## OMRON

## **Machine Automation Controller**

**NJ/NX-series** 

## **CPU Unit Built-in EtherNet/IP™ Port**

## **User's Manual**

NX701-17□□

NX701-16□□

NJ501-15□□

NJ501-14□□

NJ501-13□□

NJ301-12□□

NJ301-11□□

**NJ101-10**□□

140101-10

**NJ101-90**□□

**CPU Unit** 





#### - NOTE -

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.

No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained in this publication.

#### Trademarks

- · Sysmac and SYSMAC are trademarks or registered trademarks of OMRON Corporation in Japan and other countries for OMRON factory automation products.
- · Microsoft, Windows, Windows Vista, Excel, and Visual Basic are either registered trademarks or trademarks of Microsoft Corporation in the USA and other countries.
- EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
- ODVA, CIP, CompoNet, DeviceNet, and EtherNet/IP are trademarks of ODVA.
- The SD and SDHC logos are trademarks of SD-3C, LLC.



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

#### - Copyrights

Microsoft product screen shots reprinted with permission from Microsoft Corporation.

## Introduction

Thank you for purchasing an NJ/NX-series CPU Unit.

This manual contains information that is necessary to use the NJ/NX-series CPU Unit. Please read this manual and make sure you understand the functionality and performance of the NJ/NX-series CPU Unit before you attempt to use it in a control system.

Keep this manual in a safe place where it will be available for reference during operation.

### **Intended Audience**

This manual is intended for the following personnel, who must also have knowledge of electrical systems (an electrical engineer or the equivalent).

- Personnel in charge of introducing FA systems.
- Personnel in charge of designing FA systems.
- Personnel in charge of installing and maintaining FA systems.
- Personnel in charge of managing FA systems and facilities.

For programming, this manual is intended for personnel who understand the programming language specifications in international standard IEC 61131-3 or Japanese standard JIS B 3503.

## **Applicable Products**

This	manual	covers	the	following	products.
11113	manuai	COVEIS	uic	IOHOWING	products.

- NX-series CPU Units
  - NX701-17□□
  - NX701-16□□
- NJ-series CPU Units
  - NJ501-15□□
  - NJ501-14□□
  - NJ501-13□□
  - NJ301-12□□
  - NJ301-11□□
  - NJ101-10
  - NJ101-90□□

Part of the specifications and restrictions for the CPU Units are given in other manuals. Refer to *Relevant Manuals* on page 2 and *Related Manuals* on page 25.

## **Relevant Manuals**

The following table provides the relevant manuals for the NJ/NX-series CPU Units.

Read all of the manuals that are relevant to your system configuration and application before you use the NJ/NX-series CPU Unit.

Most operations are performed from the Sysmac Studio Automation Software. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for information on the Sysmac Studio.

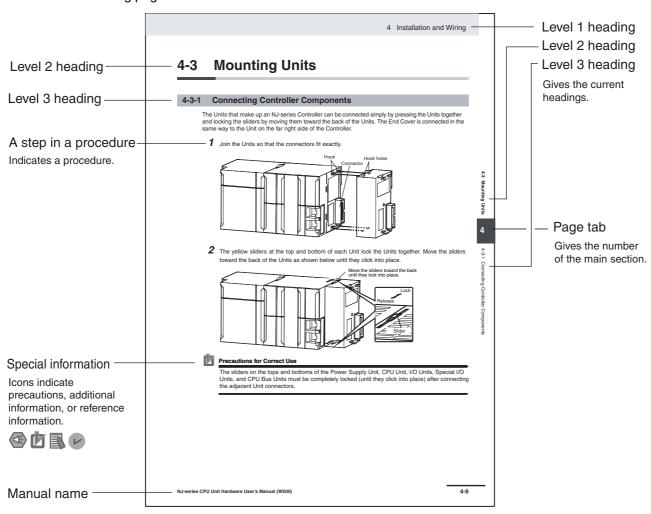
															Ма	nua	al													
			В	Bas	ic i	nfo	orm	atic	n																					
Purpose of use	Manual	Hardware User's	NX-series CPU Unit	Manual	Hardware User's	N.J-series CPU Unit	Manual Osers	NJ/NX-series CPU Unit	Manual	Instructions Reference	NJ/NX-series	Manual	Motion Control User's	NJ/NX-series CPU Unit	Reference Manual	Control Instructions	N I/NY-series Motion	Hear's Manual	NJ/NX-series CPU Unit	Port User's Manual	Built-in EtherNet/IP	NJ/NX-series CPU Unit	User's Manual	Connection CPU Unit	NJ-series Database	Manual	CPU Units User's	NJ-series SECS/GEM	Troubleshooting Manual	NJ/NX-series
Introduction to NX-series Controllers		•																										T		
Introduction to NJ-series Controllers					•																									
Setting devices and hardware														İ																
Using motion control						Ī							•																	
Using EtherCAT		_			_	ļ											T	(										T		
Using EtherNet/IP		•			•	j											T			T	•							T		
Using the database connection service						ļ											T							•				T		
Using the GEM Services						ļ											T										•	T		
Software settings						İ																						Ì		
Using motion control													•																	
Using EtherCAT																		(										T		
Using EtherNet/IP							•														•									
Using the database connection service																								•						
Using the GEM Services																											•			
Writing the user program																														
Using motion control													•			•														
Using EtherCAT																		(												
Using EtherNet/IP										•											•									
Using the database connection service																								•						
Using the GEM Services																											•			
Programming error processing																													•	,
Testing operation and debugging																														
Using motion control													•															T		_
Using EtherCAT							_										T	•		T					$\neg$			1		
Using EtherNet/IP							•	•									T				•				$\neg$			T		
Using the database connection service																	T							•				T		
Using the GEM Services																	T										•	T		
Learning about error management and						j							_								_			_	$\neg$		_	T	_	_
corrections*1		•			•		4	<b>\</b>					<b>A</b>					4	<b>A</b>		•			•			<b>A</b>			
Maintenance						j																			$\neg$			T		_
Using motion control		_			_	ļ							•				T											T		
Using EtherCAT		•			•	ſ												•							$\neg$			T		
Using EtherNet/IP						j															•							T		

<sup>\*1</sup> Refer to the *NJ/NX-series Troubleshooting Manual* (Cat. No. W503) for the error management concepts and an overview of the error items. Refer to the manuals that are indicated with triangles for details on errors for the corresponding Units.

## **Manual Structure**

## **Page Structure**

The following page structure is used in this manual.



This illustration is provided only as a sample. It may not literally appear in this manual.

## **Special Information**

Special information in this manual is classified as follows:



#### **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.



#### Additional Information

Additional information to read as required.

This information is provided to increase understanding or make operation easier.



#### **Version Information**

Information on differences in specifications and functionality for CPU Units with different unit versions and for different versions of the Sysmac Studio is given.

Note References are provided to more detailed or related information.

## **Precaution on Terminology**

In this manual, "download" refers to transferring data from the Sysmac Studio to the physical Controller and "upload" refers to transferring data from the physical Controller to the Sysmac Studio.

For the Sysmac Studio, synchronization is used to both upload and download data. Here, "synchronize" means to automatically compare the data for the Sysmac Studio on the computer with the data in the physical Controller and transfer the data in the direction that is specified by the user.

## **Sections in this Manual**

				1	10
1	Introduction	10	FTP Server	2	11
				3	12
2	Installing Ethernet Networks	11	FTP Client	4	13
	System-defined Variables Related to the Built-in	10	Automatic Clock		10
3	EtherNet/IP Port	12	Adjustment	5	14
4	Determining IP Addresses	13	SNMP Agent	6	15
5	Sysmac Studio Settings for the Built-in EtherNet/IP Port	14	Communications Per- formance and Commu- nications Load	7	А
6	Testing Communications	15	Troubleshooting	8	1
					9
7	Tag Data Link Functions	A	Appendices	_/ / /	
8	Message Communications	1	Index		
9	Socket Service				

Sections in this Manual

## **CONTENTS**

Ir	ntroduct	ion	1
R	Relevant	Manuals	2
IV	lanual S	Structure	
s	ections	in this Manual	5
T	erms an	nd Conditions Agreement	15
s	Safety Pr	ecautions	17
Р	recautio	ons for Safe Use	18
Р	recautio	ons for Correct Use	19
R	Regulatio	ons and Standards	20
V	ersions		22
R	Related N	Manuals	25
R	Revision	History	27
Section	<b>1</b>	Introduction	
1-	-1 Introc 1-1-1 1-1-2	Suction  EtherNet/IP Features  Features of Built-in EtherNet/IP Port on NJ/NX-series CPU Units	1-2
1-	-2 Syste 1-2-1 1-2-2	Perm Configuration and Configuration Devices  Devices Required to Construct a Network	1-5
1.	-3 Built- 1-3-1 1-3-2	in EtherNet/IP Port	1-7
1.	1-4-1 1-4-2 1-4-3 1-4-4 1-4-5 1-4-6 1-4-7 1-4-8 1-4-9	CIP (Common Industrial Protocol) Communications Services IP Routing BOOTP Client FTP Server FTP Client Automatic Clock Adjustment Socket Service Specifying Host Names SNMP Agent	1-15 1-18 1-19 1-19 1-20 1-20 1-21
1-	-5 Ether 1-5-1	Net/IP Communications Procedures Outline	_

Section 2	Installing Ethernet	<b>Networks</b>
-----------	---------------------	-----------------

2-1	Selecting the Network Devices	2-2
	2-1-1 Recommended Network Devices	
	2-1-2 Ethernet Switch Types	
	2-1-3 Ethernet Switch Functions	
2-2	Network Installation	
2-2	2-2-1 Basic Installation Precautions	
	2-2-2 Recommended Network Devices	
	2-2-3 Precautions When Laying Twisted-pair Cable	-
	2-2-4 Precautions When Installing and Connecting Ethernet Switches	2-10
2-3	Connecting to the Network	2-11
	2-3-1 Ethernet Connectors	
	2-3-2 Connecting the Cable	2-12
Section 3	System-defined Variables Related to the Built-in EtherNet/IP Port	
3-1	System-defined Variables Related to the Built-in EtherNet/IP Port	3-2
3-2	System-defined Variables	3-3
3-3	Specifications for Individual System-defined Variables	3-33
Section 4	Determining IP Addresses	
4-1	IP Addresses	
	4-1-1 IP Address Configuration	
	4-1-2 Allocating IP Addresses	
	4-1-4 CIDR	
4-2	Built-in EtherNet/IP Port IP Address Settings	4-4
7-2	4-2-1 Determining IP Addresses	
	4-2-2 Setting IP Addresses	
	4-2-3 Online Connection	
	4-2-4 Checking the Current IP Address	
4-3	Private and Global Addresses	4-12
	4-3-1 Private and Global Addresses	
	4-3-2 Using a Private Address for the Built-in EtherNet/IP Port	
	4-5-5 Carry a Global Address for the Built-III Etherweyn Tort	
Section 5	Sysmac Studio Settings for the Built-in EtherNet/IF	Port
5-1	TCP/IP Settings Display	5-2
5-2	Link Settings Display	5-6
5-3	FTP Display	5-7
5-4	NTP Settings Display	5-8
5-5	SNMP Settings Display	
5-6	SNMP Trap Settings Display	
3-0	Tap Collings Dioplay	5-12

## **Section 6** Testing Communications

	6-1	Testing	g Communications	6-2
		6-1-1	PING Command	6-2
		6-1-2	Using the PING Command	6-2
		6-1-3	Host Computer Operation	6-3
Sect	ion 7	-	Tag Data Link Functions	
	— 7-1	Introdu	uction to Tag Data Links	7.0
	7-1	7-1-1	Tag Data Links	
		7-1-2	Data Link Data Areas	
		7-1-3	Tag Data Link Functions and Specifications	
		7-1-4	Overview of Operation	
		7-1-5	Starting and Stopping Tag Data Links	
		7-1-6	Controller Status	
		7-1-7	Concurrency of Tag Data Link Data	
	7-2	Settino	g Tag Data Links	7-18
		7-2-1	Starting the Network Configurator	
		7-2-2	Tag Data Link Setting Procedure	
		7-2-3	Registering Devices	
		7-2-4	Creating Tags and Tag Sets	
		7-2-5	Connection Settings	
		7-2-6	Creating Connections Using the Wizard	
		7-2-7	Creating Connections by Device Dragging and Dropping	
		7-2-8	Connecting the Network Configurator to the Network	
		7-2-9	Downloading Tag Data Link Parameters	
		7-2-10	Uploading Tag Data Link Parameters	
		7-2-11	Verifying the Tag Data Links	
		7-2-12	Starting and Stopping Tag Data Links	
		7-2-13	Clearing the Device Parameters	
		7-2-14	Saving the Network Configuration File	
		7-2-15	Reading a Network Configuration File	
		7-2-16	Checking Connections	
		7-2-17	Changing Devices	
		7-2-18	Displaying Device Status	
	7-3	l addei	r Programming for Tag Data Links	
	1-0	7-3-1	Ladder Programming for Tag Data Links	
		7-3-1	Status Flags Related to Tag Data Links	
	7-4	_	ita Links with Other Models	
	7-4	iay Da	tid Liliks with Other Models	1-02
Sect	ion 8	I	Message Communications	
_	8-1	Overvi	ew of the CIP Message Communications Service	8-3
	•	8-1-1	Overview of the CIP Message Communications Service	
		8-1-2	Message Communications Service Specifications	
	0.0		·	
	8-2		essaging Communications Client Function	
		8-2-1	Overview	
		8-2-2	CIP Communications Instructions	
		8-2-3	Using CIP Communications Instructions	
		8-2-4	Route Path	
		8-2-5	Preparing Derivative Data Types to Use CIP Communications Instructions	
		0 0 6	Sample Programming for CIP Connectionless (UCMM) Message Communications	8-22
		8-2-6		
		8-2-7	Sample Programming for CIP Connection (Class 3) Message Communications	8-27
				8-27 8-34

	8-3	CIP Communication Server Functions	
		8-3-1 CIP Message Structure for Accessing CIP Objects	
		8-3-2 CIP Message Structure for Accessing Variables	
	8-4		
		8-4-1 Examples of CIP Object Specifications	
		8-4-3 Logical Segment	
		8-4-4 Data Segment	
		8-4-5 Specifying Variable Names in Request Paths	8-44
	8-5		8-47
		8-5-1 CIP Objects Sent to the Built-in EtherNet/IP Port	
		8-5-2 Identity Object (Class ID: 01 Hex)	
		8-5-3 TCP/IP Interface Object (Class ID: F5 hex)	
		8-5-4 Ethernet Link Object (Class ID: F6 Hex)	
	8-6		
	0-0	8-6-1 Read and Write Services for Variables	
		8-6-2 Write Service for Variables	
	8-7		
	0-7	8-7-1 Data Type Codes	
		8-7-2 Common Format	
		8-7-3 Elementary Data Types	
		8-7-4 Derived Data Types	8-66
Section	on 9	9 Socket Service	
	 9-1	Basic Knowledge on Socket Communications	9-2
	• .	9-1-1 Sockets	
		9-1-2 Port Numbers for Socket Services	
	9-2	Basic Knowledge on Protocols	9-3
	_	9-2-1 Differences between TCP and UDP	9-3
		9-2-2 Fragmenting of Send Data	
		9-2-3 Data Reception Processing	
		9-2-4 Broadcasting	
	9-3	Overview of Built-in EtherNet/IP Port Socket Services	
		9-3-1 Overview	-
		9-3-2 Procedure	
	9-4	3	
		9-4-1 Settings Required for the Socket Services	9-12
	9-5	Socket Service Instructions	9-13
	9-6	Details on Using the Socket Services	9-14
		9-6-1 Using the Socket Services	9-14
		9-6-2 Procedure to Use Socket Services	
		9-6-3 Timing Chart for Output Variables Used in Communications	
		9-6-4 UDP Sample Programming	
		2 2 2 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4 5 4	
	9-7		
		9-7-1 Precautions for UDP and TCP Socket Services	
		9-7-3 Precautions for TCP Socket Services	
Section	on 1	10 FTP Server	
	— 1∩₋1	1 Overview and Specifications	
	10-1	10-1-1 Overview	
		10-1-2 Specifications	_

	10-2	FTP Server Function Details	
		10-2-1 Supported Files	
		10-2-2 Connecting to the FTP Server	10-3
	10-3	Using the FTP Server Function	10-5
		10-3-1 Procedure	
		10-3-2 List of Settings Required for the FTP Server Function	10-5
	10-4	FTP Server Application Example	10-6
	10-5	Using FTP Commands	10-8
		10-5-1 Table of Commands	10-8
		10-5-2 Using the Commands	10-9
	10-6	Using SD Memory Card Operations	10-15
		10-6-1 SD Memory Card Types	10-15
		10-6-2 File Types	
		10-6-3 Initializing SD Memory Cards	
	40.7	10-6-4 Format of Variable Data	
	10-7	Application Example from a Host Computer	10-18
		4	
Section	on 1	1 FTP Client	
	11-1	Using the FTP Client to Transfer Files	11-2
		11-1-1 Transferring Files	
		11-1-2 Connectable FTP Servers	11-3
		11-1-3 File Transfer Options	
		11-1-4 Other Functions	11-4
	11-2	FTP Client Communications Instructions	11-5
		11-2-1 Functions of the FTP Client Communications Instructions	
		11-2-2 Restrictions on the FTP Client Communications Instructions	11-7
	11-3	FTP Client Application Example	11-8
Section	on 1	2 Automatic Clock Adjustment	
	12-1	Automatic Clock Adjustment	12-2
	. 4-1	12-1-1 Overview	
		12-1-2 Specifications	
	12-2	Procedure to Use the Automatic Clock Adjustment Function	12-4
	12-2	12-2-1 Procedure	
		12-2-2 Settings Required for Automatic Clock Adjustment	
Section	on 1	3 SNMP Agent	
	13-1	SNMP Agent	13-2
		13-1-1 Overview	
		13-1-2 Specifications	
		13-1-3 SNMP Messages	
		13-1-4 MIB Specifications	13-4
	13-2	Procedure to Use the SNMP Agent	13-21
		13-2-1 Procedures	
		13-2-2 Settings Required for the SNMP Agent	13-21

## Section 14 Communications Performance and Communications Load

	14-1	Communications System	
		14-1-1 Tag Data Link Communications Method	
		14-1-2 Calculating the Number of Connections	
		14-1-3 Packet Interval (RPI) Accuracy	14-5
	14-2	Adjusting the Communications Load	14-6
		14-2-1 Checking Bandwidth Usage for Tag Data Links	
		14-2-2 Tag Data Link Bandwidth Usage and RPI	
		14-2-3 Adjusting Device Bandwidth Usage	
		14-2-4 Changing the RPI	
		14-2-5 RPI Setting Examples	14-16
	14-3	I/O Response Time in Tag Data Links	
		14-3-1 Timing of Data Transmissions	
		14-3-2 Built-in EtherNet/IP Port Data Processing Time	
		14-3-3 Relationship between Task Periods and Packet Intervals (RPIs)	
		14-3-4 Maximum Tag Data Link I/O Response Time	14-25
	14-4	Message Service Transmission Delay	14-27
		•	
<b>.</b>	_		
Section	n 1	5 Troubleshooting	
	15-1	Overview of Errors	
		15-1-1 How to Check for Errors	
		15-1-2 Errors Related to the EtherNet/IP Function Module	15-8
	15-2	Troubleshooting	15-12
		15-2-1 Error Table	
		15-2-2 Error Descriptions	
		15-2-3 Troubleshooting	15-50
	15-3	Checking Status with the Network Configurator	15-51
		15-3-1 The Network Configurator's Device Monitor Function	
		15-3-2 Connection Status Codes and Troubleshooting	15-58
Apper	ndic	es	
	A-1	Functional Comparison of EtherNet/IP Ports on NJ/NX-series CPU Units and	
		Other Series	A-2
	Δ-2	Use the Sysmac Studio to Set the Tag Data Links (EtherNet/IP Connections)	Δ-3
		A-2-1 Overview of the Tag Data Links (EtherNet/IP Connections) Settings with the Sysmac S	
		A-2-2 Procedure to Make the EtherNet/IP Connection Settings with the Sysmac Studio	
		A-2-3 EtherNet/IP Connection Settings	
		A-2-4 Making the EtherNet/IP Connection Settings with the Sysmac Studio	
		A-2-5 Checking Communications Status with the Sysmac Studio and Troubleshooting	
		A-2-6 Troubleshooting	A-38
	<b>A-3</b>	EDS File Management	A-44
		A-3-1 Installing EDS Files	
		A-3-2 Creating EDS Files	
		A-3-3 Deleting EDS Files	
		A-3-4 Saving EDS Files	
		A-3-5 Searching EDS File Proportion	
		A-3-6 Displaying EDS File Properties	
		A-0-1 Creating EDO Index Files	A-48

<b>A-4</b>		utions for Using the Network Configurator on Windows XP, Window	•
	Wind	lows 7	A-49
	A-4-1	Changing Windows Firewall Settings	A-49
A-5	Variab	ole Memory Allocation Methods	A-52
	A-5-1	Variable Memory Allocation Rules	A-52
		Important Case Examples	
<b>A-6</b>	Preca	utions When Accessing External Outputs in CPU Units	A-64
<b>A-7</b>	TCP S	tate Transitions	A-65
A-8	Versio	on Information	A-67

## Index

## **CONTENTS**

## **Terms and Conditions Agreement**

## Warranty, Limitations of Liability

## **Warranties**

### Exclusive Warranty

Omron's exclusive warranty is that the Products will be free from defects in materials and workmanship for a period of twelve months from the date of sale by Omron (or such other period expressed in writing by Omron). Omron disclaims all other warranties, express or implied.

#### Limitations

OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, ABOUT NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OF THE PRODUCTS. BUYER ACKNOWLEDGES THAT IT ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE.

Omron further disclaims all warranties and responsibility of any type for claims or expenses based on infringement by the Products or otherwise of any intellectual property right.

#### Buyer Remedy

Omron's sole obligation hereunder shall be, at Omron's election, to (i) replace (in the form originally shipped with Buyer responsible for labor charges for removal or replacement thereof) the non-complying Product, (ii) repair the non-complying Product, or (iii) repay or credit Buyer an amount equal to the purchase price of the non-complying Product; provided that in no event shall Omron be responsible for warranty, repair, indemnity or any other claims or expenses regarding the Products unless Omron's analysis confirms that the Products were properly handled, stored, installed and maintained and not subject to contamination, abuse, misuse or inappropriate modification. Return of any Products by Buyer must be approved in writing by Omron before shipment. Omron Companies shall not be liable for the suitability or unsuitability or the results from the use of Products in combination with any electrical or electronic components, circuits, system assemblies or any other materials or substances or environments. Any advice, recommendations or information given orally or in writing, are not to be construed as an amendment or addition to the above warranty.

See http://www.omron.com/global/ or contact your Omron representative for published information.

## Limitation on Liability; Etc

OMRON COMPANIES SHALL NOT BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR PRODUCTION OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED IN CONTRACT, WARRANTY, NEGLIGENCE OR STRICT LIABILITY.

Further, in no event shall liability of Omron Companies exceed the individual price of the Product on which liability is asserted.

## **Application Considerations**

## Suitability of Use

Omron Companies shall not be responsible for conformity with any standards, codes or regulations which apply to the combination of the Product in the Buyer's application or use of the Product. At Buyer's request, Omron will provide applicable third party certification documents identifying ratings and limitations of use which apply to the Product. This information by itself is not sufficient for a complete determination of the suitability of the Product in combination with the end product, machine, system, or other application or use. Buyer shall be solely responsible for determining appropriateness of the particular Product with respect to Buyer's application, product or system. Buyer shall take application responsibility in all cases.

NEVER USE THE PRODUCT FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCT(S) IS PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

## **Programmable Products**

Omron Companies shall not be responsible for the user's programming of a programmable Product, or any consequence thereof.

### **Disclaimers**

### **Performance Data**

Data presented in Omron Company websites, catalogs and other materials is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of Omron's test conditions, and the user must correlate it to actual application requirements. Actual performance is subject to the Omron's Warranty and Limitations of Liability.

## **Change in Specifications**

Product specifications and accessories may be changed at any time based on improvements and other reasons. It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Product may be changed without any notice. When in doubt, special part numbers may be assigned to fix or establish key specifications for your application. Please consult with your Omron's representative at any time to confirm actual specifications of purchased Product.

### **Errors and Omissions**

Information presented by Omron Companies has been checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical or proofreading errors or omissions.

## **Safety Precautions**

Refer to the following manuals for safety precautions.

- NX-series CPU Unit Hardware User's Manual (Cat. No. W535)
- NJ-series CPU Unit Hardware User's Manual (Cat No. W500)
- NJ/NX-series CPU Unit Software User's Manual (Cat No. W501)

## **Precautions for Safe Use**

Refer to the following manuals for precautions for safe use.

- NX-series CPU Unit Hardware User's Manual (Cat. No. W535)
- NJ-series CPU Unit Hardware User's Manual (Cat No. W500)
- NJ/NX-series CPU Unit Software User's Manual (Cat No. W501)

## **Precautions for Correct Use**

Refer to the following manuals for precautions for correct use.

- NX-series CPU Unit Hardware User's Manual (Cat. No. W535)
- NJ-series CPU Unit Hardware User's Manual (Cat No. W500)
- NJ/NX-series CPU Unit Software User's Manual (Cat No. W501)

## Regulations and Standards

#### **Conformance to EC Directives**

## **Applicable Directives**

- · EMC Directives
- · Low Voltage Directive

## **Concepts**

### EMC Directive

OMRON devices that comply with EC Directives also conform to the related EMC standards so that they can be more easily built into other devices or the overall machine. The actual products have been checked for conformity to EMC standards.\*

Whether the products conform to the standards in the system used by the customer, however, must be checked by the customer. EMC-related performance of the OMRON devices that comply with EC Directives will vary depending on the configuration, wiring, and other conditions of the equipment or control panel on which the OMRON devices are installed. The customer must, therefore, perform the final check to confirm that devices and the overall machine conform to EMC standards.

\* Applicable EMC (Electromagnetic Compatibility) standards are as follows:
 EMS (Electromagnetic Susceptibility): EN 61131-2 and EN 61000-6-2
 EMI (Electromagnetic Interference): EN 61131-2 and EN 61000-6-4 (Radiated emission: 10-m regulations)

#### Low Voltage Directive

Always ensure that devices operating at voltages of 50 to 1,000 VAC and 75 to 1,500 VDC meet the required safety standards. The applicable directive is EN 61131-2.

#### Conformance to EC Directives

The NJ/NX-series Controllers comply with EC Directives. To ensure that the machine or device in which the NJ/NX-series Controller is used complies with EC Directives, the Controller must be installed as follows:

- The NJ/NX-series Controller must be installed within a control panel.
- You must use reinforced insulation or double insulation for the DC power supplies connected to DC Power Supply Units and I/O Units.
- NJ/NX-series Controllers that comply with EC Directives also conform to the Common Emission Standard (EN 61000-6-4). Radiated emission characteristics (10-m regulations) may vary depending on the configuration of the control panel used, other devices connected to the control panel, wiring, and other conditions.

You must therefore confirm that the overall machine or equipment complies with EC Directives.

### Conformance to KC Standards

Observe the following precaution if you use NX-series Units in Korea.

A 급 기기 (업무용 방송통신기자재) 이 기기는 업무용(A 급) 전자파적합기기로서 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.

Class A Device (Broadcasting Communications Device for Office Use)

This device obtained EMC registration for office use (Class A), and it is intended to be used in places other than homes.

Sellers and/or users need to take note of this.

## **Conformance to Shipbuilding Standards**

The NJ/NX-series Controllers comply with the following shipbuilding standards. Applicability to the shipbuilding standards is based on certain usage conditions. It may not be possible to use the product in some locations. Contact your OMRON representative before attempting to use a Controller on a ship.

## Usage Conditions for NK and LR Shipbuilding Standards

- The NJ/NX-series Controller must be installed within a control panel.
- Gaps in the door to the control panel must be completely filled or covered with gaskets or other material.
- The following noise filter must be connected to the power supply line.

#### **Noise Filter**

Manufacturer	Model
Cosel Co., Ltd.	TAH-06-683

## **Software Licenses and Copyrights**

This product incorporates certain third party software. The license and copyright information associated with this software is available at http://www.fa.omron.co.jp/nj\_info\_e/.

## **Versions**

Unit versions are used to manage the hardware and software in NJ/NX-series Units and EtherCAT slaves. The unit version is updated each time there is a change in hardware or software specifications. Even when two Units or EtherCAT slaves have the same model number, they will have functional or performance differences if they have different unit versions.

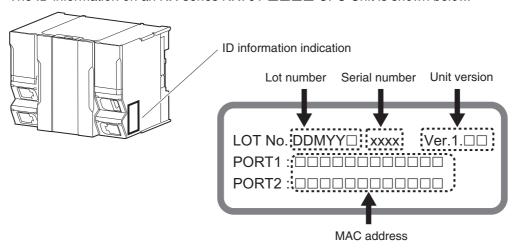
## **Checking Versions**

You can check versions in the ID information indications or with the Sysmac Studio.

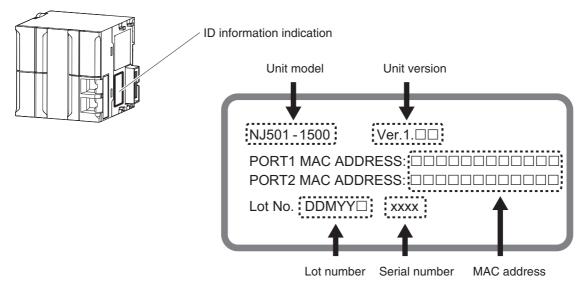
## **Checking Unit Versions on ID Information Indications**

The unit version is given on the ID information indication on the side of the product.

The ID information on an NX-series NX701-□□□□ CPU Unit is shown below.



The ID information on the NJ-series NJ501-1500 CPU Unit is shown below.



## **Checking Unit Versions with the Sysmac Studio**

You can use the Sysmac Studio to check unit versions. The procedure is different for Units and for EtherCAT slaves.

### Checking the Unit Version of an NX-series CPU Unit

You can use the Production Information while the Sysmac Studio is online to check the unit version of a Unit. You can check the unit version of only the CPU Unit.

**1** Right-click CPU Rack under Configurations and Setup – CPU/Expansion Racks in the Multiview Explorer and select *Production Information*.

The Production Information Dialog Box is displayed.

### Checking the Unit Version of an NJ-series CPU Unit

You can use the Production Information while the Sysmac Studio is online to check the unit version of a Unit. You can do this for the CPU Unit, CJ-series Special I/O Units, and CJ-series CPU Bus Units. You cannot check the unit versions of CJ-series Basic I/O Units with the Sysmac Studio.

Use the following procedure to check the unit version.

1 Double-click CPU/Expansion Racks under Configurations and Setup in the Multiview Explorer. Or, right-click CPU/Expansion Racks under Configurations and Setup and select *Edit* from the menu.

The Unit Editor is displayed.

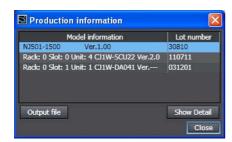
**2** Right-click any open space in the Unit Editor and select **Production Information**.

The Production Information Dialog Box is displayed.

#### Changing Information Displayed in Production Information Dialog Box

1 Click the **Show Detail** or **Show Outline** Button at the lower right of the Production Information Dialog Box.

The view will change between the production information details and outline.





**Outline View** 

**Detail View** 

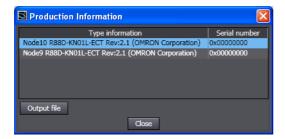
The information that is displayed is different for the Outline View and Detail View. The Detail View displays the unit version, hardware version, and software versions. The Outline View displays only the unit version.

### Checking the Unit Version of an EtherCAT Slave

You can use the Production Information while the Sysmac Studio is online to check the unit version of an EtherCAT slave. Use the following procedure to check the unit version.

- 1 Double-click EtherCAT under Configurations and Setup in the Multiview Explorer. Or, right-click EtherCAT under Configurations and Setup and select *Edit* from the menu.

  The EtherCAT Tab Page is displayed.
- 2 Right-click the master on the EtherCAT Tab Page and select *Display Production Information*.
  The Production Information Dialog Box is displayed.
  The unit version is displayed after "Rev."



## **Unit Versions of CPU Units and Sysmac Studio Versions**

The functions that are supported depend on the unit version of the NJ/NX-series CPU Unit. The version of Sysmac Studio that supports the functions that were added for an upgrade is also required to use those functions.

Refer to the *NJ/NX-series CPU Unit Software User's Manual* (Cat. No. W501) for the relationship between the unit versions of the CPU Units and the Sysmac Studio versions, and for the functions that are supported by each unit version.

## **Unit Versions of CPU Units and Peripheral Tool Versions**

When you set tag data links for the built-in EtherNet/IP port on an NJ/NX-series CPU Unit, use the versions of the Network Configurator and the Sysmac Studio that are given in the following table.

CPU Unit		Network Configurator for EtherNet/IP					Sysmac Studio		
Model	Version	Version 3.3x or lower	Version 3.40	Version 3.50 or 3.51	Version 3.53 to 3.58	Version 3.59 or higher	Version 1.09 or lower	Version 1.10 to 1.12	Version 1.13 or higher
NJ501-□□□□	Version 1.00 to 1.02	NA	ОК	OK	OK	OK	NA	ОК	ОК
NJ301-□□□	Version 1.01 or 1.02	NA	NA	OK	OK	OK	NA	ОК	ОК
NJ501-□□□□ NJ301-□□□□	Version 1.03 or later	NA	NA	NA	OK	OK	NA	ОК	ОК
NJ101-□□□□ NX701-□□□□	Version 1.10 or later	NA	NA	NA	NA	ОК	NA	NA	ОК

## **Related Manuals**

The followings are the manuals related to this manual. Use these manuals for reference.

Manual name	Cat. No.	Model numbers	Application	Description
NX-series CPU Unit Hardware User's Manual	W535	NX701-□□□□	Learning the basic specifications of the NX-series CPU Units, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NX-series system is provided along with the following information on the CPU Unit.  • Features and system configuration  • Introduction  • Part names and functions  • General specifications  • Installation and wiring  • Maintenance and inspection  Use this manual together with the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501).
NJ-series CPU Unit Hardware User's Manual	W500	NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning the basic specifications of the NJ-series CPU Units, including introductory information, designing, installation, and maintenance. Mainly hardware information is provided.	An introduction to the entire NJ-series system is provided along with the following information on the CPU Unit.  • Features and system configuration  • Introduction  • Part names and functions  • General specifications  • Installation and wiring  • Maintenance and inspection  Use this manual together with the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501).
NJ/NX-series CPU Unit Software User's Manual	W501	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning how to program and set up an NJ/NX-series CPU Unit. Mainly software information is provided.	The following information is provided on an NJ/NX-series CPU Unit.  CPU Unit operation  CPU Unit features  Initial settings  Programming based on IEC 61131-3 language specifications  Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) or <i>NX-series CPU Unit Hardware User's Manual</i> (Cat. No. W535).
NJ/NX-series Instructions Reference Manual	W502	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning detailed specifica- tions on the basic instruc- tions of an NJ/NX-series CPU Unit.	The instructions in the instruction set (IEC 61131-3 specifications) are described. When programming, use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) or <i>NX-series CPU Unit Hardware User's Manual</i> (Cat. No. W535) and with the <i>NJ/NX-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ/NX-series CPU Unit Motion Control User's Manual	W507	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning about motion control settings and programming concepts.	The settings and operation of the CPU Unit and programming concepts for motion control are described. When programming, use this manual together with the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500) or NX-series CPU Unit Hardware User's Manual (Cat. No. W535) and with the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501).

Manual name	Cat. No.	Model numbers	Application	Description
NJ/NX-series Motion Control Instructions Ref- erence Manual	W508	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning about the specifications of the motion control instructions that are provided by OMRON.	The motion control instructions are described. When programming, use this manual together with the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500) or NX-series CPU Unit Hardware User's Manual (Cat. No. W535), and with the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501) and NJ/NX-series CPU Unit Motion Control User's Manual (Cat. No. W507).
NJ/NX-series CPU Unit Built-in EtherCAT® Port User's Manual	W505	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Using the built-in EtherCAT port on an NJ/NX-series CPU Unit.	Information on the built-in EtherCAT port is provided. This manual provides an introduction and provides information on the configuration, features, and setup.  Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) or <i>NX-series CPU Unit Hardware User's Manual</i> (Cat. No. W535) and with the <i>NJ/NX-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ/NX-series CPU Unit Built-in EtherNet/IP <sup>TM</sup> Port User's Manual	W506	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Using the built-in Ether- Net/IP port on an NJ/NX- series CPU Unit.	Information on the built-in EtherNet/IP port is provided. Information is provided on the basic setup, tag data links, and other features.  Use this manual together with the <i>NJ-series CPU Unit Hardware User's Manual</i> (Cat. No. W500) or <i>NX-series CPU Unit Hardware User's Manual</i> (Cat. No. W535) and with the <i>NJ/NX-series CPU Unit Software User's Manual</i> (Cat. No. W501).
NJ-series Database Con- nection CPU Units User's Manual	W527	NJ501-1□20	Using the database connection service with NJ-series Controllers	Describes the database connection service.
NJ-series SECS/GEM CPU Unit User's Manual	W528	NJ501-1340	Using the GEM Services with NJ-series Controllers	Information is provided on the GEM Services.
NJ/NX-series Trouble- shooting Manual	W503	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	Learning about the errors that may be detected in an NJ/NX-series Controller.	Concepts on managing errors that may be detected in an NJ/NX-series Controller and information on individual errors are described.  Use this manual together with the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500) or NX-series CPU Unit Hardware User's Manual (Cat. No. W535) and with the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501).
Sysmac Studio Version 1 Operation Manual	W504	SYSMAC- SE2□□□	Learning about the operat- ing procedures and func- tions of the Sysmac Studio.	Describes the operating procedures of the Sysmac Studio.

## **Revision History**

A manual revision code appears as a suffix to the catalog number on the front and back covers of the manual.



Revision code	Date	Revised content
01	July 2011	Original production
02	March 2012	Added information on the NJ301-1□00.
		<ul> <li>Made changes accompanying the upgrade to unit version 1.01 of the CPU Unit.</li> </ul>
		Corrected mistakes.
		Added A-8 Accessing Variables with CIP Message Communications.
03	May 2012	Added information on functional support for unit version 1.02 and later of the CPU Units.
		Corrected mistakes.
04	August 2012	Added information on functional support for unit version     1.03 and later of the CPU Units.
		Corrected mistakes.
05	February 2013	Added information on functional support for unit version     1.04 and later of the CPU Units.
		Corrected mistakes.
06	April 2013	Corrected mistakes.
07	June 2013	Added information on functional support for unit version 1.06 and later of the CPU Units.
08	December 2013	Added information on functional support for unit version 1.08 and later of the CPU Units.
		Corrected mistakes.
09	July 2014	Added information on functional support for unit version     1.09 and later of the CPU Units.
		Corrected mistakes.
10	January 2015	Added information on functional support for unit version     1.10 and later of the CPU Units.
		Corrected mistakes.
11	April 2015	Added information on the NX701-□□□□ NX-series CPU Units.
		Added information on the NJ101-□□□□ NJ-series CPU Units.
		Corrected mistakes.

**Revision History** 

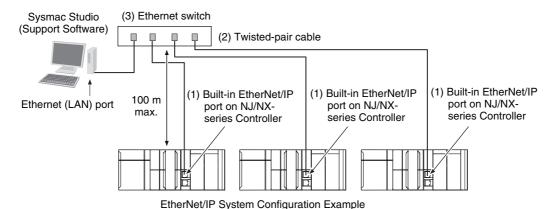
## Introduction

1-1	Introdu	iction	1-2
	1-1-1	EtherNet/IP Features	1-2
	1-1-2	Features of Built-in EtherNet/IP Port on NJ/NX-series CPU Units	1-3
1-2	System	n Configuration and Configuration Devices	1-5
	1-2-1	Devices Required to Construct a Network	1-5
	1-2-2	Support Software Required to Construct a Network	1-6
1-3	Built-in	n EtherNet/IP Port	1-7
	1-3-1	Specifications	1-7
	1-3-2	Part Names and Functions	1-10
1-4	Introdu	iction to Communications Services	1-15
	1-4-1	CIP (Common Industrial Protocol) Communications Services	1-15
	1-4-2	IP Routing	1-18
	1-4-3	BOOTP Client	1-19
	1-4-4	FTP Server	1-19
	1-4-5	FTP Client	1-20
	1-4-6	Automatic Clock Adjustment	1-20
	1-4-7	Socket Service	1-21
	1-4-8	Specifying Host Names	1-21
	1-4-9	SNMP Agent	1-22
1-5	EtherN	et/IP Communications Procedures	1-23
	1-5-1	Outline	1-23

#### Introduction 1-1

#### 1-1-1 EtherNet/IP Features

EtherNet/IP is an industrial multi-vendor network that uses Ethernet. The EtherNet/IP specifications are open standards managed by the ODVA (Open DeviceNet Vendor Association), just like DeviceNet. EtherNet/IP is not just a network between Controllers. It is also used as a field network. Because Ether-Net/IP uses standard Ethernet technology, various general-purpose Ethernet devices can be used in the network.



### High-speed, High-capacity Data Exchange through Tag Data Links

The EtherNet/IP protocol supports implicit communications, which allows cyclic communications (called tag data links in this manual) with EtherNet/IP devices. Using the NJ/NX-series CPU Unit, up to 184,832 words of data (total of 369,664 words with two ports) can be quickly exchanged between controllers and devices. Using the NJ-series CPU Unit, up to 9,600 words of data can be exchanged.

#### Tag Data Link (Cyclic Communications) Cycle Time

Tag data links (cyclic communications) operate at the cyclic period specified for each application, regardless of the number of nodes. Data is exchanged over the network at the refresh cycle set for each connection, so the communications refresh cycle will not increase even if the number of nodes is increased, i.e., the concurrency of the connection's data is maintained. Because the refresh cycle can be set for each connection, each application can communicate at its ideal refresh cycle. For example, interprocess interlocks can be transferred at high speed, while the production commands and the status monitor information are transferred at low speed.

## 1-1-2 Features of Built-in EtherNet/IP Port on NJ/NX-series CPU Units

#### Tag Data Links

Cyclic communications between Controllers or between Controllers and other devices are possible on an EtherNet/IP network. Tag data links can quickly perform data exchanges of up to 184,832 words of data using the NX-series CPU Unit (total of 369,664 words with two ports), and up to 9,600 words of data using the NJ-series CPU Unit.

### Message Communications

You can send CIP commands to devices on the EtherNet/IP network when required by execution of CIP communications instructions in a program. As a result, it is possible to send and receive data with devices on the EtherNet/IP network.

#### BOOTP Client

If the NJ/NX-series built-in EtherNet/IP port is set in the BOOTP settings, the BOOTP client operates when the Controller power is turned ON, and the IP address is obtained from the BOOTP server. It is possible to set all of the IP addresses of multiple built-in EtherNet/IP ports at the same time.

#### Built-in FTP Server for File Transfers to and from Host Computers

An FTP server is built into the Controller. You can use it to read and write data within the Controller as files from workstations and computers with FTP clients. The FTP server enables the transfer of large amounts of data from a client without any additional ladder programming.

#### • FTP Client for File Transfers to and from Host Computers

An FTP client is built into the Controller so that you can read and write files on workstations and computers that have an FTP server from the Controller.

You can use the FTP client communications instructions to transfer one or more files between the Controller and an FTP server.

#### Automatic Controller Clock Adjustment

The clocks built into Controllers connected to Ethernet can be automatically adjusted to the time of the clock in the NTP server. If all of the clocks in the system are automatically adjusted to the same time, time stamps can be used to analyze production histories.

\* A separate NTP server is necessary to automatically adjust the Controller clocks.

#### Socket Services

Socket services can be used to send/receive data between general-purpose applications and Controllers. You can use these communications services to send and receive any data to and from remote nodes, i.e., between host computers and Controllers or between Controllers. You can execute socket communications instructions in order in a program to execute communications processes with the socket services. There are two socket services, the UDP socket service and TCP socket service.

#### Host Names

You can directly specify IP addresses, but you can also use the host names instead of the IP addresses for NTP servers, SNMP managers, or the destinations of socket instructions and CIP communications instructions (DNS client or hosts settings). This is useful, for example, when server IP addresses change for system revisions because the IP addresses are automatically found when host names are used.

- \* A separate DNS server is necessary to use host names with the DNS client.
- \* The DNS server is specified directly using its IP address.

#### Network Management with an SNMP Manager

The SNMP agent passes internal status information from the built-in EtherNet/IP port to network management software that uses an SNMP manager.

\* A separate SNMP manager is necessary for network management.

### Complete Troubleshooting Functions

A variety of functions are provided to quickly identify and handle errors.

- Self-diagnosis at startup
- · Event log that records the time of occurrence and other error details



#### **Additional Information**

#### **CIP (Common Industrial Protocol)**

CIP is a shared industrial protocol for the OSI application layer. The CIP is used in networks such as EtherNet/IP, CompoNet, and DeviceNet. Data can be routed easily between networks that are based on the CIP. You can therefore easily configure a transparent network from the field device level to the host level. The CIP has the following advantages.

- Destination nodes are specified by a relative path, without fixed routing tables.
- The CIP uses the producer/consumer model. Nodes in the network are arranged on the same level and it is possible to communicate with required devices whenever it is necessary. The consumer node will receive data sent from a producer node when the connection ID in the packet indicates that the node requires the data. Because the producer can send the same data with the same characteristics in a multicast format, the time required for the transfer is fixed and not dependent on the number of consumer nodes. (Either multicast or unicast can be selected.)

## Two EtherNet/IP Communications Ports as a Standard Feature (only with the **NX-series CPU Unit)**

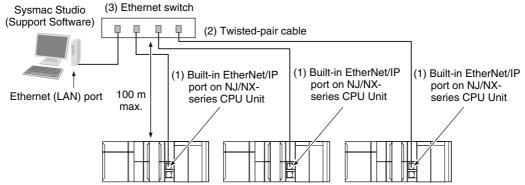
The CPU Unit is equipped with two EtherNet/IP ports for EtherNet/IP communications as standard. This feature allows you to separate the information network from the control network. In addition, the built-in EtherNet/IP ports have an IP routing function that sends IP packets to devices on other IP network segments.

\* To use the function, you must set the IP router table and default gateway settings for each device on the network appropriately for your network configuration. For details on the settings, refer to 5-1 TCP/IP Settings Display.

# 1-2 System Configuration and Configuration Devices

## 1-2-1 Devices Required to Construct a Network

The basic configuration for an EtherNet/IP system includes one Ethernet switch to which nodes are attached in star configuration using twisted-pair cable.



The following products are also required to build a network. Obtain them in advance.

	Network device	Function		
(1)	Per Node  NJ501-□□□□, 301-□□□□ or 101-□□□  NJ-series CPU Unit (built-in EtherNet/IP port)  NX701-□□□ NX-series CPU Unit (built-in EtherNet/IP port)  OMRON PLCs  CJ2H-CPU□□-EIP or CJ2M-CPU3□ CJ2 CPU  Unit (built-in EtherNet/IP port)  CJ-series CJ1W-EIP21 EtherNet/IP Unit*	These Units are used to connect to an EtherNet/IP network.		
(2)	CS-series CS1W-EIP21 EtherNet/IP Unit Twisted-pair cable	The twisted-pair cable has a RJ45 Modular Connector at each end. This cable is used to connect the built-in EtherNet/IP port or EtherNet/IP Unit to an Ethernet switch. Use an STP (shielded twisted-pair) cable of category 5, 5e, or higher.		
(3)	Ethernet switch	This is a relay device that connects multiple nodes in a star LAN. For details on recommended devices to configure a network, refer to 2-1-1 Recommended Network Devices.		

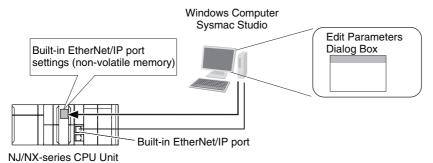
<sup>\*</sup> The CJ1W-EIP21 can be mounted only to an NJ-series CPU Unit with unit version 1.01 or later and Sysmac Studio version 1.02 or higher.

## 1-2-2 Support Software Required to Construct a Network

This section describes the Support Software that is required to construct an EtherNet/IP network. The built-in EtherNet/IP port has Ethernet Settings and Tag Data Link Settings, which are stored in the non-volatile memory of the CPU Unit. Support Software is provided for each, as described below.

### Built-in Ethernet/IP Settings: Sysmac Studio

Use the Sysmac Studio to set the basic settings, such as the local IP address and subnet mask of the built-in EtherNet/IP port. The Sysmac Studio can also be used to check if data I/O is being performed correctly for tag data links.



Refer to the Sysmac Studio Version 1 Operation Manual (Cat. No. W504) for details on the Sysmac Studio.

### Tag Data Link Settings: Network Configurator

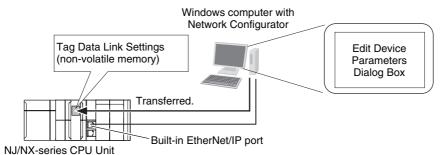
Use the Network Configurator to set the tag data links for the built-in EtherNet/IP port. (The Network Configurator is included in the Sysmac Studio Standard Edition.) The main functions of the Network Configurator are given below.

### 1) Setting and Monitoring Tag Data Links (Connections)

The network device configuration and tag data links (connections) can be created and edited. After connecting to the network, the device configuration and tag data link settings can be uploaded and monitored.

#### 2) Multi-vendor Device Connections

EDS files can be installed and deleted so that you can construct, set, and manage networks that contain EtherNet/IP devices from other companies. The IP addresses of EtherNet/IP devices can also be changed.



For details on the Network Configurator, refer to Section 7 Tag Data Link Functions.



#### **Additional Information**

You can also use the Sysmac Studio to set the tag data links. Refer to A-2 Use the Sysmac Studio to Set the Tag Data Links (EtherNet/IP Connections) for details on setting the tag data links in the Sysmac Studio.

## 1-3 Built-in EtherNet/IP Port

## 1-3-1 Specifications

		Specifications			
14.	Item			/NJ301-□□□□/	
10		Unit version 1.10 or later	Unit version 1.00 to 1.02	Unit version 1.03 or later	
Communications pro	otocol	TCP/IP or UDP/IP			
Supported services		Sysmac Studio connection, tag data link, CIP message communications, socket services, FTP server, FTP client, automatic clock adjustment (NTP client), SNMP agent, DNS client, and BOOTP client			
Number of ports		2 (With IP routing function)	1		
Physical layer		100Base-TX, 10Base-T or 1000Base-T (1000Base-T or 100Base-TX is rec- ommended.) *1	100Base-TX or 10Base-T (100Base-TX is recommended.) *1		
	Media access method	CSMA/CD			
	Modulation	Baseband			
	Transmission paths	Star form			
Transmission specifications	Baud rate	1,000 Mbps (1000Base-T)	100 Mbps (100Base-TX)		
opcomouncino	Transmission media	Shielded twisted-pair (STP) cable, Category 5, 5e, or higher			
	Transmission distance	100 m max. (distance between hub and node)			
	Number of cascade connections	There is no limitation when an Ethernet switch is used.			
	Number of connections	256 per port (total of 512 with two ports)	32		
	Packet interval	0.5 to 10,000 ms in 0.5-ms increments	10 to 10,000 ms in 1-ms increments	1 to 10,000 ms in 1- ms increments	
OID and in a Tan	(refresh cycle)	Packet intervals can be set independently for each cor (Data is refreshed over the network at the preset interv not depend on the number of nodes.)			
CIP service: Tag data links (cyclic	Allowed communi-	40,000 pps *2, *3	1,000 pps *2	3,000 pps *3	
communications)	cations bandwidth per Unit	Note The heartbeat is	s included.		
	Number of registrable tags	256 per port (total of 512 with two ports)	256		
Tag types		Network variables CIO, Work, Holding, DM, and EM Areas cannot be used.	Network variables CIO, Work, Holding, DM, or EM Areas can be used.		

		Specifications			
Ite	em	NX701-□□□		/NJ301-□□□□/ □□□□	
		Unit version 1.10 or later	Unit version 1.00 to 1.02	Unit version 1.03 or later	
	Number of tags per connection (= 1 tag set)	8 (7 tags when the tag	8 (7 tags when the tag set contains the Controller status)		
	Maximum link data size per node	739,328 bytes	19,200 bytes		
		1,444 bytes *4	600 bytes *4		
	Maximum data size per connection		ncy is maintained with Concurrency of Tag Da concurrency.		
CIP service: Tag	Number of registrable tag sets	256 per port (total of 512 with two ports) (1 connection = 1 tag set)	32 (1 connection = 1 t	tag set)	
communications)	Maximum size of 1 tag set	722 words (The Controller status uses 1 word when the tag set contains the Controller sta- tus.)	300 words (The Controller status uses 1 word when the tag set contains the Control ler status.)		
	Changing tag data link parameters when Controller is in RUN mode	Supported *5			
	Multi-cast packet filter *6	Supported			
	Class 3 (con- nected)	Connections: 128 per port (total of 256 with two ports) (cli- ents plus server)	Number of connection ers)	ns: 32 (clients + serv-	
OID	UCMM (uncon-	Number of clients that can communicate at one time:		one time: 32 max.	
CIP message service: Explicit mes-	nected)	Number of servers that can communicate at one time: 32 max.			
sages * <sup>7</sup>	CIP routing *8	Supported.  CIP routing is supported for the following remote Units: NXT  DDDD, NJ501-DDDD, NJ301-DDDD, NJ101-DDDD,  EIP21, CJ1W-EIP21, CJ2H-CPUDD-EIP, and CJ2M-CPU3		J101-□□□□, CS1W-	
		Using a combination of extended up to a max	of any Units above, communication can be kimum of 8 levels.		
SNMP	Agents	SNMPv1 or SNMPv2c			
	MIB	MIB-II	T		
EtherNet/IP conform	ance test	Conforms to CT11	Conforms to CT8		
Ethernet interface		10Base-T, 100Base-TX, or 1000Base-T Auto negotiation or	10Base-T or 100Base Auto negotiation or fix		

<sup>\*1</sup> If tag data links are being used, use 100Base-TX or 1000Base-T.

<sup>\*2</sup> Here, pps means "packets per second" and indicates the number of packets that can be processed in one second.

<sup>\*3</sup> If the two built-in EtherNet/IP ports are used simultaneously, the maximum communications data size for two ports in total will be reached.

- \*4 To use a data size of 505 bytes or higher, the system must support a large forward open (an optional CIP specification). The CS, CJ, NJ, and NX-series Units support a large forward open, but before connecting to nodes of other companies, confirm that those devices also support it.
- \*5 If the parameters of the built-in EtherNet/IP port are changed, the port is restarted. When other nodes are in communications with the affected node, the communications will temporarily time out and automatically recover after the restart.
- \*6 Because the built-in EtherNet/IP port is equipped with an IGMP client (version 2), unnecessary multicast packets can be filtered by an Ethernet switch that supports IGMP snooping.
- \*7 The built-in EtherNet/IP port uses the TCP/UDP port numbers shown in the following table. Do not set the same port number for more than one TCP/UDP service.

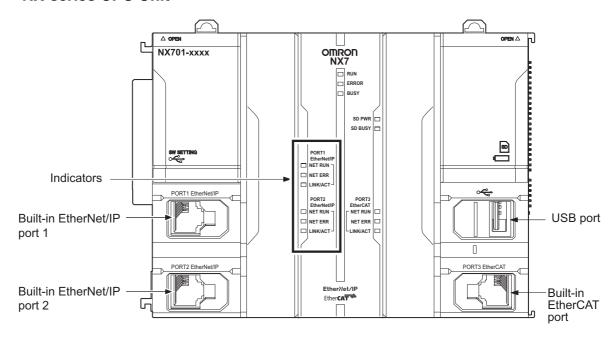
Service	Protocol	Port number	Remarks
EIP data links	UDP	2222	Fixed values
Used by system	UDP	2223, 2224	
	TCP	9610	
CIP messages	TCP	44818	
FTP client data transfer port	TCP	20	
DNS client	TCP/UDP	53	
BOOTP client	UDP	68	
HTTP server	TCP	80	
Used by system, other	TCP/UDP	9600	You can change the port number in
FTP client control port	TCP	21	the Unit Settings on the Sysmac Studio.
NTP client	UDP	123	Studio.
SNMP agent	UDP	161	
SNMP trap	UDP	162	

<sup>\*8</sup> A CPU Unit with unit version 1.01 or later and Sysmac Studio version 1.02 or higher are required to use CPU routing.

#### **Part Names and Functions** 1-3-2

## **Parts and Names**

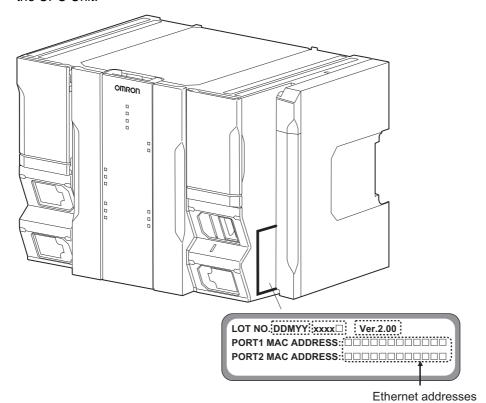
## • NX-series CPU Unit



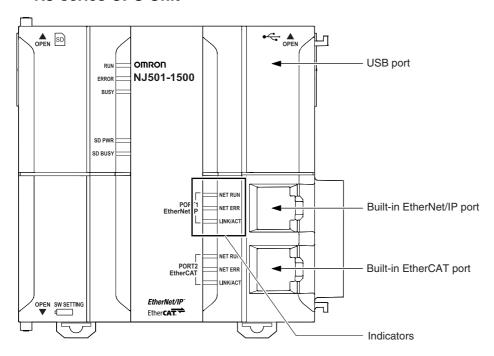
## **Ethernet Address Notation**

A specific Ethernet address is allocated to each device connected to the Ethernet network.

The built-in EtherNet/IP port's address is listed in 12-digit hexadecimal in the place shown below on the CPU Unit.

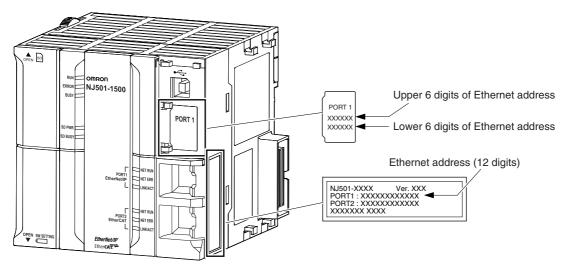


## • NJ-series CPU Unit

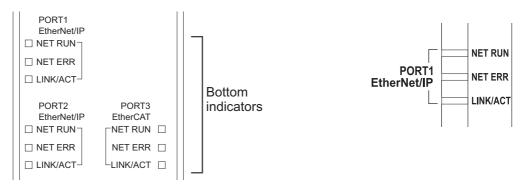


## **Ethernet Address Notation**

A specific Ethernet address is allocated to each device connected to the Ethernet network. The built-in EtherNet/IP port's address is listed in 12-digit hexadecimal in the two places shown below on the CPU Unit.



## Indicators (LEDs)



**NX-series CPU Unit** 

NJ-series CPU Unit

## NET RUN, NET ERR, and LINK/ACT

- NET RUN indicator: This shows the status of the CIP connection (tag data links, Class 3 messages).
- NET ERR indicator: This shows the network communications error status. Refer to *Section 15 Troubleshooting* for details.
- LINK/ACT indicator: This shows the Ethernet communications status.

Indicator	Color	Status	Operating status
mulcator	Color	Status	Ethernet communications are not possible.
NET RUN	Not lit		<ul> <li>The power supply is OFF or the Controller is being reset.</li> <li>A MAC address error or communications Controller error occurred.</li> <li>The same IP address is assigned to more than one node.</li> </ul>
	Green	Flashing	Ethernet communications are in progress.     Tag data link connection establishment in progress (originator operation)     IP address acquisition with BOOTP in progress.
	Green	Lit	Normal If only the target is set for the tag data link, this indicator lights regardless of whether the connection from the originator is established. It remains lit even if the data links are stopped.
	Not lit  Flashing  Red		There are no Ethernet communications errors.  • The power supply is OFF or was reset.
NET ERR			A user-recoverable error occurred.     An error occurred in TCP/IP communications or CIP communications.     FTP Server Setting Error, NTP Server Setting Error, etc.     Tag Data Link Setting Error, Tag Data Link Verification Error, etc.     The same IP address is assigned to more than one node.
		Lit	A user-non-recoverable error occurred.     A MAC address error or communications Controller error occurred.
LINK/ACT		Not lit	Link not established.  The cable is not connected.  The power supply is OFF or was reset.
	Yellow	Flashing	Data communications in progress after establishing link.
	IGHOW	Lit	Links established.



## **Additional Information**

When the built-in EtherNet/IP port is set to disable, all the indicators will not light. Refer to 5-1 TCP/IP Settings Display for the information on setting the built-in EtherNet/IP port.

# 1-4 Introduction to Communications Services

## 1-4-1 CIP (Common Industrial Protocol) Communications Services

## **Tag Data Links (Cyclic Communications)**

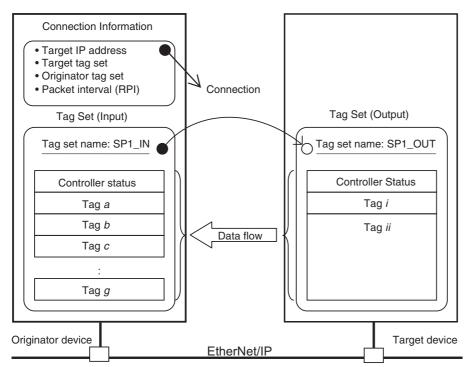
A program is not required to perform cyclic data exchanges with other devices on the EtherNet/IP network. Normally, a connection is started with the target device for each tag set that was created with the Network Configurator to start communications for tag data links for a built-in EtherNet/IP port. One connection is used per tag set.

The maximum number of connections that can be registered is shown below.

- NX-series CPU Unit: 256 connections (total of 512 connections with two ports)
- NJ-series CPU Unit: 32 connections.

The following table gives the built-in EtherNet/IP port tag and tag set specifications.

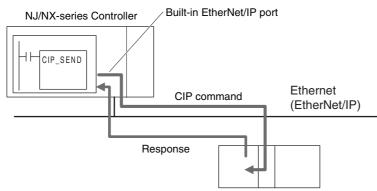
	Specification	NX-series CPU Unit	NJ-series CPU Unit
Tag	Total size of all tags	184,832 words max. (total of 369,664 words with two ports)	9,600 words max.
	Maximum size of 1 tag	722 words max. (The maximum size is 721 words when the tag set includes the Controller status)	300 words max. (The maximum size is 299 words when the tag set includes the Controller status)
	Number of registrable tags	256 per port (total of 512 connections with two ports)	32
Tag set	Maximum tag set size	722 words max. (The maximum size is 721 words when the tag set includes the Controller status)	300 words max. (The maximum size is 299 words when the tag set includes the Controller status)
	Number of tags per tag set	8 (7 tags per tag set when the tag set includes the Controller status)	8 (7 tags per tag set when the tag set includes the Controller status)
		Note Input and output variables cannot be combined.	Note Input and output variables cannot be combined.
	Number of registrable tag sets	256 per port (total of 512 connections with two ports)	32



<sup>\*</sup> In this example, a connection is established with the originator's tag list with tags a to g (inputs), which are in a tag set called SP1\_IN, and the target's tag list with tags i and ii (outputs), which are in a tag set called SP1\_OUT.

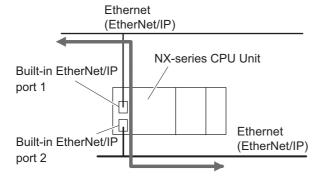
## **CIP Message Communications**

User-specified CIP commands can be sent to devices on the EtherNet/IP network. CIP commands, such as those for reading and writing data, can be sent and their responses received by executing the CIP communications instructions from the user program in the NJ/NX-series CPU Unit.

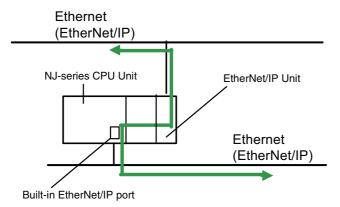


By specifying a route path, you can send CIP messages (CIP commands and responses) to a device on another CIP-based network segment via a built-in EtherNet/IP port or the EtherNet/IP Unit (CIP routing function for message communications). The maximum number of levels of CIP routing via the ports is eight for any combination of CS, CJ, NJ, and NX-series CPU Units. Note that the number of levels of IP routing using an L3 Ethernet switch is not counted in the number of levels of CIP routing via the ports.

• NX-series CPU Unit Because there are two built-in EtherNet/IP ports, CIP routing is possible by the CPU Unit alone.



NJ-series CPU Unit
 By combining the built-in EtherNet/IP port and an EtherNet/IP Unit, CIP routing can be performed.





#### **Additional Information**

In CIP routing, a node (Unit) that routes information subtracts the equivalent of one hop from the timeout, deletes its own address from the route information, and relays the information to the next node (Unit).

When a timeout is specified, the timeout for the actual request service processing is set in the last hop. In the case of relay hops, the timeout for the relay route must be added to the timeout for the request.

OMRON products that support CIP subtract 5 seconds per hop.



### **Version Information**

For NJ-series CPU Unit, you can use the EtherNet/IP Unit with the CPU unit version 1.01 or later and the Sysmac Studio version 1.02 or higher.

#### 1-4-2 IP Routing

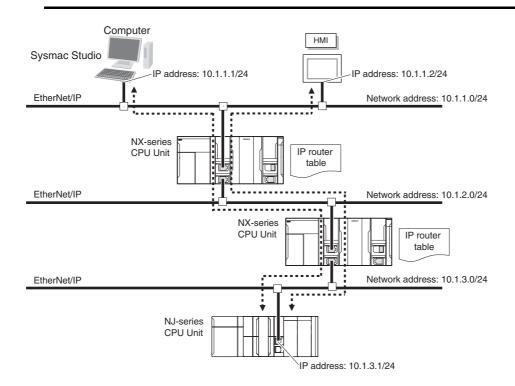
The two built-in EtherNet/IP ports on the NX-series CPU Unit both have the IP routing function. The IP routing function sends IP packets to other network segments based on the routing information set in the IP router table.

To communicate with devices on other network segments, you must set the IP router table and default gateway settings for the NX-series CPU Unit and each device on the network appropriately for your network configuration.



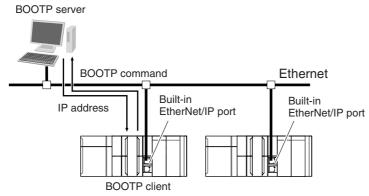
### **Precautions for Correct Use**

- You cannot create tag data links between multiple CPU Units using IP routing on the NX-series CPU Unit.
- The IP routing function can only be used with NX-series CPU Units. IP routing cannot be used with a combination of a built-in EtherNet/IP port on an NJ-series CPU Unit and an EtherNet/IP Unit.



## 1-4-3 BOOTP Client

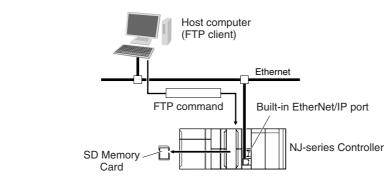
You set the built-in EtherNet/IP port in the BOOTP settings to use the BOOTP client to obtain settings, such as the built-in EtherNet/IP port IP address.

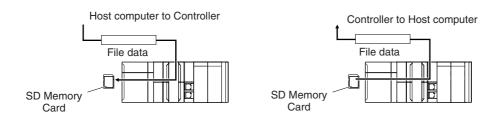


The built-in EtherNet/IP port IP address is obtained from the BOOTP server when the power is turned ON.

## 1-4-4 FTP Server

An FTP server is built into the built-in EtherNet/IP port so that files can be read from and written to the SD Memory Card in the CPU Unit of the Controller from computers at other Ethernet nodes. This makes it possible to exchange data files between a host computer and the Controller with the host computer as the FTP client and the Controller as the FTP server.

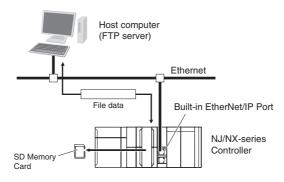


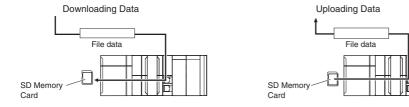


## 1-4-5 FTP Client

The built-in EtherNet/IP port contains an FTP client. With it, you can use FTP client communications instructions to transfer files between the CPU Unit and host computers on Ethernet.

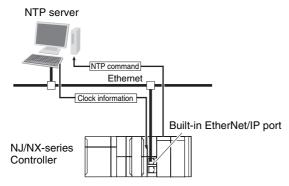
This makes it possible to exchange data files between a host computer and the Controller with the Controller as the FTP client and the host computer as the FTP server.





## 1-4-6 Automatic Clock Adjustment

With the built-in EtherNet/IP port, clock information is read from the NTP server at the specified time or at a specified interval after the power supply to the CPU Unit is turned ON. The internal clock time in the CPU Unit is updated with the read time.



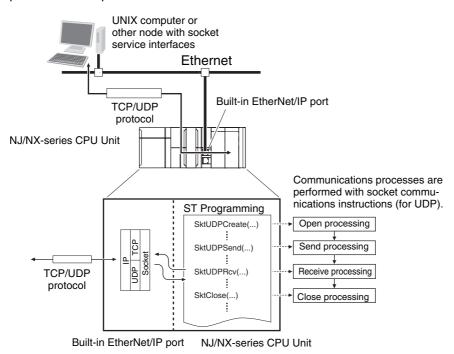


## **Precautions for Correct Use**

An NTP server is required to use automatic clock adjustment.

## 1-4-7 Socket Service

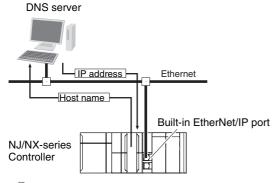
You can send data to and receive data from any node on Ethernet with the UDP or TCP protocol. To send/receive data with a socket service, you execute multiple socket communications instructions in sequence in an ST program to execute the required communications processes. After a connection with the other communications device is opened with an open instruction, the values of the variables that are specified for the send instruction are sent and the data that was received for a receive instruction is stored in the specified variables. The connection is closed with a close instruction, and communications end. For TCP, you can also read the socket status and received data. You can use a total of 16 TCP ports and UDP ports.



## 1-4-8 Specifying Host Names

You can directly specify IP addresses, but you can also use the host names instead of the IP addresses for NTP servers, SNMP managers, or the destinations of socket instructions and CIP communications instructions (DNS client or hosts settings).

Example: Setting Host Names on the DNS Server



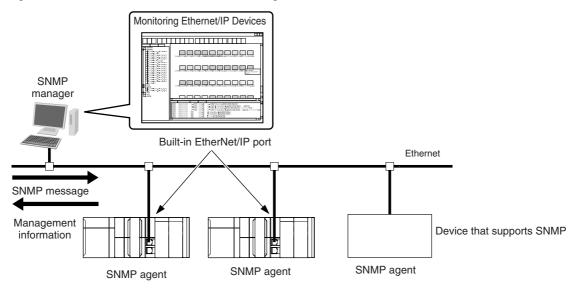
## 凼

### **Precautions for Correct Use**

A DNS server is required to use the server host names for the DNS client.

#### 1-4-9 **SNMP Agent**

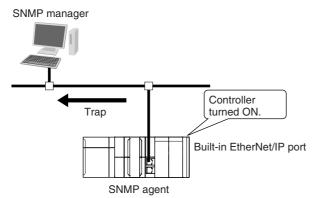
The SNMP agent passes internal status information from the built-in EtherNet/IP port to network management software that uses an SNMP manager.



## **SNMP Trap**

When specific conditions occur, the built-in EtherNet/IP port that is set as the SNMP agent sends status notification reports to the SNMP manager. The SNMP manager can learn about changes in status even without periodically monitoring of the built-in EtherNet/IP port. Status notification reports are sent under the following conditions.

- · When the Controller is turned ON
- · When links are established
- When an SNMP agent fails to be authorized



# 1-5 EtherNet/IP Communications Procedures

## 1-5-1 Outline

## Basic Operation

**1** Wire the Ethernet network with twisted-pair cable.

Section 2 Installing Ethernet Networks

1

2 Set the built-in EtherNet/IP port IP address with the Sysmac Studio.

Section 4 Determining IP Addresses

- 1. Use the Sysmac Studio to create a new project.
- 2. Set the local IP address in one of the following ways:
- · Defaults:

NX	-series CPU Unit	
	Built-in EtherNet/IP port 1	: 192.168.250.1
		(subnet mask = 255.255.255.0)
	Built-in EtherNet/IP port 2	: 192.168.251.1
		(subnet mask = 255.255.255.0)
NJ	series CPU Unit	: 192.168.250.1
		(subnet mask = 255.255.255.0)

- · Set any IP address.
- Obtain from BOOTP server.

Perform a communications test with a PING command from a computer.

Section 6 Testing Communications

**4** Use the Sysmac Studio to set the initial settings of the EtherNet/IP Function Module.

Section 5 Sysmac Studio Settings for the Built-in EtherNet/IP Port

Set the TCP/IP settings and Ethernet settings as required.

## Using Tag Data Links

1 Import the variable settings for the tags that were created on the Sysmac Studio to the Network Configurator.

7-2-4 Creating Tags and Tag Sets

2 Use the Network Configurator to create the tag data link table.

Section 7 Tag Data Link Functions

- · Create the network configuration.
- · Set the tags, tag sets, and connections.

 $\downarrow$ 

3 Connect the Network Configurator online.

J

**4** Download the tag data link setting.

 $\downarrow$ 

Start the tag data links (the links starts automatically when power is turned ON).

Ţ

**6** Check operation.

1-3-2 Part Names and Func-

• Check the built-in EtherNet/IP port indicators.

Section 15 Troubleshooting

- Use the Sysmac Studio to check the communications status with the All Tag Data Link Communications Status system-defined variable.
- Use the monitor function of the Network Configurator to confirm that the tag data links are in normal operation.

## Using the Message Communications Service

CIP Communications Instructions

**1** Execute CIP communications instructions in the user program.

Section 8 Message Communications

**2** Check operation.

1-3-2 Part Names and Func-

Use the Sysmac Studio to check the communications status with the end codes
of the instruction (Done, Error, and ErrorID).

Section 15 Troubleshooting

## Using the Socket Services

**1** Execute the socket service instructions in the user program.

Section 9 Socket Service

**2** Check operation.

Use the Sysmac Studio to check the communications status with the end codes
of the instruction (Done, Error, and ErrorID).

## Using the FTP Server

1 Use the Sysmac Studio to set the initial settings of the EtherNet/IP Function Module.

Section 10 FTP Server

Set the FTP settings (enabling FTP, login name, and password).

Ţ

**2** Connect to the FTP server in the NJ-series CPU Unit from an FTP client application.

- Input the FTP login name and password to log onto the built-in EtherNet/IP port.
- · Check the event log to see if the FTP server started.

## Using the Automatic Clock Adjustment

1 Use the Sysmac Studio to set the initial settings of the EtherNet/IP Function Module.

Section 12 Automatic Clock Adjustment

• Set the NTP settings (enabling NTP and execution conditions).

Execute automatic clock adjustment.

- Execute automatic adjustment at specified times or specified intervals.
- Use the Sysmac Studio to check the NTP Last Operation Time and NTP Operation Result system-defined variables.
- Check the event log to see if the NTP client started.

## Using the SNMP Agent

1 Use the Sysmac Studio to set the initial settings of the EtherNet/IP Function Module.

Section 13 SNMP Agent

- Set the SNMP settings.
- Set the SNMP trap settings.



**2** Check operation.

• Check the event log to see if the SNMP agent started.

## Using BOOTP

Use the Sysmac Studio to set the initial settings of the EtherNet/IP Function Module.

Section 5 Sysmac Studio Settings for the Built-in EtherNet/IP Port

• Set the BOOTP settings.



- **2** Check operation.
- Check the event log to see if BOOTP started.
- Check the Online system-defined variable.



## **Installing Ethernet Networks**

2-1	Selecti	ng the Network Devices	2-2
	2-1-1	Recommended Network Devices	. 2-2
	2-1-2	Ethernet Switch Types	. 2-3
	2-1-3	Ethernet Switch Functions	. 2-3
	2-1-4	Precautions for Ethernet Switch Selection	. 2-4
2-2	Networ	rk Installation	2-6
	2-2-1	Basic Installation Precautions	. 2-6
	2-2-2	Recommended Network Devices	. 2-6
	2-2-3	Precautions When Laying Twisted-pair Cable	. 2-6
	2-2-4	Precautions When Installing and Connecting Ethernet Switches	2-10
2-3	Conne	cting to the Network	2-11
	2-3-1	Ethernet Connectors	2-11
	2-3-2	Connecting the Cable	2-12

## **Selecting the Network Devices** 2-1

#### 2-1-1 **Recommended Network Devices**

The following table shows the devices recommended for use with the EtherNet/IP.

### Ethernet Switches

Manufacturer	Model	Description	
OMRON	W4S1-03B	Packet priority control (QoS): EtherNet/IP control data priority	
	W4S1-05B	Failure detection: Broadcast storm, LSI error detection	
	W4S1-05C	• 10/100Base-TX	
		Auto negotiation	
		Number of ports: three for the W4S1-03B, or five each for the W4S1-05B and W4S1-05C	
		Failure detection output (W4S1-05C only)	
Cisco Systems, Inc.	Consult the manufacturer.		
	http://www.cisco.com/		
CONTEC USA Inc.	Consult the manufacturer.		
	http://www.contec.com/		
Phoenix Contact USA	Consult the manufacturer.		
	https://www.phoenixcor	ntact.com	

## Twisted-pair Cables and Connectors

Applicable EtherNet/IP communications cables and connectors vary depending on the used baud rate. For 100Base-TX and 10Base-T, use an STP (shielded twisted-pair) cable of category 5 or higher. You can use either a straight or cross cable. For 1000Base-T, use an STP (shielded twistedpair) cable (double shielding with aluminium tape and braiding) of category 5e or higher. You can use either a straight or cross cable.

"100Base-TX" in the "Product" column of the table below indicates that either 100Base-TX or 10Base-T can be used.

	Product			Model
For 1000Base-T and	Sizes and conduc-	Sizes and conduc- Cables		NETSTAR-C5E
100Base-TX	tor pairs: AWG 24 ×		Cable, Ltd.	SAB 0.5 × 4P
	4 pairs		Kuramo Electric Co., Ltd.	KETH-SB
			SWCC Showa Cable Systems Co. Ltd.	FAE-5004
		RJ45 Connectors	Panduit Corporation	MPS588
For 100Base-TX Sizes and conduc-		Cables	Kuramo Electric Co., Ltd.	KETH-PSB-OMR
	tor pairs: AWG 22 × 2 pairs *1		Nihon Electric Wire & Cable Co., Ltd.	PNET/B
		RJ45 Assembly Connectors	OMRON	XS6G-T421-1
Sizes and conductor pairs: 0.5 mm × 4	Cables	Fujikura Ltd.	F-LINK-E 0.5mm × 4P	
	pairs	RJ45 Connectors	Panduit Corporation	MPS588

<sup>\*1</sup> We recommend that you use cables and connectors in above combinations.

## 2-1-2 Ethernet Switch Types

## Unmanaged Layer 2 (L2) Ethernet Switches

These Ethernet switches use the Ethernet MAC address to switch ports. Ordinary Ethernet switches have this function. Ethernet switch functions and settings cannot be changed.

## Managed Layer 2 (L2) Ethernet Switches

These Ethernet switches use the Ethernet MAC address to switch ports. Ethernet switch functions and settings can be changed with special software tools for Ethernet switches running on a network node. You can also collect analytical data. These Ethernet switches provide more-advanced functions than unmanaged layer 2 Ethernet switches.

## 2-1-3 Ethernet Switch Functions

This section describes the Ethernet switch functions that are important for an EtherNet/IP network. For a built-in EtherNet/IP port, consider whether the Ethernet switch supports these functions when you select the Ethernet switch.

- · Multicast filtering
- QoS (Quality of Service) for TCP/UDP port numbers (L4)

## Multicast Filtering

Multicast filtering transfers multicast packets to the specific nodes only. This function is implemented in the Ethernet switch as IGMP snooping or GMRP. "Specific nodes" are nodes equipped with an IGMP client that have made transfer requests to the Ethernet switch. (OMRON built-in EtherNet/IP ports are equipped with an IGMP client.) When the Ethernet switch does not use multicast filtering, multicast packets are sent to all nodes, just like broadcast packets, which increases the traffic in the network. Settings must be made in the Ethernet switch to enable this function. There must be enough multicast filters for the network.

## QoS (Quality of Service) Function for TCP/UDP Port Numbers (L4)

This function controls the priority of packet transmissions so that packets can be sent with higher priority to a particular IP address or TCP (UDP) port. The TCP and UDP protocols are called transport layer protocols, leading to the name L4 (layer 4) QoS function. When tag data links and message communications are executed on the same network, tag data links can be sent at higher priority to prevent problems such as transmission delays due to message communications traffic and packet losses due to buffer overflow. Settings must be made in the Ethernet switch to enable this function and give higher priority to tag data link packets.

Support for the above two functions is as follows for the different types of Ethernet switches.

Ethernet Switch Types	Multicast filtering	L4 QoS	Remarks
Unmanaged L2 Ethernet switches	None	None	
Managed L2 Ethernet switches	Provided.	Provided.	Both functions must be set with a special software tool.
OMRON W4S1-series Ethernet switches	None	Provided.	L4 QoS is set with a switch. No software tool is necessary.



## **Additional Information**

If the Network Configurator is used to set the connection type in the connection settings to a multicast connection, multicast packets are used. If the connection type is set to a point-to-point connection, multicast packets are not used.

#### 2-1-4 **Precautions for Ethernet Switch Selection**

The functions supported by the Ethernet switch may affect tag data link transmission delays and the settings in the Controller configurations and setup. In addition, if the Ethernet switch supports advanced functions, special settings are required for those functions. When you select an Ethernet switch, it is necessary to consider whether to select the Ethernet switch based on the kind and amount of communications you want to execute in the network. Refer to the following precautions when you select an Ethernet switch. Refer to 14-2 Adjusting the Communications Load to estimate the communications load for tag data links.

## Selecting the Ethernet Switch Based on the Types of Network **Communications**

## Executing Tag Data Links Only

We recommend that you use an L2 Ethernet switch without multicast filtering or an L2 Ethernet switch with multicast filtering. An L2 Ethernet switch with multicast filtering prevents increased traffic due to unnecessary multicast packets, so the tag data links can operate at higher speed. If either of the following conditions exists, the amount traffic will be the same for both kinds of L2 Ethernet switches (with or without multicast filtering).

- The tag data links are set to share the same data with all nodes in the network. (The multicast packets are transferred to all nodes in the network, just like a broadcast.)
- The tag data link settings are all one-to-one (unicast) and multicast packets cannot be used.

If multicast filters are being used, settings must be made in the Ethernet switch. There must be enough multicast filters for the network being used.

## Executing Tag Data Links and Message Communications

We recommend an L2 Ethernet switch with multicast filtering and L4 QoS. If you set tag data links for higher-priority transmission, it is possible to prevent problems such as transmission delays due to message communications traffic and packet losses due to buffer overflow. You must make special settings in the Ethernet switch when using the multicast filtering function and L4 QoS function.

## Selecting the Ethernet Switch Based on the Ethernet Switch's Supported Functions

## L2 Ethernet Switch without Multicast Filtering

We recommend this kind of Ethernet switch when only tag data links are executed and any of the following conditions is met.

- The tag data links are set to share the same data with all nodes in the network. (The multicast packets are transferred to all nodes in the network, just like a broadcast.)
- The tag data link settings are all one-to-one (unicast) and multicast packets cannot be used.
- · There is little traffic in the tag data links.

No special settings are required for an L2 Ethernet switch without multicast filtering.

## L2 Ethernet Switch with Multicast Filtering

We recommend this kind of Ethernet switch when only tag data links are executed and the following condition is met.

There are many 1:N links (where N represents some number of nodes in the network) in the tag
data link settings, i.e., there are many multicast packets used, or there is heavy traffic in the tag
data links.

Special settings are required for an L2 Ethernet switch with multicast filtering. There must be enough multicast filters for the network.

## L3 Ethernet Switch with Multicast Filtering and L4 QoS Functions

We recommend this kind of Ethernet switch when both tag data links and message communications are executed. If you set tag data links for higher-priority transmission, you can prevent problems such as transmission delays due to message communications traffic and packet losses due to buffer overflow. Special settings must be made in the Ethernet switch when using the multicast filtering function and L4 QoS function. There must be enough multicast filters for the network.

## Selecting the Ethernet Switch Based on the Network Communication Speed

## Executing Tag Data Links at a Communication Speed Over 100 Mbps (NX-series CPU Unit)

If you will use data tag links with the following conditions, use an Ethernet switch with a multicast filter or an Ethernet switch that supports a communication speed of 1,000 Mbps.

- Multicast
- Communication speed over 100 Mbps

If there is an Ethernet device on the same network that communicates at a speed of 100 Mbps or less, the device may affect tag data link communication and cause tag data links to be broken, even if the device is not related to tag data link communication.



## **Precautions for Correct Use**

- Ask the Ethernet switch manufacturer for setting procedures for the Ethernet switch.
- Install the Ethernet switch so that its environmental resistance specifications are not exceeded. Ask the Ethernet switch manufacturer for information on the environmental resistance of the Ethernet switch.

#### **Network Installation** 2-2

#### 2-2-1 **Basic Installation Precautions**

- Take the greatest care when you install the Ethernet System. Be sure to follow ISO 8802-3 specifications. Be sure you understand them before attempting to install an Ethernet System.
- · Unless you are already experienced in installation of communications systems, we strongly recommend that you employ a professional to install your system.
- Do not install Ethernet equipment near sources of noise. If a noisy environment is unavoidable, take adequate measures against noise interference, such as installation of network components in metal cases or the use of optical cable in the system.
- · When using a shielded cable with the shields on both ends of the cable connected to connector hoods, ground loops induced by improper grounding methods may decrease noise immunity and cause device damage. To prevent ground loops caused by differences in potential between device grounding points, the reference potential between the devices must be stabilized. Design grounding appropriately so that noise current does not flow to ground lines between the devices. For grounding methods, refer to the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500) and the NXseries CPU Unit Hardware User's Manual (Cat. No. W535).
- To obtain information on laying EtherNet/IP cable, contact ODVA. ODVA web site: http://www.odva.org
- When you install an EtherNet/IP network that combines an information network with the control system, and the communications load may be heavy due to tag data links, we recommend that you set up the network so that the load does not affect communications. For example, install the tag data links in a segment that is separate from the information network.

#### 2-2-2 **Recommended Network Devices**

Refer to 2-1 Selecting the Network Devices for the devices recommended for use with the built-in Ether-Net/IP port.

#### 2-2-3 **Precautions When Laying Twisted-pair Cable**

## Connecting the Shield to Connector Hoods

## Between an EtherNet/IP Port and an Ethernet Switch

Connect the shield to connector hoods as described below.

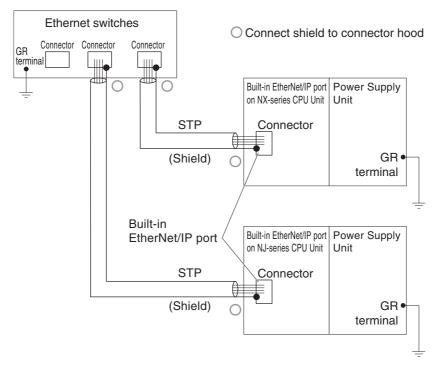
NJ-series CPU Unit		NX-series CPU Unit		
10Base-T	100Base-TX	10Base-T	100Base-TX	1000Base-T
Connect both ends or     Connect Ethernet S	switch side only	Connect both ends or     Connect the Ethern clamp core must be erNet/IP port side or	attached to the Eth-	Connect both ends

10Base-T or 100Base-TX

Connect the cable shields to the connector hoods as described in either a) or b) below.

## a) Connecting the shields at both ends of the cable

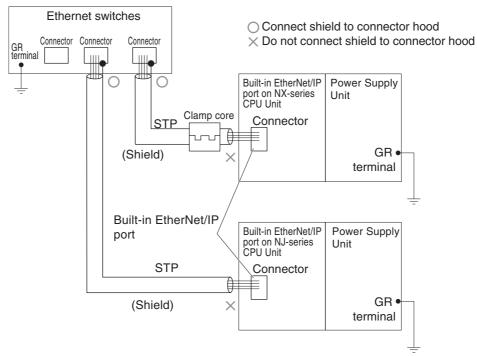
Connect the shields at both ends of the cables to connector hoods.



#### b) Connecting the shields on the Ethernet switch side only

Connect only the shield at the end of the cable on the Ethernet switch side to the connector hood.

- For an NX-series CPU Unit, a clamp core must be attached at the end of the cable on the EtherNet/IP port side. For the recommended clamp core and attachment method, refer to Recommended Clamp Core and Attachment Method. To comply with EMC standards, it is mandatory that a clamp core be attached when connecting the shield only to the connector hood on the Ethernet switch side.
- For an NJ-series CPU Unit, it is not necessary to connect a clamp core.



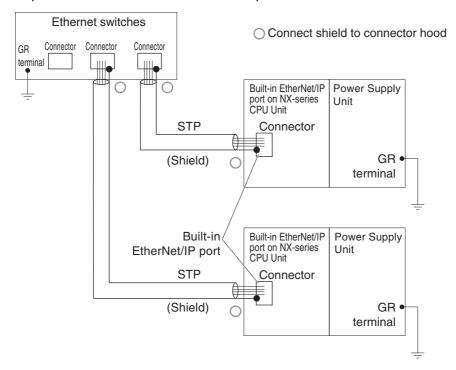


#### **Additional Information**

Noise immunity may be reduced and device damage may occur due to ground loops, which can occur due to improper shield connections and grounding methods. When using a baud rate of 100 Mbps or less, it may be possible to alleviate this problem by connecting only the Ethernet switch side as described in b), rather than connecting both ends as described in a).

#### 1000Base-T

Connect the shields at the both ends of the cable to respective connector hoods. This connection is required with 1000Base-T to ensure compliance with EMC standards.



#### Between two Ethernet switches

Regardless of which baud rate is used, check with the Ethernet switch manufacturers for information about installing the network between Ethernet switches, and in particular whether or not it is necessary to connect the cable shields to the connector hoods.

## Other Precautions When Laying the Twisted-pair Cable

- Press the cable connector in firmly until it locks into place at both the Ethernet switch and the built-in EtherNet/IP port.
- Do not lay the twisted-pair cable together with high-voltage lines.
- Do not lay the twisted-pair cable near devices that generate noise.
- · Do not lay the twisted-pair cable in locations subject to high temperatures or high humidity.
- Do not lay the twisted-pair cable in locations subject to excessive dirt and dust or to oil mist or other contaminants.

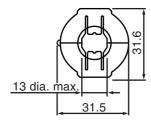
## **Recommended Clamp Core and Attachment Method**

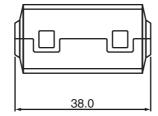
If you connect a shielded cable with both ends connected to connector hoods to an NX-series CPU Unit, you must attach a clamp core to the EtherNet/IP port side. The recommended clamp core and attachment method are given below.

## Recommended clamp core

Manufacturer	Product	Model
NEC TOKIN	Clamp core	ESD-SR-250

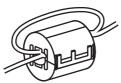
## **ESD-SR-250 dimensions**





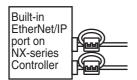
## Recommended attachment method

### (1) Cable attachment method



Make two loops with the cable as shown.

#### (2) Attachment method



Attach close to the cable connection as shown.

#### 2-2-4 **Precautions When Installing and Connecting Ethernet Switches**

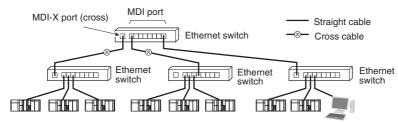
## **Precautions When Installing Ethernet Switches**

- · Do not ground the Ethernet switch in the same location as a drive-system component, such as an inverter.
- Always use a dedicated power supply for the Ethernet switch's power supply. Do not use the same power supply for other equipment, such as an I/O power supply, motor power supply, or control power supply.
- · Before installation, check the Ethernet switch's environmental resistance specifications, and use an Ethernet switch that is appropriate for the ambient conditions. Contact the Ethernet switch manufacturer for details on Ethernet switch's environmental resistance specifications.

## **Ethernet Switch Connection Methods**

 To connect Ethernet switches with twisted-pair cables, observe the followings: Connect an MDI port to an MDI-X port with a straight cable. Connect two MDI ports or two MDI-X ports with a cross cable.

Note It is very difficult to distinguish cross cables and straight cables by appearance. Incorrect cables will cause communications to fail. We recommend cascade connections with straight cables whenever possible.



 Some Ethernet switches can automatically distinguish between MDI and MDI-X. When this kind of Ethernet switch is used, straight cable can be used between Ethernet switches.



### **Precautions for Correct Use**

Adjust the built-in EtherNet/IP port's link settings to match the communications settings of the connected Ethernet switch. If the settings do not match, the link will be unstable and prevent normal communications. The following table shows the allowed settings for each Ethernet switch communications mode. (Auto-Nego: Auto negotiation, Full: Full duplex, Half: Half duplex)

Ethernet switch		Built-in EtherNet/IP port					
		Auto- Nego	10 Mbps (fixed)		100 Mbps (fixed)		1,000 Mbps (fixed)
			Full	Half	Full	Half	Full
Auto-Nego		Best		OK		OK	
10 Mbps (fixed)	Full		OK				
	Half	OK		OK			
100 Mbps (fixed)	Full				OK		
	Half	OK				OK	
1,000 Mbps (fixed)	Full						Best

Best = Recommended; OK = Allowed; --- = Not allowed.

## 2-3 Connecting to the Network

## 2-3-1 Ethernet Connectors

The following standards and specifications apply to the connectors for the Ethernet twisted-pair cable.

- Electrical specifications: Conforming to IEEE 802.3 standards.
- Connector structure: RJ45 8-pin Modular Connector (conforming to ISO 8877)
- For information on connecting shield wire to connector hoods, refer to 2-1-2 Ethernet Switch Types.

## 10Base-T and 100Base-TX



Connector pin	Signal name	Abbr.	Signal direc- tion	
1	Transmission data +	TD+	Output	
2	Transmission data –	TD-	Output	
3	Reception data +	RD+	Input	
4	Not used.			
5	Not used.			
6	Reception data –	RD-	Input	
7	Not used.			
8	Not used.			

## 1000Base-T



Connector pin	Signal name	Abbr.	Signal direc- tion	
1	Communication data DA+	BI_DA+	Input/output	
2	Communication data DA-	BI_DA-	Input/output	
3	Communication data DB+	BI_DB+	Input/output	
4	Communication data DC+	BI_DC+	Input/output	
5	Communication data DC-	BI_DC-	Input/output	
6	Communication data DB-	BI_DB-	Input/output	
7	Communication data DD+	BI_DD+	Input/output	
8	Communication data DD-	BI_DD-	Input/output	

#### 2-3-2 **Connecting the Cable**



## **Precautions for Correct Use**

- Turn OFF the Controller's power supply before connecting or disconnecting Ethernet communications cable.
- Allow extra space for the bending radius of the communications cable. For the CPU Unit dimensions when the communications cable is connected to the Unit, refer to the NJ-series CPU Unit Hardware User's Manual (Cat. No. W500) and the NX-series CPU Unit Hardware User's Manual (Cat. No. W535). The required space depends on the communications cable and connector that are used. Consult the manufacturer or sales agent.
- 1 Lay the twisted-pair cable.
- 2 Connect the cable to the Ethernet switch.
- 3 Connect the twisted-pair cable to the connector on the built-in EtherNet/IP port. Be sure to press the connectors (both the Ethernet switch side and Ethernet side) until they lock into place.



# System-defined Variables Related to the Built-in EtherNet/IP Port

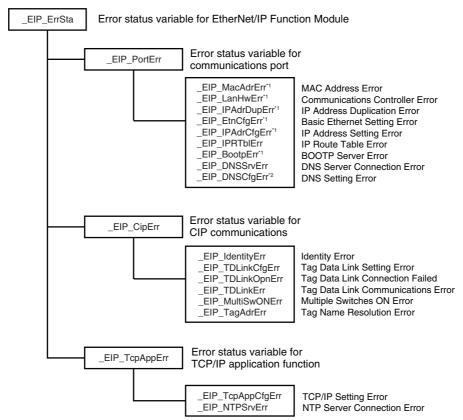
3-1	System-defined Variables Related to the Built-in EtherNet/IP Port	3-2
3-2	System-defined Variables	3-3
3-3	Specifications for Individual System-defined Variables	3-33

## System-defined Variables Related to 3-1 the Built-in EtherNet/IP Port

You can use the system-defined variables that are provided for the built-in EtherNet/IP port in programs to check the status of the built-in EtherNet/IP port.

## Checking for Errors in the Built-in EtherNet/IP Port

You can check for built-in EtherNet/IP port errors, Sysmac Studio setting errors, Network Configurator setting errors, TCP/IP application errors (e.g., FTP or NTP), etc. The following hierarchy is used. The system gives the error status at each level by logically ORing the error status information in the next lower level.



- \*1 Error status variables for errors related to NX-series Controllers are provided individually for communications port 1 and communications port 2. Refer to Hierarchical Relationship of System-defined Variables Related to EtherNet/IP Errors in the NX-series CPU Unit.
- \*2 Only NX-series CPU Units have this system-defined variable.

## 3-2 System-defined Variables

The variables are described in the tables as shown below.

Variable name	Meaning	Function	Data type	Range of values	Reference
This is the system- defined variable name. The prefix gives the category name.	This is the meaning of the variable.	The function of the variable is described.	The data type of the variable is given.	The range of values that the variable can take is given.	The page of the individual system- defined vari- able specifi- cations table is given.

## • Functional Classification: EtherNet/IP Communications Errors

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_ErrSta	Built-in EtherNet/IP Error	This is the error status variable for the built-in EtherNet/IP port.	WORD	16#0000 to 16#00F0	page 3-33
		NX-series CPU Units: Represents the collective status of the following error flags.			
		_EIP1_PortErr (Communications Port1 Error)			
		• _EIP2_PortErr (Communications Port2 Error)			
		_EIP1_CipErr (CIP Communications1 Error)			
		_EIP2_CipErr (CIP Communications2 Error)			
		_EIP_TcpAppErr (TCP Application Com- munications Error)			
		NJ-series CPU Units: Represents the collective status of the following error flags.			
		_EIP_PortErr (Communications Port Error)			
		_EIP_CipErr (CIP Communications Error)			
		_EIP_TcpAppErr (TCP Application Communications Error)			
		Note Refer to information on the meanings of the error status bits at the end of this appendix for details.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_PortErr	Communications Port Error	This is the error status variable for the communications port.  NX-series CPU Units: Represents the collective status of the following error flags.  • _EIP1_MacAdrErr (Port1 MAC Address Error)  • _EIP1_LanHwErr (Port1 Communications Controller Error)  • _EIP1_EtnCfgErr (Port1 Basic Ethernet Setting Error)  • _EIP1_IPAdrCfgErr (Port1 IP Address Setting Error)  • _EIP1_IPAdrDupErr (Port1 IP Address Duplication Error)  • _EIP1_BootpErr (Port1 BOOTP Server Error)  • _EIP_DNSCfgErr (DNS Setting Error)  • _EIP_DNSCfgErr (DNS Setting Error)  • _EIP_DNSSrvErr (DNS Server Connection Error)  • _EIP_IPRTblErr (IP Route Table Error)  NJ-series CPU Units: Represents the collective status of the following error flags.  • _EIP_MacAdrErr (MAC Address Error)  • _EIP_LanHwErr (Communications Controller Error)  • _EIP_EtnCfgErr (Basic Ethernet Setting Error)  • _EIP_EtnCfgErr (IP Address Setting Error)  • _EIP_IPAdrCpgErr (IP Address Duplication Error)  • _EIP_IPAdrDupErr (IP Route Table Error)  Note If a Link OFF Detected or Built-in EtherNet/IP Processing Error occurs, it is recorded in the event log and then the corresponding bit turns ON. Refer to information on the meanings of the error status bits at the end of this appendix for	WORD	16#0000 to 16#00F0	page 3-34
		details.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP1_PortErr	Communications Port1 Error	This is the error status variable for the communications port.  It represents the collective status of the following error flags.  • _EIP1_MacAdrErr (Port1 MAC Address Error)  • _EIP1_LanHwErr (Port1 Communications Controller Error)  • _EIP1_EtnCfgErr (Port1 Basic Ethernet Setting Error)	WORD	16#0000 to 16#00F0	page 3-35
		_EIP1_IPAdrCfgErr (Port1 IP Address Setting Error)     _EIP1_IPAdrDupErr (Port1 IP Address Duplication Error)     _EIP1_BootpErr (Port1 BOOTP Server Error)     _EIP_DNSCfgErr (DNS Setting Error)     _EIP_DNSSrvErr (DNS Server Connection Error)     _EIP_IPRTblErr (IP Route Table Error)			
		Note If a Link OFF Detected or Built-in EtherNet/IP Processing Error occurs, it is recorded in the event log and then the corresponding bit turns ON.  Refer to information on the meanings of the error status bits at the end of this appendix for details.			
_EIP2_PortErr	Communications	Note You can use this system-defined variable only for NX-series CPU Units.  This is the error status variable for the	WORD	16#0000 to	page 3-35
	Port2 Error	communications port.  It represents the collective status of the following error flags.  • _EIP2_MacAdrErr (Port2 MAC Address Error)  • _EIP2_LanHwErr (Port2 Communications Controller Error)  • _EIP2_EtnCfgErr (Port2 Basic Ethernet Setting Error)  • _EIP2_IPAdrCfgErr (Port2 IP Address Setting Error)  • _EIP2_IPAdrDupErr (Port2 IP Address Duplication Error)  • _EIP2_BootpErr (Port2 BOOTP Server Error)  • _EIP2_DNSCfgErr (DNS Setting Error)  • _EIP_DNSSrvErr (DNS Server Connection Error)  • _EIP_IPRTblErr (IP Route Table Error)  Note If a Link OFF Detected or Built-in EtherNet/IP Processing Error occurs, it is recorded in the event log and then the corresponding bit turns ON.  Refer to information on the meanings of the error status bits at the end of this appendix for details.  Note You can use this system-defined vari-		16#00F0	page o oo

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_CipErr	CIP Communications Error	This is the error status variable for CIP communications.  NX-series CPU Units: Represents the collective status of the following error flags.	WORD	16#0000 to 16#00F0	page 3-36
		_EIP1_IdentityErr (CIP Communications1 Identity Error)     _EIP1_TDLinkCfgErr (CIP Communications1 Tag Data Link Setting Error)     _EIP1_TDLinkOpnErr (CIP Communications1 Tag Data Link Connection Failed)     _EIP1_TDLinkErr (CIP Communications1 Tag Data Link Communications Error)     _EIP1_TagAdrErr (CIP Communications1 Tag Name Resolution Error)     _EIP1_MultiSwONErr (CIP Communications1 Tag Name Resolution Error)			
		NJ-series CPU Units: Represents the collective status of the following error flags.  • _EIP_IdentityErr (Identity Error)  • _EIP_TDLinkCfgErr (Tag Data Link Setting Error)  • _EIP_TDLinkOpnErr (Tag Data Link Connection Failed)  • _EIP_TDLinkErr (Tag Data Link Communications Error)  • _EIP_TagAdrErr (Tag Name Resolution Error)  • _EIP_MultiSwONErr (Multiple Switches ON Error)  Note If a Tag Name Resolution Error			
		occurs, it is recorded in the event log and this variable changes to TRUE. Refer to information on the meanings of the error status bits at the end of this appendix for details.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP1_CipErr	CIP Communications1 Error	This is the error status variable for CIP communications 1.  It represents the collective status of the following error flags.  • _EIP1_IdentityErr (CIP Communications1 Identity Error)  • _EIP1_TDLinkCfgErr (CIP Communications1 Tag Data Link Setting Error)  • _EIP1_TDLinkOpnErr (CIP Communications1 Tag Data Link Connection Failed)  • _EIP1_TDLinkErr (CIP Communications1 Tag Data Link Communications Error)  • _EIP1_TagAdrErr (CIP Communications1 Tag Name Resolution Error)  • _EIP1_MultiSwONErr (CIP Communications1 Tag Name Resolution Error)	WORD	16#0000 to 16#00F0	page 3-36
		Note If a Tag Name Resolution Error occurs, it is recorded in the event log and this variable changes to TRUE. Refer to information on the meanings of the error status bits at the end of this appendix for details.  Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_CipErr	CIP Communications2 Error	This is the error status variable for CIP communications 2.  It represents the collective status of the following error flags.  • _EIP2_IdentityErr (CIP Communications2 Identity Error)  • _EIP2_TDLinkCfgErr (CIP Communications2 Tag Data Link Setting Error)  • _EIP2_TDLinkOpnErr (CIP Communications2 Tag Data Link Connection Failed)  • _EIP2_TDLinkErr (CIP Communications2 Tag Data Link Communications Error)  • _EIP2_TagAdrErr (CIP Communications2 Tag Name Resolution Error)  • _EIP2_MultiSwONErr (CIP Communications2 Tag Name Resolution Error occurs, it is recorded in the event log and this variable changes to TRUE. Refer to information on the meanings of the error status bits at the end of this appendix for details.  Note You can use this system-defined variable only for NX-series CPU Units.	WORD	16#0000 to 16#00F0	page 3-37

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_TcpAppErr	TCP Application Communications Error	This is the error status variable for TCP application communications.  It represents the collective status of the following error flags.	WORD	16#0000 to 16#00F0	page 3-37
		_EIP_TcpAppCfgErr (TCP/IP Setting Error)     _EIP_NTPSrvErr (NTP Server Connection Error)			
		Note Refer to information on the meanings of the error status bits at the end of this appendix for details.			
_EIP_MacAdrErr	MAC Address Error	NX-series CPU Units: Indicates that an error occurred when the MAC address was read on the communications port 1 at startup.  TRUE: Error	BOOL	TRUE or FALSE	page 3-37
		FALSE: Normal  NJ-series CPU Units: Indicates that an error occurred when the MAC address was read at startup.  TRUE: Error  FALSE: Normal			
_EIP1_MacAdrErr	Port1 MAC Address Error	Indicates that an error occurred when the MAC address was read on the communications port 1 at startup.	BOOL	TRUE or FALSE	page 3-38
		TRUE: Error FALSE: Normal  Note You can use this system-defined vari-			
FIDO Mara Astron	Darlo MAO Address	able only for NX-series CPU Units.	DOOL	TDUE	
_EIP2_MacAdrErr	Port2 MAC Address Error	Indicates that an error occurred when the MAC address was read on the communications port 2 at startup.	BOOL	TRUE or FALSE	page 3-38
		TRUE: Error FALSE: Normal			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP_LanHwErr	Communications Controller Error	NX-series CPU Units: Indicates that a communications controller failure occurred on the communications port 1.  TRUE: Failure	BOOL	TRUE or FALSE	page 3-38
		FALSE: Normal			
		NJ-series CPU Units: Indicates that a communications controller failure occurred.			
		TRUE: Failure			
_EIP1_LanHwErr	Port1 Communications Controller Error	FALSE: Normal Indicates that a communications controller failure occurred on the communications	BOOL	TRUE or FALSE	page 3-38
		port 1. TRUE: Failure			
		FALSE: Normal			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP2_LanHwErr	Port2 Communications Controller Error	Indicates that a communications controller failure occurred on the communications port 2.  TRUE: Failure  FALSE: Normal  Note You can use this system-defined vari-	BOOL	TRUE or FALSE	page 3-39
		able only for NX-series CPU Units.			
_EIP_EtnCfgErr	Basic Ethernet Setting Error	NX-series CPU Units: Indicates that the Ethernet communications speed setting (Speed/Duplex) for the communications port 1 is incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-39
		TRUE: Setting incorrect or read failed FALSE: Normal			
		NJ-series CPU Units: Indicates that the Ethernet communications speed setting (Speed/Duplex) is incorrect. Or, a read operation failed.			
		TRUE: Setting incorrect or read failed			
		FALSE: Normal			
_EIP1_EtnCfgErr	Port1 Basic Ethernet Setting Error	Indicates that the Ethernet communications speed setting (Speed/Duplex) for the communications port 1 is incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-39
		TRUE: Setting incorrect or read failed			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_EtnCfgErr	Port2 Basic Ethernet Setting Error	Indicates that the Ethernet communications speed setting (Speed/Duplex) for the communications port 2 is incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-39
		TRUE: Setting incorrect or read failed			
		FALSE: Normal			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_IPAdrCfgErr	IP Address Setting Error	NX-series CPU Units: Indicates the IP address setting errors for the communications port 1.  TRUE:  There is an illegal IP address setting.  A read operation failed.  The IP address obtained from the BOOTP server is inconsistent.  The DNS settings are not correct.  FALSE: Normal  NJ-series CPU Units: Indicates the IP address setting errors.  TRUE:  There is an illegal IP address setting.  A read operation failed.  The IP address obtained from the BOOTP server is inconsistent.  The DNS settings are not correct.	BOOL	TRUE or FALSE	page 3-40
_EIP1_IPAdrCfgErr	Port1 IP Address Setting Error	FALSE: Normal  Indicates the IP address setting errors for the communications port 1.  TRUE:  • There is an illegal IP address setting.  • A read operation failed.  • The IP address obtained from the BOOTP server is inconsistent.  • The DNS settings are not correct.  FALSE: Normal  Note You can use this system-defined variable only for NX-series CPU Units.	BOOL	TRUE or FALSE	page 3-40
_EIP2_IPAdrCfgErr	Port2 IP Address Setting Error	Indicates the IP address setting errors for the communications port 2.  TRUE:  There is an illegal IP address setting.  A read operation failed.  The IP address obtained from the BOOTP server is inconsistent.  The DNS settings are not correct.  FALSE: Normal  Note You can use this system-defined variable only for NX-series CPU Units.	BOOL	TRUE or FALSE	page 3-40

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_IPAdrDupErr	IP Address Duplication Error	NX-series CPU Units: Indicates that the same IP address is assigned to more than one node for the communications port 1.  TRUE: Duplication occurred.  FALSE: Other than the above.  NJ-series CPU Units: Indicates that the same IP address is assigned to more than one node.  TRUE: Duplication occurred.  FALSE: Other than the above.	BOOL	TRUE or FALSE	page 3-41
_EIP1_IPAdrDupErr	Port1 IP Address Duplication Error	Indicates that the same IP address is assigned to more than one node for the communications port 1.  TRUE: Duplication occurred.  FALSE: Other than the above.  Note You can use this system-defined variable only for NX-series CPU Units.	BOOL	TRUE or FALSE	page 3-41
_EIP2_IPAdrDupErr	Port2 IP Address Duplication Error	Indicates that the same IP address is assigned to more than one node for the communications port 2.  TRUE: Duplication occurred.  FALSE: Other than the above.  Note You can use this system-defined variable only for NX-series CPU Units.	BOOL	TRUE or FALSE	page 3-41
_EIP_DNSCfgErr	DNS Setting Error	Indicates that the DNS or hosts settings are incorrect. Or, a read operation failed.  TRUE: Setting incorrect or read failed FALSE: Other than the above.  Note You can use this system-defined variable only for NX-series CPU Units.	BOOL	TRUE or FALSE	page 3-41
_EIP_BootpErr	BOOTP Server Error	NX-series CPU Units: Indicates that a BOOTP server connection failure occurred on the communications port 1. TRUE: There was a failure to connect to the BOOTP server (timeout).  FALSE: The BOOTP is not enabled, or BOOTP is enabled and an IP address was normally obtained from the BOOTP server.  NJ-series CPU Units: Indicates that a BOOTP server connection failure occurred.  TRUE: There was a failure to connect to the BOOTP server (timeout).  FALSE: The BOOTP is not enabled, or BOOTP is enabled and an IP address was normally obtained from the BOOTP server.	BOOL	TRUE or FALSE	page 3-42
_EIP1_BootpErr	Port1 BOOTP Server Error	Indicates that a BOOTP server connection failure occurred on the communications port 1.  TRUE: There was a failure to connect to the BOOTP server (timeout).  FALSE: The BOOTP is not enabled, or BOOTP is enabled and an IP address was normally obtained from the BOOTP server.  Note You can use this system-defined variable only for NX-series CPU Units.	BOOL	TRUE or FALSE	page 3-42

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP2_BootpErr	Port2 BOOTP Server Error	Indicates that a BOOTP server connection failure occurred on the communications port 2.  TRUE: There was a failure to connect to the BOOTP server (timeout).  FALSE: The BOOTP is not enabled, or	BOOL	TRUE or FALSE	page 3-42
		BOOTP is enabled and an IP address was normally obtained from the BOOTP server.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP_IPRTblErr	IP Route Table Error	NX-series CPU Units: Indicates that the default gateway settings or IP router table settings are incorrect. Or, a read operation failed. TRUE: Setting incorrect or read failed FALSE: Normal	BOOL	TRUE or FALSE	page 3-43
		Note For NX-series CPU Units, the host setting errors will be indicated to _EIP_DNSCfgErr.			
		NJ-series CPU Units: Indicates that the IP router table or hosts settings are incorrect. Or, a read operation failed.			
		TRUE: Setting incorrect or read failed FALSE: Normal			
		Note For NJ-series CPU Units, the default gateway setting errors will be indicated to _EIP_IPAdrCfgErr.			
_EIP_IdentityErr	Identity Error	NX-series CPU Units: Indicates that the identity information for CIP communications 1 (which you cannot overwrite) is incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-43
		TRUE: Setting incorrect or read failed FALSE: Normal			
		NJ-series CPU Units: Indicates that the identity information (which you cannot overwrite) is incorrect. Or, a read operation failed.			
		TRUE: Setting incorrect or read failed FALSE: Normal			
_EIP1_IdentityErr	CIP Communications1 Identity Error	Indicates that the identity information for CIP communications 1 (which you cannot overwrite) is incorrect. Or, a read operation failed.  TRUE: Setting incorrect or read failed FALSE: Normal	BOOL	TRUE or FALSE	page 3-43
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_IdentityErr	CIP Communications2 Identity Error	Indicates that the identity information for CIP communications 2 (which you cannot overwrite) is incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-44
		TRUE: Setting incorrect or read failed FALSE: Normal			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference		
_EIP_TDLinkCfgErr	Tag Data Link Setting Error	NX-series CPU Units: Indicates that the tag data link settings for CIP communications 1 are incorrect. Or, a read operation failed.  TRUE: Setting incorrect or read failed FALSE: Normal NJ-series CPU Units: Indicates that the	BOOL	BOOL TRUE or FALSE			page 3-44
		tag data link settings are incorrect. Or, a read operation failed.  TRUE: Setting incorrect or read failed					
		FALSE: Normal					
_EIP1_TDLinkCfgErr	CIP Communications1 Tag Data Link Setting	Indicates that the tag data link settings for CIP communications 1 are incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-44		
	Error	TRUE: Setting incorrect or read failed FALSE: Normal					
		Note You can use this system-defined variable only for NX-series CPU Units.					
_EIP2_TDLinkCfgErr	CIP Communications2 Tag Data Link Setting Error	Indicates that the tag data link settings for CIP communications 2 are incorrect. Or, a read operation failed.  TRUE: Setting incorrect or read failed FALSE: Normal	BOOL	TRUE or FALSE	page 3-44		
		Note You can use this system-defined variable only for NX-series CPU Units.					
_EIP_TDLinkOpnErr	Tag Data Link Con- nection Failed	NX-series CPU Units: Indicates that establishing a tag data link connection for CIP communications 1 failed.	BOOL	TRUE or FALSE	page 3-45		
		TRUE: Establishing a tag data link con- nection failed due to one of the following causes.					
		<ul> <li>The information registered for a target node in the tag data link parameters is different from the actual node informa- tion.</li> </ul>					
		<ul> <li>There was no response from the remote node.</li> </ul>					
		FALSE: Other than the above.					
		NJ-series CPU Units: Indicates that establishing a tag data link connection failed.					
		TRUE: Establishing a tag data link con- nection failed due to one of the following causes.					
		<ul> <li>The information registered for a target node in the tag data link parameters is different from the actual node informa- tion.</li> </ul>					
		<ul> <li>There was no response from the remote node.</li> </ul>					
-		FALSE: Other than the above.					

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP1_TDLinkOpnErr	nErr CIP Communications1 Tag Data Link Connection Failed	Indicates that establishing a tag data link connection for CIP communications 1 failed.  TRUE: Establishing a tag data link connection failed due to one of the following causes.	BOOL	TRUE or FALSE	page 3-45
		The information registered for a target node in the tag data link parameters is different from the actual node informa- tion.			
		There was no response from the remote node.			
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_TDLinkOpnErr	CIP Communications2 Tag Data Link Con-	Indicates that establishing a tag data link connection for CIP communications 2 failed.	BOOL	TRUE or FALSE	page 3-45
	nection Failed	TRUE: Establishing a tag data link con- nection failed due to one of the following causes.			
		<ul> <li>The information registered for a target node in the tag data link parameters is different from the actual node informa- tion.</li> </ul>			
		There was no response from the remote node.			
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP_TDLinkErr	Tag Data Link Com- munications Error	NX-series CPU Units: Indicates that a timeout occurred in a tag data link connection for CIP communications 1.	BOOL	TRUE or FALSE	page 3-46
		TRUE: A timeout occurred.			
		FALSE: Other than the above.			
		NJ-series CPU Units: Indicates that a timeout occurred in a tag data link connection.			
		TRUE: A timeout occurred.			
_		FALSE: Other than the above.			
_EIP1_TDLinkErr	CIP Communications1 Tag Data Link Communications Error	Indicates that a timeout occurred in a tag data link connection for CIP communications 1.	BOOL	TRUE or FALSE	page 3-46
munic	munications Endi	TRUE: A timeout occurred.			
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_TDLinkErr	CIP Communications2 Tag Data Link Com-	Indicates that a timeout occurred in a tag data link connection for CIP communications 2.	BOOL	TRUE or FALSE	page 3-46
	munications Error	TRUE: A timeout occurred.			
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_TagAdrErr	Tag Name Resolution Error	NX-series CPU Units: Indicates that tag resolution for CIP communications 1 failed (i.e., the address could not be identified from the tag name).	BOOL	TRUE or FALSE	page 3-47
		TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.			
		<ul> <li>The size of the network variable is different from the tag settings.</li> <li>The I/O direction that is set in the tag data link settings does not agree with the I/O direction of the variable in the CPU Unit.</li> </ul>			
		<ul> <li>There is no network variable in the CPU Unit that corre- sponds to the tag setting.</li> </ul>			
		FALSE: Other than the above.			
		NJ-series CPU Units: Indicates that tag resolution failed (i.e., the address could not be identified from the tag name).			
		TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.			
		<ul> <li>The size of the network variable is different from the tag settings.</li> </ul>			
		<ul> <li>The I/O direction that is set in the tag data link settings does not agree with the I/O direc- tion of the variable in the CPU Unit.</li> </ul>			
		<ul> <li>There is no network variable in the CPU Unit that corre- sponds to the tag setting.</li> </ul>			
		FALSE: Other than the above.			
_EIP1_TagAdrErr	CIP Communications1 Tag Name Resolution	Indicates that tag resolution for CIP communications 1 failed (i.e., the address could not be identified from the tag name).	BOOL	TRUE or FALSE	page 3-47
	Error	TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.			
		<ul> <li>The size of the network variable is different from the tag settings.</li> </ul>			
		The I/O direction that is set in the tag data link settings does not agree with the I/O direc- tion of the variable in the CPU Unit.			
		<ul> <li>There is no network variable in the CPU Unit that corre- sponds to the tag setting.</li> </ul>			
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Indicates that tag resolution for CIP communications 2 failed (i.e., the address could not be identified from the tag name).		ues	Reference
	BOOL	TRUE or FALSE	page 3-48
TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.			
<ul> <li>The size of the network variable is different from the tag settings.</li> </ul>			
<ul> <li>The I/O direction that is set in the tag data link settings does not agree with the I/O direc- tion of the variable in the CPU Unit.</li> </ul>			
<ul> <li>There is no network variable in the CPU Unit that corre- sponds to the tag setting.</li> </ul>			
FALSE: Other than the above.			
Note You can use this system-defined variable only for NX-series CPU Units.			
NX-series CPU Units: Indicates that more than one switch turned ON at the same time in CIP communications 1.	BOOL	TRUE or FALSE	page 3-48
TRUE: More than one data link start/stop switch changed to TRUE at the same time.			
FALSE: Other than the above.			
NJ-series CPU Units: Indicates that more than one switch turned ON at the same time.			
TRUE: More than one data link start/stop switch changed to TRUE at the same time.			
FALSE: Other than the above.			
Indicates that more than one switch turned ON at the same time in CIP communications 1.	BOOL	TRUE or FALSE	page 3-48
TRUE: More than one data link start/stop switch changed to TRUE at the same time.			
FALSE: Other than the above.			
Note You can use this system-defined variable only for NX-series CPU Units.			
Indicates that more than one switch turned ON at the same time in CIP communications 2.	BOOL	TRUE or FALSE	page 3-48
TRUE: More than one data link start/stop switch changed to TRUE at the same time.			
FALSE: Other than the above.			
Note You can use this system-defined variable only for NX-series CPU Units.			
TRUE: At least one of the set values for a TCP/IP application (FTP, NTP, SNMP) is incorrect. Or, a read operation failed.	BOOL	TRUE or FALSE	page 3-49
FALSE: Normal	POO!	TDUE	noge 0 40
the server (timeout).  FALSE: NTP is not set or the connection	BOOL	FALSE	page 3-49
T	SNMP) is incorrect. Or, a read operation failed.  ALSE: Normal  RUE: The NTP client failed to connect to the server (timeout).	SNMP) is incorrect. Or, a read operation failed.  ALSE: Normal  RUE: The NTP client failed to connect to the server (timeout).  ALSE: NTP is not set or the connection	SNMP) is incorrect. Or, a read operation failed.  ALSE: Normal  RUE: The NTP client failed to connect to the server (timeout).  ALSE: NTP is not set or the connection

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_DNSSrvErr	DNS Server Connection Error	TRUE: The DNS client failed to connect to the server (timeout).	BOOL	TRUE or FALSE	page 3-49
		FALSE: DNS is not enabled. Or, DNS is enabled and the connection was successful.			



#### **Additional Information**

## Hierarchical Relationship of System-defined Variables Related to EtherNet/IP Errors in the NJ-series CPU Unit

The system-defined variables that are related to EtherNet/IP errors have the following hierarchical relationship. For example, if the value of any of the \_EIP\_PortErr, \_EIP\_CipErr, or \_EIP\_TcpAppErr variables in the second level is TRUE, then the \_EIP\_ErrSta variable in the first level also changes to TRUE. Therefore, you can check the values of system-defined variables in a higher level to see if an error has occurred for a variable in a lower level.

Le	vel 1	Leve	el 2	Le	vel 3
Variable	Name	Variable	Name	Variable	Name
_EIP_ErrSta	Built-in Ether- Net/IP Error	_EIP_PortErr	Communica- tions Port Error	_EIP_MacAdr Err	MAC Address Error
				_EIP_LanHwErr	Communications Controller Error
				_EIP_EtnCfgErr	Basic Ethernet Set- tings Error
				_EIP_IPAdrCfg Err	IP Address Setting Error
				_EIP_IPAdrDupErr	IP Address Duplica- tion Error
				_EIP_BootpErr	BOOTP Server Error
				_EIP_DNSSrvErr	DNS Server Connection Error
				_EIP_IPRTblErr	IP Route Table Error
		_EIP_CipErr	CIP Communi- cations Error	_EIP_Identity Err	Identity Error
				_EIP_TDLink CfgErr	Tag Data Link Setting Error
				_EIP_TDLink OpnErr	Tag Data Link Con- nection Failed
				_EIP_TDLink Err	Tag Data Link Com- munications Error
				_ EIP_TagAdr Err	Tag Name Resolution Error
				_EIP_MultiSw ONErr	Multiple Switches ON Error
		_EIP_TcpAppErr	TCP Applica- tion Communi-	_EIP_TcpApp CfgErr	TCP/IP Setting Error
			cations Error	_EIP_NTPSrv Err	NTP Server Connection Error

#### Hierarchical Relationship of System-defined Variables Related to EtherNet/IP Errors in the **NX-series CPU Unit**

The system-defined variables that are related to EtherNet/IP errors have the following hierarchical relationship. For example, if the value of any of the \_EIP1\_PortErr, \_EIP2\_PortErr, EIP1\_CipErr, \_EIP2\_CipErr, and \_EIP\_TcpAppErr variables in the second level is TRUE, then the \_EIP\_ErrSta variable in the first level also changes to TRUE. Therefore, you can check the values of system-defined variables in a higher level to see if an error has occurred for a variable in a lower level.

Lev	rel 1	Lev	rel 2		Level 3
Variable	Name	Variable	Name	Variable	Name
_EIP_ErrSta	Built-in	_EIP1_Por	Communi-	_EIP1_MacAdrErr	Port1 MAC Address Error
	EtherNet/IP	tErr	cations	_EIP1_LanHwErr	Port1 Communications Controller Error
	Error		Port1	_EIP1_EtnCfgErr	Port1 Basic Ethernet Setting Error
			Error	_EIP1_IPAdrCfgErr	Port1 IP Address Setting Error
				_EIP1_IPAdrDupErr	Port1 IP Address Duplication Error
				_EIP1_BootpErr	Port1 BOOTP Server Error
				_EIP_DNSCfgErr	DNS Setting Error
				_EIP_DNSSrvErr	DNS Server Connection Error
				_EIP_IPRTblErr	IP Route Table Error
		_EIP2_Por	Communi-	_EIP2_MacAdrErr	Port2 MAC Address Error
		tErr	cations	_EIP2_LanHwErr	Port2 Communications Controller Error
			Port2	_EIP2_EtnCfgErr	Port2 Basic Ethernet Setting Error
			Error	_EIP2_IPAdrCfgErr	Port2 IP Address Setting Error
				_EIP2_IPAdrDupErr	Port2 IP Address Duplication Error
				_EIP2_BootpErr	Port2 BOOTP Server Error
				_EIP_DNSCfgErr	DNS Setting Error
				_EIP_DNSSrvErr	DNS Server Connection Error
				_EIP_IPRTblErr	IP Route Table Error
		_EIP1_Cip	CIP	_EIP1_IdentityErr	CIP Communications1 Identity Error
		Err	Communic	_EIP1_TDLinkCfgErr	CIP Communications1 Tag Data
			ations1 Error		Link Setting Error
			LIIOI	_EIP1_TDLinkOpnErr	CIP Communications1 Tag Data Link Connection Failed
				_EIP1_TDLinkErr	CIP Communications1 Tag Data Link Communications Error
				_EIP1_TagAdrErr	CIP Communications1 Tag Name Resolution Error
				_EIP1_MultiSwONErr	CIP Communications1 Multiple Switches ON Error
		_EIP2_Cip	CIP	_EIP2_IdentityErr	CIP Communications2 Identity Error
		Err	Communic ations2	_EIP2_TDLinkCfgErr	CIP Communications2 Tag Data Link Setting Error
			Error	_EIP2_TDLinkOpnErr	CIP Communications2 Tag Data Link Connection Failed
				_EIP2_TDLinkErr	CIP Communications2 Tag Data Link Communications Error
				_EIP2_TagAdrErr	CIP Communications2 Tag Name Resolution Error
				_EIP2_MultiSwONErr	CIP Communications2 Multiple Switches ON Error
		_EIP_Tcp	TCP	_EIP_TcpAppCfgErr	TCP/IP Setting Error
		AppErr	Applica- tion Com- municatio	_EIP_NTPSrvErr	NTP Server Connection Error
			ns Error		

Note You can access the same values of the system-defined variables whose variable names with \_EIP1 and the system-defined variables whose variable names with \_EIP. For example, you can access the same values of \_EIP1\_PortErr (Communications Port1 Error) and \_EIP\_PortErr (Communcations Port Error).

## Meanings of Error Status Bits

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WORD			_	_	1	_	_	_							_	_

error status of the Controller error. These bits show Unit errors. They will not change to TRUE for errors in E Net/IP communications.  TRUE: The master detected a Controller error. FALSE: The master has not detected a Controller error. (Valid for _CJB_U_ErrSta and _EC_SIvErrTbl.)  Collective slave error status: This bit indicates if a Controller error was detected for levels (e.g., a Unit, slave axis, or axes group) that are lower than the event source (i.e., for a function module).  TRUE: A Controller error has occurred at a lower level. FALSE: A Controller error has not occurred at a lower level. (Valid for _CJB_ErrSta, _MC_ErrSta, and _EC_ErrSta.)  8 to 13  Reserved.  This bit indicates whether a major fault level Controller error has occurred. TRUE: A major fault level Controller error has not occurred. FALSE: A major fault level Controller error has not occurred.  TRUE: A partial fault level Controller error has occurred. FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has not occurred.  TRUE: A minor fault level Controller error has not occurred.  TRUE: A minor fault level Controller error has not occurred.  FALSE: A minor fault level Controller error has not occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.	Bit	Meaning
FALSE: The master has not detected a Controller error. (Valid for _CJB_U_ErrSta and _EC_SIvErrTbl.)  Collective slave error status: This bit indicates if a Controller error was detected for levels (e.g., a Unit, slave axis, or axes group) that are lower than the event source (i.e., for a function module).  TRUE: A Controller error has occurred at a lower level.  FALSE: A Controller error has not occurred at a lower level. (Valid for _CJB_ErrSta, _MC_ErrSta, and _EC_ErrSta.)  8 to 13  Reserved.  This bit indicates whether a major fault level Controller error has occurred.  TRUE: A major fault level Controller error has not occurred.  FALSE: A major fault level Controller error has not occurred.  TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  TRUE: A minor fault level Controller error has not occurred.  TRUE: A minor fault level Controller error has not occurred.  TRUE: A minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has not occurred.  FALSE: An observation level Controller error has not occurred.	15	Master-detected error: This bit indicates whether the master detected a Controller error in the Unit/slave for the error status of the Controller error. These bits show Unit errors. They will not change to TRUE for errors in Ether-Net/IP communications.
(Valid for _CJB_U_ErrSta and _EC_SIvErrTbl.)  14		TRUE: The master detected a Controller error.
axis, or axes group) that are lower than the event source (i.e., for a function module).  TRUE: A Controller error has occurred at a lower level.  FALSE: A Controller error has not occurred at a lower level. (Valid for _CJB_ErrSta, _MC_ErrSta, and _EC_ErrSta.)  8 to 13  Reserved.  This bit indicates whether a major fault level Controller error has occurred.  TRUE: A major fault level Controller error has not occurred.  FALSE: A major fault level Controller error has not occurred.  TRUE: A partial fault level Controller error has occurred.  TRUE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has not occurred.  TRUE: An observation level Controller error has not occurred.  FALSE: An observation level Controller error has not occurred.		
FALSE: A Controller error has not occurred at a lower level. (Valid for _CJB_ErrSta, _MC_ErrSta, and _EC_ErrSta.)  8 to 13 Reserved.  7 This bit indicates whether a major fault level Controller error has occurred. TRUE: A major fault level Controller error has occurred. FALSE: A major fault level Controller error has not occurred.  6 This bit indicates whether a partial fault level Controller error has occurred. TRUE: A partial fault level Controller error has occurred. FALSE: A partial fault level Controller error has not occurred.  5 This bit indicates whether a minor fault level Controller error has occurred. TRUE: A minor fault level Controller error has occurred. FALSE: A minor fault level Controller error has not occurred.  4 This bit indicates whether an observation level Controller error has occurred. TRUE: An observation level Controller error has not occurred. FALSE: An observation level Controller error has not occurred. FALSE: An observation level Controller error has not occurred.	14	Collective slave error status: This bit indicates if a Controller error was detected for levels (e.g., a Unit, slave, axis, or axes group) that are lower than the event source (i.e., for a function module).
(Valid for _CJB_ErrSta, _MC_ErrSta, and _EC_ErrSta.)  8 to 13 Reserved.  7 This bit indicates whether a major fault level Controller error has occurred.  TRUE: A major fault level Controller error has not occurred.  FALSE: A major fault level Controller error has not occurred.  6 This bit indicates whether a partial fault level Controller error has occurred.  TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  5 This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  4 This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has not occurred.  FALSE: An observation level Controller error has not occurred.		TRUE: A Controller error has occurred at a lower level.
This bit indicates whether a major fault level Controller error has occurred.  TRUE: A major fault level Controller error has occurred.  FALSE: A major fault level Controller error has not occurred.  This bit indicates whether a partial fault level Controller error has occurred.  TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		
TRUE: A major fault level Controller error has occurred.  FALSE: A major fault level Controller error has not occurred.  This bit indicates whether a partial fault level Controller error has occurred.  TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.	8 to 13	Reserved.
FALSE: A major fault level Controller error has not occurred.  This bit indicates whether a partial fault level Controller error has occurred.  TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.	7	This bit indicates whether a major fault level Controller error has occurred.
This bit indicates whether a partial fault level Controller error has occurred.  TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		TRUE: A major fault level Controller error has occurred.
TRUE: A partial fault level Controller error has occurred.  FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		FALSE: A major fault level Controller error has not occurred.
FALSE: A partial fault level Controller error has not occurred.  This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.	6	This bit indicates whether a partial fault level Controller error has occurred.
This bit indicates whether a minor fault level Controller error has occurred.  TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		TRUE: A partial fault level Controller error has occurred.
TRUE: A minor fault level Controller error has occurred.  FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		FALSE: A partial fault level Controller error has not occurred.
FALSE: A minor fault level Controller error has not occurred.  This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.	5	This bit indicates whether a minor fault level Controller error has occurred.
This bit indicates whether an observation level Controller error has occurred.  TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		TRUE: A minor fault level Controller error has occurred.
TRUE: An observation level Controller error has occurred.  FALSE: An observation level Controller error has not occurred.		FALSE: A minor fault level Controller error has not occurred.
FALSE: An observation level Controller error has not occurred.	4	This bit indicates whether an observation level Controller error has occurred.
		TRUE: An observation level Controller error has occurred.
		FALSE: An observation level Controller error has not occurred.
U to 3 Heserved.	0 to 3	Reserved.

Note Bits 14 and 15 are never TRUE for the built-in EtherNet/IP port.



### **Precautions for Correct Use**

Do not use \_ErrSta (Controller Error Status), \_CJB\_ErrSta (I/O Bus Error Status), \_CJB\_MstrErrSta (I/O Bus Master Error Status), and \_CJB\_UnitErrSta (I/O Bus Unit Error Status) in the user program. There may be a delay in updating them and concurrency problems may occur with function module error status. Use this variable only to access status through communications from an external device.

## • Functional Classification: EtherNet/IP Communications Status

You can check the status of the built-in EtherNet/IP port (e.g., communications status).

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_EtnOnlineSta	Online	NX-series CPU Units: Indicates that the built-in EtherNet/IP port's communications can be used via the communications port 1 (that is, the link is ON, IP address is defined, and there are no errors).	BOOL	TRUE or FALSE	page 3-49
		TRUE: The built-in EtherNet/IP port's communications can be used.			
		FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.			
		NJ-series CPU Units: Indicates that the built-in EtherNet/IP port's communications can be used via the communications port (that is, the link is ON, IP address is defined, and there are no errors).			
		TRUE: The built-in EtherNet/IP port's communications can be used.			
		FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.			
_EIP1_EtnOnlineSta	Port1 Online	Indicates that the built-in EtherNet/IP port's communications can be used via the communications port 1 (that is, the link is ON, IP address is defined, and there are no errors).  TRUE: The built-in EtherNet/IP port's communications can be used.	BOOL	TRUE or FALSE	page 3-50
		FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_EtnOnlineSta	Port2 Online	Indicates that the built-in EtherNet/IP port's communications can be used via the communications port 2 (that is, the link is ON, IP address is defined, and there are no errors).	BOOL	TRUE or FALSE	page 3-50
		TRUE: The built-in EtherNet/IP port's communications can be used.			
		FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_TDLinkRunSta	Tag Data Link Communications Status	NX-series CPU Units: Indicates that at least one connection is in normal operation in CIP communications 1.  TRUE: Normal operation	BOOL	TRUE or FALSE	page 3-50
		FALSE: Other than the above.  NJ-series CPU Units: Indicates that at least one connection is in normal operation.  TRUE: Normal operation			
FID4 TDI int Dun Oto	CID	FALSE: Other than the above.	DOOL	TDUE	7777 0 50
_EIP1_TDLinkRunSta	CIP Communications1 Tag Data Link Communications Status	Indicates that at least one connection is in normal operation in CIP communications  1.  TRUE: Normal operation	BOOL	TRUE or FALSE	page 3-50
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_TDLinkRunSta	CIP Communications2 Tag Data Link Com-	Indicates that at least one connection is in normal operation in CIP communications 2.	BOOL	TRUE or FALSE	page 3-51
	munications Status	TRUE: Normal operation			
		FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP_TDLinkAllRunS ta	All Tag Data Link Communications Sta- tus	NX-series CPU Units: Indicates that all tag data links are communicating in CIP communications 1.	BOOL	TRUE or FALSE	page 3-51
		TRUE: Tag data links are communicating in all connections as the originator.			
		FALSE: An error occurred in at least one connection.			
		NJ-series CPU Units: Indicates that all tag data links are communicating.			
		TRUE: Tag data links are communicating in all connections as the originator.			
		FALSE: An error occurred in at least one connection.			
_EIP1_TDLinkAllRun Sta	CIP Communications1 All	Indicates that all tag data links are communicating in CIP communications 1.	BOOL	TRUE or FALSE	page 3-51
	Tag Data Link Com- munications Status	TRUE: Tag data links are communicating in all connections as the originator.			
		FALSE: An error occurred in at least one connection.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_TDLinkAllRun Sta	CIP Communications2 All	Indicates that all tag data links are com-	BOOL	TRUE or FALSE	page 3-51
Jia Jia	Tag Data Link Communications Status	municating in CIP communications 2.  TRUE: Tag data links are communicating in all connections as the originator.		I ALOL	
		FALSE: An error occurred in at least one connection.			
_		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_RegTargetSta [255]	Registered Target Node Information	NX-series CPU Units: Gives a list of nodes for which built-in EtherNet/IP connections are registered for CIP communications 1.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-52
		This variable is valid only when the built-in EtherNet/IP port is the originator.			
		Array[x] is TRUE: The connection to the node with a target node ID of x is registered.			
		Array[x] is FALSE:  The connection to the node with a target node ID of x is not registered.			
		NJ-series CPU Units: Gives a list of nodes for which built-in EtherNet/IP connections are registered.			
		This variable is valid only when the built-in EtherNet/IP port is the originator.			
		Array[x] is TRUE:  The connection to the node with a target node ID of x is registered.			
		Array[x] is FALSE:  The connection to the node with a target node ID of x is not registered.			
_EIP1_RegTargetSta [255]	CIP Communications1 Registered Target	Gives a list of nodes for which built-in EtherNet/IP connections are registered for CIP communications 1.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-52
	Node Information	This variable is valid only when the built-in EtherNet/IP port is the originator.			
		Array[x] is TRUE: The connection to the node with a target node ID of x is registered.			
		Array[x] is FALSE:  The connection to the node with a target node ID of x is not registered.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_RegTargetSta [255]	CIP Communications2 Registered Target	Gives a list of nodes for which built-in EtherNet/IP connections are registered for CIP communications 2.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-52
	Node Information	This variable is valid only when the built-in EtherNet/IP port is the originator.			
		Array[x] is TRUE: The connection to the node with a target node ID of x is registered.			
		Array[x] is FALSE:  The connection to the node with a target node ID of x is not registered.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_EstbTargetSta [255]	Normal Target Node Information	NX-series CPU Units: Gives a list of nodes that have normally established EtherNet/IP connections for CIP communications 1.  Array[x] is TRUE:  The connection to the node with a target	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-53
		node ID of x was established normally.  Array[x] is FALSE: The connection to the node with a target node ID of x was not established, or an error occurred.			
		NJ-series CPU Units: Gives a list of nodes that have normally established Ether-Net/IP connections.			
		Array[x] is TRUE: The connection to the node with a target node ID of x was established normally.			
		Array[x] is FALSE: The connection to the node with a target node ID of x was not established, or an error occurred.			
_EIP1_EstbTargetSta [255]	CIP Communications1 Normal Target Node Information	Gives a list of nodes that have normally established EtherNet/IP connections for CIP communications 1.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-53
		Array[x] is TRUE: The connection to the node with a target node ID of x was established normally.			
		Array[x] is FALSE: The connection to the node with a target node ID of x was not established, or an error occurred.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_EstbTargetSta [255]	CIP Communications2 Normal Target Node	Gives a list of nodes that have normally established EtherNet/IP connections for CIP communications 2.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-53
	Information	Array[x] is TRUE: The connection to the node with a target node ID of x was established normally.			
		Array[x] is FALSE: The connection to the node with a target node ID of x was not established, or an error occurred.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_TargetPLCMod eSta [255]	Target PLC Operating Mode	NX-series CPU Units: Shows the operating status of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP port as the originator.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-54
		The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.			
		Array[x] is TRUE: This is the operating state of the target Controller with a node address of x.			
		Array[x] is FALSE: Other than the above.			
		NJ-series CPU Units: Shows the operating status of the target node Controllers that are connected with the built-in Ether-Net/IP port as the originator.			
		The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.			
		Array[x] is TRUE: This is the operating state of the target Controller with a node address of x.			
		Array[x] is FALSE: Other than the above.			
_EIP1_TargetPLCMo deSta [255]	CIP Communications1 Target PLC Operat- ing Mode	Shows the operating status of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP port as the originator.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-54
		The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.			
		Array[x] is TRUE: This is the operating state of the target Controller with a node address of x.			
		Array[x] is FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP2_TargetPLCMo deSta [255]	CIP Communications2 Target PLC Operat- ing Mode	Shows the operating status of the target node Controllers that are connected for CIP communications 2, with the built-in EtherNet/IP port as the originator.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-54
		The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.			
		Array[x] is TRUE:  This is the operating state of the target  Controller with a node address of x.			
		Array[x] is FALSE: Other than the above.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP_TargetPLCErr [255]	Target PLC Error Information	NX-series CPU Units: Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-55
		Array[x] is TRUE: A fatal or non-fatal error occurred in the target Controller with a target node ID of x.			
		Array[x] is FALSE: Other than the above.			
		NJ-series CPU Units: Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected with the built-in EtherNet/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.			
		Array[x] is TRUE: A fatal or non-fatal error occurred in the target Controller with a target node ID of x.			
		Array[x] is FALSE: Other than the above.			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP1_TargetPLCErr [255]	CIP Communications1 Target PLC Error Information	Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected for CIP communications 1, with the built-in Ether-Net/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.  Array[x] is TRUE:  A fatal or non-fatal error occurred in the target Controller with a target node ID of x.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-55
		Array[x] is FALSE: Other than the above.  Note You can use this system-defined vari-			
_EIP2_TargetPLCErr [255]	CIP Communications2 Target PLC Error Information	able only for NX-series CPU Units.  Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected for CIP communications 2, with the built-in Ether-Net/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.  Array[x] is TRUE:  A fatal or non-fatal error occurred in the target Controller with a target node ID of x.  Array[x] is FALSE:	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-55
		Other than the above.  Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP_TargetNodeErr [255]	Target Node Error Information	NX-series CPU Units: Indicates that the connection for the Registered Target Node Information for CIP communications 1 was not established or that an error occurred in the target Controller.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-56
		The array elements are valid only when the Registered Target Node Information is TRUE.  Array[x] is TRUE:  A connection was not normally established with the target node for a target node ID of x  (the Registered Target Node Informa-			
		tion is TRUE and the Normal Target Node Information is FALSE), or a connection was established with the target node but an error occurred in the target Controller.			
		Array[x] is FALSE:  The target node is not registered for a target node ID of x (the Registered Target Node Information is FALSE), or a connection was normally established with the target node (the Registered Target Node Information is TRUE and the Normal Target Node Information is TRUE).  An error occurred in the target Controller (the Target PLC Error Information is TRUE).			
		NJ-series CPU Units: Indicates that the connection for the Registered Target Node Information was not established or that an error occurred in the target Controller.			
		The array elements are valid only when the Registered Target Node Information is TRUE.			
		Array[x] is TRUE: A connection was not normally established with the target node for a target node ID of x (the Registered Target Node Information is TRUE and the Normal Target Node Information is FALSE), or a connection was established with the target node but an error occurred in the target Controller.			
		Array[x] is FALSE: The target node is not registered for a target node ID of x (the Registered Target Node Information is FALSE), or a connection was normally established with the target node (the Registered Target Node Information is TRUE and the Normal Target Node Information is TRUE). An error occurred in the target Controller (the Target PLC Error Information is TRUE).			

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP1_TargetNodeEr r [255]	CIP Communications1 Target Node Error Information	Indicates that the connection for the Registered Target Node Information for CIP communications 1 was not established or that an error occurred in the target Controller.  The array elements are valid only when the Registered Target Node Information is TRUE.  Array[x] is TRUE:  A connection was not normally established with the target node for a target node ID of x (the Registered Target Node Information is TRUE and the Normal Target Node Information is FALSE), or a connection was established with the target node but an error occurred in the target node is not registered for a target node ID of x (the Registered Target Node Information is FALSE), or a connection was normally established with the target node (the Registered Target Node Information is TALSE) and the Normal Target Node Information is TRUE and the Normal Target Node Information is TRUE and the Normal Target PLC Error Information is TRUE).	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-56
_EIP2_TargetNodeEr r [255]	CIP Communications2 Target Node Error Information	Note You can use this system-defined variable only for NX-series CPU Units.  Indicates that the connection for the Registered Target Node Information for CIP communications 2 was not established or that an error occurred in the target Controller.  The array elements are valid only when the Registered Target Node Information is TRUE.  Array[x] is TRUE:  A connection was not normally established with the target node for a target node ID of x (the Registered Target Node Information is TRUE and the Normal Target Node Information is FALSE), or a connection was established with the target node but an error occurred in the target Controller.  Array[x] is FALSE:  The target node is not registered for a target node ID of x (the Registered Target Node Information is FALSE), or a connection was normally established with the target node (the Registered Target Node Information is TRUE and the Normal Target Node Information is TRUE and the Normal Target Node Information is TRUE.)  An error occurred in the target Controller (the Target PLC Error Information is TRUE).  Note You can use this system-defined variable only for NX-series CPU Units.	ARRAY [0255] OF BOOL	TRUE or FALSE	page 3-57

Variable name	Meaning	Function	Data type	Range of values	Reference
_EIP_NTPResult	NTP Operation Information	Use the GetNTPStatus instruction to read the NTP operation information from the user program. Direct access is not possible.	_sNTP_ RESULT		page 3-57
.ExecTime	NTP Last Operation Time	Gives the last time that NTP processing ended normally. The time that was obtained from the NTP server is stored when the time is obtained normally. The time is not stored if it is not obtained from the NTP server normally.  Note Do not use this variable in the user program. There may be a delay in updating it. Use this variable only to access status through communications from an external device.	DATE_AND_ TIME	Depends on data type.	page 3-57
.ExecNormal	NTP Operation Result	TRUE: Indicates an NTP normal end.  FALSE: Indicates that NTP operation ended in an error or has not been executed even once.  Note Do not use this variable in the user program. There may be a delay in updating it. Use this variable only to access status through communications from an external device.	BOOL	TRUE or FALSE	page 3-57



#### **Additional Information**

#### **Communications Status with Target Node**

The communications status with the target node of an NJ/NX-series Controller is shown by the combination of the values of four system-defined variables.

- \_EIP\_RegTargetSta (Registered Target Node Information)
- \_EIP\_EstbTargetSta (Normal Target Node Information)
- \_EIP\_TargetPLCErr (Target PLC Error Information)
- \_EIP\_TargetNodeErr (Target Node Error Information)

Value of _EIP_RegTarget Sta	Value of _EIP_EstbTarget Sta	Value of _EIP_TargetPL CErr	Value of _EIP_TargetNo deErr	Communications status with target node
TRUE	TRUE	FALSE	FALSE	A connection with the target node was established normally and there is no error in the target PLC.
		TRUE	TRUE	A connection with the target node was established but there is an error in the target PLC.
	FALSE	Disabled	TRUE	A connection with the target node was not established normally.
FALSE	Disabled	Disabled	Disabled	The information is not valid because the target node is not registered.

For the NX-series Controller, the communications status of CIP communications 1 and CIP communications 2 is shown by the combination of the values of four system-defined variables in the same way as shown in the above table.

- · CIP Communications 1
  - \_EIP1\_RegTargetSta (CIP Communications1 Registered Target Node Information)
  - \_EIP1\_EstbTargetSta (CIP Communications1 Normal Target Node Information)
  - EIP1 TargetPLCErr (CIP Communications1 Target PLC Error Information)
  - \_EIP\_TargetNodeErr (Target Node Error Information)
- CIP Communications 2
  - \_EIP2\_RegTargetSta (CIP Communications2 Registered Target Node Information)
  - \_EIP2\_EstbTargetSta (CIP Communications2 Normal Target Node Information)
  - \_EIP2\_TargetPLCErr (CIP Communications2 Target PLC Error Information)
  - EIP TargetNodeErr (Target Node Error Information)

## • Functional Classification: EtherNet/IP Communications Switches

You can start and stop tag data links.

Variable name	le name Meaning Function		Data type	Range of val- ues	Reference
_EIP_TDLinkStartCm d	Tag Data Link Com- munications Start Switch	NX-series CPU Units: Change this variable to TRUE to start tag data links for CIP communications 1.	BOOL	TRUE or FALSE	page 3-58
		It automatically changes back to FALSE after tag data link operation starts.			
		NJ-series CPU Units: Change this variable to TRUE to start tag data links.			
		It automatically changes back to FALSE after tag data link operation starts.			
		Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.			
_EIP1_TDLinkStartC md	CIP Communications1	Change this variable to TRUE to start tag data links for CIP communications 1.	BOOL	TRUE or FALSE	page 3-58
	Tag Data Link Com- munications Start Switch	It automatically changes back to FALSE after tag data link operation starts.			
	Switch	Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP2_TDLinkStartC md	CIP Communications2	Change this variable to TRUE to start tag data links for CIP communications 2.	BOOL	TRUE or FALSE	page 3-58
	Tag Data Link Com- munications Start Switch	It automatically changes back to FALSE after tag data link operation starts.			
		Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.			
		Note You can use this system-defined variable only for NX-series CPU Units.			
_EIP_TDLinkStopCm d	Tag Data Link Com- munications Stop Switch	NX-series CPU Units: Change this variable to TRUE to stop tag data links for CIP communications 1.	BOOL	TRUE or FALSE	page 3-59
		It automatically changes back to FALSE after tag data link operation stops.			
		NJ-series CPU Units: Change this variable to TRUE to stop tag data links.			
		It automatically changes back to FALSE after tag data link operation stops.			
		Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.			
_EIP1_TDLinkStopC md	CIP Communications1	Change this variable to TRUE to stop tag data links for CIP communications 1.	BOOL	TRUE or FALSE	page 3-59
	Tag Data Link Com- munications Stop Switch	It automatically changes back to FALSE after tag data link operation stops.			
		Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.			
		Note You can use this system-defined variable only for NX-series CPU Units.			

Variable name	Meaning	Function	Data type	Range of val- ues	Reference
_EIP2_TDLinkStopC md	CIP Communications2 Tag Data Link Com- munications Stop Switch	Change this variable to TRUE to stop tag data links for CIP communications 2.  It automatically changes back to FALSE after tag data link operation stops.  Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.	BOOL	TRUE or FALSE	page 3-59
		Note You can use this system-defined variable only for NX-series CPU Units.			



## **Additional Information**

There is a system-defined variable for communications that is related to FINS/TCP communications.

Variable name	Meaning	Description	Data type	Opera- tion/refresh timing	Range of values	Default	R/W access	Related instruc- tions
FINSTCP- ConnSta	FINS/TCP Connection Status	This is the FINS/TCP connection status.	WORD	Anytime	16#0000 to 16#FFFF	0	R	None

# 3-3 Specifications for Individual Systemdefined Variables

The specifications for each system-defined variable are given as described below.

Variable name	This is the system-defined variable name. The prefix gives the category name.			Members	The member names are given for structure variables.	
Meaning	This is the meaning of the variable.			Global/local	Global: Global variable, Local: Local variable	
Function	The function of the	e variable is descrit	oed.			
Data type	The data type of the variable is given.			Range of values	The range of values that the variable can take is given.	
R/W access	R: Read only, RW: Read/write	Retained	The Retain attri- bute of the vari- able is given.	Network Publish	The Network Publish attribute of the variable is given.	
Usage in user program	Whether you	Related instruc-	The instructions the	hat are related to th	e variable are given.	
	can use the variable directly in the user program is specified.	tions	If you cannot use the variable directly in the user program, the instructions that access the variable are given.			

### • Functional Classification: EtherNet/IP Communications Errors

Variable name	_EIP_ErrSta							
Meaning	Built-in EtherNet/I	uilt-in EtherNet/IP Error Global/local Global						
Function	This is the error status variable for the built-in EtherNet/IP port.							
	NX-series CPU U	nits: Represents the	e collective status of	f the following error	flags.			
	• _EIP1_PortErr (	Communications P	ort1 Error)					
	• _EIP2_PortErr (	Communications P	ort2 Error)					
	• _EIP1_CipErr (0	CIP Communication	s1 Error)					
	• _EIP2_CipErr (0	CIP Communication	s2 Error)					
	_EIP_TcpAppEi	rr (TCP Application	Communications E	rror)				
	NJ-series CPU Ur	nits: Represents the	collective status of	the following error	flags.			
	• _EIP_PortErr (C	communications Po	rt Error)					
	• _EIP_CipErr (CI	P Communications	Error)					
	_EIP_TcpAppEi	rr (TCP Application	Communications E	rror)				
	Note Refer to Me	anings of Error Sta	tus Bits for the mea	nings of the error s	tatus bits.			
Data type	WORD			Range of values	16#0000 to 16#00F0			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions	You can access th instruction.	is variable from the	user program with the following			
			GetEIPError					

Variable name	_EIP_PortErr								
Meaning	Communications	Port Error		Global/local	Global				
Function	This is the error st	This is the error status variable for the communications port.							
	NX-series CPU U	NX-series CPU Units: Represents the collective status of the following error flags.							
	• _EIP1_MacAdri	_EIP1_MacAdrErr (Port1 MAC Address Error)							
	• _EIP1_LanHwE	Frr (Port1 Communic	cations Controller E	rror)					
		_EIP1_EtnCfgErr (Port1 Basic Ethernet Setting Error)							
		_EIP1_IPAdrCfgErr (Port1 IP Address Setting Error)							
		` `	ess Duplication Erro	or)					
	,	r (Port1 BOOTP Se	,						
		Frr (DNS Setting Err	,						
		_EIP_DNSSrvErr (DNS Server Connection Error) _EIP_IPRTblErr (IP Route Table Error)							
		•	collective status of	f the following error	flage				
		rr (MAC Address Er		the following entor	nags.				
		r (Communications	,						
		(Basic Ethernet Se	•						
	• _EIP_IPAdrCfgl	Err (IP Address Sett	ting Error)						
	• _EIP_IPAdrDup	Err (IP Address Du	plication Error)						
	• _EIP_BootpErr	(BOOTP Server Er	ror)						
	• _EIP_DNSSrvE	rr (DNS Server Cor	nnection Error)						
	• _EIP_IPRTblEri	(IP Route Table Er	ror)						
	Note If a Link OF	F Detected or Buil	t-in EtherNet/IP Pro	ocessing Error occu	irs, it is recorded in the event log and				
	then corres bits.	ponding bit turns O	N. Refer to <i>Meaning</i>	gs of Error Status B	its for the meanings of the error status				
Detectors				Daniel of column	10//0000 1: 10//0050				
Data type	WORD	5	I	Range of values	16#0000 to 16#00F0				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instruc- tions	You can access the instruction.	is variable from the	user program with the following				
			GetEIPError						

Variable name	_EIP1_PortErr	_EIP1_PortErr							
Meaning	Communications I	Communications Port1 Error Global/local Global							
Function	This is the error st	atus variable for th	e communications	port 1.					
	It represents the c	ollective status of t	he following error fl	lags.					
	• _EIP1_MacAdrl	Err (Port1 MAC Add	dress Error)						
	• _EIP1_LanHwE	rr (Port1 Communi	cations Controller E	Error)					
	• _EIP1_EtnCfgE	rr (Port1 Basic Ethe	ernet Setting Error)	1					
	• _EIP1_IPAdrCfg	gErr (Port1 IP Addre	ess Setting Error)						
		,	ess Duplication Er	ror)					
		r (Port1 BOOTP Se							
	• _EIP_DNSCfgErr (DNS Setting Error)								
	• _EIP_DNSSrvErr (DNS Server Connection Error)								
	• _EIP_IPRTblEri	• _EIP_IPRTblErr (IP Route Table Error)							
	then corres			•	urs, it is recorded in the event log and Bits for the meanings of the error state.				
	bits.								
	Note You can use	e this system-define	ed variable only for	NX-series CPU Un	its.				
Data type	WORD			Range of values	16#0000 to 16#00F0				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions	You can access to instruction.	his variable from the	e user program with the following				
			GetEIPError						
/ariable name	_EIP2_PortErr								
Meaning	Communications I			Global/local	Global				
Function	This is the error st	atus variable for th	e communications	port 2.					
	It represents the collective status of the following error flags.								

Variable name	_EIP2_PortErr				_				
Meaning	Communications I	Communications Port2 Error Global/local Global							
Function	This is the error st	This is the error status variable for the communications port 2.							
	It represents the c	It represents the collective status of the following error flags.							
	• _EIP2_MacAdrl	• _EIP2_MacAdrErr (Port2 MAC Address Error)							
	• _EIP2_LanHwE	• _EIP2_LanHwErr (Port2 Communications Controller Error)							
	• _EIP2_EtnCfgE	_EIP2_EtnCfgErr (Port2 Basic Ethernet Setting Error)							
	• _EIP2_IPAdrCfg	<i>Err</i> (Port2 IP Addre	ess Setting Error)						
	• _EIP2_IPAdrDu	pErr (Port2 IP Addr	ess Duplication Erro	or)					
	,	r (Port2 BOOTP Se	,						
		rr (DNS Setting Err	,						
		rr (DNS Server Cor	,						
	• _EIP_IPRTblErr	(IP Route Table Er	ror)						
				•	urs, it is recorded in the event log and lits for the meanings of the error status				
	Note You can use	this system-define	ed variable only for I	NX-series CPU Uni	its.				
Data type	WORD			Range of values	16#0000 to 16#00F0				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	le. Related instructions  You can access this variable from the user program with the following instruction.							
			GetEIPError						

Variable name	_EIP_CipErr							
Meaning	CIP Communication	ons Error		Global/local	Global			
Function	This is the error st	This is the error status variable for CIP communications.						
	NX-series CPU U	nits: Represents the	e collective status o	f the following error	flags.			
	• _EIP1_IdentityE	Err (CIP Communica	ations1 Identity Erro	or)				
	• _EIP1_TDLinkC	fgErr (CIP Commu	nications1 Tag Data	a Link Setting Error)				
		<i>pnErr</i> (CIP Commu	J		,			
	• _EIP1_TDLinkE	Err (CIP Communica	itions1 Tag Data Lir	nk Communications	Error)			
		Err (CIP Communica	· ·	,				
	• _EIP1_MultiSw	ONErr (CIP Commu	ınications1 Multiple	Switches ON Error	7)			
	NJ-series CPU Ur	nits: Represents the	collective status of	f the following error	flags.			
	• _EIP_IdentityEr	r (Identity Error)						
		gErr (Tag Data Link	0 ,					
	,	onErr (Tag Data Linl		,				
		r (Tag Data Link Co		r)				
		r (Tag Name Resolu	,					
	• _EIP_MultiSwO	nErr (Multiple Switch	enes ON Error)					
			,	,	g and this variable changes to TRUE.			
		eanings of Error Sta	tus Bits for the mea		T			
Data type	WORD		T	Range of values	16#0000 to 16#00F0			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Possible. Related instructions You can access this variable from the user program with the following instruction.						
			GetEIPError					

Variable name	_EIP1_CipErr							
Meaning	CIP Communications1 Error Global/local Global							
Function	This is the error st	atus variable for CI	P communications	1.				
	It represents the c	ollective status of the	ne following error fla	ags.				
	• _EIP1_IdentityE	Err (CIP Communica	ations1 Identity Erro	r)				
	• _EIP1_TDLinkC	fgErr (CIP Commu	nications1 Tag Data	Link Setting Error)				
	• _EIP1_TDLinkC	<i>pnErr</i> (CIP Commu	inications1 Tag Data	a Link Connection I	Failed)			
	• _EIP1_TDLinkE	rr (CIP Communica	itions1 Tag Data Lin	nk Communications	Error)			
		•	itions1 Tag Name R	,				
	• _EIP1_MultiSw	ONErr (CIP Commu	inications1 Multiple	Switches ON Error	·)			
			r occurs, it is record tus Bits for the mea		g and this variable changes to TRUE. tatus bits.			
	Note You can use	e this system-define	ed variable only for I	NX-series CPU Uni	ts.			
Data type	WORD			Range of values	16#0000 to 16#00F0			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions  You can access this variable from the user program with the following instruction.						
			GetEIPError					

Variable name	_EIP2_CipErr						
Meaning	CIP Communica	ommunications2 Error Global/local Global					
Function	This is the error	status variable fo	or CIP communic	ations 2.	•		
	It represents the	e collective status	of the following	error flags.			
	• _EIP2_Identit	yErr (CIP Comm	unications2 Ident	ity Error)			
	• _EIP2_TDLin	<i>kCfgErr</i> (CIP Cor	nmunications2 Ta	ag Data Link Setting Err	or)		
	• _EIP2_TDLin	kOpnErr (CIP Co	mmunications2 7	Tag Data Link Connection	n Failed)		
	• _EIP2_TDLin	kErr (CIP Commi	unications2 Tag [	Data Link Communicatio	ns Error)		
	• _EIP2_TagAd	<i>IrErr</i> (CIP Commu	unications2 Tag N	Name Resolution Error)			
	• _EIP2_MultiS	wONErr (CIP Co	mmunications2 N	Multiple Switches ON Er	ror)		
	<b>Note</b> If a Tag Name Resolution Error occurs, it is recorded in the event log and this variable changes to TRUE. Refer to <i>Meanings of Error Status Bits</i> for the meanings of the error status bits.						
	Note You can u	use this system-d	efined variable o	nly for NX-series CPU L	Jnits.		
Data type	WORD			Range of values	16#0000 to 16#00F0		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible. Related You can access this variable from the user program with the following instruc-						
	instructions • GetEIPError						

Variable name	_EIP_TcpAppErr							
Meaning	TCP Application Communications Error Global/local Global							
Function	This is the error	This is the error status variable for TCP application communications.						
	It represents the	e collective status	s of the following	error flags.				
	• _EIP_TcpApp	CfgErr (TCP/IP	Setting Error)					
	• _EIP_NTPSr	Err (NTP Server	Connection Erro	r)				
	Note Refer to I	Meanings of Erro	r Status Bits for t	ne meanings of the error s	status bits.			
Data type	WORD			Range of values	16#0000 to 16#00F0			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program								
		instructions	GetEIPError					

Variable name	_EIP_MacAdrErr							
Meaning	MAC Address Error Global/local Global							
Function		NX-series CPU Units: Indicates that an error occurred when the MAC address was read on the communications port 1 at startup.						
	TRUE: Error							
	FALSE: Normal							
	NJ-series CPU Units: Indicates that an error occurred when the MAC address was read at startup.  TRUE: Error							
	FALSE: Normal							
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_MacAdrErr						
Meaning	Port1 MAC Add	ress Error		Global/local	Global		
Function	Indicates that an error occurred when the MAC address was read on the communications port 1 at startup.  TRUE: Error  FALSE: Normal						
Data type		use this system-c	delined variable (	only for NX-series CPU	TRUE or FALSE		
Data type  R/W access	BOOL R	Retained	Not retained.	Range of values  Network Publish	Published.		
Usage in user program	Possible.	Related instructions		Network Publish	Fublished.		
Variable name	_EIP2_MacAdr	Err					
Meaning	Port2 MAC Add	ress Error		Global/local	Global		
Function	TRUE: Error FALSE: Normal			ddress was read on the	e communications port 2 at startup.  Units.		
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					
Variable name	_EIP_LanHwEr	<u>,</u>					
Meaning		s Controller Erro	r	Global/local	Global		
Function	NX-series CPU Units: Indicates that a communications controller failure occurred on the communications port 1.  TRUE: Failure  FALSE: Normal  NJ-series CPU Units: Indicates that a communications controller failure occurred.  TRUE: Failure  FALSE: Normal						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					
Va dala da	L EID4 1. 11 =	•					
Variable name	_EIP1_LanHwE		N. E.	Clabal#s ssl	Clahal		
Meaning		cations Controlle		Global/local	Global		
Function	Indicates that a communications controller failure occurred on the communications port 1.  TRUE: Failure  FALSE: Normal  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP2_LanHwE	_EIP2_LanHwErr						
Meaning	Port2 Commun	ications Controlle	er Error	Global/local	Global			
Function	Indicates that a	Indicates that a communications controller failure occurred on the communications port 2.						
	TRUE: Failure	TRUE: Failure						
	FALSE: Norma	FALSE: Normal						
	Note You can	use this system-	defined variable	only for NX-series CPU	Units.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP_EtnCfgErr							
Meaning	Basic Ethernet	Setting Error		Global/local	Global			
Function		NX-series CPU Units: Indicates that the Ethernet communications speed setting (Speed/Duplex) for the communications port 1 is incorrect. Or, a read operation failed.						
	TRUE: Setting i	ncorrect or read	failed					
	FALSE: Normal							
	NJ-series CPU read operation		hat the Ethernet	communications speed se	etting (Speed/Duplex) is incorrect. Or, a			
	TRUE: Setting i	ncorrect or read	failed					
	FALSE: Normal							
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_EtnCfgErr							
Meaning	Port1 Basic Eth	Port1 Basic Ethernet Setting Error Global/local Global						
Function		Indicates that the Ethernet communications speed setting (Speed/Duplex) for the communications port 1 is incorrect. Or, a read operation failed.						
	TRUE: Setting i	ncorrect or read	failed					
	FALSE: Normal							
	Note You can u	use this system-o	defined variable o	only for NX-series CPU Ur	nits.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP2_EtnCfgE	_EIP2_EtnCfgErr					
Meaning	Port2 Basic Eth	ernet Setting Err	or	Global/local	Global		
Function	Indicates that the Ethernet communications speed setting (Speed/Duplex) for the communications port 2 is incorrect. Or, a read operation failed.  TRUE: Setting incorrect or read failed FALSE: Normal  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained. Network Publish Published.				
Usage in user program	Possible.	Related instructions					

Variable name	_EIP_IPAdrCfgErr						
Meaning	IP Address Set	ting Error		Global/local	Global		
Function	NX-series CPU Units: Indicates the IP address setting errors for the communications port 1.  TRUE: • There is an illegal IP address setting. • A read operation failed. • The IP address obtained from the BOOTP server is inconsistent. • The DNS settings are not correct.  FALSE: Normal  NJ-series CPU Units: Indicates the IP address setting errors.  TRUE: • There is an illegal IP address setting. • A read operation failed. • The IP address obtained from the BOOTP server is inconsistent. • The DNS settings are not correct.  FALSE: Normal						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					
Variable name	EIP1 IPAdrCf	nFrr					
Meaning	Port1 IP Addres			Global/local	Global		
Function	Indicates the IP address setting errors for the communications port 1.  TRUE: • There is an illegal IP address setting. • A read operation failed. • The IP address obtained from the BOOTP server is inconsistent. • The DNS settings are not correct.  FALSE: Normal  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					
Variable name	_EIP2_IPAdrCf						
Meaning	Port2 IP Addres			Global/local	Global		
Function	Indicates the IP address setting errors for the communications port 2.  TRUE: • There is an illegal IP address setting. • A read operation failed. • The IP address obtained from the BOOTP server is inconsistent. • The DNS settings are not correct.  FALSE: Normal  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP_IPAdrDupErr						
Meaning	IP Address Dup	lication Error		Global/local	Global		
Function	NX-series CPU Units: Indicates that the same IP address is assigned to more than one node for the communications port 1.						
	TRUE: Duplicat	ion occurred.					
	FALSE: Other to	han the above.					
	NJ-series CPU	Units: Indicates t	hat the same IP	address is assigned to mo	re than one node.		
	TRUE: Duplicat	ion occurred.					
	FALSE: Other to	han the above.					
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Published.			
Usage in user program	Possible.	Related instructions					

Variable name	_EIP1_IPAdrDupErr							
Meaning	Port1 IP Address Duplication Error Global/local Global							
Function	Indicates that th	Indicates that the same IP address is assigned to more than one node for the communications port 1.						
	TRUE: Duplicat	ion occurred.						
	FALSE: Other than the above.							
	Note You can u	Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP2_IPAdrDu	_EIP2_IPAdrDupErr						
Meaning	Port2 IP Addres	s Duplication Err	or	Global/local	Global			
Function	Indicates that th	Indicates that the same IP address is assigned to more than one node for the communications port 2.						
	TRUE: Duplicat	TRUE: Duplication occurred.						
	FALSE: Other to	FALSE: Other than the above.						
	Note You can	use this system-d	efined variable o	nly for NX-series CPU Uni	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_DNSCfgErr							
Meaning	DNS Setting Er	ror		Global/local	Global			
Function	Indicates that th	Indicates that the DNS or hosts settings are incorrect. Or, a read operation failed.						
	TRUE: Setting i	ncorrect or read t	failed					
	FALSE: Normal							
	Note You can u	Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_BootpErr							
Meaning	BOOTP Server	Error		Global/local	Global			
Function	NX-series CPU	NX-series CPU Units: Indicates that a BOOTP server connection failure occurred on the communications port 1.						
	TRUE: There w	as a failure to cor	nnect to the BOC	TP server (timeout).				
	FALSE: The BOOTP is not enabled, or BOOTP is enabled and an IP address was normally obtained from the BOOTP server.							
	NJ-series CPU	Units: Indicates t	hat a BOOTP se	rver connection failure occ	curred.			
	TRUE: There w	as a failure to cor	nnect to the BOC	TP server (timeout).				
		OTP is not enab TP server.	led, or BOOTP is	enabled and an IP addres	ss was normally obtained from the			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_BootpEi	_EIP1_BootpErr							
Meaning	Port1 BOOTP S	Server Error		Global/local	Global				
Function	Indicates that a	Indicates that a BOOTP server connection failure occurred on the communications port 1.							
	TRUE: There w	TRUE: There was a failure to connect to the BOOTP server (timeout).							
	BOOTP server.	FALSE: The BOOTP is not enabled, or BOOTP is enabled and an IP address was normally obtained from the BOOTP server.  Note You can use this system-defined variable only for NX-series CPU Units.							
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions							

Variable name	_EIP2_BootpErr							
Meaning	Port2 BOOTP S	OOTP Server Error Global/local Global						
Function	Indicates that a	Indicates that a BOOTP server connection failure occurred on the communications port 2.						
	TRUE: There w	as a failure to co	nnect to the BOO	TP server (timeout).				
	FALSE: The BOOTP is not enabled, or BOOTP is enabled and an IP address was normally obtained from the BOOTP server.  Note You can use this system-defined variable only for NX-series CPU Units.							
Data type	BOOL	<u> </u>		Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained. Network Publish Published.					
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_IPRTblEr	_EIP_IPRTblErr						
Meaning	IP Route Table	Error Global/local Global						
Function	NX-series CPU Units: Indicates that the default gateway settings or IP router table settings are incorrect. Or, a read operation failed.							
	TRUE: Setting i	ncorrect or read	failed					
	FALSE: Normal	FALSE: Normal						
	Note For NX-series CPU Units, the host setting errors will be indicated to _EIP_DNSCfgErr.							
	NJ-series CPU	Units: Indicates t	hat the IP router	table or hosts settings are	e incorrect. Or, a read operation failed.			
	TRUE: Setting i	ncorrect or read	failed					
	FALSE: Normal							
	Note For NJ-se	eries CPU Units,	the default gatew	ay setting errors will be in	dicated to _EIP_IPAdrCfgErr.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_IdentityEr	_EIP_IdentityErr						
Meaning	Identity Error	Identity Error Global/local Global						
Function	NX-series CPU Units: Indicates that the identity information for CIP communications 1 (which you cannot overwrite) is incorrect. Or, a read operation failed.							
	TRUE: Setting i	ncorrect or read f	failed					
	FALSE: Normal							
	NJ-series CPU operation failed		hat the identity in	formation (which you can	not overwrite) is incorrect. Or, a read			
	TRUE: Setting i	ncorrect or read f	failed					
	FALSE: Normal							
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_IdentityErr							
Meaning	CIP Communic	ations1 Identity E	rror	Global/local	Global			
Function	Indicates that the identity information for CIP communications 1 (which you cannot overwrite) is incorrect. Or, a read operation failed.							
	TRUE: Setting i	ncorrect or read t	failed					
	FALSE: Normal							
	Note You can	use this system-d	lefined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

3-43

Variable name	_EIP2_IdentityErr							
Meaning	CIP Communic	CIP Communications2 Identity Error Global/local Global						
Function	Indicates that the identity information for CIP communications 2 (which you cannot overwrite) is incorrect. Or, a read operation failed.							
	TRUE: Setting i	TRUE: Setting incorrect or read failed						
	FALSE: Normal							
	Note You can u	use this system-d	lefined variable o	nly for NX-series CPU Uni	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_TDLinkCfgErr						
Meaning	Tag Data Link Setting Error Global/local Global						
Function	NX-series CPU Units: Indicates that the tag data link settings for CIP communications 1 are incorrect. Or, a read operation failed.						
	TRUE: Setting i	ncorrect or read t	failed				
	FALSE: Normal						
	NJ-series CPU	Units: Indicates t	hat the tag data I	ink settings are incorrect.	Or, a read operation failed.		
	TRUE: Setting i	ncorrect or read t	failed				
	FALSE: Normal						
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user pro-	Possible.	Related					
gram		instructions					

Variable name	_EIP1_TDLinkCfgErr							
Meaning	CIP Communic	ations1 Tag Data	tions1 Tag Data Link Setting Global/local Global					
	Error							
Function	Indicates that th	ie tag data link se	ettings for CIP co	mmunications 1 are incorr	rect. Or, a read operation failed.			
	TRUE: Setting i	ncorrect or read t	failed					
	FALSE: Normal							
	Note You can u	use this system-d	lefined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained. Network Publish Published.					
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP2_TDLinkCfgErr							
Meaning	CIP Communic Error	munications2 Tag Data Link Setting Global/local Global						
Function	Indicates that the	ne tag data link se	ettings for CIP co	mmunications 2 are incor	rect. Or, a read operation failed.			
	TRUE: Setting i	ncorrect or read	failed					
	FALSE: Norma							
	Note You can	use this system-d	lefined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained. Network Publish Published.					
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_TDLinkO	_EIP_TDLinkOpnErr							
Meaning	Tag Data Link C	Connection Failed		Global/local	Global				
Function	NX-series CPU	NX-series CPU Units: Indicates that establishing a tag data link connection for CIP communications 1 failed.							
	TRUE: Establis	hing a tag data lir	nk connection fail	ed due to one of the follow	ving causes.				
		formation registe information.	red for a target n	ode in the tag data link pa	rameters is different from the actual				
	•There	was no response	from the remote	node.					
	FALSE: Other t	nan the above.							
	NJ-series CPU	Units: Indicates t	hat establishing	a tag data link connection	failed.				
	TRUE: Establis	hing a tag data lir	nk connection fail	ed due to one of the follow	ving causes.				
		formation registe information.	red for a target n	ode in the tag data link pa	rameters is different from the actual				
	•There	was no response	from the remote	node.					
	FALSE: Other t	nan the above.							
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions			1				

Variable name	_EIP1_TDLink(	_EIP1_TDLinkOpnErr						
Meaning	CIP Communic tion Failed	ations1 Tag Data	ions1 Tag Data Link Connec- Global/local Global					
Function	Indicates that es	stablishing a tag da	ata link connection	for CIP communications 1	failed.			
	TRUE: Establis	hing a tag data lir	nk connection fail	ed due to one of the follow	ving causes.			
		•The information registered for a target node in the tag data link parameters is different from the actual node information.						
	•There	was no response	from the remote	node.				
	FALSE: Other t	han the above.						
	Note You can	use this system-d	efined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP2_TDLink(	_EIP2_TDLinkOpnErr						
Meaning	CIP Communic tion Failed	ations2 Tag Data	tions2 Tag Data Link Connec- Global/local Global					
Function	Indicates that es	stablishing a tag da	ata link connection	n for CIP communications 2	2 failed.			
	TRUE: Establis	hing a tag data lir	nk connection fail	ed due to one of the follow	ving causes.			
		<ul> <li>The information registered for a target node in the tag data link parameters is different from the actual node information.</li> </ul>						
	•There	was no response	from the remote	e node.				
	FALSE: Other t	han the above.						
	Note You can	use this system-d	efined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

3-45

Variable name	_EIP_TDLinkErr						
Meaning	Tag Data Link C	Communications E	Error	Global/local	Global		
Function	NX-series CPU	Units: Indicates th	at a timeout occu	rred in a tag data link conne	ection for CIP communications 1.		
	TRUE: A timeou	ut occurred.					
	FALSE: Other th	nan the above.					
	NJ-series CPU l	Jnits: Indicates tha	at a timeout occur	red in a tag data link conne	ection.		
	TRUE: A timeou	ut occurred.					
	FALSE: Other th	han the above.					
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user pro-	Possible. Related						
gram		instructions					

Variable name	_EIP1_TDLinkE	_EIP1_TDLinkErr							
Meaning	CIP Communications1 Tag Data Link Communications Error Global/local Global								
Function	Indicates that a	timeout occurred i	in a tag data link o	connection for CIP commur	nications 1.				
	TRUE: A timeou	ut occurred.							
	FALSE: Other to	han the above.							
	Note You can	use this system-c	lefined variable o	only for NX-series CPU Ur	nits.				
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions							

Variable name	_EIP2_TDLinkErr							
Meaning	CIP Communica	IP Communications2 Tag Data Link Communi- Global/local Global						
	cations Error							
Function	Indicates that a	imeout occurred i	n a tag data link d	connection for CIP commun	ications 2.			
	TRUE: A timeou	ut occurred.						
	FALSE: Other than the above.							
	Nata Vallaga			alufar NV assisa ODLLUa				
	Note You can t	use this system-d	letined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	ined Not retained. Network Publish Published.					
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_TagAdrEr	_EIP_TagAdrErr						
Meaning	Tag Name Reso	Tag Name Resolution Error Global/local Global						
Function	NX-series CPU Units: Indicates that tag resolution for CIP communications 1 failed (i.e., the address could not be id fied from the tag name).							
	TRUE: Tag reso	TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.						
	•The si	ze of the network	variable is differ	ent from the tag settings				
		O direction that is in the CPU Unit.	s set in the tag da	ta link settings does not	agree with the I/O direction of the vari-			
	•There	is no network va	riable in the CPU	Unit that corresponds to	o the tag setting.			
	FALSE: Other t	han the above.						
	NJ-series CPU	Jnits: Indicates th	at tag resolution fa	ailed (i.e., the address cou	uld not be identified from the tag name).			
	TRUE: Tag reso	olution failed (i.e.,	the address cou	ld not be identified from	the tag name). The following causes are			
	•The si	ze of the network	variable is differ	ent from the tag settings				
		O direction that is in the CPU Unit.	s set in the tag da	ta link settings does not	agree with the I/O direction of the vari-			
	•There	is no network va	riable in the CPU	Unit that corresponds to	o the tag setting.			
	FALSE: Other t	han the above.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_TagAdrl	_EIP1_TagAdrErr						
Meaning	CIP Communic	nications1 Tag Name Resolution Global/local Global						
Function	Indicates that ta	g resolution for CII	communications	1 failed (i.e., the address o	could not be identified from the tag name).			
	TRUE: Tag res	TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.						
	•The s	ize of the network	variable is differ	ent from the tag settings.				
		O direction that is in the CPU Unit.	set in the tag da	ta link settings does not a	gree with the I/O direction of the vari-			
	•There	is no network va	riable in the CPU	Unit that corresponds to t	the tag setting.			
	FALSE: Other t	than the above.						
	Note You can	use this system-d	efined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained. Network Publish Published.					
Usage in user program	Possible.	Related instructions						

Variable name	_EIP2_TagAdrE	_EIP2_TagAdrErr					
Meaning	CIP Communica Error	ations2 Tag Name	e Resolution	Global/local	Global		
Function	Indicates that tag	Indicates that tag resolution for CIP communications 2 failed (i.e., the address could not be identified from the tag name).					
	TRUE: Tag reso possible.	TRUE: Tag resolution failed (i.e., the address could not be identified from the tag name). The following causes are possible.					
	•The si	ze of the network	variable is differ	ent from the tag settings.			
	•The I/O direction that is set in the tag data link settings does not agree with the I/O direction of the variable in the CPU Unit.						
	•There	is no network var	riable in the CPU	Unit that corresponds to t	he tag setting.		
	FALSE: Other the	nan the above.					
	Note You can u	use this system-d	efined variable o	nly for NX-series CPU Un	its.		
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP_MultiSwONErr						
Meaning	Multiple Switch	Multiple Switches ON Error Global/local Global					
Function	NX-series CPU Units: Indicates that more than one switch turned ON at the same time in CIP communications 1.						
	TRUE: More tha	an one data link s	start/stop switch o	changed to TRUE at the sa	ame time.		
	FALSE: Other to	han the above.					
	NJ-series CPU	Units: Indicates t	hat more than on	e switch turned ON at the	same time.		
	TRUE: More that	an one data link s	start/stop switch o	changed to TRUE at the sa	ame time.		
	FALSE: Other to	han the above.					
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP1_MultiSwONErr								
Meaning	CIP Communic Error	ations1 Multiple S	ons1 Multiple Switches ON Global/local Global						
Function	Indicates that n	Indicates that more than one switch turned ON at the same time in CIP communications 1.							
	TRUE: More than one data link start/stop switch changed to TRUE at the same time.								
	FALSE: Other than the above.								
	Note You can use this system-defined variable only for NX-series CPU Units.								
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions							

Variable name	_EIP2_MultiSwONErr								
Meaning	CIP Communica	CIP Communications2 Multiple Switches ON Global/local Global							
	Error								
Function	Indicates that m	Indicates that more than one switch turned ON at the same time in CIP communications 2.							
	TRUE: More than one data link start/stop switch changed to TRUE at the same time.								
	FALSE: Other than the above.								
	Note You can u	use this system-d	lefined variable o	nly for NX-series CPU Un	its.				
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user pro-	Possible.	Related							
gram		instructions							

Variable name	_EIP_TcpAppCfgErr							
Meaning	TCP/IP Setting	Error		Global/local	Global			
Function	TRUE: At least one of the set values for a TCP/IP application (FTP, NTP, SNMP) is incorrect. Or, a read operation failed.							
	FALSE: Norma	FALSE: Normal.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_NTPSrvErr						
Meaning	NTP Server Connection Error			Global/local	Global		
Function	TRUE: The NTP client failed to connect to the server (timeout).						
	FALSE: NTP is	FALSE: NTP is not set or the connection was successful.					
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user pro-	Possible.	Related					
gram		instructions					

Variable name	_EIP_DNSSrvI	_EIP_DNSSrvErr							
Meaning	DNS Server Co	nnection Error		Global/local	Global				
Function	TRUE: The DN	TRUE: The DNS client failed to connect to the server (timeout).							
	FALSE: DNS is	FALSE: DNS is not enabled. Or, DNS is enabled and the connection was successful.							
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user pro-	Possible.	Related							
gram		instructions							

## • Functional Classification: EtherNet/IP Communications Status

Variable name	_EIP_EtnOnline	_EIP_EtnOnlineSta						
Meaning	Online			Global/local	Global			
Function		NX-series CPU Units: Indicates that the built-in EtherNet/IP port's communications can be used via the communications port 1 (that is, the link is ON, IP address is defined, and there are no errors).						
	TRUE: The buil	t-in EtherNet/IP p	ort's communica	tions can be used.				
	FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.							
				rNet/IP port's communicationd there are no errors).	ons can be used via the communications			
	TRUE: The buil	t-in EtherNet/IP p	ort's communica	tions can be used.				
	FALSE: The bu cessing, or link		port's communic	ations is disabled due to a	n error in initial processing, restart pro-			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_EtnOnlineSta							
Meaning	Port1 Online Global/local Global							
Function	Indicates that the built-in EtherNet/IP port's communications can be used via the communications port 1 (that is, the link is ON, IP address is defined, and there are no errors).							
	TRUE: The built-in EtherNet/IP port's communications can be used.							
	cessing, or link	FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP2_EtnOnlineSta								
Meaning	Port2 Online	Port2 Online Global/local Global							
Function		Indicates that the built-in EtherNet/IP port's communications can be used via the communications port 2 (that is, the link is ON, IP address is defined, and there are no errors).							
	TRUE: The built-in EtherNet/IP port's communications can be used.								
	cessing, or link	FALSE: The built-in EtherNet/IP port's communications is disabled due to an error in initial processing, restart processing, or link OFF status.  Note You can use this system-defined variable only for NX-series CPU Units.							
Data type	BOOL	uoo iiio oyoloiii o	ionnod variable o	Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions							

Variable name	_EIP_TDLinkRunSta							
Meaning	Tag Data Link Communications Status			Global/local	Global			
Function	NX-series CPU	NX-series CPU Units: Indicates that at least one connection is in normal operation in CIP communications 1.						
	TRUE: Normal operation							
	FALSE: Other the	FALSE: Other than the above.						
	NJ-series CPU Units: Indicates that at least one connection is in normal operation.							
	TRUE: Normal	operation						
	FALSE: Other th	nan the above.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	ossible. Related						
gram		instructions						

Variable name	_EIP1_TDLinkRunSta						
Meaning	CIP Communica cations Status	Global					
Function	Indicates that a	least one conne	ection is in norma	I operation in CIP commu	nications 1.		
	TRUE: Normal	TRUE: Normal operation					
	FALSE: Other than the above.						
	Note You can u	use this system-d	lefined variable o	nly for NX-series CPU Un	its.		
Data type	BOOL			Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP2_TDLinkF	_EIP2_TDLinkRunSta							
Meaning	CIP Communications2 Tag Data Link Communications Status  Global/local  Global								
Function	Indicates that a	Indicates that at least one connection is in normal operation in CIP communications 2.							
	TRUE: Normal	TRUE: Normal operation							
	FALSE: Other than the above.								
	Note You can u	Note You can use this system-defined variable only for NX-series CPU Units.							
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user pro-	Possible.	Related							
gram		instructions							

Variable name	_EIP_TDLinkAllRunSta							
Meaning	All Tag Data Lin	All Tag Data Link Communications Status Global/local Global						
Function	NX-series CPU	NX-series CPU Units: Indicates that all tag data links are communicating in CIP communications 1.						
	TRUE: Tag data links are communicating in all connections as the originator.							
	FALSE: An erro	r occurred in at le	east one connect	ion.				
	NJ-series CPU Units: Indicates that all tag data links are communicating.							
	TRUE: Tag data	a links are commu	unicating in all co	nnections as the originato	r.			
	FALSE: An erro	r occurred in at le	east one connect	ion.				
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP1_TDLinkAllRunSta							
Meaning	CIP Communications Sta	•	tions1 All Tag Data Link Com- us Global/local Global					
Function	Indicates that all	tag data links are	communicating in	n CIP communications 1.				
	TRUE: Tag data	a links are commu	ınicating in all co	nnections as the originato	r.			
	FALSE: An error occurred in at least one connection.							
	Note You can use this system-defined variable only for NX-series CPU Units.							
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP2_TDLinkA	_EIP2_TDLinkAllRunSta						
Meaning		CIP Communications2 All Tag Data Link Communications Status  Global/local Global						
Function	Indicates that all	Indicates that all tag data links are communicating in CIP communications 2.						
	TRUE: Tag data	a links are comm	unicating in all co	nnections as the originat	tor.			
	FALSE: An erro	FALSE: An error occurred in at least one connection.						
	Note You can	Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_RegTargetSta [255]					
Meaning	Registered Targ	et Node Informat	ion	Global/local	Global	
Function	NX-series CPU Units: Gives a list of nodes for which built-in EtherNet/IP connections are registered for CIP communications 1.					
	This variable is	valid only when t	he built-in EtherN	let/IP port is the originator	:	
	Array[x] is TRUI	E: The connection	n to the node with	n a target node ID of x is r	egistered.	
	Array[x] is FALS	E: The connection	on to the node wit	th a target node ID of x is	not registered.	
	NJ-series CPU	Units: Gives a lis	t of nodes for whi	ich built-in EtherNet/IP co	nnections are registered.	
	This variable is	valid only when t	he built-in EtherN	let/IP port is the originator	:	
	Array[x] is TRUI	E: The connection	n to the node with	n a target node ID of x is r	egistered.	
	Array[x] is FALS	E: The connection	on to the node wit	th a target node ID of x is	not registered.	
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE	
R/W access	R	Retained	Not retained.	Network Publish	Published.	
Usage in user pro-	Possible.	Related		·	·	
gram		instructions				

Variable name	_EIP1_RegTargetSta [255]						
Meaning	CIP Communications1 Registered Target Node Information Global/local Global						
Function	Gives a list of n	odes for which bu	uilt-in EtherNet/IF	connections are registere	ed for CIP communications 1.		
	This variable is	valid only when t	he built-in EtherN	let/IP port is the originator	:		
	Array[x] is TRUE: The connection to the node with a target node ID of x is registered.						
	Array[x] is FALSE: The connection to the node with a target node ID of x is not registered.						
	Note You can u	use this system-d	lefined variable o	nly for NX-series CPU Un	its.		
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user pro-	Possible.	Related					
gram		instructions					

Variable name	_EIP2_RegTar	_EIP2_RegTargetSta [255]							
Meaning	CIP Communic Information	CIP Communications2 Registered Target Node Information Global/local Global							
Function		Gives a list of nodes for which built-in EtherNet/IP connections are registered for CIP communications 2.							
	This variable is	valid only when t	he built-in EtherN	Net/IP port is the originator	<b>f.</b>				
	Array[x] is TRU	Array[x] is TRUE: The connection to the node with a target node ID of x is registered.							
	Array[x] is FALS	SE: The connection	on to the node wi	th a target node ID of x is	not registered.				
	Note You can	use this system-d	lefined variable o	nly for NX-series CPU Un	its.				
Data type	ARRAY [0255	] OF BOOL		Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions							

Variable name	_EIP_EstbTarg	_EIP_EstbTargetSta [255]						
Meaning	Normal Target I	Node Information Global/local Global						
Function	NX-series CPU munications 1.	NX-series CPU Units: Gives a list of nodes that have normally established EtherNet/IP connections for CIP communications 1.						
	Array[x] is TRU	E: The connection	n to the node witl	n a target node ID of x was	s established normally.			
	Array[x] is FALSE: The connection to the node with a target node ID of x was not established, or an error occurred.							
	NJ-series CPU	Units: Gives a lis	t of nodes that ha	ave normally established E	EtherNet/IP connections.			
	Array[x] is TRU	E: The connection	n to the node with	n a target node ID of x was	s established normally.			
	Array[x] is FALS	E: The connectio	n to the node with	n a target node ID of x was	not established, or an error occurred.			
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_EstbTar	_EIP1_EstbTargetSta [255]							
Meaning	CIP Communic	CIP Communications1 Normal Target Node Global/local Global Global							
Function	Gives a list of r	Gives a list of nodes that have normally established EtherNet/IP connections for CIP communications 1.							
	Array[x] is TRU	Array[x] is TRUE: The connection to the node with a target node ID of x was established normally.							
	Array[x] is FALS	SE: The connection	n to the node wit	h a target node ID of x was	s not established, or an error occurred.				
	Note You can	use this system-c	lefined variable o	nly for NX-series CPU Un	its.				
Data type	ARRAY [0255	] OF BOOL		Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions			•				

Variable name	_EIP2_EstbTargetSta [255]						
Meaning	CIP Communication	ations2 Normal Target Node Global/local Global					
Function	Gives a list of n	odes that have no	ormally establish	ed EtherNet/IP connection	ns for CIP communications 2.		
	Array[x] is TRUI	E: The connection	n to the node with	h a target node ID of x was	s established normally.		
	Array[x] is FALSE: The connection to the node with a target node ID of x was not established, or an error occurred.						
	Note You can u	use this system-d	lefined variable o	nly for NX-series CPU Uni	its.		
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user pro-	Possible.	Related					
gram		instructions					

Variable name	_EIP_TargetPLCModeSta [255]						
Meaning	Target PLC Ope	rating Mode Global/local Global					
Function	NX-series CPU Units: Shows the operating status of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP port as the originator.						
	The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.						
	Array[x] is TRU	E: This is the ope	rating state of the	e target Controller with a r	node address of x.		
	Array[x] is FALS	E: Other than the	e above.				
	NJ-series CPU Units: Shows the operating status of the target node Controllers that are connected with the built-in EtherNet/IP port as the originator.						
		al Target Node In	•		lode Information is TRUE. If the corre- roller Operating Information indicates		
	Array[x] is TRUI	E: This is the ope	rating state of the	e target Controller with a r	node address of x.		
	Array[x] is FALS	E: Other than the	e above.				
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP1_TargetPLCModeSta [255]					
Meaning		ations1 Target PL	.C Operating	Global/local	Global	
	Mode					
Function	Shows the operating status of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP port as the originator.					
	The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.					
	Array[x] is TRUI	E: This is the ope	rating state of the	e target Controller with a r	node address of x.	
	Array[x] is FALS	E: Other than the	e above.			
	Note You can u	use this system-d	efined variable o	nly for NX-series CPU Un	its.	
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE	
R/W access	R	Retained	Not retained.	Network Publish	Published.	
Usage in user pro-	Possible.	Related				
gram		instructions				

Variable name	_EIP2_TargetPLCModeSta [255]						
Meaning	CIP Communic	ations2 Target PLC Operating Global/local Global					
	Mode						
Function	Shows the operating status of the target node Controllers that are connected for CIP communications 2, with the built-in EtherNet/IP port as the originator.						
	The array elements are valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.						
	Array[x] is TRU	E: This is the ope	rating state of the	e target Controller with a n	node address of x.		
	Array[x] is FALS	SE: Other than the	e above.				
	Note You can u	use this system-d	efined variable o	nly for NX-series CPU Uni	its.		
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user pro-	Possible.	Related					
gram		instructions					

Variable name	_EIP_TargetPLCErr [255]							
Meaning	Target PLC Erro	or Information		Global/local	Global			
Function	NX-series CPU Units: Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.							
	Array[x] is TRU	E: A fatal or non-f	atal error occurre	ed in the target Controller	with a target node ID of x.			
	Array[x] is FALS	SE: Other than the	e above.					
	that are connect corresponding I able is FALSE.	ted with the built- Normal Target No	in EtherNet/IP pode Information is	orts as the originator. The TRUE. The immediately	al errors) of the target node Controllers array elements are valid only when the preceding value is retained if this vari-			
	Array[x] is TRU	E: A fatal or non-t	atal error occurre	ed in the target Controller	with a target node ID of x.			
	Array[x] is FALS	SE: Other than the	e above.					
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_TargetP	_EIP1_TargetPLCErr [255]							
Meaning	CIP Communic mation	nications1 Target PLC Error Infor- Global/local Global							
Function	for CIP commu when the corre this variable is <i>Array[x]</i> is TRU	Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected for CIP communications 1, with the built-in EtherNet/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.  Array[x] is TRUE: A fatal or non-fatal error occurred in the target Controller with a target node ID of x.  Array[x] is FALSE: Other than the above.							
Data type	ARRAY [0255	] OF BOOL		Range of values	TRUE or FALSE				
R/W access	R	Retained	Not retained.	Network Publish	Published.				
Usage in user program	Possible.	Related instructions							

Variable name	_EIP2_TargetPLCErr [255]						
Meaning	CIP Communic mation	ations2 Target PL	tions2 Target PLC Error Infor- Global/local Global				
Function	Shows the error status (logical OR of fatal and non-fatal errors) of the target node Controllers that are connected for CIP communications 2, with the built-in EtherNet/IP ports as the originator. The array elements are valid only when the corresponding Normal Target Node Information is TRUE. The immediately preceding value is retained if this variable is FALSE.  Array[x] is TRUE: A fatal or non-fatal error occurred in the target Controller with a target node ID of x.  Array[x] is FALSE: Other than the above.  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP_TargetNo	deErr [255]						
Meaning	Target Node Er	ror Information	or Information Global/local Global					
Function	NX-series CPU Units: Indicates that the connection for the Registered Target Node Information for CIP communications 1 was not established or that an error occurred in the target Controller.							
	The array eleme	ents are valid only	y when the Regis	stered Target Node Inform	ation is TRUE.			
	tered Target No	de Information is	ΓRUE and the No	· ·	node for a target node ID of x (the Regision is FALSE), or a connection was			
	tion is FALSE), mation is TRUE	or a connection v	vas normally esta Target Node Info	ablished with the target no	x (the Registered Target Node Informa- de (the Registered Target Node Infor- or occurred in the target Controller (the			
		Units: Indicates the coccurred in the ta		n for the Registered Targe	t Node Information was not established			
	The array eleme	ents are valid only	y when the Regis	tered Target Node Inform	ation is TRUE.			
	tered Target No	de Information is	ΓRUE and the No		node for a target node ID of x (the Regision is FALSE), or a connection was			
	Array[x] is FALSE: The target node is not registered for a target node ID of x (the Registered Target Node Information is FALSE), or a connection was normally established with the target node (the Registered Target Node Information is TRUE and the Normal Target Node Information is TRUE). An error occurred in the target Controller (the Target PLC Error Information is TRUE).							
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user program	Possible.	Related instructions						

Variable name	_EIP1_TargetNodeErr [255]						
Meaning	CIP Communication	ations1 Target No	de Error Infor-	Global/local	Global		
Function	Indicates that the connection for the Registered Target Node Information for CIP communications 1 was not established or that an error occurred in the target Controller.						
	The array eleme	ents are valid only	when the Regis	stered Target Node Inform	ation is TRUE.		
	Array[x] is TRUE: A connection was not normally established with the target node for a target node ID of x (the Registered Target Node Information is TRUE and the Normal Target Node Information is FALSE), or a connection was established with the target node but an error occurred in the target Controller.						
	tion is FALSE), mation is TRUE Target PLC Erro	or a connection wand the Normal or Information is 1	vas normally esta Target Node Info TRUE).	ablished with the target no	x (the Registered Target Node Informa- de (the Registered Target Node Infor- or occurred in the target Controller (the its.		
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP2_TargetNodeErr [255]						
Meaning	CIP Communication	ations2 Target No	ode Error Infor-	Global/local	Global		
Function	Indicates that the connection for the Registered Target Node Information for CIP communications 2 was not established or that an error occurred in the target Controller.						
	The array eleme	ents are valid only	y when the Regis	stered Target Node Inform	ation is TRUE.		
	tered Target No	de Information is 1	ΓRUE and the No	•	node for a target node ID of x (the Regision is FALSE), or a connection was		
	Array[x] is FALSE: The target node is not registered for a target node ID of x (the Registered Target Node Information is FALSE), or a connection was normally established with the target node (the Registered Target Node Information is TRUE and the Normal Target Node Information is TRUE). An error occurred in the target Controller (the Target PLC Error Information is TRUE).  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	ARRAY [0255]	OF BOOL		Range of values	TRUE or FALSE		
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Possible.	Related instructions					

Variable name	_EIP_NTPResu	ılt		Member name	.ExecTime		
Meaning	NTP Last Opera	ation Time		Global/local	Global		
Function	Gives the last time that NTP processing ended normally.						
	The time that w	The time that was obtained from the NTP server is stored when the time is obtained normally.					
	The time is not	stored if it is not o	obtained from the	NTP server normally.			
				m. There may be a delay m an external device.	in updating it. Use this variable only to		
Data type	Structure: _sN7	P_RESULT		Range of values	Depends on data type.		
	Members: DATI	E_AND_TIME					
R/W access	R	Retained	Not retained.	Network Publish	Published.		
Usage in user program	Not possible.	Related instructions	You can read the contents of this variable with the GetNTPStatus instruction.				

Variable name	_EIP_NTPResu	lt		Member name	.ExecNormal			
Meaning	NTP Operation	Result		Global/local	Global			
Function	This variable shows if the NTP operation ended normally.							
	TRUE: Indicates	s an NTP normal	end.					
	FALSE: Indicates that NTP operation ended in an error or has not been executed even once.							
				m. There may be a delay n an external device.	in updating it. Use this variable only to			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Not possible.	Related You can read the contents of this variable with the GetNTPStatus instruction.						
gram		instructions						

## • Functional Classification: EtherNet/IP Communications Switches

Variable name	_EIP_TDLinkStartCmd						
Meaning	Tag Data Link C	Communications S	Start Switch	Global/local	Global		
Function	NX-series CPU Units: Change this variable to TRUE to start tag data links for CIP communications 1.						
	It automatically	changes back to	FALSE after tag	data link operation starts.			
	NJ-series CPU	Units: Change th	is variable to TRI	JE to start tag data links.			
	It automatically	changes back to	FALSE after tag	data link operation starts.			
	Note Demotés		ahama ta EALOI	- fue as the same and a second	w fue we then Common Chordin It also were to		
		rce this switch to utomatically.	change to FALSE	= from the user program o	r from the Sysmac Studio. It changes to		
Data type	BOOL	atomaticany.		Range of values	TRUE or FALSE		
R/W access	R/W	Retained	Not retained.	Network Publish	Published.		
	Possible.	Related		TYOUVOIN I UDIIOII	i abilotica.		
Usage in user pro-	russibile.	instructions					
gram		Instructions					

Variable name	_EIP1_TDLinkStartCmd							
Meaning		ations1 Tag Data	Link Communi-	Global/local	Global			
	cations Start Sv	vitch						
Function	Change this vari	able to TRUE to s	tart tag data links	for CIP communications 1.				
	It automatically	changes back to	FALSE after tag	data link operation starts.				
		rce this switch to utomatically.	change to FALSI	E from the user program o	r from the Sysmac Studio. It changes to			
	Note You can u	use this system-d	efined variable o	nly for NX-series CPU Un	its.			
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R/W	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	EIP2_TDLinkStartCmd							
Meaning	CIP Communications Start Sv	ations2 Tag Data vitch	Link Communi-	Global/local	Global			
Function	Change this variable to TRUE to start tag data links for CIP communications 2.							
	It automatically	changes back to	FALSE after tag	data link operation starts.				
	FALSE a	Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.  Note You can use this system-defined variable only for NX-series CPU Units.						
Data type	BOOL	<u> </u>		Range of values	TRUE or FALSE			
R/W access	R/W	Retained	Retained Not retained. Network Publish Published.					
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP_TDLinkStopCmd							
Meaning	Tag Data Link C	Communications S	ommunications Stop Switch Global/local Global					
Function	NX-series CPU Units: Change this variable to TRUE to stop tag data links for CIP communications 1.							
	It automatically	changes back to	FALSE after tag	data link operation stops.				
	NJ-series CPU	Units: Change th	is variable to TRI	JE to stop tag data links.				
	It automatically	changes back to	FALSE after tag	data link operation stops.				
		g	g					
			change to FALSI	E from the user program o	r from the Sysmac Studio. It changes to			
	FALSE a	utomatically.						
Data type	BOOL			Range of values	TRUE or FALSE			
R/W access	R/W	Retained	Not retained.	Network Publish	Published.			
Usage in user pro-	Possible.	Related						
gram		instructions						

Variable name	_EIP1_TDLink	_EIP1_TDLinkStopCmd							
Meaning	CIP Communic	tions1 Tag Data Link Communi- itch Global							
Function	Change this var	Change this variable to TRUE to stop tag data links for CIP communications 1.							
	It automatically	changes back to	FALSE after tag	data link operation stops.					
		orce this switch to automatically.	change to FALSI	E from the user program o	or from the Sysmac Studio. It changes to				
	Note You can	use this system-d	lefined variable o	nly for NX-series CPU Ur	nits.				
Data type	BOOL			Range of values	TRUE or FALSE				
R/W access	R/W	/W Retained Not retained. Network Publish Published.							
Usage in user program	Possible.	Related instructions							

Variable name	_EIP2_TDLinkStopCmd				
Meaning	CIP Communications2 Tag Data Link Communications Stop Switch			Global/local	Global
Function	Change this variable to TRUE to stop tag data links for CIP communications 2.				
	It automatically changes back to FALSE after tag data link operation stops.				
	<ul><li>Note Do not force this switch to change to FALSE from the user program or from the Sysmac Studio. It changes to FALSE automatically.</li><li>Note You can use this system-defined variable only for NX-series CPU Units.</li></ul>				
Data type	BOOL			Range of values	TRUE or FALSE
R/W access	R/W	Retained	Not retained.	Network Publish	Published.
Usage in user program	Possible.	Related instructions			



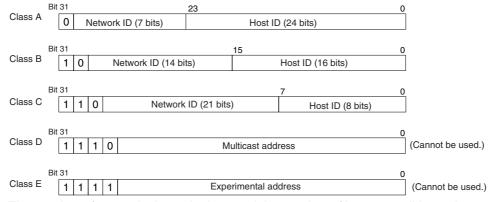
## **Determining IP Addresses**

4-1	IP Add	resses	4-2
	4-1-1	IP Address Configuration	4-2
	4-1-2	Allocating IP Addresses	4-3
	4-1-3	Subnet Masks	4-3
	4-1-4	CIDR	4-3
4-2	Built-in	EtherNet/IP Port IP Address Settings	4-4
	4-2-1	Determining IP Addresses	4-4
	4-2-2	Setting IP Addresses	4-5
	4-2-3	Online Connection	4-8
	4-2-4	Checking the Current IP Address	4-11
4-3	Private	and Global Addresses4	l-12
	4-3-1	Private and Global Addresses	4-12
	4-3-2	Using a Private Address for the Built-in EtherNet/IP Port	4-13
	4-3-3	Using a Global Address for the Built-in EtherNet/IP Port	4-14

## **IP Addresses**

#### 4-1-1 **IP Address Configuration**

IP addresses are made up of 32 bits of binary data that specify the network number (net ID) and host number (host ID). The network number identifies the network, and the host number identifies the node (or host) on the network. IP addresses are divided into three classes, A, B, and C, so that the address system can be selected according to the scale of the network. (Classes D and E are not used.)



The number of networks in each class and the number of hosts possible on the network differ according to the class.

Class	Number of networks	Number of hosts	
Class A	Small	2 <sup>24</sup> –2 max. (16,777,214 max.)	
Class B	Medium	2 <sup>16</sup> –2 max. (65,534 max.)	
Class C	Large	28-2 max. (254 max.)	

The 32 bits of binary data in an IP address are divided into four sections of eight bits each. IP addresses are represented by the decimal equivalent of each of the four octets in the 32-bit address, each separated by a period.

For example, the binary address 10000010 00111010 00010001 00100000 would be represented as 130.58.17.32.

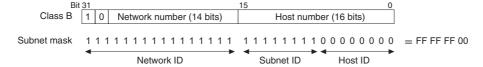
## 4-1-2 Allocating IP Addresses

You must assign IP addresses nodes so that each IP address is assigned only once in the network or between several networks.

#### 4-1-3 Subnet Masks

Operation and management of a network can become very difficult if too many nodes are connected on a single network. In such a case it can be helpful to configure the system so that a single network is divided up into several subnetworks. Internally the network can be treated as a number of subnetworks, but from the outside it acts as a single network and uses only a single network ID. To establish subnetworks, the host ID in the IP address is divided into a subnet ID and a host ID by using a setting called the subnet mask. The subnet mask indicates which part of the host ID is to be used as the subnet ID. All bits in the subnet mask that correspond to the bits in the IP address used either as the network ID or subnet ID are set to "1," and the remaining bits, which correspond to the bits in the IP address actually used for the host ID, are set to "0."

The following example shows the subnet mask for an 8-bit subnet ID used in class-B IP addresses.



Set the same subnet mask for all of the nodes on that subnetwork. The built-in EtherNet/IP port supports CIDR (Classless Inter-Domain Routing). The subnet mask can be set to 192.0.0.0 to 255.255.252. If subnetworks are not used, set the following subnet mask values for IP address classes A to C.

Class	Subnet mask
Class A	255.0.0.0
Class B	255.255.0.0
Class C	255.255.255.0

#### 4-1-4 CIDR

CIDR, or classless interdomain routing, is used to assign IP addresses that do not use classes. IP addresses that use classes are separated into blocks according to network IDs and host IDs, resulting in inefficient usage of IP address space. CIDR does not use classes, so IP address space can be divided as required to more efficiently use IP address space. For example, using a subnet mask setting with CIDR enables building a horizontally distributed network exceeding 254 nodes even if a class C address block (e.g., 192, 168...) is used.

Subnet Mask Range		
192.0.0.0 to 255.255.255.252		

## **Built-in EtherNet/IP Port IP Address** 4-2 **Settings**

#### 4-2-1 **Determining IP Addresses**

Use one of the following methods to set the IP address of the built-in EtherNet/IP port.

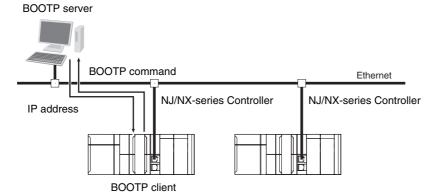
#### Setting a User-specified IP Address

If you need to change the default IP address of the built-in EtherNet/IP port or if you need to use the built-in EtherNet/IP port with another EtherNet/IP node, set the IP address to the required value. For an NX-series CPU Unit, do not set the same network address for both built-in EtherNet/IP ports 1 and 2.

#### Automatically Obtaining the IP Address from the BOOTP Server

There are two methods to automatically obtain an IP address.

- Obtain the IP address from the BOOTP server each time the power is turned ON.
- Obtain the IP address from the BOOTP server once when the power is turned ON and then do not allow it to change.

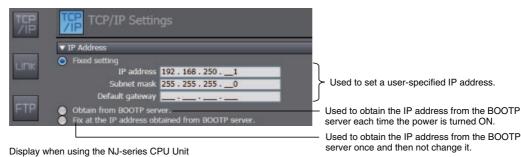


## 4-2-2 Setting IP Addresses

Use the Sysmac Studio to set the IP address of the built-in EtherNet/IP port.

**1** Select the setting method for IP addresses.

Make the following settings on the TCP/IP Settings Display of the Built-in EtherNet/IP Port Settings Tab Page in the Controller Setup to set the local IP address.



For an NX-series CPU Unit, the IP addresses must be set separately for built-in EtherNet/IP ports 1 and 2.



#### **Precautions for Correct Use**

You cannot use the same network address for both the built-in EtherNet/IP port 1 and port 2.

- 2 Connect the Sysmac Studio to the NJ/NX-series CPU Unit via a USB connection or the Ethernet network.
- **3** Connect the Sysmac Studio online to the NJ/NX-series CPU Unit. Refer to *4-2-3 Online Connection* for the procedure to connect online.
- **4** Use one of the following methods to download the IP addresses that were set on the Sysmac Studio to the NJ/NX-series CPU Unit.
  - Go online with the Controller, and then select *Synchronization* from the Controller Menu. The data on the computer and the data in the physical Controller are compared automatically.
  - Click the Transfer to Controller Button.

Note Use the Synchronization Menu of the Sysmac Studio to upload and download data.



After the IP address settings are downloaded, the IP address is reflected in the CPU Unit as follows:

#### **Setting a User-specified IP Address**

After the IP address settings are downloaded, the set IP address is automatically saved in the CPU Unit.

#### Obtaining the IP Address from the BOOTP Server Each Time the Power Is Turned ON

After the data is downloaded, the IP address from the BOOTP server is automatically saved in the CPU Unit. Each time the power supply is turned ON, the IP address from the BOOTP server is automatically saved in the CPU Unit.



#### **Additional Information**

If you cannot obtain the IP address from the BOOTP server or the obtained IP address is not correct, select the Fixed setting Option in the IP Address Area and manually set the IP address, subnet mask, and default gateway. Requests to the BOOTP server to obtain the IP address will continue if there is a failure to connect to the BOOTP server.

#### Obtaining the IP Address from the BOOTP Server Once When the Power Is Turned ON and Then Not Allow It to Change

After the I/O address is downloaded, the IP address from the BOOTP server is automatically saved in the Controller and then the same address is used.



#### **Additional Information**

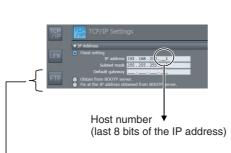
- The TCP/IP Settings Display is not updated even if the IP address is obtained normally from the BOOTP server. To check the IP address that was obtained from the BOOTP server on the TCP/IP Display, upload the project from the NJ/NX-series Controller.
- If you cannot obtain the IP address from the BOOTP server, the Fix at the IP address obtained from BOOTP server Option is selected on the TCP/IP Display. Select Fixed Setting in the IP Address Settings and manually set the IP address, subnet mask, and default gateway.
- If the Controller power supply is turned ON when the IP address was not normally obtained from the BOOTP server, the setting remains at Fix at the IP address obtained from BOOTP server.
- After the IP address is obtained from the BOOTP server, the built-in EtherNet/IP port IP address setting is automatically set to Fixed setting. Therefore, the IP address will not match when the program is verified on the Sysmac Studio.



#### **Additional Information**

For an NJ-series CPU Unit, when the local IP address of the built-in EtherNet/IP port is set, the FINS node address is automatically set as shown below. Note that the FINS node address is provided only to an NJ-series CPU Unit.

Example: Pairing IP Addresses and FINS Node Addresses with Automatic Creation Method



TCP/IP Settings Display



After the IP address is obtained from the BOOTP server, the node address of the built-in EtherNet/IP FINS is set.

The same value as the host number is set. When FINS node address are created automatically, the FINS node address is the host ID and it cannot be changed. It can be changed if you use both methods or an IP address table method.

The FINS node address is required for FINS communications (e.g., to connect to the CX-Integrator and other Support Software). When you automatically set node addresses, do not set the lower 8 bits of the IP address to 000 or 255.

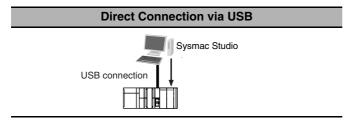
#### 4-2-3 **Online Connection**

Connect the Sysmac Studio online to the CPU Unit.

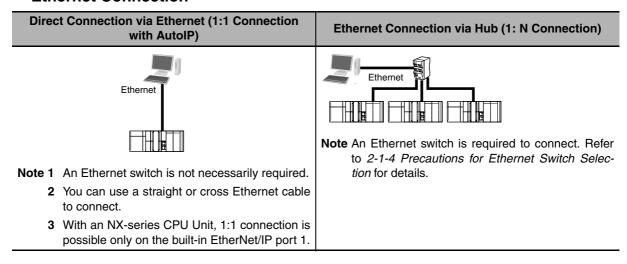
## Types of Connections between the CPU Unit and Computer That **Runs the Sysmac Studio**

The CPU Unit and the computer that runs Sysmac Studio are connected as shown below via USB or Ethernet:

#### **USB Connection**



#### **Ethernet Connection**



## Connection from USB Across Ethernet Remote Connection via USB (1:N Connection)

## Ethernet

Note An Ethernet switch is required to connect. Refer to 2-1-4 Precautions for Ethernet Switch Selection for details.



#### **Additional Information**

- Auto IP automatically assigns IP addresses in Windows 98 and later operating systems. Unique IP addresses are automatically assigned from the address 169.254.0.0 to 169.254.255.255.
- If the IP address of the connected EtherNet/IP port is changed when the Sysmac Studio is connected online via a built-in EtherNet/IP port, a timeout will occur in the Sysmac Studio. In the case, switch the Sysmac Studio status to offline, change to the IP address of the connected built-in EtherNet/IP port, and then switch back the Sysmac Studio status to online. This will allow you to reconnect.



#### **Precautions for Correct Use**

If there is more than one node with the same IP address in the EtherNet/IP network, the built-in EtherNet/IP port will connect to the node that is detected first. An IP Address Duplication Error will not occur.

### **Online Connection Procedure**

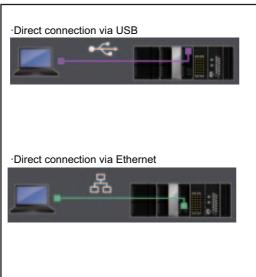
Connect the CPU Unit and the computer that runs the Sysmac Studio via USB or Ethernet, and then perform the following procedure.

1

Select *Controller - Communications Setup* and click the **OK** Button in the Sysmac Studio Project Window.

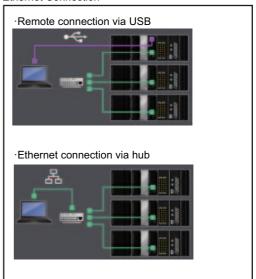
1:1 Connection

**Direct Connection** 



#### 1:N Connection

**Ethernet Connection** 





#### **Additional Information**

If there is an error in the set IP address, the CPU Unit behaves as follows:

- The NET RUN indicator on the CPU Unit does not light and the NET ERR indicator flashes red. On an NX-series CPU Unit, indicators will indicate the status of each built-in EtherNet/IP port.
- An IP Address Setting Error is recorded in the event log.

If you use a direct connection via USB or via Ethernet, it is possible to change the IP address because the Sysmac Studio can connect to the CPU Unit via the built-in EtherNet/IP port.



#### **Precautions for Correct Use**

- If the IP address is duplicated or not set correctly, communications are not possible via the EtherNet/IP network.
- The IP address range shown below is used by the system and cannot be specified. 169.254.0.0 to 169.254.255.255 192.168.255.0 to 192.168.255.255
- Due to Ethernet restrictions, you cannot specify the IP addresses that are described below.
  - An IP address that is all 0's or all 1's
  - IP addresses that start with 127, 0, or 255 (decimal)
  - IP addresses that have a host ID that is all 0's or all 1's
  - Class-D IP addresses (224.0.0.0 to 239.255.255.255)
  - Class-E IP addresses (240.0.0.0 to 255.255.255.255)

## Connecting from a Saved Project

The connection configuration that is set (USB or EtherNet/IP) is saved in the project. (The file is xxx.smc.) If you open a saved project on the Sysmac Studio, you can connect to the EtherNet/IP network without redoing the settings.

## 4-2-4 Checking the Current IP Address

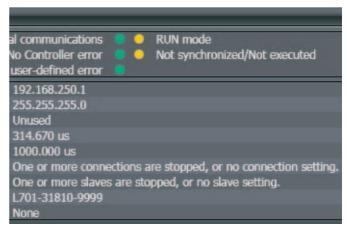
The current IP address can be confirmed in the Controller Status Pane of the Sysmac Studio either when it is set manually or obtained from the BOOTP server.

Display when using the NJ-series CPU Unit

· Basic Controller Status Pane



• Controller Status Pane with Details

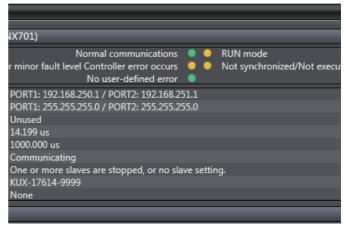


Display when using the NX-series CPU Unit

· Basic Controller Status Pane



Controller Status Pane with Details





#### **Additional Information**

- If you obtain the IP address from the BOOTP server, you can check the obtained IP address by synchronizing and uploading the controller settings from the Sysmac Studio.
- If the IP address of the built-in EtherNet/IP port is not registered due to the following reasons, the IP address field shows "0.0.0.0".
  - The IP address was not obtained from the BOOTP server.
  - The built-in EtherNet/IP port on an NX-series CPU Unit is set to disable. Refer to 5-1 TCP/IP Settings Display.

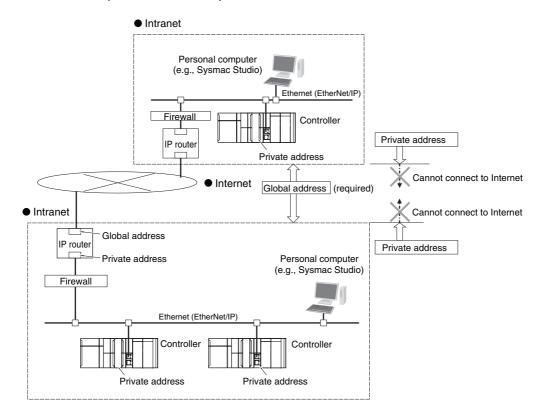
## **Private and Global Addresses**

#### 4-3-1 **Private and Global Addresses**

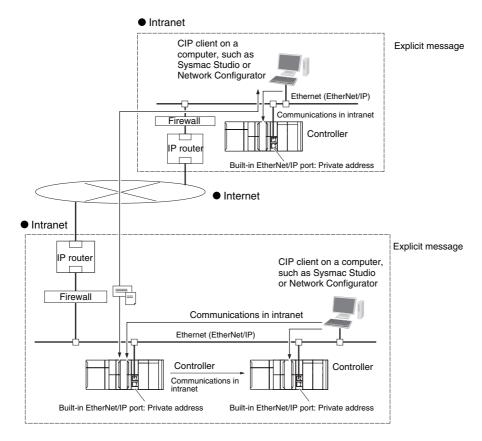
There are two kinds of IP addresses, private and global.

IP address	Description		
Global address	These are IP addresses that connect directly to the Internet. Allocated by application to NIC, each address is unique in the world, and as many as 4.3 billion can be allocated worldwide.		
Private address	These are IP addresses for Intranet (LAN) use. Direct connection to the Internet is not possible. Frames that include private IP addresses are restricted by the router from being sent outside the LAN.		

Generally, as shown below, global addresses in the intranet are allocated only to IP routers (such as broadband routers) interfaced with the Internet. All other nodes in the intranet, which includes the builtin EtherNet/ IP port, are allocated private addresses.



## 4-3-2 Using a Private Address for the Built-in EtherNet/IP Port



#### Conditions for Communications Applications

If the built-in EtherNet/IP port uses a private address, you can use explicit message communications service under the following conditions.

- The explicit message communications service can be executed on the intranet between built-in EtherNet/IP ports with private addresses only.
- A device such as a personal computer (CIP applications including the Network Configurator) cannot connect online and communicate over the Internet with a built-in EtherNet/IP port that has a private address. Explicit message communications are also not possible over the Internet between built-in EtherNet/IP ports with private addresses.

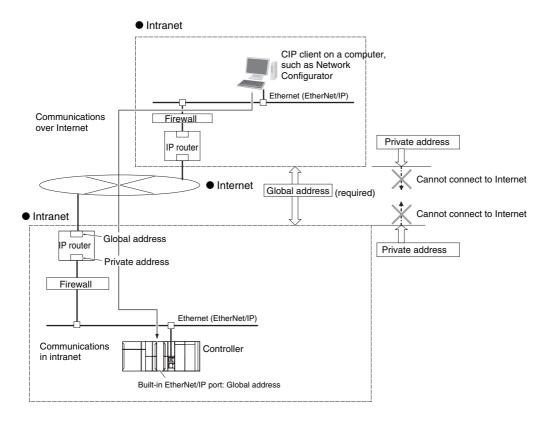


#### **Precautions for Correct Use**

#### **Network Security and Firewalls**

To set up an intranet through a global address involves network security considerations. Be sure to consult with a network specialist in advance and consider installation of a firewall. After a firewall has been set up by a communications company technician, there may be some applications that cannot be used. Be sure to check first with the communications company technician.

#### Using a Global Address for the Built-in EtherNet/IP Port 4-3-3



#### Conditions for Communications Applications

You can use the explicit message communications service over the Internet under the following conditions.

- A device such as a personal computer (a CIP application including the Network Configurator) can connect online and communicate over the Internet with a built-in EtherNet/IP port that has a global address.
- The TCP port number (44818) or UDP port number (44818) that is used for EtherNet/IP cannot be used because it is prohibited by a firewall in the communications path.



#### **Precautions for Correct Use**

#### **Network Security and Firewalls**

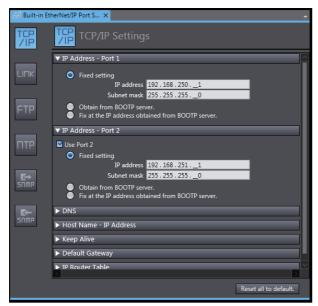
To set a global IP address for a built-in EtherNet/IP port involves network security considerations. It is recommended that the user contract with a communications company for a dedicated line, rather than for a general line such as a broadband line. Also, be sure to consult with a network specialist and consider security measures such as a firewall. After a firewall has been set up by a communications technician, there may be some applications that cannot be used. Be sure to check first with the communications technician.

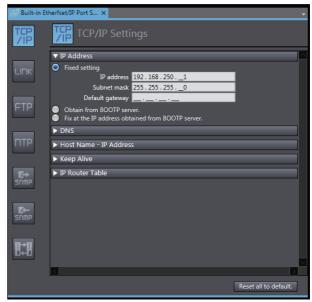


# Sysmac Studio Settings for the Built-in EtherNet/IP Port

5-1	TCP/IP Settings Display	5-2
5-2	Link Settings Display	5-6
5-3	FTP Display	5-7
5-4	NTP Settings Display	5-8
5-5	SNMP Settings Display	5-10
5-6	SNMP Trap Settings Display	5-12

#### **TCP/IP Settings Display** 5-1





**NX-series CPU Unit** 

NJ-series CPU Unit

#### IP Address - Port 1 and Port 2 (NX-series CPU Unit)

Set IP addresses for built-in EtherNet/IP port 1 and built-in EtherNet/IP port 2.

Setting	Description	Default
Use Port 2	Select the check box to use the built-in EtherNet/IP port 2.	Selected (use)
IP address setting method	Select one of the following IP address setting methods for the built-in EtherNet/IP port.  • Fixed setting  • Obtain from BOOTP server.  • Fix at the IP address obtained from BOOTP server.	Fixed setting
IP address*2	Set the IP address for the built-in EtherNet/IP port.*3	Built-in EtherNet/IP port 1: 192.168.250.1 Built-in EtherNet/IP port 2: 192.168.251.1
Subnet mask*2	Set the subnet mask for the built-in EtherNet/IP port.	255.255.255.0

<sup>\*1</sup> These settings are required if you set IP address setting method to Fixed setting.

<sup>\*2</sup> Refer to 4-2 Built-in EtherNet/IP Port IP Address Settings for details on setting IP addresses.



#### **Precautions for Correct Use**

Do not set the same network address for both Built-in EtherNet/IP Port 1 and Built-in EtherNet/IP Port 2.

## • IP Address (NJ-series CPU Unit)

Setting	Description	Default
IP address setting	Select one of the following IP address setting methods for	Fixed setting
method	the built-in EtherNet/IP port.*1	
	Fixed setting	
	Obtain from BOOTP server.	
	Fix at the IP address obtained from BOOTP server.	
IP address*2	Set the IP address for the built-in EtherNet/IP port.	192.168.250.1
Subnet mask*2	Set the subnet mask for the built-in EtherNet/IP port.	255.255.255.0
Default gateway*3	Set the IP address of the default gateway for the built-in EtherNet/IP port. This setting is not required when the default gateway is not used.	None
	When the IP address settings are obtained from a BOOTP server, the default gateway set in this item is disabled.	

<sup>\*1</sup> Refer to 4-2 Built-in EtherNet/IP Port IP Address Settings for details on setting IP addresses.

#### DNS

Setting	Description	Default
DNS	Enable using DNS if a DNS is used to resolve host names when host names are specified for the remote communications nodes in CIP communications and socket instructions.	Do not use.
	A DNS server is required to use DNS.	
Priority DNS server*1	Set the IP address of the DNS server. You can set priority and secondary IP addresses.	None
Secondary DNS server		None
Domain name*1	Sets the domain name of the domain to which the built-in EtherNet/IP port belongs. (Single-byte alphanumeric characters, dots, and hyphens: 48 characters max.)	None

<sup>\*1</sup> These settings are required when DNS is used.

#### Host Name - IP Address

Setting	Description	Default
Host Name	Addresses are converted according to this setting when host names are used to specify remote communications nodes.  They can be set even if DNS is not used. You can set up to six host names. (You can use up to 200 single-byte alphanumeric characters, dots, and hyphens with up to 63 single-byte alphanumeric characters between dots.)	None
IP Address	Set the IP addresses of the registered hosts.	None

<sup>\*2</sup> These settings are required if you set IP address setting method to Fixed setting.

<sup>\*3</sup> This setting is valid if you set IP address setting method to Fixed setting.

## Keep Alive

Setting	Description	Default
Keep Alive	Set whether to use the remote node keep alive function of connected servers and clients (such as socket services, FTP server, Sysmac Studio, and FINS/TCP) for each connection number. If the keep-alive function is used and the remote node does not respond during the set keep-alive monitoring time, the connection is disconnected. The connection is left open if the power supply to the remote node is turned OFF without warning. Use the keep-alive function whenever possible.  • Use.	Use.
	Do not use.	
Keep Alive Monitoring Time	The connection is disconnected if the keep-alive function is used and the remote node does not respond during the monitoring time set in the <i>Keep Alive Monitoring Setting</i> .	300
	Setting range: 1 to 65,535 (seconds)	
Linger option	Set whether to specify the <i>Linger Option</i> for connections to FINS/TCP or socket services. If the <i>Linger Option</i> is specified, the port number is immediately opened even before the port number is released after the socket closes (approx. 1 minute).	Do not specify.
	Specify.	
	Do not specify.	

## Default Gateway (NX-series CPU Unit)

Setting	Description	Default
Default gateway*1	Set the IP address of the default gateway for the built-in EtherNet/IP port. This setting is not required when the default gateway is not used.*2	None
	When the IP address settings are obtained from a BOOTP server, the default gateway set in this item is disabled.	

<sup>\*1</sup> This setting is only valid when the IP address setting method for either port 1 or port 2 is set to Fixed setting.

#### • IP Router Table

Parameter	Description	Default
Destination IP address	Set these settings when the built-in EtherNet/IP port is used	None
Destination mask IP address	for tag data links or CIP message communications with nodes on other IP network segments via an IP router. Also set these settings to use an NX-series CPU Unit as an IP router using the IP routing function of the built-in Ether-Net/IP port.	
Gateway address		None
	You can set up to 256 combinations of IP addresses and gateway addresses with an NX-series CPU Unit, and up to eight combinations with an NJ-series CPU Unit.  Specify 0 for the host portions of the IP addresses.	

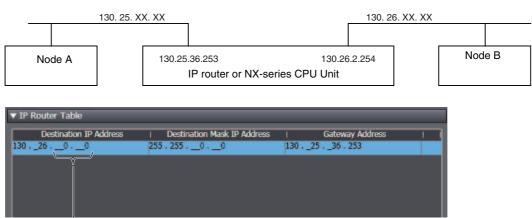
<sup>\*2</sup> Even if you select Fixed setting for both port 1 and port 2, you can only set the default gateway for one of the ports.



#### **Additional Information**

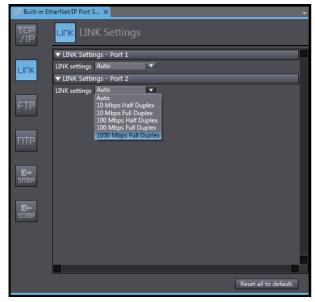
#### **IP Router Table Setting Example**

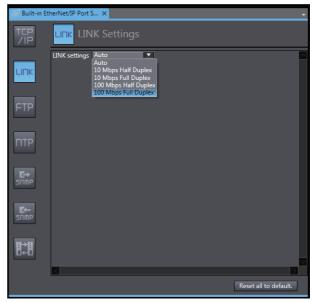
Set the following IP router table in node A to use tag data links or CIP message communications between node A and node B through the IP router. When you set the IP router table, node A sends packets to the gateway IP address (130.25.36.253) if communications instructions are executed on node A and addressed to node B.



The host fields are set to 0 in the destination IP address.

### **Link Settings Display 5-2**





**NX-series CPU Unit** 

NJ-series CPU Unit

# ● LINK settings - Port 1 and Port 2 (NX-series CPU Unit)

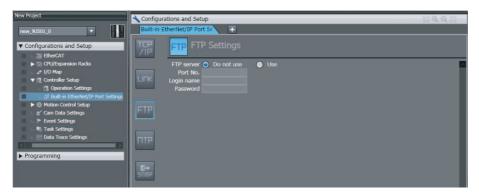
Set for each built-in EtherNet/IP port.

Setting	Description	Default
LINK settings	Set the baud rate for the built-in EtherNet/IP port.	Auto
	• Auto	
	10 Mbps Half Duplex	
	10 Mbps Full Duplex	
	100 Mbps Half Duplex	
	100 Mbps Full Duplex	
	1000 Mbps Full Duplex	

# • LINK settings (NJ-series CPU Unit)

Setting	Description	Default
LINK settings	Set the baud rate for the built-in EtherNet/IP port.	Auto
	Auto	
	10 Mbps Half Duplex	
	10 Mbps Full Duplex	
	100 Mbps Half Duplex	
	100 Mbps Full Duplex	

# 5-3 FTP Display



Setting	Description	Default
FTP server	Specify whether to use the FTP server. FTP connections from external devices will not be possible if the <i>Do not use</i> Option is selected.	Do not use.
Port No.*1*3	Set the FTP port number of the built-in EtherNet/IP port. It is normally not necessary to change this setting. The FTP control port is set here. The FTP data transfer port is always port 20.	21
Login name*1	Set the login name to externally connect to the built-in EtherNet/IP port via FTP. (You can use up to 12 alphanumeric characters.)*2	None
Password*1	Set the password to externally connect to the built-in Ether-Net/IP port via FTP. (You can use 8 to 32 alphanumeric characters.)*2	None

<sup>\*1</sup> These settings are required to use the FTP server.

<sup>\*3</sup> The following ports are used by the system and cannot be set by the user: 20, 23, 25, 80, 110, 9610, and 44818.



#### **Additional Information**

Refer to Section 10 FTP Server for details on the FTP server.

<sup>\*2</sup> The login name and password are case sensitive.

# **NTP Settings Display**



Setting	Description	Default
NTP server clock information	Set whether to obtain clock information from the NTP server to update the clock in the CPU Unit.	Do not get.
Port No.*1*2	Set the port number to use to connect to the NTP server to obtain clock information. It is normally not necessary to change this setting.	123
Server specifying method*1	Set the method to use to specify the NTP server to obtain clock information.  • IP address	IP address
IP address	Host name  Set the IP address of the NTP server. (Set this setting if the server specifying method is set to the <i>IP address</i> Option.)	None
Host name	Set the host name of the NTP server (i.e., the domain name of the host). (Set this setting if server specifying method is set to the <i>Host name</i> Option.) (You can use up to 200 single-byte alphanumeric characters, dots, and hyphens with up to 63 single-byte alphanumeric characters between dots.)	None
NTP operation timing*1	Set the time at which the NTP server is accessed to synchronize the clocks.  • Specify a clock time.	Specify a time.
	Specify a time interval.	
Time [hours:minutes:sec- onds]	The NTP server is accessed at the specified time. (Setting range: 00:00:00 to 23:59:59) (Set this setting if NTP operation timing is set to the <i>Specify a clock time</i> Option.)	00:00:00
Interval [minutes]	The NTP server is accessed when the specified period of time has passed. (Setting range: 1 to 1,440 minutes) (Set this setting if NTP operation timing is set to the Specify a time interval Option.)	60 min
Timeout time (seconds)*1	Set the timeout detection time. (Setting range: 1 to 255 seconds)	10 seconds
	If the remote host does not respond, retry processing is performed four times within the time interval that is set here. If a time interval is specified for NTP operation, timing for the next execution of the NTP operation starts when the fourth retry processing times out.	
*1 This patting is used in all to	let the cleak information from the NTD corver	I

<sup>\*1</sup> This setting is required to get the clock information from the NTP server.

<sup>\*2</sup> The following ports are used by the system and cannot be set by the user: 25, 53, 68, 110, 2222, 2223, 2224, 9600, and 44818.



# **Additional Information**

Refer to *Section 12 Automatic Clock Adjustment* for details on obtaining clock information from the NTP server.

### **SNMP Settings Display 5-5**



#### SNMP

Setting	Description	Default
SNMP	Specify whether to use the SNMP monitor service. If not using the SNMP monitor service is specified, an SNMP manager cannot connect from an external device.	Do not use.
Port No.*1	Set the port number to use to connect to the SMTP server that is used to connect from an SNMP manager. This setting does not normally need to be changed.	161
Address	and installation location as taxt information. You do not	None
Location		None
Send a recognition trap	Set whether to send an authentication trap. If you select <i>Send a recognition trap</i> and there is access from an SNMP manager that is not set in Recognition 1 or Recognition 2, an authentication trap is sent to the SNMP manager. If you select <i>Send a recognition trap</i> , specify the SNMP trap settings on the SNMP Trap Tab Page.	Do not use.

<sup>\*1</sup> The following ports are used by the system and cannot be set by the user: 25, 53, 68, 110, 2222, 2223, 2224, 9600, and 44818.



#### **Additional Information**

Refer to Section 13 SNMP Agent for details on the SNMP service.

# • Recognition 1\*1

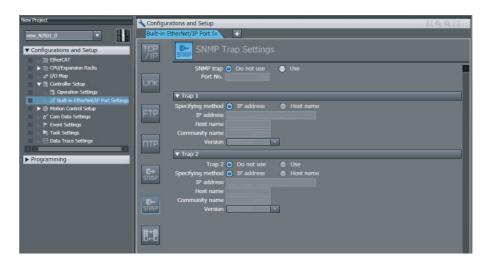
Setting	Description	Default
Recognition method	Set the method to use to specify SNMP managers for which access is permitted.	IP address
	IP address	
	Host name	
	Make these settings to permit access by only certain SNMP managers. Access is not allowed unless an IP address or host name is set.	
IP address	Set the IP address of the SNMP manager. If the default setting of 0.0.0.0 is used, access is permitted from all SNMP managers. (Set this setting if the recognition method in the recognition 1 settings is set to the <i>IP address</i> Option.)	None
Host name	Set the host name of the SNMP manager. (Set this setting if the recognition method in the recognition 1 settings is set to the <i>Host name</i> Option.) (You can use up to 200 single-byte alphanumeric characters, dots, and hyphens with up to 63 single-byte alphanumeric characters between dots.)	None
Community name	Set the community name to enable the SNMP manager to access information from the built-in EtherNet/IP port. (Single-byte alphanumeric characters, dots, and hyphens: 255 characters max.)	public

# ● Recognition 2\*1

Setting	Description	Default
Recognition 2	Specify whether to use the recognition 2 settings.  • Use.  • Do not use.	Do not use.
Recognition method	Set the method to use to specify SNMP managers for which access is permitted.	IP address
	<ul><li>IP address</li><li>Host name</li></ul>	
	Make these settings to permit access by only certain SNMP managers. Access is not allowed unless an IP address or host name is set.	
IP address	Set the IP address of the SNMP manager. If the default setting of 0.0.0.0 is used, access is permitted from all SNMP managers. (Set this setting if the recognition method in the recognition 2 settings is set to the <i>IP address</i> Option.)	None
Host name	Set the host name of the SNMP manager. (Set this setting if the recognition method in the recognition 2 settings is set to the <i>Host name</i> Option.) (You can use up to 200 single-byte alphanumeric characters, dots, and hyphens with up to 63 single-byte alphanumeric characters between dots.)	None
Community name	Set the community name to enable the SNMP manager to access information from the built-in EtherNet/IP port. (Single-byte alphanumeric characters, dots, and hyphens: 255 characters max.)	public

<sup>\*1</sup> These settings are required if the SNMP service is enabled.

### **SNMP Trap Settings Display 5-6**



# SNMP Trap

Setting	Description	Default
SNMP trap  Specify whether to use the SNMP trap (network error detection). If the SNMP trap service is not used, SNMP traps are usent to the SNMP manager.		Do not use.
Port No.*1	Set the port number to use to connect to the SNMP server. It is normally not necessary to change this setting.	162

<sup>\*1</sup> The following ports are used by the system and cannot be set by the user: 25, 53, 68, 110, 2222, 2223, 2224, 9600, and 44818.



#### **Additional Information**

Refer to 13-1-1 Overview for details on the SNMP trap.

# ● Trap 1\*1

Setting	Description	Default
Specifying method	Set the specifying method for the SNMP manager destination for SNMP traps.	IP address
	IP address	
	Host name	
IP address	Set the IP address of the SNMP manager. (Set this setting if the specifying method in the trap 1 settings is set to the <i>IP address</i> Option.)	
Host name	Set the host name of the SNMP manager. (Set this setting if the specifying method in the trap 1 settings is set to the <i>Host name</i> Option.) (You can use up to 200 single-byte alphanumeric characters, dots, and hyphens with up to 63 single-byte alphanumeric characters between dots.)	None
Community name	Set the community name. (You can use up to 255 single-byte alphanumeric characters.)	public
Version	Set the version of the SNMP manager.	SNMPv1
	SNMP version 1: SNMPv1	
	SNMP version 2C: SNMPv2C	

# ● Trap 2\*1

Setting	Description	Default
Trap 2	Specify whether to use the trap 2 settings.	Do not use.
	• Use.	
	Do not use.	
Specifying method	Set the specifying method for the SNMP manager destination for SNMP traps.	IP address
	IP address	
	Host name	
IP address	Set the IP address of the SNMP manager. (Set this setting if the specifying method in the trap 2 settings is set to the <i>IP address</i> Option.)	None
Host name  Set the host name of the SNMP manager. (Set this setting if the specifying method in the trap 2 settings is set to the <i>Host name</i> Option.) (You can use up to 200 single-byte alphanumeric characters, dots, and hyphens with up to 63 single-byte alphanumeric characters between dots.)		None
Community name	Set the community name. (You can use up to 255 single-byte alphanumeric characters.)	public
Version	Set the version of the SNMP manager.	SNMPv1
	SNMP version 1: SNMPv1	
	SNMP version 2C: SNMPv2C	

<sup>\*1</sup> These settings are required if the SNMP trap is set is enabled.



# **Testing Communications**

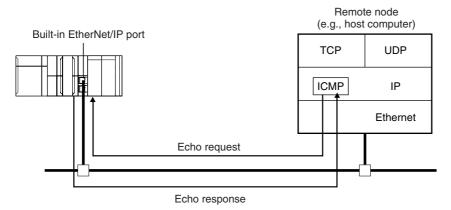
6-1	Testing	g Communications	6-2
	6-1-1	PING Command	6-2
	6-1-2	Using the PING Command	6-2
	6-1-3	Host Computer Operation	. 6-3

#### 6-1 **Testing Communications**

If the basic settings (in particular the IP address and subnet mask) have been made correctly for the built-in EtherNet/IP port, then it is possible to communicate with nodes on the EtherNet/IP network. This section describes how to use the PING command to test communications with the built-in EtherNet/IP

#### 6-1-1 **PING Command**

The PING command sends an echo request packet to a remote node and receives an echo response packet to confirm that the remote node communications are normal. The PING command uses the ICMP echo request and responses. The echo response packet is automatically returned in the ICMP. The PING command is normally used to check the connections of remote nodes when you set up a network. The built-in EtherNet/ IP port supports both the ICMP echo request and response functions. If the remote node returns a normal response to the PING command, then the nodes are physically connected correctly and Ethernet node settings are correct.



#### 6-1-2 **Using the PING Command**

The built-in EtherNet/IP port automatically returns the echo response packet in response to an echo request packet sent by another node (e.g., host computer).

## 6-1-3 Host Computer Operation

The PING command can be executed from the host computer to send an echo request packet to a built-in EtherNet/IP port. The following example shows how to use the PING command in the host computer.

# **Application Method**

Input the following command at the host computer's prompt (\$):

```
$ ping IP_address (host_name)
```

The destination is specified by its IP address or host name.



#### **Additional Information**

The PING command is not supported by some host computers.

# **Application Example**

In this example, a PING command is sent to the node at IP address 130.25.36.8. The "\$" in the example represents the host computer prompt.

#### Normal Execution

```
$ ping 130.25.36.8

PING 130.25.36.8: 56 data bytes
64 bytes from 130.25.36.8: icmp_seq=0. time=0. ms
64 bytes from 130.25.36.8: icmp_seq=0. time=0. ms
∴ ∴ ∴ ∴ ∴ ∴

64 bytes from 130.25.36.8: icmp_seq=0. time=0. ms
← Press the Ctrl+C Keys to cancel execution.

---- 130.25.36.8 PING Statistics ----
9 packets transmitted, 9 packets received, 0% packets loss
round-trip (ms) min/avg/max = 0/1/16
$
```

#### Error

```
$ ping 130.25.36.8

PING 130.25.36.8: 56 data bytes

---- 130.25.36.8 PING Statistics ----
9 packets transmitted, 0 packets received, 100% packets loss

$ Executes the PING command.

Press the Ctrl+C Keys to cancel execution.
```

Refer to the command reference manual for your computer's OS for details on using the PING command.



# **Tag Data Link Functions**

7-1	Introdu	uction to Tag Data Links	7-2
	7-1-1	Tag Data Links	7-2
	7-1-2	Data Link Data Areas	7-3
	7-1-3	Tag Data Link Functions and Specifications	7-6
	7-1-4	Overview of Operation	7-7
	7-1-5	Starting and Stopping Tag Data Links	7-9
	7-1-6	Controller Status	7-10
	7-1-7	Concurrency of Tag Data Link Data	7-13
7-2	Setting	g Tag Data Links	7-18
	7-2-1	Starting the Network Configurator	7-18
	7-2-2	Tag Data Link Setting Procedure	7-20
	7-2-3	Registering Devices	7-21
	7-2-4	Creating Tags and Tag Sets	7-23
	7-2-5	Connection Settings	7-36
	7-2-6	Creating Connections Using the Wizard	7-46
	7-2-7	Creating Connections by Device Dragging and Dropping	7-49
	7-2-8	Connecting the Network Configurator to the Network	7-52
	7-2-9	Downloading Tag Data Link Parameters	7-57
	7-2-10	Uploading Tag Data Link Parameters	7-60
	7-2-11	Verifying the Tag Data Links	7-63
	7-2-12	Starting and Stopping Tag Data Links	7-67
	7-2-13	Clearing the Device Parameters	
	7-2-14	Saving the Network Configuration File	7-71
	7-2-15	Reading a Network Configuration File	
	7-2-16	Checking Connections	7-74
	7-2-17	Changing Devices	
	7-2-18	Displaying Device Status	7-76
7-3	Ladde	r Programming for Tag Data Links	7-77
	7-3-1	Ladder Programming for Tag Data Links	7-77
	7-3-2	Status Flags Related to Tag Data Links	7-81
7-4	Tag Da	ata Links with Other Models	7-82

# **Introduction to Tag Data Links**

#### 7-1-1 Tag Data Links

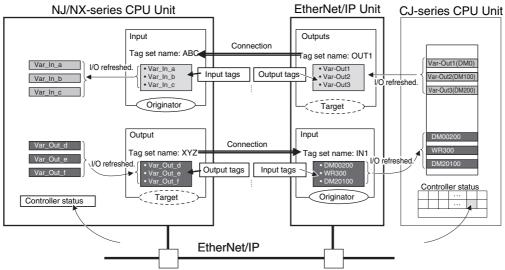
Tag data links enable cyclic tag data exchanges on an EtherNet/IP network between Controllers or between Controllers and other devices. Variables are assigned to tags. (You can also assign I/O memory addresses to tags.) The settings for tag data links are made with the Network Configurator. Refer to 7-2 Setting Tag Data Links for information on how to make the settings.



#### **Additional Information**

You can also use the Sysmac Studio to set the tag data links. Refer to A-2 Use the Sysmac Studio to Set the Tag Data Links (EtherNet/IP Connections) for details on setting the tag data links in the Sysmac Studio.

With a tag data link, one node requests the connection of a communications line to exchange data with another node. The node that requests the connection is called the originator, and the node that receives the request is called the target.



For communications between Controllers, the connection information is set in the built-in EtherNet/IP port of the Controller that receives (consumes) the data (i.e., the originator).



#### **Additional Information**

For communications between a Controller and an I/O device, the connection information is set in the built-in EtherNet/IP port that is the originator. If an I/O device is used, the Network Configurator must have an EDS file installed that includes connection information for the I/O device. Refer to A-3 EDS File Management for the installation procedure.

The output words and input words for each node for which data is exchanged are set in the connection information. These words are called the output tag set and input tag set. A tag set must specify at least one tag. The size of the data for data exchange is the total size of the tags included in the tag set. The size of the output tag set and the size of the input tag set must match.

#### 7-1-2 Data Link Data Areas

# **Tags**

A tag is a unit that is used to exchange data with tag data links. Data is exchanged between the local network variables and remote network variables specified in the tags or between specified I/O memory areas.



#### **Precautions for Correct Use**

To maintain concurrency in the values of network variables that are assigned to tags, you must set refreshing tasks. Refer to 7-1-7 Concurrency of Tag Data Link Data for details.

# **Tag Sets**

When a connection is established, from 1 to 8 tags (including Controller status) is configured as a tag set. Each tag set represents the data that is linked for a tag data link connection. Tag data links are therefore created through a connection between one tag set and another tag set. A tag set name must be set for each tag set.

**Note** A connection is used to exchange data as a unit within which data concurrency is maintained. Thus, data concurrency is maintained for all the data exchanged for the tags in one data set.

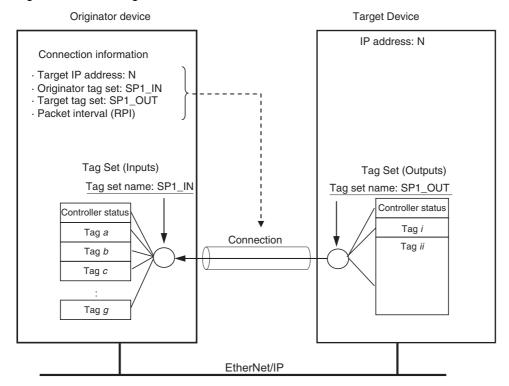


#### **Precautions for Correct Use**

Data for the tags is exchanged in the order that the tags are registered in the tag sets. Register the tags in the same order in the input and output tag sets.

#### Example

In the following example, input tags a to g at the originator are a tag set named SP1\_IN and output tags *i* and *ii* are a tag set named *SP1\_OUT*. A connection is set between these two tag sets.



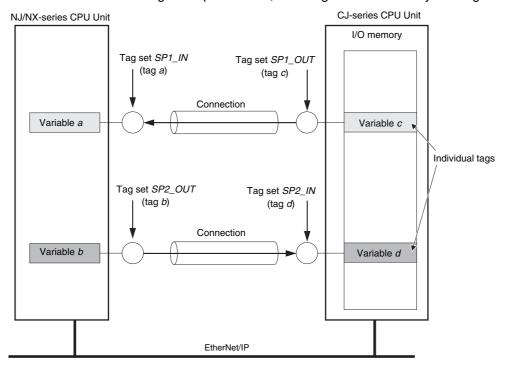
There are both input (consume) and output (produce) tag sets. Each tag set can contain only input tags or only output tags. The same input tag cannot be included in more than one input tag set.

#### Number of Tags in Tag Sets

You can set any tag sets containing one or more tags for the input and output tag sets for one connection. For example, you can set a tag set with one tag for the input tag set and set a tag set with more than one tag for the output tag set.

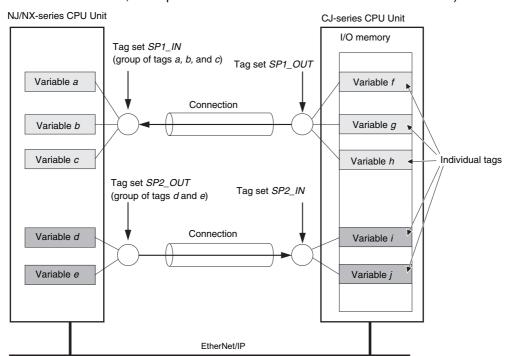
#### Tag Sets with Only One Tag Each

With basic Network Configurator procedures, each tag set contains only one tag.



#### Tag Sets Each with Multiple Tags

As shown below, tags can be grouped. You can place up to eight tags (up to 722 words in total for an NX-series CPU Unit, and up to 300 words in total for an NJ-series CPU Unit) in one tag set.



Note To enable a connection, each tag set must include only input tags or only output tags. (Both input and output tags cannot be included in the same tag set.)

#### 7-1-3 **Tag Data Link Functions and Specifications**

The tag data link and performance specifications of the NJ/NX-series CPU Unit are given below.

Item		Specification		
		NX701-□□□□	NJ501-□□□□/NJ301-□□□□/NJ101-□□□	
		Unit version 1.10 or later	Unit version 1.00 to 1.02	Unit version 1.03 or later
Communication	ns type	Standard EtherNet/IP implicit communications (connection-type cyclic communications)		
Setting method		After you have set the tags, tag sets, and connections with the Network Configurator, you must download the tag data link parameters to all devices on the EtherNet/IP network.  You can export the network variables that you created on the Sysmac Studio to a CSV file. You can then import the file to the Network Configurator and assign the network variables to tags.  After the parameters are downloaded, the EtherNet/IP Units are restarted to start the tag data links.		
Tags*1	Supported variable types	You can specify the following Global variables	ing network variables as tag	gs. *2, *3
	Maximum number of words per tag	722 words (1444 bytes)	300 words (600 bytes)	
	Maximum number of tags	256 (total of 512 with two ports)	256*4	
Tag sets	Maximum number of tags per tag set	8 (7 when Controller statu	s is included)	
	Maximum number of words per tag set	722 words (1444 bytes)	300 words (600 bytes)	
	Maximum number of tag sets	256 (total of 512 with two ports)	32	
Connections		Maximum number of connections per unit: 256	Maximum number of connections per Unit: 32	
		(total of 512 with two ports)		
Connection type		Each connection can be set for 1-to-1 (unicast) or 1-to-N (multi-cast) communications.		
Packet interval (RPI)		0.5 to 10,000 ms in 0.5-ms increments	10 to 10,000 ms in 1-ms increments	1 to 10,000 ms in 1-ms increments
		The packet interval can be set separately for each connection.		
Permissible communications band (pps)		40,000 pps *5	1,000 pps	3,000 pps
		Note The heartbeat is included.		

<sup>\*1</sup> To specify a specific I/O memory address for a tag, create a variable, use an AT specification of the I/O memory address on the Sysmac Studio, and then specify the variable with the AT specification for the tag. (This applies only for the NJ-series CPU Units.)

<sup>\*2</sup> You can import network variables created in the Sysmac Studio to the Network Configurator as tags. However, variables with a Network Publish attribute that have variable names that are the same as the I/O memory address notation, such as "0000" and "H0000" are not exported to CSV files.

\*3 The following table lists the variables that you can specify as tags.

Туре		Example	Specification
Variables with basic data types		aaa	Supported.
Enumerated variables		bbb	Supported.
Array variables	Arrays	ccc	Supported.
	Elements	ccc[2]	Supported.
Structure variables	Structures	ddd	Supported.
Structure variables	Members	ddd.xxx	Supported.
Union variables	Unions	eee	Supported.
Official variables	Members	еее.ууу	Supported.

- \*4 The maximum number of tags is given for the following conditions.
  - · All tag sets contain eight tags.
  - The maximum number of tag sets (32) is registered.
- \*5 If the two built-in EtherNet/IP ports are used simultaneously, the maximum communications data size for two ports in total will be reached.

## 7-1-4 Overview of Operation

In this manual, the connection information that is set is called tag data link parameters. This section describes how to set tag data links with the Sysmac Studio and the Network Configurator.

# **Setting Network Variables (Sysmac Studio)**

First, create any variables that you want to use for tag data links as network variables in the Sysmac Studio.

- Set the Network Publish attribute to Input or Output in the Global Variable Table for the variables you want to use for tag data links (i.e., as tags).
- **2** To maintain concurrency in tag data within a tag set, set all tags (i.e., variables with a Network Publish attribute) within the same tag set as follows:

Set a refreshing task for variables with a Network Publish attribute to maintain concurrency as described below for tag data link data.\*

Refer to 7-1-7 Concurrency of Tag Data Link Data for details on the concurrency of tag data link data.

- · Maintain concurrency in the tag data in a tag set.
- The timing of updating network variables that are assigned to tags is synchronized with the execution period of the program that accesses the network variables.
  - \* If a variable that uses an AT specification is used as a tag, you do not need to set a refreshing task. It is refreshed in the primary periodic task.



#### **Precautions for Correct Use**

You cannot use the following notation, which specifies an I/O memory address, in the variable name of any variable used in a tag data link.

- 1) Variable names that contain only single-byte numerals (Example: 001)
- 2) Variable names with the following single-byte letters (uppercase or lowercase) followed by single-byte numerals
  - H (Example: H30)
  - W (Example: w30)
  - D (Example: D100)
  - E0 to E18

# Setting and Downloading Tag Data Link Parameters (Network Configurator or Sysmac Studio)

The tag data link parameters (e.g., connection information) that are described below are created with the Network Configurator or the Sysmac Studio, and then the parameters are downloaded to all originator devices on the EtherNet/IP network. When the tag data links are used on the built-in EtherNet/IP port, use the Network Configurator to make the following settings.

Creating the Configuration Information

Register the EtherNet/IP ports and EtherNet/IP Units to create the connections that define the tag data links. Refer to 7-2-3 Registering Devices for details.

2 Setting Tags

Create CPU Unit variables for input (consume) tags and output (produce) tags. You can create up to 256 tags for a built-in EtherNet/IP port on an NJ/NX-series CPU Unit. (The maximum data size for one tag is 1,444 bytes or 722 words for an NX-series CPU Unit, and 600 bytes or 300 words for an NJ-series CPU Unit.) You can import and export network variables that are created on the Sysmac Studio to CSV files. This allows you to register them as tags on the Network Configurator. Output tags can be defined to clear output data to 0 or to hold the output data from before the error when a fatal error occurs in the CPU Unit.

3 Setting Tag Sets

Create output tag sets and input tag sets and assign tags to them. (You can create up to eight I/O tag sets.) You can create a maximum of 256 tag sets for the built-in EtherNet/IP port on an NX-series CPU Unit, and a maximum of 32 tag sets for the built-in EtherNet/IP port on an NJseries CPU Unit. (The maximum data size of one tag for an NX-series CPU Unit is 1,444 bytes (722 words) or 600 bytes (300 words).) You can specify the Controller status that indicates the CPU Unit's operating status (operating information and error information) in a tag set.

**4** Setting Connections

Link the output tag sets for the target device and the input tag sets for the originator device as connections.

## **Connection Setting Parameters**

The connection settings in step 4 above have the following setting parameters.

#### **Setting the Requested Packet Interval (RPI)**

The RPI is the I/O data refresh cycle on the Ethernet line when performing tag data links. The RPI can be set separately for each connection. You can set the RPI for the built-in EtherNet/IP port in the following ranges:

- NX-series CPU Unit: 0.5 to 10,000 ms in 0.5-ms increments. The default setting is 50 ms.
- NJ-series CPU Unit: 1 to 10,000 ms\* in 1-ms increments. The default setting is 50 ms.

With EtherNet/IP, data is exchanged on the communications line at the packet interval that is set for each connection, regardless of the number of nodes.

\* For unit version 1.03 or later, you can set the RPI to between 1 and 10,000 ms. For unit version 1.00 to 1.02, you can set the RPI to between 10 and 10,000 ms.

#### **Setting Multi-cast and Unicast Communications**

You can select a multi-cast connection or unicast (point-to-point) connection as the connection type in the tag data link connection settings. With a multi-cast connection, you can send an output tag set in one packet to multiple nodes and make allocations to the input tag sets. A unicast connection separately sends one output tag set to each node, and so it sends the same number of packets as the number of input tag sets. Therefore, multi-cast connections can decrease the communications load if one output tag set is sent to multiple nodes. If multi-cast connections are used, however, use an Ethernet switch that has multi-cast filtering, unless the tag set is received by all nodes in the network. If an Ethernet switch without multi-cast filtering is used, the multi-cast packets are broadcast to the entire network, and so packets are sent to nodes that do not require them, which will cause the communications load on those nodes to increase. To use a multi-cast connection and send an output tag set in one packet to multiple nodes, the following settings for the receiving node must be the same as the settings of the sending node: the connection type (multi-cast), the connection I/O types, packet internals (RPI), and timeout values.



#### **Precautions for Correct Use**

- The performance of communications devices is limited to some extent by the limitations of each product's specifications. Consequently, there are limits to the packet interval (RPI) settings. Refer to 14-2 Adjusting the Communications Load and set an appropriate packet interval (RPI).
- If you will use data tag links with multicast at a communication speed over 100 Mbps, use an Ethernet switch with a multicast filter or an Ethernet switch that supports a communication speed of 1000 Mbps.
  - If there is an Ethernet device on the same network that communicates at a speed of 100 Mbit/sec or less, the device may affect tag data link communication and cause tag data links to be broken, even if the device is not related to tag data link communication.



#### **Additional Information**

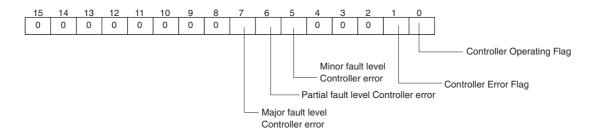
- A maximum of 256 connections can be opened for an NX-series CPU Unit, and a maximum of 32 connections can be opened for an NJ-series CPU Unit. To calculate the number of connections of each connection type, refer to 14-1-2 Calculating the Number of Connections.
- If the maximum number of connections is exceeded, you must review the number of connections for the built-in EtherNet/IP port, or the number of nodes. When you use an NJ-series CPU Unit, you can also consider adding EtherNet/IP Units.

# 7-1-5 Starting and Stopping Tag Data Links

Tag data links are automatically started when the data link parameters are downloaded from the Network Configurator and the power supply to the NJ/NX-series Controller is turned ON. Thereafter, you can start and stop tag data links for the entire network or individual devices from the Network Configurator. Starting and stopping tag data links for individual devices must be performed for the originator. Furthermore, you can use system-defined variables to start and stop the entire network. Refer to 7-2-12 Starting and Stopping Tag Data Links for details.

#### 7-1-6 **Controller Status**

You can include the Controller status as a member of a tag set in the data sent and received. The Controller status is a set of flags that indicate the operating status of the CPU Unit (operating information, error information, Controller error level). If the Controller status is specified as an output (send) tag, the Controller status is added to the start of the tag set in the following format. (Select the Include Option for the Controller Status in the upper right of the Edit Tag Set Dialog Box.)



\* Of the flags in bits 5 to 7 that indicate the current error level, only the flag for the highest error level changes to TRUE. For example, if a minor fault level Controller error and a major fault level Controller error occur at the same time, only the flag for the major fault level Controller error (bit 7) will change to TRUE and the flag for the minor fault level Controller error (bit 5) will remain FALSE.

To receive the Controller status, specify the Controller status for the In - Consume Tab Page in the dialog box used to edit the receive tag set. (Select the Include Option for the Controller Status in the upper right of the Edit Tag Set Dialog Box.) When a tag data link is started, the contents of the Controller status is stored in the system variables that are given below.

Target PLC Operating Mode

NX-series CPU Unit: \_EIP1\_TargetPLCModeSta (for built-in EtherNet/IP port 1)

\_EIP2\_TargetPLCModeSta (for built-in EtherNet/IP port 2)

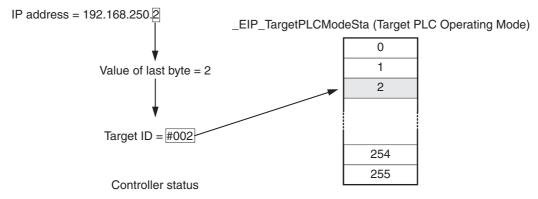
\_EIP\_TargetPLCModeSta NJ-series CPU Unit:

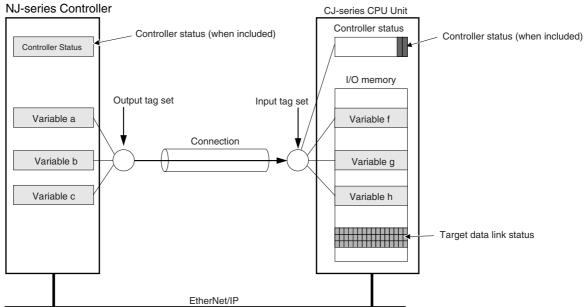
Target PLC Error Information

NX-series CPU Unit: EIP1 TargetPLCErr (for built-in EtherNet/IP port 1)

\_EIP2\_TargetPLCErr (for built-in EtherNet/IP port 2)

NJ-series CPU Unit: \_EIP\_TargetPLCErr Example: Using an NJ-series CPU Unit to send the Target PLC Operating Mode of the Target Node with an IP Address of 192.168.250.2







#### **Additional Information**

The target ID may be duplicated depending on the IP addresses of the target nodes. In this case, it is necessary to change the target ID with the Network Configurator so that the same address is not used twice. For information on how to change the target node ID, refer to step 4 under 7-2-5 Connection Settings.

When you use multiple connections to communicate with one specific node, the information in the Controller status is stored in the following variables if the Controller status is specified in the input tags and the output tags for all connections.

Controller status	Variable name	Description of operation
Controller Operating Flag	Target PLC Operating Mode	This flag shows the operation information of the Controller at the target node.
	NX-series CPU Unit:     _EIP1_TargetPLCMode     Sta (for built-in Ether-     Net/IP port 1), or     _EIP2_TargetPLCMode     Sta (for built-in Ether-     Net/IP port 2)      NJ-series CPU Unit:     EIP_TargetPLCModeSta	When the Built-in EtherNet/IP Port Is the Originator of the Connection  The array element that corresponds to the target ID at the target is TRUE when all information for all connections of the relevant target node is shows operating status. You can change the target ID of the IP address from the Network Configurator. This status information is enabled when the Controller status is included in the communications data in both the originator and target node. This variable is updated when necessary.
Controller Error Flag	Target PLC Error Information  NX-series CPU Unit: _EIP1_TargetPLCErr (for built-in EtherNet/IP port 1), or _EIP2_TargetPLCErr (for built-in EtherNet/IP port 2)  NJ-series CPU Unit: _EIP_TargetPLCErr	This variable shows the error status (logical OR of fatal and nonfatal errors) of the target node Controllers.  When the Built-in EtherNet/IP Port Is the Originator of the Connection  You can change the target ID of the IP address from the Network Configurator. The Controller Error Flags are enabled when the Controller status is included in the communications data for both the originator and target. This variable is updated when necessary.



#### **Additional Information**

Even if you specify including the Controller status in the output (produce) tags, you do not necessarily need to include it in the input (consume) tags. If you do not include the Controller status in an input (consume) tag, the contents of the Controller status is not updated in the Target PLC Operating Mode and Target PLC Error Information variables, but it is sent in the input (consume) tag. Therefore, you can use the Controller status data that was received in the input (consume) tag as receive data.

## 7-1-7 Concurrency of Tag Data Link Data

To maintain the concurrency of data in a tag data link, you must set a refreshing task for each network variable that is assigned to a tag.

- Maintain concurrency in the tag data in a tag set.
- The timing of updating network variables that are assigned to tags is synchronized with the execution period of the program that accesses the network variables.



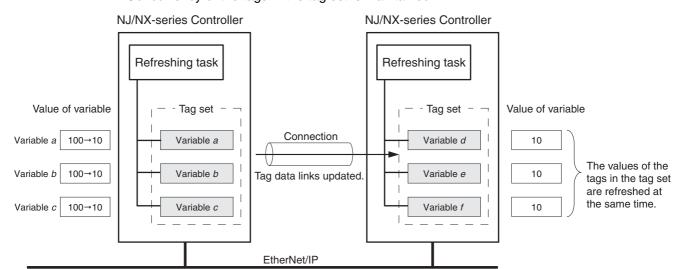
#### **Additional Information**

A refreshing task maintains concurrency of the value of a global variable from all tasks that access that global variable. This is achieved by specifying a single task that can write to that global variable and not allowing any other task to write to that global variable. For details on refreshing tasks, refer to the *NJ/NX-series CPU Unit Software User's Manual* (Cat. No. W501).

### Maintaining Concurrency in the Tag Data in a Tag Set

To maintain concurrency in the values of multiple tags in a tag set, the tags (variables with a Network Publish attribute) must satisfy the following four conditions.

- (1) The tags must be assigned to the same tag set (connection).
- (2) A refreshing task must be set for the network variables assigned to the tags, and the refreshing task must be the same for all tags in the tag set.\*
- (3) In the case of a NJ-series CPU Unit, a tag that uses AT specification must not be included in the same tag set.
- (4) The variable access time set for each task must be set to a higher value than is required to transfer the tag data. Refer to 14-3-3 Relationship between Task Periods and Packet Intervals (RPIs) for details on the variable access time and data transfer.
  - If you set a refreshing task for a network variable, you must set a variable access time to allocate enough time to access the network variable from outside of the Controller.
- Setting Refreshing Tasks for Tags (Network Variables) Concurrency of the tags in the tag set is maintained.

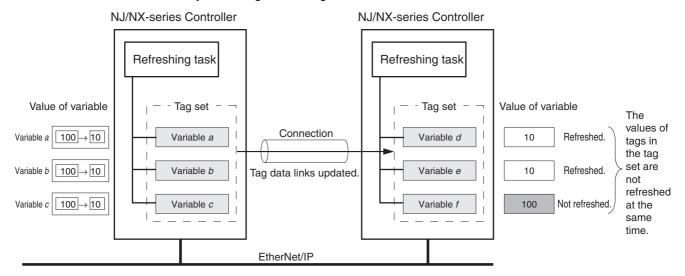




#### **Additional Information**

When using an NJ-series CPU Unit, you do not need to set a refreshing task for variables (tags) that use an AT specification because they are updated in the primary periodic task.

• Not Setting Refreshing Tasks for Tags (Network Variables) Concurrency of the tags in the tag set is not maintained.



## Synchronizing the Update Timing of Network Variables (Tags) with the User Program Execution Period

To have the values of network variables (tags) updated to the latest tag data values each time the user program that accesses those network variables is executed, set the refreshing task for the network variables (tags) to the same type of the task as for the user program that accesses the network variables (tags).

The difference between the operation of tags with a refreshing task that is the same as the user program and tags without a refreshing task is described below.

• Tag (network variable) with a refreshing task

The tag is refreshed each time the program with the task that is set as the refreshing task is executed.

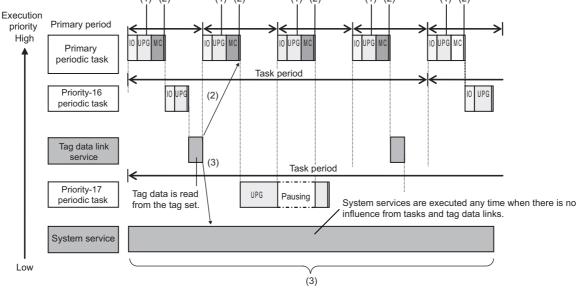
· Tag (network variable) without a refreshing task

The tag is refreshed by the system service.

Refreshing is not synchronized to the execution timing of the program.

#### • NX-series CPU Unit

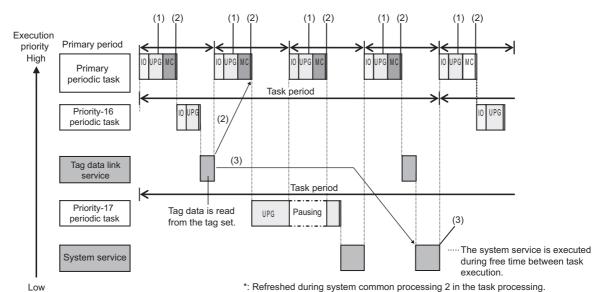
- (1) Execution timing of the program
- (2) Refresh timing of network variables (tags) with the primary periodic task\* set as the refreshing task\*
- (3) Refresh timing of network variables (tags) that do not have the primary periodic task set as the refreshing task



\*: Refreshed during system common processing 2 in the task processing.

#### • NJ-series CPU Unit

- (1) Execution timing of the program
- (2) Refresh timing of network variables (tags) with the primary periodic task\* set as the refreshing task\*
- (3) Refresh timing of network variables (tags) that do not have the primary periodic task set as the refreshing task





#### **Additional Information**

If a program needs to access a network variable with an AT specification, set the program in the primary periodic task so that it matches the refresh timing of the network variable that uses an AT specification. (This applies to the NJ-series CPU Unit only.)

and the input tags.



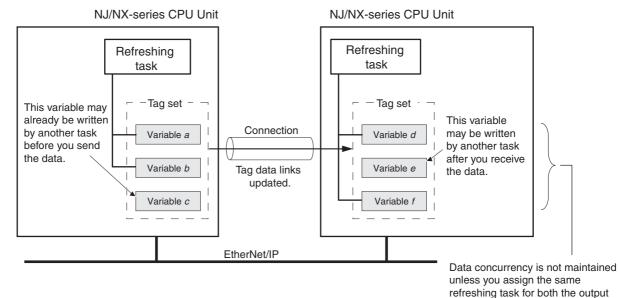
#### **Additional Information**

#### Relationship between Refreshing Tasks and Data Concurrency in Tag Data Links

If you do not specify a refreshing task for global variables in tag data links, the following may occur.

- 1. When the data is sent for the output tag set, another task may have already written different values before that data is sent, depending on the timing of the task.
- 2. When data is received by an input tag set, another task may write different values after that data is received, depending on the timing of the task.

Therefore, to maintain the concurrency of data in tag data links, you must specify the same refreshing task on both the output CPU Unit and the input CPU Unit.



# Required Processing Time to Maintain Concurrency

When you set a refreshing task for tags (network variables) to maintain the concurrency of data link data, the processing time required for that specified task increases. Due to this increase in task processing time, the refreshing of tag data link data may not occur during the packet interval (RPI) period set for each connection. Therefore, you need to adjust the packet interval (RPI) settings to match the period of the task specified as the refreshing task. Refer to 14-3-3 Relationship between Task Periods and Packet Intervals (RPIs) for details.

#### Task Setup Procedure

- (1) Set the global variables for which to specify a refreshing task, and set the refreshing tasks and accessing tasks in the Settings for Exclusive Control of Variables in Tasks in the Task Setup on the Sysmac Studio.
- (2) Set the variable access time for each refreshing task.

For details, refer to the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501).

#### 7-2 **Setting Tag Data Links**



### **Additional Information**

You can also use the Sysmac Studio to set the tag data links. Refer to A-2 Use the Sysmac Studio to Set the Tag Data Links (EtherNet/IP Connections) for details on setting the tag data links in the Sysmac Studio.

#### 7-2-1 Starting the Network Configurator

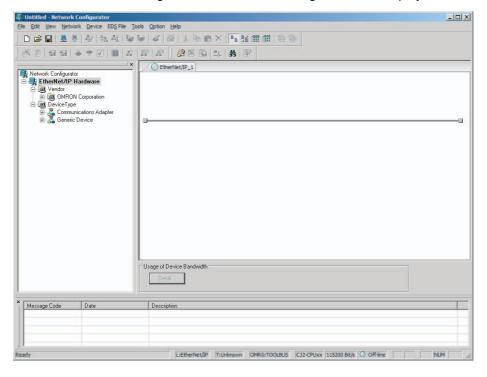
# **Procedure**

Tag data links are set from the Network Configurator. Use the following procedure to start the Network Configurator.

#### Using the Windows Start Menu

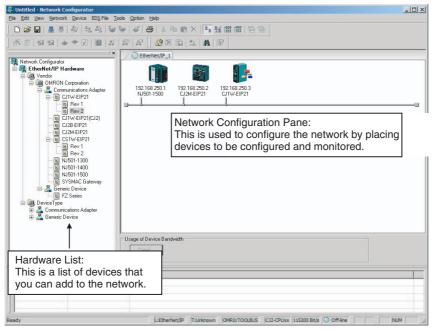
To start the Network configurator, select OMRON - Sysmac Studio - Network Configurator for **EtherNetIP** – **Network Configurator** from the Windows Start Menu.

When the Network Configurator starts, the following window is displayed.

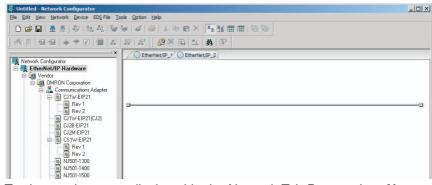


## **Main Window**

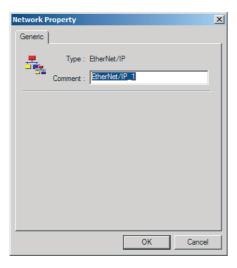
The Main Window consists of a Hardware List and a Network Configuration Pane, as shown in the following diagram.



To manage two or more networks, you can select **Network** – **Add** to add a new Network Configuration Pane. You can add a new Network Configuration Pane.



To change the name displayed in the Network Tab Page, select *Network – Property*. You can change the name set in the Comment Field of the Network Property Dialog Box.



#### 7-2-2 Tag Data Link Setting Procedure

This section describes the procedure to set tag data links (i.e., connection information). For data links between Controllers, the connection information is set only in the originator, i.e., the node that receives data.

- Create the network configuration.
  - (1) Register all built-in EtherNet/IP ports for which to create connections in the EtherNet/IP Network Configuration Pane. (Refer to 7-2-3 Registering Devices.)
    - \* If a system has already been installed, connect online to the EtherNet/IP network and upload the network configuration. (Refer to 7-2-10 Uploading Tag Data Link Parameters.)
- Create the tag and tag set connections.

Set the connections with one of the following methods.

- (1) Basic Operation:
  - 1. Create tags and tag sets for all registered devices (built-in EtherNet/IP ports). (Refer to 7-2-4 Creating Tags and Tag Sets.)
  - 2. Create a connection for the originator device (i.e., the registered device that receives data as input data). (Refer to 7-2-5 Connection Settings.)
- (2) Create the connections by dragging registered devices. (Refer to 7-2-10 Uploading Tag Data Link Parameters.)
- Download the tag data link parameters. (Refer to 7-2-9 Downloading Tag Data Link Parameters.)

1

- Make sure that the tag data links are operating normally by using the indicators for the built-in EtherNet/IP port (refer to 15-1 Overview of Errors) and the Network Configurator monitor functions. (Refer to 14-1 Communications System.)
- Make sure that the output tag data is updated in the input tags by using the Sysmac Studio's Watch Tab Page.

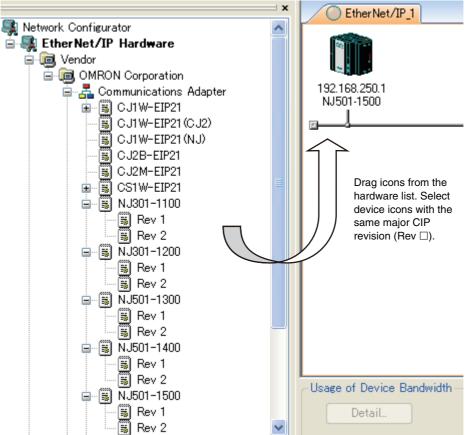
Note Refer to the Sysmac Studio Version 1 Operation Manual (Cat. No. W504) for the procedure.

# 7-2-3 Registering Devices

Register all of the devices required in the equipment (such as EtherNet/IP Units performing tag data links) in the network configuration.

1 Register the devices that will participate in the tag data links by dragging the devices from the Hardware List and dropping them in the Network Configuration Pane on the right. (To drag and drop an icon, click and hold the left mouse button over the icon, move the icon to the destination, and release the mouse button.)

You can also select a device in the Hardware List and press the **Enter** Key to register it. The icon of the device is displayed in the Network Configuration Pane, as shown in the following diagram.



The device names and major CIP revisions (Rev  $\Box$ ) are displayed in the hardware list. For NJ/NX-series CPU Units, device names and major CIP revisions are as shown in the following table.

Device name in	Unit version	CIP revisions	
hardware list		Major revision	Revision name in hardware list
NX701	Unit version 1.10 or later	2	None
NJ501-□□□□	Unit version 1.00 to 1.02	1	Rev 1
	Unit version 1.03 or later	2	Rev 2
NJ301-□□□□	Unit version 1.01 or 1.02	1	Rev 1
	Unit version 1.03 or later	2	Rev 2
NJ101	Unit version 1.10 or later	2	None



#### **Precautions for Correct Use**

Make sure that you select the devices with the same device names and the same major CIP revisions as the devices that you use in actual operation. The following will occur if any device names or CIP revisions are different when you attempt to download tag data link parameters on the Network Configurator.

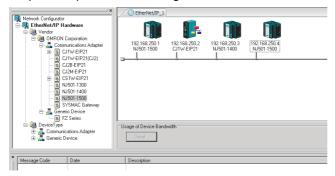
- · If a device name is different, an error message "Specified device can not be accessed, or wrong device type." will be displayed and the download will fail.
- If a revision is different, "Wrong unit revision." will be displayed and the download will fail.

The above also applies when uploading or comparing tag data link parameters. In any of the above cases, refer to 7-2-17 Changing Devices and change the device.

Right-click the registered device's icon to display the pop-up menu, and select Change UNKNOWN Address.



- Set the IP address to match the node address (IP address) actually used in the device and click the **OK** Button.
- Repeat steps 1 to 3, and register all of the devices that participate in the tag data links.



# 7-2-4 Creating Tags and Tag Sets

You must create the tag sets and set member tags required to create connections for a registered EtherNet/IP Unit. You can set the network variables used in control programs for tags.

This section first describes the basic procedure to create tags and tag sets (1, below). Then it explains how to import variables with a Network Publish attribute from the Sysmac Studio to the Network Configurator (2, below).

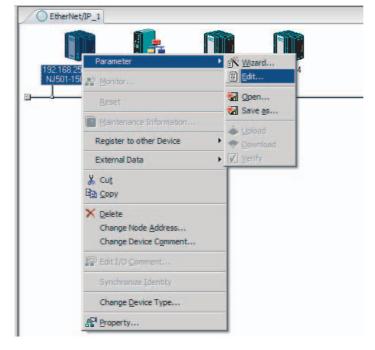
Then it explains how to effectively use network variables for tags.

- (1) Creating Tags and Tag Sets with the Network Configurator's Device Parameter Editing Function
- (2) Importing Variables with a Network Publish Attribute Created in the Sysmac Studio to the Network Configurator

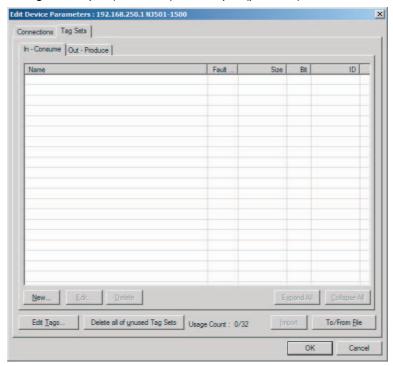
# (1) Creating Tags and Tag Sets with the Network Configurator's Device Parameter Editing Function

#### Creating a Tag Set

1 Double-click the icon of the device for which to create a tag set to display the Edit Device Parameters Dialog Box. Right-click the icon to display the pop-up menu, and select *Parameter – Edit*.



Click the Tag Sets Tab at the top of the Edit Device Parameters Dialog Box. There are two kinds of tag sets: input (consume) and output (produce).

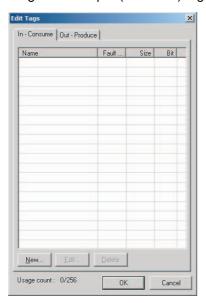


# Creating and Adding Tags

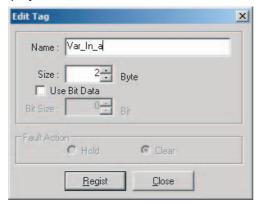
**3** Click the **Edit Tags** Button.

The Edit Tags Dialog Box is displayed.

Register the input (consume) tags and output (produce) tags separately.



4 Click the In - Consume Tab, and then click the New Button. The Edit Tag Dialog Box is displayed.



**5** Enter the variable name directly into the *Name* Box. (Example: Var\_In\_a)



#### **Additional Information**

- You can use the following characters in tag names. 0 to 9, A to Z, a to z, single-byte kana, \_
   (underbar), and multi-byte characters (e.g., Japanese)
- You cannot use the following characters in tag names. ! " # \$ & '() \* + , . / : ; < = > ? @ [] ^ '
   % spaces or text strings that start with numerals (0 to 9)
- The maximum length of a tag name is 255 bytes.
- Specify array variables, structure variables, and union variables as shown below.
  - Specifying array elements: array [2][3] (or array [2,3]) and array [2][3][4] (or array [2,3,4])
  - Specifying structure members: Struct.member (Separate the member name with a period.)
  - Specifying union members: Union.member (Separate the member name with a period.)
- To specify a specific I/O memory address for a tag, create a variable with an AT specification for the I/O memory address on the Sysmac Studio. (This applies only for the NJ-series CPU Units.) If you specify the variable with the AT specification for the tag on the Network Configurator, a tag data link is created to the I/O memory address that was set for the AT specification.

7-25

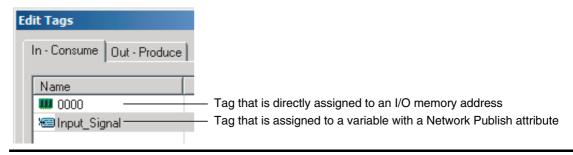


#### **Precautions for Correct Use**

Specify variable names for tags. To specify an I/O memory address for a tag, do not specify the I/O memory address for the tag directly. Instead, create a variable, set an AT specification of the I/O memory address on the Sysmac Studio, and then specify the variable with the AT specification for the tag. (This applies only for the NJ-series CPU Units.) If you enter I/O memory addresses for tag names on the Network Configurator, as shown below, the tags are directly assigned to the I/O memory addresses in the CPU Unit, and not to the variables. Always specify variable names for tags.

- 1) Variable names that contain only single-byte numerals from 0000 to 6143
- 2) Variable names with the following single-byte letters (uppercase or lowercase) followed by single-byte numerals
  - H (H000 to H511)
  - W (w000 to w511)
  - D (D00000 to D32767)
  - E0\_ to E18\_ (E0\_00000 to E0\_32767, to E18\_00000 to E18\_32767)

You cannot register a notation representing an I/O memory address as a tag name to the NXseries CPU Units. You can check the memory address or variable to which a tag is assigned with the icons in the Edit Tags Dialog Box.

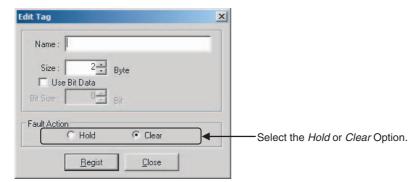


- Input the size of the tag in bytes in the Size Field. Input the tag size so that it is the same as the data type size of the variable. Select the Use Bit Data Check Box and change the bit size to 1 to use BOOL variables.
- Click the **Regist** Button to register the tag. If an I/O memory address is specified for a tag name, the Edit Tag Dialog Box is displayed with the next consecutive address as the tag name for the next tag. After you have registered all of the tags, click the **Close** Button.
- Click the Out Produce Tab, and then click the New Button. The Edit Tag Dialog Box is displayed. Input the output tag in the same way. Use the Fault Action setting of the output (produce) tag to specify whether to clear the output data or continue to send it when a major fault occurs in the CPU Unit.

The Fault Action setting is not required for input (consume) tag sets.

- Retain output for major fault: *Hold* (default)

  Output data maintains its previous status even after a major fault occurs.
- Clear output at major fault: Clear
   Output data is cleared to 0 when a major fault occurs.

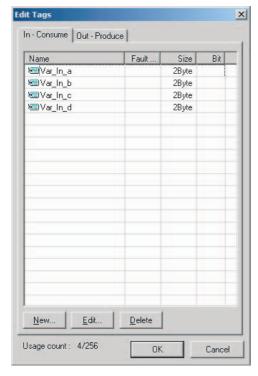




#### **Precautions for Correct Use**

Connections are cut off if any of the following errors occurs in the CPU Unit that is the originator while tag data links are active.

- Major fault level Controller error
- Partial fault level Controller error
- **9** After you register all of the required tags, click the **OK** Button in the Edit Tags Dialog Box.

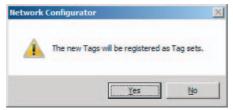




#### **Precautions for Correct Use**

Make the following settings to refresh all of the tag data in the same tag set at the same time.

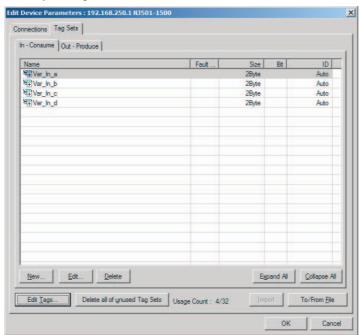
- Use the Sysmac Studio to specify, in advance, the same refreshing task for all of the variables that are assigned to tags in the tag set.
- Do not place tag variables that have AT specifications in I/O memory and tag variables that do not have AT specifications in the same tag set.
- 10 At this point, a confirmation dialog box is displayed to check whether the registered tag names are used as the tag set names. A tag set can contain up to eight tags, but tag sets are registered with one tag per tag set if the tag names are registered as tag set names. In this case, click the Yes Button.



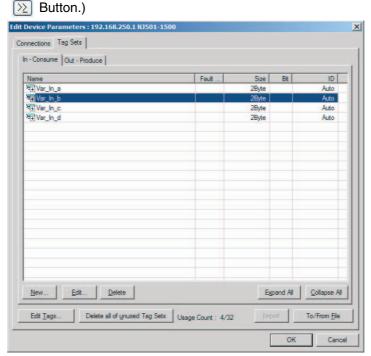
If the **No** Button is clicked, you can add more tags to the tag set. Refer to step 18 for details on how to register new tags first and add more tags to the tag set later.

#### Changing and Registering Tag Sets

11 The following dialog box is displayed when the tags in the Edit Tags Dialog Box are registered directly as tag sets.



If an input tag is already registered in an input tag set, and you want to change its registration to a different input tag set, it is necessary to delete the tag from the tag set in which it was originally registered. Open the Edit Device Parameters Dialog Box, select the tag set containing the tag that you want to delete on the Tag Sets Tab Page, and click the **Delete** Button in the Edit Tags Dialog Box. (If there are other tags registered in that tag set, it is possible to delete just one tag by selecting the tag that you want to delete in the Edit Tag Set Dialog Box and clicking the

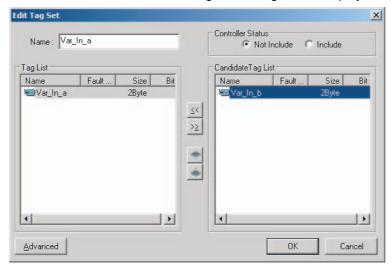


A confirmation message is displayed.



If the **No** Button is clicked, only the tag set is deleted. Click the **No** Button.

To edit a registered tag set and add tags, either double-click the tag set, or select the tag set and click the Edit Button. The Edit Tag Set Dialog Box is displayed.

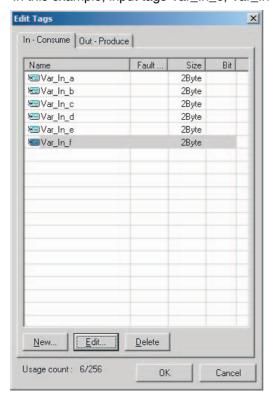


The Tag List on the left side of the dialog box shows the tags that are already registered, and the Candidate Tag List on the right side of the dialog box shows the other tags that are not registered yet. To add a tag, select it in the Candidate Tag List and click the 🔣 Button.

14 To include the Controller status in the tag set, select the *Include* Option at the upper-right corner of the dialog box.



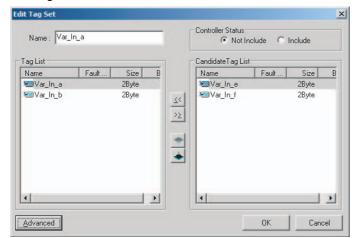
- To confirm a change, click the **OK** Button in the Edit Tag Set Dialog Box.
- Click the **OK** Button in the Edit Device Parameters Dialog Box.
- If you want to just add new tags and register the tag set, first register the tags with steps 1 to 9. In this example, input tags Var\_In\_e, Var\_In\_f are newly added.



- **18** After you register all of the required tags, click the **OK** Button at the bottom of the Edit Tags Dialog Box.
- At this point, a confirmation dialog box is displayed to check whether the registered tag names are used as the tag set names. Tags are just added in this case, so click the **No** Button. Just the tags are registered. The tags are not registered as tag sets.

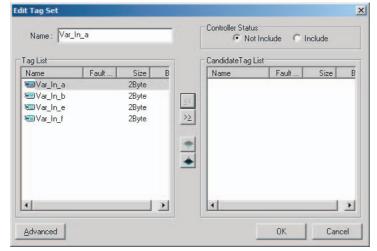


**20** To register the newly added tags in a tag set, either double-click the desired tag set, or select the tag set and click the **Edit** Button.



The Tag List on the left side of the dialog box shows the tags that are already registered, and the Candidate Tag List on the right side of the dialog box shows the other tags that are not registered yet.

**21** Select the tags that you want to add from the Candidate Tag List and click the 🔣 Button.



You can register up to eight tags in a tag set. (If you include the Controller status in the tag set, you can register up to only seven tags, and two bytes are added to the size.)

Data is sent and received in the order it is displayed in the tag list. To change the order of a tag, select the tag and click the Up and Down Buttons ( ).

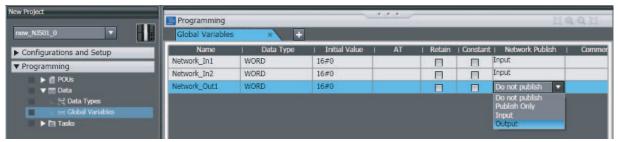
- **22** To confirm the changes, click the **OK** Button at the bottom of the Edit Tag Set Dialog Box.
- **23** Click the **OK** Button in the Edit Device Parameters Dialog Box.

# (2) Importing Variables with a Network Publish Attribute Created on the Sysmac Studio to the Network Configurator

You can create network variables in the Sysmac Studio and import these variables to the Network Configurator to assign them to tags and tag sets. Use the following procedure.

## Creating Global Variables on the Sysmac Studio

1 Create a global variable with the Global Variable Editor of the Sysmac Studio and select Input or Output for the Network Publish attribute of the variable. Save the project when you are finished.



Select Export Global Variables - Network Configurator... from the Tools Menu.

Any global variables with *Input* or *Output* set for the Network Publish attribute are imported from the csv file for the import procedure described below (Importing to the Network Configurator).

#### Importing to the Network Configurator



#### **Precautions for Correct Use**

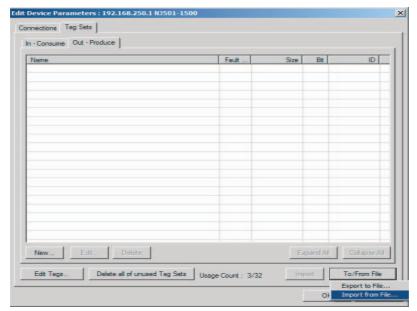
Variables with a Network Publish attribute that have variable names that are the same as the I/O memory address notation, such as "0000" and "H0000" are not exported to CSV files.

- 1) Variable names that contain only single-byte numerals (Example: 001)
- 2) Variable names with the following single-byte letters (uppercase or lowercase) followed by single-byte numerals
  - H (Example: H30) • W (Example: w30)
  - D (Example: D100)
  - E0\_ to E18\_ (Example: EA\_100)

Double-click the icon of the device registered in the Network Configurator for which you want to import the variable with a Network Publish attribute to display the Edit Device Parameters Dialog

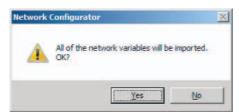
Right-click the icon to display the pop-up menu, and select **Device - Parameter - Edit**.



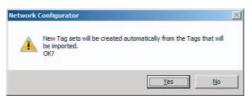


A confirmation dialog box is displayed that asks you how you want to import the variables as shown below.

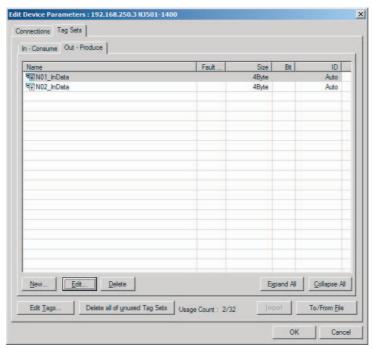
• To import all variables with a Network Publish attribute, click the **Yes** Button. To import only some of these variables, click the **No** Button.



After you import the variables to the tags, click the **Yes** Button to automatically create tag sets, or click the **No** Button to set up tag sets manually.



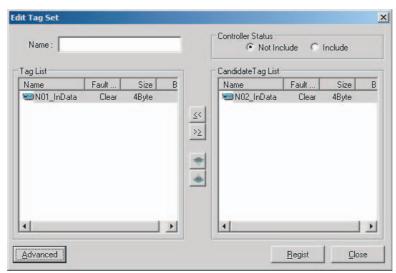
The variables will be imported as shown below on the Tag Sets Tab Page. Each variable will be imported into a different tag set and the device parameters will be automatically edited. (The variable name will be used for the tag set name.)



To place more than one input variable (input tag) imported from the Sysmac Studio into one tag set, you must delete the input tags that were registered. Select the tag set containing the variables you want to put into a tag set, then click the **Delete** Button. A confirmation dialog box is displayed to confirm that you want to delete the selected tag set and the tags contained in that tag set. You only want to delete the tag set, so click the **No** Button.



Click the **New** Button to create a new tag set. To place more than one tag in an existing tag set, double-click the tag set, or select it and click the Edit Button. The Edit Tag Set Dialog Box is displayed. Imported tags that are not registered in another tag set are displayed in the Candidate Tag List on the right. Click the Button to add tags individually.



- **3** You can change tag set names in this dialog box. To confirm a change, click the **Regist** Button in the Edit Tag Set Dialog Box.
- 4 Perform steps 1 to 3 for all the devices to import variables and to create tag sets.

#### 7-2-5 **Connection Settings**

After you create the tag sets, click the Connections Tab at the top of the Edit Device Parameters Dialog Box, and set the following connection information.

- The target devices with which connections are opened
- The connection type (multi-cast or unicast)
- The length of the packet intervals (RPI)
- · Connection name (optional)

Make the connections settings in the originator only. The connections settings are not necessary in the target device.

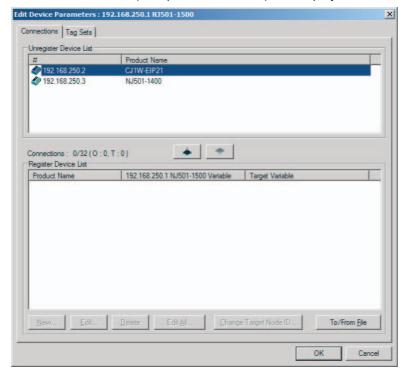


#### **Precautions for Correct Use**

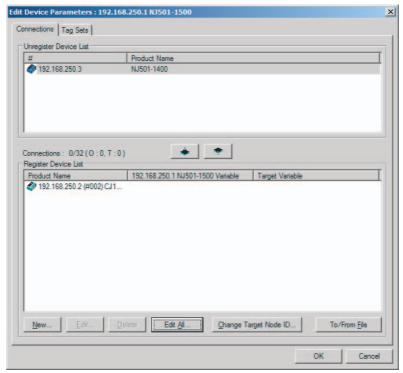
Make the connections settings after you create tag sets for all of the devices involved in tag data links.

# **Connection Settings (Connections Tab Page)**

- Registering Devices in the Register Device List
  - Double-click the icon of the device for which to make originator settings in the Network Configuration Pane of the Network Configurator. The Edit Device Parameters Dialog Box is displayed. Right-click the icon to display the pop-up menu, and select *Parameter - Edit*.
  - Click the Connections Tab in the Edit Device Parameters Dialog Box. All of the devices registered in the network (except the local node) are displayed.

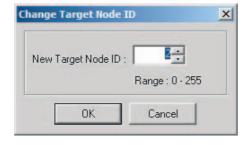


In the Unregister Device List, click the target device that requires connection settings so its color changes to gray, and click the Button. The selected target device is displayed in the Register Device List, as shown in the following diagram.



4 Target node IDs are assigned to the devices that are registered in the Register Device List.

The target node ID serves as the bit array position for the following variables in the originator Controller: Target Node Controller Mode, Target Node Controller Error Information, Target Node Error Information, Registered Target Node Information, and Normal Target Node Information. By default, the target ID is automatically set to the rightmost 8 bits of the IP address. In the example above, the target device's IP address is 192.168.250.2, so the device number is #002. If a target node ID is duplicated and you want to change the device number, click the **Change Target Node ID** Button and change the target ID.



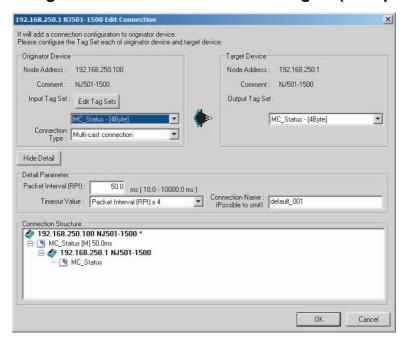
#### Editing Settings for Individual Connections

You can edit each connection separately.

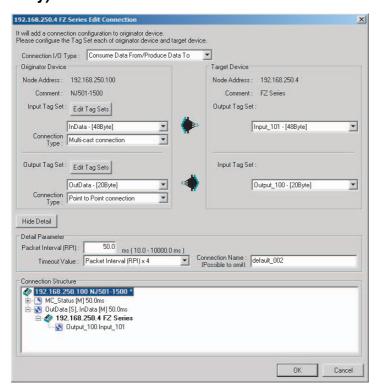
Note Refer to the following section for information on how to perform batch editing in a table format.

Click the **Connections** Tab and then click the **New** Button. The following Edit Connection Dialog Box is displayed according to the type of device that is selected.

#### Using Built-in EtherNet/IP Ports as Targets (for Input Only)



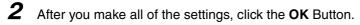
## Using Other EtherNet/IP Devices as Targets (for Settings Other Than Input Only)



The settings are as follows:

Setting		Description
Connection I/O Type	Select Input Only (tag type) to use tag data links with a CS1W-EIP21, CJ1W-EIP21, CJ2B-EIP21, CJ2M-EIP21, CJ1W-EIP21(CJ2), CJ1W-EIP21(NJ), NX701, NJ501-    NJ301-  O	
Connection Type	Select whether the data is sent in multi-cast or unicast (point-to-point) format. The default setting is multi-cast.	
	Multi-cast connection:	Select when the same data is shared by multiple nodes. This setting is usually used.
	Point-to-Point connection:	Select when the same data is not shared by multiple nodes. In a unicast transmission, other nodes are not burdened with an unnecessary load.
		w of Operation for details on using multi-cast and unicast ing the number of connections.
The Connection Structu	ure Area and the following items	are not displayed if the Hide Detail Button is clicked.
Packet Interval (RPI)		., the packet interval) of each connection between the ault setting is 50 ms (i.e., data is updated once every 50
	NX-series CPU Unit: Set the RPI to between 0.5	and 10,000 ms in 0.5-ms increments.
	NJ-series CPU Unit:     Set the RPI to between 1 a	nd 10,000 ms in 1-ms increments.*1
Timeout Value	Set the time until a connection timeout is detected. The timeout value is set as a multiple of the packet interval (RPI) and can be set to 4, 8, 16, 32, 64, 128, 256, or 512 times the packet interval. The default setting is 4 times the packet interval (RPI).	
Connection Name	Set a name for the connection. (32 single-byte characters max.)	

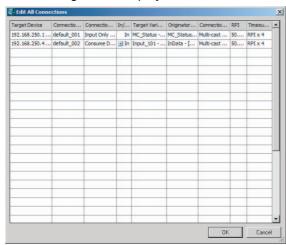
<sup>\*1</sup> For CPU unit version 1.02 or earlier, you can set the RPI with 1-ms increments between 10 and 10,000 ms.



#### Editing Settings for All Connections

You can edit the connection settings between the originator and all of the target devices selected in the Register Device List together in a table.

Click the **Connections** Tab, and then click the **Edit All** Button. The following Edit All Connections Dialog Box is displayed.



### The settings are as follows:

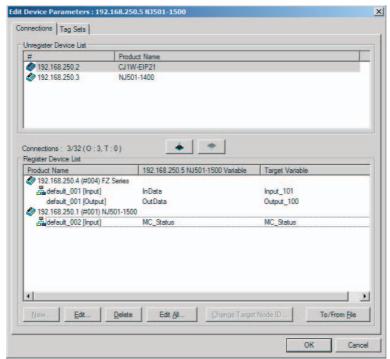
Setting		Description
Target Device	Select the target device.	
Connection Name	Any name can be given to the connection (32 single-byte characters max.). If this field is left blank, a default name is assigned. The connection name is used as a comment.	
Connection I/O Type	Select InputOnly (tag type) to use tag data links with a CJ1W-EIP21(CJ2), CJ1W-EIP21(NJ), NX701, NJ501-□□□, NJ301-□□□, or NJ101. When you create tag data links for other devices, select the connection I/O type specified in that device's EDS file. Use the Input Only (ID type) setting when another company's node is the originator and does not support connection settings with a Tag type setting.	
In/Out	The connections I/O is automatically displayed based on the selected connection.	
	<ul> <li>Input Only: Just In is display</li> </ul>	ayed.
Target Variable	Select the target node's tag	•
	In: Select the target's outp	· ·
	Out: Select the target's inp	
Originator Variable	Select the originator node's	
	In: Select the originator's i	
	Out: Select the originator's	
Connection Type	Select whether the data is sent in multi-cast or unicast (point- The default setting is multi-cast.	
	Multi-cast connection:	Select when the same data is shared by multiple nodes. This setting is usually used.
	Point-to-point connection:	Select when the same data is not shared by multiple nodes. In a unicast connection, other nodes are not burdened with an unnecessary load.
		w of Operation for details on using multi-cast and nd counting the number of connections.
RPI	Set the data update cycle (i.e., the packet interval) of each connection between the originator and target. The default setting is 50 ms (i.e., data i updated once every 50 ms).	
		5 and 10,000 ms in 0.5-ms increments.
	NJ-series CPU Unit:     For unit version 1.03 or later, set the RPI to between 1 and 10,000 ms in 1-ms increments.     For unit version 1.02 or earlier, set the RPI to between 10 and 10,000 ms in 1-ms increments.	
Timeout Value	Set the time until a connection timeout is detected. The timeout value is set as a multiple of the packet interval (RPI) and can be set to 4, 8, 16, 32, 64, 128, 256, or 512 times the packet interval. The default setting is 4 times the packet interval (RPI).	

**2** After you make all of the settings, click the **OK** Button.

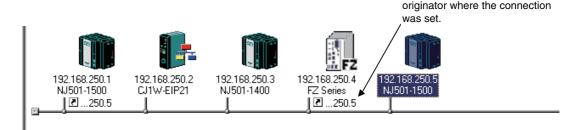
Indicates the IP address of the

## Confirming the Connections Settings

**1** An overview of the connections that were set in the Register Device List is displayed in the Connections Tab Page.



**2** Click the **OK** Button. The following kind of diagram is displayed.



**3** Repeat the connections setting procedure until all of the connections are set.



#### **Precautions for Correct Use**

After you have made all of the settings, always click the **OK** Button before you close the Edit Device Parameters Dialog Box. If the **Cancel** Button is clicked and the dialog box is closed, the new settings are discarded.

4 If the tag set's size is changed in either the originator or target after the connection was set, the size will not match the other node and a parameter data mismatch will occur. In this case, if you change the connection settings, be sure to check the connections. (Refer to 7-2-16 Checking Connections.)

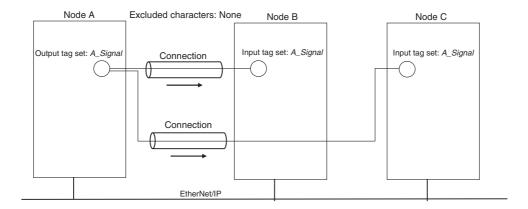
# **Automatically Setting Connections (Network – Auto Connection)**

You can use automatic detection of the tag set names that are set for devices to automatically set connections between input and output tag sets with the same name (or the same names excluding specified ellipses). Connections are automatically set under the following conditions.

Output tag set names for connection setting	Except for specified ellipses, the output tag set name must be the same as the input tag set name. Ellipses can be set for the beginning or end of tag set names.
Input tag set names for connection settings	Except for specified ellipses, the input tag set name must be the same as the output tag set name. Ellipses can be set for the beginning or end of tag set names.
Connection type	The connection type must be Input Only. Multi-cast or single-cast connections can be specified for a connection.
RPI	The default setting is used.
Timeout	The default setting is used.

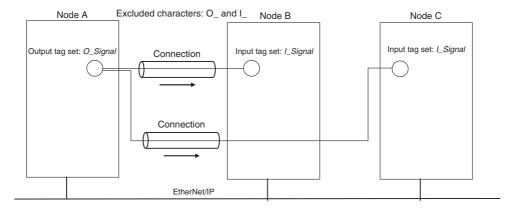
#### **Example 1: Automatic Connections with the Same Tag Set Names**

The following connections are automatically set with the same tag set name (A\_Signal) if there is an output (produce) tag set named A\_Signal at node A and input (consume) tag sets named A\_Signal at nodes B and C.



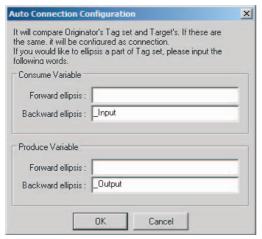
## **Example 2: Automatic Connections with the Ellipses**

The following connections are automatically set with the same tag set name (Signal) if there is an output (produce) tag set named O\_Signal at node A and input (consume) tag sets named I\_Signal at nodes B and C, and "O\_" and "I\_" are set as forward ellipses.



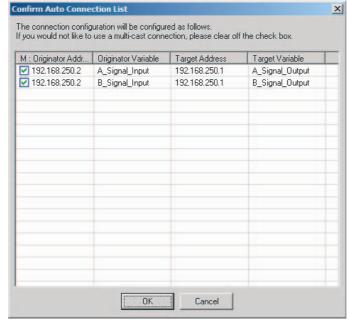
Set the same tag set names for the output and input tag sets for the connection. The tag set names can also include forward and backward ellipses.

2 Select *Auto Connection Configuration* from the Network Menu. The connections will be set automatically. A dialog box will appear to set forward and backward ellipses for both output and input tag sets as soon as automatic connection setting processing has begun.



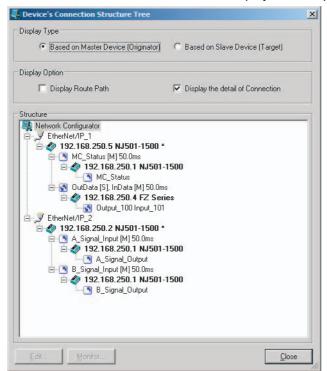
Input the ellipses and click the **OK** Button. Processing for automatic setting is started.

**3** If there are tag sets that meet the conditions for automatic connection setting, they are displayed.



Click the **OK** Button. Processing for automatic setting is started.

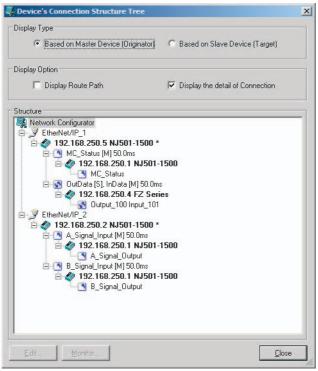
**4** A device connection structure tree is displayed when processing is completed.



Use the device connection structure tree as required to change the RPI and timeout settings.

# **Device Connection Structure Tree**

Connection settings can be displayed on the network configuration. Select *View Device's Connection Structure Tree* from the Network Menu.



- You can use the Display the detail of Connection Check Box to switch between device-level and connection-level displays of tag data link communications.
- An asterisk is displayed after the device name of the originator set for the connection.
- The Edit Device Parameters Dialog Box is displayed if you select a connection and click the Edit Button. You can edit the connections in this dialog box.

#### **Creating Connections Using the Wizard** 7-2-6

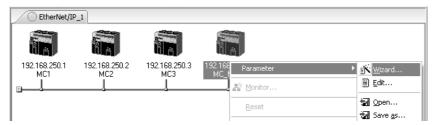
You can use the Network Configurator's Wizard to easily create connections between OMRON PLCs following the instructions provided by the Wizard.

Note The Wizard can be used only with the following OMRON EtherNet/IP devices.

Device name	Remarks
CJ1W-EIP21(NJ)	CJ1W-EIP21 mounted to NJ-series Controller
CJ1W-EIP21	CJ1W-EIP21 mounted to CJ1 CPU Unit
CJ1W-EIP21(CJ2)	CJ1W-EIP21 mounted to CJ2 CPU Unit
CJ2B-EIP21	Built-in EtherNet/IP port in CJ2H CPU Unit
CJ2M-EIP21	Built-in EtherNet/IP port in CJ2M CPU Unit
CS1W-EIP21	CJ1W-EIP21 mounted to CS1 CPU Unit
NX701	Built-in EtherNet/IP port on NX-series CPU Unit
NJ301-□□□□	Built-in EtherNet/IP port on NJ-series CPU Unit
NJ501-□□□□	
NJ101	

Use the following procedure to create connections (i.e., data links) with the Wizard.

- Set tags and tag sets for all devices before starting the Wizard. Refer to 7-2-4 Creating Tags and Tag Sets for the setting procedure.
- For tag data links between OMRON PLCs, a connection is created in the PLC (i.e., the originator device) that receives data as input data. First, select the registered device for which you want to create a connection in the Network Configuration Window of the Network Configurator, and then select Device - Parameters - Wizard from the menus.

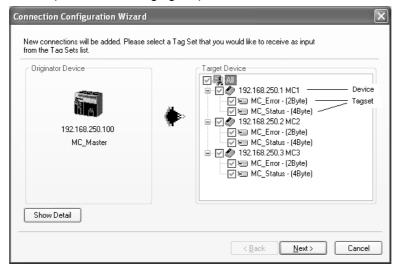


The following dialog box will be displayed before the Wizard starts.



Click the Yes Button to delete the connections that have been set with OMRON PLCs before starting the Wizard.

**3** Create the connection following the instructions that are given by the Wizard after the Wizard starts. (See the following figure.)



**4** A list of tag sets is displayed on the right side of the Wizard Dialog Box with target devices that support receiving input data.

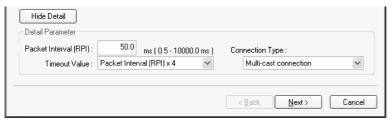
Select the tag sets that you want to receive at the originator device.

The following tables describes the meanings of the icons and check marks displayed in the tag set list.

Icon	Display position	Status
<u>~</u>	All	All output tag sets for all devices are selected.
_	Device	All output tag sets for the applicable device are selected.
	Tag set	The applicable output tag sets are selected. These are the tag sets that will be set in the connection.
V	All	All or some output tag sets for some devices are selected.
_	Device	Some output tag sets for applicable devices are selected.
Г	All	All output tag sets for all devices are not selected.
_	Device	All output tag sets for applicable devices are not selected.
	Tag set	The applicable output tag sets are not selected. The connections for this tag set will be deleted.
	Device	No applicable tag sets.

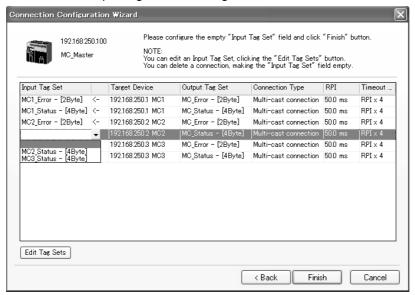
Note Tag sets that are used in connections that are already set are not displayed.

The following display will appear when you click the Show Detail Button.



The specified values for detailed parameters will be displayed. Change the values as required. The connection name cannot be set. They are automatically created using the following rule. default\_N (where N is a 3-digit number (001, 002, etc.) starting from 1)

Click the Next Button to switch to the table in the following Wizard Dialog Box. Follow the instructions to select and input from the list box the input tag set of the originator device that receives the output tag set of the target device.

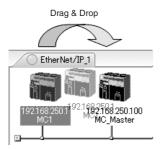


- The blank area in the Input Tag Set Column is the connection that you are creating.
- The rows in which there are input tag sets are connections that are already set.
- To prevent duplicate settings, input tag sets that have been used are not displayed in the list box for input tag sets.
- If there is no applicable input tag set, you can edit a tag set or create a new one by using the Edit Tag Sets Button and Edit Tag Button.
- **6** Once the input tag set settings have been completed, click the **Finish** Button. You can check the set connection by selecting Network - View Devices Connection Structure Tree from the menus.
  - The Wizard can be ended even if the input tag set includes a blank row. In that case, a connection is not created for the blank row.
  - You can delete a connection by deleting the input tag sets that were previously set.

# 7-2-7 Creating Connections by Device Dragging and Dropping

You can create a connection to the originator by dragging a target device and dropping it at the originator device. Network Configurator version 3.10 or higher is required to drag and drop devices to make connections.

Example: Drag the target device at 192.168.250.1 and drop it at the originator device at 192.168.250.100.





#### **Additional Information**

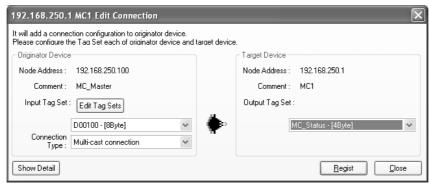
The EtherNet/IP originator device (i.e., a device in which connections can be set) must be one of the following OMRON EtherNet/IP devices.

Device name	Remarks
CJ1W-EIP21(NJ)	CJ1W-EIP21 mounted to NJ-series CPU Unit
CJ1W-EIP21	CJ1W-EIP21 mounted to CJ1 CPU Unit
CJ1W-EIP21(CJ2)	CJ1W-EIP21 mounted to CJ2 CPU Unit
CJ2B-EIP21	Built-in EtherNet/IP port in CJ2H CPU Unit
CJ2M-EIP21	Built-in EtherNet/IP port in CJ2M CPU Unit
CS1W-EIP21	CJ1W-EIP21 mounted to CS1 CPU Unit
NX701	Built-in EtherNet/IP port on NX-series CPU Unit
NJ301-□□□□	Built-in EtherNet/IP port on NJ-series CPU Unit
NJ501-□□□□	
NJ101	

Use the following procedure to create connections (i.e., data links) by dragging and dropping devices.

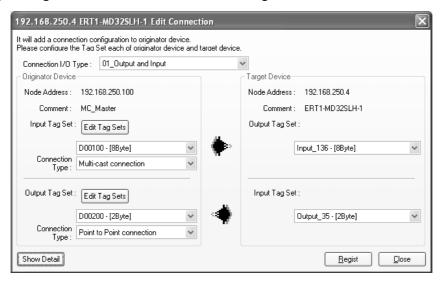
- Set the tags and tag sets for the target device that will be dragged.
  - (1) Refer to 7-2-4 Creating Tags and Tag Sets for information on creating the settings if the target is one of the OMRON EtherNet/IP devices given above.
  - (2) If the target is another EtherNet/IP device, refer to the manual of that device and perform settings as required.

- A dialog box as in the following figure for connection allocation will be displayed when you drag the target device and drop it at the OMRON EtherNet/IP device.
  - (1) Using One of the Above OMRON EtherNet/IP Devices As Target



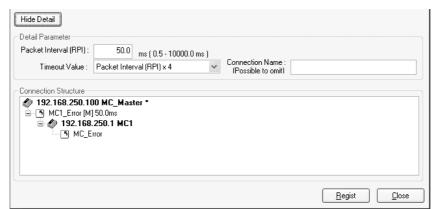
Select the output tag set from Target Device Area on the right side of the Edit Connection Dialog Box, and then select the input tag set to receive the output tag set in the Originator Device Area on the left.

- If there is no applicable input tag set at the originator, you can create a new one by using the Edit Tag Sets Button and Edit Tag Button.
- (2) Using Other EtherNet/IP Devices as Target



The connection I/O type list box in the upper part of the Connection Settings Dialog Box displays the connection I/O types that can be selected. Select the connection I/O type according to your application.

- The connection I/O types that can be selected depend on the target device.
- Items that can be selected will depend on the connection I/O type that is selected.
- Select the output, input, or both output and input tag sets at the target and specify the corresponding input, output, or both input and output tag sets at the originator.
- If there is no applicable tag set at the originator, you can create a new one by using the Edit Tag Sets Button and Edit Tag Button.



The following display will appear when you click the **Show Detail** Button.

The specified values for detailed parameters will be displayed. Change the values as required. Connection names are automatically created using the following rule.

default\_N (where N is a 3-digit number (001, 002, etc.) starting from 1)



#### **Additional Information**

The following dialog box will be displayed if a target device that does not have I/O data is dropped.



Before dropping again, refer to the manual of the applicable device and create the I/O data (i.e., output tag sets) required to create a connection.

After you have set all of the connection, click the Regist Button to create the connection. When creating the connection has been completed, the input tag set and output tag set will be blank. Next, you can continue to create connections by selecting the connection I/O type and setting a tag set.

#### **Connecting the Network Configurator to the Network** 7-2-8

This section describes how to Connect the Network Configurator online.



#### **Additional Information**

Although all NX-series CPU Units provide two EtherNet/IP ports, the Network Configurator treats these two ports as two different units and connects them individually.

# Connecting through Ethernet

Connect to the built-in EtherNet/IP port on the CPU Unit via an Ethernet switch.



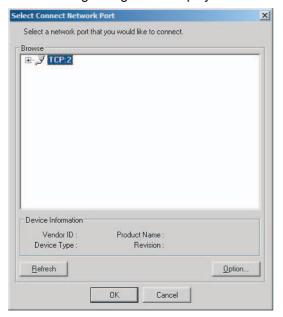
#### **Precautions for Correct Use**

The first time you connect via Ethernet with Windows XP (SP2 or higher), Windows Vista, or Windows 7, you much change the Windows firewall settings. For the procedure, refer to A-4 Precautions for Using the Network Configurator on Windows XP, Windows Vista, or Windows 7.

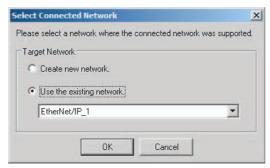
Select Option - Select Interface - Ethernet I/F.

Select Network - Connect. If there are multiple Ethernet interfaces on the computer, the Select Connect Network Port Dialog Box is displayed. Select the interface to connect, and press the **OK** Button.

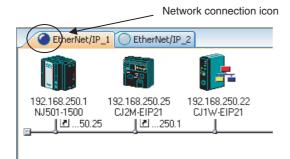
The following dialog box is displayed.



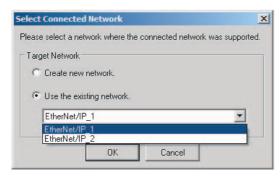
**3** Click the **OK** Button. Select the network to connect.



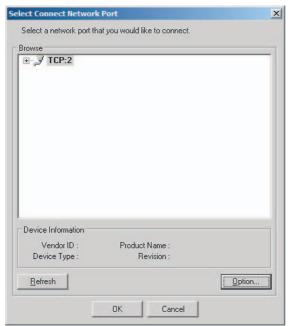
The Network Configurator will connect to the EtherNet/IP network. If the Network Configurator goes online normally, "Online" is displayed in the status bar at the bottom of the window. The network connection icon is displayed in blue on the Network Tab Page in which the Network Configurator is connected.



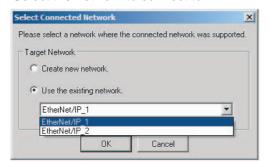
Select Network - Change Connect Network to switch the connected network.



The following dialog box is displayed.



Click the **OK** Button. Select the network to connect to.



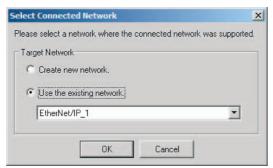
# Connections through NJ/NX-series CPU Unit's USB Port

Use the following procedure to connect to the built-in EtherNet/IP port via the USB port on the CPU Unit.

- Select the communications interface. Select Option - Select Interface - NJ/NX Series USB Port.
- Select Network Connect. The following dialog box is displayed.



**3** Select *TCP:2* and then click the **OK** Button. The following dialog box is displayed.



4 Select the network to connect and click the **OK** Button.

The Network Configurator will connect to the EtherNet/IP network. If the Network Configurator goes online normally, "On-line" is displayed in the status bar at the bottom of the window.

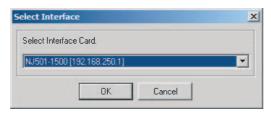


# Direct Connection via Ethernet to Built-in EtherNet/IP Port on NJ/NXseries CPU Units

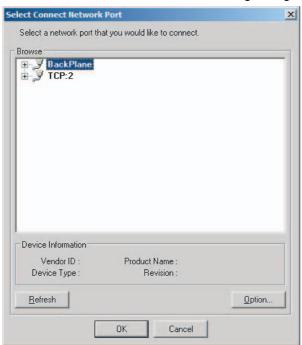
Use the following procedure to directly connect to the built-in EtherNet/IP port via Ethernet. You can connect to the built-in EtherNet/IP port even if the IP address has not been set on the computer.

- Select the communications interface.
  Select Option Select Interface NJ/NX Series Ethernet Direct I/F.
- 2 Select **Network Connect**.

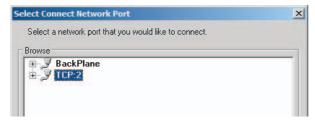
The Setup Interface Dialog Box is displayed if there are several Controllers that you can connect to. Select NX701-\(\subseteq\) (IP number), NJ501-\(\subseteq\) (IP number), NJ301-\(\subseteq\) (IP number), or NJ101-\(\subseteq\) (IP number).



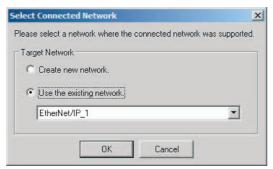
Select *Network* – *Connect*. The following dialog box is displayed.



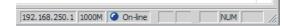
Select TCP:2 and then click the OK Button.



Select the network to connect.



The Network Configurator will connect to the EtherNet/IP network. If the Network Configurator goes online normally, "On-line" is displayed in the status bar at the bottom of the window.



# 7-2-9 Downloading Tag Data Link Parameters

To make tag data links, you must download tag data link parameters, such as tag set settings and connection settings, to all devices in the EtherNet/IP network. When the download operation is executed, the tag data link parameters are transferred to the EtherNet/IP Units that require the settings.

The following procedure shows how to download the tag data link parameters. Refer to *7-2-8 Connecting the Network Configurator to the Network* for information on how to Connect the Network Configurator online.



#### **Precautions for Correct Use**

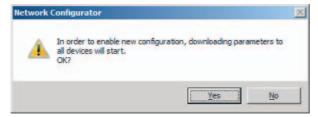
- If the node addresses (IP addresses) are not set correctly, you may connect to the wrong Controller and set incorrect device parameters. Download data only after you confirm that you are connected to the correct Controller.
- If incorrect tag data link parameters are set, it may cause equipment to operate unpredictably. Even when the correct tag data link parameters are set, make sure that there will be no effect on equipment before you transfer the data.
- When network variables are used in tag settings, a connection error will result if the variables
  are not also set in the CPU Unit. Before downloading the tag data link parameters, check to
  confirm that the network variables are set in the CPU Unit. Check whether the network variable, tag, and connection settings are correct. On the Connection and Tag Status Tab Pages
  described in 15-3-1 The Network Configurator's Device Monitor Function.
- If a communications error occurs, the output status depends on the specifications of the
  device being used. When a communications error occurs for a device that is used along with
  output devices, check the operating specifications and implement safety countermeasures.
- If the CIP revision of the Network Configurator is earlier than that of the CPU Unit, you cannot
  download the tag data link parameters in the Network Configurator to the CPU Unit. Upgrade
  the CIP revision of the Network Configurator to be the same as that of the CPU Unit, and
  download the parameters.
- Make sure that the major CIP revision of the device registered with the Network Configurator is
  the same as the major CIP revision of the NJ/NX-series CPU Unit that you use. If major CIP
  revisions are not the same, the parameters may not be downloaded. To determine whether
  downloading is possible, refer to 7-2-3 Registering Devices.
- Do not disconnect the Ethernet cable or reset or turn OFF the power to the EtherNet/IP Unit during the parameter download.
- Tag data links (data exchange) between relevant nodes is stopped during a download. Before
  you download data in RUN mode, make sure that it will not affect the controlled system. Also
  implement interlocks on data processing in ladder programming that uses tag data links when
  the tag data links are stopped or a tag data link error occurs.
- For EtherNet/IP Units with revision 1, you can download tag data link parameters only when the CPU Unit is in PROGRAM mode.
- Even for Units with revision 2 or later, all CPU Units must be in PROGRAM mode to download the parameters if any Units with revision 1 are included in the network.

- Connect the Network Configurator online.
- There are two ways to download the parameters.

#### **Downloading to All Devices in the Network**

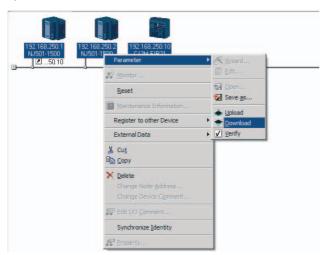
Select Network - Download.

The following dialog box is displayed.

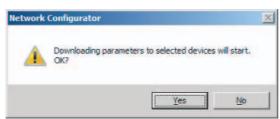


#### **Downloading Individually to Particular Devices**

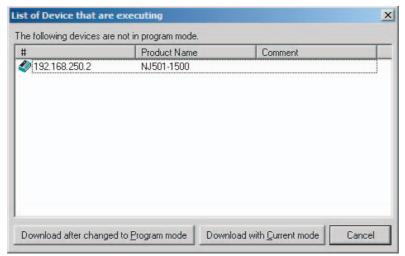
Select the icon of the EtherNet/IP Unit to which you want to download. To select multiple nodes, hold down the Shift Key or the Ctrl Key while you click the icons. (In the following example, 2 nodes are selected: 192.168.250.1 and 192.168.250.2.) Right-click the icon to display the popup menu, and select *Parameter – Download*.



The following dialog box is displayed.



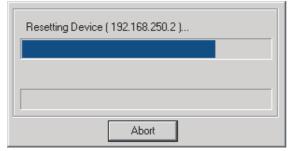
3 Click the Yes Button to download the tag data link parameters to the EtherNet/ IP Unit. The following dialog box is displayed if any of the CPU Units is not in PROGRAM mode.



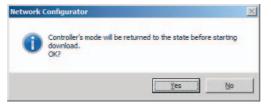
If the **Download after changed to Program mode** Button is clicked, all CPU Units are changed to PROGRAM mode and the parameters are downloaded. Confirm safety for all controlled equipment before you change the CPU Units to PROGRAM mode. You can restore the operating modes after the parameters are downloaded.

You can click the **Download with Current mode** Button to download the parameters even when one or more CPU Units is in RUN mode. The **Download with Current mode** Button is disabled if the EtherNet/IP Unit does not support this function (e.g., revision 1 of CJ1W-EIP21 or CS1W-EIP21).

During the download, the following progress monitor is displayed to show the progress of the download.



If the operating mode of one or more CPU Units was changed to download the parameters, you can return the CPU Units to the previous operating modes. If the **No** Button is clicked, the CPU Units remain in PROGRAM mode.



4 The following dialog box is displayed to show that the download was completed.



#### 7-2-10 **Uploading Tag Data Link Parameters**

You can upload tag data link parameters (such as the tag set settings and connection settings) from EtherNet/IP Units in the EtherNet/IP network. The following procedure shows how to upload the parameters. For details on how to connect to the network from the Network Configurator, refer to 7-2-8 Connecting the Network Configurator to the Network.



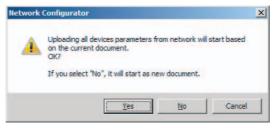
#### **Precautions for Correct Use**

Make sure that the major CIP revision of the device registered with the Network Configurator is the same as the major CIP revision of the NJ/NX-series CPU Unit that you use. If the major CIP revisions are not the same, the parameters may not be uploaded. To determine whether uploading is possible, refer to 7-2-3 Registering Devices.

There are two ways to upload the parameters.

#### Uploading from All Devices in the Network

- Connect the Network Configurator online, and then select *Upload* from the Network Menu.
- The following dialog box is displayed.



#### Clicking the Yes Button:

The tag data link parameters in the current project are uploaded.

#### Clicking the No Button:

You open a new project to upload the tag data link parameters. The current project is closed.

#### **Clicking the Cancel Button:**

The upload operation is cancelled. The upload is not performed.

If you click the **Yes** Button in step 2, the following dialog box is displayed.



#### Clicking the Yes Button:

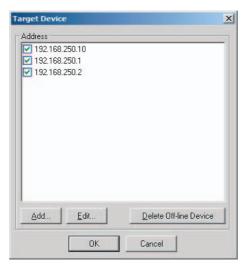
Parameters are uploaded only from the devices registered in the Network Configuration Pane. Parameters are not uploaded from devices that are not registered in the Network Configuration Pane.

#### Clicking the No Button:

Performing a Batch Upload over the Network

Parameters are uploaded from all devices on the network. The current Network Configuration Information will be lost.

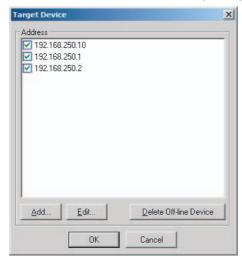
The following dialog box will be displayed. Select the devices for which to upload parameters and click the **OK** Button.



#### **Clicking the Cancel Button:**

The upload operation is cancelled. The upload is not performed.

4 If you click the **No** Button in step 2, the following dialog box is displayed. Select the devices for which to upload parameters and click the **OK** Button.



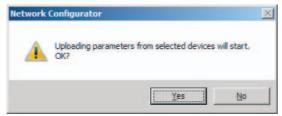
#### **Uploading Individually from Particular Devices**

Connect the Network Configurator online and select the icon of the EtherNet/IP Unit from which you want to upload the parameters. To select multiple nodes, press and hold the Shift Key or the Ctrl Key while you select additional icons. (In the following example, 2 nodes are selected: 192.168.250.1 and 192.168.250.2.)

Right-click the icon to display the pop-up menu, and select *Parameter - Upload*.

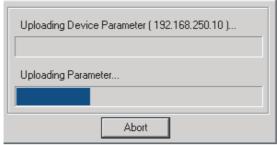


The following dialog box is displayed.



Click the Yes Button or the No Button.

During the upload, the following progress monitor is displayed to show the progress of the upload.



The following dialog box is displayed to show that the upload was completed.



# 7-2-11 Verifying the Tag Data Links

Tag data link parameters (such as the tag set settings and connection settings) can be compared with the parameters of the built-in EtherNet/IP ports in the EtherNet/IP network. The following procedure shows how to compare the parameters. For details on how to connect to the network from the Network Configurator, refer to 7-2-8 Connecting the Network Configurator to the Network.



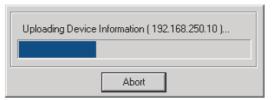
#### **Precautions for Correct Use**

Make sure that the major CIP revision of the device registered with the Network Configurator is the same as the major CIP revision of the NJ/NX-series CPU Unit that you use. If the major CIP revisions are not the same, the parameters may not be compared. To determine whether comparison is possible, refer to *7-2-3 Registering Devices*.

# **Verifying the Network Configuration**

You can use the following procedure to compare the list of registered devices in the Network Configuration Pane with the devices connected on the EtherNet/IP network, and check the IP addresses and device types. This function does not verify device parameters.

- **1** Connect the Network Configurator online.
- Select Network Verify Structure. The following progress monitor is displayed to show the progress as data is read from the network and compared.

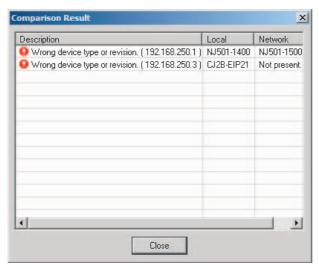


3 The results of the comparison between the network configuration file and data from the network are displayed as shown below.

#### **Differences Not Found in the Comparison**



## **Differences Found in the Comparison**



#### **Differences Found in the Device Type**

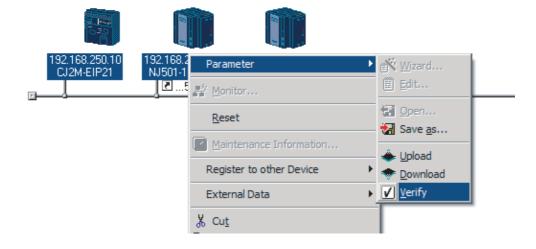


Click the **OK** Button or the **Close** Button.

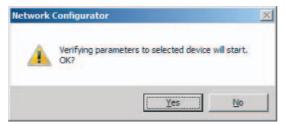
# Verifying the Device Parameters

Use the following procedure to compare the device parameters for the devices selected in the Network Configuration Pane with those of the devices connected on the EtherNet/IP network. The IP addresses, device types, and device parameters are compared.

- Connect the Network Configurator online.
- Click the icon of the built-in EtherNet/IP port to verify. To select multiple nodes, press and hold the Shift Key or the Ctrl Key while you select additional icons. (In the following example, 2 nodes are selected: 192.168.250.1 and 192.168.250.2.) Right-click the icon to display the pop-up menu and select *Parameter – Verify*.



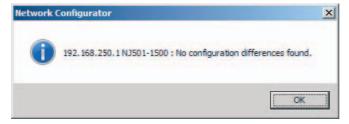
**3** The following dialog box is displayed.



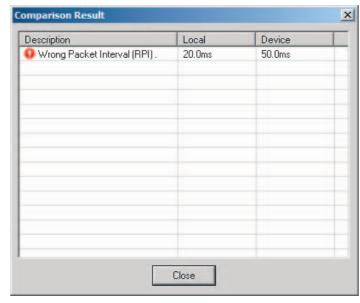
Click the Yes Button or the No Button.

**4** The following dialog box is displayed.

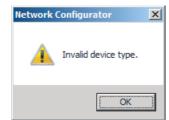
# **Differences Not Found in the Comparison**



#### **Differences Found in the Comparison**

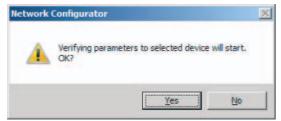


#### **Differences Found in the Device Type**



Click the **OK** Button or the **Close** Button.

If multiple nodes have been selected and compared, the following message is displayed. Click the Yes Button.



The comparison results are displayed in order of the selected nodes.

# 7-2-12 Starting and Stopping Tag Data Links

# **Automatically Starting Tag Data Links**

Tag data links are automatically started immediately after the data link parameters are downloaded from the Network Configurator. (They are automatically started after the CPU Unit's power is turned ON or the Unit is restarted.)



#### **Additional Information**

With CPU Units with unit version 1.04 or later, a Tag Data Link Connection Timeout error will occur if a connection is not established with the target device within 1 minute after the tag data links are started in operation as the originator device. Reconnection processing is continued periodically even after this error occurs to automatically recover. If the application environment allows this error to be ignored, such as when a target device is started later than the originator device, you can change the event level to the observation level.

# Starting and Stopping Tag Data Links for the Entire Network

You can start and stop tag data links for the entire network from the user program or from the Network Configurator.



#### **Precautions for Correct Use**

Use the same method (i.e., either the user program or the Network Configurator) to both start and stop tag data links. For example, if you use the \_EIP\_TDLinkStopCmd (Tag Data Link Communications Stop Switch) system-defined variable stop tag data links, you cannot start them from the Network Configurator.

# Using Commands in the User Program

You can change the corresponding elements in the following system-defined variables to TRUE in the user program to start and stop tag data links for individual devices. (Refer to Section 3 System-defined Variables Related to the Built-in EtherNet/IP Port.)

- NX-series CPU Unit
  - You can individually start and stop tag data links for each built-in EtherNet/IP port.
    - Tag data links start/stop operation switch for built-in EtherNet/IP port 1
       \_EIP\_TDLinkStartCmd, \_EIP1\_TDLinkStartCmd (Tag Data Link Communications Start Switch)
       \_EIP\_TDLinkStopCmd, \_EIP1\_TDLinkStopCmd (Tag Data Link Communications Stop Switch)
    - Tag data links start/stop operation switch for built-in EtherNet/IP port 2
       \_EIP2\_TDLinkStartCmd (Tag Data Link Communications Start Switch)
       \_EIP2\_TDLinkStopCmd (Tag Data Link Communications Stop Switch)
- NJ-series CPU Unit
  - EIP TDLinkStartCmd (Tag Data Link Communications Start Switch)
  - EIP TDLinkStopCmd (Tag Data Link Communications Stop Switch)



#### **Additional Information**

- Change the Tag Data Link Start Switch to TRUE, while the Tag Data Link Communications Stop Switch is FALSE. If the Tag Data Link Stop Switch is TRUE, the tag data links do not start even if the Tag Data Link Start Switch is changed to TRUE. Furthermore, if the Tag Data Link Start Switch and the Tag Data Link Stop Switch are both TRUE, an error occurs, the Multiple Switches TRUE Error system-defined variable changes to TRUE, and the event is recorded in the event log.
- · After you start the tag data links, do not force the Tag Data Link Start Switch to change to FALSE from the user program or from the Sysmac Studio. It will change to FALSE automati-

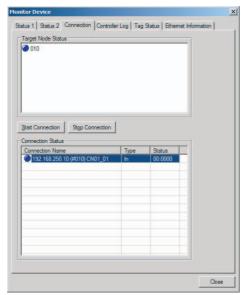
### Using the Network Configurator

You can select I/O Connection - Start or I/O Connection - Stop from the Network Menu to start and stop tag data links for individual devices.

# Starting and Stopping Tag Data Links for Individual Devices

#### Using the Network Configurator

You can start and stop tag data links for individual devices using the Connection Tab Page in the Monitor Device Dialog Box. This applies only to tag data links for which the device is the originator. Select *Monitor* from the Device Menu to access the Monitor Device Dialog Box. When using an NXseries CPU Unit, you can individually start and stop the tag data links of the built-in EtherNet/IP ports 1 and 2 connected to the Network Configurator.



#### **Start Connection Button:**

Starts all connections for which the device is the originator.

#### **Stop Connection Button:**

Stops all connections for which the device is the originator.

# 7-2-13 Clearing the Device Parameters

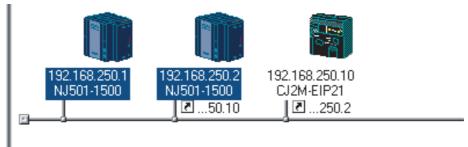
You can clear the tag data link settings (or return them to their factory settings) that are saved in the registered EtherNet/IP device. The following procedure shows how to clear the tag data link settings. For details on how to connect to the network from the Network Configurator, refer to 7-2-8 Connecting the Network Configurator to the Network.



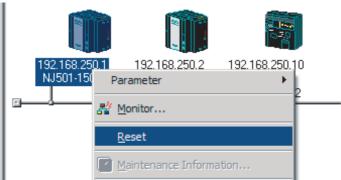
#### **Precautions for Correct Use**

#### For a CPU Unit with Unit Version 1.10 or Later

- Use the Network Configurator version 3.58 or higher to perform the following procedures to clear the tag data link settings.
- If you perform the following procedures from the Network Configurator version 3.57 or lower, the tag data link settings are not cleared. Refer to *Additional Information* in this section for the procedures to clear the tag data link settings from the Network Configurator version 3.57 or lower.
- **1** Connect the Network Configurator online.
- Select the icon of the device from which you want to clear the device parameters. In the following example, 2 nodes are selected: 192.168.250.1 and 192.168.250.2. To select multiple nodes, press and hold the **Shift** Key while you select additional icons.



3 Select *Device - Reset*. You can also right-click the icon and select *Reset* from the popup menu.



**4** The following dialog box is displayed.



#### Yes Button:

The following dialog box is displayed.



Select the Initialize tag data link configuration, and then emulate cycling power Option and then click the OK Button.



#### **Precautions for Correct Use**

The Controller is not restarted. Only the built-in EtherNet/IP port is reset.

#### No Button:

The tag data link settings will not be cleared and the built-in EtherNet/IP port will not be reset.



#### **Additional Information**

You can also execute the Reset service of the Identity object for the CPU Unit to clear the tag data link settings. The procedures to execute the service from the Network Configurator is given below.

- (1) Connect the Network Configurator to the network.
- (2) Select *Tool Setup Parameters* on the main window.

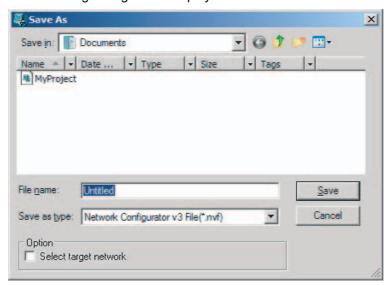
Then the dialog box for the general parameter settings are displayed.

- (3) Specify the target device and message to send.
  - Target Node Address: Enter the IP address of the target device.
  - Service: Select Reset. · Class: Enter 01. • Instance: Enter 01. Attribute: Enter 00. • Data: Enter 02.\*
- (4) Click the **Send** Button.
- \* For a CPU Unit version 1.09 or earlier, specify 01.

# 7-2-14 Saving the Network Configuration File

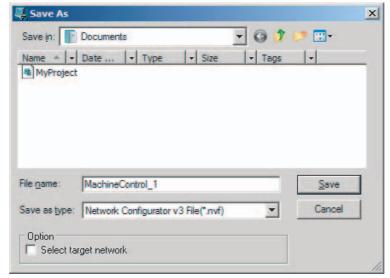
You can save device parameters set in the Network Configurator or device parameters uploaded from the network in a network configuration file.

Select File – Save As.
The following dialog box is displayed.



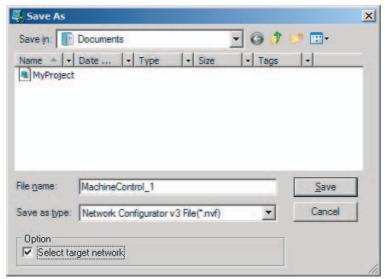
Untitled.nvf is displayed as the default file name.

**2** Input the file name, and then click the **Save** Button.

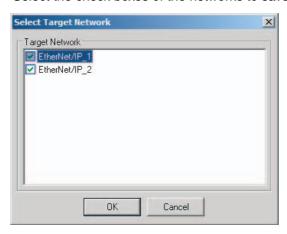


This completes the network configuration file save operation.

- When the network configuration is changed later, you can overwrite the existing network configuration file if you select *File Save* or click the Button.
- 4 You can select the *Select target network* Check Box in the Option Area to save a network configuration file with only the required networks.



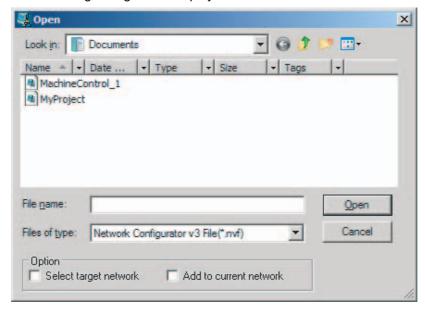
Select the check boxes of the networks to save and click the **OK** Button.



#### **Reading a Network Configuration File** 7-2-15

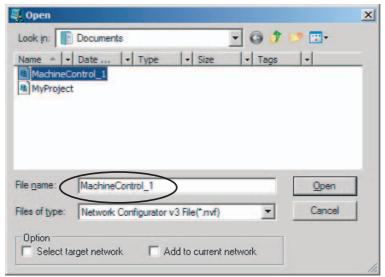
You can read a previously saved network configuration file into the Network Configurator.

Select File - Open or click the Button. The following dialog box is displayed.

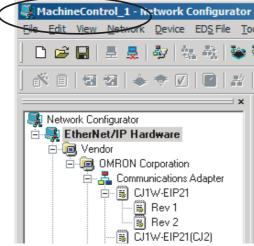


If the network configuration file that you want to read is not displayed, change to another folder.

**2** If you select the network configuration file that you want to read, that file name is displayed in the *File name* Field.



- **3** Click the **Open** Button to read the network configuration file.
- 4 The Network Configurator's Title Bar will display the name of the file that was read.



**5** Select any of the options as necessary. The options are listed below.

Setting	Description
Select target network	Allows you to select specific networks from the network configuration and open them.
Add to current document	Allows you to add the networks from the network configuration file that is currently open to the current configuration file.



#### **Additional Information**

The save format will depend on the Network Configurator version. You can import configuration files (\*.ncf) created with the Network Configurator for EtherNet/IP (version 2 or lower) if you select *External Data – Import* from the File Menu.

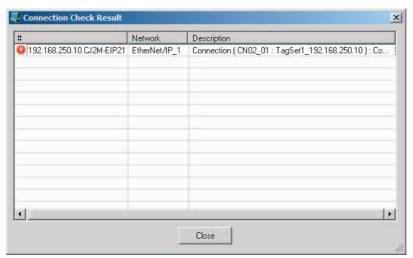
# 7-2-16 Checking Connections

You can check the consistency of connection parameters for network configuration files with device parameters that were set with the Network Configurator or device parameters uploaded from the network.

1 Select Check Connection from the Network Menu. The following dialog box is displayed if parameters are normal.



The following dialog box is displayed if there are parameter errors. Check the displayed details and review the settings.



If an inconsistency is found, open the originator's Edit Device Parameter Dialog Box and click the Connection Tab. The inconsistent connection in the Register Device List is displayed with a 📇 icon (instead of the normal 🚣 icon). To change the connection setting and select a different target variable, select the connection as shown below and click the Edit Button.



# 7-2-17 Changing Devices

You can change devices that are registered in a network configuration with the Network Configurator. Select *Change Device* from the Device Menu to display a list of the possible devices to change to. Select the desired device. You can change a device only when there is complete or upward compatibility with the device.

# **Device Changes**

Model afte	er	CJ1W- EIP21	CS1W- EIP21	CJ1W- EIP21	CS1W- EIP21	CJ1W- EIP21(NJ)	CJ2W- EIP21(CJ2)	CJ2B- EIP21	CJ2M- EIP21	NJ501-E NJ301-E		NJ101	NX701
Model before change	CIP Rev	1.01	1.01	2.01	2.01	2.04	2.01	2.01	2.01	1.0□*1	2.0□*2	2.0□	2.0□
CJ1W -EIP21	1.01		Yes	Yes	Yes	Yes	Yes	Yes	*2	*3	*3	*3	No
CS1W -EIP21	1.01	Yes		Yes	Yes	Yes	Yes	Yes	*2	*3	*3	*3	No
CJ1W -EIP21	2.01	No	No		Yes	*4	Yes	Yes	*2	*3 *4	*3 *4	*3 *4	No
CS1W -EIP21	2.01	No	No	Yes		*4	Yes	Yes	*2	*3 *4	*3 *4	*3 *4	No
CJ1W- EIP21(NJ)	2.04	No	No	*1 *7	*1 *7		*7	*7	*6 *7	*3	*3	*3	Yes
CJ1W- EIP21(CJ2)	2.01	No	No	*1	*1	*4		Yes	*2	*3 *4	*3 *4	*3 *4	*4
CJ2B -EIP21	2.01	No	No	*1	*1	*4	Yes		*2	*3 *4	*3 *4	*3 *4	*4
CJ2M -EIP21	2.01	No	No	*1	*1	*4	Yes	Yes		*3 *4	*3 *4	*3 *4	*4
NJ501-	1.0□ *1	No	No	*4 *7	*4 *7	*7	*7	*7	*6 *7		Yes	Yes	Yes
NJ301- □□□□	2.0□ *2	No	No	*4 *7	*4 *7	*7	*7	*7	*6 *7	*3		Yes	Yes
NJ101	2.0□	No	No	*4 *7	*4 *7	*7	*7	*7	*6 *7	*3	Yes		Yes
NX701	2.0□	No	No	No	No	*7	*7	*7	*6 *7	*3	*3	*3	

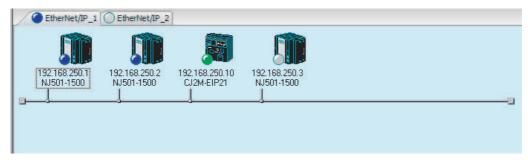
Yes: Can be changed.

No: Cannot be changed.

- \*1 Cannot be changed if a network variable is specified as a tag.
- \*2 Cannot be changed if the following items exceed the permissible settings of the device after the change: Number of I/O connections, number of tags, number of tag sets, and size of one tag set.
- \*3 Cannot be changed in any of the following cases:
  - The number of I/O connections, number of tags, number of tag sets, or size of one tag set exceeds the permissible settings for the device after the change.
  - RPI exceeds the permissible settings or is set in 0.5-ms increments (such as 10.5 ms).
- \*4 Cannot be changed if a variable that is set in a tag or the tag size is an odd number of bytes.
- \*5 Cannot be changed if a tag or tag set size is an odd number of bytes.
- \*6 Cannot be changed if tags or tag sets exceed the permissible settings for the refreshing size, or one of the tag sizes is an odd number of bytes.
- \*7 Cannot be changed if the maximum size of a tag name or tag set name (size after conversion into UTF-8) exceeds 48 bytes.
- \*8 CPU Unit with a unit version 1.00 to 1.02
- \*9 CPU Unit with a unit version 1.03 or later

# 7-2-18 Displaying Device Status

Device status is displayed using the following icons in Maintenance Mode. To enter Maintenance Mode, select *Large Icons – Maintenance Mode* from the View Menu.



lcon	Status
0	Offline
(white)	
(gray)	Default (including no Controller Configurations and Setup)
(green)	Idle (including when CPU Unit of Controller is in PROGRAM mode)
(blue)	Normal communications state (including when Controller is in RUN mode)
(yellow)	Warning status (including when there is a partial fault or non-fatal error)
(red)	Alarm status (including when there is a major fault or fatal error in the Controller)

# 7-3 Ladder Programming for Tag Data Links

# 7-3-1 Ladder Programming for Tag Data Links

If data in the ladder program is linked by tag data links, add conditions 1 to 3 in the ladder program for that data. If you want to use target node Controller information in the input conditions, add conditions 4 and 5.

- Conditions to enable the NJ/NX-series Built-in EtherNet/IP Port's tag data links:
  - (1) The following error bits in the Built-in EtherNet/IP Error variable (\_EIP\_ErrSta) are FALSE.

Major fault: Bit 7, Partial fault: Bit 6, and Minor fault: Bit 5

- (2) Also, the Online variable (\_EIP\_EtnOnlineSta)\*1 is TRUE.
- Conditions showing that connections are established with the target device, and tag data links are operating:
  - (3) The bit corresponding to the target node address in the Normal Target Node Information variable (\_EIP\_EstbTargetSta)\*2 is TRUE.
- Condition for the Target PLC Operating Mode (operating or stopped)
   (\_EIP\_TargetPLCModeSta)\*3 (valid for OMRON Controllers only):
  - (4) The Controller operating mode of the target node is ON.
- Condition for the Controller error status (fatal or non-fatal error) of the target node (valid for OMRON Controllers only):
  - (5) The target node Controller error information element in the Target PLC Error Information (\_EIP\_TargetPLCErr)\*4 is FALSE.

When you want to use the Target Node Controller Error Flag, the Controller status must be included in the tag sets for both the originator and target. Include the Controller status by using the Network Configurator to select the Include Options in the Edit Tag Set Dialog Boxes.

- \*1 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EtnOnlineSta Built-in EtherNet/IP port 2: \_EIP2\_EtnOnlineSta
- \*2 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EstbTargetSta Built-in EtherNet/IP port 2: \_EIP2\_EstbTargetSta
- \*3 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TargetPLCModeSta Built-in EtherNet/IP port 2: \_EIP2\_TargetPLCModeSta
- \*4 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TargetPLCErr Built-in EtherNet/IP port 2: \_EIP2\_TargetPLCErr

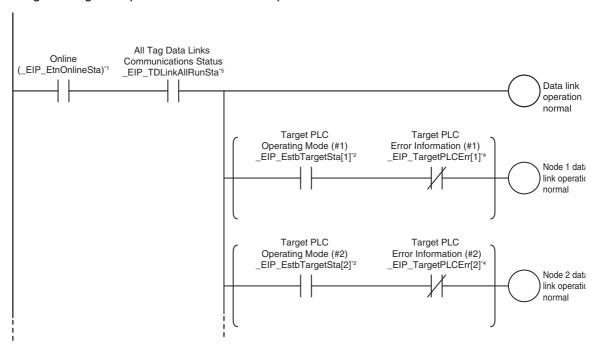
#### Programming Example to Detect Normal Operation

The following programming can be used to confirm that normal communications are being performed for each target node. If the Controller status is included in the tag data, the status of the Controller can also be detected.

Programming Example 1 to Detect Normal Operation

```
Normal Target Node
                                                           Target PLC Operating
                                                                                              Target PLC Error
       Online
                            Information (#1)
                                                                 Mode (#1)
                                                                                              Information (#1)
( EIP EtnOnlineSta)*1
                       EIP EstbTargetSta[1]*
                                                        EIP TargetPLCModeSta[1]*3
                                                                                            EIP TargetPLCErr[1]*4
                                                                                                                                  Node 1 dat
                                                                                                                                  link operation
                                                                                                                                  normal
                          Normal Target Node
                                                          Target PLC Operating
                                                                                             Target PLC Error
                               Information
                                                                                              Information (#2)
                                   (#2)
                                                                Mode (#2)
                                                                                           _EIP_TargetPLCErr[2]*4
                         _EIP_EstbTargetSta[2]<sup>*2</sup>
                                                       _EIP_TargetPLCModeSta[2]<sup>+3</sup>
                                                                                                                                  Node 2 dat
                                                                                                                                  link operation
                                                                                                                                  normal
```

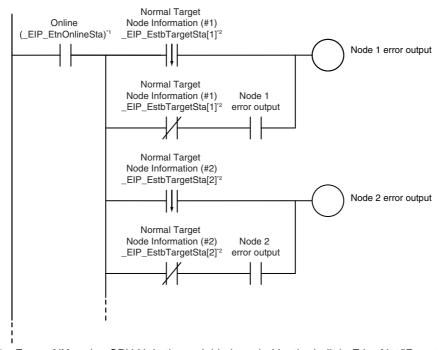
Programming Example 2 to Detect Normal Operation



- \*1 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EtnOnlineSta Built-in EtherNet/IP port 2: \_EIP2\_EtnOnlineSta
- \*2 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EstbTargetSta Built-in EtherNet/IP port 2: \_EIP2\_EstbTargetSta
- \*3 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TargetPLCModeSta Built-in EtherNet/IP port 2: \_EIP2\_TargetPLCModeSta
- \*4 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TargetPLCErr Built-in EtherNet/IP port 2: \_EIP2\_TargetPLCErr
- \*5 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TDLinkAllRunSta Built-in EtherNet/IP port 2: EIP2 TDLinkAllRunSta

# Programming Example to Detect Errors

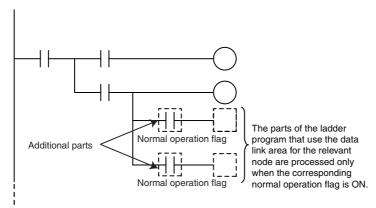
The following programming can be used to check for tag data link errors for each target node. This programming is used to detect errors only after the data links for all nodes have started normally.



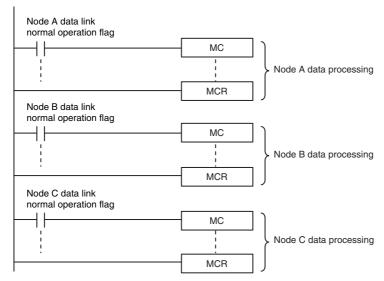
- \*1 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EtnOnlineSta Built-in EtherNet/IP port 2: \_EIP2\_EtnOnlineSta
- \*2 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EstbTargetSta Built-in EtherNet/IP port 2: \_EIP2\_EstbTargetSta

#### Data Processing Programming Example

The following type of programming can be used to process data only when the data links are operating normally.



• You can use MC - MCR instructions and JMP instructions to process data only when the data links are operating normally as shown below.





#### **Precautions for Correct Use**

Even if an error occurs in communications with a target device, the input data from the target device will remain stored in words allocated in memory to the local node. To prevent malfunctions, write the user program so that no input processing is performed when the following Built-in EtherNet/IP Error (\_EIP\_ErrSta) bits are TRUE.

Major fault: Bit 7 Partial fault: Bit 6 Minor fault: Bit 5

# 7-3-2 Status Flags Related to Tag Data Links

The status of the tag data links is reflected in the following system-defined variables.

Variable name	Description
_EIP_TargetPLCModeSta[255]*1 (Target PLC Operating Mode)  * Corresponds to the operating information in the Controller status.	This variable shows the operating status of the target node Controllers that are connected with the built-in EtherNet/IP port as the originator. The information in this area is valid only when the corresponding Normal Target Node Information is TRUE. If the corresponding Normal Target Node Information is FALSE, the Target Node Controller Operating Information indicates the previous operating status.
	Array[x] is TRUE: The target Controller with a node address of x is in operating status.
	Array[x] is FALSE: Other than the above.
_EIP_TargetNodeErr[255]*2 (Target Node Error Information)  * Corresponds to the Controller Error Information in the Controller status.	This variable indicates that the connection for the Registered Target Node Information was not established or that an error occurred in the target Controller. The information in this area is valid only when the Registered Target Node Information is TRUE.
	Array[x] is TRUE: The Registered Target Node Information for a node address of x is TRUE, and the Normal Target Node Information is FALSE or the Target PLC Error Information is TRUE.
	Array[x] is FALSE: When the Registered Target Node Information for a node address of x is FALSE, or when the Registered Target Node Information is TRUE and the Normal Target Node Error Information is TRUE.
	This is when the Controller Error Information is FALSE.
_EIP_EstbTargetSta[255]*3 (Normal Target Node Information)	This variable gives a list of nodes that have normally established EtherNet/IP connections.
* This status is not included in the Controller status.	Array[x] is TRUE: The connection to the node with a node address of x is established normally.
	Array[x] is FALSE: A connection is not established or an error has occurred.

- \*1 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TargetPLCModeSta Built-in EtherNet/IP port 2: \_EIP2\_TargetPLCModeSta
- \*2 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_TargetNodeErr Built-in EtherNet/IP port 2: \_EIP2\_TargetNodeErr
- \*3 For an NX-series CPU Unit, the variable is varied by the built-in EtherNet/IP port number to communicate with. Built-in EtherNet/IP port 1: \_EIP1\_EstbTargetSta Built-in EtherNet/IP port 2: \_EIP2\_EstbTargetSta

# **Tag Data Links with Other Models**

The performance of tag data links depends on the CPU Unit and EtherNet/IP Unit model as shown below. When you use tag data links between the built-in EtherNet/IP port on the NJ/NX-series CPU Unit and another CPU Unit or EtherNet/IP Unit, use tag data link settings of the Unit with the lower communications performance.

### Differences in Tag Data Link Performance Specifications

			N.I. corios	CDUUmit	CJ2M-	СРИЗ□	CS1W-EIP21,
Ito	em	NX-series	NJ-series	CPU Unit	Unit v	ersion	CJ1W-EIP21,
		CPU Unit	Unit version 1.00 to 1.02	Unit version 1.03 or later	2.0	2.1 or later	or CJ2H- CPU6□-EIP
Tags	Total size of all tags	184,832 words (total of 369,664 words with two ports)	9,600 words		640 words		184,832 words
	Maximum size of 1 tag	722 words (721 words when the tag set includes the Controller status)	the tag set includes the Controller status)		20 words (19 words when the tag set includes the Controller status)	640 words (639 words when the tag set includes the Controller status)	722 words (721 words when the tag set includes the Controller status)
	Number of registrable tags	256 (total of 512 tags with two ports)	256* <sup>3</sup>		32		256
Tag sets	Maximum size of 1 tag set	722 words (721 words when the tag set includes the Controller status)	then the tag set includes the Con- troller status)		20 words (19 words when the tag set includes the Controller status)	640 words (639 words when the tag set includes the Controller status)	722 words (721 words when the tag set includes the Controller status)
	Number of tags per tag set		the tag set incl		·		
	Number of registrable tag sets	256 (total of 512 tags with two ports)	32		32		256
Connections	Number of connections	256 (total of 512 connec- tions with two ports)			32		256
Maximum data size per connection		722 words*2 (Data con- currency is maintained within each connection.)	300 words (Refer to 7-1-7 Concurrency of Tag Data Link Data for information on the conditions to maintain concurrency in the data for one connection.)		20 words 640 words (Data concurrency is maintained within each connection.)		252 words or 722 words*2 (Data con- currency is maintained within each connection.)
Packet intervals (RPIs)		0.5 to 10,000 ms in 0.5-ms increments	10 to 10,000 ms in 1-ms increments	1 to 10,000 ms in 1-ms increments	1 to 10,000 ms increments	s in 0.5-ms	0.5 to 10,000 ms in 0.5-ms increments
Communication used (pps)	ons bandwidth	40,000 pps*1	1,000 pps*1	3,000 pps*1	3,000 pps*1		6,000 pps*1

<sup>\*1</sup> Here, pps means "packets per second" and indicates the number of packets that can be processed in one sec-

- \*2 To use a data size of 505 bytes or more, the system must support a large forward open (an optional CIP specification). The SYSMAC CS/CJ-series Units support Large\_Forward\_Open, but before you connect to nodes of other companies, confirm that those devices also support it.
- \*3 The maximum number of tags is given for the following conditions.
  - · All tag sets contain eight tags.
  - The maximum number of tag sets (32) is registered.

#### Specifying Tags

You can specify where to assign a tag either with a variable or with a I/O memory address. However, some CPU Units may not support both of these methods. Communications with the devices are possible regardless of whether the remote node tags are set using I/O memory addresses or network variables.

The supported tag specification methods for each CPU Unit are listed in the table below.

Yes: Supported, No: Not supported

CPU Unit		Name in Hardware List of	Network variable	I/O memory address
	EtherNet/IP Unit	Network Configurator	name specification	specification
NX-series CPU Unit		NX701	Yes	No
NJ-series CPU Unit		NJ501-□□□□	Yes	Yes*
		NJ301-□□□□		
		NJ101		
	CJ1W-EIP21	CJ1W-EIP21(NJ)	Yes	Yes*
CJ2H-CPU6□-EIP		CJ2B-EIP21	Yes	Yes
	CJ1W-EIP21	CJ1W-EIP21(CJ2)	Yes	Yes
CJ2H-CPU6□	CJ1W-EIP21	CJ1W-EIP21(CJ2)	No	Yes
CJ2M-CPU3□		CJ2M-EIP21	Yes	Yes
	CJ1W-EIP21	CJ1W-EIP21(CJ2)	Yes	Yes
CJ2M-CPU1□	CJ1W-EIP21	CJ1W-EIP21(CJ2)	No	Yes
CJ1 CPU Unit	CJ1W-EIP21	CJ1W-EIP21	No	Yes
CS1 CPU Unit	CS1W-EIP21	CS1W-EIP21	No	Yes

<sup>\*</sup> To specify an I/O memory address for a tag, do not specify the I/O memory address for the tag directly. Instead, create a variable, set an AT specification of the I/O memory address on the Sysmac Studio, and then specify the variable with the AT specification for the tag.



# **Message Communications**

8-1	Overv	iew of the CIP Message Communications Service 8-3
	8-1-1	Overview of the CIP Message Communications Service 8-3
	8-1-2	Message Communications Service Specifications
8-2	CIP Me	essaging Communications Client Function8-4
	8-2-1	Overview 8-4
	8-2-2	CIP Communications Instructions
	8-2-3	Using CIP Communications Instructions 8-
	8-2-4	Route Path
	8-2-5	Preparing Derivative Data Types to Use CIP Communications Instructions
	8-2-6	Sample Programming for CIP Connectionless (UCMM) Message Communications 8-22
	8-2-7	Sample Programming for CIP Connection (Class 3) Message Communications
	8-2-8	Operation Timing
	8-2-9	Response Codes
8-3	CIP Co	ommunication Server Functions
	8-3-1	CIP Message Structure for Accessing CIP Objects 8-40
	8-3-2	CIP Message Structure for Accessing Variables 8-40
8-4	Reque	est Path Used in the Server Function8-4
	8-4-1	Examples of CIP Object Specifications 8-4
	8-4-2	Examples of Variable Specifications
	8-4-3	Logical Segment 8-42
	8-4-4	Data Segment
	8-4-5	Specifying Variable Names in Request Paths 8-4-
		·

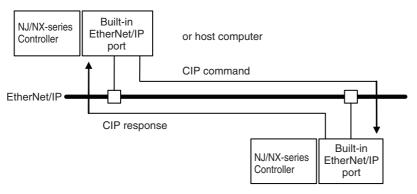
8-5	CIP Ob	pject Services	8-47
	8-5-1	CIP Objects Sent to the Built-in EtherNet/IP Port	. 8-47
	8-5-2	Identity Object (Class ID: 01 Hex)	. 8-47
	8-5-3	TCP/IP Interface Object (Class ID: F5 hex)	. 8-51
	8-5-4	Ethernet Link Object (Class ID: F6 Hex)	. 8-54
	8-5-5	Controller Object (Class ID: C4 Hex)	. 8-58
8-6	Read a	and Write Services for Variables	8-59
	8-6-1	Read and Write Services for Variables	. 8-59
	8-6-2	Write Service for Variables	. 8-61
8-7	Variab	le Data Types	8-63
	8-7-1	Data Type Codes	. 8-63
	8-7-2	Common Format	. 8-64
	8-7-3	Elementary Data Types	. 8-64
	8-7-4	Derived Data Types	. 8-66

# 8-1 Overview of the CIP Message Communications Service

# 8-1-1 Overview of the CIP Message Communications Service

CIP commands can be sent to devices on the EtherNet/IP network whenever they are required. You execute CIP\_SEND instructions in a program in the NJ/NX-series CPU Unit to send CIP commands, such as those to read and write data and to receive the responses.

You can use CIP messages from the client to read and write memory in the Controller with the server without adding any special programming to the user program of the Controller with the server.



# 8-1-2 Message Communications Service Specifications

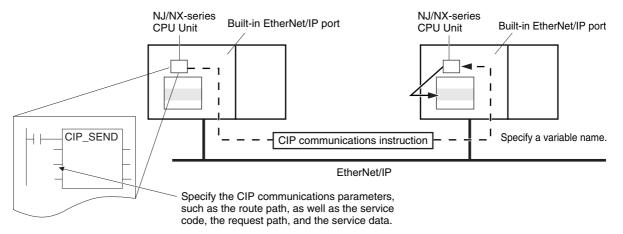
Ite	em	Specification	
Message type		Either of the following can be selected.	
		CIP UCMM connectionless messages	
		CIP class 3 connection messages	
Execution method		CIPSend (Send Explicit Message Class 3) instruction or CIPUCMM (Read Variable UCMM Explicit) instruction	
Data contents		Sending required CIP commands and receiving responses	
Communications pa	arameters	Message type, timeout value, and root path specification	
Maximum length per connection	Non-connection type (UCMM)	502 bytes	
Connection type (class 3)		Using Forward_Open     502 bytes	
		Using Large_Forward_Open     NX-series CPU Unit: 8192 bytes     NJ-series CPU Unit: 1994 bytes	

# 8-2 **CIP Messaging Communications Client Function**

#### **Overview** 8-2-1

The CIP Messaging Communications Client function is available for an NJ/NX-series CPU Unit. This function allows you to request a service by sending CIP messages to an external device and specifying a self-contained object of a device supporting the CIP Communication Server functions.

The NJ/NX-series CPU Unit executes CIP communications instructions in the user program and send CIP messages. In this way, you can use CIP messages to execute instructions such as reading and writing variables (data) of the NJ/NX-series CPU Unit on the EtherNet/IP network.



# 8-2-2 CIP Communications Instructions

The following CIP communications instructions are available. For details on CIP communications instructions, refer to the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

Instruction	Name	Description	Communica- tions method
CIPUCMMRead	Read Variable UCMM Explicit	Reads the value of a variable with a Network Publish attribute from the specified remote Controller on the CIP network and stores the value in a variable at the local Controller.	CIP UCMM connectionless messages
CIPUCMMWrite	Write Variable UCMM Explicit	Writes the value of a variable at the local controller to a variable with a Network Publish attribute at the specified remote Controller on the CIP network.	
CIPUCMMSend	Send Explicit Message UCMM	Sends a specified CIP command to the specified remote Controller on the CIP network.  Refer to 8-2-9 Response Codes and 8-5 CIP Object Services for information on the service codes and response codes that are used with the NJ/NX-series CPU Units.	
CIPOpen	Open CIP Class 3 Connection (Large_Forward _Open)	Opens a CIP class 3 connection (Large_Forward_Open) with the specified remote node.	CIP class 3 con- nection mes- sage
CIPOpenWithD ataSize	Open CIP Class 3 Connection with Specified Data Size	Opens a CIP class 3 connection with the specified remote node that allows class 3 explicit messages of the specified data length or shorter to be sent and received.	
CIPRead	Read Variable Class 3 Explicit	Reads the value of a variable with a Network Publish attribute from the specified remote Controller on the CIP network and stores the value in a variable at the local Controller.	
CIPWrite	Write Variable Class 3 Explicit	Writes the value of a variable at the local controller to a variable with a Network Publish attribute at the specified remote Controller on the CIP network.	
CIPSend	Send Explicit Message Class 3	Sends a specified class-3 CIP command to the specified remote Controller on the CIP network.  Refer to 8-2-9 Response Codes and 8-5 CIP Object Services for information on the service codes and response codes that are used with the NJ/NX-series CPU Units.	
CIPClose	Close CIP Class 3 Connection	This instruction closes the CIP class 3 connection that is specified by the handle.	



#### **Version Information**

A CPU Unit with unit version 1.06 or later and Sysmac Studio version 1.07 or higher are required to use the CIPOpenWithDataSize instruction.

#### **Using CIP Communications Instructions** 8-2-3

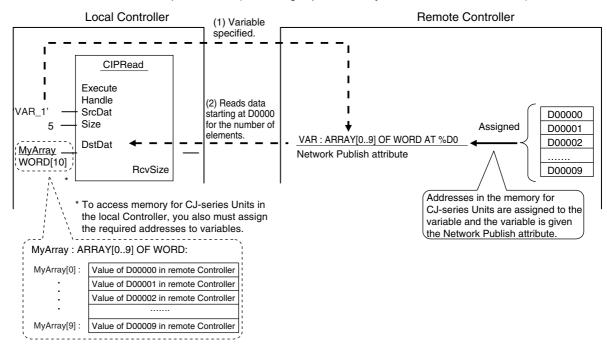
CIP message communications include the following processes. If CIP class 3 connections are used, the open and close processes are required before and after the data is sent and received.

Process	Description	Instruction
Open process (only for CIP class 3 connec- tions)*1	Execute this process before you use a CIP message. Open processing is continued until a CIP class 3 connection is established.	CIPOpen CIPOpenWithDataSize
Sending and receiving variable data*2	This process is used to read and write data for specified variables with the Network Publish attributes.	CIPUCMMRead CIPUCMMWrite CIPRead CIPWrite
Sending CIP commands	You can set the required CIP command.	CIPUCMMSend CIPSend
Close process (only for CIP class 3 connec- tions)	This process closes the connection.	CIPClose

- The following is the maximum number of connection handles that can be obtained simultaneously by the opening process.
  - NJ-series CPU Unit: 32
  - NX-series CPU Unit: 128

Even if a connection is broken for a timeout, the handle is not released. Execute the CIPClose instruction.

Addresses in memory for CJ-series Units (e.g., D0000) cannot be specified directly. To access memory for CJ-series Units, access a variable with an AT specification. (Accessing is possible only for an NJ-series CPU Unit.)





#### **Precautions for Correct Use**

You can execute up to 32 CIP communications instructions at the same time regardless of the instruction types.

Use exclusive control in the user program so that the number of CIP communications instructions executed at the same time does not exceed the above numbers.

#### 8-2-4 **Route Path**

The route path indicates the path from the local CPU Unit to the remote Controller on the network. Routing is performed for CIP communications instructions based on route paths.

# **Route Path Notation**

The EPATH data type is used to give route paths. The basic format is shown below.

Network\_type\_number \Remote\_address

#### NX-series CPU Unit

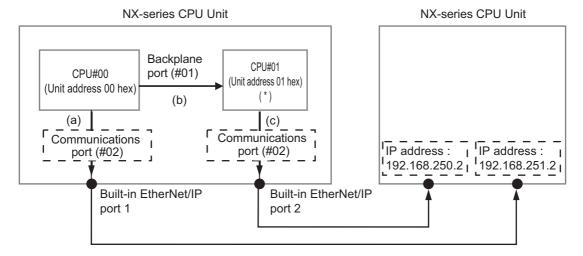
Two internal CPU Units are provided (each with a unique unit address) to control the two built-in EtherNet/IP ports.

- For built-in EtherNet/IP port 1: CPU Unit with a unit address of 00 hex (CPU #00)
- For built-in EtherNet/IP port 2: CPU Unit with a unit address of 01 hex (CPU #01)

In the route path of the CIP communication instruction (RoutePath in-out variable), add entries that recognize the above two CPU Units (CPU#00 and CPU#01) before sending the CIP communication instruction.

Route path when sending a CIP communication instruction

- When issuing a CIP communication instruction, issue from CPU #00. (a)
- To output from built-in EtherNet/IP port 2, output from CPU #00 via CPU #01. (b) to (c)



	Route	Content entered in route path		
Route	notation	Network type number (hexadecimal)	Destination address (hexadecimal)	
Output from built-in EtherNet/IP port 1	(a)	#02 (communications port)	IP address	
Output from built-in EtherNet/IP port 2	(b)	#01 (backplane port)	#01(unit address of CPU Unit) (CPU #01 for built-in EtherNet/IP port 2 communication)	
	(c)	#02 (communications port)	IP address	

Route Path

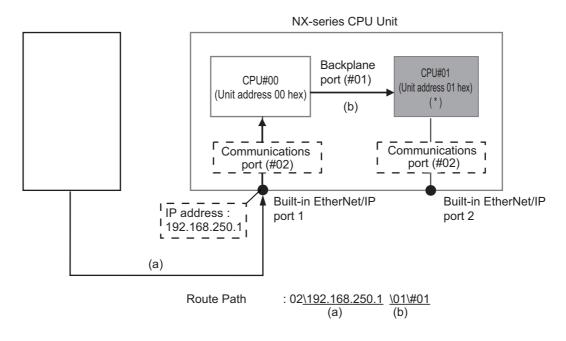
Output from built-in EtherNet/IP port 1: <u>02\192.168.250.2</u>
(a)

Output from built-in EtherNet/IP port 2 : 01\#01 \02\192.168.251.2 (b) (c)

Regardless of to which communications port (EtherNet/IP port 1 or 2) the data is actually input, you
can access both CPU Units (CPU#00 and CPU#01) that control the EtherNet/IP ports, via backplane
port.

### Example:

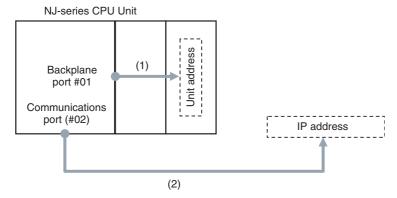
Inputting an Ethernet Link object (class ID: F6 hex) through built-in EtherNet/IP port 1 of the destination NX-series CPU Unit, and reading the settings and status of built-in EtherNet/IP port 2



### NJ-series CPU Unit

The network type number and the remote address are determined as shown in the following table according to whether the route type is (1) a Unit on the CPU Rack or (2) a communications port on a Communications Unit.

Route	Network type number (hexadecimal)	Remote address (hexadecimal)
Output to Unit on CPU Rack	#01 (backplane port)	Remote Unit address (Refer to Additional Information below.)
Output from communications port on Communications Unit	#02 (Built-in EtherNet/IP port)	IP address



### (1) When Routing the Output to a Unit on the CPU Rack

Output the command to the backplane port as a network with the CPU Rack. Specify the Unit address as the address of the remote Unit.

### (2) When Routing the Output to a Communications Port on a Communications Unit

Output the command to an EtherNet/IP port. Specify the IP address as the address of the remote node.



### **Additional Information**

**Unit Addresses** 

A Unit address is used to discriminate between several devices connected to a single node on a network.

Set the unit address as shown below.

• CPU Unit: 00 hex, 01 hex

• CPU Bus Units (EtherNet/IP Units): Unit number + 10 hex

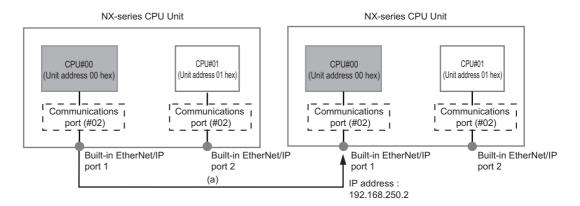
## **Route Path Notation Examples**

### NX-series CPU Unit

The route path notation is different for communication using built-in EtherNet/IP port 1 (CPU#00) and for communication using built-in EtherNet/IP port 2 (CPU#01). This section provides examples of route paths.

This example explains communication via an NX-series CPU Unit.

# (1) Using built-in EtherNet/IP port 1 (local CPU #00) (Local CPU #00 to destination CPU #00)

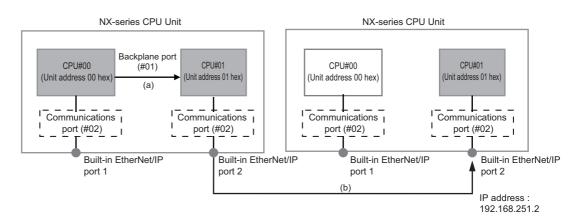


Local CPU #00 to destination IP address

Network type number: "02" (Output to communication port)
 Destination address: Specify the destination IP address

Route Path : 02\192.168.250.2

# (2) Using built-in EtherNet/IP port 2 (local CPU #01) (Local CPU #00 to destination CPU #01 via local CPU #01)



### (a) Local backplane to local CPU #01

Network type number: "01" (Output to Backplane port)

Destination address: "#01" (CPU#01)\*1

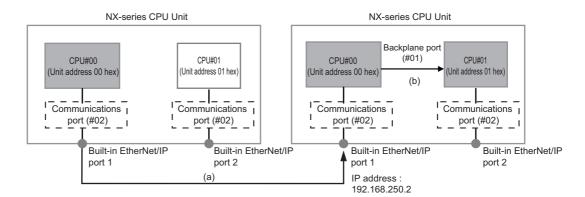
\*1 This is in order to output using the sender IP address of built-in EtherNet/IP port 2.

### (b) Local CPU #01 to destination IP address

Network type number: "02" (Output to communication port)
 Destination address: Specify the destination IP address

Route Path: 01\#01\02\192.168.250.2 (a) (b)

(3) Communicating with destination built-in EtherNet/IP port 2 (destination CPU #01) via destination built-in EtherNet/IP port 1 (destination CPU #00) (Local CPU #00 to destination CPU #01 via destination CPU #00)



### (a) Local CPU #00 to destination IP address

 Network type number: "02" (Output to communication port) · Destination address: Specify the destination IP address

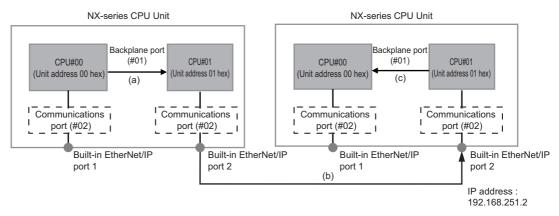
### (b) Destination backplane to destination CPU #01

 Network type number: "01" (Output to Backplane port)

· Destination address: "#01" (CPU#01)

Route Path: 02\192.168.250.2 \01\#01 (a)

(4) Communicating with destination built-in EtherNet/IP port 1 (destination CPU #00) via destination built-in EtherNet/IP port 2 (destination CPU #01) (Local CPU #00 to destination CPU #00 via destination CPU #01)



### (a) Local backplane to local CPU #01

• Network type number: "01" (Output to Backplane port)

"#01" (CPU#01)\*1 Destination address:

\*1 This is in order to output using the sender IP address of built-in EtherNet/IP port 2.

### (b) Local CPU #01 to destination IP address

Network type number: "02" (Output to communication port)
 Destination address: Specify the destination IP address

Route Path: 01\#01\02\192.168.251.2 (a) (b)

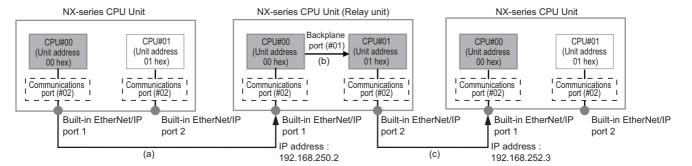
### (c) Destination CPU #01 to destination CPU #00

• Network type number: "01" (Output to Backplane port)

Destination address: "#00" (CPU#00)

Route Path: 01\#01\02\192.168.251.2\01\#00
(a) (b) (c)

# (5) Using an NX-series CPU as a relay unit (built-in EtherNet/IP port 1 to built-in EtherNet/IP port 2) (Local CPU #00 to destination CPU #00 via relay CPU #00 and relay CPU #01)



### (a) Local CPU #00 to relay IP address

• Network type number: "02" (Output to communication port)

Destination address: Specify the relay IP address

### (b) Relay backplane to relay CPU #01

• Network type number: "01" (Output to Backplane port)

Destination address: "#01" (CPU#01)\*1

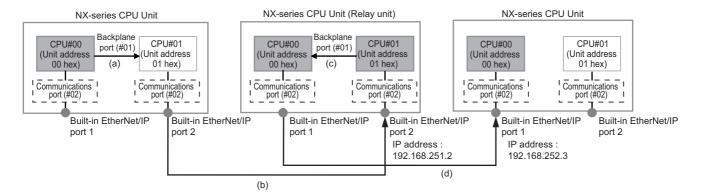
\*1 This is in order to output using the sender IP address of built-in EtherNet/IP port 2.

### (c) Relay CPU #01 to destination IP address

Network type number: "02" (Output to communication port)
 Destination address: Specify the destination IP address

Route Path: 02\192.168.250.2\01\#01\02\192.168.252.3 (a) (b) (c)

### (6) Using an NX-series CPU as a relay unit (built-in EtherNet/IP port 2 to built-in Ether-Net/IP port 1) (Local CPU #00 to destination CPU #00 via local CPU #01, relay CPU #01, and relay **CPU #00)**



### (a) Own CPU #00 to relay IP address

 Network type number: "01" (Output to Backplane port)

• Destination address: "#01" (CPU#01)\*1

This is in order to output using the sender IP address of built-in EtherNet/IP port 2.

### (b) Relay backplane to relay CPU #01

 Network type number: "02" (Output to communication port)

· Destination address: Specify the relay IP address

### (c) Relay backplane to relay CPU #00

 Network type number: "01" (Output to Backplane port)

"#00" (CPU#00)\*1 · Destination address:

\*1 This is in order to output using the sender IP address of built-in EtherNet/IP port 1.

### (d) Relay CPU #00 to destination IP address

 Network type number: "02" (Output to communication port) · Destination address: Specify the destination IP address

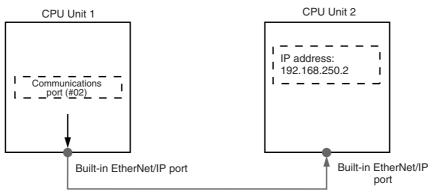
Route Path: 01\#01\02\192.168.251.2\01\#00\02\192.168.252.3 (a) (b) (c) (d)

### NJ-series CPU Unit

The notation of the route path is different for communications on the built-in EtherNet/IP port and for communication on an EtherNet/IP Unit. This section provides examples of route paths.

### Communicating between Built-in EtherNet/IP Ports

Example: Communicating between Built-in EtherNet/IP Ports on CPU Unit 1 and CPU Unit 2



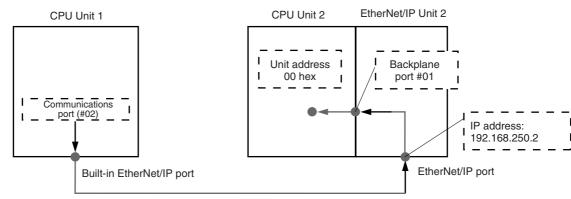
Network type number: "#02" (Output the command via EtherNet/IP port.)

Remote address: Specify the remote IP address.

• Route path: 02\192.168.250.2

### Communicating from a Built-in EtherNet/IP Port to an EtherNet/IP Unit

Example: Communicating from the built-in EtherNet/IP port on CPU Unit 1 to CPU Unit 2 via the EtherNet/IP Unit mounted to CPU Unit 2



### (1) CPU Unit 1 to EtherNet/IP Unit 2

- Network type number: "#02" (Output the command via EtherNet/IP port.)
- Remote address: Specify the remote IP address.

### (2) EtherNet/IP Unit 2 to CPU Unit 2

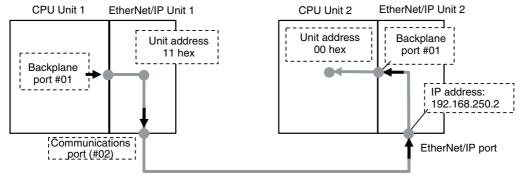
• Network type number: "#01" (Output the command via internal backplane port.)

• Remote address: "#00" (unit address of CPU Unit)

Route path :  $02\192.168.250.2\01\#00$ 

### Communicating between EtherNet/IP Units

Example: Communicating via EtherNet/IP Units Mounted to CPU Unit 1 and CPU Unit 2



### (1) CPU Unit 1 to EtherNet/IP Unit 1

- Network type number: "#01" (Output the command via internal backplane port.)
- Remote address: "#11" (Unit address of EtherNet/IP Unit (Unit number: 1+10 hex))

### (2) EtherNet/IP Unit 1 to EtherNet/IP Unit 2

- Network type number: "#02" (Output the command via EtherNet/IP port.)
- Remote address: Specify the remote IP address.

### (3) EtherNet/IP Unit 2 to CPU Unit 2

- Network type number: "#01" (Output the command via internal backplane port.)
- Remote address: "#0" (unit address of CPU Unit)

01\#11\02\192.168.250.2\01\#00 Route path (1) (2) (3)

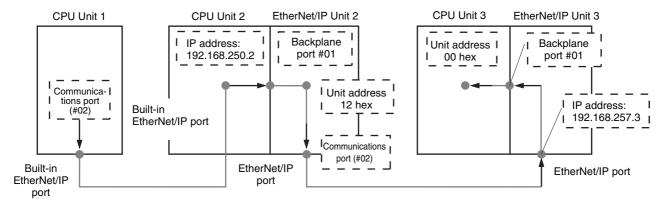


### **Version Information**

You can use the CJ1W-EIP21 EtherNet/IP Unit mounted to an NJ-series Controller with a CPU Unit with unit version 1.01 or later and Sysmac Studio version 1.02 or higher.

### Accessing via a Relay Node

Example: Communicating from CPU Unit 1 to CPU Unit 3 via CPU Unit 2



### (1) CPU Unit 1 to CPU Unit 2

- Network type number: "#02" (Output the command via EtherNet/IP port.)
- Remote address: Specify the remote IP address.

### (2) CPU Unit 2 to EtherNet/IP Unit 2

- Network type number: "#01" (Output the command via internal backplane port.)
- Remote address: "#12" (Unit address of EtherNet/IP Unit (Unit number: 2+10 hex = 12 hex))

### (3) EtherNet/IP Unit 2 to EtherNet/IP Unit 3

- Network type number: "#02" (Output the command via EtherNet/IP port.)
- Remote address: Specify the remote IP address.

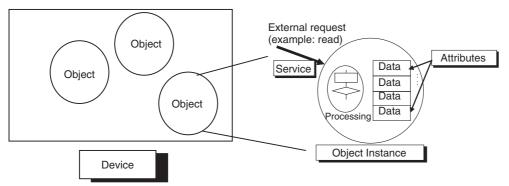
### (4) EtherNet/IP Unit 3 to CPU Unit 3

- Network type number: "#01" (Output the command via internal backplane port.)
- Remote address: "#00" (unit address of CPU Unit)

Route path : 02\192.168.250.2\01\#12\02\192.168.257.3\01\#00 (1) (2) (3) (4)

# **Request Path (IOI)**

In the CIP world, each device is modeled as a collection of objects. An Object abstractly represents the specific configuration elements of a device.

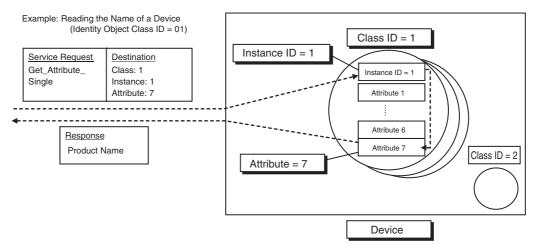


In the CIP Common Specification, "Object," "Class," "Instance," "Attribute" and "Service" are defined as follows: (Source: CIP Common Specification)

Term	Definition
Object	An abstract representation of a particular component within a device.
Class	A set of objects that all represent the same kind of system component.
Instance	A specific and real (physical) occurrence of an object.
Attribute	A description of an externally visible characteristic or feature of an object.
Service	A request from an external object (e.g., to read data).

### You use the Class ID, Instance ID, and Attribute ID to access an object.

You specify these three IDs to designate an object in a device. When you make a request from an external device for a service, you must specify the Class ID, Instance ID, and Attribute ID. (The Instance ID and Attribute ID are not required for some services.)



These are called Internal Object Identifiers (IOI) because they identify the Class ID, Instance ID, and Attribute ID within the device. Refer to 8-5 CIP Object Services for the class ID, instance ID, attribute ID, and service code for each object.

# 8-2-5 Preparing Derivative Data Types to Use CIP Communications Instructions

To use CIP communications instructions, you must prepare derivative data type variables to input communications parameter settings in advance. Refer to the *NJ/NX-series CPU Unit Software User's Manual* (Cat. No. W501) for the specifications and setting procedures for the variables that are described in this section.

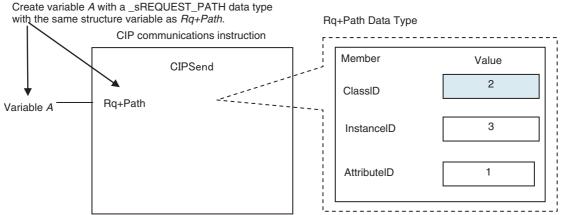
# **Providing the Structure Variables to Input Request Paths**

A CIP communications instruction combines all of the objects in a request path into a single structure variable.

### Creating Structure Variables

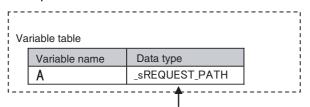
To input a value into the structure variable of a CIP communications instruction, you must create a user-defined variable with the same configuration in advance.

Example: Creating a Variable to Input Data to the CIPSend Instruction Array Variable Rq+Path



To create a variable in a variable table, select the pre-registered CIP communications instruction data type \_sREQUEST\_PATH. This is a structure variable with the same configuration as Rq+Path.

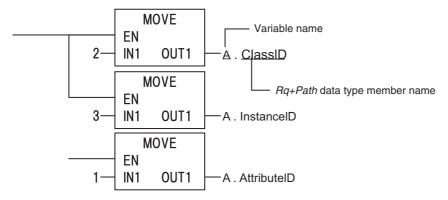
### Example:



Select the same data type as Rq+Path for the data type of variable A.

### Inputting the Values for Each Structure Variable Member

Input the following values into the communications parameters that were registered as members of the structure variable.



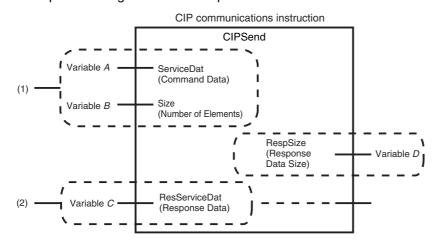
# Preparing Array Variables to Input and Output Service Data and Response Data

CIP communications instructions send and receive data that is stored in array variables.

### Creating Array Variables

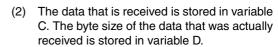
To input a value into the array variable of a CIP communications instruction, you must create a variable with the same configuration as the array variable in advance.

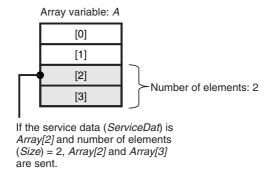
Example: Creating a Variable to Input Data to the CIPSend Instruction Array Variables

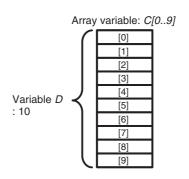


(1) Input the service data to send.

The data to send is stored in array variable A. If only certain elements are specified in array variable A, specify the number of elements in variable B.





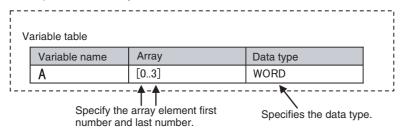


Use the following procedure to create a variable in the variable table.

Select the *Array* Check Box.

**2** Specify the element first number, the element last number, and the data type.

Example: UINT Array



# • CIP Communications Instructions That Use Array Variables

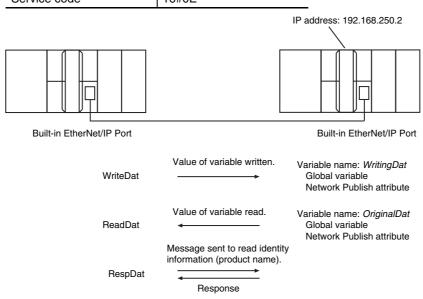
Instruction	Structure variable name		
Input variable		In-out variable	Output variable
CIPRead			DstDat (Read Data)
CIPWrite	SrcDat (Write Data)		
CIPSend	ServiceDat (Command Data)	RespServiceDat (Response Data)	

#### Sample Programming for CIP Connectionless (UCMM) Message 8-2-6 **Communications**

This sample uses CIP UCMM messages to write a variable, read a variable, and send a message. The Controllers are connected to an EtherNet/IP network. The IP address of the remote node is 192.168.250.2. The following procedure is used.

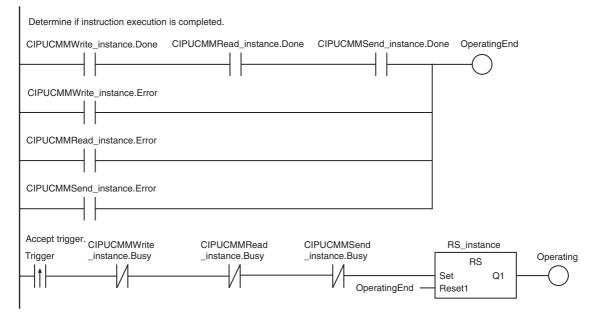
- The CIPUCMMWrite instruction is used to write the value of a variable at a remote node. The variable name at the remote node is WritingDat and the contents of the WriteDat is written to it. WritingDat must be defined as a global variable at the remote node and the Network Publish attribute must be set.
- The CIPUCMMRead instruction is used to read the value of a variable at a remote node. The value of the variable OriginalDat at the other node is read and the read value is stored in the ReadDat variable. OriginalDat must be defined as a global variable at the remote node and the Network Publish attribute must be set.
- 3 The CIPUCMMSend instruction is used to send an explicit message to a remote node. The contents of the message is to read identity information (product name). The class ID, instance ID, attribute ID, and service code are as follows. The response data is stored in the RespDat variable.

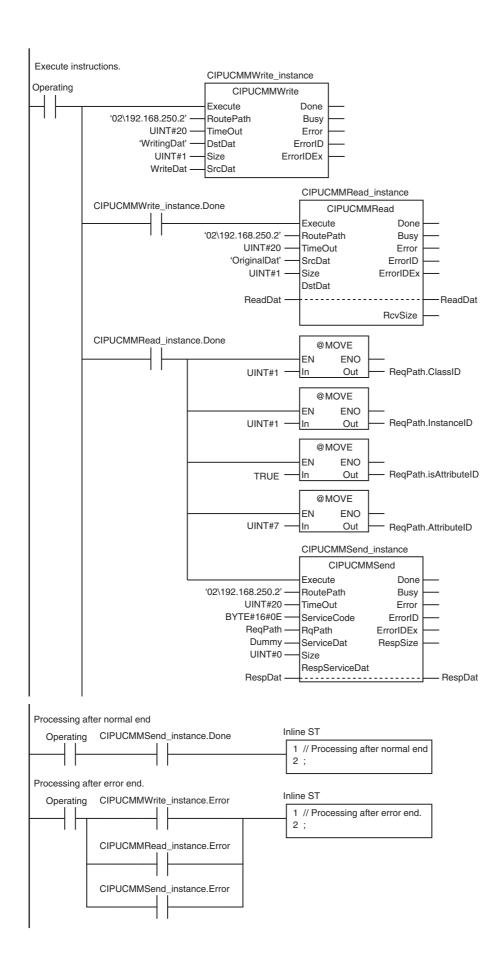
Item	Value
Class ID	1
Instance ID	1
Attribute ID	7
Service code	16#0F



### LD

Variable	Data type	Initial value	Comment
OperatingEnd	BOOL	False	Processing finished.
Trigger	BOOL	False	Execution condition
Operating	BOOL	False	Processing
WriteDat	INT	1234	Source data
ReadDat	INT	0	Read data
ReqPath	_sREQUEST_	(ClassID:=0, InstanceID:=0,	Request path
	PATH	isAttributeID:=False, AttributeID:=0)	
RespDat	ARRAY[010] OF BYTE	[11(16#0)]	Response data
Dummy	BYTE	16#0	Dummy
RS_instance	RS		
CIPUCMMWrite_instance	CIPUCMMWrite		
CIPUCMMRead_instance	CIPUCMMRead		
CIPUCMMSend_instance	CIPUCMMSend		





### ST

Internal Variables	Variable	Data type	Initial value	Comment
	Trigger	BOOL	False	Execution condition
	DoUCMMTrigger	BOOL	False	Processing
	Stage	INT	0	Status change
	WriteDat	INT	1234	Write data
	ReadDat	INT	0	Read data
	ReqPath	_sREQUEST_	(ClassID:=0, InstanceID:=0,	Request path
		PATH	isAttributeID:=False, AttributeID:=0)	
	RespDat	ARRAY[010] OF BYTE	[11(16#0)]	Response data
	Dummy	BYTE	16#0	Dummy
	CIPUCMMWrite_instance	CIPUCMMWrite		
	CIPUCMMRead_instance	CIPUCMMRead		
	CIPUCMMSend_instance	CIPUCMMSend		

External Variables	Variable	Data type	Constant	Comment
	_EIP_EtnOnlineSta*1	BOOL	<	Online

<sup>\*1</sup> For an NX-series CPU Unit, replace the variable with \_EIP1\_EtnOnlineSta (communication port 1 online) or \_EIP2\_EtnOnlineSta (communication port 2 online), depending on the built-in EtherNet/IP port which is used.

```
// Start sequence when Trigger changes to TRUE.
IF ((Trigger=TRUE) AND (DoUCMMTrigger=FALSE) AND (_Eip_EtnOnlineSta=TRUE))
THEN
  DoUCMMTrigger
                          :=TRUE;
                          :=INT#1;
  Stage
  CIPUCMMWrite_instance(
      Execute
                          :=FALSE,
                                                  // Initialize instance.
      SrcDat
                          :=WriteDat);
                                                  // Dummy
  CIPUCMMRead_instance(
                                                  // Initialize instance.
      Execute
                          :=FALSE,
                                                  // Dummy
      DstDat
                          :=ReadDat);
                                                  // Dummy
   CIPUCMMSend_instance(
      Execute
                          :=FALSE,
                                                  // Initialize instance.
      ServiceDat
                          := Dummy,
                                                  // Dummy
      RespServiceDat
                          :=RespDat);
                                                  // Dummy
END IF;
IF (DoUCMMTrigger=TRUE) THEN
  CASE Stage OF
   1:
                                                  // Request writing value of variable.
      CIPUCMMWrite_instance(
         Execute
                          :=TRUE,
         RoutePath
                          :='02\192.168.250.2',
                                                  // Route path
                                                  // Timeout value
         TimeOut
                          :=UINT#20,
         DstDat
                                                  // Source variable name
                          :='WritingDat',
         Size
                          :=UINT#1,
                                                  // Number of elements to write
                                                  // Write data
         SrcDat
                          :=WriteDat);
      IF (CIPUCMMWrite instance.Done=TRUE) THEN
                                                  // Normal end
```

:=INT#2;

Stage

```
ELSIF (CIPUCMMWrite instance.Error=TRUE) THEN
                          :=INT#10;
                                                 // Error end
         Stage
      END_IF;
  2:
                                                 // Request reading value of variable.
      CIPUCMMRead_instance(
         Execute
                          :=TRUE,
         RoutePath
                          :='02\192.168.250.2',
                                                 // Route path
         TimeOut
                          :=UINT#20,
                                                 // Timeout value
         SrcDat
                          :='OriginalDat',
                                                 // Source variable name
                                                 // Number of elements to read
         Size
                          :=UINT#1,
         DstDat
                          :=ReadDat);
                                                 // Read data
      IF (CIPUCMMRead_instance.Done=TRUE) THEN
                          :=INT#3;
                                                 // Normal end
      ELSIF (CIPUCMMRead_instance.Error=TRUE) THEN
                          :=INT#40:
                                                 // Error end
         Stage
      END_IF;
  3:
                                                 // Send message
      ReqPath.ClassID:=UINT#01;
      ReqPath.InstanceID:=UINT#01;
      ReqPath.isAttributeID:=TRUE;
      ReqPath.AttributeID:=UINT#07;
      CIPUCMMSend_instance(
         Execute
                          :=TRUE,
         RoutePath
                         :='02\192.168.250.2',
                                                 // Route path
         TimeOut
                         :=UINT#20,
                                                 // Timeout time
         ServiceCode
                         :=BYTE#16#0E,
                                                 // Service code
         RqPath
                         :=ReqPath,
                                                 // Request path
         ServiceDat
                         := Dummy,
                                                 // Service data
         Size
                          :=UINT#0,
                                                 // Number of elements
         RespServiceDat :=RespDat);
                                                 // Response data
      IF (CIPUCMMSend instance.Done=TRUE) THEN
                          :=INT#0:
                                                 // Normal end
         Stage
      ELSIF (CIPUCMMSend_instance.Error=TRUE) THEN
         Stage
                          :=INT#30:
                                                 // Error end
      END_IF;
  0:
                                                 // Processing after normal end
      DoUCMMTrigger:=FALSE;
      Trigger
                       :=FALSE;
  ELSE
                                                 // Processing after error end
      DoUCMMTrigger:=FALSE;
      Trigger
                       :=FALSE;
  END_CASE;
END_IF;
```

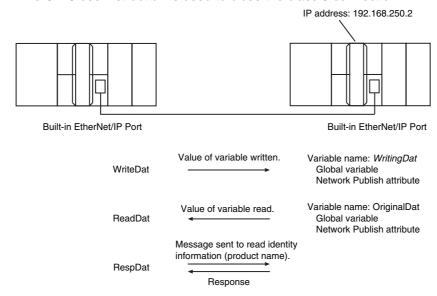
# 8-2-7 Sample Programming for CIP Connection (Class 3) Message Communications

This sample uses CIP class 3 messages to write a variable, read a variable, and send a message. The Controllers are connected to an EtherNet/IP network. The IP address of the remote node is 192.168.250.2. The following procedure is used.

- **1** The CIPOpen is used to open a class 3 connection (Large\_Forward\_Open). The timeout time is 2 s.
- The CIPWrite instruction is used to write the value of a variable at a remote node. The variable name at the remote node is *WritingDat* and the contents of the *WriteDat* is written to it. *WritingDat* must be defined as a global variable at the remote node and the Network Publish attribute must be set.
- The CIPRead instruction is used to read the value of a variable at a remote node. The value of the variable *OriginalDat* at the other node is read and the read value is stored in the *ReadDat* variable. *OriginalDat* must be defined as a global variable at the remote node and the Network Publish attribute must be set.
- The CIPSend instruction is used to send an explicit message to a remote node. The contents of the message is to read identity information (product name). The class ID, instance ID, attribute ID, and service code are as follows: The response data is stored in the RespDat variable.

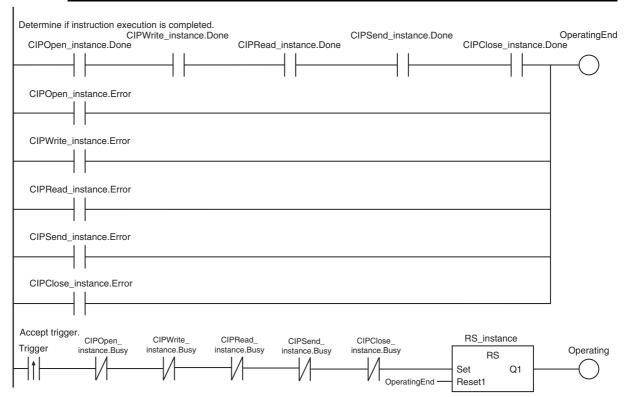
Item	Value
Class ID	1
Instance ID	1
Attribute ID	7
Service code	16#0E

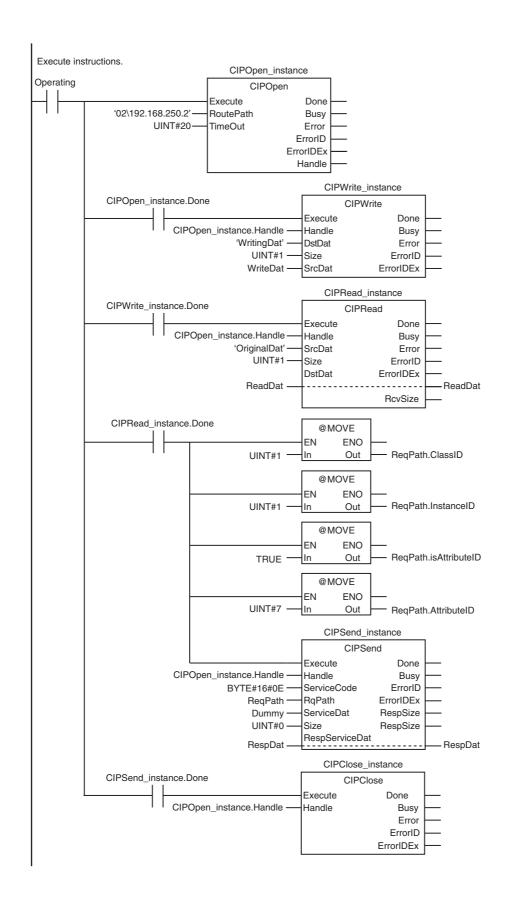
**5** The CIPClose instruction is used to close the class 3 connection.

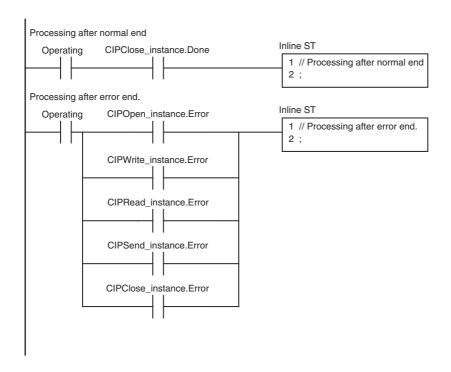


### LD

Variable	Data type	Initial value	Comment
OperatingEnd	BOOL	False	Processing finished.
Trigger	BOOL	False	Execution condition
Operating	BOOL	False	Processing
WriteDat	INT	1234	Source data
ReadDat	INT	0	Read data
ReqPath	_sREQUEST_ PATH	(ClassID:=0, InstanceID:=0, isAt-tributeID:=False, AttributeID:=0)	Request path
RespDat	ARRAY[010] OF BYTE	[11(16#0)]	Response data
Dummy	BYTE	16#0	Dummy
RS_instance	RS		
CIPOpen_instance	CIPOpen		
CIPWrite_instance	CIPWrite		
CIPRead_instance	CIPRead		
CIPSend_instance	CIPSend		
CIPClose_instance	CIPClose		







### ST

Internal Variables	Variable	Data type	Initial value	Comment
	Trigger	BOOL	False	Execution condition
	DoCIPTrigger	BOOL	False	Processing
	Stage	INT	0	Status change
	WriteDat	INT	1234	Write data
	ReadDat	INT	0	Read data
	ReqPath	_sREQUEST_	(ClassID:=0, InstanceID:=0,	Request path
		PATH	isAttributeID:=False, AttributeID:=0)	
	RespDat	ARRAY[010] OF BYTE	[11(16#0)]	Response data
	Dummy	BYTE	16#0	Dummy
	CIPOpen_instance	CIPOpen		
	CIPWrite_instance	CIPWrite		
	CIPRead_instance	CIPRead		
	CIPSend_instance	CIPSend		
	CIPClose_instance	CIPClose		

External Variables	Variable	Data type	Constant	Comment
	_EIP_EtnOnlineSta*1	BOOL	<	Online

<sup>\*1</sup> For an NX-series CPU Unit, replace the variable with \_*EIP1\_EtnOnlineSta* (communication port 1 online) or \_*EIP2\_EtnOnlineSta* (communication port 2 online), depending on the built-in EtherNet/IP port which is used.

// Initialize instance.

```
// Start sequence when Trigger changes to TRUE.
IF ((Trigger=TRUE) AND (DoCIPTrigger=FALSE) AND (_Eip_EtnOnlineSta=TRUE))THEN
   DoCIPTrigger:=TRUE;
   Stage
                    :=INT#1;
                                                  // Initialize instance.
   CIPOpen_instance(Execute:=FALSE);
   CIPWrite_instance(
                                                  // Initialize instance.
      Execute
                    :=FALSE,
      SrcDat
                    :=WriteDat);
                                                  // Dummy
   CIPRead_instance(
                                                  // Initialize instance.
                                                  // Dummy
      Execute
                    :=FALSE,
      DstDat
                    :=ReadDat);
                                                  // Dummy
   CIPSend_instance(
      Execute
                                                  // Initialize instance.
                    :=FALSE,
      ServiceDat
                    := Dummy,
                                                  // Dummy
      RespServiceDat :=RespDat);
                                                  // Dummy
```

CIPClose\_instance(Execute:=FALSE);

END\_IF;

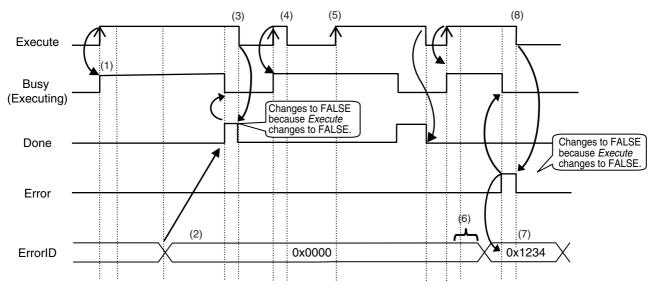
```
IF (DoCIPTrigger=TRUE) THEN
   CASE Stage OF
   1:
                                                  // Open CIP Class 3 Connection (Large_Forward_Open)
      CIPOpen_instance(
         Execute
                   :=TRUE,
         TimeOut :=UINT#20,
                                                  // Timeout time: 2.0 s
         RoutePath :='02\192.168.250.2');
                                                  // Route path
      IF (CIPOpen_instance.Done=TRUE) THEN
         Stage
                    :=INT#2;
                                                  // Normal end
      ELSIF (CIPOpen_instance.Error=TRUE) THEN
                    :=INT#10:
                                                  // Error end
         Stage
      END_IF;
   2:
                                                  // Request writing value of variable.
      CIPWrite instance(
         Execute
                   :=TRUE,
                                                  // Handle
         Handle
                    :=CIPOpen_instance.Handle,
         DstDat
                                                  // Source variable name
                    :='WritingDat',
         Size
                    :=UINT#1,
                                                  // Number of elements to write
                    :=WriteDat);
                                                  // Write data
         SrcDat
      IF (CIPWrite_instance.Done=TRUE) THEN
         Stage
                    :=INT#3;
                                                  // Normal end
      ELSIF (CIPWrite_instance.Error=TRUE) THEN
                    :=INT#20;
                                                  // Error end
         Stage
      END_IF;
   3:
                                                  // Request reading value of variable.
      CIPRead_instance(
         Execute
                   :=TRUE.
         Handle
                    :=CIPOpen_instance.Handle,
                                                  // Handle
         SrcDat
                   :='OriginalDat',
                                                  // Source variable name
         Size
                    :=UINT#1,
                                                  // Number of elements to read
         DstDat
                   :=ReadDat);
                                                  // Read data
      IF (CIPRead instance.Done=TRUE) THEN
                    :=INT#4;
                                                  // Normal end
         Stage
      ELSIF (CIPRead_instance.Error=TRUE) THEN
                                                  // Error end
         Stage
                    :=INT#30;
      END IF;
```

```
4:
                                                 // Send message
      ReqPath.ClassID :=UINT#01;
      ReqPath.InstanceID :=UINT#01;
      ReqPath.isAttributeID:=TRUE;
      ReqPath.AttributeID :=UINT#07;
      CIPSend_instance(
         Execute
                   :=TRUE,
         Handle
                   :=CIPOpen_instance.Handle,
                                                 // Handle
         ServiceCode:=BYTE#16#0E,
                                                 // Service code
         RqPath
                   :=ReqPath,
                                                 // Request path
         ServiceDat :=Dummy,
                                                 // Service data
         Size
                   :=UINT#0,
                                                 // Number of elements
         RespServiceDat:=RespDat);
                                                 // Response data
      IF (CIPSend_instance.Done=TRUE) THEN
         Stage
                   :=INT#5;
                                                 // Normal end
      ELSIF (CIPSend_instance.Error=TRUE) THEN
                   :=INT#40;
                                                 // Error end
         Stage
      END_IF;
  5:
                                                 // Request closing CIP class 3 connection.
      CIPClose_instance(
         Execute
                   :=TRUE,
         Handle
                   :=CIPOpen_instance.Handle);
                                                // Handle
      IF (CIPClose_instance.Done=TRUE) THEN
                   :=INT#0;
         Stage
      ELSIF (CIPClose_instance.Error=TRUE) THEN
         Stage
                   :=INT#50;
      END_IF;
  0:
                                                 // Processing after normal end
      DoCIPTrigger:=FALSE;
      Trigger
                   :=FALSE;
  ELSE
                                                 // Processing after error end
      DoCIPTrigger :=FALSE;
      Trigger
                   :=FALSE;
  END_CASE;
END_IF;
```

## 8-2-8 Operation Timing

# **Output Variable Operation and Timing**

You can monitor the values of the output variables to determine the status throughout instruction execution. The following timing chart shows the operation of the output variables.



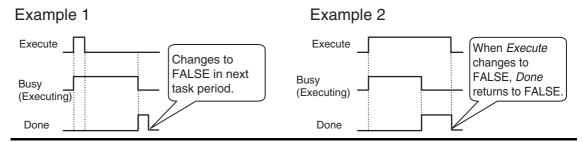
- **1** When *Execute* changes to TRUE, the instruction is executed and *Busy* changes to TRUE.
- **2** After the results of instruction execution are stored in the output variables, *Done* changes to TRUE and *Busy* changes to FALSE.
- **3** When Execute changes to FALSE, Done returns to FALSE.
- 4 When Execute changes to TRUE again, Busy changes to TRUE.
- **5** Execute is ignored if it changes to TRUE during instruction executed (i.e., when Busy is TRUE).
- **6** If an error occurs, several retries are attempted internally. The error code in the *ErrorID* is not updated during the retries.
- When a communications error occurs, *Error* changes to TRUE and the *ErrorID* is stored. Also, *Busy* and *Done* change to FALSE.
- **8** When *Execute* changes to FALSE, *Error* changes to FALSE.



### **Precautions for Correct Use**

If *Execute* changes back to FALSE before *Done* changes to TRUE, *Done* stays TRUE for only one task period. (Example 1)

If you want to see if *Done* is TRUE at any time, make sure to keep *Execute* TRUE until you confirm that *Done* is TRUE. If *Execute* is TRUE until *Done* changes to TRUE, *Done* stays TRUE until *Execute* changes to FALSE. (Example 2)

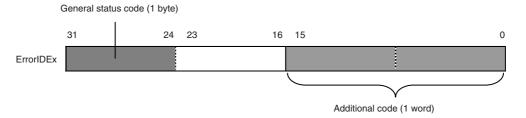


### 8-2-9 Response Codes

This section describes the response codes stored in the output variable *ErrorIDEx* if an error occurs during the execution of a CIP message communications instruction.

### **General Status Codes**

As response codes, general codes are stored in the *ErrorIDEx* output variable (DWORD data) after execution of a CIP communications instruction is completed. If an additional code is added, the additional code is also stored.



General sta- tus code (hex)	Status name	Description of status
00	Success	Service was successfully performed by the object specified.
01	Connection failure	A connection related to service failed along the connection path.
02	Resource unavailable	Resources needed for the object to perform the requested service were unavailable.
03	Invalid parameter value	See Status Code 20 hex.
04	Path segment error	The path segment identifier or the segment syntax was not understood by the processing node. Path processing stops when a path segment error occurs.
05	Path destination unknown	The path is referencing an object class, instance, or structure element that is not known or is not contained in the processing node. Path processing stops when a Path Destination Unknown Error occurs.
06	Partial transfer	Only part of the expected data was transferred.
07	Connection lost	The message connection was lost.

General sta- tus code (hex)	Status name	Description of status
08	Service not supported	The requested service was not supported or was not defined for this object class/instance.
09	Invalid attribute value	Invalid attribute data was detected.
0A	Attribute list error	An attribute in the Get_Attribute_List or Set_Attribute_List response has a non-zero status.
0B	Already in requested mode/state	The object is already in the mode/state being requested by the service.
0C	Object state conflict	The object cannot perform the requested service in its current mode/state.
0D	Object already exists	The requested instance of object to be created already exists.
0E	Attribute not settable	A request to modify a non-modifiable attribute was received.
0F	Privilege violation	A permission/privilege check failed.
10	Device state conflict	The device's current mode/state prohibits the execution of the requested service.
11	Reply data too large	The data to be transmitted in the response buffer is larger than the allocated response buffer.
12	Fragmentation of a primitive value	The service specified an operation that is going to fragment a primitive data value, i.e. half a REAL data type.
13	Not enough data	The requested service did not supply enough data to perform the specified operation.
14	Attribute not supported	The attribute specified in the request is not supported.
15	Too much data	The service supplied more data than was expected.
16	Object does not exist	An object that does not exist was specified for the requested service.
17	Service fragmentation sequence not in progress	The fragmentation sequence for this service is not currently active for this data.
18	No stored attribute data	The attribute data of this object was not saved prior to the requested service.
19	Store operation failure	The attribute data of this object was not saved due to a failure during the attempt.
1A	Routing failure (request packet too large)	The service request packet was too large for transmission on a network in the path to the destination. The routing device was forced to abort the service.
1B	Routing failure (response packet too large)	The service response packet was too large for transmission on a network in the path from the destination. The routing device was forced to abort the service.
1C	Missing attribute list entry data	The service did not supply an attribute in a list of attributes that was needed by the service to perform the requested behavior.
1D	Invalid attribute value list	The service is returning the list of attributes supplied with status information for those attributes that were invalid.
1E	Embedded service error	An embedded service resulted in an error.
1F	Vendor specific error	A vendor-specific error occurred. The Additional Code Field of the error response defines the error. This is a gen- eral error code that is used only for errors that do not cor- respond to any of the error codes in this table and are not in an object class definition.
20	Invalid parameter	A parameter for the requested service is invalid. This code is used when a parameter does not meet the requirements of the specification and/or the requirements defined in an application object specification.

General sta- tus code (hex)	Status name	Description of status
21	Write-once value or medium already written	An attempt was made to write to a write-once medium (e.g. WORM drive or PROM) that was previously written or cannot be changed.
22	Invalid Reply Received	An invalid reply was received. (For example, the reply service code does not match the request service code, or the reply message is shorter than the minimum expected reply size.) This status code is used for other causes of invalid replies.
23-24		Reserved by CIP for future extensions.
25	Key Failure in path	The key segment that was included as the first segment in the path does not match the destination module. The object specific status must indicate which part of the key check failed.
26	Path Size Invalid	The size of the path that was sent with the service request is either too large or too small for the request to be routed to an object.
27	Unexpected attribute in list	An attempt was made to set an attribute that is not able to be set at this time.
28	Invalid Member ID	The member ID specified in the request does not exist in the specified class, instance, and attribute.
29	Member not settable	A request to modify a non-modifiable member was received.
2A	Group 2 only server general failure	This error code is reported only by group 2 only servers with 4K or less of code space and only in place of Service not supported, Attribute not supported, or Attribute not settable.
2B-CF		Reserved by CIP for future extensions.
D0-FF	Reserved for Object Class and service errors	This range of error codes is to be used to indicate object class-specific errors. This code range is used only when none of the error codes in this table accurately reflect the error that occurred. The additional code field is used to describe the general error code in more detail.

# • Examples of Additional Status When General Status Is 01 Hex

(Status of Connection Manager Object)

General Sta- tus (hex)	Additional Status (hex)	Description
01	0100	Connection in use or duplicate forward open.
01	0103	Transport class and trigger combination not supported.
01	0106	Ownership conflict.
01	0107	Connection not found at target application.
01	0108	Invalid connection type. There is a problem with either the connection type or priority of the connection.
01	0109	Invalid connection size.
01	0100	Device not configured.
01	0111	RPI not supported. May also indicate problem with connection time-out multiplier, or production inhibit time.
01	0113	Connection Manager cannot support any more connections.
01	0114	Either the vendor ID or the product code in the key segment does not match the device.
01	0115	Device type in the key segment does not match the device.

General Sta- tus (hex)	Additional Status (hex)	Description
01	0116	Major or minor revision information in the key segment does not match the device.
01	0117	Invalid connection point.
01	0118	Invalid configuration format.
01	0119	Connection request failed because there is no controlling connection currently open.
01	011A	Target application cannot support any more connections.
01	011B	RPI is smaller than the production inhibit time.
01	0203	Connection cannot be closed because the connection has timed out.
01	0204	Unconnected_Send service timed out while waiting for a response.
01	0205	Parameter error in Unconnected_Send service.
01	0206	Message too large for unconnected message service.
01	0207	Unconnected acknowledgement without reply.
01	0301	No buffer memory available.
01	0302	Network bandwidth not available for data.
01	0303	No tag filters available.
01	0304	Not configured to send real-time data.
01	0311	Port that was specified in port segment is not available.
01	0312	Link address that was specified in port segment is not available.
01	0315	Invalid segment type or segment value in path.
01	0316	Path and connection were not equal when closing the connection.
01	0317	Either the segment is not present or the encoded value in the network segment is invalid.
01	0318	Link address to self is invalid.
01	0319	Resources on secondary are unavailable.
01	031A	Connection is already established.
01	031B	Direct connection is already established.
01	031C	Others
01	031D	Redundant connection mismatch.
01	031E	There are no more reception resources available on the sending module.
01	031F	No connection resources exist for the target path.
01	0320-07FF	Vendor specific.

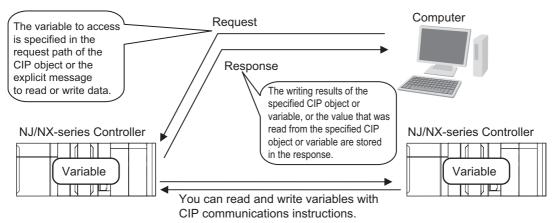
# 8-3 CIP Communication Server Functions

The CIP Communication Server functions are exclusively available for the NJ/NX-series Controllers. These functions execute services for a specified self-contained object in the CPU Unit after receiving the CIP messages from external devices.

This section provides information on CIP messages structure along with information about how to use CIP messages in a program that runs on a computer or by other means and uses the CIP Communication Server functions to perform following:

- Writing CIP objects and the values of variables to the NJ/NX-series Controller,
- Reading CIP objects and the values of variables from the NJ/NX-series Controller.

To read and write CIP objects or the values of variables between NJ/NX-series Controllers, use the CIP communications instructions. Refer to *8-2 CIP Messaging Communications Client Function* for information about how to use CIP communications instructions for CIP message communications.

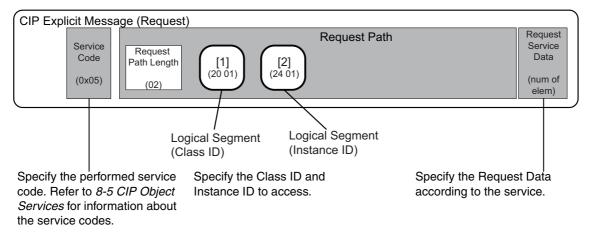


#### 8-3-1 CIP Message Structure for Accessing CIP Objects

This section shows how to specify messages to access CIP objects.

The CIP objects to access are expressed by connecting the segments that are defined in the CIP Common specifications in the request path field in a CIP explicit message.

Example: Performing the Reset service (0x05) to the Instance (01 hex) of the Identity object (class: 01



#### 8-3-2 **CIP Message Structure for Accessing Variables**

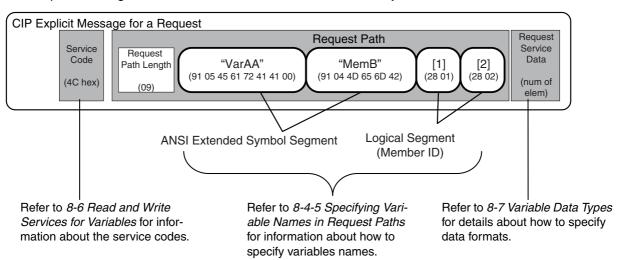
This section shows how to specify messages to access variables.

The variables to access are given by connecting the segments that are defined in the CIP Common specifications so that explicit message can be set in the request path field.

The following elements are combined to make the specification.

Specifying the variable to access: The elements are stored in the CIP segments and then joined to make the message.

Example: Reading the Present Value of One Member of the VarAA.MemB[1.2] Structure Variable Example for Using the CIP Read Data Service for a Variable Object



# 8-4 Request Path Used in the Server Function

The variable name, structure member name, and array index are specified for the request path. In CIP, the EPATH data type is used for the request path. With this method, the request path is divided into segments and a value is assigned to each segment. The request path notation shows the path to the final destination when the data segments are joined together. Each segment includes the segment type information and the segment data.

The first byte gives the interpretation method for the segment. It consists of two parts; a 3-bit segment type and a 5-bit segment format.

Segment Type					Se	egme	ent F	orm	at
7 6 5				4	3	2	1	0	

The segment type specifications are defined as follows in the CIP specifications.

Seg	Segment Type		Meaning
7	6	5	Wearing
0	0	0	Port Segment
0	0	1	Logical Segment
0	1	0	Network Segment
0	1	1	Symbolic Segment
1	0	0	Data Segment
1	0	1	Data Type
1	1	0	Data Type
1	1	1	Reserved

The specifications of segment format are different for each segment type. Use the segment format to request a service from a particular object of a particular device. Logical segments and data segments, which are needed to specify variables in CIP message communications, are described below.

# 8-4-1 Examples of CIP Object Specifications

Logical Segments are joined to form the request path that specifies the object to access.

Logical Segment (Class ID)	Logical Segment (Instance ID)	Logical Segment (Attribute ID)	
Specify the Class ID.	Specify the Instance ID.	Specify the Attribute ID.	

# 8-4-2 Examples of Variable Specifications

Segments are joined to form the request path that specifies the variable to access.

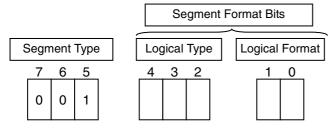
Data Segment (ANSI Extended Symbol Segment)	Logical Segment
, , , , , , , , , , , , , , , , , , , ,	

Specify the variable name and the member name.

Specify the array index.

#### **Logical Segment** 8-4-3

A logical segment is used to give the range of the CIP Object or variable (array) in the request path.



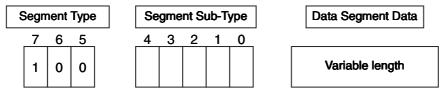
Lo	Logical Type		Mooning		
4	3	2	Meaning		
0	0	0	Class ID		
0	0	1	Instance ID		
0	1	0	Member ID		
0	1	1	Connection Point		
1	0	0	Attribute ID		
1	0	1	Special (Do not use the logical addressing definition for the Logical Format.)		
1	1	0	Service ID (Do not use the logical addressing definition for the Logical Format.)		
1	1	1	Reserved		

Logical	Format	Mooning					
1	0	Meaning Meaning					
0	0	8 bit logical address					
0	1	16 bit logical address					
1	0	32 bit logical address					
1	1	Reserved					

An 8-bit or 16-bit logical address can be used for the class ID and attribute ID. An 8-bit,16-bit, or 32-bit logical address can be used for the instance ID.

# 8-4-4 Data Segment

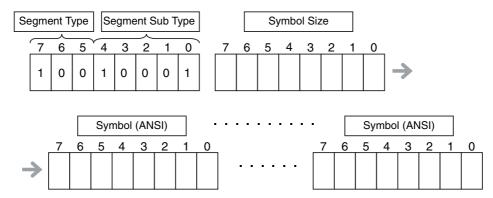
A data segment is used to give the specified variable name in the request path.



	Segment Sub-Type				Meaning	
4	3	2	1	0	Wealing	
0	0	0	0	0	Simple Data Segment	
1	0	0	0	1	ANSI Extended Symbol Segment	

A data segment is mainly used for an ANSI extended symbol segment. This segment sub-type is used to read and write the values of variables.

# **ANSI Extended Symbol Segment**



# 8-4-5 Specifying Variable Names in Request Paths

# **Variable Names**

A variable name is specified as a symbolic segment (ANSI extended symbol segment).

Variable Name Specification Format

BYTE 91 hex

BYTE Length in BYTE

Array of cotet Variable\_name
:

Octet (pad)

ANSI Extended Symbol Segment Length of variable name in bytes Variable name encoded in UTF-8

00 hex. One byte is padded if the variable name length is an odd number of bytes.

### Variable Names

Variable names are encoded in UTF-8.

### **Structure Member Names**

Structure member names are specified in the same way as variable names. (Store UTF-8 character codes in the ANSI extended symbol segment.)

# **Array Indices**

Specify the array index in a logical segment that is set as a member ID. You can specify an array index ([x]) in a variable name.

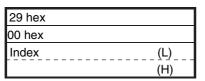
Specification Method 1: 8-bit Index

BYTE 28 hex
USINT Index

Logical Segment (Member ID) Array index from 0 to 255

Specification Method 2: 16-bit Index

BYTE octet UINT



Logical Segment (Member ID) Pad Array index from 0 to 65,535

# Range Specifications with the Num of Element Field

There is a Num of Element field in the request data for the variable read and variable write services. You can use these services to access the specified range of an array with the following specifications.

- Specify the first element in the range of elements to access in the array variable as the variable to read or write.
- Specify the number of elements to access in the Num of Element field.

### **Specification Examples**

This example shows how to specify *VarAA.MemB[1.2]* for the following structure variable.

Variable Name Specification Format

BYTE	91 hex				
BYTE	05 hex				
Array of	'V'				
octet	'a'				
	'r'				
	'A'				
	'A'				
Octet	00 hex				
BYTE	91 hex				
BYTE	04 hex				
Array of	'M'				
octet	'e'				
	'm'				
	'B'				
BYTE	28 hex				
USINT	01 hex				
BYTE	28 hex				
USINT	02 hex				

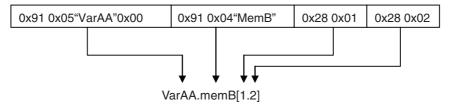
ANSI Extended Symbol Segment Length of variable name in bytes Variable name

Pad ANSI Extended Symbol Segment Length of variable name in bytes Variable name

Logical Segment (Member ID) Array index for first element Logical Segment (Member ID) Array index for second element

The variable name that is specified in the symbolic segment (ANSI extended symbol segment) must be converted to a text string to pass it to the communications thread. The following conversion rules apply.

### **Specification Example for Structure Members and Array Elements**



This example shows how to specify VarAA[1].MemB[1.2] for the following structure variable.

```
struct
{
  UINT
               MemA;
               MemB[10][10];
  BOOL
} VarAA[3];
```

#### Variable Name Specification Format

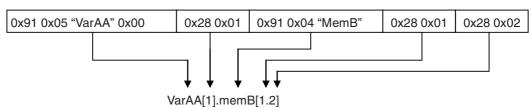
BYTE	91 hex					
BYTE	05 hex					
Array of	'V'					
octet	'a'					
	'r'					
	'A'					
	'A'					
Octet	00 hex					
BYTE	28 hex					
USINT	01 hex					
BYTE	91 hex					
BYTE	04 hex					
Array of	'M'					
octet	'e'					
	'm'					
	'B'					
BYTE	28 hex					
USINT	01 hex					
BYTE	28 hex					
USINT	02 hex					
CONT	OZ IIOA					

**ANSI Extended Symbol Segment** Length of variable name in bytes Variable name

Pad Logical Segment (Member ID) Array index ANSI Extended Symbol Segment Length of variable name in bytes Variable name

Logical Segment (Member ID) Array index for first element Logical Segment (Member ID) Array index for second element

#### Specification Example for Structure Array



# 8-5 CIP Object Services

This section shows services that specify the CIP object in the Request Path and access the CIP message server function of the NJ/NX-series Controllers.

### 8-5-1 CIP Objects Sent to the Built-in EtherNet/IP Port

The following CIP objects can be sent to an EtherNet/IP port.

Object name	Function	Reference
Identity object	Reads ID information from the CPU Unit.	page 8-47
	Resets the built-in EtherNet/IP port.	
TCP/IP interface object	Writes and reads TCP/IP settings.	page 8-51
Ethernet link object	Reads Ethernet settings.	page 8-54
	Reads Ethernet status.	
Controller object	Gets the Controller status.	page 8-58
	Changes the operating mode of the Controller.	

### 8-5-2 Identity Object (Class ID: 01 Hex)

This object reads the ID information of the CPU Unit and resets the built-in EtherNet/IP port. When using an NX-series CPU Unit, use the route path to specify the port number (1 or 2) of the built-in EtherNet/IP port to access.

### **Service Codes**

Specify the service to execute with the service code.

Service	Parameter name	Description	Supported services		
code	Parameter mame	Description	Classes	Instances	
01 hex	Get_Attribute_All	Reads the values of the attributes.	Supported.	Supported.	
0E hex	Get_Attribute_Single	Reads the value of the specified attribute.	Supported.	Supported.	
05 hex	Reset	Resets the built-in EtherNet/IP port. This parameter is used to reset the built-in EtherNet/IP port when you change the IP address or other parameter settings and want to apply them. Input one of the following values for the <i>ServiceDat</i> input variable to the CIPSend instruction to specify the reset method.  00 hex: Resets the built-in EtherNet/IP port.  02 hex*1: Clears the saved tag data link settings and resets the built-in EtherNet/IP port.	Not sup- ported.	Supported.	

<sup>\*1</sup> The value is 01 hex for a CPU Unit with unit version 1.09 or earlier.

### Class ID

Specify 01 hex.

### **Instance ID**

Specify 00 or 01 hex.

### **Attribute ID**

The attribute ID specifies the information to read.

#### Class Attribute ID

The class attribute ID specifies the attribute of the entire object.

Attribute ID				Read data		
	Parameter name	Description	Attribute	Data	Value	
				type	value	
01 hex	Revision	Revision of the object	Read	UINT	0001 hex	
02 hex	Max Instance	The maximum instance number	Read	UINT	0001 hex	

#### Instance Attribute ID

The instance attribute ID specifies the attribute of the instance.

				Read data		
Attribute ID	Parameter name	Description	Attribute	Data type	Value	
01 hex	Vendor ID	Vendor ID	Read	UINT	002F hex	
02 hex	Device Type	Device type	Read	UINT	000C hex	
03 hex	Product Code	Product code	Read	UINT	Refer to (1) Product Codes for Each Model, below	
04 hex	Revision	Device revision	Read	Struct		
	Major Revision	Major revision	Read	USINT	Refer to (2) Major and Minor CIP	
	Minor Revision	Minor revision	Read	USINT	Revisions for Individual Models, below	
05 hex	Status	Status of the built-in EtherNet/IP port	Read	WORD	Refer to (3) Status Details of the Built-in EtherNet/IP Port, below	
06 hex	Serial Number	Serial number	Read	UINT	Set value	
07 hex	Product Name	Product name	Read	STRING	Set value	

#### (1) Product Codes for Each Model

Туре	Product Code
NX701-□□□	067D hex
NJ501-13□□	0665 hex
NJ501-14□□	0666 hex
NJ501-15□□	0667 hex
NJ301-11□□	066B hex
NJ301-12□□	066C hex
NJ101-□□□□	0680 hex

#### (2) Major and Minor CIP Revisions for Individual Models

Model	Unit version	CIP revisions			
Wodei	Offic version	Major revision	Minor revision		
NX701-□□□□	Unit version 1.10 or later	02 hex	03 hex		
NJ501-□□□□	Unit version 1.00	01 hex	01 hex		
	Unit version 1.01 or 1.02	01 hex	03 hex		
	Unit version 1.03 to 1.08	02 hex	01 hex		
Unit version 1.09		02 hex	02 hex		
	Unit version 1.10 or later	02 hex	03 hex		

Model	Unit version	CIP revisions			
Woder	Offic version	Major revision	Minor revision		
NJ301-□□□□	Unit version 1.01 or 1.02	01 hex	03 hex		
	Unit version 1.03 to 1.08	02 hex	01 hex		
	Unit version 1.09	02 hex	02 hex		
	Unit version 1.10 or later	02 hex	03 hex		
NJ101-□□□□	Unit version 1.10 or later	02 hex	03 hex		

#### (3) Status Details of the Built-in EtherNet/IP Port

Bit	Name	Description							
0	Owned		Indicates when the built-in EtherNet/IP port has an open connection as the						
		_	target of a tag data link.						
1	Reserved	_	Always FALSE.						
2	Configured	Tag da			gs exi	st.			
3	Reserved	Always							
4 to 7	Extended Device Status	Indicates the status of the built-in EtherNet/IP port.							
		b7	b6	b5	b4				
		0	1	0	1	A major fault occurred.			
		0	0	1	0	A timeout occurred in one or more target connections.			
		0	0	1	1	Indicates that there are no tag data link settings.			
		0	1	1	0	Indicates that one or more connections are performing communications normally.			
		0	1	1	1	Other than the above.			
	Minar Dansverskia Fault	TDUE			ما ماد	Harring a green a corre			
8	Minor Recoverable Fault			-		llowing errors occurs.			
		• IP R			_				
						n Failed			
		• Tag			_				
		• Tag	Data I	₋ink Tiı	meout				
		• Tag	Data I	ink Co	nnect	ion Timeout			
		• FTP	Serve	er Setti	ng Err	or			
		• NTF	Clien	t Settir	ng Erro	or			
		• SNN	/IP Set	ting E	rror				
		• NTF	Serve	er Con	nectio	n Failed			
		• Tag	Name	Resol	ution I	Error			
9	Minor Unrecoverable Fault	TRUE	when	the fol	lowing	error occurs.			
		• Iden	itity Er	ror					
10	Major Recoverable Fault	TRUE	when	any of	the fo	llowing errors occurs.			
		• IP A	ddres	s Dupli	cation	Error			
				-		ction Error			
		Basi							
		• IP A			•				
11	Major Unrecoverable Fault	_				llowing errors occurs.			
	-			-		roller Failure			
		• MAC	C Addr	ess Er	ror				
12 to 15	Reserved	Always FALSE.							
	1	1/mayo 1/1EoE.							

### Request Paths (IOIs) to Specify Objects

When you specify an object, specify the request path (IOI) for each service code as given below.

Service code Class		Class ID	Instance ID	Attribute ID	
01 hex	Get_Attribute_All	01 hex	Specifying a service for a class: 00 hex	Not required.	
0E hex	Get_Attribute_Single		Specifying a service for an instance:     Always 01 hex	Reading a class attribute: 01 or 02 hex	
				Reading an instance attribute: 01 to 07 hex	
05 hex	Reset		Always 01 hex	Not required.	

### 8-5-3 TCP/IP Interface Object (Class ID: F5 hex)

This object is used to read and write settings such as the IP address, subnet mask, and default gateway. It is necessary to use the route path of the CIP communication command (in-out variable "Route-Path") to specify the port number (1 or 2) of the built-in EtherNet/IP port to access.

### **Service Codes**

Specify the service to execute with the service code.

Service	Parameter name	Description	Supported services		
code	rarameter mame	Description	Classes	Instances	
01 hex	Get_Attribute_All	Reads the values of the attributes.	Sup- ported.	Not sup- ported.	
0E hex	Get_Attribute_Single	Reads the value of the specified attribute.	Sup- ported.	Sup- ported.	
10 hex	Set_Attribute_Single	Writes a value to the specified attribute. The built-in EtherNet/IP port restarts automatically after the value is written to the attribute. When the restart process is not completed and the next Set_Attribute_Single is executed, the general status "OC hex" (Object State Conflict) is returned.	Not sup- ported.	Sup- ported.	

### **Class ID**

Specify F5 hex.

### Instance ID

Specify 00 or 01 hex.

### **Attribute ID**

The attribute ID specifies the information to read.

#### Class Attribute ID

The class attribute ID specifies the attribute of the entire object.

	Parameter name			Read data		
Attribute ID		Description	Attribute	Data type	Value	
01 hex	Revision	Revision of the object	Read	UINT	0001 hex: Unit version 1.01 or earlier	
					0002 hex: Unit version 1.02 or later	
02 hex	Max Instance	The maximum instance number	Read	UINT	0001 hex	

#### • Instance Attribute ID

The instance attribute ID specifies the attribute of the instance.

				Read/write data		
Attribute ID	Parameter name	Description	Attribute	Data type	Value	
01 hex	Interface Configuration Status	Indicates the IP address settings status of the built-in EtherNet/IP port.		DWORD	Bits 0 to 3: Interface Configuration Status:  0 = IP address is not set.	
		por a			(This includes when BOOTP is starting.)	
					1 = IP address is set.	
					Bits 4 and 5: Reserved (always FALSE).	
					Bit 6: AcdStatus:*1	
					FALSE = IP address collisions have not been detected.	
					TRUE = IP address collisions have been detected.	
					Bits 7 to 31: Reserved (always FALSE).	
02 hex	Configuration Capability	Indicates a Controller Configurations and Setup that can be set to	Read	DWORD	Bit 0: BOOTP Client: Always TRUE.	
		the built-in EtherNet/IP port.			Bit 1: DNS Client: Always TRUE.	
					Bit 2: DHCP Client: Always FALSE.	
					Bit 3: DHCP-DNS Update: Always FALSE.	
					Bit 4: Configuration Settable: Always TRUE.	
					Bit 5: Hardware Configurable: Always FALSE.	
					Bit 6: Interface Configuration Change Requires Reset: Always FALSE.	
					Bit 7: ACD Capable:*1 Always TRUE.	
					Bits 8 to 31: Reserved (always FALSE).	
03 hex	Configuration Control	Sets the method used to set the	Write	DWORD	Bit 0: Static IP address.	
		IP address when the EtherNet/IP port starts.			Bit 1: Set by BOOTP.	
04 hex	Physical Link Object	The path to the link object in the physical layer.	Read	Struct		
	Path size	The path size (WORD size).		UINT	0002 hex	
	Path	The path to the link object in the physical layer (static).		EPATH	20 F6 24 01 hex	
05 hex	Interface Configuration	The built-in EtherNet/IP port settings.	Write	Struct		
	IP Address	IP address.		UDINT	Set value	
	Network Mask	Subnet mask.	]	UDINT	Set value	
	Gateway Address	The default gateway.		UDINT	Set value	
	Nama Server	The primary name server.	]	UDINT	Set value	
	Nama Server2	The secondary name server.	]	UDINT	Set value	
	Domain Name	The domain name.		STRING	Set value	
06 hex	Host Name	The host name (reserved).	Write	STRING	Always 0000 hex.	

<sup>\*1</sup> The value is always FALSE for a CPU Unit with unit version 1.01 or earlier.

### Request Paths (IOIs) to Specify Objects

When you specify an object, specify the request path (IOI) for each service code as given below.

	Service code	Class ID	Instance ID	Attribute ID
01 hex	Get_Attribute_All	F5 hex	Specifying a service for a class: 00 hex	Not required.
0E hex	Get_Attribute_Single		Specifying a service for an instance: 01	Reading a class attribute: 01 or 02
10 hex	Set_Attribute_Single		hex	hex
				Reading and writing an instance attribute: 01 to 06 hex

#### **Ethernet Link Object (Class ID: F6 Hex)** 8-5-4

This object is used to set and read Ethernet communications and read Ethernet communications status information. It is necessary to use the route path of the CIP communication command (in-out variable "RoutePath") to specify the port number (1 or 2) of the built-in EtherNet/IP port to access.

### **Service Codes**

Specify the service to execute with the service code.

Service	Parameter name	Description	Supported services		
code	rafailletei fiaille	Description		Instances	
0E hex	Get_Attribute_Single	Reads the value of the specified attribute.	Supported.	Supported.	
10 hex	Set_Attribute_Single	Writes a value to the specified attribute.	Supported.	Supported.	
4C hex	Get_and_Clear	Specify Attribute4 or Attribute5 to reset the value of the attribute to 0.	Not sup- ported.	Supported.	

### **Class ID**

Specify F6 hex.

### **Instance ID**

Specify 00 or 01 hex.

### **Attribute ID**

The attribute ID specifies the information to read.

#### Class Attribute ID

The class attribute ID specifies the attribute of the entire object.

			Attribute	Read data		
Attribute ID	Parameter name	Description		Data	Value	
				type	value	
01 hex	Revision	Revision of the object	Read	UINT	0002 hex	
02 hex	Max Instance	The maximum instance number	Read	UINT	0001 hex	

#### Instance Attribute ID

The instance attribute ID specifies the attribute of the instance.

					Read/write data
Attribute ID	Parameter name	Description	Attribute	Data type	Value
01 hex	Interface Speed	Gives the baud rate for the built-in EtherNet/IP port.	Read	UDINT	Reads the current value.
02 hex	Interface Flags	Gives the status of the built-in EtherNet/IP port.	Read	DWORD	Refer to (1) Interface Flag Details, below.
03 hex	Physical Address	Gives the MAC address of the built-in EtherNet/IP port.	Read	ARRAY [05] OF USINT	Reads the current value of the MAC address.

				Read/write data		
Attribute ID	Parameter name	Description	Attribute	Data type	Value	
04 hex	Interface Counters	The path to the link object in the physical layer	Read	Struct		
	In Octets	The number of octets received through the interface.		UDINT	Reads the current value.	
		This includes unnecessary multi- cast packets and discarded pack- ets counted by <i>InDiscards</i> .				
	In Unicast Packets	The number of unicast packets received through the interface. This does not include discarded packets counted by <i>InDiscards</i> .		UDINT	Reads the current value.	
	In NonUnicast Packets	The number of packets besides unicast packets received through the interface. This includes unnecessary multicast packets, but does not include discarded packets counted by InDiscards.		UDINT	Reads the current value.	
	In Discards	The number of discarded incoming packets received through the interface.		UDINT	Reads the current value.	
	In Errors	The number of incoming packets that had errors. This is not included in <i>InDiscards</i> .		UDINT	Reads the current value.	
	In Unknown Protos	The number of incoming packets that were of an unknown protocol.		UDINT	Reads the current value.	
	Out Octets	The number of octets sent through the interface.		UDINT	Reads the current value.	
	Out Unicast Packets	The number of unicast packets sent through the interface.		UDINT	Reads the current value.	
	Out NonUnicast Packets	The number of packets besides unicast packets sent through the interface.	Read	UDINT	Reads the current value.	
	Out Discards	The number of discarded sent packets.		UDINT	Reads the current value.	
	Out Errors	The number of sent packets that had errors.		UDINT	Reads the current value.	

				Read/write data		
Attribute ID	Parameter name	Description	Attribute	Data type	Value	
05 hex	Media Counters	Media counters for the built-in EtherNet/IP port.	Read	Struct		
	Alignment Errors	Number of frames received that were not octets in length.		UDINT	Reads the current value.	
	FCS Errors	Number of frames received that did not pass the FCS check.		UDINT	Reads the current value.	
	Single Collisions	Number of frames sent successfully with only one collision.		UDINT	Reads the current value.	
	Multiple Collisions	Number of frames sent successfully with two or more collisions.		UDINT	Reads the current value.	
	SQE Test Errors	Number of times a SQE test error message was generated.		UDINT	Reads the current value.	
	Deferred Transmissions	The number of frames for which the first attempt to send was delayed because the media was busy.		UDINT	Reads the current value.	
	Late Collisions	The number of collisions detected in packets that were sent after 512 bit times.		UDINT	Reads the current value.	
	Excessive Collisions	The number of frames that failed to be sent because of excessive collisions.		UDINT	Reads the current value.	
	MAC Transmit Errors	The number of frames that failed to be sent due to an internal MAC sublayer transmission error.		UDINT	Reads the current value.	
	Carrier Sense Errors	The number of times the carrier sense condition was lost or the number of times an assertion did not occur when an attempt was made to send the frame.		UDINT	Reads the current value.	
	Frame Too Long	The number of frames received that exceeded the maximum allowed frame size.		UDINT	Reads the current value.	
	MAC Receive Errors	The number of frames that could not be received through the interface due to an internal MAC sublayer reception error.		UDINT	Reads the current value.	
06 hex	Interface Control	Counter for the built-in Ether- Net/IP port.	Write	Struct		
	Control Bits	Auto Nego for Ethernet communications that specifies full duplex.		WORD	Refer to (2) Control Bit Details, below.	
	Forced Interface Speed	Gives the set value of the Ethernet baud rate.		UINT	Reads the set value.	

### (1) Interface Flag Details

Bit	Name	Description
0	LinkStatus	FALSE: The link is down.
		TRUE: The link is up.
1	Half/FullDuplex	FALSE: Half duplex
		TRUE: Full duplex
2 to 4	Negotiation Status	00 hex: Auto-negotiation is in progress.
		01 hex: Auto-negotiation and speed detection failed.
		02 hex: Auto-negotiation failed, but speed detection succeeded.
		03 hex: Speed and duplex mode negotiation succeeded.
		04 hex: Auto-negotiation was not attempted.
5	Manual Setting Requires Speed	Always FALSE: Changes can be applied automatically.
6	Local Hardware Fault	Always FALSE
7 to 31	Reserved	Always FALSE

#### (2) Control Bit Details

Bit	Name	Description
0	Auto-negotiate	FALSE: Auto-negotiation is disabled.
		TRUE: Auto-negotiation is enabled.
1	ForcedDuplex Mode	FALSE: Half duplex
		TRUE: Full duplex
		* When auto-negotiation is enabled (bit 0 is TRUE), this should always be FALSE.
2 to 16	Reserved	Always FALSE

# Request Paths (IOIs) to Specify Objects

When you specify an object, specify the request path (IOI) for each service code as given below.

	Service code	Class ID	Instance ID	Attribute ID
0E hex	Get_Attribute_Single	F6 hex	Specifying a service for a class: 00 hex	Not required.
10 hex	Set_Attribute_Single		Specifying a service for an instance:     Always 01 hex	Reading a class attribute: 01 or 02 hex
				Reading and writing an instance attribute: 01 to 06 hex
4C hex	Get_and_Clear			Specify an attribute to reset its value to 0: 01 to 05 hex

#### **Controller Object (Class ID: C4 Hex)** 8-5-5

This object is used to get the status of the Controller or to change the operating mode of the Controller.

### **Service Codes**

Specify the service to execute with the service code.

Service	Parameter name	Description	Supported services	
code	rafametel mame	Description	Classes	Instances
0E hex	Get_Attribute_Single	Reads the value of the specified attribute.	Supported.	Not supported.
10 hex	Set_Attribute_Single	Writes a value to the specified attribute.	Supported.	Not supported.
51 hex	Reset_System_Alarm_All	Resets all errors in the CPU Unit.	Supported.	Not supported.

### **Class ID**

Specify C4 hex.

### **Instance ID**

Specify 00 hex.

#### Class Attribute ID

The class attribute ID specifies the attribute (value) of the entire object.

				Read/write data		
Attribute ID	Parameter name	Description	Attribute	Data type	Value	
01 hex	Revision	Revision of the object	Read	UINT	Always 0002 hex.	
02 hex	Max Instance	The maximum instance number	Read	UINT	Always 0001 hex	
64 hex	PLC Mode	This can be used to read and modify the Controller operating	Write	UINT	Specify this when you want to write to an attribute.	
		mode.			0000 hex: PROGRAM mode	
					0004 hex: RUN mode	
65 hex	PLC Error Status	Indicates when there is a Control- ler error. Changes to TRUE when	Read	UINT	0000 hex: There is no Controller error.	
		a fatal or non-fatal error occurs.			0001 hex: There is a Controller error.	
66 hex	PLC Model	Indicates the model of the Controller. The length is always 2 bytes for the size + 20 bytes for the name. Unused area is padded with spaces.	Read	STRING		

#### Instance Attribute ID

None

### **Request Paths (IOIs) to Specify Objects**

When you specify an object, specify the request path (IOI) for each service code as given below.

Service code		Class ID	Instance ID	Attribute ID
0E hex	Get_Attribute_Single	C4 hex	00 hex	Specifies the attribute of the class to
10 hex	Set_Attribute_Single			read or write: 01 hex, 02 hex, or 64 to 66 hex

## 8-6 Read and Write Services for Variables

This section shows services that specify the CIP object in the Request Path and access the CIP message server function of the NJ/NX-series Controllers.

#### 8-6-1 Read and Write Services for Variables

Specify service code 4C hex to read the value of the variable that is specified by the request path.

### **Service Code: 4C Hex**

### Request Data Format

Request Path Data	
Variable name specification	J

Specify the number of elements to read for an array variable.

Response Service Data

Data Type	USINT	Data type of variable to read
AddInfo Length	USINT	Additional information: Field length in bytes
(AddInfo)		Additional information: CRC value of structure
Actual data*	]	

<sup>\*</sup> The actual data is stored in little-endian format.

Data Type	Code for data type of variable to read	
	Refer to 8-7-1 Data Type Codes.	
AddInfoLength	The size of the AddInfo area is stored only when accessing a structure variable. Set 02 hex for a structure variable. Otherwise, set 00 hex.	
AddInfo	The CRC code of the structure definition is stored only when accessing a structure variable. In this case, the size of AddInfo will be 2 bytes.	
Actual data	The actual data is stored in little-endian format. If 0001 hex is specified for an array, the actual data is stored in the same format as when you access a variable with the data type of the elements of the array.	

#### Response Codes

CIP status	Meaning	Add status	Cause
00	SUCCESS		The service ended normally.
02	RESOURCE_UNAVAILABLE		The internal processing buffer is not available.
04	PATH_SEGMENT_ERROR		The request path specification is not correct.
05	PATH_DESTINATION_UNKNOWN		The variable specification is not correct.
0C	OBJECT_STATE_CONFLICT	8010	Downloading, starting up
		8011	There is an error in tag memory.
11	REPLY_DATA_TOO_LARGE		The response exceeds the maximum response length.
13	NOT_ENOUGH_DATA		The data length is too short for the specified service.
15	TOO_MUCH_DATA		The data length is too long for the specified service.
1F	VENDOR_SPECIFIC_ERROR	0101, 1101	The variable type specification is not correct.
		0101, 1103	The read start address exceeds the range of the variable area.
		0101, 1104	The read end address exceeds the range of the variable area.
		0101, 110B	There are too many elements.
		0101, 1004	The combined size of the variable type and read address is not correct.
		0104, 1101	A variable type is out of range.
		8001	An internal error occurred.
		800D	There is an error in the registered tag information.
		8014	An internal error occurred.
		8016	A variable is not correctly registered.
20	INVALID_PARAMETER	8017	More than one element was specified for a variable that does not have elements.
		8018	Zero elements or data that exceeded the range of the array was specified for an array.

### 8-6-2 Write Service for Variables

Specify service code 4D hex to write the value of the variable that is specified by the request path.

### **Request Data Format for Writing a Variable**

	Request Path Data		
Г	Variable name specification		
_		_	

Request Service Data

Data Type	
AddInfo Length	
(AddInfo)	
Num of Element	(L)
	(H)
Actual data*	

USINT Data type of variable to write

USINT Additional information: Field length in bytes
Additional information: CRC value of structure

UINT

<sup>\*</sup> Data to write: Store the data to write in little-endian format.

Data Type	Code for data type of variable to write
	Refer to 8-7 Variable Data Types.
AddInfoLength	Specify the size of the AddInfo area only when accessing a structure variable. Set 02 hex for a structure variable. Otherwise, set 00 hex.
AddInfo	The CRC code of the structure definition is specified only when accessing a structure variable. In this case, the size of AddInfo will be 2 bytes.
NumOfElement	Specifying the number of elements in the array. Do not specify 0000 hex (an error will occur). For variables other than arrays, set 0001 hex.
Actual data	Specify the actual data in little-endian format. If 0001 hex is specified for an array, specify the actual data in the same format as when you access a variable with the data type of the elements of the array.

Response Service Data: There is no response service data.

#### Response Codes

CIP status	Meaning	Add status	Cause
00	SUCCESS		The service ended normally.
02	RESOURCE_UNAVAILABLE		The internal processing buffer is not available.
04	PATH_SEGMENT_ERROR		The request path specification is not correct.
05	PATH_DESTINATION_UNKNOWN		The link was followed to the end, but the variable was not found.
0C	OBJECT_STATE_CONFLICT	8010	Downloading, starting up
		8011	There is an error in tag memory.
13	NOT_ENOUGH_DATA		The data length was too short for the specified service.
15	TOO_MUCH_DATA		The data length was too long for the specified service.
1F	VENDOR_SPECIFIC_ERROR	0102, 1003	The number of elements does not match the size of the write data.
		0102, 1101	The variable type specification is not correct.
		0102, 2101	A read-only area is included in the write area.
		8001	An internal error occurred. (An information inconsistency was detected in the interface in the Module.)
		800D	There is an error in the registered tag information.
		8014	An internal error occurred.
		8016	A variable is not correctly registered.
20	INVALID_PARAMETER	8017	More than one element was specified for a variable that does not have elements.
		8018	Zero elements or data that exceeded the range of the array was specified for an array.
		8022	The data type that is specified in the request service data does not agree with the tag information. The AddInfo Length in the request service data is not 0.

# 8-7 Variable Data Types

This section provides the data types of variables that can be used with CIP message communications.

### 8-7-1 Data Type Codes

The following codes are given to variable data types.

Data Type	Code(Hex)	Group*
Boolean (bit)	C1	CIP Common
SINT (1-byte signed binary)	C2	CIP Common
INT (1-word signed binary)	C3	CIP Common
DINT (2-word signed binary)	C4	CIP Common
LINT (4-word signed binary)	C5	CIP Common
USINT (1-byte unsigned binary)	C6	CIP Common
UINT (1-word unsigned binary)	C7	CIP Common
UDINT (2-word unsigned binary)	C8	CIP Common
ULINT (4-word unsigned binary)	C9	CIP Common
REAL (2-word floating point)	CA	CIP Common
LREAL (4-word floating point)	СВ	CIP Common
STRING	D0	CIP Common
BYTE (1-byte hexadecimal)	D1	CIP Common
WORD (1-word hexadecimal)	D2	CIP Common
DWORD (2-word hexadecimal)	D3	CIP Common
TIME (8-byte data)	DB	CIP Common
LWORD (4-word hexadecimal)	D4	CIP Common
Abbreviated STRUCT	A0	CIP Common
STRUCT	A2	CIP Common
ARRAY	A3	CIP Common
UINT BCD (1-word unsigned BCD)	04	Vendor Specific
UDINT BCD (2-word unsigned BCD)	05	Vendor Specific
ULINT BCD (4-word unsigned BCD)	06	Vendor Specific
ENUM	07	Vendor Specific
DATE_NSEC	08	Vendor Specific
TIME_NSEC	09	Vendor Specific
DATE_AND_TIME_NSEC	0A	Vendor Specific
TIME_OF_DAY_NSEC	0B	Vendor Specific
Union	0C	Vendor Specific

<sup>\* &</sup>quot;CIP Common" indicates codes that are defined in the CIP Common Specifications. "Vendor Specific" indicates codes that are assigned by OMRON.

#### 8-7-2 **Common Format**

The basic format on the data line is shown below.

#### **Data Format**

Data Type **USINT** USINT AddInfo Length (AddInfo) UINT Num of Element (L) (H) Actual data

Refer to Data Type Codes on page 8-63 for specific values.

Additional information: Field length in bytes

Additional information: CRC value of structure or other information

This field exists only in the parameters for the variable write service.

#### 8-7-3 **Elementary Data Types**

### **Fixed-length Byte Data**

Applicable data types: BYTE, USINT, and SINT

**Data Format** 

**USINT** Data Type **USINT** 00h UINT (L) Num of Elem (H) **USINT** Data **USINT** 0 (padding)

01 hex 00 hex

00 hex

### **Fixed-length 2-byte Data**

Applicable data types: INT, UINT, UINT BCD, and WORD

**Data Format** 

**USINT USINT** UINT

Data Type	
00h	
Num of Elem	(L)
 	(H)
Data	(L)
	(H)

01 hex 00 hex

### **Fixed-length 4-byte Data**

Applicable data types: DINT, UDINT, UDINT BCD, REAL, and DWORD

**Data Format** 

**USINT USINT** UINT

(L)
(H)
(LL)
(LH)
(HL)
(HH)

01 hex 00 hex

### **Fixed-length 8-byte Data**

Applicable data types: LINT, ULINT, ULINT BCD, LREAL, and LWORD

Data Format

USINT USINT UINT

Data Type	
00 hex	
Num of Elei	m (L)
	(H)
Data	(Least-significant byte)
	:
	:
	:
	:
	:
	:
	(Most-significant byte)

01 hex 00 hex

### **Boolean Data**

#### Data Format

USINT	Data Type		
USINT	00 hex		
UINT	Num of Elem (L)		
	(H)		
USINT	Status		
USINT	Forced set/reset information*		
* Chacify O when writing data			

<sup>\*</sup> Specify 0 when writing data.

C1 hex

01 hex 00 hex

01 hex: TRUE, 00 hex : FALSE 01 hex: Forced, 00 hex: Not forced

#### 8-7-4 **Derived Data Types**

Arrays and structures are handled as derived data types.

### Accessing One Member

The data format for accessing one element of an array or one member of a structure is the same as the data format for the corresponding elementary data type.

Example: If you specify Var[5] to access a variable defined with UINT Var[10], use the same data format as for UINT data.

### **Accessing More Than One Element at the Same Time**

#### Arrays

· Accessing an Entire Array

If you access an array variable without specifying an element, the entire array is accessed. The following data format is used.

#### **Data Format**

USINT	Data Type	
USINT	00 hex	
UINT	Num of Elem	(L)
		(H)
	Data	
	:	
	Data	

Data type of array elements (A1 hex is not used.)

Gives the number of elements in the array.

The actual data for the elements of the array are given in order in the same format as when the elements are accessed individually.\*

- \* For STRING data, the output format differs from the format when accessing individual elements in the following ways.
  - There are no fields for the text string lengths. Only the text strings (including NULL) are given.
  - The transferred data length is not the combined lengths of the text strings, but the memory size that is allocated to the STRING variable.
  - Upper and lower bytes are reversed.

Example: The outputs will be as follows for a STRING array named s that has two elements (with the data quantity around elements is set to 4 bytes) when s[0] is "ab" and s[1] is "d".

Individual [0]: D0 00 03 00 61 62 63 (hex)

Entire array: D0 00 62 61 ?? 00 00 64 ?? ?? (hex) (??: Invalid data.)

#### Handling Multi-dimensional Array

Elements for a multi-dimensional array are given in order from the deepest elements. For example, the data is read in the following format when *Var* is specified for a variable defined with *UINT Var*[2][2].

#### **Data Format**

USINT	C7 hex	
USINT	00 hex	
UINT	Value of <i>Var[0][0]</i>	(L)
		(H)
UINT	Value of <i>Var[0][1]</i>	(L)
		(H)
UINT	Value of <i>Var[1][0]</i>	(L)
		(H)
UINT	Value of Var[1][1]	(L)
		(H)

Data type code for UINT

The following data format is used for a BOOL array (using BOOL b[2][3] as an example).

#### **Data Format**

USINT USINT (WORD)

C1 hex (data type code for BOOL)							
00 hex							
rsv	rsv	b[1][2]	b[1][1]	b[1][0]	b[0][2]	b[0][1]	b[0][0]
rsv	rsv	rsv	rsv	rsv	rsv	rsv	rsv

#### · Exceptions When Specifying the Num of Element Field

The following data format is used if a specification is made in the Num of Element field for a BOOL array. (Refer to *8-4-5 Specifying Variable Names in Request Paths* for information on the Num of Element field.) The status (TRUE/FALSE) is given in order for each element of the BOOL variable.

#### **Data Format**

USINT	Data Type	
USINT	00 hex	
UINT	Num of Elem	(L)
	I	(H)
USINT	Status	
:	:	
USINT	Status	

C1 hex

Gives the number of elements in the array.

01 hex: TRUE, 00 hex: FALSE

#### Structure Variables

· Accessing an Entire Structure

If a structure variable is specified, it is treated as an access request for all of the members of the structure.

#### Data Format

USINT Dat
USINT 02 |
UINT CR

Data Type	
02 hex	
CRC	(L)
	(H)
Num of Elem	(L)
! 	(H)
:	
Data	
:	

A0 Hex (Abbreviated STRUCT)

CRC value for the structure definition

01 hex 00 hex



# **Socket Service**

9-1	Basic	Knowledge on Socket Communications	9-2
	9-1-1	Sockets	
	9-1-2	Port Numbers for Socket Services	9-2
9-2	Basic	Knowledge on Protocols	9-3
	9-2-1	Differences between TCP and UDP	9-3
	9-2-2	Fragmenting of Send Data	9-5
	9-2-3	Data Reception Processing	9-7
	9-2-4	Broadcasting	9-10
9-3	Overvi	iew of Built-in EtherNet/IP Port Socket Services	9-11
	9-3-1	Overview	9-11
	9-3-2	Procedure	9-11
9-4	Setting	gs Required for the Socket Services	9-12
	9-4-1	Settings Required for the Socket Services	9-12
9-5	Socke	t Service Instructions	9-13
9-6	Details	s on Using the Socket Services	9-14
	9-6-1	Using the Socket Services	
	9-6-2	Procedure to Use Socket Services	9-15
	9-6-3	Timing Chart for Output Variables Used in Communications	9-17
	9-6-4	UDP Sample Programming	9-19
	9-6-5	TCP Sample Programming	9-24
9-7	Precau	utions in Using Socket Services	9-30
	9-7-1	Precautions for UDP and TCP Socket Services	9-30
	9-7-2	Precautions for UDP Socket Services	9-30
	9-7-3	Precautions for TCP Socket Services	9-30

### **Basic Knowledge on Socket** 9-1 **Communications**

#### 9-1-1 **Sockets**

A socket is an interface that allows you to directly use TCP or UDP functions from the user program. In a host computer (e.g., personal computer), sockets are provided in the form of a C language interface library. If you load the library, you can program communications via TCP and UDP in the user program. In a UNIX computer, a socket interface is provided in the format of system calls. For the built-in Ether-Net/IP port, you execute instructions in the user program to use sockets. You can use these communications services to send and receive any data to and from remote nodes, i.e., between host computers and Controllers or between Controllers. The built-in EtherNet/IP port provides a UDP socket service and a TCP socket service.

#### 9-1-2 Port Numbers for Socket Services

Ports 0 to 1023 used for TCP/IP are reserved as well-known ports. In addition, ports 1024 to 49151 are reserved as registered ports by the protocols that are used. Therefore, we recommend that you use port numbers 49152 to 65535 for applications other than the protocols that are registered with the

You cannot specify port number 0 for the built-in EtherNet/IP port. Also, make sure that the following ports, which are used by the built-in EtherNet/IP port, are not used for other purposes.

Application	UDP	TCP
FTP server		20, 21*
DNS	53	
BOOTP client	68	
HTTP server		80
NTP server	123*	
SNMP Agent	161*	
SNMP trap	162*	
FINS/UDP	9600	
FINS/TCP		9600*
Used by system.	2222, 2223, and 2224	9610
CIP messages	44818	44818

<sup>\*</sup> These are the default port numbers. You can change the port numbers in the settings.

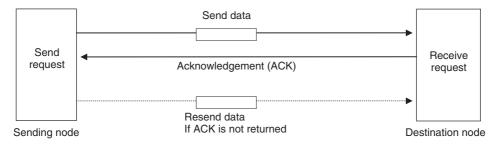
# 9-2 Basic Knowledge on Protocols

#### 9-2-1 Differences between TCP and UDP

The TCP and UDP functions used on socket services differ as shown below.

### **TCP Communications**

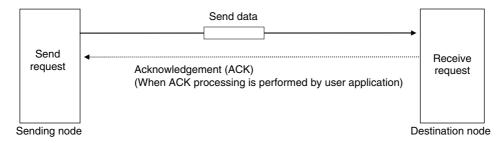
The following operations are performed each time data is sent to ensure that it reaches the destination node. The destination node returns an acknowledgement (ACK) when data is received normally. The sending node sends the next data after ACK is returned. It resends the same data if ACK is not received within a certain length of time.



In TCP, the remote IP address and the remote TCP port number are specified when a request is made to open a socket. The variables that store the data to send are specified when the send request is made.

### **UDP Communications**

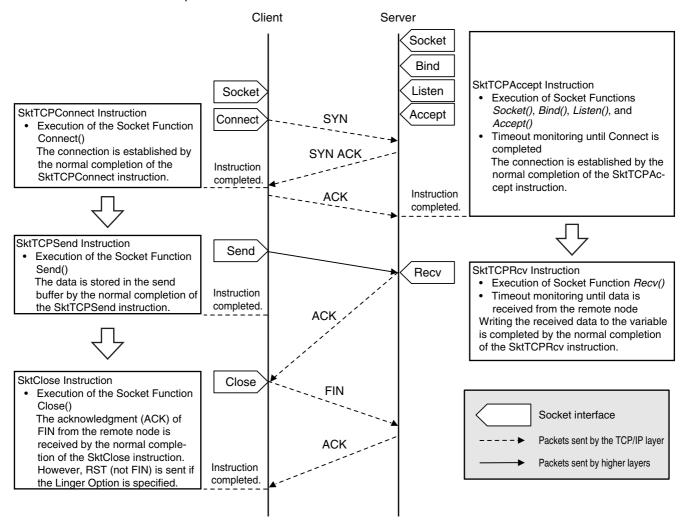
Data is simply sent to the destination node and no acknowledgement or resends are performed like they are for TCP. To increase the reliability of communications, the user application must perform data resend processing.



In UDP, the remote IP address and the remote UDP port number are not specified when a request is made to open a socket. The variables that store the remote IP address, the remote UDP port number, and the data to send are specified when the send request is made. (The received data is stored in the response data to specify which UDP port was used from which IP address.) Furthermore, once a socket is open in UDP, communications with other remote nodes is possible even if the socket is not closed.

### **TCP Communications Procedure**

You execute socket communications instructions in sequence to perform TCP communications for the built-in EtherNet/IP port.



### 9-2-2 Fragmenting of Send Data

The receive buffer for the built-in EtherNet/IP port is a maximum of 9,000 bytes per handle. The maximum number of handles that can be used at the same time is given in the following table. If any data that is larger than 9,000 bytes is received, any data over 9,000 bytes is discarded.

		Number of sockets	
UDP/TCP	NX-series CPU Unit	NJ-series	CPU Unit
	NA-Series CPO Utilit	Unit version 1.00 to 1.02	Unit version 1.03 or later
UDP socket service	Total of 30 sockets	Total of 16 sockets	Total of 30 sockets
TCP socket service			

Up to 2,000 bytes can be received for a single request. In this case, the data is sent in fragments as described below.

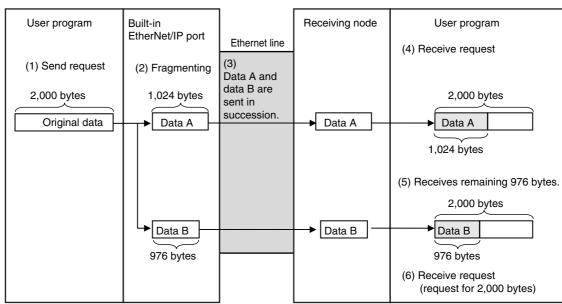
### **Using TCP**

The conditions shown in the following figure occur when data is sent in fragments in TCP communications.

- (1) A send request is sent from the user program at the sending node. It specifies a variable with a data length of 2,000 bytes.
- (2) The built-in EtherNet/IP port separates the send data into 1,024 bytes in data A and 976 bytes in data B.
- (3) Data A and data B are sent in sequence by the sending node.
- (4) After data A is received, the remaining data B is received.

Sending node (Controller)

Receiving node (host computer)



Data in TCP communications is delivered to the user program in a fragmented form, as shown above. The number of bytes of data that is received must be checked to confirm all data was received before the next receive request is made. (You can use the *RecvDataSize* output variable of the socket receive request instruction to check the received data.)



#### **Additional Information**

If TCP is used to send data to a different segment, the data is separated into 536-byte fragments.

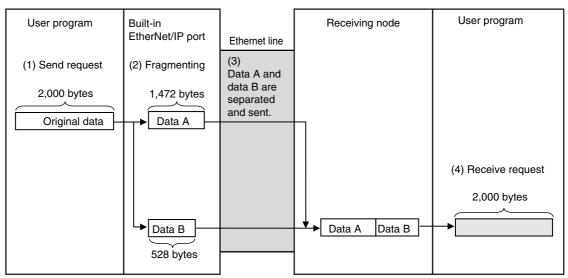
### **Using UDP**

The conditions shown in the following figure occur when data is sent in fragments in UDP communications.

- (1) A send request is sent from the user program at the sending node. It specifies a variable with a data length of 2,000 bytes.
- (2) The built-in EtherNet/IP port separates the send data into 1,472 bytes in data A and 528 bytes in data B.
- (3) Data A and data B are sent in sequence by the sending node.
- (4) Data A and data B are joined to restore the original send data, and the data is passed to the user program.

Sending node (Controller)

Receiving node (host computer)



Because UDP communications are performed in datagram units as shown above, send data is returned to the original data format before it is passed to the user program.

### 9-2-3 Data Reception Processing

This section describes data reception processing for TCP and UDP.

#### TCP Receive Processing

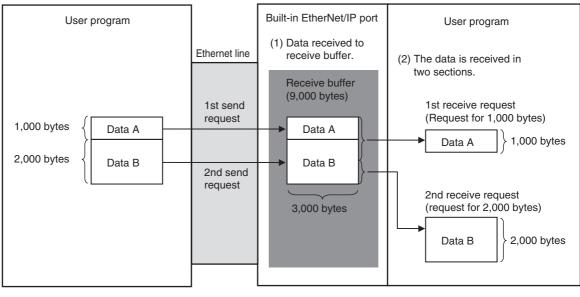
With TCP communications, the receive data that is stored in the receive buffer can be separated and received. (The receive buffer holds up to 9,000 bytes.) This can be used to receive data that exceeds the size that can be received with one data request (2,000 bytes), i.e., more than one receive request is used to receive all of the data. Also, all of the receive data can be received if there is less data in the receive buffer than the size of the variable that was specified in the receive request.

Example: Receiving 3,000 Bytes of Receive Data in Two Sections

- 1. The data that was sent from the sending node in two sends is stored in the receive buffer.
- 2. Two receive requests are used to receive all of the send data.

Sending node (host computer)

Receiving node (Controller)



#### UDP Receive Processing

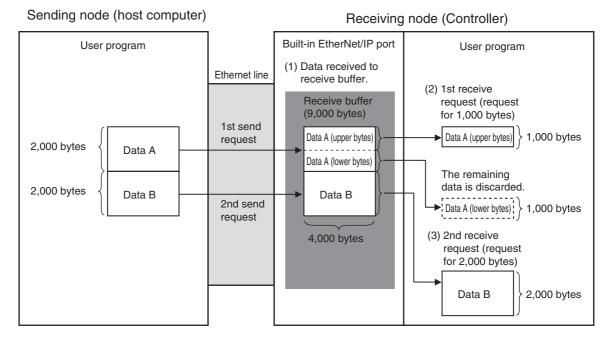
With UDP communications, the receive data that is stored in the receive buffer cannot be separated and received. (The receive buffer holds up to 9,000 bytes.) Therefore, the data that is sent for each send request must be received with one receive request. Because of this, attention must be given to the following at the receiving node when data is received.

• When the Size of the Variable Specified in the Receive Request Is Smaller Than the Data Size Sent with the Send Request

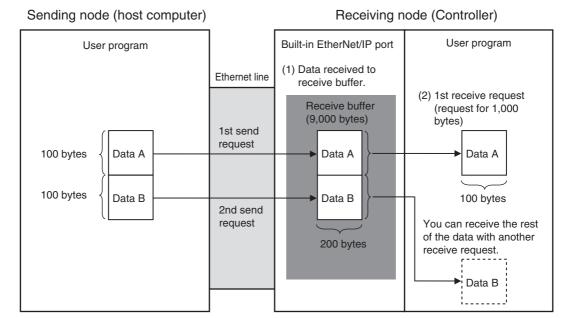
The received data that exceeds the size of the variable that is specified in the receive request is discarded. All of the receive data is received if there is less data in the receive buffer than the size of the variable that was specified in the receive request.

Example 1: Receive Request for 1,000 Bytes when 2,000 Bytes of Data Was Received

- 1. The data that was sent from the sending node in two sends is stored in the receive buffer.
- 2. If a receive request is made for 1,000 bytes for the first send data, the remaining 1,000 bytes is discarded.
- 3. If the next receive request is for 2,000 bytes, all of the send data for the second send is received.



- When There Is Only One Receive Request for Data Sent for Multiple Send Requests You cannot use just one receive request to receive the data that was sent for multiple send requests, regardless of the size of the data.
- Example 2: Receive Request for 1,000 Bytes when 200 Bytes of Data Was Sent for Two Send Requests
- 1. The data that was sent from the sending node in two sends is stored in the receive buffer.
- 2. Even if a receive request is made for 2,000 bytes of data, only the 100 bytes that was sent for the first send request is received.



#### **Broadcasting** 9-2-4

You can set the destination IP address to a broadcast address for a UDP socket to broadcast data to the host and all nodes on the same network as the EtherNet/IP port. If there is a router on the network, packets are not sent beyond the router. You can broadcast up to 1,472 bytes of data. Data larger than 1,472 bytes cannot be broadcast.

You can specify either of the two following types of broadcast addresses.

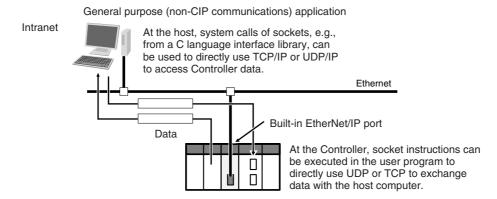
- Local Broadcast If no destination IP address is specified, the following IP address is specified automatically. Network segment: The network segment of the local IP address is set. Host segment: All bits are set to 1.
- · Global Broadcast Specify this type when the IP address of the local node or the subnet to which the local node belongs is unknown. As shown below, every bit of the 32-bit address is set to 1.

255.255.255.255

# 9-3 Overview of Built-in EtherNet/IP Port Socket Services

#### 9-3-1 Overview

Socket services on the built-in EtherNet/IP port are used to exchange data between Controllers and general-purpose applications that do not support CIP message communications. The Controller requests the socket service from the user program.



### **Overview of Socket Services with Socket Service Instructions**

You can use socket service instructions to use the socket services. The maximum total number of UDP and TCP sockets that you can use is given in the following table.

		Number of sockets	
UDP/TCP	NX-series CPU Unit	NJ-series CPU Unit	
	NA-Series CFO Offic	Unit version 1.00 to 1.02	Unit version 1.03 or later
UDP socket service	Total of 30 sockets	Total of 16 sockets	Total of 30 sockets
TCP socket service			

#### 9-3-2 Procedure

Make the settings that are required for socket services.

Refer to 9-4 Settings Required for the Socket Services.

2 Execute the socket service instructions from the user program.

Refer to 9-5 Socket Service Instructions.

### **Settings Required for the Socket** 9-4 **Services**

#### **Settings Required for the Socket Services** 9-4-1

Make the following settings in the Unit Setup to use the socket services.

Sysmac Studio Unit Settings Tab Page	Setting	Setting conditions
Setting	Local IP Address	Required
	Subnet Mask	Required
	TCP/IP Keep Alive	Optional (Change when the default setting of 5 minutes is unacceptable.)
	Linger Option	Optional



#### **Additional Information**

Make this setting in the TCP/IP Dialog Box. Refer to 5-1 TCP/IP Settings Display for information on the TCP/IP Dialog Box.

## 9-5 Socket Service Instructions

You can use the following socket service instructions for socket services use. Refer to the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502) for information on the socket service instructions.

UDP/TCP	Instruction	Socket service
UDP sockets	SktUDPCreate	Create UDP Socket instruction
	SktUDPRcv	UDP Socket Receive instruction
	SktUDPSend	UDP Socket Send instruction
TCP sockets	SktTCPAccept	Accept TCP Socket instruction
	SktTCPConnect	Connect TCP Socket instruction
	SktTCPRcv	TCP Socket Receive instruction
	SktTCPSend	TCP Socket Send instruction
	SktGetTCPStatus	Read TCP Socket Status instruction
Services for both	SktClose	Close UDP/TCP Socket instruction
UDP and TCP sockets	SktClearBuf	Clear UDP/TCP Socket Receive Buffer instruction



#### **Precautions for Correct Use**

You can execute a maximum of 32 socket service instructions at the same time. Use exclusive control in the user program so that no more than 32 socket service instructions are executed at the same time.

#### **Details on Using the Socket Services** 9-6

#### **Using the Socket Services** 9-6-1

The built-in EtherNet/IP port has the total number of sockets for TCP and UDP that is given in the following table.

	Number of sockets			
UDP/TCP	NX-series CPU Unit	NJ-series CPU Unit		
	NA-Series CFO Offic	Unit version 1.00 to 1.02	Unit version 1.03 or later	
UDP socket service	Total of 30 sockets	Total of 16 sockets	Total of 30 sockets	
TCP socket service				

To use these sockets for communications, special ST instructions for sockets are executed to execute the following processes.

Open processing: This process places the socket in a usable state. This is the first process that is exe-

cuted when using socket services. With TCP, open processing is performed until a

connection is established.

Close processing: This process ends the use of the socket. With TCP, it closes the connection.

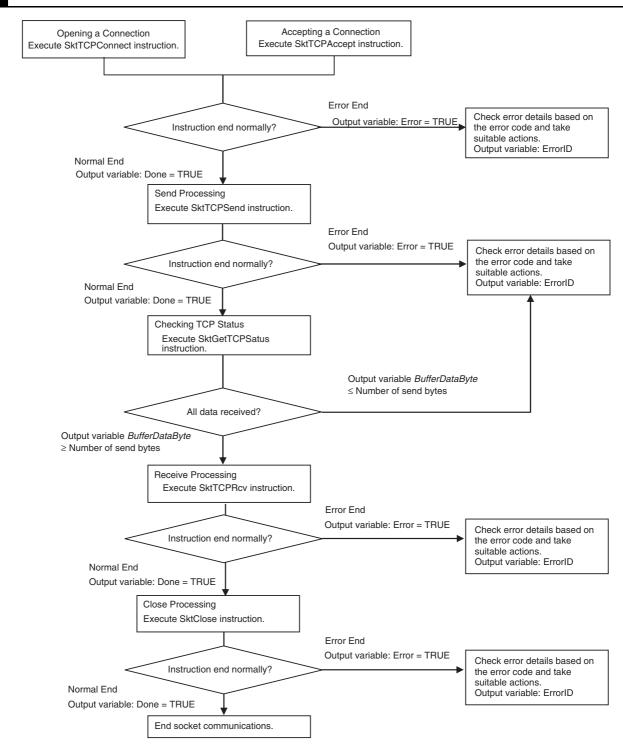
Send processing: This process sends data from the socket. Receive processing: This process receives data from the socket.

Clear processing: This process clears data in the receive buffer received from the remote node.

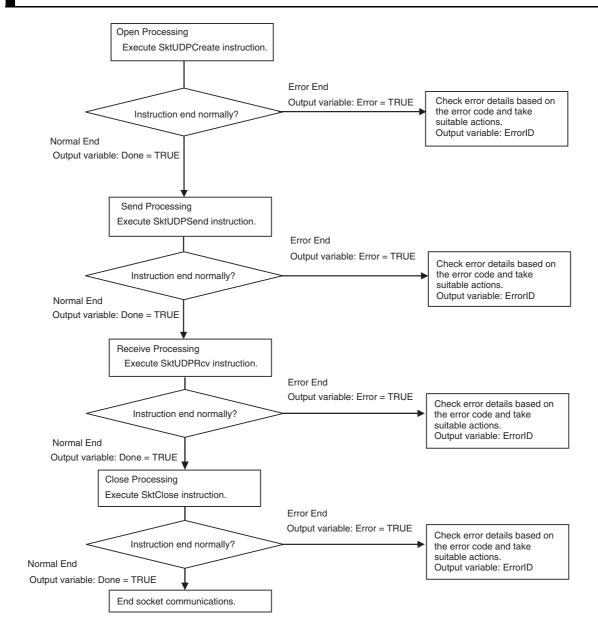
### 9-6-2 Procedure to Use Socket Services

You execute special instructions for sockets in sequence to use the socket services according to the procedure shown below. Use the values of the output variables for each instruction to confirm that each instruction is completed normally.

## TCP



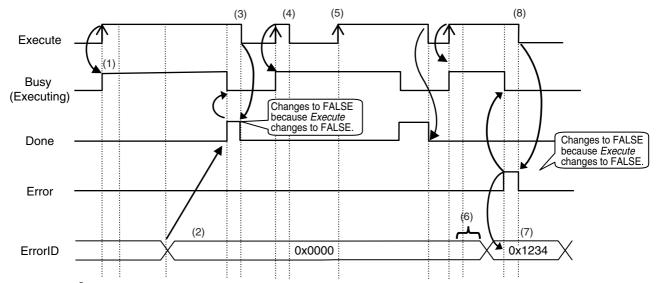
## **UDP**



## 9-6-3 Timing Chart for Output Variables Used in Communications

## **Output Variable Operation and Timing**

You can monitor the values of the output variables to determine the status throughout instruction execution. The following timing chart shows the operation of the output variables.



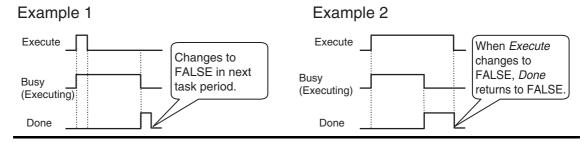
- **1** When *Execute* changes to TRUE, the instruction is executed and *Busy* changes to TRUE.
- **2** After the results of the instruction execution of the instruction are stored in the output variables, *Done* changes to TRUE and *Busy* changes to FALSE.
- **3** When Execute changes to FALSE, Done returns to FALSE.
- 4 When Execute changes to TRUE again, Busy changes to TRUE.
- **5** Execute is ignored if it changes to TRUE during instruction executed (i.e., when Busy is TRUE).
- **6** If an error occurs, several retries are attempted internally. The error code in the *ErrorID* is not updated during the retries.
- When a communications error occurs, *Error* changes to TRUE and the *ErrorID* is stored. Also, *Busy* and *Done* change to FALSE.
- **8** When *Execute* changes to FALSE, *Error* changes to FALSE.



#### **Precautions for Correct Use**

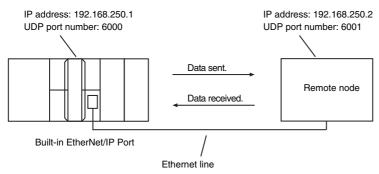
If Execute changes back to FALSE before Done changes to TRUE, Done stays TRUE for only one task period. (Example 1)

If you want to see if Done is TRUE at any time, make sure to keep Execute TRUE until you confirm that Done is TRUE. If Execute is TRUE until Done changes to TRUE, Done stays TRUE until Execute changes to FALSE. (Example 2)



## 9-6-4 UDP Sample Programming

In this sample, the UDP socket service is used for data communications between the NJ/NX-series Controller and a remote node.



The processing procedure is as follows:

- The SktUDPCreate instruction is executed to request creation of a UDP socket.
- 2 Execute the SktUDPSend instruction to request sending. The data in SendSocketDat[] is sent.
- **3** The SktUDPRcv instruction is executed to request reception. The receive data is stored in *Rcv-SocketDat[]*.
- **4** The SktClose instruction is executed to close the socket.

Internal Variables	Variable	Data type	Initial value	Comment
	Trigger	BOOL	False	Execution condition
	DoSendAndRcv	BOOL	False	Processing
	Stage	INT	0	Status change
	RcvSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Receive data
	WkSocket	_sSOCKET	(Handle:=0, SrcAdr:=(PortNo:=0, lpAdr:="), DstAdr:=(PortNo:=0, lpAdr:="))	Socket
	SendSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Send data
	SktUDPCreate_instance	SktUDPCreate		
	SktUDPSend_instance	SktUDPSend		
	SktUDPRcv_instance	SktUDPRcv		
	SktClose_instance	SktClose		

External Variables	Variable	Data type	Constant	Comment
_EIP_EtnOnlineSta*1		BOOL	>	Online

<sup>\*1</sup> For an NX-series CPU Unit, replace the variable with \_EIP1\_EtnOnlineSta (communication port 1 online) or \_EIP2\_EtnOnlineSta (communication port 2 online), depending on the built-in EtherNet/IP port to use.

```
// Start sequence when Trigger changes to TRUE.
IF((Trigger=TRUE) AND(DoSendAndRcv=FALSE) AND (_Eip_EtnOnlineSta=TRUE))THEN
  DoSendAndRcv :=TRUE;
  Stage
                   :=INT#1;
  SktUDPCreate_instance(Execute:=FALSE);
                                                 // Initialize instance.
  SktUDPSend_instance(
                                                 // Initialize instance.
     Execute
                   :=FALSE,
      SendDat
                   :=SendSocketDat[0]);
                                                 // Dummy
                                                 // Initialize instance.
  SktUDPRcv instance(
      Execute
                   :=FALSE,
      RcvDat
                   :=RcvSocketDat[0]);
                                                 // Dummy
  SktClose instance(Execute:=FALSE);
                                                 // Initialize instance.
END_IF;
IF (DoSendAndRcv=TRUE) THEN
  CASE Stage OF
  1:
                   // Request creating socket.
      SktUDPCreate_instance(
         Execute :=TRUE,
         SrcUdpPort:=UINT#6000,
                                                 // Local UDP port number
         Socket
                   =>WkSocket);
                                                 // Socket
      IF (SktUDPCreate_instance.Done=TRUE) THEN
                                                 // Normal end
         Stage
                   :=INT#2;
      ELSIF (SktUDPCreate_instance.Error=TRUE) THEN
                   :=INT#10;
                                                 // Error end
         Stage
      END_IF;
  2:
                   // Send request
      WkSocket.DstAdr.PortNo :=UINT#6001;
      WkSocket.DstAdr.lpAdr:='192.168.250.2';
      SktUDPSend_instance(
         Execute :=TRUE,
         Socket
                   :=WkSocket.
                                                 // Socket
         SendDat :=SendSocketDat[0],
                                                 // Send data
         Size
                   :=UINT#2000);
                                                 // Send data size
      IF (SktUDPSend_instance.Done=TRUE) THEN
         Stage
                   :=INT#3;
                                                 // Normal end
      ELSIF (SktUDPSend_instance.Error=TRUE) THEN
         Stage
                   :=INT#20:
                                                 // Error end
      END_IF;
```

```
3:
                   // Request receiving data.
      SktUDPRcv_instance(
         Execute
                   :=TRUE,
         Socket
                    :=WkSocket,
                                                  // Socket
         TimeOut :=UINT#0,
                                                  // Timeout value
         Size
                    :=UINT#2000,
                                                  // Receive data size
         RcvDat
                    :=RcvSocketDat[0]);
                                                  // Receive data
      IF (SktUDPRcv_instance.Done=TRUE) THEN
                    :=INT#4:
                                                  // Normal end
         Stage
      ELSIF (SktUDPRcv_instance.Error=TRUE) THEN
                    :=INT#30:
                                                  // Error end
         Stage
      END_IF;
                          // Request closing data.
      SktClose instance(
         Execute
                    :=TRUE,
                                                  // Socket
         Socket
                    :=WkSocket);
      IF (SktClose_instance.Done=TRUE) THEN
         Stage
                    :=INT#0;
                                                  // Normal end
      ELSIF (SktClose_instance.Error=TRUE) THEN
                   :=INT#40;
                                                  // Error end
         Stage
      END_IF;
  0:
                          // Normal end
      DoSendAndRcv:=FALSE;
      Trigger
                   :=FALSE;
  ELSE
                          // Interrupted by error.
      DoSendAndRcv:=FALSE;
      Trigger
                    :=FALSE;
  END_CASE;
END_IF;
```

#### Remote Node Programming

In this example, programming is also required in the remote node. The order of sending and receiving is reversed in comparison with the above procedure.

- The SktUDPCreate instruction is used to request creating a UDP socket.
- 2 The SktUDPRcv instruction is executed to request reception. The receive data is stored in *Rcv-SocketDat[*].
- 3 Execute the SktUDPSend instruction to request sending. The data in SendSocketDat[] is sent.
- **4** The SktClose instruction is executed to close the socket.

Internal Variables	Variable	Data type	Initial value	Comment
	Trigger	BOOL	False	Execution condition
	DoSendAndRcv	BOOL	False	Processing
	Stage	INT	0	Status change
	RcvSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Receive data
	WkSocket	_sSOCKET	(Handle:=0, SrcAdr:=(PortNo:=0, IpAdr:="), DstAdr:=(PortNo:=0, IpAdr:="))	Socket
	SendSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Send data
	SktUDPCreate_instance	SktUDPCreate		
	SktUDPSend_instance	SktUDPSend		
	SktUDPRcv_instance	SktUDPRcv		
	SktClose_instance	SktClose		

External Variables	Variable	Data type	Constant	Comment
_EIP_EtnOnlineSta*1		BOOL	>	Online

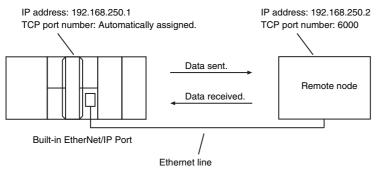
<sup>\*1</sup> For an NX-series CPU Unit, replace the variable with \_EIP1\_EtnOnlineSta (communication port 1 online) or \_EIP2\_EtnOnlineSta (communication port 2 online), depending on the built-in EtherNet/IP port to use.

```
// Start sequence when Trigger changes to TRUE.
IF((Trigger=TRUE) AND (DoSendAndRcv=FALSE) AND (_Eip_EtnOnlineSta=TRUE))THEN
  DoSendAndRcv :=TRUE;
                   :=INT#1;
  SktUDPCreate_instance(Execute:=FALSE);
                                                 // Initialize instance.
  SktUDPSend instance(
                                                 // Initialize instance.
      Execute
                   :=FALSE,
      SendDat:=SendSocketDat[0]);
                                                 // Dummy
  SktUDPRcv_instance(
                                                 // Initialize instance.
                   :=FALSE,
      Execute
      RcvDat
                   :=RcvSocketDat[0]);
                                                 // Dummy
  SktClose_instance(Execute:=FALSE);
                                                 // Initialize instance.
END_IF;
IF (DoSendAndRcv=TRUE) THEN
  CASE Stage OF
  1:
                         // Request creating socket.
      SktUDPCreate instance(
         Execute :=TRUE,
         SrcUdpPort:=UINT#6001,
                                                 // Local UDP port number
         Socket
                   =>WkSocket);
                                                 // Socket
      IF (SktUDPCreate_instance.Done=TRUE) THEN
         Stage
                   :=INT#2;
                                                 // Normal end
      ELSIF (SktUDPCreate instance.Error=TRUE) THEN
                                                // Error end
                   :=INT#10;
         Stage
      END_IF;
```

```
2:
                         // Request receiving data.
      SktUDPRcv_instance(
         Execute :=TRUE,
         Socket
                   :=WkSocket,
                                                // Socket
         TimeOut :=UINT#0,
                                                // Timeout value
         Size
                   :=UINT#2000,
                                                 // Receive data size
         RcvDat
                   :=RcvSocketDat[0]);
                                                // Receive data
      IF (SktUDPRcv_instance.Done=TRUE) THEN
                   :=INT#3;
                                                 // Normal end
         Stage
      ELSIF (SktUDPRcv_instance.Error=TRUE) THEN
         Stage
                   :=INT#20;
                                                 // Error end
      END_IF;
                         // Send request
  3:
      WkSocket.DstAdr.PortNo:=UINT#6000;
      WkSocket.DstAdr.lpAdr :='192.168.250.1';
      SktUDPSend_instance(
         Execute :=TRUE,
         Socket
                   :=WkSocket,
                                                // Socket
                                                // Send data
         SendDat :=SendSocketDat[0],
                   :=UINT#2000);
                                                 // Send data size
         Size
      IF (SktUDPSend_instance.Done=TRUE) THEN
                   :=INT#4;
                                                 // Normal end
      ELSIF (SktUDPSend_instance.Error=TRUE) THEN
                   :=INT#30;
                                                 // Error end
         Stage
      END_IF;
  4:
                         // Request closing data.
      SktClose instance(
         Execute :=TRUE,
         Socket
                   :=WkSocket);
                                                 // Socket
      IF (SktClose_instance.Done=TRUE) THEN
         Stage
                   :=INT#0;
                                                 // Normal end
      ELSIF (SktClose instance.Error=TRUE) THEN
         Stage
                   :=INT#40;
                                                 // Error end
      END_IF;
  0:
                         // Normal end
      DoSendAndRcv:=FALSE;
      Trigger
                   :=FALSE;
  ELSE
                         // Interrupted by error.
      DoSendAndRcv:=FALSE;
                   :=FALSE;
      Trigger
  END_CASE;
END IF;
```

#### **TCP Sample Programming** 9-6-5

In this sample, the TCP socket service is used for data communications between the NJ/NX-series Controller and a remote node.



The processing procedure is as follows:

- The SktTCPConnect instruction is executed to request connecting to the TCP port on the remote node.
- The SktClearBuf instruction is executed to clear the receive buffer for a TCP socket.
- The SktGetTCPStatus instruction is executed to read the status of a TCP socket.
- The SktTCPSend instruction is executed to request sending data. The data in SendSocketDat[] is sent.
- The SktTCPRcv instruction is executed to request receiving data. The receive data is stored in RcvSocketDat[].
- The SktClose instruction is executed to close the socket.

Internal Variables	Variable	Data type	Initial value	Comment
	Trigger	BOOL	False	Execution condition
	DoTCP	BOOL	False	Processing
	Stage	INT	0	Status change
	RcvSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Receive data
	WkSocket	_sSOCKET	(Handle:=0,SrcAdr:=(PortNo:=0, lpAdr:="), DstAdr:=(PortNo:=0, lpAdr:="))	Socket
	SendSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Send data
	SktTCPConnect_ instance	SktTCPConnect		
	SktClearBuf_instance	SktClearBuf		
	SktGetTCPStatus_ instance	SktGetTCPStatus		
	SktTCPSend_instance	SktTCPSend		
	SktTCPRcv_instance	SktTCPRcv		
	SktClose_instance	SktClose		

External Variables	Variable	Data type	Constant	Comment
_	_EIP_EtnOnlineSta	BOOL	>	Online

```
// Start sequence when Trigger changes to TRUE.
IF ((Trigger=TRUE) AND (DoTCP=FALSE) AND (_Eip_EtnOnlineSta=TRUE)) THEN
  DoTCP
                    :=TRUE;
  Stage
                    :=INT#1;
  SktTCPConnect_instance(Execute:=FALSE);
                                                  // Initialize instance.
  SktClearBuf_instance(Execute:=FALSE);
                                                  // Initialize instance.
  SktGetTCPStatus_instance(Execute:=FALSE);
                                                  // Initialize instance.
  SktTCPSend_instance(
                                                  // Initialize instance.
      Execute
                    :=FALSE,
      SendDat
                    :=SendSocketDat[0]);
                                                  // Dummy
  SktTCPRcv_instance(
                                                  // Initialize instance.
      Execute
                   :=FALSE.
      RcvDat
                    :=RcvSocketDat[0]);
                                                  // Dummy
   SktClose_instance(Execute:=FALSE);
                                                  // Initialize instance.
END_IF;
IF (DoTCP=TRUE) THEN
  CASE Stage OF
   1:
                          // Request a connection.
      SktTCPConnect_instance(
         Execute
                   :=TRUE,
                                                  // Local TCP port number: Automatically assigned.
         SrcTcpPort :=UINT#0,
                   :='192.168.250.2',
                                                  // Remote IP address
         DstAdr
         DstTcpPort:=UINT#6000,
                                                  // Destination TCP port number
                   =>WkSocket);
         Socket
                                                  // Socket
      IF (SktTCPConnect_instance.Done=TRUE) THEN
         Stage
                    :=INT#2;
                                                  // Normal end
      ELSIF (SktTCPConnect_instance.Error=TRUE) THEN
                    :=INT#10;
                                                  // Error end
         Stage
      END IF;
  2:
                          // Clear receive buffer.
      SktClearBuf_instance(
         Execute
                   :=TRUE,
         Socket
                    :=WkSocket);
                                                  // Socket
      IF (SktClearBuf_instance.Done=TRUE) THEN
         Stage
                    :=INT#3;
                                                  // Normal end
      ELSIF (SktClearBuf_instance.Error=TRUE) THEN
                    :=INT#20;
                                                  // Error end
         Stage
      END_IF;
```

```
3:
                      // Request reading status.
   SktGetTCPStatus_instance(
      Execute :=TRUE,
      Socket
                :=WkSocket);
                                              // Socket
   IF (SktGetTCPStatus_instance.Done=TRUE) THEN
                :=INT#4;
                                             // Normal end
      Stage
   ELSIF (SktGetTCPStatus_instance.Error=TRUE) THEN
      Stage
                :=INT#30;
                                              // Error end
   END_IF;
4:
                       // Send request
   SktTCPSend_instance(
      Execute
               :=TRUE,
      Socket
                :=WkSocket,
                                             // Socket
      SendDat :=SendSocketDat[0],
                                              // Send data
                :=UINT#2000);
                                              // Send data size
      Size
   IF (SktTCPSend_instance.Done=TRUE) THEN
      Stage
                :=INT#5;
                                              // Normal end
   ELSIF (SktTCPSend_instance.Error=TRUE) THEN
                :=INT#40;
                                              // Error end
      Stage
   END_IF;
                       // Request receiving data.
5:
   SktTCPRcv_instance(
      Execute :=TRUE,
      Socket
                :=WkSocket,
                                              // Socket
      TimeOut :=UINT#0,
                                              // Timeout value
      Size
                :=UINT#2000,
                                              // Receive data size
      RcvDat :=RcvSocketDat[0]);
                                              // Receive data
   IF (SktTCPRcv_instance.Done=TRUE) THEN
      Stage
                :=INT#6:
                                              // Normal end
   ELSIF (SktTCPRcv_instance.Error=TRUE) THEN
      Stage
                :=INT#50;
                                              // Error end
   END IF;
6:
                      // Request closing data.
   SktClose_instance(
      Execute :=TRUE.
      Socket
                :=WkSocket);
                                              // Socket
   IF (SktClose instance.Done=TRUE) THEN
      Stage
                :=INT#0;
                                              // Normal end
   ELSIF (SktClose_instance.Error=TRUE) THEN
      Stage
                :=INT#60;
                                              // Error end
   END_IF;
```

0: // Normal end

DoTCP :=FALSE; Trigger :=FALSE;

ELSE // Interrupted by error.

DoTCP :=FALSE; Trigger :=FALSE;

END\_CASE;

END\_IF;

### Remote Node Programming

In this example, programming is also required in the remote node. The order of sending and receiving is reversed in comparison with the above procedure.

- The SktTCPAccept instruction is executed to request accepting a TCP socket.
- **2** The SktTCPRcv instruction is executed to request receiving data. The receive data is stored in *RcvSocketDat[]*.
- **3** The SktTCPSend instruction is executed to request sending data. The data in SendSocketDat[] is sent.
- **4** The SktClose instruction is executed to close the socket.

Internal Variables	Variable	Data type	Initial value	Comment
	Trigger	BOOL	False	Execution condition
	DoTCP	BOOL	False	Processing
	Stage	INT	0	Status change
	RcvSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Receive data
	WkSocket	_sSOCKET	(Handle:=0, SrcAdr:=(PortNo:=0,lpAdr:="), DstAdr:=(PortNo:=0,lpAdr:="))	Socket
	SendSocketDat	ARRAY[01999] OF BYTE	[2000(16#0)]	Send data
	SktTCPAccept_instance	SktTCPAccept		
	SktTCPSend_instance	SktTCPSend		
	SktTCPRcv_instance	SktTCPRcv		
	SktClose_instance	SktClose		

	External Variables	Variable	Data type	Constant	Comment
_EIP_EtnOnlineSta		BOOL	<b>&gt;</b>	Online	

```
// Start sequence when Trigger changes to TRUE.
IF ((Trigger=TRUE) AND (DoTCP=FALSE) AND (_Eip_EtnOnlineSta=TRUE)) THEN
  DoTCP
                          :=TRUE;
  Stage
                          :=INT#1;
  SktTCPAccept_instance(Execute:=FALSE);
                                                 // Initialize instance.
  SktTCPSend_instance(
                                                 // Initialize instance.
     Execute
                   :=FALSE,
      SendDat
                   :=SendSocketDat[0]);
                                                 // Dummy
                                                 // Initialize instance.
  SktTCPRcv instance(
     Execute
                   :=FALSE,
      RcvDat
                   :=RcvSocketDat[0]);
                                                 // Dummy
  SktClose instance(Execute:=FALSE);
                                                 // Initialize instance.
END_IF;
IF (DoTCP=TRUE) THEN
  CASE Stage OF
  1:
                          // Request accepting a socket connection.
      SktTCPAccept_instance(
         Execute :=TRUE,
         SrcTcpPort:=UINT#6000,
                                                 // Local TCP port number
                                                 // Timeout value
         TimeOut :=UINT#0,
                                                 // Socket
         Socket
                 =>WkSocket);
      IF (SktTCPAccept_instance.Done=TRUE) THEN
                   :=INT#2;
                                                 // Normal end
      ELSIF (SktTCPAccept_instance.Error=TRUE) THEN
                   :=INT#10;
                                                 // Error end
         Stage
      END_IF;
                          // Request reception.
  2:
      SktTCPRcv_instance(
         Execute :=TRUE,
         Socket
                   :=WkSocket,
                                                 // Socket
         TimeOut :=UINT#0,
                                                 // Timeout value
         Size
                  :=UINT#2000,
                                                 // Receive data size
         RcvDat :=RcvSocketDat[0]);
                                                 // Receive data
      IF (SktTCPRcv_instance.Done=TRUE) THEN
         Stage
                   :=INT#3;
                                                 // Normal end
      ELSIF (SktTCPRcv_instance.Error=TRUE) THEN
                   :=INT#20:
                                                 // Error end
         Stage
      END_IF;
```

```
3:
                         // Send request
      SendSocketDat:=RcvSocketDat;
      SktTCPSend_instance(
         Execute :=TRUE,
                                                // Socket
         Socket
                   :=WkSocket,
         SendDat :=SendSocketDat[0],
                                                // Send data
                                                // Send data size
         Size
                   :=UINT#2000);
      IF (SktTCPSend_instance.Done=TRUE) THEN
                   :=INT#4;
                                                // Normal end
         Stage
      ELSIF (SktTCPSend_instance.Error=TRUE) THEN
         Stage
                   :=INT#30;
                                                // Error end
      END_IF;
  4:
                         // Request closing data.
      SktClose_instance(
         Execute
                   :=TRUE,
         Socket
                   :=WkSocket);
                                                // Socket
      IF (SktClose_instance.Done=TRUE) THEN
                   :=INT#0;
                                                // Normal end
         Stage
      ELSIF (SktClose_instance.Error=TRUE) THEN
                   :=INT#40;
                                                // Error end
         Stage
      END_IF;
  0:
                         // Normal end
      DoTCP
                   :=FALSE;
      Trigger
                   :=FALSE;
  ELSE
                         // Interrupted by error.
      DoTCP
                   :=FALSE;
      Trigger
                   :=FALSE;
  END_CASE;
END IF;
```

# **Precautions in Using Socket Services**

#### **Precautions for UDP and TCP Socket Services** 9-7-1

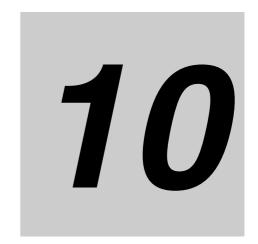
- Communications processing are sometimes delayed when multiple functions of the built-in Ether-Net/IP port are used simultaneously or due to the contents of the user program.
- · Communications efficiency is sometimes reduced by high communications traffic on the network line.
- · The close processing for a close request instruction discards all of the buffered send and receive data for the socket. For example, send data from a send request instruction immediately before the close processing is sometimes not sent.
- After a socket is open, the built-in EtherNet/IP port provides a receive buffer of 9,000 bytes per TCP socket and 9,000 bytes per UDP socket to enable data to be received at any time. If the receive buffer is full, data received by that socket is discarded. Make sure that the user application always executes receive requests to prevent the internal buffer from becoming full.

#### **Precautions for UDP Socket Services** 9-7-2

- The destination IP address can be set to a broadcast address for a UDP socket to broadcast data to all nodes on the network. However, in this case, the maximum length of send data is 1,472 bytes. Data lengths broken into multiple fragments (1,473 bytes or more in UDP) cannot be sent.
- For UDP socket, controls to confirm the reliability of communications, such as the confirmation of send data, are not performed. To improve the reliability of communications when you use UDP sockets, make sure the user program confirms that data is sent and resends data when necessary.

#### 9-7-3 **Precautions for TCP Socket Services**

- If the TCP socket is closed on the remote node without warning during communications (i.e., if the connection is closed), the socket at the local node must also be closed. You can use the Read TCP Socket Status instruction (SktGetTCPstatus) to see if the connection is closed. Immediately close the socket at the local node if the TCP socket at the remote node is closed.
- If the remote node's TCP socket closes without warning, the data to send may remain in the buffer at the local node. The remaining data is discarded in the local node's TCP close processing. The steps that are required in applications to avoid this include sending data from the sending node that permits closing and closing the socket only after checking the remote node.
- While open processing is performed for a TCP socket, a port that was closed first cannot be opened again for 60 seconds from the time the close processing is performed for the remote socket. However, this is not true if you specified 0 (automatic assignment by the Unit) as the port for the SktTCP-Connect instruction.
- You can use Connect from another socket to open a connection to a socket that was opened with Accept. A connection is not opened if you try to use Connect from another socket to open a connection to a socket that was opened with Connect. Also, a connection is not opened if you attempt to use Accept from another socket to open a socket that was opened with Accept. Furthermore, you cannot use Connect from more than one other node to establish multiple connections with a single TCP socket that was opened with Accept on the built-in EtherNet/IP port.
- You can use the keep-alive function for TCP sockets at the built-in EtherNet/IP port. The keep alive function checks whether a connection is normally established when no data is sent or received for a certain period on the communications line where the connection was established. The built-in Ether-Net/IP port responds to checks from other nodes even if keep alive is not specified.

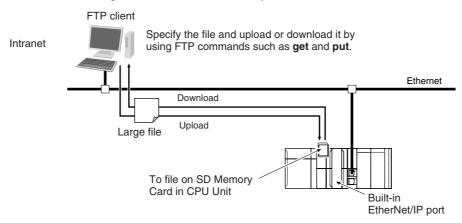


# **FTP Server**

10-1	Overvi	ew and Specifications1	0-2
	10-1-1	Overview 1	10-2
	10-1-2	Specifications	10-2
10-2	FTP Se	erver Function Details 1	0-3
	10-2-1	Supported Files 1	10-3
	10-2-2	Connecting to the FTP Server	10-3
10-3	Using t	the FTP Server Function	0-5
	10-3-1	Procedure 1	10-5
	10-3-2	List of Settings Required for the FTP Server Function 1	10-5
10-4	FTP Se	erver Application Example	0-6
10-5	Using I	FTP Commands	0-8
	10-5-1	Table of Commands	10-8
	10-5-2	Using the Commands	10-9
10-6	Using 9	SD Memory Card Operations	-15
	10-6-1	SD Memory Card Types	)-15
	10-6-2	File Types	)-16
	10-6-3	Initializing SD Memory Cards	)-16
	10-6-4	Format of Variable Data	)-17
10-7	Applica	ation Example from a Host Computer	-18

### 10-1-1 Overview

The built-in EtherNet/IP port contains an FTP (file transfer protocol) server. You can therefore send FTP commands from an FTP client software application on a computer on an Ethernet network to upload and download large files on the SD Memory Card.



## 10-1-2 Specifications

Item	Specification
Executable commands	open: Connects the specified host FTP server.
	user: Specifies the user name for the remote FTP server.
	ls: Displays file names in the remote host.
	mls: Displays file names in the remote host.
	dir: Displays file names and details in the remote host.
	mdir: Displays file names and details in the remote host.
	rename: Changes a file name.
	mkdir: Creates a new directory in the working directory in the remote host.
	rmdir: Deletes a directory from the working directory in the remote host.
	cd: Changes the work directory in the remote host to the specified directory.
	pwd: Displays the work directory on the remote host.
	type: Specifies the data type of transferred files.
	get: Transfers the specified remote file to the local host.
	mget: Transfers the specified multiple remote files to the local host.
	put: Transfers the specified local file to the remote host.
	mput: Transfers the specified multiple local files to the remote host.
	delete: Deletes the specified file from the remote host.
	mdelete: Deletes the specified multiple files from the remote host.
	append: Uses the currently specified file data type and appends a local file to the remote host.
	close: Disconnects the FTP server.
	bye: Closes the FTP client.
	quit: Closes the FTP client.
Protection	Login name (up to 12 characters)
	Password consists of 8 to 32 characters
Protocol used	FTP (port number: 20/TCP, 21/TCP)
Number of connections	6

## 10-2 FTP Server Function Details

### 10-2-1 Supported Files

The file system in the Controller that can be accessed by the built-in EtherNet/IP port includes files in any SD Memory Card mounted in the CPU Unit. The directory tree is shown below.

/: root

MEMCARD1: SD

usr
bin
dev

Used by system

A connection is initially made to the root directory.



#### **Additional Information**

- The date of the MEMCARD1 directory displayed for **Is**, **dir**, or **mkdir** commands in the root directory is the date of the file system volume label.
- The login date is displayed for MEMCARD1 if a volume label has not been created.

## 10-2-2 Connecting to the FTP Server

Input the FTP login name and password to login to the built-in EtherNet/IP port from an FTP client application. Use the Built-in EtherNet/IP Port Settings in the Sysmac Studio to set the FTP login name and password.



#### **Additional Information**

When a general-purpose FTP application is used, you can use a graphical user interface similar to Explorer to transfer and read files.

## Login Name and Password Setting

The FTP login name and password are not set by default. Use the Built-in EtherNet/IP Port Settings to set any login name and password.

#### Login Messages

Status	Message			
Normal connection	220 xxx.xx.xx FTP server ready. xxx.xx.xx: CPU Unit model (example: NJ501-1300)			
Connected to maximum number of connections (6)	530 FTP server busy, Goodbye.			

#### Setting Restrictions

The following restrictions apply to login names and passwords.

- Only single-byte alphanumeric characters can be used in the login name and password. The login name and password are case sensitive.
- A login name consists of up to 12 characters.
- A password consists of 8 to 32 characters.
- Always set a password when you set a new login name. The login name will not be valid unless a password is set for it.
- The login name is invalid if the login name is not set or characters other than single-byte alphanumeric characters are used.

#### FTP File Transfer Mode

FTP has two file transfer modes: ASCII mode and binary mode. Before you start to transfer files, use the **type** command (specifies the data type of transferred files) to select the required mode.

- To transfer a file in binary format: Select binary mode.
- To transfer a file in ASCII format: Select ASCII mode.

#### Multiple Accesses to the Same File

Files accessed with the FTP server may be simultaneously accessed from multiple sources with communications commands from other FTP servers or programming instructions. Exclusive control is required to prevent multiple accesses. This is to prevent reading and writing the same file at the same time. The CPU Unit automatically performs exclusive control as shown below only when the following combinations of instructions are used. In other cases, use file operation instructions (Change File Name, Copy File, etc.) or communications commands and perform exclusive control.

Exclusive Control When Accessing the Same File on the SD Memory Card

	First access	Instru	uction	Communicati	ons command
Later access		Read	Write	Read	Write
Instruction	Read	Exclusive control is automatically implemented and the read instruction results in an error (first-comefirst-serve).		(Exclusive control is not required.)	Implement exclusive control.
	Write	Exclusive control implemented and tion results in an efirst serve)	the write instruc-	Implement exclusive control.	
Communica- tions command	Read	(Exclusive control is not required.)	Implement exclusive control.	(Exclusive control is not required.)	
Write Implement exclusive control.			•		

# 10-3 Using the FTP Server Function

#### 10-3-1 Procedure

- Make the basic settings.Refer to 1-5-1 Outline for basic operations.
- **2** Set up the FTP server on the Sysmac Studio. (Refer to *5-3 FTP Display*.)
- 3 Select Built-in EtherNet/IP Port Settings under Configuration Controller Setup on the Sysmac Studio. Make the following settings on the FTP Settings Display.
  - FTP server
  - · Port number
  - Login name
  - Password
- 4 Place the CPU Unit online and transfer the settings to the Controller.
- **5** Insert the SD Memory Card into the CPU Unit.
- 6 Connect to the built-in EtherNet/IP port from an FTP client.
- Input the FTP login name and password that you set in the Built-in EtherNet/IP Port Settings to log in to the built-in EtherNet/IP port.

Note Once logged in, the ftp commands can be used, such as cd (Change Directory), and get (Obtain File).

- **8** After you are logged in, you can use the ftp commands, such as cd (Change Directory) and get (Obtain File) for the MEMCARD1 directory in the SD Memory Card in the Controller.
- **9** Close the connection.

## 10-3-2 List of Settings Required for the FTP Server Function

Make the following settings for the unit setup when the FTP server function is used.

Built-in EtherNet/IP Port Settings Tab Page on Sysmac Studio	Setting	Setting conditions	Reference
FTP	FTP server	Required	page 5-7
	Port No.	Optional*	
		Note Required when changing the default value of 21.	
	Login name	Required*	
	Password	Required*	

<sup>\*</sup> Settings are not required if the FTP server is not used.



#### **Additional Information**

Make the settings in the FTP Settings Dialog Box if the FTP server is used. Refer to 5-3 FTP Display for information on the FTP Settings Dialog Box.

An example application of the FTP server when the login name is "user1" and the password is "password" is shown below.



#### **Additional Information**

When a general-purpose FTP application is used, you can use a graphical user interface similar to Explorer to transfer and read files.

#### Step

- Make sure that an SD Memory Card is inserted in the CPU Unit and turn ON the power supply to the Controller.
- Connect to the FTP server from a computer on the Ethernet by entering the text that is underlined in the following diagram.

IP address of built-in EtherNet/IP port C:\>ftp 192.168.250.1 Connected to 192.168.250.1. Results 220 NJ501-1500 FTP server ready. User (192.168.250.1: (none)): user1 < Login name 331 Password required for user1. Password: Password 230 User user1 logged in. (hidden) ftp> ftp> bye 221-Data traffic for this session was 0 bytes in 0 files. Total traffic for this session was 204 bytes in 0 transfers. 221 Thank you for using the FTP service on 192.168.250.1. C:\>

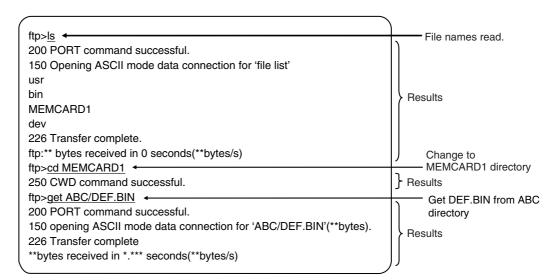
**3** Enter FTP commands (underlined in the following diagram) to read and write files. The following directory tree is used in this example.

```
/ (root directory)

MEMCARD1

ABC (subdirectory)

DEF.BIN(file)
```



# **10-5 Using FTP Commands**

This section describes the FTP commands which the host computer (FTP client) can send to the FTP server of the built-in EtherNet/IP port. The descriptions should also apply to most workstations, but slight differences may arise. Refer to your workstation's operation manuals for details.

#### **Table of Commands** 10-5-1

The FTP commands which can be sent to the built-in EtherNet/ IP port are listed in the following table.

Command	Description			
open	Connects the specified host FTP server.			
user	Specifies the user name for the remote FTP server.			
Is	Displays file names in the remote host.			
mls	Displays file names in the remote host.			
dir	Displays file names and details in the remote host.			
mdir	Displays file names and details in the remote host.			
rename	Changes a file name.			
mkdir	Creates a new directory in the working directory in the remote host.			
rmdir	Deletes a directory from the working directory in the remote host.			
cd	Changes the work directory in the remote host to the specified directory.			
pwd	Displays the work directory on the remote host.			
type	Specifies the data type of transferred files.			
get	Transfers the specified remote file to the local host.			
mget	Transfers the specified multiple remote files to the local host.			
put	Transfers the specified local file to the remote host.			
mput	Transfers the specified multiple local files to the remote host.			
delete	Deletes the specified file from the remote host.			
mdelete	Deletes the specified multiple files from the remote host.			
append	Uses the file data type that is specified by the <b>type</b> command to append the local file to the remote host.			
close	Disconnects the FTP server.			
bye	Closes the FTP client.			
quit	Closes the FTP client.			

- Note 1 "Remote host" refers to the built-in EtherNet/IP port.
  - 2 A "remote file" is a file on the SD Memory Card in the CPU Unit.
  - 3 "Local host" refers to the host computer (FTP client).
  - 4 "Local file" refers to a file on the host computer (FTP client).

## 10-5-2 Using the Commands

## open

#### Format

open [IP address or host name of FTP server]

#### Function

Connects the FTP server. Normally when the FTP client is booted, the FTP server IP address is specified to execute this command automatically.

#### user

#### Format

user [user name]

#### Function

- Specifies the user name. Specify the FTP login name set in the built-in EtherNet/IP port system setup.
- The user name is automatically requested immediately after connection to the FTP server.

#### Is

#### Format

Is [-I] [remote\_file\_name [local\_file\_name]]

#### Function

- Displays the names of files on the remote host (on the SD Memory Card).
- Set the switch [-I] to display not only the file names but the creation dates and sizes as well. If the switch is not set, only the file names are displayed.
- Specify a file on the SD Memory Card for the remote file name.
- If a local file name is specified, the file information is stored in the specified file.

#### mls

#### Format

mls remote\_file\_name local\_file\_name

- Displays a list of the names of files on the remote host (on the SD Memory Card).
- Specify the directory or file name on the SD Memory Card you wish to list for remote file name. Input an asterisk (\*) to display a list of the current working directory.
- If a local file name is specified, the file information is stored in the specified file. Input a hyphen (-) to display a list of the remote host but not store the list of file names.

### dir

#### Format

dir [remote\_file\_name [local\_file\_name]]

#### Function

- · Displays the names, creation dates, and sizes of files on the remote host (on the SD Memory
- It displays the same information as command [Is -I].
- Specify a file on the SD Memory Card for the remote file name.
- If a local file name is specified, the file information is stored in the specified file.

#### mdir

#### Format

mdir remote\_file\_name local\_file\_name

#### Function

- · Displays the names of files, subdirectories, creation dates, and sizes on the remote host (on the SD Memory Card).
- Specify the directory or file name on the SD Memory Card to list as the remote file name. Input a hyphen (-) to display a list of the current working directory.
- If a local file name is specified, the file information is stored in the specified file. Input a hyphen (-) to display a list of the remote host and not store the list of file names.

#### Format

mdir RemoteFiles [...] LocalFile

Remotefiles: Enter a hyphen (-) to use the current working directory.

Localfiles: Enter a hyphen (-) to display a list.

#### Function Details

• A list of the files and subdirectories in the remote directory is displayed.

#### rename

#### Format

rename current\_file\_name new\_file\_name

- Changes the specified current file name to the specified new file name.
- If the new file name is already used on the remote host (on the SD Memory Card), the existing file is overwritten by the file for which the name was changed.
- rename can be used only to change the file name. It cannot be used to move the file to a different directory.

### mkdir

#### Format

mkdir directory\_name

#### Function

- Creates a directory of the specified name at the remote host (on the SD Memory Card).
- An error will occur if a file or directory of the same name already exists in the working directory.

### rmdir

#### Format

rmdir directory\_name

#### Function

- Deletes the directory with the specified name from the remote host (from the SD Memory Card).
- The directory must be empty to delete it.
- · An error will occur if the specified directory does not exist or is not empty.

## pwd

#### Format

pwd

#### Function

• Displays the work directory on the remote host.

## append

#### Format

append local\_file\_name [remote\_file\_name]

#### Function

Uses the file data type that is specified by the **type** command to append the local file to the remote host (on the SD Memory Card).

#### cd

#### Format

cd [directory\_name]

- Changes the remote host work directory to the specified remote directory.
- Files on the SD Memory Card are stored in the MEMCARD1 directory under the root directory (/).

• The root directory (/) is the directory that is used when you log onto the built-in EtherNet/IP port. The MEMCARD1 directory does not exist if an SD Memory Card is not inserted in the CPU Unit or if the SD Memory Card power indicator on the CPU Unit is not lit.

## tvpe

#### Format

type data\_type

#### Function

- Specifies the file data type.
- The following data types are supported: ascii: Files are transferred as ASCII data. binary (image): Files are transferred as binary data. The CPU Unit handles binary files. Use the type command to specify binary transfers before you upload or download files. File contents cannot be guaranteed if transferred as ASCII data.
- The default file type is ASCII.

## get

#### Format

get file\_name [receive\_file\_name]

#### Function

- Transfers the specified remote file from the SD Memory Card to the local host.
- A receive file name can be used to specify the name of the file in the local host.

## mget

#### Format

mget file\_name

#### Function

 You can include wildcards (\*) in the file name to transfer multiple remote files from the SD Memory Card to the local host.

## put

#### Format

put file\_name [destination\_file\_name]

- Transfers the specified local file to the remote host (to the SD Memory Card).
- You can specify the destination file name to specify the name the file is stored under on the SD Memory Card.
- Any existing file with the same name in the remote host (on the SD Memory Card) is overwritten by the contents of the transferred file.

## mput

#### Format

mput file\_name

#### Function

- You can include wildcards (\*) in the file name to transfer multiple local files to the remote host (to the SD Memory Card).
- Any existing file with the same name in the remote host (on the SD Memory Card) is overwritten by the contents of the transferred file.

## delete

#### Format

delete file\_name

#### Function

• Deletes the specified remote file (on the SD Memory Card).

### mdelete

#### Format

mdelete file\_name

#### Function

• You can include wildcards (\*) in the file name to delete multiple remote files from the SD Memory Card.

#### close

#### Format

close

#### Function

• Disconnects the FTP server of the built-in EtherNet/IP port.

## bye

#### Format

bye

#### Function

• Ends the FTP sessions.

# quit

Format

quit

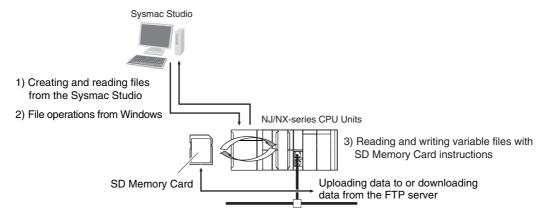
- Function
  - Ends the FTP sessions.

# 10-6 Using SD Memory Card Operations

The built-in EtherNet/IP port can be used to upload and download the following data between the SD Memory Card and the FTP server.

· Variables files (binary format)

The following three methods are used by the CPU Unit to store and read data on the SD Memory Card.



## 10-6-1 SD Memory Card Types

Use the SD or SDHC Memory Card that is specified below. Operation may not be dependable if another SD or SDHC Memory Card is used.

Model number	Card type	Capacity [GB]	Format	Number of overwrites	Weight
HMC-SD291	SD	2	FAT16	100,000 writes	2 g max.
HMC-SD491	SDHC	4	FAT32		

## **File Names**

Files are distinguished by assigning file names and extensions. The following characters can be used in file names and extensions: File names are not case sensitive.

A to Z, a to z, 0 to 9, and the following symbols: % ' - \_ @ ! '() ~ = # & + ^ [] {}, .;

The following characters cannot be used in files names and extensions:

Blanks, multi-bytes characters, and the following symbols: /\? \* ": < > etc.

The maximum file name length with the extension is 65 characters. The first period (.) in a file name is taken as the delimiter between the file name and extension. Extensions are determined by the file type.

## **Directory**

You can create up to five levels of directories to store files on the SD Memory Card (count the root directory as one level). A maximum of 65 characters can be used in a directory name.

## File Names Handled by CPU Unit

The files described in the following table can be read or written by the CPU Unit.

File type	File names	Extension	Contents	Description
Variables file (binary format)	Refer to 10-6-2 File Types.	.bin	Specified variables	This variables file contains the values of specified variables (which include arrays and structures) in binary format (.bin).

Refer to the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501) for details.

## 10-6-3 Initializing SD Memory Cards

1 Insert the SD Memory Card into the CPU Unit.

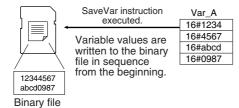
2 Use the Sysmac Studio to initialize the SD Memory Card.

### 10-6-4 Format of Variable Data

## **Binary Format**

The IOM format is a data format used for binary data specified by the ladder instructions, FileReadVar (Read Variables File) and FileWriteVar (Save Variables File), in the CPU Unit. You can also read and save arrays and structures. Data is created as shown below when the data of variable  $Var_A$  is placed in an attached file in binary format.

SD Memory Card





#### **Additional Information**

- When you handle a binary file on the NJ/NX-series CPU Unit, always specify the binary data type with the type command before you read or write the file via FTP. (Refer to 10-5-2 Using the Commands.)
- For details on how to use ladder diagram instructions to process files, refer to the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

The following procedure provides an example of FTP operations from a host computer. In this example, the following assumptions are made.

- The IP address of the built-in EtherNet/IP port is registered in the hosts as host name [ni].
- The FTP login name is "LogIn".
- Manufacturing results is stored in the SD Memory Card in the CPU Unit in a file that is named RESULT.BIN.
- A processing instructions data file called PLAN.BIN already exists on the workstation.

In the following procedure, the manufacturing results file (RESULT.BIN) in the SD Memory Card in the CPU Unit is transferred to a workstation, and then a manufacturing instructions file (PLAN.BIN) on the workstation is transferred to the SD Memory Card in the CPU Unit. Underlined text is keyed in from the FTP client. The workstation prompt is indicated as \$ and the cursor is indicated as ■.

**1** Start the FTP application and connect to the built-in EtherNet/IP port.

```
$ ftp nj ← connected to nj
220 **IPaddress** NJ501-1300 FTP server(FTP**version**)ready
Name(nj:root): ■
```

**2** Enter the login name.

```
Name(nj:root):LogIn ←
331 Password required for LogIn.
Password:
230 LogIn logged in.
ftp> ■

Enter the login name.

Enter the password.
```

Make sure the Memory Card is correctly inserted. The MEMCARD1 directory is displayed if there is an SD Memory Card in the CPU Unit.

```
ftp> Is 

200 PORT command successful.

150 opening data connection for Is(**IPaddress**port#**)(0 bytes).

MEMCARD1

226 Transfer complete.

15 bytes received in 0 seconds(**bytes/s)

ftp> ■
```

**4** Change to the MEMCARD1 directory.

```
ftp> cd MEMCARD1 ← Change the directory.
250 CWD command successful.
ftp> ■
```

**5** Change data type to binary.

```
ftp> type binary ∠200 Type set to I.

ftp> ■
```

**6** Read the file RESULT.BIN and transfer it to the workstation.

```
ftp> get RESULT.BIN 
200 PORT command successful.
150 opening data connection for result.bin (**IPaddress**port#**) (**bytes).
226 Transfer complete.
** bytes received in *.*** seconds (**bytes/s)
ftp> ■
```

7 Write the file PLAN.BIN to the Memory Card.

```
ftp> put PLAN.BIN 
200 PORT command successful.
150 opening data connection for plan.bin (**IPaddress**port#**) .
226 Transfer complete.
** bytes received in *.** seconds (**bytes/s)
ftp>
```

**8** End the FTP session.

```
ftp> bye ← 221 Goodbye. $ ■
```



# **FTP Client**

11-1	Using t	he FTP Client to Transfer Files 1	11-2
	11-1-1	Transferring Files	11-2
	11-1-2	Connectable FTP Servers	11-3
	11-1-3	File Transfer Options	11-3
	11-1-4	Other Functions	11-4
11-2	FTP Cli	ent Communications Instructions 1	11-5
	11-2-1	Functions of the FTP Client Communications Instructions	11-5
	11-2-2	Restrictions on the FTP Client Communications Instructions	11-7
11-3	FTP Cli	ent Application Example	11-8

## **Using the FTP Client to Transfer Files**

You can use the FTP client to transfer files between the FTP client and an FTP server. You can transfer files in either direction: download data from the FTP server to the FTP client or upload data from the FTP client to the FTP server.



#### **Version Information**

A CPU Unit with unit version 1.08 or later is required to use the FTP client.

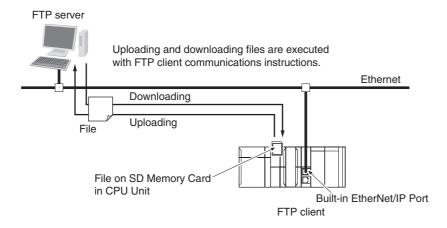
#### 11-1-1 **Transferring Files**

All file transfers that use the FTP client are executed with FTP client communications instructions in the user program. The file transfer settings are all made with the parameters of the FTP client communications instructions. No settings are required from the Sysmac Studio.

The FTP client communications instructions and their functions are given in the following table. You can execute up to three FTP client communications instructions at the same time.

Instruction	Function
FTPGetFileList	Gets a file list from the FTP server.
FTPGetFile	Downloads one or more files from the FTP server.
FTPPutFile	Uploads one or more files to the FTP server.
FTPRemoveFile	Deletes one or more files on the FTP server.
FTPRemoveDir	Deletes a directory on the FTP server.

Downloaded files are stored on the SD Memory Card. When uploading files, files that are stored on the SD Memory Card are uploaded to the FTP server. Therefore, to upload or download files, an SD Memory Card must be inserted in the NJ/NX-series CPU Unit.



#### 11-1-2 Connectable FTP Servers

An NJ/NX-series CPU Unit can connect to the following FTP servers. Refer to the relative manuals for information on setting and using the FTP servers.

- FTP server in a built-in EtherNet/IP port on NJ/NX-series CPU Unit
- CJ-series EtherNet/IP Unit with unit version 2.0 or later
- CJ-series CJ2 CPU Unit with Built-in EtherNet/IP
- CJ-series CJ1M CPU Unit with Ethernet Functions
- · CJ-series Ethernet Unit
- Windows7: Windows Server 2008 R2 (Internet Information Services (IIS) 7.5)
- Windows8: Windows Server 2012 (IIS8.0)
- Linux

### 11-1-3 File Transfer Options

You can use the following options for file transfers. All options are specified in the parameters of the FTP client communications instructions.

- · File transfer mode
- Open mode for data connection
- · Deleting files after transfer
- Overwriting

The following sections describe each of these options.

#### File Transfer Mode

There are two file transfer modes, ASCII Mode and Binary Mode, which differ in how line feeds in the text data are processed. The following table describes the differences.

Transfer mode	Handling of line feeds in text data
ASCII Mode	Line feeds are converted to the line feed code of the destination system, e.g., Unix or Windows.
Binary Mode	Line feeds are transferred without conversion.

#### Open Mode for Data Connection

Opening a TCP connection between the FTP server and FTP client is necessary to transfer files. TCP connections include control connections to control communications and data connections to transfer data. There are also an Active Mode and a Passive Mode to open a data connection that differ in which party makes the connection request. The following table describes the differences.

Open mode	Request to establish a connection
Active Mode	The FTP server makes the connection request.
Passive Mode	The FTP client makes the connection request.

For example, if the FTP server is not on the Internet and you use Active Mode to open the data connection, the connection request from the FTP server may not be permitted due to security policies. In that case, you must use Passive Mode to open the data connection when you request a connection from the FTP client.

#### File Deletion after Transfer

You can specify whether to delete the source files after the file transfer. If the file transfer fails for any reason, the source files are not deleted even if deletion is specified.

#### Overwriting

You can specify whether to overwrite a file of the same name at the file transfer destination. If you specify not overwriting files and a file of the same name as the transferred file exists at the transfer destination, the source file is not transferred.

#### 11-1-4 Other Functions

You can also use the following two functions for file transfers.

- · Retrying connection processing with the FTP server
- · Using wildcards to specify the files to transfer

These functions are described in the following sections.

#### Retrying Connection Processing with the FTP Server

You can specify automatically retrying connection processing up to three times when connection processing to the FTP server fails. You can set the timeout time that is used to determine connection failure, the number of retries, and the retry interval.

#### Using Wildcards to Specify the Files to Transfer

You can use wildcards to specify the names of the files to transfer. This allows you to transfer more than one file at one time.

# 11-2 FTP Client Communications Instructions

All file transfers that use the FTP client are executed with FTP client communications instructions. The FTP client communications instructions and their functions are given in the following table.

Instruction	Function
FTPGetFileList	Gets a file list from the FTP server.
FTPGetFile	Downloads one or more files from the FTP server.
FTPPutFile	Uploads one or more files to the FTP server.
FTPRemoveFile	Deletes one or more files on the FTP server.
FTPRemoveDir	Deletes a directory on the FTP server.

For details on the FTP client communications instructions, refer to the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

#### 11-2-1 Functions of the FTP Client Communications Instructions

This section describes the functions of the FTP client communications instructions.

### FTPGetFileList Instruction

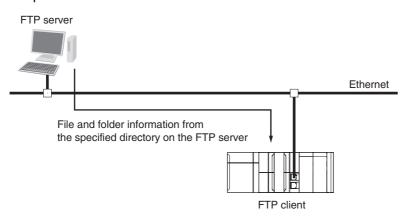
The FTPGetFileList instruction gets a list of the files and folders in a specified directory on the FTP server.

The following information is obtained.

- The number of files and folders in the specified directory
- · The names of the files and folders
- The last updated dates of the files and folders
- The file sizes
- · The read-only attributes of the files and folders

You can specify the following option.

· Open mode for data connection



## **FTPGetFile Instruction**

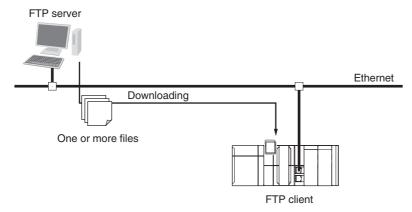
The FTPGetFile instruction downloads the specified file from the specified directory on the FTP server to the specified directory in the SD Memory Card.

You can use wildcards to specify the file name to allow you to download more than one file at the same time.

If the directory specified for the download does not exist in the SD Memory Card, the directory is created and the data is downloaded in it.

You can specify the following options.

- Transfer mode
- · Open mode for data connection
- · Deleting files after transfer
- Overwriting



## **FTPPutFile Instruction**

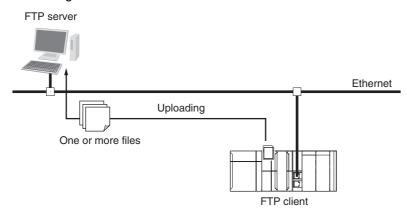
The FTPPutFile instruction uploads the specified file from the specified directory in the SD Memory Card to the specified directory on the FTP server.

You can use wildcards to specify the file name to allow you to upload more than one file at the same time.

If the directory specified for the upload does not exist on the FTP server, the directory is created and the data is uploaded in it.

You can specify the following options.

- · Transfer mode
- · Open mode for data connection
- · Deleting files after transfer
- Overwriting



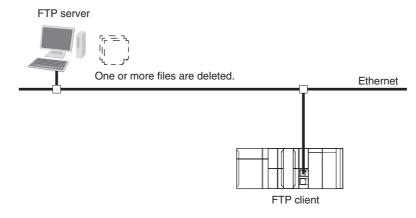
#### FTPRemoveFile Instruction

The FTPRemoveFile instruction deletes the specified file in the specified directory on the FTP server.

You can use wildcards to specify the file name to allow you to delete more than one file at the same time.

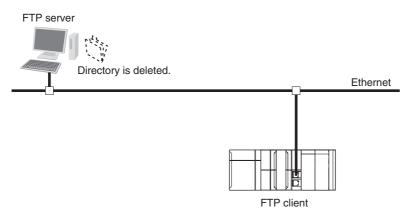
You can specify the following option.

· Open mode for data connection



#### **FTPRemoveDir Instruction**

The FTPRemoveDir instruction deletes the specified directory from the FTP server.



#### 11-2-2 Restrictions on the FTP Client Communications Instructions

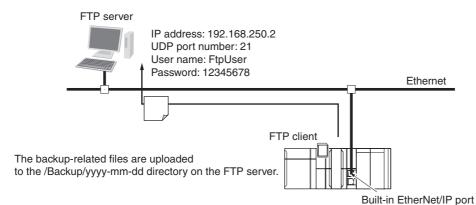
The following restrictions apply to the FTP client communications instructions. Be careful not to exceed these restrictions when you create the user program.

- If you execute more than one FTP client communications instruction to read and write data in an SD
  Memory Card at a time, unexpected operation may result, such as reading data from a file to which
  data is being written. Perform exclusive control of the instructions in the user program.
- If you execute an FTP client communications instruction to read or write data in an SD Memory Card
  at the same time as an operation to read or write data in the SD Memory Card, unexpected operation
  may result, such as reading data from a file to which data is being written. Perform exclusive control
  of the instructions in the user program. Refer to the NJ/NX-series CPU Unit Software User's Manual
  (Cat. No. W501) for information on exclusive control of file access in SD Memory Cards.

## 11-3 FTP Client Application Example

All FTP client functionality is executed with FTP client communications instructions. This section provides sample programming that uses the FTP client communications instructions.

This programming executes an SD Memory Card backup and then uploads all of the backup-related files to the /Backup/yyyy-mm-dd directory on the FTP server.



The Controller is connected to the FTP server through an EtherNet/IP network. The settings of the parameters to connect to the FTP server are given in the following table.

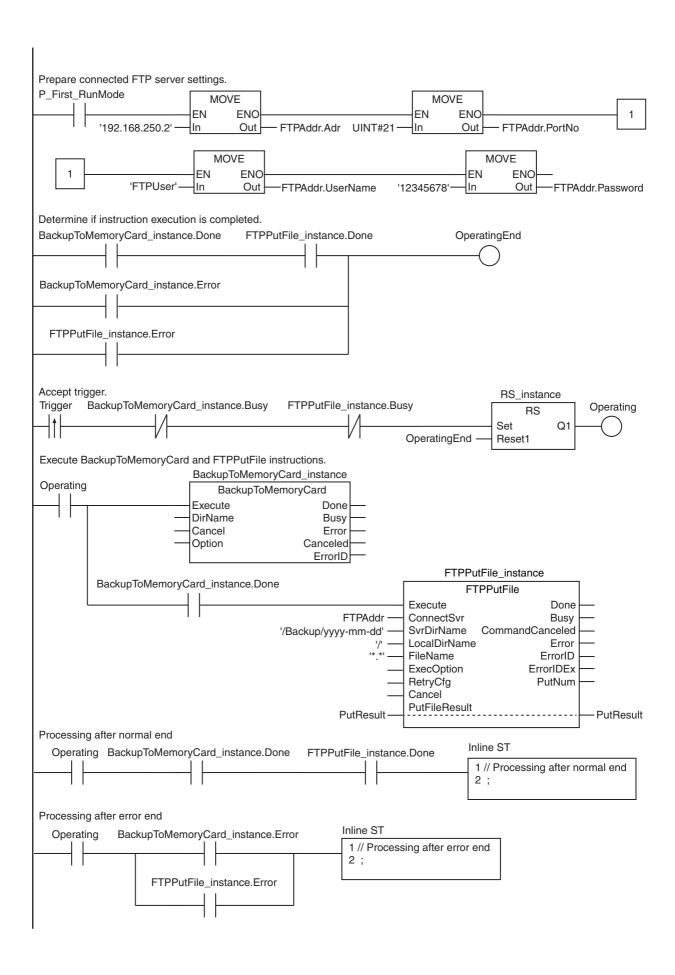
Parameter	Value
IP address	192.168.250.2
UDP port number	21
User name	FtpUser
Password	12345678

The following procedure is used.

- The BackupToMemoryCard instruction is used to save NJ/NX-series Controller backup-related files to the root directory on the SD Memory Card.
- The FTPPutFile instruction is used to upload the backup-related files to the /Backup/yyyy-mmdd directory on the FTP server. The wildcard specification \*.\* is used to specify the names of the files to transfer.
- Normal end processing is executed if all processing ends normally. Processing for an error end is performed if an error occurs.

#### LD

Internal Vari- ables	Variable	Data type	Initial value	Comment
	FTPPutFile_insta nce	FTPPutFile		Instance of FTPPutFile instruction
	FTPAddr	_sFTP_CONNECT _SVR	(Adr := ", PortNo := 0, User- Name := ", Password := ")	Connected FTP server settings
	PutResult	ARRAY[00] OF _sFTP_FILE_RES ULT	[(Name := ", TxError := False, RemoveError := False, Reserved := [4(16#0)])]	Uploaded file results
	RS_instance	RS		Instance of RS instruction
	OperatingEnd	BOOL	FALSE	Processing completed
	Trigger	BOOL	FALSE	Execution condition
	Operating	BOOL	FALSE	Processing
	BackupToMemor yCard_instance	BackupToMemory- Card		Instance of Backup- ToMemoryCard instruction



ST

Internal Vari- ables	Variable	Data type	Initial value	Comment
	R_TRIG_instance	R_TRIG		Instance of R_TRIG instruction
	UP_Q	BOOL	FALSE	Trigger output
	FTPPutFile_insta	FTPPutFile		Instance of FTPPutFile instruction
	DoFTPTrigger	BOOL	FALSE	Execution condition for BackupToMemoryCard and FTPPutFile
	FTPAddr	_sFTP_CONNECT _SVR	(Adr := ", PortNo := 0, User- Name := ", Password := ")	Connected FTP server settings
	PutResult	ARRAY[00] OF _sFTP_FILE_RES ULT	[(Name := ", TxError := False, RemoveError := False, Reserved := [4(16#0)])]	Uploaded file results
	Stage	UINT	0	Instruction execution stage
	Trigger	BOOL	FALSE	Execution condition
	BackupToMemory Card_instance	BackupToMemory- Card		Instance of Backup- ToMemoryCard instruction

```
// Prepare connected FTP server settings.
IF P_First_RunMode THEN
   TPAddr.PortNo := '192.168.250.2';// Address
FTPAddr.PortNo := UINT#21; // Port number
FTPAddr.UserName := 'FtpUser'; // User name
   FTPAddr.Password := '12345678';
                                        // Password
END_IF;
// Accept trigger.
R TRIG instance(Trigger, UP Q);
IF ( (UP Q = TRUE) AND (BackupToMemoryCard instance.Busy = FALSE) AND
     (FTPPutFile_instance.Busy = FALSE) ) THEN
   DoFTPTrigger := TRUE;
   Stage := INT#1;
   BackupToMemoryCard instance( // Initialize instance.
      Execute := FALSE) ;
   FTPPutFile instance( // Initialize instance.
      Execute := FALSE,
      ConnectSvr := FTPAddr,
      SvrDirName := '/Backup/yyyy-mm-dd',
      LocalDirName := '/',
      FileName := '*.*',
      PutFileResult := PutResult) ;
END_IF;
IF (DoFTPTrigger = TRUE) THEN
   CASE Stage OF
      1: // Execute BackupToMemoryCard instruction.
         BackupToMemoryCard instance(
            Execute := TRUE, // Execution
         IF (BackupToMemoryCard instance.Done = TRUE) THEN
            Stage := INT#2; // To next stage
         ELSIF (BackupToMemoryCard_instance.Error = TRUE) THEN
                   := INT#10; // Error end
            Stage
         END IF;
      2: // Execute FTPPutFile instruction.
```

```
FTPPutFile_instance(
           Execute := TRUE,
ConnectSvr := FTPAddr,
                                                  // Execution
                                                 // Connected FTP server
           SvrDirName := '/Backup/yyyy-mm-dd',// FTP server directory name
           LocalDirName := '/',
                                                 // Local directory name
           FileName := '*.*',
                                                  // File name
           PutFileResult := PutResult) ;
                                                 // Uploaded file results
         IF (FTPPutFile instance.Done = TRUE) THEN
           Stage := INT#0; // Normal end
         ELSIF (FTPPutFile instance.Error = TRUE) THEN
           Stage := INT#20; // Error end
        END IF;
      0: \ \ // \ {\tt Processing} \ {\tt after} \ {\tt normal} \ {\tt end}
        DoFTPTrigger:=FALSE;
        Trigger :=FALSE;
     ELSE // Processing after error end
        DofTPTrigger:=FALSE;
        Trigger :=FALSE;
  END CASE;
END_IF;
```



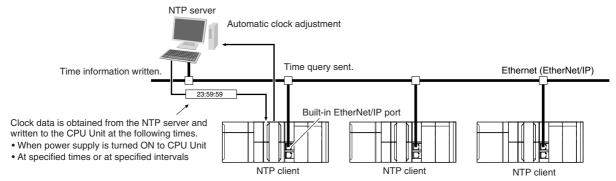
# **Automatic Clock Adjustment**

12-1	Automa	atic Clock Adjustment	12-2
	12-1-1	Overview	12-2
	12-1-2	Specifications	12-3
12-2	Proced	ure to Use the Automatic Clock Adjustment Function	12-4
	12-2-1	Procedure	12-4
	12-2-2	Settings Required for Automatic Clock Adjustment	12-4

# 12-1 Automatic Clock Adjustment

#### 12-1-1 Overview

With the built-in EtherNet/IP port, clock information is read from the NTP\* server at the specified time or at a specified interval after the power supply to the CPU Unit is turned ON. The internal clock time in the CPU Unit is updated with the read time.

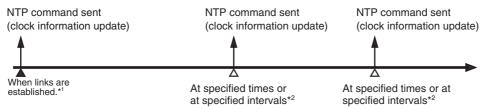


<sup>\*</sup> The NTP (Network Time Protocol) server is used to control the time on the LAN.

### 12-1-2 Specifications

Item	Specification		
Protocol	NTP		
Port No.	123 (UDP)		
	However, you can change the port nun the Sysmac Studio.	nber in the Built-in EtherNet/IP Port Settings on	
Access to NTP server	Writes the clock information from the NTP server to the local CPU Unit.	Obtains the clock information from the NTP server set up on the Network, and applies the information obtained to the local CPU Unit.	
NTP Operation Timing	Clock information is automatically updated at the following times if the NTP functi used.		
	<ul> <li>After links are established when the power supply to the CPU Unit is turned ON</li> <li>At specified times or at specified intervals (according to the option selected for the NTP Operation Timing)</li> </ul>		

Clock information is updated at the following times.



- \*1 The clock information is updated if acquiring the NTP server clock information is enabled on the NTP Tab Page.
- \*2 According to the option that is selected for the NTP operation timing on the NTP Tab Page.



#### **Additional Information**

- NTP clock synchronization is normally performed as follows:
  - If the clock deviation is within 128 ms: The clock is synchronized every 0.5 ms.
  - If the clock deviation exceeds 128 ms: The clock is synchronized immediately.
- If the NTP operation timing is set for a specified time interval, the timing will not change even if the time in the CPU Unit is changed during operation. The time information is updated automatically. (For example, if the time interval is set to 60 minutes, the information is updated 60 minutes after the last time it was updated even if the time in the CPU Unit is changed.)

# 12-2 Procedure to Use the Automatic **Clock Adjustment Function**

#### 12-2-1 Procedure

- Make the basic settings. Refer to 1-5 EtherNet/IP Communications Procedures for the flow of basic operations.
- 2 Select Built-in EtherNet/IP Port Settings from the Controller Setup on the Sysmac Studio, and then set the following on the NTP Settings Display.
  - NTP server settings (required)
  - NTP operation timing
- Select Synchronization from the Controller Menu. The built-in EtherNet/IP port settings are transferred to the CPU Unit.

## 12-2-2 Settings Required for Automatic Clock Adjustment

The following Built-in EtherNet/IP Port Settings are made from the Sysmac Studio to use automatic clock adjustment.

Tab page		Setting	Setting conditions	Reference
NTP	NTP server clock information		Required.	page 5-8
	Port No.		Specified by user.	
			Note Required to change from the default value of 123.	
	Server s	pecifying method	Required*1	
		IP address	One of these must be set, depending on	
		Host name	the Server specification type setting.	
	NTP operation timing		Required*1	
		Specify a Time	One of these must be set. (Set according	
		Specify a time interval	to the option selected for the NTP Operation Timing.)	
		Timeout time	Specified by user.	
			Note Required to change from the default value of 10 seconds.	



#### **Additional Information**

Make the settings in the NTP Settings Dialog Box if automatic clock adjustment is used. Refer to 5-4 NTP Settings Display for information on the NTP Settings Dialog Box.

<sup>\*1</sup> This setting is required to get the clock information from the NTP server.

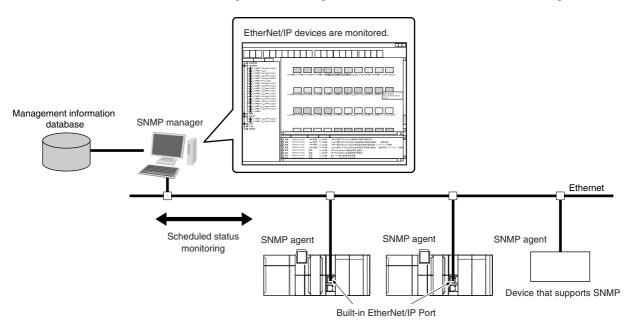
<sup>\*2</sup> The following ports are used by the system and cannot be set by the user: 25, 53, 68, 110, 2222, 2223, 2224, 9600, and 44818.

# **SNMP Agent**

13-1 SN	IMP A	Agent	13-2
13-	1-1	Overview	13-2
13-	1-2	Specifications	13-3
13-	1-3	SNMP Messages	13-3
13-	1-4	MIB Specifications	13-4
13-2 Pro	ocedu	ure to Use the SNMP Agent1	3-21
13-	2-1	Procedures	13-21
13-	2-2	Settings Required for the SNMP Agent	13-21

# 13-1 SNMP Agent

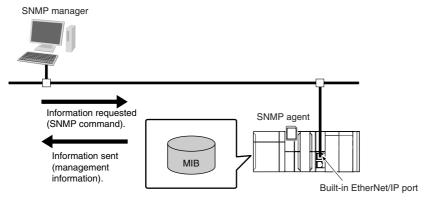
The SNMP (simple network management protocol) is a network management protocol. You can use the SNMP to manage any network that consists of devices that support SNMP. The server that manages the network is called the SNMP manager. The managed network devices are called SNMP agents.



#### 13-1-1 Overview

## **SNMP Agent**

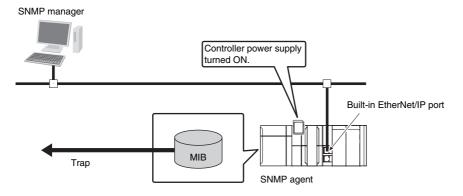
The built-in EtherNet/IP port has its own management information called the MIB (management information base). This information can be provided to the SNMP manager. The SNMP manager is software that gathers and processes information about devices on the SNMP network and provides that information to the network administrator. You can use the SNMP manager to monitor the built-in EtherNet/IP port.



The SNMP manager has a SNMP command to request MIB information. The built-in EtherNet/IP port SNMP agent function supports SNMPv1 (RFC1157) and SNMPv2C (RFC1901). Use the SNMPv1 or SNMPv2C protocol to manage the built-in EtherNet/IP port with the SNMP manager. You can also use both the SNMPv1 and SNMPv2C protocols together at the same time.

## **SNMP Traps**

When a failure or some other specific problem occurs, a status report called a trap is sent. This enables monitoring changes in status even if the SNMP manager does not monitor the built-in EtherNet/IP port periodically. However, traps use UDP. Therefore, you cannot check to see if the SNMP manager receives traps from the EtherNet/IP port. Thus, depending on the network status, some traps may not reach the SNMP manager.

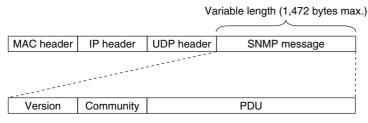


## 13-1-2 Specifications

Item	Specification
Protocol	SNMP
Agent	SNMPv1, SNMPv2c
MIB	MIB-II
Port No.	SNMP agent: 161 (UDP)
	SNMP trap: 162 (UDP)
	These can be changed in the Built-in EtherNet/IP Port Settings from the Sysmac Studio.
Timing of SNMP trap opera-	Status reports are sent to the SNMP manager at the following times.
tion	When the Controller is turned ON
	When links are established
	When an SNMP agent fails to be authorized
Supported MIB commands	GetRequest/GetNextRequest

## 13-1-3 SNMP Messages

The structure of SNMP messages is as follows:



Item	Set value
Version	This value gives the SNMP version.
	SNMPv1: 0
	SNMpv2c: 1
Community	Community name for verification
PDU	This depends on the PDU type.

## 13-1-4 MIB Specifications

This section describes the specifications of the MIB that is supported by the built-in EtherNet/IP port.

## **MIB System Diagram**

The built-in EtherNet/IP port MIB consists of the following tree structure.

```
L iso (1)
      L org (3)
              L <sub>dod (6)</sub>
                    Linternet (1)
                           L mgmt (2)
                                   L mib-2 (1)
                                          -system (1)
                                                               Unit information
                                          interface (2)
                                                               Interface information
                                               ip(4)
                                                               IP information
                                                               ICMP information
                                              icmp(5)
                                              tcp(6)
                                                               TCP information
                                              udp(7)
                                                               UDP information
                                              snmp(11)
                                                               SNMP information
```

## **MIB Groups**

	MIB grou	Stored information	
Standard MIB	system group		The MIB for information related to the device.
	interfaces group		The MIB for information related to the inter-
			face.
	<i>ip</i> group	ip	The MIB for IP information.
		ipAddrTable	The MIB for addressing table information
			related to IP addresses.
		ipRouteTable	The MIB for information related to IP routing
			tables.
		ipNetToMediaTable	The MIB for information related to IP address
			conversion tables.
		ipForward	The MIB for information related to IP forward-
			ing tables.
	icmp group		The MIB for ICMP information.
	tcp group	tcp	The MIB for TCP information.
	<i>udp</i> group	udp	The MIB for UDP information.
	snmp group	snmp	The MIB for SNMP information.

## **Detailed Descriptions of MIB Objects**

## • System Group

Subtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
sysDescr	(1) RO Device information (including hardware, OS,	Sup- ported.	"OMRON Corporation" + CPU Unit model + CPU Unit version
	software names, and versions) ASCII characters only.		CPU Unit model (example):     NJ501-1300
			CPU Unit version (example):     Version 1.0
sysObjectID	(2) RO	Sup-	NX-series CPU Unit:
	Vendor OID.	ported.	1.3.6.1.4.1.16838.1.1025.5
	Tells where this device information was assigned in the private MIB.		• NJ-series CPU Unit: 1.3.6.1.4.1.16838.1.1025.4
sysUpTime	(3) RO	Sup-	According to the standard.
	The time elapsed since the system was started (unit: 1/100 s).	ported.	
sysContact	(4) RW	Sup-	Set by the user.
	How to contact the administrator and information on the administrator.	ported.	
sysName	(5) RW	Sup-	CPU Unit name
	The name for management. Sets the full domain name of the device.	ported.	
sysLocation	(6) RW	Sup-	Set by the user.
	The physical location of the device.	ported.	
sysServices	(7) RO	Sup-	64
	The value of the provided service.	ported.	

## • Interfaces Group

Subtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ifNumber	(1) RO The number of network interfaces.	Sup- ported.	NX-series CPU     Unit: 3     NJ-series CPU     Unit: 2
ifTable	(2) NA		
	Interface entity table		
ifEntry	(1) NA Row data for interface information The index is <i>ifIndex</i> .		
ifIndex	(1) RO A number used to identify the interface.	Sup- ported.	NX-series CPU     Unit: 1 to 3     NJ-series CPU     Unit: 1 to 2
ifDescr	(2) RO Information related to the interface (includes manufacturer name, product name, and hardware interface version).	Sup- ported.	NX-series CPU     Unit: 10/100/1000M     Gigabit Ethernet     Port     NJ-series CPU     Unit: 10/100M Fast     Ethernet Port
ifType	(3) RO The type of interface classified according to the physical/link layer protocol directly under the network layer of the protocol stack.	Sup- ported.	ethernet-csmacd(6)
ifMtu	(4) RO MTU value The maximum size (in octets) of datagrams that can be sent and received through this interface.	Sup- ported.	1,500
ifSpeed	(5) RO Estimated bandwidth If a stable, accurate value cannot be obtained for the bandwidth, a nominal value is set instead.	Sup- ported.	NX-series CPU     Unit:     10000000/     1000000000      NJ-series CPU     Unit:     10000000/     10000000/     100000000/
ifPhysAddress	(6) RO MAC address The physical address under the network layer of the interface.	Sup- ported.	The MAC address of the EtherNet/IP port.
ifAdminStatus	(7) RW The preferred status of the interface. You cannot send normal packets in the testing state. up(1) down(2) testing(3)	Sup- ported.	According to the standard.
ifOperStatus	(8) RO The current status of the interface. You cannot send normal packets in the testing state. up(1) down(2) testing(3)	Sup- ported.	According to the standard.

btree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ifLastChange	(9) RO	Sup-	According to the
	The sysUpTime (in 0.01 seconds) at the last change in ifOperStatus for this interface.	ported.	standard.
ifInOctets	(10) RO	Sup-	According to the
	The number of octets received through this interface. This includes framing characters.	ported.	standard.
ifInUcastPkts	(11) RO	Sup-	According to the
	The number of unicast packets reported to a higher level protocol.	ported.	standard.
ifInNUcastPkts	(12) RO	Sup-	According to the
	The number of non-unicast packets (broadcast or multicast packets) reported to a higher level protocol.	ported.	standard.
ifInDiscards	(13) RO	Sup-	According to the
	The number of packets that had no errors but could not be passed to a higher level protocol (i.e., the number of packets received but discarded due to a buffer overflow).	ported.	standard.
ifInErrors	(14) RO	Sup-	According to the
	The number of packets discarded because they contained errors.	ported.	standard.
ifInUnknown	(15) RO	Sup-	According to the
Protos	The number of packets received, but discarded because they were of an illegal or unsupported protocol.	ported.	standard.
	For example, Ethernet packets did not have IP set for the field that identifies their higher level protocol.		
ifOutOctets	(16) RO	Sup-	According to the
	The number of octets of packets sent through this interface.	ported.	standard.
	This includes framing characters.		
ifOutUcast	(17) RO	Sup-	According to the
Pkts	The number of unicast packets sent by higher level protocols.	ported.	standard.
	This includes discarded packets and unsent packets.		
ifOutNUcast	(18) RO	Sup-	According to the
Pkts	The number of non-unicast packets sent by higher level protocols.	ported.	standard.
	This includes discarded packets and unsent packets.		
ifOutDiscards	(19) RO	Sup-	According to the
	The number of packets that had no errors but were discarded in the sending process (due to a send buffer overflow, etc.).	ported.	standard.
ifOutErrors	(20) RO	Sup-	According to the
	The number of packets that could not be sent because of an error.	ported.	standard.
ifOutQLen	(21) RO	Sup-	Always 0.
	The size of the send packet queue (i.e., the number of packets).	ported.	

Subtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ifSpecific	(22) RO	Sup-	0.0
	The object ID that represents a reference to the media-specific MIB for the interface.	ported.	
	For example, for Ethernet, set the object ID of the MIB that defines Ethernet. If there is no information, set { 0.0 }.		

## • Ip Group: Ip

Subtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ipForwarding	(1) RW	Sup-	NX-series CPU
	Indicates if the device operates as a gateway. IP gateways can transfer datagrams, but IP hosts can perform only source routing. Some nodes take only one of these values. Therefore, if you attempt to change this object from the SNMP Manager, a <i>badValue</i> error is returned.	ported.	Unit: Forwarding(1)  NJ-series CPU Unit: Not-forwarding (2)
	Forwarding (1) Not-forwarding (2)		
lpDefaultTTL	(2) RW	Sup-	64
,podauni i z	The default value set for the IP header TTL if no TTL value was given by the transport layer protocol.	ported.	
IpInReceives	(3) RO	Sup-	According to the
	The number of all IP datagrams that reached the interface, including errors.	ported.	standard.
IpInHdrErrors	(4) RO	Sup-	According to the
	The number of received datagrams that were discarded because of an IP header error (checksum error, version number error, format error, TTL error, IP option error, etc.).	ported.	standard.
IpInAddrErrors	(5) RO	Sup-	According to the
	The number of packets that were discarded because the destination address in the IP header was not valid.	ported.	standard.
ipForwDatagrams	(6) RO	Sup-	According to the
	The number of IP datagrams that were transferred to their final destination. If this node does not operate as an IP gateway, this is the number of datagrams that were successfully transferred through source routing.	ported.	standard.
ipInUnknownProtos	(7) RO	Sup-	According to the
	The number of IP datagrams that were received but discarded because they were of an unsupported or unrecognized protocol.	ported.	standard.
ipInDiscards	(8) RO The number of IP datagrams that could have continued to be processed without any problems, but were discarded (for example, because of insufficient buffer space).	Sup- ported.	According to the standard.
ipInDelivers	(9) RO	Sup-	According to the
	The number of datagrams delivered to an IP user protocol (any higher level protocol, including ICMP).	ported.	standard.
ipOutRequests	(10) RO	Sup-	According to the
	The number of times a send request was made for an IP datagram by a local IP user protocol (any higher level protocol, including ICMP). This counter does not include <i>ipForwDatagrams</i> .	ported.	standard.
ipOutDiscards	(11) RO	Sup-	According to the
	The number of IP datagrams that could have been sent without any problems, but were discarded (for example, because of insufficient buffer space).	ported.	standard.

Sı	ubtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ipOutNo	DRoutes	(12) RO The number of IP datagrams that were discarded because there was no transmission path. This counter includes datagrams that attempted to be sent through <i>ipForwDatagrams</i> , but were discarded because they were set with no-route. This value indicates the number of datagrams that could not be transferred because the default gateway was down.	Sup- ported.	According to the standard.
ipReasr	mTimeout	(13) RO The maximum number of seconds to wait to receive all IP datagrams for reassembly if a fragmented IP datagram is received.	Sup- ported.	60 s
ipReasr	nReqds	(14) RO The number of IP datagrams received that require reassembly. There is a flag in the IP header that indicates if the datagram is fragmented. You can use that flag to identify fragments.	Sup- ported.	According to the standard.
ipReasr	mOKs	(15) RO The number of IP datagrams received that were successfully reassembled.	Sup- ported.	According to the standard.
ipReasr	nFails	(16) RO The number of IP datagrams received that were not successfully reassembled.	Sup- ported.	According to the standard.
ipFragO	Ks	(17) RO The number of IP datagrams that were successfully fragmented.	Sup- ported.	According to the standard.
ipFragF	ails	(18) RO The number of IP datagrams that were not successfully fragmented. (For example, because the Don't Fragment flag was set for the IP datagram.)	Sup- ported.	According to the standard.
ipFragC	reates	(19) RO The number of IP datagrams created as a result of fragmentation.	Sup- ported.	According to the standard.
ipAddrT	able	(20) NA An address information table for IP addresses.		
ip <i>A</i>	AddrEntry	(1) NA Row data of address information for IP addresses. The index is <i>ipAdEntAddr</i> .		
	ipAdEntAddr	(1) RO The IP address.	Sup- ported.	According to the standard.
	ipAdEntIfIndex	(2) RO The index value of the interface that this entry applies to. This is the same value as <i>ifIndex</i> .	Sup- ported.	According to the standard.
	ipAdEntNet Mask	(3) RO The subnet mask for the IP address of this entry.	Sup- ported.	According to the standard.
	ipAdEntB- castAddr	(4) RO The value of the least significant bit of the address when an IP broadcast is sent. An address represented by all 1 bits is used for broadcasting as an Internet standard. In that case, this value is always 1.	Sup- ported.	According to the standard.

	Subtree name	Standard [(identifier) attribute]	Sup-	Implementation
			port	specifications
	ipAdEn- tReasmMax- Size	(5) RO The maximum IP packet size that can be reassembled from IP fragmented input IP datagrams received through the interface.	Sup- ported.	According to the standard.
ipRo	outeTable	(21) NA		
		The IP routing table for this entity.		
	ipRouteEntry	(1) NA		
		Route information for a specific destination. The index is <i>ipRouteDest</i> .		
	ipRouteDest	(1) RW	Sup-	According to the
		The destination IP address for this route. A value of 0.0.0.0 for this entry indicates the default route.	ported.	standard.
	ipRoutelfIndex	(2) RW	Sup-	According to the
		The ID number of the interface required to send to the next destination host in this route. This ID number is the same number as <i>ifIndex</i> , which is used to identify the interface.	ported.	standard.
	ipRouteMetric1	(3) RW	Sup-	According to the
		The primary routing metric for this route. This value is determined based on the protocol specified in <i>ipRouteProto</i> . Set to –1 if you do not want to use this metric (this is also the same for <i>ipRouteMetric</i> 2 through 4).	ported.	standard.
	ipRouteMetric2	(4) RW	Sup-	According to the
		The alternative routing metric for this route.	ported.	standard.
	ipRouteMetric3	(5) RW	Sup-	According to the
		The alternative routing metric for this route.	ported.	standard.
	ipRouteMetric4	(6) RW	Sup-	According to the
		The alternative routing metric for this route.	ported.	standard.
	ipRouteNext	(7) RW	Sup-	According to the
	Нор	The IP address of the next hop in this route (for	ported.	standard.
		routes connected by a broadcast or media, this is the agent address or address of that interface).		
	ipRouteType	(8) RW	Sup-	According to the
	h Mr.	The type of route.	ported.	standard.
		Other (1): Not any of the following types.		
		Invalid (2): An invalid route.		
		Direct (3): A direct connection.		
		Indirect (4): An indirect connection (not connected to LOCAL).		

Subtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ipRouteProto	(9) RO This is the routing mechanism used to determine routes. Some values correspond to gateway routing protocols, but be aware that the host may not support those protocols.	Sup- ported.	According to the standard.
	Other (1): Other than the following items.		
	Local (2): A route set on the local machine.		
	Netmgmt (3): A route set by network management.		
	Icmp (4): A route set by an ICMP redirect or some other ICMP function.		
	Egp (5): EGP		
	The following are gateway protocols:		
	Ggp (6): GGP		
	Hello (7): HELLO		
	Rip (8): RIP		
	is-is (9)		
	es-is (10)		
	ciscolgrp (11)		
	bbnSpflgp (12)		
	ospf (13): OSPF		
	bgp (14)		
ipRouteAge	(10) RW	Sup-	According to the
	The elapsed time since this route was updated (in seconds).	ported.	standard.
ipRouteMask	(11) RW	Sup-	According to the
	The subnet mask value in relation to <i>ipRouteDest</i> . On systems that do not support a custom subnet mask value, this value is based on the address class of the <i>ipRouteDest</i> field. If <i>ipRouteDest</i> is 0.0.0.0, this value is also 0.0.0.0.	ported.	standard.
ipRouteMetric5	(12) RW	Sup-	According to the
	The alternative routing metric.	ported.	standard.
ipRouteInfo	(13) RO	Sup-	0.0
	The MIB object ID for the routing protocol used by this route. If not defined, set to {0.0}.	ported.	

	Subtree name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
ipNe	etToMediaTable	(22) NA		
		The IP address conversion table used to map IP addresses to physical addresses.		
	ipNetToMediaEntry	(1) NA		
		Row data for the conversion table. The indices are ipNetToMedialfIndex and ipNetToMediaNetAddress.		
	ipNetToMedi-	(1) RW	Sup-	According to the
	alfIndex	The interface ID number for this entry. The value of <i>ifIndex</i> is used for this value.	ported.	standard.
	ipNetToMedi-	(2) RW	Sup-	According to the
	aPhysAddress	The media-dependent physical address.	ported.	standard.
	ipNetToMedi-	(3) RW	Sup-	According to the
	aNetAddress	The IP address that corresponds to the media- dependent physical address.	ported.	standard.
	ipNetToMedi-	(4) RW	Sup-	According to the
	аТуре	The address conversion method.	ported.	standard.
		Other (1): A method other than the following items.		
		Invalid (2): An invalid value.		
		Dynamic (3): Dynamic conversion.		
		Static (4): Static conversion.		
ipRo	outingDiscards	(23) RO	Sup-	According to the
		The number of routing entries that were valid but discarded. For example, if there was not enough buffer space because of other routing entries.	ported.	standard.

## • Ip Group: Icmp

Name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
icmpInMsgs	(1) RO	Sup-	According to the
	The total number of received ICMP messages. This includes messages counted by <i>icmpInErrors</i> .	ported.	standard.
icmpInErrors	(2) RO	Sup-	According to the
·	The number of received ICMP message errors. (Checksum errors, frame length errors, etc.)	ported.	standard.
icmpInDestUnreachs	(3) RO	Sup-	According to the
•	The number of <i>Destination Unreachable</i> messages received.	ported.	standard.
icmpInTimeExcds	(4) RO	Sup-	According to the
	The number of <i>Time Exceed</i> messages received.	ported.	standard.
icmpInParmProbs	(5) RO	Sup-	According to the
	The number of <i>Parameter Problem</i> messages received.	ported.	standard.
icmpInSrcQuenchs	(6) RO	Sup-	According to the
	The number of <i>Source Quench</i> messages received.	ported.	standard.
icmpInRedirects	(7) RO	Sup-	According to the
	The number of <i>Redirect</i> messages received.	ported.	standard.
icmplnEchos	(8) RO	Sup-	According to the
	The number of <i>Echo (request)</i> messages received.	ported.	standard.
icmpInEchoReps	(9) RO	Sup-	According to the
	The number of <i>Echo Reply</i> messages received.	ported.	standard.
icmpInTimestamps	(10) RO	Sup-	According to the
	The number of <i>Timestamp</i> messages received.	ported.	standard.
icmpInTimestampReps	(11) RO	Sup-	According to the
	The number of <i>Timestamp Reply</i> messages received.	ported.	standard.
icmplnAddrMasks	(12) RO	Sup-	According to the
	The number of <i>Address Mask Request</i> messages received.	ported.	standard.
icmplnAddrMaskReps	(13) RO	Sup-	According to the
	The number of <i>Address Mask Reply</i> messages received.	ported.	standard.
icmpOutMsgs	(14) RO	Sup-	According to the
	The total number of ICMP messages sent. This includes messages counted by <i>icmpOutErrors</i> .	ported.	standard.
icmpOutErrors	(15) RO	Sup-	According to the
	The number of ICMP messages that could not be sent because of an error.	ported.	standard.
icmpOutDestUnreachs	(16) RO	Sup-	According to the
	The number of <i>Destination Unreachable</i> messages sent.	ported.	standard.
icmpOutTimeExcds	(17) RO	Sup-	According to the
	The number of <i>Time Exceed</i> messages sent.	ported.	standard.
icmpOutParmProbs	(18) RO	Sup-	According to the
	The number of <i>Parameter Problem</i> messages sent.	ported.	standard.
icmpOutSrcQuenchs	(19) RO	Sup-	According to the
•	The number of Source Quench messages sent.	ported.	standard.

Name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
icmpOutRedirects	(20) RO	Sup-	According to the
	The number of Redirect messages sent.	ported.	standard.
icmpOutEchos	(21) RO	Sup-	According to the
	The number of <i>Echo (request)</i> messages sent.	ported.	standard.
icmpOutEchoReps	(22) RO	Sup-	According to the standard.
	The number of Echo Reply messages sent.	ported.	
icmpOutTimestamps	(23) RO	Sup-	According to the
	The number of <i>Timestamp</i> messages sent.	ported.	standard.
icmpOutTimestampReps	(24) RO	Sup-	According to the
	The number of <i>Timestamp Reply</i> messages sent.	ported.	standard.
icmpOutAddrMasks	(25) RO	Sup-	According to the
	The number of Address Mask Request messages	ported.	standard.
	sent.		
icmpOutAddrMaskReps	(26) RO	Sup-	According to the
	The number of <i>Address Mask Reply</i> messages sent.	ported.	standard.

## • Ip Group: Tcp

Name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
tcpRtoAlgorithm	(1) RO	Sup-	According to the
	The algorithm used to determine the timeout value for resending.	ported.	standard.
	Other (1): Other than the following items.		
	Constant (2): A constant RTO value.		
	Rsre (3): The algorithm specified by the MIL-STD-1778 standard.		
	Vanj (4): The Van Jacobson algorithm.		
tcpRtoMin	(2) RO	Sup-	According to the
	The minimum resend timeout value (in 0.01 s). This value depends on the algorithm used to determine the resend timeout value.	ported.	standard.
tcpRtoMax	(3) RO	Sup-	According to the
	The maximum resend timeout value (in 0.01 s). This value depends on the algorithm used to determine the resend timeout value.	ported.	standard.
tcpMaxConn	(4) RO	Sup-	According to the
	The total number of supported TCP connections. If the maximum number of connections is dynamic, this value is -1.	ported.	standard.
tcpActiveOpens	(5) RO	Sup-	According to the
	The number of times the TCP connection changed from the CLOSE state directly to the SYN-SENT state. (Active connection establishment.)	ported.	standard.
tcpPassiveOpens	(6) RO	Sup-	According to the
	The number of times the TCP connection changed from the LISTEN state directly to the SYN-RCVD state. (Passive connection establishment.)	ported.	standard.
tcpAttemptFails	(7) RO	Sup-	According to the
	The total number of times the TCP connection changed from the SYN-SENT or SYN-RCVD state directly to the CLOSE state and from the SYN-RCVD state directly to the LISTEN state.	ported.	standard.
tcpEstabResets	(8) RO	Sup-	According to the
·	The number of times the TCP connection changed from the ESTABLISHED or the CLOSE-WAIT state directly to the CLOSE state.	ported.	standard.
tcpCurrEstab	(9) RO	Sup-	According to the
·	The total number of TCP connections currently in the ESTABLISHED or the CLOSE-WAIT state.	ported.	standard.
tcpInSegs	(10) RO	Sup-	According to the
	The total number of received segments. This includes the number of error segments.	ported.	standard.
tcpOutSegs	(11) RO	Sup-	According to the
	The total number of sent segments. This includes the number of segments for the current connec- tion, but does not include the number of segments for resent data only.	ported.	standard.
tcpRetransSegs	(12) RO	Sup-	According to the
	The total number of resent segments.	ported.	standard.

Name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
tcpConnTable	(13) NA The information table specific to the TCP connection.		According to the standard.
tcpConnEntry	(1) NA Entry information related to a specific TCP connection. This value is deleted if the connection changes to the CLOSE state. The indices are tcp-ConnLocalAddress, tcpConnLocalPort, tcpConnRemAddress, and tcpConnRemPort.		According to the standard.
tcpConnState	(1) RW The status of the TCP connection. closed(1) listen(2) synSent(3) synReceived(4) established(5) finWait1(6) finWait2(7) closeWait(8) lastAck(9) closing(10) timeWait(11)	Sup- ported.	According to the standard.
tcpConnLoca- IAddress	(2) RO The local IP address of this TCP connection. A value of 0.0.0.0 is used for connections in the LISTEN state that accept connections from any IP interface related to the node.	Sup- ported.	According to the standard.
tcpConnLocal- Port tcpConnRe-	(3) RO The local port number for this TCP connection. (4) RO	Sup- ported. Sup-	According to the standard.  According to the
mAddress tcpConnRem-	The remote IP address for this TCP connection. (5) RO	ported. Sup-	standard.  According to the
tcpInErrs	The remote port number for this TCP connection.  (14) RO  The total number of error segments received (TCP checksum errors, etc.).	Sup- ported.	According to the standard.
tcpOutRsts	(15) RO The number of segments sent with the RST flag (the number of times the TCP connection was reset).	Sup- ported.	According to the standard.

## • Ip Group: Udp

Name	Standard [(identifier) attribute]	Sup- port	Implementation specifications
udpInDatagrams	(1) RO	Sup-	According to the stan-
	The total number of UDP datagrams (i.e., the number of packets) sent to the UDP user.	ported.	dard.
udpNoPorts	(2) RO	Sup-	According to the stan-
	The number of UDP datagrams that were received but did not start an application at the destination port.	ported.	dard.
udpInErrors	(3) RO	Sup-	According to the stan-
	The number of UDP datagrams that were not sent to a higher level protocol for a reason other than <i>udpNoPorts</i> .	ported.	dard.
udpOutDatagrams	(4) RO	Sup-	According to the stan-
	The total number of sent UDP datagrams.	ported.	dard.
udpTable	(5) NA		According to the stan-
	The information table for the UDP listener.		dard.
udpEntry	(1) NA		According to the stan-
	An entry related to a specific UDP listener. The indices are <i>udpLocalAddress</i> and <i>udpLocalPort</i> .		dard.
udpLocal	(1) RO	Sup-	According to the stan-
Address	The local IP address of this UDP listener. A value of 0.0.0.0 is used for UDP listeners that accept datagrams from any IP interface related to the node.	ported.	dard.
udpLocalPort	(2) RO	Sup-	According to the stan-
	The local port number for this UDP listener.	ported.	dard.

## • Ip Group: Snmp

Name			Standard [(identi- fier) attribute]
snmpInPkts	(1) RO	Sup-	According to the
	The total number of SNMP messages received.	ported.	standard.
snmpOutPkts	(2) RO	Sup-	According to the
	The total number of SNMP messages sent.	ported.	standard.
snmpInBadVersions	(3) RO	Sup-	According to the
	The total number of messages received of an	ported.	standard.
	unsupported version.		
snmpInBadCommunity-	(4) RO	Sup-	According to the
Names	The total number of messages received from an	ported.	standard.
	unregistered community.		
snmpInBadCommunity-	(5) RO	Sup-	According to the
Uses	The total number of messages received that spec-	ported.	standard.
	ify an operation that is not allowed by that commu-		
	nity.		
snmpInASNParseErrs	(6) RO	Sup-	According to the
	The total number of messages received that	ported.	standard.
	resulted in an ASN.1 error or BER error during		
	decoding.		A P 1
snmpInTooBigs	(8) RO	Sup-	According to the standard.
	The total number of PDUs received with an error status of <i>tooBig</i> .	ported.	รเลกนสาน.

Name			Standard [(identi- fier) attribute]
snmpInNoSuchNames	(9) RO	Sup-	According to the
	The total number of PDUs received with an error status of <i>noSuchName</i> .	ported.	standard.
snmpInBadValues	(10) RO	Sup-	According to the
	The total number of PDUs received with an error status of <i>badValue</i> .	ported.	standard.
snmpInReadOnlys	(11) RO	Sup-	According to the
	The total number of PDUs received with an error status of <i>readOnly</i> .	ported.	standard.
snmpInGenErrs	(12) RO	Sup-	According to the
	The total number of PDUs received with an error status of <i>genErr</i> .	ported.	standard.
snmpInTotalReqVars	(13) RO	Sup-	According to the
	The total number of MIB objects read normally after receiving GetRequest or GetNextRequest.	ported.	standard.
snmpInTotalSetVars	(14) RO	Sup-	According to the
	The total number of MIB objects updated normally after receiving <i>SetRequest</i> .	ported.	standard.
snmpInGetRequests	(15) RO	Sup-	According to the
	The total number of <i>GetRequest</i> PDUs received.	ported.	standard.
snmpInGetNexts	(16) RO	Sup-	According to the
	The total number of <i>GetNextRequest</i> PDUs received.	ported.	standard.
snmpInSetRequests	(17) RO	Sup-	According to the
	The total number of SetRequest PDUs received.	ported.	standard.
snmpInGetResponses	(18) RO	Sup-	According to the
	The total number of <i>GetResponse</i> PDUs received.	ported.	standard.
snmpInTraps	(19) RO	Sup-	According to the
	The total number of trap PDUs received.	ported.	standard.
snmpOutTooBigs	(20) RO	Sup-	According to the
	The total number of PDUs sent with an error status of <i>tooBig</i> .	ported.	standard.
snmpOutNoSuchNames	(21) RO	Sup-	According to the
	The total number of PDUs sent with an error status of <i>noSuchName</i> .	ported.	standard.
snmpOutBadValues	(22) RO	Sup-	According to the
	The total number of PDUs sent with an error status of badValue.	ported.	standard.
snmpOutGenErrs	(24) RO	Sup-	According to the
	The total number of PDUs sent with an error status of <i>genErr</i> .	ported.	standard.
snmpOutGetRequests	(25) RO	Sup-	According to the
	The total number of GetRequest PDUs sent.	ported.	standard.
snmpOutGetNexts	(26) RO	Sup-	According to the
	The total number of GetNextRequest PDUs sent.	ported.	standard.
snmpOutSetRequests	(27) RO The total number of <i>SetRequest</i> PDUs sent.	Sup- ported.	According to the standard.
snmpOutGetResponses	(28) RO	Sup-	According to the
i	The total number of <i>GetResponse</i> PDUs sent.	ported.	standard.
snmpOutTraps	(29) RO	Sup-	According to the
	The total number of trap PDUs sent.	ported.	standard.

Name			Standard [(identi- fier) attribute]
snmpEnableAuthen	(30) RW	Sup-	According to the
Traps	Determines if the agent generates verification failed traps.	ported.	standard.
	Enabled (1)		
	Disabled (2)		

# 13-2 Procedure to Use the SNMP Agent

#### 13-2-1 Procedures

- Make the basic settings.
  Refer to 1-5 EtherNet/IP Communications Procedures for the flow of basic operations.
- 2 Select Built-in EtherNet/IP Port Settings from the Controller Setup on the Sysmac Studio, and then set the following on the SNMP Settings Display or SNMP Trap Settings Display.
  - SNMP Service
  - · Recognition 1
  - · Recognition 2
- 3 Select *Transfer to Controller* from the Controller Menu and click the **Yes** Button. The built-in EtherNet/IP port settings are transferred to the CPU Unit.

# 13-2-2 Settings Required for the SNMP Agent

The following Built-in EtherNet/IP Port Settings are made from the Sysmac Studio to use the SNMP agent.

Tab page		Setting	Setting conditions	Reference
SNMP Settings	SNMP s	service	Required.	page 5-10
	Port No.		Specified by user.	
			Note Required to change from the default value of 161.	
	Contact	, location	Specified by user.	
	Send a	recognition trap	Specified by user.	
			Select this check box to send a recognition trap if there is access from an SNMP manager that is not specified (Access other than Recognition 1 and 2).	
	Recogn	ition 1 and Rec-	Specified by user.	page 5-11
	ognition	2	Make these settings to permit access by	
		IP address	only certain SNMP managers.	
		Host name		
		Community name		

Tab page		Setting	Setting conditions	Reference
SNMP Trap Settings	SNMP trap		Required.	page 5-12
	Port No	).	Specified by user.	
			Note Required to change from the default value of 162.	
	Trap 1 a	and trap 2		page 5-13
		IP address	Required.	
		Host name	Set an IP address or a host name as the SNMP trap destination.	
		Community name	Specified by user.	
		Version	Required.	]
			Set the version of the SNMP manager.	



## **Additional Information**

Make the settings in the SNMP Settings Dialog Box and SNMP Trap Dialog Box if the SNMP agent is used.

Refer to 5-5 SNMP Settings Display for information on the SNMP Settings Dialog Box. Refer to 5-6 SNMP Trap Settings Display for information on the SNMP Trap Dialog Box.



# **Communications Performance** and Communications Load

14-1 Comm	unications System	14-2
14-1-1	Tag Data Link Communications Method	14-2
14-1-2	Calculating the Number of Connections	14-4
14-1-3	Packet Interval (RPI) Accuracy	14-5
14-2 Adjust	ing the Communications Load	14-6
14-2-1	Checking Bandwidth Usage for Tag Data Links	14-7
14-2-2	Tag Data Link Bandwidth Usage and RPI	14-8
14-2-3	Adjusting Device Bandwidth Usage	14-9
14-2-4	Changing the RPI	14-10
14-2-5	RPI Setting Examples	14-16
14-3 I/O Res	sponse Time in Tag Data Links	14-21
14-3-1	Timing of Data Transmissions	14-21
14-3-2	Built-in EtherNet/IP Port Data Processing Time	14-22
14-3-3	Relationship between Task Periods and Packet Intervals (RPIs)	14-24
14-3-4	Maximum Tag Data Link I/O Response Time	14-25
14-4 Messa	ge Service Transmission Delay	14-27

# 14-1 Communications System

# **Tag Data Link Communications Method**

# **Requested Packet Interval (RPI) Settings**

In tag data links for the built-in EtherNet/IP port, the data transmission period is set for each connection as the RPI.

The target device will send data (i.e., output tags) once each RPI, regardless of the number of nodes.

Also, the heartbeat frame is sent from the originator to the target for each connection. The target uses the heartbeat to check to see if errors have occurred in the connection with the originator. The data transmission period of the heartbeat frame depends on the RPI settings.

#### **Heartbeat Frame Transmission Period**

- If packet interval < 100 ms, the heartbeat frame transmission period is 100 ms.
- If packet interval ≥ 100 ms, the heartbeat frame transmission period is the same as the RPI.

#### **Example:**

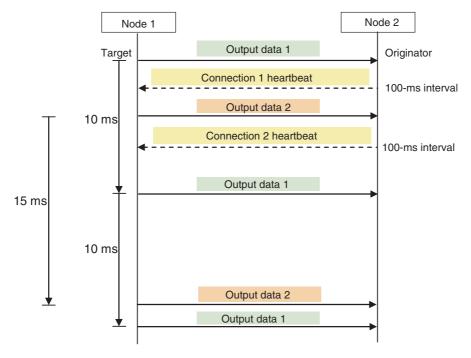
In this example, 2 tag data link connections are set for node 2 (the originator) and node 1 (the tar-

The RPI for output data 1 is set to 10 ms.

The RPI for output data 2 is set to 15 ms.

In this case, output data 1 is sent from node 1 to node 2 every 10 ms, and output data 2 is sent from node 1 to node 2 every 15 ms, as shown in the following diagram.

Also, data is sent from node 2 (the originator) to node 1 (the target) with a heartbeat of 100 ms for connection 1 and a heartbeat of 100 ms for connection 2.



# Requested Packet Interval (RPI) and Bandwidth Usage (PPS)

The number of packets transferred each second is called the used bandwidth or PPS (packets per second).

The PPS is calculated from the RPI and heartbeat as follows for each connection:

PPS used in a connection (pps) = (1,000 ÷ RPI (ms)) + (1,000 ÷ Heartbeat transmission period (ms))

Use the following equation to calculate the total number of packets transferred by each built-in Ether-Net/IP port (Unit) in 1 second.

Built-in EtherNet/IP port's total PPS = Total PPS of originator connections + Total PPS of target connections\*

\* Connections set as target connections must be added, too.

The maximum number of packets that the built-in EtherNet/IP port for each series can transfer in 1 second (called the allowed Unit bandwidth) is as follows, so set the connection below this maximum value.

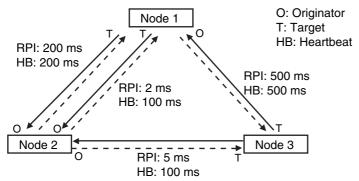
- NX-series CPU Unit: 40,000 pps
- NJ-series CPU Unit: 3,000 pps\*
- \* Note that the allowed Unit bandwidth is 1,000 pps for the NJ-series CPU Unit with a unit version 1.00 to 1.02.

#### Example:

Node 1 has both originator and target connections, with send RPI of 200 ms and 2 ms, and receive RPI of 500 ms.

Node 2 has originator connections only, with receive RPIs of 200 ms, 2 ms, and 5 ms.

Node 3 has target connections only, with send RPIs of 5 ms and 1 ms.



Each node's total PPS is calculated as follows:

- Total PPS of node 1's Unit
  - = 1000 / 200 ms + 1000 / 2 ms + 1000 / 500 ms (for data)
  - + 1000 / 200 ms + 1000 / 100 ms + 1000 / 500 ms (for heartbeat)
  - = 524 pps
- Total PPS of node 2's Unit
  - = 1000 / 200 ms + 1000 / 2 ms + 1000 / 5 ms (for data)
  - + 1000 / 200ms + 1000 / 100 ms + 1000 / 100 ms (for heartbeat)
  - = 730 pps
- · Total PPS of node 3's Unit
  - = 1000 / 5 ms + 1000 / 500 ms (for data)
  - + 1000 / 100 ms + 1000 / 500 ms (for heartbeat)
  - =214pps

All of the Units are within the allowed Unit bandwidth, so they can transfer data.

# 14-1-2 Calculating the Number of Connections

The maximum number of connections for the built-in EtherNet/IP port for each series is as follows:

NX-series CPU Unit: 256

• NJ-series CPU Unit: 32

The number of connections must be set to the above numbers or less combining both connections that the Unit opens as the originator and connections that are opened from an originator with the Unit as the target.

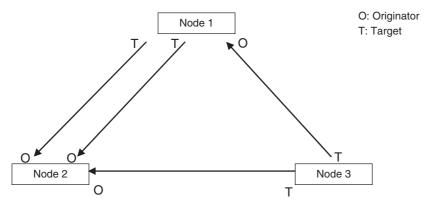
#### Example:

Node 1 opens two connections as the target with node 2 and one connection as the originator with node 3. Therefore, the total is three connections.

Node 2 opens two connections as the originator with node 1 and one connection as the originator with node 2. Therefore, the total is two connections.

Node 3 opens one connection as the target with node 1 and one connection as the target with node 2. Therefore, the total is two connections.

In either case, the connections can be opened because the numbers of connections for all nodes are within the maximum number of connections for the built-in EtherNet/IP port.

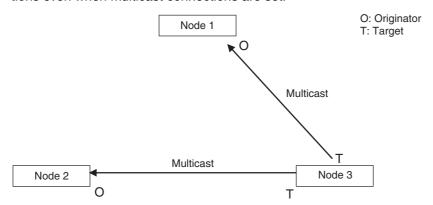


Also, if multicast is set, one packet will be sent, but the number of connections will be consumed.

#### Example:

Node 3 sends one multicast packet to node 1 and node 2. Node 3 opens one connection as the target with node 1 and one connection as the target with node 2.

Caution is required because the number of connections consumed is the same as for unicast connections even when multicast connections are set.

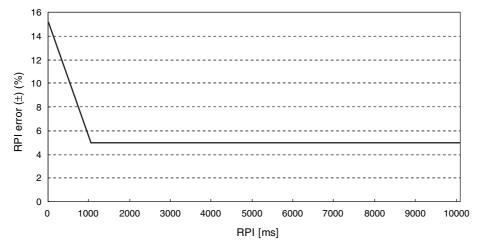


# 14-1-3 Packet Interval (RPI) Accuracy

The send processing delay occurs in the built-in EtherNet/IP port when data packets are sent once each packet interval (RPI). This delay varies with the RPI error shown in the following graph, so the send processing delay time is the maximum value for each RPI.

Packet interval (RPI)	RPI error (±) (%)
0.5 ms to 1,000 ms (NX-series CPU Unit), 1 ms to 1,000 ms (NJ-series CPU Unit)*1	15 – (RPI [ms]/100)
1,000 to 10,000 ms	5% of the RPI

<sup>\*1</sup> RPI is 10 ms to 1,000 ms on NJ-series CPU Units with unit versions 1.00 to 1.02.



# 14-2 Adjusting the Communications Load

In an Ethernet network using an Ethernet switch, the network bandwidth is not shared by all of the nodes; independent transmission paths are established between individual nodes through the Ethernet switch.

A dedicated communications buffer is established in the Ethernet switch for communications between the nodes and full-duplex communications (simultaneous transmission and reception) are performed asynchronously with other transmission paths. The communications load in other transmission paths does not affect communications, so packet collisions do not occur and stable, high-speed communications can be performed.

The Ethernet switch functions shown in the following table determine the performance of tag data links.

Item	Description
Buffer capacity	This is the amount of data that can be buffered when packets accumulate at the Ethernet switch.
Multicast filtering	This function transfers multicast packets to specific nodes only.
QoS function	This function performs priority control on packet transfers.

The following table shows the setting ranges of the tag data link settings that can be made for a built-in EtherNet/IP port.

			NJ-series	CPU Unit
Item	Description	NX-series CPU Unit	Unit version 1.00 to	Unit ver- sion 1.03 or later
Network bandwidth	Physical Ethernet baud rate	1,000 Mbps	100 Mbps or 1	0 Mbps
Allowed tag data link communications bandwidth	Maximum number of tag data link packets that can be processed in 1 second (pps: packets per second)	40,000 pps max. (total of 40,000 with two ports)	1,000 pps max.	3,000 pps max.
Connection resources	Number of connections that can be established	256 max. (total of 512 with two ports)	32 max.	
Packet interval (RPI: Requested Packet Interval)	Refresh period for tag data	0.5 to 10,000 ms in 0.5-ms increments	10 to 10,000 ms in 1-ms increments	1 to 10,000 ms in 1-ms increments

When the tag data link settings exceed the capabilities of the Ethernet switch being used, increase the packet interval (RPI) value. Particularly when using an Ethernet switch that does not support multicast filtering, the settings must be made considering that multicast packets will be sent even to nodes without connection settings.



#### **Additional Information**

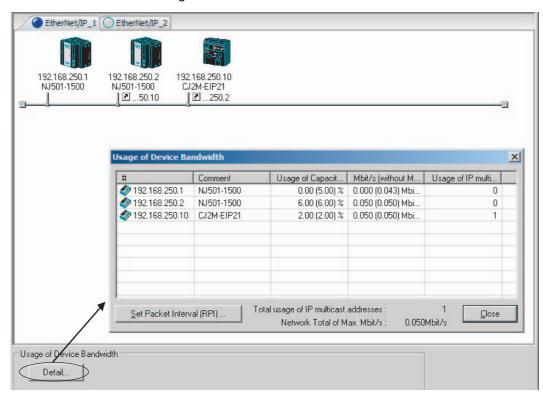
If the Network Configurator is used to set the connection type in the connection settings to a multicast connection, multicast packets will be used. If the connection type is set to a point-to-point connection, multicast packets are not used.

In addition, if the required tag data link performance cannot be achieved with the Ethernet switch's capabilities, re-evaluate the overall network configuration and correct it by taking steps such as selecting a different Ethernet switch or splitting the network.

The following sections show how to check the device bandwidth being used by the tag data links in the designed network, and how to set the appropriate values.

# 14-2-1 Checking Bandwidth Usage for Tag Data Links

The Network Configurator can display the bandwidth actually used for tag data links at each built-in EtherNet/IP port, based on the connections set in the network configuration. The device bandwidth used by tag data links can be checked by clicking the **Detail** Button in the Usage of Device Bandwidth Area at the bottom of the Network Configuration Window.



Item	Description
#	The IP address of the device.
Comment	A description of the device. The comment is displayed below the device icon. The model number of the device is displayed by default.
Usage of Capacity (without Multi- cast Filter)	The percentage of the allowable communications bandwidth used for tag data links for the device is displayed. Bandwidth used ÷ Allowable tag data link bandwidth
	The values outside parentheses are for when multicast filtering is used.
	The values inside parentheses are for when multicast filtering is not used.
Mbit/s (without Multicast Filter)	The bandwidth used for communications by the device of the network bandwidth is shown.
	The values outside parentheses are for when multicast filtering is used.
	The values inside parentheses are for when multicast filtering is not used.
Usage of IP Multicast Addresses	The number of multicast IP addresses actually used for communications by the device is shown.
Total usage of IP multicast addresses	The number of multicast IP addresses used in the entire network is shown. This value is used to estimate the number of multicast filters for switching.
Network Total of Max. Mbit/s	The total network bandwidth used for tag data link communications in the entire network is shown. Tag data links will not operate normally if the network bandwidth that can be set is exceeded.

## Checking the Usage of Capacity and Network Bandwidth for Tag Data Links

The percentage of the allowable communications bandwidth for tag data links for each built-in Ether-Net/IP port is displayed as the Usage of Capacity and the bandwidth used for tag data link communications in the entire network is displayed as the Mbit/s. The usage of capacity and used network bandwidth that are displayed in parentheses are for an Ethernet switch that does not use multicast filtering. In this case, multicast packets will be sent to even the nodes without connection settings, so the displayed values will include these packets as well. These values can be adjusted according to instructions in 14-2-4 Changing the RPI.

#### Checking the Total Number of Multicast IP Addresses in the Network

When using an Ethernet switch that provides multicast filtering, there must be enough multicast filters for the network being used. The number of multicast IP address used in the entire network that is displayed by the Network Configurator is based on connection settings. Make sure that the number of multicast IP addresses used in the entire network does not exceed the number of multicast filters supported by the Ethernet switch. If necessary, change to an Ethernet switch with enough multicast filters, or adjust the usage of capacity and network bandwidth for tag data links (Mbit/s) values given for an Ethernet switch without multicast filtering (i.e., the values in parentheses). These values can be adjusted according to instructions in 14-2-4 Changing the RPI.

#### Checking the Total Maximum Network Bandwidth

The Network Configurator displays the total maximum bandwidth that can be used for the entire network. This value indicates the maximum bandwidth that can be used on the transmission paths when Ethernet switches are cascaded. If the value exceeds the bandwidth of a cascade connection in the actual network (e.g., 1,000 Mbps for an NX-series CPU Unit, or 100 Mbps for an NJ-series CPU Unit), the maximum bandwidth for part of the communications path may be exceeded, depending on how the network is wired. This may prevent the tag data links from operating correctly. If this occurs, either calculate the bandwidth usage for each communications path and be sure that the maximum bandwidth is not exceeded for any cascade connection, or adjust the bandwidth for all cascade connections so that the total maximum network bandwidth is not exceeded. Adjust the bandwidth according to instructions in 14-2-4 Changing the RPI.

#### 14-2-2 Tag Data Link Bandwidth Usage and RPI

The usage of capacity without multicast filtering can be adjusted against the tag data link's allowable bandwidth by using the packet interval (RPI) setting. If the RPI is made shorter, the usage of capacity will increase. If the RPI is made longer, the usage of capacity will decrease.

The RPI can be set in any one of the following ways.

- Setting the same interval for all connections
- Setting a particular device's connection
- Setting a particular connection

When the same RPI is set for all connections, the usage of capacity will basically increase proportionally as the RPI is made shorter.

Example: If the RPI is set to 50 ms for all connections and the usage of capacity is 40%, the usage of capacity may increase to 80% when the RPI is reduced to 25 ms for all connections.



#### **Precautions for Correct Use**

Performing message communications or other network operations from the Network Configurator (such as monitoring or other operations that place a load on the network) or from the user application when the tag data link bandwidth usage of capacity is between 80% and 100% can temporarily create an excessive load on the network and result in timeouts. If timeouts occur, increase one or all of the RPI settings and reduce the usage of capacity.

# 14-2-3 Adjusting Device Bandwidth Usage

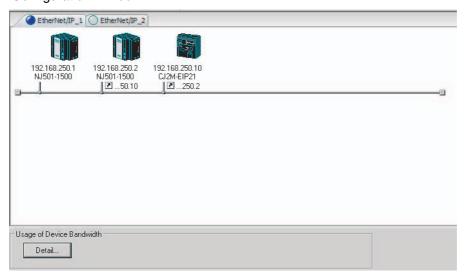
- Ethernet Switches without Multicast Filtering (Supports the Maximum Network Bandwidth That Can Be Set, i.e., 1,000 Mbps for an NX-series CPU Unit or 100 Mbps for an NJ-series CPU Unit)
  - Is the network bandwidth usage without multicast filtering under the network bandwidth that can be set for each node?
    - If any node exceeds its maximum network bandwidth that can be set, change the connections settings, such as the RPI.
  - Is the usage of capacity without multicast filtering under 100% for each node?
     If any node exceeds 100%, change the connections settings, such as the RPI.
  - Is the total network bandwidth usage under the network bandwidth that can be set? If the total bandwidth usage exceeds the network bandwidth that can be set, the bandwidth of part of the transmission path (e.g., an Ethernet switch or media converter) may be exceeded as the result of how the network was wired (e.g., cascade connections of Ethernet switches), causing a tag data link to operate abnormally. Check the bandwidth of the transmission path for all cascade connections. If the bandwidth is exceeded, rewire the network or increase the bandwidth between Ethernet switches (e.g., to 1 Gbps). If these countermeasures are not possible, change the connection settings, e.g., the RPI settings, and adjust the bandwidth for all cascade connections until the total network bandwidth is not exceeded.
- Ethernet Switches with Multicast Filtering (Supports the Maximum Network Bandwidth That Can Be Set, i.e., 1,000 Mbps for an NX-series CPU Unit or 100 Mbps for an NJ-series CPU Unit)
  - Is the network bandwidth usage under the network bandwidth that can be set for each node? If any node exceeds its maximum network bandwidth that can be set, change the connections settings, such as the RPI.
  - Is the usage of capacity under 100% for each node?
     If any node exceeds 100%, change the connections settings, such as the RPI.
  - Is the total network bandwidth usage under the network bandwidth that can be set? If the total bandwidth usage exceeds the network bandwidth that can be set, the bandwidth of part of the transmission path (e.g., an Ethernet switch or media converter) may be exceeded as the result of how the network was wired (e.g., cascade connections of Ethernet switches), causing a tag data link to operate abnormally. Check the bandwidth of the transmission path for all cascade connections. If the bandwidth is exceeded, rewire the network or increase the bandwidth between Ethernet switches (e.g., to 1 Gbps). If these countermeasures are not possible, change the connection settings, e.g., the RPI settings, and adjust the bandwidth for all cascade connections until the total network bandwidth is not exceeded.
  - Is the network bandwidth usage without multicast filtering under the network bandwidth that can be set for each node or is the usage of capacity without multicast filtering under 100% for each node?
    - If any node exceeds the maximum network bandwidth that can be set or the usage of 100%, check whether the multicast filtering on the Ethernet switch is functioning correctly. If the number of multicast filters on the Ethernet switch is less than the total usage of IP multicast addresses, bandwidth overloads may occur in some paths and prevent tag data links from operating correctly depending on the network connection (e.g., cascade connections of Ethernet switches). Calculate the number of multicast filters required by each Ethernet switch on the network and make sure that the number does not exceed the number of Ethernet switch multicast filters. If the number of Ethernet switch multicast filters is not sufficient, use switches with enough multicast filters or revise connection settings, such as the RPI settings.

# 14-2-4 Changing the RPI

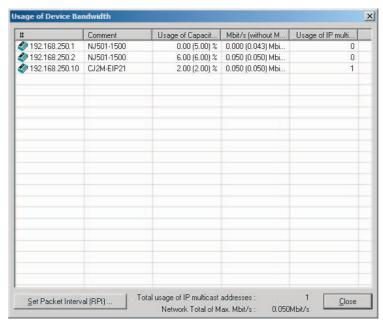
You can check the usage of capacity offline without multicast filtering against the tag data link's allowable bandwidth by following the procedures in 14-2-1 Checking Bandwidth Usage for Tag Data Links. The usage of capacity without multicast filtering can be adjusted against the tag data link's allowable bandwidth by changing the packet interval (RPI).

If the required communications performance cannot be achieved by changing the settings, re-evaluate the network starting with the network configuration.

- Make the required settings in the Network Configurator's Network Configuration Window.
- Click the **Detail** Button in the Usage of Device Bandwidth Area at the bottom of the Network Configuration Window.



The Usage of Device Bandwidth Dialog Box will be displayed.



The Usage of Capacity (without multicast filter) column will show the percentage of the allowed tag data link bandwidth being used, and the Mbit/s (without multicast filter) column will show the network bandwidth being used.

The usage of capacity without multicast filtering can be adjusted against the tag data link's allowable bandwidth by changing the associated devices' packet interval (RPI) settings.

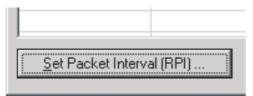
The RPI settings can be changed with the following three methods.

#### Method 1:

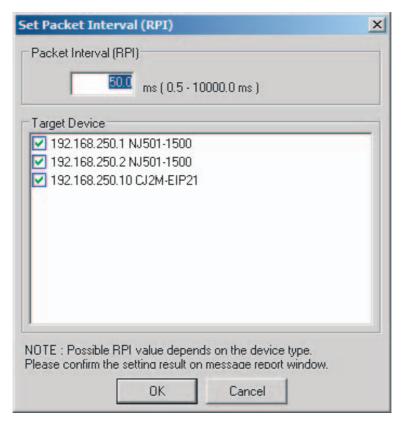
Change All Connections to the Same RPI

The usage of capacity without a multicast filter can be adjusted for all devices by changing the packet intervals (RPI) settings for all of the device's connections to the same RPI at the same time.

(1) Click the Set Packet Interval (RPI) Button at the bottom of the Usage of Device Bandwidth Dialog Box.



(2) The Set Packet Interval (RPI) Dialog Box will be displayed. Input a new RPI value, and click the OK Button.



#### Method 2:

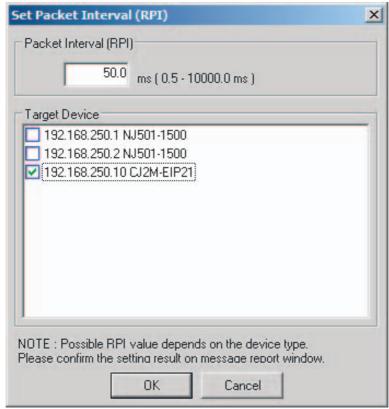
Change a Particular Device's Packet Interval (RPI) Setting:

The usage of capacity without multicast filtering can be adjusted for only a particular device against the tag data link's allowable bandwidth by changing the packet intervals (RPI) settings for all of the device's connections together. In this case, the usage of capacity will also change for the target devices of the connection for which the packet interval is changed.

(1) Click the Set Packet Interval (RPI) Button at the bottom of the Usage of Device Bandwidth Dialog Box.



(2) The Set Packet Interval (RPI) Dialog Box will be displayed. In the *Target Device* Area, deselect the target devices that are not being adjusted by removing the check marks.



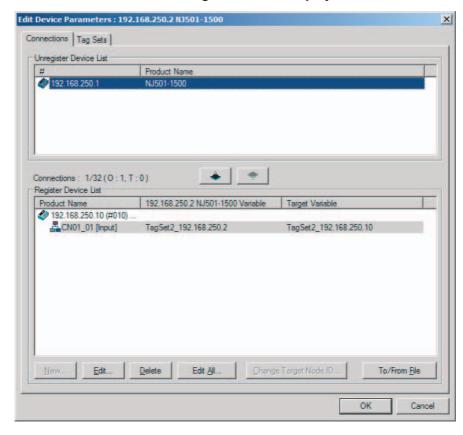
(3) Input a new RPI value, and click the OK Button.

#### Method 3:

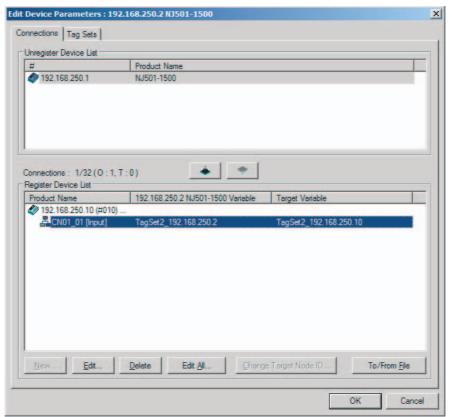
Changing a Particular Connection's Packet Interval (RPI) Setting:

The usage of capacity without multicast filtering can be adjusted against the tag data link's allowable bandwidth by individually changing the packet interval (RPI) for a particular connection. In this case, the usage of capacity will also change for target device of the connection for which the packet interval is changed.

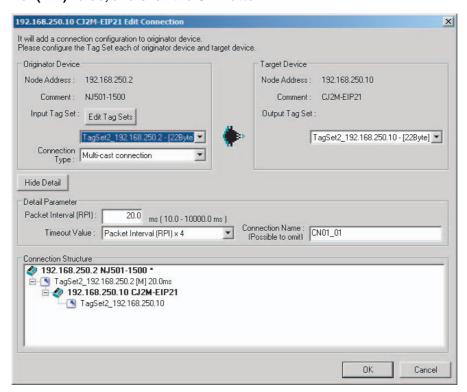
- (1) Click the Close Button at the bottom of the Usage of Device Bandwidth Dialog Box.
- (2) Double-click the device that is set as the originator of the desired connection. The Edit Device Parameters Dialog Box will be displayed.



(3) In the Register Device List, select the connection for which you want to change the RPI, and click the Edit Button.



(4) The device's Edit Connection Dialog Box will be displayed. Input a new packet interval (RPI) value, and click the OK Button.



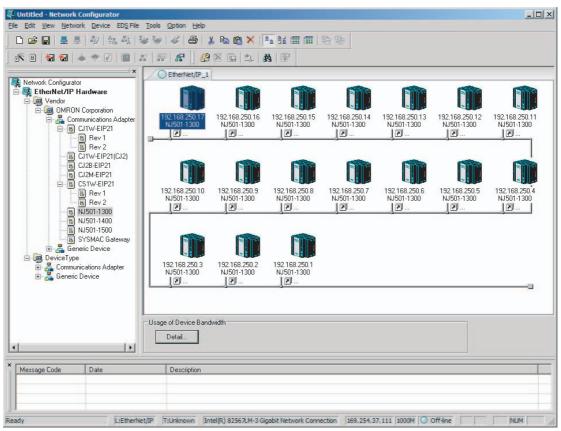
- 4 If the usage of capacity cannot be adjusted to the desired level when the setting described above has been performed, reconsider the network configuration considering the following points. Refer to 14-2-3 Adjusting Device Bandwidth Usage.
  - Reduce the number of nodes and number of connections.
  - · Split the network.
- Check the bandwidth usage again.

  If you have changed the connection settings, click the **Detail** Button in the Usage of Device Bandwidth Area at the bottom of the Network Configuration Window and check bandwidth usage according to the instructions in 14-2-1 Checking Bandwidth Usage for Tag Data Links. It is particularly important to check the usage of capacity when an individual connection's RPI setting was changed without using the **Set Packet Interval (RPI)** Button.
- 6 Run user tests to verify that there are no problems with the new settings.

# 14-2-5 RPI Setting Examples

The following examples explain how to calculate the packet intervals (RPIs) in the following network configuration.

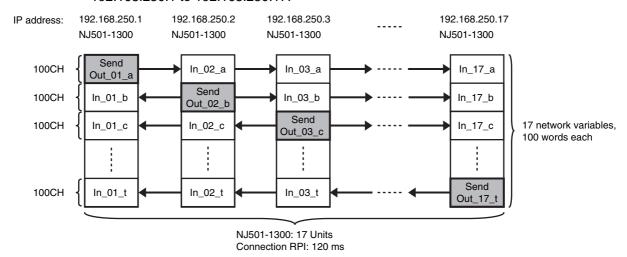
# **Conditions**



#### Connections:

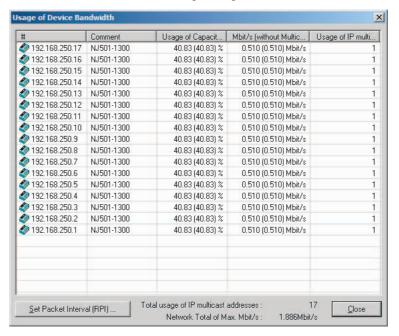
Example: Seventeen NJ501-1300 Units are connected to the network.

Each device has one 100-word tag for transmission and sixteen 100-word tags for reception, so that the Units exchange data mutually. By default, the packet intervals (RPIs) are set to 120 ms for all of the connections. The devices' IP addresses range from 192.168.250.1 to 192.168.250.17.



# **Checking the Device Bandwidth Usage**

When the **Detail** Button is clicked in the Usage of Device Bandwidth Area, it is apparent that the percentage of the allowed tag data link bandwidth being used by each device's tag data link (Usage of Capacity) is 40.83%, as shown in the following dialog box.

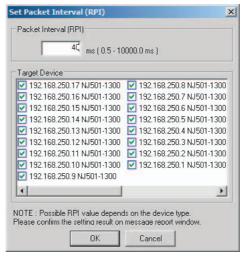


# **Changing Settings**

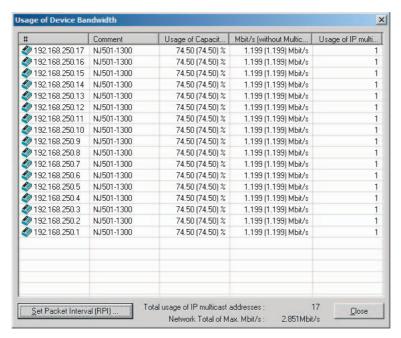
## Method 1: Same Packet Interval Setting for All Connections

The percentage of the allowed tag data link bandwidth being used (Usage of Capacity) was 40.83% with the RPI set to 120 ms for all of the connections, so the RPI will be set to 40 ms, with a target of 80% or less of the allowable bandwidth.

Click the **Set Packet Interval (RPI)** Button at the bottom of the Usage of Device Bandwidth Dialog Box to display the Set Packet Interval (RPI) Dialog Box. Input 40 ms as the new RPI value, then click the **OK** Button.



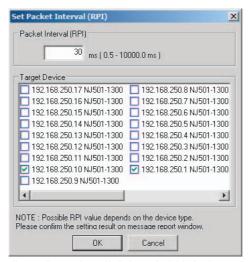
If the packet interval for all connections has been set to the same setting, the dialog box will show that the usage of capacity for the tag data link's allowable communications bandwidth is 74.50% and the fastest set value is 40 ms.



Method 2: Changing the Packet Interval (RPI) of Only Specific Devices

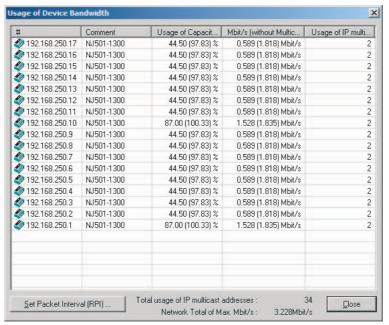
In this example, we want faster tag data links for devices 192.168.250.1 and 192.168.250.10 only. To do this, click the Set Packet Interval (RPI) Button at the bottom of the Usage of Device Bandwidth Dialog Box. The Set Packet Interval (RPI) Dialog Box is displayed.

In the Target Device Area, clear the selections of all devices other than 192.168.250.1 and 192.168.250.10. Input 30 ms as the new RPI value, then click the **OK** Button.



The percentage of the allowed tag data link bandwidth being used (Usage of Capacity) increases to 87.00% for devices 192.168.250.1 and 192.168.250.10, which indicates that the RPI is set to a higher speed for these devices' connections.

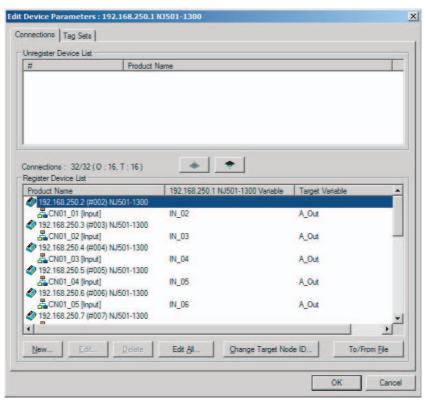
The Usage of Capacity values also indicate that the Usage of Capacity has increased (from 40.83% to 44.67%) for all of the other devices, which connect with devices 192.168.250.1 and 192.168.250.10.



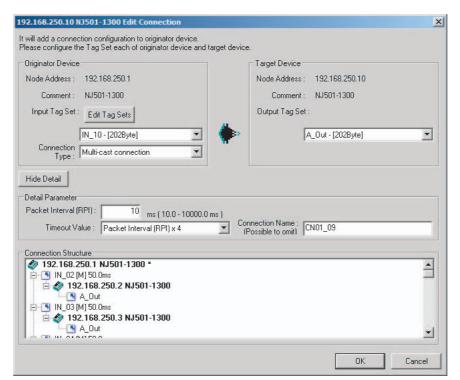
In this case, if there is no multicast filter, the value becomes 100.33%. If there is no multicast filter for an Ethernet switch, communications errors may occur depending on the communications load of the built-in EtherNet/ IP Unit port.

## Method 3: Changing the Packet Intervals (RPIs) of Only Specific Connections

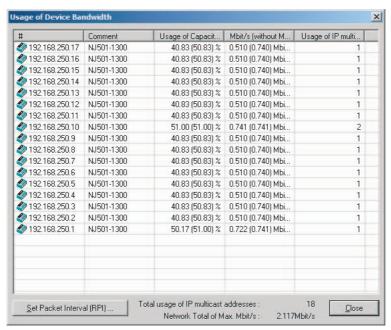
In this example, we want a faster tag data links for just a particular connection of device 192.168.250.1. Double-click device 192.168.250.1 in the Network Configuration Window.



Information about the connection with device 192.168.250.10 is registered in the Register Device List. Double-click this connection to edit the settings.



In the Edit Connection Dialog Box, input 10 ms as the new RPI value, and click the **OK** Button. The tag data link bandwidth used by device 192.168.250.1 (Usage of Capacity) increases to 50.17%, which indicates that a RPI is set to a higher speed for this device.



In this case, the tag data link bandwidth that is used by device 192.168.250.10 (Usage of Capacity) also increases (from 40.83% to 51.00%).

# 14-3 I/O Response Time in Tag Data Links

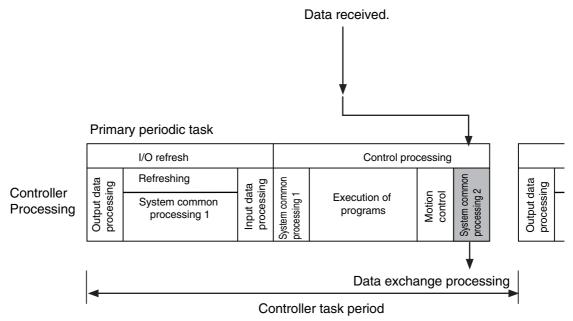


#### **Additional Information**

This section describes the response time of the NX- and NJ-series CPU Unit built-in Ethernet/IP ports. The data processing times for the EtherNet/IP Unit, the built-in EtherNet/IP port on the CJ2H-CPU6□-EIP CPU Unit and the built-in EtherNet/IP port on the CJ2M-CPU3□ CPU Unit are different. For details, refer to *7-4 Tag Data Links with Other Models*.

# 14-3-1 Timing of Data Transmissions

The following diagram shows the timing of tag data link transfers between the built-in EtherNet/IP port and the CPU Unit. Data is transferred when system common processing 2 is performed for the task set as the refreshing task.



You can set either of the following types of tasks as the refreshing task.

- Primary periodic task
   The primary periodic task has the highest execution priority. It executes processes with high speed and high precision.
- Periodic tasks
   Periodic tasks are executed during the time between executions of the primary periodic task.

When using an NJ-series CPU Unit, you do not need to specify a refreshing task for tags that use an AT specification. Data is transferred for these tags during the primary periodic task. The task during which to perform tag data link processing is specified for each tag. Set the refreshing task on the Sysmac Studio for each variable you want to set as a tag. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for details on setting refreshing tasks.

# 14-3-2 Built-in EtherNet/IP Port Data Processing Time

This section describes the data processing time required to transfer data between the built-in Ether-Net/IP port and the CPU Unit.

# **Data Processing Time Overview**

The time required for data processing consists of the following three elements.

#### 1. Variable Access Time

First, calculate the time required to transfer tag data (or the time required to access variables). This calculation is performed for each task. Therefore, if the same refreshing task is set for multiple tag sets, calculate the total time required for all tags in the tag sets.

Use the following equation to calculate the variable access time.

Variable access time [ $\mu$ s] = Total size of variables [bytes]  $\times$  a + Number of variables  $\times$  b + Number of accesses × c + d

Number of accesses: Number of tag sets a to d: Constant values as given below

CPU Unit model		Constant	value [µs]	
	а	b	С	d
NX701-□□□□	0.0005	0.033	2.67	7.22
NJ501-□□□□	0.001	0.49*1	1.41	6.68
NJ301-□□□□	0.0015 <sup>*2</sup>	0.56 <sup>*3</sup>	2.15	7.52
NJ101-□□□□	0.0015	0.56	3.83	7.52

<sup>\*1</sup> The value is 0.58 for a CPU Unit with unit version 1.02 or earlier.

#### 2. Number of Data Transfers

Tag data is transferred as a part of task processing.

If the time required to process the data transfer is greater than the "variable access time" \*2, the data cannot be sent entirely in one task period and is sent separately instead.

Number of data transfers = "Time required to send the data entirely" \*1

- ÷ "Variable access time" \*2 set for the task
- \*1 This is the variable access time as calculated in step 1 above.
- \*2 The "variable access time" refers to the maximum processing time for accessing variables. Set the time for each task on the Task Setup Display, which is displayed by selecting Configurations and Setup - Task Setup in the Sysmac Studio.



#### **Precautions for Correct Use**

The maximum number of tag data link words that can be transferred through the built-in Ether-Net/IP port is 184,832 words on an NX-series CPU Unit (total of 369,664 words with two ports), or 9,600 words on an NJ-series CPU Unit. If the number of tag data link words exceeds the number of words that can be exchanged with the CPU Unit at one time, the data is divided and transferred in multiple data exchanges.

<sup>\*2</sup> The value is 0.0009 for a CPU Unit with unit version 1.02 or earlier.

<sup>\*3</sup> The value is 1.03 for a CPU Unit with unit version 1.02 or earlier.

#### 3. Actual Time Required for Data Transfer

You can use the task period of the refreshing task and the number of data transfers as calculated in (2) above to calculate the actual time required to transfer the data.

Task period × Number of data transfers

# **Data Processing Time Calculation Example**

Here we provide an example of how to perform the tag data link calculations described earlier for the following tag data transfers.

- Model Numbers of CPU Units for Tag Data Links NJ501-□□□□
- Refreshing task
   Primary periodic task

Task period: 500 µs (variable access time: 3%)

Setting Tag Sets

Tag set	Refreshing task	Number of variables	Total size of variables
Tag set A	Primary periodic task	8	600 bytes
Tag set B	Primary periodic task	4	200 bytes
Tag set C	Primary periodic task	10	1,000 bytes

Calculate the variable access time as shown below.

[(600 + 200 + 1,000) bytes  $\times$  0.001  $\mu$ s] + [(8 + 4 + 10) variables  $\times$  0.49  $\mu$ s] + 3  $\times$  1.41  $\mu$ s + 6.68  $\mu$ s = 23.49  $\mu$ s

**2** Calculate the number of data transfers.

Time required for data transfer: "Variable access time" in step 1 = 23.49  $\mu$ s

Variable access time set for the task:  $500 \mu s \times 0.03 = 15 \mu s$ 

Number of data transfers:  $23.49 \,\mu s \div 15 \,\mu s = 1.6 \,times$ 

Thus, approximately two data transfers are required.

**3** Calculate the actual time required for the data transfer.

 $500 \mu s \times 2 \text{ times} = 1,000 \mu s$ 

# Relationship between Task Periods and Packet Intervals (RPIs)

# **Effect of Tag Data Links on Task Periods**

The tag data is transferred during task processing. Therefore, the tag data transfer process is added to the task processing for tasks set as a tag's refreshing task. This requires you to make adjustments to the variable access time and task period in the Task Setup so that these processes are completed within a single task period.

- Calculate the time required for the data transfer and set the result as the "variable access time". For the formula for calculating the time required for the data transfer, refer to Data Processing Time Calculation Example.
  - \* If the same refreshing task is set for multiple tag sets, calculate the total time required for all tags in tag
- Set the variable access time in the Task Setup to a value equal to or greater than the value calculated in step 1 above.

Adjust the task period time after adding in the time calculated in step 1. Use the Sysmac Studio to set the variable access time and task period settings. For details, refer to the NJ/NX-series CPU Unit Software User's Manual (Cat. No. W501).

# Adjusting Packet Intervals (RPIs) According to the Task Periods

Tag data is transferred based on the actual time required for the transfer (task period x number of data transfers), regardless of the packet interval (RPI) setting. Therefore, set the packet interval (RPI) according to the following guideline.

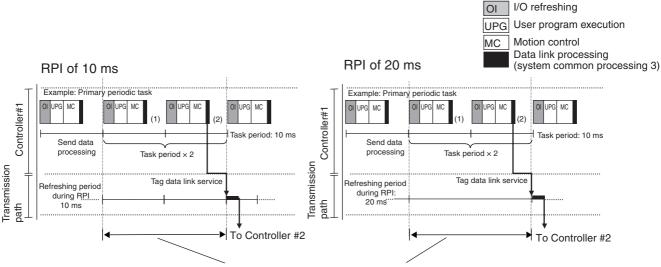
Actual time required for data transfer (Task period × Number of data transfers) < RPI

For details on the actual time required to transfer data, refer to 14-3-2 Built-in EtherNet/IP Port Data Processing Time.

Example: Relationship between the RPI Setting and the Time Required for Data Transfer

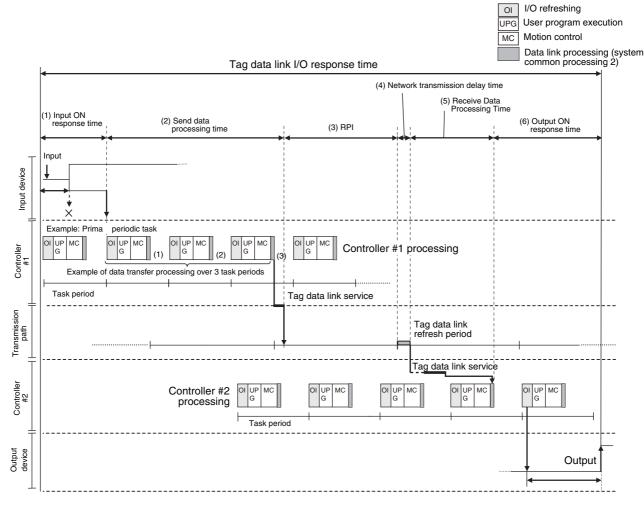
- Task period: 10 ms
- Number of data transfers: 2
- Actual time required for data transfer: 10 ms × 2 times = 20 ms

Regardless of the RPI value, the time required for the data transfer is 20 ms.



# 14-3-4 Maximum Tag Data Link I/O Response Time

You can find the maximum I/O response time from the total of (1) to (6) in the following figure.





#### **Additional Information**

- With unit version 1.03 or later, processing for tag data links is executed in the tag data link service.
- With unit version 1.00 to 1.02, processing for tag data links is executed in the system services.
   If a tag data link timeout occurs, reconsider the execution time for system services.

### (1) Input ON Response Time

This is the delay time for the external input device from when the input occurs until the switch actually changes to ON and the time until the input data is stored in the memory area of the CPU Unit. Refer to the input delay of each device for the input switch delay time. Also, one task period is required until the data is stored in the memory area of the CPU Unit. Therefore, the input ON response time is obtained as shown below.

Input ON response time = Input device delay time + Task period

#### (2) Send Data Processing Time

This is the time until the variables in the CPU Unit are transferred to the built-in EtherNet/IP port. Data is transferred during task processing. Therefore, the time required for send data processing is the same as the task period. If the data that is transferred is larger than the amount of data that can be sent during a single task (as set in the variable access time for the task), the data is transferred over multiple task periods. Therefore, add (task period x the number of transfers) to the time required. For details on how to determine the time required to send data, refer to 14-3-2 Built-in EtherNet/IP Port Data Processing Time.

#### (3) Packet Interval (RPI)

This is the communications refresh period set for each connection using the Network Configurator.

#### (4) Network Transmission Delay Time

The transmission delay on an Ethernet line is 50 µs or less. This delay time can be ignored.

## (5) Receive Data Processing Time

This is the time required to transfer data received on the built-in EtherNet/IP port to a variable in the CPU Unit. Data is received during task processing. Therefore, the time required for receive data processing is the same as the task period. If the data that is transferred is larger than the amount of data that can be received during a single task (as set in the variable access time for the task), the data is transferred over multiple task periods. Therefore, add (task period × the number of transfers) to the time required. For details on how to determine the time required to receive data, refer to 14-3-2 Built-in EtherNet/IP Port Data Processing Time. Data is transferred once in each task period. Therefore, if data transfer has ended in the task period in which data is received, the start of transmission for received data will be delayed by one Controller task period.



#### **Additional Information**

The total amount of data transferred increases if there are connections with multiple nodes, and the data that is transferred may exceed the amount that can be processed in a single transfer. In this case, the number of data transfers increases.

#### (6) Output ON Response Time

This is the delay time for the external output device from when the Controller specified turning ON the output until the output is actually turned ON.

Output ON response time = Output device delay time + Task period

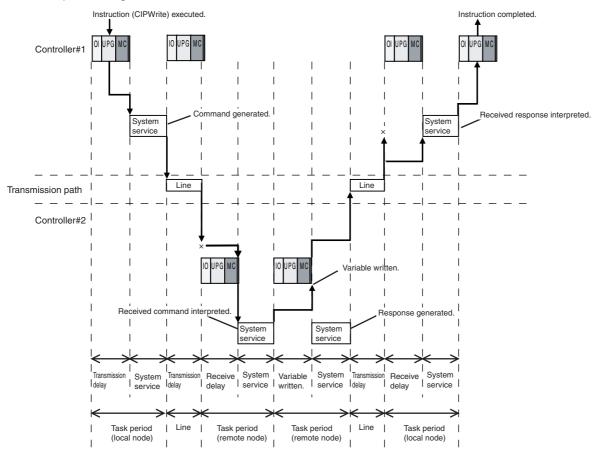


#### **Additional Information**

The I/O response time may be longer due to noise, or other events.

# 14-4 Message Service Transmission Delay

This section describes the delay time that occurs in CIP communications instruction (CIPWrite) service processing.



Maximum transmission delay time = Send delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission delay + Receive delay + System service execution time + Transmission time + Tr

+ Variable write time + System service execution time + Transmission delay + Receive delay + System service execution time

(Remote node task period)

Processes that cause a delay time are processed in the task periods at each node as shown in the above diagram. Line-based delays are as follows:

#### Transmission Delay

The transmission delay on an Ethernet line is 50 µs or less. This delay time can be ignored.



# **Additional Information**

- Depending on the actual operating environment, the transmission time may be longer than the one calculated with the equations given here. The following factors can cause longer transmission times: other traffic on the network, window sizes of network nodes, other traffic at the built-in EtherNet/ IP port itself (e.g., simultaneous tag data link communications), and the system configuration.
- CIP communications processing is executed as a system service. If a timeout occurs for a CIP communications instruction, reconsider the execution time for system services.



# **Troubleshooting**

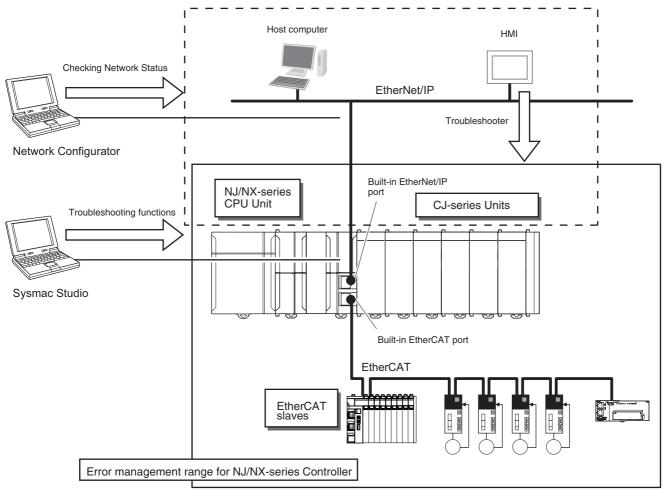
This section describes the items to check when errors occur in the EtherNet/IP Function Module. It includes error diagnosis and countermeasures for error indications, and error diagnosis and countermeasures for operating conditions.

15-1	Overvie	ew of Errors	15-2
	15-1-1	How to Check for Errors	15-3
	15-1-2	Errors Related to the EtherNet/IP Function Module	15-8
15-2	Trouble	eshooting	5-12
	15-2-1	Error Table	15-12
	15-2-2	Error Descriptions	15-18
	15-2-3	Troubleshooting	15-50
15-3	Checki	ng Status with the Network Configurator	5-51
	15-3-1	The Network Configurator's Device Monitor Function	15-51
	15-3-2	Connection Status Codes and Troubleshooting	15-58

# **Overview of Errors**

You manage all of the errors that occur on the NJ/NX-series Controller as events. This allows you to see what errors have occurred and find corrections for them with the same methods for the entire range of errors that is managed (i.e., CPU Unit, NX-series Slave Terminals, EtherCAT slaves,\* and CJ-series

\* Only Sysmac devices are supported.



You can use the troubleshooting functions of the Sysmac Studio or the Troubleshooter on an HMI to quickly check for errors that have occurred and find corrections for them.

To perform troubleshooting from an HMI, connect the HMI to the built-in EtherNet/IP port on the CPU Unit.

You can also use the Network Configurator to check the network status. Refer to 15-3 Checking Status with the Network Configurator for the procedure to check network status.

This manual describes the errors that originate in the EtherNet/IP Function Module. Refer to the NJ/NXseries Troubleshooting Manual (Cat. No. W503) for specific corrections when errors occur and for troubleshooting information on the entire NJ/NX-series Controller.



#### **Precautions for Correct Use**

- · CJ-series Units can be used only with an NJ-series CPU Unit.
- Refer to the appendices of the NJ/NX-series Troubleshooting Manual (Cat. No. W503) for the applicable range of the HMI Troubleshooter.

#### 15-1-1 How to Check for Errors

You can check to see if an error has occurred with the following methods.

Checking method	What you can check
Checking the indicators	You can check the error status of the CPU Unit and EtherNet/IP port.
Checking with the trouble- shooting function of Sysmac Studio	You can check for current Controller errors, a log of past Controller errors, error sources, error causes, and corrections.
Checking with the Trouble- shooter of an HMI*	You can check for current Controller errors, a log of past Controller errors, error sources, error causes, and corrections.
Checking with instructions that read function module error status	You can check the highest-level status and highest-level event code in the current Controller errors.
Checking with system-defined variables	You can check the current Controller error status for each function module.
Checking communications status with the Network Configurator	You can check the communications status (e.g., tag data link connection status) for each device on the EtherNet/IP network.

<sup>\*</sup> To perform troubleshooting from an HMI, connect the HMI to the built-in EtherNet/IP port on the CPU Unit. Refer to the appendices of the *NJ/NX-series Troubleshooting Manual* (Cat. No. W503) for the applicable range of the HMI Troubleshooter.

This section describes the above checking methods.

# **Checking the Indicators**

#### CPU Unit Operating Status

You can use the PWR indicator on the Power Supply Unit and the RUN and ERROR indicators on the CPU Unit to determine the event level for an error. The following table shows the relationship between the Controller's indicators and the event level.

The following table shows the relationship between the CPU Unit indicators and the event level for an NX-series CPU Unit.

Indicators			CDI Unit energing status	Error confirmation with the Sysmac Stu-	
PWR	RUN	ERROR	CPU Unit operating status	dio or an HMI	
Not lit	Not lit	Not lit	Power Supply Error	Not possible:	
Lit	Not lit	Not lit	CPU Unit Reset*1	Refer to the <i>NJ/NX-series Troubleshooting Manual</i> (Cat. No. W503).	
Lit	Not lit or flashing	Lit	CPU Unit Error*2	Manual (Sat. 140. 44000).	
Lit	Flashing for 30 s or longer	Not lit	System Initialization Error		
Lit	Not lit	Lit	Major fault level*2	Possible:	
Lit	Lit	Flashing	Partial fault level	Connect the Sysmac Studio or an HMI and check the cause of and correction for the error in the troubleshooting functions of the	
Lit	Lit	Flashing	Minor fault level		
Lit	Lit	Not lit	Observation	Sysmac Studio or the Troubleshooter of the HMI.	
Lit	Lit	Not lit	Normal operation in RUN mode		
Lit	Not lit	Not lit	Normal operation in PRO- GRAM mode*1		

Indicators			- CPU Unit operating status	Error confirmation with the Sysmac Stu-
PWR	RUN	ERROR	CPO Offic operating status	dio or an HMI
Lit	Flashing	Not lit	Normal operation in startup state	

- \*1 If you can connect communications to the CPU Unit from the Sysmac Studio with a direct USB connection, the CPU Unit is in PROGRAM mode. If you cannot connect communications, the CPU Unit is being reset.\*3
- \*2 If you can connect communications to the CPU Unit from the Sysmac Studio with a direct USB connection, a major fault level error has occurred. If you cannot connect communications, a CPU Unit Error has occurred.\*3
- \*3 If you cannot connect communications to the CPU Unit from the Sysmac Studio, it is also possible that the USB cable is faulty or that the network type on the Sysmac Studio is not set for a direct USB connection. Refer to the NJ/NX-series Troubleshooting Manual (Cat. No. W503) if you cannot connect communications to the CPU

The following table shows the relationship between the CPU Unit indicators and the event level for an NJ-series CPU Unit.

Indicators		;	CRIL Unit operating status	Error confirmation with the Sys-	
PWR	RUN	ERROR	CPU Unit operating status	mac Studio or an HMI	
Not lit	Not lit	Not lit	Power Supply Error	Not possible: Refer to the NJ/NX-	
Lit	Not lit	Not lit	CPU Unit Reset*1	series Troubleshooting Manual (Cat. No. W503).	
Lit	Flashing	Lit	Incorrect Power Supply Unit Connected		
Lit	Not lit	Lit	CPU Unit Watchdog Timer Error*2		
Lit	Not lit	Lit	Major fault level*2	Possible: Connect the Sysmac Stu-	
Lit	Lit	Flashing	Partial fault level	dio or an HMI and check the cause of and correction for the error in the troubleshooting functions of the	
Lit	Lit	Flashing	Minor fault level		
Lit	Lit	Not lit	Observation	Sysmac Studio or the Trouble- shooter of the HMI.	
Lit	Lit	Not lit	Normal operation in RUN mode		
Lit	Not lit	Not lit	Normal operation in PROGRAM mode*1		
Lit	Flashing	Not lit	Normal operation in startup state		

- \*1 If you can connect communications to the CPU Unit from the Sysmac Studio with a direct USB connection, the CPU Unit is in PROGRAM mode. If you cannot connect communications, the CPU Unit is being reset.\*3
- \*2 If you can connect communications to the CPU Unit from the Sysmac Studio with a direct USB connection, a major fault level error has occurred. If you cannot connect communications, a watchdog timer error has occurred in the CPU Unit.\*3
- \*3 If you cannot connect communications to the CPU Unit from the Sysmac Studio, it is also possible that the USB cable is faulty or that the network type on the Sysmac Studio is not set for a direct USB connection. Refer to the NJ/NX-series Troubleshooting Manual (Cat. No. W503) if you cannot connect communications to the CPU Unit.

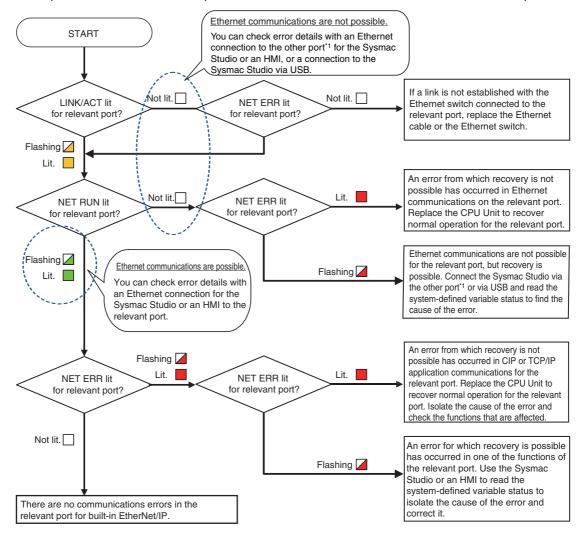
#### EtherNet/IP Port Error Status

If the EtherNet/IP NET ERR indicator is also lit in addition to the PWR, RUN, and ERROR indicators, then an error in the minor fault level or a higher level has occurred in the EtherNet/IP port. The indicator lets you check the status given in the following table.

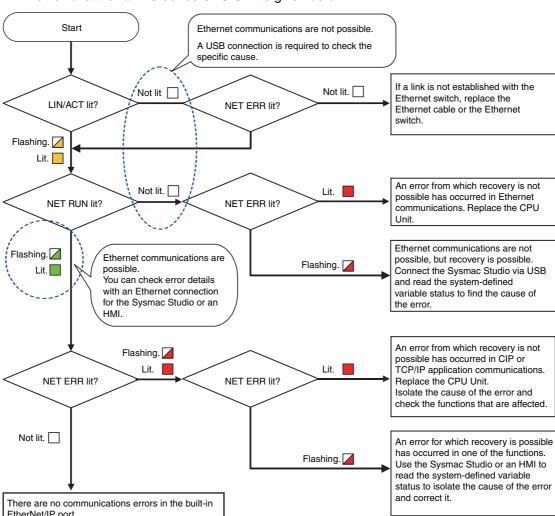
Indicators	Indicated status	
EtherNet/IP NET ERR	EtherNet/IP Port Status	
	• Lit: An error for which normal status cannot be recovered through user actions (i.e., errors for which you must replace the CPU Unit or contact your OMRON representative) has occurred.	
	Flashing: An error for which normal status can be recovered through user actions has occurred.	
	Not lit: There is no minor fault level or higher-level error.	

#### Flowchart to Check Error Status on EtherNet/IP Function Module with Indicators

The flowchart for an NX-series CPU Unit is given below. To differentiate between the two communications ports in the flowchart, the port for which to isolate the error is called the relevant port.



<sup>\*1.</sup> You can connect the Sysmac Studio or an HMI to the other port to confirm error details only when Ethernet communications are possible for the other port.



#### The flowchart for an NJ-series CPU Unit is given below.

## Checking with the Troubleshooting Function of Sysmac Studio

When an error occurs, you can connect the Sysmac Studio online to the Controller to check current Controller errors and the log of past Controller errors. You can also check the cause of the error and corrections.

Refer to the NJ/NX-series Troubleshooting Manual (Cat. No. W503) for the procedures to check for errors with the Sysmac Studio.

## Checking with the Troubleshooter of an HMI

If you can connect communications between an HMI and the Controller when an error occurs, you can check for current Controller errors and the log of past Controller errors. You can also check the cause of the error and corrections.

To perform troubleshooting from an HMI, connect the HMI to the built-in EtherNet/IP port on the CPU

Refer to the NJ/NX-series Troubleshooting Manual (Cat. No. W503) for the procedures to check for errors with an HMI.

# **Checking with Instructions That Read Error Status**

You can use instructions in the user program to check the error status of each function module. The following table gives the instruction that is used to get error information for the EtherNet/IP Function Module.

Instruction	Name	Outline of function				
GetEIPError		The GetEIPError instruction gets the highest level status (partial fault or minor fault) and highest level event code of the current Controller errors in the EtherNet/IP Function Module.				

For details on the instructions that get error status, refer to the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

## **Checking with System-defined Variables**

You can use system-defined variables to check for errors in the EtherNet/IP Master Function Module.

#### Error Status Variables

You can check for errors in each function module of the NJ/NX-series Controller with error status variables. The following variables show the error status of the EtherNet/IP Function Module.

Variable name	Data type	Meaning	Function
_EIP_ErrSta	WORD	Built-in EtherNet/IP Error	Gives the collective error status of all error status for the EtherNet/IP Error Function Module.
_EIP_PortErr	WORD	Communications Port Error	Gives the collective error status of all error status for the EtherNet/IP communications port on an NJ-series CPU Unit. For an NX-series CPU Unit, this variable gives the same status as the Communications Port 1 Error variable and you can use it instead of the Communications Port 1 Error variable.
_EIP1_PortErr	WORD	Communications Port 1 Error	Gives the collective error status of all error status for the EtherNet/IP communications port 1 on an NX-series CPU Unit.
_EIP2_PortErr	WORD	Communications Port 2 Error	Gives the collective error status of all error status for the EtherNet/IP communications port 2 on an NX-series CPU Unit.
_EIP_CipErr	WORD	CIP Communications Error	Gives the collective error status of all error status for CIP communications on an NJ-series CPU Unit. For an NX-series CPU Unit, this variable gives the same status as the CIP Communications 1 Error variable and you can use it instead of the CIP Communications 1 Error variable.
_EIP1_CipErr	WORD	CIP Communications 1 Error	Gives the collective error status for CIP communications 1 for an NX-series CPU Unit.
_EIP2_CipErr	WORD	CIP Communications 2 Error	Gives the collective error status for CIP communications 2 for an NX-series CPU Unit.
_EIP_TcpAppErr	WORD	TCP Application Communications Error	Gives the collective error status for TCP applications communications.

The meanings of the individual bits in the above error status variables are given below.

Bit	Name	Description	Value	Meaning
15	Master Detection*	This bit indicates whether the master	TRUE	Error
		detected an error in the slaves that it manages.	FALSE	No error
14	Slave Summary*	Indicates whether there is an error at	TRUE	Error
		a level below the function module.	FALSE	No error
8 to 13	Not used.			
7	Major Fault	Indicates if there is a major fault level error.	TRUE	Error
			FALSE	No error
6	Partial Fault Indicates if there is a partial fault level		TRUE	Error
		error.	FALSE	No error
5	Minor Fault	Indicates if there is a minor fault level	TRUE	Error
		error.	FALSE	No error
4	Observation	Indicates if there is an observation	TRUE	Error
		level error.	FALSE	No error
0 to 3	Not used.			

<sup>\*</sup> This bit is not used in the error status variables for the EtherNet/IP Function Module.

#### Other System-defined Variables Related to Errors

For the EtherNet/IP Function Module, there are other system-defined variables that show error status in addition to the error status variables. Refer to 3-1 System-defined Variables Related to the Built-in EtherNet/IP Port for system-defined variables that are related to errors.

## **Checking Communications Status with the Network Configurator**

You can use the Network Configurator to check the communications status (e.g., tag data link connection status) for each device on the EtherNet/IP network. Refer to 15-3 Checking Status with the Network Configurator.

#### 15-1-2 Errors Related to the EtherNet/IP Function Module

This section describes the errors that are related to the EtherNet/IP Function Module.

## **Classifications**

There are the following sources of errors in the EtherNet/IP Master Function Module.

#### NX-series CPU Units

Classification	Description		
Communications port 1 errors	If an error is detected for EtherNet/IP communications port 1, the corresponding bit in the Communications Port 1 Error status variable shows the error.		
Communications port 2 errors	If an error is detected for EtherNet/IP communications port 2, the corresponding bit in the Communications Port 2 Error status variable shows the error.		

Classification	Description			
CIP communications 1 errors	If an error that is related to the tag data links or CIP message communications is detected for EtherNet/IP communications port 1, the corresponding bit in the CIP Error Communications Error status variable shows the error.			
CIP communications 2 errors	If an error that is related to the tag data links or CIP message communications is detected for EtherNet/IP communications port 2, the corresponding bit in the CIP Error Communications Error status variable shows the error.			
TCP application errors	If an error that is related to the FTP server, NTP, or SMNP client is detected, the corresponding bit in the TCP Application Communications Error status variable shows the error.			

<sup>\*</sup> Other Ethernet communications are not affected.

#### NJ-series CPU Units

Classification	Description			
Communications port errors	If an error is detected in overall Ethernet communications, the corresponding bit in the Communications Port Error status variable shows the error.			
CIP communications errors	If an error that is related to the tag data links or CIP message communications is detected, the corresponding bit in the CIP Error Communications Error status variable shows the error.*			
TCP application errors	If an error that is related to the FTP server, NTP, or SMNP client is detected, the corresponding bit in the TCP Application Communications Error status variable shows the error.			

<sup>\*</sup> Other Ethernet communications are not affected.

# **Event Levels**

This section describes the operation of the EtherNet/IP Function Module for each event level.

Event level of the error	Operation
Major fault	All NJ/NX-series Controller control operations stop for errors in this event level.  A major fault level error never occurs for the EtherNet/IP Function Module.
Partial fault	All control operations for one of the function modules in the NJ/NX-series Controller stop for errors in this event level. If a partial fault level error occurs in the EtherNet/IP Function Module, all functions of the EtherNet/IP Function Module, such as axis operation, stop.
Minor fault	Some of the control operations for one of the function modules in the NJ/NX-series Controller stop for errors in this event level. If a minor fault level error occurs for the EtherNet/IP Function Module, EtherNet/IP communications are possible, but control may be affected. User action is required.
Observation	Errors in the observation level do not affect NJ/NX-series Controller control operations. Observations are reported in order to prevent them from developing into errors at the minor fault level or higher.
Information	Events that are classified as information provide information that do not indicate errors.

# **Event Levels for Built-in EtherNet/IP Errors**

The event levels for errors that are related to built-in EtherNet/IP are given in the following table.

#### NX-series CPU Units

Source Level	Communications port	CIP communications	TCP application		
Major fault	None	None	None		
Partial fault	EtherNet/IP Processing Error	None	None		
Minor fault	Communications Controller Failure     MAC Address Error     IP Route Table Setting Error     Basic Ethernet Setting Error     IP Address Setting Error     DNS Setting Error     DNS Server Connection Error     IP Address Duplication Error     BOOTP Server Connection Error	<ul> <li>Identity Error</li> <li>Tag Data Link Setting Error</li> <li>Tag Name Resolution Error</li> <li>Controller Insufficient Memory Warning</li> <li>Tag Data Link Connection Failed</li> <li>Tag Data Link Timeout</li> <li>Tag Data Link Connection Timeout</li> <li>Tag Data Link Equipment Total Allowable Bandwidth Exceeded</li> </ul>	FTP Server Setting Error     NTP Client Setting Error     SNMP Setting Error     NTP Server Connection Error		
Access Detected Outside Range of Variable     Packet Discarded Due to Full Reception Buffer     Link OFF Detected		None	None		
Information	<ul><li>Link Detected</li><li>Restarting Ethernet Port</li><li>IP Address Fixed</li><li>BOOTP Client Started</li></ul>	<ul> <li>Tag Data Link Download Started</li> <li>Tag Data Link Download Finished</li> <li>Tag Data Link Stopped</li> <li>Tag Data Link Started</li> <li>Tag Data Link All Run</li> </ul>	<ul><li>FTP Server Started</li><li>NTP Client Started</li><li>SNMP Started</li></ul>		

#### NJ-series CPU Units

Source Level	Communications port	CIP communications	TCP application		
Major fault	None	None	None		
Partial fault	<ul> <li>Communications Controller Failure</li> <li>MAC Address Error</li> <li>EtherNet/IP Processing Error</li> <li>Basic Ethernet Setting Error</li> <li>IP Address Setting Error</li> <li>IP Address Duplication Error</li> <li>BOOTP Server Connection Error</li> </ul>	• None	• None		
Minor fault	DNS Server Connection Error     IP Route Table Setting Error	<ul> <li>Identity Error</li> <li>Tag Data Link Setting Error</li> <li>Tag Name Resolution Error</li> <li>Controller Insufficient Memory Warning</li> <li>Tag Data Link Connection Failed</li> <li>Tag Data Link Timeout</li> <li>Tag Data Link Connection Timeout</li> <li>Tag Data Link Equipment Total Allowable Bandwidth Exceeded</li> </ul>	<ul> <li>FTP Server Setting Error</li> <li>NTP Client Setting Error</li> <li>SNMP Setting Error</li> <li>NTP Server Connection Error</li> </ul>		
Observation	<ul> <li>Access Detected Outside Range of Variable</li> <li>Packet Discarded Due to Full Reception Buffer</li> <li>Link OFF Detected</li> </ul>	None	None		
<ul> <li>Restarting Ethernet Port</li> <li>IP Address Fixed</li> <li>BOOTP Client Started</li> </ul>		<ul> <li>Tag Data Link Download Started</li> <li>Tag Data Link Download Finished</li> <li>Tag Data Link Stopped</li> <li>Tag Data Link Started</li> <li>Tag Data Link All Run</li> </ul>	<ul><li>FTP Server Started</li><li>NTP Client Started</li><li>SNMP Started</li></ul>		

You can change the event level for some events. Refer to *NJ/NX-series CPU Unit Software User's Manual* (Cat. No. W501) for information on changing event levels. Refer to *15-2 Troubleshooting* in this manual to see the events for which you can change the event level.

# 15-2 Troubleshooting

This section describes the errors that can occur and the corrections for them.

### 15-2-1 Error Table

The errors (i.e., events) that can occur in the EtherNet/IP Function Module are given on the following pages. The following abbreviations and symbols are used in the event level column.

Abbreviation	Name
Maj	Major fault level
Prt	Partial fault level
Min	Minor fault level
Obs	Observation
Info	Information

Symbol	Meaning
S	Event levels that are defined by the system.
U	Event levels that can be changed by the user.*

<sup>\*</sup> This symbol appears only for events for which the user can change the event level.

A version in parentheses in the Event code column is the unit version of the CPU Unit when the event occurs for only specific unit versions of the CPU Unit.

Refer to the NJ/NX-series Troubleshooting Manual (Cat. No. W503) for all of the event codes that may occur in an NJ/NX-series Controller.

Event code Event name			Assumed cause	Level				Deference	
Event code	Event name	Meaning	Assumed cause		Prt	Min	Obs	Info	Reference
04200000 hex	Communications Controller Failure	A hardware error was detected in the communications controller of the built-in EtherNet/IP port.	Communications Controller hardware error		S				page 15-19
14200000 hex	MAC Address Error	The MAC address in non-volatile memory was not read correctly.	Non-volatile memory failure		S				page 15-19
14220000 hex	EtherNet/IP Processing Error	A fatal error was detected in the Eth- erNet/IP Function Module.	Hardware has failed.		S				page 15-20
34210000 hex	Basic Ether- net Setting Error	An error was detected in the Ethernet settings.	Setting error     Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.     Memory error		S				page 15-20
34220000 hex	IP Address Setting Error	An error was detected in the IP address settings.	Setting error     Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.     Memory error     The IP address acquired from BOOTP server is illegal.		S				page 15-21

						Leve	I		Defe
Event code	Event name	Meaning	Assumed cause	Maj	Prt	Min	Obs	Info	Reference
84010000 hex	IP Address Duplication Error	The same IP address is used more than once.	The IP address of the built-in EtherNet/IP port is also used as the IP address of another node.		S				page 15-22
84020000 hex	BOOTP Server Con- nection Error	Connection with the BOOTP server failed.	Server setting error Server is down. An error occurred in the communications path.		S				page 15-23
0421 0000 hex (Ver. 1.10 or later)	Communica- tions Control- ler Failure	A hardware error was detected in the communications controller of the built-in EtherNet/IP port.	Hardware error in the communications controller			S			page 15-23
14210000 hex	Identity Error	The CIP identity information in non-volatile memory was not read correctly.	Non-volatile memory failure			S			page 15-24
14230000 hex (Ver. 1.10 or later)	MAC Address Error	The MAC address in non-volatile memory was not read correctly.	Non-volatile memory failure			S			page 15-24
34200000 hex	Tag Data Link Setting Error	An error was detected in the communications settings for tag data links.	Power was interrupted when a download was in progress for the data link settings.     Memory error			S			page 15-25
34230000 hex	IP Route Table Setting Error	An error was detected in the IP router tables or default gateway.	Setting error     Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.     Memory error			S			page 15-26
34240000 hex	FTP Server Setting Error	An error was detected in the FTP server settings.	Setting error     Power was interrupted when a download was in progress for the FTP server settings.     Memory error			S			page 15-27
3425 0000 hex	NTP Client Setting Error	An error was detected in the NTP client settings.	Setting error     Power was interrupted when a download was in progress for the NTP client settings.     Memory error			S			page 15-28
34260000 hex	SNMP Set- ting Error	An error was detected in the SNMP agent/trap settings.	Setting error     Power was interrupted when a download was in progress for the SNMP agent/trap settings.     Memory error			S			page 15-29

Event code	Event name	Meaning	Assumed cause			Leve	I		Reference
Event code	Event name	Meaning	Assumed cause	Maj	Prt	Min	Obs	Info	Reference
34270000 hex	Tag Name Resolution Error	Resolution of a tag used in a tag data link failed.	· 1			S			page 15-30
34280000 hex (Ver. 1.10 or later)	Basic Ether- net Setting Error	An error was detected in the Ethernet settings.	Parameter error Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings. A memory error occured.			S			page 15-31
3429 0000 hex (Ver. 1.10 or later)	IP Address Setting Error	An error was detected in the IP address settings.	<ul> <li>Parameter error</li> <li>Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.</li> <li>The IP address acquired from BOOTP server is illegal.</li> <li>A memory error occured.</li> </ul>			S			page 15-32
342A 0000 hex (Ver. 1.10 or later)	DNS Setting Error	An error was detected in the DNS settings or Hosts settings.	Parameter error  Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.  A memory error occured.			S			page 15-33
5001 0000 hex (Ver. 1.02 or later)	Controller Insufficient Memory Warning	The amount of data for the EtherCAT slave configuration, network-published information, or other data exceeds the value that is specified for the CPU Unit. You may not be able to perform online editing or other operations.	The amount of data for the EtherCAT slave configuration, network-published information, or other data exceeds the value that is specified for the CPU Unit.			S			page 15-34
84030000 hex	DNS Server Connection Error	Connection with the DNS server failed.	<ul> <li>Parameter error</li> <li>Server is down.</li> <li>An error occurred in the communications path.</li> </ul>			S			page 15-35
84040000 hex	NTP Server Connection Error	Connection with the NTP server failed.	<ul> <li>Parameter error</li> <li>Server is down.</li> <li>An error occurred in the communications path.</li> </ul>			S			page 15-35
84070000 hex	Tag Data Link Connection Failed	Establishing a tag data link connection failed.	The tag data link connection information is not the same for the originator and target. Insufficient connections			S			page 15-36

Frank and	Fromt	Maarring	A			Leve	ı		Deferre
Event code	Event name	Meaning	Assumed cause	Maj	Prt	Min	Obs	Info	Reference
84080000 hex	Tag Data Link Timeout	A timeout occurred in a tag data link.	The power supply to the target node is OFF.			S			page 15-37
			Communications with the target node stop.						
			The Ethernet cable for Ether- Net/IP is disconnected.						
			The Ethernet cable for Ether- Net/IP is broken.						
			Noise						
			The link to the built-in Ether- Net/IP port is OFF.						
84090000 hex (Ver. 1.04 or	Tag Data Link Connection	A timeout occurred while trying to	The power supply to the target node is OFF.			S	U		page 15-38
later)	Timeout	establish a tag data link connection.	Communications at the target node are stopped.						
			The Ethernet cable connector for EtherNet/IP is disconnected.						
			The Ethernet cable for Ether- Net/IP is broken.						
			An error occurred in the communications path.						
840A0000 hex (Ver. 1.10 or later)	IP Address Duplication Error	The same IP address is used more than once.	The IP address of the built-in EtherNet/IP port is also used as the IP address of another node.			S			page 15-39
840B0000 hex	воотр	Connection with the	Server setting error			S			page 15-40
(Ver. 1.10 or later)	Server Con- nection Error	BOOTP server failed.	The server is down.						
	Hodion End	lanou.	An error occurred in the communications path.						
840C 0000 hex (Ver. 1.10 or later)	Tag Data Link Equipment Total Allow- able Band- width Exceeded	The total band- width for the con- nections that are set or established exceeded the allow- able tag data link bandwidth for all of the EtherNet/IP ports in the equip- ment.	An attempt was made to establish a connection that would cause the PPS total of the packet transfer rates of the tag data links that use the Ether-Net/IP ports in the equipment to exceed the allowable bandwidth of the equipment.			S			page 15-41
54E00000 hex	Access Detected Outside Range of Variable	Accessing a value that is out of range was detected for a tag variable that is used in a tag data link.	<ul> <li>An out-of-range value was written by an EtherNet/IP tag data link for a variable with a specified range. A value that does not specify an enumerator was written by an EtherNet/IP tag data link for an enumeration variable.</li> </ul>				Ø		page 15-38
84050000 hex	Packet Discarded Due to Full Reception Buffer	A packet was discarded.	A network convergence occurred.				Ø		page 15-42

Event code	Event norm	Maanin	Accommoderate			Leve			Reference
Event code	Event name	Meaning	Assumed cause	Maj	Prt	Min	Obs	Info	Reference
84060000 hex	Link OFF Detected	An Ethernet Link OFF was detected.	An Ethernet cable is broken, disconnected, or loose. The Ethernet switch's power supply is turned OFF. Baud rate mismatch. Noise The Identity object was reset. Settings for EtherNet/IP were downloaded from the Network Configurator or Sysmac Studio, or the Clear All Memory operation was performed. EtherNet/IP was restarted.			U	S		page 15-43
9401 0000 hex	Tag Data Link Download Started	Changing the tag data link settings started.	Changing the tag data link set- tings started.					S	page 15-44
94020000 hex	Tag Data Link Download Finished	Changing the tag data link settings finished.	Changing the tag data link set- tings finished.					S	page 15-44
94030000 hex	Tag Data Link Stopped	Tag data links were stopped by the Network Configurator, Sysmac Studio, or manipulation of a system-defined variable. Or, the data link table was downloaded from the Network Configurator or Sysmac Studio again.	Tag data links were stopped by the Network Configurator, Sys- mac Studio, or manipulation of a system-defined variable.					S	page 15-45
94040000 hex	Tag Data Link Started	Tag data links were started by the Network Configurator, Sysmac Studio, or manipulation of a system-defined variable. Or, the data link table was downloaded from the Network Configurator or Sysmac Studio again.	Tag data links were started by the Network Configurator, Sys- mac Studio, or manipulation of a system-defined variable.					S	page 15-46
94050000 hex	Link Detected	Establishment of an Ethernet link was detected.	Establishment of an Ethernet link was detected.					S	page 15-46
94060000 hex	Restarting Ethernet Port	The built-in Ether- Net/IP port was restarted.	The built-in EtherNet/IP port was restarted.					S	page 15-47
94070000 hex	Tag Data Link All Run	Tag data link con- nections to all nodes have been established.	Tag data link connections to all target nodes have been estab- lished.					S	page 15-47
94080000 hex	IP Address Fixed	The correct IP address has been determined and Ethernet communications can start.	The correct IP address has been determined and Ethernet communications can start.					S	page 15-48
9409 0000 hex	BOOTP Cli- ent Started	The BOOTP client started requesting an IP address.	The BOOTP client started requesting an IP address.					S	page 15-48

Event code	Event name	Meaning	Assumed cause	Level					Reference
Event code	Event name	Meaning	Assumed cause		Prt	Min	Obs	Info	neierence
940A0000 hex	FTP Server Started	The FTP agent started normally.	The FTP agent started nor- mally.					S	page 15-49
940B0000 hex	NTP Client Started	The NTP client started normally and a request for the NTP server to obtain the time started.	The NTP client started normally and a request for the NTP server to obtain the time started.					S	page 15-49
940C0000 hex	SNMP Started	The SNMP agent started normally.	The SNMP agent started nor- mally.					S	page 15-49

#### 15-2-2 Error Descriptions

This section describes the information that is given for individual errors.

## **Controller Error Descriptions**

The items that are used to describe individual errors (events) are described in the following copy of an error table.

Event name	Gives the name o	Gives the name of the error.  Event code  Gives the code of the error.						
Meaning	Gives a short des	cription of the error.						
Source	Gives the source	of the error.	Source details	Gives details on the source of the error.	Detection timing	Tells when the error is detected.		
Error attributes	Level	Tells the level of influence on control.*1	Recovery	Gives the recovery method.*2	Log category  Tells which log the error is save in.*3			
Effects	User program	Tells what will hap- pen to execution of the user pro- gram.*4	Operation	Provides special ir from the error.	formation on the op	peration that results		
Indicators		of the built-in EtherNe rCAT Master Function	•	•		us is given only for		
System-defined	Variable		Data type		Name			
variables		names, data types, a ffected by the error, o	,		•	t error notification,		
Cause and cor-	Assumed cause		Correction		Prevention			
rection	Lists the possible causes, corrections, and preventive measures for the error.							
Attached information	This is the attache	This is the attached information that is displayed by the Sysmac Studio or an HMI.*5						
Precautions/ Remarks	Provides precautions, restrictions, and supplemental information. If the user can set the event level, the event levels that can be set, the recovery method, operational information, and other information is also provided.							

\*1 One of the following:

Major fault: Major fault level Partial fault: Partial fault level Minor fault: Minor fault level

Observation Information

\*2 One of the following:

Automatic recovery: Normal status is restored automatically when the cause of the error is removed.

Error reset: Normal status is restored when the error is reset after the cause of the error is removed.

Cycle the power supply: Normal status is restored when the power supply to the Controller is turned OFF and then back ON after the cause of the error is removed.

Controller reset: Normal status is restored when the Controller is reset after the cause of the error is removed.

Depends on cause: The recovery method depends on the cause of the error.

\*3 One of the following:

System: System event log Access: Access event log

\*4 One of the following:

Continues: Execution of the user program will continue.

Stops: Execution of the user program stops. Starts: Execution of the user program starts.

Refer to the appendices of the NJ/NX-series Troubleshooting Manual (Cat. No. W503) for the applicable range of the HMI Troubleshooter.

# **Error Descriptions**

Event name	Communications (	Controller Failure		Event code	0420 0000 hex			
Meaning	A hardware error	was detected in the c	communications cor	troller of the built-in I	EtherNet/IP port.			
Source	EtherNet/IP Funct	ion Module	Source details	Communications port	Detection timing	Continuously		
Error attributes	Level	Partial fault	Recovery	Cycle the power supply.	Log category System			
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not	operate.		
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINK/ACT			
	OFF		Lights.					
System-defined	Variable		Data type Name					
variables	_EIP_LanHwErr		BOOL		Communications Controller Error			
Cause and	Assumed cause		Correction		Prevention			
correction	Communications (	Controller hardware	Replace the CPU	Unit.	None			
Attached information	None							
Precautions/ Remarks		ter the _EIP_LanHwErr system-defined variable changes to TRUE, it will not change to FALSE unless the power sup- y to the Controller is cycled.						

Event name	MAC Address Err	ror		Event code	1420 0000 hex			
Meaning	The MAC address	s in non-volatile me	emory was not read c	orrectly.	•			
Source	EtherNet/IP Fund	tion Module	Source details	Communications port	Detection timing	At power ON or Controller reset		
Error attributes	Level	Partial fault	Recovery	Cycle the power supply.	Log category System			
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not	operate.		
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT			
	OFF		Lights.	Lights.				
System-defined	Variable		Data type		Name			
variables	_EIP_MacAdrErr		BOOL		MAC Address Er	ror		
Cause and	Assumed cause		Correction		Prevention			
correction	Non-volatile mem	ory failure	Replace the CPU	Unit.	None			
Attached information	None							
Precautions/ Remarks	After the _EIP_MacAdrErr system-defined variable changes to TRUE, it will not change to FALSE unless the power supply to the Controller is cycled.							

Event name	EtherNet/IP Processing Error Event code 1422 0000 hex						
Meaning	A fatal error was detected in the EtherNet/IP Function Module.						
Source	EtherNet/IP Func	tion Module	Source details	Communications port	Detection Continuous timing		
Error attributes	Level	Partial fault	Recovery	Cycle the power supply.	Log category System		
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not	operate.	
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET	EtherNet/IP NET ERR EtherNet/IP LINK/ACT			
	OFF		Lights.				
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	Hardware has fail	ed.	Replace the CPU	Unit.	None		
Attached information	None						
Precautions/ Remarks	None						

Event name	Basic Ethernet Se	tting Error		Event code	3421 0000 hex			
Meaning	An error was dete	cted in the Ethernet	settings.					
Source	EtherNet/IP Funct	on Module	Source details	Communications port	Detection timing	At power ON or Controller reset		
Error attributes	Level	Partial fault	Recovery	Automatic recovery, cycle the power supply, or reset Controller	Log category	System		
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not op	oerate.		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK//	ACT		
	OFF	OFF		rvals.				
System-defined	Variable		Data type		Name			
variables	_EIP_EtnCfgErr		BOOL		Basic Ethernet Set	tting Error		
Cause and	Assumed cause		Correction		Prevention			
correction	Setting error		Identify the error fr information, correct then download the	t the setting, and	None			
	Power was interru load was in progre EtherNet/IP port s		Perform the Clear tion or download th		Do not turn OFF the the Controller while progress for the but port settings.	e a download is in		
	Memory error		If operation is not recovered by the above, replace the CPU Unit.		None			
Attached information	Attached informati	on 2: Error details		-	ror, 02 hex: Inconsistency in settings) , 12 hex: Unsupported baud rate)			
Precautions/ Remarks	The cause of error	can be identified wi	th the attached infor	mation.	g,			

Event name	IP Address Setting	Error		Event code	34220000 hex	34220000 hex			
Meaning	An error was detec	ted in the IP addres	s settings.	•					
Source	EtherNet/IP Functi	on Module	Source details	Communications port	Detection timing	At power ON or Controller reset			
Error attributes	Level	Partial fault	Recovery	Automatic recovery (after downloading the settings), cycle the power supply, or reset Controller	Log category	System			
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not	operate.			
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINE	(/ACT			
	OFF		Flashes at 1-s int	ervals.					
System-defined	Variable	Variable		Data type					
variables	_EIP_IPAdrCfgErr	_EIP_IPAdrCfgErr		BOOL		ng Error			
Cause and	Assumed cause		Correction		Prevention				
correction	Setting error		,	from the attached ect the setting, and e settings again.	None				
	Power was interrupted was in progree EtherNet/IP port so	ss for the built-in		All Memory opera- the settings again.	Do not turn OFF the power suppl the Controller while a download is progress for the built-in EtherNet/ port settings.				
	The IP address ac BOOTP server is it		vided to this port server so that it is	•	to this port by the	ss that was provided a BOOTP server so e range specified for ntroller.			
	Memory error		If operation is not above, replace th	•	None				
Attached	Attached informati	on 1: Type of errors	(01 hex: Non-volati	e memory access er	ror, 02 hex: Incons	istency in settings)			
information	Attached informati	on 2: Error details (0	00 hex: Non-volatile	memory access erro	r				
				, 12 hex: Illegal subne S setting error, 16 he					
Precautions/ Remarks	The cause of error	can be identified wi	be identified with the attached information.						

Event name	IP Address Duplica	ation Error		Event code	8401 0000 hex		
Meaning	The same IP addre	ess is used more tha	an once.				
Source	EtherNet/IP Functi	on Module	Source details	Communications port	Detection timing	After link is established	
Error attributes	Level	Partial fault	Recovery	Automatic recovery	Log category	System	
Effects	User program	Continues.	Operation		unications will not op ocal IP address are o		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/	ACT	
	OFF		Flashes at 1-s inte	rvals.			
System-defined	Variable		Data type		Name		
variables	_EIP_IPAdrDupErr	•	BOOL		IP Address Duplication Error		
Cause and	Assumed cause		Correction		Prevention		
correction	The IP address of the built-in Ether-Net/IP port is also used as the IP address of another node.		settings so that the not used by more . Remove the nod	rer supply to the the Controller. dresses of other ct the IP address he same address is e than one node.	Perform allocations addresses of node are used for only o	s on the network	
Attached information	Attached information 1: Duplicated IP address (example: C0A8FA01 hex = address 192.168.250.1)						
Precautions/ Remarks	A duplicated address error occurs if an ARP is sent with the set IP address and there is an ARP response.						

Event name	BOOTP Server C	onnection Error		Event code	84020000 hex		
Meaning	Connection with t	he BOOTP server fa	iled.				
Source	EtherNet/IP Func	tion Module	Source details	Communications port	Detection timing	At BOOTP operation	
Error attributes	Level	Partial fault	Recovery	Automatic recovery	Log category	System	
Effects	User program	Continues.	Operation	the BOOTP server of from the BOOTP server of Function Module will		unications will not operate. Requests to will continue until there is a response erver. Data refreshing with the PLC vill continue. An IP address was not set port when it was supposed to be set erver.	
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINE	K/ACT	
	OFF		Flashes at 1-s in	Flashes at 1-s intervals.			
System-defined	Variable		Data type	Data type			
variables	_EIP_BootpErr		BOOL		BOOTP Server Error		
Cause and	Assumed cause		Correction		Prevention		
correction	Server setting error			Correct the server settings at the remote connection.		Check to make sure that the server settings at the remote connection are correct.	
	Server is down.		Check if the server at the remote connection is operating normally and set it to operate normally if it is not.		Check to make sure that the server a the remote connection is operating normally.		
	An error occurred tions path.	in the communica-	Check the communications path to the server and take corrective measures if there are any problems.		None		
Attached information	None				1		
Precautions/ Remarks	None						
Event name	Communications	Controller Failure		Event code	0421 0000 hex*		
Meaning	A hardware error	was detected in the	communications co	ntroller of the built-in	EtherNet/IP port.		
Source	EtherNet/IP Func	tion Module	Source details	Communications port 1 or 2	Detection timing	Continuously	
Error attributes	Level	Minor fault	Recovery	Cycle the power supply.	Log category	System	
Effects	User program	Continues	Operation	EtherNet/IP communications are not possible for the rel			

Event name	Communications Controller Failure			Event code	0421 0000 hex*		
Meaning	A hardware error	was detected in the	communications co	ntroller of the built-in	EtherNet/IP port.		
Source	EtherNet/IP Function Module		Source details	Communications port 1 or 2	Detection timing	Continuously	
Error attributes	Level	Minor fault	Recovery	Cycle the power supply.	Log category	System	
Effects	User program	Continues.	Operation	EtherNet/IP comm	nunications are not pons port.	ossible for the rele-	
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET	EtherNet/IP NET ERR		ACT	
	OFF		Lights.	Lights.			
System-defined	Variable		Data type	Data type			
variables	_EIP1_LanHwErr		BOOL	BOOL		tions Controller	
	_EIP2_LanHwErr		BOOL	BOOL		Port2 Communications Controller Error	
Cause and	Assumed cause		Correction		Prevention		
correction	Hardware error in the communications controller		Replace the CPL	Replace the CPU Unit.		None	
Attached information	None						
Precautions/ Remarks	After the _EIP1_LanHwErr or _EIP2_LanHwErr system-defined variable changes to TRUE, it will not change to FALSE unless the power supply to the Controller is cycled.						

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	Identity Error			Event code	1421 0000 hex			
Meaning	The CIP identity in	The CIP identity information in non-volatile memory was not read correctly.						
Source	EtherNet/IP Func	tion Module	Source details	Source details CIP,*1 CIP1,*2 or CIP2*2		At power ON or Controller reset		
Error attributes	Level	Minor fault	Recovery	Cycle the power supply.	Log category	System		
Effects	User program	Continues.	Operation	Operation EtherNet/IP communications are not possible for vant communications port.				
Indicators	EtherNet/IP NET RUN EtherNet/IP N		EtherNet/IP NET	RR EtherNet/IP LINK/ACT				
		Lights.						
System-defined	Variable		Data type	Data type		Name		
variables	_EIP_IdentityErr*1		BOOL	BOOL		Identify Error		
	_EIP1_IdentityErr*2		BOOL	BOOL		CIP Communications1 Identity Error		
	_EIP2_IdentityErr*2		BOOL	BOOL		CIP Communications2 Identity Error		
Cause and	Assumed cause		Correction	Correction		Prevention		
correction	Non-volatile mem	ory failure	Replace the CPU	Replace the CPU Unit.		None		
Attached information	None				•			
Precautions/ Remarks	None							

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	MAC Address Error			Event code	14230000 hex*		
Meaning	The MAC addres	s in non-volatile m	emory was not read c	orrectly.	•		
Source	EtherNet/IP Function Module		Source details	Communications port 1 or 2	Detection timing	At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Cycle the power supply.	Log category	System	
Effects	User program	Continues.	Operation	Operation EtherNet/IP communications are not possible for th vant communications port.			
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT	
	OFF		Lights.	Lights.			
System-defined	Variable		Data type	Data type		Name	
variables	_EIP1_MacAdrE	_EIP1_MacAdrErr		BOOL		Port1 MAC Address Error	
	_EIP2_MacAdrE	rr	BOOL	BOOL		Port2 MAC Address Error	
Cause and	Assumed cause		Correction		Prevention		
correction	Non-volatile mem	nory failure	Replace the CPU	Replace the CPU Unit.		None	
Attached information	None						
Precautions/ Remarks			22_MacAdrErr system- the Controller is cycled		nges to TRUE, it wil	I not change to	

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	Tag Data Link Sett	ing Error		Event code	3420 0000 hex	
Meaning	An error was detec	cted in the communic	cations settings for to	ag data links.		
Source	EtherNet/IP Functi	on Module	Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	At power ON or Controller reset
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the tag data link settings), cycle the power supply, or reset Controller	Log category	System
Effects	User program	Continues.	Operation	Tag data link comm	nunications will not o	perate.
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	ERR	EtherNet/IP LINK/	ACT
	Flashes at 1-s inte	rvals.	Flashes at 1-s intervals.			
System-defined	Variable		Data type		Name	
variables	_EIP_TDLinkCfgErr <sup>*1</sup>		BOOL	BOOL		ing Error
	_EIP1_TDLinkCfgErr*2		BOOL	BOOL		ns1 Tag Data Link
	_EIP2_TDLinkCfg	Err <sup>*2</sup>	BOOL		CIP Communications2 Tag Data Link Setting Error	
Cause and	Assumed cause		Correction		Prevention	
correction	Power was interrupted when a download was in progress for the data link settings.		sures.  • Perform the Clearation.  • Download the taragain.	<ul><li>Perform the Clear All Memory operation.</li><li>Download the tag data link settings</li></ul>		ne power supply to e a download is in g data link settings.
	Memory error		If operation is not above, replace the	,	None	
Attached information	Attached informati	on 1: Type of errors	(01 hex: Non-volatile	e memory access err	or, 02 hex: Inconsist	ency in settings)
Precautions/ Remarks	None					

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	IP Route Table Se	tting Error		Event code	3423 0000 hex		
Meaning	An error was dete	cted in the IP router	tables or default gat	eway.			
Source	EtherNet/IP Function Module		Source details	Communications port	Detection timing	At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the settings), cycle the power supply, or reset Controller	Log category	System	
Effects	User program	Continues.	Operation	Communications to gateway are not po	nat use the IP router ossible.	tables or default	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
			Flashes at 1-s intervals.				
System-defined	Variable		Data type	Data type			
variables	_EIP_IPRTblErr		BOOL	BOOL		or	
Cause and	Assumed cause		Correction		Prevention		
correction	Setting error	Setting error		Identify the error from the attached information, correct the setting, and then download the settings again.		None	
	Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.			Perform the Clear All Memory operation or download the settings again.		Do not turn OFF the power supply to the Controller while a download is in progress for the built-in EtherNet/IP port.	
	Memory error	Memory error		If operation is not recovered by the above, replace the CPU Unit.		None	
Attached	Attached informati	on 1: Type of errors	(01 hex: Non-volatile	e memory access err	or, 02 hex: Inconsis	tency in settings)	
information	Attached informati	on 2: Error details (0	00 hex: Non-volatile	memory access erro	r)		
	When the settings default gateway)	are inconsistent (11	hex: Illegal IP route	er table settings, 12 h	ex: Illegal Hosts set	ting, 13 hex: Invalid	
Precautions/ Remarks	The cause of error	can be identified wi	th the attached infor	rmation.			

Event name	FTP Server Setting Error			Event code	34240000 hex		
Meaning	An error was dete	cted in the FTP serv	er settings.				
Source	EtherNet/IP Func	tion Module	Source details	Source details FTP		At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the FTP settings), cycle the power supply, or reset Controller	Log category	System	
Effects	User program	Continues.	Operation	FTP will not operate	ite.		
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
			Flashes at 1-s int	Flashes at 1-s intervals.			
System-defined	Variable		Data type	Data type			
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	Setting error		information, corre	Identify the error from the attached information, correct the setting, and then download the settings again.		None	
		Power was interrupted when a download was in progress for the FTP server settings.		Perform the Clear All Memory operation or download the settings again.		Do not turn OFF the power supply to the Controller while a download is in progress for the FTP server settings.	
	Memory error	Memory error		If operation is not recovered by the above, replace the CPU Unit.		None	
Attached information	Attached informat	ion 1: Type of errors	(01 hex: Non-volati	e memory access err	ror, 02 hex: Inconsis	stency in settings)	
Precautions/ Remarks	The cause of erro	r can be identified wi	th the attached info	rmation.			

Event name	NTP Client Setting	g Error		Event code	3425 0000 hex		
Meaning	An error was dete	cted in the NTP clier	nt settings.				
Source	EtherNet/IP Funct	EtherNet/IP Function Module		NTP	Detection timing	At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the NTP settings), cycle the power supply, or reset Controller	Log category	System	
Effects	User program	Continues.	Operation	NTP operation sto	ps.		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
			Flashes at 1-s intervals.				
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause	Assumed cause		Correction			
correction	Setting error		information, correct	Identify the error from the attached information, correct the setting, and then download the settings again.		None	
		Power was interrupted when a down- load was in progress for the NTP cli- ent settings.		Perform the Clear All Memory operation or download the settings again.		Do not turn OFF the power supply to the Controller while a download is in progress for the NTP client settings.	
	Memory error	Memory error		If operation is not recovered by the above, replace the CPU Unit.		None	
Attached information	Attached informati	on 1: Type of errors	(01 hex: Non-volatile	e memory access er	ror, 02 hex: Inconsist	ency in settings)	
Precautions/ Remarks	The cause of error	can be identified wi	th the attached infor	mation.			

Event name	SNMP Setting Erro	or		Event code	34260000 hex		
Meaning	<u> </u>	cted in the SNMP ag	ent/trap settings.				
Source	EtherNet/IP Functi	on Module	Source details	SNMP	Detection timing	At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the SNMP settings), cycle the power supply, or reset Controller	Log category	System	
Effects	User program	Continues.	Operation	SNMP operation s	tops.		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
			Flashes at 1-s intervals.				
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause	Assumed cause		Correction			
correction	Setting error		information, corre	Identify the error from the attached information, correct the setting, and then download the settings again.		None	
	Power was interrupted when a download was in progress for the SNMP agent/trap settings.			Perform the Clear All Memory operation or download the settings again.		Do not turn OFF the power supply to the Controller while a download is in progress for the SNMP agent/trap set- tings.	
	Memory error	Memory error		If operation is not recovered by the above, replace the CPU Unit.		None	
Attached	Attached informati	on 1: Type of errors	(01 hex: Non-volatil	e memory access er	ror, 02 hex: Inconsi	stency in settings)	
information	Attached informati 02 hex: SNMP trap	·	when there is an in	consistency in the se	ttings (01 hex: SNM	IP agent settings,	
Precautions/ Remarks	The cause of error	can be identified wi	th the attached info	rmation.			

Event name	Tag Name Resolu	ution Error		Event code	34270000 hex		
Meaning	Resolution of a ta	g used in a tag data l	ink failed.		•		
Source	EtherNet/IP Func	tion Module	Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	At power ON, at Controller reset, when variables are changed from the Sysmac Stu- dio, or when the data link table is changed from the Network Configu- rator	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the tag settings)	Log category	System	
Effects	User program	Continues.	Operation	Data links will not for other tags will o	operate for unresolved tags. Data links operate.		
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
	Flashes at 1-s int	ervals.	Flashes at 1-s inte	Flashes at 1-s intervals.			
System-defined variables	Variable		Data type		Name		
	_EIP_TagAdrErr*	_EIP_TagAdrErr <sup>*1</sup>			Tag Name Resolu	tion Error	
	_EIP1_TagAdrErr*2		BOOL		CIP Communication Resolution Error	ons1 Tag Name	
	_EIP2_TagAdrErr*2		BOOL	BOOL		ons2 Tag Name	
Cause and	Assumed cause		Correction		Prevention		
correction	The size of the network variable is different from the tag settings.		Correct the sizes in the tag settings to match the network variables.		Set the sizes in the tag settings to match the network variables.		
	and the I/O direct	The I/O direction set for a tag data link and the I/O direction of the Controller variable do not match.		Correct the tag settings or the settings of the Controller variables so that the I/O direction for the tag data links match the I/O direction of the Controller variable.		Set the tag settings or the settings of the Controller variables so that the I/O directions for the tag data links match the I/O directions of the Controller variables.	
	There are no netw Controller tag set	vork variables for the tings.		Correct the tag settings so that existing network variables are set for the tags.		Set the tag settings so that existing network variables are set for the tags.	
	for a tag data link Publish attribute s	A variable in the Controller that is set for a tag data link has the Network Publish attribute set to Input but also has the Constant attribute.		Remove the Constant attribute from the Controller variable that has the Network Publish attribute set to Input.		Do not set the Constant attribute for a Controller variable that has the Network Publish attribute set to Input.	
Attached information	None						
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Basic Ethernet Se	tting Error		Event code	34280000 hex*		
Meaning	An error was dete	cted in the Ethernet	settings.				
Source	EtherNet/IP Funct	ion Module	Source details	Communications port 1 or 2	Detection timing	At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Automatic recovery, cycle the power supply, or reset Controller.	Log category	System	
Effects	User program	Continues.	Operation	EtherNet/IP comm vant communication	unications are not pons port.	ossible for the rele-	
Indicators	EtherNet/IP NET I	RUN	EtherNet/IP NET E	ERR	EtherNet/IP LINK/	ACT	
	OFF		Flashes at 1-s inte	ervals.			
System-defined	ed Variable		Data type		Name		
variables	_EIP1_EtnCfgErr		BOOL	BOOL		net Setting Error	
	_EIP2_EtnCfgErr		BOOL		Port2 Basic Etherr	net Setting Error	
Cause and	Assumed cause		Correction		Prevention		
correction	Parameter error	Parameter error		Identify the error from the attached information, correct the setting, and then download the settings again.		None	
	load was in progre	Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.		Perform the Clear All Memory operation or download the settings.		Do not turn OFF the power supply to the Controller while a download is in progress for the built-in EtherNet/IP port settings.	
	A memory error of	A memory error occurred.		recovered by the CPU Unit.	None		
Attached	Attached informati	on 1: Type of errors	(01 hex: Non-volatile	e memory access er	ror, 02 hex: Inconsis	tency in settings)	
information	Attached informati hex: Unsupported	on 2: Error details (0 baud rate)	0 hex: Non-volatile r	nemory access error	, 11 hex: Incorrect b	aud rate setting, 12	
Precautions/ Remarks	The cause of error	r can be identified wi	th the attached infor	mation.			

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	IP Address Setting	Error		Event code	3429 0000 hex*		
Meaning	An error was detec	cted in the IP addres	ss settings.				
Source	EtherNet/IP Functi	on Module	Source details	Communications port 1 or 2	Detection timing	At power ON or Controller reset	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the settings), cycle the power supply, or reset Controller.	Log category	System	
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications are not pons port.	ossible for the rele-	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	ERR	EtherNet/IP LINK//	ACT	
	OFF		Flashes at 1-s inte	ervals.			
System-defined	Variable		Data type		Name		
variables	_EIP1_IPAdrCfgErr		BOOL	BOOL		Setting Error	
	_EIP2_IPAdrCfgErr		BOOL		Port2 IP Address S	Setting Error	
Cause and	Assumed cause		Correction		Prevention		
correction	Parameter error		information, correct	Identify the error from the attached information, correct the setting, and then download the settings again.			
	Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.			Perform the Clear All Memory operation or download the settings again.		Do not turn OFF the power supply to the Controller while a download is in progress for the built-in EtherNet/IP port settings.	
	The IP address acquired from BOOTP server is illegal.		vided to this port be server so that it is	Correct the IP address that was provided to this port by the BOOTP server so that it is within the range specified for an NX-series Controller.		Correct the IP address that was provided to this port by the BOOTP server so that it is within the range specified for an NX-series Controller.	
	A memory error od	ccurred.	If operation is not above, replace the	·	None		
Attached	Attached informati	on 1: Type of errors	(01 hex: Non-volatile	e memory access er	or, 02 hex: Inconsist	tency in settings)	
information	Attached informati	on 2: Error details (0	00 hex: Non-volatile i	memory access erro	r)		
	When the settings	are inconsistent					
	(11 hex: Illegal IP	address, 12 hex: Ille	gal subnet mask)				
Precautions/ Remarks	The cause of error	can be identified wi	th the attached infor	mation.			

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	DNS Setting Error			Event code	342A0000 hex*				
Meaning	An error was dete	cted in the DNS setti	ngs or Hosts setting	gs.					
Source	EtherNet/IP Funct	ion Module			Detection timing	At power ON or Controller reset			
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the settings), cycle the power supply, or reset Controller.	Log category System				
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not o	perate.			
Indicators	EtherNet/IP NET I	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINK/ACT				
	OFF		Flashes at 1-s into	ervals.					
System-defined	Variable		Data type		Name				
variables	_EIP_DNSCfgErr		BOOL		DNS Setting Error				
Cause and	Assumed cause		Correction		Prevention				
correction	Parameter error		Identify the error from the attached information, correct the setting, and then download the settings again.		None				
	Power was interrupted when a download was in progress for the built-in EtherNet/IP port settings.		Perform the Clear All Memory operation or download the settings again.		Do not turn OFF the power supply to the Controller while a download is in progress for the built-in EtherNet/IP port settings.				
	A memory error occurred.		If operation is not recovered by the above, replace the CPU Unit.		None				
Attached	Attached informati	Attached information 1: Type of errors (01 hex: Non-volatile memory access error, 02 hex: Inconsistency in settings)							
information	Attached informati	Attached information 2: Error details (00 hex: Non-volatile memory access error)							
	When the settings	When the settings are inconsistent							
	(14 hex: Preferred	DNS setting error, 1	5 hex: Alternate DN	IS setting error, 16 he	ex: Illegal domain n	ame)			
Precautions/ Remarks	The cause of erro	r can be identified wi	th the attached info	rmation.					

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	Controller Insufficient Memory Warning			Event code	5001 0000 hex*		
Meaning	The amount of data for the EtherCAT slave configuration, network-published information, or other data exceeds the value that is specified for the CPU Unit. You may not be able to perform online editing or other operations.						
Source	EtherCAT Master Function Module or EtherNet/IP Function Module		Source details	Master or CIP	Detection timing	At power ON, download, or online editing	
Error attributes	Level	Minor fault	Recovery	Automatic recovery	Log category	System	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	The amount of data for the EtherCAT slave configuration, network-published information, or other data exceeds the value that is specified for the CPU Unit.		Reduce the number of PDOs that are used by the EtherCAT slaves.  Reduce the number of data types that are used for network variables or reduce the length of the text strings that are used for names.				
Attached information	None						
Precautions/ Remarks	You may not be at	You may not be able to perform online editing or other operations.					

 $<sup>^{\</sup>star}\,$  This event code occurs for unit version 1.02 or later of the CPU Unit.

Event name	DNS Server Connection Error			Event code	84030000 hex			
Meaning	Connection with the	Connection with the DNS server failed.						
Source	EtherNet/IP Function Module Source details		Source details	Communications port	Detection timing	At DNS operation		
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the DNS settings)	Log category	System		
Effects	User program	Continues.	Operation	Communications u	sing DNS stop.			
Indicators	EtherNet/IP NET I	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK//	ACT		
			Flashes at 1-s inte	rvals.				
System-defined	Variable		Data type		Name			
variables	_EIP_DNSSrvErr		BOOL		DNS Server Connection Error			
Cause and	Assumed cause		Correction		Prevention			
correction	Parameter error		If there is a mistake with the specifications of the connected server, correct the server specifications and download them again.		Make sure that the connected server is specified correctly.			
	Server is down.		Check if the server at the remote con- nection is operating normally and set it to operate normally if it is not.		Check to make sure that the server at the remote connection is operating normally.			
	An error occurred in the communications path.		Check the communications path to the server and take corrective measures if there are any problems.		None			
Attached information	None		•		•			
Precautions/ Remarks	None							

Event name	NTP Server Connection Error			Event code	84040000 hex		
Meaning	Connection with t	Connection with the NTP server failed.					
Source	EtherNet/IP Func	tion Module			Detection timing	At NTP operation	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the NTP settings)	Log category	System	
Effects	User program	Continues.	Operation	Time cannot be ac	quired from NTP.	•	
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK//	ACT	
			Flashes at 1-s inte	ervals.			
System-defined	Variable		Data type		Name		
variables	_EIP_NTPSrvErr		BOOL		NTP Server Connection Error		
Cause and	Assumed cause		Correction		Prevention		
correction	Parameter error		If there is a mistake with the specifications of the connected server, correct the server specifications and download them again.		Make sure that the connected server is specified correctly.		
	Server is down.		Check if the server at the remote con- nection is operating normally and set it to operate normally if it is not.		Check to make sure that the server at the remote connection is operating normally.		
	An error occurred in the communications path.		Check the communications path to the server and take corrective measures if there are any problems.		None		
Attached information	None		1		1		
Precautions/ Remarks	If TCP Server Run is recorded in the event log after the correction is made, then the CPU Unit is correctly connected to the DNS server.						

Event name	Tag Data Link Cor	nection Failed		Event code	ent code 8407 0000 hex	
Meaning	Establishing a tag	data link connection	failed.			
Source	EtherNet/IP Function Module S		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	When establishing tag data link connection
Error attributes	Level	Minor fault	Recovery Automatic recovery		Log category	System
Effects	User program	Continues.	Operation		perate for connections that could not a links for other connections will open	
Indicators	EtherNet/IP NET I	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/	ACT
	Flashes at 1-s inte	ervals.	Flashes at 1-s intervals.			
System-defined	Variable		Data type		Name	
variables	_EIP_TDLinkOpnErr*1		BOOL		Tag Data Link Connection Failed	
	_EIP1_TDLinkOpnErr*2		BOOL		CIP Communications1 Tag Data Link Connection Failed	
	_EIP2_TDLinkOpnErr*2		BOOL		CIP Communications2 Tag Data Link Connection Failed	
Cause and	Assumed cause		Correction		Prevention	
correction	The tag data link connection information is not the same for the originator and target.		Correct the tag data link connection information, and then download the device parameters or connection settings from the Network Configurator or Sysmac Studio.		Before you use the tag data links, make sure that the tag data link connection information in the originator and target are suitable.	
	Insufficient connections		Reduce the number of class-3 messages.		Reduce the number of data links and class-3 messages that are used.	
Attached information	Attached information 1: Target node IP address (example: C0A8FA01 hex = address 192.168.250.1)					
Precautions/	You can investigat	e a detailed cause fr	om the connection s	tatus.		
Remarks	Refer to 15-3-2 Co	onnection Status Cod	des and Troubleshoo	oting.		

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link Timeout			Event code	84080000 hex	
Meaning	A timeout occurred	d in a tag data link.				
Source	EtherNet/IP Function Module		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	Continuously after starting tag data link commu- nications
Error attributes	Level			Automatic recovery	Log category	System
Effects	User program	Continues.	Operation		nk connection will stop. Reconnection dically repeated for the tag data link	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/	ACT
	Flashes at 1-s inte	rvals.	Flashes at 1-s inte	rvals.		
System-defined	Variable		Data type		Name	
variables	_EIP_TDLinkErr*1		BOOL		Tag Data Link Con	nmunications Error
	_EIP1_TDLinkErr*2		BOOL		CIP Communications1 Tag Data Link Communications Error	
	_EIP2_TDLinkErr*2		BOOL	CIP Communications2 Tag Communications Error		-
Cause and	Assumed cause		Correction		Prevention	
correction	The power supply to the target node is OFF.				Use the tag data link after you confirm that the target node is normal.	
correction		to the target node	Check the status o and start it normall			
Conection						
Correction	is OFF.  Communications w	vith the target node		y. inector and make		e is normal.
Correction	is OFF. Communications w stop. The Ethernet cable	vith the target node	and start it normall  Reconnect the con	nnector and make rrectly.	that the target nod	e is normal.
Correction	is OFF.  Communications w stop.  The Ethernet cable disconnected.  The Ethernet cable	vith the target node	and start it normall  Reconnect the consure it is mated consure it is mated consumptions.	nnector and make rrectly. net cable. ountermeasures if	Connect the connect None	ector securely.
COTTOCUTOR	is OFF. Communications w stop. The Ethernet cable disconnected. The Ethernet cable broken.	with the target node e for EtherNet/IP is e for EtherNet/IP is	Reconnect the consure it is mated connect the Ethern	inector and make rrectly. Inet cable.  ountermeasures if noise.  OFF Detected error r the assumed	Connect the connect the connect the connect the connect the connect there is excessive	ector securely.  ountermeasures if noise.  DFF Detected error r the assumed
Attached	is OFF.  Communications w stop.  The Ethernet cable disconnected.  The Ethernet cable broken.  Noise  The link to the built port is OFF.	with the target node e for EtherNet/IP is e for EtherNet/IP is	Reconnect the consure it is mated co. Replace the Ethern Implement noise or there is excessive Refer to the Link C (84060000 hex) for causes and other is	inector and make rrectly.  net cable.  ountermeasures if noise.  OFF Detected error r the assumed infomation on link-	Connect the connect the connect the connect the connect the connect there is excessive  Refer to the Link C (8406 0000 hex) for causes and other is	ector securely.  ountermeasures if noise.  DFF Detected error r the assumed
	is OFF.  Communications w stop.  The Ethernet cable disconnected.  The Ethernet cable broken.  Noise  The link to the built port is OFF.	with the target node e for EtherNet/IP is e for EtherNet/IP is t-in EtherNet/IP	Reconnect the consure it is mated co. Replace the Ethern Implement noise of there is excessive Refer to the Link C (84060000 hex) for causes and other in OFF.	nnector and make rrectly.  net cable.  ountermeasures if noise.  OFF Detected error r the assumed nfomation on link-	None Implement noise of there is excessive Refer to the Link C (8406 0000 hex) for causes and other in OFF.	ector securely.  ountermeasures if noise.  OFF Detected error r the assumed nfomation on link-
Attached	is OFF.  Communications w stop.  The Ethernet cable disconnected.  The Ethernet cable broken.  Noise  The link to the built port is OFF.  Attached informatic Attached informatic	with the target node e for EtherNet/IP is e for EtherNet/IP is t-in EtherNet/IP	Reconnect the consure it is mated conference the Ethern Implement noise of there is excessive  Refer to the Link C (84060000 hex) for causes and other in OFF.  Stance No. (0 to 255) address (example:	nnector and make rrectly.  net cable.  ountermeasures if noise.  OFF Detected error r the assumed nfomation on link-	None Implement noise of there is excessive Refer to the Link C (8406 0000 hex) for causes and other in OFF.	ector securely.  ountermeasures if noise.  OFF Detected error r the assumed nfomation on link-
Attached information	is OFF.  Communications w stop.  The Ethernet cable disconnected.  The Ethernet cable broken.  Noise  The link to the built port is OFF.  Attached informatic Attached informatic	with the target node e for EtherNet/IP is e for EtherNet/IP is t-in EtherNet/IP on 1: Connection ins on 2: Target node IP s are not included in	Reconnect the consure it is mated conference the Ethern Implement noise of there is excessive  Refer to the Link C (84060000 hex) for causes and other in OFF.  Stance No. (0 to 255) address (example:	nnector and make rrectly.  net cable.  ountermeasures if noise.  OFF Detected error r the assumed nfomation on link-	None Implement noise of there is excessive Refer to the Link C (8406 0000 hex) for causes and other in OFF.	ector securely.  ountermeasures if noise.  OFF Detected error r the assumed nfomation on link-

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link Connection Timeout			Event code	84090000 hex*1		
Meaning	A timeout occurred	d while trying to esta	blish a tag data link	connection.	•		
Source	EtherNet/IP Function Module		Source details	CIP,*2 CIP1,*3 or CIP2*3	Detection timing	When establishing tag data link connection	
Error attributes	Level	Minor fault	Recovery	Automatic recovery Log category		System	
Effects	User program	Continues.	Operation		operate for connections that timed out essing is periodically repeated for the ned out.		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/	ACT	
	Flashes at 1-s inte	rvals.	Flashes at 1-s inte	rvals.			
System-defined	Variable		Data type		Name		
variables	_EIP_TDLinkOpnErr*2		BOOL		Tag Data Link Cor	nnection Failed	
	_EIP1_TDLinkOpnErr*3		BOOL		CIP Communications1 Tag Data Link Connection Failed		
	_EIP2_TDLinkOpr	_EIP2_TDLinkOpnErr*3		CIP Communication Connection Failed		tions2 Tag Data Link d	
Cause and cor-	Assumed cause		Correction		Prevention		
rection	The power supply is OFF.	to the target node	Check the status of the target node and start it normally.		Use the tag data link after you confirm that the target node is normal.		
	Communications a are stopped.	at the target node					
	The Ethernet cable erNet/IP is disconr	e connector for Eth- nected.	Reconnect the connector and make sure it is mated correctly.		Connect the connector securely.		
	The Ethernet cable for EtherNet/IP is broken.		Replace the Ethernet cable.		None		
	An error occurred in the communications path.		Check the communications path and take corrective measures if there are any problems.		None		
Attached information	Attached informati	on 1: Target node IP	address (example:	C0A8FA01 hex = ad	dress 192.168.250.	1)	
Precau- tions/Remarks			bservation level. If yed to "" (no chang			vel, the EtherNet/IP	
	The following cases are not included in this error.						
	Connections as	a target					
	Connection time	outs due to a Link O	FF detection for an	Ethernet switch			
	to the state of the CRITICAL						

<sup>\*1</sup> This event code occurs for unit version 1.04 or later of the CPU Unit.

<sup>\*2</sup> This applies to an NJ-series CPU Unit.

<sup>\*3</sup> This applies to an NX-series CPU Unit.

Event name	IP Address Duplic	IP Address Duplication Error			840A0000 hex*			
Meaning	The same IP add	The same IP address is used more than once.						
Source	EtherNet/IP Function Module		Source details	Communications port 1 or 2	Detection timing	After link is established		
Error attributes	Level	Minor fault	Recovery	Automatic recovery	Log category System			
Effects	User program	Continues.	Operation	vant communication	unications are not possible for the re ns port. Packets addressed to the loc elevant communications port are dis-			
Indicators	EtherNet/IP NET	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINK/	ACT		
	OFF		Flashes at 1-s intervals.					
System-defined	Variable		Data type		Name			
variables	_EIP1_IPAdrDupErr		BOOL		Port1 IP Address Duplication Error			
	_EIP2_IPAdrDupErr		BOOL	BOOL		Port2 IP Address Duplication Error		
Cause and	Assumed cause		Correction	Correction				
correction	The IP address of the built-in Ether-Net/IP port is also used as the IP address of another node.		settings so that not used by mo • Remove the nod	ver supply to the t the Controller. Idresses of other ect the IP address the same address is re than one node.	addresses of nodes on the network are used for only one node.  other lidress lidress is node.  the			
Attached information	Attached informat	ion 1: Duplicated IP		COA8FA01 hex = add	L ress 192.168.250.1)			
Precautions/ Remarks	A duplicated addr	ess error occurs if a	n ARP is sent with th	ne set IP address and	there is an ARP re	sponse.		

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	BOOTP Server Connection Error			Event code	840B0000 hex*		
Meaning	Connection with th	ne BOOTP server fai	led.				
Source	EtherNet/IP Function Module		Source details	Communications port 1 or 2	Detection timing	At BOOTP operation	
Error attributes	Level	Minor fault	Recovery	Automatic recovery	Log category System		
Effects	User program	Continues.	Operation	vant communicatio will continue until the server. Data refres continue. An IP ad	unications are not possible for the rele- ns port. Requests to the BOOTP server here is a response from the BOOTP hing with the PLC Function Module will dress was not set for the EtherNet/IP upposed to be set from the BOOTP		
Indicators	EtherNet/IP NET I	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
	OFF		Flashes at 1-s intervals.				
System-defined	Variable		Data type		Name		
variables	_EIP1_BootpErr		BOOL		Port1 BOOTP Server Error		
	_EIP2_BootpErr		BOOL	Port2 BOOTP Server Erro		ver Error	
Cause and	Assumed cause		Correction	Prevention			
correction	Server setting error		Correct the server settings at the remote connection.		Check to make sure that the server settings at the remote connection are correct.		
	The server is down.		Check if the server at the remote connection is operating normally and set it to operate normally if it is not.		Check to make sure that the server at the remote connection is operating normally.		
	An error occurred tions path.	An error occurred in the communications path.		Check the communications path to the server and take corrective measures if there are any problems.		None	
Attached information	None	None					
Precautions/ Remarks	None						

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	Tag Data Link Equ Exceeded	ipment Total Allowat	ole Bandwidth	Event code	840C0000 hex*		
Meaning		th for the connections /IP ports in the equip		ablished exceeded th	ne allowable tag data	a link bandwidth for	
Source	EtherNet/IP Function Module		Source details	CIP1 or CIP2	Detection timing	When establishing tag data link connection	
Error attributes	Level	Minor fault	Recovery	Automatic recovery (after downloading the settings), cycle the power supply, or reset Controller.	Log category	System	
Effects	User program	Continues.	Operation		not operate in the bandwidth that able total PPS for the equipment. Other tions will continue.		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINK/	ACT	
	Flashes at 1-s inte	ervals.	Flashes at 1-s inte	Flashes at 1-s intervals.			
System-defined	Variable		Data type		Name		
variables	_EIP_TDLinkOpnErr		BOOL		Tag Data Link Cor	nection Failed	
	_EIP1_TDLinkOpnErr		BOOL		CIP Communications1 Tag Data Link Connection Failed		
	_EIP2_TDLinkOpr	_EIP2_TDLinkOpnErr		BOOL		CIP Communications2 Tag Data Link Connection Failed	
Cause and	Assumed cause		Correction		Prevention		
correction	total of the packet tag data links that	ould cause the PPS transfer rates of the use the EtherNet/IP ment to exceed the	node so that the to the EtherNet/IP po	orts in the equip- ceed the allowable	data size for all EtherNet/IP por the equipment does not exceed total allowable bandwidth for the		
Attached information	None				,		
Precautions/ Remarks	None						

<sup>\*</sup> This event code occurs for unit version 1.10 or later of the CPU Unit.

Event name	Access Detected C	Outside Range of Var	riable	Event code	54E00000 hex		
Meaning	Accessing a value	that is out of range v	was detected for a ta	g variable that is use	ed in a tag data link.		
Source	EtherNet/IP Function Module		Source details	Communications port	Detection timing	When variable is written	
Error attributes	Level	Observation	Recovery		Log category	System	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK//	ACT	
System-defined	ystem-defined Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	An out-of-range value was written by an EtherNet/IP tag data link for a variable with a specified range. A value that does not specify an enumerator was written by an EtherNet/IP tag data link for an enumeration variable.		Correct the value that is written to the variable with a specified range so that the value is in the range. Correct the value that is written to the enumeration variable so that the value specifies an enumerator.		Write values that are in range for variables with specified ranges. Write values that specify enumerators to enumeration variables.		
Attached information	None						
Precautions/ Remarks		for out-of-range values of				mally.	

Event name	Packet Discarded	Due to Full Recept	ion Buffer	Event code	84050000 hex		
Meaning	A packet was disc	arded.			•		
Source	EtherNet/IP Function Module		Source details	Communications port,*1 communications port 1,*2 or communications port 2*2	Detection timing	After link is established	
Error attributes	Level	Observation	Recovery		Log category	System	
Effects	User program	Continues.	Operation	Not affected.		<u> </u>	
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
System-defined	Variable		Data type	Data type			
variables	None						
Cause and	Assumed cause		Correction	Correction		Prevention	
correction	A network convergence occurred.		Check whether the send unnecessal on the network a After that, check number of frame:	The load on the network is too high. Check whether there are nodes that send unnecessary broadcast frames on the network and remove them. After that, check that the received number of frames has reduced in the network statistical information.		Make sure that unnecessary broad- cast frames are not sent on the net- work. Do not connect the Ethernet cable in a loop.	
Attached information	None		<u> </u>		•		
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Link OFF Detected	t		Event code	84060000 hex		
Meaning	An Ethernet Link C	OFF was detected.					
Source	EtherNet/IP Function Module		Source details	Communications port,*1 communications port 1,*2 or communications port 2*2	Detection timing	Continuously	
Error attributes	Level	Observation	Recovery		Log category	System	
Effects	User program	Continues.	Operation	EtherNet/IP comm	unications will not o	perate.	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/	ACT	
					OFF		
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	An Ethernet cable is broken, disconnected, or loose.		Connect the Ethernet cable securely. If the cable is broken, replace it.		Connect the Ethernet cable securely. Check the cable to make sure that it is not disconnected.		
	The Ethernet switch power supply is turned OFF.		Turn ON the power supply to the Ethernet switch. Replace the Ethernet switch if it fails.		Do not turn OFF the power supply to the Ethernet switch.		
	Baud rate mismatch.		Correct the settings so that the same baud rate is used as for the remote communications nodes.		Set the same baud rate as for the remote communications nodes.		
	Noise	Noise		Implement noise countermeasures if there is excessive noise.		Implement noise countermeasures.	
	One of the following	g operations was	None		None		
	<ul> <li>performed.</li> <li>The Identity object was reset.</li> <li>Settings for EtherNet/IP were downloaded from the Network Configurator or Sysmac Studio, or the Clear All Memory operation was performed.</li> <li>EtherNet/IP was restarted.</li> </ul>		This error occurs when the operations on the left are performed.		This error occurs when the operations on the left are performed.		
Attached information	None						
Precautions/ Remarks		e event level to the n changed to "Automa		ou change the level to	the minor fault leve	el, the Recovery col-	

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link Download Started			Event code	9401 0000 hex		
Meaning	Changing the tag	Changing the tag data link settings started.					
Source	EtherNet/IP Function Module		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	At user operation	
Error attributes	Level	Information	Recovery		Log category	Access	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT	
	Flashes at 1-s intervals.						
System-defined	Variable		Data type		Name		
variables	None	None					
Cause and	Assumed cause		Correction		Prevention		
correction	Changing the tag data link settings started.						
Attached information	Attached information 1: Controller status (01 hex: PROGRAM mode, 02 hex: RUN mode)						
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link Dow	nload Finished		Event code	94020000 hex			
Meaning	Changing the tag	Changing the tag data link settings finished.						
Source	EtherNet/IP Function Module		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	At user operation		
Error attributes	Level	Information	Recovery		Log category	Access		
Effects	User program	Continues.	Operation	Not affected.		•		
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/ACT			
	Flashes at 1-s inte	rvals.						
System-defined	Variable		Data type		Name			
variables	None							
Cause and	Assumed cause		Correction	Correction		Prevention		
correction	Changing the tag of finished.	data link settings						
Attached information	Attached information	Attached information 1: Controller status (01 hex: PROGRAM mode, 02 hex: RUN mode)						
Precautions/ Remarks	None							

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link Stopped Event code			Event code	94030000 hex		
Meaning				Sysmac Studio, or ma Configurator or Sysm		em-defined variable.	
Source	EtherNet/IP Function Module		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	At user operation	
Error attributes	Level	Information	Recovery		Log category	Access	
Effects	User program	Continues.	Operation	Not affected.		•	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET I	ERR	EtherNet/IP LINK	ACT	
	Flashes at 1-s inte	rvals.					
System-defined	Variable		Data type		Name		
variables	_EIP_TDLinkStopCmd*1		BOOL		Tag Data Link Communications Stop Switch		
	_EIP1_TDLinkStopCmd*2		BOOL	BOOL		CIP Communications1 Tag Data Link Communications Stop Switch	
	_EIP2_TDLinkStopCmd*2		BOOL		CIP Communications2 Tag Data Link Communications Stop Switch		
Cause and	Assumed cause		Correction	Correction		Prevention	
correction	Tag data links were stopped by the Network Configurator, Sysmac Studio, or manipulation of a systemdefined variable.						
Attached information	Attached information 1: Controller status 01 hex: PROGRAM mode, 02 hex: RUN mode						
	Attached informati	on 2: Operation met	'	ation from the Networ	•	ysmac Studio,	
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link Sta	rted		Event code	94040000 hex		
Meaning		e started by the Netvable was downloaded				m-defined variable.	
Source	EtherNet/IP Function Module		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	At user operation	
Error attributes	Level	Information	Recovery		Log category	Access	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET I	RUN	EtherNet/IP NET E	RR	EtherNet/IP LINK/	ACT	
	Flashes at 1-s inte	ervals.					
System-defined	Variable		Data type		Name		
variables	_EIP_TDLinkStartCmd*1		BOOL		Tag Data Link Communications Start Switch		
	_EIP1_TDLinkSta	_EIP1_TDLinkStartCmd*2		BOOL		ons1 Tag Data Link Start Switch	
	_EIP2_TDLinkStartCmd*2		BOOL		CIP Communications2 Tag Data Link Communications Start Switch		
Cause and	Assumed cause		Correction	Correction		Prevention	
correction	Tag data links were started by the Network Configurator, Sysmac Studio, or manipulation of a systemdefined variable.						
Attached information	Attached informati	ion 1: Controller stati	us 01 hex: PROGRAM mode, 02 hex: RUN mode				
	Attached informat	ion 2: Operation met	thod 01 hex: Operation from the Network Configurator or Sysmac Studio 02 hex: Operation with system-defined variable				
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Link Detected			Event code	94050000 hex		
Meaning	Establishment of a	n Ethernet link was	detected.	•			
Source	EtherNet/IP Function Module		Source details	Communications port,*1 communications port 1,*2 or Communications port 2*2	Detection timing	When establishing link	
Error attributes	Level	Information	Recovery		Log category	System	
Effects	User program	Continues.	Operation	Not affected.		•	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
					Lights.		
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	Establishment of a detected.	Establishment of an Ethernet link was detected.					
Attached information	None				•		
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Restarting Ethernet Port			Event code	94060000 hex		
Meaning	The built-in EtherN	let/IP port was resta	rted.				
Source	EtherNet/IP Function Module		Source details	Communications port,*1 communications port 1,*2 or Communications port 2*2	Detection timing	At user operation	
Error attributes	Level	Information	Recovery		Log category	Access	
Effects	User program	Continues.	Operation	Not affected.		•	
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction	Correction		Prevention	
correction	The built-in EtherN restarted.	The built-in EtherNet/IP port was restarted.					
Attached information	None					,	
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	Tag Data Link All I	Run		Event code	94070000 hex		
Meaning	Tag data link conn	ections to all nodes I	have been establish	ned.			
Source	EtherNet/IP Function Module		Source details	CIP,*1 CIP1,*2 or CIP2*2	Detection timing	When establishing tag data link connection	
Error attributes	Level	Information	Recovery		Log category	System	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET F	RUN	EtherNet/IP NET	ERR	EtherNet/IP LINK/	ACT	
	Lights.						
System-defined	Variable		Data type		Name		
variables	_EIP_TDLinkAllRunSta*1		BOOL		All Tag Data Link Communications Status		
	_EIP1_TDLinkAllRunSta*2		BOOL		CIP Communications1 All Tag Data Link Communications Status		
	_EIP2_TDLinkAllRunSta*2		BOOL		CIP Communications2 All Tag Data Link Communications Status		
Cause and	Assumed cause		Correction		Prevention		
correction	Tag data link conn	ections to all target established.					
Attached information	None						
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	IP Address Fixed			Event code	94080000 hex		
Meaning	The correct IP add	dress has been deter	communications ca	n start.			
Source	EtherNet/IP Function Module		Source details	Communications port,*1 communi- cations port 1,*2 or Communica- tions port 2*2	Detection timing	At power ON or Controller reset	
Error attributes	Level	Information	Recovery		Log category	System	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET ERR		EtherNet/IP LINK/ACT		
	Lights.						
System-defined	Variable	Variable		Data type		Name	
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	The correct IP address has been determined and Ethernet communications can start.						
Attached information	Attached Informat	ion 1: IP address (ex	ample: C0A8FA01 h	ex = address 192.16	68.250.1)		
Precautions/ Remarks	None						

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	BOOTP Client Started			Event code	94090000 hex	
Meaning	The BOOTP client	started requesting a	ın IP address.			
Source	EtherNet/IP Function Module		Source details	Communications port,*1 communications port 1,*2 or Communications port 2*2	Detection timing	At power ON or Controller reset
Error attributes	Level	Information	Recovery		Log category	System
Effects	User program	Continues.	Operation	Not affected.		
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET ERR		EtherNet/IP LINK/ACT	
System-defined	Variable		Data type		Name	
variables	None					
Cause and	Assumed cause		Correction		Prevention	
correction	The BOOTP client started requesting an IP address.					
Attached information	None					
Precautions/ Remarks	None					

<sup>\*1</sup> This applies to an NJ-series CPU Unit.

<sup>\*2</sup> This applies to an NX-series CPU Unit.

Event name	FTP Server Started		Event code	940A0000 hex			
Meaning	The FTP agent st	arted normally.					
Source	EtherNet/IP Function Module Source details		Source details	FTP	Detection timing	At power ON or Controller reset	
Error attributes	Level	Information	Recovery		Log category	System	
Effects	User program	Continues.	Operation Not affected.		•		
Indicators	EtherNet/IP NET RUN		EtherNet/IP NET	EtherNet/IP NET ERR		EtherNet/IP LINK/ACT	
System-defined	Variable		Data type	Data type		Name	
variables	None						
Cause and	Assumed cause		Correction	Correction		Prevention	
correction	The FTP agent started normally.						
Attached information	None						
Precautions/ Remarks	None						

Event name	NTP Client Started		Event code	940B0000 hex		
Meaning	The NTP client sta	rted normally and a	request for the NTP	server to obtain the	time started.	
Source	EtherNet/IP Function Module Source details		Source details	NTP	Detection timing	At power ON or Controller reset
Error attributes	Level	Information	Recovery		Log category	System
Effects	User program	Continues.	Operation	Not affected.		
Indicators	ators EtherNet/IP NET RUN		EtherNet/IP NET E	RR	EtherNet/IP LINK//	ACT
System-defined	Variable		Data type		Name	
variables	None					
Cause and	Assumed cause		Correction		Prevention	
correction	The NTP client started normally and a request for the NTP server to obtain the time started.					
Attached information	None					
Precautions/ Remarks	None					

Event name	SNMP Started			Event code	940C 0000 hex		
Meaning	The SNMP agent	started normally.					
Source	EtherNet/IP Function Module Sou		Source details	SNMP	Detection timing	At power ON or Controller reset	
Error attributes	Level	Information	Recovery		Log category	System	
Effects	User program	Continues.	Operation	Not affected.			
Indicators	EtherNet/IP NET I	EtherNet/IP NET RUN		EtherNet/IP NET ERR		EtherNet/IP LINK/ACT	
System-defined	Variable		Data type		Name		
variables	None						
Cause and	Assumed cause		Correction		Prevention		
correction	The SNMP agent started normally.						
Attached information	None	None					
Precautions/ Remarks	None						

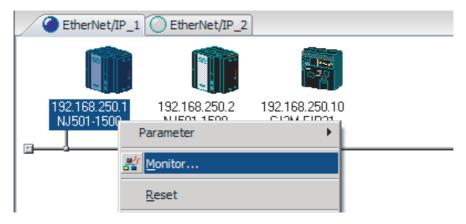
### 15-2-3 Troubleshooting

Problem	Correction
Tag data is not concurrent.	Check the following items and correct the user program.
	<ul> <li>Data concurrency is maintained for each connection between the CPU Unit and the built-in EtherNet/IP port. To maintain data concurrency for tag data links, set a refreshing task for the network variables that are assigned to tags. Refer to 7-1-7 Concurrency of Tag Data Link Data for details.</li> <li>Refer to the product manuals for products from other manufacturers.</li> </ul>
At startup, some of the receive data is FALSE when it should be TRUE.	<ul> <li>If the user program uses receive data, make sure that the All Tag Data Link Communications Status in communications status 1 or the Controller Operating Mode for the target node is TRUE before you use the receive data. To use operation information from the Controller, use Controller status in the tag sets on both the sending and receiving nodes.</li> <li>If the Fault Action setting of the output (produce) tag is enabled, The output (produce) data changes to FALSE when a fatal error occurs in the CPU Unit. Check the error status at the output (producing) Controller.</li> </ul>
Tag data link communications are not stable.	<ul> <li>For an NX-series CPU Unit, use a 1,000 Mbps Ethernet switch if 10 or 100 Mbps is set or if you are using a 10 Mbps repeater hub, a 100 Mbps repeater hub, or a 1,000 Mbps repeater hub. The performance of the tag data links assumes that an Ethernet switch is used to achieve a 40,000-pps bandwidth for full-duplex, 1,000-Mbps auto-negotiation communications.</li> <li>For an NJ-series CPU Unit, use a 100 Mbps Ethernet switch if 100 Mbps is set or if you are using a 10 Mbps or 100 Mbps repeater hub. For unit version 1.03 or later, the performance of the tag data links assumes that an Ethernet switch is used to achieve a 3,000-pps bandwidth for full-duplex, 100-Mbps auto-negotiation communications. For unit version 1.00 to 1.02, a 1,000-pps bandwidth is assumed.</li> <li>Refer to 15-3 Checking Status with the Network Configurator to access the error counters and discarded packed counters on the Ethernet Information Tab Page. Use the information to check for noise on the communications path, non-standard cables, damaged cables/connectors, unexpectedly high communications traffic, and incorrect loops in connections between Ethernet switches.</li> <li>Contact the Ethernet switch manufacturer if there are problems with the transfer capacity of the Ethernet switches in the communications path. If Ethernet switches are cascaded, the load may be concentrated on the middle Ethernet switches. Change the network configuration so that the load is not concentrated.</li> <li>Also, refer to 15-3 Checking Status with the Network Configurator and use the connection status on the Connections Tab Page to remove the cause of the error according to information in 15-3-2 Connection Status Codes and Troubleshooting.</li> </ul>

# 15-3 Checking Status with the Network Configurator

### 15-3-1 The Network Configurator's Device Monitor Function

Connect the Network Configurator online, select the device to be checked, right-click to display the popup menu, and select *Monitor*.



The Monitor Device Dialog Box will be displayed.



#### **Additional Information**

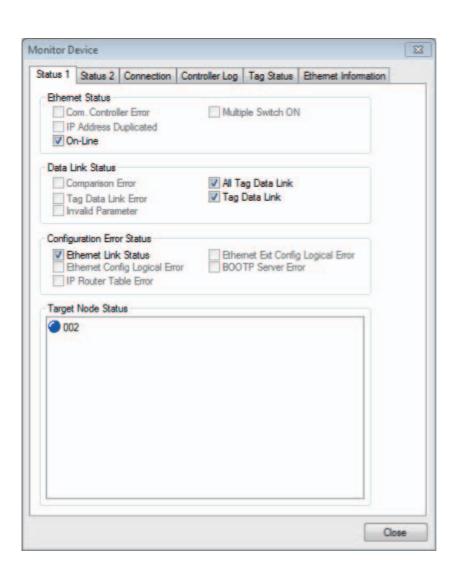
If a communications error occurs during monitoring, the dialog box will continue to show the last information that was collected. To start monitoring again, close the Monitor Device Dialog Box, and then open the dialog box again.

### Status 1 Tab Page

The following check boxes are displayed for the status. If a check box is selected, the status is TRUE.

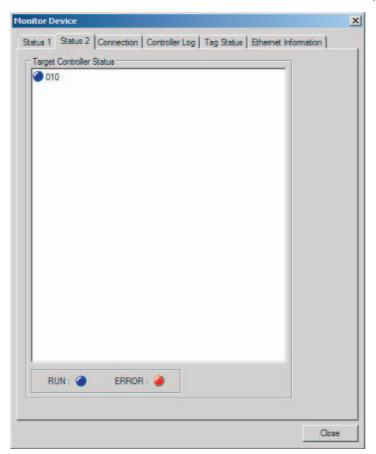
Classification	Item	Corresponding system-defined variable		
Ethernet Status	Com. Controller Error	An error occurred in the communications controller.		
	IP Address Duplicated	The same IP address is assigned to more than one node.		
	On-Line	Indicates that the Unit is online. (The EtherNet/IP Unit can perform communications processing.)		
	Multiple Switch ON	Indicates that more than one data link start/stop switch changed to TRUE at the same time.		
Data Link Status	Comparison Error	The remote node information in the tag data link parameters was different from the actual node information.		
		Main causes: The specified target does not exist.  The variable name does not match.  The connection size is different.  Connection resources are not sufficient.		
	Tag Data Link Error	There were two or more errors in a connection as an originator.		
	Invalid Parameter	An error was found in the validation check of the parameters for tag data links that are saved in non-volatile memory.		
	All Tag Data Links	Tag data links are communicating in all connections as the originator.		
	Tag Data Link	Tag data links are communicating in one or more connections as the originator.		
Configuration Error	Ethernet Link Status	TRUE when a link is established with the Ethernet switch.		
Status	Ethernet Basic Settings Logic	TRUE when the following settings are incorrect:		
	Error	TCP/IP settings (IP address, subnet mask, or link settings)		
	IP Router Table Error	TRUE when there is a mistake in the IP router table information.		
	Ethernet Ext Config Logical Error	Always FALSE.		
	BOOTP Server Error	TRUE when one of the following errors occurs when using the BOOTP server.		
		<ul> <li>The IP address received from the BOOTP server is incorrect.</li> <li>A communications timeout occurred with the server.</li> </ul>		

Information about the target node that acts as the originator is displayed. If all tag data link connections to the node are established and normal, this information is displayed in blue. However, if any connection is broken it is displayed in red.



### Status 2 Tab Page

This tab page displays information on nodes with tag data link originator settings based on \_EIP\_TargetPLCModeSta (Target PLC Operating Mode)\*1 and \_EIP\_TargetPLCErr (Target PLC Error Information)\*1 in the EIP EstbTargetSta (Normal Target Node Information)\*1 system-defined variable. This information is in blue if the connection is normal, or red if an error occurred.



- \*1 These are the system-defined variables for an NJ-series CPU Unit. The system-defined variables for an NXseries CPU Unit are given below.
  - CIP Communications 1:
  - \_EIP1\_EstbTargetSta (CIP Communications 1 Normal Target Node Information)
  - \_EIP1\_TargetPLCModeSta (CIP Communications 1 Target PLC Operating Mode)
  - \_EIP1\_TargetPLCErr (CIP Communications 1 Target PLC Error Information)
  - CIP Communications 2:
  - \_EIP2\_EstbTargetSta (CIP Communications 2 Normal Target Node Information)
  - \_EIP2\_TargetPLCModeSta (CIP Communications 2 Target PLC Operating Mode)
  - \_EIP2\_TargetPLCErr (CIP Communications 2 Target PLC Error Information)

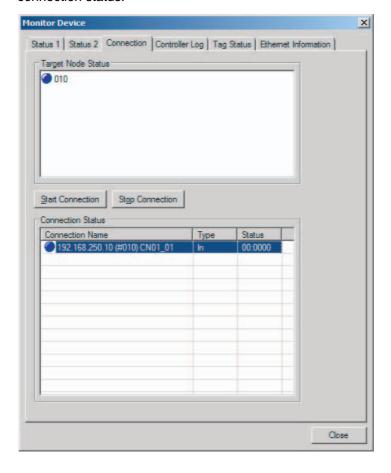


#### **Additional Information**

The target Controller status can be used when the Controller status is selected for all the target sets for both originator and target connections. If it is not selected, it is grayed out on the display.

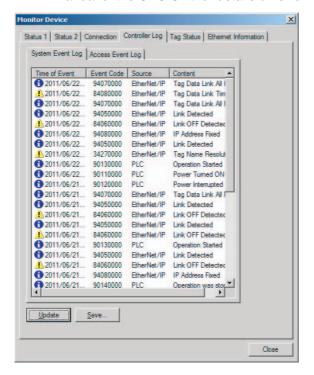
### Connection Tab Page

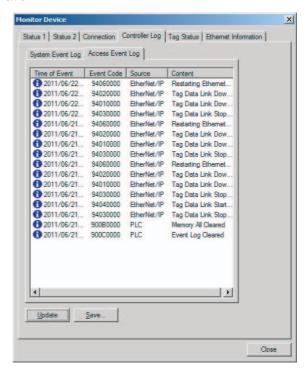
Information about the target node that acts as the originator is displayed. If all tag data link connections to the node are established and normal, this information is displayed in blue. However, if any connection is broken it is displayed in red. However, this information is displayed in gray if the connection to the node is stopped. In addition, the *Connection Status* Area shows the current status of each connection that is set as the originator. This information can be used to identify the cause of tag data link errors. Refer to 15-3-2 Connection Status Codes and Troubleshooting for details on the connection status.



### Controller Event Log Tab Page

This tab page displays the Controller event log that is stored in the CPU Unit. The error history shows errors that have occurred. It can be saved in a file in the computer. Refer to the operation manual of the CPU Unit for details on error information.





### Tag Status Tab Page

This tab page displays if the tag settings for each tag for tag data links are set so that data can be exchanged with the CPU Unit. The following status is displayed depending on the status that is set.

• Normal resolution completed: Normal data exchange is possible.

Resolving: The variables with tags are being resolved. When the

resolution is completed normally, a connection will be

established and the data exchange will start.

• Size does not match error: Different sizes are set for the network variables and the

tag settings. A connection will not be established for a

tag for which this error occurs.

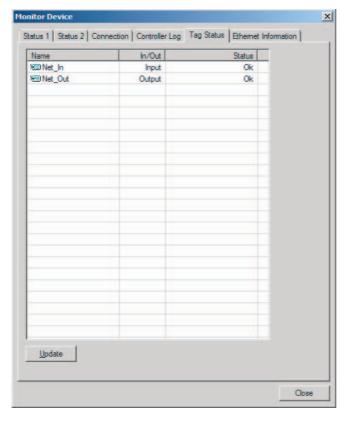
• No tag: A network variable is not set in the variable table in the

CPU Unit for the specified tag setting. A connection will not be established for a tag for which this error occurs.

Attribute error: Writing is not possible for Read Only and Constant attri-

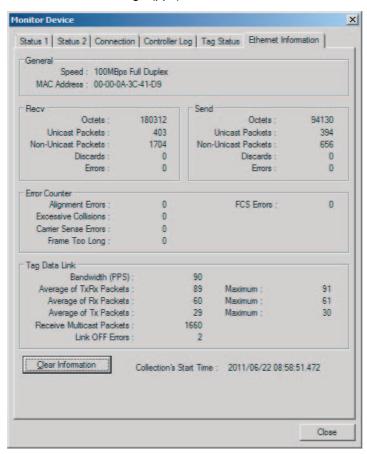
butes.

If the status is not "Normal resolution completed," check the tag data link settings or the network variable settings in the variable table in the NJ/NX-series CPU Unit.



### Ethernet Information Tab Page

This tab page displays the communications status at the communications driver level of the built-in Ethernet/IP port. The error counter information can be used to confirm whether communications problems have occurred. The tag data link information can be used to confirm characteristics such as the bandwidth usage (pps).



### 15-3-2 Connection Status Codes and Troubleshooting

This section explains how to identify and correct errors based on the tag data link's connection status. The connection status can be read using the Connection Tab Page of the Network Configurator's Monitor Device Window. Refer to 15-3-1 The Network Configurator's Device Monitor Function for details.



### **Additional Information**

The connection status has the same meaning as the Connection Manager's General and Additional error response codes, as defined in the CIP specifications.

The following table shows the likely causes of the errors causes for each configuration and connection status (code).

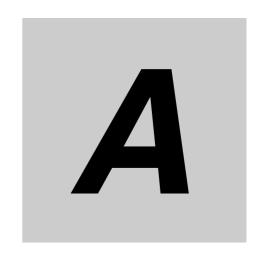
	Originator	Target
Configuration 1	CS1W-EIP21, CJ1W-EIP21, CJ2H-CPU□□- EIP, CJ2M-CPU3□, NJ501-□□□, NJ301- □□□□, NJ101-□□□□, and NX701- □□□□	CS1W-EIP21, CJ1W-EIP21, CJ2H- CPU□□-EIP, CJ2M-CPU3□, NJ501- □□□□, NJ301-□□□□, NJ101-□□□□, and NX701-□□□□
Configuration 2	CS1W-EIP21, CJ1W-EIP21, CJ2H-CPU	Products from other manufacturers
Configuration 3	Products from other manufacturers	CS1W-EIP21, CJ1W-EIP21, CJ2H- CPU□□-EIP, CJ2M-CPU3□, NJ501- □□□□, NJ301-□□□□, NJ101-□□□□, and NX701-□□□□

Connection status			Handling			
General Status (hex)	Additional Status (hex)	Source of error	Configuration 1	Configuration 2	Configuration 3	
00	0000	Normal status code: The connection has been opened and the tag data link is communicating normally.				
01	0100	Error code returned from target: Attempted to open multiple connections for the same connection.	This error does not occur.	Depends on the target's specifications. (This error should not occur. If is does, contact the target device's manufacturer.)	Depends on the originator's specifications. (This error should not occur. If is does, contact the originator device's manufacturer.)	
01	0103	Error code returned from target: Attempted to open a connection with an unsupported transport class.	This error does not occur.	Confirm that the target supports Class 1.	Confirm that the originator supports Class 1.	
01	0106	Duplicate consumers: Attempted to open multiple connections for single-consumer data.	If the tag data link is stopped or started, this error may occur according to the tim- ing, but the system will recover automat- ically.	Depends on the target's specifications. (Contact the target device's manufacturer.)	If the tag data link is stopped or started, this error may occur according to the tim- ing, but the system will recover automatically.	
01	0107	Error code returned from target: Attempted to close a connection, but that connection was already closed.	This error does not occur.	This error does not occur.	This is not an error because the connection is already closed.	
01	0108	Error code returned from target: Attempted to open a connection with an unsupported connection type.	This error does not occur.	Check which connection types can be used by the target. (Contact the manufacturer.) Only multicast and point-topoint connections can be set.	Check which connection types can be used by the originator. (An error will occur if a connection other than a multicast or point-topoint connection is set.)	
01	0109	Error code returned from target: The connection size settings are dif- ferent in the originator and target.	Check the connection	sizes set in the originator	and target.	
01	0110	Error code returned from target: The target was unable to open the connection, because of its operating status, such as downloading settings.	Check whether the tag data link is stopped at the target. (Restart the tag data link communications with the software switch.)	Depends on the target's specifications. (Contact the target device's manufacturer.)	Check whether the tag data link is stopped at the target. (Restart the tag data link communications with the software switch.)	
01	0111	Error code returned from target: The RPI was set to a value that exceeds the specifications.	This error does not occur.	Check the target's RPI setting specifications.	Set the originator's RPI setting to 10 seconds or less.	

Connec	tion status		Handling				
General Status (hex)	Additional Status (hex)	Source of error	Configuration 1	Configuration 2	Configuration 3		
01	0113	Error code generated by originator or returned from target: Attempted to open more connections than allowed by the specifications (32).	Check the connection settings (number of connections) at the originator and target.	Check the connection settings (number of connections) at the originator and target. Check the connection specifications for devices from other manufacturers.	Check the connection settings (number of connections) at the originator and target. Check the connection specifications for devices from other manufacturers.		
01	0114	Error code returned from target: The Vendor ID and Product Code did not match when opening con- nection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.) Check that the target device's EDS file is correct.	Check the originator's connection settings.		
01	0115	Error code returned from target: The Product Type did not match when opening connection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.) Check that the target device's EDS file is correct.	Check the originator's connection settings.		
01	0116	Error code returned from target: The Major/Minor Revisions did not match when opening connection.	Check the major and minor revisions set for the target device and connection. If necessary, obtain the most recent EDS file and set it again.	Depends on the target's specifications. (Contact the target device's manufacturer.) Check that the target device's EDS file is correct.	Check the originator's connection settings.		
01	0117	Error code returned from target: The tag set specified in the connection's target variables does not exist.	Check whether the originator and target tag sets and tags are set correctly.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Check the originator's connection settings. Check whether the target tag sets and tags are set correctly.		
01	011A	Error code generated by originator: Connection could not be established because the buffer was full due to high traffic.	Unexpected network traffic may have been received. Use the Network Configurator Device Monitor or the Ethernet Tab Page to check the bandwidth usage, and correct the load. If there are places where broadcast storms occur, such as loop connections in the network connection format, then correct them.	Unexpected network traffic may have been received. Use the Network Configurator Device Monitor or the Ethernet Tab Page to check the bandwidth usage, and correct the load. If there are places where broadcast storms occur, such as loop connections in the network connection format, then correct them.	Depends on the target's specifications. (Contact the target device's manufacturer.)		
01	011B	Error code returned from target: The RPI was set to a value that is below the specifications.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Set the originator's RPI setting to 1 ms or greater.		
01	0203	Error code generated by originator: The connection timed out.	Tag data link communications from the target timed out. Check the power supply and cable wiring of the devices in the communications path, including the target and switches. If performance has dropped due to heavy traffic, change the performance settings. For example, increase the timeout time or RPI setting.				
01	0204	Error code generated by originator: The connection open process timed out.	I	se from the target. Check t vices in the communication			

Connec	tion status		Handling				
General Status (hex)	Additional Status (hex)	Source of error	Configuration 1	Configuration 2	Configuration 3		
01	0205	Error code returned from target: There was a parameter error in the frame used to open the connection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
01	0302	Error code generated by originator or returned from target: The tag data link's allowable bandwidth (pps) was exceeded.	Check the connection settings (number of connections and RPI) at the originator and target.	Check the target's con- nection settings (num- ber of connections and RPI). Check the con- nection settings (num- ber of connections and RPI) at the originator and target.	Check the connection settings (number of connections and RPI) at the originator and target.		
01	0311	Error code returned from target: There was a parameter error in the frame used to open the connection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
01	0312	Error code returned from target: There was a parameter error in the frame used to open the connection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
01	0315	Error code returned from target: There was a parameter error in the frame used to open the connection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
01	0316	Error code returned from target: There was a parameter error in the frame used to close the connection.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
01	031C	Error code generated by originator: Some other error occurred.	This error does not occur.	The originator generates this code when an unsupported response code is returned from the target in reply to an open request.	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
08		Error code returned from target: There is no Forward Open or Large Forward Open service in the target device.	This error does not occur.	Depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		
DO	0001	Error code generated by originator: The connection operation is stopped.	The connection was stopped because the Tag Data Link Stop Bit was turned ON, or the settings data is being downloaded. Either turn ON the Tag Data Link Start Switch, or wait until the settings data has been downloaded. This code includes fatal Controller errors and Unit failure. To handle these errors, refer to 15-1-1 How to Check for Errors.	The meaning of this error code is defined by each vendor, so it depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the originator's specifications. (Contact the originator device's manufacturer.)		
D0	0002	Error code generated by originator: The connection is being opened (opening processing in progress).	Wait until the opening processing is completed.	The meaning of this error code is defined by each vendor, so it depends on the target's specifications. (Contact the target device's manufacturer.)	Depends on the origi- nator's specifications. (Contact the originator device's manufacturer.)		

Connec	tion status			Handling	
General Status (hex)	Additional Status (hex)	Source of error	Configuration 1	Configuration 2	Configuration 3
OMRON e	rror code				
01	0810	Error code returned from target: New data could not be obtained from the CPU Unit when opening connec- tion. (The Unit will automatically recover, and attempt to open the connection again.)	This error may occur if the CPU Unit's task period was long when opening the connection or some problem in the Controller caused the Controller to stop. If the task period was too long, operation recovers automatically. If the Controller has stopped, identify the error from the error information in the CPU Unit.	The meaning of this error code is defined by each vendor, so it depends on the target's specifications. (Contact the target device's manufacturer.)	The meaning of this error code is defined by each vendor, so it depends on the originator's specifications. (Contact the originator device's manufacturer.)
01	0811	Error code generated by originator: New data could not be obtained from the CPU Unit when opening connec- tion. (The Unit will automatically recover, and attempt to open the connection again.)	This error may occur if the CPU Unit's task period was long when opening the connection. If the task period was too long, operation recovers automatically.	The meaning of this error code is defined by each vendor, so it depends on the target's specifications. (Contact the target device's manufacturer.)	The meaning of this error code is defined by each vendor, so it depends on the originator's specifications. (Contact the originator device's manufacturer.)



# **Appendices**

<b>A-1</b>		onal Comparison of EtherNet/IP Ports on NJ/NX- CPU Units and Other Series	A-2
<b>A-2</b>		e Sysmac Studio to Set the Tag Data Links Net/IP Connections)	Λ 2
	A-2-1	Overview of the Tag Data Links (EtherNet/IP Connections) Settings with Sysmac Studio	the
	A-2-2	Procedure to Make the EtherNet/IP Connection Settings with the Sysmac Studio	
	A-2-3	EtherNet/IP Connection Settings	
	A-2-4	Making the EtherNet/IP Connection Settings with the Sysmac Studio	
	A-2-5	Checking Communications Status with the Sysmac Studio and Troubleshooting	
	A-2-6	Troubleshooting	
		•	
A-3		ile Management	
	A-3-1	Installing EDS Files	
	A-3-2	Creating EDS Files	
	A-3-3	Deleting EDS Files	
	A-3-4	Saving EDS Files	
	A-3-5	Searching EDS Files	
	A-3-6	Displaying EDS File Properties	
	A-3-7	Creating EDS Index Files	A-48
A-4		Itions for Using the Network Configurator on Windows XP,	A 40
		ws Vista, or Windows 7	
	A-4-1	Changing Windows Firewall Settings	
A-5		le Memory Allocation Methods	
	A-5-1	Variable Memory Allocation Rules	
	A-5-2	Important Case Examples	A-60
<b>A-6</b>	Precau	itions When Accessing External Outputs in CPU Units	<b>A-6</b> 4
<b>A-7</b>	TCP St	ate Transitions	A-65
<b>A-8</b>	Versio	n Information	A-67

### A-1 Functional Comparison of EtherNet/IP Ports on NJ/NX-series CPU Units and Other Series

OK: Supported., ---: Not supported.

Item	Built-in Ether- Net/IP port on NX-series CPU Unit	Built-in Ether- Net/IP port on NJ-series CPU Unit	CJ-series Ethernet Unit	EtherNet/IP Unit (built-in port on CJ2 CPU Unit)		
				Unit ver- sion 1.0	Unit ver- sion 2.0	Unit ver- sion 2.1
Tag data link communications service	OK	OK		OK	OK	OK
CIP message communications service	OK	OK		OK	OK	OK
IP routing	OK					
Socket service	OK	OK	OK			
FTP server	OK	OK	OK		OK	OK
FTP client	OK	OK				
Mail send/receive			OK			
Web functions			OK			
Automatic adjustment of PLC/Controller's internal clock	OK	OK	OK		OK	OK
Error history	OK <sup>*1</sup>	OK <sup>*1</sup>	OK	OK	OK	OK
Response to PING command	OK	OK	OK	OK	OK	OK
SNMP/SNMP trap	OK	OK			OK	OK
CIDR function for IP addresses	OK	OK			OK	OK
Online connection via EtherNet/IP using CX-One			OK		OK	OK
Online connection via EtherNet/IP using Network Configurator	OK	OK		OK	OK	OK
Mounting in an NJ-series CPU Unit						OK <sup>*2</sup>
Connection settings using the Sysmac Studio	OK	OK				OK
Mounting in an NX-series CPU Unit						

<sup>\*1</sup> This is equivalent to the event log in the built-in EtherNet/IP of an NJ-series Controller.

<sup>\*2</sup> You cannot use the following functions if you connect to the CPU Unit through an EtherNet/IP Unit.

Placing the Sysmac Studio online with the CPU Unit (However, you can place the Network Configurator online.)

<sup>•</sup> Using the Troubleshooter of an NS-series PT

# A-2 Use the Sysmac Studio to Set the Tag Data Links (EtherNet/IP Connections)

# A-2-1 Overview of the Tag Data Links (EtherNet/IP Connections) Settings with the Sysmac Studio

You can use the Sysmac Studio to set the settings required for creating tag data links (EtherNet/IP connections)\*1 between NJ/NX-series Controllers.

\*1 The tag data links and EtherNet/IP connections enable cyclic tag data exchanges on an EtherNet/IP network between Controllers or between Controllers and other devices. Here, "EtherNet/IP connection" refers to both the tag data links and the EtherNet/IP connections.



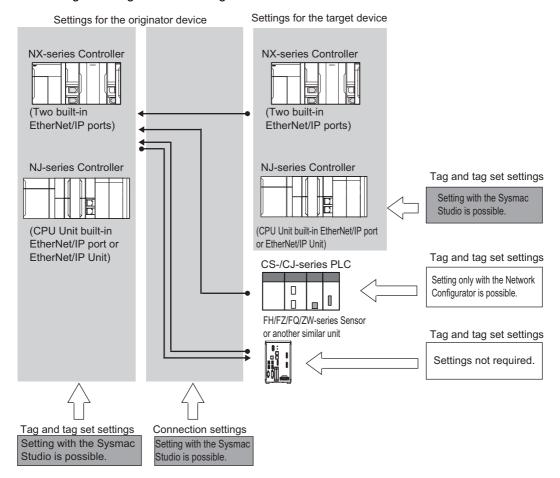
#### **Version Information**

Sysmac Studio version 1.10 or higher is required to use the Tag Data Link (EtherNet/IP Connection) Settings.

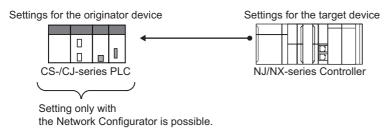
# Acceptable System Configuration Conditions for Setting the EtherNet/IP Connection Settings on the Sysmac Studio

If an NJ/NX-series Controller operates as the originator device, you can use the Sysmac Studio to set the originator device settings for the EtherNet/IP connections.

Similarly, if an NJ/NX-series Controller operates as the target device, you can use the Sysmac Studio to set the tags and tag sets of the target device.



Use the Network Configurator if a CS-/CJ-series PLC operates as the originator device.



# A-2-2 Procedure to Make the EtherNet/IP Connection Settings with the Sysmac Studio

**1** Registering devices

Register devices with which the EtherNet/IP connections are established to the project.

• Main Window

**2** Creating network variables(\*1)

. Refer to Registering the Network Variable for the Originator Device on page A-12. Setup Window

Global Variable Table on the Sysmac Studio

**3** Registering tags and tag sets

. Refer to Registering the Tag and Tag Set on page A-14.

EtherNet/IP Connection Settings (Tag Set Display)

Register the network variables that are set in step 2 as tags and tag sets.

**4** Setting Connections

... Refer to Setting Connections for the Originator Device on page A-17.

EtherNet/IP Connection Settings (Connections Display)

Specify devices (i.e., target devices and originator devices) and tag sets to communicate with using the EtherNet/IP connections.

- **5** Going online from the Sysmac Studio
- .. Refer to *Transferring the Connection Settings Data* on page A-29.
- Main Window

**6** Downloading EtherNet/IP connection settings

**Note** Connections automatically start after the download.

- . Refer to *Transferring the Connection Settings Data* on page A-29.
- Synchronization Window / Transfer to Controller Dialog Box
- EtherNet/IP Connection Settings

- Checking operationStopping and starting connections
- Refer to A-2-5 Checking Communications Status with the Sysmac Studio and Troubleshooting on page A-34.

EtherNet/IP Connection Monitor Tab Page

<sup>\*1</sup> Variables with its Network Publish attribute set to *Output* or *Input* in the Global Variable Table are called network variables.

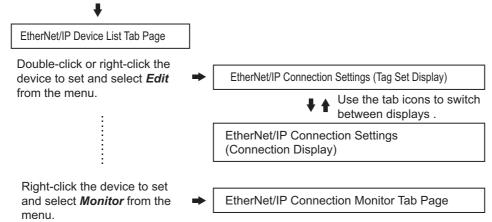
### A-2-3 EtherNet/IP Connection Settings

This section describes the screen configuration for EtherNet/IP connection settings.

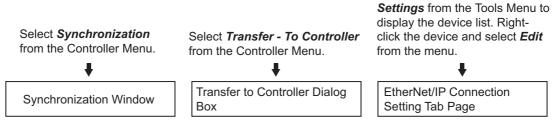
### Screen Transitions in the EtherNet/IP Connection Settings

• Connection Settings

Select EtherNet/IP Connection Settings from the Tools Menu.



· Transferring connection settings to the Controller from the computer





### **Precautions for Correct Use**

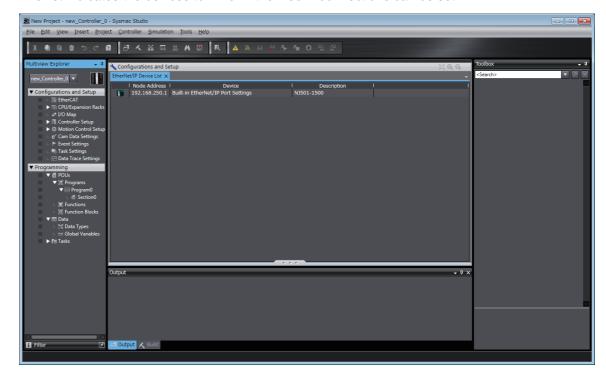
To transfer only the connection settings, execute Transfer from the EtherNet/IP Connection Setting Tab Page.

The connection settings are not transferred from the Synchronization Window, the Transfer to Controller Dialog Box, and the Transfer from Controller Dialog Box, even if you clear the *Do not transfer the connection setting* Check Box, as long as the data in the computer and in the Controller is the same.

Select EtherNet/IP Connection

## ● EtherNet/IP Device List Tab Page (Refer to *Registering the Tag and Tag Set* on page A-14.)

The list indicates the devices to which EtherNet/IP connections can be set.



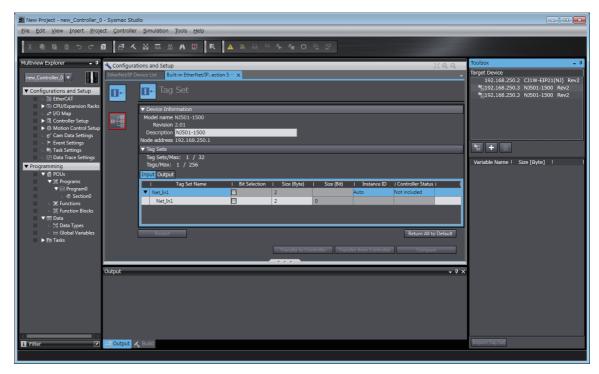
### • EtherNet/IP Connection Settings (Tag Set Display) (Refer to *Registering the Tag and Tag Set* on page A-14.)

Register tag sets required to create connections. Each tag set represents the data that is sent and received through a connection. You can register up to eight tags in one tag set.

The name and size of the tag must be the same as those of the network variable \*1.

Set whether to include the Controller status information in tags for the tag sets. You can also set the data output operation at a fatal error occurrence for output tags.

\*1 Variables with its Network Publish attribute set to *Output* or *Input* in the Global Variable Table are called network variables.



### EtherNet/IP Connection Settings (Connection Display) (Refer to Setting Connections for the Originator Device on page A-17.)

Specify the target devices and set their connections.

For each connection, set the following information: Connection Name, Connection I/O Type, I/O, target device tag set (target variable), originator device tag set (originator variable), Packet Interval (RPI), and Timeout Value.

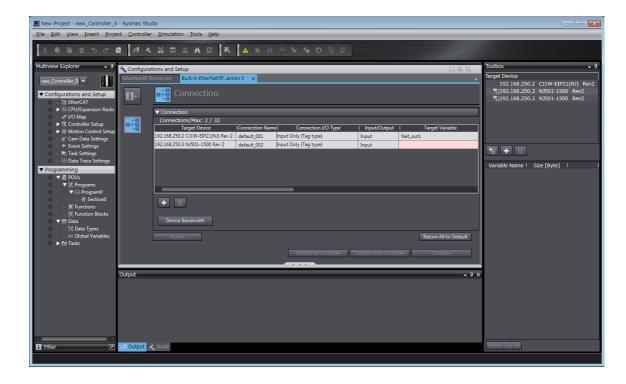


#### **Precautions for Correct Use**

If you changed the IP address, model or revision of the target device after making the connection settings, perform the following.

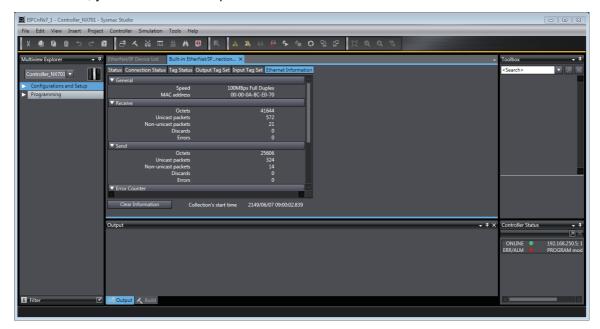
With the Sysmac Studio version 1.11 or higher, change the connection settings entirely.

With the Sysmac Studio version 1.10 or lower, create the connections again.



• EtherNet/IP Connection Monitor Tab Page (Refer to A-2-5 Checking Communications Status with the Sysmac Studio and Troubleshooting on page A-34.)

You can check the EtherNet/IP connection setting status offline and communications status online. When online, you can start and stop connections.



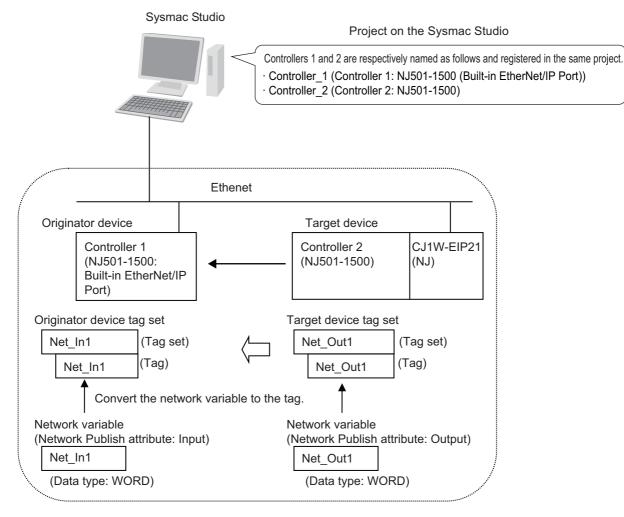
## A-2-4 Making the EtherNet/IP Connection Settings with the Sysmac Studio

This section describes the procedure to make the EtherNet/IP connection settings with the Sysmac Studio.

Here, we take the following system configuration as an example to describe how to set the EtherNet/IP connection settings.

Example: System that connects the built-in EtherNet/IP port on Controller 1 and the built-in EtherNet/IP port on Controller 2 via Ethernet

- Set the settings so that values in the network variable *Net\_Out1* allocated for Controller 2 are sent to the network variable *Net\_In1* allocated for Controller 1 at the set RPI of 50 ms cycle.
- This example assumes the programs for both Controllers 1 and 2 are registered in the same project.

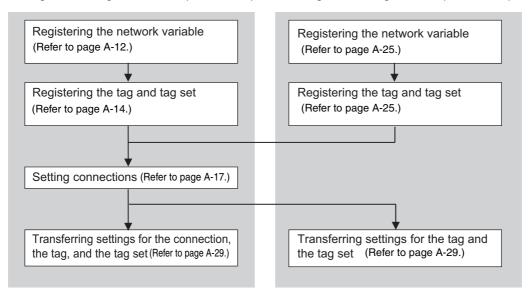


Follow the flow below to set the settings to Controllers 1 and 2 for which to establish EtherNet/IP connections.

The required settings for the originator device and the target device are shown below.

Settings for the originator device (Controller 1)

Settings for the target device (Controller 2)



### Registering the Network Variable for the Originator Device

Register the network variable that is sent and received using the EtherNet/IP connections.

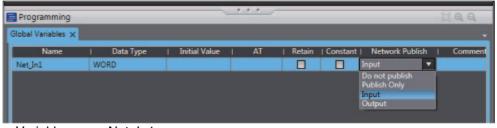
Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No. W504) for the operations for registering variables.

Assign the network variable to the tag used for the EtherNet/IP connection for Controller 1 (originator device).

This network variable receives data from Controller 2 (target device).

Set the Network Publish attribute to *Input* or *Output* in the Global Variable Table for the variable so that the variable serves as a network variable, i.e., the variable can be used for the Ether-Net/IP connections.

In this example, set the network variable for Controller 1 as shown below.



Variable name: Net\_In1

Data type: WORD

Network Publish attribute: Input

#### Network Variables Used for the EtherNet/IP Connections

· Network variable name

You cannot specify an I/O memory address for a tag name in the EtherNet/IP connection settings. Thus, do not specify an I/O memory address for the network variable name that is to be assigned to a tag.

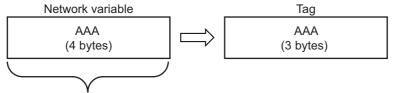
The following text strings are recognized as the I/O memory address names.

- (1) Variable names that contain only single-byte numerals from 0000 to 6143
- (2) Variable names with the following single-byte letters (uppercase or lowercase) followed by single-byte numerals
  - H (H000 to H511)
  - W (W000 to W511)
  - D (D00000 to D32767)
  - E0 to E18 (E0 00000 to E0 32767, to E18 00000 to E18 32767)

To specify an I/O memory address in the NJ-series CPU Unit for a tag, do not specify the I/O memory address for the tag directly. Instead, create a variable, set an AT specification of the I/O memory address on the Sysmac Studio, and then specify the variable with the AT specification for the tag.

#### · Size of variables

To use an EtherNet/IP Unit as an EtherNet/IP device, set an even number of bytes for the size of the network variable used for the EtherNet/IP connections regardless of an odd number of bytes for the tag size.



The CPU Unit memory is consumed in units of two bytes. To assign tags of odd numbers of bytes to network variables, specify even byte numbers (i.e., sizes of the tags + 1) to the network variables.

#### Data concurrency

To maintain concurrency in the values of network variables that are assigned to tags, you must set refreshing tasks.

Refer to 7-1-7 Concurrency of Tag Data Link Data for details.

### **Registering the Tag and Tag Set**

Register the required tag and tag set for the EtherNet/IP connections.

You can register tags and tag sets in the EtherNet/IP Connection Setting Tab Page.



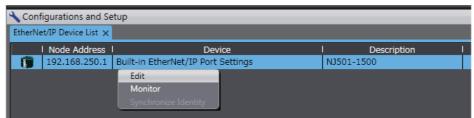
#### **Precautions for Correct Use**

Make the following settings to refresh all of the tag data in the same tag set at the same time.

- Use the Sysmac Studio, in advance, to specify the same refreshing task for all of the variables that are assigned to tags in the tag set.
- If you use the NJ-series CPU Unit, do not place tag variables that have AT specifications in I/O memory and tag variables that do not have AT specifications in the same tag set.
- 1 Select EtherNet/IP Connection Settings from the Tools Menu.

The EtherNet/IP Device List Tab Page is displayed.

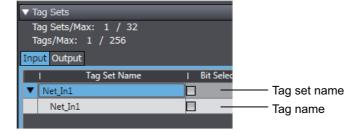
2 In this example, right click *Built-in EtherNet/IP Port Settings* for the originator device and select *Edit* from the menu to open the EtherNet/IP Connection Setting Tab Page.



- Click the [1] (Show Tag Set Display) icon in the EtherNet/IP Connection Setting Tab Page.
- 4 Click the **Input** tab to switch to the Input Tab Page. Register the tag set and the tag.

Use one of the following methods to register the tag set and the tag.

- Independent registration : Manually registers network variables in the Controller
  - as tags.
- Batch registration : Registers all network variables in the Controller as tags at the same time.
- 5 Register tags and tag sets independently.
  - (1) Right-click anywhere in the Input Tab Page of the EtherNet/IP Connection Setting Tab Page and select *Create New Tag Set* from the menu.
  - (2) Enter tag set name Net\_In1 directly into the list in the Input Tab Page.
  - (3) Right-click anywhere in the Input Tab Page and select *Create New Tag* from the menu.
  - (4) Enter tag name Net\_In1.





#### **Precautions for Correct Use**

Any name can be specified for the tag set if the name matches one of the registered network variable names in the Controller.

As you enter characters (or immediately after you press the Ctrl + Space Keys), the Sysmac Studio Entry Assistance provides a list of variable names registered in the Controller. Select the variable name from the list.



#### **Additional Information**

You can register up to 8 tags in a tag set.

Set as shown below to register multiple tags.

#### Example:

	Tag set name	
▼	Network_Input_Value	(Tag set name)
	Net_In1	(Tag name)
	Net_In2	(Tag name)

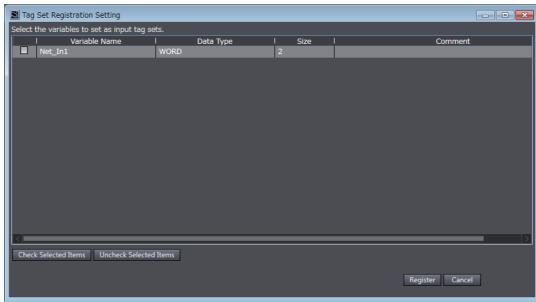
6

Register all tags and tag sets at the same time.

(1) Right-click anywhere in the Input Tab Page of the EtherNet/IP Connection Setting Tab Page and select *Register All Tag Sets* to display the Tag Set Registration Setting Dialog Box.

This dialog box lists the following variables that are registered in the Global Variable Table.

- When registering all tag sets in the Input Tab Page: lists all variables that are published to the network as inputs.
- When registering all tag sets in the Output Tab Page: lists all variables that are published to the network as outputs.



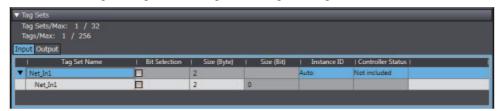
(2) Select the variable to register as a tag, and then click the Register Button.

# (3) The automatically registered tag is added to the list in the EtherNet/IP Connection Setting Tab Page.

With automatic registration, the tag is registered under a tag set having the same name as the tag, i.e., a single tag is registered in a single tag set.



7 Set the following settings for the registered tag and tag set.



#### · Setting for Tag Sets

Name	ltem
Tag Set Name	Enter the tag set name.
	You can change the names as required.
Size (Byte)	Gives the total size of the tag in bytes.
Instance ID	Gives the instance ID.
	Auto
	• IN_{min}IN_{max}
	{min} represents the minimum number of Produced Assembly identification numbers recorded in the EDS files for the relevant devices.
	{max} represents the maximum number of Produced Assembly identification numbers recorded in the EDS files for the relevant devices.
Controller Status	Specify whether to include the Controller status in the tag set.

#### · Setting for Tags

Name	Item
Tag Name	Enter the tag name.
	Specify the tag name that matches one of the registered network variable names in the Controller.
Bit Selection	Specify whether to set the tag data size in bits.
	Selected: Set the size in bits.
	Not selected: Set the size in bytes.
Size (Byte)	Gives the size of the tag in bytes.
Size (Bit)	Gives the size of the tag in bits.
Output at Fatal Error	Specify whether to clear the output data or continue to send it when a major fault level Controller error occurs in the Controller.
	Retained
	Cleared

# **Setting Connections for the Originator Device**

After the tag set registration, set the connection settings for transferring data using the EtherNet/IP connections.

Make the connection settings in the originator device (i.e., Controller 1 in this example) only.

Register the tag and tag set for Controller 2 (Target device) before setting the connection settings as described in this example.

Refer to Registering the Tag and Tag Set for the Target Device on page A-25 for the operations for registering tags and tag sets.



#### **Precautions for Correct Use**

If you change the IP address, model, or revision of the target device after making the connection settings, you must also change the target device settings that are included in the connection settings. For information on how to change the target device settings in the connection settings, refer to *Changing the Target Device Settings after Making Connection Settings*.

- 1 Select *EtherNet/IP Connection Settings* from the Tools Menu to display the EtherNet/IP Device List Tab Page.
- **2** Right click *Built-in EtherNet/IP Port Settings* for Controller 1 (originator device in this example) and select *Edit* from the menu.

The EtherNet/IP Connection Setting Tab Page is displayed.

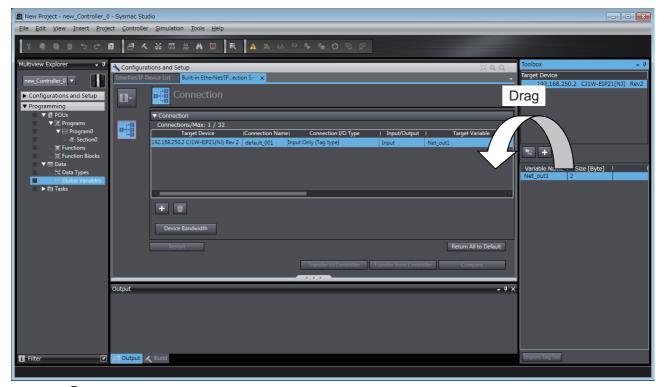


- 3 Click the (Show Connection Display) icon in the EtherNet/IP Connection Setting Tab Page.
- 4 Select CJ1W-EIP21(NJ) from Target Device in the Toolbox on the right of the tab page.

This operation displays the target device tag set *Net\_Out1* that is set for Controller 2 in the **Variable Name** column.

**5** Drag the target device tag set *Net\_Out1* in the **Variable Name** column of the Toolbox to the connection list.

As you enter characters (or immediately after you press the Ctrl + Space Keys), a list of target device variables that can be set for the connection is provided. Select the value from the list.



**6** Specify **Originator Variable** and its **Size** [Byte] for the tag set *Net\_Out1* added in step 5. Here, specify *Net\_In1* for **Originator Variable** and *2* for its **Size** [Byte].

Change the other settings as required.

You can set the following items in the connection settings.

Name	Setting Methods
Target Device	Select the target device.
Connection Name	Any name can be given to the connection (32 single-byte characters max.).
Connection I/O Type	Input Only (Tag type) is selected if EtherNet/IP connections are used with a CS1W-EIP21, CJ1W-EIP21, CJ2B-EIP21, CJ2M-EIP21, CJ1W-EIP21(CJ2), CJ1W-EIP21(NJ), NX701-□□□□, NJ501-□□□□, NJ301-□□□□, or NJ101-□□□□.
	When you create EtherNet/IP connections for other target devices, select the connection I/O type specified in that device's EDS file.
	Use the <b>Input Only (ID type)</b> setting when another company's node is the originator and does not support connection settings with a Tag type setting.
Input/Output	The connection's input/output is automatically displayed based on the selected connection.
	Input Only: Just Input is displayed.
Target Variable	Select the target node's tag set to assign it.
	Input is specified for Input/Output: Select the target's output (produce) tag set.
	Output is specified for Input/Output: Select the target's input (consume) tag set.

Name	Setting Methods
Size [Byte]	The data sizes of the target variables are displayed.
Originator Variable	Select the originator node's tag set to assign it.
	• Input is specified for Input/Output: Select the originator's input (consume) tag set.
	Output is specified for Input/Output: Select the originator's output (produce) tag set.
Size [Byte]	Enter the data sizes of the originator variables.
Connection Type	Select whether the data is sent in multi-cast or unicast (point-to-point) form. The default setting is multi-cast.
	Multi-cast connection: Select when the same data is shared by multi- ple nodes. This setting is usually used.
	Point-to-point connection: Select when the same data is not shared by multiple nodes.
	In a unicast transmission, other nodes are not burdened with an unnecessary load.
	<b>Note</b> Refer to <i>7-1-4 Overview of Operation</i> for details on using multicast and unicast connections, and counting the number of connections.
RPI [ms]	Set the data update cycle (i.e., the packet interval) of each connection between the originator and target.
	The default setting is 50 ms (i.e., data is updated once every 50 ms).
Timeout Value	Set the time until a connection timeout is detected.
	The timeout value is set as a multiple of the packet interval (RPI) and can be set to 4, 8, 16, 32, 64, 128, 256, or 512 times the packet interval.
	The default setting is RPI x 4.
	The timeout value must be at least 10 ms.

The Toolbox displays the target devices if the devices are registered in the same Sysmac Studio project as where the originator devices are registered.

You can use one of the following methods to add unregistered devices in the same Sysmac Studio project as where the originator devices are registered to the Target Device List.

- Importing devices that are registered in another project
   You can import NJ/NX-series Controllers registered in another project data and add them to the Device List.
- Registering devices using user-specified settings
   You can manually add target devices to the device list.

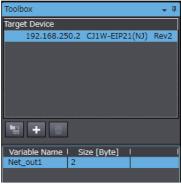


#### **Additional Information**

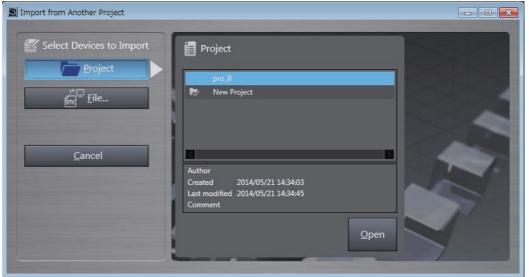
You can add target devices to the Device List by installing EDS files that include connection information for the devices in the Sysmac Studio and register the devices to the project.

Refer to Adding EDS Files on page A-22 for details.

- 8 Import devices that are registered in another project.
  - (1) Click the [12] (Import a device from another project) Button in the Toolbox on the right of the EtherNet/IP Connection Setting Tab Page.

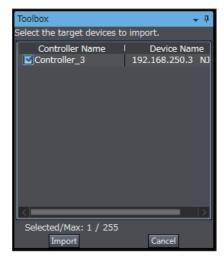


(2) The Import from Another Project Dialog Box is displayed. Click the Project Button, select a project to import and click the Open Button.



(3) The list of EtherNet/IP devices registered in the selected project will be displayed. Select the target devices to import, and click the Import Button.

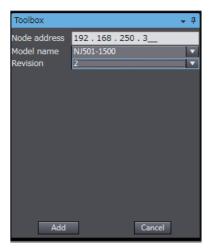
**Note** Only the project for which the EtherNet/IP connection settings are set will be displayed. The imported EtherNet/IP devices are added to the Target Device List in the Toolbox.



- **9** Register devices as required.
  - (1) Click the + Button under the Target Device List in the Toolbox.

The Add Target Device Pane is displayed.

(2) Enter relevant items for the target devices to add.



Menu	Description
Node address	Enter the target device IP address.
Model name	Select the target device model.
Revision	Select the revision of the target device.

(3) Here, set the following items for Controller 3 and click the Add Button.

The target device is added to the Target Device List in the Toolbox.

Node address: 192.168.250.3 Model name: NJ501-1500

Revision: 2

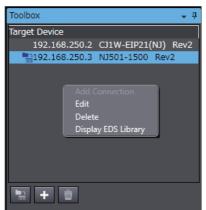
(4) You can click the Import Tag Set Button to import the tag sets that are set in the Network Configurator to the target devices.

Select *Export to File* from the **To/From File** Button in the Tag Sets Tab Page of the Edit Device Parameters Dialog Box to generate CSV files to import.

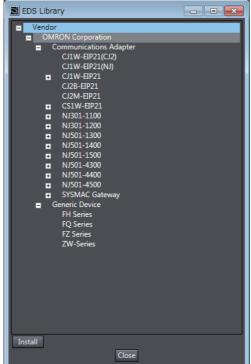
# Adding EDS Files

Note The Modular EDS device is supported by the Sysmac Studio version 1.11 or higher.

1 Right-click anywhere in the Target Device List in the Toolbox of the EtherNet/IP Connection Setting Tab Page and select *Display EDS Library* from the menu.



**2** The EDS Library Dialog Box is displayed. Click the Install Button.



- 3 Select the EDS file to add, and then click the Open Button.
  The EDS file is added.
- 4 The EtherNet/IP device with the EDS file installed is added to the EDS Library.

  Devices listed in the EDS Library are used as a candidate device list when adding devices to the Target Device List in the Toolbox of the EtherNet/IP Connection Setting Tab Page.

#### Changing the Target Device Settings after Making Connection Settings

If you change the IP address, model, or revision of the target device after making the connection settings, you must also change the target device settings that are included in the connection settings. You can change the target device settings entirely.



#### **Precautions for Correct Use**

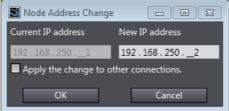
When you use the Sysmac Studio version 1.10 or lower, create the connections again if you changed the target device after configuring the connection settings.

# **Changing the IP Addresses for All Target Devices**

1 Right-click one of the connection lines and select *Change Node Address* from the menu.



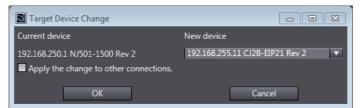
**2** The Node Address Change Dialog Box is displayed. Enter a new IP address in *New IP address*.



- To apply the same change to other connections, select the *Apply the change to other connections* Check Box.
- 4 Click the **OK** Button.

# **Changing All Target Device Information including Model Names and Revisions**

- **1** Right-click one of the connection lines and select *Change Node Address* from the menu.
- **2** The Target Device Change Dialog Box is displayed. Select a target device from *New device*.





#### **Precautions for Correct Use**

- Changeable target devices are limited to ones that have "OMRON" in the Vendor ID and is an EDS device of the Communications Adapter in the Device Type.
- To display a device in the list of selectable new target devices, the device must be registered as the target device in the Toolbox.
- **3** To apply the same change to other connections, select the *Apply the change to other connections* Check Box.
- 4 Click the **OK** Button.

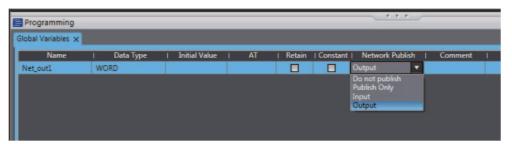
# **Registering the Network Variable for the Target Device**

**1** Assign the network variable to the tag used for the EtherNet/IP connection for Controller 2 (target device).

This network variable stores data to send to Controller 1 (originator device).

Set the Network Publish attribute to *Input* or *Output* in the Global Variable Table for the variable so that the variable serves as a network variable, i.e., the variable can be used for the Ether-Net/IP connections.

In this example, set the network variable for Controller 1 as shown below.



Name: Net\_Out1Data type: WORD

· Network Publish attribute: Output

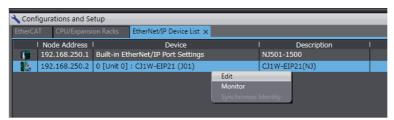
# Registering the Tag and Tag Set for the Target Device

Set the tag and tag set for the target device.

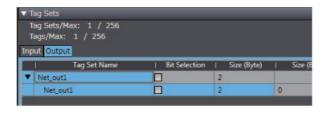
Select *EtherNet/IP Connection Settings* from the Tools Menu.
The EtherNet/IP Device List Tab Page is displayed.

**2** Right-click CJ1W-EIP21, the EtherNet/IP Unit connected to the Controller 2 (originator device in this example), and select **Edit** from the menu.

The EtherNet/IP Connection Setting Tab Page is displayed.



- 3 Click the (Show Tag Set Display) icon in the EtherNet/IP Connection Setting Tab Page.
- Click the Output tab to switch to the Output Tab Page. Register the following tag and tag set.
  The tag and tag set can be registered in the same way as for the target device. (Refer to Registering the Tag and Tag Set on page A-14.)



# **Checking the Device Bandwidth Usage**

The bandwidth usage for the device can be displayed from the EtherNet/IP Connection Setting Tab Page.

This value is for when multicast filtering is used.



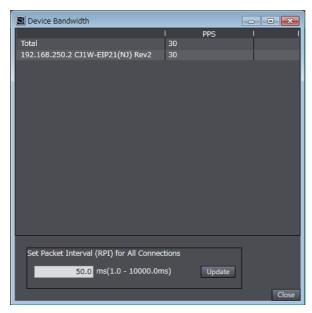
#### **Precautions for Correct Use**

In the Device Bandwidth Dialog Box, you can only check the bandwidth being used for the Ether-Net/IP connections from one originator device to its target devices.

The actual bandwidth used for the EtherNet/IP network must be calculated by taking into account of all bandwidths used on the EtherNet/IP network (i.e., bandwidths used for connections for the other devices in the EtherNet/IP network than the one given on the dialog box must be included into the calculation).

#### Procedure

Click the **Device Bandwidth** Button in the EtherNet/IP Connection Setting Tab Page for the target device.



Menu	Description
PPS	Gives the bandwidth used for each target device and total bandwidth used for all target devices.
Set Packet Interval (RPI) for All Connections	Changes all Packet Interval (RPI) values for all target devices.



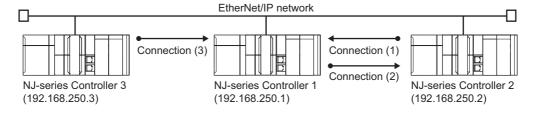
#### **Additional Information**

You can specify a value in **Set Packet Interval (RPI) for All Connections** and click the **Update** Button to change packet interval (RPI) values set in the connection settings for all target devices to the specified value.

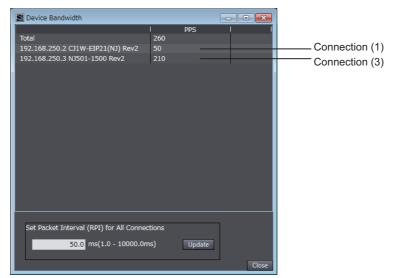
## Calculation Example for Bandwidth Used (PPS) for Each Device by the EtherNet/IP Connections

Establishing following three EtherNet/IP connections between Controllers (1) to (3) in the EtherNet/IP network

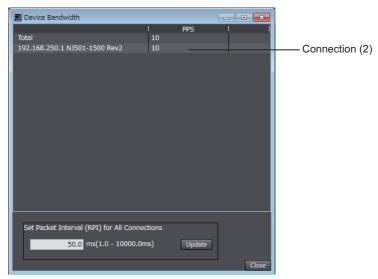
Connection type	Relevant devices in the EtherNet/IP connections	Device bandwidth usage (PPS)
Connection (1)	NJ-series Controller 2 (target device) to NJ-series Controller 1 (originator device)	50 pps
Connection (2)	NJ-series Controller 1 (target device) to NJ-series Controller 2 (originator device)	10 pps
Connection (3)	NJ-series Controller 3 (target device) to NJ-series Controller 1 (originator device)	210 pps



Bandwidth used (PPS) for each EtherNet/IP device is as given below.



EtherNet/IP connection settings for Controller 1



EtherNet/IP connection settings for Controller 2

In this example, the PPS for Connection (1) is 50 pps, the PPS for Connection (2) is 10 pps, and the PPS for Connection (3) is 210 pps. Therefore, bandwidth used (PPS) for each EtherNet/IP device is as given below.

192.168.250.1: 270 pps = 50 pps (for Connection (1)) + 10 pps (for Connection (2)) + 210 pps (for Connection (3))

192.168.250.2: 60 pps = 50 pps (for Connection (1)) + 10 pps (for Connection (2))

192.168.250.3: 210 pps = 210 pps (for Connection (3))

#### Adjusting Method

If the calculation result value exceeds the values in the specifications of the devices used in the EtherNet/IP connections, re-evaluate the overall network configuration and correct it by taking steps such as selecting a different Ethernet switch or splitting the network.

If the RPI is made longer, the PPS for the EtherNet/IP connections will decrease.

You can change the RPI value in the connection settings for all target devices by specifying a value in Set Packet Interval (RPI) for All Connections in this dialog box.

Refer to 14-2-2 Tag Data Link Bandwidth Usage and RPI on page 14-8 for the relationship between the PPS for the device and the RPI.

# **Transferring the Connection Settings Data**

resolved (i.e., the operations agree the settings).



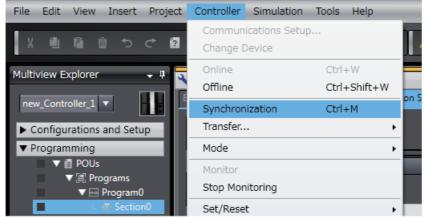
#### **Precautions for Correct Use**

- If the node addresses (IP addresses) are not set correctly, you may connect to the wrong Controller and set incorrect device parameters. Download data only after you confirm that you are connected to the correct Controller.
- If incorrect connection settings are set, it may cause equipment to operate unpredictably. Even
  when the correct connection settings are set, make sure that there will be no effect on equipment before you transfer the data.
- A connection error will result if the network variables that are used in the tag settings are not set in the Controller. Before downloading the connection settings, check to confirm that the network variables used in the tag settings are set in the Controller.
- If a communications error occurs, the output status depends on the specifications of the device being used. When a communications error occurs for a device that is used along with output devices, check the operating specifications and implement safety countermeasures.
- The built-in EtherNet/IP port and the port on the EtherNet/IP Unit are automatically restarted after the parameters are downloaded. This restart is required to enable the tag set and connection information. Before you download the parameters, check to confirm that problems will not occur with the equipment when the port is restarted.
- Do not disconnect the Ethernet cable or reset or turn OFF the power to the EtherNet/IP Unit during the parameter download.
- The EtherNet/IP connections between relevant nodes is stopped during a download. Before
  you download data in RUN mode, make sure that it will not affect the controlled system.
   Also implement interlocks on data processing in ladder programming that uses EtherNet/IP
  - connections when the connections are stopped or a connection error occurs.
- In the EtherNet/IP network, if the device bandwidth usage (PPS) exceeds the unit's allowable bandwidth (PPS), the EtherNet/IP connection operations may not agree with the settings.
   If you increase the RPI value in such a case, there are cases when the problem can be

#### Synchronizing/Transferring a Whole Project

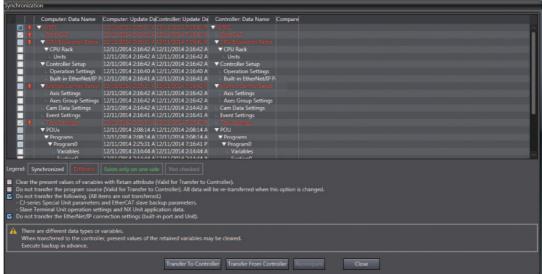
You can synchronize and transfer the EtherNet/IP connection settings along with the program data. You can also transfer all the EtherNet/IP connection settings along with the program data.

- · Synchronizing and transferring the data
- **1** Establish an online connection between the computer and the Controller and then select **Synchronization** from the Controller Menu. (Or, click the Button on the Toolbar.)



The Synchronization Window is displayed, and comparison of the user program and parameter settings between the Sysmac Studio and the Controller is started.

**2** The following Uploading and Downloading Data Window is displayed after the automatic comparison.



Glear the Do not transfer the EtherNet/IP connection settings (built-in port and Unit) Check Box and click the Transfer to Controller Button.

Then the EtherNet/IP connection settings are transferred along with the not-synchronized data. If no EtherNet/IP connection settings are set in the Sysmac Studio, no data will be sent.

- · Transferring all data
  - Establish an online connection between the computer and the Controller and then select *Transfer To Controller* from the Controller Menu. (Or, click the Button on the Toolbar.)
  - **2** The **Transfer to Controller** Dialog Box is displayed.

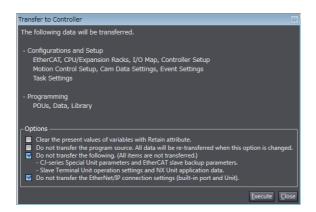
Clear the *Do not transfer the EtherNet/IP connection settings (built-in port and Unit)* Check Box and click the **Execute** Button.



#### **Precautions for Correct Use**

To transfer only the connection settings, execute Transfer from the EtherNet/IP Connection Setting Tab Page.

The connection settings are not transferred from the Synchronization Window, the Transfer to Controller Dialog Box, and the Transfer from Controller Dialog Box, even if you clear the *Do not transfer the connection setting* Check Box, as long as the data in the computer and in the Controller is the same.



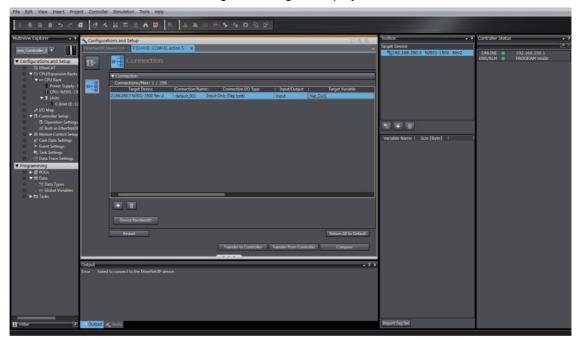
# Transferring Only the EtherNet/IP Connection Settings

You can transfer tag sets and connections to the EtherNet/IP devices.

- **1** Establish an online connection with the Controller.
- 2 Click the Transfer to Controller or Transfer from Controller Button in the EtherNet/IP Connection Setting Tab Page.

The tag settings and connection settings set at that time are transferred to the Controller connected online.

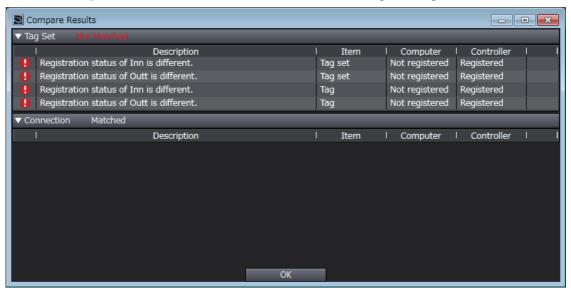
**3** If the Controller connected online is in RUN mode, the dialog box to confirm whether to switch to PROGRAM mode before transferring the settings is displayed.



#### Comparison

The differences in the tag set and connection settings between the project and the EtherNet/IP devices can be displayed.

1 Click the Compare Button in the EtherNet/IP Connection Setting Tab Page.



# Starting and Stopping EtherNet/IP Connections

#### Automatically Starting EtherNet/IP Connections

The EtherNet/IP device is automatically restarted and EtherNet/IP connections are automatically started immediately after the connection settings are downloaded from the Sysmac Studio.



#### **Precautions for Correct Use**

Connections are adversely cut off if any of the following errors occurs in the CPU Unit that is the originator while EtherNet/IP connections are active.

- Major fault level Controller error
- · Partial fault level Controller error

### Starting and Stopping the EtherNet/IP Connections for the Entire Network

You can start and stop EtherNet/IP connections from the user program or from the Sysmac Studio.



#### **Precautions for Correct Use**

Use the same method (i.e., either the user program or the tool software) to both start and stop EtherNet/IP connections.

For example, if you use the \_EIP\_TDLinkStopCmd (Tag Data Link Communications Stop Switch) system-defined variable to stop EtherNet/IP connections, you cannot start them from the Sysmac Studio and the Network Configurator.

# A-2-5 Checking Communications Status with the Sysmac Studio and Troubleshooting

You can monitor the communications status of the EtherNet/IP connections after their settings are set. You can also check errors.



#### **Precautions for Correct Use**

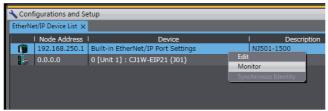
Make sure that the connection settings in both the Sysmac Studio and the Controller are consistent before using the monitor functions. You can use the *Comparison* on page A-32 to see if they are the same.

# Checking Communications Status with the Sysmac Studio

You can check the communications status on the EtherNet/IP connections in the EtherNet/IP Connection Monitor Tab Page.

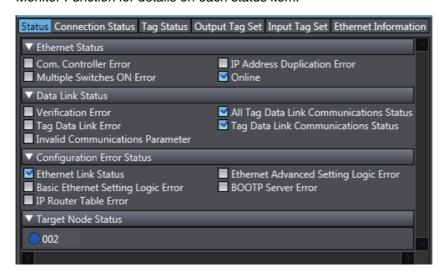
- Select EtherNet/IP Connection Settings from the Tools Menu to display the EtherNet/IP Device List Tab Page.
- 2 Right-click the Controller you want to check the communications status and select *Monitor* from the menu.

The pane to monitor the EtherNet/IP connection is displayed. This pane has six tabs for each communications status.

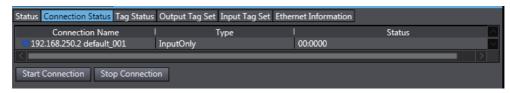


- 3 Select one of the six tabs for which you want to confirm the communications status.
  - Status Tab Page

This tab page gives the TRUE/FALSE status of the system-defined variables that monitors the tag data link errors and communication status. If any of the variables are TRUE, the checkbox in front of the variable will be selected. Refer to 15-3-1 The Network Configurator's Device Monitor Function for details on each status item.



Connection Status Tab Page
 Current status of each connection is given.



Name	Description
Connection Name	Gives the current status of each connection with the following text colors.
	Green: Normal
	Red: There is at least one connection that has not been established.
	Gray: There are no connections or the connection operation is stopped.
Туре	Gives the connection type.
Status	Gives the current status on each connection with codes.
	Normal operation: 00:0000
	Abnormal operation: Gives an error code.     This information can be used to identify the cause of EtherNet/IP connection errors. Refer to 15-3-2 Connection Status Codes and Trouble-shooting on page 15-58 for details on the connection status.

#### • Tag Status Tab Page

This tab page gives if the tag settings for each tag for EtherNet/IP connections are set so that data can be exchanged with target devices.



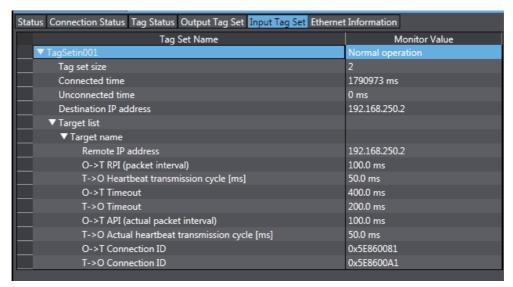
Name	Description
Tag Name	Gives the tag name.
Input/Output	Gives the type of the tag.
Status	The following status is displayed depending on the status that is set.
	Normally resolved: Normal data exchange is possible.
	• Different sizes: Different sizes are set for the network variables and the tag settings.
	A connection will not be established for a tag for which this error occurs.
	No tag: A network variable is not set in the variable table in the CPU Unit for the specified tag setting.
	A connection will not be established for a tag for which this error occurs.
	Attribute error: Writing is not possible for Constant attributes.

• Output Tag Set and Input Tag Set Tab Pages

You can monitor the status of each input/output tag set that is used for the EtherNet/IP connections.

Note The tag set status monitor is not available for a built-in EtherNet/IP port on NJ-series Controller version 1.08 or earlier.

Click ▼ of each tag to display its detailed information.

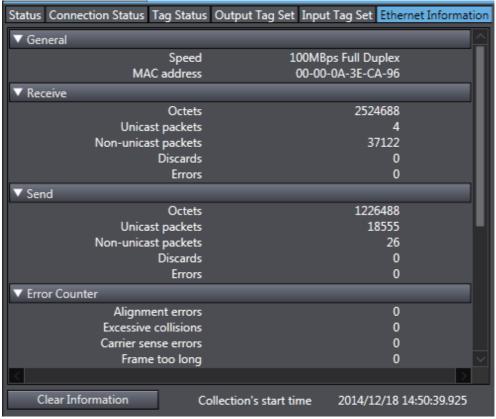


Name	Description
Tag Set Name	Gives the connection status.
	If there is a connection error, "Not connected or error" is given.
Tag set size	Gives the size of the tag set in bytes.
Connected time	Gives the total connection duration in milliseconds.
Unconnected time	Gives the total disconnection duration in milliseconds.
Number of connections (in the Output Tag Set Tab Page)	Gives the number of connections.
Number of connected originators (in the Output Tag Set Tab Page)	Gives the number of the connected originator devices.
Originator list (in the Output Tag Set Tab Page), Target list (in the Input Tag Set Tab Page)	Gives the detailed information of the connected originators.
Originator name (in the Output Tag Set Tab Page), Produced tag name (in the Input Tag Set Tab Page)	Gives no information.
IP address (in the Output Tag Set Tab Page), Remote IP address (in the Input Tag Set Tab Page)	Gives the IP addresses allocated for the originators.
Connected time (in the Output Tag Set Tab Page)	Gives the total duration of connection with the originator in milliseconds.
Unconnected time (in the Output Tag Set Tab Page)	Gives the total duration of disconnection with the originator in milliseconds.

Name	Description
Destination IP address (in the Output Tag Set Tab Page)	Gives the destination IP addresses. If the multi-cast connections are used, its own multi-cast address is displayed.
O->T RPI (packet interval)	Gives the RPI of connection from the originator to the target in milliseconds.
T->O Heartbeat trans- mission cycle (ms)	Gives the heartbeat transmission period of the connections from the target to the originator in milliseconds.
O->T Timeout	Gives the timeout time for the connections from the originator to the target in milliseconds.
T->O Timeout	Gives the timeout time for the connections from the target to the originator in milliseconds.
O -> T API (actual packet interval)	Gives the API of connection from the originator to the target in milliseconds.
T->O Actual heartbeat transmission cycle (ms)	Gives the actual heartbeat transmission period of the connections from the target to the originator in milliseconds.
O->T Connection ID	Gives the connection identification for the connections from the originator to the target in hexadecimal.
T->O Connection ID	Gives the connection identification for the connections from the target to the originator in hexadecimal.

#### • Ethernet Information Tab Page

This tab page displays the communications status at the communications driver level of the built-in EtherNet/IP port. The error counter information can be used to confirm whether communications problems have occurred. Under the Tag Data Link, you can confirm characteristics such as the bandwidth usage (PPS).



Display example for an NJ-series CPU Unit
With an NX-series CPU Unit, the statuses for each port are given.

## A-2-6 Troubleshooting

In the case that there is a setting error or a communications error in the EtherNet/IP networks, the Sysmac Studio displays the error in the Troubleshooting Dialog Box.

Refer to 15-2 Troubleshooting on page 15-12 for the confirmation methods for errors and information on errors.

# Troubleshooting When Transferring and Monitoring the EtherNet/IP Connection Settings Fail with Sysmac Studio Version 1.10 or Higher

The first time you establish an online connection between the Controller and the computer that runs the Sysmac Studio version 1.10 or higher with Windows Firewall on the computer enabled, the dialog box to confirm the connection may be displayed. If that occurs, make the following selection in the dialog box.

- Unblock (on Windows XP/Vista)
- Allow access (on Windows 7)

If you make other selections than above, there are cases when transferring and monitoring the Ether-Net/IP connection settings cannot properly be performed even if the online connection is successfully established.

Disabling a part of Windows Firewall settings can resolve this.

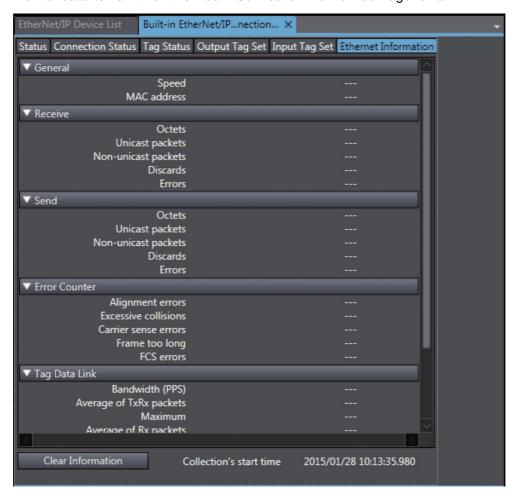
(Refer to *Method 1: Changing the Memory Configuration of the Structure Variable in the NJ/NX-series CPU Unit* on page A-61 for details.)

## Problems

• The connection setting data cannot be transferred

Data Transmission Screen	Problem
	1.102.0
Synchronization Window	The Sysmac Studio displays the following error message and the data will not be transferred.
	■ Do not transfer the EtherNet/IP connection settings (built-in port and Unit).
	Pailed to transfer the EtherNet/IP connection settings from the Controller. (Communication error)
Transfer to Controller Dialog Box	The Sysmac Studio displays the following error dialog box and the data will not be transferred.
	Transfer from Controller
	Failed to transfer the EtherNet/IP connection settings from the Controller. (Communication error) Process was aborted.
	OK
EtherNet/IP Connection Setting Tab Page	The Transfer to Controller and Transfer from Controller Buttons are grayed out and the data cannot be transferred/compared.
	Configurations and Setup
	EtherNet/IP Device List  Built-in EtherNet/IPection Se ×  Connection  Connection
	Connections/Max: 1 / 32 Target Device   Connection Name  Connection I/O Type 192.168.250.10 NJ301-1200 Rev 2   default_001   Input Only (Tag type)
	<b>-</b>
	Device Bandwidth
	Restart Return All to Default
	Transfer to Controller Transfer from Controller Compare

Monitoring the settings cannot be performed
 Monitor data items in the EtherNet/IP Connection Monitor Tab Page remain "---".



#### Method 1: Disabling Windows Firewall Settings



#### **Precautions for Correct Use**

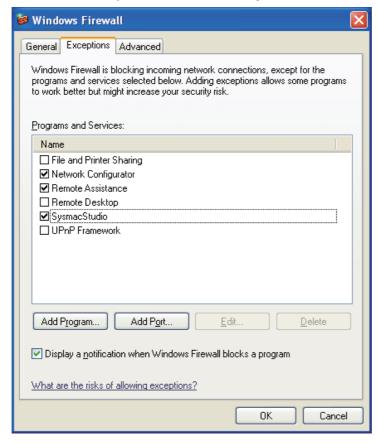
The main function of the firewall is to prevent unwanted access from external sources (e.g., the Internet).

The changes that are made with the following procedures are to allow the Sysmac Studio and the NJ/NX-series Controller to connect. If your computer is on an inhouse network, make sure that security will not be jeopardized before you change the settings.

- Windows XP
  - 1 Open the Control Panel from the Windows Start Menu and then select Windows Firewall icon.

The Windows Firewall Dialog Box is displayed.

2 Click on the Exceptions tab and select Sysmac Studio in the Programs and Services list.



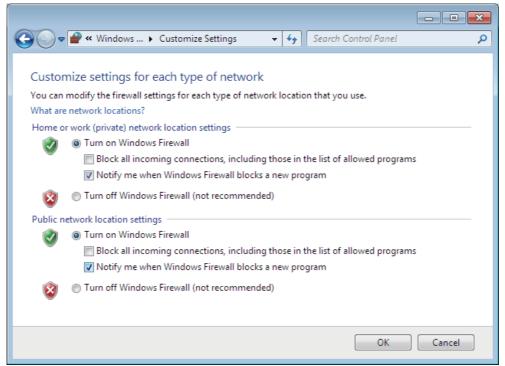
- Windows Vista, Windows 7, or later version
  - 1 Open the Control Panel from the Windows Start Menu and then select Windows Firewall icon.

The Windows Firewall Dialog Box is displayed.

2 Select Turn Windows Firewall on or off.

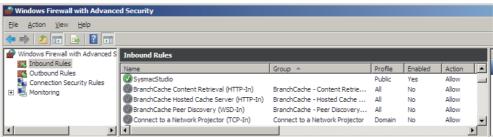
The Customize Settings Dialog box is displayed.

3 Clear the Block all incoming connections, including those in the list of allowed programs Check Box and click the OK Button.



- $m{4}$  Select Advanced settings in the Windows Firewall Dialog Box.
  - The Windows Firewall with Advanced Security Dialog Box is displayed.
- **5** Click *Inbound Rules* in the left pane and then double click **SysmacStudio** in the **Inbound Rules** list.

The SysmacStudio Properties Dialog Box is displayed.



6 In the General Tab Page of the dialog box, set the following settings.

Select Enabled under the General section.

Select Allow the connection under the Action section.

## • Method 2: Cycle the power supply to the Controller

Cycle the power supply to the NJ/NX-series Controller and transfer/monitor the EtherNet/IP connections settings again.

**Note** You may need to cycle the power supply when reflecting the changes in the IP address of the built-in EtherNet/IP port or executing Transfer to Controller.

# A-3 EDS File Management

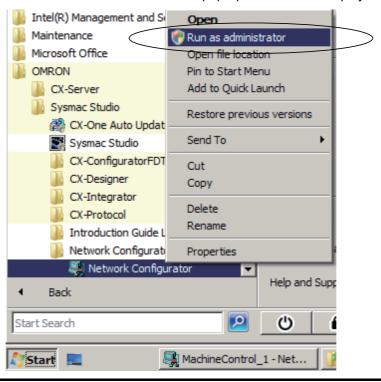
This section describes the EDS file management functions used in the Network Configurator.



#### **Precautions for Correct Use**

For Windows Vista or Windows 7, we recommend that you start the Network Configurator as the administrator. Otherwise, the following condition will result due to user management for Windows security functions. The results of the following operations are not applied for logins with other user accounts and must be repeated: installing, creating, and deleting EDS files, and creating EDS index files. You can run the Network Configurator as the administrator with the following procedure.

- 1. Select the Network Configurator from the Start Menu, and then right-click.
- 2. Select *Run as administrator* from the pop-up menu that is displayed.



# A-3-1 Installing EDS Files

# **EDS File - Install**

The Network Configurator can support new devices if the proper EDS files are installed.

To install the EDS file, use the following procedure.

- Select EDS File Install.
  The Install EDS File Dialog Box is displayed.
- 2 Select the EDS file to install and click the **Open** Button. Next, select the icon file (\*.ico). The EDS file is added to the Hardware List as a new device. If the hardware already exists, the new Hardware List will overwrite the previous one. If the hardware versions are different, a hardware device is added to the Hardware List for each version.

# A-3-2 Creating EDS Files

#### **EDS File – Create**

The EDS files are required by the Network Configurator to create a network configuration. To create an EDS file, use the following procedure.

- 1 Select EDS File Create.
- 2 Set the device information. You can obtain the device information from the device on the network if the network is online.
- **3** The device is added to the Hardware List as a new device, just like when you install an EDS file.



#### **Additional Information**

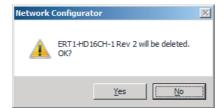
You cannot set device parameters with the Network Configurator's EDS file creation function. Obtain a proper EDS file from the manufacturer of the device to make device parameter settings for the device.

# A-3-3 Deleting EDS Files

# **EDS File – Delete**

To delete an EDS file, use the following procedure.

- **1** Select the device from the Hardware List.
- Select EDS File Delete.
  The following confirmation dialog box is displayed.



Glick the **Yes** Button.

The selected device is deleted from the Hardware List together with the EDS file.

# A-3-4 Saving EDS Files

# **EDS File – Save**

To save the EDS file, use the following procedure.

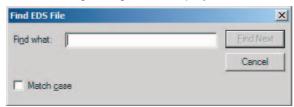
- Select the target hardware device in the Hardware List, and then select EDS File Save.
  A Save EDS File Dialog Box is displayed.
- Input the folder and file names and click the Save Button. The EDS file is saved.

# A-3-5 Searching EDS Files

# **EDS File – Find**

To search the devices (EDS files) displayed in the Hardware List, use the following procedure.

Select EDS file – Find.
The following dialog box is displayed.



- 2 Input the character string to search for and click the Find Next Button.
- **3** When a matching device is found, the cursor moves to that position.
- **4** To guit the search operation, click the **Cancel** Button.



#### **Additional Information**

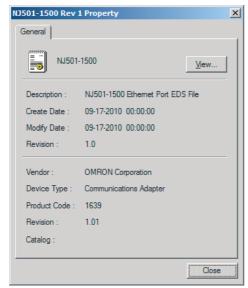
- The device is found only if it is located below the present cursor position in the Hardware List.
- To search all the devices, select *Hardware* in the Hardware List before you perform the search procedure.

# A-3-6 Displaying EDS File Properties

# **EDS File – Property**

To display the properties of the EDS file, use the following procedure.

- **1** Select the desired hardware (device) from the Hardware List.
- Select EDS File Property.
  The following dialog box is displayed.



The time and date when the EDS file was created is displayed, along with the device information.

# A-3-7 Creating EDS Index Files

# EDS File – Create EDS Index File

To manually add an EDS file or if a device is not displayed correctly in the hardware list, use the following procedure to recreate the EDS index file. (This applies to Network Configurator version 3.30 or higher.)

- 1 Select EDS File Create EDS Index File.
- Restart the Network Configurator.

# A-4 Precautions for Using the Network Configurator on Windows XP, Windows 7

Better firewall security for Windows XP (SP2 or higher), Windows Vista, and Windows 7 has increased the restrictions for data communications. Therefore, you must perform the corresponding procedure given below to change the settings of the Windows firewall before you use the following operations to perform communications with the Network Configurator connected to an NJ/NX-series CPU Unit.

- If you select Option Select Interface Ethernet I/F
- If you select Option Select Interface NJ/NX Series Ethernet Direct I/F
- If you select Option Select Interface NJ/NX Series USB Port



#### **Precautions for Correct Use**

The main function of the firewall is to prevent unwanted access from external sources (e.g., the Internet). The changes that are made with the following procedures are to allow the Network Configurator and the NJ/NX-series CPU Unit to connect. If your computer is on an inhouse network, make sure that security will not be jeopardized before you change the settings.

# A-4-1 Changing Windows Firewall Settings

# Windows XP

- When you attempt to connect to the NJ/NX-series CPU Unit from the Network Configurator, the Windows Security Warning Dialog Box is displayed.
- **2** Click the **Unblock** Button.

A USB or EtherNet/IP connection will be approved for the Network Configurator, and you will be able to connect the Network Configurator in the future.

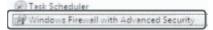
# **Windows Vista or Windows 7**

Use the following procedure to change the settings. Always perform steps 1 to 6 if you cannot go online. The User Account Control Dialog Box may be displayed during this procedure. If it appears, click the **Continue** Button and continue with the procedure.

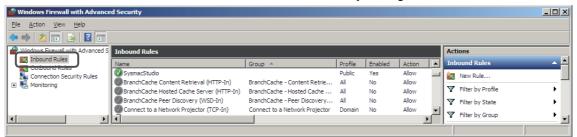
1 Select Control Panel from the Windows Start Menu and change the display to Classic View.



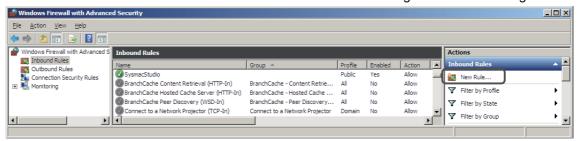
**2** Open the Administrative Tools and select *Windows Firewall with Advanced Security* from the dialog box that is displayed.



3 Select Inbound Rules under Windows Firewall with Advanced Security on Local Computer on the left side of the Windows Firewall with Advanced Security Dialog Box.



4 Select New Rule under Inbound Rules in the Actions Area on the right side of the dialog box.

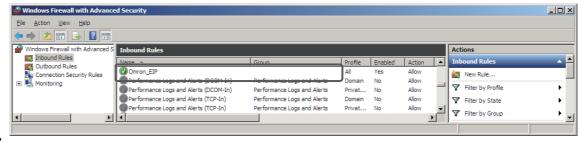


Make the following settings for each step in the New Inbound Rule Wizard Dialog Box, and click the Next Button to move between steps.

Rule Type	Select Custom.
Program	Select All Programs.
Protocol and support	Select ICMPv4 as the protocol type.  Protocol type: ICMPv4  Protocol number: 1 A
Scope	Select Any IP address for everything.
Action	Select Allow the connection.
Profile	Select <i>Domain</i> , <i>Private</i> , and <i>Public</i> .
Name	Enter any name, e.g., Omron_EIP.

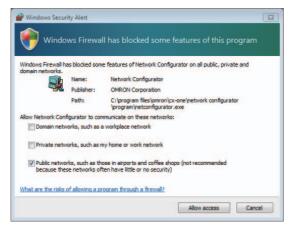
6 Click the Finish Button. The rule that you defined will be registered in the Inbound Rules (e.g., Omron\_EIP).

Close the Windows Firewall with Advanced Security Dialog Box.



When you attempt to connect to the NJ/NX-series CPU Unit from the Network Configurator, the Windows Security Warning Dialog Box is displayed.

## 8 Click the Unblock Button.



#### (Windows 7)

A USB or EtherNet/IP connection will be approved for the Network Configurator, and you will be able to connect the Network Configurator in the future.

# A-5 Variable Memory Allocation Methods

You must be aware of the way in which memory is allocated to variables to align the memory locations of the members of structure or union variables with variables in other devices. Adjustments are necessary mainly when structure variables are used in the following type of communications with other devices.

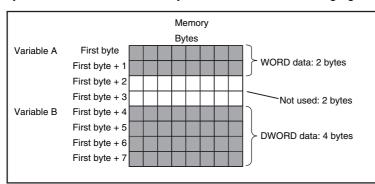
- When using EtherNet/IP tag data links or CIP messages to access variables between NJ/NX-series CPU Units and other CPU Units
- When using structure variables to exchange data with devices other than CPU Units, such as ID Tags

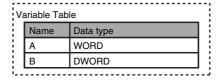
### A-5-1 Variable Memory Allocation Rules

The amount of memory and the memory locations that are allocated for a variable depend on the data type of the variable. The amount of memory and the memory locations that are allocated for array elements, structure members, and union members depend on the data types, but also on the declarations that are made for the arrays, structures, and unions.

## **Data Type Alignment and Memory Allocation Amounts**

The data size is determined for each data type. The data size is the minimum amount of memory that is required to store the value or values of that data type. On the other hand, memory for variables is automatically structured by the Controller for the most efficient access. Therefore, the total amount of memory that is required for variables is not necessarily the total of the data sizes of the variables. For example, if WORD and DWORD variables are declared, the total of the data sizes is six bytes, but eight bytes are allocated in memory, as shown in the following figure.





This information for determining the location of a variable in memory is called the alignment. The alignment is determined for each data type. The amount of memory and the memory locations for the variables are given below.

Item	Specification	
Amount of memory that is allocated	An integral multiple of the alignment. However, the minimum amount of memory is the data size.	
Locations in memory	At an integral multiple of the alignment starting from the start of the variable in memory.	

The alignments and the amounts of memory that are allocated for the basic data types and enumerations are given below.

Data type	Alignment [bytes]	Amount of memory that is allocated [bytes]
BOOL	2	2
BYTE, USINT, or SINT	1	1
WORD, UINT, or INT	2	2
DWORD, UDINT, or DINT	4	4
LWORD, ULINT, or LINT	8	8
REAL	4	4
LREAL	8	8
TIME, DATE, TIME_OF_DAY, or DATE_AND_TIME	8	8
STRING[N+1] <sup>*1</sup>	1	N+1
Enumerations	4	4

<sup>\*1</sup> N is the maximum number of characters handled. For example, if a maximum of 10 single-byte characters are handled, the NULL character is added, so memory for 11 characters must be reserved.

The elements of arrays and the members of structures and unions are located in memory for the most efficient access. The alignments and the amounts of memory that are allocated for arrays, structures, and unions are determined by the variable declarations, as described below.

Data type	Alignment	Amount of memory that is allocated
Array	Same as alignment of the data type of the elements	(Amount of memory that is allocated for the data type of the elements) × Number of elements*
Structure	The largest alignment of all of the members	The integral multiple of the alignment that is larger than the total amount of memory that is allocated when the members are arranged in order at integral multiples of the alignment of the data types of the members
Union	The largest alignment of all of the members	The largest amount of memory that is allocated for any of the members

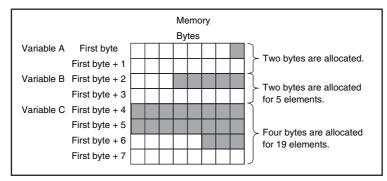
<sup>\*</sup> BOOL arrays are an exception. Refer to *Precautions for Correct Use*, below, for the amount of memory that is allocated for BOOL arrays.



#### **Precautions for Correct Use**

#### **Amount of Memory That Is Allocated for BOOL Arrays**

Two bytes are allocated in memory for individual BOOL variables, BOOL structure members, and BOOL union variables. However, for a BOOL array, two bytes of memory are not allocated for each element. One bit is allocated in order for each element. For the entire array, a multiple of two bytes of memory is allocated (including unused bits).



¦۷۶	ariable Tab	le
:	Name	Data type
:	Α	BOOL
	В	ARRAY[15]OF BOOL
:	С	ARRAY[018]OF BOOL

Therefore, the following formula gives the amount of memory that is allocated for a BOOL array. For 1 to 16 elements, 2 bytes are allocated. For 17 to 32 elements, 4 bytes are allocated.

Amount of memory = 
$$2\left[\begin{array}{c} \text{Number of elements} - 1 \\ \hline 16 \end{array}\right] + 2$$

Truncate the decimal portion of the result of the calculation in brackets.

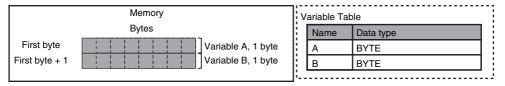
Specific examples of the rules for memory allocation for variables of each data type are given below.

## **Basic Data Types**

#### Variables with One-Byte Alignments (e.g., BYTE)

One byte of memory is allocated for the one-byte alignment.

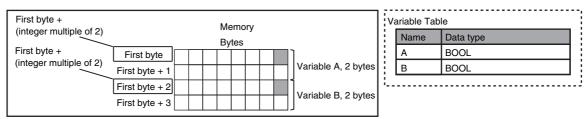
Example: Two consecutive BYTE variables



#### Variables with Two-byte Alignments (e.g., BOOL and WORD)

Two bytes of memory are allocated for the two-byte alignment.

Example: Two consecutive BOOL variables

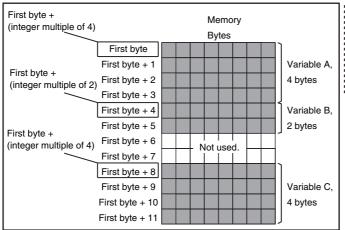


### Variables with Four-byte Alignments (e.g., DWORD)

Four bytes of memory are allocated for the four-byte alignment.

The location of the first byte of data in memory is an integer multiple of four bytes. Therefore, if a variable with a two-byte alignment, such as WORD data, is inserted, two bytes of unused memory will remain.

Example: Consecutive variables in the following order: DWORD, WORD, and DWORD

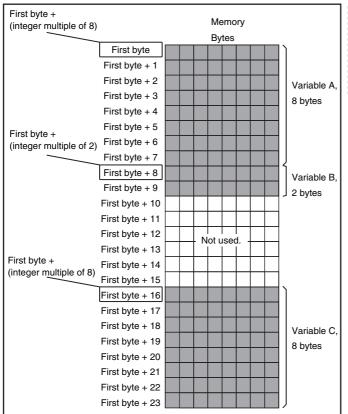


#### Variables with Eight-byte Alignments (e.g., LWORD)

Eight bytes of memory are allocated for the eight-byte alignment.

The location of the first byte of data in memory is an integer multiple of eight bytes. Therefore, if a variable with a two-byte alignment, such as WORD data, is inserted, six bytes of unused memory will remain. If a variable with a four-byte alignment, such as DWORD data, is inserted, four bytes of unused memory will remain.

Example: Consecutive variables in the following order: LWORD, WORD, and LWORD

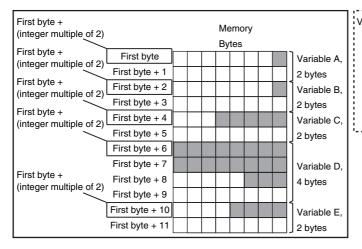


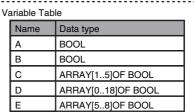
١	/ariable Ta	ble	
	Name	Data type	
	Α	LWORD	
	В	WORD	
	С	LWORD	

## **Arrays**

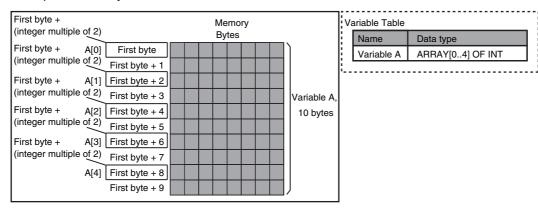
A continuous section of memory is allocated for the elements of the array based on the data size of the data type of the array variable. The alignment of an array is the same as alignment of the data type of the elements.

Example: Continuous variables in the following order: two BOOL variable, one BOOL array with five elements, one BOOL array with 19 elements, and one BOOL array with four elements





Example: INT array with five elements

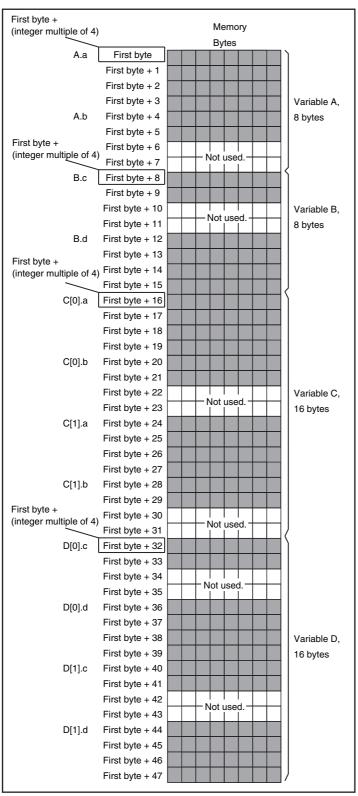


#### **Structures**

For a structure variable, the members are located in memory in the order that they are declared. Each member is located at an integer multiple of the alignment of the data type of the member. Therefore, there can be unused memory between members or at the end of members. The alignment of a structure is the largest alignment of all of the members. The amount of memory that is allocated is the integral multiple of the alignment that is larger than the total amount of memory that is allocated when the members are arranged in order at integral multiples of the alignment of the data types of the members.

Example: The alignments and the amounts of memory that are allocated for the four variable declarations given in the following figure are given in the following table.

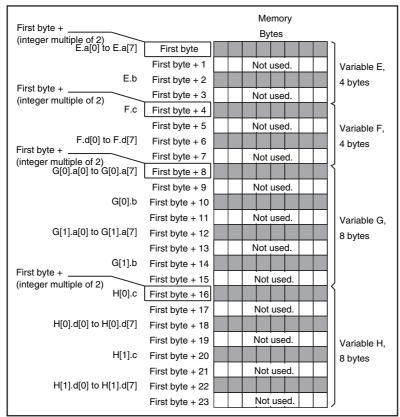
Variable	Alignment [bytes]	Amount of memory that is allocated [bytes]
A	4	8
В	4	8
С	4	16
D	4	16



Name		Data type
Structure 5	STR_A	STRUCT
а		DINT
b		INT
Name		Data type
Structure 3	STR_B	STRUCT
С		INT
d		DINT
/ariable Table	Э	
Name	Data	type
Variable A	Struc	ture <i>STR_A</i>
Variable 71	Struc	ture <i>STR_B</i>
Variable B		AY[01] OF STR_A
		AY[01] OF STR_A

Example: The alignments and the amounts of memory that are allocated for the four variable declarations given in the following figure are given in the following table.

Variable	Alignment [bytes]	Amount of memory that is allocated [bytes]
Е	2	4
F	2	4
G	2	8
Н	2	8



Name		Data type
Structure S	TR_C	STRUCT
а		ARRAY[07] OF BOOL
b		BYTE
Name		Data type
Structure S	TR_D	STRUCT
С		BYTE
d		ARRAY[07] OF BOOL
/ariable Table		į, j
/ariable Table	Data	type
	_	. ,
	Stru	type
Name Variable E	Struc	type

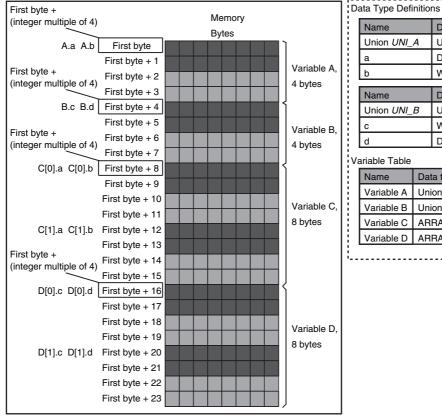
Data Type Definitions

#### **Unions**

For a union variable, the members overlap in the same memory locations. The alignment of a union is largest alignment of all of the members. The amount of memory that is allocated is the largest amount of memory that is allocated for any of the members.

Example: The alignments and the amounts of memory that are allocated for the four variable declarations given in the following figure are given in the following table.

Variable	Alignment [bytes]	Amount of memory that is allocated [bytes]
Α	4	4
В	4	4
С	4	8
D	4	8



Name		Data type	
Union UNI_	Α	UNION	
а		DWORD	
b		WORD	
Name		Data type	
Union UNI_	В	UNION	
С		WORD	
d		DWORD	
/ariable Table			
Name	Data type		
Variable A	Union <i>UNI_A</i>		
Variable B	Union <i>UNI_B</i>		
Variable C	ARRAY[01] OF UNI_A		
Variable D	ARRAY[01] OF UNI_B		

## A-5-2 Important Case Examples

When you exchange structure variable data between an NJ/NX-series CPU Unit and a remote device, you must align the memory configuration of the structure variable members with those of the remote device. This section describes what to do in either the NJ/NX-series CPU Unit or in the remote device.



#### **Additional Information**

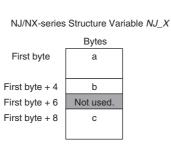
This is not necessary when you exchange data between NJ/NX-series CPU Units.

## Aligning the Memory Configuration with a Remote Device

There are two methods that you can use to align the memory configuration with a remote device. For example, the differences in the memory configuration for structure variables between an NJ/NX-series CPU Unit and a CJ-series CPU Unit are shown below.

This section describes how to align the memory configuration for these Units.

	Data Type Definition	ons
	Name	Data type
	Structure Y	STRUCT
	а	DINT
	b	INT
	С	DINT
٧	ariable Table	
	Name	Data type
	Variable <i>NJ_X</i>	Structure Y



Data Type Definitions				
	Name	Data type		
	Structure Y	STRUCT		
	а	DINT		
	b	INT		
	С	DINT		
Variable Table				
	Name	Data type		
	Variable <i>CJ_X</i>	Structure Y		

#### Method 1: Changing the Memory Configuration of the Structure Variable in the NJ/NX-series CPU Unit

With an NJ/NX-series CPU Unit, you can specify member offsets to change the memory configuration of the members of a structure variable. You can change the memory configuration of the members of a structure variable in the NJ/NX-series CPU Unit so that it is the same as the memory configuration in a remote device that the CPU Unit will communicate with. Specify the member offsets for a structure variable when you register the structure data type.

To communicate with a CJ-series CPU Unit, you can set the offset type to *CJ* to automatically use the CJ-series memory structure. You can set the offset type to *User* to freely set your own offsets.



#### **Version Information**

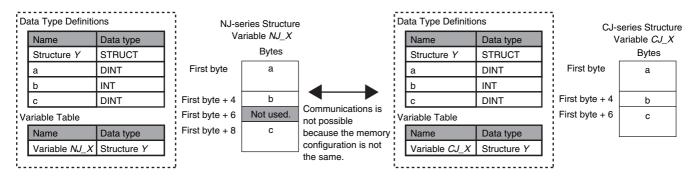
The following table gives the unit version of the CPU Units and the Sysmac Studio version that are required to specify member offsets.

Unit version of CPU Unit	Sysmac Studio version			
Office Version of GFO Office	1.01 or lower	1.02	1.03 or higher	
1.01 or later	Not possible.	Possible.*	Possible.	
1.00	Not possible.	Not possible.	Not possible.	

You cannot select the memory offset type. You can set member offsets.

If you change the memory configuration of a structure variable by setting offsets, you must make the same changes for the same structure variable in other NJ/NX-series CPU Units on the network. Refer to the *Sysmac Studio Version 1 Operation Manual* (Cat. No W504-E1-03 or higher) for the procedure to change the memory configuration of a structure variable.

Example: The following example shows how the memory configuration of the structure variable members in the NJ-series CPU Unit is changed to match the memory configuration of the structure variable members in the CJ-series CPU Unit.

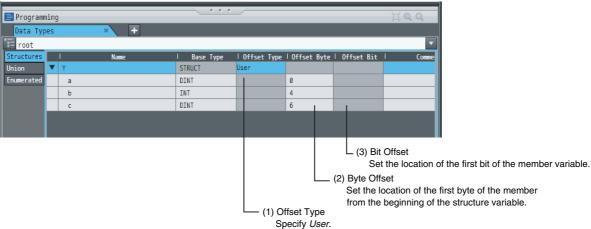


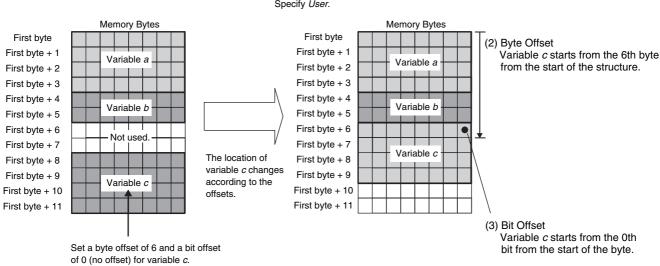
To align the memory configurations in the NJ-series and CJ-series CPU Units, offsets are set in the Sysmac Studio.



Here, the following offsets are set for member c of data type Y of the structure variable NJ\_X.



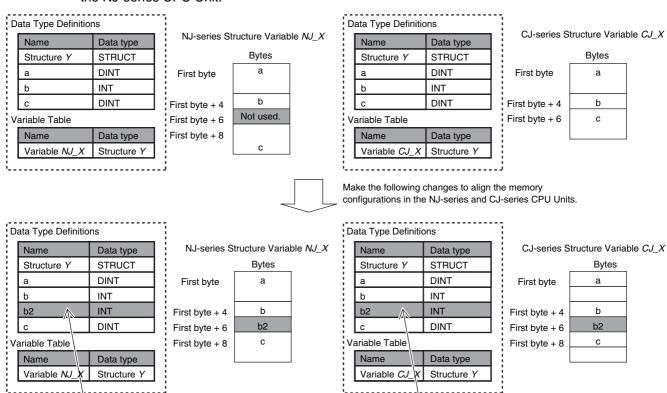




#### Method 2: Changing the Memory Configuration of the Structure Variable in the Remote Device

You can insert a member into the structure variable of the remote device to change it to match the memory configuration of the structure variable in the NJ/NX-series CPU Unit. Both the memory configuration and the data types must be the same between the two structure variables. You therefore need to create the same members in both the remote device and the NJ/NX-series CPU Unit.

Example: The following example shows how the memory configuration of the structure variable in the CJ-series CPU Unit is changed to match the memory configuration of the structure variable in the NJ-series CPU Unit.



<sup>(2)</sup> Add the dummy variable *b2* that you created in the CJ-series CPU Unit to the NJ-series CPU Unit as well.

<sup>(1)</sup> Add a dummy member variable *b2* that matches the unused memory location on the NJ-series CPU Unit.

# A-6 Precautions When Accessing External Outputs in CPU Units

Observe the following precautions when you access variables or I/O memory addresses that are assigned to external outputs in an NJ/NX-series CPU Unit.

Precaution on Writing from External Devices, Variables\* That Are Assigned to External Outputs
Any value that is written to a variable\* that is assigned to an external output in an NJ/NX-series CPU
Unit through a tag data link or communications instruction will be overwritten by the execution results
of the user program. The value that is written from the tag data link or communications instruction will
therefore not be output to the external device.

The following types of variable are assigned to the external outputs.

- The devices variables (or global variables) that are assigned to an I/O port of a CJ-series Basic Output Unit\*
- The devices variables (or global variables) that are assigned to an I/O port of an EtherCAT output slave
- The global variables with AT specifications to output bits that are assigned to CJ-series Basic Output Units\*
- \* NJ-series CPU Unit only
- Precaution When Directly Writing to I/O Memory Addresses Assigned to Output Bits for CJ-series Basic Output Units

Any value that is written to an I/O memory address that corresponds to an output bit that is assigned to a CJ-series Basic Output Unit through a tag data link will be overwritten by the execution results of the user program. The value that is written directly to the I/O memory address from the tag data link will therefore not be output to the external device.

# **A-7** TCP State Transitions

TCP protocol operates in 11 states for connections.

You can check the TCP state of the socket service with the TCP connection status that is output by the SktGetTCPStatus (Read TCP Socket Status) instruction.

The TCP states and their meanings are given in the following table.

TCP state	Meaning
CLOSED	The connection was closed.
LISTEN	The server is waiting for a connection request (SYN) with a passive open.
SYN SENT	The client sent a connection request (SYN) for an active open and is waiting for acknowledgement (SYN + ACK).
SYN RECEIVED	The server sent an acknowledgement (SYN + ACK) to a connection request (SYN) and is waiting for acknowledgement (ACK).
ESTABLISHED	A connection was established.
CLOSE WAIT	The server sent acknowledgement (ACK) to a connection close request (FIN) and is waiting for the server application to be ready to close.
FIN WAIT-1	The client sent a connection close request (FIN) and is waiting for acknowledgement (ACK).
CLOSING	The client and server simultaneously received a connection close request (FIN) and are waiting for acknowledgement (ACK).
LAST-ACK	The server sent a connection close request (FIN) and is waiting for acknowledgement (ACK).
FIN WAIT-2	The client is waiting for a connection close request (FIN).
TIME WAIT	The client received acknowledgement (ACK) to a connection close request (FIN) and is waiting for it to be received and processed by the server.

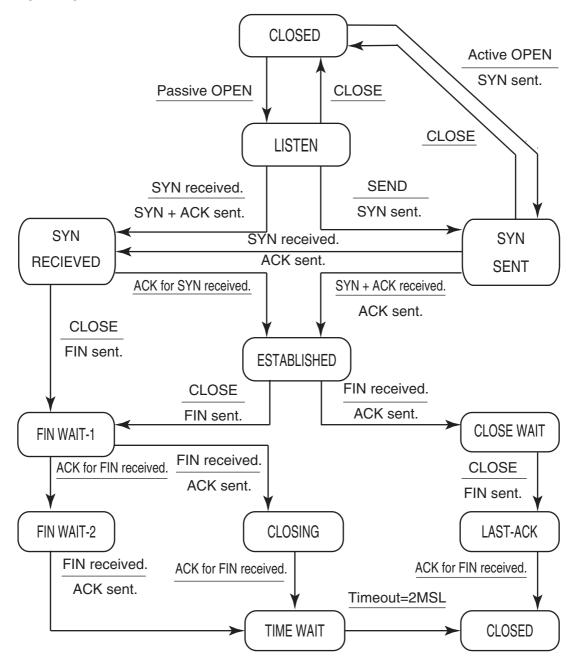
The TCP state changes as requests and acknowledgements are received from the remote node, and as TCP socket connection and close instructions are executed in the user program.

When the state changes, connection requests (SYN), close requests (FIN), and acknowledgements (ACK) to those requests are sent to and received from the remote node.

The following figure shows TCP state transitions.

The TCP states are given in the boxes in the figure. Between the states, the text on top is the condition for a change in state and the text on the bottom is the action that is performed when the state changes. (If no action is given, then none is performed.)

Example: When SYN and ACK are received in SYN SENT state, ACK is sent and the state changes to ESTABLISHED.



# **A-8** Version Information

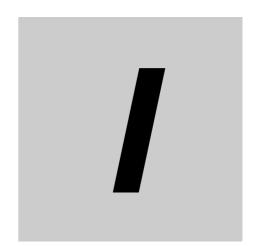
This appendix describes the changes in and additions to functional specifications that were made for changes in the unit version of the CPU Units.

#### Changes in and Additions to Functional Specifications

The following table shows the unit version of the CPU Units and the version of the Sysmac Studio that correspond to changes in or additions to the functional specifications.

	Item	Change or addition	Reference	Unit ver- sion	Sysmac Studio version
CIP routing		Addition	1-17	1.01	1.02
Support for mount Net/IP Unit	ing CJ1W-EIP Ether-	Addition	1-5		
Offsets for struc-	User	Addition	A-60		
ture members	CJ	Addition	A-60	1.02	1.03
CIP objects	Identity object	Change	8-48	1.01	
	TCP/IP Interface object	Change	8-52	1.02	
Tag data links	Packet intervals (RPIs)	Change	7-6	1.03	1.04
	Permissible commu- nications band	Change			
CIP message communications	CIPOpenWithData- Size instruction	Addition	8-5	1.06	1.07
Socket services	Number of sup- ported sockets	Change	9-11	1.03	1.04
FTP client		Addition	11-1	1.08	1.09
Troubleshooting	Tag Data Link Con- nection Timeout	Addition	15-37	1.04	1.05
Connection settings		Addition	A-3	1.09	1.10

Appendices



# Index

# Index

A	connection settings
	automatically setting connections7-42
Accept TCP Socket instruction9-1	3 batch editing7-39
addresses5-1	aditing individual connections 7.20
array variables	Register Device List7-36
preparing array variables to input	connection status
and output service data and response data 8-2	ocodes and troubleshooting15-58
assumed causes	Connection Tob Dogo 15 EE
Auto Connection Configuration7-4	O : T 700 740
automatically setting connections7-4	aannaatiana
automationly dotting dominoutons	checking7-74
В	Controller Event Log Tab Page15-56
	Controller Object8-58
handwidth usage	Controller status7-10
bandwidth usage	Create UDP Socket instruction9-13
requested packet intervals	avalia communicatione 1.16
tag data links	
binary format	13
BOOTP client1-1	
broadcasting	
built-in EtherCAT port1-1	
Built-in EtherNet/IP Error3-3, 3-33, 15-	da oo
built-in EtherNet/IP port1-1	14.22
settings5	data transmissions
socket services	
overview9-1	50.54
specifications1	destination IP addresses5-4
	destination in addresses
С	device bandwidth usage
	adjusting14-9
CIDR4	Device Connection Structure Tree
CIP Communications1-1	5 Device Connection Structure free
CIP Communications Error3-36, 15	• /
CIP communications errors15	device parameters
CIP communications instructions8	clearing
using8	6 verifying
CIP message communications8	4 device status
CIP message communications service	displaying7-76
overview8	3 devices
specifications8	3 changing
Clear UDP/TCP Socket Receive Buffer instruction 9-1	registering7-21
clock	DNS
automatic adjustment1-20, 12-	domain names5-3
required settings12	4 _
specifications12-3, 12-	<b>E</b>
Close CIP Class 3 Connection instruction8	5
Close UDP/TCP Socket instruction9-1	EDC files
communications load	managementA-44
adjusting14	6EIP2_TDLinkStartCmd7-67
Communications Port Error3-4, 3-34, 15-	
communications port errors	
community names5-11, 5-1	EID EOt- 0.0.00 4E 7
Connect TCP Socket instruction9-1	FID FothTorgatCta 15 E4
Connection I/O Type7-39, 7-4	
Connection Name	

\_EIP\_TargetPLCModeSta ......7-10, 15-54

_EIP_TcpAppErr	15-7	FTP server	1-19, 5-7
_EIP_TDLinkStartCmd		application example	10-6
_EIP_TDLinkStopCmd	7-67	application example from host computer	10-18
error status	3-19	application procedure	10-5
error status variables		commands	
errors		overview	
checking communications with Network	work Configurator	specifications	
		functions	
checking for errors	15-3	comparison between NJ Series and Other S	Series
classifications	15-8		A-2
descriptions	15-18		
event levels	15-9, 15-10	G	
indicators	15-3		
reading with instructions	15-7	gateway addresses	5-4
sources	15-8	general status	
Troubleshooter of an HMI	15-6	general status codes	
troubleshooting with Sysmac Studio	o 15-6	Get EtherNet/IP Error Status instruction	
Ethernet addresses		global addresses	
Ethernet connectors		global broadcasting	
Ethernet Information Tab Page		global broadcasting	5 10
Ethernet Link Object		Н	
Ethernet switches			
functions			5 44 5 40
selection precautions		host names5-3, 5-8	
types		specifying	1-21
event codes		•	
event codesevent names		I	
events			
events	10-2	Identity Object	
F		indicators1-10	), 1-13, 15-3
Г		information event level	15-9
		input ON response time	14-25
FINS/TCP Connection Status		instructions	
FINSTCPConnSta	3-32	CIPClose	8-5
fragmenting		CIPOpen	8-5
send data	9-5	CIPOpenWithDataSize	8-5
FTP commands		CIPRead	8-5
append	10-11	CIPSend	
bye	10-13	CIPUCMMRead	8-5
cd	10-11	CIPUCMMSend	
close	10-13	CIPUCMMWrite	
delete	10-13	CIPWrite	
dir	10-10	GetEIPError	
get	10-12	reading errors	
ls		SktClearBuf	
mdelete			
mdir		SktGetTCPStatus	
mget		SktTCPAccept	
mkdir		SktTCPConnect	
		SktTCPRcv	
mls		SktTCPSend	
mput		SktUDPCreate	
open		SktUDPRcv	
put		SktUDPSend	9-13
pwd		interval	
quit	10-14	NTP operation timing	5-8
rename	10-10	IOI	
rmdir	10-11	IP address configuration	
type	10-12		
user			

FTP Display ......5-7

IP addresses 5-2, 5-3, 5-8, 5-11, 5		
allocation		
checking current addresses4	observation event level	15-9
setting	.4-5 Originator Variable	7-40
17	output ON response time	14-26
K	output variables	
	operation and timing	8-34, 9-17
Keep Alive Monitoring Time setting		
Keep Alive setting	5-4 <b>P</b>	
L	Packet Interval (RPI)	7-39
	partial fault event level	
Linger option		
LINK settings		
LINK/ACT indicator		
local broadcasting		
location	precautions	14-3
SNMP		0_30
login names	COCKET COLVICOS	
10g.11 10a.1100	p, 2.12 co. 10.	
M	private addresses	4-12
	programming	0.00
major fault avant laval	CIP message communications	
major fault event level		
message communications	1-17 socket services	9-19, 9-24
message service	1-27 <b>R</b>	
transmission delay14 MIB		
groups1		9-13
system design1	Read Variable Class 3 Explicit instruction .	8-5
MIB objects	Read Variable UCMM Explicit instruction	8-5
detailed descriptions1		14-26
minor fault event level1	Recognition 1 settings	5-11
multi-cast communications	7-9 Recognition 2 settings	5-11
multicast filtering	.2-3 recognition method	5-11
	recognition trap	
N	sending	5-10
	registering devices	7-21
NET ERR indicator	I-13 request path	8-18
NET RUN indicator	1-13 requested packet intervals	7-8, 14-26
network configuration	accuracy	14-5
verifying7	7-63 adjusting according to task periods	14-24
network configuration file	bandwidth usage	14-3
reading7	7-72 changing	14-10
saving7	7-71 settings	14-2
Network Configurator		
connections through NJ-series CPU Unit's USB por		
	7-54 RPI setting	7-40
connections via Ethernet to NJ-series CPU Unit7	7-55	
Ethernet connections7	7-52 <b>S</b>	
network transmission delay time14	1-26	
network variables	7-7 SD Memory Cards	
importing to Network Configurator7		10-16
Normal Target Node Information15	1, pee	
NTP operation timing		
NTP server clock information		
NTP Settings Display	4,000	
number of connections	send data processing time	
calculating1		
•	server specifying method	

settings	5-1	tag data links	1-15
SNMP	5-10	automatically starting	7-67
SNMP agent	1-22, 13-2	bandwidth usage and requested packet interva	als
application procedure			
required settings		communications method	
SNMP messages		data concurrency	
SNMP Settings Display		effect on task periods	14-24
SNMP specifications		introduction	
SNMP Trap Settings Display	5-12	maximum response time	
SNMP traps		models other than NJ-series CPU Units	
socket service	1-21	setting	
socket service communications		starting and stopping	
data reception processing	9-7	starting and stopping for individual devices	7-67
fragmenting of send data	9-5	verifying	
socket service instructions	9-13	tag sets	7-3
socket services		creating	7-23
application procedure	9-14	Tag Status Tab Page	15-57
built-in EtherNet/IP port		tags	7-3
overview	9-11	Target Device	7-40
port numbers	9-2	Target PLC Error Information7-	10, 15-54
precautions	9-30	Target PLC Operating Mode7-	10, 15-54
required settings	9-12	Target Variable	7-40
sockets	9-2	task periods	
specifying method	5-13	relationship with requested packet intervals	14-24
Status 1 Tab Page	15-52	TCP Application Communications Error	
Status 2 Tab Page		TCP application errors	
structure variables		TCP communications	
providing to input request paths	8-19	procedure	
subnet masks		TCP Socket Receive instruction	
Sysmac Studio		TCP Socket Send instruction	
system-defined variables		TCP/IP Display	
EtherNet/IP communications errors		TCP/IP Interface Object	
EtherNet/IP communications status		time	
EtherNet/IP communications switches		NTP operation timing	5-8
		timeout time	
T		NTP operation timing	5-8
		Timeout Value7	
tag data link		Trap 1 settings	
functions	7.6	Troubleshooter	
specifications		troubleshooting	
•	7-0	twisted-pair cable	
tag data link parameters	7 57	installation environment precautions	
downloading		installation precautions	
setting and downloading		installation precautions	2-0
uploading all		U	
uploading from individual devices			
Tag Data Link Start Switch		LIDD : r:	0.0
Tag Data Link Stop Switch	/-6/	UDP communications	
		UDP Socket Receive instruction	
		UDP Socket Send instruction	
		unicast communications	
		USB port	1-10
		V	
		V	

variable memory allocation

### W

Windows firewall settings	
changing	A-49
Write Variable Class 3 Explicit instruction	8-5
Write Variable UCMM Explicit instruction	8-5

**OMRON Corporation Industrial Automation Company** 

Tokyo, 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.

Owner Asia Pacific PTE. 11D.

No. 438A Alexandra Road # 05-05/08 (Lobby 2),
Alexandra Technopark,
Singapore 119967
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:** 

© OMRON Corporation 2011 All Rights Reserved. In the interest of product improvement, specifications are subject to change without notice.

Cat. No. W506-E1-11