence	End Sequence	Type	Target
	New Protocol	000	099	USER	SCB (Not Fitted)

11 With New Protocol List selected, select **Download Protocols** from the **Protocol** Menu.



12 The dialog box on the right is displayed. Select the check box for **Include Source Information**, and click **Compile**.



Protocol Compile / Download

Protocol password protection

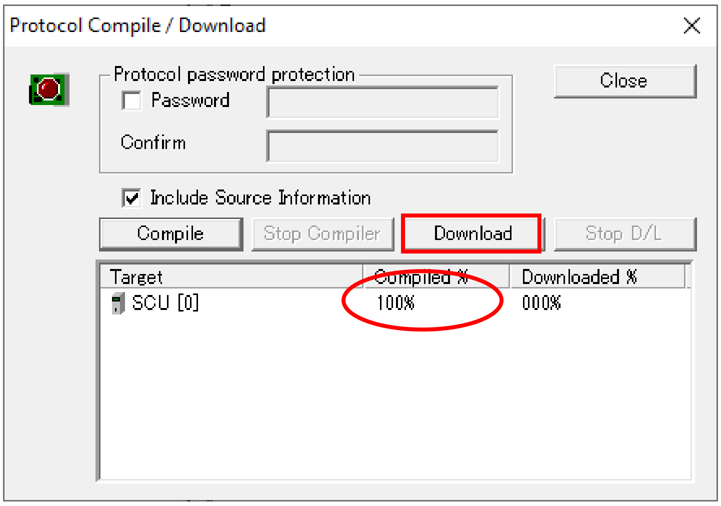
□ Password

Confirm

☑ Include Source Information

Target	Compiled %	Downloaded %
SCU [0]	000%	000%

13 The compile process is complete when **Compiled %** shows *100%* in the dialog box on the right.
Confirm that the compile process is successfully completed as shown on the right, and then click **Download**.



Protocol Compile / Download

Protocol password protection

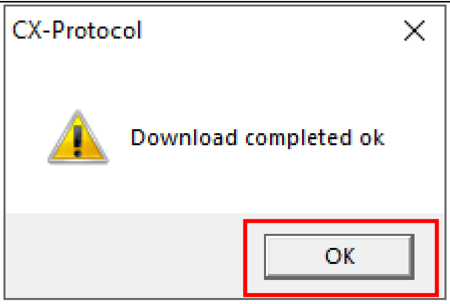
□ Password

Confirm

☑ Include Source Information

Target	Compiled %	Downloaded %
SCU [0]	100%	000%

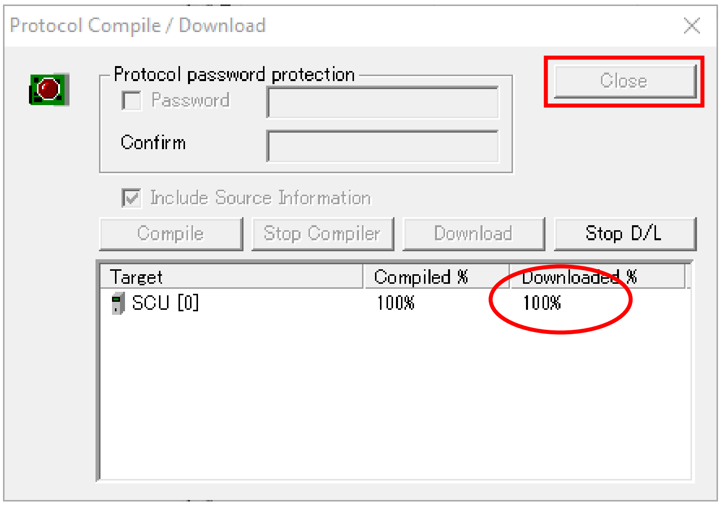
14 A dialog box as shown in the figure on the right appears. Click **OK**.



CX-Protocol

Download completed ok

15 Confirm that **Downloaded %** shows *100%* in the dialog box on the right, and click **Close**.



Protocol Compile / Download

Protocol password protection

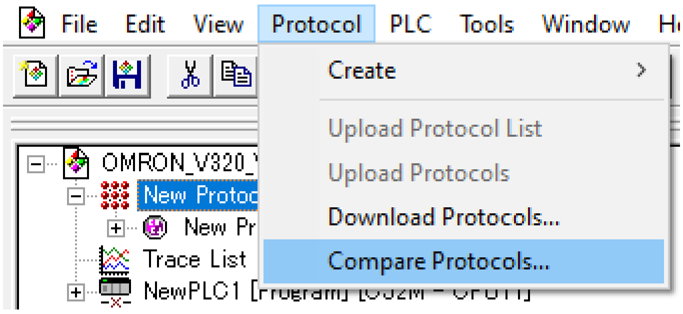
□ Password

Confirm

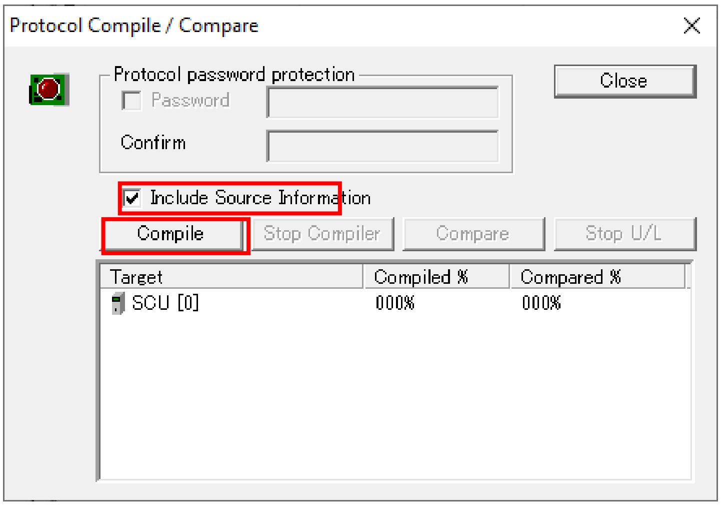
☑ Include Source Information

Target	Compiled %	Downloaded %
SCU [0]	100%	100%

16 With New Protocol List selected, select **Compare Protocols** from the **Protocol** Menu.



17 The dialog box on the right is displayed. Select the check box for **Include Source Information**, and click **Compile**.



Protocol Compile / Compare

Protocol password protection

□ Password

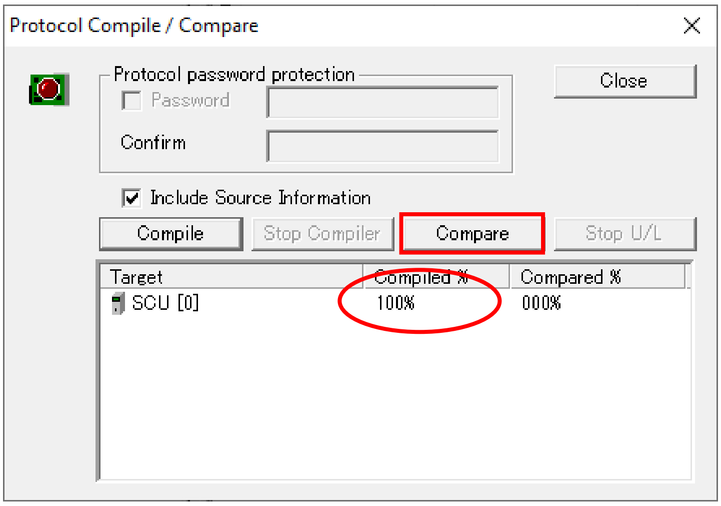
Confirm

☑ Include Source Information

Target	Compiled %	Compared %
SCU [0]	000%	000%

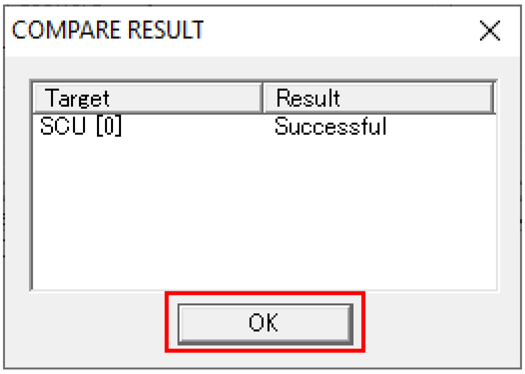
18 The compile process is complete when **Compiled %** shows *100%* in the dialog box on the right.

Confirm that the compile process is successfully completed as shown on the right, and then click **Compare**.


Protocol Compile / Compare
Protocol password protection
□ Password
Confirm
Close
☑ Include Source Information
Compile
Stop Compiler
Compare
Stop U/L

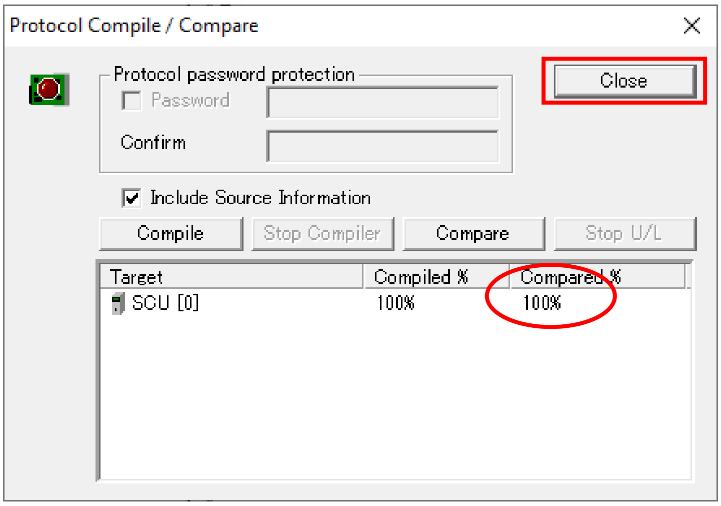
Target	Compiled %	Compared %
SCU [0]	100%	000%

19 A dialog box as shown in the figure on the right appears. Click **OK**.



Target	Result
SCU [0]	Successful

20 Confirm that **Compared %** shows *100%* in the dialog box on the right, and click **Close**.


Protocol Compile / Compare
Protocol password protection
□ Password
Confirm
Close
☑ Include Source Information
Compile
Stop Compiler
Compare
Stop U/L

Target	Compiled %	Compared %
SCU [0]	100%	100%

7.4. Checking That Serial Communications Work

Execute the program to check that serial communications work correctly.

Caution

Sufficiently confirm safety before you change the values of variables on a Watch Tab Page when the Sysmac Studio is online with the CPU Unit. Incorrect operation may cause the devices that are connected to Output Units to operate regardless of the operating mode of the controller.

Precautions for Correct Use

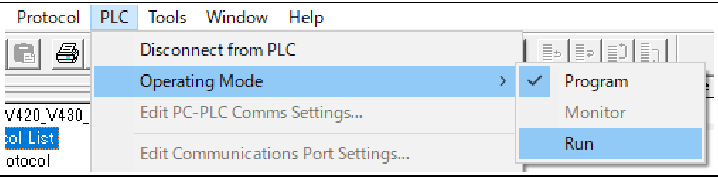
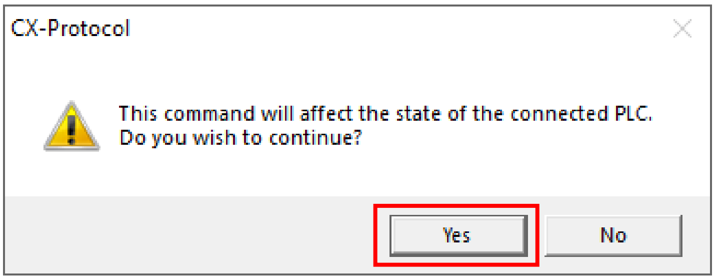
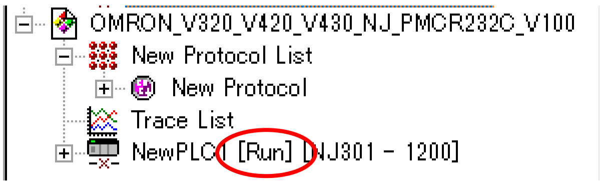
This document assumes that the code reader is used in the factory default settings. Be sure to initialize the code reader according to *8.2. Initializing the Code Reader* if it is not in the factory default settings.

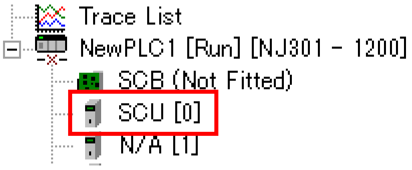
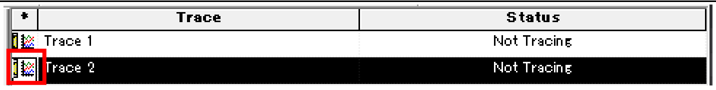
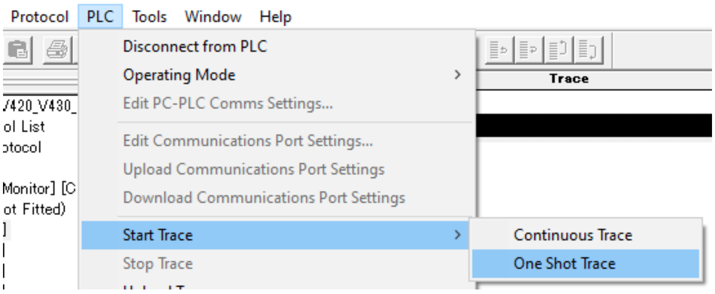
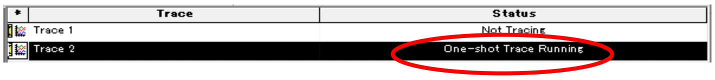
Precautions for Correct Use

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

7.4.1. Starting a Trace

Start a trace in the CX-Protocol.

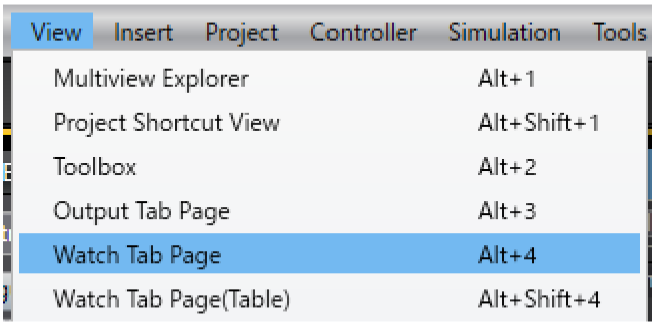
1	In the CX-Protocol, select **Operating Mode – Run** from the **PLC** Menu.	 Protocol PLC Tools Window Help Disconnect from PLC Operating Mode Edit PC-PLC Comms Settings... Edit Communications Port Settings... Program Monitor Run V420_V430_ ol List otocol
2	A dialog box as shown in the figure on the right appears. Click **Yes**.	CX-Protocol This command will affect the state of the connected PLC. Do you wish to continue?
3	Confirm that the operating mode indication has changed to **Run**, and double-click **NewPLC1**.	OMRON_V320_V420_V430_NJ_PMCR232C_V100 New Protocol List New Protocol Trace List NewPLC1 [Run] [NJ301 - 1200]

4	The tree structure under **NewPLC1** is open. Select the Serial Communications Unit (**SCU[0]** in the figure on the right).	 Trace List NewPLC1 [Run] [NJ301 - 1200] SCB (Not Fitted) SCU [0] N/A [1]
5	In the project window, select the **Trace 2** icon (). Confirm that **Trace 2** is highlighted as shown on the right. * **Trace 2** corresponds to "Port 2" on the Serial Communications Unit.	 * Trace Status Trace 1 Not Tracing Trace 2 Not Tracing
6	Select **Start Trace** – **One Shot Trace** from the **PLC** Menu.	 Protocol PLC Tools Window Help Disconnect from PLC Operating Mode Edit PC-PLC Comms Settings... Edit Communications Port Settings... Upload Communications Port Settings Download Communications Port Settings Start Trace Stop Trace Continuous Trace One Shot Trace Trace /420_V430_ ol List otocol Monitor] [C ot Fitted)
7	In the project window, confirm that the status of **Trace 2** has changed to **One-shot Trace Running**.	 * Trace Status Trace 1 Not Tracing Trace 2 One-shot Trace Running

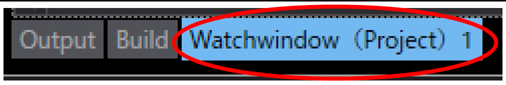
7.4.2. Executing the Program

Execute the program in the Sysmac Studio.

1 In the Sysmac Studio, select **Watch Tab Page** from the **View** Menu.



2 The **Watchwindow (Project) 1** Tab Page is displayed under the Edit Pane.

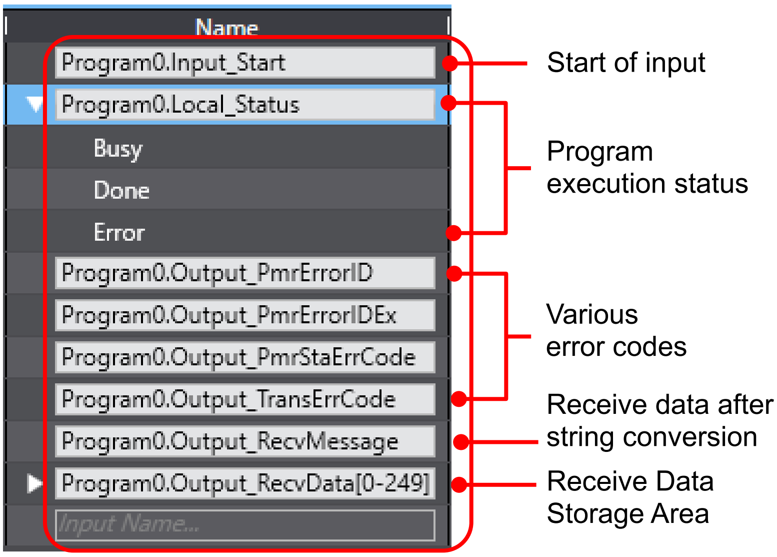


3 Confirm that the variables shown in the figure on the right are listed in the **Name** column.

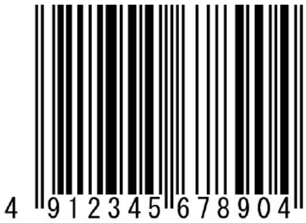
* To add a variable, click **Input Name**.

* If ▶ is displayed in the place of ▼ to the left of the variable, click on the ▶ to expand the structure variable.

* In the following description, “Program0” of the variable names in the **Name** column is omitted.


Name
Program0.Input_Start
Program0.Local_Status
Busy
Done
Error
Program0.Output_PmrErrorID
Program0.Output_PmrErrorIDEx
Program0.Output_PmrStaErrCode
Program0.Output_TransErrCode
Program0.Output_RecvMessage
Program0.Output_RecvData[0-249]
Input Name...
Start of input
Program execution status
Various error codes
Receive data after string conversion
Receive Data Storage Area

4 This document uses a barcode as shown in the figure on the right as an example of reading.
Set the code reader to the position where it can read the barcode in the figure on the right.

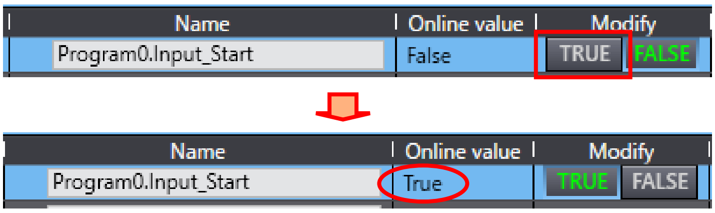


5 Click **TRUE** in the **Modify** column of *Input_Start*.

The **Online value** of *Input_Start* changes to *True*.

The program starts running and the controller performs serial communications with the code reader to read the barcode .

* The reading is successful if the PASS LED indicator on the code reader is lit green.



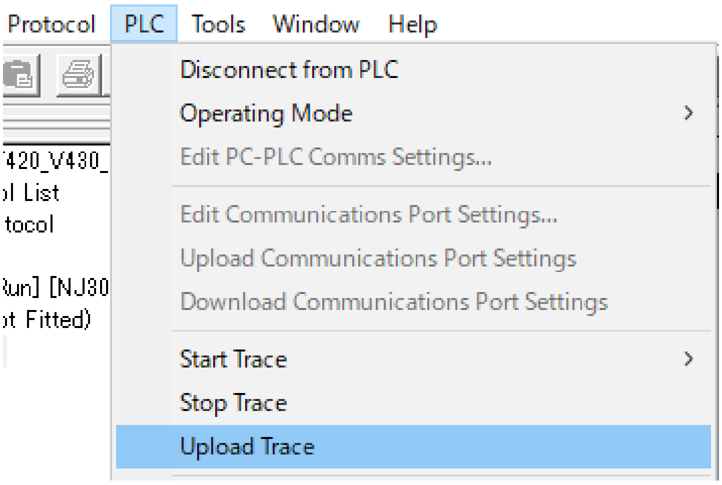
Name	Online value	Modify
Program0.Input_Start	False	TRUE FALSE

Name	Online value	Modify
Program0.Input_Start	True	TRUE FALSE

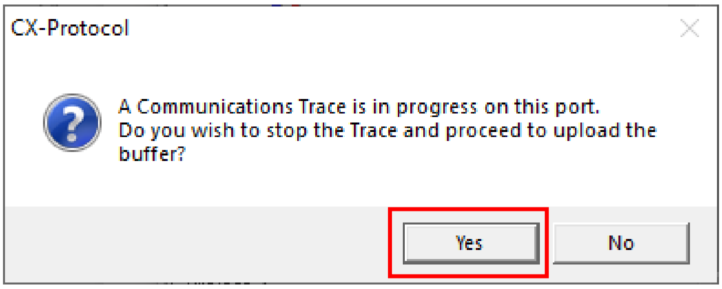
7.4.3. Checking the Trace Data

Using the trace data of the CX-Protocol, check that the correct data is sent and received.

1 In the CX-Protocol, select **Upload Trace** from the **PLC** Menu.


Protocol
PLC
Tools
Window
Help
420_V430_
l List
tocol
un] [NJ30
t Fitted)
Disconnect from PLC
Operating Mode
Edit PC-PLC Comms Settings...
Edit Communications Port Settings...
Upload Communications Port Settings
Download Communications Port Settings
Start Trace
Stop Trace
Upload Trace

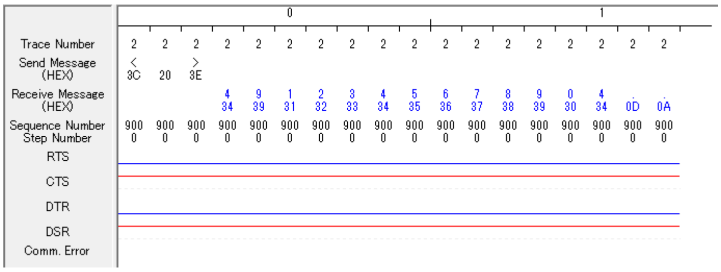
2 A dialog box as shown in the figure on the right appears. Click **Yes**.


CX-Protocol

A Communications Trace is in progress on this port.
Do you wish to stop the Trace and proceed to upload the buffer?

3 Check the Send Message and Receive Message in the trace data file shown in the figure on the right.

* In the example on the right, you can see that the controller has sent the "Read trigger" command and received the barcode data shown in step 4 in section 7.4.2.



Trace Number	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Send Message (HEX)	< 3C	20	> 3E															
Receive Message (HEX)				4 34	9 39	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	0 30	4 34	. 0D	. 0A
Sequence Number Step Number	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0	900 0

■ Send Data
Hex: 3C 20 3E
String: < >
■ Receive Data
Hex: 34 39 31 32 33 34 35 36 37 38 39 30 34 0D 0A
String: "4912345678904"[CR][LF]

7.4.4. Checking the Receive Data

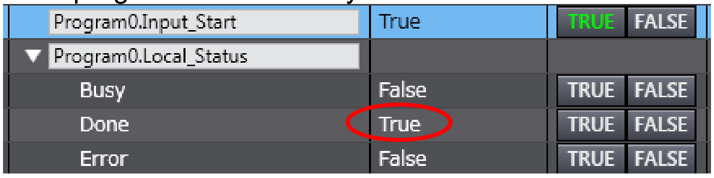
In the Sysmac Studio, check that correct data is written to the controller variables.

1 Also confirm that the **Online value** of *Local_Status.Done* indicating the program execution status is *True*.

* This indicates that the program has ended normally.

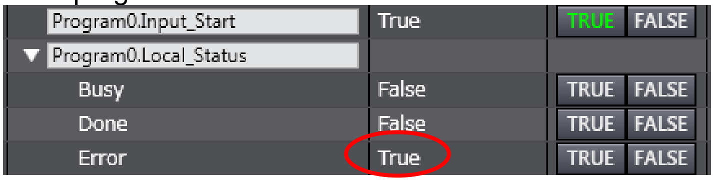
* If the program ends with an error, the value of *Local_Status.Error* is *True*. Refer to step 2 and make a correction.

If the program ends normally



Program0.Input_Start	True	TRUE FALSE
▼ Program0.Local_Status		
Busy	False	TRUE FALSE
Done	True	TRUE FALSE
Error	False	TRUE FALSE

If the program ends with an error

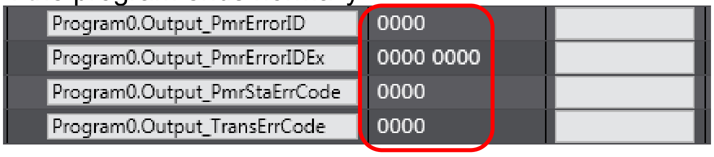


Program0.Input_Start	True	TRUE FALSE
▼ Program0.Local_Status		
Busy	False	TRUE FALSE
Done	False	TRUE FALSE
Error	True	TRUE FALSE

2 If the program ends normally, check that the error codes are *0*.

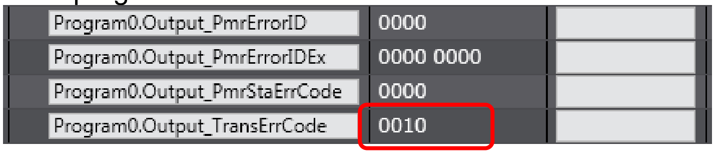
* If the program ends with an error, the error code is stored according to the error that occurred. Refer to *9.8 Error Processing* and make a correction.

If the program ends normally



Program0.Output_PmrErrorID	0000
Program0.Output_PmrErrorIDEx	0000 0000
Program0.Output_PmrStaErrCode	0000
Program0.Output_TransErrCode	0000

If the program ends with an error

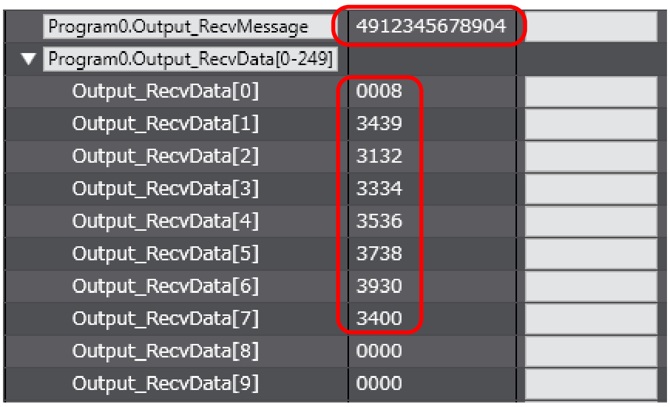


Program0.Output_PmrErrorID	0000
Program0.Output_PmrErrorIDEx	0000 0000
Program0.Output_PmrStaErrCode	0000
Program0.Output_TransErrCode	0010

3 Check the received data (read code) in the Watch Tab Page of the Sysmac Studio.

* In the example on the right, the data stored in *Output_RecvMessage* is "4912345678904", which is the same as the trace data in step 3 in section 7.4.3.

* *Output_RecvData[0]* stores the number of words (*0008* in hex) used to store received data. The content of the read barcode is stored in *Output_RecvData[1]* to *Output_RecvData[7]*, so the number of words used is 8 including *Output_RecvData[0]*.



Program0.Output_RecvMessage	4912345678904
▼ Program0.Output_RecvData[0-249]	
Output_RecvData[0]	0008
Output_RecvData[1]	3439
Output_RecvData[2]	3132
Output_RecvData[3]	3334
Output_RecvData[4]	3536
Output_RecvData[5]	3738
Output_RecvData[6]	3930
Output_RecvData[7]	3400
Output_RecvData[8]	0000
Output_RecvData[9]	0000

Received data (Hex)

- Output_RecvData[0]: "0008"

Number of words used to store receive data 8 words, [0] to [7]

- Output_RecvData[1]: "3439"
- Output_RecvData[2]: "3132"
- Output_RecvData[3]: "3334"
- Output_RecvData[4]: "3536"
- Output_RecvData[5]: "3738"
- Output_RecvData[6]: "3930"
- Output_RecvData[7]: "3400"

Barcode value (String)
Output_RecvMessage
"4912345678904"

8. Initializing the System

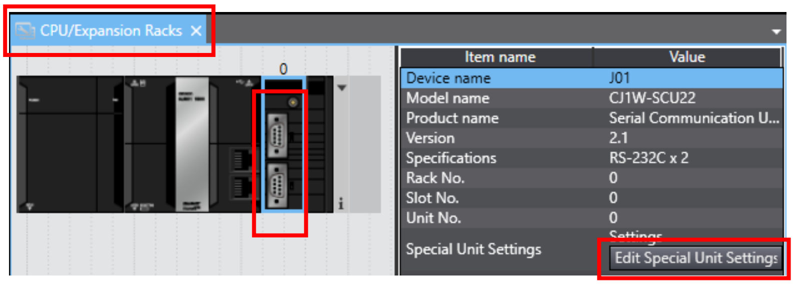
This document assumes that each device uses the factory default settings.
If you change their settings from the defaults, you may not be able to perform various setting procedures as described.

8.1. Initializing the Controller

In order to initialize the controller, both the CPU Unit and the Serial Communications Unit must be initialized. Please put the controller in PROGRAM mode before initialization.

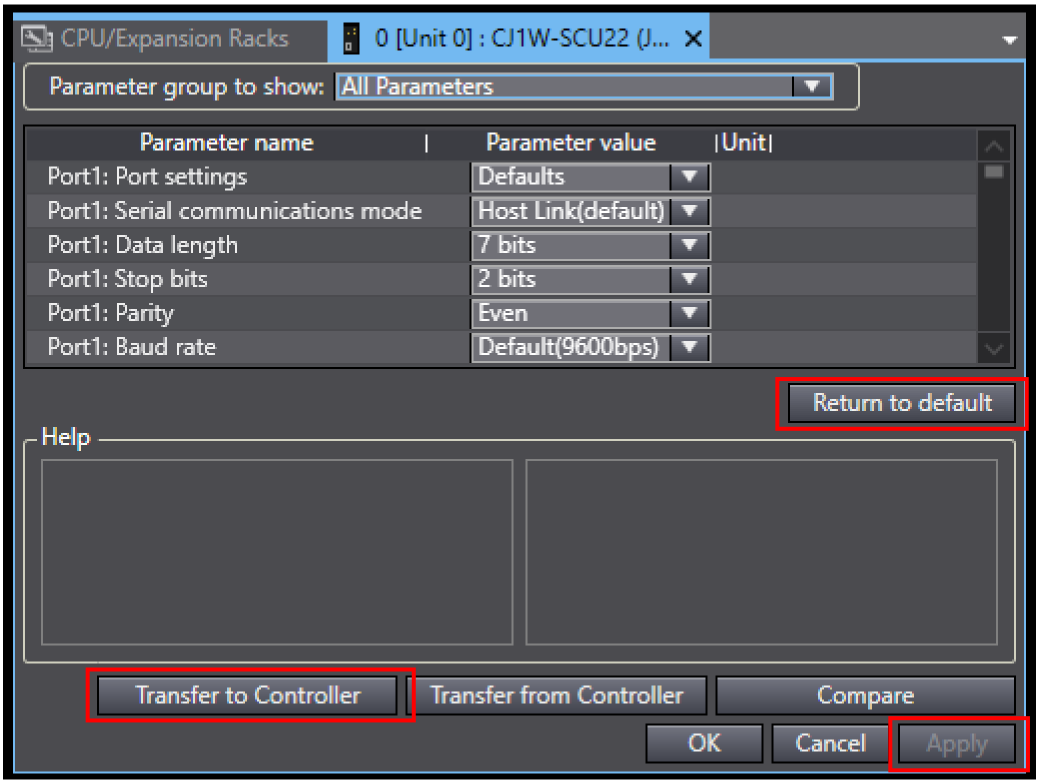
8.1.1. Serial Communications Unit

To return the CJ1W-SCU22 Serial Communication Unit to its default settings, in the Sysmac Studio, select it in the **CPU/Expansion Racks** Tab Page and click **Edit Special Unit Settings**.



Item name	Value
Device name	J01
Model name	CJ1W-SCU22
Product name	Serial Communication U...
Version	2.1
Specifications	RS-232C x 2
Rack No.	0
Slot No.	0
Unit No.	0
Special Unit Settings	Settings

Select **Return to default**, select **Apply**, and then select **Transfer to Controller**.

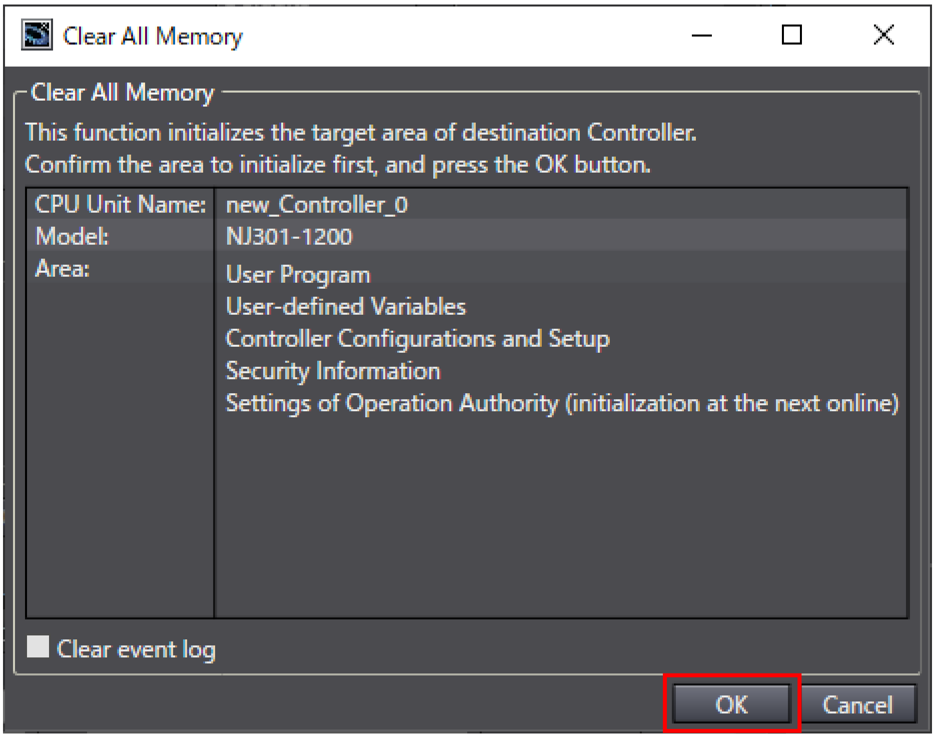


Parameter group to show: All Parameters

Parameter name	Parameter value	Unit
Port1: Port settings	Defaults	
Port1: Serial communications mode	Host Link(default)	
Port1: Data length	7 bits	
Port1: Stop bits	2 bits	
Port1: Parity	Even	
Port1: Baud rate	Default(9600bps)	

CPU Unit

To return the controller to its default settings, select **Clear All Memory** from **Controller** Menu in the Sysmac Studio. When the Clear All Memory Dialog Box is displayed, click **OK**.



Clear All Memory

Clear All Memory

This function initializes the target area of destination Controller.
Confirm the area to initialize first, and press the OK button.

CPU Unit Name:	new_Controller_0
Model:	NJ301-1200
Area:	User Program User-defined Variables Controller Configurations and Setup Security Information Settings of Operation Authority (initialization at the next online)

□ Clear event log

8.2. Initializing the Code Reader

For information on initializing the code reader, please refer to *How to initialize the settings?* in *Q&A* in *Appendices* of the *MicroHAWK V320-F/V330-F/V420-F/V430-F Series Barcode Reader User Manual* (Cat. No. Z432).

9. Program

This section describes the details of the program and protocol macro data used in this document.

9.1. Overview

This section describes the specifications and functions of the program and protocol macro data used to check the connection between the code reader and the controller (Serial Communications Unit (hereinafter SCU unit)).

This program and protocol macro data use the protocol macro function of the SCU unit to send and receive a “Read trigger” command to and from the code reader, and judge whether the normal end or error end has reached.
Here, “normal end” of the program means that the communications sequence of the protocol macro has ended successfully.
On the other hand, “error end” means that the communications sequence of a protocol macro has ended with an error.

In this section, if it is necessary to distinguish between decimal data and hexadecimal data, add "10#" (optional) to the beginning of decimal data and "16#" to the beginning of hexadecimal data. (Example: “1000” or “10#1000” for decimal data, “16#03E8” for hexadecimal data)
To specify the data type, add "<*data type*> #" at the beginning. (Example: "WORD#16#03E8")

Note

We have verified in our test configuration that the program and protocol macro data enable 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.

9.1.1. Communications Data Flow

This is the flow from issuing command data from the controller (SCU unit) to the code reader through serial communications and receiving response data from the code reader.

1.	Communications Sequence Execution Processing	The protocol macro instruction (Mnemonic: ExecPMCR) for the sequence number specified in the program is executed to call the communications sequence (protocol macro data) registered in the SCU unit.
	▼	
2.	Command Send Processing	Based on the sequence number specified in step 1, the SCU unit issues a send message (command data) to the code reader.
	▼	
3.	Response Receive Processing	The SCU unit stores the receive message (response data) received from the code reader in the internal variable of the specified CPU Unit.

9.1.2. Protocol Macro Execution Function Block and Send/Receive Messages

This section provides an overview of the protocol macro execution function block (Mnemonic: PMCR, hereinafter referred to as "ExecPMCR instruction") and the general movement of send and receive messages.

Note

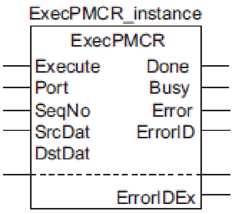
For details, refer to *Serial Communications Instructions (ExecPMCR)* in *Section 2 Instruction Descriptions* in the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

● ExecPMCR Instruction

This program uses the following standard instructions to implement serial communications.

Name	Function block	Description
Protocol Macro	ExecPMCR	Requests the execution of the communications sequence (protocol data) registered in the SCU unit.

● ExecPMCR Instruction Argument Data

Instruction	Name	FB/ FUN	Graphic expression	ST expression
ExecPMCR	Protocol Macro	FB		ExecPMCR_instance(Execute, Port, SeqNo, SrcDat, DstDat, Done, Busy, Error, ErrorID, ErrorIDEx);

Variables

	Meaning	I/O	Description	Valid range	Unit	Default
Port	Destination port	Input	Destination port	---	---	---
SeqNo	Communications sequence number		Communications sequence number	0 to 999		0
SrcDat[] (array)	Send data array		Send data array	Depends on data type.		*1
DstDat[] (array)	Receive data array	In-out	Receive data array	Depends on data type.	---	---

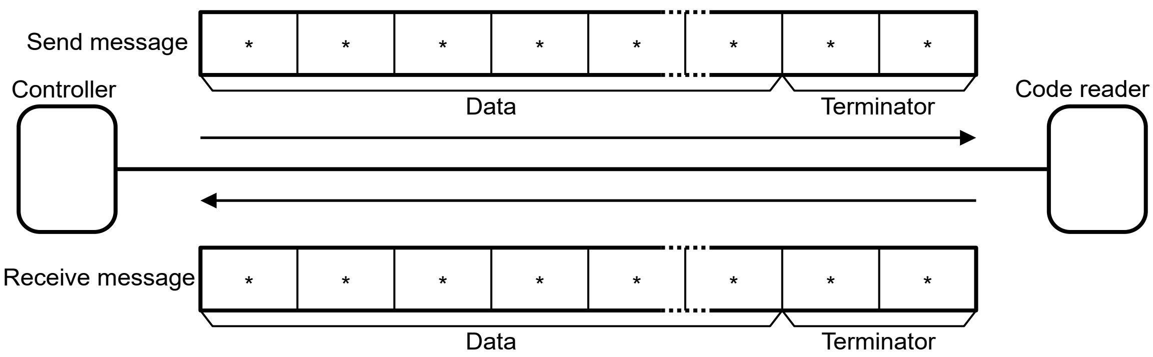
*1. If you omit the input parameter, the default value is not applied. A building error will occur.

• Data Type of Destination Port Specification (_sPORT)

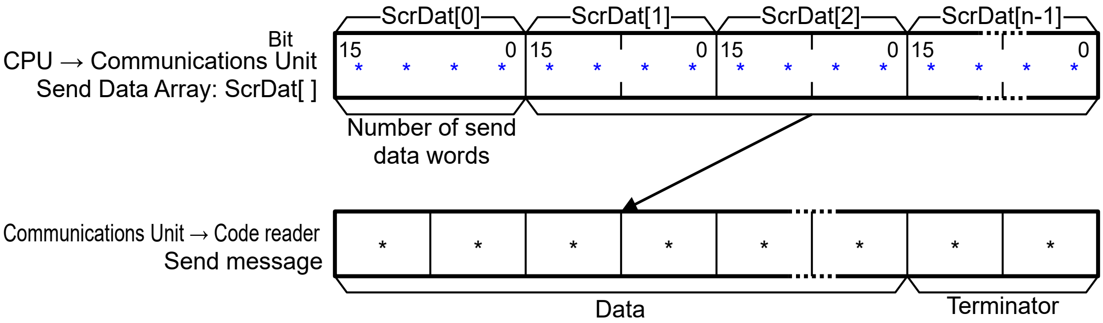
Name	Meaning	Description	Data type	Valid range	Unit	Default
Port	Destination port	Destination port	_sPORT	---	---	---
UnitNo	Unit number	Unit number of Serial Communications Unit	_eUnitNo	_CBU_No00 to _CBU_No15	---	_CBU_No00
PhysicPortNo	Serial port number	Serial port number on Serial Communications Unit	USINT	1 or 2		1

- Send and Receive Messages

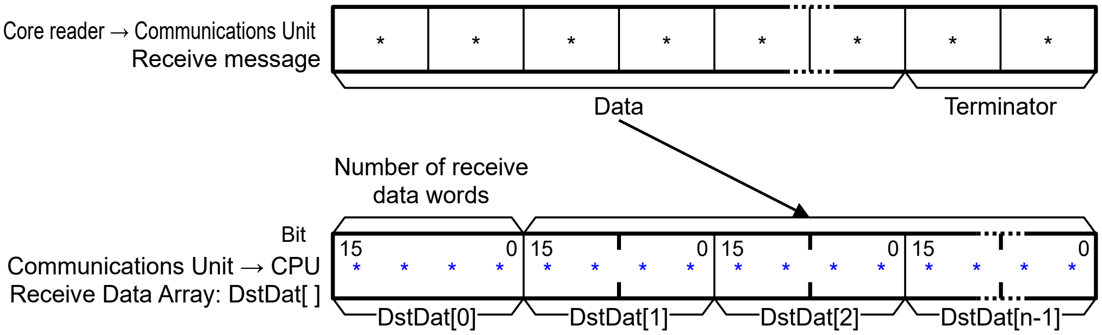
Overview of Send and Receive Messages



Relationship between Send Data Array *SrcDat[ ]* and Send Message


ScrDat[0]
ScrDat[1]
ScrDat[2]
ScrDat[n-1]
Bit
CPU → Communications Unit
Send Data Array: ScrDat[ ]
15
0
Number of send data words
Communications Unit → Code reader
Send message
Data
Terminator

Relationship between Receive Message and Receive Data Array *DstDat[ ]*


Core reader → Communications Unit
Receive message
Data
Terminator
Number of receive data words
Bit
Communications Unit → CPU
Receive Data Array: DstDat[ ]
15
0
DstDat[0]
DstDat[1]
DstDat[2]
DstDat[n-1]

9.2. Communications Sequence

This section describes the communications sequence (protocol macro data) that can be used with ExecPMCR instructions in the program.

9.2.1. Communications Sequence Number

The communications sequence (protocol macro data) registered in the SCU unit is identified by the communications sequence number. By specifying the communications sequence number with an ExecPMCR instruction, the corresponding code reader command is executed in the code reader.

The following communications sequence is available with the protocol macro data.

No.	Command	Description
900	Read trigger	Issues a trigger to read the code.

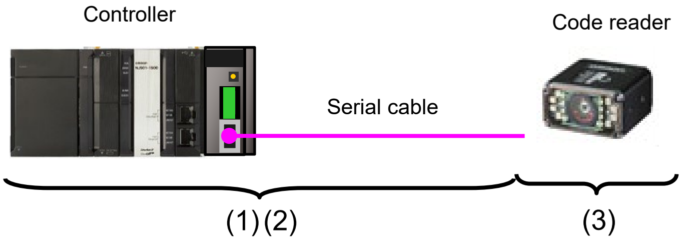
9.2.2. ExecPMCR Instruction Input Variable Settings

The table below shows the ExecPMCR Instruction input variable settings for Communications Sequence No. 900 "Read trigger".

Variable		Name (Data type)	Data description
Port		Destination Port Specification (_sPROT)	---
	UnitNo	Unit No. (_eUnitNo)	_eUnitNo#_CBU_No00 (Specifies the SCU unit number.)
	PhysicPortNo	Serial Port No. (USINT)	USINT#10#2 (Uses Serial Port No. 2.)
SeqNo		Communications Sequence No. (UINT)	UINT#10#900 (Specifies the Read trigger.)
SrcDat[ ]		Send Data Array (WORD)	WORD#16#0000 (Number of send data words 0: Specifies that the command has no required parameters.)
DstDat[ ]		Receive Data Array (WORD)	(No setting is required for receive data.)

9.3. Error Judgment Processing

This program performs error judgment processing for the following three types of errors (1) to (3). Refer to *9.8. Error Processing* for error codes.



(1) Error that occurred during ExecPMCR instruction execution (ExecPMCR instruction error)
An error that occurred during ExecPMCR instruction execution, such as main unit error or communications setting error, is judged as an “ExecPMCR instruction error”. Judgment is made based on the error code (ErrorID) and expansion error code (ErrorIDEx) of the ExecPMCR instruction.

(2) Error that occurred during protocol macro execution (Protocol macro error)
An error that occurred for some reason during protocol macro execution is judged as a “protocol macro error”. Judgment is made based on the Sequence Abort Completion Flag (J01_P2_PmrSeqAbtSta).

(3) Code reader error
An error that occurred in the code reader, such as command error, parameter error, data error, or inexecutable error, is judged as a “code reader error”. The judgment is made based on the response data returned from the code reader. The program detects this error as a timeout error in (2) since no response is returned from the code reader when it occurs.

Since the causes of errors (1) and (2) above may be correlated, all the following error codes are stored if any of the errors occurs.

- ExecPMCR Instruction Error Code (ErrorID)
- ExecPMCR Instruction Expansion Error Code (ErrorIDEx)
- Protocol Macro Error Code (Lower 4 bits of J01_P2_PmrSta)
- Transmission Error Status (J01_P2_TransErrSta)

9.4. Variables Used

This section describes variables used in the program.

9.4.1. User-defined Variables

The tables below list data types, external variables (user-defined global variables, device variables for CJ-series Unit, and system-defined variables), and internal variables used in this program.

● Data Type (Structure)

Communications Processing Status Flag

Name		Data type	Description
sStatus		STRUCT	Structure of Communications Processing Status Flag String
	Busy	BOOL	Communications Processing Execution Flag TRUE: Executing, FALSE: Not executing
	Done	BOOL	Communications Processing Normal End Flag TRUE: Normal end, FALSE: Other than normal end
	Error	BOOL	Communications Processing Error End Flag TRUE: Error end, FALSE (Other than error end)

● External Variables

User-defined Global Variables

<table>
<tr><th>Variable name</th><th>Data type</th><th colspan="2">Description</th></tr>
<tr><td>Input_Start</td><td>BOOL</td><td colspan="2">Communications Start Switch
The program changes when the value changes from <i>FALSE</i> to <i>TRUE</i>.</td></tr>
<tr><td>Input_SendData</td><td>ARRAY[0..249] OF WORD</td><td colspan="2">Send Data Storage Area (500 bytes of WORD data)</td></tr>
<tr><td rowspan="2">Output_RecvData</td><td rowspan="2">ARRAY[0..249] OF WORD</td><td rowspan="2">Receive Data Storage Area (500 bytes of WORD data)
Area for storing <i>DstDat[ ]</i> of the ExecPMCR instruction</td><td>AT specification</td></tr>
<tr><td>%5500
(*)</td></tr>
<tr><td>Output_RecvMessage</td><td>STRING[498]</td><td colspan="2">String Converted Receive Data Storage Area (498 characters)</td></tr>
<tr><td>Output_PmrErrorID</td><td>WORD</td><td colspan="2">ExecPMCR Instruction Error Code Storage Area
Normal end: 16#0000</td></tr>
<tr><td>Output_PmrErrorIDEx</td><td>DWORD</td><td colspan="2">ExecPMCR Instruction Extended Error Code Storage Area
Normal end: 16#00000000</td></tr>
<tr><td>Output_PmrStaErrCode</td><td>WORD</td><td colspan="2">Protocol Macro Error Code Storage Area
Normal end: 16#0000</td></tr>
<tr><td>Output_TransErrCode</td><td>WORD</td><td colspan="2">Transmission Error Status Flag String Storage Area
Area for storing <i>J01_P2_TransErrSta</i>
Normal end: 16#0000</td></tr>
</table>

* For information on the area for storing *DstDat[]* of the ExecPMCR instruction, AT specification is required for the address of memory used for CJ-series Units.

Device Variables for CJ-series Units (SCU Units)

Variable name	Data type	Description
J01_P2_PmrSta	WORD	Protocol Macro Operation Status Bits 03 to 00: Protocol macro error codes
J01_P2_PmrExecSta	BOOL	Protocol Macro Execution Flag
J01_P2_PmrSeqEndSta	BOOL	Sequence End Completion Flag
J01_P2_PmrSeqAbtSta	BOOL	Sequence Abort Completion Flag
J01_P2_TransErrSta	WORD	Transmission Error Status

Note

For information on variables used for the SCU Unit, refer to *2-3 Device Variable for CJ-series Unit* in the *CJ-series Serial Communications Units Operation Manual for NJ-series CPU Unit* (Cat. No. W494).

System-defined Variables

Variable name	Data type	Description
_Port_isAvailable	BOOL	Network Communications Instruction Enabled Flag TRUE: A port is available. FALSE: No port is not available.

Note

For information on system-defined variables used with the ExecPMCR instruction, refer to *Related System-defined Variables* in *Serial Communications Instruction (ExecPMCR)* in Section 2 *Instruction Descriptions* of the *Machine Automation Controller NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

● Internal Variables (Instance Variables)

The table below lists internal variables used for executing function blocks in the program. These internal variables are called “Instances”, each of which specifies the name of the function block used as data type of the variable.

Variable name	Data type	Description
ExecPMCR_instance	ExecPMCR	Requests the execution of the communications sequence (protocol data) registered in the SCU unit.
F_TRIG_instance	F_TRIG	Outputs TRUE for one task period only at the falling edge of the input signal.

Note

For information on the ExecPMCR instruction, refer to *Serial Communications Instruction (ExecPMCR)* in *Section 2 Instruction Descriptions* in the *Machine Automation Controller NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

Note

For information on the F_TRIG instruction, refer to *Sequence Input Instructions (F_TRIG)* in *Section 2 Instruction Descriptions* in the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

● Internal Variables

Variable name	Data type	Description
Local_Status	sStatus	Communications Processing Status Flag String Defined by structure type “sStatus”
Local_State	DINT	State Processing No.
Local_ExecFlg	BOOL	Communications Instruction Execution Flag
Local_EndExecPMCR	BOOL	Protocol Macro Execution End Judgment Flag
Local_InPort	_sPort	Specifies the port to be used.
Local_SeqNo	UINT	Communications Sequence No.
Local_PmrStaCode	WORD	Protocol macro error code editing area
Local_RecvWordData	ARRAY[0..248] OF WORD	Area for extracting and storing the part to be converted to a string from the receive data (498 bytes)
Local_RecvWordSize	UINT	Data size of the data to be converted to a string (Unit: Words)
Local_RecvByteData	ARRAY[0..497] OF BYTE	Area for converting the data to be converted to a string in a byte array (498 bytes)
Local_RecvByteSize	UINT	Data size of the data to be converted to a character string (unit: BYTE)

9.5. Program in ST Language

9.5.1. Functional Components of the Program

This program is written in the ST language. The functional components of the program are as follows.

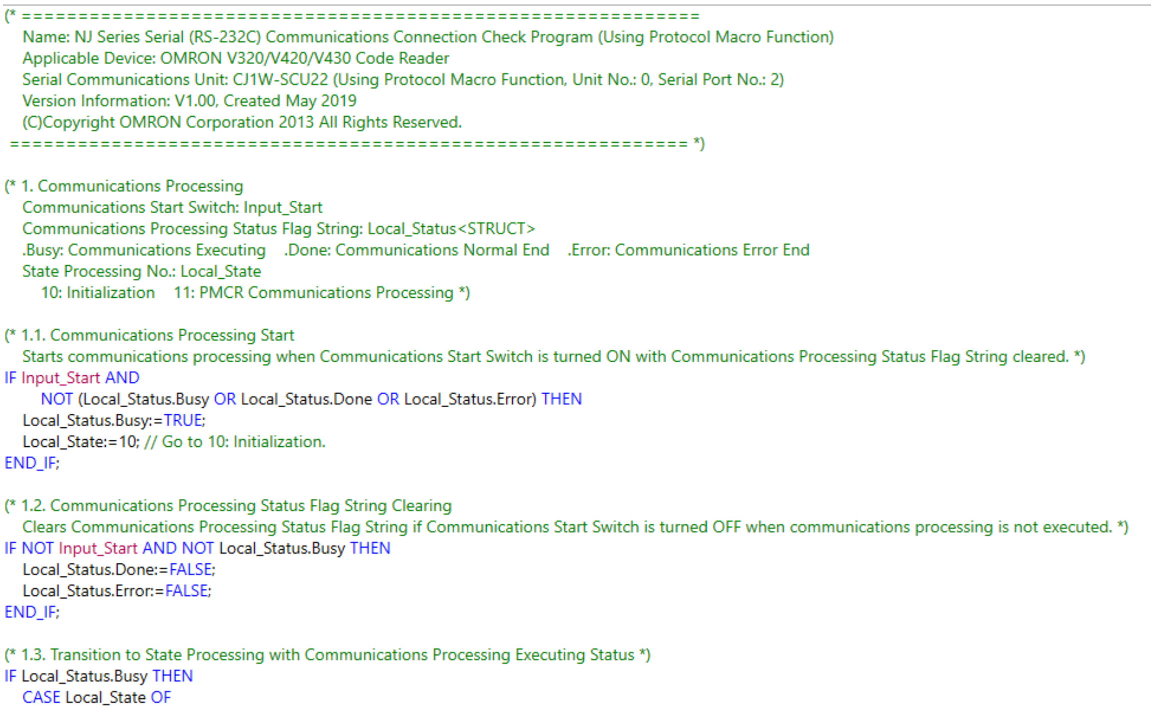
Category	Subcategory	Description
1. Communications Processing	1.1. Communications Processing Start 1.2. Communications Processing Status Flag String Clearing 1.3. Transition to State Processing with Communications Executing Status	Executes communications processing.
2. Initialization	2.1. Communications Instruction Initialization 2.2. Communications Instruction Execution Flag Initialization 2.3. Error Code Storage Area Initialization 2.4. ExecPMCR Instruction Input Variable Setting 2.5. Receive Data Storage Area Setting 2.6. Initialization End Processing	Set the parameters of the communications instruction (ExecPMCR Instruction) and clears the Receive Data Storage Area.
3. PMCR Communications Processing	3.1. Communications Processing Status Judgment and Execution Flag Setting 3.2. Communications Instruction Execution	Executes the ExecPMCR Instruction. Judges whether execution result is normal or error.
4. Processing No. Error Processing	---	Executes error processing if a non-existent processing number is detected.

9.5.2. Contents of the Program

The contents of the program are shown below.

● Program: Program0 (Serial Communications Connection Check Program)

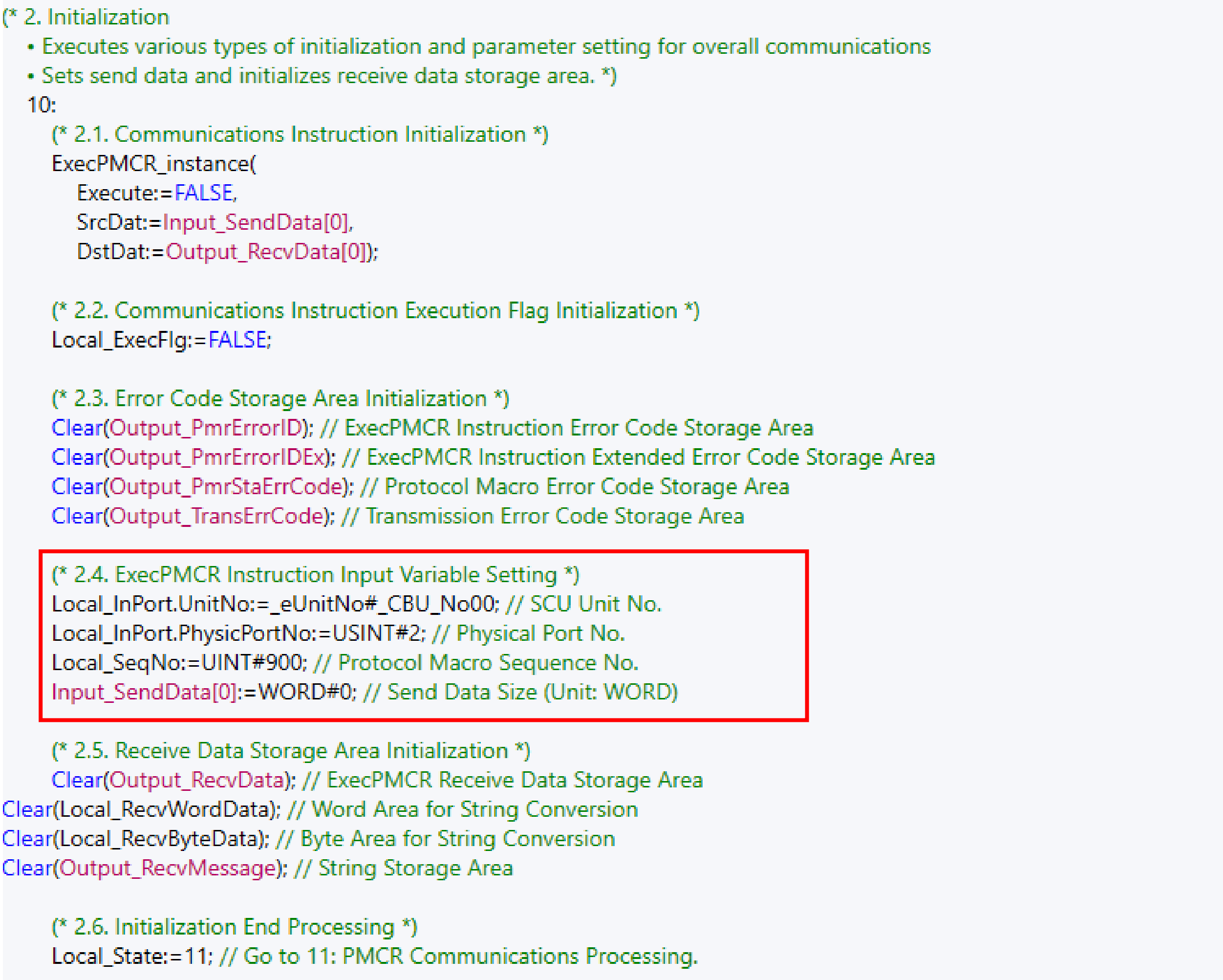
1. Communications Processing



```
(* ============================================================
   Name: NJ Series Serial (RS-232C) Communications Connection Check Program (Using Protocol Macro Function)
   Applicable Device: OMRON V320/V420/V430 Code Reader
   Serial Communications Unit: CJ1W-SCU22 (Using Protocol Macro Function, Unit No.: 0, Serial Port No.: 2)
   Version Information: V1.00, Created May 2019
   (C)Copyright OMRON Corporation 2013 All Rights Reserved.
 ============================================================ *)

(* 1. Communications Processing
   Communications Start Switch: Input_Start
   Communications Processing Status Flag String: Local_Status<STRUCT>
   .Busy: Communications Executing    .Done: Communications Normal End    .Error: Communications Error End
   State Processing No.: Local_State
      10: Initialization    11: PMCR Communications Processing *)

(* 1.1. Communications Processing Start
   Starts communications processing when Communications Start Switch is turned ON with Communications Processing Status Flag String cleared. *)
IF Input_Start AND
      NOT (Local_Status.Busy OR Local_Status.Done OR Local_Status.Error) THEN
   Local_Status.Busy:=TRUE;
   Local_State:=10; // Go to 10: Initialization.
END_IF;

(* 1.2. Communications Processing Status Flag String Clearing
   Clears Communications Processing Status Flag String if Communications Start Switch is turned OFF when communications processing is not executed. *)
IF NOT Input_Start AND NOT Local_Status.Busy THEN
   Local_Status.Done:=FALSE;
   Local_Status.Error:=FALSE;
END_IF;

(* 1.3. Transition to State Processing with Communications Processing Executing Status *)
IF Local_Status.Busy THEN
   CASE Local_State OF
```

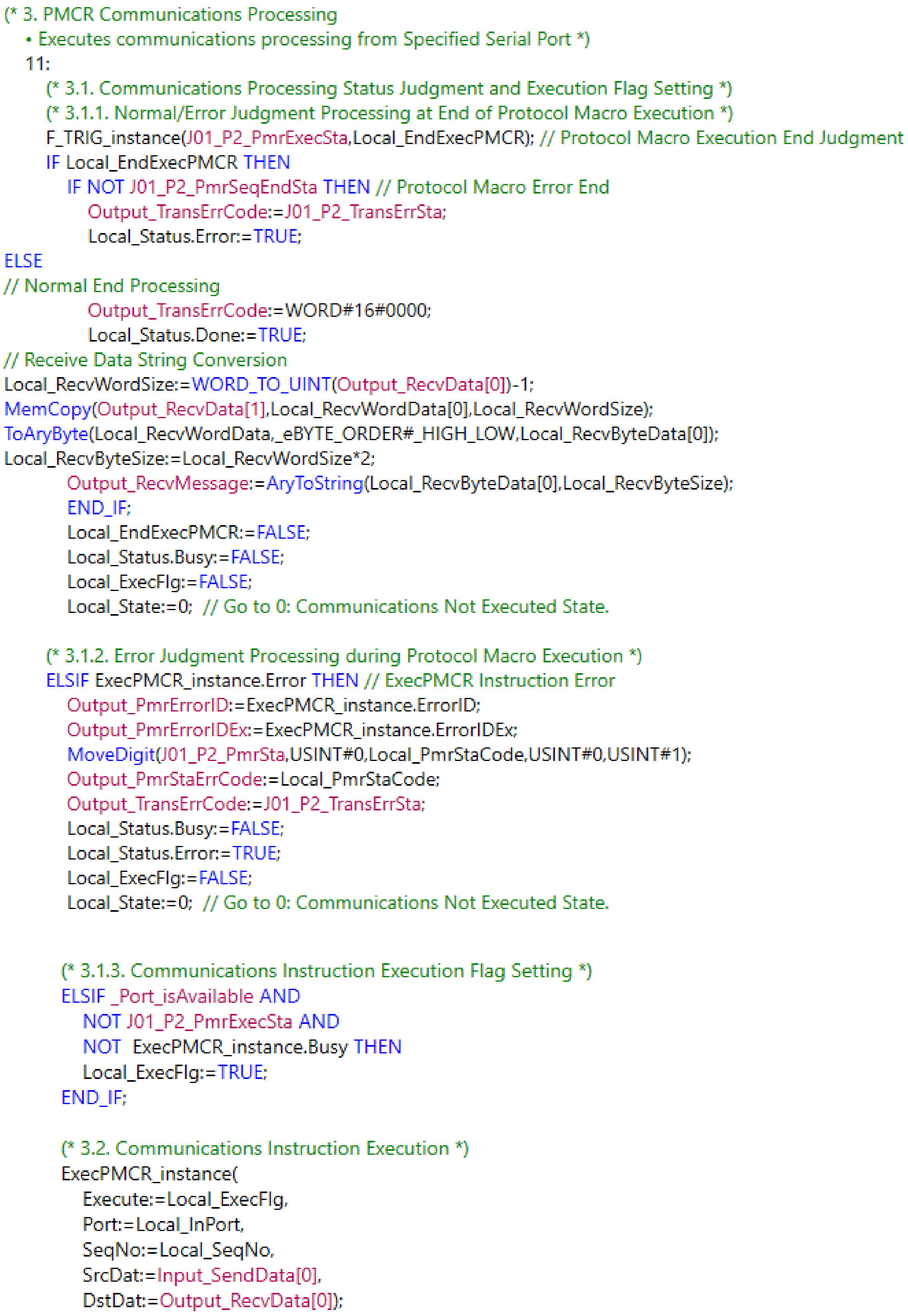
2. Initialization

To change the input values such as SCU Unit number, port number to be used, protocol macro sequence number, and send data size, change the settings shown in the red frame.



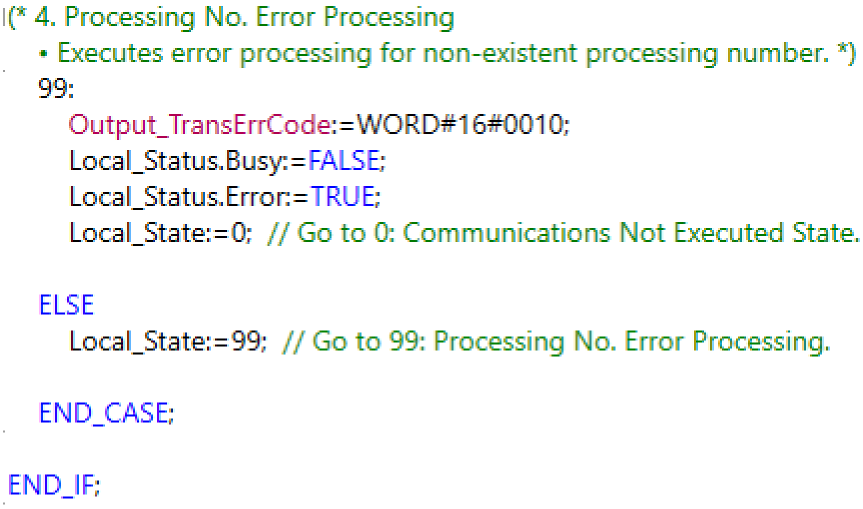
```
(* 2. Initialization
 • Executes various types of initialization and parameter setting for overall communications
 • Sets send data and initializes receive data storage area. *)
  10:
    (* 2.1. Communications Instruction Initialization *)
    ExecPMCR_instance(
      Execute:=FALSE,
      SrcDat:=Input_SendData[0],
      DstDat:=Output_RecvData[0]);

    (* 2.2. Communications Instruction Execution Flag Initialization *)
    Local_ExecFlg:=FALSE;

    (* 2.3. Error Code Storage Area Initialization *)
    Clear(Output_PmrErrorID); // ExecPMCR Instruction Error Code Storage Area
    Clear(Output_PmrErrorIDEx); // ExecPMCR Instruction Extended Error Code Storage Area
    Clear(Output_PmrStaErrCode); // Protocol Macro Error Code Storage Area
    Clear(Output_TransErrCode); // Transmission Error Code Storage Area

    (* 2.4. ExecPMCR Instruction Input Variable Setting *)
    Local_InPort.UnitNo:=_eUnitNo#_CBU_No00; // SCU Unit No.
    Local_InPort.PhysicPortNo:=USINT#2; // Physical Port No.
    Local_SeqNo:=UINT#900; // Protocol Macro Sequence No.
    Input_SendData[0]:=WORD#0; // Send Data Size (Unit: WORD)

    (* 2.5. Receive Data Storage Area Initialization *)
    Clear(Output_RecvData); // ExecPMCR Receive Data Storage Area
Clear(Local_RecvWordData); // Word Area for String Conversion
Clear(Local_RecvByteData); // Byte Area for String Conversion
Clear(Output_RecvMessage); // String Storage Area

    (* 2.6. Initialization End Processing *)
    Local_State:=11; // Go to 11: PMCR Communications Processing.
```

3. PMCR Communications Processing



```
(* 3. PMCR Communications Processing
  • Executes communications processing from Specified Serial Port *)
  11:
    (* 3.1. Communications Processing Status Judgment and Execution Flag Setting *)
    (* 3.1.1. Normal/Error Judgment Processing at End of Protocol Macro Execution *)
    F_TRIG_instance(J01_P2_PmrExecSta,Local_EndExecPMCR); // Protocol Macro Execution End Judgment
    IF Local_EndExecPMCR THEN
      IF NOT J01_P2_PmrSeqEndSta THEN // Protocol Macro Error End
        Output_TransErrCode:=J01_P2_TransErrSta;
        Local_Status.Error:=TRUE;
ELSE
// Normal End Processing
        Output_TransErrCode:=WORD#16#0000;
        Local_Status.Done:=TRUE;
// Receive Data String Conversion
Local_RecvWordSize:=WORD_TO_UINT(Output_RecvData[0])-1;
MemCopy(Output_RecvData[1],Local_RecvWordData[0],Local_RecvWordSize);
ToAryByte(Local_RecvWordData,_eBYTE_ORDER#_HIGH_LOW,Local_RecvByteData[0]);
Local_RecvByteSize:=Local_RecvWordSize*2;
      Output_RecvMessage:=AryToString(Local_RecvByteData[0],Local_RecvByteSize);
      END_IF;
      Local_EndExecPMCR:=FALSE;
      Local_Status.Busy:=FALSE;
      Local_ExecFlg:=FALSE;
      Local_State:=0;  // Go to 0: Communications Not Executed State.

    (* 3.1.2. Error Judgment Processing during Protocol Macro Execution *)
    ELSIF ExecPMCR_instance.Error THEN // ExecPMCR Instruction Error
      Output_PmrErrorID:=ExecPMCR_instance.ErrorID;
      Output_PmrErrorIDEx:=ExecPMCR_instance.ErrorIDEx;
      MoveDigit(J01_P2_PmrSta,USINT#0,Local_PmrStaCode,USINT#0,USINT#1);
      Output_PmrStaErrCode:=Local_PmrStaCode;
      Output_TransErrCode:=J01_P2_TransErrSta;
      Local_Status.Busy:=FALSE;
      Local_Status.Error:=TRUE;
      Local_ExecFlg:=FALSE;
      Local_State:=0;  // Go to 0: Communications Not Executed State.


     (* 3.1.3. Communications Instruction Execution Flag Setting *)
     ELSIF _Port_isAvailable AND
       NOT J01_P2_PmrExecSta AND
       NOT  ExecPMCR_instance.Busy THEN
       Local_ExecFlg:=TRUE;
     END_IF;

     (* 3.2. Communications Instruction Execution *)
     ExecPMCR_instance(
       Execute:=Local_ExecFlg,
       Port:=Local_InPort,
       SeqNo:=Local_SeqNo,
       SrcDat:=Input_SendData[0],
       DstDat:=Output_RecvData[0]);
```

4. Processing No. Error Processing



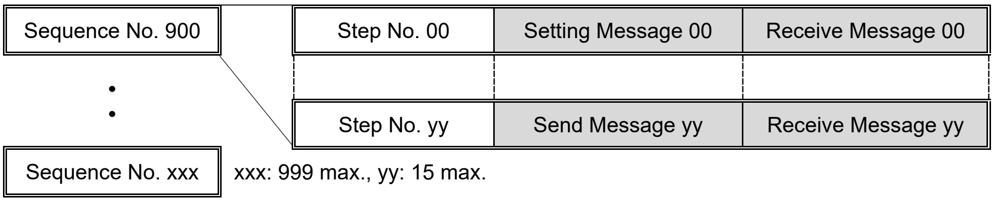
```
(* 4. Processing No. Error Processing
    • Executes error processing for non-existent processing number. *)
    99:
        Output_TransErrCode:=WORD#16#0010;
        Local_Status.Busy:=FALSE;
        Local_Status.Error:=TRUE;
        Local_State:=0;  // Go to 0: Communications Not Executed State.

    ELSE
        Local_State:=99;  // Go to 99: Processing No. Error Processing.

    END_CASE;

END_IF;
```

9.6. Protocol Macro Data

Protocol macro data consists of the following components: Sequence, Step, Send Message, Receive Message, and Reception Matrix.

● When there is only one receive message format for one step (i.e., single send and receive operation)

• One receive message and one send message are set for the step.



Sequence No. 900

Step No. 00 | Setting Message 00 | Receive Message 00

Step No. yy | Send Message yy | Receive Message yy

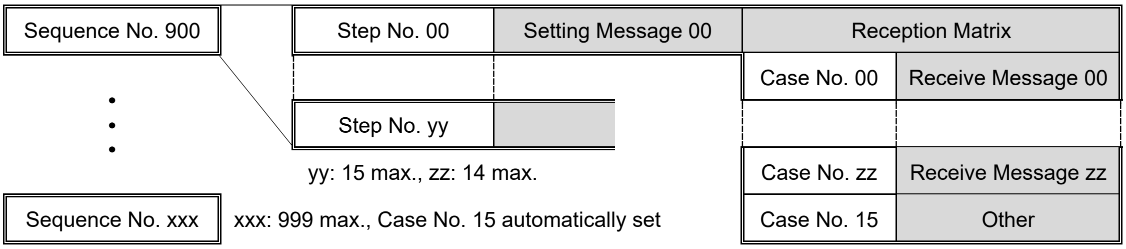
Sequence No. xxx

xxx: 999 max., yy: 15 max.

● When there are several receive message formats for one step (i.e., single send and receive operation)

• One send message and one reception matrix are set for the step.

• Several “cases” (receive messages) are set for the reception matrix.



Sequence No. 900

Step No. 00 | Setting Message 00 | Reception Matrix

Case No. 00 | Receive Message 00

Step No. yy

Case No. zz | Receive Message zz

Case No. 15 | Other

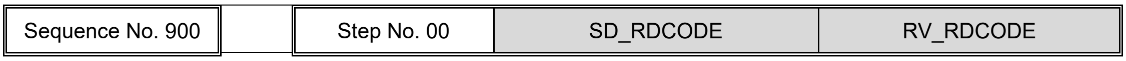
yy: 15 max., zz: 14 max.

Sequence No. xxx

xxx: 999 max., Case No. 15 automatically set

9.6.1. Structure of the Protocol Macro Data

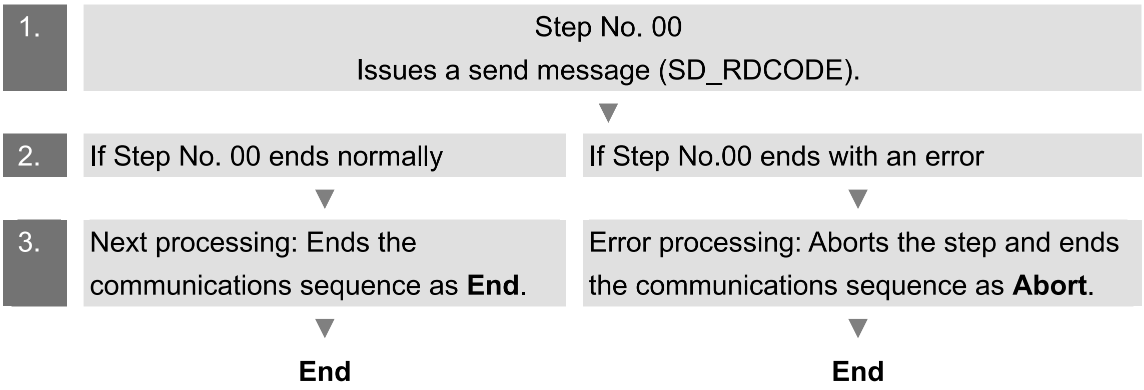
This protocol macro data has a structure without the reception matrix as shown below, since it has only one receive message (SD_RDCODE) for one send message (SD_RDCODE). For more information, refer to *9.6.6. Setting the Receive Message*.



Sequence No. 900 | Step No. 00 | SD_RDCODE | RV_RDCODE

9.6.2. Protocol Macro Processing Procedure

The protocol macro processing procedure is as follows.



1. Step No. 00
Issues a send message (SD_RDCODE).

2. If Step No. 00 ends normally

If Step No.00 ends with an error

3. Next processing: Ends the communications sequence as **End**.

Error processing: Aborts the step and ends the communications sequence as **Abort**.

End

End

9.6.3. Sequence Settings

The protocol macro data uses a “Read trigger” (Communications sequence No. 900) to read the code. The setting items for the communications sequence include **Timeout**.

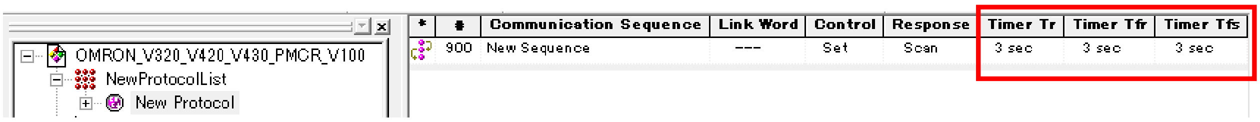
Note

For details on the sequence settings, refer to *3-2 Sequence Attributes (Common to All Steps)* in the *CX-Protocol Operation Manual (Cat. No. W344)*.

● Timeout Settings

The timeout settings (**Timer Tr**, **Timer Tfr**, and **Timer Tfs**) for the sequence are as follows.

Communications Sequence Setting Screen



OMRON_V320_V420_V430_PMCR_V100
NewProtocolList
New Protocol

*	#	Communication Sequence	Link Word	Control	Response	Timer Tr	Timer Tfr	Timer Tfs
	900	New Sequence	---	Set	Scan	3 sec	3 sec	3 sec

Settings

Item	Meaning	Description
Timer Tr	Receive Wait Monitoring Time	Monitors the time until the code reader receives the first data (header) after it enters the receive wait state in each step in the sequence. In the protocol macro data, it is set to 3 seconds.
Timer fr	Receive Completed Monitoring Time	Monitors the time until the code reader completes the receiving after it receives the first data (header) in each step in the sequence. In the protocol macro data, it is set to 3 seconds.
Timer Tfs	Send Completed Monitoring Time	Monitors the time until the code reader receives the last data after it sends the header. In the protocol macro data, it is set to 3 seconds.

Note

For details on how to calculate the monitoring time, refer to *4-5 Calculation Method of Monitoring Time* in the *CX-Protocol Operation Manual* (Cat. No. W344).

9.6.4. Step Settings

The step settings for communications sequence No.900 are shown below. The step settings include **Retry**, **Send Message** and **Recv Message** (message names), **Next**, and **Error**. The sequence in the protocol macro data consists of Step No. 00 only.

Note

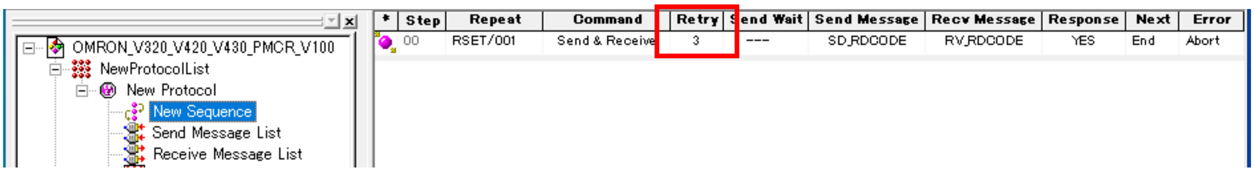
For details on the step settings, refer to *3-3 Step Attributes* in the *CX-Protocol Operation Manual* (Cat. No. W344).

- Retry Count Setting

The **Retry** setting for the step is shown below. If an error occurs, the protocol macro will retry the step the specified number of times (0 to 9 times). If the error persists even after the retries, it will go to Error Processing.

The retry count is valid only for the Send & Receive command.

Step Setting Screen



OMRON_V320_V420_V430_PMCR_V100
NewProtocolList
New Protocol
New Sequence
Send Message List
Receive Message List

*	Step	Repeat	Command	Retry	Send Wait	Send Message	Recv Message	Response	Next	Error
	00	RSET/001	Send & Receive	3	---	SD_RDCODE	RV_RDCODE	YES	End	Abort

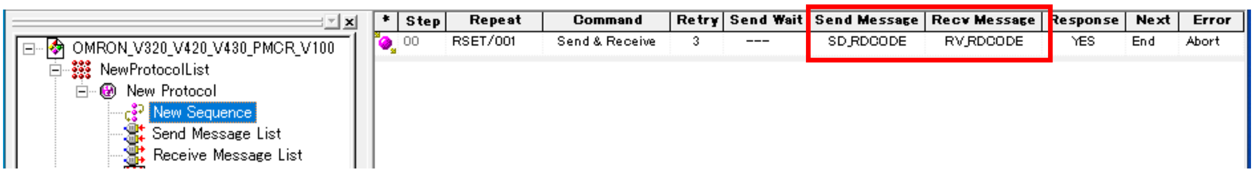
Settings

Step	Retry
00	3

- Send/Receive Message (Message Name) Settings

The **Send Message** and **Recv Message** settings for the step are shown below. The send message name and receive message name that are separately registered are set.

Step Setting Screen



OMRON_V320_V420_V430_PMCR_V100
NewProtocolList
New Protocol
New Sequence
Send Message List
Receive Message List

*	Step	Repeat	Command	Retry	Send Wait	Send Message	Recv Message	Response	Next	Error
	00	RSET/001	Send & Receive	3	---	SD_RDCODE	RV_RDCODE	YES	End	Abort

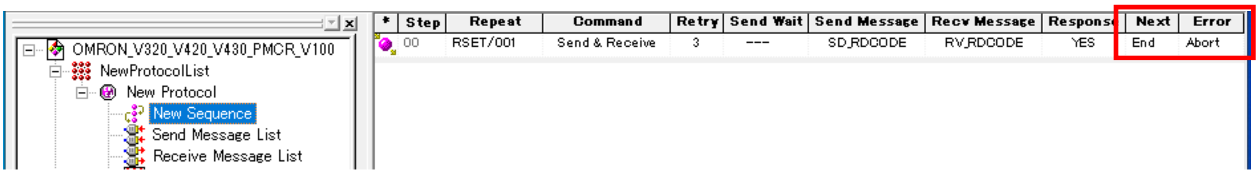
Settings

Step	Send Message	Recv Message
00	SD_RDCODE	RV_RDCODE

- Next Processing and Error Processing Settings

  The **Next** and **Error** processing settings for the step are shown below. The **Next** processing setting will be executed when the step execution ends normally. The **Error** processing setting will be executed when a communication error occurs.

Step Setting Screen



OMRON_V320_V420_V430_PMCR_V100
- NewProtocolList
  - New Protocol
    - New Sequence
    - Send Message List
    - Receive Message List

*	Step	Repeat	Command	Retry	Send Wait	Send Message	Recv Message	Respons	Next	Error
●	00	RSET/001	Send & Receive	3	---	SD_RDCODE	RV_RDCODE	YES	End	Abort

Settings

Step	Next	Error
00	End	Abort

Processing Items

Processing	Description
End	Ends the communications sequence.
Next	Advances to the next step No.
Abort	Aborts the step and ends the communications sequence.
Goto	Jumps to the specified step No.
Reception Matrix	Executes the processing specified by the reception matrix setting.

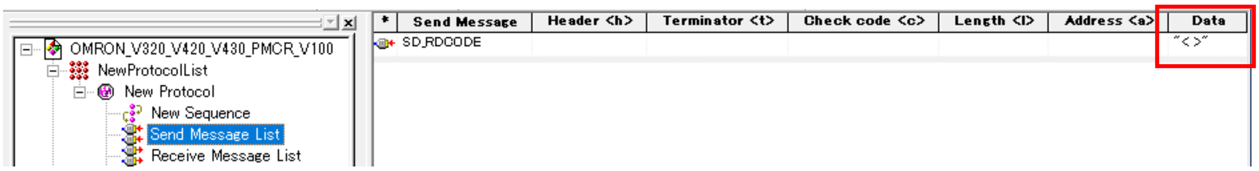
9.6.5. Send Message Settings

The send message settings are shown below.

Note

For details on the Send Message settings, refer to *3-4 Communication Message Attributes* in the *CX-Protocol Operation Manual* (Cat. No. W344).

Send Message Setting Screen



*	Send Message	Header <h>	Terminator <t>	Check code <c>	Length <l>	Address <a>	Data
	SD_RDCODE						"< >"

● Settings for Send Message SD_RDCODE

Settings

$\underset{(1)}{\underline{< >}}$

No.	Code	Description
(2)	"< >"	Constant in ASCII

* Header <h> and Footer <f> is set to *None*.

Send Message Command Format

This is the command format of messages sent from the SCU unit to the code reader according to the SD_RDCODE setting.

"< >"

Command	Number of bytes	Remarks
"< >"	1	Fixed: “<>” (16# 3C203E) (Code reader command: “Read Trigger”)

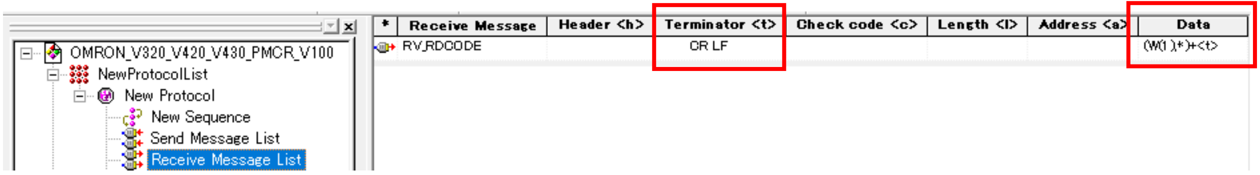
9.6.6. Receive Message Settings

The receive message settings are shown below.

Note

For details on the receive message settings, refer to *3-4 Communication Message Attributes* in the *CX-Protocol Operation Manual* (Cat. No. W344).

Receive Message Setting Screen



*	Receive Message	Header <h>	Terminator <t>	Check code <c>	Length <l>	Address <a>	Data
	RV_RDCODE		CR LF				(W(1),*)+<t>

● Settings for Receive Message RV_RDCODE

Settings

(W(1),*)+<t>
(1) (2)

No.	Code	Description
(1)	(W(1),*)	Message data: Variable All variable-length data is fetched and stored in array elements after the 1st array element of the array variable [n] specified in *DstDat* of ExecPMCR instruction (from *Output_RecvData [1]* onwards for this program).
(2)	<t>(Terminator)	Message data: Constant in hex, Type: Code, Data: CR LF

* Header <h> is set to *None*.

Response Format of Receive Message

This is the response format of the receive message from the code reader that the SCU unit receives according to the RV_RDCODE setting.

Code value read	<CR><LF>

Command	Number of bytes	Remarks
Data	Variable length	Variable: Code value read
Terminator	1	Fixed: <CR> <LF>(16#0D0A)

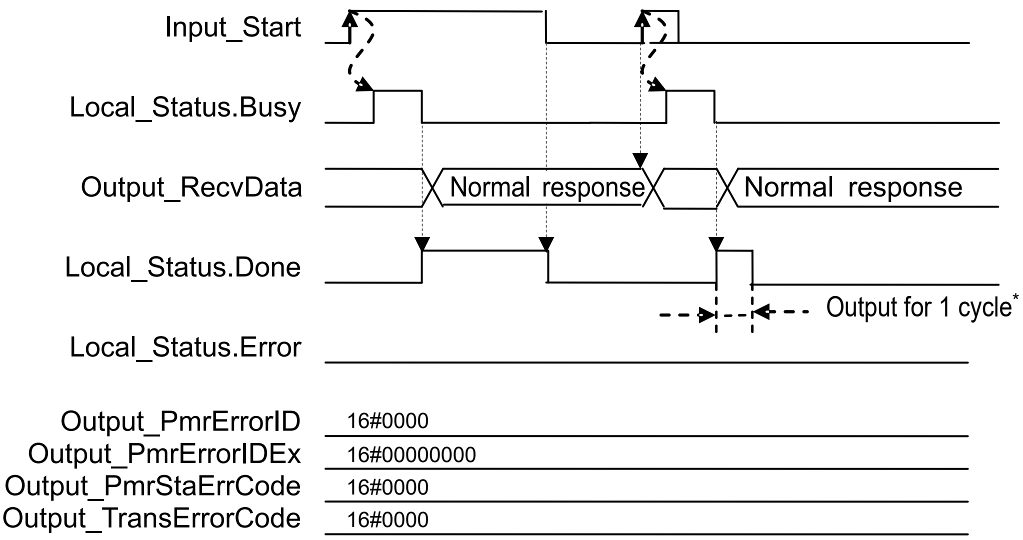
9.7. Timing Chart

The timing chart for the program is shown below.

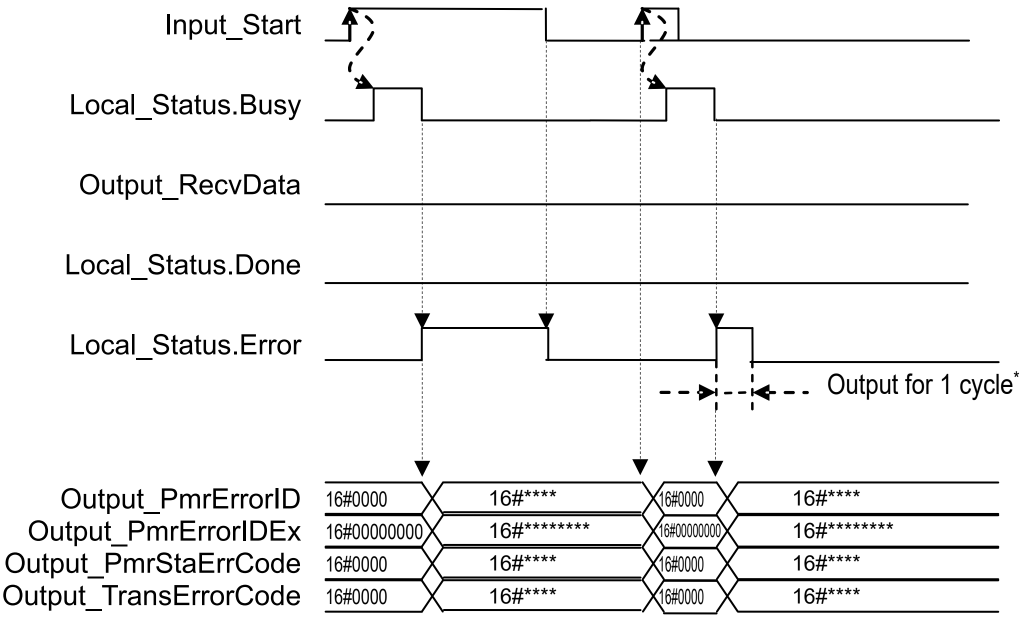
The timing chart patterns are defined as follows.

Pattern	Normal End	Error End 1 ExecPMCR Instruction Error	Error End 2 Protocol Macro Error
Command	Normal	Error	Error
Code reader	Normal	Normal or Error	Normal or Error
Response	Yes	No	No

● Normal End



● Error End 1, 2



* If *Input_Start* changes from *TRUE* to *FALSE* during execution (Busy = TRUE), Normal or Error is output for one cycle after completion of the processing (Busy = FALSE).

9.8. Error Processing

This section provides information on errors that can occur during the execution of the program.

9.8.1. ExecPMCR Instruction Error

The table below shows error codes that occur when the ExecPMCR instruction ends with an error.

- Output_PmrErrorID

After execution of the ExecPMCR instruction, the content of *ExecPMCR_instance.ErrorID* is set in this variable.

Code	Description
16#0000	Normal End
16#0400	An input parameter for an instruction exceeded the valid range for an input variable.
16#0406	A memory address or data size that was specified for the instruction is not suitable.
16#0407	The calculation result of the instruction exceeded the valid range for the data area for output parameters.
16#0800	An error occurred in FINS command communications.
16#0801	The FINS port is in use.

Note

For details on ErrorID, refer to *A-1 Error Codes That You Can Check with ErrorID* and *A-2 Error Codes* in *Appendices* in the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

- Output_PmrErrorIDEx

When the value of *ExecPMCR_instance.ErrorID* is *16#0800*, the content of *ExecPMCR_instance.ErrorIDEx* is set in this variable.

Code	Description
16#00000000	Normal End
16#00001106	The value of *SeqNo* is the communications sequence number that has not been registered.
16#00002201	This instruction is already being executed. The values of Busy and *J01_P2_PmrExecSta* are *TRUE*.
16#00002202	The instruction cannot be executed because the protocol is being switched.
16#00002401	The protocol macro data has an incorrect sum value, or is being transferred.

Note

For details on ErrorIDEx, refer to *Serial Communications Instructions (ExecPMCR)* in *Section 2 Instruction Descriptions* in the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

9.8.2. Protocol Macro Error Codes

The table below shows the protocol macro error codes for errors caused by the protocol macro.

They are Set in *Output_PmrStaErrCode*.

- Output_PmrStaErrCode

The contents of the lower four bits of *J01_P2_PmrSta* are set in this variable.

Code	Description
16#0000	No error
16#0002	Sequence number error
16#0003	Data read/write area exceeded error
16#0004	Protocol data syntax error

Note

For details on the protocol macro error code (PmrStaErrCode), refer to *4-2-6 Protocol Status* in the *CJ-Series Serial Communications Units Operation Manual for NJ-series CPU Unit* (Cat. No. W494).

9.8.3. Transmission Error Status

The table below shows the error flag string for transmission errors that can occur.

They are Set in *Output_TransErrCode*.

- Output_TransErrCode

The contents of *J01_P2_TransErrSta* are set in this variable.

Bit	Description	
15	1: Transmission error	0: No transmission error
14	1: Send finished monitoring time exceeded	0: Normal
13	1: Receive finished monitoring time exceeded	0: Normal
12	1: Receive wait monitoring time exceeded	0: Normal
8 to 11	Retry	
7	1 :FCS check error	0: Normal
6	1 :Commands error	0: Normal
5	1: Timeout	0: Normal
4	1: Overrun error	0: Normal
3	1: Framing error	0: Normal
2	1: Parity error	0: Normal
0, 1	(Not used)	

Note

For details on the Transmission Error Status (TransErrCode), refer to *4-2-5 Device Variables for CJ-series Unit for Status* in the *CJ-series Serial Communications Units Operation Manual for NJ-series CPU Unit* (Cat. No. W494).

10. Revision History

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

OMRON Corporation **Industrial Automation Company**

Kyoto, JAPAN **Contact : www.ia.omron.com**

Regional Headquarters

OMRON EUROPE B.V.
Wegalaan 67-69, 2132 JD Hoofddorp
The Netherlands
Tel: (31) 2356-81-300 Fax: (31) 2356-81-388

OMRON ASIA PACIFIC PTE. LTD.
438B Alexandra Road, #08-01/02 Alexandra
Technopark, Singapore 119968
Tel: (65) 6835-3011 Fax: (65) 6835-2711

OMRON ELECTRONICS LLC
2895 Greenspoint Parkway, Suite 200
Hoffman Estates, IL 60169 U.S.A.
Tel: (1) 847-843-7900 Fax: (1) 847-843-7787

OMRON (CHINA) CO., LTD.
Room 2211, Bank of China Tower,
200 Yin Cheng Zhong Road,
PuDong New Area, Shanghai, 200120, China
Tel: (86) 21-5037-2222 Fax: (86) 21-5037-2200

Authorized Distributor:

Cat. No. Z430-E1-01 0323