OMRON

EJ1 EtherCAT® Slave Unit

EJ1N-HFUC-ECT

User's Manual

Cat. No. H192-E1-01

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.
- EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

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 EJ1 EtherCAT Slave Unit.

This manual provides information required to use the EJ1 EtherCAT Slave Unit, including information on functions, performance, and application methods.

Observe the following items when you use the EJ1 EtherCAT Slave Unit.

- Allow only a specialist with knowledge of electrical systems to handle the EJ1 EtherCAT Slave Unit.
- Read and understand this user's manual completely and use the EJ1 EtherCAT Slave Unit correctly.
- Keep this user's manual in a safe place where it will be available for reference when required.

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

Notation and Definitions for Precautionary Information

The following notation is used in this manual to provide precautions required to ensure safe usage of the EJ1 EtherCAT Slave Unit.

The safety precautions that are provided here are extremely important to safety. Always read and heed information provided in all safety precautions.

The following notation is used.

● Definitions of Warning and Caution

⚠ WARNING	Indicates a potentially hazardous situation which, if not avoided, is likely to result in slight or moderate injury or occasionally, death or serious injury. And serious property damage may occur as well.
⚠ CAUTION	Indicates a potentially hazardous situation which, if not avoided, is likely to result in minor or moderate injury or property damage.

Symbols

Symbol		Meaning
Caution		• General Caution Indicates non-specific general cautions, warnings, and dangers.
		• Electrical Shock Caution Indicates possibility of electric shock under specific conditions.
Prohibition		• General Prohibition Indicates non-specific general prohibitions.
		• Disassembly Prohibition Indicates prohibitions when there is a possibility of injury, such as from electric shock, as the result of disassembly.
Mandatory Caution		• General Caution Indicates non-specific general cautions, warnings, and dangers.

Warnings and Cautions

⚠ WARNING

Never use the product without installing protective circuit in network. Doing so may possibly cause abnormal operation, and result in serious injury, property damage, or accident. To operate your total system safely even if any equipment failure occurs, or any trouble is caused by an external element, be sure to configure an external-control-circuit that consists of emergency stop, interlock and limit circuits to provide double or triple safeguard.

Do not attempt to take the Unit apart. In particular, parts with high power supply voltages are present in Units that supply power while power is supplied or immediately after power is turned OFF. Touching any of these parts may result in electric shock. There are sharp parts inside the Unit that may cause injury.

⚠ CAUTION

Do not touch the terminals while power is being supplied.

Doing so may occasionally result in minor injury due to electric shock.

Use power supplies that comply with the reinforced insulation specified in IEC 60664 for the EJ1 external power supply and for the power supplies for devices connected to the EJ1. If non-compliant power supplies are used, electric shock may occasionally result in minor injury.

Do not allow pieces of metal, wire clippings, or fine metallic chips generated during installation to enter the product. Doing so may occasionally result in electric shock, fire, or malfunction.

Do not use the product where subject to flammable or explosive gas.

Otherwise, minor injury from explosion may occasionally occur.

Never disassemble, modify, or repair the product or touch any of the internal parts. Minor electric shock, fire, or malfunction may occasionally occur.

Execute online editing only after confirming that no adverse effects will be caused by deviations in the timing of I/O. If you perform online editing, the task execution time may exceed the task period, I/O may not be refreshed with external devices, input signals may not be read, and output timing may change.

Always confirm safety at the destination node before you transfer Unit configuration information, parameters, settings, or other data from tools such as the Sysmac Studio.

The devices or machines may operate unexpectedly, regardless of the operating mode of the controller.

Precautions for Safe Use

- The product is designed for indoor use only. Do not use the product outdoors or in any of the following locations.
  - Places directly subject to heat radiated from heating equipment.
  - Places subject to splashing liquid or oil atmosphere.
  - Places subject to direct sunlight.
  - Places subject to dust or corrosive gas (in particular, sulfide or ammonia gas)
  - Places subject to intense temperature change.
  - Places subject to icing or condensation.
  - Places subject to vibration or strong shocks.
- Use and store the product within the rated temperature and humidity ranges. Provide forced-cooling if required.
- To allow heat to escape, do not block the area around the temperature controller. Also, do not block its ventilation holes.
- Be sure to wire properly with correct polarity of terminals.
- Do not wire terminals that do not have an identified use.
- Secure as much space as possible between the product and devices that generates a strong high-frequency or surge. Separate the high-voltage or large-current power lines from other lines, and avoid parallel or common wiring with the power lines when you are wiring to the terminals.
- Use the product within the rated load and power supply voltage.
- Make sure that the rated voltage is attained within two seconds of turning ON the power.
- The switch or circuit breaker must be located within an easy reach of the operator, and must be marked as a disconnecting means for this unit.
- Do not use paint thinner or similar chemical to clean with. Use standard grade alcohol.
- Never touch the electric components, connectors, or patterns in the product with bare hands. Always hold the product by its enclosure. Inappropriate handling of the product may occasionally damage internal components due to static electricity.
- Use a switch, relay, or other device for turning OFF the power supply quickly. Gradually lowering the voltage of the power supply may result in incorrect outputs or memory errors.
- Connect only the specified number of products in only a specified configuration.
- Mount the product to a DIN Rail mounted vertically to the ground.
- Make sure that the data transfer distance for EtherCAT is within the specified range, and use the specified cables only.
- Do not bend a communications cable past its natural bending radius or pull in it with excessive force. Also, do not place heavy objects on top of the cables or other wiring lines. Doing so may break the cable.
- When transporting any Unit, use the special packing box for it. Also, do not subject the Unit to excessive vibration or shock during transportation.
- Do not drop any Unit or subject it to abnormal vibration or shock. Doing so may result in Unit malfunction or burning.
- Be sure that connectors with locking devices are properly locked into place.
- When wiring or installing the Units, do not allow metal fragments to enter the Units.
- Double-check all switches and other settings and double-check all wiring to make sure that they are correct before turning ON the power supply. Use the correct wiring parts and tools when you wire the system.
- Always turn OFF the external power supply to the Units before attempting any of the following.
  - Mounting, removing, or replacing the Unit
  - Setting the rotary switches

  - Connecting cables, wiring, or changing the configuration
- If you change the fail-soft operation setting, the output status when the error occurs may also change. Confirm safety before you change the fail-soft operation setting.
- Confirm that the controlled system will not be adversely affected before you perform any of the following operations.
  - Changing the operating mode of the CPU Unit (including changing the setting of the Operating Mode at Startup)
  - Changing the user program or settings
  - Changing set values or present values
  - Forced refreshing
- Dispose of the product according to local ordinances as they apply.
- Do not allow foreign matter to enter the openings in the Unit. Doing so may result in Unit burning, electric shock, or failure.
- Before installing the Unit, do not remove the enclosed cover seal from the connector opening on the left end of the Unit.
- You can connect up to 16 Units for one End Unit. However, you can connect only one Advanced Unit. Do not connect more than 16 Units.
- You can connect up to 32 Basic Units with branch wiring. Do not connect more than 32 Basic Units.

Observe the following precautions for EtherCAT communications.

- Make sure that the communications distance, number of nodes connected, and method of connection for EtherCAT are within specifications.

  Do not connect EtherCAT Coupler Units to EtherNet/IP, a standard in-house LAN, or other networks.

  An overload may cause the network to fail or malfunction.
- Malfunctions or unexpected operation may occur for some combinations of EtherCAT revisions of the master and slaves. If you disable the revision check in the network settings, check the slave revision settings in the master and the actual slave revisions, and then make sure that functionality is compatible in the manuals or other references. You can check the slave versions in the settings from the Sysmac Studio and you can check the actual slave revisions from the Sysmac Studio or on slave nameplates.
- After you transfer the user program, the CPU Unit is restarted and communications with the EtherCAT slaves are cut off. During that period, the slave outputs behave according to the slave settings.

  The time that communications are cut off depends on the EtherCAT network configuration.

  Before you transfer the user program, confirm that the system will not be adversely affected.
- EtherCAT communications are not always established immediately after the power supply is turned ON. Use the system-defined variables in the user program to confirm that communications are established before attempting control operations.
- If frames sent to EtherCAT slaves are lost due to noise or other causes, slave I/O data is not communicated, and the intended operation is sometimes not achieved. Perform the following processing if noise countermeasures are necessary.

  Program the *_EC_InDataInvalid* (Input Data Invalid), *_EC_InData1Invalid* (Input Data 1 Invalid), and *_EC_InData2Invalid* (Input Data 2 Invalid) system-defined variables as interlock conditions in the user program.

  Set the *PDO communications consecutive timeout detection count* setting in the EtherCAT master to at least 2.

  Refer to the *NJ/NX-series CPU Unit Built-in EtherCAT Port User's Manual* (Cat. No. W505) for details.
- When an EtherCAT slave is disconnected, communications will stop and control of the outputs will be lost not only for the disconnected slave, but for all slaves connected after it. Confirm that the system will not be adversely affected before you disconnect a slave.
- If you disconnect the cable from an EtherCAT slave to disconnect it from the network, any current communications frames may be lost. If frames are lost, slave I/O data is not communicated, and the

intended operation is sometimes not achieved. Perform the following processing for a slave that needs to be replaced.

Program the *_EC_InDataInvalid* (Input Data Invalid), *_EC_InData1Invalid* (Input Data 1 Invalid), and *_EC_InData2Invalid* (Input Data 2 Invalid) system-defined variables as interlock conditions in the user program.

Set the *PDO communications consecutive timeout detection count* setting in the EtherCAT master to at least 2.

Refer to the *NJ/NX-series CPU Unit Built-in EtherCAT Port User's Manual* (Cat. No. W505) for details.

- Make sure that the PDO recording area for EtherCAT communications is correct.

Precautions for Correct Use

Installation

- Connect the EJ1 EtherCAT Slave Unit to the left side of a Basic Unit or an End Unit.
- The EJ1 cannot be used linked to a CJ-series PLC.

Service Life

- Use the product within the following temperature and humidity ranges.

  Temperature: −10 to 55°C (with no icing or condensation)

  Humidity: 25% to 85%

  When the Temperature Controller is incorporated in a control panel, make sure that the controller's ambient temperature and not the panel's ambient temperature does not exceed 55°C.
- The service life of electronic devices like the Temperature Controller is determined by the service life of internal electronic components. Component service life is affected by the ambient temperature: the higher the temperature, the shorter the service life and the lower the temperature, the longer the service life. Therefore, the service life can be extended by lowering the temperature of the Temperature Controller.
- Mounting two or more Temperature Controllers side by side, or mounting Temperature Controllers above each other may cause heat to build up inside the Temperature Controllers, which will shorten their service life. If the Temperature Controllers are mounted above each other or side by side, use forced cooling by fans or other means of air ventilation to cool down the Temperature Controllers.

  However, be sure not to cool only the terminals. Doing so will result in measurement errors.
- Take appropriate and sufficient countermeasures when installing the controller in the following locations.

  Locations subject to static electricity or other forms of noise

  Locations subject to strong electromagnetic fields

  Locations subject to possible exposure to radioactivity

  Locations close to power lines

Precautions for Operation

- It takes a certain amount of time for the outputs to turn ON from after the power supply is turned ON. Due consideration must be given to this time when designing control panels, etc.
- It takes 30 minutes from the time the product is turned ON until the correct temperature is indicated. Always turn ON the power supply at least 30 minutes before starting temperature control.
- Avoid using the Temperature Controller near a radio, television set, or other wireless device. Its use would result in reception disturbance.

Preparations for Use

Be sure to thoroughly read and understand the manual provided with the product, and check the following points.

Timing	Check point	Details
Purchasing the product	Product appearance	After purchase, check that the product and packaging are not dented or otherwise damaged. Damaged internal parts may prevent optimum control.
	Product model and specifications	Make sure that the purchased product meets the required specifications.
Setting the Unit	Product installation location	Provide sufficient space around the product for heat dissipation. Do not block the vents on the product.
Wiring	Terminal wiring	Do not subject the terminal screws to excessive stress (force) when tightening them. Make sure that there are no loose screws after tightening terminal screws to the specified torque of 0.25 to 0.30 N·m.
		Be sure to confirm the polarity for each terminal before wiring the terminal block and connectors.
	Power supply inputs	Wire the power supply inputs correctly. Incorrect wiring will result in damage to the internal circuits.
Operating environment	Ambient temperature	The ambient operating temperature for the product is −10 to 55°C (with no condensation or icing). To extend the service life of the product, install it in a location with an ambient temperature as low as possible. In locations exposed to high temperatures, if necessary, cool the products using a fan or other cooling method.
	Vibration and shock	Check whether the standards related to shock and vibration are satisfied at the installation environment. (Install the product in locations where the conductors will not be subject to vibration or shock.)
	Foreign particles	Install the product in a location that is not subject to liquid or foreign particles entering the product. If sulfide, chlorine, or other corrosive gases are present, remove the source of the gas, install a fan, or use other countermeasures to protect the product.

Regulations and Standards

Conformance to EU Directives

Applicable Directives

EMC Directives

Concepts

EMC Directives

OMRON devices that comply with EU 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 EU 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.

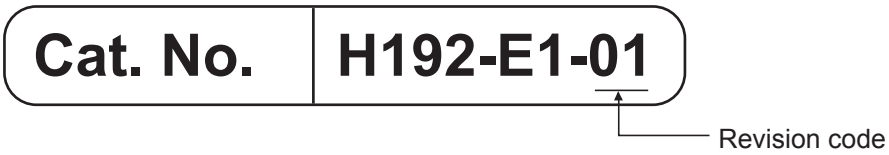
Conformance to EU Directives

To ensure that the machine or device in which the EJ-series Unit is used complies with EU Directives, the Unit must be installed as follows:

- The EJ1 EtherCAT Slave Unit must be installed within a control panel.
- You must use reinforced insulation or double insulation for the DC power supplies used for the communications power supply, internal power supply, and I/O power supplies.
- EJ-series Units that comply with EU Directives also conform to the Common Emission Standard (EN 61326). 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 EU Directives.
- The Unit is a Class A product (products for industrial environments). In residential environment areas it may cause radio interference, in which case the user may be required to take adequate measures to reduce interference.

Revision History

A manual revision code appears as a suffix to the catalog number at the bottom of the back cover of the manual.



Revision code	Date	Revised content
01	October 2015	Original production

Related Manuals

The manuals related to the EJ1 EtherCAT Slave Unit are configured as shown in the following tables. Refer to these manuals as required.

EJ1

Cat. No.	Model number	Manual name	Application	Description
H192 (This manual)	EJ1N-HFUC-ECT	EJ1 EtherCAT® Slave Unit User's Manual	Learning the basic specifications of the EJ1 EtherCAT Slave Unit, including introductory information, designing, and maintenance.	The following information is provided on the EJ1 EtherCAT Slave Unit. • Overview and features • System configuration • Installation and wiring • Troubleshooting
H142	EJ1N-TC2□ EJ1N-TC4□ EJ1C-EDU□	EJ1 Modular Temperature Controllers User's Manual	Learning the basic specifications of the EJ1 Temperature Controllers, including introductory information, designing, installation, and maintenance.	The following information is provided on the EJ1 Temperature Controllers. • Overview and features • System configuration • Mounting and wiring • Troubleshooting
--- Online Help	EST2-2C-MV4	CX-Thermo Ver. 4.□ (online help)	Learning the operating procedures of the CX-Thermo.	Describes how to set parameters and adjust devices (i.e., components such as Temperature Controllers) using the CX-Thermo.

NJ/NX-series Master Units

Cat. No.	Model number	Manual name	Application	Description
W505	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	NJ/NX-series CPU Unit Built-in EtherCAT® Port User's Manual	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 information on the configuration, features, and setup. Use this manual together with the *NJ-series CPU Unit Hardware User's Manual* (Cat. No. W500) and *NJ/NX-series CPU Unit Software User's Manual* (Cat. No. W501).
W500	NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	NJ-series CPU Unit Hardware User's Manual	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).

Cat. No.	Model number	Manual name	Application	Description
W535	NX701-□□□□	NX-series CPU Unit Hardware User's Manual	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).
W501	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	NJ/NX-series CPU Unit Software User´s Manual	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 • Use this manual together with IEC 61131-3-based programming language specifications and the *NJ-series CPU Unit Hardware User's Manual* (Cat. No. W500).
W502	NX701-□□□□ NJ501-□□□□ NJ301-□□□□ NJ101-□□□□	NJ/NX-series Instructions Reference Manual	Learning detailed specifications on the basic instructions 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 *NJ-series CPU Unit Hardware User's Manual* (Cat. No. W500) and *NJ/NX-series CPU Unit Software User's Manual* (Cat. No. W501).
W504	SYSMAC-SE2□□□	Sysmac Studio Version 1 Operation Manual	Learning about the operating procedures and functions of the Sysmac Studio.	Describes the operating procedures of the Sysmac Studio.

G3ZA Multi-channel Power Controllers

Cat. No.	Model number	Manual name	Application	Description
Z200	G3ZA-4H203-FLK-UTU G3ZA-4H403-FLK-UTU G3ZA-8A203-FLK-UTU G3ZA-8A403-FLK-UTU	G3ZA Multi-channel Power Controller User's Manual	Using the G3ZA to perform high-precision control of heater power with RS-485 serial communications (CompoWay/F) from an EJ1 or PLC.	The following information is provided on the G3ZA. • Overview and features • Installation and wiring • Setting RS-485 serial communications • Basic functions

G3PW Power Controllers

Cat. No.	Model number	Manual name	Application	Description
Z280	G3PW-A220EC-C-FLK G3PW-A230EC-C-FLK G3PW-A245EC-C-FLK G3PW-A260EC-C-FLK G3PW-A220EC-S-FLK G3PW-A230EC-S-FLK G3PW-A245EC-S-FLK G3PW-A260EC-S-FLK	G3PW-series Power Controller User's Manual	Using the G3PW under phase control or cyclic control to perform control that achieves higher precision than with the G3ZA on heater power with a continuous proportional output or with RS-485 serial communications (CompoWay/F) from an EJ1.	The following information is provided on the G3PW. • Overview and features • Installation and wiring • Setting RS-485 serial communications • Basic functions

Meanings of Abbreviations and Terms

The following abbreviations and terms are used in this manual.

Abbreviation or term	Meaning
CompoWay/F commands/responses	Also called “FINS-mini commands/responses.” This manual uses the term “CompoWay/F commands/responses.”
EDU	EJ1 End Unit
Basic Unit	EJ1 Basic Unit • TC4: Four-channel Basic Unit • TC2: Two-channel Basic Unit
Advanced Unit	EJ1 EtherCAT Slave Unit
Temperature Controller	One of the EJ1 Basic Units given above.
communications unit number	An identification number in CompoWay/F communications for a Temperature Controller connected to an EJ1 EtherCAT Slave Unit. Set the communications unit number with rotary switch 1 and DIP switch 2 on each Basic Unit.

Sections in this Manual

1 Features and System Configuration

I Index

2 Specifications and Procedures

3 Part Names and EtherCAT Communications Overview

4 Installation and Connection

5 Using the EJ1 EtherCAT Slave Unit with PDO Communications

6 Using the EJ1 EtherCAT Slave Unit with SDO Communications

7 Troubleshooting

8 Maintenance and Inspection

A Appendices

CONTENTS

Introduction 1

Terms and Conditions Agreement 2
Warranty, Limitations of Liability 2
Application Considerations 3
Disclaimers 3

Safety Precautions 4
Notation and Definitions for Precautionary Information 4
Symbols 4
Warnings and Cautions 5

Precautions for Safe Use 6

Precautions for Correct Use 9

Preparations for Use 10

Regulations and Standards 11
Conformance to EU Directives 11

Revision History 12

Related Manuals 13
EJ1 13
NJ/NX-series Master Units 13
G3ZA Multi-channel Power Controllers 14
G3PW Power Controllers 14

Meanings of Abbreviations and Terms 15

Sections in this Manual 17

CONTENTS 18

Section 1 Features and System Configuration

1-1 Overview and Features 1-2
1-1-1 Overview 1-2
1-1-2 Features 1-2

1-2 System Configuration 1-3

1-3 Unit Models and Overview of Functions 1-5
1-3-1 Unit Model Number 1-5
1-3-2 Overview of Functions 1-5

Section 2 Specifications and Procedures

2-1 Specifications 2-2
2-1-1 General Specifications 2-2
2-1-2 Function Specifications 2-2
2-1-3 EtherCAT Slave Communications Specifications 2-3

2-2 Application Procedures 2-4
2-2-1 Hardware Setup Procedure 2-4
2-2-2 Software Setup Procedure 2-5

Section 3 Part Names and EtherCAT Communications Overview

3-1 Part Names and Functions 3-2
3-1-1 Appearance and Names of Parts on the Front Panel 3-2
3-1-2 Meanings of Indicators 3-3
3-1-3 Node Address Setting Switches 3-3
3-1-4 EtherCAT Communications Connectors 3-4
3-2 EtherCAT Communications Overview 3-5
3-2-1 Structure of CAN Application Protocol over EtherCAT (CoE) 3-5
3-2-2 EtherCAT Slave Information File (ESI Files) 3-6
3-2-3 State Transitions for EtherCAT Communications 3-7
3-2-4 Process Data Objects (PDOs) 3-8
3-2-5 Service Data Objects (SDOs) 3-10

Section 4 Installation and Connection

4-1 Installing the EJ1 EtherCAT Slave Unit 4-2
4-2 EtherCAT Network Wiring 4-3
4-2-1 Installation Precautions 4-3
4-2-2 Preparations for Installation 4-3
4-2-3 Recommended EtherCAT Cables 4-4
4-3 Wiring the Temperature Controllers 4-5

Section 5 Using the EJ1 EtherCAT Slave Unit with PDO Communications

5-1 EJ1 PDO Communications Overview 5-2
5-2 How Data Is Sent and Received with PDO Communications 5-3
5-2-1 CompoWay/F Commands and Responses 5-3
5-2-2 Control Commands and Responses 5-5
5-3 PDO Mapping Data Overview 5-6
5-3-1 Types of PDO Mapping Data 5-6
5-3-2 PDO Mapping Data List 5-7
5-4 PDO Mapping Data Details 5-8
5-4-1 Output Notification Data (Communications Master to EJ1 EtherCAT Slave Unit) 5-8
5-4-2 Input Notification Data (EJ1 EtherCAT Slave Unit to Communications Master) 5-10
5-4-3 Sysmac Error Status 5-14
5-5 Writing User Programming 5-15
5-5-1 Handshaking with the EJ1 EtherCAT Slave Unit with User Programming 5-15
5-5-2 Handshaking with the EJ1 EtherCAT Slave Unit with User Programming 5-16
5-5-3 Procedures for Sending Control Commands and Receiving Responses 5-17
5-6 Example Procedures for Sending CompoWay/F Commands and Receiving Responses 5-18
5-6-1 Example of Sending and Receiving Data without Dividing into Parts 5-18
5-6-2 Example of Sending and Receiving Data in Parts 5-21
5-7 Setting CompoWay/F Frames 5-25
5-7-1 Output Data and Input Data Contents 5-25
5-7-2 Storing Output Data and Input Data in Array Variables 5-27
5-7-3 Setting the PDO Mappings for I/O Data 5-28
5-8 Assigning Device Variables 5-32

Section 6 Using the EJ1 EtherCAT Slave Unit with SDO Communications

6-1 Overview **6-2**
6-2 Abort Codes **6-3**

Section 7 Troubleshooting

7-1 Error Notification Methods **7-2**
7-2 Error List **7-3**
7-2-1 Errors Detected during Boot Process 7-3
7-2-2 EtherCAT Communications Errors 7-4
7-2-3 Errors Detected during a Software Process 7-5
7-3 Error Notifications Based on the Sysmac Error Status **7-6**
7-3-1 Sysmac Error Status 7-6
7-3-2 Checking and Responding to Event Codes 7-7
7-4 Error Notifications Provided with Emergency Messages **7-8**
7-4-1 Emergency Messages 7-8
7-5 Error Notifications Based on the AL Status **7-9**

Section 8 Maintenance and Inspection

8-1 Cleaning and Inspection **8-2**
8-1-1 Cleaning 8-2
8-1-2 Inspection Methods 8-2
8-1-3 Inspection Items 8-3
8-2 Replacing Units **8-4**
8-2-1 Unit Replacement Precautions 8-4
8-2-2 Unit Replacement Procedure 8-4

Appendices

A-1 Dimensions **A-2**
A-1-1 EJ1N-HFUC-ECT A-2
A-2 Sample Programming **A-3**
A-2-1 Items Common to All Programming Samples A-3
A-2-2 Sending CompoWay/F Commands and Receiving Responses A-6
A-2-3 Restart Port A-8
A-3 CoE Objects **A-9**
A-3-1 Object Dictionary Area A-9
A-3-2 Data type A-9
A-3-3 Communications Objects A-10
A-3-4 PDO Mapping Objects A-11
A-3-5 Sync Manager Communications Objects A-14
A-3-6 Manufacturer-specific Object 1 ..... A-15
A-3-7 Device Profile Area A-17

Index

1

Features and System Configuration

This section describes the features of the EJ1 EtherCAT Slave Unit and the system configuration in which the EJ1 EtherCAT Slave Unit is used.

1-1 Overview and Features 1-2
1-1-1 Overview 1-2
1-1-2 Features 1-2
1-2 System Configuration 1-3
1-3 Unit Models and Overview of Functions 1-5
1-3-1 Unit Model Number 1-5
1-3-2 Overview of Functions 1-5

1-1 Overview and Features

This section introduces the EJ1 EtherCAT Slave Unit and describes its features.

1-1-1 Overview

The EJ1 EtherCAT Slave Unit is a communications unit used to connect EJ1 Modular Temperature Controllers as slaves on an EtherCAT network.

You can send CompoWay/F commands from a user program through PDO communications to perform handshaking with an EJ1 EtherCAT Slave Unit in order to write/read values in EJ1 Basic Units connected to the EJ1 EtherCAT Slave Unit.

1-1-2 Features

The EJ1 EtherCAT Slave Unit has the following features.

- You can connect up to 15 Basic Units in EJ1 Modular Temperature Controllers to communicate between the EtherCAT communications master and the Basic Units. If you use distributed placement, you can connect up to 32 Basic Units.
- You can send the required commands with the CompoWay/F communications protocol for OMRON components from the EtherCAT communications master to read/write present values and set values or to perform other processing.

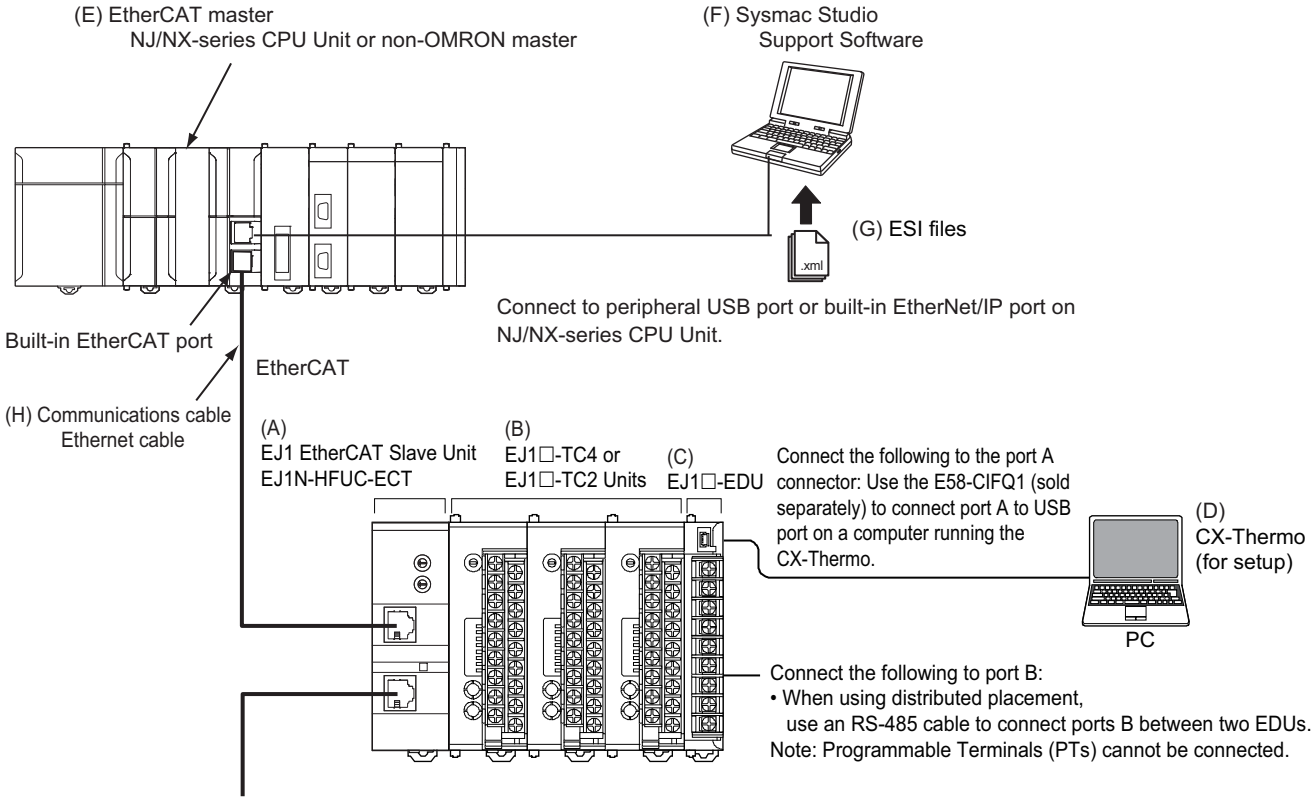
1-2 System Configuration

This section describes the overall system configuration in which an EJ1 EtherCAT Slave Unit is used.

The EJ1 EtherCAT Slave Unit is connected to an EtherCAT network that is connected to an NJ/NX-series CPU Unit or a non-OMRON EtherCAT communications master.

The EJ1 EtherCAT Slave Unit is connected as the far left module, and EJ1 Basic Units (TC4/TC2) are connected on the right of it. An EJ1 End Unit (EDU) is connected as the far right module.

The elements in the system configuration are described in the following system configuration diagram.



A maximum of 15 Basic Units can be connected side by side next to an EJ1 EtherCAT Slave Unit to perform EtherCAT communications through the EJ1 EtherCAT Slave Unit.

If you use distributed placement through a RS-485 communications cable from the EDU, you can connect a maximum of 32 Basic Units next to the EJ1 EtherCAT Slave Unit to perform EtherCAT communications through the EJ1 EtherCAT Slave Unit.

Refer to the *EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142) for information on distributed placement.

Letter	Item	Description
(A)	EJ1 EtherCAT Slave Unit	This Unit is an EJ1 Advanced Unit.
(B)	EJ1 Basic Units	The Basic Units are modular temperature controllers with I/O. There are two models of Basic Units: The TC4 with four I/O channels and the TC2 with two I/O channels. You can connect up to eight G3ZA Multi-channel Power Controllers or V1.1 G3PW Power Controllers to each Basic Unit.
(C)	EJ1 End Unit (EDU)	This Unit provides power to the connected Basic Units and Advanced Unit. The End Unit must be connected as the far right module when you use the EJ1 EtherCAT Slave Unit. You can connect up to 16 Units for one End Unit. However, you can connect only one Advanced Unit.
(D)	CX-Thermo (EST2-2C-MV□)	This computer software application is used to set up and monitor the EJ1 Basic Units. An E58-CIFQ1 USB-Serial Conversion Cable is required to connect the CX-Thermo. You can also use a K3SC-10 Interface Converter to convert to RS-232C/RS-485 and thereby enable connecting to more than one EJ1 EtherCAT Slave Unit at the same time.
(E)	EtherCAT communications master	This is an EtherCAT communications master, such as an NJ/NX-series CPU Unit or a communications master from another manufacturer.
(F)	Sysmac Studio Support Software	This computer software application is used to set up the EtherCAT network and make other settings. It is used to register an EJ1 EtherCAT Slave Unit in the EtherCAT network configuration. The Sysmac Studio is also used to set the PDO mappings for the EtherCAT communications master. To change the default PDO sizes, edit the PDO mappings in the communications master of the EJ1 EtherCAT Slave Unit in EtherCAT slave settings on the Sysmac Studio.
(G)	ESI files	ESI files contain information that is unique to the EJ1 EtherCAT Slave Unit in XML format. ESI files are required to connect OMRON EtherCAT slaves to controllers from other companies. To connect an EJ1 EtherCAT Slave Unit to a controller from another company, install the ESI file for the EJ1 EtherCAT Slave Unit in the configuration software that you are using. For details on ESI files, refer to *3-2-2 EtherCAT Slave Information File (ESI Files)* on page 3-6
(H)	EtherCAT communications cable	Use a double-shielded cable with aluminum tape and braiding of Ethernet category 5 (100BASE-TX) or higher, and use straight wiring.

1-3 Unit Models and Overview of Functions

This section introduces the EJ1 EtherCAT Slave Unit model number and provides an overview of the functions of the EJ1 EtherCAT Slave Unit.

1-3-1 Unit Model Number

The model number and specifications for the EJ1 EtherCAT Slave Unit are given in the following table.

Unit name	Model number	Specifications
EJ1 EtherCAT Slave Unit	EJ1N-HFUC-ECT	• In the EJ1 Series, this Unit is classified as an Advanced Unit. • As an EtherCAT communications device, this Unit is classified as an EtherCAT slave. You can use PDO communications to perform CompoWay/F command/response communications between the EJ1 EtherCAT Slave Unit and Basic Units. • Power supply: 24 VDC provided from the End Unit.

1-3-2 Overview of Functions

The following table provides an overview of the functions provided by an EJ1 EtherCAT Slave Unit.

Function	Description	Reference
PDO communications	The following two types of communications commands can be sent and received by using PDO communications from a user program to perform handshaking. 1) CompoWay/F commands: Commands to read the present values or write set values in Basic Units. 2) Control commands: Commands used to control an EJ1 EtherCAT Slave Unit.	*Section 5 Using the EJ1 EtherCAT Slave Unit with PDO Communications*
SDO communications	You can use SDO communications to access objects that cannot be registered in PDO mappings. (You cannot use it for objects that can be registered in PDO mappings.)	*Section 6 Using the EJ1 EtherCAT Slave Unit with SDO Communications*

2

Specifications and Procedures

This section provides the general specifications and individual Unit specifications, and describes how to use the EJ1 EtherCAT Slave Unit.

2-1 Specifications 2-2
2-1-1 General Specifications 2-2
2-1-2 Function Specifications 2-2
2-1-3 EtherCAT Slave Communications Specifications 2-3
2-2 Application Procedures 2-4
2-2-1 Hardware Setup Procedure 2-4
2-2-2 Software Setup Procedure 2-5

2-1 Specifications

This section gives the general specifications of the EJ1 EtherCAT Slave Unit.

2-1-1 General Specifications

The following table gives the general specifications of the EJ1 EtherCAT Slave Unit.

Item	Specification
Power supply voltage	End Unit power supply: 24 VDC
Allowable voltage range	End Unit power supply: 20.4 to 26.4 VDC
Power consumption (at maximum load)	2 W max.
Ambient operating temperature	−10 to 55°C (with no condensation or icing)
Ambient operating humidity	25% to 85%
Ambient storage temperature	−25 to 65°C (with no condensation or icing)
Ambient storage humidity	25% to 85%
Vibration resistance	10 to 55 Hz, 10 m/s^2 for 2 hours each in X, Y, and Z directions
Shock resistance	150 m/s^2 max. 3 times each in 3 axes, 6 directions
Dielectric strength	600 VAC, 50 or 60 Hz for 1 min
Insulation resistance	20 MΩ min. (at 500 VDC)
Enclosure rating	IP20
Memory protection	EEPROM, 100,000 write operations
EJ1 internal bus communications conditions	Communications protocol: CompoWay/F, Baud rate: 115,200 bps Data length: 7 bits, Stop bit: 2 bits, Communications parity: Even
Weight	100 g max.

2-1-2 Function Specifications

The following table gives the function specifications of the EJ1 EtherCAT Slave Unit.

Item	Specification
Maximum number of Basic Units that can be connected	32 Units You can connect up to 16 Units for one End Unit. However, you can connect only one Advanced Unit.

2-1-3 EtherCAT Slave Communications Specifications

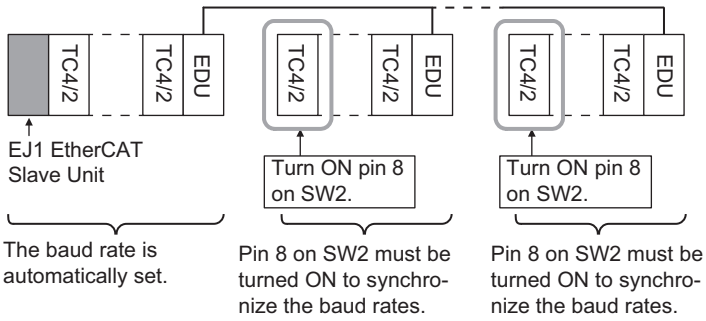
This section provides the EtherCAT slave communications specifications for the EJ1 EtherCAT Slave Unit.

Item		Specification
Communications protocol		EtherCAT protocol
Communications	PDO communications	Cyclic I/O is supported between the communications master and the following PDO-mappable objects in an EJ1 EtherCAT Slave Unit. • Input Notification Data: Port Status, Input SID, Input Data, etc. • Output Notification Data: Output SID, Output Data, etc.
	SDO communications	SDO communications is supported only when it is required to access objects that cannot be assigned to PDO mappings.
Modulation		Baseband
Baud rate		100 Mbps
Physical layer		100BASE-TX
Topology		Daisy chain, T-junction
Communications media		STP category 5 or higher
Communications distance		Distance between nodes: 100 m or less
Noise immunity		Conforms to IEC 61000-4-4, 1 kV or higher
PDO size		The following process data can be allocated as required.
	Input (received by communications master)	• Input Notification Data: 10 + (4 × n) bytes (n = 1, 2, 3, ... 20), where n is the number of input data items (default: 5) The input data size is between 4 and 80 bytes (default: 20 bytes). (The input data size can be changed by editing the PDO mappings.) • Sysmac Error Status: 1 byte.
	Output (sent by communications master)	• Output Notification Data: 8 + (4 × m) bytes (m = 1, 2, 3, ... 20), where m is the number of output data items (default: 5) The output data size is between 4 and 80 bytes (default: 20 bytes). (The output data size can be changed by editing the PDO mappings.)
Address range		0 to 255: Based on the two rotary switches on the front panel 0 to 65,535: Based on software settings*1, *2 *1. The addresses that you can set in the software settings depend on the EtherCAT master that you use. (For example, 1 to 192 for an NJ-series CPU Unit or 1 to 512 for an NX-series CPU Unit.) *2. Use the EtherCAT master's configuration tool to configure the software settings. (For example, use the Sysmac Studio if using an NJ-series CPU Unit or NX-series CPU Unit.)
Address setting method		Set on the hexadecimal rotary switches or in the software settings.
Indicators		L/A, RUN, and ERR
Cycle Time		FreeRun
Message protocol		CoE

2-2 Application Procedures

This section provides the procedures to use the EJ1 EtherCAT Slave Unit. The hardware and setup procedures are given separately.

2-2-1 Hardware Setup Procedure

Step	Item	Description	Reference
1	Connect the EJ1 EtherCAT Slave Unit and the Basic Units.	Place the EJ1 EtherCAT Slave Unit on the far left, connect up to 15 Basic Units on the right of it, and connect an End Unit on the far right.	*Section 1 Features and System Configuration* *Section 4 Installation and Connection*
2	Mount the connected Units to a DIN Rail.	Mount the connected Units to a DIN Rail. To ensure secure mounting, always attach an End Plate to each end.	*Section 4 Installation and Connection*
3	If you use distributed placement or connect to multiple Units from the CX-Thermo, connect an RS-485 communications cable.	• If you use a multiblock configuration for the Basic Units connected to the EJ1 EtherCAT Slave Unit, connect an RS-485 communications cable to port B on the End Unit of each block. • If you use the CX-Thermo configuration tool on a computer to set up the Basic Units connected by distributed placement, connect a communications cable to port A (i.e., the terminal block).	*EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142)
4	Connect the power supply.	Connect a 24-VDC power supply to the power supply terminals of the End Unit. Note Do not turn ON the power supply at this time. This power supply is used as the internal circuit power supply of the Basic Units and the EJ1 EtherCAT Slave Unit.	
5	Wire the Basic Units.	Wire the Basic Unit temperature inputs, control outputs, and other signals. Note Do not turn ON the power supply to any peripheral devices at this time.	
6	Set the communications unit numbers of the Basic Units.	Set the CompoWay/F communications unit number of each Basic Unit using the rotary switch and DIP switch on the Basic Unit. Set a unique CompoWay/F communications unit number for each Basic Unit.	
7	Set the Basic Unit baud rate.	• If you use distributed placement, turn ON pin 8 on SW2 on the Basic Unit connected at the far left for all blocks not connected to an EJ1 EtherCAT Slave Unit.  	
8	Set the node address.	Set the EtherCAT node address (0 to 255) of the EJ1 EtherCAT Slave Unit using the rotary switches. Set a unique node address for each slave connected to the same communications master.	*Section 3 Part Names and EtherCAT Communications Overview*

Step	Item	Description	Reference
9	Connect the EtherCAT communications connector.	Connect the EtherCAT communications connector to the EJ1 EtherCAT Slave Unit.	*Section 4 Installation and Connection*

2-2-2 Software Setup Procedure

Step	Item	Description	Reference
1	Create the EtherCAT network configuration.	Register the EJ1 EtherCAT Slave Unit when you create the EtherCAT network configuration on the Sysmac Studio or another EtherCAT communications master tool.	*Sysmac Studio Operation Manual* (Cat. No. W504)
2	Set the PDO mappings for I/O data.	Edit the I/O data PDO mappings used by the EJ1 EtherCAT Slave Unit on the Sysmac Studio or another EtherCAT communications master tool.	*Section 5 Using the EJ1 EtherCAT Slave Unit with PDO Communications*
3	Assign I/O ports to device variables and create the user program.	If you use an NJ/NX-series CPU Unit as the communications master, perform the following tasks on the Sysmac Studio: • Assign I/O port device variables for the EJ1 EtherCAT Slave Unit. • Create the user program to send CompoWay/F commands, receive CompoWay/F responses, and perform handshaking for that data.	*Section 5 Using the EJ1 EtherCAT Slave Unit with PDO Communications*
4	Turn ON the power supply to the EJ1 system.	Turn ON the power supply connected to the End Unit. The Basic Units will start. There are no restrictions on the order for turning ON the power supply.	*EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142)
5	Configure the Basic Unit settings.	Connect an E58-CIFQ1 USB-Serial Conversion Cable between a COM port on the computer on which the CX-Thermo is installed and port A on the End Unit. Configure the Basic Unit settings on the CX-Thermo.	
6	Turn ON the power supply to the EtherCAT communications master.	Turn ON the power supply to the EtherCAT communications master.	---
7	Check the PWR operation indicator on the EJ1 EtherCAT Slave Unit.	Confirm that the PWR operation indicator on the EJ1 EtherCAT Slave Unit is as follows: PWR: The power supply status is normal if the indicator is lit green.	*Section 3 Part Names and EtherCAT Communications Overview*
8	Transfer the network configuration information and the user program.	Transfer the network configuration information and the user program created on the Sysmac Studio or other EtherCAT communications master tool software to the controller.	*Sysmac Studio Operation Manual* (Cat. No. W504)
9	Execute the user program and check operation.	Execute the user program and confirm that communications between the communications master and the Basic Units are working.	*Section 5 Using the EJ1 EtherCAT Slave Unit with PDO Communications*

Precautions for Correct Use

Do not send any CompoWay/F commands from the EtherCAT master to the EJ1 EtherCAT Slave Unit while you are changing EJ1 EtherCAT Slave Unit settings or Basic Unit settings from the CX-Thermo.

3

Part Names and EtherCAT Communications Overview

This section gives the names of the parts of the EJ1 EtherCAT Slave Unit, describes the functions of the parts, and provides an overview of EtherCAT communications.

3-1 Part Names and Functions 3-2

3-1-1 Appearance and Names of Parts on the Front Panel 3-2

3-1-2 Meanings of Indicators 3-3

3-1-3 Node Address Setting Switches 3-3

3-1-4 EtherCAT Communications Connectors 3-4

3-2 EtherCAT Communications Overview 3-5

3-2-1 Structure of CAN Application Protocol over EtherCAT (CoE) 3-5

3-2-2 EtherCAT Slave Information File (ESI Files) 3-6

3-2-3 State Transitions for EtherCAT Communications 3-7

3-2-4 Process Data Objects (PDOs) 3-8

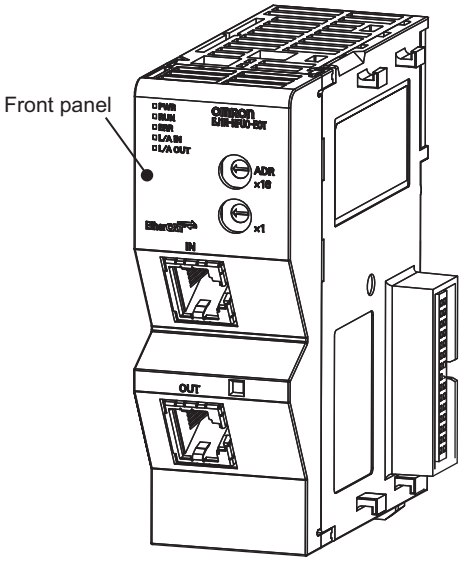
3-2-5 Service Data Objects (SDOs) 3-10

3-1 Part Names and Functions

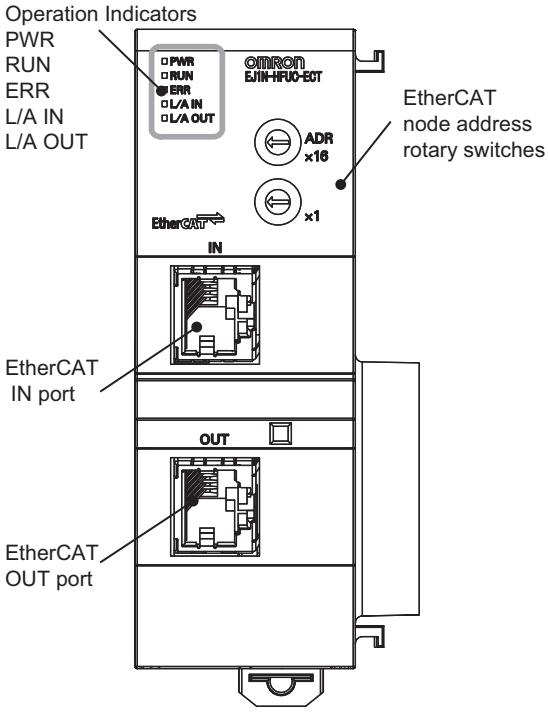
This section gives the names of the parts of the EJ1 EtherCAT Slave Unit and describes its external appearance.

3-1-1 Appearance and Names of Parts on the Front Panel

● Appearance



● Names of Parts on the Front Panel



3-1-2 Meanings of Indicators

The following table gives the meanings of the operation indicators on the front panel of the EJ1 EtherCAT Slave Unit.

● Operation Indicators

Indicator	Name	Color	Indicator	Description
PWR	Power supply status	Green	Lit.	Power is supplied to the Unit. *1
		---	Not lit.	No power is supplied to the Unit.
RUN	Operating status	Green	Lit.	Operational state
			Single flash	Safe-Operational state
			Flashing	Pre-Operational state
		---	Not lit.	Init state
ERR	Error status	Red	Lit.	• Error detected during boot process. • Error detected during Unit process.
			Double flash	A process data WDT error occurred.
			Flashing	An EtherCAT communications error other than a process data WDT error occurred.
		---	Not lit.	No error
L/A IN	EtherCAT IN port	Green	Lit.	A link was established in the physical layer.
			Flickering	A link was established and output communications are in operation.
		---	Not lit.	A link was not established in the physical layer.
L/A OUT	EtherCAT OUT port	Green	Lit.	A link was established in the physical layer.
			Flickering	A link was established and output communications are in operation.
		---	Not lit.	A link was not established in the physical layer.

*1. It will take a few moments until the indicators light after the power supply to the Unit is turned ON.

3-1-3 Node Address Setting Switches

The front panel rotary switches are used to set the node address of the EJ1 EtherCAT Slave Unit.

There is a difference in setting the node addresses between an OMRON EtherCAT master and one made by another manufacturer.

Setting on node address switches	Node address	
	OMRON NJ501-1□00 or CJ1W-NC□82 EtherCAT Master Unit	EtherCAT master from another manufacturer
00	The node address set with the configuration tool is used. (Default setting: 0, Setting range: 1 to 65,535)	The node address set with the configuration tool is used. (The address set on the switches is ignored.)
01 to 99	Address set on node address switches is used.	

Precautions for Correct Use

- The setting on the node address switches is read only once when the power supply is turned ON. Even if the setting is changed after the power supply is turned ON, the new setting will not be used until the next time the power supply is turned ON.
- When an NJ/NX-series CPU Unit is used, an error will occur and operation will stop if the same node address is set for more than one slave.
- When an NJ/NX-series CPU Unit is used and you set the node address switches to 00, the EJ1 EtherCAT Slave Unit will not operate if the node address setting from the configuration tool is not changed from 0.
  Be sure to use the configuration tool to change the node address to a value between 1 and 65,535.

Setting the EtherCAT Node Address

Use the two hexadecimal rotary switches to set the first digit ($\times 16^0$) and second digit ($\times 16^1$) of the EtherCAT node address for the EJ1 EtherCAT Slave Unit. You can set the address to any value between 00 and FF hex (0 to 255 decimal).

Note The default address setting is 00 hex.

Precautions for Correct Use

- Make sure that the power supply to the EJ1 EtherCAT Slave Unit is OFF before manipulating the switches. The setting is enabled when the power supply is turned ON.
- Set the switches with a small flat-blade screwdriver. Do not set the switches midway between two settings.

3-1-4 EtherCAT Communications Connectors

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

Connect EtherCAT network communications cables to the communications connectors.

There are two connectors: one for the IN (input) port and one for the OUT (output) port.

The specifications of the EtherCAT connectors are as follows:

- Electrical specifications: Conforming to IEEE 802.3 standards.
- Connector structure: RJ45 8-pin Modular Connector (conforming to ISO 8877).

For detailed wiring methods, refer to *4-2 EtherCAT Network Wiring* on page 4-3.

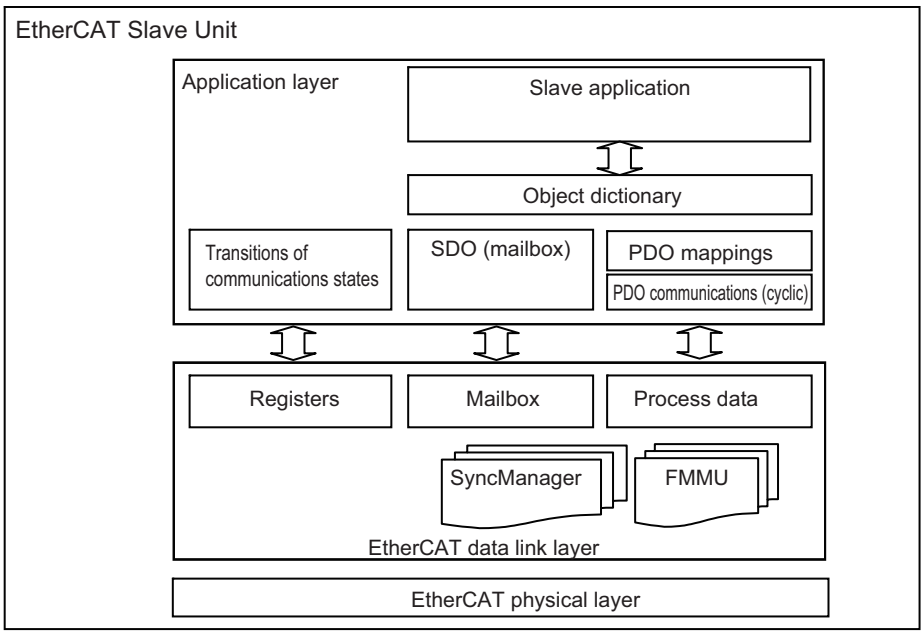
3-2 EtherCAT Communications Overview

This section provides an overview of EtherCAT communications.

3-2-1 Structure of CAN Application Protocol over EtherCAT (CoE)

EtherCAT allows the use of multiple protocols for communications. However, the EtherCAT Slave Unit uses the CAN application protocol over EtherCAT (CoE) as the device profile for the CAN application protocol. The CoE is a communications interface that is designed to provide compatibility with EtherCAT devices. The CAN application protocol is an open network standard.

The following figure shows how the CoE is structured for an EtherCAT Slave Unit.



The objects in the object dictionary for the CAN application protocol are broadly divided into PDOs (process data objects) and SDOs (service data objects).

PDOs are contained in the object dictionary. The PDOs can be mapped in the object dictionary. The process data is defined by the PDO mappings. PDOs are used in PDO communications for periodic exchange of process data.

SDOs are the objects that can be read and written. SDOs are used in non-periodic SDO communications (event-driven message communications).

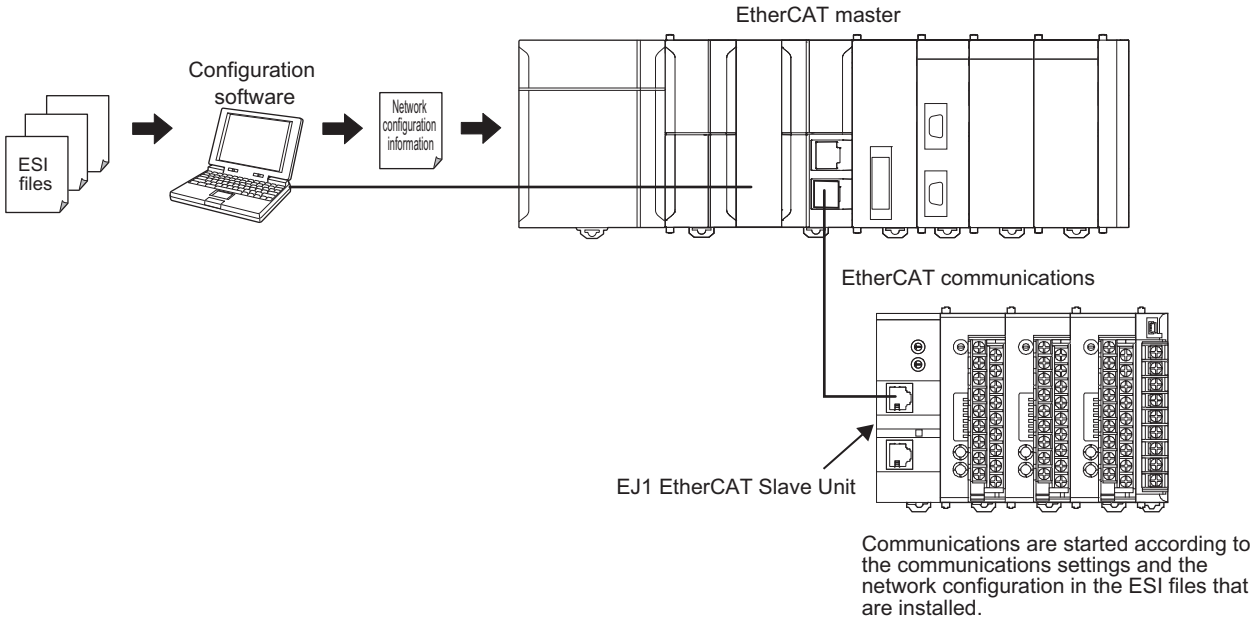
If you use the CoE interface to set the object dictionary for PDOs and SDOs, you can provide EtherCAT devices with the same device profiles as the CAN application protocol.

3-2-2 EtherCAT Slave Information File (ESI Files)

The setting information for an EtherCAT slave is provided in an ESI file (EtherCAT slave information). The EtherCAT communications settings are defined based on the ESI files of the connected slaves and the network connection information.

You can create the network configuration information by installing ESI files into the network configuration software.

You can download the network configuration information to the EtherCAT master to configure the EtherCAT network.

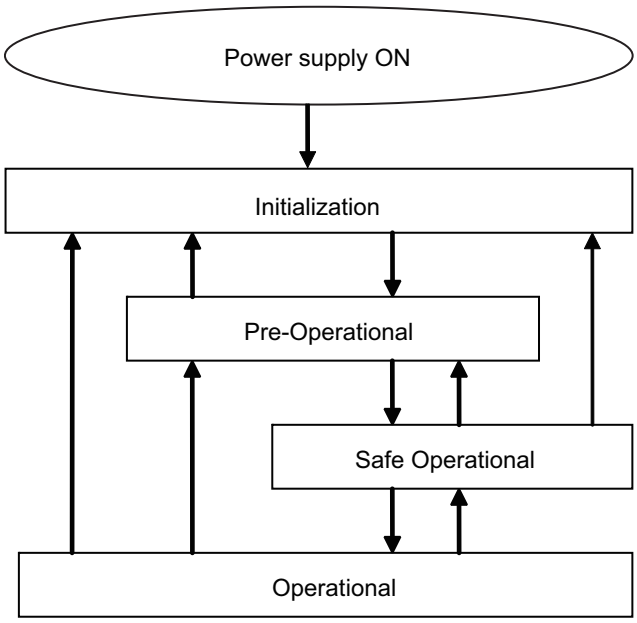


ESI files are required to connect OMRON EtherCAT slaves to controllers from other companies. To connect an EJ1 EtherCAT Slave Unit to a controller from another company, install the ESI file for the EJ1 EtherCAT Slave Unit in the configuration software that you are using.

3-2-3 State Transitions for EtherCAT Communications

The state machine that is used to control communications for the EJ1 EtherCAT Slave Unit is controlled by the EtherCAT master. The state machine is based on the ESI definition information and network connection information of the connected EJ1 EtherCAT Slave Units.

The following figure shows the communications state transitions from when the power supply is turned ON.



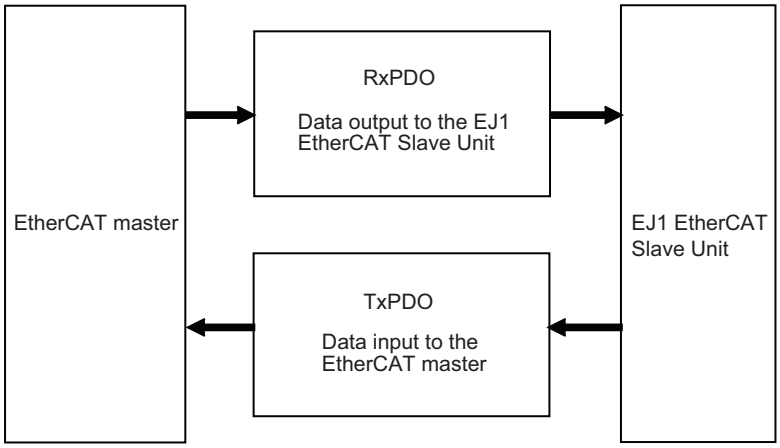
State	SDO communications	Sending PDOs	Receiving PDOs	Description
Initialization (Init)	Not possible.	Not possible.	Not possible.	Communications are being initialized. Communications are not possible.
Pre-Operational (Pre-Op)	Possible.	Not possible.	Not possible.	Only SDO communications (message communications) are possible in this state. This state is entered after initialization is completed. It is used to initialize network settings.
Safe-Operational (Safe-Op)	Possible.	Possible.	Not possible.	In this state, you can use both SDO communications (message communications) and send PDOs from the Slave Unit. You can send PDOs to send status and other information from the EJ1 EtherCAT Slave Unit.
Operational (Op)	Possible.	Possible.	Possible.	This is the normal state for communications. PDO communications are used to control the I/O data.

3-2-4 Process Data Objects (PDOs)

Overview

Process data objects (PDOs) are used to transfer data during cyclic communications in realtime.

There are two types of process data objects (PDOs): RxPDOs, which are used by the EJ1 EtherCAT Slave Unit to receive data from the EtherCAT master, and TxPDOs, which are used by the EJ1 EtherCAT Slave Unit to send data to the EtherCAT master.



The EtherCAT application layer can hold more than one object to enable the transfer of various process data of the EJ1 EtherCAT Slave Unit.

The contents of the process data is defined in the PDO mapping objects.

The EJ1 EtherCAT Slave Unit supports PDO mapping for I/O control.

PDO Mappings

PDO mapping objects contain the I/O data for the EJ1 EtherCAT Slave Unit. PDO mapping objects for the RxPDOs are managed in the object dictionary from indexes 1600 to 17FF hex, and for the TxPDOs from indexes 1A00 to 1BFF hex.

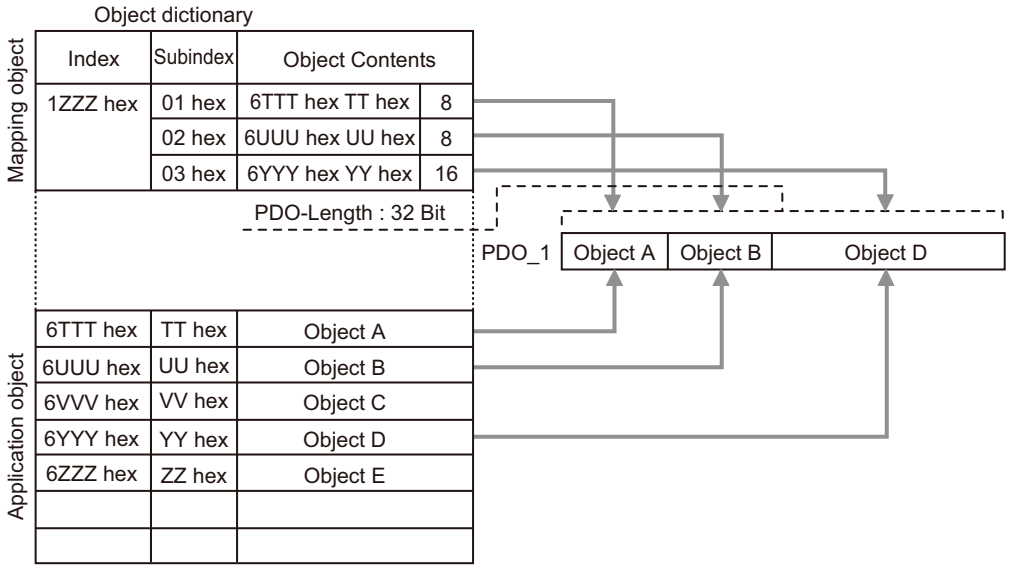
● PDO Mapping Scheme in EtherCAT

The PDO mapping scheme in EtherCAT is described below.

Three application objects (objects A, B, and D) are allocated to the PDO (name: PDO_1) at index 1ZZZ hex.

As described here, PDO mappings show how application objects are assigned to PDOs.

Indexes and subindexes are also assigned to application objects.



3-2-5 Service Data Objects (SDOs)

The EJ1 EtherCAT Slave Unit also supports SDO communications. Use SDO communications to monitor object settings and the status of the EJ1 EtherCAT Slave Unit. The communications master can read and write data in entries in the object dictionary to make parameter settings and monitor status.

4

Installation and Connection

This section describes how to install the EJ1 EtherCAT Slave Unit and wire communications.

4-1 Installing the EJ1 EtherCAT Slave Unit 4-2

4-2 EtherCAT Network Wiring 4-3

4-2-1 Installation Precautions 4-3

4-2-2 Preparations for Installation 4-3

4-2-3 Recommended EtherCAT Cables 4-4

4-3 Wiring the Temperature Controllers 4-5

4-1 Installing the EJ1 EtherCAT Slave Unit

This section describes how to connect the EJ1 EtherCAT Slave Unit to an EJ1 system.

Connect the Basic Units on the right side of the EJ1 EtherCAT Slave Unit.

Connect the End Unit to the right end.

1. Align the connectors and connect the Units to each other.
2. The yellow sliders at the top and bottom of each Unit lock the Units together. Move the sliders until they click into place.
3. Mount the EJ1 EtherCAT Slave Unit and Basic Units to a DIN Rail. Use screws to attach the DIN Rail to the control panel in at least three places.

Precautions for Correct Use

If you use distributed placement, connect the End Units of all of the blocks to each other with RS-485 communications cables. If you use distributed placement, connect ports B on all of the End Units to each other. If you connect to anything other than ports B on the End Units, the EJ1 EtherCAT Slave Unit may not operate correctly.

Refer to the *EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142) for information on how to mount Units to a DIN Rail.

4-2 EtherCAT Network Wiring

This section describes how to install an EtherCAT network.

4-2-1 Installation Precautions

Basic precautions for the installation of EtherCAT networks are provided below.

● Precautions when Installing a Network

- When you install an EtherCAT network, take sufficient safety precautions and perform the installation according to standards and specifications. (Refer to JIS X5252 or to electrical facility technical references.)

  An expert well versed in safety measures and the standards and specifications should be asked to perform the installation.
- Do not install EtherCAT network equipment near sources of noise.
- If the network must be installed in an area with noise, take steps to address the noise, such as placing equipment in metal cases.

● Precautions when Installing Communications Cables

- Check the following items on the communications cables that are used in the network.
  - Are there any breaks?
  - Are there any shorts?
  - Are there any connector problems?
- When you connect the cable to the communications connectors on devices, firmly insert the communications cable connector until it locks in place.
- Do not lay the communications cables together with high-voltage lines.
- Do not lay the communications cable near devices that generate noise.
- Do not lay the communications cables in locations subject to high temperatures or high humidity.
- Do not lay the communications cables in locations subject to excessive dirt and dust or to oil mist or other contaminants.
- There are limitations on the bending radius of communications cables. Check the specifications of the communications cable for the bending radius.

4-2-2 Preparations for Installation

Prepare the following devices.

Product	Remarks
Twisted-pair cable (Cables with the following connectors can also be used.)	100BASE-TX (Category 5 or higher) Double shielding with aluminum tape and braiding
RJ45 connectors	Shielded

4-2-3 Recommended EtherCAT Cables

Recommended products are given in the following tables.

Cables with Connectors

● Sizes and Conductor Pairs: AWG 27 × 4 pairs

Product name	Manufacturer	Cable length (m) *1	Model
Standard-type Cables with Connectors on Both Ends (RJ45/RJ45) 	OMRON Corporation	0.3	XS6W-6LSZH8SS30CM-Y
		0.5	XS6W-6LSZH8SS50CM-Y
		1	XS6W-6LSZH8SS100CM-Y
		10	XS6W-6LSZH8SS1000CM-Y

*1. Cables are available in the following lengths: 0.2, 0.3, 0.5, 1, 1.5, 2, 3, 5, 7.5, 10, 15, and 20 m. Refer to the *Industrial Ethernet Connectors Catalog* (Cat. No. G019)

Cables and Connectors

● Sizes and Conductor Pairs: AWG 24 × 4 pairs

Connecting Communications Cables and Connectors

Use straight wiring with a cable that has two or four twisted pairs.

- For cables with four twisted pairs, use T568A or T568B cables.
- For cables with two twisted pairs, refer to the datasheet for the connector.
- After wiring, use a LAN cable tester to check the operation of the network wiring.

4-3 Wiring the Temperature Controllers

Refer to the *EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142) for information on wiring the Temperature Controllers.

● Complying with Safety Standards

The power supply terminals must be supplied from a SELV, limited-current source. A SELV (separated extra-low voltage) source is a power supply having double or reinforced insulation between the primary and the secondary circuits and having an output voltage of 30 V r.m.s. max. and 42.4 V peak max. or 60 VDC max.

Recommended power supply: S8VM Series or S8VS Series (both manufactured by OMRON)

- Select a Power Supply with a capacity that is suitable for the application conditions.
- To comply with the EN 61326 Class A noise terminal voltage standard, insert a noise filter (TDK's RSMN2006 or the equivalent) on the DC line as close as possible to Temperature Controller.

5

Using the EJ1 EtherCAT Slave Unit with PDO Communications

This section describes how to send data to and receive data from the EJ1 EtherCAT Slave Unit by using PDO communications from the communications master.

5-1 EJ1 PDO Communications Overview 5-2

5-2 How Data Is Sent and Received with PDO Communications 5-3

5-2-1 CompoWay/F Commands and Responses 5-3

5-2-2 Control Commands and Responses 5-5

5-3 PDO Mapping Data Overview 5-6

5-3-1 Types of PDO Mapping Data 5-6

5-3-2 PDO Mapping Data List 5-7

5-4 PDO Mapping Data Details 5-8

5-4-1 Output Notification Data (Communications Master to EJ1 EtherCAT Slave Unit) 5-8

5-4-2 Input Notification Data (EJ1 EtherCAT Slave Unit to Communications Master) 5-10

5-4-3 Sysmac Error Status 5-14

5-5 Writing User Programming 5-15

5-5-1 Handshaking with the EJ1 EtherCAT Slave Unit with User Programming . . . 5-15

5-5-2 Handshaking with the EJ1 EtherCAT Slave Unit with User Programming . . . 5-16

5-5-3 Procedures for Sending Control Commands and Receiving Responses 5-17

5-6 Example Procedures for Sending CompoWay/F Commands and Receiving Responses 5-18

5-6-1 Example of Sending and Receiving Data without Dividing into Parts 5-18

5-6-2 Example of Sending and Receiving Data in Parts 5-21

5-7 Setting CompoWay/F Frames 5-25

5-7-1 Output Data and Input Data Contents 5-25

5-7-2 Storing Output Data and Input Data in Array Variables 5-27

5-7-3 Setting the PDO Mappings for I/O Data 5-28

5-8 Assigning Device Variables 5-32

5-1 EJ1 PDO Communications Overview

This section provides an overview of PDO communications when the EJ1 EtherCAT Slave Unit is used with EJ1 Basic Units as slave devices.

● Reading EJ1 Basic Unit Present Values and Writing Target Values

If you want to access an EJ1 Basic Unit connected to an EJ1 EtherCAT Slave Unit on the EtherCAT network from the communications master to read present values or write target values, you can use PDO communications to send a CompoWay/F command to the Basic Unit and receive a CompoWay/F response back from that Basic Unit. To do this, user programming is used to perform handshaking and to send and receive data.

● Restarting a Port on the EJ1 EtherCAT Slave Unit

To restart a port or clear the send and receive buffers in the EJ1 EtherCAT Slave Unit from the communications master, you can send control commands to control the EJ1 EtherCAT Slave Unit and then receive the response back for those commands. To do this, user programming is used to perform handshaking and to send and receive data.

5-2 How Data Is Sent and Received with PDO Communications

This section describes how data is sent and received between the communications master and the EJ1 EtherCAT Slave Unit with PDO communications.

The operation is described separately when CompoWay/F commands and responses are executed and when control commands and responses are executed.

5-2-1 CompoWay/F Commands and Responses

This section describes how data is sent and received when CompoWay/F commands and responses are used.

● How Data Is Sent

The following procedure is used to send data.

1 In each process data communications cycle, the Output Notification Data from the EtherCAT master is automatically sent to the send buffer in the EJ1 EtherCAT Slave Unit.

The size of the data sent at one time depends on the PDO map settings that are made from the Sysmac Studio. The PDO mapping sizes for Output Data and Input Data can be set from 4 to 80 bytes. The default setting is 20 bytes.)

2 The EJ1 EtherCAT Slave Unit is notified through user programming that the Output Notification Data was updated, and handshaking is performed when the EJ1 EtherCAT Slave Unit receives that notification.

3 The data in the send buffer in the EJ1 EtherCAT Slave Unit is used to automatically generate a CompoWay/F command and send it to the destination Basic Unit via the lateral EJ1 bus.

● How Data Is Received

1 The CompoWay/F commands are processed by the Basic Unit, a CompoWay/F response is sent back from the Basic Unit via the lateral EJ1 bus, and the response is stored in the EJ1 EtherCAT Slave Unit's receive buffer.

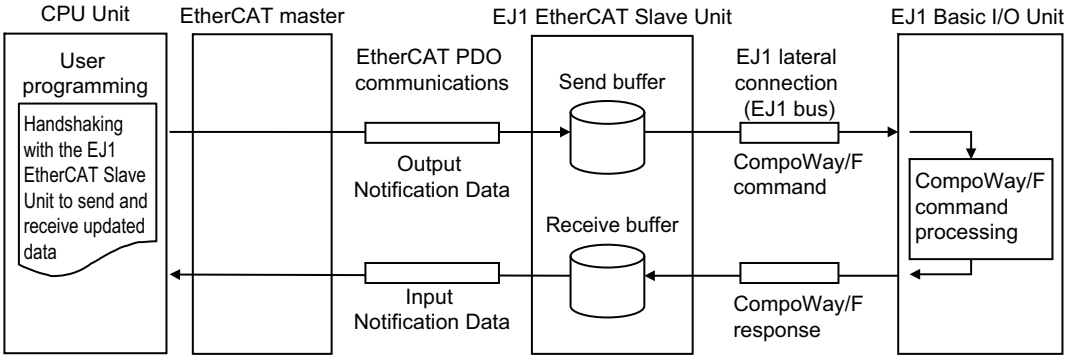
2 In each process data communications cycle, the Input Notification Data in the receive buffer in the EJ1 EtherCAT Slave Unit is automatically sent to the EtherCAT master.

The size of the data received at one time depends on the PDO map settings that are made from the Sysmac Studio. (The PDO mapping sizes for Output Data and Input Data can be set from 4 to 80 bytes. The default setting is 20 bytes.)

Precautions for Correct Use

If you use broadcast CompoWay/F commands, you do not need to perform reception processing for CompoWay/F responses.

3 The EJ1 EtherCAT Slave Unit sends a notification through user programming that the Input Notification Data was updated, and handshaking is performed when the notification is received.



5-2-2 Control Commands and Responses

This section describes how data is sent and received when control commands and responses are used.

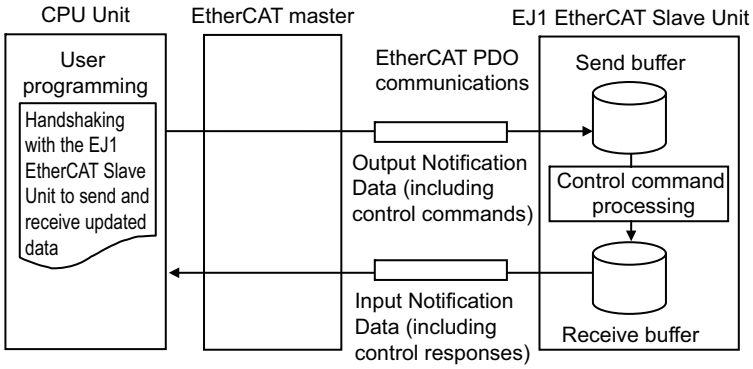
● How Data Is Sent

The following procedure is used to send data.

1. In each process data communications cycle, the Output Notification Data, which includes control commands, is automatically sent from the EtherCAT master to the send buffer in the EJ1 EtherCAT Slave Unit.
2. The EJ1 EtherCAT Slave Unit is notified through user programming that the Output Notification Data was updated, and handshaking is performed when the EJ1 EtherCAT Slave Unit receives that notification.

● How Data Is Received

1. The control commands are processed by the EJ1 EtherCAT Slave Unit.
2. In each process data communications cycle, the Input Notification Data, which includes control responses, is automatically sent to the EtherCAT master.
3. The EJ1 EtherCAT Slave Unit sends a notification through user programming that the Input Notification Data was updated, and handshaking is performed when the notification is received.



5-3 PDO Mapping Data Overview

This section provides an overview of mapping data for the communications master of the EJ1 EtherCAT Slave Unit for PDO communications.

The following section describes the types of PDO mapping data and provides a list of PDO entries.

5-3-1 Types of PDO Mapping Data

The following data is input and output cyclically in each process data communications cycle between the communications master and the EJ1 EtherCAT Slave Unit.

Data name	Description	Index	Size
Output Notification Data	This data is cyclically output from the communications master to the EJ1 EtherCAT Slave Unit.	7000 hex	The Output Data consists of 8 + (4 × m) bytes (where m = 1, 2, 3, ... 20).
Input Notification Data	This data is cyclically input from the EJ1 EtherCAT Slave Unit to the communications master.	6000 hex	The Input Data consists of 10 + (4 × n) bytes (where n = 1, 2, 3, ... 20).
Sysmac Error Status	The Sysmac Error Status data is cyclically input from the EJ1 EtherCAT Slave Unit to the communications master.	2002 hex (subindex 01 hex)	1 byte

5-3-2 PDO Mapping Data List

The following table lists the PDO mapping data by PDO entry.

Precautions for Correct Use

Set the assignments to meet the following conditions for PDO mapping. If the following conditions are not met, it will not be possible to change the EtherCAT communications state to the Safe-Operational (Safe–Op) state to enable PDO communications.

- Mapping must be in ascending order by index and subindex.
- Do not map a PDO at index 0000 hex, subindex 00 hex.
- Do not delete required PDO entries when mapping. Refer to *5-3-2 PDO Mapping Data List* on page 5-7 for details on the required PDO entries.

Data name	Index	Subindex	PDO entry *1	Set by default	PDO entry name, I/O port name *2	Object name	Size
Output Notification Data	7000 hex	01 hex	Required.	O	Output SID	Output SID	1 byte
		02 hex	Required.	O	Input SID Response	Input SID Response	1 byte
		03 hex	Required.	O	Output Data Type	Output Data Type	2 bytes
		04 hex	Required.	O	Output Sub Info	Output Sub Info	2 bytes
		05 hex	Required.	O	Output Data Length	Output Data Length	2 bytes
		06 hex	Required.	O	Output Data 01 to Output Data 20 (Defaults: Output Data 01 to Output Data 05)	Output Data 01 to Output Data 20 (Defaults:Output Data 01 to Output Data 05)	4 bytes × 1 to 4 bytes × 20 (Default: 4 × 5 bytes)
		07 to 0A hex	Any value	O			
		0B to 19 hex	Any value	---			
Input Notification Data	6000 hex	01 hex	Required.	O	Port Status	Port Status	2 bytes
		02 hex	Required.	O	Input SID	Input SID	1 byte
		03 hex	Required.	O	Output SID Response	Output SID Response	1 byte
		04 hex	Required.	O	Input Data Type	Input Data Type	2 bytes
		05 hex	Required.	O	Input Sub Info	Input Sub Info	2 bytes
		06 hex	Required.	O	Input Data Length	Input Data Length	2 bytes
		07 hex	Required.	O	Input Data 01 to Input Data 20 (Defaults: Input Data 01 to Input Data 05)	Input Data 01 to Input Data 20 (Defaults:Input Data 01 to Input Data 05)	4 bytes × 1 to 4 bytes × 20 (Default: 4 × 5 bytes)
		08 to 0B hex	Any value	O			
		0C to 1A hex	Any value	---			
Sysmac Error Status	2002 hex	01 hex	Any value	---	Sysmac Error Status	Sysmac Error Status	1 byte

*1. The following PDO entries are required. Do not delete these PDO entries when you set the PDO mappings.
Index 7000 hex, subindices 01, 02, 03, 04, 05, and 06 hex
Index 6000 hex, subindices 01, 02, 03, 04, 05, 06, and 07 hex

*2. The I/O port names are used to assign variables on the Sysmac Studio if you use an NJ/NX-series communications master.

5-4 PDO Mapping Data Details

This section provides details on the PDO mapping data. It is separated into Output Notification Data, Input Notification Data, and Sysmac Error Status.

Note The data in the PDO entry name, I/O port name column in the following tables match the PDO entry names and I/O port names (i.e., the names assigned to variables on the Sysmac Studio if you use an NJ/NX-series communications master) in the PDO mapping. However, the I/O port name column lists only the I/O port names.

5-4-1 Output Notification Data (Communications Master to EJ1 EtherCAT Slave Unit)

The following data is cyclically output from the communications master to the EJ1 EtherCAT Slave Unit.

PDO entry name, I/O port name	Object name	Type*1, *2	Size	R/W	Function	Value	Default
Output SID	Output SID *3	USINT	1 byte	R/W	This parameter tells the EJ1 EtherCAT Slave Unit whether or not the Output Notification Data was updated.	00 hex: Initial state 01 hex to FF hex: The Output SID is incremented each time data is sent by user programming. The value returns to 01 hex after FF hex.	00 hex
Input SID Response	Input SID Response *4	USINT	1 byte	R/W	This is a response parameter to the Input SID (update notification for Input Notification Data) in the Input Notification Data. • The EJ1 EtherCAT Slave Unit will not send notification of the next Input Data until the Input SID Response and Input SID match. The last value is retained.	00 hex: Initial state 01 hex to FF hex: Set SID with user programming.	00 hex
Output Data Type	Output Data Type	WORD	2 bytes	R/W	This entries specifies the control command relationship and the send data status.	Refer to the *Output Data Type* on page 5-9 for details.	0000 hex
Output Sub Info	Output Sub Info	WORD	2 bytes	R/W	Reserved area		0000 hex
Output Data Length	Output Data Length	UINT	2 bytes	R/W	This entry gives the byte size of data sent as Output Data.	0 to 80	0000 hex

PDO entry name, I/O port name	Object name	Type*1, *2	Size	R/W	Function	Value	Default
Output Data 01	Output Data 1	ARRAY[0..3] OF BYTE	4 bytes	R/W	This data is sent from the EJ1 EtherCAT Slave Unit to the Basic Unit. The text portion of the CompoWay/F command frame is written here. For details, refer to the *5-7-1 Output Data and Input Data Contents* on page 5-25.	Binary data of 0 to 80 decimal (00 hex to 50 hex) bytes. You can edit the PDO map settings on the Sysmac Studio to add PDO entries to increase or decrease the maximum output data size to 4 × m (where m = 1 to 20) bytes.	00000000 hex
Output Data m	Output Data m (m = 2 to 20) (Default: m = 5)	ARRAY[0..3] OF BYTE	Same as above.	Same as above.			00000000 hex

*1. WORD data is displayed as UINT data except on the Sysmac Studio.

On the Sysmac Studio, each bit of WORD data can be accessed as BOOL data. You cannot access these bits as BOOL data except from the Sysmac Studio.

*2. BYTE data is displayed as USINT data except on the Sysmac Studio.

On the Sysmac Studio, each bit of BYTE data can be accessed as BOOL data. You cannot access these bits as BOOL data except from the Sysmac Studio.

*3. How to Use Output SIDs in User Programming

Perform the following processing in user programming.

① Each time you update the Output Notification Data, increment the Output SID (previous value + 1).*

② If the Output SID Response in the next received Input Notification Data matches the Output SID incremented in step 1, it means that the EJ1 EtherCAT Slave Unit received the updated Output Notification Data.

*: If the EJ1 EtherCAT Slave Unit detects that the Output SID was incremented (previous value + 1), it receives the Output Notification Data by itself. If the Output SID value is not equal to the previous value plus 1, the data is considered invalid and is not received.

*4. How to Use Input SID Responses in User Programming

Perform the following processing in user programming.

- When Input Notification Data is received, set the Input SID Response to the value of the Input SID.

● Output Data Type

The following table shows how the Output Data type specifies the control command relationship and the send data status.

Bit position	I/O port name	Parameter name	Value		Default
00 to 07	None (Specifications are made in the lower byte of the Output Data Type WORD.)	Command Code of the Control Command*1, *2	Specifies the control command code. Refer to the Control Command Code List, below. Bit 08, below, is valid when the Control Command Flag is 1 (ON).		0
08	Send Command Flag	Control Command Flag*3	0: No control command sent. 1: Control command sent.		0
09 to 11	Reserved	Reserved area	Always 0 (OFF).		0
12	Send Data Flag	Intermediate Send Data Flag*3	1: There is intermediate send data.	If both bits are 0 (OFF): No send data. Note Both of these bits cannot be 1 (ON) at the same time.	0
13	Send Last Data Flag	Final Send Data Flag*3	1: There is final send data.		0
14 to 15	Reserved	Reserved area	Always 0 (OFF).		0

*1. Do not send a control command while output data is being sent for a CompoWay/F command or while input data is being received for a CompoWay/F response.

*2. If an out-of-range value is set for the control command, the Input Sub Info will return Undefined Command (0002 hex).

*3. If an out-of-range value is specified for the Control Command Flag, Intermediate Send Data Flag, or Final Send Data Flag when WORD data is specified, nothing is processed.

● Control Command Code List

Command code	Command	Description
01 hex	Restart Port	Clears the send and receive buffers and resets the Input SID to 0 to restore the port to its initial state. *1
06 hex	Clear Send and Receive Buffers	Clears the send and receive buffers.

*1. The Output SID is not reset if the port is restarted.

5-4-2 Input Notification Data (EJ1 EtherCAT Slave Unit to Communications Master)

The following data is cyclically input from the communications EJ1 EtherCAT Slave Unit to the communications master.

PDO entry, I/O port name	Object name	Type*1, *2	Size	R/W	Function	Value	Default
Port Status	Port Status	WORD or BOOL	2 bytes	RO	This status provides information on the communications state.	Refer to the *Port Status* on page 5-12 for details.	0000 hex
Input SID	Input SID *3	USINT	1 byte	RO	This parameter allows the EJ1 EtherCAT Slave Unit to tell the communications master whether or not the Input Notification Data was updated. • When the EJ1 EtherCAT Slave Unit receives a CompoWay/F command response or a control command response from a Basic Unit, this Input SID is automatically incremented (previous value + 1) by the EJ1 EtherCAT Slave Unit.	00 hex: Initial state (no received data) 01 hex to FF hex: The Input SID is automatically incremented when the EJ1 EtherCAT Slave Unit updates the data. The value returns to 01 hex after FF hex. • If the EJ1 EtherCAT Slave Unit performs a port restart, the input SID is reset to its initial state of 00 hex. If the Input SID Response is set to 00 hex, a Restart Port control command response is returned, so the input SID will be 01 hex.*4	00 hex

PDO entry, I/O port name	Object name	Type*1, *2	Size	R/W	Function	Value	Default
Output SID Response	Output SID Response *5	USINT	1 byte	RO	This is a response parameter to the Output SID (update notification for Output Notification Data) in the Output Notification Data. When Output Notification Data is received, the EJ1 EtherCAT Slave Unit automatically sets the Output SID Response to the value of the Output SID.	00 hex: Initial state 01 to FF hex: The Output SID received by the EJ1 EtherCAT Slave Unit is automatically stored in the Output SID Response.	00 hex
Input Data Type	Input Data Type	WORD	2 bytes	RO	The Input Data Type contains the control command-related response and received data status.	Refer to the *Input Data Type* on page 5-13 for details.	0000 hex
Input Sub Info	Input Sub Info	WORD	2 bytes	RO	When a control command is executed, the results of that command execution is stored here.	0000 hex: Control command ended normally. 0002 hex: Control command is undefined.	0000 hex
Input Data Length	Input Data Length	UINT	2 bytes	RO	This entry gives the byte size of valid data in the Input Data.	0 to 80 decimal (00 hex to 50 hex) bytes	00 hex
Input Data 01	Input Data 1	ARRAY[0..3] OF BYTE	4 bytes	RO	The data received by the EJ1 EtherCAT Slave Unit from the Basic Unit. The text portion of the CompoWay/F response frame is written to this data. For details, refer to the *5-7-1 Output Data and Input Data Contents* on page 5-25 later in this manual.	Binary data of 0 to 80 decimal (00 hex to 50 hex) bytes. * You can edit the PDO map settings on the Sysmac Studio to add PDO entries to increase or decrease the maximum input data size to 4 × n (n = 1 to 20) bytes.	00000000 hex
. . .	. . .	. . .					
Input Data n	Input Data n (where n = 2 to 20) (Default: n = 5)	ARRAY[0..3] OF BYTE	Same as above.	RO			00000000 hex

*1. WORD data is displayed as UINT data except on the Sysmac Studio.

On the Sysmac Studio, each bit of WORD data can be accessed as BOOL data. You cannot use these bits as BOOL data except on the Sysmac Studio.

*2. BYTE data is displayed as USINT data except on the Sysmac Studio.

On the Sysmac Studio, each bit of BYTE data can be accessed as BOOL data. You cannot access these bits as BOOL data except from the Sysmac Studio.

*3. How to Use Input SIDs in User Programming

Perform the following processing in user programming.

① When the value of this Input SID is equal to the previous value +1, the Input Notification Data is received. If the Input SID value is not changed, the Input Notification Data is not received.

② When the Input Notification Data is received, the Input SID Response in the Output Notification Data is set to the value of the Input SID incremented (previous value + 1) in step 1.

*4. If the Input SID Response is 00 hex and Restart Port is executed, a Restart Port control command response is returned immediately, so the input SID will be 01 hex.

*5. How to Use Output SID Responses in User Programming:

Perform the following processing in user programming.

- If the received Output SID Response matches the previously incremented Output SID, it means that the EJ1 EtherCAT Slave Unit received the sent data.

● Port Status

The port status provides information on the communications state, as given in the following table.

Bit position	I/O port name	Parameter name	Function	Value	Default
00	Send Data Exist	Send Data Exists Flag	Indicates whether CompoWay/F command data exists in the EJ1 EtherCAT Slave Unit send buffer. After all of the CompoWay/F command is sent to the Basic Unit, the value changes to 0 (OFF).	0: There is no data in the send buffer. 1: There is data in the send buffer.	0
01	Send Completed Toggle Bit	Send Completed Toggle Bit	This bit toggles each time a CompoWay/F command is sent from the EJ1 EtherCAT Slave Unit to the Basic Unit.	The value is toggled between 0 and 1 after each send operation.	0
02	Send Buffer Full Flag	Send Buffer Full Flag	This bit changes to 1 (ON) when the send data exceeds the maximum size of the buffer. The maximum send data buffer size is 2,048 bytes. To reset this bit to 0 (OFF), restart the port or use the Clear Send and Receive Buffers control command.	0: No error occurred. 1: Error occurred.	0
03 to 10	Reserved	Reserved area	Always 0.		0
11	Receive Data Exist	Final Data Reception Flag	This bit changes to 1 (ON) when the entire CompoWay/F response is received (up to the BCC) by the receive buffer in the EJ1 EtherCAT Slave Unit from a Basic Unit. The bit changes back to 0 (OFF) when the communications master reads the received CompoWay/F response data from the receive buffer of the EJ1 EtherCAT Slave Unit.	0: There is no data in the receive buffer. 1: There is data in the receive buffer.	0
12 to 13	Reserved	Reserved area	Always 0.		0

Bit position	I/O port name	Parameter name	Function	Value	Default
14	EJ1Bus Communication Error	EJ1 Bus Communications Error Flag	This bit changes to 1 (ON) when a communications error occurs on the EJ1 bus (the internal bus between the EJ1 EtherCAT Slave Unit and Basic Unit) due to noise or other factors. *1 This flag resets to 0 (OFF) when a correct response is received.	0: Normal 1: EJ1 Bus Communications Error occurred.	0
15	BCC Error	BCC Error Flag	This bit changes to 1 (ON) when the BCC check of the received response indicates that communications failed. The received data is discarded. This flag resets to 0 (OFF) when a correct response is received.	0: Normal 1: BCC error occurred.	0

*1. If an EJ1 bus communications error occurs, the EJ1 EtherCAT Slave Unit may still be receiving a CompoWay/F response. Allow at least 32 ms after detecting an EJ1 bus communications error before executing the next output data notification.

● Input Data Type

The Input Data Type contains the control command-related response and received data status.

Bit position	I/O port name	Parameter name	Value		Default
00 to 07	None (Specifications are made in the lower byte of the Input Data Type WORD.)	Control Command Response	The control command code that was sent in the control command of the Output Data Type is returned as is. Bit 09, below, is valid when the Control Command Response Flag is 1 (ON).		00 hex
08	Reserved	Reserved area	Always 0.		0
09	Receive Command Flag	Control Command Response Flag	0: There is no control command response. 1: There is a control command response.		0
10 to 11	Reserved	Reserved area	Always 0.		0
12	Receive Data Flag	Intermediate Data Reception Flag *1	1: There is Intermediate Receive Data from the Basic Unit.	When both bits are 0 (OFF): There is no received data from the Basic Unit. Note Both of these bits cannot be 1 (ON) at the same time.	0
13	Receive Last Data Flag	Final Data Reception Flag *1	1: There is Final Receive Data from the Basic Unit.		0
14	Reserved	Reserved area	Always 0.		0

Bit position	I/O port name	Parameter name	Value	Default
15	Receive Data Error Flag	Receive Data Error Flag *2	0: No error 1: There is an error. This flag resets to 0 (no errors) when a correct response is received.	0

*1. The Final Data Reception Flag changes to 1 (ON) when the response to the received CompoWay/F command is the final data.

The Final Data Reception Flag will always be 1 (ON) for responses that do not require division into smaller parts.

If a response must be divided into smaller parts, the Intermediate Data Reception Flag changes to 1 (ON) for intermediate data, and the Final Data Reception Flag changes to 1 (ON) for the final data.

*2. If an error occurs during the reception of data from the Basic Unit, the Receive Data Error Flag changes to 1 (ON).

One of the following errors is the cause.

- The received data fails the BCC check.
- The received data is corrupted due to noise or some other cause.

5-4-3 Sysmac Error Status

PDO entry name, I/O port name	Object name	Type	Size	R/W	Function	Value	Default
Sysmac Error Status	Sysmac Error Status	BYTE	1 byte	RO	This is the Sysmac error status for the EJ1 EtherCAT Slave Unit. 1 (ON): An error at the relevant event level occurred. 0 (OFF): There are no errors at the relevant event level. Refer to *7-3 Error Notifications Based on the Sysmac Error Status* on page 7-6 for details.	Bit 05: Minor fault Bit 04: Observation	00 hex

5-5 Writing User Programming

This section describes how to write user programming to perform data transfer between the communications master and the EJ1 EtherCAT Slave Unit.

It is necessary to use handshaking to confirm that the Input Notification Data and Output Notification Data that are updated between the communications master and the EJ1 EtherCAT Slave Unit are received normally by the other party.

Additional Information

"SID" in "output SID" and "input SID" stands for sequential ID. The SID is an identifier used to determine the sequential number of data in a sequential data set.

5-5-1 Handshaking with the EJ1 EtherCAT Slave Unit with User Programming

Create user programming based on the following procedure.

1. Increment the Output SID to notify the EJ1 EtherCAT Slave Unit that the Output Notification Data was updated.
2. Check whether the Output SID Response matches the above Output SID to confirm that the EJ1 EtherCAT Slave Unit received the Output Notification Data successfully.
3. See if the Input SID was incremented to confirm when the Input Notification Data is updated.
4. Set the Input SID Response to the above Input SID to notify the EJ1 EtherCAT Slave Unit that the Input Notification Data was received.

5-5-2 Handshaking with the EJ1 EtherCAT Slave Unit with User Programming

Create user programming based on the following procedure when you send a CompoWay/F command, receive a CompoWay/F response, and perform handshaking as discussed previously.

1 Set the Output Data Type to 1000 hex (intermediate send data) or 2000 hex (final send data). At the same time, set the CompoWay/F command in the Output Data.

2 Increment the Output SID from the previous value (+1). This allows the communications master to notify the EJ1 EtherCAT Slave Unit that the Output Notification Data was updated.

3 If the received Output SID Response matches the value of the Output SID sent in step 2 above, it means that the EJ1 EtherCAT Slave Unit received the sent data.

4 If the Send Completed Toggle Bit in the Port Status was changed, it means that the CompoWay/F command was successfully sent to the Basic Unit.

5 Check the value of the Input SID. If the Input SID was incremented (+1) from its previous value (it has an initial value of 00 hex), check the Input Data Type.

(If the Receive Data Error Flag is ON, a BCC error or EJ1 bus communications error occurred. Check the Port Status for error details.)

If the Intermediate Data Reception Flag or Final Data Reception Flag is ON, it means that the Input Notification Data was updated and the Input Data is read.

If the Input Data Type is 2000 hex, the data received from the Basic Unit is the final data.

If the Input Data Type is 1000 hex, the data received from the Basic Unit is not the final data and is considered to be intermediate data (i.e., there is still more data to receive).

Repeat steps 5 and 6 for the next data reception.

6 Set the Input SID Response to the Input SID from step 5 above to notify the EJ1 EtherCAT Slave Unit that the Input Notification Data was received.

7 If the Receive Data Exists Flag in the Port Status is 0 (OFF), the data was received.

5-5-3 Procedures for Sending Control Commands and Receiving Responses

Create user programming based on the following procedure when you send a control command, receive a control command response, and perform handshaking as discussed previously.

1 Set the Output Data Type to 0101 hex (Port Restart) or 0106 hex (Clear Send and Receive Buffers).

2 Increment the Output SID from the previous value (+1). This allows the communications master to notify the EJ1 EtherCAT Slave Unit that the Output Notification Data was updated.

3 If the received Output SID Response matches the value of the Output SID sent in step 2 above, it means that the EJ1 EtherCAT Slave Unit received the sent data.

4 Check the value of the Input SID.

If the value is 0101 hex (Restart Port):

If the Input SID is not 0 before restarting the port, the Input SID changes to 0 after the port is restarted.

If the Input SID is 0, set the Input SID Response to 0.

When the Input SID is updated to 1, check the Input Data Type. If the Control Command Response Flag is 1 (ON), there is a control command response.

If the value is 0106 hex (Clear Send and Receive Buffers):

If the Input SID was incremented (+1) from its previous value (it has an initial value of 00 hex), check the Input Data Type. If the Control Command Response Flag is 1 (ON), there is a control command response.

5 Set the Input SID Response to the Input SID from step 4 above to notify the EJ1 EtherCAT Slave Unit that the Input Notification Data was received.

5-6 Example Procedures for Sending CompoWay/F Commands and Receiving Responses

This section provides example procedures for sending CompoWay/F commands and receiving responses.

Whether or not the data must be sent or received in parts depends on whether the Output Data or Input Data exceeds the assigned PDO mapping size.

The following examples demonstrate sending and receiving data both not dividing the data and dividing the data into parts.

5-6-1 Example of Sending and Receiving Data without Dividing into Parts

If the Output Data and Input Data do not exceed the assigned PDO mapping sizes, the Output Data and Input Data are sent and received without division into parts.

<table>
<tr><th>Step</th><th>Process</th><th>Communications master processing (processing in user programming)</th><th>Communications direction</th><th>EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)</th></tr>
<tr><td>1</td><td>No data reception</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The Input SID is 00 hex, so the master knows that the received Input Notification Data was not updated.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. The EJ1 EtherCAT Slave Unit sets the Input SID to 00 hex to notify the communications master that the Input Notification Data was not updated.
<table>
<tr><td>Port Status</td><td>0000 hex</td></tr>
<tr><td>Input SID</td><td>00 hex</td></tr>
<tr><td>Output SID Response</td><td>00 hex</td></tr>
<tr><td>Input Data Type</td><td>0000 hex</td></tr>
<tr><td>Input Sub Info</td><td>0000 hex</td></tr>
<tr><td>Input Data Length</td><td>0000 hex</td></tr>
<tr><td>Input Data</td><td>None</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Step</th><th>Process</th><th>Communications master processing (processing in user programming)</th><th>Communications direction</th><th>EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)</th></tr>
<tr><td>2</td><td>Input response for no Input Data + output of undivided send data</td><td>1. The communications master outputs the following Output Notification Data.
2. The communications master has not yet received input data, so the Input SID Response remains at 00 hex.
3. The communications master sets the Output Data Type to 2000 hex to indicate that the Output Data is the final data (because it is not divided).
4. The communications master increments the Output SID by 1, then passes the Output Data Length and Send Data to the EJ1 EtherCAT Slave Unit.
<table>
<tr><td>Output SID</td><td>01 hex</td></tr>
<tr><td>Input SID Response</td><td>00 hex</td></tr>
<tr><td>Output Data Type</td><td>2000 hex</td></tr>
<tr><td>Reserved.</td><td>0000 hex</td></tr>
<tr><td>Output Data Length</td><td>nnnn hex</td></tr>
<tr><td>Output Data</td><td>Send data</td></tr>
</table></td><td>→</td><td>The EJ1 EtherCAT Slave Unit receives the Output Notification Data given on the left.</td></tr>
<tr><td>3</td><td>Output response for the send data</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The communications master knows that sent data was received by the EJ1 EtherCAT Slave Unit because the Output SID Response in the Input Notification Data matches the value of the Output SID output in step 2.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. The EJ1 EtherCAT Slave Unit changes the Send Data Exists Flag (bit 00) in the Port Status to 1 (ON) to indicate that there is data in the send buffer.
3. When the value of the Output SID Response is 01 hex, i.e., the value of the Output SID from step 2, it tells the communications master that the Output Notification Data was received normally.
<table>
<tr><td>Port Status</td><td>0001 hex</td></tr>
<tr><td>Input SID</td><td>00 hex</td></tr>
<tr><td>Output SID Response</td><td>01 hex</td></tr>
<tr><td>Input Data Type</td><td>0000 hex</td></tr>
<tr><td>Input Sub Info</td><td>0000 hex</td></tr>
<tr><td>Input Data Length</td><td>0000 hex</td></tr>
<tr><td>Input Data</td><td>None</td></tr>
</table></td></tr>
<tr><td>4</td><td>Sending the send data</td><td>———</td><td></td><td>The EJ1 EtherCAT Slave Unit sends the data to the Basic Unit.</td></tr>
</table>

<table>
<tr><th>Step</th><th>Process</th><th>Communications master processing (processing in user programming)</th><th>Communications direction</th><th>EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)</th></tr>
<tr><td>5</td><td>Confirmation of sending the send data</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The Send Completed Toggle Bit in the Port Status in the Input Notification Data changed, so the communications master knows that the EJ1 EtherCAT Slave Unit sent the send data to the Basic Unit.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data to tell the communications master that the final send data was sent to the Basic Unit.
2. The EJ1 EtherCAT Slave Unit changes the Send Completed Toggle Bit in the Port Status.
<table><tr><td>Port Status</td><td>0002 hex</td></tr><tr><td>Input SID</td><td>00 hex</td></tr><tr><td>Output SID Response</td><td>01 hex</td></tr><tr><td>Input Data Type</td><td>0000 hex</td></tr><tr><td>Input Sub Info</td><td>0000 hex</td></tr><tr><td>Input Data Length</td><td>0000 hex</td></tr><tr><td>Input Data</td><td>None</td></tr></table></td></tr>
<tr><td>6</td><td>Receiving data from Basic Unit</td><td>---</td><td></td><td>The EJ1 EtherCAT Slave Unit receives the receive data from the Basic Unit.</td></tr>
<tr><td>7</td><td>Data input</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The Input SID changed from 00 to 01 hex, so the communications master knows that the received Input Data in the Input Notification Data was updated.
3. The Input Data Type is 2000 hex, so the communications master knows that the Input Data is the final data.
4. The communications master reads the Input Data.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. The EJ1 EtherCAT Slave Unit changes the Input SID from 00 hex to 01 hex to tell the communications master that the Input Data in the Input Notification Data was updated.
<table><tr><td>Port Status</td><td>0002 hex</td></tr><tr><td>Input SID</td><td>01 hex</td></tr><tr><td>Output SID Response</td><td>01 hex</td></tr><tr><td>Input Data Type</td><td>2000 hex</td></tr><tr><td>Input Sub Info</td><td>0000 hex</td></tr><tr><td>Input Data Length</td><td>nnnn hex</td></tr><tr><td>Input Data</td><td>Received data</td></tr></table></td></tr>
<tr><td>8</td><td>Input response</td><td>1. The communications master outputs the following Output Notification Data.
2. The communications master sets the Input SID Response to the value of the Input SID received in step 7 (01 hex) to tell the EJ1 EtherCAT Slave Unit that the communications master normally received the Input Data.
<table><tr><td>Output SID</td><td>01 hex</td></tr><tr><td>Input SID Response</td><td>01 hex</td></tr><tr><td>Output Data Type</td><td>0000 hex</td></tr><tr><td>Reserved.</td><td>0000 hex</td></tr><tr><td>Output Data Length</td><td>0000 hex</td></tr><tr><td>Output Data</td><td>None</td></tr></table></td><td>→</td><td>1. The EJ1 EtherCAT Slave Unit receives the Output Notification Data given on the left.
2. The EJ1 EtherCAT Slave Unit knows that the communications master normally received the Input Notification Data because the value of the Input SID Response in the Output Notification Data from step 7 is 01 hex.</td></tr>
</table>

5-6-2 Example of Sending and Receiving Data in Parts

If the Output Data or Input Data exceeds the assigned PDO mapping size, the Output Data or Input Data is divided to be sent or received.

<table>
<tr><th>Step</th><th>Process</th><th>Communications master processing (processing in user programming)</th><th>Communications direction</th><th>EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)</th></tr>
<tr><td>1</td><td>No data reception</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The Input SID is 00 hex, so the master knows that the received Input Notification Data was not updated.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. The EJ1 EtherCAT Slave Unit sets the Input SID to 00 hex to notify the communications master that the Input Notification Data was not updated.
<table>
<tr><td>Port Status</td><td>0000 hex</td></tr>
<tr><td>Input SID</td><td>00 hex</td></tr>
<tr><td>Output SID Response</td><td>00 hex</td></tr>
<tr><td>Input Data Type</td><td>0000 hex</td></tr>
<tr><td>Input Sub Info</td><td>0000 hex</td></tr>
<tr><td>Input Data Length</td><td>0000 hex</td></tr>
<tr><td>Input Data</td><td>None</td></tr>
</table></td></tr>
<tr><td>2</td><td>Input response for no Input Data + output of non-final send data</td><td>1. The communications master outputs the following Output Notification Data.
2. The communications master has not yet received input data, so the Input SID Response remains at 00 hex.
3. The communications master sets the Output Data Type to 1000 hex to indicate that the Output Data is not the final data.
4. The communications master increments the Output SID by 1, then passes the Output Data Length and send data (not the final data) to the EJ1 EtherCAT Slave Unit.
<table>
<tr><td>Output SID</td><td>01 hex</td></tr>
<tr><td>Input SID Response</td><td>00 hex</td></tr>
<tr><td>Output Data Type</td><td>1000 hex</td></tr>
<tr><td>Reserved.</td><td>0000 hex</td></tr>
<tr><td>Output Data Length</td><td>nnnn hex</td></tr>
<tr><td>Output Data</td><td>Send data that is not the final send data</td></tr>
</table></td><td>→</td><td>The EJ1 EtherCAT Slave Unit receives the Output Notification Data given on the left.</td></tr>
</table>

<table>
<tr><th>Step</th><th>Process</th><th>Communications master processing (processing in user programming)</th><th>Communications direction</th><th>EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)</th></tr>
<tr><td>3</td><td>Output response for send data that is not the final send data</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The communications master knows that sent data was received by the EJ1 EtherCAT Slave Unit because the Output SID Response in the Input Notification Data matches the value of the Output SID output in step 2.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. The EJ1 EtherCAT Slave Unit changes the Send Data Exists Flag (bit 0) in the Port Status to 1 (ON) to indicate that there is data in the send buffer.
3. When the value of the Output SID Response is 01 hex, it tells the communications master that the Output Notification Data was received normally.
<table>
<tr><td>Port Status</td><td>0001 hex</td></tr>
<tr><td>Input SID</td><td>00 hex</td></tr>
<tr><td>Output SID Response</td><td>01 hex</td></tr>
<tr><td>Input Data Type</td><td>0000 hex</td></tr>
<tr><td>Input Sub Info</td><td>0000 hex</td></tr>
<tr><td>Input Data Length</td><td>0000 hex</td></tr>
<tr><td>Input Data</td><td>None</td></tr>
</table></td></tr>
<tr><td>4</td><td>Outputting the final send data</td><td>1. The communications master outputs the following Output Notification Data.
2. The communications master sets the Output Data Type to 2000 hex to indicate that the Output Data is the final data.
3. The communications master increments the Output SID by 1, then passes the Output Data Length and final send data to the EJ1 EtherCAT Slave Unit.
<table>
<tr><td>Output SID</td><td>02 hex</td></tr>
<tr><td>Input SID Response</td><td>00 hex</td></tr>
<tr><td>Output Data Type</td><td>2000 hex</td></tr>
<tr><td>Reserved.</td><td>0000 hex</td></tr>
<tr><td>Output Data Length</td><td>nnnn hex</td></tr>
<tr><td>Output Data</td><td>This is the final send data.</td></tr>
</table></td><td>→</td><td>The EJ1 EtherCAT Slave Unit receives the Output Notification Data given on the left.</td></tr>
<tr><td>5</td><td>Output response for the final send data</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The communications master knows that sent data was received by the EJ1 EtherCAT Slave Unit because the Output SID Response in the Input Notification Data matches the value of the Output SID output in step 4.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. When the value of the Output SID Response is 02 hex, it tells the communications master that the final send data was received normally.
<table>
<tr><td>Port Status</td><td>0001 hex</td></tr>
<tr><td>Input SID</td><td>00 hex</td></tr>
<tr><td>Output SID Response</td><td>02 hex</td></tr>
<tr><td>Input Data Type</td><td>0000 hex</td></tr>
<tr><td>Input Sub Info</td><td>0000 hex</td></tr>
<tr><td>Input Data Length</td><td>0000 hex</td></tr>
<tr><td>Input Data</td><td>None</td></tr>
</table></td></tr>
<tr><td>6</td><td>Sending the send data</td><td>---</td><td></td><td>The EJ1 EtherCAT Slave Unit sends the data to the Basic Unit.</td></tr>
</table>

Step	Process	Communications master processing (processing in user programming)	Communications direction	EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)
7	Checking sending of the final send data	1. The communications master receives the Input Notification Data given on the right. 2. The Send Completed Toggle Bit in the Port Status in the Input Notification Data changed, so the communications master knows that the EJ1 EtherCAT Slave Unit sent the final send data to the Basic Unit.	←	1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data to tell the communications master that the final send data was sent to the Basic Unit. 2. The EJ1 EtherCAT Slave Unit sets the Send Data Exists Flag (bit 00) in the Port Status to 0 (OFF) to indicate that the CompoWay/F command was sent to the Basic Unit. 3. The EJ1 EtherCAT Slave Unit changes the Send Completed Toggle Bit in the Port Status. Port Status: 0002 hex Input SID: 00 hex Output SID Response: 02 hex Input Data Type: 0000 hex Input Sub Info: 0000 hex Input Data Length: 0000 hex Input Data: None
8	Receiving data from Basic Unit	—		The EJ1 EtherCAT Slave Unit receives the receive data from the Basic Unit.
9	Inputting the first data	1. The communications master receives the Input Notification Data given on the right. 2. The Input SID changed from 00 to 01 hex, so the communications master knows that the received Input Data in the Input Notification Data was updated. 3. The Input Data Type is 1000 hex, so the communications master knows that the Input Data is not the final data. 4. The communications master reads the Input Data.	←	1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data. 2. The EJ1 EtherCAT Slave Unit sets the Receive Data Exists Flag (bit 11) in the Port Status to 1 (ON) to indicate that the CompoWay/F response was received from the Basic Unit. 3. The EJ1 EtherCAT Slave Unit changes the Input SID from 00 hex to 01 hex to tell the communications master that the Input Data in the Input Notification Data was updated. Port Status: 0802 hex Input SID: 01 hex Output SID Response: 02 hex Input Data Type: 1000 hex Input Sub Info: 0000 hex Input Data Length: nnnn hex Input Data: First received data

<table>
<tr><th>Step</th><th>Process</th><th>Communications master processing (processing in user programming)</th><th>Communications direction</th><th>EJ1 EtherCAT Slave Unit processing (EJ1 EtherCAT Slave Unit firmware processing)</th></tr>
<tr><td>10</td><td>First input response</td><td>1. The communications master outputs the following Output Notification Data.
2. The communications master sets the Input SID Response to the value of the Input SID received in step 9 (01 hex) to tell the EJ1 EtherCAT Slave Unit that the communications master normally received the first Input Data.
<table><tr><td>Output SID</td><td>02 hex</td></tr><tr><td>Input SID Response</td><td>01 hex</td></tr><tr><td>Output Data Type</td><td>0000 hex</td></tr><tr><td>Reserved.</td><td>0000 hex</td></tr><tr><td>Output Data Length</td><td>0000 hex</td></tr><tr><td>Output Data</td><td>None</td></tr></table></td><td>→</td><td>1. The EJ1 EtherCAT Slave Unit receives the Output Notification Data given on the left.
2. The EJ1 EtherCAT Slave Unit knows that the communications master normally received the first Input Data because the Input SID Response in the Output Notification Data is 01 hex from step 9.</td></tr>
<tr><td>11</td><td>Inputting the second data</td><td>1. The communications master receives the Input Notification Data given on the right.
2. The communications master checks the Input SID.
3. The Input SID changed from 01 to 02 hex, so the communications master knows that the received Input Data in the Input Notification Data was updated.
4. The Input Data Type is 2000 hex, so the communications master knows that the Input Data is the final data.
5. The communications master reads the Input Data.</td><td>←</td><td>1. The EJ1 EtherCAT Slave Unit inputs the following Input Notification Data.
2. The EJ1 EtherCAT Slave Unit changes the Input SID from 01 hex to 02 hex to tell the communications master that the Input Data in the Input Notification Data was updated.
<table><tr><td>Port Status</td><td>0002 hex</td></tr><tr><td>Input SID</td><td>02 hex</td></tr><tr><td>Output SID Response</td><td>02 hex</td></tr><tr><td>Input Data Type</td><td>2000 hex</td></tr><tr><td>Input Sub Info</td><td>0000 hex</td></tr><tr><td>Input Data Length</td><td>nnnn hex</td></tr><tr><td>Input Data</td><td>Second received data</td></tr></table></td></tr>
<tr><td>12</td><td>Second input response</td><td>1. The communications master outputs the following Output Notification Data.
2. The communications master sets the Input SID Response to the value of the Input SID received in step 11 (02 hex) to tell the EJ1 EtherCAT Slave Unit that the communications master normally received the second Input Data.
<table><tr><td>Output SID</td><td>02 hex</td></tr><tr><td>Input SID Response</td><td>02 hex</td></tr><tr><td>Output Data Type</td><td>0000 hex</td></tr><tr><td>Reserved.</td><td>0000 hex</td></tr><tr><td>Output Data Length</td><td>0000 hex</td></tr><tr><td>Output Data</td><td>None</td></tr></table></td><td>→</td><td>1. The EJ1 EtherCAT Slave Unit receives the Output Notification Data given on the left.
2. The EJ1 EtherCAT Slave Unit knows that the communications master normally received the second Input Data because the Input SID Response in the Output Notification Data is 02 hex from step 11.</td></tr>
</table>

5-7 Setting CompoWay/F Frames

This section describes what data to set and what data is set in the Output Data included in Output Notification Data and Input Data included in Input Notification Data for the CompoWay/F frames during CompoWay/F communications. This section also describes how to arrange that data in array variables.

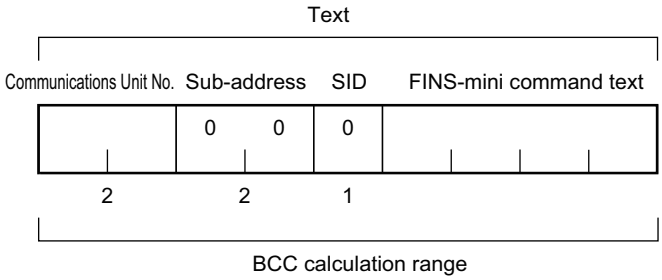
5-7-1 Output Data and Input Data Contents

This section describes the information that you must set in the Output Data (i.e., the data sent from the EJ1 EtherCAT Slave Unit to a Basic Unit) from user programming and the information that is set automatically in the Input Data (i.e., the data received by the EJ1 EtherCAT Slave Unit from the Basic Unit).

● Output Data Contents

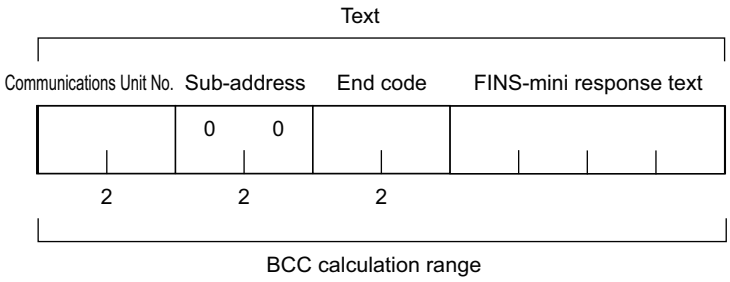
Set the text portion from the communications unit number to the FINS-mini command text in the CompoWay/F command frame to the Output Data (index: 7000 hex, subindex: 06 hex to 19 hex) with user programming.

Do not include the leading STX (02 hex) and ETX (03 hex) + BCC at the end.



● Input Data Contents

Set the text portion from the communications unit number to the FINS-mini response text in the CompoWay/F response frame to the Input Data (index: 6000 hex, subindex: 07 hex to 1A hex) with user programming. Do not include the leading STX (02 hex) and ETX (03 hex) + BCC at the end.



Frame element		Command	Response
STX: Not included in the Output Data or the Input Data.		This code designates the beginning of the communications frame.	
		The EJ1 EtherCAT Slave Unit adds the STX automatically. With the EJ1 EtherCAT Slave Unit, you do not need to send the STX from the communications master.	The response data is returned with the STX automatically removed.
Text: Included in the Output Data or the Input Data.	**Communications unit number**	Set the communications unit number that is set on switches SW1 and SW2 on the Basic Unit. Specify XX as the communications unit number for broadcasting. No response is returned for broadcasting. No response is returned for any communications unit number other than the above one.	
	Sub-address	The sub-address is not used with the EJ1 EtherCAT Slave Unit. Always set it to 3030 hex.	
	SID	The SID is not used with the EJ1 EtherCAT Slave Unit. Always set it to 30 hex.	---
	End code	---	The end code is stored.
	FINS-mini command text	This is the command.	---
	FINS-mini response text	---	This is the response.
ETX: Not included in the Output Data or the Input Data.		This code designates the end of the communications frame.	
		The EJ1 EtherCAT Slave Unit adds the ETX automatically. With the EJ1 EtherCAT Slave Unit, you do not need to send the ETX from the communications master.	The response data is returned with the ETX automatically removed.
BCC: Not included in the Output Data or the Input Data.		This is the BCC calculation result from the communications unit number to the ETX.	
		The EJ1 EtherCAT Slave Unit adds the BCC automatically. With the EJ1 EtherCAT Slave Unit, you do not need to send the BCC from the communications master.	The response data is returned with the BCC automatically removed.

Refer to the *EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142) for details on CompoWay/F commands and responses.

5-7-2 Storing Output Data and Input Data in Array Variables

If the communications master is an NJ/NX-series CPU Unit, the Output Data and Input Data are set in array variables of four bytes each.
Specifically, the data in the frame is stored in bytes in ascending order of the array element numbers starting from the communications unit number.
This is shown in the following table.

● Storing the Output Data for a CompoWay/F Command in Array Variables

Element No. ← One byte →	
Output Data 1	
0	Communications Unit No. upper byte
1	Communications Unit No. lower byte
2	30 hex (sub-address upper byte)
3	30 hex (sub-address lower byte)
Output Data 2	
0	30 hex (SID)
1	MRC upper byte
2	MRC lower byte
3	SRC upper byte
Output Data 3	
0	SRC lower byte
1	Data 1
2	Data 2
3	Data 3
. . .	. . .
Output Data m (m = 20 max.)	
0	Data
1	Data
2	Data
3	Data

FINS-mini command text

● Storing Input Data for a CompoWay/F Response in Array Variables

Element No. ← One byte →	
Input Data 01	
0	Communications Unit No. upper byte
1	Communications Unit No. lower byte
2	30 hex (sub-address upper byte)
3	30 hex (sub-address lower byte)
Input Data 2	
0	End code upper byte
1	End code lower byte
2	MRC upper byte
3	MRC lower byte
Input Data 3	
0	SRC upper byte
1	SRC lower byte
2	Response code MRES upper byte
3	Response code MRES lower byte
Input Data 4	
0	Response code SRES upper byte
1	Response code SRES lower byte
2	Data 1
3	Data 2
. . .	. . .
Input Data m (m = 20 max.)	
0	Data
1	Data
2	Data
3	Data

FINS-mini command text

Precautions for Correct Use

If the CompoWay/F command or CompoWay/F response to send or receive exceeds the PDO assignment sizes, send the data divided into parts. Refer to *Example of Sending and Receiving Data in Parts* on page 5-21 for how to send and receive data divided into parts.

5-7-3 Setting the PDO Mappings for I/O Data

You must set the PDO mappings for Input Data and Output Data used for PDO communications between the communications master and EJ1 EtherCAT Slave Unit.

You can edit the Input Data and Output Data PDO mapping sizes on the Sysmac Studio by using the PDO map settings to set the PDO sizes to between 4 and 80 bytes for both inputs and outputs.

You can adjust the Output Data and Input Data PDO mapping sizes to prioritize either CompoWay/F communications performance or PDO communications performance. For guidelines on how to determine the PDO sizes, refer to the *Guidelines for Determining the Output Data and Input Data PDO Mapping Sizes* on page 5-31 later in this manual.

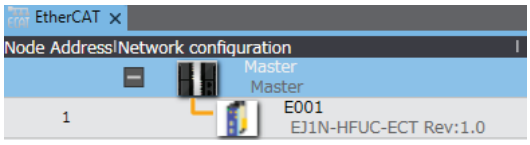
How to set the PDO mapping sizes when an EJ1 EtherCAT Slave Unit is registered to the EtherCAT network configuration is described next.

Precautions for Correct Use

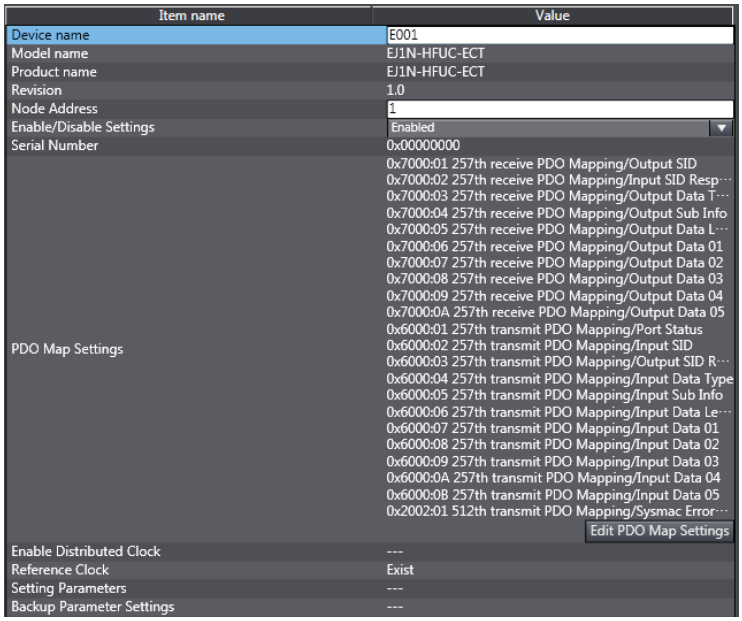
Set the assignments to meet the following conditions for PDO mapping. If the following conditions are not met, it will not be possible to change the EtherCAT communications state to the Safe-Operational (Safe–Op) state to enable PDO communications.

- Mapping must be in ascending order by index and subindex.
- Do not map a PDO at index 0000 hex, subindex 00 hex.
- Do not delete required PDO entries when mapping. Refer to *5-3-2 PDO Mapping Data List* on page 5-7 for details on the required PDO entries.

1 Click the EJ1N-HFUC-ECT in the EtherCAT network configuration.

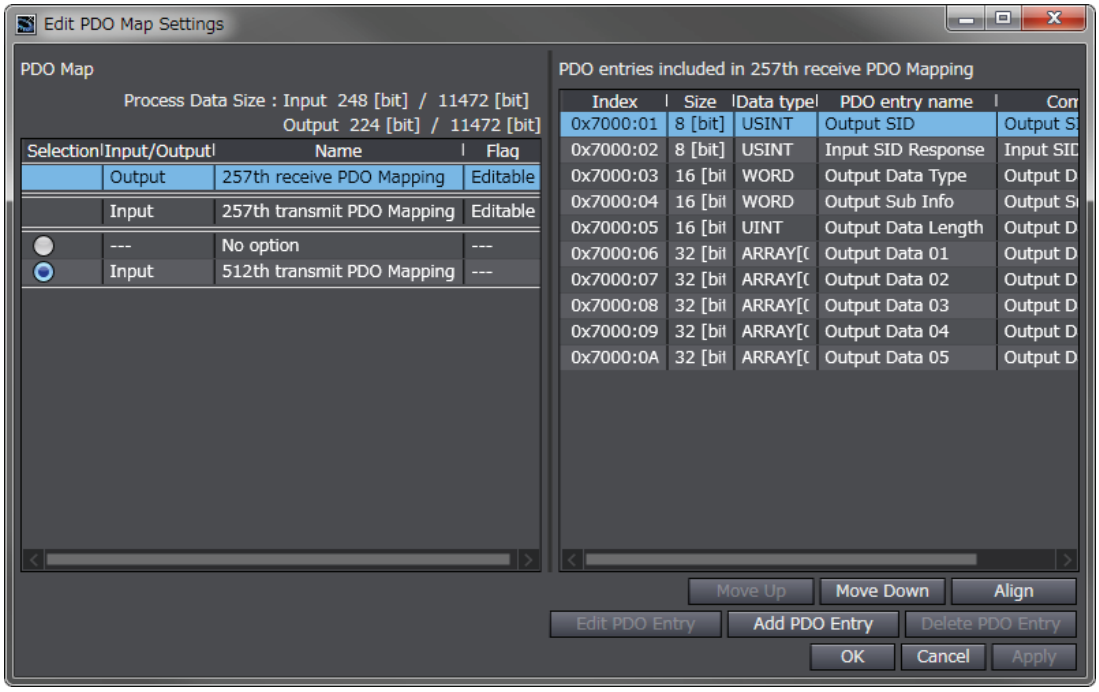


The following dialog box is displayed.



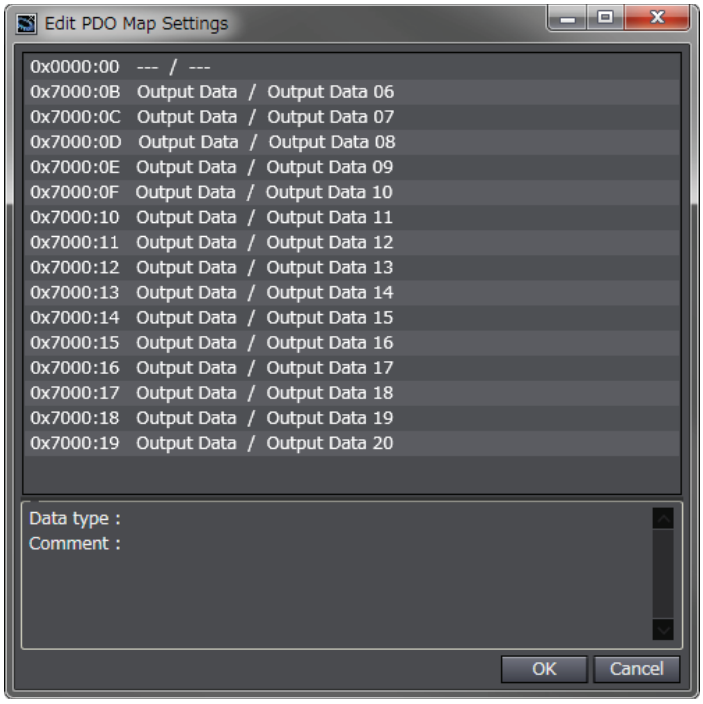
2 Click the **Edit PDO Map Settings** Button.

The following dialog box is displayed.



3 Select **257th receive PDO Mapping** and click the **Add PDO Entry** Button.

The following dialog box is displayed.



4 Select the PDO to add and click the **OK** Button.

The PDO is added.

5 Set the other PDO entries in the same way.

- The Output Data entries are 4 bytes each.

  If OutputData01 to OutputData20 are all registered, the Output Data size will be 80 bytes and the resulting Output Notification Data will be 88 bytes.
- The Input Data entries are 4 bytes each.

  If InputData01 to InputData20 are all registered, the Input Data size will be 80 bytes and the resulting Output Notification Data will be 90 bytes.

6 Click the **OK** Button or **Apply** Button in the Edit PDO Map Settings Dialog Box.

The PDO entries are added to the EJ1 EtherCAT Slave Unit settings.

Guidelines for Determining the Output Data and Input Data PDO Mapping Sizes

The PDO mapping sizes for Output Data and Input Data are determined based on the cases given in the following table.

Case		PDO mapping sizes	
		Size of Output Data in the Output Notification Data	Size of Input Data in the Input Notification Data
Case 1	Giving priority to CompoWay/F communications (As a result, the PDO mapping sizes will be larger and the process data cycle will be longer.)	The Output Data size must be larger than the maximum size of the data from the communications unit number to the FINS-mini command text in any CompoWay/F command that is used.	The Input Data size must be larger than the maximum size of the data from the communications unit number to the FINS-mini response text in any CompoWay/F response that is used.
Case 2	Giving priority to communications for frequently used CompoWay/F commands. It must be acceptable that less frequently used CompoWay/F commands are slower.	The Output Data size must be larger than the maximum size of the data from the communications unit number to the FINS-mini command text in frequently used CompoWay/F commands.	The Input Data size must be larger than the maximum size of the data from the communications unit number to the FINS-mini response text in frequently used CompoWay/F responses.
		CompoWay/F commands or CompoWay/F responses that do not meet the above conditions must be divided into parts to be sent and received with user programming.	
Case 3	Giving priority to the process data cycle. You can perform CompoWay/F communications with as little effect as possible on the process data cycle. (Conversely, it must be acceptable that CompoWay/F communications are slower as a result.)	First, determine the PDO mapping sizes to obtain a satisfactory target process data cycle. Based on those results, divide the data into parts to send and receive with user programming in the following cases: • When the maximum size of the data from the communications unit number to the FINS-mini command text in the CompoWay/F command is larger than the Output Data size or • When the maximum size of the data from the communications unit number to the FINS-mini response text in the CompoWay/F response is larger than the Input Data size Note The allowed size of process data for communications is determined by the EtherCAT communications master based on the process data cycle time. For details, refer to the manual for your EtherCAT communications master.	

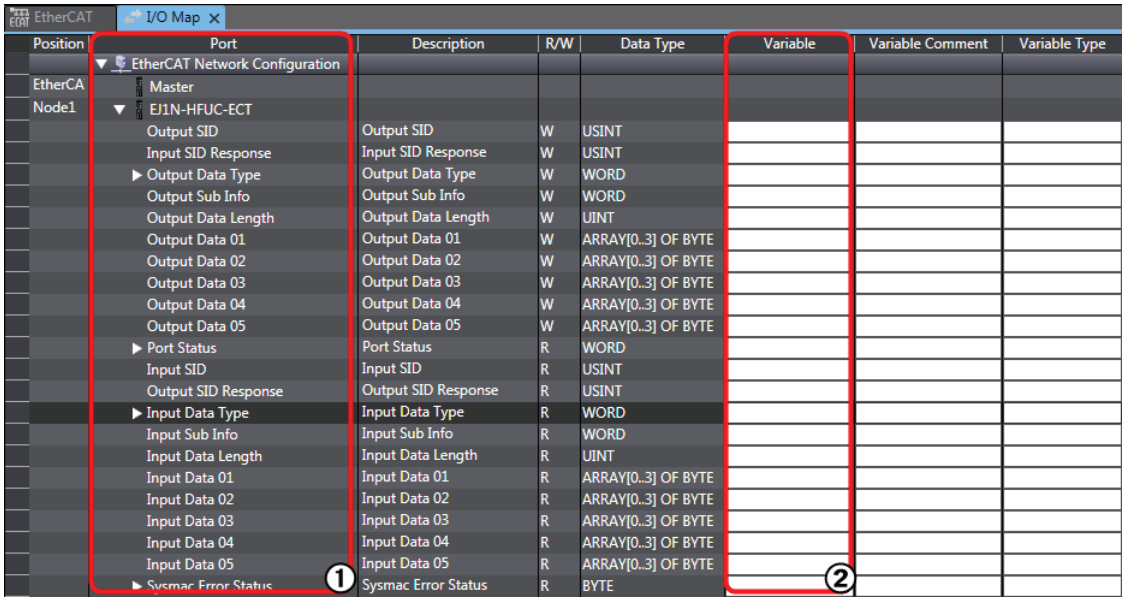
5-8 Assigning Device Variables

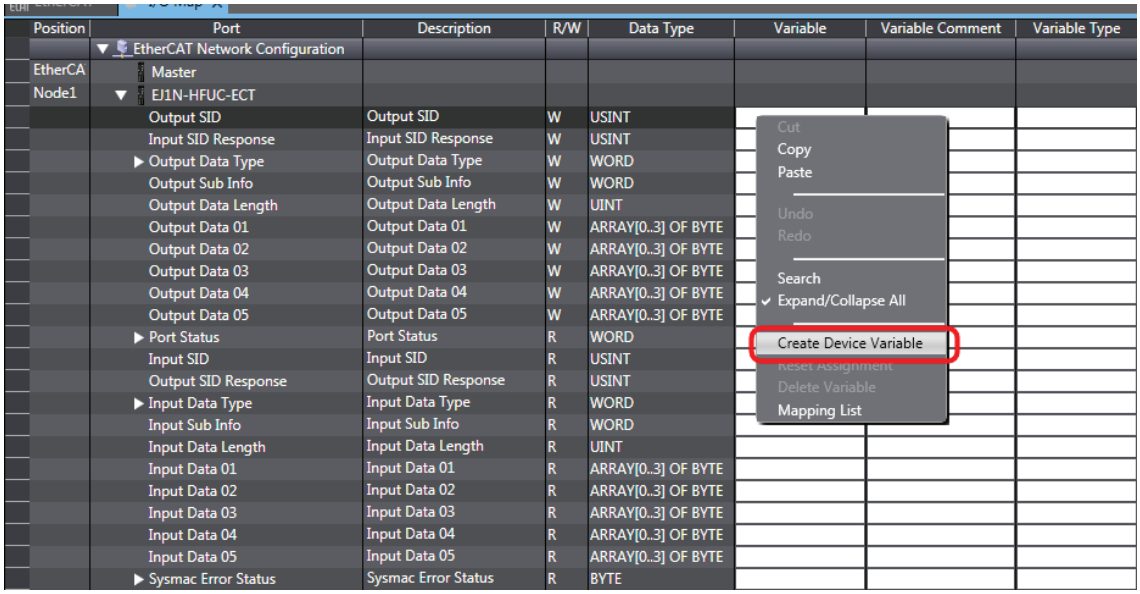
This section describes how to assign device variables to I/O ports when the communications master is an NJ/NX-series CPU Unit.

When the communications master is an NJ/NX-series CPU Unit, you must assign device variables to the I/O ports on the I/O Map Tab Page of the Sysmac Studio to access PDO entries from user programming.

Area (1), below, gives the I/O port names for the EJ1N-HFUC-ECT EJ1 EtherCAT Slave Unit.

You can right-click on any of these I/O ports and select **Create Device Variable** to assign device variables.





6

Using the EJ1 EtherCAT Slave Unit with SDO Communications

This section describes how to read and write data in the EJ1 EtherCAT Slave Unit by using SDO communications from the communications master.

6-1 Overview **6-2**

6-2 Abort Codes **6-3**

6-1 Overview

The EJ1 EtherCAT Slave Unit supports SDO message communications.

The communications master can read and write data in entries in the object dictionary with SDO communications to make parameter settings and monitor status when necessary.

If the communications master is an NJ/NX-series CPU Unit, the following EtherCAT communications instructions are used.

Instruction	Function
EC_CoESDORead	Reads a value from a CoE object in the specified slave.
EC_CoESDOWrite	Writes a value to a CoE object in the specified slave.

For details on the EtherCAT communications instructions, refer to the *NJ/NX-series Instructions Reference Manual* (Cat. No. W502).

6-2 Abort Codes

The following table lists the abort codes for SDO communications errors.

Value	Meaning
06010000 hex	Unsupported access to an object
06010002 hex	Attempt to write to a read-only object.
06020000 hex	The object does not exist in the object directory.
06040041 hex	The object cannot be mapped to the PDO.
06040042 hex	The number and length of the objects to be mapped would exceed the PDO length
06070010 hex	Data type does not match or length of service parameter does not match.
06090011 hex	Subindex does not exist.
06090030 hex	Value range of parameter exceeded (only for write access)
06090031 hex	Value of parameter that was written is too high.
06090032 hex	Value of parameter that was written is too low.
08000021 hex	Data cannot be transferred or stored to the application because of local control.
08000022 hex	Data cannot be transferred or stored to the application because of the present device state.

7

Troubleshooting

This section describes the notification and confirmation methods and checking procedures when an error occurs in an EJ1 EtherCAT Slave Unit.

7-1 Error Notification Methods 7-2

7-2 Error List 7-3

7-2-1 Errors Detected during Boot Process 7-3

7-2-2 EtherCAT Communications Errors 7-4

7-2-3 Errors Detected during a Software Process 7-5

7-3 Error Notifications Based on the Sysmac Error Status 7-6

7-3-1 Sysmac Error Status 7-6

7-3-2 Checking and Responding to Event Codes 7-7

7-4 Error Notifications Provided with Emergency Messages 7-8

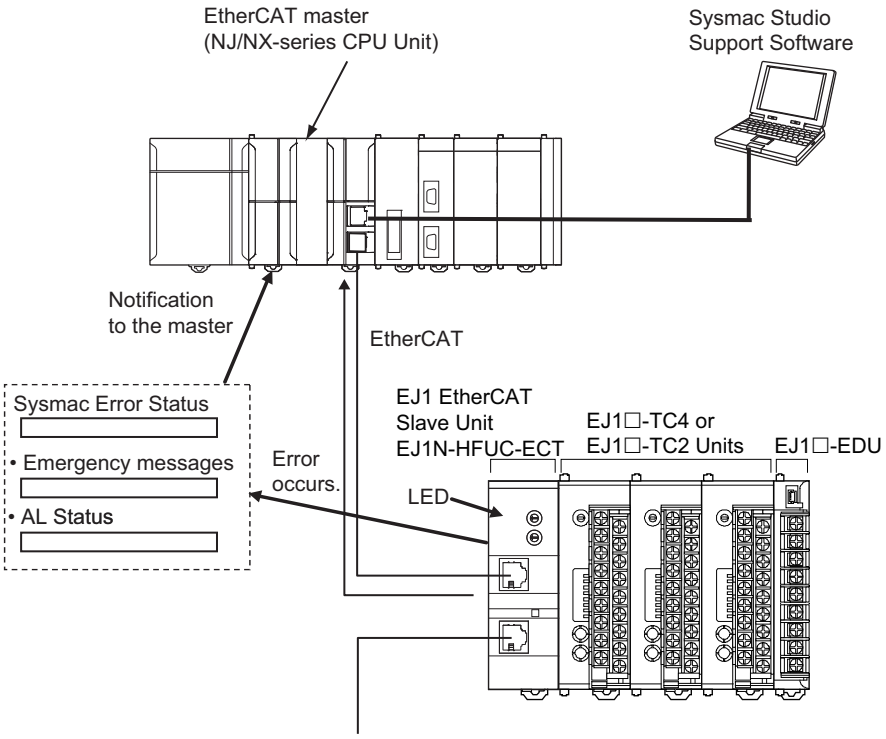
7-4-1 Emergency Messages 7-8

7-5 Error Notifications Based on the AL Status 7-9

7-1 Error Notification Methods

This section describes how the communications master and other devices are notified when an error occurs in an EJ1 EtherCAT Slave Unit.

Use the following methods to check the status of errors in the EJ1 EtherCAT Slave Unit.



Type of error notification	Description	Notification method	Reference
Indicators	Operation indicators on the front panel of the EJ1 EtherCAT Slave Unit.	---	*3-2 EtherCAT Communications Overview* on page 3-5 *7-2 Error List* on page 7-3
Sysmac error status	This status reports errors that are detected in the application layer.	The status is assigned to a TxPDO and sent cyclically to the master.	*7-3 Error Notifications Based on the Sysmac Error Status* on page 7-6
Emergency messages	The emergency messages report application level errors. The error code contains the CiA-defined error code as well as an additional error code in the vendor-specific area.	When an error occurs, the EJ1 EtherCAT Slave Unit notifies the communications master.	*7-4 Error Notifications Provided with Emergency Messages* on page 7-8
AL status	This status reports errors related to EtherCAT communications. The ETG-defined method is used for the error detection and error code.	When an error occurs, the error is written to the AL status register to notify the communications master.	*7-5 Error Notifications Based on the AL Status* on page 7-9

Precautions for Correct Use

Refer to the *EJ1 Modular Temperature Controllers User's Manual* (Cat. No. H142) for information on errors that occur in the Basic Units.

7-2 Error List

This section provides a list of errors that can occur in the EJ1 EtherCAT Slave Unit.

7-2-1 Errors Detected during Boot Process

Error name	Description	Detection state	Indicators		Operation		Master notified?			Possible correction
			ECAT		EtherCAT state transition	State restriction	AL status code	Emergency code	Event code	
			RUN	ERR						
ESC Error *1, *2	An error occurred in the EtherCAT slave communications controller.	I	Not lit.	Lit.	Stops in Init state.	Init	No	No	No	If this error is not cleared when the power supply to the Unit is cycled, there is a Unit hardware failure. Replace the Unit.
ESC Initialization Error*1, *2	Initialization of the EtherCAT slave communications controller failed.	I	Not lit.	Lit.	Stops in Init state.	Init	No	No	No	If this error is not cleared when the power supply to the Unit is cycled, there is a Unit hardware failure. Replace the Unit.
Slave Hardware Error *1, *2	A hardware error occurred in the Slave Unit.	I	Not lit.	Lit.	Stops in Init state.	Init	No	No	No	If this error is not cleared when the power supply to the Unit is cycled, there is a Unit hardware failure. Replace the Unit.

*1. This error is detected as a Slave Initialization Error (event code: 84230000 hex) or a Network Configuration Verification Error (event code: 84220000 hex) in an NJ/NX-series CPU Unit.

*2. If this type of error occurs, the user is unable to determine the type of error because EtherCAT communications will be unusable.

Note The meanings for the symbols in the Detection state column above are as follows:

I: Init state, I → P: During transition from Init state to Pre-Op state, P: Pre-Op state, P → S: During transition from Pre-Op state to Safe-Op state, S: Safe-Op state, S → O: During transition from Safe-Op to Op state, O: OP state, and −: Undefined

7-2-2 EtherCAT Communications Errors

Error name	Description	Detection state	Indicators		Operation		Master notified?			Possible correction
			ECAT		EtherCAT state transition *1	State restriction	AL status code	Emergency code	Event code	
			RUN	ERR						
Slave Unit Verification Error*2	An error occurred in Slave Unit verification.	I→P	Not lit.	Flashing	Init + ERR	No	0014 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
Mailbox Setting Error*2	An incorrect mailbox setting was detected for the Sync Manager.	I→P	Not lit.	Flashing	Init + ERR	No	0016 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
RxPDO Setting Error*2	An error was detected in the RxPDO settings.	P→S	Flashing	Flashing	Pre-Op + ERR	No	001D hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
TxPDO Setting Error*2	An error was detected in the TxPDO settings.	P→S	Flashing	Flashing	Pre-Op + ERR	No	001E hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
PDO WDT Setting Error*2	An incorrect PDO WDT setting was detected.	P→S	Flashing	Flashing	Pre-Op + ERR	No	001F hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
Init State Transition Request	The slave sent a request to the EtherCAT master to change to the Init state.	P→S	Flashing	Flashing	Pre-Op + ERR	No	0021 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
TxPDO Mapping Error*2	An incorrect TxPDO was set.	P→S	Flashing	Flashing	Pre-Op + ERR	No	0024 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
RxPDO Mapping Error*2	An incorrect RxPDO was set.	P→S	Flashing	Flashing	Pre-Op + ERR	No	0025 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
Illegal State Transition Request Received*2	An incorrect state transition request was received.	---	(Undefined.)	Flashing	Current status + ERR	No	0011 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
Error State Transition Received*2	An unclear state transition request was received.	---	(Undefined.)	Flashing	Current status + ERR	No	0012 hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9
Process Data WDT Error*2	Process data communications were stopped for more than the specified period of time.	O	Single flash	Double flash	Safe-Op + ERR	No	001B hex	No	No	*7-5 Error Notifications Based on the AL Status* on page 7-9

*1. + ERR in the EtherCAT state transition column means that an error defined in the EtherCAT specifications has occurred. In this state, the user is notified of an AL status error.

*2. This error is detected as a Slave Application Error (event code: 84280000 hex) in an NJ/NX-series CPU Unit. The AL status code that the EJ1 EtherCAT Slave Unit reports to the master is stored as Sub Info 1 of the Slave Application Error (event code: 84280000 hex).

Note The meanings for the symbols in the Detection state column above are as follows:

I: Init state, I → P: During transition from Init state to Pre-Op state, P: Pre-Op state, P → S: During transition from Pre-Op state to Safe-Op state, S: Safe-Op state, S → O: During transition from Safe-Op to Op state, O: Op state, and −: Undefined

7-2-3 Errors Detected during a Software Process

Error name	Description	Detection state	Indicators		Operation		Master notified?			Possible correction
			ECAT		EtherCAT state transition	State restriction	AL status code	Emergency code	Event code	
			RUN	ERR						
Unit Processing Error*1*2	A fatal error was detected in the EJ1 EtherCAT Slave Unit.	---	Not lit.	Lit.	Changed to Init state.	Init	No	No	No	If this error is not cleared when the power supply to the Unit is cycled, there is a Unit hardware failure. Replace the Unit.
EJ1 Bus Communications Error *3	An EJ1 bus communications error occurred due to noise or other cause.	---	(Undefined.)	(Undefined.)	Current state + Sysmac Error Status (Observation) generated	No	No	FF01 hex	08300000 hex	Cause: Noise influence Correction: Implement noise countermeasures if there is excessive noise.
BCC Error*3	The BCC check on the response received from the Basic Units failed. The data received from the Basic Units will be discarded.	---	(Undefined.)	(Undefined.)	Current state + Sysmac Error Status (Observation) generated	No	No	FF02 hex	08310000 hex	Cause: Incomplete connection between the EJ1 EtherCAT Slave Unit and the Basic Units. Correction: Install the Units properly.

*1. This error is detected as a Slave Initialization Error (event code: 84230000 hex) or a Process Data Communications Error (event code: 842C0000 hex) in an NJ/NX-series CPU Unit.

*2. If this type of error occurs, the user is unable to determine the type of error because EtherCAT communications will be unusable.

*3. In an NJ/NX-series CPU Unit, this error is detected as a Sysmac Error Status.

Note The meanings for the symbols in the Detection state column above are as follows:

I: Init state, I → P: During transition from Init state to Pre-Op state, P: Pre-Op state, P → S: During transition from Pre-Op state to Safe-Op state, S: Safe-Op state, S → O: During transition from Safe-Op to Op state, O: Op state, and −: Undefined

7-3 Error Notifications Based on the Sysmac Error Status

This section describes how the communications master is notified of a Sysmac Error Status.

7-3-1 Sysmac Error Status

The Sysmac Error Status is assigned to a TxPDO to provide cyclic notifications of the level of current errors in the EJ1 EtherCAT Slave Unit to the communications master. This allows you to use the same operation to check errors and corrections with an NJ/NX-series CPU Unit and the Sysmac Studio.

With an NJ/NX-series CPU Unit, you can use the *_EC_SlavErr* system-defined variable or a device variable for the EJ1 EtherCAT Slave Unit (Sysmac Error Status) to detect errors that are shown in the Sysmac Error Status.

Errors shown in the Sysmac Error Status are retained even if the cause of the error is removed. You can write 1 for the Sysmac Error Status Clear object (index: 2002 hex, subindex: 02 hex) to clear the error status detected by the EJ1 EtherCAT Slave Unit.

With an NJ/NX-series CPU Unit, you can use an instruction to clear the error status detected by the EJ1 EtherCAT Slave Unit. You can also clear this error status from the Sysmac Studio with the Reset All Button in the Troubleshooter.

7-3-2 Checking and Responding to Event Codes

When an error occurs, you can view all current errors according to their Sysmac Error Status level through the communications master.

You can check the errors that are managed by the EJ1 EtherCAT Slave Unit through the Sysmac Error Status.

You can connect the NJ/NX-series CPU Unit and the Sysmac Studio to check an error detected by the EJ1 EtherCAT Slave Unit with the event code given in Sub Info 3 under *Emergency Message Detected* on the Controller Event Log Tab Page. Refer to *Event Code List* on page 7-7 for details on event codes.

For non-OMRON masters, you can check the current error details by reading data from the objects at indexes 2003 hex and 2004 hex. For details, refer to *A-3-6 Manufacturer-specific Object 1* on page A-15 under *A-3 CoE Objects* on page A-9.

● Event Code List

The events that occur in the EJ1 EtherCAT Slave Unit are listed below.

The following abbreviations are used in the Level column.

Abbreviation	Name
Maj	Major fault level
Prt	Partial fault level
Min	Minor fault level
Obs	Observation
Info	Information

Symbol	Name
✓	Event levels that are defined by the system.

Event code	Event name	Meaning	Assumed cause	Correction	Level				
					Maj	Prt	Min	Obs	Info
08300000 hex	EJ1 Bus Communications Error	An EJ1 bus communications error occurred due to noise or other cause.	A problem, such as noise, caused one of the following communications errors on the EJ1 bus: Parity Error, Framing Error, Overrun Error, or Noise Error.	If this error occurs irregularly, implement noise countermeasures.				✓	
08310000 hex	BCC Error	The BCC check failed on the CompoWay/F response received from the Basic Units. The data received from the Basic Units will be discarded.	An error occurred in the CompoWay/F BCC due to a Basic Unit error or some other problem, such as noise.	If this error occurs irregularly, implement noise countermeasures.				✓	

7-4 Error Notifications Provided with Emergency Messages

This section describes how the communications master is notified with emergency messages.

7-4-1 Emergency Messages

The EJ1 EtherCAT Slave Unit reports emergency messages to the communications master by using SDO communications if it detects an error at the application level.

When the power supply to the EJ1 EtherCAT Slave Unit is turned ON, the EJ1 EtherCAT Slave Unit will always start with emergency message notifications enabled.

With an NJ/NX-series CPU Unit, an Emergency Message Detected event (event code: 6420 0000 hex) occurs when the EJ1 EtherCAT Slave Unit sends an emergency message.

When this event code error is detected, place the Sysmac Studio online with the NJ/NX-series CPU Unit. Details on the emergency message, including the emergency error code, are stored in Sub Info 1 to Sub Info 3 under *Emergency Message Detected* on the Controller Event Log Tab Page.

An emergency message consists of 8 bytes of data as shown below.

Byte	0	1	2	3	4	5	6	7
Meaning	Emergency error code (Sub Info 1)		Reserved		Event code (Sub Info 3)			

● Emergency Error Code List

The following table provides details on and corrections for the emergency error codes used by the EJ1 EtherCAT Slave Unit.

Emergency error code	Error name	Meaning	Possible correction
FF01 hex	EJ1 Bus Communications Error	An EJ1 bus communications error occurred due to noise or other cause.	If this error occurs irregularly, implement noise countermeasures.
FF02 hex	BCC Error	The BCC check on the response received from the Basic Units failed. The data received from the Basic Units will be discarded.	If this error occurs irregularly, implement noise countermeasures. If the error still occurs, replace the Basic Unit.

Emergency error codes allow you to isolate the causes of errors.

Emergency messages cannot be sent while there is an EtherCAT communications error.

Refer to *Event Code List* on page 7-7 for details on event codes.

7-5 Error Notifications Based on the AL Status

This section describes how the communications master is notified of the AL status (an error status related to EtherCAT communications).

When an error related to EtherCAT communications occurs, the AL status code is written to the AL status register in the EJ1 EtherCAT Slave Unit. This allows the communications master to detect errors related to EtherCAT communications with the EJ1 EtherCAT Slave Unit.

If an AL status error occurs in the EJ1 EtherCAT Slave Unit, it is detected as a Slave Application Error (event code: 8428 0000 hex) in an NJ/NX-series CPU Unit.

When this event code error is detected, place the Sysmac Studio online with the NJ/NX-series CPU Unit. The following AL status codes are stored in Sub Info 1 under Slave Application Error in the event log.

AL status codes allow you to isolate the causes of errors.

● AL Status Code List

The following table provides details on and corrections for the AL status error codes used by the EJ1 EtherCAT Slave Unit.

AL status code	Name	Cause	Possible correction
0011 hex	Illegal State Transition Request Received	An incorrect state transition request was received.	Change the state correctly.
0012 hex	Error State Transition Received	An unclear state transition request was received.	Change the state correctly.
0014 hex	Slave Unit Verification Error	The non-volatile memory for SII back-ups does not match the information in the SII (VendorID, ProductCode, RevisionNo, and Serial No.).	Cycle the power supply. If the error still occurs, replace the Unit.
0016 hex	Mailbox Setting Error	An incorrect mailbox setting was detected for the Sync Manager.	Correct the settings, and then download the settings to the communications master again.
001B hex	Process Data WDT Error	A timeout was detected for an I/O data send frame.	Review the following items, and restart the slave based on the specifications of the connected communications master. • Wire the EtherCAT communications cable correctly. • Check to see if the EtherCAT communications cable is exposed to excessive noise.
001D hex	RxPDO Setting Error	An error was detected in the RxPDO settings.	Correct the settings, and then download the settings to the communications master again.
001E hex	TxPDO Setting Error	An error was detected in the TxPDO settings.	Correct the settings, and then download the settings to the communications master again.

AL status code	Name	Cause	Possible correction
001F hex	PDO WDT Setting Error	An incorrect PDO WDT setting was detected.	Correct the settings, and then download the settings to the communications master again.
0021 hex	Init State Transition Request	The slave sent a request to the EtherCAT master to change to the Init state.	Set the Slave Unit to Init state, and then resume communications.
0024 hex	TxPDO Mapping Error	An illegal TxPDO was set. • An incorrect TxPDO was set, e.g., the index, subindex, or size was outside of the allowable range.	Correct the settings, and then download the settings to the communications master again.
0025 hex	RxPDO Mapping Error	An illegal RxPDO was set. • An incorrect RxPDO was set, e.g., the index, subindex, or size was outside of the allowable range.	Correct the settings, and then download the settings to the communications master again.

8

Maintenance and Inspection

This section describes how to clean, inspect, and maintain an EJ1 EtherCAT Slave Unit.

8-1 Cleaning and Inspection 8-2

8-1-1 Cleaning 8-2

8-1-2 Inspection Methods 8-2

8-1-3 Inspection Items 8-3

8-2 Replacing Units 8-4

8-2-1 Unit Replacement Precautions 8-4

8-2-2 Unit Replacement Procedure 8-4

8-1 Cleaning and Inspection

This section describes the cleaning and inspection methods recommended as regular maintenance.

8-1-1 Cleaning

Clean the EJ1 EtherCAT Slave Unit regularly as described below in order to keep the network in optimal operating condition.

- Wipe the equipment over with a soft, dry cloth when performing daily cleaning.
- If dirt remains even after wiping with a soft, dry cloth, wipe over with a cloth that has been wet with a sufficiently diluted detergent (2%) and wrung dry.
- Smudges may remain on the Unit from gum, vinyl, or tape that was left on for a long time. Remove these smudges when cleaning.

Precautions for Correct Use

Never use volatile solvents, such as paint thinner or benzene, or chemical wipes to clean the Unit. These substances may deteriorate the surface coating of the EJ1 EtherCAT Slave Unit.

8-1-2 Inspection Methods

Perform regular inspections to maintain optimal operating conditions.

In general, inspect the system once every 6 to 12 months, but inspect more frequently if the system is used in hot, humid, or dusty conditions.

● Equipment Required for Inspection

Prepare the following equipment for inspection.

● Regularly Required Items

- Flat-blade and Phillips screwdrivers
- A screwdriver for connecting communications connectors
- A voltage tester (or digital voltmeter)
- Industrial alcohol and pure cotton cloth

● Equipment Required Occasionally

- Synchroscope
- Pen-writing oscilloscope
- Thermometer and hygrometer (humidity meter)

8-1-3 Inspection Items

Check the items in the following table for any condition that does not meet the criteria. Adjust the EJ1 EtherCAT Slave Unit or improve the operating environment to correct the situation if any of the following criteria is not met.

Inspection Item	Inspection details	Criteria	Inspection method
Environmental conditions	Are the ambient and cabinet temperatures correct?	-10 to 55°C	Thermometer
	Are the ambient and cabinet humidities correct?	25% to 85%	Hygrometer
	Has dust or dirt accumulated?	No dust or dirt	Visual inspection
Installation conditions	Is the Unit installed securely?	No looseness	Phillips screwdriver
	Are the connectors of the communications cables fully inserted?	No looseness	Visual inspection
	Are the external wiring screws tight?	No looseness	Phillips screwdriver
	Are the connecting cables undamaged?	No externally visible damage	Visual inspection

8-2 Replacing Units

The EJ1 EtherCAT Slave Unit is part of an EtherCAT network. The entire network is affected when a Unit is faulty, so a faulty Unit must be repaired or replaced quickly. We recommend having spare Units available to restore network operation as quickly as possible.

8-2-1 Unit Replacement Precautions

Observe the following precautions when you replace a faulty Unit.

- Check the new Unit to make sure that there are no errors.
- If returning malfunctioning devices for repair, attach a detailed description of the malfunction to the device and send the device to the OMRON representative listed at the end of this manual or to your OMRON representative.
- If there is faulty contact, try wiping the contacts with a clean, lint-free cotton cloth dampened with alcohol.

8-2-2 Unit Replacement Procedure

Step	Item	Description
1	System power supply OFF	Turn OFF the power supply to the system.
2	Unit replacement	Replace the Unit.
3	Settings after Unit replacement	• After replacing a Unit, set the new Unit's EtherCAT node address rotary switches to the same address setting as on the old Unit. • If you use a software setting for the EtherCAT node address, use the Sysmac Studio or other EtherCAT master support software to set the EtherCAT node address. If you use the Sysmac Studio's backup function, you can restore the settings from a backup to recover the EtherCAT node address software settings. • Connect the same model of EJ1 EtherCAT Slave Unit to the Basic Units as the previous EJ1 EtherCAT Slave Unit.

Note If you do not use a software setting for the EtherCAT node address, you do not need to perform backup and restore operations before and after you replace the Unit.

A

Appendices

The appendices provide additional information for the EJ1 EtherCAT Slave Unit, such as dimensions and object lists.

A-1 Dimensions A-2

A-1-1 EJ1N-HFUC-ECT A-2

A-2 Sample Programming A-3

A-2-1 Items Common to All Programming Samples A-3

A-2-2 Sending CompoWay/F Commands and Receiving Responses A-6

A-2-3 Restart Port A-8

A-3 CoE Objects A-9

A-3-1 Object Dictionary Area A-9

A-3-2 Data type A-9

A-3-3 Communications Objects A-10

A-3-4 PDO Mapping Objects A-11

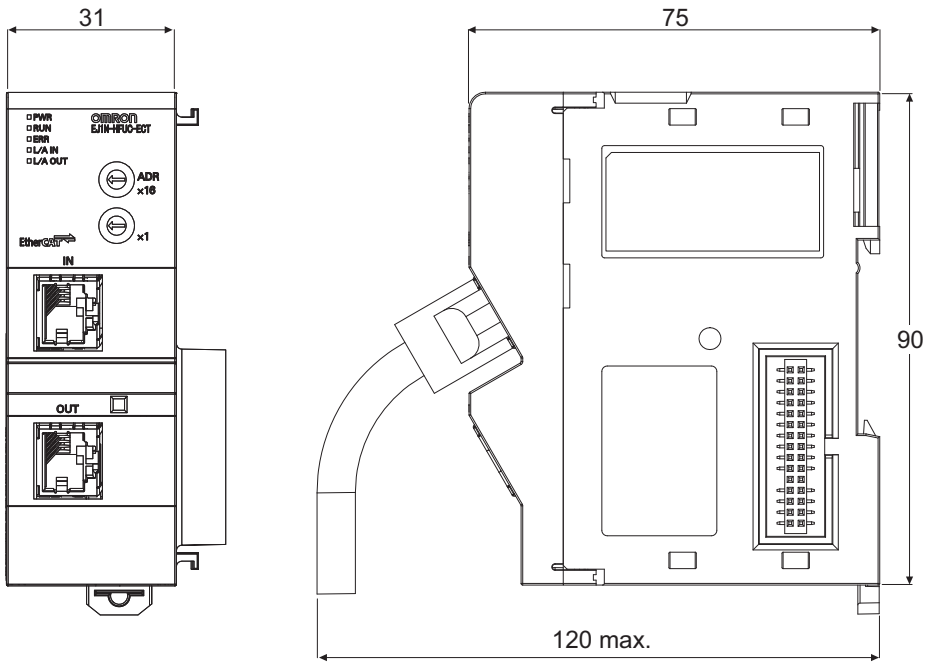
A-3-5 Sync Manager Communications Objects A-14

A-3-6 Manufacturer-specific Object 1 . . . . . A-15

A-3-7 Device Profile Area A-17

A-1 Dimensions

A-1-1 EJ1N-HFUC-ECT



(Unit: mm)

A-2 Sample Programming

Precautions for Correct Use

- These programming samples do not implement processing for when a timeout occurs while waiting to receive a CompoWay/F response.
- Implement timeout and retry processing during CompoWay/F response confirmation in your application.

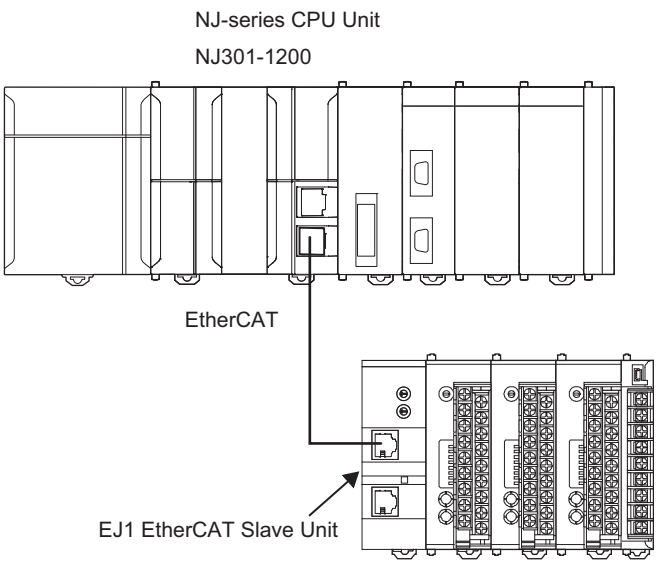
A-2-1 Items Common to All Programming Samples

This section specifies the items that are the same in all three ST programming samples.

System Configuration

The following system configuration is used.

Item	Specification
Communications master	NJ301-1200
EJ1 EtherCAT Slave Unit	EJ1-HFUC-ECT
EJ1 System	EJ1 Basic I/O Units



PDO Size Settings

The following PDO sizes are set from the Sysmac Studio as the default values.

Data	I/O entry name	Size [bytes]
Input Data	Input Data 01 to 05	20
Output Data	Output Data 01 to 05	20

I/O Map

The following I/O map settings for variables are made from the Sysmac Studio.

Unit	Port	Description	Variable
EJ1-HFUC-ECT	Output SID	Output SID	E001_Output_SID
	Input SID Response	Input SID Response	E001_Input_SID_Response
	Output Data Type	Output Data Type	E001_Output_Data_Type
	Output Sub Info	Output Sub Info	E001_Output_Sub_Info
	Output Data Length	Output Data Length	E001_Output_Data_Length
	Output Data 01	Output Data 01	E001_Output_Data_01
	Output Data 02	Output Data 02	E001_Output_Data_02
	Output Data 03	Output Data 03	E001_Output_Data_03
	Output Data 04	Output Data 04	E001_Output_Data_04
	Output Data 05	Output Data 05	E001_Output_Data_05
	Port Status	Port Status	E001_Port_Status
	Input SID	Input SID	E001_Input_SID
	Output SID Response	Output SID Response	E001_Output_SID_Response
	Input Data Type	Input Data Type	E001_Input_Data_Type
	Input Sub Info	Input Sub Info	E001_Input_Sub_Info
	Input Data Length	Input Data Length	E001_Input_Data_Length
	Input Data 01	Input Data 01	E001_Input_Data_01
	Input Data 02	Input Data 02	E001_Input_Data_02
	Input Data 03	Input Data 03	E001_Input_Data_03
	Input Data 04	Input Data 04	E001_Input_Data_04
	Input Data 05	Input Data 05	E001_Input_Data_05
	Sysmac Error Status	Sysmac Error Status	E001_Sysmac_Error_Status

Internal Variable Table

The following variable table is set from the Sysmac Studio. All of the variables that you registered in the I/O map are registered as global variables.

Name	Data type	Initial value	Comment
rcvCommand	ARRAY[0..255] OF BYTE		Buffer for input data storage
sendCommand	ARRAY[0..255] OF BYTE		Buffer for output data storage
State	INT	0	State transition
sendLength	INT		Output data size
Offset	INT		Data offset when sending or receiving data in parts
outSid	USINT		Output SID
inSid	USINT		Input SID
sendString	STRING[256]		CompoWay/F command text string to send
rcvString	STRING[256]		Received CompoWay/F response text string

Common Function: nextSid

This function is used in the programming samples.

It returns the next SID after the SID it is given.

- Variable Table I/O

Name	Data type	Comment
currentSid	USINT	Present value of the SID argument to the nextSid function

- Variable Table Return Value

Name	Data type	Comment
nextSid	USINT	The return value of the nextSid function, which is the next SID

- nextSid Function

```
1  nextSid:=currentSid+1;
2  IF(nextSid=USINT#0)THEN
3      nextSid:=1;
4  END_IF;
```

Add 1 to the given SID.

If the resulting SID from the above calculation is 0, set the SID to 1.

A-2-2 Sending CompoWay/F Commands and Receiving Responses

Sending and Receiving Data without Dividing It

The following programming sample sends a FINS-mini echoback test to communications unit number 0.

```
CASE state OF
  0: //Set the CompoWay/F command.
     sendString:='0000008011234';
     sendLength:=StringToAry(sendString , sendCommand[0]);
     inSid:=nextSid(E001_Input_SID);
     state:=10;
  10: //Send CompoWay/F command.
     outSid:=nextSid(E001_Output_SID_Response);
     E001_Output_SID:=outSid;
     E001_Output_Data_Type:=16#2000;
     E001_Output_Data_Length:=sendLength;
     AryMove(sendCommand[0],E001_Output_Data_01[0] ,UINT#4);
     AryMove(sendCommand[4],E001_Output_Data_02[0] ,UINT#4);
     AryMove(sendCommand[8],E001_Output_Data_03[0] ,UINT#4);
     AryMove(sendCommand[12] ,E001_Output_Data_04[0] ,UINT#4);
     AryMove(sendCommand[16] ,E001_Output_Data_05[0] ,UINT#4);
     state:=20;
  20: //Confirm that the command was sent.
     IF(E001_Output_SID_Response=outSid)THEN
        state:=30;
     END_IF;
  30: //Check CompoWay/F response reception.
     IF(E001_Input_SID=inSid)THEN
        state:=40;
     END_IF;
  40: //Receive the CompoWay/F response.
     E001_Input_SID_Response:=inSid;
     IF(E001_Receive_Data_Error_Flag=TRUE)THEN
        state:=50;
     END_IF;
     IF(E001_Receive_Last_Data_Flag=TRUE)THEN
        AryMove(E001_Input_Data_01[0] ,rcvCommand[0],UINT#4);
        AryMove(E001_Input_Data_02[0] ,rcvCommand[4],UINT#4);
        AryMove(E001_Input_Data_03[0] ,rcvCommand[8],UINT#4);
        AryMove(E001_Input_Data_04[0] ,rcvCommand[12],UINT#4);
        AryMove(E001_Input_Data_05[0] ,rcvCommand[16],UINT#4);
        rcvString:=AryToString(rcvCommand[0] ,E001_Input_Data_Length);
        state:=60;
     ELSE
        state:=50;
     END_IF;
  50: //Error occurs.
     ;
  60: //Normal end
     ;
END_CASE;
```

①Initial processing
Set the CompoWay/F command.
Copy data to *sendCommand*.
Calculate next Input SID.
Go to step ②.

②Send CompoWay/F command.
Calculate next Output SID.
Set the Output SID.
Set Output Data Type.
Set Output Data Length.
Set Output Data.
Go to step ③.

③Send processing and confirm that the command was sent.
If the Output SID Response is updated, go to step ④.

④Check that a CompoWay/F response was received.
If the Input SID is updated, go to step ⑤.

⑤Receive CompoWay/F response.
Update the Input SID Response.
If a receive data error occurs, go to step ⑥.
Receive the final data.
Read the Input Data.
Store the Input Data in *rcvString*.
Go to step ⑦.
If the received data is not the final data, go to step ⑥.

⑥Processing after error.

⑦Processing after normal end.

Sending and Receiving Data in Parts

The following programming sample sends a FINS-mini echoback test to communications unit number 0. In this sample, the test data is long and must be sent and received in parts through data exchange with the EJ1 EtherCAT Slave Unit using PDO communications.

```
 1  CASE state OF
 2    0: //Set the CompoWay/F command.
 3       sendString:='00000080112345678ABCD012345678ABCD012345678ABCD';
 4       sendLength:=StringToAry(sendString , sendCommand[0]);
 5       inSid:=nextSid(E001_Input_SID);
 6       offset:=0;
 7       state:=10;
 8    10: //Send CompoWay/F command.
 9       outSid:=nextSid(E001_Output_SID_Response);
10       E001_Output_SID:=outSid;
11       IF(sendLength>20)THEN
12          E001_Output_Data_Type:=16#1000;
13          E001_Output_Data_Length:=20;
14          sendLength:=sendLength-20;
15       ELSE
16          E001_Output_Data_Type:=16#2000;
17          E001_Output_Data_Length:=sendLength;
18       END_IF;
19       AryMove(sendCommand[offset+0],E001_Output_Data_01[0] ,UINT#4);
20       AryMove(sendCommand[offset+4],E001_Output_Data_02[0] ,UINT#4);
21       AryMove(sendCommand[offset+8],E001_Output_Data_03[0] ,UINT#4);
22       AryMove(sendCommand[offset+12] ,E001_Output_Data_04[0] ,UINT#4);
23       AryMove(sendCommand[offset+16] ,E001_Output_Data_05[0] ,UINT#4);
24       offset:=offset+20;
25       state:=20;
26    20: //Confirm that the command was sent.
27       IF(E001_Output_SID_Response=outSid)THEN
28          IF(E001_Output_Data_Type=16#2000)THEN
29             offset:=0;
30             state:=30;
31          ELSE
32             state:=10;
33          END_IF;
34       END_IF;
35    30: //Check CompoWay/F response reception.
36       IF(E001_Input_SID=inSid)THEN
37          state:=40;
38       END_IF;
39    40: //Receive the CompoWay/F response.
40       E001_Input_SID_Response:=inSid;
41       inSid:=nextSid(E001_Input_SID);
42       IF(E001_Receive_Data_Error_Flag=TRUE)THEN
43          state:=50;
44       END_IF;
45       IF((E001_Receive_Data_Flag=TRUE ) OR (E001_Receive_Last_Data_Flag=TRUE))THEN
46          AryMove(E001_Input_Data_01[0] ,rcvCommand[offset+0],UINT#4);
47          AryMove(E001_Input_Data_02[0] ,rcvCommand[offset+4],UINT#4);
48          AryMove(E001_Input_Data_03[0] ,rcvCommand[offset+8],UINT#4);
49          AryMove(E001_Input_Data_04[0] ,rcvCommand[offset+12],UINT#4);
50          AryMove(E001_Input_Data_05[0] ,rcvCommand[offset+16],UINT#4);
51          IF(E001_Receive_Data_Flag=TRUE )THEN
52             offset:=offset+20;
53             state:=30;
54          ELSE
55             rcvString:=AryToString(rcvCommand[0] ,offset+E001_Input_Data_Length);
56             state:=60;
57          END_IF;
58       ELSE
59          state:=50;
60       END_IF;
61    50: //Error occurs.
62        ;
63    60: //Normal end
64        ;
65  END_CASE;
```

①Initial processing
Set the CompoWay/F command.
Copy data to *sendCommand*.
Calculate next Input SID.
Initialize *Offset*.
Go to step ②.

②Send CompoWay/F command.
Calculate next Output SID.
Set the Output SID.
If intermediate send data:
Set Output Data Type.
Set Output Data Length.
and calculate the remaining send size.
If final send data:
Set Output Data Type.
Set Output Data Length.

Set Output Data.

Increment *Offset* by 20 bytes.
Go to step ③.

③Confirm that the command was sent for send processing.
If the Output SID Response is updated:
If final send data,
initialize *Offset*.
Go to step ④.
If intermediate send data,
Go to step ②.

④Check CompoWay/F response reception.
If the Input SID is updated,
go to step ⑤.

⑤Receive CompoWay/F response.
Update the Input SID Response.
Calculate next Input SID.
If a Receive Data Error occurs,
go to step ⑥.

If Input Data is received

Read the Input Data.

If intermediate receive data,
increment *Offset* by 20 bytes.
Go to step ④.
If final receive data,
store the Input Data in *rcvString*.
Go to step ⑦.

If the received data is not the final data,
go to step ⑥.

⑥Processing after error.

⑦Processing after normal end.

A-2-3 Restart Port

This programming sample restarts a port.

Restarting a port is used to clear the send and receive buffers and the Input SID of EJ1 EtherCAT Slave Unit when resuming operation of the EtherCAT master controller after it has been paused during a CompoWay/F command send or receive operation.

```
CASE state OF
  0: // Initialization
    state:=10;
  10: //Request control command execution.
    outSid:=nextSid(E001_Output_SID_Response);
    E001_Output_SID:=outSid;
    E001_Output_Data_Type:=16#0101; //PORT RESTART
    E001_Output_Data_Length:=0;
    state:=20;
  20: //Confirm that the control command execution request was sent.
    IF(E001_Output_SID_Response=outSid)THEN
      state:=30;
    END_IF;
  30: //Confirm that the the port was restarted.
    IF((E001_Input_SID=0)AND(E001_Input_SID_Response<>0))THEN
      E001_Input_SID_Response:=0;
      state:=40;
    END_IF;
    IF((E001_Input_SID=1)AND(E001_Input_SID_Response=0))THEN
      state:=40;
    END_IF;
  40: //Receive the control command response.
    IF((E001_Input_SID=1) AND (E001_Input_Data_Type=16#0201))THEN
      E001_Input_SID_Response:=1;
      state:=50;
    END_IF;
  50: //Normal end
     ;
END_CASE;
```

①Initial processing

②Request execution of the control command.
Calculate next Output SID.
Set the Output SID.
Set the Restart Port command.
Set Output Data Length.
Go to step ③.

③Confirm that the control command execution request was sent.
If the Output SID Response is updated, go to step ④.

④Confirm that the Restart Port command was executed.
If the Input SID is 0 when the Input SID Response is not 0, update the Input SID Response.
Go to step ⑤.

If the Input SID is 1 when the Input SID Response is 0, go to step ⑤.

⑤Receive the Control Command Response.
When the Control Command Response is received, update the Input SID Response.
go to step ⑥.

⑥Processing after normal end.

A-3 CoE Objects

This section describes the CoE objects that are implemented by the EJ1 EtherCAT Slave Unit.

A-3-1 Object Dictionary Area

CAN application protocol over EtherCAT (CoE) is based on the object dictionary for the CAN application protocol. All objects are assigned 4-digit hexadecimal indexes. The objects are structured in the following areas.

Index	Area	Description
0000 hex to 0FFF hex	Data Type Area	This area contains the data type definitions.
1000 hex to 1FFF hex	CoE Communications Area	The objects in this area are defined for use by all servers that perform specialized communications. • PDO mapping objects
2000 hex to 2FFF hex	Manufacturer-specific Area 1	The objects in this area are defined for all OMRON products.
3000 hex to 5FFF hex	Manufacturer-specific Area 2	The objects in this area are defined for the EJ1 EtherCAT Slave Unit.
6000 hex to 9FFF hex	Device Profile Area	The objects in this area are defined by the CiA401 Generic I/O Module Device Profile (a profile that specifies the CAN application protocol interface for devices with digital I/O and analog I/O).
A000 hex to FFFF hex	Reserved area	This area is reserved for future use.

A-3-2 Data type

The following data types are used in this profile.

Data type	Size	Range of values
Unsigned8	1 byte	0 to 255
Unsigned16	2 bytes	0 to 65,535
Unsigned32	4 bytes	0 to 4,294,967,295
Visible String	---	---

A-3-3 Communications Objects

The following table lists the communications objects for the EJ1 EtherCAT Slave Unit.

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
1000 hex	---	Device Type	RO	UNSIGNED32	---	00000000 hex	N/A	N/A
1008 hex	---	Device name	RO	VISIBLE STRING	---	“EJ1N-HFU-CECT” padded with 7 spaces character 20 hex	N/A	N/A
1009 hex	---	Hardware Version	RO	VISIBLE STRING	---	20 spaces (character 20 hex)	N/A	N/A
100A hex	---	Software Version	RO	VISIBLE STRING	---	“V1.00” padded with 15 spaces character 20 hex	N/A	N/A
1018 hex	---	ID Information	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	---	04 hex	N/A	---
	01	Vendor ID	RO	UNSIGNED32	---	00000083 hex	N/A	---
	02	Product Code	RO	UNSIGNED32	---	000000C0 hex	N/A	---
	03	Revision	RO	UNSIGNED32	---	00010000 hex	N/A	---
	04	Serial Number	RO	UNSIGNED32	---	Set to a unique value for each Unit by default.	N/A	---
10E0 hex	---	Reload Node Address	---	---	---	---	---	N/A
	00	Number of Entries	RO	UNSIGNED8	03 hex	03 hex	N/A	---
	01	Configured Station Alias Register Value	RW	UNSIGNED16	0000 hex to FFFF hex	0000 hex	N/A	---
	03	ID-Selector Validation	RW	UNSIGNED16	0000 hex to FFFF hex	0000 hex	N/A	---

A-3-4 PDO Mapping Objects

The following table lists the PDO mapping objects for the EJ1 EtherCAT Slave Unit.

● PDO Mapping Objects: Output Notification Data

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
1700 hex	---	257th receive PDO Mapping	---	---	---	---	---	Possible.
	00	Number of Entries	RW	UNSIGNED8	00 hex to 19 hex	0A hex	N/A	---
	01	Output SID	RW	UNSIGNED32	00000000 hex, 70000108 hex, 70000208 hex, 70000310 hex, 70000410 hex, 70000510 hex, 70000620 hex, 70000720 hex, 70000820 hex, 70000920 hex, 70000A20 hex, 70000B20 hex, 70000C20 hex, 70000D20 hex, 70000E20 hex, 70000F20 hex, 70001020 hex, 70001120 hex, 70001220 hex, 70001320 hex, 70001420 hex, 70001520 hex, 70001620 hex, 70001720 hex, 70001820 hex, or 70001920 hex	70000108 hex	N/A	---
	02	Input SID Response	RW	UNSIGNED32		70000208 hex	N/A	---
	03	Output Data Type	RW	UNSIGNED32		70000310 hex	N/A	---
	04	Output Sub Info	RW	UNSIGNED32		70000410 hex	N/A	---
	05	Output Data Length	RW	UNSIGNED32		70000510 hex	N/A	---
	06	Output Data 01	RW	UNSIGNED32		70000620 hex	N/A	---
	07	Output Data 02	RW	UNSIGNED32		70000720 hex	N/A	---
	08	Output Data 03	RW	UNSIGNED32		70000820 hex	N/A	---
	09	Output Data 04	RW	UNSIGNED32		70000920 hex	N/A	---
	0A	Output Data 05	RW	UNSIGNED32		70000A20 hex	N/A	---
	0B	Output Data 06	RW	UNSIGNED32		00000000 hex	N/A	---
	0C	Output Data 07	RW	UNSIGNED32		00000000 hex	N/A	---
	0D	Output Data 08	RW	UNSIGNED32		00000000 hex	N/A	---
	0E	Output Data 09	RW	UNSIGNED32		00000000 hex	N/A	---
	0F	Output Data 10	RW	UNSIGNED32		00000000 hex	N/A	---

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
1700 hex	10	Output Data 11	RW	UNSIGNED32	00000000 hex, 70000108 hex, 70000208 hex, 70000310 hex, 70000410 hex, 70000510 hex, 70000620 hex, 70000720 hex, 70000820 hex, 70000920 hex, 70000A20 hex, 70000B20 hex, 70000C20 hex, 70000D20 hex, 70000E20 hex, 70000F20 hex, 70001020 hex, 70001120 hex, 70001220 hex, 70001320 hex, 70001420 hex, 70001520 hex, 70001620 hex, 70001720 hex, 70001820 hex, or 70001920Hex	00000000 hex	N/A	---
	11	Output Data 12	RW	UNSIGNED32		00000000 hex	N/A	---
	12	Output Data 13	RW	UNSIGNED32		00000000 hex	N/A	---
	13	Output Data 14	RW	UNSIGNED32		00000000 hex	N/A	---
	14	Output Data 15	RW	UNSIGNED32		00000000 hex	N/A	---
	15	Output Data 16	RW	UNSIGNED32		00000000 hex	N/A	---
	16	Output Data 17	RW	UNSIGNED32		00000000 hex	N/A	---
	17	Output Data 18	RW	UNSIGNED32		00000000 hex	N/A	---
	18	Output Data 19	RW	UNSIGNED32		00000000 hex	N/A	---
	19	Output Data 20	RW	UNSIGNED32		00000000 hex	N/A	---

Assign data from objects in the 7000 hex range for PDO mapping of Output Notification Data.

● PDO Mapping Objects: Input Notification Data

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
1B00 hex	---	257th transmit PDO mapping	---	---	---	---	---	Possible.
	00	Number of Entries	RW	UNSIGNED8	00 hex to 1A hex	0B hex	N/A	---
	01	Port Status	RW	UNSIGNED32	00000000 hex, 60000110 hex, 60000208 hex, 60000308 hex, 60000410 hex, 60000510 hex, 60000610 hex, 60000720 hex, 60000820 hex, 60000920 hex, 60000A20 hex, 60000B20 hex, 60000C20 hex, 60000D20 hex, 60000E20 hex, 60000F20 hex, 60001020 hex, 60001120 hex, 60001220 hex, 60001320 hex, 60001420 hex, 60001520 hex, 60001620 hex, 60001720 hex, 60001820 hex 60001920 hex, or 60001A20 hex	60000110 hex	N/A	---
	02	Input SID	RW	UNSIGNED32		60000208 hex	N/A	---
	03	Output SID Response	RW	UNSIGNED32		60000308 hex	N/A	---
	04	Input Data Type	RW	UNSIGNED32		60000410 hex	N/A	---
	05	Input Sub Info	RW	UNSIGNED32		60000510 hex	N/A	---
	06	Input Data Length	RW	UNSIGNED32		60000610 hex	N/A	---
	07	Input Data 01	RW	UNSIGNED32		60000720 hex	N/A	---
	08	Input Data 02	RW	UNSIGNED32		60000820 hex	N/A	---
	09	Input Data 03	RW	UNSIGNED32		60000920 hex	N/A	---
	0A	Input Data 04	RW	UNSIGNED32		60000A20 hex	N/A	---
	0B	Input Data 05	RW	UNSIGNED32		60000B20 hex	N/A	---
	0C	Input Data 06	RW	UNSIGNED32		00000000 hex	N/A	---
	0D	Input Data 07	RW	UNSIGNED32		00000000 hex	N/A	---
	0E	Input Data 08	RW	UNSIGNED32		00000000 hex	N/A	---
	0F	Input Data 09	RW	UNSIGNED32		00000000 hex	N/A	---
	10	Input Data 10	RW	UNSIGNED32		00000000 hex	N/A	---
	11	Input Data 11	RW	UNSIGNED32		00000000 hex	N/A	---
	12	Input Data 12	RW	UNSIGNED32		00000000 hex	N/A	---
	13	Input Data 13	RW	UNSIGNED32		00000000 hex	N/A	---
	14	Input Data 14	RW	UNSIGNED32		00000000 hex	N/A	---
	15	Input Data 15	RW	UNSIGNED32		00000000 hex	N/A	---
	16	Input Data 16	RW	UNSIGNED32		00000000 hex	N/A	---
	17	Input Data 17	RW	UNSIGNED32		00000000 hex	N/A	---
	18	Input Data 18	RW	UNSIGNED32		00000000 hex	N/A	---
	19	Input Data 19	RW	UNSIGNED32		00000000 hex	N/A	---
	1A	Input Data 20	RW	UNSIGNED32		00000000 hex	N/A	---
1BFF hex	---	512th transmit PDO mapping	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	01 hex	01 hex	N/A	---
	01	Sysmac Error Status:	RO	UNSIGNED32	20020108 hex	20020108 hex	N/A	---

Assign data from objects in the 6000 hex range for PDO mapping of Input Notification Data.

A-3-5 Sync Manager Communications Objects

The EtherCAT communications memory is set with objects from 1C00 to 1C13 hex.

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
1C00 hex	---	Sync Manager Communication Type	---	---	---	---	---	Possible.
	00	SM Channel Number	RO	UNSIGNED8	04 hex	04 hex	N/A	---
	01	Communications Type SM0	RO	UNSIGNED8	01 hex	01 hex	N/A	---
	02	Communications Type SM1	RO	UNSIGNED8	02 hex	02 hex	N/A	---
	03	Communications Type SM2	RO	UNSIGNED8	03 hex	03 hex	N/A	---
	04	Communications Type SM3	RO	UNSIGNED8	04 hex	04 hex	N/A	---
1C10 hex	---	Sync Manager 0 PDO Assignment	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	00 hex	00 hex	N/A	---
1C11 hex	---	Sync Manager 1 PDO Assignment	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	00 hex	00 hex	N/A	---
1C12 hex	---	Sync Manager 2 PDO Assignment	---	---	---	---	---	Possible.
	00	Number of Entries	RW	UNSIGNED8	00 hex to 01 hex	01 hex	N/A	---
	01	Receive PDO Mapping Assignment 1	RW	UNSIGNED16	0000 hex or 1700 hex	1700 hex	N/A	---
1C13 hex	---	Sync Manager 3 PDO Assignment	---	---	---	---	---	Possible.
	00	Number of Entries	RW	UNSIGNED8	00 hex to 02 hex	02 hex	N/A	---
	01	Send PDO Mapping Assignment 1	RW	UNSIGNED16	0000 hex 1B00 hex or 1BFF hex	1B00 hex	N/A	---
	02	Send PDO Mapping Assignment 2	RW	UNSIGNED16	0000 hex 1B00 hex or 1BFF hex	1BFF hex	N/A	---

A-3-6 Manufacturer-specific Object 1

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
2002 hex	---	Sysmac Error	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	02 hex	02 hex	N/A	---
	01	Sysmac Error Status *1	RO	UNSIGNED8	00 hex to FF hex	00 hex	T	---
	02	Sysmac Error Status Clear *2	RW	UNSIGNED8	00 hex or 01 hex	00 hex	N/A	---

*1. The assignments of bits in the Sysmac error status at subindex 01 hex are listed below.
The applicable bit is 0 (FALSE) if no error exists, or 1 (TRUE) if an error exists.
Bits 6 to 15: Reserved
Bit 5: Minor fault
Bit 4: Observation
Bits 2 to 3: Reserved
Bit 1: Reserved
Bit 0: Reserved

*2. Subindex 02 hex is used to clear the Sysmac Error Status.
Write 01 hex to clear the Sysmac Error Status. If you write a value other than 01 hex, the command is invalid and the abort code is returned.
When a read is performed, 00 hex is given.

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
2003 hex	---	Sysmac Observation	---	---	---	---	---	Possible.
	00	Number of Observations	RO	UNSIGNED8	00 hex to 05 hex	00 hex	N/A	---
	01	Observation 1	RO	OCTET_STRING	---	---	N/A	---
	02	Observation 2	RO	OCTET_STRING	---	---	N/A	---
	03	Observation 3	RO	OCTET_STRING	---	---	N/A	---
	04	Observation 4	RO	OCTET_STRING	---	---	N/A	---
	05	Observation 5	RO	OCTET_STRING	---	---	N/A	---

Note 1. Observations are stored in the order that they occur from subindexes 01 to 05 hex.
2. The logs are cleared when 1 (TRUE) is written to Sysmac Error Status Clear (02 hex) in the Sysmac Error (2002 hex).
3. The following table gives the format of each log.

Item	Data type	Details
Error code	UNSIGNED32	Event code (stored in little endian)
Reserved area	UNSIGNED32	---
Reserved area	UNSIGNED32	---

Refer to *Event Code List* on page 7-7 under *Section 7 Troubleshooting* for details on event codes.

Index (hex)	Sub-index (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
2004 hex	---	Sysmac Minor Fault	---	---	---	---	---	Possible.
	00	Number of Minor Faults	RO	UNSIGNED8	00 hex to 05 hex	00 hex	N/A	---
	01	Minor Fault 1	RO	OCTET_STRING	---	---	N/A	---
	02	Minor Fault 2	RO	OCTET_STRING	---	---	N/A	---
	03	Minor Fault 3	RO	OCTET_STRING	---	---	N/A	---
	04	Minor Fault 4	RO	OCTET_STRING	---	---	N/A	---
	05	Minor Fault 5	RO	OCTET_STRING	---	---	N/A	---

Note 1. Minor faults are stored in the order that they occur from subindexes 01 to 05 hex.

2. The logs are cleared when 1 (TRUE) is written to Sysmac Error Status Clear (02 hex) in the Sysmac Error (2002 hex).

3. The following table gives the format of each log.

Item	Data type	Details
Error code	UNSIGNED32	Event code (stored in little endian)
Reserved area	UNSIGNED32	---
Reserved area	UNSIGNED32	---

Refer to *Event Code List* on page 7-7 under *Section 7 Troubleshooting* for details on event codes.

A-3-7 Device Profile Area

The following objects are available for I/O assignment in the device profile area of the EJ1N-HFUC-ECT. Refer to *5-4 PDO Mapping Data Details* on page 5-8 for information on individual objects.

Index (hex)	Subindex (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
6000 hex	---	Input Data	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	1A hex	1A hex	N/A	---
	01	Port Status	RO	UNSIGNED16	---	0000 hex	T	---
	02	Input SID	RO	UNSIGNED8		00 hex	T	---
	03	Output SID Response	RO	UNSIGNED8		00 hex	T	---
	04	Input Data Type	RO	UNSIGNED16		0000 hex	T	---
	05	Input Sub Info	RO	UNSIGNED16		0000 hex	T	---
	06	Input Data Length [bytes]	RO	UNSIGNED16		0000 hex	T	---
	07–1A	Input Data 01 to 20	RO	ARRAY [0..3] OF BYTE		0000 0000 hex	T	---

Index (hex)	Subindex (hex)	Object name	Access RO: Read only RW: Read/write	Data Type	Data range	Initial value	PDO mapping N/A: Not mappable R: RxPDO mappable T: TxPDO mappable	Complete access
7000 hex	---	Output Data	---	---	---	---	---	Possible.
	00	Number of Entries	RO	UNSIGNED8	19 hex	19 hex	N/A	---
	01	Output SID	RW	UNSIGNED8	---	00 hex	R	---
	02	Input SID Response	RW	UNSIGNED8		00 hex	R	---
	03	Output Data Type	RW	UNSIGNED16		0000 hex	R	---
	04	Output Sub Info	RW	UNSIGNED16		0000 hex	R	---
	05	Output Data Length [bytes]	RW	UNSIGNED16		0000 hex	R	---
	06 to 19	Output Data 01 to 20	RW	ARRAY [0..3] OF BYTE		0000 0000 hex	R	---

I

Index

Index

A

AL status 7-9
Assigning device variables 5-32

B

Basic Unit baud rate setting 2-4
Basic Units 1-4
BCC Error Flag 5-13

C

CoE objects A-9
Command code of the control command 5-9
CompoWay/F command frame 5-25
CompoWay/F commands 1-5
CompoWay/F commands and responses 5-3
CompoWay/F response frame 5-25
Configuration tool 3-3
control command code list 5-10
control command flag 5-9
control command response 5-13
Control Command Response Flag 5-13
control commands 1-5
Control commands and responses 5-5
CX-Thermo 1-4

D

determining the output data and input data PDO mapping sizes 5-31

E

editing PDO map settings 5-28
EJ1 Bus Communications Error Flag 5-13
emergency messages 7-8
End Unit (EDU) 1-4
error list 7-3
error notification methods 7-2
error notifications based on the Sysmac Error Status 7-6
ESI file 3-6
event code 7-7

F

Final Data Reception Flag 5-12, 5-13
final send data 5-9

G

G3ZA 1-4

I

Init 3-7
input data 5-11
input data length 5-11
input data type 5-11, 5-13
input notification data 5-7, 5-10
input SID 5-10
input SID response 5-8
input sub info 5-11
Intermediate Data Reception Flag 5-13
Intermediate send data 5-9

N

node address setting 3-4

O

Op 3-7
Operational 3-7
output data 5-9
output data and input data contents 5-25
Output data type 5-8, 5-9
output notification data 5-7, 5-8
output SID 5-8
output SID response 5-11

P

PDO 3-8
PDO communications 1-5, 5-2
PDO mapping data overview 5-6
PDO mappings 3-8
PDO size 5-28
Port status 5-12
Pre-Op 3-7
Pre-Operational 3-7
procedure for sending a CompoWay/F command and receiving a CompoWay/F response 5-16, 5-18
Procedures for sending control commands and receiving responses 5-17

R

Receive Data Error Flag 5-14
Restart Port 5-2, 5-10, A-8

S

Safe-Op 3-7
Safe-Operational 3-7
SDO 3-10
SDO communications 1-5, 6-2

Send Buffer Full Flag 5-12
Send Completed Toggle Bit 5-12
Send Data Exists Flag 5-12
setting the PDO mappings for I/O data 5-28
storing output data and input data in array variables 5-27
Sysmac Error status 5-7, 5-14, 7-6

W

with SDO communications 6-1

OMRON Corporation **Industrial Automation Company**
Kyoto, JAPAN
Contact: www.ia.omron.com

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

OMRON 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 ASIA PACIFIC PTE. LTD.
No. 438A Alexandra Road # 05-05/08 (Lobby 2),
Alexandra Technopark,
Singapore 119967
Tel: (65) 6835-3011/Fax: (65) 6835-2711

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 2015 All Rights Reserved.
In the interest of product improvement,
specifications are subject to change without notice.

Cat. No. H192-E1-01 1015