Modicon M580 BMENOC0302(H)

High Performance Ethernet Communication Module

Installation and Configuration Guide

Original instructions

NNZ44174.01
08/2025

www.se.com

Schneider Electric

Legal Information

The information provided in this document contains general descriptions, technical characteristics and/or recommendations related to products/solutions.

This document is not intended as a substitute for a detailed study or operational and site-specific development or schematic plan. It is not to be used for determining suitability or reliability of the products/solutions for specific user applications. It is the duty of any such user to perform or have any professional expert of its choice (integrator, specifier or the like) perform the appropriate and comprehensive risk analysis, evaluation and testing of the products/solutions with respect to the relevant specific application or use thereof.

The Schneider Electric brand and any trademarks of Schneider Electric SE and its subsidiaries referred to in this document are the property of Schneider Electric SE or its subsidiaries. All other brands may be trademarks of their respective owner.

Schneider Electric reserves the right to make changes or updates with respect to or in the content of this document or the format thereof, at any time without notice.

To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this document, as well as any non-intended use or misuse of the content thereof.

Table of Contents

Safety Information

Important Information

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

⚡ The addition of this symbol to a “Danger” or “Warning” safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.

⚠ This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result** in minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

Please Note

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠WARNING

UNGUARDED EQUIPMENT

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

> **NOTE:** Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

Start-up and Test

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check are made and that enough time is allowed to perform complete and satisfactory testing.

⚠WARNING

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.

- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

Operation and Adjustments

The following precautions are from the NEMA Standards Publication ICS 7.1-1995:

(In case of divergence or contradiction between any translation and the English original, the original text in the English language will prevail.)

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer’s instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer’s instructions and the machinery used with the electrical equipment.
- Only those operational adjustments required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

About the Book

Document Scope

This guide describes the BMENOC0302 and BMENOC0302H Ethernet communications modules and their roles in a Modicon M580 system.

The BMENOC0302(H) module is the communications interface between the M580 controller and other Ethernet network devices through the EtherNet/IP or Modbus TCP communication protocols.

The BMENOC0302(H) is a high-performance module that facilitates the advanced migration of Quantum and Premium Ethernet offers to the M580 offer.

NOTE: The specific configuration settings contained in this guide are intended to be used for instructional purposes only. The settings for your specific configuration may differ from the examples presented in this guide.

Validity Note

This document has been created for the M580 system when used with EcoStruxure™ Control Expert 16.2.

The characteristics of the products described in this document are intended to match the characteristics that are available on www.se.com. As part of our corporate strategy for constant improvement, we may revise the content over time to enhance clarity and accuracy. If you see a difference between the characteristics in this document and the characteristics on www.se.com, consider www.se.com to contain the latest information.

NOTE: After clicking one of the above download links, you may need to select your country before you can download the document.

General Cybersecurity Information

In recent years, the growing number of networked machines and production plants has seen a corresponding increase in the potential for cyber threats, such as unauthorized access, data breaches, and operational disruptions. You must, therefore, consider all possible cybersecurity measures to help protect assets and systems against such threats.

To help keep your Schneider Electric products secure and protected, it is in your best interest to implement the cybersecurity best practices as described in the Cybersecurity Best Practices document.

Schneider Electric provides additional information and assistance:

- Subscribe to the Schneider Electric security newsletter.
- Visit the Cybersecurity Support Portal web page to:
  - Find Security Notifications.
  - Report vulnerabilities and incidents.
- Visit the Schneider Electric Cybersecurity and Data Protection Posture web page to:
  - Access the cybersecurity posture.
  - Learn more about cybersecurity in the cybersecurity academy.
  - Explore the cybersecurity services from Schneider Electric.

Available Languages of the Document

The information contained herein is available in these languages:

- English (NNZ44174)

Related Documents

Title of documentation	Reference number
Modicon M580, Frequently Used Architectures, System Guide	HRB62666 (ENG) HRB65318 (FRE) HRB65319 (GER) HRB65320 (ITA) HRB65321 (SPA) HRB65322 (CHS)
Modicon M580, Complex Topologies, System Guide	NHA58892 (ENG) NHA58893 (FRE) NHA58894 (GER) NHA58895 (ITA) NHA58896 (SPA) NHA58897 (CHS)
Modicon M580 Hot Standby, Frequently Used Architectures, System Guide	NHA58880 (ENG) NHA58881 (FRE) NHA58882 (GER) NHA58883 (ITA) NHA58884 (SPA) NHA58885 (CHS)
Modicon M580, Hardware, Reference Manual	EIO0000001578 (ENG) EIO0000001579 (FRE) EIO0000001580 (GER) EIO0000001581 (SPA) EIO0000001582 (ITA) EIO0000001583 (CHS)
Modicon M580, M340, and X80 I/O Platforms, Standards and Certifications	EIO0000002726 (ENG) EIO0000002727 (FRE) EIO0000002728 (GER) EIO0000002730 (ITA) EIO0000002729 (SPA) EIO0000002731 (CHS)
Modicon M580, Change Configuration on the Fly, User Guide	EIO0000001590 (ENG) EIO0000001591 (FRE) EIO0000001592 (GER) EIO0000001594 (ITA) EIO0000001593 (SPA) EIO0000001595 (CHS)
M580 BMENOS0300, Network Option Switch, Installation and Configuration Guide	NHA89117 (English) NHA89119 (French) NHA89120 (German) NHA89121 (Italian) NHA89122 (Spanish) NHA89123 (Chinese)

Title of documentation	**Reference number**
Modicon X80, BMXNRP0200/0201 Fiber Converter Modules, User Guide	EIO0000001108 (ENG) EIO0000001113 (CHS) EIO0000001109 (FRE) EIO0000001110 (GER) EIO0000001112 (ITA) EIO0000001111 (SPA)
Modicon eX80, BMEAHI0812 HART Analog Input Module & BMEAHO0412 HART Analog Output Module, User Guide	EAV16400 (ENG) EAV28404 (FRE) EAV28384 (GER) EAV28413 (ITA) EAV28360 (SPA) EAV28417 (CHS)
Modicon X80, Analog Input/Output Modules, User Manual	35011978 (ENG) 35011980 (FRE) 35011979 (GER) 35011982 (ITA) 35011981 (SPA) 35011983 (CHS)
Modicon X80, Discrete Input/Output Modules, User Manual	35012474 (ENG) 35012476 (FRE) 35012475 (GER) 35012478 (ITA) 35012477 (SPA) 35012479 (CHS)
Modicon X80, BMXEHC0200 Counting Module, User Manual	35013355 (ENG) 35013357 (FRE) 35013356 (GER) 35013359 (ITA) 35013358 (SPA) 35013360 (CHS)
Electrical installation guide	EIGED306001EN (English)
Control Panel Technical Guide, How to protect a machine from malfunctions due to electromagnetic disturbance	CPTG003_EN (English) CPTG003_FR (French)
EcoStruxure™ Control Expert, Program Languages and Structure, Reference Manual	35006144 (ENG) 35006145 (FRE) 35006146 (GER) 35013361 (ITA) 35006147 (SPA) 35013362 (CHS)
EcoStruxure™ Control Expert, Operating Modes	33003101 (ENG) 33003102 (FRE) 33003103 (GER) 33003104 (SPA) 33003696 (ITA) 33003697 (CHS)

Title of documentation	Reference number
EcoStruxure™ Control Expert, Installation Manual	35014793 (ENG) 35014792 (FRE) 35014794 (GER) 35014795 (SPA) 35014796 (ITA) 35012191 (CHS)
Modicon Controllers Platform Cyber Security, Reference Manual	EIO0000001999 (ENG) EIO0000002004 (CHS) EIO0000002001 (FRE) EIO0000002000 (GER) EIO0000002002 (ITA) EIO0000002003 (SPA)
EcoStruxure™ Automation Device Maintenance, User Guide	EIO0000004033 (ENG) EIO0000004048 (FRE) EIO0000004046 (GER) EIO0000004049 (ITA) EIO0000004047 (SPA) EIO0000004050 (CHS)
Modicon X80 Racks and Power Supplies, Hardware, Reference Manual	EIO0000002626 (ENG) EIO0000002631 (CHS) EIO0000002627 (FRE) EIO0000002628 (GER) EIO0000002630 (ITA) EIO0000002629 (SPA)
EcoStruxure™ Control Expert, Communication, Block Library	33002527 (ENG) 33002528 (FRE) 33002529 (GER) 33003682 (ITA) 33002530 (SPA) 33003683 (CHS)
Modicon M340, BMXNOC0401 Ethernet Communication Module, User Manual	S1A34009 (ENG) S1A34010 (FRE) S1A34011 (GER) S1A34013 (ITA) S1A34012 (SPA) S1A34014 (CHS).
Premium using EcoStruxure™ Control Expert, Hot Standby, User Manual	35012068 (ENG) 35012070 (FRE) 35012069 (GER) 35012072 (ITA) 35012071 (SPA) 35012073 (CHS)

Title of documentation	Reference number
Quantum EIO, Control Network, Installation and Configuration Guide	S1A48993 (ENG) S1A48994 (FRE) S1A48995 (GER) S1A48997 (ITA) S1A48998 (SPA) S1A48999 (CHS)
EcoStruxure™ Control Expert, System Bits and Words, Reference Manual	EIO0000002135 (ENG) EIO0000002136 (FRE) EIO0000002137 (GER) EIO0000002139 (SPA) EIO0000002138 (ITA) EIO0000002140 (CHS)

To find documents online, visit the Schneider Electric download center (www.se.com/ww/en/download/).

Product Related Information

⚠⚠DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION, OR ARC FLASH

- Disconnect all power from all equipment, including connected devices, prior to removing any covers or doors or installing or removing any accessories, hardware, cables, or wires except under the specific conditions specified in the appropriate hardware guide for this equipment.
- Always use a properly rated voltage-sensing device to confirm the power is off where and when indicated.
- Replace and secure all covers, accessories, hardware, cables, and wires and confirm that a proper ground connection exists before applying power to the equipment.
- Use only the specified voltage when operating the equipment and any associated products.

Failure to follow these instructions will result in death or serious injury.

⚠WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.[1]
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

The examples given herein are for information only.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

Adapt examples given herein to the specific functions and requirements of your industrial application before you implement them.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

Follow all local and national safety codes and standards.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Terminology Derived from Standards

The technical terms, terminology, symbols and the corresponding descriptions in the information contained herein, or that appear in or on the products themselves, are generally derived from the terms or definitions of international standards.

In the area of functional safety systems, drives and general automation, this may include, but is not limited to, terms such as *safety*, *safety function*, *safe state*, *fault*, *fault reset*, *malfunction*, *failure*, *error*, *error message*, *dangerous*, etc.

Among others, these standards include:

Standard	Description
IEC 61131-2:2007	Programmable controllers, part 2: Equipment requirements and tests.
ISO 13849-1:2023	Safety of machinery: Safety related parts of control systems. General principles for design.
EN 61496-1:2020	Safety of machinery: Electro-sensitive protective equipment. Part 1: General requirements and tests.
ISO 12100:2010	Safety of machinery - General principles for design - Risk assessment and risk reduction
EN 60204-1:2006	Safety of machinery - Electrical equipment of machines - Part 1: General requirements

Standard	Description
ISO 14119:2013	Safety of machinery - Interlocking devices associated with guards - Principles for design and selection
ISO 13850:2015	Safety of machinery - Emergency stop - Principles for design
IEC 62061:2021	Safety of machinery - Functional safety of safety-related electrical, electronic, and electronic programmable control systems
IEC 61508-1:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: General requirements.
IEC 61508-2:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Requirements for electrical/electronic/programmable electronic safety-related systems.
IEC 61508-3:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Software requirements.
IEC 61784-3:2021	Industrial communication networks - Profiles - Part 3: Functional safety fieldbuses - General rules and profile definitions.
2006/42/EC	Machinery Directive
2014/30/EU	Electromagnetic Compatibility Directive
2014/35/EU	Low Voltage Directive

In addition, terms used in the present document may tangentially be used as they are derived from other standards such as:

Standard	Description
IEC 60034 series	Rotating electrical machines
IEC 61800 series	Adjustable speed electrical power drive systems
IEC 61158 series	Digital data communications for measurement and control – Fieldbus for use in industrial control systems

Finally, the term *zone of operation* may be used in conjunction with the description of specific hazards, and is defined as it is for a *hazard zone* or *danger zone* in the *Machinery Directive* (*2006/42/EC*) and *ISO 12100:2010*.

NOTE: The aforementioned standards may or may not apply to the specific products cited in the present documentation. For more information concerning the individual standards applicable to the products described herein, see the characteristics tables for those product references.

Characteristics of the BMENOC0302(H) Module

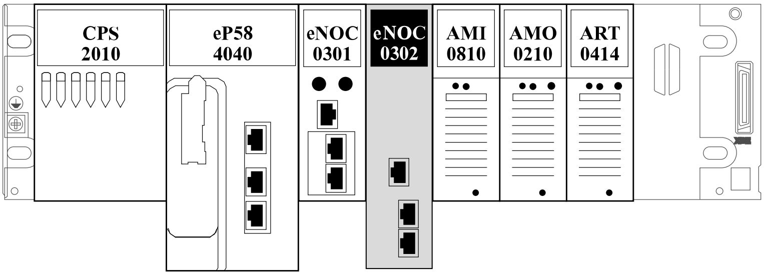
Introduction

Use the BMENOC0302(H) Ethernet communications module to enable distributed device communications in a Modicon M580 system.

Module Introduction

Introduction

The BMENOC0302(H) Ethernet communications module is installed on a local net rack in the Modicon M580 system:



The BMENOC0302(H) module is an advanced Ethernet communications module that provides access to devices that are connected to the controller module's device network ports when you enable the Ethernet rack, page 93.

NOTE: Do not mount the BMENOC0302(H) module on a BMX (X Bus–only) rack. The module powers up only on a BME (Ethernet) rack. (Refer to the rack and power descriptions in the *Modicon X80 Racks and Power Supplies, Hardware, Reference Manual*.)

Extreme Environments

The BMENOC0302H module is a ruggedized version of the standard BMENOC0302 equipment. It can be used in standard and harsh temperature environments.

These are the operating temperature ranges for the respective modules:

Module	Celsius (°C)	Fahrenheit (°F)
BMENOC0302	0 ... +60	32 ... 140
BMENOC0302H	-25 ... +70	-13 ... 158

These characteristics apply to use at altitudes up to 2000 m (6560 ft). When the module operates above 2000 m (6560 ft), apply additional derating of approximately 1 °C/400 m (33.8 °F/1,312 ft), isolation 150 V/1,000 m/3,281 ft. For accurate temperature derating calculation, refer to IEC 61131-2 Ed4.0 Annex A.

NOTE:

- To compare the performance and characteristics of the BMENOC0302(H) with the BMENOC0301 and BMENOC0311 Ethernet communications modules, refer to table that presents their respective features, page 35.
- For more information, refer to these topics in the *Modicon M580, M340, and X80 I/O Platforms, Standards and Certifications* guide:
  - *Installation in More Severe Environments*
  - *Operating and Storage Conditions*

Key Features

The BMENOC0302(H) advanced Ethernet communications module has these features and capabilities:

- The BMENOC0302(H) supports up to six Ethernet communication modules in a single local rack, depending on the M580 system type and the selected controller module. The number of supported BMENOC0302(H) modules depends on the selected controller module:

Controller	Supported BMENOC0302(H) Modules
BMEP581020	2
BMEP582020	2
BMEP582040	2
BMEP583020	3
BMEP583040	3
BMEP584020	4
BMEP584040	4
BMEP585040	6
BMEP586040	6

  NOTE: When your local rack includes different Ethernet communication modules (for example, a combination of BMENOC0302(H), BMENOC0301, BMENOC0311, or BMENOC0321 modules), verify that the power characteristics of each module and the selected controller module do not exceed your power budget. These resources provide more information:

  - Refer to the power characteristics for the BMENOC0302(H) module, page 37.
  - Refer to the *Modicon X80 Racks and Power Supplies, Hardware, Reference Manual*.

- You can configure up to six BMENOC0302(H) modules in an M580 local rack. (Refer to the *Modicon M580, Hardware, Reference Manual*.)
- You can configure a maximum 16KB input and 16KB output of implicit messages for the exchange of data between the BMENOC0302(H) module and the controller module.
- You can configure up to 32 messages for each controller MAST task cycle for explicit message (Port 502, EtherNet/IP Class3).
- The maximum packet rate for high-performance Modbus TCP and EtherNet/IP is 12,000 packets per second.
- The transmission rate for the two device ports (ETH 1, ETH 2) is 1 GB/sec.

- Cybersecurity features:
  - secure boot
  - firmware signing and integrity check
  - firmware upgrade through HTTPS
  - HTTPS-based web pages
  - IPsec/IKEv2 (initiator/responder)
  - SNMPv3
- To support the automatic hot swap of IP parameters, enable or disable **Enable IP A/B** in a Hot Standby configuration on the IP configuration screen in EcoStruxure Control Expert, page 157.
- The module is available in a harsh variant (BMENOC0302H).

When you convert an existing M580 project, replace the BMENOC0301 or BMENOC0311 modules with BMENOC0302(H) modules to capitalize on these advanced features.

Hot Swapping Considerations

The BMENOC0302(H) module is a hot swappable device that supports these IP management modes in an M580 Hot Standby system:

- *swappable:* The BMENOC0302(H) module in the new primary local rack uses the main IP address from the previous standby IP address (main IP address + *1*).
- *fixed:* The primary and secondary BMENOC0302(H) modules use fixed IP address (IP A, IP B), which does not change, even during the auto-swap of the controller.

Hot swapping is the ability to remove a module from its bus base and then to replace it with an identical module, while the M580 system is powered up, without disrupting the normal operations of the controller. When the electronic module is placed back to its bus base or replaced with another electronic module with the same reference, it starts to operate again.

⚠⚠DANGER

EXPLOSION OR ELECTRIC SHOCK

- Perform a hot swap operation only in locations known and confirmed to be non-hazardous.
- Replace an electronic module only with an identical reference.

Failure to follow these instructions will result in death or serious injury.

⚠WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.[1]
- Test each implementation of a system for proper operation before placing it into service.

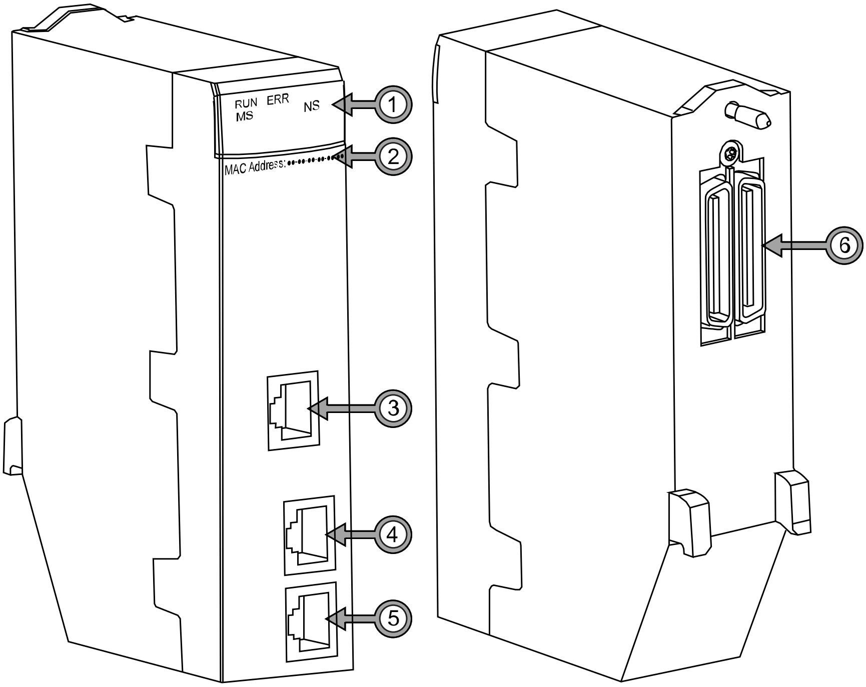
Failure to follow these instructions can result in death, serious injury, or equipment damage.

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Hardware Description

Physical Description

This illustration shows the external features of the BMENOC0302(H) Ethernet communications module:

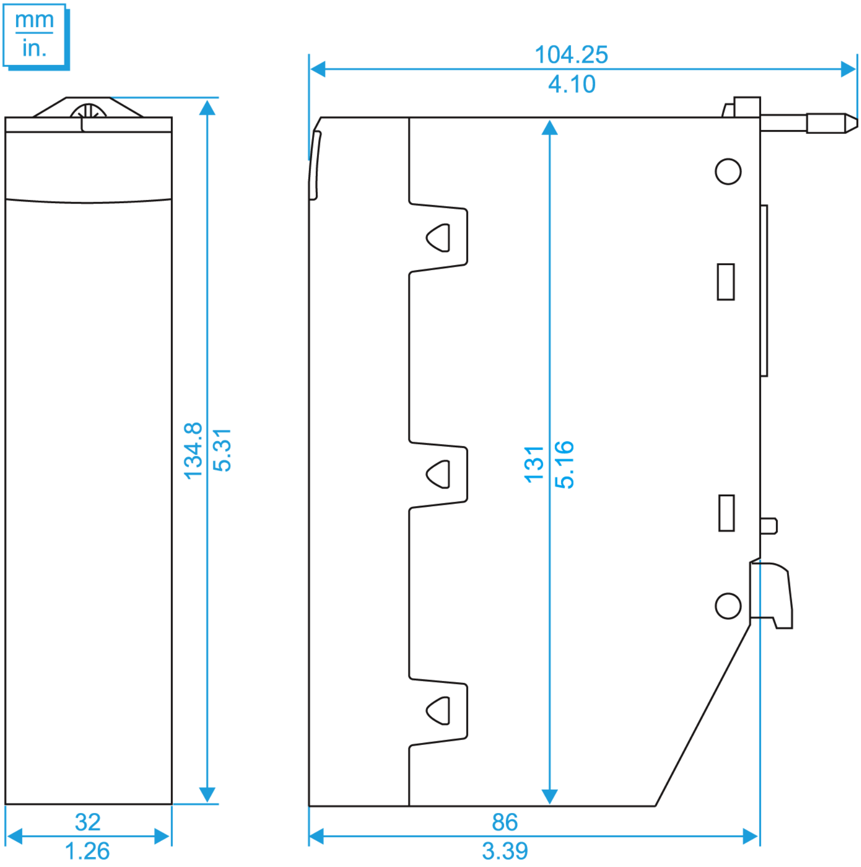


Legend:

Item	Description	Function
1	LED array	To diagnose the module, observe the LED display, page 253.
2	MAC address	This media access control (MAC) address corresponds to the Ethernet module hardware.
3	SERVICE port (ETH1)	Use the RJ45 Ethernet connector for a SERVICE port. **NOTE:** Refer to the service port configuration, page 114.
4	DEVICE NETWORK port (ETH2)	Two RJ45 DEVICE NETWORK ports (ETH 2 and ETH 3) facilitate these functions: • Theses ports support Ethernet communications (10/100/1000 Mbps). • Connections to these ports facilitate communications to distributed devices. • These ports provide cable redundancy through a daisy chain loop architecture when RSTP is enabled.
5	DEVICE NETWORK port (ETH3)	
6	rack connector	This connection to the Modicon M580 rack supports Ethernet communications.

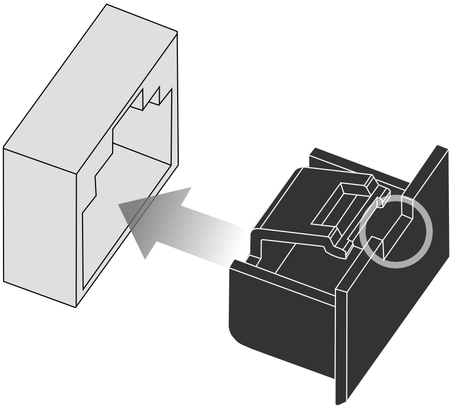
Dimensions

The BMENOC0302(H) module is the width of a single-slot M580 communications module but the same height as a controller module:



Dust Cover

To help keep dirt particles and moisture out of the Ethernet ports, cover unused ports with the dust cover:



Insert or remove the dust cover:

insert	1. Set the dust cover in the port with your hand. 2. Apply mild pressure with your fingers to push the dust cover into the port. 3. Stop when the dust cover snaps into place.
remove	1. When the dust cover is in the port, insert a small screwdriver in the slot (circled in the above image). 2. Gently push down on the locking mechanism to unlock the dust cover. 3. Gently pry the dust cover out of the port.

Key Module Features

Product Features

This list shows the key features of the BMENOC0302(H) Ethernet communications module:

- Server Services:
  - Port 502 server
  - EtherNet/IP messaging server
  - Address server
- Client Services:
  - I/O scanning
  - Client explicit messaging
- Redundancy:
  - Hot Standby
  - RSTP (enabled or disabled)
- Diagnostics:
  - DTM online diagnostic screens
  - Modicon M580 BMENOC0302(H) website
- Security:
  - HTTPS
  - Certificate-based authentication, IPsec IKEv2 (initiator/responder)
  - SNMPv1, SNMPv3
  - NTPv4 client
- Enhanced Hardware:
  - 1Gbits dual port (ETH 2, ETH 3)
  - Lower power consumption

Service Descriptions

Configure the BMENOC0302(H) module to provide these Ethernet services:

Modbus scanner	Use this service to exchange I/O data between the BMENOC0302(H) and Modbus TCP devices. The service supports Modbus function codes 3 (read), 16 (write), and 23 (read-write).
EtherNet/IP scanner	The BMENOC0302(H) module acts as a scanner and exchanges I/O data (embedded in assembly objects) with EtherNet/IP devices. This service supports communications between the BMENOC0302(H) module and distributed EtherNet/IP devices to repetitively exchange I/O data across the network.
I/O server	The BMENOC0302(H) module acts as an EtherNet/IP I/O server that exchanges data with EtherNet/IP scanners.
Modbus server	Use this service to access the Modbus controller-module server or the local Modbus server (for diagnostics data). Sample clients include these: • Modicon M580 BMENOC0302(H) website • SCADA Modbus • Modbus HMI • EcoStruxure Control Expert
EtherNet/IP adapter	Configure the BMENOC0302(H) as an EtherNet/IP adapter to access the local slaves for I/O controller-module data and local diagnostics data (through CIP diagnostic objects). Sample clients include these: • Modicon M580 BMENOC0302(H) website • SCADA over EtherNet/IP • HMI over EtherNet/IP • EcoStruxure Control Expert DTM **NOTE:** Refer to the description of the EtherNet/IP adapter, page 226.
EtherNet/IP Modbus translator	Use this service to access Modbus data with an EtherNet/IP client using standard CIP messaging.

NOTE: Refer to the communication specifications, page 50.

Redundancy:

RSTP	For all RSTP-enabled devices in the network, the RSTP service creates a loop-free logical network path for EtherNet/IP devices that are part of a topology that includes redundant physical paths. When the network experiences an interruption in service, the RSTP-enabled module automatically restores network communication by activating redundant links. Configure the RSTP service with the EcoStruxure Control Expert DTM to help protect against a single point of error in the network. Refer to the RSTP configuration instructions, page 109.

Quality of Service:

QoS	This service adds *Differentiated Services Code Point* (DSCP) tags to the IP header of Ethernet packets so that network infrastructure devices can prioritize the transmission and forwarding of Ethernet frames for specific services.

Diagnostics:

Controller Module Application	Some module diagnostics (I/O connection health, redundancy status, etc.) are available through the controller-module application and are updated every cycle.
Local Modbus Server, page 288	Some module diagnostics (I/O connection, extended health, redundancy status, FDR server, etc.) are available to Modbus clients that read the local Modbus server area with Modbus function code 3 when the unit ID is set to 100 or through Modbus function code 3, 8/21, 8/22, or 43/14.
CIP Objects, page 290	Some module diagnostics (Ethernet interface, redundancy, EtherNet/IP scanner, etc.) are available through CIP objects that EtherNet/IP devices such as SCADA or HMI can read.
SNMP, page 114	Some module diagnostics (IP parameters, redundancy, Ethernet port statistics, etc.) are available through the SNMP service (an SNMPv1 or SNMPv3 agent). Configure the SNMP service with the EcoStruxure Control Expert DTM to access diagnostic information for the BMENOC0302(H) module and event notification for some services (like a change in the Ethernet port link state). You can configure SNMP manager IP addresses (MIB browser, CNM, etc.) as trap (event) notification destinations. The standard MIB-II (including the bridge MIB) provides diagnostic information through the SNMP service.
Port Mirroring, page 114	Diagnose network issues by examining packets to and from Ethernet ports when the service port is configured for port mirroring.
Modicon M580 BMENOC0302(H) website, page 362	The embedded Modicon M580 BMENOC0302(H) website provides diagnostics data through a web browser.
Device DDT, page 257	View the device DDT diagnostics for the BMENOC0302(H) module.

Web Server:

Web Server, page 362	The web server in this module processes requests from web pages.

Firmware Upgrade:

Firmware Upgrade, page 360	Use the EcoStruxure™ Automation Device Maintenance (EADM) software tool to upgrade the module firmware.

Services and Addresses

This table shows the availability of network services in terms of the relationship between the ports in the BMENOC0302(H) module and its IP and MAC addresses:

Service	IP Address	MAC address
EtherNet/IP scanner	main IP	module MAC
Modbus client	main IP	module MAC
FDR server and DHCP	main IP	module MAC
other services (for example, web server, EtherNet/IP adapter, Modbus server/FTP)	main IP	module MAC
Syslog	main IP	module MAC
SNMPv3 source IP address	main IP	module MAC
NTPv4 client source IP address	main IP or IPA/IPB if enabled	module MAC
LLDP	main IP	port MAC = (module MAC + 1, 2, 3, or 4) (1)
RSTP	main IP	port MAC = (module MAC + 1, 2, or 3) (1)
(1)Ports: • **port 1:** module MAC + 1 (service port) • **port 2:** module MAC + 2 • **port 3:** module MAC + 3 • **port 4:** module MAC + 4 (Ethernet rack)		

Modicon M580 Ethernet Communications Module Evolution

Available Features

This tables shows the evolution of module functionality from the BMENOC0301 and BMENOC0311 Modicon M580 Ethernet communications modules to the BMENOC0302(H) Modicon M580 high-performance Ethernet communications module:

Feature	M580 Ethernet Communications Module	M580 High Performance Ethernet Module
commercial reference	BMENOC0301/BMENOC0301C BMENOC0311/BMENOC0311C	BMENOC0302, BMENOC0302H
power consumption (per module)	1.8A (3.3V) BMENOC0301 0.9mA (3.3V) for PV>=13 BMENOC0311 0.9mA (3.3V) for PV>=14	0.155A (24V)
maximum number of modules per rack	4 (BMENOC0301 PV>=13, BMENOC0311 PV>=14) 3 (BMENOC0301 PV<13, BMENOC0311 PV<14)	6
configuration software	EcoStruxure Control Expert V14.0 or subsequent supporting versions	EcoStruxure Control Expert V16.2 or subsequent supporting versions
cybersecurity	• firmware signing and encryption • disable/enable protocols by conf and FB • ACL • IPsec responder • Syslog client • Achilles L2 • CSPN • California law compliance	All BMENOC0301/BMENOC0311 features, plus these: • https • SNMPv3 • NTPv4 • California law compliance • IPsec initiator • secure firmware upgrade • IKEv2
IP A, IP B (Hot Standby)	BMENOC0301.2, BMENOC0311.2, BMENOC0301.3, BMENOC0311.3	Enable or disable this feature with the **Enable IP A/B** check box on the IP configuration screen in EcoStruxure Control Expert, page 157.
firmware update tool	Unity Loader with FTP	EADM with https

Feature	**M580 Ethernet Communications Module**	**M580 High Performance Ethernet Module**
ODVA EtherNet/IP compliance	CT12	CT19
SNMP agent	SNMPv1	SNMPv1 or SNMPv3 with a check box selection
NTP client	SNTPv1	NTPv4
device network ports (ETH 2, ETH 3)	10/100 Mbps	10/100/1000 Mbps
RSTP through the device network ports (ETH 2, ETH 3)	always enabled	Enable or disable this feature on the RSTP configuration screen
web site	diagnostics web pages (including **Alarm Viewer**)	diagnostics web pages, including: • **Alarm Viewer** • **Rack Viewer** • **Program Viewer**
maximum implicit I/O data size over the rack	*X Bus:* 8KB *in*, 8KB *out*	*Ethernet bus:* 16KB *in*, 16KB *out*
crashlog accessibiltity from EcoStruxure Control Expert	no	yes
port 502 and EtherNet/IP client	16 requests (in DATA_EXCH EFs)	32 requests (in DATA_EXCH EFs)

The BMENOC0302(H) Module in Modicon M580 Networks

Functionality

The BMENOC0302(H) Ethernet communications module plays these major roles in Modicon M580 systems:

Role	Description
I/O Scanner	The main purpose of the module is to provide EtherNet/IP and Modbus TCP scanner services to distributed equipment on a device network or DIO network.
Modbus TCP server	Use the Ethernet communications module to access configuration and diagnostic data through the Modicon M580 controller module.
HTTPS server	The module includes a hypertext transfer protocol secure (HTTPS) server that provides access to the Ethernet communications module from standard Internet browsers (including but not limited to Internet Explorer).

Modicon M580 System Overview

Install a BMENOC0302(H) module on a local Modicon M580 rack to manage DIO and connected devices in distributed device networks.

Features of the Modicon M580 system:

- The system supports the connection to a control network through the service port.
- The system supports connection to a device network.
- The system facilitates the operation of Ethernet gateway devices (like Profibus and CANopen masters) as distributed devices (that use an Ethernet rack, for example).

⚠WARNING

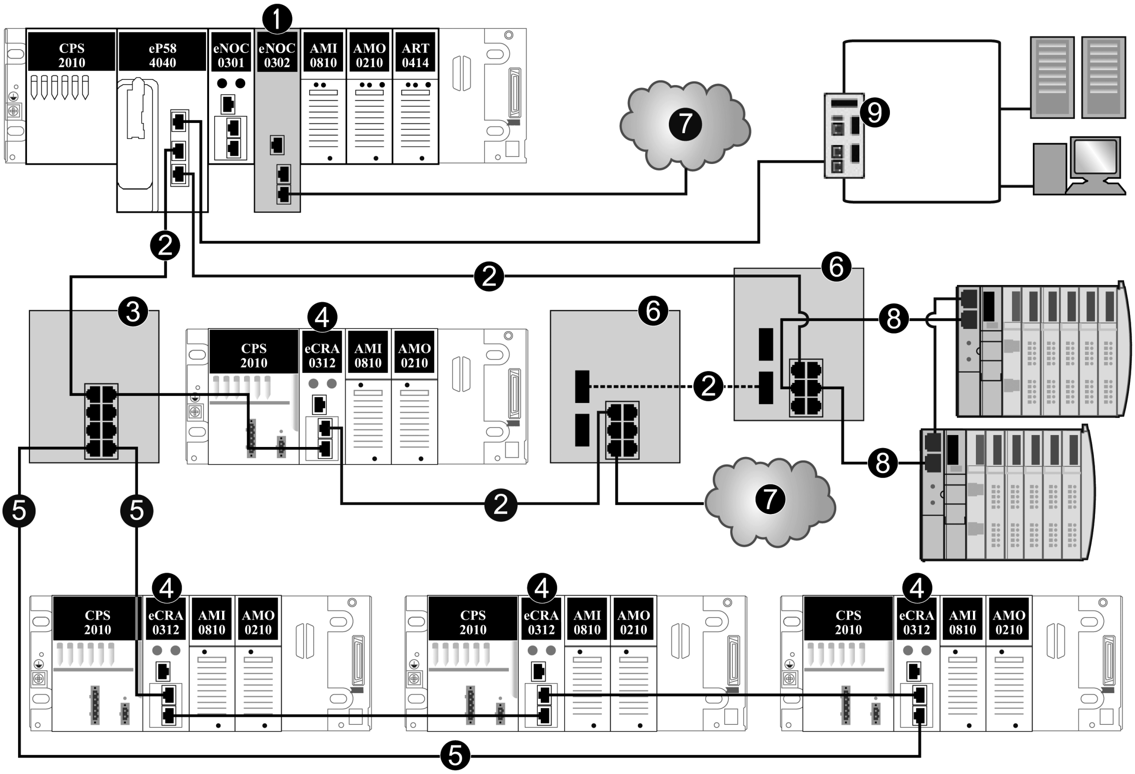
UNINTENDED EQUIPMENT OPERATION

Use managed switches with VLANs and/or routers to segregate the networks when two or more service ports are connected to the control network.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

You can configure M580 controllers and BMENOC0302(H) communications modules as DHCP servers that assign IP addresses to the Ethernet DIO drops. Ignoring the previous instruction for managed switches can result in the M580 controller module or the BMENOC0302(H) module controlling unintended Ethernet DIO drops.

In this network, the Ethernet rack (1) includes a controller module and a BMENOC0302(H) communications module. The controller connects to both the main ring (2) and a control network (9). The communications module is connected to a distributed device cloud (7):



1 main rack (including a controller and a BMENOC0302(H) module)

2 main RIO ring

3 DRS connected to the RIO main ring and RIO sub-ring

4 Ethernet RIO drops (which include a BM•CRA312•0 (e)X80 EIO adapter module)

5 RIO sub-ring

6 DRSs configured for copper-to-fiber and fiber-to-copper transition connect a DIO cloud (7) and a DIO sub-ring (8) to the main RIO ring

7 DIO cloud

8 DIO sub-ring

9 control network (connected to the controller module in the local rack)

Characteristics

When you use a BMENOC0302(H) Ethernet communications module on a Modicon M580 rack, observe these characteristics and guidelines:

- The BMENOC0302(H) module on the Ethernet bus communicates with the controller module and the other Ethernet communications modules on the rack when its rack port is enabled. These modules are in the same network to support network transparency.
- Each BMENOC0302(H) Ethernet communications module supports distributed equipment that are connected to its device network port(s) on the front panel.
- The BMENOC0302(H) module can manage 128 scanned devices. That number includes up to 16 connections to local slaves.
- The number of scanned local slaves subtracts from the number of available connections. That is, a network that makes the maximum number (16) of local slave connections has only 112 available connections to scan distributed equipment (128 – 16 = 112).
- Account for these power characteristics when you add one or more BMENOC0302 or BMENOC0302H modules to your system design:

Power Characteristic	BMENOC0302	BMENOC0302H
power rail	24 Vdc	24 Vdc
current consumption	<= 155 mA 24 Vdc	<= 155 mA 24 Vdc
MTBF reliability	1,011,833 hours 86°F/30°C continuous	601,650 hours 140°F/60°C continuous

NOTE:

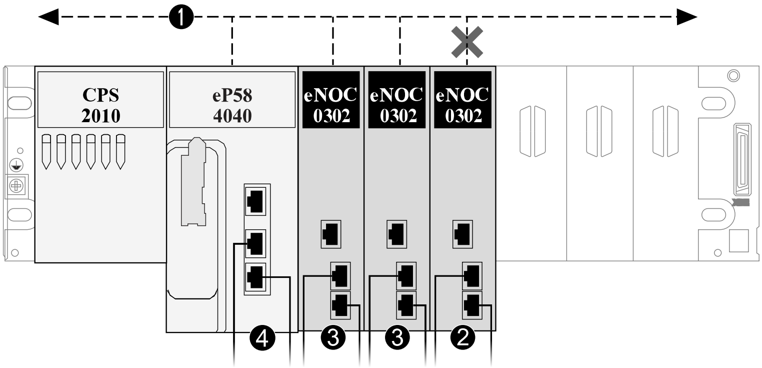
- For other Ethernet communication modules (BMENOC0301, BMENOC0311), verify that the respective power characteristics do not exceed the capabilities of the selected power supply module.
- For M580 Hot Standby systems that include RIO and DIO rings, disable the rack port of the BMENOC0302(H) module. Refer to the *Modicon M580 Hot Standby, Frequently Used Architectures, System Guide*.

Isolate the DIO Network

By default, Ethernet racks are disabled in both M580 standalone and Hot Standby systems. Therefore, a BMENOC0302(H) module installed on the rack in an M580 network, without enabling the Ethernet rack port, is isolated from the network.

An isolated DIO network is not part of the RIO network. It is an Ethernet-based network that contains distributed equipment that is connected to its device network ports (**ETH 2**, **ETH 3**). If you use dual-port distributed equipment that supports RSTP, you can connect the equipment in a daisy-chain loop to the two device network ports on a BMENOC0302(H) module. In M580 Hot Standby systems, isolate a DIO network by using a DIO ring.

Use the EcoStruxure Control Expert DTM to configure the BMENOC0302(H) module to manage an isolated DIO network. This illustration shows that one of the BMENOC0302(H) modules can manage an isolated DIO network when the connection between its Ethernet port and the Ethernet bus on the rack is disabled (as indicated by the **X**):



Legend:

1. Ethernet communications run across the rack.
2. With its rack port blocked (as indicated by the **X**), one BMENOC0302(H) module manages an isolated network that connects to its device network ports (**ETH 2**, **ETH 3**).
3. The device network ports (**ETH 2**, **ETH 3**) on two BMENOC0302(H) modules connect to separate networks or RSTP rings.
4. The device network ports on the controller module connect to an RSTP ring (such as an RIO/DIO network).

NOTE: Consider these points when you connect daisy chain loops to the BMENOC0302 (H) modules in the local rack through their device network ports (**ETH 2**, **ETH 3**):

- When the rack port for a single BMENOC0302(H) module is disabled and the module's device network ports connect to a daisy chain loop of DIO devices, the DIO network is *isolated*.
- When the rack ports for three BMENOC0302(H) modules are disabled and the device network ports for all three modules connect to their own daisy chain loops of DIO devices, all three networks are *isolated*. (That is, there are no connections between the multiple BMENOC0302(H) modules.)
- Do not enable the rack ports for two BMENOC0302(H) modules if both modules connect to a single daisy chain loop (RSTP ring) of DIO devices through their device network ports. Such a configuration creates a potential loop between that RSTP ring and the controller module's RSTP ring.

Connecting more than one module to both the rack and an Ethernet network can cause a broadcast storm.

⚠CAUTION

RISK OF BROADCAST STORM

Do not connect more than one module in a local rack to both the Ethernet rack and an Ethernet network.

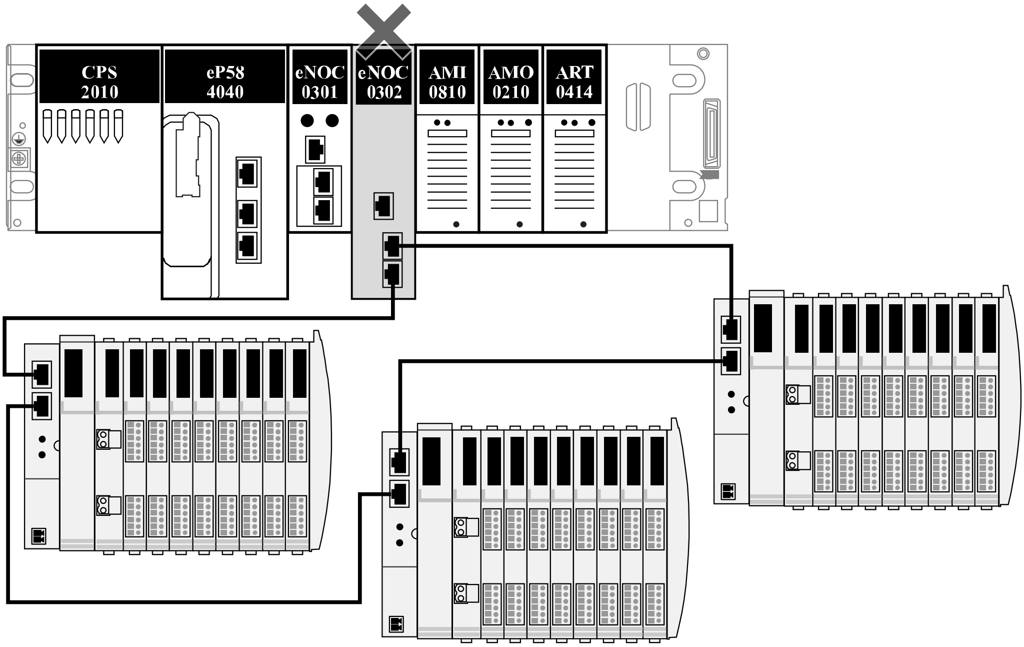
Failure to follow these instructions can result in injury or equipment damage.

These modules are available in the local rack to connect an Ethernet network to the Ethernet rack:

- controller module (when remote I/O are used)
- BMENOS0300 network option switch
- BMENOC0301, BMENOC0311, BMENOC0302(H), or BMENOP0300 communications module

Dual Attachment

You can attach a BMENOC0302(H) module to a distributed network with daisy-chain loop topology that supports RSTP for cable redundancy. In this case, disable the rack port (indicated by the **X** in this network illustration) and make a dual connection to the network with the **ETH 2** and **ETH 3** ports on the front of the BMENOC0302(H) module:

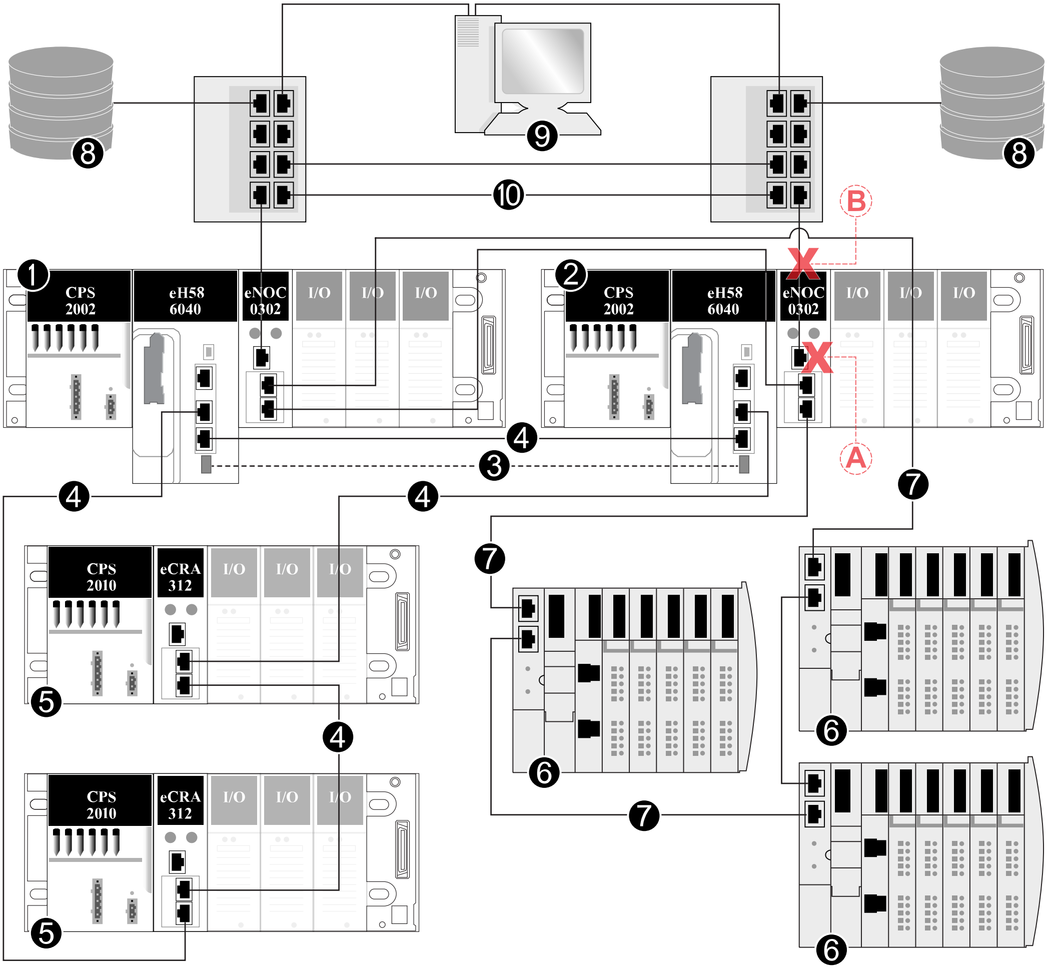


Ethernet port **ETH 2** or **ETH 3** functions as both an Ethernet switch and an interface to the module. In this case, information flows through the device to the STB islands in the loop.

The BMENOC0302(H) Module in Hot Standby Systems

Sample Network Architecture

This network topology shows an M580 Hot Standby system with an Ethernet RIO main ring, a DIO network scanned by the BMENOC0302(H) module on the local rack that communicates with the control network, and a single SCADA connection to the service ports of the primary and standby BMENOC0302(H) modules. The RIO network, device network, and control network are on the same subnet.



Legend:

A In this topology, where SCADA is connected to the Hot Standby system through the service ports of the BMENOC0302(H) modules, confirm that you select (check) **Automatic blocking of service port on Standby NOC** in the **ServicePort** configuration tab to help avoid network communication loss (formed by cables 7 and 10).

B In this topology, where a DIO ring/cloud network communicates with the control network via the BMENOC0302(H) modules, the Ethernet rack port of the standby BMENOC0302(H) module is automatically disabled to help avoid network communication loss (formed by cables 4 and 7).

Hardware:

1. The primary local rack includes a primary controller and a BMENOC0302(H) Ethernet communications module.
2. The standby local rack includes a standby controller and a BMENOC0302(H) Ethernet communications module.
3. A Hot Standby communications link connects the primary and the standby controllers.
4. Ethernet RIO main ring
5. Ethernet RIO drops include eX80 BM•CRA312•0 adapter modules.
6. Distributed equipment connects to the Ethernet DIO ring.
7. Ethernet DIO ring
8. SCADA server
9. engineering workstation with dual Ethernet
10. control room network

Device DDT Behavior

During a Hot Standby switchover, the Device DDT for the BMENOC0302(H) module is not transferred from the primary controller module to the standby controller module. Thus, the health bits in the DIO_HEALTH array are local to each Ethernet communication module. One exception is the DIO_CTRL array, which is transferred from primary to standby.

The Device DDT variables for EtherNet/IP and Modbus devices that contain freshness data and input and output data are transferred from the primary controller module to the standby controller to diagnose the health of the device and its associated data. The DIO_HEALTH array in the Device DDT for the BMENOC0302(H) can diagnose the health of the connection.

NOTE: Refer to the individual descriptions of the DIO_HEALTH array and the DIO_CTRL array in the BMENOC0302(H) module's Device DDT, page 257.

IP Addresses at Switchover

These actions occur during the switchover:

- The connections to distributed devices close.
- The BMENOC0302(H) modules swap their IP addresses:
  - The module on the new primary rack takes the configured IP address.
  - The module on the new standby rack takes the configured IP address + *1*.
- The connections to distributed devices are re-established at intervals of 600ms for Modbus devices when all services are running.

  NOTE:

  - Refer to the Hot Standby addressing considerations in the description of TCP/IP properties, page 94.
  - Refer to the Hot Standby considerations, page 98.

Ethernet Communication Modules on the Local Rack

This table shows the controller modules that are available for Modicon M580 Hot Standby systems. The table also shows the maximum number of Ethernet communication modules (when you use BMENOC0302(H) modules exclusively) in the local rack with the different controller modules:

Controller	Ethernet Communications Modules in the Local Rack
BMEH582040	2
BMEH584040	4
BMEH586040	6

NOTE: For a comprehensive list of the maximum number of communications modules supported by standalone and Hot Standby controller modules, refer to the *Controller Characteristics* tables in the *Performance Characteristics* topic of the *Modicon M580, Hardware, Reference Manual*.

Broadcast Storms

A broadcast storm can occur under one of these conditions:

- The rack port is enabled for a BMENOC03•• Ethernet communications module that is installed in a Hot Standby system.

  NOTE: A disabled rack port decreases the amount of network traffic.

- The service port for either the BMENOC03•• module or the Hot Standby controller is connected to the same switch without network segmentation (through VLAN, for example).

Connecting more than one module to both the rack and an Ethernet network can cause a broadcast storm.

⚠CAUTION

RISK OF BROADCAST STORM

Do not connect more than one module in a local rack to both the Ethernet rack and an Ethernet network.

Failure to follow these instructions can result in injury or equipment damage.

These modules are available in the local rack to connect an Ethernet network to the Ethernet rack:

- controller module (when remote I/O are used)

- BMENOS0300 network option switch
- BMENOC0301, BMENOC0311, BMENOC0302(H), BMENOP0300 communications module

  NOTE: You can have multiple BMENOC0301, BMENOC0311, BMENOC0302(H) or BMENOP0300 modules in a local rack, each with its rack port enabled, provided the embedded switch ports (the service port and the two network ports) are not used.

Hot Standby and the ETH_PORT_CTRL Function Block

When a Hot Standby system is configured to enable or disable Ethernet services and protocols, configure the ETH_PORT_CTRL function block to run in the controllers in both primary and standby rack configurations. (Refer to the instructions to configure the ETH_PORT_CTRL function block for Hot Standby systems, page 147.)

Standards and Certifications

Download

Click the link that corresponds to your preferred language to download standards and certifications (PDF format) that apply to the modules in this product line:

Title	Languages
Modicon M580, M340, and X80 I/O Platforms, Standards and Certifications	• English: EIO0000002726 • French: EIO0000002727 • German: EIO0000002728 • Italian: EIO0000002730 • Spanish: EIO0000002729 • Chinese: EIO0000002731

Communication Specifications

Introduction

The BMENOC0302(H) Ethernet communications module provides support for I/O scanning using EtherNet/IP and Modbus TCP.

These specifications describe the I/O communication and the implicit and explicit messaging capacities of the BMENOC0302(H) module.

NOTE: Refer to the description of system throughput considerations (including packets per cycle) in the *Modicon M580, Frequently Used Architectures, System Guide*.

I/O Communication Specifications

These tables present the I/O communications features for the module.

EtherNet/IP (CIP Implicit Messaging):

Feature		Maximum Capacity
scanner	number of devices	128 EtherNet/IP devices
	message size	input: 505 bytes (excluding header) output: 509 bytes (excluding header)
adapter	number of instances	16 adapter instances
	number of connections	2 connections per instance
	message size	511 bytes (including header)
	inputs	505 bytes (excluding header)
	outputs	509 bytes (excluding header)

Modbus TCP (Modbus I/O Scanner):

Feature		Maximum Capacity
registers	number of devices	128 devices shared with EtherNet/IP
	read	125 registers
	write	120 registers
message size	read	250 bytes (125 words) (excluding header)

Feature		Maximum Capacity
	write	240 bytes (120 words) (excluding header)

The total number of distributed devices that can be scanned by the BMENOC0302(H) modules in a controller configuration is limited to 512.

Combined EtherNet/IP Scanner/Adapter and Modbus Scanner:

I/O Data Exchange with the controller module		
Feature	**Maximum Capacity**	**Comment**
input data size	16KB (8KW), including overhead	16 KB of data includes user configurable data and overhead. The overhead includes module diagnostic data, data object headers, and the number of headers depending on the user configuration. As a result, the maximum user configurable input data size is approximately 15.42KB (1 KB = 1,024 Bytes).
output data size	16KB (8KW), including overhead	16 KB of data includes user configurable data and overhead. The overhead includes module control data, data object headers, and the number of headers depending on the user configuration. As a result, the maximum user configurable output data size is approximately 15.56KB (1 KB = 1,024 Bytes).

Default RPI values:

Task	**CRA > M580 controllers**	**M580 controllers > CRA**
Periodic MAST	Mast_Period / 2 (rounded down to ms)	Mast_Period (+ 10% or 2 ms, whichever is larger) (rounded down to ms)
Cyclic MAST	Watchdog / 4 (rounded down to ms)	Watchdog / 4 (rounded down to ms)
Periodic FAST	Fast_period /2 (first value > 50ms to allow transparent loop recovery	fast_period (+ 10% or 2 ms, whichever is larger) (rounded down to ms)
		Output are published synchronously at end of FAST cycle
Aux 0	Aux0_period / 2	Aux_period (+ 10% or 2 ms, whichever is larger) (rounded down to ms)
Aux 1	Aux1_period / 2	Aux1_period (+ 10% or 2 ms, whichever is larger) (rounded down to ms)

NOTE:

- If lower minimum values or default values are chosen, the system does not function.
- The requested packet interval (RPI) accuracy requirement is derived from ODVA:
  - Mean measured packet interval (MPI) with respect to reported API < +-10%
  - Average jitter of the measured mean MPI < +- 10%
  - Maximum jitter of the measured mean MPI < +-50%

Most of the data to be exchanged between the M580 controller and the BMENOC0302(H) module are performed through the Ethernet backplane. These exchanges may have some impact on the MAST task cycle time of the controller. Calculate the minimum cycle time required by the MAST cycle duration to correctly perform these exchanges with this formula:

Mast Cycle Min (ms) = the greater value between (implicit_data_IN + implicit_data_OUT) / 2 and 8ms

The formula uses these variables:

- *implicit_data_IN:* This variable represents the size (KB) of implicit data inputs between the controller and all BMENOC0302(H) scanners (Modbus or EtherNet/IP requests) configured in the application.
- *implicit_data_OUT:* This variable represents the size (KB) of implicit data outputs between the controller and all BMENOC0302(H) scanners (Modbus or EtherNet/IP requests) configured in the application.

Explicit Messaging Specifications

These tables present the explicit messaging features of the module.

NOTE: These tables report the maximum capacity for a single BMENOC0302(H) module.

EtherNet/IP (CIP explicit messaging):

Feature		**Maximum Capacity**
client	simultaneous requests	32 requests in data exchange blocks
	message size	1024 bytes
server	simultaneous requests	32 connections
	message size	1024 bytes

Modbus TCP (Modbus explicit messaging):

Feature		**Maximum Capacity**
client	simultaneous requests	32 requests in data exchange blocks
	message size	1024 bytes
server	simultaneous requests	64 connections
	message size	1024 bytes

Install the BMENOC0302(H) Module

Introduction

Use the information below to install the BMENOC0302(H) Ethernet communications module in an M580 system.

NOTE: An M580 rack supports up to six BMENOC0302(H) modules, depending on the selected controller. Refer to these resources:

- The available controller modules are listed in these guides:
  - *Modicon M580, Frequently Used Architectures, System Guide*
  - *Modicon M580 Hot Standby, Frequently Used Architectures, System Guide*
  - *Modicon X80 Racks and Power Supplies, Hardware, Reference Manual*
- Refer to the power characteristics for the BMENOC0302(H) module, page 37.

Mount the BMENOC0302(H) Module on a Modicon M580 Rack

Introduction

Use these instructions to install a BMENOC0302(H) high-performance Ethernet communications module in a single slot on an Ethernet rack.

NOTE: Fitting operations (installation, assembly, and disassembly) are described below.

Rack Selection

These racks and their corresponding hardened (*H*) versions support the BMENOC0302(H) module:

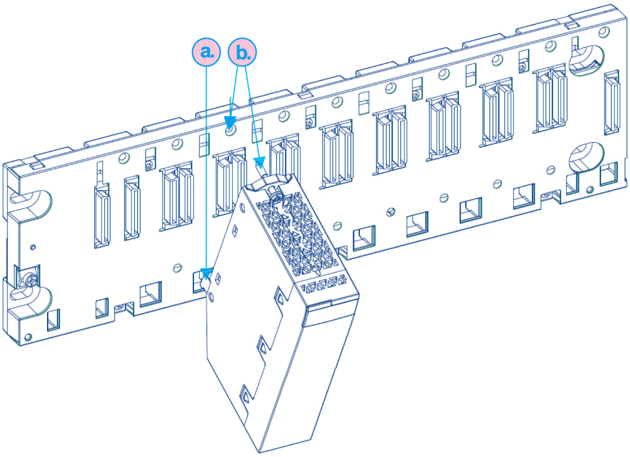
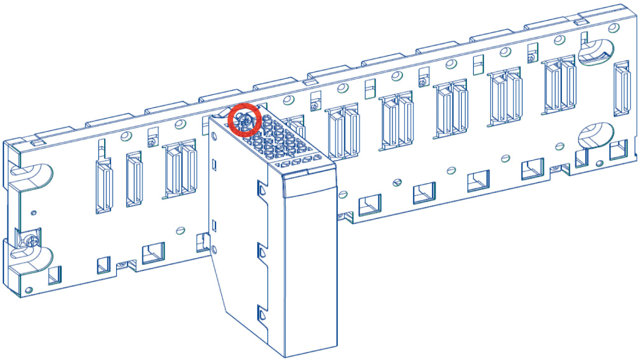
- BMEXBP0400, BMEXBP0400H (4 slots)
- BMEXBP0602, BMEXBP0602H (6 slots)
- BMEXBP0800, BMEXBP0800H (8 slots)
- BMEXBP1002, BMEXBP1002H (10 slots)
- BMEEBP1200, BMEXBP1200H (12 slots)

NOTE:

- In local racks, slots 00 and 01 are reserved for the controller module.
- In 10- and 12-slot Ethernet racks, slots 02, 08, 10, and 11 are X Bus-only slots. Install a BMENOC0302(H) communication module in a different rack slot.

Install the Module on the Rack

Mount the BMENOC0302(H) module in a single slot on one of the supporting racks:

Step	Action
1	Turn off the power supply to the rack.
2	Remove the protective cover from the module interface on the rack.
3	 a. Insert the locating pins on the bottom of the module in the corresponding slots in the rack. b. Use the locating pins as a hinge and pivot the module until it is flush with the rack. (Insert the twin connector on the back of the module in the connectors on the rack.)
4	Tighten the retaining screw to hold the module in place on the rack:  **NOTE:** Tightening torque: 1.1...1.5 N•m (0.81...1.10 lbf-ft).

Grounding Considerations

This section describes the wiring guidelines and best practices to be respected when installing and cabling the BMENOC0302(H) Ethernet communications module.

⚠⚠DANGER

HAZARD OF ELECTRIC SHOCK, EXPLOSION OR ARC FLASH

- Disconnect all power from all equipment including connected devices prior to removing covers or doors or installing or removing accessories, hardware, cables, or wires except under the specific conditions specified in the appropriate hardware guide for this equipment.
- Always use a properly rated voltage sensing device to verify that the power is off where and when indicated.
- Replace and secure all covers, accessories, hardware, cables, and wires and verify that a proper ground connection exists before applying power to the unit.
- Use only the specified voltage when operating this equipment of associated products.

Failure to follow these instructions will result in death or serious injury.

⚠WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.[1]
- Test each implementation of a system for proper operation before placing it into service.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

Apply these rules when your connect cables to the BMENOC0302(H) Ethernet communications module:

- Use twisted pair, shielded (ferrule) cables with the proper rating for your installation/ environment.
- Verify that communication wiring is separate from the power wiring. Route these two types of wiring in separate cable ducting.
- Verify that the operational conditions and environment are within the values cited in the present document and the other user guides associated with this equipment.

If you do not use proper, shielded (ferrule) cables for these connections, electromagnetic interference can cause signal degradation. Degraded signals can cause the controller or other attached modules and equipment to perform in an unintended manner.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

- Use shielded (ferrule) cables for all communication signals.
- Ground (ferrule) cable shields for all communication signals at a single point[1].
- Route communication separately from power cables.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

[1] Multipoint grounding is permissible if connections are made to an equipotential ground plane dimensioned to help avoid cable shield damage in the event of power system short-circuit currents.

Use fiber-optic cable to establish a communications link when it is not possible to equalize the potential between the two grounds.

NOTE: Refer to the ground protection information provided in the *Electrical installation guide* and *Control Panel Technical Guide, How to protect a machine from malfunctions due to electromagnetic disturbance*.

Module Replacement

You can replace a BMENOC0302(H) Ethernet communications module on the rack with another BMENOC0302(H) module that implements the same version of the firmware (or a subsequent supporting version).

In this case, the replacement module obtains its operating parameters over the rack connection from the controller. The transfer occurs immediately at the next cycle to the device.

Cable Installation

Introduction

Use four-pair, twisted-pair, copper-shielded cables to connect a BMENOC0302(H) module to a DIO network in an M580 system.

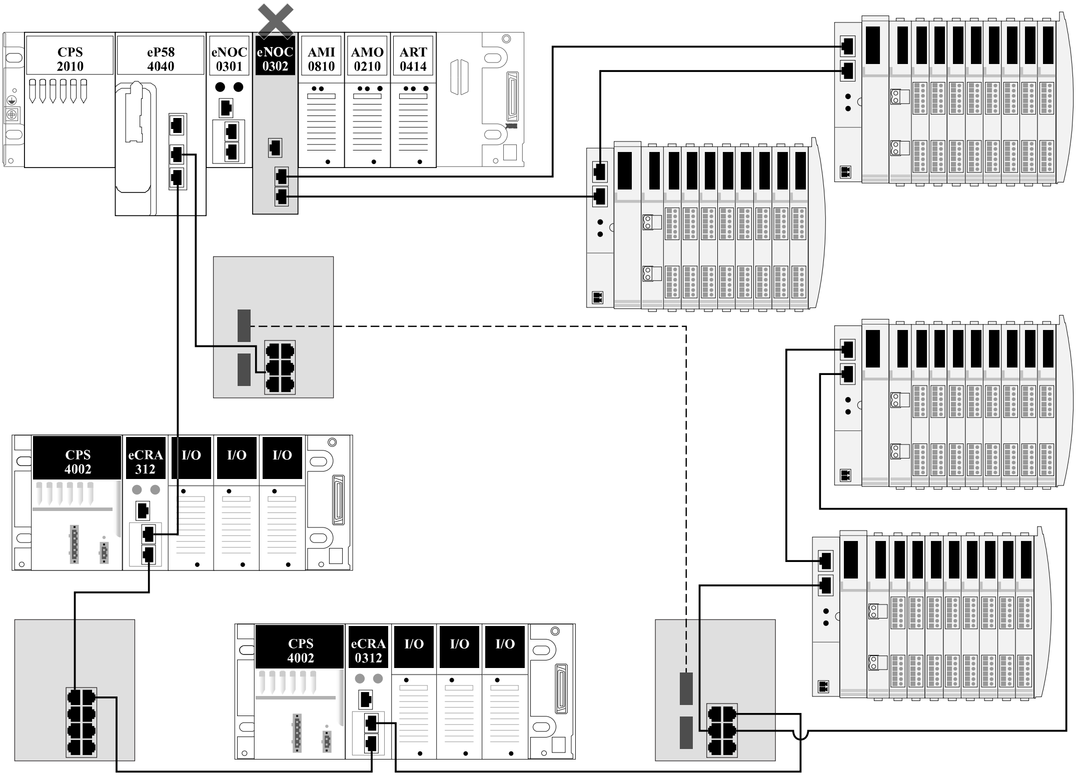
Select the appropriate cable according to the transmission speed:

Speed	Cable
10/100 Mbps	Use CAT5e or CAT6 cables for transmission speeds that are less than or equal to 100 Mbps.
10/100/1000 Mbps	Use only CAT6 cables for transmission speeds greater than 1000 Mbps.
NOTE: Use only *four*-pair, twisted-pair cables for these connections. Do not use *two*-pair cables.	

Use two-pair, twisted-pair, copper-shielded CAT5e (10/100 Mbps) cables for these connections in M580 systems.

Connections Between Devices

This example shows the cable lengths between RIO and DIO devices in an M580 device network:



Legend:

- *solid line:* Use copper cables for distances less than or equal to 100 m.
- *dashed line:* Use fiber-optic cables for distances greater than 100 m.
- ***X***: The *X* indicates that the connection between the Ethernet port on the BMENOC0302 (H) module and the Ethernet bus is disabled to facilitate the management of the DIO network that is isolated and connected to the device ports. (Refer to the description of an isolated DIO network, page 41.)

A BMENOC0302(H) module supports distributed equipment through the device network port (s) on the front of the module while respecting the limitation of 128 scanned devices per BMENOC0302(H) module. (Refer to the description of **Switch** properties for the rack connection, page 93.)

BMENOC0302(H) Module Configuration

Introduction

Use the EcoStruxure Control Expert Classic programming software to select and configure the BMENOC0302(H) Ethernet communications module on the local rack.

> **NOTE:** The device configuration procedure is valid when configuring a project with EcoStruxure Control Expert. When you configure your device from a system project, some commands are disabled in the EcoStruxure Control Expert editor. In this case, configure these parameters at the system level by using EcoStruxure Control Expert (with Topology Manager).

For more information on creating and/or modifying EcoStruxure Control Expert projects, refer to *EcoStruxure™ Control Expert, Operating Modes*.

Add the BMENOC0302(H) Module

Add a BMENOC0302(H) Ethernet communications module to the EcoStruxure Control Expert project:

Step	Action
1	View the available communication modules (**Hardware Catalog > Modicon M580 local drop > Communication**).
2	Drag and drop the BMENOC0302(H) Ethernet communications module to an open slot in the rack to see the **New Device** window.
3	Examine the topological address for the module in the **New Device** window and click **OK** to see the **General** tab of the **Properties of device** window. **NOTE:** The **General** tab contains configurable information. The other tabs contain read-only information.
4	Examine the provided **DTM name** for the module and click **OK**. You can use this field to configure a different **DTM name**: • When you change the **DTM name**, EcoStruxure Control Expert changes the base input and output type and variable names to match the new **DTM name**. • Assign a unique **DTM name** to each communication module to distinguish between modules of the same type. • The **DTM name** is used elsewhere in EcoStruxure Control Expert: ◦ It is the **Network name** when you view the module properties. ◦ It is the module name in the **DTM Browser** under the **Host PC**.
5	Confirm that the **PLC bus** displays the BMENOC0302(H) module, and save the project (**File > Save**).

Communication Module and Remote Device Node Commands

Right-click the BMENOC0302(H) module in the **PLC bus** to use these commands:

Name	Description	Alternate Command
Delete Module	• Delete the selected module from the rack. • Delete the selected module from the **DTM Browser**. • Delete the corresponding DTM and its sub-node DTMs from the DTM connectivity tree.	**Edit > Delete**
Open Module	See a description of the selected communications module.	**Edit > Open** Double-click the module in the **PLC bus**.
Move Module	Move the selected module to the rack slot that you designate.	**Edit > Move Module...**
Go to DTM	Display the DTM for the module in the **DTM Browser**.	**Tools > DTM Browser**
Power Supply and IO Budget	View the power consumption of the module and a list of networks that include the module. **NOTE:** The **Power Supply and IO Budget** is described in more detail below.	**Services > Power Supply and IO Budget**

Power Supply and I/O Budget

For each communication module on the local rack, you can view information about the power supply and I/O budget for the application-specific channels in the bar charts.

Right-click the BMENOC0302(H) module in the **PLC bus** and scroll to **Power Supply and IO Budget** to view these tabs:

- **Power supply:** This tab shows the total power that the module uses and the amount of power that is discharged in the module for each voltage it uses.
- **I/O:** This tab shows the number of application-specific channels that are configured in the module.

The bar charts on these tabs indicate the state of the budget according to this color scheme:

- *green:* This is the number of configured channels.

- *white:* This is the number of available channels.
- *red:* This is the number of channels that are not managed by the BMENOC0302(H) module. (In this case, a message reports the excess of unmanaged channels.)

Export the EcoStruxure Control Expert Module Configuration

At a Glance

In EcoStruxure Control Expert, use the **PLC bus** configuration window to export the BMENOC0302(H) module configuration and the devices configured behind the NOC master DTM.

The entire configuration is copied to a `.ZHW` file.

Export the Configuration

Export the module configuration:

Step	Action
1	Expand (**+**) the **Project Browser** to see the **PLC bus** (**Project > Configuration > PLC bus**).
2	Expand (**+**) **PLC bus** to see the M580 rack, and expand the rack to see the BMENOC0302(H) module.
3	Right-click the BMENOC0302(H) module and scroll to **Export** to launch the **Export** dialog box.
4	Use standard Windows commands to select a destination directory for the export.
5	Enter the file name in the **File name** text field.
6	Click the **Export** button and monitor the progress bar.

A message in the **output window** tells you that export is complete.

Import the Configuration for an Ethernet Communications Module

At a Glance

Use the import function to import the module configuration for one of these devices and the devices configured behind the BMENOC03•• master DTM:

- BMENOC0301
- BMENOC0311
- BMENOC0302(H)

Consideration

- You can import files that were previously exported as `.ZHW` configuration files.
- You cannot import the same file to the same application more than once.

Import the Configuration

Import a module configuration:

Step	Action
1	Create a file conversion with the **M580ApplicationUpdate.exe** tool that is located in the same program directory as the EcoStruxure Control Expert software. **NOTE:** The **Import** command is enabled in these cases: • The controller module is not connected to EcoStruxure Control Expert. • An empty slot of the M580 BMEXBP•••• main rack is selected.
2	Expand (**+**) the **Project Browser** to see the **PLC bus** in the **Structural view** (**Project Browser > Project > Configuration > PLC bus**).
3	Expand **PLC bus** to see the M580 rack.
4	Right-click an empty rack slot in the **Structural view** and scroll to **Import** to launch the **Import** dialog box.
5	Use standard Windows commands to find and select a file for import. **NOTE:** • A dedicated tooltip indicates the type of content for the `.ZHW` file. • The name of the file appears in the **File name** field.
6	Click the **Import** button and monitor the progress bar. **NOTE:** A pop-up message tells you that the import is complete.

Configuration with the EcoStruxure Control Expert DTM

Introduction

Use the instruction in this section to configure an Ethernet communications module with the EcoStruxure Control Expert DTM.

Introduction to FDT/DTM

Overview

EcoStruxure Control Expert incorporates the Field Device Tool (FDT) / Device Type Manager (DTM) approach to integrate distributed devices with your process control application. EcoStruxure Control Expert includes an FDT container that interfaces with the DTMs for EtherNet/IP and Modbus TCP devices.

An EtherNet/IP device or Modbus TCP device is defined by a collection of properties in its DTM. For each device in your configuration, add the corresponding DTM to the EcoStruxure Control Expert **DTM Browser**. From the **DTM Browser**, you can open the device's properties and configure the parameters presented by the DTM.

Device manufacturers may provide a DTM for each of their EtherNet/IP or Modbus TCP devices. However, if you use an EtherNet/IP or Modbus TCP device that has no DTM, configure the device with one of these methods:

- Configure a generic DTM that is provided in EcoStruxure Control Expert.
- Import the EDS file for the device. EcoStruxure Control Expert populates the DTM parameters based on the content of the imported EDS file.

  NOTE: The DTM for a BMENOC0302(H) module is automatically added to the **DTM Browser** when the module is added to the **PLC bus**.

Open the DTM Browser

View the configuration options for the BMENOC0302(H) Ethernet communications module in the EcoStruxure Control Expert **DTM Browser**:

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the EcoStruxure Control Expert **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module.
4	Double-click the module name to open the configuration window.
5	View the DTM configuration parameters for the module in the open dialog box: • *Channel Properties*, page 89 • IPsec configuration, page 118. • *EtherNet/IP Local Slaves*, page 226 • *Device List*, page 152 • *Logging*, page 163

DTM Types

The **DTM Browser** presents a hierarchical list of DTM nodes on a connectivity tree. The DTM nodes in the list were added to your EcoStruxure Control Expert project. Each node represents a module or device in your Ethernet network.

There are two kinds of DTMs:

- *master DTMs*: This DTM is both a device DTM and a communication DTM. The master DTM is a pre-installed component of EcoStruxure Control Expert.
- *generic DTMs*: The EcoStruxure Control Expert FDT container is the integration interface for a device’s communication DTM.

This list contains these node types:

DTM Type	Description
communication (master)	Communication DTMs appear under the root node (host computer). A communication DTM supports gateway DTMs or device DTMs as children if their protocols are compatible.
gateway	A gateway DTM supports other gateway DTMs or device DTMs as children if their protocols are compatible.
device	A device DTM does not support child DTMs.

Node Names

Each DTM node that is inserted in the browser gets a default name. The default name for gateway and device DTMs is in the format *<protocol:address> device name*. (For example, < EtherNet IP:192.168.20.3 > BMENOC0302_from_EDS.)

This table describes the components of the default node name:

Element	Description
channel	This is the name of the channel communication medium into which the device is plugged. This name is read from the DTM and is set by the device vendor. **Example**: EtherNet/IP, Modbus
address	This is the bus address of the device that defines the connection point on its parent gateway network (for example, the device IP address).
device name	The default name is determined by the vendor in the device DTM, but you can edit the name.

Node Status

The **DTM Browser** contains graphics to indicate the status of each DTM node in the connectivity tree:

Status	Description
Built / Not-built	A blue checkmark is superimposed on a device icon to indicate that the node (or one of its sub-nodes) is not built. This means that some property of the node has changed, so the information stored in the physical device is not with the local project.
Connected / Disconnected	A connected DTM appears in **bold** text. An unconnected DTM appears in plain text. **NOTE:** • Connecting a DTM to its physical device automatically connects higher level parent nodes up to the root node. • Disconnecting a DTM from its physical device automatically disconnects its lower-level child nodes. **NOTE:** Connecting or disconnecting a DTM to or from its device does not also connect or disconnect EcoStruxure Control Expert to or from the device. DTMs can be connected/disconnected while EcoStruxure Control Expert is either offline or online.
Installed / Not-installed	A red **X** is superimposed on a device icon to indicate that the DTM for that device is not installed on the computer.

Handling Invalid Nodes

A red **X** superimposed on a node indicates that the DTM for that node is not installed on the computer. To resolve this situation, right-click the node and open a pop-up menu with these commands:

Command	Description
Delete	Remove the selected node (and its sub-nodes) from the **DTM Browser**.
Properties	Open the **Properties of ...** dialog box to identify the name of the missing DTM.

NOTE: After you install the DTM, reopen the EcoStruxure Control Expert application.

DTM Browser Menu Commands

Introduction

The EcoStruxure Control Expert **DTM Browser** includes these commands for the selected DTM associated with a BMENOC0302(H) module:

- Universal commands (determined by the selected node level):
  - Host PC node (level 1)
  - Communication module node (level 2)
  - Remote device node (level 3)
- Device-specific commands (determined by the device DTM)

Host PC Node Commands

In the EcoStruxure Control Expert **DTM Browser**, right-click **Host PC** to access these commands:

Name	Description
Add...[1]	Open the **Add** window (a subset of the **Hardware Catalog**). Select a device DTM to add to the **DTM Browser**.
Check DTM devices[1]	Examine the project for invalid DTMs or DTMs that are not installed on the computer. Invalid or not-installed DTMs appear in the **User errors** tab in the information window and a red **X** is superimposed over their icons in the **DTM Browser**.
DTM services[1]	Display the communication DTMs and the device topology along with their respective IP addresses and connection states. For each device, you can connect, disconnect, load data from devices, or store data to devices. You can also choose to stop communications or continue an activity when errors are detected.
DTM hardware catalog[1]	Display the **DTM catalog** tab in the **Hardware Catalog**.
Expand all[2]	Display and expand every DTM in the project in the **DTM Browser**.
Collapse all[2]	Display only the communication DTMs in the project.
1. This command is also in the **Edit** menu. 2. This command is also in the **View** menu.	

Communication Module and Device Commands

Right-click the desired module or device in the **DTM Browser** and scroll to these commands:

Name	Description
Open[1]	View the configuration options for the selected module or device. **NOTE:** You can also double-click the DTM in the **DTM Browser** to open this window.
Add[1]	Open the **Add** dialog box to view a subset of available DTMs in the **Hardware Catalog**. **NOTE:** EcoStruxure Control Expert filters the content of the **Add** dialog box to present only the DTMs that are compatible with the selected DTM.
Delete[1]	Delete the selected DTM and its sub-node DTMs from the DTM connectivity tree when the selected DTM supports this function.
Field Bus Discovery[1]	Scan the connected physical devices to create the corresponding field bus topology. **NOTE:** Refer to the *Field Bus Discovery Service topic*, page 79.
Sort by	Sort the DTMs according to the sub-selections in this pull-down menu: • **Protocol** • **Device Name** • **IP Address**
Connect[1]	Connect the DTM to its physical device on the network. This connection does not depend on the controller module online/offline status of the EcoStruxure Control Expert project application. **NOTE:** Connecting a gateway or device DTM implicitly connects its parent DTM.
Disconnect[1]	Disconnect the DTM from its physical device. This disconnection depends on the online/offline status of the controller module in the EcoStruxure Control Expert project application. **NOTE:** The disconnection of a gateway or device DTM implicitly disconnects its parent DTM.
Load data from device[1]	This service is not authorized for the BMENOC0302(H) DTM.
Store data to device[1]	This service is not authorized for the BMENOC0302(H) DTM.
Copy[1]	Copy the selected device DTM.
Paste[1]	Paste the selected device DTM.
Go to module or device[1]	Delete a pre-configured module DTM: • Right-click the desired DTM node. • Select **Go to module or device**. • Right-click the module, and select **Delete**. **NOTE:** You cannot use this feature if you manually open the window that displays the module/device that is selected for deletion.

Name	Description
Device menu	Open a sub-menu that contains device-specific commands, as determined by the device vendor. **NOTE:** The selections within the **Device menu** are described in the next table.
Properties[1]	Open the Ethernet communications module's **Properties** window. • *General:* The **DTM name management** field shows the name of the module. *Device Information:* This tab displays manufacturer, version, and date information about the module. *DTM Information:* This tab displays manufacturer, version, and date information about the DTM. *Protocol information:* This tab displays information about the implemented and supported protocols.
Print device[1]	Display the device documentation (including configuration settings) in the computer's default Internet browser. That information can then be printed. **NOTE:** • This function is not supported by all DTMs. • Device information can be printed for only one device DTM at a time when the DTM is not open for editing in the **Device Editor**. • Device information can be printed only when the DTM is disconnected from the physical device.
Zoom out[2]	This returns to the display of the entire DTM connectivity tree.
Expand all[2]	Display the DTMs below the selected DTM.
Collapse all[2]	Display only the selected DTM.
1. This command is also in the **Edit** menu. 2. This command is also in the **View** menu.	

Communication Module Commands

Right-click the desired module or device in the **DTM Browser** and scroll to **Device menu** to open a sub-menu that contains these commands:

Name	Description
Offline Parameter	This command is disabled.
Online Parameter	This command is disabled.
Compare	Compare two devices (**Offline Compare** or **Online Compare**).
Configuration	Open the **Device Editor** for the selected communication module when the module and its DTM are disconnected.
Observe	This command is disabled.
Diagnosis	Open the **Diagnosis Window** for the selected communication module when the module and its DTM are connected.
Additional functions	**Add EDS to library**
	Remove EDS from library
	Export EDS library
	Import EDS library
	Store Device Conf to FDR
	Online Action
	EtherNet/IP Explicit Message
	Modbus Explicit Message
	About: View module information.
	Advanced Mode: Display or hide expert-level properties that help define Ethernet connections.

Enable Advanced Mode

Use the contextual menu in the **DTM Browser** to toggle EcoStruxure Control Expert in or out of **Advanced Mode**, thereby displaying or hiding expert-level properties that help define Ethernet connections.

> **NOTE:** To maintain system performance, verify that the **Advanced Mode** properties are configured by persons with a solid understanding of communication protocols.

Enable and disable **Advanced Mode**:

Step	Action
1	Close the configuration windows associated with the Ethernet communication module.
2	In the **DTM Browser**, right-click the Ethernet communication module.
3	Scroll to **Additional functions** (**Device menu > Additional functions**) to see the status of the **Advanced Mode**: • *enabled (selected):* **Advanced Mode** is enabled. • *disabled (deselected):* **Advanced Mode** is disabled. **NOTE:** If a configuration or properties window that is associated with the device or module is open, the **Advanced Mode** is not available (grayed out).
4	Select **Advanced Mode** to toggle its status. For example, if **Advanced Mode** is checked (enabled), select it to disable it.

Configure these items in **Advanced Mode**:

- EtherNet/IP features, page 116 (timeout parameters and DIO scanner behavior)
- RSTP parameters, page 109
- Online Action, page 284 (refresh data and reset devices)

Managing DTM Connections

Introduction

Use these instructions to connect or disconnect a device DTM or module DTM to or from a physical device or module.

Connecting and Disconnecting

Connect or disconnect a DTM and the associated device or module through the contextual pop-up menu in the EcoStruxure Control Expert **DTM Browser**:

Step	Action
1	In the EcoStruxure Control Expert **DTM Browser**, locate the DTM that you want to connect to or disconnect from.
2	Right-click to view a pop-up menu.
3	Select **Connect** or **Disconnect** from the pull-down menu (or access the **Connect** and **Disconnect** commands in the EcoStruxure Control Expert **Edit** menu): • **Connect**: Perform these tasks with a connection: ◦ Configure Ethernet communication modules, distributed devices, and their common Ethernet connections. ◦ Monitor and diagnose the real-time operation of the device or module. • **Disconnect**: Perform these tasks without a connection: ◦ Configure an Ethernet communication module or distributed device by editing its properties. ◦ A disconnected DTM appears in normal text (not **bold**). (The **Connect** command is available only for disconnected DTMs.)

The **DTM Browser** indicates the relationship between the DTM and the remote module or device:

- A connected DTM appears in **bold** text.

  A disconnected DTM appears in regular (not **bold**) text.

  NOTE: The **Disconnect** and **Connect** commands are available only for connected DTMs.

To connect to the BMENOC0302(H) module, set the **Source IP Address** in the channel properties configuration, page 89 to the same network as the communications module.

Field Bus Discovery Service

Introduction

Use the field bus discovery service to detect and add to your EcoStruxure Control Expert application, network devices that are situated on a local network. The field bus discovery service is available when the Ethernet communication module DTM is connected to its physical device.

Only the first level devices below the communication DTM are detected.

Performing Field Bus Discovery

The results of the scanning process are compared to the registered DTMs in the DTM catalog of the computer. If a match is found in the DTM catalog for a scanned device, the results are accompanied with a matching type that gives the accuracy of the match.

These are the available matching types:

- *Exact match*: Every identification attribute matches. The correct device type is found.
- *Generic match*: At least the **Vendor** and device **Type ID** attributes match. The support level of the DTM is “Generic Support.”
- *Uncertain match*: At least the **Vendor** and device **Type ID** attributes match. The support level of the DTM is *not* “Generic Support.”

Use the field bus discovery service:

Step	Action
1	In the **DTM Browser**, select an appropriate DTM. **NOTE:** The field bus discovery service limits its search to the range of IP addresses that is pre-configured for the selected channel in the **Channel Properties** configuration, page 89.
2	Right-click the DTM and scroll to **Field bus discovery** to open the corresponding dialog box.
3	Make selections in the **Channel** and **Protocol** menus: • The DTM has more than one channel. • The channel supports more than one protocol.
4	Click **OK** to start the detection of devices on the selected channel by the service.
5	If at least one matched device is found, the **Field Bus Discovery** dialog box displays a list of **Scanned Devices**.
6	Use the controls of the **Field Bus Discovery** dialog box to select the devices to add to your EcoStruxure Control Expert application.
7	After you have selected devices to add to the **Field Bus Discovery** dialog box, click **OK**.
8	If the field bus discovery process finds at least one device with an IP address that is used in the project, you are prompted to continue and replace the existing project device(s): • **Yes**: Open the **Properties of device** dialog box (and continue to the next step). • **No**: Cancel automatic field bus discovery.
9	In the **Properties of device** dialog box, select the **General** tab, enter the **Alias name** for the device to be added, and click **OK**. The dialog box closes and reopens if there is another device to be added to the application. **NOTE:** When if first opens, the **Properties of device** dialog box displays the default name for the first discovered device to be added.
10	Repeat the above step for each additional discovered device.
11	After you finish adding devices to the application, configure each device for operation as part of the application: • Disconnect the Ethernet communication module from its DTM. In the **DTM Browser**, select the Ethernet communication module and disconnect it (**Edit > Disconnect**). • Configure the device properties in the DTMs for both the Ethernet communication module, and the added remote device.

Field Bus Discovery Dialog Box

The **Field bus discovery** dialog box appears when at least one matched device is found. It lists the scanned and matched devices. When you select the matched devices to be created in the EcoStruxure Control Expert project, those devices appear in the **Selected Devices** list:

Device Type	**Description**
Scanned Devices	This table displays the devices (matched and unmatched) that are found during the scan.
Matched Devices	This table displays the matched DTMs that are found in the workstation DTM catalog for the device that you selected in the **Scanned Devices** list. Each time a scanned device is selected in the **Scanned Devices** list, the contents of the **Matched Devices** list is updated to display the matched device DTMs found for the selected scanned device. The matching process can yield one or more matched devices for a given scanned device. In this case, only one DTM was discovered for the selected scanned device.
Selected Devices	This table displays the device DTMs that are selected in the **Matched Devices** list for inclusion in the EcoStruxure Control Expert project.

The lists use these colored icons:

Color	**Meaning**
green	The device is selected.
yellow	The device is matched.
red	The device is **not** matched.
black	Information about the address of the scanned device: • In the **Scanned Devices** list, the device has an address that isidentical to a DTM in the EcoStruxure Control Expert project • In the **Matched Devices** list, the device will be assigned an address that is identical to a DTM in the EcoStruxure Control Expert project.

NOTE: An icon can consist of two colors. For example, a search can discover a device that:

- has a matching DTM, and
- has an IP address identical to a device in the EcoStruxure Control Expert application

In this case, the icon next to the discovered device is:

- half yellow and half black before it is selected, and
- half green and half black after it is selected

These buttons appear in the dialog box:

Name		**Description**
Add All	+	Automatically add the most closely matched (according to the matching types listed above) device DTM for each device found in the **Matched Devices** list to the **Selected Devices** list.
Add One	+	Add the matched device DTM that is selected in the **Matched Devices** list.
Remove	−	Remove one or more devices from the **Selected Devices** list.
OK		Insert the device DTMs in the **Selected Devices** list in the EcoStruxure Control Expert project. If there are one or more devices in the **Selected Devices** list that have the same address in the EcoStruxure Control Expert project, a message box prompts you to continue. If you click **OK**, devices in the EcoStruxure Control Expert project that have identical addresses as the selected devices are **deleted** and **replaced** by the DTMs selected in the **Selected Devices** list.
Cancel		Cancel the field bus discovery scan and discard the information in the three lists.

Configure DTM Properties

Introduction

You can edit and view parameters in the **Device List** that are associated with the M580 DTM.

Open the Device List

View the **Device List**:

Step	Action
1	Open the **DTM Browser** in EcoStruxure Control Expert (**Tools > DTM Browser**).
2	Double-click the M580 DTM in the **DTM Browser**.
3	In the configuration tree associated with the M580 DTM, click **Device List**.

Property Configuration

While you edit a parameter field in the **Device Editor**, an icon appears next to the edited field in the navigation tree. The icon refers to the value of the edited parameter:

!	*exclamation point:* A value is not valid.
✎	*pencil:* A parameter is edited.

Click the appropriate button:

- **Apply:** Save your changes and keep the page open.

  NOTE: The **Apply** button is inactive until you enter a valid value.
- **OK:** Save your changes and close the page.
- **Cancel:** Cancel changes.

  NOTE: Your changes do not take effect until they are successfully downloaded from your computer to the controller module and from the controller module to the communication modules and network devices.

Upload and Download DTM-Based Applications

Introduction

You can use EcoStruxure Control Expert to download an application file from local computer to the controller module and upload an application file from the controller module to the local computer.

To perform a successful upload, verify that the application file includes specific upload-related information as part of the application.

Download DTM-Based Applications

EcoStruxure Control Expert applications that include DTM files use more memory than traditional EcoStruxure Control Expert applications. These products employ DTMs for network configuration:

- BMENOC0302(H) Ethernet communication modules
- 140NOC77101 Ethernet communication module for Quantum
- TSXETC101 Ethernet communication module for Premium
- BMXNOC0401 Ethernet communication module for M340
- 140NOC78•00 Ethernet communication module for Quantum
- BMEP58•0•0 controller modules for M580
- BMEH58•040 controller modules for M580

In some cases, the application configurations created for these modules (and the data associated with them) require more memory than is available in the controller module. In this case, EcoStruxure Control Expert displays a message during the build process, before the application is downloaded to the controller module.

When this situation occurs, change the EcoStruxure Control Expert configuration to exclude additional upload-related information from the application to complete the build and enable the application download:

Step	Action
1	Open the **Project Settings** window (**Tools > Project Settings...**).
2	Expand the structure (**Project Settings > General > PLC embedded data**).
3	In the right pane, deselect the checkbox that corresponds to the **Upload information** row.
4	Click **OK** to save your changes and close the dialog box.

After the **Upload information** setting is disabled, you can build the application and download it to the controller module.

> **NOTE:** You cannot upload an application with a disabled **Upload information** setting from the controller module to the computer.

Upload DTM-Based Applications

DTM-based applications that are successfully downloaded to the controller module (with the project **Upload information** setting enabled) can be uploaded from the controller module to a local computer on which these files are installed:

File	Description
EcoStruxure Control Expert	The EcoStruxure Control Expert version is the same (or a subsequent supporting version) as the version used to create the application. (The configuration includes the DTMs for the modules.)
DTM	The device DTMs for the DTM-based devices are attached to the network. **NOTE:** • Verify that the DTMs are the same version (or a subsequent supporting version) as each device DTM used in the configuration. • Verify that each of the DTM components is installed on the target computer *before* you execute the upload.
EDS	The device EDS files for an EtherNet/IP device are used in the configuration. **NOTE:** Verify that the EDS files are the same version (or a subsequent supporting version) as each device EDS file used in the configuration.

When these files are installed on the target computer, you can upload a DTM-based EcoStruxure Control Expert application from a controller module.

Input and Output Items

Introduction

Create input and output items to support peer-to-peer data transfers between and among scanners. Use the EcoStruxure Control Expert DTM to create input and output items and to define the name and data type of each item.

> **NOTE:** The BMENOC0302(H) module performs the function of a network scanner. You can, however, enable its local slaves to make the module perform the role of an EtherNet/IP adapter. In that case, network EtherNet/IP scanners can read from and write to controller-module data through the enabled local slaves. (Refer to the local slave configuration instructions, page 229.)

Create input and output items in these groups:

- one or more single bits
- 8-bit bytes
- 16-bit words
- 32-bit DWORDS
- 32-bit IEEE floating values

The number of items you create depends upon the data type and size of each item.

Access Items

View the **Items** configuration tabs:

Step	Action
1	Open an M580 project in EcoStruxure Control Expert.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, double-click the DTM that corresponds to the Ethernet communication module.
4	*device connections*: Expand **Device List** and select **Items** for the appropriate connection. *local slaves*: Expand **EtherNet/IP Local Slaves** and select **Items** for the appropriate local slave.

Create Input Items

Create input items:

Step	Action
1	Select the **Input** tab.
2	In the **Default Item Name Root** field, enter a context-sensitive name.
3	Select the first two table rows (0 and 1).
4	Click **Define Item(s)** to open the **Item Name Definition** dialog box.
5	In the **New Item(s) Data Type** field, scroll to **Word**. **NOTE:** The number of selected rows conforms to the data type: • **Byte**: Select a single row. • **WORD**: Select two rows beginning with the next available whole word.
6	Click **OK** to see the new item on the **Input** tab.
7	Click **Apply** to save the item and leave the page open.
8	Repeat these steps to create additional input items that populate the next available row(s) in the table.
9	Save your changes (**File > Save**).

Create Input Bit Items

Create input bit items:

Step	Action
1	Select the **Input (bit)** tab.
2	In the **Default Item Name Root** field, enter a context-sensitive name to monitor the device status.
3	Click the **Define Items** button.
4	Enter a name in the **Item Name** (or accept the default name).
5	Click **OK** to see the new bit item on the **Input** tab.
6	Click **Apply** to save the item and leave the page open.
7	Repeat these steps to create additional input bit items.
8	Save your changes (**File > Save**).

Create Output Items

Create output items:

Step	Action
1	Select the **Output** tab.
2	In the **Default Item Name Root** field, enter a context-sensitive name.
3	Select the first two table rows (0 and 1). **NOTE:** The number of selected rows conforms to the data type: • **Byte**: Select a row. • **WORD**: Select two rows beginning with the next available whole word.
4	Click **Define Item(s)** to open the **Item Name Definition** dialog box.
5	In the **New Item(s) Data Type** field, scroll to **Word**.
6	Click **OK** to see the new item on the **Output** tab.
7	Click **OK** to close the **Items** window.
8	Save your changes (**File > Save**).

Create Output Bit Items

Create output bit items:

Step	Action
1	Select the **Output (bit)** tab.
2	In the **Default Item Name Root** field, enter a context-sensitive name to monitor the device status.
3	Click the **Define Items** button.
4	Enter a name in the **Item Name** (or accept the default name).
5	Click **OK** to see the new bit item on the **Input** tab.
6	Repeat these steps to create additional input bit items.
7	Apply the changes: • Click **Apply** to save the new items and leave the page open. • Click **OK** to save the new items and close the page.

Channel Properties

Overview

This section describes how to configure channel properties for the Ethernet network.

Access Channel Properties

Introduction

The **IP Address Source (PC)** pull-down menu is a list of IP addresses that are configured for a computer that has the EcoStruxure Control Expert DTM installed.

To make a connection, choose an address from this menu that is in the same network as the BMENOC0302(H) module.

Execute these tasks through this connection:

- Perform fieldbus discovery.
- Execute Online Actions.
- Send an explicit message to an EtherNet/IP device.
- Send an explicit message to a Modbus TCP device.
- Diagnose modules.

  NOTE: To establish transparency between a USB connection and a device network, refer to the *Modicon M580, Frequently Used Architectures, System Guide*.

Open the Page

View the **Channel Properties** for the Ethernet communications module:

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the **DTM Browser** (**Tools** > **DTM Browser**).
3	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module.
4	Double-click the name of the BMENOC0302(H) to open the configuration window. **NOTE:** You can also right-click the module and scroll to **Open** to view the configuration window.
5	Select **Channel Properties** in the navigation pane.

Property Descriptions

Select **Channel Properties** in the navigation tree to configure these properties:

Field	Parameter	Description
Source Address	*Source IP Address (PC)*	This is a list of IP addresses assigned to the network interface cards installed on your computer. **NOTE:** If the configured main IP address of the controller module is not in the subnet of an IP configured on the computer's interface cards, the first interface card IP is suggested by default.
	Sub-Network Mask	This read-only address is for the subnet mask associated with the selected source IP address.
EtherNet/IP Network Detection	*Begin detection range address*	The first IP address in the address range for automatic field bus discovery of EtherNet/IP devices.
	End detection range address	The last IP address in the address range for automatic field bus discovery of EtherNet/IP devices.
Modbus Network Detection	*Begin detection range address*	The first IP address in the address range for automatic field bus discovery of Modbus TCP devices.
	End detection range address	The last IP address in the address range for automatic field bus discovery of Modbus TCP devices.

Make the Connection

Connect to the **Source IP Address (PC)**:

Step	Action
1	Select an IP address from the **Source IP Address (PC)** pull-down menu.
2	Click the **Apply** button.
3	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module.
4	Right-click the name of the controller and scroll to **Connect**.

TCP/IP Monitoring

Expand (**+**) the **Channel Properties** heading in the configuration tree and select the **TCP/IP** item at level 1.

The read-only information on this page monitors the IP parameters that were configured in EcoStruxure Control Expert.

Managing Source IP Addresses for Multiple Computers

When you connect a computer to a DTM-based EcoStruxure Control Expert application, EcoStruxure Control Expert requires that you define the IP address of the personal computer (PC) connected to the controller module, which is referred to as the *source IP address (PC)*. Rather than having to perform a **Build** in EcoStruxure Control Expert each time you connect a computer to the controller module, the *source IP address (PC)* is selected automatically when you import the EcoStruxure Control Expert application. During application import, the DTM retrieves the available configured NIC addresses of a connected computer and matches the subnet mask of the master with the available NIC list.

- If a match between the subnet mask of the master and the NIC list exists, EcoStruxure Control Expert automatically selects the matched IP address as the *source IP address (PC)* in the **Channel Properties** page.
- If multiple matches exist, EcoStruxure Control Expert automatically selects the IP address nearest to the subnet mask.
- If no match exists, EcoStruxure Control Expert automatically selects the IP address to the nearest available subnet mask.

Switch Properties

Introduction

Use the **Switch** properties to perform these tasks:

- Enable or disable the Ethernet ports on the BMENOC0302(H) Ethernet communication module.
- View and edit the baud rate for each port, including the transmission speed and duplex mode.

NOTE: The Ethernet communication module supports only the Ethernet II frame type.

Access the Switch Properties

View the **Switch** properties for the BMENOC0302(H) module:

Step	Action
1	To view the **Channel Properties** for the module, open the **DTM Browser** (**Tools > DTM Browser**).
2	Expand (**+**) the **Channel Properties** to see the **Switch** page.
3	Select the **Switch** page to see the configurable properties.

Property Descriptions

Configure the **Switch** properties for the ETH 2 and ETH 3 ports to conform to your application:

Column	Description
Port	This read-only column shows the Ethernet ports that are connected to the module's internal switch (ETH 1, ETH 2, etc.) and the rack port.
Enabled	Scroll to enable (**Yes**) or disable (**No**) a port. **NOTE:** Consider these performance characteristics for the **rack Port** when it is not enabled: • *port blocked:* The port is blocked. However, rack traffic between the controller module and the BMENOC0302(H) remains active for low-level SNAP (high-speed transmission between the controller module and the BMENOC0302(H)), LLDP, and LLC communications. The DDT status of the rack port, therefore, retains a value of *1*, but there is no Ethernet traffic between the front port and the rack of the BMENOC0302(H). • *DIO equipment:* The BMENOC0302(H) module does not manage the DIO equipment on the device network. It can support an isolated DIO network only. • *IPsec:* When you enable IPsec, the DTM automatically disables the rack Ethernet port on the BMENOC0302(H). This isolates the IPsec network (control room network) from the device network.
Baud Rate	Select a baud rate for the enabled port from this pull-down menu. **NOTE:** The available baud rates are listed below.

The baud rate for the enabled rack port is **100 Mbits/sec Full duplex**.

Select a baud rate for an enabled Ethernet port (ETH 1, ETH 2, ETH 3):

- Automatic:
  - Service port (**ETH 1**): 10/100 Mbits
  - **ETH 2**, **ETH 3**: 10/100/1000 Mbits
- 100 Mbits/sec Half duplex
- 100 Mbits/sec Full duplex
- 10 Mbits/sec Half duplex
- 10 Mbits/sec Full duplex

NOTE: Use the default baud rate (**Auto 10/100Mbits/sec**). With this setting, connected devices perform auto-negotiation and thereby determine the fastest common transmission rate and duplex mode.

TCP/IP Properties

Introduction

The read-only table on the **TCP/IP** page shows the IP parameters that were configured in EcoStruxure Control Expert.

Use the channel **Configuration** tab of the module to perform these tasks:

- Select a configuration mode to specify the manner in which the Ethernet communication module obtains its IP addressing settings.
- Edit the IP addressing settings when the configuration mode is set.

Access the Configuration Tab

Access the channel **Configuration** tab for the BMENOC0302(H) Ethernet communications module:

Step	Action
1	In the **Project Browser**, double-click **Project > Configuration > PLC bus**.
2	In the **PLC Bus** dialog box, right-click the BMENOC0302(H) module and click **Open** to access the module's configuration window.
3	Select **Channel 0** to view the **Configuration** tab.

Select a Configuration Mode

In the **IP Address configuration** part of the **Configuration** tab, select one of these modes to determine how the communication module obtains its IP address at startup:

Mode	Description	Standalone	Hot Standby
Static	The module uses the scanner IP address, gateway IP address, and sub-network mask configured in this page.	✓	✓
BOOTP	The module uses an IP address assigned by a BOOTP server.	✓	—
DHCP	The module uses an IP address assigned by a DHCP server.	✓	—
✓ : Supported – : Not supported			

Configure the parameters that are available in each *Configuration Type*:

Type	IP Parameter	Description
Static	*Main IP Address*	This 32-bit identifier consists of a network address and a host address that are assigned to a device that is connected to a TCP/IP Internet network through the Internet Protocol (IP). **NOTE:** In Hot Standby systems, the primary controller is assigned the *Main IP Address* and the standby controller is assigned *Main IP Address + 1*. After a switchover, the new primary controller (former standby) is assigned the *Main IP Address* and the standby controller is assigned *Main IP Address + 1*.
	Main IP Address +1	This address is automatically assigned to the standby controller module during a Hot Standby switchover.
	Enable IP A/B	Select (check) this functionality and configure the address fields for both (primary and standby) controller modules: • **IP address A:** Enter a fixed IP address. • **IP address B:** Enter a fixed IP address. **NOTE:** Refer to the independent descriptions of IP addresses A and B, page 98, and Hot Standby Considerations, page 98.
	Subnet Mask	This 32-bit value hides (or masks) the host portion of the IP address to set the network address of the module.
	Gateway	When required by the application, this device address serves as a gateway to other parts of the network.
	Automatic Device(s) subnet update (checkbox)	The subnet mask of Modbus devices behind a BMENOC0302(H) module is automatically updated if the NOC subnet is changed (for example during commissioning or maintenance). When deselected (unchecked), the subnets for the devices are unchanged.

Type	IP Parameter	Description
DHCP	*Main IP Address*	This 32-bit identifier consists of a network address and a host address that are assigned to a device that is connected to a TCP/IP Internet network through the Internet Protocol (IP).
	Subnet Mask	This 32-bit value hides (or masks) the host portion of the IP address to set the network address of the module.
	Gateway	When required by the application, this device address serves as a gateway to other parts of the network.
	DHCP Server IP address	Click this button to populate the *Main IP Address* field with the new parameters after the module obtains valid IP parameters from the requested server. **NOTE:** • This value is optional. • If no value is set, the first received DHCP offer is accepted. • If a value is entered, the client accepts only a DHCP offer from this IP address.
	Backup DHCP server IP address	Enter the address of a DHCP server in the local subnet or a remote subnet (through a DHCP relay).
	DHCP Identifier	This auto-generated field is in the format `8M_#_modulename`, for which: • `8M` is a constant. • `#` is a one- or two-digit slot number. • `modulename` represents the module type (for example, `BMENOC0302`). **NOTE:** You can edit this auto-generated identifier.
BOOTP	*Main IP Address*	This 32-bit identifier consists of a network address and a host address that are assigned to a device that is connected to a TCP/IP Internet network through the Internet Protocol (IP). **NOTE:** • When the Ethernet communications module cannot reach a BOOTP server, use the default IP address in this format: `10.10.MAC5.MAC6` • The module uses the default address until it obtains valid IP parameters from the server.
	Subnet Mask	This 32-bit value hides (or masks) the host portion of the IP address to set the network address of the module.
	BOOTP Server IP address	Enter the address of a BOOTP server in the local subnet or a remote subnet (through a DHCP relay).

IP Addresses A and B

When your connect EcoStruxure Control Expert to a Hot Standby system, you can use **IP Address A** or **IP Address B** from the TCP/IP configuration (above) to access the controller. These independent addresses are configured on the controller for redundancy or Hot Standby purposes.

The two addresses apply to the two modules that can be either the primary or standby controllers. You do not assign the primary and standby roles here. Typically, the first controller to power up becomes the primary controller, regardless of its designation as A or B; the secondary controller to power up becomes the standby.

Considerations:

- The functionality of **IP Address A** or **IP Address B** is automatically disabled when the cybersecure IPsec protocol is enabled in a Hot Standby system. These address assignments are not affected by a change in the controller's primary or standby status.
- When **IP Address A** or **IP Address B** is enabled, NTPv4 clients in the primary and standby controllers get their times from an NTP server without closing the re-opened connection during the Hot Standby switchover.

Typically, the first controller to power up becomes the primary controller, regardless of its designation as *A* or *B*; the secondary controller to power up becomes the standby.

Hot Standby Considerations

In a Hot Standby system, distributed equipment uses the **Main IP address** setting of the controller module to communicate with the primary controller module over an Ethernet network.

NOTE:

- Refer to the *Modicon M580 Hot Standby, Frequently Used Architectures, System Guide*.
- Configure the *Main IP Address* in the **IP Config** tab for the M580 controller module. (Refer to the *Modicon M580 Hot Standby, Frequently Used Architectures, System Guide*.)

On switchover, the *Main IP Address* setting is automatically transferred from the former primary controller module to the former standby controller module (now the new primary controller module). Similarly, on switchover, the *Main IP Address +1* setting is automatically transferred from the former standby controller module to the new standby controller module.

In this way, the configured links between the distributed equipment and the primary controller module do not require editing in the event of a switchover.

Default Address Configurations and Software Default Address

The communications module uses a default address when it is not configured or when a duplicate IP address is detected. The default address is based on the MAC address of the module and makes it possible for several Schneider Electric devices to use their default network configuration on the same network.

The default address is 10.10.MAC5.MAC6, wherein MAC5.MAC6 represents the last numbers of the module MAC address. When a module is added to the controller configuration in an M580 local rack, EcoStruxure Control Expert generates a default IP configuration that observes these rules. Verify that the generated IP parameters are appropriate to your application requirements, or edit the addresses as needed.

The Ethernet communication module provides these basic services when it uses the default IP address (and the services are enabled in the configuration):

- FTP server
- HTTPS/Web server
- Modbus TCP server
- EtherNet/IP explicit message server
- SNMPv1/SNMPv3 agent
- RSTP

Duplicate Address Checking

The module looks for duplicate IP addresses before it applies the configured IP address:

Response	Meaning
yes	Another network device is using the proposed IP address.
	The module does not use the proposed IP address. It uses the default IP address.
no	The module uses the proposed IP address (along with the associated network parameters).

To improve performance during a network power-up operation, power up the network switches before you power up a system component (Ethernet communications module, Modicon M580 rack, controller module, and so forth.).

NOTE: When the entire network powers up at once, some switches may be slower to complete the process. The relatively slow switch response can cause some ARP messages to be dropped, resulting in an incomplete detection of duplicate IP addresses.

Ethernet Services

Enable and Disable Ethernet Services

Introduction

The BMENOC0302(H) Ethernet communications module provides several Ethernet services. Use the **Services** page in the EcoStruxure Control Expert DTM to enable and disable those services.

Enable and Disable Services

View the module **Services**:

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, find the name that you assigned to the module.
4	Double-click the module name to open the configuration window. **NOTE:** You can also right-click the module and scroll to **Open** to open the configuration window.
5	Select **Services** in the navigation tree.
6	Enable or disable each feature in the pull-down menu: • **Enabled:** Scroll to **Enabled** to enable a service in the appropriate pull-down menu. • **Disabled:** Scroll to **Disabled** to disable the service.
7	Click a button: • **Apply**: Save changes with the window open. • **OK**: Save changes and close the window.
8	Expand (**+**) **Services** in the navigation tree to view the enabled services.

NOTE:

- The available services appear in the expanded **Services** tree when they are enabled.
- You can configure the settings for an enabled service. If you enable a service that you do not configure, the EcoStruxure Control Expert DTM applies the default settings.

Available Services

These Ethernet services are provided by the BMENOC0302(H) Ethernet communications module:

Service	Description	Default
Address Server, page 103	Provide IP addressing parameters and operating parameters to other Ethernet devices.	enabled
SNMP, page 107	• Serve as an SNMPv1 or SNMPv3 agent. • Provide trap information for one or two devices that are configured as SNMP managers.	enabled
RSTP, page 109	Employ RSTP in combination with other similarly-configured network devices to manage redundant physical connections and create a loop-free logical path that connects network devices.	enabled
QoS, page 112	Add DSCP tags to Ethernet packets so that network switches can prioritize the transmission and forwarding of Ethernet packets. **NOTE:** Before you enable tagging, verify that the devices connected to the Ethernet communication module support QoS DSCP tagging, page 112.	enabled
Service Port, page 114	The system supports the connection to a network through the service port.	enabled

FDR Address Server

About the FDR Service

The Ethernet communications module includes a fast device replacement (FDR) server. That server provides operating parameter settings to replacement Ethernet devices that are equipped with FDR client functionality.

A networked Ethernet device that is equipped with FDR client functionality can subscribe to the Ethernet communications module's FDR service. The module can store up to 1 MB of FDR client operating parameter files. When this file storage capacity is reached, the module cannot store additional client FDR files.

The Ethernet communications module can store FDR client files for up to 128 devices, depending on the size of each stored file. For example, if the size of each FDR client file is small (not more than 8 KB) the module can store a maximum of 128 parameter files.

In an M580 Hot Standby system, the parameter configuration (.prm) files that are managed by the FDR server in both controller modules are synchronized when the applications in both of those modules match.

NOTE: Refer to the discussion of FDR in Hot Standby systems in the *Modicon M580 Hot Standby, Frequently Used Architectures, System Guide*.

Configure the FDR Address Server

Configure the address server service with the EcoStruxure Control Expert DTM to set IP parameters for an Ethernet device that is based on a unique name (device name) or the MAC address of the device:

Step	Action
1	Enable the **Address Server** in the **Services** configuration, page 101.
2	Expand (**+**) **Services** and select **Address Server**.
3	In the **FDR Server** menu, scroll to **Enabled** to enable the server.
4	View these tables: • **Automatically Added Devices:** This table shows the devices (and the corresponding IP addresses) that are automatically included in the module configuration. • **Manually Added Devices:** This table shows the devices (and the corresponding IP addresses) that you add to the module configuration. **NOTE:** • The addition of devices to these tables is described below. • The same IP address cannot appear in both the **Manually Added Devices** table and the **Automatically Added Devices** table.
5	Click a button to finish: • **Apply**: Save changes with the window open. • **OK**: Save changes and close the window.

This service also enables a device to store the configuration of the communications module in local non-volatile memory. The address server automatically provides correct network and device parameters for replacement devices without stopping the process.

View the Auto-Generated Client List

The **Automatically Added Devices** table automatically displays a list of devices that fit these criteria:

- The devices correspond to devices in the **Device List**.
- The devices subscribe to the Ethernet communications module's IP addressing service.

NOTE: You cannot add devices to this table. Instead, use the configuration pages for the remote device to subscribe to this service.

These columns appear in the **Automatically Added Devices** list:

Column	Description
Device No	This number is assigned to the device in the EcoStruxure Control Expert configuration.
IP Address	This address corresponds to the client device.
DHCP	**TRUE** indicates that the device subscribes to the DHCP service.
Identifier Type	*Identifier Type*: This is the type of value that the client device uses to identify itself to the FDR server: • MAC address • Device Name
Identifier	This is the MAC address or device name.
Netmask	This is the subnet mask of the client device.
Gateway	This is the IP address of the network device that a DHCP client device uses to access other devices that are not located on the local subnet. A value of *0.0.0.0* constrains the DHCP client device by restricting its communications to devices on the local subnet.

Manually Add Remote Devices to the DHCP Service

The **Manually Added Devices** table is a list of device DTMs for devices in the **Device List**. Devices in this table are equipped with DHCP or BOOTP client software.

Click the **Add** button to add a device DTM to the table and configure these parameters for that client device:

Parameter	Description
IP Address	Double-click the cell in the *IP Address* column and enter an IP address for the client device.
Identifier Type	Scroll to the type of value that the client device uses to identify itself to the FDR server (*MAC Address* or *Device Name*).
Identifier	Depending upon the identifier type, enter the client device setting for the MAC address or name.
Netmask	Enter the client device subnet mask.
Gateway	Enter the gateway address that remote devices can use to communicate with devices located on other networks. Use *0.0.0.0* if remote devices do not communicate with devices on other networks.

Example: A DHCP Server Provides IP Addresses for Local and Remote Subnets

To view an example of a DHCP server configuration that provides IP addresses to devices in local and remote subnets, refer to the description of the Device DDT for the Ethernet communications module, page 257.

SNMP Agent (SNMPv1, SNMPv3)

Introduction

The BMENOC0302(H) Ethernet communications module supports SNMPv1 or SNMPv3 agents. You can select and configure the agent type that is most appropriate for your particular application.

An SNMP agent is a software component that runs on the communication module to support access to the module's diagnostic and management information through the SNMP service. SNMP browsers, network management software, and other tools typically use SNMP to access this data.

As part of the SNMP service, you can configure the SNMP agent with the IP addresses for two SNMP manager devices (typically computers that run network management software). Those devices can then receive event-driven trap messages that contain reports of events such as cold starts and unauthorized access.

Access the SNMP Page

Access the **SNMP** page to configure the SNMP agent in the BMENOC0302(H) module:

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module
4	Double-click the name of the BMENOC0302(H) to open the configuration window. **NOTE:** You can also right-click the module and scroll to **Open** to open the configuration window.
5	Expand (**+**) **Services** in the navigation tree.
6	Select **SNMP** to see the configuration options.

SNMP Properties

Edit the available SNMPv1 or SNMPv3 properties on the **SNMP** page:

Property		Description
Configuration Type	*SNMP V1*	Select the radial button for the version (**SNMP V1** or **SNMP V3**) that is most appropriate for your particular application.
	SNMP V3	
IP Address Managers	*IP Address Manager 1*	This is the IP address of the first SNMP manager to which the SNMP agent sends trap notices.
	IP Address Manager 2	This is the IP address of the second SNMP manager to which the SNMP agent sends trap notices.
Agent	*Location (SysLocation)*	This is the device location (32 characters maximum).
	Contact (SysContact)	This field contains the name of the person to contact for device maintenance (32 characters maximum),
	SNMP User	*Disabled*: In this case you can configure the *Location (SysLocation)* and *Contact (SysContact)* settings.
		Enabled: In this case you cannot edit the *Location (SysLocation)* and *Contact (SysContact)* settings. (Those settings are managed by the SNMP Manager.)
Community Names (SNMPv1 only)	*Set*	The SNMP agent requires the password to execute write commands from an SNMP manager (default = **private**).
	Get	The SNMP agent requires the password to execute read commands from an SNMP manager (default = **public**).
	Trap	The SNMP agent requires the password to accept trap notices from the agent (default = **alert**).
Security (SNMPv1 only)	*Enable "Authentication failure" Trap*	**Enabled:** The SNMP agent sends trap notices to the SNMP manager if an unauthorized manager sends a Get or Set command to the agent.
		Disabled (default)**:** This function is disabled.
SNMP User (SNMPv3 only)	*SNMP User Name*	This is the name of the SNMP user.

Click a button to apply the configuration:

- **Apply**: Save changes.
- **OK**: Save changes and close the window.

Rapid Spanning Tree Protocol (RSTP)

Introduction

The Ethernet DEVICE NETWORK ports (**ETH 2**, **ETH 3**) on the front of the BMENOC0302 (H) Ethernet communications module support the *Rapid Spanning Tree Protocol*. RSTP is an OSI layer 2 protocol defined by IEEE 802.1D 2004 that performs these services:

- RSTP creates a logical network path for Ethernet devices that are part of a topology that includes redundant physical paths.
- RSTP automatically restores network communications by activating redundant links when a network event causes an interruption in service.

NOTE: When an RSTP link is connected in an Ethernet RIO network, the RSTP service acts on an event and forwards traffic through the correct port. During this re-connect time, some packets may be lost.

RSTP software, operating simultaneously in all network switches, obtains information from each neighboring switch to enable the software to create a hierarchical logical network topology. RSTP is a flexible protocol that can be implemented on many physical topologies, including ring, mesh, or a combination of ring and mesh.

NOTE: RSTP can be implemented when every network switch is configured to support RSTP.

NOTE: In a redundant architecture, configure all the BMENOC0302(H) modules as RSTP participant.

Recovery Time

When forty BMENOC0302(H) modules are configured at 100 Mbit/sec full-duplex, the RSTP recovery time is less than 100 milliseconds. In this case, configure the BMENOC0302(H) modules at 100 Mbit/sec full-duplex for RSTP for the fastest recovery time.

NOTE:

- If you add a device with a baud rate in auto-negotiation of 10/100 Mbit/sec, the connection switches to half-duplex.
- A configured RSTP speed of 1 Gbit/sec may yield slow recovery times (up to several seconds) owing to the technical limitations of the hardware.

Access the RSTP Page

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Enable **RSTP** in the **Services** configuration, page 101.
3	Open the **DTM Browser** (**Tools > DTM Browser**).
4	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module.
5	Double-click the name of the BMENOC0302(H) to open the configuration window. **NOTE:** You can also right-click the module and scroll to **Open** to open the configuration window.
6	Expand (**+**) **Services** in the navigation tree.
7	Select **RSTP** to see the **General** and **Advanced** configuration tabs. **NOTE:** The **Advanced** tab is available when you enable the DTM's Advanced Mode, page 77.

Assign the Bridge Priority

The bridge priority is a two-byte value for the switch. The range for valid values is 0 ... 65535, with a default value of 32768 (the midpoint).

Select the **General** tab to configure the Bridge Priority:

Step	Action
1	Make a selection in the **Bridge Priority** pull-down list in the **RSTP Operational State** group box: • **Root (0)** • **Backup Root (4096)** • **Participant (32768)** (default)
2	Finish the configuration: • **OK**: Assign the *Bridge Priority* and close the window. • **Apply**: Assign the *Bridge Priority* and keep the window open.

NOTE: The *Bridge Priority* value establishes the relative position of the switch in the RSTP hierarchy.

Advanced Configuration

Configure these parameters on the **Advanced** tab:

Field	Property	Description
Bridge Parameters	**Maximum Age Time**	The switch waits this length of time (6 ... 40 sec) for receipt of the next hello message before it initiates a change to the RSTP topology. (Default = 40 sec.)
	Transmit Hold Count	The maximum number of BPDUs (1 ... 40) that the switch can transmit per second. (Default = 40.)
	Hello Time	The embedded switch sends heartbeat BPDUs at this (read-only) frequency (2 sec).
Port Parameters (ETH 2, ETH 3)	**RSTP**	This (read-only) property is set to **Enabled** in the **Services** page.
	Priority	The priority assigned to the switch port, an integer from 0 to 240 in increments of 16. (Default = 0.) The RSTP process uses this value to break a tie between two ports on the same switch when it identifies the port type: • *root port:* This port on a non-root switch is closest to the root bridge in terms of path cost. • *designated port:* Traffic passes through this port at one end of a network segment on its way to the root bridge.
	RSTP Cost	Select a method to determine the RSTP cost of the path through the embedded switch: • **Auto**: The RSTP protocol automatically assigns a value to the switch by operation of the RSTP algorithm. • **Manual**: Input the RSTP cost integer (1 ... 200000000) in the **Value** field.
	Edge Port	Set to a fixed (read-only) value of **Auto**. The RSTP process automatically determines if the port is an RSTP edge port.
	Point to Point	Set to a fixed (read-only) value of **Auto**. The RSTP process automatically determines if the port is an RSTP point-to-point port.

DSCP Values for QoS

Description

The BMENOC0302(H) Ethernet communication module can be configured to use the Different Service Code Point (DSCP) service in the IP packets. When you enable QoS, the module adds a DSCP value to the IP header of the Ethernet frame to indicate the frame priority.

NOTE: The BMENOC0302(H) module supports the OSI layer 3 Quality of Service (QoS) standard defined in IEEE RFC 2475.

Use the **QoS** page to view or edit the QoS DSCP prioritization values.

Configure the QoS Service

Step	Action
1	Enable **QoS** in the Ethernet **Services** configuration, page 101.
2	Expand (**+**) the **Services** page to see **QoS** in navigation tree.
3	Select the **QoS** node to see the configurable parameters.
4	Enter changes in the appropriate fields on the **QoS** configuration page. (The table below describes the traffic settings.)
5	Click a button to finish: • **Apply**: Save changes with the window open. • **OK**: Save changes and close the window.

QoS Settings

Use these guidelines to effectively implement QoS settings in your Ethernet network:

- Use network switches that support QoS.
- Consistently apply DSCP values to network devices and switches that support DSCP.
- Verify that switches apply a consistent set of rules for sorting DSCP tags when transmitting and receiving Ethernet packets.

Use the EcoStruxure Control Expert DTM to configure these values for *EtherNet/IP Traffic*, *Modbus TCP Traffic*, and *Network Time Protocol Traffic*:

Field	Traffic	Default
EtherNet/IP Traffic	DSCP Value for I/O Data Scheduled Priority Messages	47
	DSCP Value for Explicit Messages	27
	DSCP Value for I/O Data Urgent Priority Messages (See the note below.)	55
	DSCP Value for I/O Data High Priority Messages (See the note below.)	43
	DSCP Value for I/O Data Low Priority Messages (See the note below.)	31
Modbus TCP Traffic	DSCP Value for I/O Messages	43
	DSCP Value for Explicit Messages	27
Network Time Protocol Traffic	DSCP Value for Network Time Protocol Messages	59
NOTE: These fields are available only when you enable **Advanced Mode**, page 77.		

Service Port

Introduction

Use these instructions to configure the ETH 1 port on the front of the BMENOC0302(H) Ethernet communications module as an access port or port mirroring port.

When configured for port mirroring, the communications module can monitor the functionality of network links depending on which links are connected to the network.

Access the Service Port Configuration Page

Enable **Service Port** configuration:

Step	Action
1	Enable the **Service Port** in the Ethernet **Services** configuration, page 101.
2	Expand (**+**) the **Services** page to see the **Service Port** parameters.
3	Select **Service Port** in the navigation tree.
4	In the **Service Port Mode** pull-down menu, select **Access Port** or **Port Mirroring** mode. (These modes are discussed in detail below.)
5	Click a button to finish: • **Apply**: Save changes with the window open. • **OK**: Save changes and close the window.

Access Port Mode

In **Access Port** mode (the default selection), the **ETH 1** port is set to **Enabled**. (You cannot disable this port.) In this mode you can connect these types of devices to the **ETH 1** port:

- HMI
- a computer with EcoStruxure Control Expert software

You can communicate with the controller module or the BMENOC0302(H) module itself. You can also access other devices that are connected to the network.

Port Mirroring Mode

Select the **Port Mirroring** mode to configure the port to monitor and capture traffic to support a network analyzer (like Wireshark). In this mode, the SERVICE port is a read-only port. That is, you cannot communicate with Ethernet devices through the SERVICE port.

In the **Port Mirroring Configuration** table, use the **Source Port(s)** property to enable specific ports:

- **Yes**: Traffic to and from this port is mirrored to the SERVICE port.
- **No**: Traffic to and from this port is not monitored by the SERVICE port.

The SERVICE port monitors traffic to the enabled ports:

Source Port	Description
Internal Port	Monitor Ethernet traffic to and from the module through the SERVICE port.
ETH 2	Ethernet traffic to and from port **ETH 2** is sent to the SERVICE Port.
ETH 3	Ethernet traffic to and from port **ETH 3** is sent to the SERVICE Port.
Rack Port	Ethernet traffic to and from the rack port is sent to the SERVICE Port.

NOTE: If a device that is connected to the SERVICE port is configured for a speed that exceeds 100 Mbps, the Ethernet link may not be established between the device and the module through the SERVICE port.

Hot Standby Configuration for the Service Port

In an M580 Hot Standby configuration, some topologies may unintentionally create a loop that interferes with network communication. These topologies are essentially related to the management of flat network.

To help avoid the creation of an unintentional loop that is caused by a connection to the SERVICE port, select (check) the **Automatic blocking of service port on NOC** box in the **ServicePort** tab in the configuration dialog box.

No loop condition can exist on the Ethernet rack connection by connecting one or more BMENOC0302(H) modules to the Ethernet rack of an M580 Hot Standby configuration. This module automatically blocks its rack port on the standby local rack.

The BMENOC0302(H) module automatically blocks its rack port on the standby local rack.

To see topology examples in which this issue exists, refer to the ServicePort configuration instructions, page 114.

Advanced Settings for EtherNet/IP

Introduction

The EtherNet/IP **Advanced** tab is available for Ethernet communication modules that support the DIO scanner service.

Access the Advanced Tab

View the EtherNet/IP **Advanced** tab:

Step	Action
1	Close configuration windows associated with the Ethernet communication module.
2	Find the Ethernet communication module in the EcoStruxure Control Expert **DTM Browser**.
3	Right-click the module and scroll to **Device menu > Additional functions > Advanced Mode**.
4	Double-click the module in the **DTM Browser** to view the **Channel Properties**.
5	Expand (**+**) **Channel Properties**.
6	Select **EtherNet/IP** to view the items in the **Group/Parameter** column: • **Timeout**: EtherNet/IP timeout settings • **Behavior**: EtherNet/IP scanner behavior

Timeout Settings

These timeout settings are in the EtherNet/IP **Timeout** field:

Parameter	Comment	Default
FW_Open I/O Connection Timeout (msec)	Configure the amount of time the scanner waits for a FW_Open response for an I/O connection.	4960
FW_Open EM Connection Timeout (msec)	Configure the amount of time the scanner waits for FW_Open response for an explicit message connection.	3000
EM Connection RPI (msec)	Set T->O and O->T RPI for all explicit messaging connections.	10000
EM Request Timeout (sec)	Configure the amount of time the scanner waits between the request and the response of an explicit message.	10

Scanner Behavior

Configure the behavior of the DIO scanner in the EtherNet/IP **Behavior** field:

Parameter	Value	Comment
Allow RESET via explicit message	*False*	(Default.) The scanner ignores the Identity object reset service request.
	True	The scanner resets if an Identity object reset service request is received.
Behavior when CPU state is STOP	*Idle*	(Default.) The EtherNet/IP I/O connection stays open, but the **Run/Idle** flag is set to Idle.
	Stop	The EtherNet/IP I/O connection is closed.

Security

IP Secure Communications

Introduction to IPsec

IPsec (Internet Protocol Security) is an open set of protocol standards that make IP communication sessions private and encrypted. Its authentication and encryption algorithms require user-defined cryptographic keys that process communications packets during an IPsec session.

NOTE: The Internet Engineering Task Force developed and designed IPsec. For more information, refer to that organization website (www.IETF.org).

BMENOC0302(H) Support for IPsec

A BMENOC0302(H) Ethernet communications module can serve as an IPsec initiator or responder when these requirements are met:

IPsec Feature	Requirement
transmission mode	IPsec uses transport mode for data flows that are local to the module.
IKE version	IPsec uses the *Internet Key Exchange Version 2* (IKEv2) protocol to establish secure internet connections between VPN clients and servers.
key encryption	IPsec uses AES-128CBC to encrypt the key exchange.
secure hashtag	IPsec uses the SHA-256 cryptographic hash function.
secure protocol	IPsec uses a Diffie-Hellman (DH) key exchange algorithm with a key size of 2048 bits.
administrative privileges	You have the administrative rights to execute the script.
command line	You can run scripts within Windows PowerShell.
confidentiality	IPsec supports Encapsulating Security Payload (ESP) with and without confidentiality in accordance with the user configuration in the DTM.
multiple channels	The module supports a maximum of eight simultaneous IKEv2 IPsec connections.
open channels	The BMENOC0302(H) module can open these IPsec channels: • Open a channel to a computer that runs the Windows 10 or Windows 11 operating system. • Open a channel to another BMENOC0302(H) module.

IPsec Feature	Requirement
computer hardware	At least one Windows computer can generate or sign certificates using the scripts.
SSL protocol instance	OpenSSL is installed on your system and configured.
user privileges	You have obtained the administrative rights to execute the script.
script running	The computer and OS can run scripts with Windows PowerShell.
time synchronization	The time for the BMENOC0302(H) module is synchronized to meet the certificate validity requirements.

NOTE: To obtain these permissions and installation requirements, contact your local IT department or the service responsible for security policies.

Limitations

The BMENOC0302(H) Ethernet communications module observes these limitations in regard to IPsec functionality:

- *tunnel mode:* The BMENOC0302(H) module cannot create secure connections through tunnel mode for IPsec.
- *PSK authentication:* The BMENOC0302(H) module does not support pre-shared key authentication and therefore does not support IPsec communications with BMENOC0301, BMENOC0311, BMENOC0321, or BMENUA0100 modules.
- *implicit messages:* The BMENOC0302(H) module does not support implicit communications (EtherNet/IP scanner, Modbus TCP scanner) through IPsec. It supports only explicit peer-to-peer communications between itself and a Windows (v10, v11) computer or between itself and another BMENOC0302(H) module.
- Communication interruptions in explicit messaging are to be managed in the EcoStruxure Control Expert application as the maximum number of EFs that can be simultaneously used with IPsec communications is lower than without IPsec communications.
- An interval of at least 5 minutes is required between consecutive hot swap operations or power cycles.

Prerequisites

Manually configure each computer that supports IPsec:

- These directions apply to computers that run Windows versions 10 or 11 (or a subsequent supporting version).
- Install the appropriate version of EcoStruxure Control Expert.

  NOTE: The BMENOC0302(H) module supports IPsec communications in EcoStruxure Control Expert 16.2 (or a subsequent supporting version).
- Harden the computer that hosts the IPsec client to decrease the attack surface and observe the defense-in-depth concept.

  NOTE: Refer to the *Hardening the PC* discussion in the *Modicon Controller Systems Cybersecurity, User Guide*.

Configuration Process Overview

The IPsec configuration includes these stages:

Stage	Description
1	Configure IPsec in the EcoStruxure Control Expert DTM, as shown below.
2	Configure the Windows firewall to use IPsec, page 122.
3	Verify that the connection is valid, page 131.

Open the IPsec Configuration Screen

Open the IPsec configuration for the BMENOC0302(H) module in EcoStruxure™ Control Expert:

Step	Action
1	Open the **DTM Browser** in EcoStruxure™ Control Expert (**Tools > DTM Browser**).
2	Right-click the name that you assigned to the BMENOC0302(H) module, and select **Open**.
3	Select **Security** in the navigation pane.
4	In the **Services** group box, select **Enabled** from the **IPsec IKev2** pull-down menu.
5	Click the **Apply** button to apply IPsec to the application.
6	Expand **Security** in the navigation pane to see **IPSec** in the structure.
7	Select **IPSec** in the structure to open the **IPSec Service** configuration screen.

Configure IPsec Parameters

Configure these parameters in the **IPSec Service** group box:

Parameter		Description
NTP authorized outside IPSEC		*disabled:* De-select (uncheck) this box to exchange NTP packets through IPsec only.
		enabled: Select (check) this box to authorize the NTP client in the module to communicate with the remote NTP server (that is, not IPsec capable) outside IPsec even though IPsec is enabled in this module.
IPSec Links	**PC**	• **Yes:** This address is configured for a Windows computer. • **No:** This link is associated with a connection to a device that is not identified as a Windows computer. **NOTE:** • Your IPsec configuration requires at least one Windows computer to run the certificate and make the IPsec connection. • This column identifies the designated IPsec link for the Windows computer when it configures the script. It is not applicable to the Ethernet communications module.
	Remote IP Address	This column shows the IPv4 address of the remote device at the other end of the IPsec connection.
	Confidentiality	• **Yes:** Communications are encrypted for the remote IP address. • **No:** Communications are not encrypted for the remote IP address.
	Update	Click this button to confirm the new link.
	Remove Link	Remove the selected remote link from the **IPSec Links** table.
Add Link	**Remote IP Address**	This column contains the IPv4 address for the remote device at one end of the IPsec connection.
	Confidentiality	• **Yes:** Communications are encrypted for the remote IP address. • **No:** Communications are not encrypted for the remote IP address.
	PC	• **Yes:** Communications with the Windows computer are encrypted. • **No:** Communications with the Windows computer are not encrypted.
	Download Script	Click this button to download the Windows script for remote IP addresses to configure the Windows firewall settings, page 122. If the IPsec setting is changed for some protocols, download the Windows script again to execute it on Windows. **NOTE:** • If these buttons are not visible, use the scroll bar to view the far-right portion of the *IPSec* window. • This script is generated by DTM when you configure IPsec, page 118.

Parameter	Description
IKEv2	**Is PKI Enrolment Available:** • **Yes:** PKI is available. • **No:** PKI is not available. **NOTE:** • The *Public Key Infrastructure* (PKI) uses certificates to verify the identity of remote addresses to secure communications. • This selection affects the script version that is downloaded when you click the **Download Script** button.
	Root CA Certificate Thumbprint: Configure the Certificate Authority thumbprint that corresponds to your specific PKI tool. **NOTE:** • The Certificate Authority (CA) is a trusted issuer of the certificates used in the identification and encryption of IPsec tunnels. • The CA thumbprint is a unique identifier for the individual digital certificate. • When PKI enrollment is available, the thumbprint comes from the CA that signs certificate Signing Requests (CSRs).

After you configure these parameters, click the **Apply** button finish the security configuration for the BMENOC0302(H) module.

IKE/IPsec Windows Firewall Configuration Scripts

To run IPsec on a Windows computer that hosts either the EcoStruxure™ Control Expert configuration software, add a network configuration to the host firewall.

The following examples present Windows firewall configuration scripts with and without IPsec confidentiality. Each script example includes values for these variables:

- `endpoint1`: This is the remote IP address in the IPsec configuration.
- `endpoint2`: This is the address of the control port in the BMENOC0302(H) module.

Create an IPsec Channel

Use these directions to create an IPsec channel between the Windows computer and the BMENOC0302(H) module.

Use the instruction tables below to download a Certificate Signing Request for each BMENOC0302(H) module in the EcoStruxure Control Expert application.

Prepare for the download:

Step	Action
1	In EcoStruxure Control Expert, transfer an application that includes at least one BMENOC0302(H) module to the controller to allow access to the Modicon M580 BMENOC0302(H) website.
2	Log in to the website with these credentials: • *username:* loader • *password:* Use the password set in the EcoStruxure Control Expert application for **Firmware**.
3	Create a local folder with a unique name (example: *Test_IPsec*). **NOTE:** • Script operations occur in this folder. • Start with this new and empty folder.
4	Inside the folder you just created (*Test_IPsec*), create a folder named *CSRs*. **NOTE:** The Certificate Signing Requests (CSRs) that are hereafter created are saved to the *CSRs* folder. CSRs are signed messages to the CA from an identified device that requests a certificate for authenticating IPsec connections.
5	Expand **Security** in the module DTM navigation pane and select **IPsec**.
6	Prepare the IPsec configuration in the EcoStruxure Control Expert application: • Add the required links. • Verify that the connection to the Windows computer (**PC**) is configured for IPsec.
7	Continue to the instructions that correspond to the availability of PKI: • **PKI is available** • **PKI is not available** **NOTE:** The availability of PKI corresponds to the configuration of the **IKEv2** parameter (**Is PKI Enrolment Available**).

PKI is available. Use these instructions when PKI enrollment is *available*:

Step	Action
1	Configure the ROOT CA thumbprint to match the CA to be used by PKI in following steps.
2	Download the EcoStruxure Control Expert script (.ps1) from the **IPSec Service** configuration screen. **NOTE:** For those infrequent cases in which the script is not directly downloaded from a EcoStruxure Control Expert instance that runs on your Windows computer, you may be required to perform *one* of these tasks: • Right-click the icon for the script file, scroll to **Properties**, and select the **Unblock** checkbox. • Run this command: `Set-ExecutionPolicy Unrestricted -Scope Process` ◦ Run this command from any directory. ◦ Owing to the performance characteristics of Windows PowerShell sessions, it is a good practice to run this command immediately before you run the script (in the next step).
3	Open Windows PowerShell with administrative rights and run the downloaded script file (.ps1). **NOTE:** • The execution of the script configures a channel for IPsec transport mode. • Connections to the website are not affected because inbound and outbound connections are made only by request. • You are not required to run the script from the *CSRs* folder that contains the script because the script generates a request file (request.req) only for the certificate for the Windows computer.
4	Copy the downloaded CSR files and the request file (.req) generated by the script and paste them into the directory on the windows computer that hosts the PKI.
5	Create signed certificates for the BMENOC0302(H) module(s) and Windows computers that are applicable to the IPsec communications.
6	Install the certificate for the Windows computer that is generated by the PKI tool on the Windows computer for which IPsec communications will be established. **NOTE:** The certificate appears in the computer’s *Certificates* directory (*Certificates - Local Computer* > *Personal* > *Certificates*).
7	Skip the next table (**PKI is not available**) and continue to the instructions to **Resume the configuration**.

PKI is not available. Use these instructions when PKI enrollment is *not available*:

Step	Action
1	Download the EcoStruxure Control Expert script from the **IPSec Service** configuration screen. **NOTE:** For those infrequent cases in which the script is not directly downloaded from a EcoStruxure Control Expert instance that runs on your Windows computer, you may be required to perform *one* of these tasks: • Right-click the icon for the script file, scroll to **Properties**, and select the **Unblock** checkbox. • Run this command: `Set-ExecutionPolicy Unrestricted -Scope Process` **NOTE:** ◦ You can run this command from any directory. ◦ Owing to the performance characteristics of Windows PowerShell sessions, it is a good practice to run this command immediately before you run the script (in the next step).
2	Open Windows PowerShell with administrative rights and run the script. **NOTE:** The execution of the script has these results: • The Certificate Authority and the signed certificates for the BMENOC0302(H) modules are generated. ◦ The Certificate Authority (ca-cert.pem) appears in the root directory. ◦ Signed certificates for the BMENOC0302(H) modules appear in the *SignedCerts* directory. • Two certificates are generated for the Windows computer. ◦ Execute a `certmgr` (Manage computer certificates) Window command to open the `certlm` dialog box and select the *Certificates* directory from the navigation tree (*Certificates - Local Computer* > *Personal* > *Certificates*). • A channel is configured for the IPsec transport mode.
3	Proceed to the next table (**Resume the configuration**).

NOTE: In the absence of PKI, you can establish an IPsec connection to only one Windows computer.

Resume the configuration.

Step	Action
1	Open these items: • Open the **Root Certificate** tab on the Modicon M580 BMENOC0302(H) website (**CONFIGURATION > Root Certificate**). • Open the local *Test_IPsec* folder.
2	Drag and drop the certificate for a BMENOC0302(H) module (ca-cert.pem file) from the *Test_IPsec* directory to the **ADD ROOT CERTIFICATE** area on the **Root Certificate** webpage. **NOTE:** The certificate for the BMENOC0302(H) is added to the **ROOT CERTIFICATES** table on the **Root Certificate** page.
3	Drag and drop the corresponding certificates for the BMENOC0302(H) module from the *SignedCerts* directory to the **SET SIGNED CERTIFICATE** area on the **Signed Certificates** web page. **NOTE:** Repeat this step for every BMENOC0302(H) module that requires IPsec communications.
4	Download the EcoStruxure Control Expert application to the controller module for your project.

Communication Roles and Authentication

When IPsec is implemented, the device at either end of a VPN tunnel performs the role of *initiator* or *responder*, depending on which device started (*initiated*) the communications.

The BMENOC0302(H) module can perform both the *initiator* and *responder* VPN roles in the IPsec tunnel. These remote IPsec *initiator/responder* relationships are available:

Initiator	Responder	Requirement
BMENOC0302(H)	BMENOC0302(H)	Both BMENOC0302(H) modules run firmware version V1.01 or a subsequent supporting version.
Windows	BMENOC0302(H)	The operating system version is Windows v10 or v11 or a subsequent supporting version.

Network Considerations for IPsec

Network Characteristic	Consideration
Hot Standby	When the standby controller in a Hot Standby assumes the responsibilities of the primary controller as a result of a hot swap, the new primary controller automatically switches from *IP+1* to *main IP* to reestablish the IPsec channel (connection) before it resumes other communications.
RSTP	In the event of a network reconfiguration (an RSTP daisy-chain loop, for example), the IPsec initiator re-initiates the IPsec channel (connection) automatically.

IPsec Connection and Reconnection Times

The IPsec connection and reconnection times can vary according to network conditions, device performance, or the specific IPsec configuration.

There is no standard formula to consistently calculate the precise connection, but the connection or reconnection process usually occurs in this order:

Phase	Description
1	ISAKMP/IKE Phase 1 (IKE_SA_INIT): - 2 messages: • Negotiate cryptographic algorithms. • Exchange IPsec nonces and Diffie-Hellman values.
2	ISAKMP/IKE Phase 2 (IKE_AUTH): - 2 messages: • Exchange certificates for authentication. • Authenticate peers using the provided certificates. • Establish the first security association for a secure transfer of data. **NOTE:** A *security association* (SA) describes the relationship between devices in terms of the manner in which they conduct secure communications.

NOTE: Each of the above steps involves encryption, decryption, and sometimes complex calculations, depending on the chosen algorithms.

You can roughly estimate the IPsec connection and reconnection times with this formula:

```
T approx = N x (L + P)
```

The formula uses these variables:

- *T* (time) = the elapsed time in establishing the IPsec connection.

- *N* (number) = the number of messages that are exchanged as a requirement of each of the above phases. For example:
  - Two messages are exchanged for IKE_SA_INIT - Phase 1.
  - Two messages are exchanged for IKE_AUTH - Phase 2.

    NOTE: These are the possible values:

    - *N = 2:* Use this value for a reconnection (two messages).
    - *N = 4:* Use this value for a new connection (four messages).
- *L* (latency) = the round-trip time (RTT) between two endpoints in the communications.
- *P* (processing) = the length of processing time that elapses for each device in processing ISAKMP/IKE messages.

    NOTE:

    - Because IKEv2 uses certificate authentication, this variable determines the connection time. The variable depends solely on the cryptographic capabilities and the hardware involved.
    - Faster devices reduce the `P` time.

Example:

Sample Values	Result
L = 100ms	The connection time is 360 ms: `T_{reconnection} approx.= 2 x (100ms + 80ms) = 2 x 180ms = 360ms`
P = 80ms	
N = 2	

These times can vary widely based on network conditions, device performance, and specific IPsec configurations.

Additional factors (like network congestion, device load, specific IP settings) can affect the connection and reconnection times.

NOTE: Considerations for reconnection times:

- Because certificates are exchanged for authentication, the reconnection time for a complete reauthentication can vary significantly depending on the cryptographic capabilities and security settings in the configuration.
- A typical reconnection reestablishes lost security associations. In general, the reconnection takes less time than the initial connection but uses some of the same steps. If you refresh only the security associations, the reconnection time is usually faster than that of a full reconnection.

Windows Firewall Script with Encryption

The DTM can generate a Windows firewall script with confidentiality enabled.

Transmitted data is confidential when the transmission medium is encrypted. In this case, only parties with the decryption key can read the data.

The content of the generated script conforms to the values configured in the IPsec parameters. These sample lines of code provide important information about the IKEv2 IPsec connection:

```
$PCIPAddress = "192.168.10.200"
```

The above string shows the IP address of the IPsec connection that is configured as the Windows (**PC**) device. The address is read from the **IPSec Service** configuration.

```
$NOCIPAddress = @("192.168.15.1", "192.168.15.2")
```

The above string shows the IP addresses of IPsec connections that are not configured for the Windows (**PC**) computer and the IP addresses that are being configured for BMENOC0302(H) modules. These addresses are read from the **IPSec Service** configuration.

```
$DistinguishedName= "CN=CA-IPSEC-SCHNEIDER"
```

The above string differentiates the self-signed CA from the other certificates on the Windows computer. The CN (common name) value identifies the domain name, host name, or IP address of the device associated with the certificate. It is applicable only when a PKI is not available. This hard-coded value is defined during the generation of a self-signed CA.

```
$ROOTCATHUMBPRINT= "65fb5831afdb6b713d5757d38bc8966e818aef2a"
```

The user-configured value for the *$ROOTCATHUMBPRINT* variable is applicable only when a PKI is available. This value is read from the **Root CA Certificate Thumbprint** field in the **IPSec Service** configuration.

```
netsh advfirewall set global mainmode mmsecmethods dhgroup14:aes128-sha256
```

The above string defines the proposed IKEv2 properties. The properties in this case are Diffie-Hellman group 14 (DHGroup14), the 128-bit advanced encryption standard (AES-128), and the 256-bit secure hash algorithm (SHA-256).

```
$qMProposal_confidentiality = New-NetIPsecQuickModeCryptoProposal -Encapsulation
ESP -ESPHash SHA256 -Encryption None
```

The above string defines the proposed IPsec confidentiality properties. The properties in this case include the Encapsulating Security Payload (*ESP*) protocol, SHA-256, and absence of encryption (*None*).

NOTE: IPsec supports Encapsulating Security Payload (ESP) with and without confidentiality in accordance with the user configuration in the DTM.

```
$qMProposal_confidentiality = New-NetIPsecQuickModeCryptoProposal -Encapsulation
ESP -ESPHash SHA256 -Encryption AES128
```

The above string defines the proposed IPsec definitions. The definitions in this case include the Encapsulating Security Payload (ESP) protocol, SHA-256, and encryption AES-128.

```
$certprop = New-NetIPsecAuthProposal -machine -cert -Authority $DistinguishedName
-AuthorityType Root
```

The above string defines the certificate properties to use in the IKEv2 authentication method. The definitions in this case are for certificates that have a CA with the same name as the *$DistinguishedName* variable.

```
$certprop = New-NetIPsecAuthProposal -machine -cert -Authority $reversedSubject
-AuthorityType Root
```

The above string defines the certificate properties to use in the IKEv2 authentication method. The definitions in this case are for certificates signed by the CA configured as the *$ROOTCATHUMBPRINT* value in the **IPSec Service** configuration.

```
$myauth = New-NetIPsecPhase1AuthSet -DisplayName "IKEv2TestPhase1AuthSet" -proposal
$certprop
```

The above string defines the IKEv2 authentication method. The properties of the applicable certificates depend on the availability of **PKI Enrolment**, as indicated in the **IPSec Services** configuration.

```
New-NetIPsecRule -DisplayName "BMENOC302_IPSec_Rule" -Mode Transport -RemoteAddress
$NOCIPAddress -Phase1AuthSet $myauth.Name -InboundSecurity Request
-OutboundSecurity Request -QuickModeCryptoSet $qMCryptoSet.Name -KeyModule IKEv2
```

The above string establishes the IPsec IKEv2 rule that uses the previously defined properties (IP address, proposals, authentication methods, certificate properties). It also defines the display name (*BMENOC302_IPSec_Rule*). It uses transport mode.

NOTE:

- Open the downloaded script file (.ps1) to see its complete content.
- The exact contents of the script file depend on the DTM configuration for the specific module.

Verify the IPsec Connection

You can verify the IPsec connection only after you perform these tasks:

- Configure the DTM, page 121.
- Configure the Windows firewall, page 129.
- Download the EcoStruxure Control Expert application to the controller, page 84.

NOTE: The application includes the configuration for the BMENOC0302(H) and its corresponding IPsec configuration.

Verify the connection:

<table>
<tr><th>Step</th><th colspan="2">Action</th></tr>
<tr><td>1</td><td colspan="2">Send a constant ping from the Windows computer to the BMENOC0302(H) module to verify that IPsec connections are operational.
<b>NOTE:</b> The first few pings may time out while the connection is being established.</td></tr>
<tr><td>2</td><td colspan="2">Use a network analyzer (like Wireshark) or the Windows Security Console to verify that the ping requests and replies are secured with IPsec.</td></tr>
<tr><td rowspan="3">3</td><td rowspan="3">Use standard Windows IPsec diagnostic tools to troubleshoot IPsec communications. (These steps use the Microsoft Management Console (MMC) service for management applications.)
<b>NOTE:</b> You cannot reset the values. To refresh the count values, relaunch the Microsoft Management Console.</td><td>a. In Windows, create a Microsoft Management Console that includes the IP Security Monitor snap-in and Windows Firewall with Advanced Security snap-in.</td></tr>
<tr><td>b. In the Windows Firewall with Advanced Security snap-in, expand the <b>Monitoring</b> selection and the <b>Security Association</b> section to view the active <b>Main Mode</b> and <b>Quick Mode</b> connections. You can also view the entries for each active IPsec connection.</td></tr>
<tr><td>c. In the <b>IP Security Monitor</b>, expand the <b>Quick Mode</b> selection and click <b>Statistics</b> to view the number of bytes that are received and sent through secured connections.</td></tr>
</table>

IPsec Web Interface Description

NOTE: An expanded discussion of the web interface for the BMENOC0302(H) module is discussed elsewhere, page 362.

Connect to the Modicon M580 BMENOC0302(H) website through an internet browser. Enter the secure extension of the Hypertext Transfer Protocol (`https://`) in your browser’s address bar, followed by the IP address of the module (example: `https://172.168.12.158`).

Use these credentials to open the IPsec website for the Modicon M580 BMENOC0302(H) module:

- *username:* loader
- *password:* Use the firmware password set in the EcoStruxure Control Expert application. For more information, refer to Firmware Protection topic in EcoStruxure Control Expert Operating Modes.

View these IPsec webpages for the BMENOC0302(H) module when you select the **CONFIGURATION** webpage and expand the **Cybersecurity** menu:

- **Root Certificates:** Use this webpage to push, list, or remove one Certificate Authority at a time. The page contains these parameters:
    - **ADD ROOT CERTIFICATE:** This area contains the **Drag & Drop your certificate or Browse** button. Drag-and-drop the certificate to this area or drive to the local certificate and upload it to here with Windows commands.
    - **ROOT CERTIFICATES:** This table contains these columns:
        - **ID:** The root certificate is identified by this name.
        - **Subject:** The subject includes information that identifies the device in the IPsec tunnel by its characteristics, like its organization, location, Common Name, and so forth.
        - **Issuer:** The issuing device creates the IPsec tunnel, thereby initiating a security association for the VPN connection.
        - **Expiration:** This is the time at which the IPsec security association for a specific tunnel expires. IPsec automatically disconnects upon expiration (unless the security association is renewed beforehand).
- **Signed Certificates:** Use the **Signed Certificates** webpage to generate a Certificate Signing Request, set a signed certificate for the device, or download the device and the IPsec certificates. The page contains these parameters:
    - **CERTIFICATE REQUEST:** Click the **Download CSR** button to download the Certificate Signing Request (CSR) for the BMENOC0302(H) module. Repeat this action for every BMENOC0302(H) module that requires IPsec communications. Save the downloaded script to a root directory from which it can be run. The BMENOC0302 (H) module gives the destination folder for the download this name: *CSRs*.

      NOTE: The *CSRs* directory remains the destination folder for the actions that follow.
    - **SET SIGNED CERTIFICATE:** This area contains the **Drag & Drop your certificate or Browse** button. Drag-and-drop the certificate to this area or drive to the local certificate and upload it to here with Windows commands.
    - **SIGNED CERTIFICATES:** This is the name of the signed certificate.
        - **Service Name:** This unique name links to a specific IPsec tunnel, as verified by the Certificate Authority.
        - **Device:** This name identifies the endpoints of an IPsec tunnel.
        - **IPSec**
        - **Download:** Click this button to download signed certificates.

IPsec Web Functionality

Configure the IPsec functionality for your application through dedicated webpages:

Step	Action
1	On the **Signed Certificates** page (**Configuration > Signed Certificates**), click the **Download CSR** button to download the script for a Certificate Signing Request (CSR).
2	Save the downloaded script file to the *CSRs* directory.
3	Configure IPsec IKEv2 in the EcoStruxure Control Expert application.

The result of the download depends on the availability of PKI Enrollment. These are examples of the file names that appear in the destination directory (*CSRs*):

- PKI Enrollment is not available: `Client_has_no_pki.ps1`
- PKI Enrollment is available: `BMENOC0302_DeviceCSR.pem`, `BMENOC0302_DeviceCSR1.pem`, `BMENOC0302_DeviceCSR2.pem`

The generated script includes its own Certificate Authority. It signs the certificate for the Windows computer that is later used to establish IPsec communications. It also signs the Certificate Signing Requests for the relevant BMENOC0302(H) modules.

The resulting certificates to be uploaded to the BMENOC0302(H) module are located in a folder called *SignedCerts*, which is created by the script during the generation and signing of certificates. It appears in the same folder as the downloaded scripts (.ps1, .pem) and the Certificate Signing Requests (*CSRs*). These are examples of the file names that appear in the download target folder:

- `signed_BMENOC0302_DeviceCSR.pem`
- `signed_BMENOC0302_DeviceCSR1.pem`
- `signed_BMENOC0302_DeviceCSR2.pem`

During the certificate upload, upload the exported Certificate Authority certificate (`ca-cert.pem`) that is installed on the BMENOC0302(H) module. Then upload the signed certificate to each corresponding webpage. If there is a mismatch, IPsec communications stop.

NOTE:

- In this case the Windows computer certificate and the Certificate Authority are installed automatically by the same script.
- The *Privacy-Enhanced Mail* file format (.pem) stores and exchanges cryptographic keys, certificates, and other private information that supports secure communications.

Because PKI Enrollment is available, these are examples of the file names that appear in the *CSRs* destination folder:

- `client_has_a_pke.ps1`

- `request.req`

Considerations:

- As described above, the private key for the Windows certificate is saved in the local Windows certificate. Use the Certificate Import Wizard to install this computer-signed certificate at this location: *Certificate > Local Machine > Personal*
- The Certificate Authority that was used to sign the certificates is trusted. Use the Certificate Import Wizard to install the CA at this location: *Certificate > Local Machine > Trusted Root Certificate Authorities*
- It is required that the Certificate Signing Request for the BMENOC0302(H) module be signed by the same Certificate Authority. Upload each Certificate Authority to the corresponding webpages.

Troubleshoot IPsec Communications

Debug Connections

Debug IPsec connections:

Step	Action
1	Type *mmc* in the Windows **Run** menu to start the Microsoft Management Console. **NOTE:** Accept the prompts to continue.
2	Open the **Add/Remove Snap-in** dialog box (**File > Add/Remove Snap-in...**).
3	Select these items from the **Available snap-ins** pane individually and click **Add** to add them to the **Selected snap-ins** pane: • **IP Security Monitor**: View the details of the active security associations. • **Windows Defender Firewall:** Use the radial buttons to select a computer: ◦ **Local computer (the computer this console is running on)** ◦ **Another computer** **NOTE:** You can change many settings that are configured by the script here. However, use the netsh commands to change some settings.
4	Click **OK** to send the selected items to the **Console** dialog box.

Troublehoot IPsec Communications

Facilitate communications when IPsec is enabled:

Behavior	Reason	Solution
There is no communication with the BMENOC0302(H) when IPsec is enabled on a Windows computer. (See the note at the end of the table.)	The IPsec policy agent is not running on the computer.	To automatically start the computer, configure IPsec, page 118.
	IPsec is not enabled on the BMENOC0302(H).	Enable IPsec on the **Security** tab of the BMENOC0302(H) DTM.
	IPsec is not configured properly in Windows.	Verify that the parameters in the Windows configuration match those in the IPsec implementation: • Verify the IP address of the BMENOC0302(H) in the DTM. • Disable **Perfect Forward Secrecy** for both communication endpoints in Windows.
EcoStruxure Control Expert cannot connect to the BMENOC0302 (H) through Ethernet. (See the note at the end of the table.)	IPsec is not enabled on both the BMENOC0302(H) and the Windows computer.	Verify that the DTM configuration and the Windows **Local Security Policy** are enabled for IPsec.
	IPsec is not configured properly in Windows.	Verify that the parameters in the Windows configuration match those in the IPsec implementation: • Verify the IP address of the BMENOC0302(H) in the DTM. • Disable **Perfect Forward Secrecy** for both communication endpoints in Windows.
	The power to the BMENOC0302(H) was recently cycled.	Choose a solution: • Wait five minutes for the Windows security associations to timeout. • Unassign then reassign the local security policy in Windows to reset the security associations.

<table>
<tr><th>Behavior</th><th>Reason</th><th>Solution</th></tr>
<tr><td rowspan="3">The firmware update tool cannot connect to the BMENOC0302(H) through Ethernet. (See the note at the end of the table.)</td><td>IPsec is not enabled on both the BMENOC0302(H) and the Windows computer.</td><td>Verify that the DTM configuration and the Windows <b>Local Security Policy</b> are enabled for IPsec.</td></tr>
<tr><td>IPsec is not configured properly in Windows.</td><td>Verify that the parameters in the Windows configuration match those in the IPsec implementation:
• Verify the IP address of the BMENOC0302(H) in the DTM.
• Disable <b>Perfect Forward Secrecy</b> for both communication endpoints in Windows.</td></tr>
<tr><td>The power to the BMENOC0302(H) was recently cycled.</td><td>Choose a solution:
• Wait 5 minutes for the Windows security associations to timeout.
• Unassign then reassign the local security policy in Windows to reset the security associations.</td></tr>
<tr><td colspan="3"><b>NOTE:</b>
• The IKE and IPsec ports may be blocked by a firewall or another program associated with antivirus applications.
• Verify that the IKE port (UDP 500), IPsec Authentication Header port (51), and ESP port (501) are open on a firewall between the computer application and the controller module, including the firewalls associated with antivirus applications (like McAfee or Symantec).</td></tr>
</table>

Configure the Service to Start Automatically in Windows

The IPsec policy agent does not run if you see this message:

The service cannot be started ....

In that case, configure Windows to start the service automatically:

Step	Action
1	In Windows, expand (**+**) **Administrative Tools**.
2	Double-click **Services** to access the local services.
3	Double-click **IPsec Policy Agent** to open its properties.
4	Select the **General** tab.
5	In the **Startup type** pull-down menu, scroll to **Automatic**.
6	In the **Service status**, click **Start**. **NOTE:** When **Start** is greyed out, the service is running.
7	Click **OK** to apply the changes and close the window.

NOTE: When you enable IPsec, the DTM automatically disables the rack Ethernet port on the BMENOC0302(H). This isolates the IPsec network (control room network) from the device network.

Configure Security Services

Introduction

The EcoStruxure Control Expert DTM provides security services to the BMENOC0302(H) Ethernet communication module. Enable and disable these services on the **Security** tab in the EcoStruxure Control Expert DTM.

Access the Security Tab

View the **Security** configuration options:

Step	Action
1	Open your EcoStruxure Control Expert project.
2	Open the **DTM Browser** (**Tools** > **DTM Browser**).
3	In the **DTM Browser**, double-click the name that you assigned to the BMENOC0302(H) module, and open the configuration window. **NOTE:** You can also right-click the module, and select **Open**.
4	Select **Security** in the navigation tree to view the configuration options.

NOTE: For general safety-related information, refer to the *Modicon Controllers Platform Cyber Security, Reference Manual*.

Service Selection

This table describes the available services:

Service	Description
FTP	Enable or disable (default) these items: • firmware upgrade • device configuration management using the FDR service **NOTE:** Local data storage remains operational, but remote access to data storage is disabled.
TFTP	Enable or disable (default) the ability to read X80 I/O module configuration files using the FDR service. **NOTE:** In M580 Hot Standby systems, you can disable TFTP services in the Ethernet screen for the BMENOC0302(H) module. (Its DIO modules either do not push their configuration in the FDR server or they use only FTP.) In such cases, the Hot Standby FDR synchronization does not work (because it is based on TFTP).
HTTPS	Enable or disable (default) the web access service.
Access Control	**Enabled** (default): Deny Ethernet access to the Modbus and EtherNet/IP server by unauthorized network devices.
	Disabled: There is no restriction on which network devices can access the Modbus and EtherNet/IP server.
IPsec	Enable or disable (default) secure communications for traffic between the IP address that corresponds to a BMENOC0302(H) module and another IP address using IPsec.
Enable Confidentiality	Check this box to enable and encrypt all Ethernet services. **NOTE:** This check box is disabled when IPsec is enabled.
DHCP / BOOTP	Enable or disable (default) the automatic assignment of IP addressing settings. For DHCP, also enables/disables automatic assignment of subnet mask, gateway IP address, and DNS server names.
SNMPv1, SNMPv3	Enable or disable (default) the protocol used to monitor network-attached devices.
EIP	Enable or disable (default) access to the EtherNet/IP server and its electronic data sheets (EDS), which classify each network device and its functionality.

NOTE:

- The default settings represent a moderate level of security. The increased security reduces the communication capabilities and the access to communication ports.
- Services that are selected online (through Control Expert or ETH_PORT_CTRL) apply only to the rack on which the EF runs.
- Refer to the ETH_PORT_CTRL (see Modicon M580, Hardware, Reference Manual) topic for information regarding using this function block to enable/disable the FTP, TFTP, HTTP, and DHCP/BOOTP protocols.

Enable Security

Set the **Security** tab parameters before you download the application to the controller module. When they are disabled, security services can be enabled only when you download an application.

Set the security level:

Step	Action
1	Select **Enabled** in the associated pull-down menu. **NOTE:** When you enable or disable a service, the pencil symbol indicates that you are editing the security settings.
2	Choose a security level: • Click **Enforce Security** to reset the services to the default states and implement the highest level of security. • Click **Unlock Security** to use the lowest level security settings (opposite of default settings).
3	Click **Apply** to enable the service. **NOTE:** The pencil symbol disappears.
4	Save your project (**File > Save**).

Access Control

Introduction to Access Control

Use the **Access Control** page to restrict device access to the BMENOC0302(H) module or the controller module communication server service through the BMENOC0302(H) module in its role as either a Modbus TCP, EtherNet/IP, FTP, TFTP, HTTPS, or SNMP server. When you enable access control in the **Security** dialog box, add the IP addresses of the devices that you want to configure for communications with the BMENOC0302(H) module to the list of authorized addresses:

- Support EtherNet/IP or Modbus TCP communications between a device in the subnet and the BMENOC0302(H) module with one of these methods:
  - Use the IP address of the BMENOC0302(H) module (default method).
  - Use the controller module communication-server service through the BMENOC0302 (H) module when **Subnet** is set to *Yes*.
- Add the IP address of a client device that may send a request to the BMENOC0302(H) module or the controller module communication-server service through the BMENOC0302(H) module, which, in this case, acts as a Modbus TCP or EtherNet/IP server.
- Add the IP address of your maintenance computer to communicate with the controller module through the BMENOC0302(H) module or the controller module’s communication-server service through the BMENOC0302(H) module (using EcoStruxure Control Expert to configure and diagnose your application).
- A service column in the table of authorized addresses is not available when the respective service is disabled in the **Services** field.

You can enter a maximum of 128 authorized IP addresses.

View the Access Control Table

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the **Security** tab, page 140.
3	Scroll down to the **Access Control** group box.
4	Select **Enabled** from the pull-down menu.

These columns appear in the table:

Column	Value
Subnet	*Yes:* Use the controller module communication-server service through the BMENOC0302(H) module
	No: Select the appropriate action: • Add a single IP address. • Remove the device that corresponds to the IP address from the table of authorized addresses.
IP Address	Enter the IP addresses for devices that you want to configure for communications with the BMENOC0302(H) module.
Subnet Mask	Enter the subnet mask for devices that you want to configure for communications with the BMENOC0302(H) module.
(protocols and services)	Refer to the service selection table below.

Service Selection

This table describes the services you can enable for the device at the corresponding IP address in the list of authorized addresses.

NOTE: A complete configuration procedure for these services follows this table.

Service	Description
FTP	Enable or disable the device configuration management using the FDR service. **NOTE:** Local data storage remains operational, but remote access to data storage is disabled.
TFTP	Enable or disable the ability to read X80 I/O module configuration files using the FDR service. **NOTE:** Enable TFTP for M580 Hot Standby systems.
HTTPS	Enable or disable (default) the firmware upgrade through web access.
IPsec	Enable or disable (default) secure communications for traffic between the IP address that corresponds to a BMENOC0302(H) module and another IP address using IPsec. **NOTE:** This service is available only when you enable it in the IPsec configuration, page 118.
EIP	Enable or disable (default) access to the EtherNet/IP server and its electronic data sheets (EDS), which classify each network device and its functionality.
SNMPv1, SNMPv3	Enable or disable the protocol that monitors network-attached devices.

NOTE:

- The default settings represent a moderate level of security. The increased security reduces the communication capabilities and the access to communication ports.
- Services that are selected online (through EcoStruxure Control Expert or ETH_PORT_CTRL, page 147) apply only to the rack on which the EF runs.
- To use this function block to enable or disable the FTP, TFTP, HTTPS, and DHCP/BOOTP protocols, refer to the description of the ETH_PORT_CTRL in the *Modicon M580, Hardware, Reference Manual*.

Add the Addresses of Authorized Devices

To grant access to a particular device, add its IP address to the list of authorized devices:

Step	Action
1	Set **Access Control** to **Enabled**.
2	In the **IP Address** column of the **Authorized Addresses** list, double-click the default IP address (0.0.0.0) to enter an IP address.
3	Enter the address of the device to access the BMENOC0302(H) module or the controller module communication-server service through the BMENOC0302(H) module with either of these methods: • *Add a single IP address*: Enter the IP address of the device and select **No** in the **Subnet** column. • *Add a subnet*: Enter a subnet address in the **IP Address** column. Select **Yes** in the **Subnet** column. Enter a subnet mask in the **Subnet Mask** column. **NOTE:** A red exclamation point (!) indicates a detected error in the entry. You can save the configuration only after the detected error is addressed.
4	Repeat these steps for each additional device or subnet for which you want to grant access to the BMENOC0302(H) module or the controller module communication-server service through the BMENOC0302(H) module. **NOTE:** You can enter up to 128 authorized IP addresses or subnets.
5	Click **Apply**.

Remove the Addresses of Devices

To deny access to a particular device, set its IP address to *0.0.0.0* in the list of authorized devices:

Step	Action
1	In the **Authorized Addresses** list, select the IP address of the device to delete.
2	Set the IP address to *0.0.0.0*.
3	Select **No** in the **Subnet** column.
4	Click either: • **OK**: Save changes and close the window. • **Apply**: Save changes and leave the window open. • **Cancel**: Cancel changes.

ETH_PORT_CTRL: Execute a Security Command in an Application

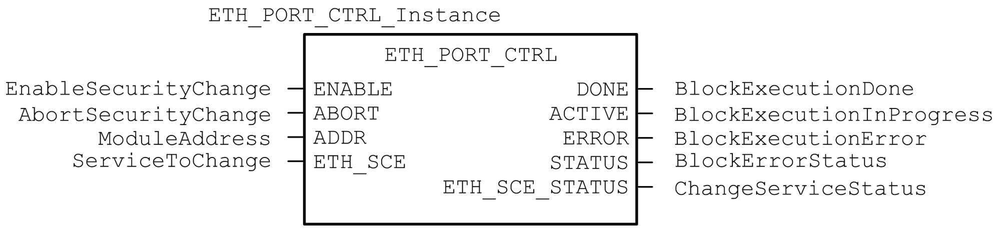
Function Description

Use the ETH_PORT_CTRL function block to control the FTP, TFTP, HTTPS, and DHCP / BOOTP protocols when they are enabled in the **Security** screen of the EcoStruxure Control Expert DTM. (By default, these protocols are disabled.) Map the inputs on variables and on unlocated variables in which the HMI property is disabled. (The variable is not in the data dictionary.)

The additional parameters `EN` and `ENO` may also be configured.

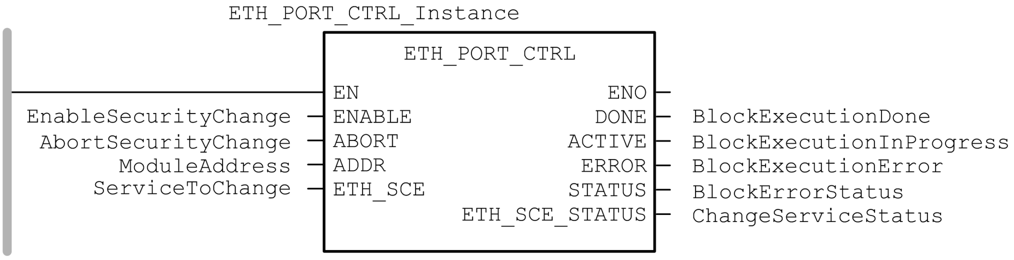
FBD Representation

Representation:



LD Representation

Representation:



IL Representation

```
CAL ETH_PORT_CTRL_Instance (ENABLE := EnableSecurityChange, ABORT :=
AbortSecurityChange, ADDR := ModuleAddress, ETH_SCE := ServiceToChange,
DONE => BlockExecutionDone, ACTIVE => BlockExecutionInProgress, ERROR
=> BlockExecutionError, STATUS => BlockErrorStatus, ETH_SCE_STATUS =>
ChangeServiceStatus)
```

ST Representation

```
ETH_PORT_CTRL_Instance (ENABLE := EnableSecurityChange, ABORT :=
AbortSecurityChange, ADDR := ModuleAddress, ETH_SCE := ServiceToChange,
DONE => BlockExecutionDone, ACTIVE => BlockExecutionInProgress, ERROR
=> BlockExecutionError, STATUS => BlockErrorStatus, ETH_SCE_STATUS =>
ChangeServiceStatus);
```

Input Parameters

Parameter	Type	Comment
ENABLE	BOOL	Set to *1* to enable the operation.
ABORT	BOOL	Set to *1* to abort the active operation.
ADDR	ANY_ARRAY_INT	This array contains the address of the Ethernet communications module for which you want to change the security state (the result of the ADDMX or ADDM function). For example: • ADDM('0.0.3') for an M580 controller module • ADDM('0.3.0') for a BMENOC in slot 03 of the main rack • ADDMX('0.0.3{192.168.10.2}SYS) for a BMXCRA with the IP address 192.168.10.2 **NOTE:** • To address a module in the local rack, enter *0.0.10* (the controller module's main server address). • In M580 Hot Standby systems, ADDR represents the address of the primary controller. If you disable TFTP you disable the synchronization of the FDR service, page 103.
ETH_SCE	WORD	For each protocol, use these binary values to control the protocol: • 00: The protocol is unchanged. • 01: Enable the protocol. • 10: Disable the protocol. • 11: Reserved **NOTE:** A value of 11 reports a detected error in ETH_SCE_STATUS. These bits are used for the different protocols: • 0, 1: FTP • 2, 3: TFTP (Modicon M580 only) • 4, 5: HTTPS • 6, 7: DHCP / BOOTP • 8...15: Reserved (value = 0)

Output Parameters

Parameter	Type	Comment
DONE	BOOL	Operation completed indication. Set to *1* when the execution of the operation is completed successfully.
ACTIVE	BOOL	Operation in progress indication. Set to *1* when the execution of the operation is in progress.
ERROR	BOOL	Set to *1* if an error is detected by the function block.
STATUS	WORD	Code providing the detected error identification. **NOTE:** Refer to the *EcoStruxure™ Control Expert, Communication, Block Library* guide.
ETH_SCE_STATUS	WORD	For each protocol, these values contain the response to an attempt to enable or disable the FTP, TFTP, HTTPS, or DHCP / BOOTP protocols: • 0: command executed • 1: command not executed Reasons for not executing the command can be: • The communication service is disabled by the configuration. • The communication service is in the state requested by the command (**Enabled** or **Disabled**). • The communication service (x) is not supported by the module or is a non-existing service. These bits are used for the different protocols: • 0: FTP • 1: TFTP • 2: HTTPS • 3: DHCP / BOOTP • 4 ... 15: Reserved (value = 0)

Execution Type

When used on a BMENOC0302(H) module, the ETH_PORT_CTRL function block is executed ***asynchronously*** and may take several cycles until the DONE output turns *ON*. Therefore, the ACTIVE output is set to *ON* until the completion of the ETH_PORT_CTRL function block.

Use the ETH_PORT_CTRL EFB

Step	Action
1	Set the bits of the services to be activated in `ETH_SCE`.
2	Set the `ENABLE` input to activate the EFB.
3	Reset the `ENABLE` input as soon as the `ACTIVE` output is reset by the EFB.
4	Assess the output value of the `STATUS` output: • `STATUS`<>0: There is a communication status code. • `STATUS` = 0: Check `ETH_SCE_STATUS`. The services for which the bits are set were not properly modified.

Hot Standby Considerations

When a Hot Standby system is configured to enable or disable Ethernet services and protocols, configure the ETH_PORT_CTRL function block to run in the controllers in both primary and standby rack configurations:

Step	Action
1	In the EcoStruxure Control Expert **Project Browser**, expand (**+**) **Configuration** and **PLC Bus**.
2	Expand the rack configuration.
3	Right-click the controller module and scroll to **Open** to open its configuration tabs.
4	Select the **Hot Standby** tab.
5	In the **Behaviour of the CPU in Wait and Standby mode** group box, select *All sections* from the **CPU executes** pull-down menu.
6	Rebuild and save the project to apply your changes.

Device List

Device List Configuration and Connection Summary

Introduction

The **Device List** contains read-only properties that summarize these items:

- configuration data:
  - input data image
  - output data image
  - maximum number or implemented numbers of devices, connections, and packets
- Modbus request and EtherNet/IP connection summary

Open the Page

Open the **Device List** page:

Step	Action
1	Open your EcoStruxure Control Expert project.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module.
4	Double-click the name of the BMENOC0302(H) to open the configuration window. **NOTE:** You can also right-click the module and scroll to **Open** to open the configuration window.
5	Select **Device List** in the navigation tree.

Configuration Summary Data

Select **Device List** and view the **Configuration Summary** table on the **Summary** tab to see values for these items:

- *Input*
- *Output*
- *Configuration Size*

Expand (**+**) the *Input* or *Output* row to view these values:

Parameter	Description
Input Current Size *Output Current Size*	This value is the sum of the Modbus requests and EtherNet/IP connection sizes for either inputs and outputs. Configure these values in the **General** page for a selected distributed device and connection.
Input Maximum Size *Output Maximum Size*	This value represents the maximum size of the total Modbus requests and EtherNet/IP connection sizes for either inputs and outputs.

The maximum size of the implicit input or output is 16 KB (16,384 bytes), including overhead.

You can calculate the maximum size of the overhead for the variable by considering these components of the variable:

- The variable contains a descriptor followed by a value that represents the number of input or output data objects.
- Each data object contains an object header followed by the input or output data.
- The number of data objects and the size of the input or output data depend on the configuration.

These are the maximum sizes of the items listed above:

		Size	Subtotal
descriptor		16 bytes	16
object header + input/output data		3 bytes	3
input or output data objects	scanned devices or local slaves supported by the BMENOC0302(H) module	128 bytes	129
	input or output object for the scanner service DDT	1 byte	

Using the values in the *Subtotal* column, you can calculate the maximum size of the overhead at 403 bytes with this formula:

```
16 + (3 * 129) = 403
```

Therefore, the maximum input or output current size is approximately 15.6 KB.

NOTE: The current input and output sizes also include scanner DDT input data:

- *Input Current Size:* 188 bytes of scanner DDT input data
- *Output Current Size:* 48 bytes of scanner DDT input data

the BMENOC0302(H) module supports plus one (1) input or output object for the scanner Device DDT).

Expand (**+**) the **Configuration Size** row in the **Connection Summary** table to view these values:

Name	Description	Source
Maximum Number of DIO Devices	This value represents the maximum number of distributed devices allowed in the configuration.	capability of the module
Current Number of DIO Devices	This is the number of active and inactive distributed devices and local slaves in the configuration.	number of devices in the **Device List**
Maximum Number of DIO Connections	This value represents the maximum number of connections that the Ethernet communications module can manage.	capability of the module
Current Number of DIO Connections	This is the number of connections to active devices and local slaves in the configuration.	device configuration in the EcoStruxure Control Expert **Device Editor**
Maximum Number of Packets	This is the maximum number of Ethernet I/O scanning packets per second that the Ethernet communications module supports.	capability of the module
Current Number of Input Packets	This is an estimate of the number of input packets per second that the active configuration generates.	device configuration in the EcoStruxure Control Expert **Device Editor**
Current Number of Output Packets	This is an estimate of the number of output packets per second that the active configuration generates.	device configuration in the EcoStruxure Control Expert **Device Editor**
Current Number of Total Packets	This number is an estimate of the total number of Ethernet I/O scanning packets per second that the active configuration generates.	device configuration in the EcoStruxure Control Expert **Device Editor**
Maximum Number of CSIO Devices	These CSIO (CIP safety-related I/O) values do not apply to the BMENOC0302(H) module.	
Current Number of CSIO Devices		
Maximum Number of CSIO Connections		
Current Number of CSIO Connections		

Request/Connection Summary

Select **Device List** and view the **Request / Connection Summary** table on the **Summary** tab. The EcoStruxure Control Expert DTM uses this information to calculate the total bandwidth that distributed devices consume:

Column	Description
Connection Bit	• *health bits:* Connection health bits display the status of each device with one or more connections. • *control bits:* Use object IDs to toggle connection control bits on and off.
Task	The task type (FAST, MAST).
Input Object	This input object number is associated with the request or connection.
Output Object	This output object number is associated with the request or connection.
Device	This device number is used for the Health and Control Bit index.
Device Name	This is a label for the device in the **Device List**.
Type	This is the target device type: • EtherNet/IP • local slave • Modbus TCP
Address	This is the target device IP address (except for local slaves).
Rate (msec)	This is the EtherNet/IP Request Packet Interval (RPI) or the Modbus TCP Repetitive Rate. **NOTE:** The rate does not apply to local slaves.
Input Packets per second	This is the number of input (T->O) Ethernet packets per second generated by this request or connection.
Output Packets per second	This is the number of output (O->T) Ethernet packets per second generated by this request or connection.
Packets per second	This is the sum of input packets per second and output packets per second for the request or connection.
Bandwidth Usage	This is the total amount of network bandwidth (total bytes traffic) that the request or connection consumes.
Size In	This is the number of input words that are configured for this request or connection.
Size Out	This is the number of output words that are configured for this request or connection.

Device List Parameters

Introduction

Configure parameters for devices in the **Device List** on these tabs:

- **Properties**
- **Address Setting**
- **Request Setting** (Modbus devices only)

View the Configuration Tabs

Navigate to the **Device List** configuration tabs:

Step	Action
1	In the **DTM Browser** (**Tools > DTM Browser**), double-click the DTM that corresponds to the Ethernet communication module.
2	To view the associated Modbus TCP and EtherNet/IP devices in the navigation pane, expand (**+**) the **Device List**, page 152.
3	Select a device from the **Device List** to view the **Properties**, **Address Setting**, and **Request Setting** tabs. **NOTE:** These tabs are described in detail below.

Properties Tab

Configure the **Properties** tab to perform these tasks:

- Add the device to the configuration.
- Remove the device from the configuration.
- Edit the base name for that variables and data structures that the device uses.
- Indicate how input and output items are created and edited.

Configure the **Properties** tab:

Field	Parameter	Description
Properties	*Number*	This value represents the relative position of the device in the list.
	Active Configuration	**Enabled:** Add this device to the EcoStruxure Control Expert project configuration.
		Disabled: Remove this device from the EcoStruxure Control Expert project configuration.
	Comment	You may enter a relevant comment (optional).
IO Structure Name	*Structure Name*	EcoStruxure Control Expert automatically assigns a structure name based on the variable name.
	Variable Name	An auto-generated variable name is based on the alias name.
	Default Name	Click this button to restore the default variable and structure names.
Items Management	*Import Mode*	**Manual:** Manually add I/O items to the **Device Editor**. **NOTE:** Changes to the device DTM do not affect the I/O items list.
		Automatic: I/O from the device DTM are updated if the items list in the device DTM changes. **NOTE:** You cannot edit items in the **Device Editor**.
	Reimport Items	Click this button to import the I/O items list from the device DTM and overwrite manual I/O item changes. This function is available only when *Import Mode* is set to **Manual**. **NOTE:** This button is enabled only when you select **Manual** for the *Import Mode*.

Click the **Apply** button to save your changes and leave the window open for additional editing.

Address Setting Tab

Configure the **Address Setting** page to perform these tasks:

- Configure the IP address for a device.
- Enable or disable DHCP client software for a device.

NOTE: When the DHCP client software is enabled in a Modbus device, it obtains its IP address from the DHCP server in the Ethernet communication module.

In the **Address Setting** page, edit these parameters to conform to your application's design and functionality:

Field	Parameter	Description
IP Configuration	*IP Address*	By default, the first three octet values equal the first three octet values of the Ethernet communication module. By default, the fourth octet value equals this device number setting. In this case, the default value is 004.
	Subnet Mask	The device subnet mask.
	Gateway	The gateway address reaches this device. An address of *0.0.0.0* indicates this device is located on the same subnet as the Ethernet communication module.
Address Server	*DHCP for this Device*	**Enabled**: Activate the DHCP client in this device. The device obtains its IP address from the DHCP service provided by the Ethernet communication module. Refer to the description of the **DHCP** column in the auto-generated client list, page 104.
		Disabled (default): Deactivates the DHCP client in this device.
	Identified by	If **DHCP for this Device** is **Enabled**, it indicates the device identifier type: • **MAC Address** • **Device Name**
	Identifier	If DHCP for this device is enabled, the specific device MAC Address or Name value.

Click **Apply** to save your changes, and leave the window open for additional editing.

Request Setting Tab

Configure the **Request Setting** tab to add, configure, or remove Modbus requests for the Modbus device. Each request represents a separate link between the communication module and the Modbus device.

NOTE: The **Request Setting** tab is available only when you select a Modbus TCP device in the **Device List**.

Use the instructions below to create or remove a request with the **Request Setting** tab.

Create a Request

Create a Modbus request in the **Request Setting** tab:

Step	Action
1	Click the **Add Request** button on the **Request Setting** tab: • The new request appears in the table. • The corresponding request items appear in the **Device List**. **NOTE:** The **Add Request** function is enabled only when **Import Mode** on the **Properties** tab is set to **Manual**.
2	Configure the request settings according to the table below.
3	Repeat these steps to create additional requests.
4	Click **Apply** to save the request.

When you create a request, these **Request Settings** parameters are available:

Setting	Description
Connection Bit	This bit indicates the read-only offset for the health bit for this connection. Offset values (starting at 0) are auto-generated by the EcoStruxure Control Expert DTM based on the connection type.
Unit ID	The Unit ID number identifies the connection target. **NOTE:** Consult the manufacturer's user manual for the specific target device to find its Unit ID.
Health Time Out	This value represents the maximum allowed interval between device responses before a time out is detected: • valid range: 5 ... 65535 ms (default: 1500 ms) • interval: 5 ms
Repetitive Rate	This value represents the data scan rate in intervals of 5 ms. (The valid range is 0...60000 ms. The default is 60 ms.)
RD Address	Data that is read from the remote device at this address is stored in the input data image of the Ethernet communication module.
RD Length	This value represents the number of words (0...125) in the Modbus device that the communication module reads.
Last Value	This value represents the behavior of input data in the application if communications are lost: • **Hold Value** (default) • **Set To Zero**
WR Address	The output data image in the Ethernet communication module's data structure is written to this address in the remote Modbus device.
WR Length	This value represents the number of words (0...120) in the Modbus device to which the communication module writes.
Gateway/Bridge Device	This feature lets slower TCP/IP network devices (like gateways and bridges) communicate with the I/O Scanner: • Select (check) this box to enable this feature. The communication module doubles the timeout setting by increasing the number of re-transmissions to 6 (instead of the typical setting of 3). • Deselect (uncheck) this box to disable this feature.

Remove a Request

Remove a Modbus request in the **Request Setting** tab:

Step	Action
1	Click a row in the table.
2	Click the **Remove** button to remove the request and the corresponding items from the **Device List**.
3	Click **Apply** to save the configuration.

Logging DTM Events to an EcoStruxure Control Expert Logging Screen

Introduction

EcoStruxure Control Expert maintains a log of events for:

- the EcoStruxure Control Expert embedded FDT container
- each Ethernet communication module DTM
- each EtherNet/IP remote device DTM

Events relating to the EcoStruxure Control Expert FDT container are displayed in the **FDT log event** page of the **Output Window**.

Events relating to a communication module or remote EtherNet/IP device are displayed:

- in configuration mode: in the **Device Editor**, by selecting the **Logging** node in the left pane
- in diagnostic mode: in the **Diagnostics** window, by selecting the **Logging** node in the left pane

Access the Logging Page

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) Ethernet communications module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, find the name that you assigned to the BMENOC0302(H) module.
4	Double-click the name of the BMENOC0302(H) (or right-click **Open**) to open the configuration window.
5	Select **Logging** in the navigation tree.

Logging Attributes

The **Logging** window displays the result of an operation or function performed by EcoStruxure Control Expert. Each log entry includes these attributes:

Attribute	Description
Date/Time	The time the event occurred, displayed in the format: *yyyy-mm–dd hh:mm:ss*
Log Level	The level of event importance includes these values: • *Information:* The operation is successfully completed. • *Warning:* An EcoStruxure Control Expert operation finished, but the result may be a subsequent detected error. • *Error:* EcoStruxure Control Expert cannot complete the operation.
Message	A brief description of the core meaning of the event.
Detail Message	A more detailed description of the event, which may include parameter names, location paths, etc.

Log DTM and Module Events to the Syslog Server

Configure the Syslog Server

Configure the syslog server address for logging DTM and module events:

Step	Action
1	In EcoStruxure Control Expert, select **Tools > Project Settings**.
2	In the left pane of the **Project Settings** window, select **Project Settings > General > PLC diagnostics**.
3	In the right pane: • Select (check) the **Event logging** box. • In the **SYSLOG server address** field enter the IP address of the syslog server. • In the **SYSLOG server port number** field, enter the port number. **NOTE:** The **SYSLOG server protocol** is not configurable. It is set to *tcp* by default.

NOTE: To set up a syslog server in your system architecture, refer to the *Modicon Controller Platform Cybersecurity, User Guide*.

DTM Events Logged to the Syslog Server

These DTM events are logged to the syslog server:

- Configuration parameter change
- Adding a device
- Deleting a device
- Switching to **Advanced Mode**
- A **Rebuild All Project** command
- A **Build Changes** command
- Renaming of I/O variables
- Adding tasks
- Modifying tasks

Ethernet Module Events Logged to the Syslog Server

Many types of events for Ethernet modules are logged by the syslog server, including these examples:

- TCP connection denied due to **Access Control** list
- Enabling/Disabling communication services outside configuration
- Ethernet port link up/down events
- RSTP topology change
- Configuration download of COM services
- Program operating mode change of COMs (Run, Stop, Init)

Explicit Messaging

Introduction to Explicit Messaging

About Explicit Messaging

The BMENOC0302(H) Ethernet communications module supports explicit messaging through the EtherNet/IP and Modbus TCP protocols:

- *EtherNet/IP*: Use the `DATA_EXCH` function block in application logic to create an EtherNet/IP explicit message.
- *Modbus TCP*: Use the `DATA_EXCH` function block or `WRITE_VAR` and `READ_VAR` function blocks in application logic to create a Modbus TCP explicit message.

  NOTE: A single EcoStruxure Control Expert application can contain more than 32 explicit messaging blocks, but only 32 explicit messaging blocks can be active at the same time.

The remainder of this discussion describes the configuration of both EtherNet/IP and Modbus TCP explicit messages through these mechanisms:

- `DATA_EXCH` function block (in application logic)
- EcoStruxure Control Expert graphical interface

Explicit Messaging Using the DATA_EXCH Block

Overview

Use this overview of the `DATA_EXCH` function block to configure both EtherNet/IP and Modbus TCP explicit messages.

These instructions describe the configuration of the `DATA_EXCH` function block's management parameter, which is common to both Modbus TCP and EtherNet/IP explicit messaging.

Configure Explicit Messaging with DATA_EXCH

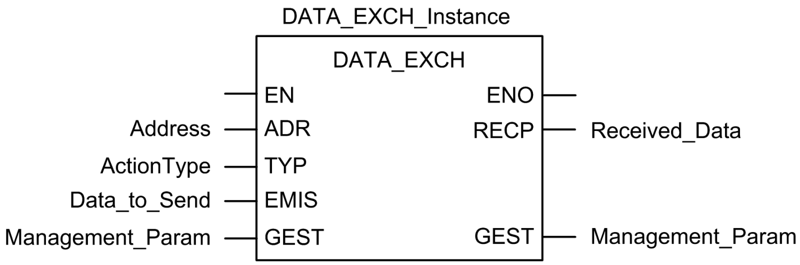
Overview

Use the DATA_EXCH function block to configure both Modbus TCP explicit messages and connected and unconnected EtherNet/IP explicit messages.

The Management_Param, the Data_to_Send, and the Received_Data parameters define the operation.

EN and ENO can be configured as additional parameters.

FBD Representation



Input Parameters

Parameter	Data type	Description
EN	BOOL	This parameter is optional. When this input is set to one, the block is activated and can solve the function block's algorithm. When this input is set to zero, the block is deactivated and does not solve the function block algorithm.
Address	Array [0...7] of `INT`	The path to the destination device, the content of which can vary depending on the message protocol. Use the `Address` function as an input to the block parameter `ADR`. Refer to a description of the `Address` parameter for these messages: • EtherNet/IP messages, page 175 • Modbus TCP messages, page 190
ActionType	`INT`	The type of action to perform. For both the EtherNet/IP and Modbus TCP protocols, this setting = 1 (transmission followed by await reception).
Data_to_Send	Array [n...m] of `INT`	The content of this parameter is specific to the protocol, either EtherNet/IP or Modbus TCP. For EtherNet/IP explicit messaging, refer to the instructions to configure the Data_To_Send parameter, page 175. For Modbus TCP explicit messaging, refer to the EcoStruxure Control Expert online help.

Input/Output Parameters

The *Management_Param* array is local:

Parameter	Data type	Description
Management_Param	Array [0...3] of `INT`	This is a four-word management parameter, page 171.

Do not copy this array during a switchover from a primary to a standby controller module in a Hot Standby system. Uncheck the *Exchange On STBY* variable in EcoStruxure Control Expert when you configure a Hot Standby system.

NOTE: Refer to the description of Hot Standby system data management and the T_M_ECPU_HSBY DDT in the *Modicon M580 Hot Standby, Frequently Used Architectures, System Guide*.

Output Parameters

Parameter	Data type	Description
ENO	BOOL	This parameter is optional. When you select this output you also get the EN input. The ENO output is activated upon successful execution of the function block.
Received_Data	Array [n...m] of `INT`	The EtherNet/IP (CIP) response, page 176 ... or the Modbus TCP response. (Refer to the configuration of explicit messaging with the DATA_EXCH function block in the *Modicon M340, BMXNOC0401 Ethernet Communication Module, User Manual*.) The structure and content depend upon the specific protocol.

Configuring the DATA_EXCH Management Parameter

Introduction

The structure and content of the management parameter of the DATA_EXCH block is common to both EtherNet/IP and Modbus TCP explicit messaging.

Configuring the Management Parameter

The management parameter consists of four contiguous words:

Data source	Register	Description	
		High Byte (MSB)	Low Byte (LSB)
Data managed by the system	Management_Param[0]	Exchange number	Two read-only bits: • Bit 0 = Activity bit, page 171 • Bit 1 = Cancel bit
	Management_Param[1]	Operation report, page 396	Communication report, page 395
Data managed by the user	Management_Param[2]	Block timeout. Values include: • 0 = infinite wait • other values = timeout x 100 ms, for example: ◦ 1 = 100 ms ◦ 2 = 200 ms	
	Management_Param[3]	Length of data sent or received: • Input (before sending the request): length of data in the Data_to_Send parameter, in bytes • Output (after response): length of data in the Received_Data parameter, in bytes	

Activity Bit

The activity bit is the first bit of the first element in the table. The value of this bit indicates the execution status of the communication function:

- **1**: The bit is set to 1 when the function launches.
- **0**: The bit returns to 0 upon the completion of the execution. (The transition from 1 to 0 increments the exchange number. If an error is detected during the execution, search for the corresponding code in the operation and communication report, page 395.)

For example, you can make this declaration in the management table:

```
Management_Param[0] ARRAY [0..3] OF INT
```

For that declaration, the activity bit corresponds to this notation:

```
Management_Param[0].0
```

> **NOTE:** The notation previously used requires configuration of the project properties in such a way as to authorize the extraction of bits on integer types. If this is not the case, `Management_Param[0].0` cannot be accessed in this manner.

EtherNet/IP Explicit Messaging Using DATA_EXCH

Overview

This section shows you how to configure the DATA_EXCH function block for EtherNet/IP explicit messages.

Explicit Messaging Services

Overview

Every explicit message performs a service. Each service is associated with a service code. Identify the explicit messaging service by its name, decimal number, or hexadecimal number.

You can execute explicit messages using the DATA_EXCH function block in the EcoStruxure Control Expert DTM.

Services

The services available in EcoStruxure Control Expert include, but are not limited to, these service codes:

Service Code		Description	Available in...	
Hex	Dec		DATA_EXCH block	EcoStruxure Control Expert
1	1	Get_Attributes_All	X	X
2	2	Set_Attributes_All	X	X
3	3	Get_Attribute_List	X	—
4	4	Set_Attribute_List	X	—
5	5	Reset	X	X
6	6	Start	X	X
7	7	Stop	X	X
8	8	Create	X	X
9	9	Delete	X	X
A	10	Multiple_Service_Packet	X	—

Service Code		Description	Available in...	
Hex	Dec		DATA_EXCH block	EcoStruxure Control Expert
B-C	11-12	(*Reserved*)	—	—
D	13	Apply_Attributes	X	X
E	14	Get_Attribute_Single	X	X
10	16	Set_Attribute_Single	X	X
11	17	Find_Next_Object_Instance	X	X
14	20	Error Response (DeviceNet only)	—	—
15	21	Restore	X	X
16	22	Save	X	X
17	23	No Operation (NOP)	X	X
18	24	Get_Member	X	X
19	25	Set_Member	X	X
1A	26	Insert_Member	X	X
1B	27	Remove_Member	X	X
1C	28	GroupSync	X	—
1D-31	29-49	(*Reserved*)	—	—
"X" indicates the service is available.				
"—" indicates the service is not available.				

Configure EtherNet/IP Explicit Messaging with DATA_EXCH

Configure the Address Parameter

To configure the Address parameter, use the `ADDM` function to convert the character string, described below, to an address that is input into the ADR parameter of the `DATA_EXCH` block:

ADDM('rack.slot.channel{ip_address}message_type.protocol'), where:

Field	Description
rack	This number is assigned to the rack that contains the communication module.
slot	This value represents the position (slot number) of the communication module in the rack.
channel	This is the communication channel (set to *0*).
ip_address	The is the IP address of the remote device (for example, *193.168.1.6*).
message_type	The message is of one of these types (represented as a three-character string): • **UNC** (unconnected message) • **CON** (connected message)
protocol	This three-character string represents the protocol type (**CIP**).

Configure the Data_to_Send Parameter

The *Data_to_Send* parameter varies in size. It consists of contiguous registers that include the message type and the CIP request in sequence:

Offset (words)	Length (bytes)	Data Type	Description
0	2 bytes	Bytes	Message type: • High byte = size of the request in words • Low byte = EtherNet/IP service code
1	`Management_Param[3]` (size of `Data_to_Send`) minus 2	Bytes	The CIP request **NOTE:** • The structure and size of the CIP request depends on the EtherNet/IP service. • Structure the CIP request in little endian order.

Contents of the Received_Data Parameter

The *Received_Data* parameter contains only the CIP response. The length of the CIP response varies, and is reported by `Management_Param[3]` after the response is received. The format of the CIP response is described, below:

Offset (words)	Length (bytes)	Data Type	Description
0	2	Byte	• High byte (MSB) = reserved • Low byte (LSB): reply service
1	2	Byte	• High byte (MSB): length of additional status • Low byte (LSB): EtherNet/IP general status.
2	length of additional status	Byte array	Additional Status **NOTE:** Refer to *The CIP Networks Library, Volume 1, Common Industrial Protocol* at section 3-5.6 *Connection Manager Object Instance Error Codes*.
...	`Management_Param[3]` (size of `Received_Data`) minus 4, and minus the additional status length	Byte array	Response data

NOTE: The response is structured in little endian order.

Check the Received_Data Response for System and CIP Status

Use the contents of the Received_Data parameter to examine both the system status and the CIP status of the Ethernet communication module when handling the explicit message.

Step	Action
1	Examine the value of the high byte (MSB) of the first response word, positioned at offset 0. If the value of this byte is: • equal to 0: the system properly handled the explicit message • not equal to 0: a system-based event occurred Refer to the list of EtherNet/IP Explicit Messaging Event Codes, page 391 for an explanation of the system-based event code contained in the second response word, positioned at offset 1.
2	If the system properly handled the explicit message, and the high byte of the first response word equals 0, check the value of the second response word, positioned at offset 1. If the value of this word is: • equal to 0: the explicit message was properly handled by the CIP protocol • not equal to 0: a CIP protocol-based event occurred **NOTE:** Refer to your CIP documentation for an explanation of the CIP status displayed in this word.

EtherNet/IP Explicit Message Example: Get_Attribute_Single

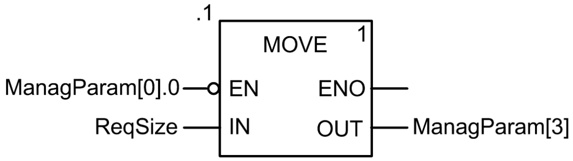
Overview

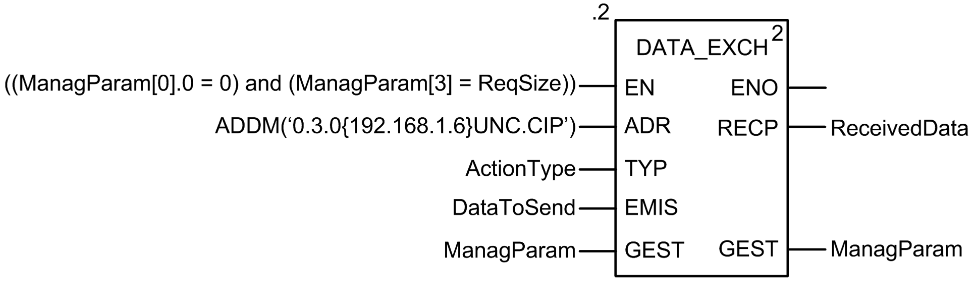
The following unconnected explicit messaging example shows you how to use the DATA_EXCH function block to retrieve diagnostic data from a remote device (at IP address 192.168.1.6). This example executes a Get_Attribute_Single of assembly instance 100, attribute 3.

You can perform the same explicit messaging service using the **EtherNet/IP Explicit Message** window, page 197.

Implement the DATA_EXCH Function Block

To implement the DATA_EXCH function block, create and assign variables for these blocks:





Configure the Address Variable

The *Address* variable identifies the explicit message originating device (in this example, the communication module) and the target device.

The *Address* variable does not include the X Way address elements {Network.Station} because the bridge is not through another controller-module station. For example, use the `ADDM` function to convert this character string to an address:

`ADDM`(‘0.2.0{192.168.1.6}UNC.CIP’), where:

- rack = 0
- module (slot number) = 2
- channel = 0
- remote device IP address = 192.168.1.6
- message type = unconnected
- protocol = CIP

Configure the ActionType Variable

The *ActionType* variable identifies the function type for the `DATA_EXCH` function block:

Variable	Description	Value
ActionType	Transmission followed by wait for response	01 hex

Configure the DataToSend Variable

The *DataToSend* variable identifies the details of the CIP explicit message request:

Variable	Description	Value
DataToSend[0]	CIP request service information: • High byte = request size in words: 03 hex (3 decimal) • Low byte = service code: 0E hex (14 decimal)	030E hex
DataToSend[1]	CIP request class information: • High byte = class: 04 hex (4 decimal) • Low byte = class segment: 20 hex (32 decimal)	0420 hex
DataToSend[2]	CIP request instance information: • High byte = instance: 64 hex (100 decimal) • Low byte = instance segment: 24 hex (36 decimal)	6424 hex
DataToSend[3]	CIP request attribute information: • High byte = attribute: 03 hex (3 decimal) • Low byte = attribute segment: 30 hex (48 decimal)	0330 hex

View the Response

Use an EcoStruxure Control Expert animation table to display the *ReceivedData* variable array. The *ReceivedData* variable array consists of the entire data buffer.

Display the CIP response:

Step	Action
1	In EcoStruxure Control Expert, open the **Project Browser** (**Tools > Project Browser**).
2	In the **Project Browser**, right-click the **Animation Tables** and scroll to **New Animation Table** to open a pop-up window.
3	In the **Properties** dialog, edit these values: • **Name:** Enter a table name. **NOTE:** Enter *ReceivedData* to follow this procedure through these steps. • **Functional module:** Accept the default (<*None*>). • **Comment:** Enter a comment here (optional). • **Number of animated characters:** Enter *100* to represent the size of the data buffer in words.
4	Click **OK** to close the dialog.
5	In the animation table's **Name** column, enter the name of the variable assigned to the RECP pin (*ReceivedData*) and click **Enter**. The animation table displays the *ReceivedData* variable.
6	Expand the *ReceivedData* variable to display its word array, where you can view the CIP response contained in the *ReceivedData* variable. **NOTE:** Each array entry presents two bytes of data in little endian format, where the least significant byte is stored in the smallest memory address. For example, *8E* in *word[0]* is the lower byte, and *00* is the upper byte.

EtherNet/IP Explicit Message Example: Read Modbus Object

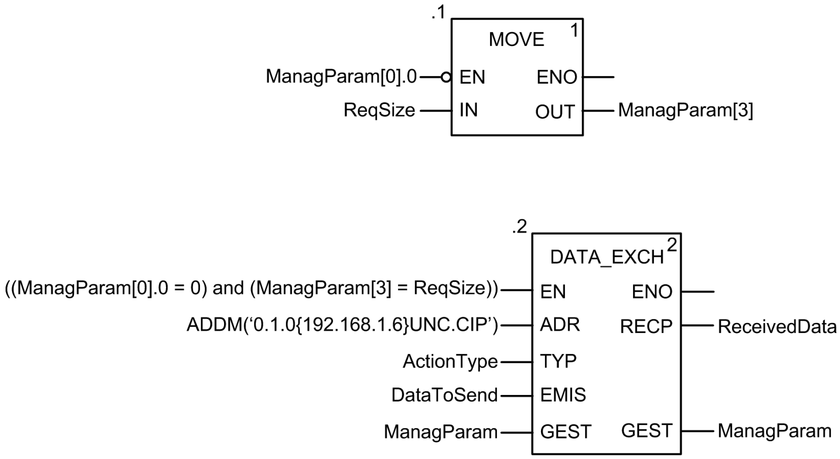
Overview

This example of unconnected explicit messaging shows the use of the DATA_EXCH function block to read data from a remote device (for example, an STBNIP2212 network interface module at IP address 192.168.1.6) with the Read_Holding_Registers service of the Modbus Object.

You can perform the same explicit messaging service using the **EtherNet/IP Explicit Message** window, page 197.

Implement the DATA_EXCH Function Block

To implement the DATA_EXCH function block, create and assign variables for these blocks:



Declare Variables

With the variables defined, you can use different variable names in your explicit messaging configuration. View the configured variables in the **Variables** tab in the **Data Editor**.

Configure the Address Variable

The *Address* variable identifies the explicit message originating device (in this case, the Ethernet communications module) and the target device. The *Address* variable does not include the X Way address elements {Network.Station} because the bridge is not through another controller-module station. Use the `ADDM` function to convert this character string to an address: `ADDM`(‘0.1.0{192.168.1.6}UNC.CIP’)

In this case:

- rack = 0
- module (slot number) = 1
- channel = 0
- remote device IP address = 192.168.1.6
- message type = unconnected
- protocol = CIP

Configure the ActionType Variable

The *ActionType* variable identifies the function type for the `DATA_EXCH` function block:

Variable	Description	Value
ActionType	Transmission followed by wait for response	01 hex

Configure the DataToSend Variable

The DataToSend variable identifies the type of explicit message and the CIP request:

Variable	Description	Value
DataToSend[0]	CIP request service information: • High byte = request size in words: 02 hex (2 decimal) • Low byte = service code: 4E hex (78 decimal)	024E hex
DataToSend[1]	CIP request class information: • High byte = class: 44 hex (68 decimal) • Low byte = class segment: 20 hex (32 decimal)	4420 hex
DataToSend[2]	CIP request instance information: • High byte = instance: 01 (1 decimal) • Low byte = instance segment: 24 hex (36 decimal)	0124 hex
DataToSend[3]	Location of first word to be read: • High byte = 00 hex (0 decimal) • Low byte = 31 hex (49 decimal)	0031 hex
DataToSend[4]	Number of words to read: • High byte = attribute: 00 hex (0 decimal) • Low byte = attribute segment: 01 hex (1 decimal)	0001 hex

View the CIP Response

The *ReceivedData* variable array consists of the entire data buffer. Use an EcoStruxure Control Expert Animation table to view the *ReceivedData* variable array:

Step	Action
1	In EcoStruxure Control Expert, open the **Project Browser** (**Tools > Project Browser**).
2	Right-click **Animation Tables** and scroll to **New Animation Table** to open the **New Animation Table** dialog box.
3	Configure the new animation table (give it a name in the **Name** field) and click **OK** to close the **New Animation Table** dialog box.
4	Right click the new animation able in the Project Browser and scroll to **Properties** to open the **Properties** dialog box.
5	Edit these values: • **Name:** Enter a table name. **NOTE:** Enter *ReceivedData* to follow this procedure through these steps. • **Functional module:** Accept the default (*<None>*). • **Comment:** Enter a comment here (optional). • **Number of animated characters:** Enter *49* to represent the size of the data buffer in words. • **Temporary Table:** Use this button to make the copied animation table temporary or permanent. • **Include in upload only:** Include the upload information when the project is transferred to the controller module. **NOTE:** ◦ You can change the **Include in upload only** property only if *Animation tables* is selected (checked) in the *Project Settings* (**Tools > Project Settings > General > PLC embedded data > Upload information > Animation tables**). ◦ Refer to the *Update of Upload Information* and *General Project Settings* topics in the *EcoStruxure™ Control Expert, Operating Modes* guide.
6	Click **OK** to close the dialog box.
7	In the animation table's **Name** column, enter the name of the variable assigned to the `RECP` pin (*ReceivedData*) and click **Enter** to see the *ReceivedData* variable appear in the animation table.
8	Expand the *ReceivedData* variable to view its word array and the CIP response in the *ReceivedData* variable. **NOTE:** Each array entry presents two bytes of data in little endian format in which the least significant byte is stored in the smallest memory address. For example, `CE` in word[0] is the lower byte, and `00` is the upper byte.

EtherNet/IP Explicit Message Example: Write Modbus Object

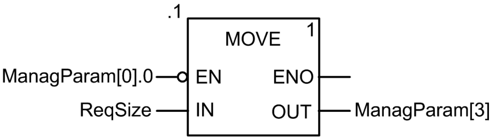
Overview

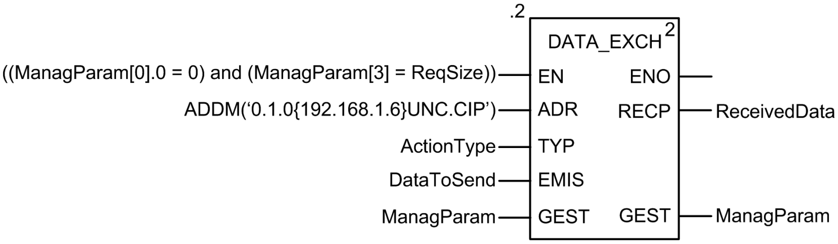
This example of unconnected explicit messaging describes the use of the DATA_EXCH function block to write data to a remote device at IP address 192.168.1.6 using the *Write_Holding_Registers* service of the Modbus object.

You can perform the same explicit messaging service using the **EtherNet/IP Explicit Message** window in the EcoStruxure Control Expert DTM, page 197.

Implement the DATA_EXCH Function Block

To implement the DATA_EXCH function block, create and assign variables for these blocks:





Declare Variables

These variables are defined for the `DATA_EXCH` function block:

- `ActionType`
- `DataToSend`
- `MangParam`
- `ReceivedData`
- `ReqSize`

NOTE: You can use different variable names in your explicit messaging configurations.

Configure the Address Variable

The *Address* variable identifies the explicit message originating device (in this case, the communications module) and the target device. The *Address* variable does not include the X Way address elements {Network.Station} because we are not using a controller module at a bridge. Use the `ADDM` function to convert this character string to an address:

`ADDM`('0.1.0{192.168.1.6}UNC.CIP'), where:

- rack = *0*
- module (slot number) = *1*
- channel = *0*
- remote device IP address = *192.168.1.6*
- message type = *unconnected*
- protocol = *CIP*

Configure the ActionType Variable

The *ActionType* variable identifies the function type for the `DATA_EXCH` function block:

Variable	Description	Value
ActionType	Transmission followed by wait for response	01 hex

Configure the DataToSend Variable

The *DataToSend* variable identifies the type of explicit message and the CIP request:

Variable	Description	Value
DataToSend[0]	CIP request service information: • High byte = request size in words: 02 hex (2 decimal) • Low byte = service code: 50 hex (80 decimal)	0250 hex
DataToSend[1]	CIP request class information: • High byte = class: 44 hex (68 decimal) • Low byte = class segment: 20 hex (32 decimal)	4420 hex
DataToSend[2]	CIP request instance information: • High byte = instance: 01 (1 decimal) • Low byte = instance segment: 24 (36 decimal)	0124 hex
DataToSend[3]	Location of first word to write (+ %MW1): • High byte = 00 (0 decimal) • Low byte = 00 (0 decimal)	0000 hex
DataToSend[4]	Number of words to write: • High byte = attribute: 00 hex (0 decimal) • Low byte = attribute segment: 01 hex (1 decimal)	0001 hex
DataToSend[5]	Data to write: • High byte = attribute: 00 hex (0 decimal) • Low byte = attribute segment: 6F hex (111 decimal)	006F hex

View the Response

Use an EcoStruxure Control Expert animation table to display the *ReceivedData* variable array. The *ReceivedData* variable array consists of the entire data buffer.

View the CIP response:

Step	Action
1	In EcoStruxure Control Expert, open the **Project Browser** (**Tools > Project Browser**).
2	Open the **New Animation Table** dialog box. **NOTE:** Right-click **Animation Tables** and scroll to **New Animation Table**.
3	Select **New Animation Table** in the pop-up menu to see the new animation table and its properties dialog box.
4	In the **Properties** dialog box, edit these values: • **Name:** Enter a table name. **NOTE:** Enter *ReceivedData* to follow this procedure through these steps. • **Functional module:** Accept the default *(<None>).* • **Comment:** Enter a comment in this field (optional). • **Number of animated characters:** This value represents the size of the data buffer in words. Enter *49* as a value to follow these steps.
5	Click **OK** to close the dialog box.
6	In the animation table's **Name** column, enter the name of the variable assigned to the `RECP` pin) *ReceivedData*) and click **Enter** to see the *ReceivedData* variable appear in the animation table.
7	Expand the *ReceivedData* variable to display its word array and view the CIP response contained in the *ReceivedData* variable. **NOTE:** Each array entry presents two bytes of data in little endian format, where the least significant byte is stored in the smallest memory address. For example, *D0* in *word[0]* is the lower byte, and *00* is the upper byte.

Modbus TCP Explicit Messaging Using DATA_ EXCH

Overview

This section shows you how to configure DATA_EXCH function block parameters for Modbus TCP explicit messages.

Modbus TCP Explicit Messaging Function Codes

Overview

You can execute Modbus TCP explicit messages using either an EcoStruxure Control Expert DATA_EXCH function block or the Modbus Explicit Message Window.

NOTE: Configuration modifications that are made to an Ethernet module are not saved to the operating parameters stored in the controller module and are therefore not sent by the controller to the module upon startup.

Function Codes

The function codes supported by the EcoStruxure Control Expert graphical user interface include these standard explicit messaging functions:

Function Code (dec)	Description
1	Read bits (%M)
2	Read input bits (%I)
3	Read words (%MW)
4	Read input words (%IW)
15	Write bits (%M)
16	Write words (%MW)

NOTE: You can use the DATA_EXCH function block to execute a Modbus function through the program logic. Refer to the Modbus IDA website for a complete list of Modbus functions (https://www.modbus.org/).

Configure Modbus TCP Explicit Messaging Using DATA_ EXCH

Introduction

When you use the DATA_EXCH block to create an explicit message for a Modbus TCP device, configure this block the same way you configure it for other Modbus communications. For more information, refer to the description of explicit messaging through the DATA_EXCH block, page 167.

Configure ADDM Block Unit ID Settings

When you configure the DATA_EXCH block, use the ADDM block to set the DATA_EXCH block's Address parameter. The ADDM block presents the configuration format ADDM('rack. slot.channel[ip_address]UnitID.message_type.protocol') where:

Parameter	Description
rack	This number is assigned to the rack that contains the communication module.
slot	The communication module occupies this position (slot number) in the rack.
channel	This is the communication channel (set to *0*).
ip_address	This is the IP address of the remote device (for example, *192.168.1.7*).
Unit ID	This is the destination node address, also known as the Modbus Plus on Ethernet Transporter (MET) mapping index value. **NOTE:** The value of *Unit ID* in a Modbus message indicates the destination of the message.
message_type	The message type is identified by a three-character string (**TCP**).
protocol	The protocol is identified by a three-character string (**MBS**).

Contents of the Received_Data Parameter

The *Received_Data* parameter contains the Modbus response. The length of the response varies, and is reported by `Management_Param[3]` after the response is received. The format of the Modbus response is described, below:

Offset (words)	Length (bytes)	Description
0	2	First word of the Modbus response: • High byte (MSB): ◦ if successful: Modbus Function Code ◦ if not: Modbus function code + 80 hex • Low byte (LSB): ◦ if successful: depends on the request ◦ if not: Modbus exception code
1	Length of the `Received_Data` parameter – 2	Remainder of the Modbus response: depends on the specific Modbus request)

NOTE:

- Structure the response in little endian order.
- For some detected errors, *Received_Data* is also used to evaluate the type of detected error along with *Management_Param*.

Modbus TCP Explicit Message Example: Read Register Request

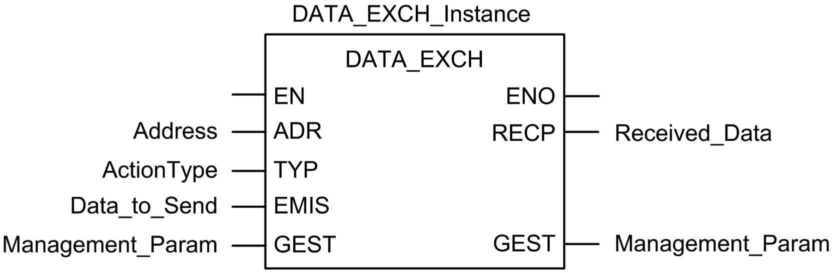
Introduction

Use the DATA_EXCH function block to send a Modbus TCP explicit message to a remote device at a specific IP address to read a single word located in the remote device.

The Management_Param Data_to_Send, and Received_Data parameters define the operation. You can configure EN and ENO as additional parameters.

Implement the DATA_EXCH Function Block

To implement the DATA_EXCH function block, create and assign variables for these inputs and outputs:



Configure the Address Variable

The *Address* variable identifies the explicit message originating device and the target device. This variable does not include the X Way address elements {Network.Station} because you are not bridging through another controller-module station. Use the ADDM function to convert this character string to an address:

ADDM(‘0.1.0{192.168.1.7}TCP.MBS’), where:

- rack = *0*
- module (slot number) = *1*
- channel = *0*
- remote device IP address = *192.168.1.7*
- message type = *TCP*
- protocol = *Modbus*

Configure the ActionType Variable

The *ActionType* variable identifies the function type for the `DATA_EXCH` function block:

Variable	Description	Value
ActionType	Transmission followed by wait for response	01 hex

Configure the DataToSend Variable

The *DataToSend* variable contains the target register address and the number of registers to read:

Variable	Description	Value
DataToSend[0]	• High byte = Most significant byte (MSB) of register address 15 hex (21 decimal) • Low byte = function code: 03 hex (03 decimal)	1503 hex
DataToSend[1]	• High byte = Most significant byte (MSB) of the number of registers to read: 00 hex (0 decimal) • Low byte = Least significant byte (LSB) of register address: 0F hex (15 decimal)	000F hex
DataToSend[2]	• High byte = not used: 00 hex (0 decimal) • Low byte = Least significant byte (LSB) of the number of registers to read: 01 hex (1 decimal)	0001 hex

NOTE: For detailed information about M580 network topologies, refer to the *Modicon M580, Frequently Used Architectures, System Guide* and *Modicon M580, Complex Topologies, System Guide*.

View the Response

Use an EcoStruxure Control Expert animation table to display the `ReceivedData` variable array. This array consists of the entire data buffer.

Display the Modbus TCP response:

Step	Action
1	In EcoStruxure Control Expert, select **Tools > Project Browser**.
2	In the **Project Browser**, select the **Animation Tables** folder, and right-click to open a pop-up menu.
3	Select **New Animation Table** in the pop-up menu to open a new animation table and its properties dialog box.
4	In the **Properties** dialog box, edit these values: • **Name:** Enter a table name. **NOTE:** Enter *ReceivedData* to follow this procedure through these steps. • **Functional module:** Accept the default (<*None*>). • **Comment:** Enter a comment here (optional). • **Number of animated characters:** Enter *100* to represent the size of the data buffer in words.
5	Click **OK** to close the dialog box.
6	In the animation table's **Name** column, enter the name of the variable that is assigned to the data buffer *(ReceivedData)* and click **Enter** to see the *ReceivedData* variable.
7	Expand the *ReceivedData* variable to display its word array, where you can view the CIP response contained in the *ReceivedData* variable. **NOTE:** Each array entry presents two bytes of data in little endian format. For example, '03' in word[0] is the low byte, and '02' is the high byte.

Send Explicit EtherNet/IP and Modbus TCP Messages in EcoStruxure Control Expert

Overview

Introduction

With the Modbus Explicit Message window in the EcoStruxure Control Expert DTM, page 200, you can send an explicit message to a Modbus TCP module or distributed device on the network. You can also use explicit messaging to perform many different services, but not all Modbus TCP devices support all services.

Connect the DTM

Before you can configure explicit messaging for EtherNet/IP or Modbus TCP devices, make the connection between the DTM for the target communication module and the physical module:

Step	Action
1	In the **DTM Browser**, find the name that is assigned to your Ethernet communications module.
2	Right-click the module name.
3	Scroll to **Connect**.

Send Explicit Messages to EtherNet/IP Devices

Overview

Use the **EtherNet/IP Explicit Message** window in the EcoStruxure Control Expert DTM to send an explicit message to an EtherNet/IP module or distributed device on the network.

Explicit messages can be sent as connected or unconnected messages:

- *connected*: With connected messaging, node resources are reserved in advance of the data transfer and are dedicated and available.
- *unconnected*: With unconnected messaging, a CIP connection to the destination is not established before the point-to-point transfer of data.

You can use explicit messaging to perform many different services. Not every EtherNet/IP device supports every service.

The EtherNet/IP explicit message configuration window presents an example of both the configuration of an EtherNet/IP explicit message and the response. The explicit message is addressed to a distributed module to obtain diagnostic information.

Send Explicit Messages

Use these instructions to execute an EtherNet/IP explicit message.

Access the configuration:

Step	Action
1	In the **DTM Browser**, select the communication module that is upstream of the target device.
2	Right-click the module and select **Device menu > Additional functions > EtherNet/IP Explicit Message** to access the configuration.

Configure the messages in the **EtherNet/IP Explicit Message** window:

Field	Description
IP Address	The IP address of the target device identifies the target of the explicit message.
Class	The class identifier of the target device is used to constructs the message path. It is an integer value (1...65535). (See the note below.)
Instance	The class instance of the target device is used to construct the message path. It is an integer value (1...65535). (See the note below.)
Attribute	Select (check) this box to enable the optional *Attribute* field. The specific device attribute (or property) is the target of the explicit message that is used to construct the message path. It is an integer value (1...65535). (See the note below.)
Number	The integer (1...127) associated with the service to be performed by the explicit message. If you select **Custom Service** as the named service, enter a service number. For other services, this is a read-only field.
Name	Select the service the explicit message is intended to perform.
Enter Path	Select (check) this box to enable the message path field and manually enter the entire path to the target device: • This field is available only when **Advanced Mode** is enabled. • When available, the implementation of this field is optional.
Data	The data to be sent to the target device, for services that send data.
Messaging	Select the type of explicit message to send (connected or unconnected, page 197): • **Connected** • **Unconnected**
Repeat 500 ms	Select (check) this box to re-send the explicit message every 500 ms.

NOTE: For *Class*, *Instance*, and *Attribute*, refer to the user manual for your specific EtherNet/IP device to obtain the appropriate values.

Send the message to the device:

Step	Action
1	After the explicit message is configured, click **Send to Device** to send the data in the **Response (hex)** area to the configuration tool by the target device in hexadecimal format. **NOTE:** Messages in the **Status** area indicate whether or not the explicit message has succeeded.
2	Click **Close** to close the window.

Send Explicit Messages to Modbus TCP Devices

Overview

Use the **Modbus Explicit Message** window in the EcoStruxure Control Expert DTM to send an explicit message from an EtherNet/IP module or distributed device on the network.

You can use explicit messaging to perform many different services. Not every Modbus TCP device supports every service.

The Modbus TCP explicit message configuration window shows both the configuration of a Modbus TCP explicit message and the response.

Send Explicit Messages

Execute a Modbus TCP explicit message:

Step	Action
1	In the **DTM Browser**, select the communication module that is upstream of the target device.
2	Open the **Modbus TCP Explicit Message** dialog box. (Right-click the module and select **Device menu > Additional functions > Modbus TCP Explicit Message**).
3	Configure explicit messages in these fields: • **IP Address:** The IP address of the target device identifies the target of the explicit message. • **Start Address:** A component of the addressing path. • **Quantity:** A component of the addressing path. • **Read Device Id Code:** Read-only identification of the service that the explicit message is intended to perform. • **Object Id:** (read-only) Specify the object the explicit message is intended to access.
4	Refer to your Modbus TCP device user manual for *Start Address*, *Quantity*, *Read Device Id Code*, and *Object Id* values: • **Unit Id:** This number identifies the connection target. **NOTE:** Consult the manufacturer’s user manual for the specific target device to find its Unit ID. • **Number:** This read-only integer (0 ... 255) is associated with the service performed by the explicit message. • **Name:** Select the service that the explicit message is intended to perform. • **Repeat 500ms:** Select (check) this box to re-send the explicit message every 500 ms. Leave this box deselected.

Step	Action
5	After your explicit message is configured, click **Send to Device**: • Data in the **Response** area was sent to the configuration tool by the target device in hexadecimal format. • Messages in the **Status** area indicate the success of the explicit message transmission.
6	Click **Close** to close the window.

Implicit Messaging

Introduction

Use implicit messaging to create a communications link between the BMENOC0302(H) module on an M580 rack and network devices.

The BMENOC0302(H) module manages the communications link to facilitate the exchange of I/O data between the M580 controller module and Modbus TCP and EtherNet/IP devices on the network. Using this module as a local slave is another example of implicit messaging.

Add an EtherNet/IP Device to the Network

Introduction

The following topics address a sample EcoStruxure Control Expert application that includes EtherNet/IP communications to a modular distributed I/O device (an STBNIC2212 network interface module in an Advantys island). Instructions will be provided in accordance with these process steps:

- Add an STBNIC2212 EtherNet/IP network interface module to your EcoStruxure Control Expert application.
- Configure the STBNIC2212 module.
- Configure EtherNet/IP connections to link the Ethernet communications module and the STBNIC2212 network interface module.
- Configure I/O items for the Advantys island.

NOTE: The STBNIC2212 is Schneider Electric’s EtherNet/IP network interface module for Advantys islands.

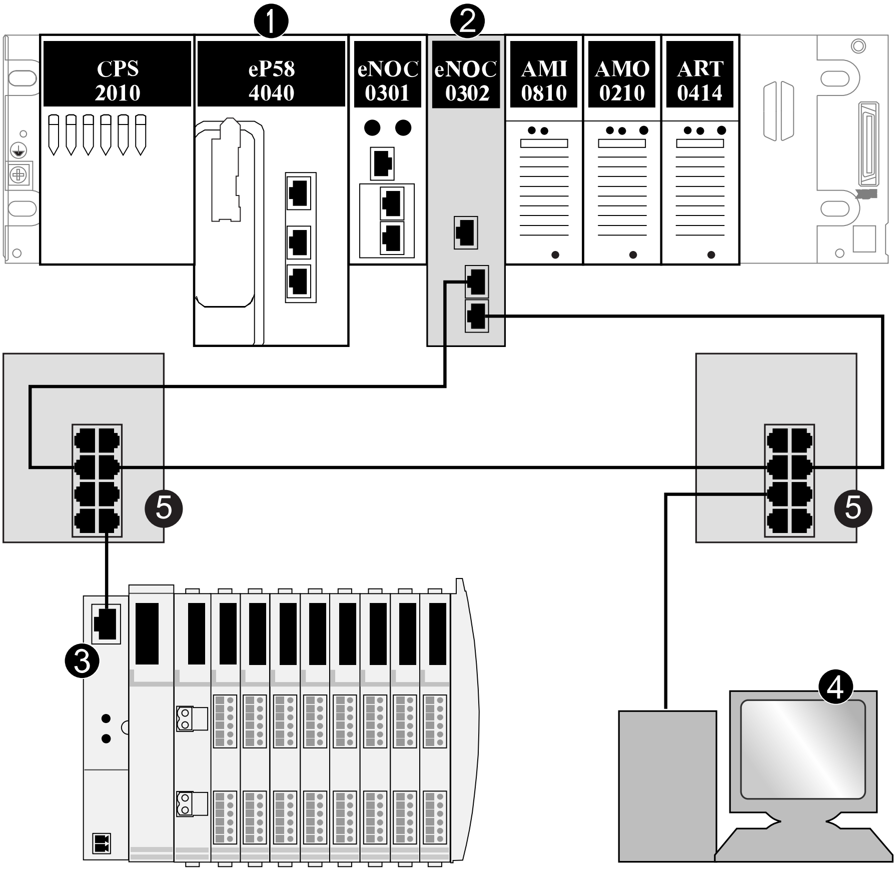
Set Up Your Network

Introduction

Use this information to establish communications between a BMENOC0302(H) module in an M580 rack and an Advantys STBNIC2212 network interface module (NIM).

Network Topology

The Ethernet network devices used in this configuration include these:



1. An M580 controller module is installed on the local rack.
2. A BMENOC0302(H) Ethernet communications module in slot 3 runs the DIO scanner service.

3. An STBNIC2212 NIM manages communications to an Advantys island.
4. A computer runs the EcoStruxure Control Expert software.
5. Dual-ring switches support network traffic.

NOTE:

- Use the IP addresses from your own configuration for the BMENOC0302(H) module, the computer, and the STBNIC2212 module when you configure a network like this.
- Configure the M580 controller module in the EcoStruxure Control Expert software that runs in the computer. In this case, the computer is indirectly wired to the controller module Ethernet port through the Ethernet switch. Alternatively, you can bypass the switch and wire the computer directly to the controller module service port.

Add an STBNIC2212 Device

Overview

You can use the EcoStruxure Control Expert device library to add a remote device to your project. Only a remote device that is part of your EcoStruxure Control Expert device library can be added to your project. The examples in this guide use an STBNIC2212 network interface module as the remote device.

Alternatively, when a remote device is included in your device library, you can use automatic device discovery to populate your project. Perform automatic device discovery with a **Field bus discovery** command to a communication module selected in the **DTM Browser**.

Add an STBNIC2212 Remote Device

NOTE: This example uses a DTM that is specific to the STBNIC2212 network interface module. If you do not have a device-specific DTM, EcoStruxure Control Expert provides a generic device DTM.

Add the STBNIC2212 to your project:

Step	Action
1	In the **DTM Browser**, right-click the DTM that corresponds to the Ethernet communication module.
2	Scroll to **Add** to open the **Add** dialog box.
3	In the **Protocol** pull-down menu, select *EtherNet IP* to view EtherNet/IP devices in the table.
4	Select **STBNIC2212 (from EDS)** in the **Device** column. **NOTE:** Click a column name to sort the list of available devices. (For example, click **Device** to view the items in the first column in alphabetical order.)
5	Click the **Add DTM** button to open the **Properties of device** dialog box.
6	Configure the DTM name for the STBNIC2212 device in the **Name** field or accept the default. **NOTE:** EcoStruxure Control Expert uses this name as the base for both structure and variable names. The name is the only editable parameter on this tab. (Other parameters are read-only.)
7	Click **OK** to view the STBNIC2212 DTM as a child DTM to the module DTM in the **DTM Browser**.

You can now configure the device you added to the project.

Configure STBNIC2212 Properties

Introduction

Use EcoStruxure Control Expert to edit the settings for the STBNIC2212 device.

Access the Device Properties

View the **Properties** tab:

Step	Action
1	Disconnect from a device.
2	In the **DTM Browser**, double-click the DTM for the BMENOC0302(H) module in slot 3 (<192.168.20.10> BMENOC0302_slot3) to access the configuration.
3	To view the associated local slave instances in the navigation tree, expand (**+**) **Device List**, page 152.
4	Select the device that corresponds to the name **NIC2212_01** to view these tabs: • **Properties**, page 206 • **Address Setting**, page 207

Properties Tab

Configure the **Properties** tab to perform these tasks:

- Add the STBNIC2212 to the configuration.
- Remove the STBNIC2212 from the configuration.
- Edit the base name for variables and data structures used by the STBNIC2212.
- Indicate how input and output items are created and edited.

In the **Properties** tab, use these values and names that you configured elsewhere for the STBNIC2212 EtherNet/IP device:

Field	Parameter	Description
Properties	*Number*	Accept the auto-generated value.
	Active Configuration	Accept the default (**Enabled**).
	Comment	Enter a relevant comment.
IO Structure Name	*Structure Name*	EcoStruxure Control Expert automatically assigns a structure name based on the variable name, in this case **T_STBNIC2212_from_EDS**.
	Variable Name	Accept the auto-generated variable name (based on the alias name): **STBNIC2212_from_EDS**.
Items Management	*Import Mode*	Select **Manual** from this pull-down menu.
	Reimport Items	Click this button to import the I/O items list from the device DTM and overwrite the manual I/O item modifications. Enabled only when **Import mode** is set to **Manual**.

Click **Apply** to save your changes and leave the window open.

> **NOTE:** The table above provides the specific configuration for the STBNIC2212 EtherNet/IP device. Elsewhere in this guide is a complete description of parameters in the **Properties** tab.

Address Setting Tab

Use the **Address Setting** tab to enable the DHCP client in the STBNIC2212 network interface module. When the DHCP client is enabled in the remote device, it obtains its IP address from the DHCP server in the Ethernet communication module

Configure the **Address Setting** page to perform these tasks:

- Configure the IP address for a device.
- Enable or disable DHCP client software for a device.

In the **Address Setting** tab, use these values and names that you already configured for the STBNIC2212 EtherNet/IP device:

Field	Parameter	Description
IP Configuration	**IP Address**	Enter the IP address *192.168.1.6*.
Address Server	**DHCP for this device**	Select **Enabled** from the pull-down menu.
	Identified by	Select **Device Name** from the pull-down menu.
	Identifier	Accept the default setting (based on the **Name**).
	Mask	Accept the default value.
	Gateway	Accept the default value.

Click the **Enter** key on your keyboard to display and click the **Apply** button. You can now configure the connection between the communication module and the remote device.

NOTE: The table above provides the specific configuration for the STBNIC2212 EtherNet/IP device. Elsewhere in this guide is a complete description of parameters in the **Address Setting** tab, page 103.

Configure EtherNet/IP Connections

Overview

An EtherNet/IP connection provides a communication link between two or more devices. Properties for a single connection can be configured in the DTMs for the connected devices.

The following example presents settings for a connection between the Ethernet communication module and a remote STBNIC2212 network interface module. Configuration modifications are made to the DTMs for each device.

When making DTM modifications, disconnect the selected DTM from the module or device, page 78.

The BMENOC0302(H) module can manage traffic of up to 12 kpps, including EtherNet/IP and Modbus frames.

You must consider the broadcast frames and EtherNet/IP multicast frames from the other products in the network (even when the BMENOC0302(H) module is not the destination of the multicast module) in the calculation of the number of frames received by the BMENOC0302(H).

When setting up your network architecture with a BMENOC0302(H), use the multicast mode; otherwise, use a point-to-point connection or a switch with an IGMP snooping feature.

Access the Connection Information

Step	Action
1	Double-click the DTM for the BMENOC0302(H) in slot 3 (<192.168.20.10> BME_NOC0302) to access the configuration.
2	To view the associated local slave instances in the navigation tree, expand (**+**) the **Device List**, page 152.
3	Expand (**+**) the device that corresponds to the name **NIC2212_01**.
4	Select **Read Input/ Write Output Data** to view the **Connection Settings** and **Connection Information** tabs.

Connection Settings

EcoStruxure Control Expert creates a connection between a communication module and remote device when the remote device is added to the EcoStruxure Control Expert project.

You can modify the connection in the DTM for the remote device. Some connection parameters, however, are configured on the **Connection Settings** tab in the DTM for the communication module. Use settings that are appropriate to your application:

Parameter	Description
Connection Bit	This (read-only) offset bit applies to the health bit and the control bit for this connection. **NOTE:** The EcoStruxure Control Expert DTM auto-generates offset values.
Request Packet Interval (RPI)O->T	This is the refresh period for this output connection in ms (2 ... 65535). Enter *30* ms. **NOTE:** This parameter can be set in the DTM for the communication module or the remote device.
Time-out Multiplier	This setting, multiplied against the RPI, produces a value that triggers an inactivity timeout. Setting selections include: x4, x8, x16, x32, x64, x128, x256 and x512. **NOTE:** To be consistent with the STBNIC2212 EtherNet/IP device, use the value *x4*. You can view the *Time-out Multiplier* parameter only when EcoStruxure Control Expert operates in **Advanced Mode**.
Request Packet Interval (RPI)T->O	This is the refresh period for this input connection in ms (2 ... 65535). Enter *30* ms. **NOTE:** This parameter can be set in the DTM for the communication module or the remote device.
Input Fallback Mode	This value is *Set To Zero* when communication is lost.

Click **OK** to save your settings.

NOTE: This information page is read-only when the communication module is selected. Set this information in the DTM for the remote device.

Configure Connection Settings in the Remote Device DTM

Connections between a communication module and remote device can be created and edited in the DTM for the remote device.

Configuration modifications are made to the connection that EcoStruxure Control Expert automatically created when the remote device was added to the project. Use settings that are appropriate for your unique application:

Step	Action
1	In the **DTM Browser**, expand the master DTM for the BMENOC0302(H) in slot 3 (<192.168.20.10> BMENOC0302_slot3).
2	Double-click the device DTM that corresponds to the name **NIC2212_01** to open the configuration window.
3	Expand (**+**) **NIC2212_01** in the navigation pane to view the connection type. If the connection type is not of the type **Read Input / Write Output Data**, delete the existing connection and add a new one: a. Select the connection in the left pane. b. Click the **Remove Connection** button to remove the existing connection. c. Click the **Add Connection** button to open the **Select the connection to add** dialog box. d. Scroll to the **Read Input / Write Output Data** connection type. e. Click **OK** to close the **Select the connection to add** dialog box and add the new connection node to the **NIC2212_01**. f. Click **Apply** to save the new connection and leave the configuration window open.

Connection Tab

In the navigation pane, select **Read Input / Write Output Data** to open the **Connection** tab and edit the settings. These descriptions suggest values that are consistent with the EtherNet/IP communications to the STBNIC2212 network interface module:

Parameter	Description
RPI	For this refresh period for the connection, accept the value of *30* ms for the requested packet interval (RPI). (Set this parameter in the DTM for the communication module or the remote device.)
Input size	The number of bytes (0 ... 509) configured in the STBNIC2212 module. **NOTE:** Enter *19* to reserve 20 bytes of input memory.
Input mode	Transmission type: • Multicast • Point to Point Accept the default (*Multicast*).
Input type	Ethernet packet type (fixed or variable length) to be transmitted. (Only **Fixed** length packets are supported.)
Input priority	The transmission priority value depends upon the device DTM. These are the available values: • Low • High • Scheduled Accept the default selection (*Scheduled*). **NOTE:** For remote modules that support more than one priority value, you can use this setting to specify the order in which the Ethernet communication module handles packets. For more information, refer to the topic that describes the prioritization of QoS packets, page 372.
Input trigger	These are the available transmission trigger values: • Cyclic • Change of state or application For input I/O data, select **Cyclic**.
Output size	The number of bytes configured in the target STBNIC2212 module in increments of 4 bytes (2 words). Enter *6* to reserve 8 bytes of output memory.
Output mode	Accept the default (**Point to Point**).
Output type	(Read-only). Only **Fixed** length packets are supported.
Output priority	Accept the default (**Scheduled**).

Click **Apply** to save your settings and leave the window open.

Identity Check Tab

Use the **Identity Check** tab to set rules for comparing the identity of the network devices (as defined by their DTM or EDS files) against the identity of the network device.

Use the *Check Identity* parameter to set the rules that the BMENOC0302(H) uses to compare the configured device to the remote device. Select one of these options from the pull-down menu in the **Value** column:

- *Must match exactly:* The DTM or EDS file exactly matches the remote device.
- *Disable:* No checking occurs. The identity portion of the connection is filled with zero values by default.
- *Must be compatible:* If the remote device is not the one defined by the DTM/EDS, it emulates the DTM/EDS definitions.
- *Custom:* Enable the following parameter settings, to be set individually.
- *None:* No checking occurs. The identity portion of the connection is omitted.

Edit the settings in the **Identity Check** tab:

Parameter	Description
Check Identity	This property defines the rule that EcoStruxure Control Expert uses to compare the configured with the actual remote device. These are the available settings: • *Must match exactly:* The DTM or EDS file exactly matches the remote device. • *Disable:* The checking function does not run. The identity portion of the connection is filled with zero values. • *Must be compatible:* When the remote device is not the same as defined by the DTM/EDS, it emulates the DTM/EDS definitions. • *Custom:* Enable the following parameter settings individually. • *None:* No checking occurs; the identity portion of the connection is omitted.
Compatibility Mode	*True:* For each of the following selected tests, the DTM/EDS and remote device are compatible.
	False: For each of the following selected tests, the DTM/EDS and remote device match exactly.
Minor Version	Make a selection for each of these parameters: • *Compatible:* Include the parameter in the test. • *Not checked:* The parameter is not included in the test.
Major Version	
Product Code	
Product Type	
Product Vendor	

Click **OK** to save your settings and close the window.

You can now configure the I/O settings.

Configure I/O Items

Overview

The final task in this example is to add I/O items to the configuration of the STBNIC2212 and its I/O modules. To accomplish this:

- Use the Advantys configuration software to identify the relative position of each I/O module's inputs and outputs.
- Use the EcoStruxure Control Expert **Device Editor** to create input and output items that define each item's name and data type.

I/O Item Types and Sizes

The goal is to create a collection of input items and output items that equal the input size and output size specified for the STBNIC2212. (Refer to the *Premium using EcoStruxure™ Control Expert, Hot Standby, User Manual*.)

The EcoStruxure Control Expert **Device Editor** provides great flexibility in creating input and output items. You can create input and output items in groups of one or more single bits, 8-bit bytes, 16-bit words, 32-bit words, or 32-bit IEEE floating values. The number of items you can create depends on the data type and size of each item.

Map the Input and Output Items

Use the **Fieldbus Image** page of the **I/O Image Overview** window in the Advantys configuration software to identify the number and type of I/O items you create:

Step	Action
1	In the Advantys configuration software, select **Island > I/O Image Overview** to open the **I/O Image** window to the **Fieldbus Overview** page.
2	Select the first cell (word 1, cell 0) in the **Input Data** table to view a description of the cell data and its source module.
3	Make a note of the word, bit(s), module, and item information for that cell.
4	Repeat the above steps for each cell that contains an *S* or an integer.

NOTE:

- The fieldbus image presents input and output data in 16-bit words (starting with word *1*). Rearrange this data for the EcoStruxure Control Expert Ethernet Configuration Tool, which presents the same data in 8-bit bytes (starting with byte *0*).
- When you create items, align items of data type `WORD` and `DWORD`:
  - `WORD` items: Align these items on a 16-bit boundary.
  - `DWORD` items: Align these items on a 32-bit boundary.

This example shows you how to create input bytes and output bytes. To use space efficiently, this example creates items in this sequence:

- input bit items
- input byte and word items
- output bit items
- output byte and word items

Refer to the description of the appropriate input and output sizes for input and output items in EcoStruxure Control Expert, page 86.

Create Input Bit Items

Create input bit items for the STBNIC2212 example, beginning with discrete inputs for the STBNIC2212 status:

Step	Action
1	Open the **Items** configuration in EcoStruxure Control Expert.
2	Select the **Input (bit)** tab and follow directions to create input bit items. Use the default root name to represent the device status (DDI3232_in_data) in the **Default Items Name Root** field.
3	In the **Items List**, select the first two rows in the table. (These rows represent bits 0-1 in byte.)
4	Click the **Define Item(s)** button to open the **Item Name Definition** dialog box. **NOTE:** An asterisk (*) in the **Item Name** field indicates that discrete items with the same root name are created.
5	Accept the default **Item Name** and click **OK** to create two discrete input items.
6	Click **Apply** to save the items and leave the page open.
	Repeat these steps to create additional groups of discrete input items.

Create Input Items

Create input items for the STBNIC2212 example, beginning with an input data byte that contains the low byte status for the STBNIC2212 module:

Step	Action
1	Select the **Input** tab. **NOTE:** In this example, both the **Offset/Device** and **Offset/Connection** columns represent the byte address. The items you create are 8-bit bytes or a 16-bit words.
2	Enter **NIC22212_01_LO_st** in the **Default Item Name Root** field.
3	Select a single row at byte 8.
4	Click the **Define Item(s)** button to open the **Item Name Definition** dialog box.
5	Select **Byte** as the **New Item(s) Data Type**.
6	Click **OK** to create the byte.
7	Click **Apply** to save the items and leave the page open.
8	Repeat these steps to create additional byte or word input items.

Create Output Bit Items

Create output bit items for the STBNIC2212 example, beginning with two output bits for an STBDDO3200 module:

Step	Action
1	Select the **Output (bit)** tab. **NOTE:** In this example, both the **Offset/Device** and **Offset/Connection** columns represent the byte address of an output. The **Position in Byte** column indicates the bit position (within the byte) of each discrete output item.
2	Enter **DDO3200_out_data** in the **Default Item Name Root** field.
3	Select the rows that correspond to bits 0 and 1 in byte 0 (the first two rows).
4	Click the **Define Item(s)** button to open the **Item Name Definition** dialog box. **NOTE:** An asterisk (*) in the **Item Name** field indicates that discrete items with the same root name are created.
5	Accept the default **Item Name** and click **OK** to create two discrete output items.
6	Click **Apply** to save the items and leave the page open.
7	Repeat these steps to create additional output items.

Create Numeric Output Items

Create output items for the STBNIC2212 example, beginning with an output data word for the STBAVO1250 module:

Step	Action
1	Select the **Output** tab. **NOTE:** In this example, both the **Offset/Device** and **Offset/Connection** columns represent the byte address. The items you create are 16-bit words comprising two bytes.
2	Enter **AVO1250_CH1_out_data** in the **Default Item Name Root** field.
3	Starting at the next available whole word, select two rows (rows 2 and 3).
4	Click the **Define Item(s)** button to open the **Item Name Definition** dialog box.
5	Click **OK** to create the output word.
6	Click **Apply** to save the items and leave the page open.
7	Repeat these steps to create a word for the STBAVO1250 channel 2 output data (at bytes 4 and 5).
8	Click **OK** to close the **Items** window.
9	Save your changes (**File > Save**).

EtherNet/IP Implicit Messaging

Overview

For EtherNet/IP implicit message connections, use a requested packet interval (RPI) that is half of the MAST cycle time. If the RPI is less than 25 ms, the implicit message connections may be adversely affected when the diagnostic features of the BMENOC0302(H) module are accessed through explicit messages or the DTM.

In this situation, use these time-out multiplier settings:

RPI (ms)	Timeout Multiplier	Connection Timeout (ms)
5	32	160
10	16	160
20	8	160
25	4	100

NOTE: If you use values that are lower than those in this table, the network can consume unnecessary bandwidth. That can affect the performance of the module within the system. Refer to the time-out multiplier settings in the description of connection settings, page 209.

Add a Modbus TCP Device to the Network

Overview

This section extends the sample EcoStruxure Control Expert application. It includes these instructions:

- Add a Modbus TCP module to your EcoStruxure Control Expert application.
- Configure the Modbus TCP module.
- Configure a Modbus TCP connection that links the Ethernet communication module and the Modbus TCP module.

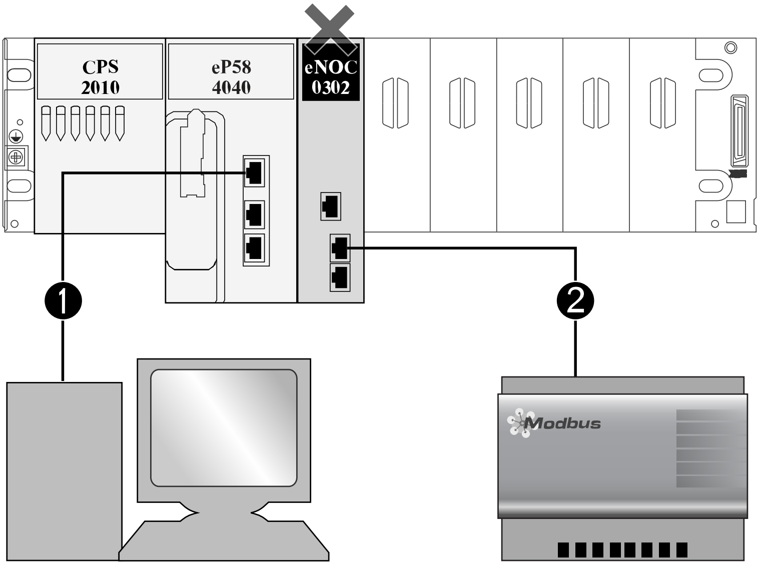
Connection to a Modbus TCP Device

Introduction

Use this example to establish communications between the M580 rack and a single-port Modbus TCP device.

Standalone Network Topology

This is an example of a generic Modbus TCP device in a simple configuration:



Legend:

1 An M580 controller module in the local rack connects to a computer that runs EcoStruxure Control Expert.

2 A BMENOC0302(H) Ethernet communications module in the local rack connects to a generic Modbus TCP device. (In this case, the rack port for the BMENOC0302(H) module is disabled, as indicated by the *X*.)

To re-create this example, use the IP addresses from your own configuration for these items:

- local computer
- BMENOC0302(H) Ethernet communications module
- generic Modbus TCP device

Add a Modbus Device to an EcoStruxure Control Expert Project

Overview

Use these instructions to add a Modbus device to your M580 EcoStruxure Control Expert project.

Add the Device

Add a Modbus device to your EcoStruxure Control Expert project:

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, right-click the name that you assigned to the BMENOC0302(H) module.
4	Scroll to **Add...** to see the **Add** dialog box.
5	From the **Device** column in the **Add** dialog box, select **Modbus Device**. **NOTE:** This selection (**Modbus Device**) is the generic Modbus DTM. If available, use the manufacturer-specified DTM that corresponds to your particular device.
6	Click **Add DTM** to open the **Properties of device** window for the Modbus device.
7	On the **General** tab, assign this **Alias name**: *MB1* **NOTE:** • EcoStruxure Control Expert uses the **Alias name** (*MB1*) as the based name for structure and variable names. No additional editing is required on this tab. Except for the **Alias name** field, parameters are read-only. • The Modbus DTM is added to the BMENOC0302(H) module in the **DTM Browser** as a subnode (<*IP_address*> Modbus:192.68.20.12).
8	Click the **OK** button and save you configuration (**File > Save**).

You can now configure the device you added to the project.

Access the Modbus Device Properties

Introduction

Use EcoStruxure Control Expert to edit the settings for a Modbus device.

NOTE:

- To edit these settings, disconnect the DTM from a device.
- These instructions assume that you selected **Modbus Device** from the **Add** window when you added a Modbus device to the project, page 222.

Access the Device Properties

For Modbus TCP devices, navigate to the configuration tabs:

Step	Action
1	In the **DTM Browser** (**Tools > DTM Browser**), double-click the DTM that corresponds to the Ethernet communication module that is associated with DTM of the generic Modbus device (**...MB1**).
2	To view the associated Modbus TCP and EtherNet/IP devices in the navigation pane, expand (**+**) the **Device List**, page 152.
3	Select the Modbus device in this example (**MB1: <MBD:192.168.20.12>**).

For more information about the different configuration tabs available for Modbus devices, refer to Device List Parameters, page 157.

Scan a Modbus TCP Device Across an External Router

Introduction

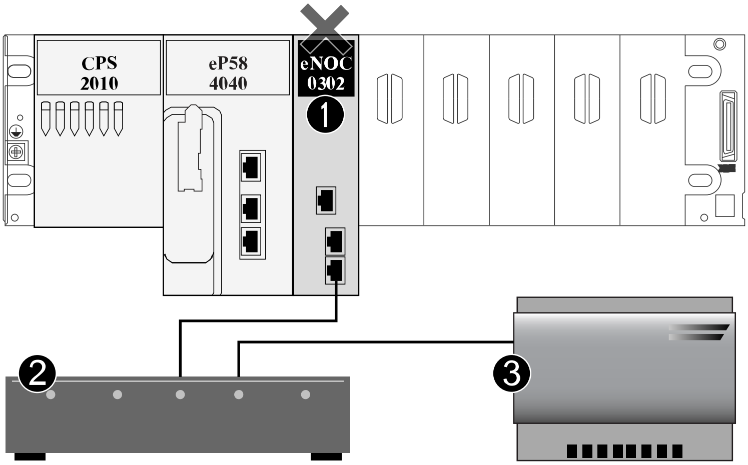
You can use the BMENOC0302(H) Ethernet communication module to configure the Modbus TCP scanner to scan a Modbus TCP slave device across an external router. (This feature does not apply to EtherNet/IP devices.)

Scanner Configuration

Verify that your network contains a BMENOC0302(H) module, an external router, and a Modbus TCP slave device. The Modbus TCP scanner traffic can cross the router under these conditions:

- The gateway IP address of the BMENOC0302(H) module is the same as the IP address of the router on the side of the module.
- The gateway IP address of the Modbus TCP slave device is the same as the IP address of the router on the side of the Modbus TCP slave device.

This is an illustrative example of a Modbus TCP scanner across a router:



Legend:

Item	Module/Device	Main IP Address	Subnet Mask	Gateway IP Address
1	BMENOC0302(H)	192.168.20.1	255.255.255.0	192.168.20.240
2	router	—	—	—
3	Modbus TCP slave	192.168.30.1	255.255.255.0	192.168.30.240

NOTE:

- In this case, the rack port for the BMENOC0302(H) module is disabled, as indicated by the *X*.
- Configure the BMENOC0302(H) module's IP parameters (main IP address, subnet mask, gateway IP address) on the IP address configuration screen. To open this screen, double-click the module in the **PLC bus** view in an M580 EcoStruxure project.
- Refer to the instructions to configure a Modbus TCP device under the BMENOC0302 (H) node, page 223.

Configuring the BMENOC0302(H) Module as an EtherNet/IP Adapter

Introduction

This section describes the configuration of the BMENOC0302(H) Ethernet communications module as an EtherNet/IP adapter using local slave functionality.

Introduction to Local Slaves

About Local Slaves

The BMENOC0302(H) Ethernet communications module scans network modules on behalf of the M580 controller module.

However, you can enable the communications module as an EtherNet/IP adapter (or local slave). When the local slave functionality is enabled, network scanners can access the M580 controller-module data that is mapped to local slave assembly objects in the controller-module program, page 296.

NOTE: The BMENOC0302(H) module continues to function as a scanner when it is enabled as an EtherNet/IP adapter.

The module supports up to 16 instances of local slaves (**Local Slave 1** ... **Local Slave 16**). Each enabled local slave instance supports these connections:

- one exclusive owner connection
- one listen-only connection

Process Overview

Configure a local slave:

Stage	Description
1	Enable and configure the BMENOC0302(H) module as a local slave, page 229.
2	Configure local slave instances in the scanner device, page 231. (Local slave instances correspond to each enabled local slave that is scanned.)
3	Specify the size of local slave input and output assemblies in the scanner device (originator). (Use sizes that match the input and output sizes of the enabled local slave.)

Implicit and Explicit Messaging

In its role as an EtherNet/IP adapter, the BMENOC0302(H) module responds to these requests from network scanners:

- *implicit messages*: Implicit messaging requests are sent from a network scanner device to the communications module. When the local slave functionality is enabled, network scanners can perform these tasks:
  - Read messages from the communications module.
  - Write messages to the communications module.

  Implicit messaging is especially suited to the exchange of peer-to-peer data at a repetitive rate.
- *explicit messages*: The communications module responds to explicit messaging requests that are directed to its CIP objects. When local slaves are enabled by the controller module, explicit messaging requests can access the communications module's CIP assembly instances. (This is a read-only function.)

Scanner Configuration

Configure the scanner:

Configuration	Description
EcoStruxure Control Expert	If the scanner device that communicates with the local slave can be configured using EcoStruxure Control Expert, use the DTMs that correspond to the BMENOC0302(H) modules to add those modules to your configuration.
third-party scanner	Third-party EtherNet/IP scanners that access the local slave assembly instances through the BMENOC0302(H) module do so in accordance with the assembly mapping table, page 235. That module is delivered with its corresponding EDS file. Third-party scanners can use the contents of the EDS file to map inputs and outputs to the appropriate assembly instances of the module.

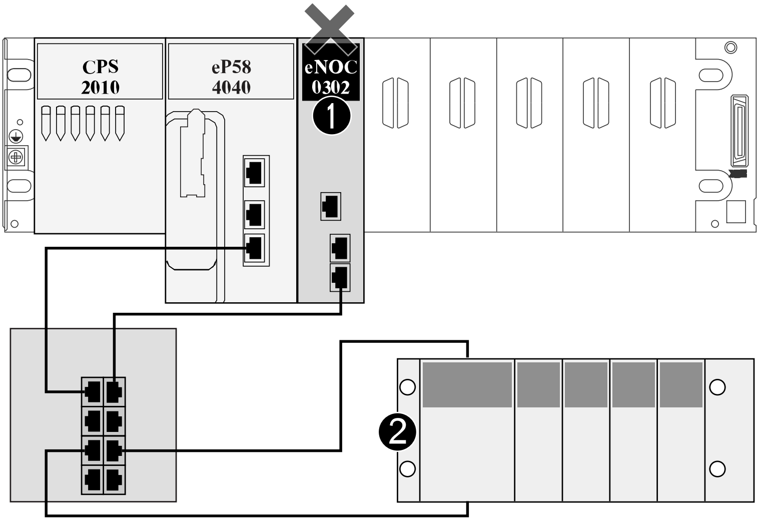
Local Slave Configuration Example

Introduction

Use these instructions to create a simple local slave configuration that includes a network scanner (originator, **O**) and a BMENOC0302(H) Ethernet communications module that is enabled as a local slave (target, **T**).

Originator and Target Devices

This simple network shows the enabled local slave and the master device:



Legend:

1 BMENOC0302(H): This Ethernet communications module is in slot 3 of the local M580 rack. (Note that its rack port is disabled, as indicated by the *X*.) In this example, you enable this module as a local slave device (or target, **T**)..

2 Modicon M340 rack: In this example, the scanner (or originator, **O**) on this rack scans the controller-module data on the M580 rack through the enabled local slave (BMENOC0302(H)).

Enable a Local Slave

Introduction

Enable **Local Slave 4** and **Local Slave 5** for the sample configuration.

Use the instructions below to enable **Local Slave 4** and **Local Slave 5** in the BMENOC0302 (H) module configuration.

Enable a Local Slave

Enable the BMENOC0302(H) module in the M580 local rack as a target device (local slave):

Step	Action
1	Open a Modicon M580 EcoStruxure Control Expert project.
2	Add a BMENOC0302(H) module to slot 3 in the local rack.
3	In the DTM Browser, right-click the module and select **Properties** from the menu.
4	On the **General** tab, assign this **Alias name** to the BMENOC0302(H) module: *BMENOC0302_slot3*
5	In the **DTM Browser** (**Tools > DTM Browser**), double-click the DTM that corresponds to the alias name of the BMENOC0302(H) module to open the configuration window.
6	In the navigation pane, expand (**+**) **EtherNet/IP Local Slaves** to see the available local slaves.
7	Select a local slave to see its properties. (For this example, select **Local Slave 4**.)
8	In the pull-down list (**Properties > Active Configuration**), scroll to **Enabled**.
9	Click **Apply** to enable **Local Slave 4**.
10	Click **OK** to apply the changes and close the configuration window.

You now have enabled **Local Slave 4** for a BMENOC0302(H) at IP address 192.168.20.10.

EtherNet/IP scanners that scan the network for the BMENOC0302(H) at that IP address can use implicit messages to read from and write to the assembly instances that are associated with the local slave instance, page 231.

Enable Another Local Slave

This example uses two local slave connections. Make a second connection for **Local Slave 5**:

Step	Action
1	Repeat the steps above to enable a second local slave (**Local Slave5**). **NOTE:** The appropriate IP address for this example (192.168.20.10) was assigned to the BMENOC0302(H) module in the assignment of **Local Slave4**.
2	Continue to the next procedure to configure the network scanner (originator, **O**).

Access Local Slaves with a Scanner

Introduction

Use these instructions to map the local slave instances in a network scanner to the enabled local slaves in the BMENOC0302(H) Ethernet communications module (**Local Slave 4**, **Local Slave 5**).

In this example, the BMENOC0302(H) module functions as a network scanner (originator, **O**) that scans a BMENOC0302(H) module that is enabled as a local slave (target, **T**).

Configure the BMENOC0302(H) module in an M580 EcoStruxure Control Expert project.

Add the Device DTM

Create a local slave instance that corresponds to an enabled local slave by name:

Step	Action
1	Open an M580 EcoStruxure Control Expert project that includes a BMENOC0302(H) Ethernet communication module.
2	Right-click the BMENOC0302(H) module in the **DTM Browser** (**Tools > DTM Browser**) and scroll to **Add** to open the **Add** dialog box.
3	Select the DTM that corresponds to the BMENOC0302(H) module (**BMENOC0302 (from EDS)**). **NOTE:** • The DTM used in this example corresponds to the BMENOC0302(H) module. For other target devices, use the DTM from the manufacturer that corresponds to your scanner device. • The corresponding input I/O vision and output I/O vision variables are automatically created with the respective suffixes **_IN** and **_OUT**.
4	Click the **Add DTM** button to open the **Properties of device** dialog window.
5	Assign a context-sensitive **Alias name** that corresponds to **Local Slave 4** for the M580 BMENOC0302(H) module. (For this example, enter *BMENOC0302_from_EDS_LS4*.)
6	Click **OK** to see the local slave instance in the **DTM Browser**.

Map Local Slave Numbers

In the M580 EcoStruxure Control Expert project, associate the local slave instances in the BMENOC0302(H) scanner with the specific local slaves that are enabled for the BMENOC0302(H) module:

Step	Action
1	In the **DTM Browser**, double-click the local slave instance that corresponds to **Local Slave 4** in the BMENOC0302(H) target device (*BMENOC0302_from_EDS_LS4*). **NOTE:** The default connection is **Local Slave 1 - Exclusive Owner**, which is most applicable to **Local Slave 1** in the target device. It is not appropriate for the local slave instance *BMENOC0302_from_EDS_LS4*, which is associated with **Local Slave 4** by the assigned context-sensitive name (... _LS4).
2	Select **Local Slave 1 - Exclusive Owner**.
3	Click **Remove Connection** to delete the connection to **Local Slave 1**.
4	Click **Add Connection** to open the dialog box (**Select connection to add**).
5	Scroll to **Local Slave 4 - Exclusive Owner**.
6	Click the **Apply** button.

The local slave (**Local Slave 4**) is now the target of a local slave instance with a context-sensitive connection name (**Local Slave 4 - Exclusive Owner**).

Map IP Addresses

Associate the IP address of the local slave (target, **T**) with the local slave instances in the scanner (originator, **O**) configuration:

Step	Action
1	Double-click the BMENOC0302(H) module in the **DTM Browser**.
2	In the navigation pane, expand the **Device List**, page 152.
3	Select a local slave instance (*BMENOC0302_from_EDS_LS4*).
4	Select the **Address Setting** tab.
5	In the **IP Address** field, enter the IP address of the local slave device (*192.168.20.10*).
6	Click in the navigation pane to make the **Apply** button active. **NOTE:** You may have to select **Disabled** in the pull-down menu (**DHCP for this device**) to activate the **OK** and **Apply** buttons.

Step	Action
7	Configure the data size. **NOTE:** Refer to the instructions for configuring input and output items, page 86.
8	Click **Apply**.

Configure an Additional Connection

You have created one local slave instance that corresponds by name and IP address to an enabled local slave. That is, the local slave instance *BMENOC0302_from_EDS_LS4* in the M340 EcoStruxure Control Expert project corresponds to **Local Slave 4** in the M580 EcoStruxure Control Expert project.

Because this example uses two local slave connections, make another connection (for **Local Slave 5**):

Step	Action
1	Repeat the above steps to create a second local slave instance that corresponds to **Local Slave 5**.
2	Build the EcoStruxure Control Expert project.

Access the Device DDT Variables

Step	Action
1	In the **Project Browser** (**Tools > Project Browser**) expand **Variables & FB instances**.
2	Double-click **Device DDT Variables** to see the device DDTs that correspond to the BMENOC0302 (H) module in slot 3.

Local Slave Parameters

Access the Configuration

Open the **EtherNet/IP Local Slaves** configuration page:

Step	Action
1	Open the EcoStruxure Control Expert project that includes a BMENOC0302(H) module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, double-click the name that you assigned to the BMENOC0302(H) to open the configuration window. **NOTE:** You can also right-click the module and scroll to **Open** to open the configuration window.
4	Expand (+) **Device List** in the navigation tree to see the local slave instances.
5	Select the local slave instance **BMENOC0302_from_EDS_LS4 <EIP:192.168.20.10>** to view the **Properties** and **Assembly** configuration tabs.

Properties

Identify and enable (or disable) the local slave on the **Properties** tab:

Parameter	Description	
Number	The EcoStruxure Control Expert DTM assigns a unique identifier (number) to the device. These are the default values: • *local slave 1*: 112 • *local slave 2*: 113 • *local slave 3*: 114 • ... • *local slave 16*: 127	
Active Configuration	Enabled	Enable the local slave with the configuration information in the **Assembly** fields when the BMENOC0302(H) module is an adapter for the local slave node.
	Disabled	Disable and deactivate the local slave. Retain the active local slave settings.
Comment	Enter an optional comment (maximum: 80 characters).	
Connection Bit	The auto-generated value in this field represents the association to the local slave in the **Request/Connection Summary** table, page 360. **NOTE:** This setting is auto-generated after the local slave settings are edited and the network configuration is saved.	

Assembly

Use the **Assembly** area of the **Local Slave** page to configure the size of the local slave inputs and outputs. Each device is associated with these assembly instances:

- Outputs
- Inputs
- Configuration
- Heartbeat (The heartbeat assembly instance is for listen-only connections only.)

The EcoStruxure Control Expert assembly numbers are fixed according to this table, where **O** indicates the originator (scanner) device and **T** indicates the target device:

Local Slave	Number		Connection
	Device	Assembly	
1	112	101	Outputs (T->O)
		102	Inputs (O->T)
		103	Configuration Size
		199	Heartbeat
2	113	111	Outputs (T->O)
		112	Inputs (O->T)
		113	Configuration Size
		200	Heartbeat
3	114	121	Outputs (T->O)
		122	Inputs (O->T)
		123	Configuration Size
		201	Heartbeat
4	115	131	Outputs (T->O)
		132	Inputs (O->T)
		133	Configuration Size
		202	Heartbeat
5	116	136	Outputs (T->O)
		137	Inputs (O->T)
		138	Configuration Size
		202	Heartbeat

Local Slave	Number		Connection
	Device	Assembly	
6	117	141	Outputs (T->O)
		142	Inputs (O->T)
		143	Configuration Size
		202	Heartbeat
7	118	146	Outputs (T->O)
		147	Inputs (O->T)
		148	Configuration Size
		202	Heartbeat
8	119	151	Outputs (T->O)
		152	Inputs (O->T)
		153	Configuration Size
		202	Heartbeat
9	120	156	Outputs (T->O)
		157	Inputs (O->T)
		158	Configuration Size
		202	Heartbeat
10	121	161	Outputs (T->O)
		162	Inputs (O->T)
		163	Configuration Size
		202	Heartbeat
11	122	166	Outputs (T->O)
		167	Inputs (O->T)
		168	Configuration Size
		202	Heartbeat
12	123	171	Outputs (T->O)
		172	Inputs (O->T)
		173	Configuration Size
		202	Heartbeat

Local Slave	Number		Connection
	Device	Assembly	
13	124	176	Outputs (T->O)
		177	Inputs (O->T)
		178	Configuration Size
		202	Heartbeat
14	125	181	Outputs (T->O)
		182	Inputs (O->T)
		183	Configuration Size
		202	Heartbeat
15	126	186	Outputs (T->O)
		187	Inputs (O->T)
		188	Configuration Size
		202	Heartbeat
16	127	191	Outputs (T->O)
		192	Inputs (O->T)
		193	Configuration Size
		202	Heartbeat

NOTE: You can read the BMENOC0302(H) module’s assembly instance through explicit messaging only when you allocate sufficient room for the response. The response size equals the sum of *assembly size + 1 byte* (reply service) + *1 byte* (general status).

Limitations (from the perspective of the local slave):

- *maximum RPI value*: 65535 ms requested packet interval (RPI)
- *maximum timeout value*: 512 * RPI
- *outputs (T->O)*: 509 bytes maximum
- *inputs (O->T)*: 505 bytes maximum
- *configuration for the Ethernet communication module*: 0 (fixed)

Working with Device DDTs

Introduction

Use EcoStruxure Control Expert to create a collection of device derived data types (Device DDTs) and variables that support communications and the transfer of data between the controller module and the various local slaves, distributed devices, and corresponding I/O modules.

You can create Device DDTs and corresponding variables in the EcoStruxure Control Expert DTM. Those program objects support your network design.

Use the Device DDTs for these tasks:

- Read status information from the Ethernet communication module.
- Write control instructions to the Ethernet communication module.

Double-click the name of the Device DDT in the **Project Browser** to view its properties and open the corresponding EDS file.

> **NOTE:** For applications that require multiple Device DDTs, create an **Alias name** that logically identifies the Device DDT with the configuration (module, slot, local slave number, etc.).

Device DDT Variables

You can access the Device DDTs and the corresponding variables in EcoStruxure Control Expert and add them to a user-defined **Animation Table**. Use that table to monitor read-only variables and edit read-write variables.

Use these data types and variables to perform these tasks:

- Read the status of connections and communications between the Ethernet communication module and distributed EtherNet/IP and Modbus TCP devices:
  - The status is displayed as a 32-byte HEALTH_BITS array.
  - A bit value of 0 indicates a lost connection or a communication module that stopped communicating with the distributed device.
- Toggle a connection ON (1) or OFF (0) by writing to a selected bit in a 16-word DIO_CONTROL array
- Monitor the value of local slave and distributed device input and output items that you created in EcoStruxure Control Expert.

Display Order for Input and Output Items

In the **Project Browser**, view the Device DDTs, page 240.

The **Data Editor** displays each input and output variable. When you open the first input and output variables, you can see both the connection health bits (DEVICE_OBJ_HEALTH) and the connection control (DEVICE_OBJ_CTRL) bits.

This table shows the rule assignment for connection numbers:

Inputs	Order	Outputs
health bits (note 1)	1	control bits (note 1)
Modbus TCP input variables (note 2)	2	Modbus TCP output variables (note 2)
local slave input variables (note 3)	3	local slave output variables (note 3)
EtherNet/IP input variables (note 2)	4	EtherNet/IP output variables (note 2)

NOTE 1: Health and control bits are in this format:

- **i.** By device type:
  - a. Modbus TCP
  - b. local slave
  - c. EtherNet/IP
- **ii.** Within each device type:
  - a. by device or local slave number
  - b. within a device (by connection number)

NOTE 2: Device variables are in this format:

- **i.** by device number
- **ii.** within a device (by connection number)
- **iii.** within a connection (by item offset)

NOTE 3: Local slave variables are in this format:

- **i.** by local slave number
- **ii.** within each local slave (by item offset)

Accessing Device DDT Variables

Device DDTs and Scanned Devices

Introduction

You can access the device DDT for EtherNet/IP and Modbus TCP devices that are scanned by the Ethernet communication module after you perform one of these tasks:

- Add an EtherNet/IP device to the network, page 202.
- Add a Modbus TCP device to the network, page 219.
- Configure the Ethernet communication module as an EtherNet/IP adapter, page 226.

Access the Device DDT Variables

Access the device DDT for the Ethernet communication module in EcoStruxure Control Expert:

Step	Action
1	Open the EcoStruxure Control Expert **Project Browser** (**Tools > Project Browser**).
2	Expand (**+**) **Variables & FB instances**.
3	Double-click **Device DDT Variables**.

To read the status and set the device control bit, add the variable to an Animation Table, page 181.

> **NOTE:** The red arrow and lock icons in the **Device DDT** table indicate that the variable name was auto-generated by EcoStruxure Control Expert based on the configuration of the communication module, local slave, or distributed device. (You cannot edit the variable name.)

This table describes the input and output bits associated with the EtherNet/IP and Modbus TCP devices:

Name	Description
Freshness	This is a global bit: • **1:** Input objects below (**Freshness_1**, **Freshness_2**, etc.) for the associated device are true (**1**) and provide up-to-date data. • **0:** One or more inputs (below) is not connected and does not provide up-to-date data.
Freshness_1	These bits represent individual input objects for the device: • **1:** The input object in this row is connected and provides up-to-date data. • **0:** The input object is not connected and does not provide up-to-date data.
Freshness_2	These bits represent individual input objects for the device: • **1:** The input object in this row is true (**1**) and provides up-to-date data. • **0:** The input object is not connected (**0**) and does not provide up-to-date data.
Freshness_3	
...	
(available)	The rows after the **Freshness** data are organized in groups of **Inputs** and **Outputs** that have user-defined names. The number of input and output rows depends on the number of input and output requests configured for a particular device.

Hardware Catalog

DTM Files

The EcoStruxure Control Expert **Hardware Catalog** contains a list of modules and devices that you can add to an EcoStruxure Control Expert project. EtherNet/IP and Modbus TCP devices are located in the **DTM Catalog** tab at the bottom of the **Hardware Catalog**. Each module or device in the catalog is represented by a DTM that defines its parameters.

EDS Files

Not every device in today’s market offers device-specific DTMs. Some devices are defined by device-specific EDS files. EcoStruxure Control Expert displays EDS files in the form of a DTM. In this way, you can use EcoStruxure Control Expert to configure devices that are defined by an EDS file in the same way you configure a device defined by its DTM.

Other devices lack both a DTM and an EDS file. Configure those devices by using the generic DTM on the **DTM Catalog** page.

View the Hardware Catalog

Open the EcoStruxure Control Expert **Hardware Catalog**:

Step	Action
1	Open EcoStruxure Control Expert.
2	Find the **PLC bus** in the **Project Browser**.
3	Use one method to open the catalog: • Use the pull-down menu (**Tools > Hardware Catalog**). • Double-click an empty slot in the **PLC bus**.

Add a DTM to the EcoStruxure Control Expert Hardware Catalog

A Manufacturer-Defined Process

Before the EcoStruxure Control Expert **Hardware Catalog** can use a DTM, install the DTM on the *host* PC (the computer that runs EcoStruxure Control Expert).

The installation process for the DTM is defined by the device manufacturer. Consult the documentation from the device manufacturer to install a device DTM on your computer.

NOTE: After a device DTM is successfully installed on your computer, update the EcoStruxure Control Expert **Hardware Catalog** to see the new DTM in the catalog. The DTM can then be added to an EcoStruxure Control Expert project.

Add an EDS File to the Hardware Catalog

Introduction

You may want to use an EtherNet/IP device for which no DTM is in the catalog. In that case, use these instructions to import the EDS files to the catalog to create a corresponding DTM.

EcoStruxure Control Expert includes a wizard you can use to add one or more EDS files to the EcoStruxure Control Expert **Hardware Catalog**. The wizard presents instruction screens to execute these commands:

- Simplify the addition of EDS files to the **Hardware Catalog**.
- Perform a redundancy check for duplicate EDS files in the **Hardware Catalog**.

NOTE: The EcoStruxure Control Expert **Hardware Catalog** displays a partial collection of DTMs and EDS files that are registered with the ODVA. This library includes DTMs and EDS files for products that are not manufactured or sold by Schneider Electric. The non-Schneider Electric EDS files are identified by vendor in the catalog. Contact the identified device's manufacturer for inquiries regarding the corresponding non-Schneider Electric EDS files.

Add EDS Files

Open the **EDS Addition** dialog box:

Step	Action
1	Open an EcoStruxure Control Expert project that includes an Ethernet communication module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	In the **DTM Browser**, select a communication module.
4	Right-click the communication module and scroll to **Device menu > Additional functions > Add EDS to library**.
5	In the **EDS Addition** window, click **Next** to open the **EDS Addition** dialog box.

Add one or more EDS files to the library:

Step	Action
1	Use these commands in the **Select the Location of the EDS File(s)** area of the **EDS Addition** dialog box to identify the location of the EDS files: • **Add File(s)**: Add one or more EDS files that are individually selected. • **Add all the EDS from the Directory**: Add the files from a selected folder. **NOTE:** Select (check) the **Look in Subfolders** box to add the EDS files from the folders within the selected folder.
2	Click **Browse** to open a navigation dialog box.
3	Select the location of the EDS file(s): • Navigate to at least one EDS file. • Navigate to a folder that contains EDS files. **NOTE:** Keep the location selected (highlighted).
4	Click **Select** to close the navigation window and see your selection in the **Directory or File Name** field.
5	Choose the naming convention rule for the EDS DTM name creation: • The naming convention is based on Model Name / Product Name and Revision. A random character is automatically suffixed when Model Name / Product Name and Revision of an EDS file in the library is identical. The new naming convention is irrespective of the order in which EDS files are added to device library. • By default, the **New Naming Convention** box is selected (checked) and the new naming rule applies. • To keep backward compatibility with Unity Pro/EcoStruxure Control Expert versions, deselect (uncheck) the **New Naming Convention** box and the naming rule is based on Model Name / Product Name. **NOTE:** Unity Pro is the former name of EcoStruxure Control Expert for version 13.1 or earlier.
6	Click **Next** to compare the selected EDS files to the files in the library.
7	The next page of the **EDS Addition** wizard opens. It indicates the status of each device you attempted to add: • checkmark ✓ (green): The EDS file can be added. • informational icon ℹ (blue): There is a redundant file. • exclamation point ! (red): There is an invalid EDS file. **NOTE:** Click **View Selected File** to open and view the selected file.
8	Click **Next** to add the non-duplicate files. **NOTE:** The next page of the **EDS Addition** wizard opens to indicate that the action is complete.
9	Click **Finish** to close the wizard to automatically update the hardware catalog.

Remove an EDS File from the Hardware Catalog

Introduction

You can remove a module or device from the list of available devices in the EcoStruxure Control Expert **Hardware Catalog** by removing its **EDS** file from the library.

When you remove an EDS file from the library, the device or module disappears from the **DTM Catalog**. However, removing the file from the library does not delete the file from its stored location, so you can import the file again later.

Remove the File from the Catalog

Remove an EDS file from the EcoStruxure Control Expert catalog:

Step	Action
1	Open the EcoStruxure Control Expert **DTM Browser** (**Tools > DTM Browser**).
2	In the **DTM Browser**, select an Ethernet communication module.
3	Right-click the module and scroll to **Device menu > Additional functions > Remove EDS from library** to open the **EDS Deletion from Device Library** window:
4	Use the selection lists in the heading of this window to specify how EDS files are displayed: • **Display:** Choose criteria to filter the list of EDS files: ◦ **All EDS** (no filtering) ◦ **Only Devices** ◦ **Only Chassis** ◦ **Only Modules** • **Sort by:** Choose criteria to sort the list of displayed EDS files: ◦ **File Name** ◦ **Manufacturer** ◦ **Category** ◦ **Device Name** • **Displayed Name:** Choose the identifier for each device: ◦ **Catalog Name** ◦ **Product Name**
5	Expand (**+**) the **Device Library** navigation tree and select the EDS file you want to remove. **NOTE:** Click **View Selected File** to see the read-only contents of the selected EDS file.
6	Click the **Delete Selected File(s)** button to open the **DeleteEDS** dialog box.
7	Click **Yes** to remove the selected EDS file from the list.

Step	Action
8	Repeat these steps for each EDS file you want to delete.
9	Click **Finish** to close the wizard and automatically update the hardware catalog.

Export and Import the EDS Library

Introduction

To use the same project on two EcoStruxure Control Expert installations (for example a source, and a target host computer), you may have to update the DTM **Hardware Catalog** of the target host computer.

Instead of adding one by one the missing EDS files in the host computer, update the DTM **Hardware Catalog** with these steps:

- Export the EDS library from the source host computer.
- Import the EDS library to the target host computer.

NOTE: When you export the EDS library, the software generates a .dlb file that contains the DTMs created from the EDS files.

Export the EDS Library

Open the **Export EDS Library** dialog box:

Step	Action
1	Open an EcoStruxure Control Expert project that includes an Ethernet communication module.
2	Open the **DTM Browser** (**Tools** > **DTM Browser**).
3	Open the **Export EDS library** window: *a.* In the **DTM Browser**, right-click the name of the communication module. *b.* Scroll: **Device menu** > **Additional functions** > **Export EDS library**

Use the features in the **Export EDS library** window to configure the new archived EDS library:

Feature	Description
EDS Device Library Path	This field shows the location of the EDS device library.
Enter / Select EDS Library File Name	Enter the full folder path along with the file name in this field.
Browse	Click this button to use standard Windows commands to name and save the EDS library.

Feature	Description
Export	Click this button to create the archived EDS library (in the .dlb format) and save it to the selected location. **NOTE:** A pop-up window tells you that the export is successful.
Close	Click this button to close the **Export EDS library**.

Import the EDS Library

Import an archived EDS library:

Step	Action
1	Open an EcoStruxure Control Expert project that includes an Ethernet communication module.
2	Open the **DTM Browser** (**Tools > DTM Browser**).
3	Open the **Import EDS library** window: *a.* In the **DTM Browser**, right-click the name of the communication module. *b.* Scroll: **Device menu > Additional functions > Import EDS library**

Use the features in the **Import EDS library** window to import an archived EDS library:

Feature	Description
Enter / Select EDS Library File Name	Enter the full folder path along with the file name in this field.
Browse	Click this button to use standard Windows commands to select the EDS library.
Import	Click this button to import the archived EDS library. **NOTE:** A pop-up window tells you that the import is successful.
Close	Click this button to close the **Import EDS library**.

Managing Connection Bits

Connection Health Bits and Connection Control Bits

Introduction

Use these instructions to configure these bits:

- *connection health bits*: Display the status of each device with one or more connections.
- *connection control bits*: Toggle each connection on and off using object IDs.

Identify the Connection Health Bits

For the Ethernet communications module, discover the health bit that is mapped to a specific distributed device.

The module supports a maximum of 128 connections for distributed devices. The health of each device is represented in a single bit value:

- 1: Every connection that is configured for the device is active.
- 0: One or more connections that are configured for the device are not active.

In the EcoStruxure Control Expert **Project Browser**, double-click **Variables & FB** instances to view health bits in an eight-word array.

EtherNet/IP Connection Health Bits

For EtherNet/IP devices, navigate to a connection node:

Step	Action
1	In the DTM Browser (**Tools > DTM Browser**), double-click the DTM that corresponds to the appropriate Ethernet communications module.
2	In the navigation pane, expand the **Device List**.
3	Select the connection that corresponds to a node in the **Device List**.
4	Select the **Connection Settings** tab, page 209.
5	Locate the device number within the brackets to the left of the Device DTM name (example: *[002]*). **NOTE:** For example, a **Device Number** value of 2 maps to the third bit in both the BME_NOC0302.DIO_CTRL array and the BME_NOC0302.DIO_HEALTH array.

NOTE: To diagnose the health of the device, refer to the device DDTs for the Ethernet communication module, page 190.

Modbus TCP Connection Health Bits

For Modbus TCP devices, navigate to the main device node:

Step	Action
1	In the DTM Browser (**Tools > DTM Browser**), double-click the DTM that corresponds to the communications module. **NOTE:** These instructions assume that you selected **Modbus Device** when you created a local slave instance in the **Add** window, page 105.
2	In the navigation pane, expand the **Device List**.
3	Select a Modbus TCP device.
4	Select the **Request Setting** tab.
5	Locate the device number within the brackets to the left of the Device DTM name (example: *[000]*). **NOTE:** For example, a **Device Number** value of 0 maps to the first bit in both the BME_NOC0302.DIO_CTRL array and the BME_NOC0302.DIO_HEALTH array.

Access the Modbus connection settings:

Step	Action
1	In the **DTM Browser**, select a communications module for which a Modbus device is configured.
2	Double-click the communications module to open the configuration window.
3	In the navigation pane, expand the **Device List**.
4	Select the Modbus device.
5	Select the **Request Setting** tab.
6	Configure requests: • *Add a request*: Click **Add Request** to see the request data in the next available row. • *Remove a request*: Click the row that corresponds to the request and click **Remove**. • When you add or remove a request, the corresponding request is added or removed in the navigation pane (**Request 001: Items**; **Request 002: Items**; **Request 003: Items**; etc.). You can select a request to configure its input data.
7	Click **Apply**. **NOTE:** You can add or remove multiple requests before you click **Apply**.

Monitor the Connection Health Bits in an Animation Table

Use an animation table to monitor the status of connection health bits and other variables. Add health bits to an animation table:

Step	Action
1	In the **Project Browser**, right-click **Animation Tables** and scroll to **New Animation Table**.
2	In the **New Animation Table**, type these values for these fields: • **Name**: Connection_Health_Bits • **Number of animated characters**: Accept the default (100).
3	Click **OK** to open the **Connection_Health_Bits** animation table.
4	Double-click the first empty row in the **Name** column.
5	Click the ellipsis (**...**) button to open the **Instance Selection** dialog box.
6	Find the health bits and select the entire array.
7	Click **OK** to add the array to the **Connection_Health_Bits** animation table. **NOTE:** Remember that each row represents a word that contains 16 individual connection health bits. When the DTM for the Ethernet communication module is connected to the physical module, the **Value** field displays a value for the entire word.

Diagnostics

Overview

The following information describes the diagnostics for Modicon M580 modules.

NOTE: For details on diagnostics at the system level, refer to the systems diagnostics topic in the *Modicon M580 Frequently Used Architectures System Guide*.

LED Indicators

Visual Indicators on the BMENOC0302(H) Module

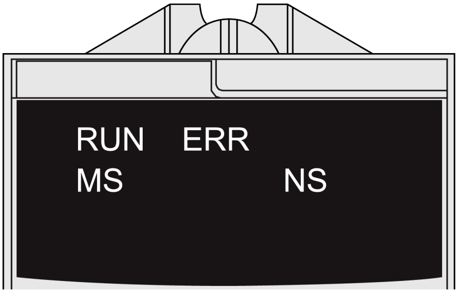
Introduction

There are two sets of LED indicators on the front of the BMENOC0302(H) module:

- LEDs in the module display screen report the performance of the module and its communications with the network appear.
- Small LEDs on the RJ45 connectors report the activity and connectivity status of the associated Ethernet ports.

Module Performance LED Indications

This is the LED display on the front of the BMENOC0302(H) module:



Duplicate IP addresses can cause unpredictable module/network behavior.

⚠WARNING

UNINTENTED EQUIPMENT BEHAVIOR

Verify that each module has a unique IP address.

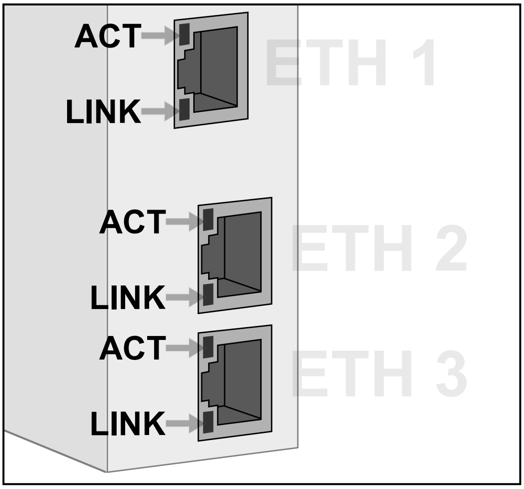
Failure to follow these instructions can result in death, serious injury, or equipment damage.

These LEDs provide information about the status and performance of the BMENOC0302(H) module:

LED	Color	State	Description
RUN	green	on	The configured module operates normally
		off	There is no power to the module, or the module is not configured.
		flashing	The communications module is in self-test mode or the controller module is stopped.
ERR	red	off	The running module *does not* detect an error.
		flashing	The module is not configured, the configuration is in progress, or an error is detected for rack communications.
		on	An error is detected.
MS (module status)	–	off	There is no power to the module.
	green	on	The module is operating correctly.
		flashing	The module is not configured.
	red	on	A major non-recoverable error (a firmware error, for example) is detected.
		flashing	A recoverable error is detected.
NS (network status)	–	off	There is no power to the module, or there is no link on the external ports including the Ethernet rack port.
	green	on	At least one CIP connection for which the module is the originator is established.
		flashing	The module has an IP address, but there is no detected CIP connection.
	red	on	The module has a duplicate IP address, or the module is receiving an operating system update.
		flashing	At least one exclusive owner CIP connection (for which the module is the target) is timed out. **NOTE:** The LED flashes until the connection is reestablished or the module is reset.

Ethernet Port LED Indications

Each RJ45 connector on the front of the BMENOC0302(H) module has **ACT** and **LINK** LEDs that report the activity and connectivity of the Ethernet port:



LED indications:

LED	Color	State	Description
LINK (link/speed)	green	on	The link runs at its maximum speed: • **ETH 1:** 100 Mbps (Fast Ethernet) • **ETH 2, ETH 3:** 1000 Mbps (Gigabit Ethernet)
	yellow	on	The link is detected: • **ETH 1:** The 10 Mbps link is detected. • **ETH 2, ETH 3:** The 10/100 Mbps link is detected.
	—	off	No link to the port is detected.
ACT (activity)	green	flashing	There is transmit or receive activity on the port.
		on	The link is detected, but there is no activity on the port.
		off	There is no link to the port.

Device DDT for the Ethernet Communications Module

T_BMENOC0302 Device DDT

Access Input/Output Parameters

In the EcoStruxure EcoStruxure Control Expert **Project Browser**, double-click **Variables & FB instances** to access the Device DDT variables for the BMENOC0302(H) Ethernet communications module.

Input Parameters

This following tables describe the input parameters in the device DDT for the module.

ETH_STATUS: This table describes the bits associated with the ETH_STATUS (Ethernet status) word:

Parameter	**Type**	**Bit**	**Description**
PORT1_LINK	BOOL	0	0: Ethernet port 1 (ETH 1) link is down. 1: Ethernet port 1 (ETH 1) link is up.
PORT2_LINK	BOOL	1	0: Ethernet port 2 (ETH 2) link is down. 1: Ethernet port 2 (ETH 2) link is up.
PORT3_LINK	BOOL	2	0: Ethernet port 3 (ETH 3) link is down. 1: Ethernet port 3 (ETH 3) link is up.
ETH_BKP_PORT_LINK	BOOL	3	0: rack port link is down. 1: rack port link is up.
SCANNER_OK	BOOL	6	0: I/O scanner operations are not normal.
			1: At least one configured device is scanned.
GLOBAL_STATUS	BOOL	7	0: At least one service is not operating normally.
			1: Services are operating normally.
NETWORK_HEALTH	BOOL	8	0: A potential network broadcast storm is detected. **NOTE:** Verify that the wiring and configuration of the controller module and BMENOC0302(H) module are valid.
			1: A network broadcast storm is not detected.

SERVICE_STATUS: This table describes the bits associated with the SERVICE_STATUS (word):

Parameter	Type	Bit	Description
RSTP_SERVICE	BOOL	0	0: The RSTP service is not operating normally.
			1: The RSTP service is operating normally or is disabled.
NTPv4_SERVICE	BOOL	1	0: The NTPv4 service is not operating normally.
			1: The NTPv4 service is operating normally or is disabled.
PORT502_SERVICE	BOOL	2	0: Port 502 is not operating normally.
			1: Port 502 is operating normally or is disabled.
SNMP_SERVICE	BOOL	3	0: SNMP is not operating normally.
			1: SNMP service is operating normally or is disabled.
MAIN_IP_ADDRESS_STATUS	BOOL	4	0: The main IP address is duplicated or not assigned.
			1: The main IP address is unique and valid.
*EIP_SCANN*ER	BOOL	7	0: The EtherNet/IP scanner service is not operating normally.
			1: The EtherNet/IP scanner service is operating normally or is disabled.
MODBUS_SCANNER	BOOL	8	0: The Modbus scanner service is not operating normally.
			1: The Modbus scanner service is operating normally or is disabled.
SNTP_CLIENT	BOOL	10	0: The SNTP client service is not operating normally.
			1: The SNTP client service is operating normally or is disabled.
WEB_SERVER	BOOL	11	0: The web server service is not operating normally.
			1: The web server service is operating normally or is disabled.
FIRMWARE_UPGRADE	BOOL	12	0: The firmware upgrade service is not operating normally.
			1: The firmware upgrade service is operating normally or is disabled.
FTP	BOOL	13	0: The FTP server service is not operating normally.
			1: The FTP server service is operating normally or is disabled.

Parameter	Type	Bit	Description
FDR_SERVER	BOOL	14	0: The FDR server service is not operating normally.
			1: The FDR server service is operating normally or is disabled.
EIP_ADAPTER	BOOL	15	0: The EtherNet/IP adapter service is not operating normally.
			1: The EtherNet/IP adapter service is operating normally or is disabled.

SERVICE_STATUS2: This table describes the parameters associated with the SERVICE_STATUS2 (word):

Parameter	Type	Bit	Description
A_B_IP_ADDRESS_STATUS	BOOL	0	0: Duplicate IP address for controller module A/B.
			1: Unique IP addresses for controller modules A/B.
LLDP_SERVICE	BOOL	1	0: The LLDP service is not operating normally.
			1: The LLDP service is operating normally or is disabled.
EVENT_LOG_STATUS	BOOL	2	0 = Event log service is not operating normally.
			1 = Event log service is operating normally or is disabled.
LOG_SERVER_NOT_REACHABLE	BOOL	3	1 = No acknowledgment received from the syslog server.
			0 = Acknowledgment received from the syslog server
NTP_SYNC	BOOL	4	1 = Synchronized to the NTP server
			0 = Not synchronized to the NTP server
NTP_QUALITY_WARNING	BOOL	5	1 = Quality of the clock out of range defined in the configuration
			0 = All other instances

Other Input Parameters: The scanner device DDT contains these other parameters:

Parameter	Type	Description
ETHERNET_PORT_1_2_STATUS (BYTE)	Bits 1...0	0: ETH 1 disabled
		1: ETH 1 access port
		2: ETH 1 port mirroring
		3: ETH 1 device network port
	Bits 3...2	reserved (0)
	Bits 5...4	0: ETH 2 disabled
		1: ETH 2 access port
		2: ETH 2 port mirroring
		3: ETH 2 device network port
	Bits 7...6	0: ETH 2 alternate RSTP port
		1: ETH 2 backup RSTP port
		2: ETH 2 designated RSTP port
		3: ETH 2 root RSTP port
ETHERNET_PORT3_BKP_STATUS (BYTE)	Bits 1...0	0: ETH 3 disabled
		1: ETH 3 access port
		2: ETH 3 port mirroring
		3: ETH 3 device network port
	Bits 3...2	0: ETH 3 alternate RSTP port
		1: ETH 3 backup RSTP port
		2: ETH 3 designated RSTP port
		3: ETH 3 root RSTP port
	Bits 5...4	0: The Ethernet rack port is disabled, page 93.
		3: The Ethernet rack port is enabled, page 93 (to support Ethernet communications).
	Bits 7...6	reserved (0)
FIRMWARE_VERSION	WORD	MSB = major revision; LSB = minor revision
FDR_USAGE	BYTE	% of FDR sever usage
IN_PACKETS	UINT	number of packets received by the module
IN_ERRORS	UINT	number of inbound packets that contain detected errors

Parameter	Type	Description
OUT_PACKETS	UINT	number of packets sent from the module
OUT_ERRORS	UINT	number of packets from the module that contain detected errors
CONF_SIG	UINT	signatures of the PRM files on the local module FDR server
NTP_WITHIN	UINT	Estimated accuracy of the clock in milliseconds
NTP_NB_SERVER_CONNECTED	UINT	Number of connected servers
IPSEC_CHANNEL_TYPE	ARRAY [0...7] OF BOOL	One bit per channel: • 0: initiator • 1: responder
IPSEC_CHANNEL_STATUS	ARRAY [0...7] OF BOOL	One bit per channel: • 0: OK • 1: NOK
IPSEC_CHANNEL_PPS	ARRAY [0...7] OF UINT	packets per second per channel

Device Health Bits

This table describes the health bits of the devices that are scanned by the module:

Parameter	Type	Bit	Description
LS_HEALTH	BOOL	0: Local slaves and distributed equipment are not operating normally. 1: Local slaves and distributed equipment are operating normally or are disabled.	local slave health bits (local slave 1 to 16) ARRAY [1...16] of BOOL
DIO_HEALTH	BOOL		distributed equipment health bits (1 bit per distributed device up to 128 devices) ARRAY [0...127] of BOOL

Output Parameters

This table describe the output parameters in the device DDT for the module:

Parameter	Type	Bit	Description
DIO_CTRL	BOOL	0: Enable normal communications to the DIO device. 1: Disable communications to the device. In this case, outputs are not written and inputs are not updated.	distributed equipment control bits (1 bit per distributed device up to 128 devices) ARRAY [0...127] of BOOL

NOTE: The array index for the DIO device is mapped to the device number in the request/connection summary in the communication module's **Device List**, page 156.

Diagnostics through the EcoStruxure Control Expert DTM Browser

Introducing Diagnostics in the EcoStruxure Control Expert DTM

Introduction

The EcoStruxure DTM provides diagnostics information that is collected at configured polling intervals. Use this information to diagnose the operation of your Ethernet communications module.

Connect the DTM

Before you can open the diagnostics page, make the connection between the DTM for the target communication module:

Step	Action
1	Open an EcoStruxure project that includes the Ethernet communications module.
2	Open the EcoStruxure **DTM Browser** (**Tools > DTM Browser**).
3	Find the name that is assigned to your Ethernet communications module in the **DTM Browser**.
4	Right-click the module name and scroll to **Connect**.

Open the Diagnosis Page

Access the **Diagnosis** information:

Step	Action
1	Right-click the name that is assigned to your Ethernet communications module in the **DTM Browser**.
2	Scroll to **Device Menu > Diagnosis** to view the available diagnostics pages.

Diagnostics Information

The diagnostics window has two distinct areas:

- left pane: LED icons indicate the operating status of modules, devices, and connections.
- right pane: These pages show diagnostics data for these items:
  - Ethernet communications module
  - local slave nodes that are activated for the communication module
  - EtherNet/IP connections between the communication module and a remote EtherNet/IP device

When the appropriate DTM is connected to the physical communication module, EcoStruxure sends an explicit message request once per second to detect the state of the communication module, the remote devices, and the EtherNet/IP connections linked to that module.

EcoStruxure places one of these status icons over the module, device, or connection in the left pane of the **Diagnostic** window to indicate its status:

Icon	Communication module	Connection to a remote device
🟢	Run state is indicated.	The health bit for every EtherNet/IP connection and Modbus TCP request (to a remote device, sub-device, or module) is set to active (1).
🔴	One of these states is indicated: • undetermined • stopped • not connected	The health bit for at least one EtherNet/IP connection or Modbus TCP request (to a remote device, sub-device, or module) is set to inactive (0).

Communication Module Ethernet Diagnostics

Introduction

Use the **Ethernet Diagnostic** page to view the dynamic and static data for the Ethernet ports on the Ethernet communications module.

NOTE: Before you can open the diagnostics page, create a connection between the DTM for the target communication module and the physical module, page 283.

Open the Page

Access the diagnostic information:

Step	Action
1	Open the diagnosis information for your module, page 264.
2	In the left pane of the **Diagnosis** window, select the communication module node.
3	Select the appropriate tab: • **Ethernet Diagnostic**, page 267 • **Bandwidth**, page 270 • **RSTP Diagnostic**, page 272

NOTE: The number of ports on the communication module determines the number of columns displayed in this page.

Data Display

For any of these diagnostic tabs, select the **Refresh Every 500ms** checkbox to view static or dynamic data:

Checkbox	Description
selected	• Display data that is dynamically updated every 500 ms. • Increment the number at the top of the table each time data is refreshed.
deselected	• Display static data. • Do not increment the number at the top of the table. That number now represents a constant value.

Ethernet Diagnostic Parameters

Introduction

The **Ethernet Diagnostic** page displays parameters for each communication module port (ETH 1, ETH 2, ETH 3, rack Port).

General Parameters

Parameter	Description
Interface speed	Valid values include: 0 (no link), 10, 100, 1000 (Mbits/s)
Interface flags	Bit 0: Link Status (0 = **Inactive link**; 1 = **Active link**)
	Bit 1: Duplex Mode (see below)
	Bits 2...4: Negotiation Status (see below)
	Bit 5: Manual Setting Requires Reset (see below)
	Bit 6: Local Hardware Fault (see below)
Duplex mode	0 = half duplex 1 = full duplex
Negotiation Status	3 = successfully negotiated speed and duplex 4 = forced speed and link
Manual Setting Require Reset	0 (automatic, **Inactive link**): The interface can activate changes to link parameters automatically. 1 (**Active link**): Devices require a reset service to be issued to its Identity.
Local Hardware Fault	0 = no event 1 = event detected
Physical Address	Module MAC Address

Input Parameters

Parameter	**Description**
Octets	Octets received on the interface
Unicast Packets	Unicast packets received on the interface
Non-Unicast Packets	Non-unicast packets received on the interface
Discards	Inbound packets received on the interface, but discarded
Errors	Inbound packets that contain detected errors (does not include In Discards)

Output Parameters

Parameter	**Description**
Octets	Octets received on the interface
Unicast Packets	Unicast packets received on the interface
Non-Unicast Packets	Non-unicast packets received on the interface
Discards	Inbound packets received on the interface, but discarded
Errors	Outbound packets that contain detected errors (does not include In Discards)
Unknown Protocols	Outbound packets of an unidentified protocol

Error Counter Parameters

Parameter	Description
Alignment Errors	Frames that are not an integral number of octets in length
FCS Errors	Frames received that do not pass the frame check sequence (FCS)
Single Collisions	Successfully transmitted frames that experienced exactly one collision
Multiple Collisions	Successfully transmitted frames that experienced more than one collision
SQE Test Errors	Number of times the SQE test error is detected and generated
Deferred Transmissions	Frames for which first transmission attempt is delayed because the medium is busy
Late Collisions	Number of times a collision is detected later than 512 bit times into the transmission of a packet
Excessive Collisions	Frames for which transmission does not finish due to excessive collisions
MAC Transmit Errors	Frames for which transmission does not finish due to detected internal MAC sublayer transmit error
Carrier Sense Errors	Times that the carrier sense condition was lost or never asserted when attempting to transmit a frame
Frame Too Long	Frames received that exceed the maximum permitted frame size
MAC Receive Errors	Frames for which reception on an interface does not finish due to a detected internal MAC sublayer receive error

Communication Module Bandwidth Diagnostics

Introduction

Use the **Bandwidth** page to view the dynamic and static data for the bandwidth use by the Ethernet communications module.

NOTE: Before you can open the diagnostics page, create a connection between the DTM for the target communication module and the physical module, page 283.

Open the Page

Access the **Bandwidth** information:

Step	Action
1	Open the Diagnosis information for your module, page 264.
2	In the left pane of the **Diagnosis** window, select the communication module node.
3	Select the **Bandwidth** tab to open that page.

Bandwidth Diagnostic Parameters

The **Bandwidth** page displays these parameters for the communication module:

Parameter	**Description**
I/O - Scanner:	
EtherNet/IP Sent	The number of EtherNet/IP packets the module has sent in packets/second.
Ether Received	The number of EtherNet/IP packets the module has received in packets/second.
Modbus TCP Received	The number of Modbus TCP requests the module has sent in packets/second.
Modbus TCP Responses	The number of Modbus TCP responses the module has received in packets/second.
I/O - Adapter:	
EtherNet/IP Sent	The number of EtherNet/IP packets (per second) the module has sent in the role of a local slave.
EtherNet/IP Received	The number of EtherNet/IP packets (per second) the module has received in the role of a local slave.
I/O - Module:	
Module Capacity	The maximum number of packets (per second) that the module can process.
Module Utilization	The percentage of communication module capacity used by the application.
Messaging - Client:	
EtherNet/IP Activity	The number of explicit messages (packets per second) sent by the module using the EtherNet/IP protocol.
Modbus TCP Activity	The number of explicit messages (packets per second) sent by the module using the Modbus TCP protocol.
Messaging - Server:	
EtherNet/IP Activity	The number of server messages (packets per second) received by the module using the EtherNet/IP protocol.
Modbus TCP Activity	The number of server messages (packets per second) received by the module using the Modbus TCP protocol.
Module:	
Processor Utilization	The percent of the processing capacity of the Ethernet communication module that is used by the present level of communication activity.

Communication Module RSTP Diagnostics

Introduction

Use the **RSTP Diagnostic** page to view the status of the RSTP service of the Ethernet communications module. The page displays dynamically generated and static data for the module.

NOTE: Before you can open the diagnostics page, create a connection between the DTM for the target communication module and the physical module, page 283.

Open the Page

Access the **RSTP Diagnosis** information:

Step	Action
1	Open the **Diagnosis** window, page 270.
2	In the left pane, select the communication module node.
3	Select the **Bandwidth** tab to open that page.
4	In the left pane of the **Diagnosis** window, select the communication module node.
5	Select the **RSTP Diagnostic** tab to open that page.

RSTP Diagnostic Parameters

The **RSTP Diagnostic** page displays the following parameters for each communication module port.

Bridge RSTP Diagnostic Parameters:

Parameter	Description
Bridge Priority	This eight-byte field contains the two-byte value that is assigned to the module embedded Ethernet switch.
MAC Address	The Ethernet address of the module, found on the front of the module.
Designated Root ID	The Bridge ID of the root device.
Root Path Cost	The aggregate cost of port costs from this switch back to the root device.
Default Hello Time	The interval at which Configuration BPDU messages are transmitted during a network convergence. For RSTP this is a fixed value of 2 seconds.

Parameter	Description
Learned Hello Time	This active Hello Time value is learned from the root switch.
Configured Max Age	The value (6 ... 40) that other switches use for *MaxAge* when this switch acts as the root.
Learned Max Age	The maximum age learned from the root switch. The switch uses this active value.
Total Topology Changes	The total number of topology changes detected by this switch since the management entity was last reset or initialized.

Ports ETH 2 and ETH 3 RSTP Statistics:

Parameter	Description
Status	The state of the port as defined by the RSTP protocol. This state controls the action the port takes when it receives a frame. Possible values are: *disabled*, *discarding*, *learning*, *forwarding*.
Role	The role of the port according to the RSTP protocol. Possible values are: *root port*, *designated port*, *alternate port*, *backup port*, *disabled port*.
Cost	The logical cost of this port as a path to the root switch. If this port is configured for AUTO, the cost is determined based on the connection speed of the port.
STP Packets	A value in this field indicates that a device on the network has the STP protocol enabled. **NOTE:** • Other devices that are enabled for STP can severely affect the network convergence times. Disable the STP protocol (but not the RSTP protocol) for every network device that supports STP. • The communication module does not support the STP protocol. The module embedded switch ignores STP packets.

Network Time Service Diagnostics

Introduction

In EcoStruxure Control Expert, configure the network time service before you use the **Network Time Service Diagnostic** page to view the dynamically generated data that describes the operation of the network time protocol (NTPv4).

NOTE:

- Before you can open the diagnostics page, create a connection between the DTM for the target communication module and the physical module, page 283.
- For detailed diagnostic information, refer to the *System Time Stamping, User Guide*.

Open the Page

Access the **NTP Diagnostic** information:

Step	Action
1	Open the **Diagnosis** information for your module, page 264.
2	In the left pane of the **Diagnosis** window, select the communication module node.
3	Select the **NTP Diagnostic** tab to open that page.

Click the **Reset Counter** button to reset the counting statistics on this page to *0*.

Network Time Service Diagnostic Parameters

This table describes the time synchronization service parameters:

Parameter	Description
Refresh Every 500ms	Select (check) this box to dynamically update the page every 500ms. The number of times this page was refreshed appears immediately to the right.
Network Time Service	Monitor the operational status of the service in the module: • *green*: operational • *orange*: disabled
Network Time Server Status	Monitor the communication status of the NTP server: • *green*: The NTP server is reachable. • *red*: The NTP server is not reachable.
Last Update	Elapsed time, in seconds, since the most recent NTP server update.
Current Date	System date
Current Time	The system time is presented in the *hh:mm:ss* format.
DST Status	Set the status of the automatic daylight savings service: • *ON*: The automatic adjustment of daylight savings is enabled and implemented. • *OFF*: The automatic adjustment of daylight savings is disabled. (The implemented date and time may not reflect daylight savings time.)
Quality	This correction (in seconds) applies to the local counter at every NTP server update. Numbers greater than 0 indicate increasingly excessive traffic condition or an NTP server overload.
Requests	This value represents the total number of client requests sent to the NTP server.
Responses	This value represents the total number of server responses sent from the NTP server.
Errors	This value represents the total number of unanswered NTP requests.
Last Error	This value indicates the last detected error code received from the NTP client: • 0: valid NTP configuration • 1: late NTP server response (can be caused by excessive network traffic or server overload) • 2: NTP not configured • 3: invalid NTP parameter setting • 4: NTP component disabled • 5: primary and secondary IP addresses that are not NTP server addresses • 7: unrecoverable NTP transmission • 9: invalid NTP server IP address • 15: invalid syntax in the custom time zone rules file

Parameter	Description
Primary / Secondary NTP Server IP	The IP addresses correspond to the primary and secondary NTP servers. **NOTE:** A green LED to the right of the primary or secondary NTP server IP address indicates the active server.
Auto Adjust Clock for Daylight Savings	Configure the daylight savings adjustment service: • enabled • disabled
DST Start / DST End	Specify the day on which daylight savings time starts and stops: • *Month:* Set the month in which daylight savings time starts or stops. • *Day of Week:* Set the day of the week on which daylight savings time starts or stops. • *Week#:* Set the occurrence of the specified day within the specified month.
Time Zone	Select the time zone plus or minus Universal Time, Coordinated (UTC).
Offset	Configure the time (in minutes) to be combined with the time zone selection (above) to produce the system time.
Polling Period	Set the frequency with which the NTP client requests an updated time from the NTP server.

Local Slave / Connection Diagnostics

Introduction

Use the **Local Slave Diagnostic** page and the **Connection Diagnostic** page to view the I/O status and production/consumption information for a selected local slave or connection.

NOTE: Before you can open the diagnostics page, create a connection between the DTM for the target communication module and the physical module, page 283.

Open the Page

Access the diagnostics information:

Step	Action
1	Open the **Diagnosis** information for your module, page 264.
2	In the left pane of the **Diagnosis** window, select the communication module node.
3	Select the **Local Slave Diagnostic** tab or the **Connection Diagnostic** tab to open that page.

Data Display

Use the **Refresh Every 500ms** checkbox to view static or dynamic data:

Checkbox	Description
selected	• Display data that is dynamically updated every 500 ms. • Increment the number at the top of the table each time data is refreshed.
deselected	• Display static data. • Do not increment the number at the top of the table. That number now represents a constant value.

Local Slave / Connection Diagnostic Parameters

This following tables display the diagnostic parameters for the selected local slave or scanner connection.

This table shows the **Status** diagnostic parameters for the selected connection:

Parameter	**Description**
Input	This integer represents the input status.
Output	This integer represents the output status.
General	This integer represents the basic connection status.
Extended	This integer represents the extended connection status.

The **Input** and **Output** status diagnostic parameters can present these values:

Input/Output Status (dec)	**Description**
0	OK
33	Time-out
53	IDLE
54	Connection established
58	Not connected (TCP)
65	Not connected (CIP)
68	Connection establishing
70	Not connected (EPIC)
77	Scanner stopped

This table shows the **Counter** diagnostic parameters for the selected connection:

Parameter	**Description**
Frame Error	Increments each time a frame is not sent by missing resources or is impossible to send.
Time-Out	Increments each time a connection times out.
Refused	Increments when connection is refused by the remote station.
Production	Increments each time a message is produced.
Consumption	Increments each time a message is consumed.
Production Byte	Total of produced messages, in bytes, since the communication module was last reset.
Consumption Byte	Total of consumed messages, in bytes, since the communication module was last reset.
Theoretical Packets per second	Packets per second calculated using the active configuration value.
Real Packets per second	Number of packets per second generated by this connection.

This table shows the **Diagnostic** parameters for the selected connection:

Parameter	**Description**
CIP Status	An integer representing CIP status.
Extended Status	An integer representing extended CIP status.
Production Connection ID	The connection ID.
Consumption Connection ID	The connection ID.
O -> T API	Packet interval (API) of the production connection.
T -> O API	Packet interval (API) of the consumption connection.
O -> T RPI	Requested packet interval (RPI) of the production connection.
T -> O RPI	Requested packet interval (RPI) of the consumption connection.

This table shows the **Socket Diagnostics** diagnostic parameters for the selected connection:

Parameter	Description
Socket ID	Internal Identification of the socket.
Remote IP Address	IP address of the remote station for this connection.
Remote Port	Port number of the remote station for this connection.
Local IP Address	IP address of the communication module for this connection.
Local Port	Port number of the communication module for this connection.

Local Slave or Connection I/O Value Diagnostics

Introduction

Use the **I/O Values** page to view both the input data image and output data image for the selected local slave or scanner connection.

NOTE: Before you can open the diagnostics page, connect the DTM to the module, page 283.

Open the Page

Access the **I/O Values** information:

Step	Action
1	Open the Diagnosis information for your module, page 264.
2	In the left pane of the **Diagnosis** window, select the communication module node.
3	Select the **I/O Values** tab.

Data Display

Use the **Refresh Every 500ms** checkbox to view static or dynamic data:

Checkbox	Description
selected	• Display data that is dynamically updated every 500 ms. • Increment the number at the top of the table each time data is refreshed.
deselected	• Display static data. • Do not increment the number at the top of the table. That number now represents a constant value.

Local Slave / Scanner Connection I/O Values

This page displays theses parameters for either a local slave or a remote device connection input and output values:

Parameter	Description
Input/Output data display	This parameter displays the input or output data image for a local slave or remote device.
Length	The *Length* parameter shows the number of bytes in an input or output data image.
Status	The *Status* parameter indicates the status of the scanner diagnostic object that is reported in the input or output data image: • *0*: The connection is OK. • *54*: The connection is in progress. The I/O data are not exchanged. • *33*: There is no connection. • *53*: A notification of IDLE is received.

Online Action

About Online Action

Online Parameters

Use the **Online Action** page in the EcoStruxure Control Expert DTM to view and edit online parameters for the Ethernet communications module. Online actions support these tasks:

- Display EtherNet/IP objects for the Ethernet communications module or a distributed EtherNet/IP device.
- View and edit the SERVICE port configuration parameters for the Ethernet communications module.
- Ping the Ethernet communications module or a distributed EtherNet/IP or Modbus TCP device to verify that it is active on the Ethernet network.
- Connect to a distributed device to perform these actions:
  - View the default parameter settings for the device.
  - View the active parameter settings for the device.
  - Edit and download to the device its editable parameter settings.

Connect the DTM

Before you can open the **Online Action** page, make the connection between the DTM for the target communication module and the physical module:

Step	Action
1	In the **DTM Browser**, find the name that is assigned to your Ethernet communications module.
2	Right-click the module name and scroll to **Connect**.
3	Right-click the module name and scroll to **Online Action** (**Device menu > Additional functions > Online Action**).

You can now view these **Online Action** tabs:

- **EtherNet/IP Objects**, page 284
- **Port Configuration**, page 285
- **Ping**, page 286

EtherNet/IP Objects Tab

Introduction

Select the **EtherNet/IP Objects** tab after you access the **Online Action** information to perform these actions:

- Retrieve and display data that describes the state of CIP objects for the selected communication module or remote EtherNet/IP device.
- Reset the selected communication module or remote EtherNet/IP device.

Available CIP Objects

You can retrieve CIP objects according to the EcoStruxure Control Expert operating mode:

Mode	Available CIP Objects
Standard	Identity object, page 291
Advanced	Identity object, page 291
	Connection Manager object, page 299
	TCP/IP Interface object, page 311
	Ethernet Link object, page 314
	QoS object, page 304

Advanced Mode

Select an object in the list after you enable Advanced Mode, page 77. These buttons are available:

Button	Action
Refresh	Click this button to update the data.
Reset Device	Click this button to reset a communication module or remote EtherNet/IP device.

Port Configuration Tab

Introduction

Use the **Port Configuration** tab in the **Online Action** window to view and edit communication port properties for a distributed EtherNet/IP device. Use this tab to execute these commands:

- *Refresh*: Use a Get command to retrieve port configuration settings from a distributed EtherNet/IP device.
- *Update*: Use a Set command to write the selected edited values to the same distributed EtherNet/IP device.

The information on the **Port Configuration** tab is sent in EtherNet/IP explicit messages that employ the address and messaging settings configured for EtherNet/IP explicit messaging (below).

Access the Page

Open the **Port Configuration** tab:

Step	Action
1	Select the **Port Configuration** tab after you access the **Online Action** information, page 283.
2	Configure the Service port with the instructions from the offline configuration. (Refer to the service port configuration directions, page 114.)
3	Click the **Update** button to apply the new configuration.

Ping Tab

Ping Functionality

Select the **Ping** tab after you access the **Online Action** information. Then you can send an ICMP echo request to a target Ethernet device to:

- Determine if the target device is present.
- Report the elapsed time until the echo response from the present target device.

Field	Parameter	Description
Address	*IP Address*	Enter a valid IP address for the ping target device.
Ping	*Repeat (100ms)*	Select (check) this box to repeat the ping if no reply is received.
	Stop on Error	Select (check) this box to stop repeating the ping if an error is detected when *Repeat (100ms)* is selected.
	Ping	Click this button to ping the selected *IP Address*.
	Clear	Click this button to clear the contents of the *Ping Result* display.
	Ping Result	This text field displays the result of the ping.

Ping a Network Device

Step	Action
1	In the **DTM Browser**, select the communication module upstream of the remote EtherNet/IP device you want to ping.
2	Right-click to select **Device Menu > Online Action** in the pop-up menu. The **Online Action** window opens.
3	In the **Online Action** window, select the device you want to ping. The window displays pages containing online information for the selected device. **NOTE:** The specific collection of displayed pages depends on the type of device selected: • the communications module • a remote EtherNet/IP device • a remote Modbus TCP device
4	Select the **Ping** page: • *selected:* Check the **Repeat** box to send a series of pings (every 100ms). • *deselected:* Uncheck the **Repeat** box to send a single ping.
5	(Optional) Select **Stop on Error** to stop pinging an unsuccessful communication.

Step	Action
6	Click **Ping** once to begin pinging.
7	Click **Ping** again to stop repeated pinging when an error is not detected.
8	The **Ping Result** box displays the ping outcome. Click **Clear** to empty the **Ping Result** box.

Diagnostics Available through Modbus/TCP

Modbus Diagnostic Codes

Introduction

Controllers and BMENOC0302(H) communication modules in M580 systems support the diagnostic codes in these tables.

Function Code 3

Some module diagnostics (I/O connection, extended health, redundancy status, FDR server, etc.) are available to Modbus clients that read the local Modbus server area. Use Modbus function code 3 with the unit ID set to 100 for register mapping:

Type	Offset Modbus Address	Size (Words)
Basic Networks Diagnostic Data	0	39
Ethernet Port Diagnostics Data (Internal port)	39	103
Ethernet Port Diagnostics Data (**ETH 1**)	142	103
Ethernet Port Diagnostics Data (**ETH 2**)	245	103
Ethernet Port Diagnostics Data (**ETH 3**)	348	103
Ethernet Port Diagnostics Data (rack)	451	103
Modbus TCP/Port 502 Diagnostic Data	554	114
Modbus TCP/Port 502 Connection Table Data	668	515
SNTP Diagnostics	1218	57
QoS Diagnostics	1275	11
Identify	2001	24

For a description of available function codes refer to the list of supported Modbus diagnostic codes in the *Quantum EIO, Control Network, Installation and Configuration Guide*.

Read Device Identification

Modbus function code 43, subcode 14: A Modbus request associated with function code 43 (Read Device Identification) asks a Modbus server to return the vendor name, product name, version number, and other optional fields:

Category	Object ID	Object Name	Type	Requirement
Basic	00 hex	VendorName (vendor name)	ASCII string	required
	01 hex	ProductCode (product code)	ASCII string	required
	02 hex	MajorMinorRevision (version number)	ASCII string	required
Regular	03 hex	VendorUrl (vendor URL)	ASCII string	optional
	04 hex	ProductName (product name)	ASCII string	optional
	05 hex	ModelName (model name)	ASCII string	optional
	06 hex	UserApplicationName (user application name)	ASCII string	optional
	07...7F hex	(reserved)	ASCII string	optional
Extended	80...FF hex	device-dependent		optional

This table provides sample responses to the Modbus request (function code 43, subcode 14):

Module	00 hex Vendor ID	01 hex Part Number	02 hex Version
BMENOC0302(H)	Schneider Electric	BMENOC0302(H)	V01.01 build 0032

Diagnostics Available through EtherNet/IP CIP Objects

Introduction

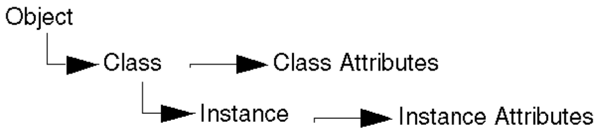
Modicon M580 applications use CIP within a producer/consumer model to provide communication services in an industrial environment. This section describes the available CIP objects for Modicon M580 modules.

About CIP Objects

Overview

The Ethernet communication module can access CIP data and services located in connected devices. The CIP objects and their content depend on the design of each device.

CIP object data and content are exposed and accessed hierarchically in these nested levels:



NOTE: Use explicit messaging to access these items:

- Access a collection of instance attributes by including only the class and instance values for the object in the explicit message.
- Access a single attribute by adding a specific attribute value to the explicit message with the class and instance values for the object.

The following information describes the CIP objects that the Ethernet communication module exposes to remote devices.

Identity Object

Overview

The Identity object presents the instances, attributes and services described below.

Class ID

01

Instance IDs

The Identity object presents two instances:

- 0: class
- 1: instance

Attributes

Identity object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET
hex	dec				
01	01	Vendor ID	UINT	X	—
02	02	Device Type	UINT	X	—
03	03	Product Code	UINT	X	—
04	04	Revision	STRUCT	X	—
		Major	USINT		
		Minor	USINT		
05	05	Status: • bit 2: 01 hex = the module is configured • bits 4...7: ◦ 03 hex = no I/O connections established ◦ 06 hex = at least one I/O connection in run mode ◦ 07 hex = at least one I/O connection established in IDLE mode • bit 8: TRUE = internal, recoverable detected error that does not result in an error state • bit 9: TRUE = internal, unrecoverable detected error that does not result in an error state • bit 10: TRUE = internal detected error that sends device to recoverable error state • bit 11: TRUE = internal detected error that sends device to unrecoverable error state • bits 12...15: reserved (0 or vendor-specific) = EDS indicates that the device instance follows a vendor-specific definition for these bits with the *DeviceStatusAssembly2* keyword.	Word	X	—
06	06	Serial Number	UDINT	X	—
07	07	Product Name	STRING	X	—
12	18	Modbus Identity	STRUCT	X	—
X = supported — = not supported					

Services

The Identity object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns: • all class attributes (instance = 0) • instance attributes 1 to 7 (instance = 1)
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported					

Message Router Object

Overview

The Message Router object provides a messaging connection point through which a client may address a service to any object class or instance residing in the physical device.

Class ID

02 (hex and decimal)

Instance IDs

The Message Router object presents two instances:

- 0: class
- 1: instance

Attributes

Message Router object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID (hex and dec)	**Description**	**GET**	**SET**
01	Revision	X	—
02	Maximum Instance	X	—
03	Number of Instances	X	—
04	Optional Attribute List	X	—
05	Optional Service List	X	—
06	Maximum Number of Class Attributes	X	—
07	Maximum Number of Instance Attributes	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Object list	STRUCT of	X	—	A list of supported objects (i.e. a structure with an array of object class codes supported by the device)
		Number	UINT	X	—	The number of supported classes (i.e. class codes) in the classes array
		Classes	Array of UINT	X	—	List of supported class codes supported by the device
02	02	Number Available	UINT	X	—	Maximum number of connections supported
03	03	Number Active	UINT	X	—	Number of connections allocated to system communication
X = supported — = not supported						

Services

The Message Router object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns: • all class attributes (instance = 0) • instance attributes 1 to 7 (instance = 1)
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported					

Assembly Object

Overview

The assembly object consists of the attributes and services. Assembly instances exist only when you configure local slaves for the Ethernet communications module, page 229.

You can send an explicit message to the assembly object only when no other connections are established to read from this object. For example, you can send an explicit message to the assembly object if a local slave instance is enabled, but no other module is scanning that local slave.

Class ID

04

Instance IDs

The assembly object presents these instance identifiers:

- 0: class
- 101, 102, 111, 112, 121, 122, 131, 132, 136, 137, 141, 142, 146, 147, 151, 152, 156, 157, 161, 162, 166, 167, 171, 172, 176, 177, 181, 182, 186, 187, 191, 192: instance

Attributes

The assembly object consists of these attributes:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
03	Number of Instances	X	—
X = supported — = not supported			

Instance attributes:

Instance ID	Attribute ID	Description	Type	GET	SET
101	03	Local slave 1: T->O (output data)	Array of BYTE	X	—
102		Local slave 1: O>T (input data)	Array of BYTE	X	—
111	03	Local slave 2: T->O (output data)	Array of BYTE	X	—
112		Local slave 2: O>T (input data)	Array of BYTE	X	—
121	03	Local slave 3: T->O (output data)	Array of BYTE	X	—
122		Local slave 3: O>T (input data)	Array of BYTE	X	—
131	03	Local slave 4: T->O (output data)	Array of BYTE	X	—
132		Local slave 4: O>T (input data)	Array of BYTE	X	—
136	03	Local slave 5: T->O (output data)	Array of BYTE	X	—
137		Local slave 5: O>T (input data)	Array of BYTE	X	—
141	03	Local slave 6: T->O (output data)	Array of BYTE	X	—
142		Local slave 6: O>T (input data)	Array of BYTE	X	—
146	03	Local slave 7: T->O (output data)	Array of BYTE	X	—
147		Local slave 7: O>T (input data)	Array of BYTE	X	—
151	03	Local slave 8: T->O (output data)	Array of BYTE	X	—
152		Local slave 8: O>T (input data)	Array of BYTE	X	—
156	03	Local slave 9: T->O (output data)	Array of BYTE	X	—
157		Local slave 9: O>T (input data)	Array of BYTE	X	—
161	03	Local slave 10: T->O (output data)	Array of BYTE	X	—
162		Local slave 10: O>T (input data)	Array of BYTE	X	—
166	03	Local slave 11: T->O (output data)	Array of BYTE	X	—
167		Local slave 11: O>T (input data)	Array of BYTE	X	—
171	03	Local slave 12: T->O (output data)	Array of BYTE	X	—
172		Local slave 12: O>T (input data)	Array of BYTE	X	—
176	03	Local slave 13: T->O (output data)	Array of BYTE	X	—
177		Local slave 13: O>T (input data)	Array of BYTE	X	—
181	03	Local slave 14: T->O (output data)	Array of BYTE	X	—
182		Local slave 14: O>T (input data)	Array of BYTE	X	—

Instance ID	Attribute ID	Description	Type	GET	SET
186	03	Local slave 15: T->O (output data)	Array of BYTE	X	—
187		Local slave 15: O>T (input data)	Array of BYTE	X	—
191	03	Local slave 16: T->O (output data)	Array of BYTE	X	—
192		Local slave 16: O>T (input data)	Array of BYTE	X	—
X = supported — = not supported					

Services

The CIP assembly object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute
X = supported					

Connection Manager Object

Overview

The Connection Manager object presents the instances, attributes and services described below.

Class ID

06

Instance IDs

The Connection Manager object presents two instance values:

- 0: class
- 1: instance

Attributes

Connection Manager object attributes are associated with each instance, as follows.

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Open Requests	UINT	X	—	Number of Forward Open service requests received
02	02	Open Format Rejects	UINT	X	—	Number of Forward Open service requests that were rejected due to incorrect format
03	03	Open Resource Rejects	UINT	X	—	Number of Forward Open service requests that were rejected due to lack of resources
04	04	Open Other Rejects	UINT	X	—	Number of Forward Open service requests that were rejected for reasons other than incorrect format or lack of resources
05	05	Close Requests	UINT	X	—	Number of Forward Close service requests received
06	06	Close Format Requests	UINT	X	—	Number of Forward Close service requests that were rejected due to incorrect format
07	07	Close Other Requests	UINT	X	—	Number of Forward Close service requests that were rejected for reasons other than incorrect format
08	08	Connection Timeouts	UINT	X	—	Total number of connection timeouts that occurred in connections controlled by this connections manager
09	09	Connection Entry List	STRUCT	X	—	0 (Unsupported optional item)
0B	11	CPU_Utilization	UINT	X	—	0 (Unsupported optional item)
0C	12	MaxBuffSize	UDINT	X	—	0 (Unsupported optional item)
0D	13	BufSize Remaining	UDINT	X	—	0 (Unsupported optional item)
X = supported — = not supported						

Services

The Connection Manager object performs these services on the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported					

Modbus Object

Overview

The Modbus object converts EtherNet/IP service requests to Modbus functions, and Modbus exception codes to CIP General Status codes. It presents the instances, attributes and services described below.

Class ID

44 (hex), 68 (decimal)

Instance IDs

The Modbus object presents two instance values:

- 0: class
- 1: instance

Attributes

The Modbus object consists of these attributes:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID	Description	Type	GET	SET
—	No instance attributes are supported	—	—	—
— = not supported				

Services

The Modbus object performs these services upon the listed object types:

Service ID		Description	Class	Instance
hex	dec			
0E	14	Get_Attribute_Single	X	X
4B	75	Read_Discrete_Inputs	—	X
4C	76	Read_Coils	—	X
4D	77	Read_Input_Registers	—	X
4E	78	Read_Holding_Registers	—	X
4F	79	Write_Coils	—	X
50	80	Write_Holding_Registers	—	X
51	81	Modbus_Passthrough	—	X
X = supported — = not supported				

Quality Of Service (QoS) Object

Overview

The QoS object implements Differentiated Services Code Point (DSCP or *DiffServe*) values for the purpose of providing a method of prioritizing Ethernet messages. The QoS object presents the instances, attributes and services described below.

Class ID

48 (hex), 72 (decimal)

Instance IDs

The QoS object presents two instance values:

- 0: class
- 1: instance

Attributes

The QoS object consists of these attributes:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID	Description	Type	GET	SET	Value
04	DSCP Urgent	USINT	X	—	For CIP transport class 0/1 Urgent priority messages.
05	DSCP Scheduled	USINT	X	—	For CIP transport class 0/1 Urgent priority messages.
06	DSCP High	USINT	X	—	For CIP transport class 0/1 Urgent priority messages.
07	DSCP Low	USINT	X	—	For CIP transport class 0/1 Urgent priority messages.
08	DSCP Explicit	USINT	X	—	For CIP explicit messages (transport class 2/3 and UCMM).
X = supported — = not supported					

NOTE: A change in the instance attribute value takes effect on device re-start, for configurations made from flash memory.

Services

The QoS object performs these services upon the listed object types:

Service ID		Description	Class	Instance
hex	dec			
0E	14	Get_Attribute_Single	X	X
X = supported				

Port Object

Overview

The Port object describes the communication interfaces that exist on the device and that are visible to CIP.

Class ID

F4 (hex), 244 (decimal)

Instance IDs

The Port object presents two instances:

- 0: class
- 1: instance

Attributes

Port object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID (hex and dec)	Description	GET	SET
01	Revision	X	—
02	Maximum Instance	X	—
03	Number of Instances	X	—
04	Optional Attribute List	X	—
05	Optional Service List	X	—
06	Optional Maximum Number of Class Attributes	X	—
07	Optional Maximum Number of Instance Attributes	X	—
08	Entry Port Returns the instance of the Port object that describes the port through which this request entered the device	X	—

Attribute ID (hex and dec)	Description	GET	SET
09	Port Instance Information Array of structures containing instance attributes 1 and 2 (see below) from each port instance	X	—
	Port Type (see Instance attribute 01)	X	—
	Port Number (see Instance attribute 02)	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Port Type	UINT	X	—	• 0: Routing not supported • 1: Vendor specific • 2: ControlNet • 3: ControlNet Redundant • 4: EtherNet/IP (formerly TCP/IP) • 5: DeviceNet • 6-199: Vendor specific • 200: CompoNet • 201: Modbus/TCP • 202: Modbus/SL • 203: SERCOS III • 204: HART • 205: IO-Link • 206-65535: Reserved
02	02	Port Number		X	—	The CIP number
03	03	Logical Link Object	STRUCT of	X	—	A list of supported objects (i.e. a structure with an array of object class codes supported by the device)
		Path Length	UINT	X	—	The number of 16-bit words in the following path.
		Link Path	Padded EPATH	X	—	Logical path segments that identify the object for this port.
04	04	Port Name	SHORT_ STRING	X	—	String name of port interface name, up to 64 characters

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
05	05	Port Type Name	SHORT_ STRING	X	—	String name of port interface type, up to 64 characters
06	06	Port Description	SHORT_ STRING	X	—	String that describes the port
07	07	Port Number and Node Address	Padded EPATH	X	—	A single port segment containing the Port Number of this port and the Link Address of this device on this port.
08	08	Port Node Range	STRUCT of	X	—	
		Minimum Node Number	UINT	X	—	For example, on port.
		Maximum Node Number	UINT	X	—	For example, on port.
09	09	Chassis Identity	Padded EPATH	X	—	Electronic key of the chassis to which this port is attached. This attribute is a single Logical Electronic Key Segment with Format 4 of the Logical Electronic Key segment.
A	10	Port Routing Capabilities	DWORD	X	—	Bit string defining the routing capabilities of this port, where 0= not-supported, 1=supported: • bit 0: Incoming unconnected messages • bit 1: Outgoing unconnected messages • bit 2: Incoming transport class 0/ 1 connections • bit 3: Outgoing transport class 0/ 1 connections • bit 4: Incoming transport class 2/ 3 connections • bit 5: Outgoing transport class 2/ 3 connections • bit 6: Outgoing DeviceNet CIP safety-related connections (only for DeviceNet ports) • bits 7-31: Reserved

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
B	11	Associated Communication Objects	STRUCT of	X	—	List of communication object instances associated with this instantiated Port Object (see list, below)
		Number of entries in following Array:	USINT	X	—	
			Array of STRUCT of	X	—	
		Number of 16 bit words in the following path	USINT	X	—	
		Logical path segments that identify an associated communication object instance	Padded EPATH	X	—	
X = supported — = not supported						

The list of Associated Communication Objects in Attribute 11 (dec) / B (hex) includes:

DeviceNet Object – 0x03	RSTP Port Object – 0x55	TCP/IP Interface Object – 0xF5
Modbus Object – 0x44	Parallel Redundancy Protocol Object – 0x56	Ethernet Link Object – 0xF6
Modbus Serial Link Object – 0x46	PRP Nodes Table Object – 0x57	0xF6 • CompoNet Link Object – 0xF7
Device Level Ring Object – 0x47	EtherNet/IP Security Object – 0x5E	CompoNet Repeater Object – 0xF8
QoS Object – 0x48	ControlNet Object – 0xF0	CompoNet Repeater Object – 0xF8
SERCOS III Link Object – 0x4C	ControlNet Keeper Object – 0xF1	IO-Link Master PHY Object – 0x10C
RSTP Bridge Object – 0x54	ControlNet Scheduling Object – 0xF2	

Services

The port object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns: • all class attributes (instance = 0) • instance attributes 1 to 7 (instance = 1)
10	10	Set_Attribute_Single	—	X	Modifies an attribute.
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported — = not supported					

TCP/IP Interface Object

Overview

The TCP/IP interface object presents the instances (per network), attributes and services described below.

Class ID

F5 (hex), 245 (decimal)

Instance IDs

The TCP/IP interface object presents two instance values:

- 0: class
- 1: instance

Attributes

TCP/IP interface object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID	Description	Type	GET	SET	Value
01	Status	DWORD	X	—	01 hex
02	Configuration Capability	DWORD	X	—	01 hex = from BOOTP 11 hex = from flash 00 hex = other
03	Configuration Control	DWORD	X	—	01 hex = out-of-box default
04	Physical Link Object	STRUCT	X	—	
	Path Size	UINT			
	Path	Padded EPATH			
05	Interface Configuration	STRUCT	X	—	00 hex = out-of-box default
	IP Address	UDINT			
	Network Mask	UDINT			
	Gateway Address	UDINT			
	Name Server	UDINT			
	Name Server 2	UDINT			
	Domain Name	STRING			
06	Host Name	STRING	X	—	
X = supported — = not supported					

Services

The TCP/IP interface object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported					

Ethernet Link Object

Overview

The Ethernet Link object consists of the instances, attributes, and services described below.

Class ID

F6 (hex), 246 (decimal)

Instance IDs

The Ethernet Link object presents these instance values:

- 0: class
- 1: ETH 1
- 2: ETH 2
- 3: ETH 3
- 4: rack port
- 255: internal port

Attributes

The Ethernet Link object presents these attributes:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
03	Number of Instances	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Interface Speed	UDINT	X	—	Valid values: 0, 10, 100.
02	02	Interface Flags	DWORD	X	—	Bit 0: link status 0 = Inactive 1 = Active
						Bit 1: duplex mode 0 = half duplex 1 = full duplex
						Bits 2...4: negotiation status 3 = successfully negotiated speed and duplex 4 = forced speed and link
						Bit 5: manual setting requires reset 0 = automatic 1 = reset the device
						Bit 6: local hardware detected error 0 = no event 1 = event detected
03	03	Physical Address	ARRAY of 6 USINT	X	—	module MAC address

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
04	04	Interface Counters	STRUCT	X	—	
		In octets	UDINT			octets received on the interface
		In Ucast Packets	UDINT			unicast packets received on the interface
		In NUcast Packets	UDINT			non-unicast packets received on the interface
		In Discards	UDINT			inbound packets received on the interface, but discarded
		In Errors	UDINT			inbound packets with detected errors (does not include in discards)
		In Unknown Protos	UDINT			inbound packets with unidentified protocol
		Out Octets	UDINT			octets sent on the interface
		Out Ucast Packets	UDINT			unicast packets sent on the interface
		Out NUcast Packets	UDINT			non-unicast packets sent on the interface
		Out Discards	UDINT			outbound packets discarded
		Out Errors	UDINT			outbound packets with detected errors

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
05	05	Media Counters	STRUCT	X	—	
		Alignment Errors	UDINT			frames that are not an integral number of octets in length
		FCS Errors	UDINT			invalid CRC — frames received do not pass the FCS check
		Single Collisions	UDINT			successfully transmitted frames that experienced exactly 1 collision
		Multiple Collisions	UDINT			successfully transmitted frames that experienced more than 1 collision
		SQE Test Errors	UDINT			number of times the detected SQE test error is generated
		Deferred Transmissions	UDINT			frames for which first transmission attempt is delayed because the medium is busy
		Late Collisions	UDINT			number of times a collision is detected later than 512 bit times into the transmission of a packet
		Excessive Collisions	UDINT			frames that do not transmit due to excessive collisions
		MAC Transmit Errors	UDINT			frames that do not transmit due to a detected internal MAC sublayer transmit error
		Carrier Sense Errors	UDINT			times that the carrier sense condition was lost or not asserted when attempting to transmit a frame
		Frame Too Long	UDINT			frames received that exceed the maximum permitted frame size
		MAC Receive Errors	UDINT			frames not received on an interface due to a detected internal MAC sublayer receive error

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
06	06	Interface Control	STRUCT	X	—	API of the connection
		Control Bits	WORD			Bit 0: Auto-negotiation disabled (0) or enabled (1). **NOTE:** When auto-negotiation is enabled, 0C hex (object state conflict) is returned when attempting to set either: • forced interface speed • forced duplex mode
						Bit 1: forced duplex mode (if auto-negotiation bit = 0) 0 = half duplex 1 = full duplex
		Forced Interface Speed	UINT			Valid values include 10000000 and 100000000. **NOTE:** An attempt to set any other value returns the detected error 09 hex (invalid attribute value).
10	16	Interface Label	SHORT_ STRING	X	—	A fixed textual string identifying the interface. The maximum number of characters is 64. **NOTE:** A valid instance of this strings includes *internal* for internal interfaces.
X = supported — = not supported						

Services

The Ethernet Link object performs these services upon the listed object types:

Service ID		Description	Class	Instance
hex	dec			
01	01	Get_Attributes_All	X	X
0E	14	Get_Attribute_Single	X	X
4C	76	Get_and_Clear	—	X
X = supported — = not supported				

Module Diagnostic Object

Overview

The Module Diagnostic object presents the instances, attributes and services described below.

Class ID

300 (hex), 768 (decimal)

Instance IDs

The Module Diagnostic object presents two instances:

- 0: class
- 1: instance

Attributes

Module Diagnostic object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Maximum Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Module Status	WORD	X	—	• 0x01 = STARTED • 0x02 = STOPPED • 0x03 = RUNNING
02	02	CNF Version	WORD	X	—	
03	03	CRC	UDINT	X	—	
04	04	Connection Status	STRUCT of	X	—	
		Size Table	WORD			In bytes -16 bytes
		Table	WORD[ ]			Padded on word • Describes I/O connections. • Each bit describes one I/O connection. The first bit is the first I/O connection. • Value 1 indicates that INPUT and OUTPUT status of an I/O connection are OK (status equal to 0). • Value 0 indicates that INPUT and OUTPUT status of an I/O connection are not OK (status not equal to 0). • The table consists of 8 words (128 I/O connections).
05	05	CCO Mode	WORD	X	—	• 0x00 = Block access to connection configuration object (CCO) • 0x01 = STOPPED
X = supported — = not supported						

Services

The Module Diagnostic object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
10	16	Set_Attribute_Single	—	X	Sets the value of the specified attribute.
X = supported — = not supported					

Scanner Diagnostic Object

Overview

The Scanner Diagnostic object presents the instances, attributes and services described below.

Class ID

301 (hex), 769 (decimal)

Instance IDs

The Scanner Diagnostic object presents two instances:

- 0: class
- 1: instance

Attributes

Scanner Diagnostic object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Maximum Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Control Bits	WORD	X	—	• TRUE = Activate checking time for production and consumption • FALSE = Inactive (default)
02	02	ST_DIAG_CNT	STRUCT of	X	—	
		wErrFrameCnt	UINT			Incremented each time a frame is not sent for lack of resources or was impossible to send.
		wErrTimeOutCnt	UINT			Incremented when one connection is timed out.
		wErrRefusedCnt	UINT			Incremented when one connection is refused by the remote station.
		dwProdCnt	UDINT			Incremented at each production.
		dwConsCnt	UDINT			Incremented at each consumption.
		dwProdByteCnt	UDINT			Total bytes produced.
		dwConsByteCnt	UDINT			Total bytes consumed.
03	03	Input Status	WORD	X	—	See below.
04	04	Output Status	WORD	X	—	See below.
05	05	ST_LINK	STRUCT of	X	—	
		CIP Status	UINT			See below.
		Extended Status	UINT			See below.
		Production Connection ID	DWORD			
		Consumed Connection ID	DWORD			
		OtoT API	UDINT			API of the Connection
		TtoO API (API of the Connection)	UDINT			API of the Connection
		OtoT RPI (RPI of the Connection)	UDINT			RPI of the Connection
		TtoO RPI (RPI of the Connection)	UDINT			RPI of the Connection

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
06	06	ST_SOCK_PARAM	STRUCT of	X	—	
		IpSockId	DWORD			Internal identifier
		IpForeign	DWORD			Remote station IP
		wPortForeign	UINT			Remote station port number
		IpLocal	DWORD			Local station IP
		wPortLocal	UINT			Local station port number
07	07	ST_PRODUCTION	STRUCT of	X	—	
		bValid	WORD			• 0 = STRUCT production data is not valid • 1 = STRUCT production data is valid
		dwCurrentTime	UDINT			Internal: number of ticks before next production
		dwProductionTime	UDINT			Internal: number of ticks between productions
		SequenceNumber	UDINT			Number of the sequence in the production
		stCheckTime	STRUCT of			
		dwLastTime	UDINT			Internal use
		dwMaxTime	UDINT			Maximum time between productions
		dwMinTime	UDINT			Minimum time between productions
		dwRPI	UDINT			Connection API
		wOverRun	UINT			Number of times the production was too long
		wUnderRun	UINT			Number of times the production was too fast
		dwCurrentTime	UDINT			Internal use

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
08	08	ST_ CONSUMPTION	STRUCT of	X	—	
		bValid	WORD			• 0 = STRUCT consumption data is not valid • 1 = STRUCT consumption data is valid
		dwCurrentTime	UDINT			Internal: number of ticks before timeout
		dwConsumption-Time	UDINT			Internal: number of ticks of the timeout
		SequenceNumber	UDINT			Number of the sequence in the consumption
		stCheckTime	STRUCT of			
		dwLastTime	UDINT			Internal use
		dwMaxTime	UDINT			Maximum time between consumptions
		dwMinTime	UDINT			Minimum time between consumptions
		dwRPI	UDINT			Connection API
		wOverRun	UINT			Number of times the consumption was too long
		wUnderRun	UINT			Number of times the consumption was too fast
		dwCurrentTime	UDINT			Internal use
09	09	CCO Status	STRUCT of	X	—	Status of the Connection Configuration Object – see below
		byGeneralStatus	BYTE			
		byReserved	BYTE			
		Extended	WORD			
X = supported — = not supported						

Status values for the Scanner Diagnostic object:

Status	Description	CIP Status	Extended	Context
0	OK	0	0	The IO data are correctly exchanged
33	Time-Out	0xFB	0xFB0B	Timeout detected on consumption
53	IDLE	0	0	An IDLE notification is received
54	Connection established	0	0	The connection is established, but the IO data are not consumed yet
		0xFB	0xFB08	Impossible to start the production
		0xFB	0xFB09	Impossible to start the consumption
		0xFB	0xFB0A	Not enough resources to manage the connection
58	Not connected (TCP)	0xFE	TCP Error	Error on TCP connection
65	Not connected (CIP)	status	extended	The Fw_Open response indicates a detected error.
		0xFB	0xFB01	Timeout for Fw_Open response
		0xFB	0xFB02	Incorrect format of the Fw_Open response
		0xFB	0xFB03	Incorrect parameters in the response (OT Net Par)
		0xFB	0xFB04	Incorrect parameters in the response (TO Net Par)
		0xFB	0xFB05	Asking port number different than 2222
		0xFB	0xFB06	Error in joining the UDP multicast group
		0xFB	0xFB07	Optimization error / indeterminable MAC address
68	Connection establishing	0xD0	0x0001	Connection is closed
		0xD0	0x0002	Connection is pending
70	Not connected (EPIC)	0xFD	Status	Error code in register session response
		0xFD	Status	Error code in the frame
		0xFD	Status	Encapsulation session unregistered
77	Scanner stopped	0	0	Connection is stopped

Services

The Scanner Diagnostic object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
61	97	Get_Output	—	X	Returns the status and value the output: • Offset 0 / UINT / Status • Offset 2 / USINT [0...409] / Output data
62	98	Get_Input	—	X	Returns the status and value the input: • Offset 0 / UINT / Status • Offset 2 / USINT [0...409] / Intput data
63	99	Set_DiagCounters	—	X	Sets the value of ST_Diag_CNT to 0..
X = supported — = not supported					

NOTE: If a service is addressed on an instance that does not exist or is not an I/O connection for the scanner, the service detects the following error: 0x05 – Path destination unknown.

Adapter Diagnostic Object

Overview

The Adapter Diagnostic object presents the instances, attributes and services described below.

Class ID

302 (hex), 770 (decimal)

Instance IDs

The Adapter Diagnostic object presents two instances:

- 0: class
- 1: instance

Attributes

Adapter Diagnostic object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Maximum Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Control Bits	WORD	X	—	• 0 = Deactivate (default) • 1 = Activate checking time for production and consumption.
02	02	ST_DIAG_CNT	STRUCT of	X	—	
		wErrFrameCnt	UINT			Incremented each time a frame is not sent for lack of resources or was impossible to send.
		wErrTimeOutCnt	UINT			Incremented when one connection is timed out.
		wErrRefusedCnt	UINT			Incremented when one connection is refused by the remote station.
		dwProdCnt	UDINT			Incremented at each production
		dwConsCnt	UDINT			Incremented at each consumption
		dwProdByteCnt	UDINT			Total bytes produced
		dwConsByteCnt	UDINT			Total bytes consumed
03	03	Input Status	WORD	X	—	See below.
04	04	Output Status	WORD	X	—	See below.
05	05	ST_LINK	STRUCT of	X	—	
		CIP Status	UINT			See below.
		Extended Status	UINT			See below.
		Production Connection ID	DWORD			
		Consumed Connection ID	DWORD			
		OtoT API	UDINT			API of the connection
		TtoO API	UDINT			API of the connection
		OtoT RPI	UDINT			RPI of the connection
		TtoO RPI	UDINT			RPI of the connection

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
06	06	ST_SOCK_PARAM	STRUCT of	X	—	
		IpSockId	DWORD			Internal Identifier
		IpForeign	DWORD			Remote station IP
		wPortForeign	UINT			Remote station port number
		IpLocal	DWORD			Local station IP
		wPortLocal	UINT			Local station port number
07	07	ST_PRODUCTION	STRUCT of	X	—	
		bValid	WORD			• 0 = STRUCT production data is not valid. • 1 = STRUCT production data is valid
		dwCurrentTime	UDINT			Internal – Number of ticks before next production
		dwProductionTime	UDINT			Internal – Number of ticks between production
		SequenceNumber	UDINT			Number of the sequence in the production
		stCheckTime	STRUCT of			
		dwLastTime	UDINT			Internal use
		dwMaxTime	UDINT			Maximum time between two productions
		dwMinTime	UDINT			Minimum time between two productions
		dwRPI	UDINT			API of the connection
		wOverRun	UINT			Number of times the production was too long
		wUnderRun	UINT			Number of times the production was too fast
		dwCurrentTime	UDINT			Internal use

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
08	08	ST_ CONSUMPTION	STRUCT	X	—	
		bValid	WORD			• 0 = STRUCT consumption data is not valid. • 1 = STRUCT consumption data is valid
		dwCurrentTime	UDINT			Internal – Number of ticks before timeout
		dwconsumption-Time	UDINT			Internal – Number of ticks of the timeout
		SequenceNumber	UDINT			Number of the sequence in the consumption
		stCheckTime	STRUCT			
		dwLastTime	UDINT			Internal use
		dwMaxTime	UDINT			Maximum time between two consumptions
		dwMinTime	UDINT			Minimum time between two consumptions
		dwRPI	UDINT			API of the connection
		wOverRun	UINT			Number of times the consumption was too long
		wUnderRun	UINT			Number of times the consumption was too fast
		dwCurrentTime	UDINT			Internal use
09	09	ASM Status	STRUCT of			See below.
		byGeneralStatus	BYTE			
		byReserved	BYTE			
		Extended Status	WORD			

X = supported

— = not supported

Adapter Diagnostic status values include the following:

Status	Description	CIP Status	Extended	Context
0	OK	0	0	The IO data are correctly exchanged
54	Connection in progress	0	0	The connection is in progress, but the IO data are not consumed yet.
33	No connection	0	0	No connection
		0xFB	0xFB01	Connection in timeout
		0xFB	0xFB07	Optimization error / indeterminable MAC address
		0xFB	0xFB0B	Timeout on consumption
		0xFB	0xFB0C	Connection closed by a forward close
		0xFB	0xFB0E	Module in STOP
		0xFD	Status	Error from Encapsulation layer
		0xFE	TCP Error	Error on TCP connection
		0x02	0	No more resource to handle the connections
		0x20	0	Connections refused because of incorrect format or parameters
53	IDLE	0	0	A notification of IDLE is received

Services

The Adapter Diagnostic object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns: • all class attributes (instance = 0) • instance attributes 1 to 7 (instance = 1)
61	97	Get_Output	—	X	Returns the status and value the output: • Offset 0 / UINT / Status • Offset 2 / USINT [0...409] / Output data
62	98	Get_Input	—	X	Returns the status and value the input: • Offset 0 / UINT / Status • Offset 2 / USINT [0...409] / Intput data
63	99	Set_DiagCounters	—	X	Sets the values of: • ST_Diag_CNT to 0. • ST_CHECK_TIME – both production and consumption – to 0 (but not the fields dwLastTime and dwCurrentTime)
X = supported — = not supported					

NOTE: If a service is addressed on an instance that does not exist, the service detects the following error: 0x05 – Path destination unknown.

EtherNet/IP Interface Diagnostics Object

Overview

The EtherNet/IP Interface Diagnostics object presents the instances, attributes and services described below.

Class ID

350 (hex), 848 (decimal)

Instance IDs

The EtherNet/IP Interface object presents two instance values:

- 0: class
- 1: instance

Attributes

EtherNet/IP Interface Diagnostics object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID	Description	Type	GET	SET	Value
01	Protocols Supported	UINT	X	—	
02	Connection Diagnostics	STRUCT	X	—	
	Max CIP IO Connections opened	UINT			Number of Class 1 connections opened since the last reset
	Active CIP IO Connections	UINT			Number of open Class 1 connections
	Max CIP Explicit Connections opened	UINT			Number of Class 3 connections opened since the last reset
	Active CIP Explicit Connections	UINT			Number of open Class 3 connections
	CIP Connections Opening Errors	UINT			Increments each time a Forward Open is not successful (Originator and Target)
	CIP Connections Timeout Errors	UINT			Increments when a connection times out (Originator and Target)
	Max EIP TCP Connections opened	UINT			Number of TCP connections (used for EIP, as client or server) opened since the last reset
	Active EIP TCP Connections	UINT			Number of open TCP connections (used for EIP, as client or server)
03	IO Messaging Diagnostics	STRUCT	X	—	
	IO Production Counter	UDINT			Increments each time a Class 0/1 message is sent
	IO Consumption Counter	UDINT			Increments each time a Class 0/1 message is received
	IO Production Send Errors Counter	UINT			Increments each time a Class 0/1 message is not sent
	IO Consumption Receive Errors Counter	UINT			Increments each time a consumption is received with a detected error

Attribute ID	Description	Type	GET	SET	Value
04	Explicit Messaging Diagnostics	STRUCT	X	—	
	Class 3 Msg Send Counter	UDINT			Increments each time a Class 3 message is sent (client and server)
	Class 3 Msg Receive Counter	UDINT			Increments each time a Class 3 message is received (client and server)
	UCMM Msg Receive Counter	UDINT			Increments each time a UCMM message is sent (client and server)
	UCMM Msg Receive Counter	UDINT			Increments each time a UCMM message is received (client and server)
X = supported — = not supported					

Services

The EtherNet/IP Interface Diagnostics object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
0E	14	Get_Attribute_Single	—	X	Returns the value of the specified attribute.
4C	76	Get_and_Clear	—	X	Returns and clears the values of all instance attributes.
X = supported — = not supported					

EtherNet/IP IO Scanner Diagnostics Object

Overview

The EtherNet/IP IO Scanner Diagnostics object presents the instances, attributes and services described below.

Class ID

351 (hex), 849 (decimal)

Instance IDs

The EtherNet/IP IO Scanner Diagnostics object presents two instances:

- 0: class
- 1: instance

Attributes

EtherNet/IP IO Scanner Diagnostics object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID	Description	Type	GET	SET
01	IO Status Table	STRUCT	X	—
	Size	UINT		
	Status	ARRAY of UNINT		
X = supported — = not supported				

Services

The EtherNet/IP IO Scanner Diagnostics object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported					

IO Connection Diagnostics Object

Overview

The IO Connection Diagnostics object presents the instances, attributes and services described below.

Class ID

352 (hex), 850 (decimal)

Instance IDs

The IO Connection Diagnostics object presents two instance values:

- 0 (class)
- 257 ... 400 (instance): The instance number matches the connection number in the **Connection Settings** configuration tab, page 209.

  NOTE: The Instance ID number = the Connection ID. For *M580* specifically, you can look up the **Connection ID** on the DTM **Device List** screen.

Attributes

IO Connection Diagnostics object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 to 256 (instance attributes):

Attribute ID	Description	Type	GET	SET	Value
01	IO Communication Diagnostics	STRUCT	X	—	
	IO Production Counter	UDINT			Increments at each production
	IO Consumption Counter	UDINT			Increments at each consumption
	IO Production Send Errors Counter	UINT			Increments each time a production is not sent
	IO Consumption Receive Errors Counter	UINT			Increments each time a consumption is received with a detected error
	CIP Connection Timeout Errors	UINT			Increments when a connection times out
	CIP Connection Opening Errors	UINT			Increments each time a connection is unable to open
	CIP Connection State	UINT			State of the Connection Bit
	CIP Last Error General Status	UINT			General status of the last error detected on the connection
	CIP Last Error Extended Status	UINT			Extended status of the last error detected on the connection
	Input Communication Status	UINT			Communication status of the inputs (see table, below)
	Output Communication Status	UINT			Communication status of the outputs (see table, below)
X = supported — = not supported					

Attribute ID	Description	Type	GET	SET	Value
02	Connection Diagnostics	STRUCT	X	—	
	Production Connection ID	UDINT			Connection ID for production
	Consumption Connection ID	UDINT			Connection ID for consumption
	Production RPI	UDINT			RPI for production
	Production API	UDINT			API for production
	Consumption RPI	UDINT			RPI for consumption
	Consumption API	UDINT			API for consumption
	Production Connection Parameters	UDINT			Connection parameters for production
	Consumption Connection Parameters	UDINT			Connection parameters for consumption
	Local IP	UDINT			—
	Local UDP Port	UINT			—
	Remote IP	UDINT			—
	Remote UDP Port	UINT			—
	Production Multicast IP	UDINT			Multicast IP used for production (or 0)
	Consumption Multicast IP	UDINT			Multicast IP used for consumption (or 0)
	Protocols Supported	UINT			Protocol supported on the connection: 1 = EtherNet/IP
X = supported — = not supported					

These values describe the structure of the instance attributes: *CIP Connection State*, *Input Communication Status*, and *Output Communication Status*:

Bit Number	Description	Values
15...3	*Reserved*	0
2	Idle	0 = no idle notification 1 = idle notification
1	Consumption inhibited	0 = consumption started 1 = no consumption
0	Production inhibited	0 = production started 1 = no production

Services

The EtherNet/IP Interface Diagnostics object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns the value of all attributes.
0E	14	Get_Attribute_Single	—	X	Returns the value of the specified attribute.
4C	76	Get_and_Clear	—	X	Returns and clears the values of all instance attributes.
X = supported — = not supported					

EtherNet/IP Explicit Connection Diagnostics Object

Overview

The EtherNet/IP Explicit Connection Diagnostics object presents the instances, attributes and services described below.

Class ID

353 (hex), 851 (decimal)

Instance IDs

The EtherNet/IP Explicit Connection Diagnostics object presents two instance values:

- 0: class
- 1...*N*: instance (*N* = maximum concurrent number of explicit connections)

Attributes

EtherNet/IP Explicit Connection Diagnostics object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID hex	Description	Value	GET	SET
01	Revision	1	X	—
02	Max Instance	0...N	X	—
X = supported — = not supported				

Instance ID = 1 to *N* (instance attributes):

Attribute ID hex	Description	Type	GET	SET	Value
01	Originator connection ID	UDINT	X	—	Originator to target connection ID
02	Originator IP	UINT	X	—	
03	Originator TCP Port	UDINT	X	—	
04	Target connection ID	UDINT	X	—	Target to originator connection ID
05	Target IP	UDINT	X	—	
06	Target TCP Port	UDINT	X	—	
07	Msg Send Counter	UDINT	X	—	Incremented each time a Class 3 CIP message is sent on the connection
08	Msg Receive counter	UDINT	X	—	Increments each time a Class 3 CIP message is received on the connection
X = supported — = not supported					

Services

The EtherNet/IP Explicit Connection Diagnostics object performs these services upon the listed object type:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	This service returns the contents of all attributes of the class or of the instance of the object.
X = supported					

EtherNet/IP Explicit Connection Diagnostics List Object

Overview

The EtherNet/IP Explicit Connection Diagnostics List object presents the instances, attributes and services described below.

Class ID

354 (hex), 852 (decimal)

Instance IDs

The EtherNet/IP Explicit Connection Diagnostics List object presents two instance values:

- 0: class
- 1: instance

Attributes

EtherNet/IP Explicit Connection Diagnostics List object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Max Instance	X	—
X = supported — = not supported			

Instance ID = 1 to 2 (instance attributes):

Attribute ID	Description	Type	GET	SET	Value
01	Number of connections	UINT	X	—	Total number of opened explicit connections
02	Explicit Messaging Connections Diagnostic List	ARRAY of STRUCT	X	—	
	Originator connection ID	UDINT			O->T connection ID
	Originator IP	UINT			—
	Originator TCP port	UDINT			—
	Target connection ID	UDINT			T->O connection ID
	Target IP	UDINT			—
	Target TCP port	UDINT			—
	Msg Send counter	UDINT			Increments each time a Class 3 CIP message is sent on the connection
	Msg Receive counter	UDINT			Increments each time a Class 3 CIP message is received on the connection
X = supported — = not supported					

Services

The EtherNet/IP Explicit Connection Diagnostics object performs these services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	—	Returns the value of all attributes.
08	08	Create	X	—	—
09	09	Delete	—	X	—
4B	75	Explicit_Connections_Diagnostic_Read	—	X	—
X = supported — = not supported					

RSTP Diagnostics Object

Overview

The RSTP Diagnostics object presents the instances, attributes and services described below.

Class ID

355 (hex), 853 (decimal)

Instance IDs

The RSTP Diagnostics object presents these instance values:

- 0: class
- 1: instance

Attributes

RSTP Diagnostics object attributes are associated with each instance.

Instance ID = 0 (class attributes):

Attribute ID	Description	Type	GET	SET
01	Revision: This attribute specifies the implemented revision of the RSTP Diagnostic Object. The revision is increased by 1 with each object update.	UINT	X	—
02	Max Instance: This attribute specifies the maximum number of instances that may be created for this object on a per device basis (for example, an RSTP Bridge). There is 1 instance for each RSTP port on a device.	UINT	X	—
X = supported — = not supported				

Instance ID = 1 to *N* (instance attributes):

Attribute ID	Description	Type	GET	CLEAR	Value
01	**Switch Status**	STRUCT	X	—	—
	Protocol Specification	UINT	X	—	Refer to RFC-4188 for attribute definitions and value range. In addition, this value is defined: [4]: the protocol is IEEE 802.1D-2004 and IEEE 802.1W
	Bridge Priority	UDINT	X	—	Refer to RFC-4188 for attribute definitions and value range.
	Time Since Topology Change	UDINT	X	—	
	Topology Change Count	UDINT	X	X	Refer to RFC-4188 for attribute definitions and value range.
	Designated Root	String	X	—	Refer to RFC-4188 for attribute definitions and value range.
	Root Cost	UDINT	X	—	
	Root Port	UDINT	X	—	
	Max Age	UINT	X	—	
	Hello Time	UINT	X	—	
	Hold Time	UDINT	X	—	
	Forward Delay	UINT	X	—	
	Bridge Max Age	UINT	X	—	
	Bridge Hello Time	UINT	X	—	
	Bridge Forward Delay	UINT	X	—	

X = supported

— = not supported

Attribute ID	Description	Type	GET	CLEAR	Value
02	**Port Status**	STRUCT	X	—	—
	Port	UDINT	X	—	Refer to RFC-4188 for attribute definitions and value range.
	Priority	UDINT	X	—	
	State	UINT	X	—	
	Enable	UINT	X	—	
	Path Cost	UDINT	X	—	
	Designated Root	String	X	—	
	Designated Cost	UDINT	X	—	
	Designated Bridge	String	X	—	
	Designated Port	String	X	—	
	Forward Transitions Count	UDINT	X	X	Refer to RFC-4188 for attribute definitions and value range. Services: • Get_and_Clear: The active value of this parameter is returned with the response message. • other services: The active value of this parameter is returned but not cleared.
X = supported — = not supported					

Attribute ID	Description	Type	GET	CLEAR	Value
03	**Port Mode**	STRUCT	X	—	—
	Port Number	UINT	X	—	This attribute indicates the port number for a data query. The value range is configuration dependent. For a 4-port Ethernet device, as an instance, the valid range is 1...4.
	Admin Edge Port	UINT	X	—	This attribute indicates if this is a user-configured edge port: • 1: true • 2: false Other values are not valid.
	Oper Edge Port	UINT	X	—	This attribute indicates if this port is an active edge port: • 1: true • 2: false Other values are not valid.
	Auto Edge Port	UINT	X	—	This attribute indicates if this port is a dynamically determined edge port: • 1: true • 2: false Other values are not valid.
X = supported — = not supported					

Services

The RSTP Diagnostics object performs these services:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	This service returns: • all attributes of the class • all attributes of the instance of the object
0E	14	Get_Attribute_Single	X	X	This service returns: • the contents of a single attribute of the class • the contents of the instance of the object as specified Specify the attribute ID in the request for this service.
4C	76	Get_and_Clear	—	X	This service returns the contents of a single attribute of the instance of the object as specified. Then the relevant counter-like parameter(s) within the specified attribute are cleared. (Specify the attribute ID in the request for this service.)
X = supported — = not supported					

Service Port Control Object

Overview

The Service Port Control object is defined for port control purposes.

Class ID

400 (hex), 1024 (decimal)

Instance IDs

The Service Port Control object presents these instance Values:

- 0: class
- 1: instance

Attributes

Service Port Control object attributes are associated with each instance.

Required class attributes (instance 0):

Attribute ID	Description	Type	Get	Set
01	Revision	UINT	X	—
02	Max Instance	UINT	X	—
X = supported — = not supported				

Required instance attributes (instance 1):

Attribute ID		Description	Type	Get	Set	Value
hex	dec					
01	01	Port Control	UINT	X	—	0 (default): disabled 1: access port 2: port mirroring
02	02	Mirror	UINT	X	—	bit 0 (default): ETH 2 port bit 1: ETH 3 port bit 2: rack port bit 3: internal port
X = supported — = not supported						

NOTE:

- If the SERVICE port is not configured for port mirroring, the mirror attribute is ignored. If the value of a parameter request is outside the valid range, the service request is ignored.
- In port mirroring mode, the SERVICE port acts like a read-only port. That is, you cannot access devices (ping, connection to EcoStruxure Control Expert, etc.) through the SERVICE port.

Services

The Service Port Control object performs these services for these object types:

Service ID		Name	Class	Instance	Description
hex	dec				
01	01	Get_Attributes_All	X	X	Get all attributes in a single message.
0E	14	Get_Attribute_Single	X	X	Get a single specified attribute.
X = supported — = not supported					

Hot Standby FDR Sync Object

Overview

The Hot Standby FDR Sync object presents the instances, attributes and services described below.

Class ID

406 (hex), 1030 (decimal)

Instance IDs

The Hot Standby FDR Sync object presents two instances:

- 0: class
- 1: instance

Attributes

Hot Standby FDR Sync object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Maximum Instance	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Status	UDINT	X	—	• bit 0: 0 = service not running; 1 = service is running • bit 1:0 = service has no detected error; 1 = service has detected an error
02	02	Checksum of the parameter (.prm) files	UDINT	X	—	
X = supported — = not supported						

Services

The Hot Standby FDR Sync object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns: • all class attributes (instance = 0) • instance attributes 1 to 7 (instance = 1)
07	07	Stop	—	X	In Standby state, start the synchronization service. In Primary state, no action.
0E	14	Get_Attribute_ Single	X	X	Returns the value of the specified attribute.
4B	75	Copy_Primary_ to_Standby	X	X	Applicable only if the device is in Standby state. Otherwise, an error is detected.
4C	76	Copy_Standby_ to_Primary	X	X	Applicable only if the device is in Standby state. Otherwise, an error is detected.
4D	77	Clear_Files_in_ Primary	X	X	Applicable only if the device is in Primary state. Otherwise, an error is detected.
X = supported — = not supported					

Ethernet Backplane Diagnostics Object

Overview

The Ethernet Backplane Diagnostics object presents the instances, attributes and services described below.

Class ID

407 (hex), 1031 (decimal)

Instance IDs

The Ethernet Backplane Diagnostics object presents two instances:

- 0: class
- 1: instance

Attributes

Ethernet Backplane Diagnostics object attributes are associated with each instance, as follows:

Instance ID = 0 (class attributes):

Attribute ID	Description	GET	SET
01	Revision	X	—
02	Maximum Instance	X	—
03	Number of Instances	X	—
X = supported — = not supported			

Instance ID = 1 (instance attributes):

Attribute ID		Description	Type	GET	SET	Value
hex	dec					
01	01	Backplane Ethernet Port Status	UINT	X	—	Link status/health of each module on the backplane: • bit 0-14: 0 = link is up, 1 = link is down • bit 15: 0 = backplane is in normal operating state • bit 15: 1= backplane is not in normal operating state
02	02	Extended Health of Ethernet Backplane	UINT	X	—	For all bits, below, 0 = no error detected, 1 = error detected: • Bit 0: SMI error detected • Bit 1: HUBIX error detected • Bit 2: Undervoltage detected • Bit 3: Overvoltage detected • Bit 4: Backplane head did not respond • Bit 14: Backplane firmware is not compatible • Bit 15: Backplane did not respond • Other bits: reserved
X = supported — = not supported						

Services

The Ethernet Backplane Diagnostics object performs the following services upon the listed object types:

Service ID		Description	Class	Instance	Notes
hex	dec				
01	01	Get_Attributes_All	X	X	Returns: • all class attributes (instance = 0) • instance attributes 1 to 7 (instance = 1)
0E	14	Get_Attribute_Single	X	X	Returns the value of the specified attribute.
X = supported — = not supported					

Firmware Update

Introduction

The following information describes the steps for updating the firmware for the BMENOC0302(H) Ethernet communications module.

Firmware Upgrade

Tools

Upgrade the firmware for the BMENOC0302(H) module with the EcoStruxure Automation Device Maintenance (EADM) software tool. Use this tool to execute these functions:

- Automatically or manually discover one or more BMENOC0302(H) modules in your project, based on IP addresses.
- Upgrade the latest firmware version that is applicable to those modules at se.com.

  NOTE:

  - To download and use this tool, refer to the *EcoStruxure™ Automation Device Maintenance, User Guide*.
  - You cannot use the Unity Loader™ software tool to upgrade the firmware for the BMENOC0302(H) module.

Both the update and downgrade procedures are maintenance operations that require changes to the module firmware. In this case, stop the module and disconnect it from the systems and applications that it affects.

NOTICE

INOPERABLE EQUIPMENT

Stop the module before a firmware update or firmware evaluation with the EcoStruxure Automation Device Maintenance software.

Failure to follow these instructions can result in equipment damage.

Hot Standby Considerations

Refer to the instructions for upgrading the firmware in the communication modules that are configured on the local rack (including BMENO•03•• modules) in a Hot Standby system.

Modicon M580 BMENOC0302(H) Website

Introduction

As a Modicon M580 device, the BMENOC0302(H) Ethernet communications module supports a standard set of web pages. These pages provide tools to diagnose the basic functionality of the modules. The web site is not customizable.

NOTE: To use the website to configure secure IPsec communications, refer to the description of the IPsec web interface, page 131.

Introduction to the Embedded Website

Introduction

Use this website to perform diagnostics for the BMENOC0302(H) Ethernet communications module to view real-time diagnostic data for both the communications module and other networked devices.

These are the main features of the secure website:

- secure HTTPS connectivity
- authentication and authorization for users

  NOTE: In some cases, a single user can fulfill multiple roles simultaneously.
- diagnostic pages

NOTE: To use the website to configure secure IPsec communications, refer to the description of the IPsec web interface, page 131.

Open the Modicon M580 BMENOC0302(H) Website

Step	Action
1	Open an EcoStruxure Control Expert project that includes a BMENOC0302(H) module and unlock security for the module, page 140.
2	Enter the IP address (https://*xxx.xxx.xxx.xxx*) of the BMENOC0302(H) module in the address bar of your internet browser. **NOTE:** Refer to the list of appropriate browser versions below.
3	Click **Enter**, and follow the Windows prompts to continue.
4	Select one of these tabs: • Home, page 365 (**Status Summary** page) • Diagnostics, page 366

Alternately, you can access the website from within the EcoStruxure Control Expert project:

Step	Action
1	In EcoStruxure Control Expert, switch to online mode (**PLC > Connect**).
2	Expand (**+**) the **Project Browser** to see the **PLC bus** in the **Structural view** (**Project Browser > Project > Configuration > PLC bus**).
3	Select the **Web : Main IP** tab.
4	Select (click) the **Web browser** button to launch the website for the module.

Software Requirements

The embedded web server displays data in standard HTML website.

Access the embedded website on a computer with any of these browsers:

Browser	Required Version
Google Chrome	v88 or a subsequent supporting version
Mozilla Firefox	v78 or a subsequent supporting version
Microsoft Edge	v89 or a subsequent supporting version

Login Page

Menu Items

Log in to the website for the M580 Ethernet communications module:

Step	Action
1	Select your sign-in criteria: • **Language:** Select your preferred language from this pull-down menu. • **User Name:** Enter your user name. • **Password:** Enter the password you configured in the **Project & Controller Protection** tab.
2	Click the **Login** button.

Status Summary (Home Page)

Access the Page

Access the **Status Summary** page on the **Diagnostics** tab (**Module > Summary**).

Diagnostic Information

This table describes the components of the **Status Summary** page (**Home** tab):

Parameters	**Description**	
LED display	This graphical representation of the LED display for module shows the state of the individual LED indicators. **NOTE:** The diagnostics information is explained in the description of LED activity and indications, page 253.	
SERVICE STATUS	green	The available service is operational and running.
	red	An error is detected in an available service.
	black	The available service is not present or not configured.
NETWORK INFORMATION	This list contains the network and hardware address information and connectivity that corresponds to the module.	
CPU SUMMARY	This list describes the controller module hardware and the applications that run on it.	
VERSION INFORMATION	This list describes the software versions that are running on the Ethernet communications module.	

Diagnostics Menu

Introduction

Use the website to perform diagnostics for the M580 Ethernet communications module to display real-time diagnostic data for the module itself and other networked devices.

Menu Items

Access these tabs from the **Diagnostics** menu when you open the module's website, page 363:

Menu Items		Description
Module	*Status Summary*, page 365	The objects on this home page provide information about the module status.
	Performance, page 367	View performance statistics for the module.
	Port Statistics, page 368	View statistical information for each port on the module.
Connected Devices	*I/O Scanner*, page 370	View the scanner status and connection statistics for the communications module.
	Messaging, page 371	View information for open Modbus TCP connections on port 502.
Services	*QoS*, page 372	View information about the QoS service.
	NTP, page 374	View the operating parameters for the network time service.
	Redundancy, page 376	View the configured values for the RSTP configuration of the communications module.
System	*Alarm Viewer*, page 377	View the diagnostics information that corresponds to running services and communications module operations.
	Rack Viewer, page 378	Access a graphical view of the local rack that contains the module.
	Program Viewer, page 379	View and monitor the EcoStruxure Control Expert programs that are in run mode.

Performance Page

Access the Page

Access the **Performance** page from the **Diagnostics** tab (**Diagnostics > Menu > Module > Performance**).

NOTE: This page is updated every 5 seconds.

Diagnostic Information

This table describes the **Performance** statistics:

Field	**Description**
MODULE I/O UTILIZATION	This graph shows the actual number of packets per second for implicit I/O communications.
MESSAGING STATISTICS	This graph shows actual number of packets per second for Modbus TCP and EtherNet/IP explicit messages.

Port Statistics Page

Access the Page

Access the **Port Statistics** page from the **Diagnostics** tab (**Diagnostics > Menu > Connected Devices > Port Statistics**).

Diagnostic Information

This page shows the statistics for each port on the communications module. This information is associated with the configuration of the Ethernet ports and the configuration of the SERVICE port.

The frame color indicates the port activity:

- *green*: active (configured and connected)
- *gray*: disabled (not connected)
- *yellow*: not configured
- *red:* error detection

View these statistics on the **Port Statistics** page:

Statistic	Description
Speed	The configured port speed (10, 100, 1000 Mbps)
Duplex	The active duplex mode is composed of some combination of these elements: • TP/Fiber • -Full/-Half/-None • Link/(no word) **NOTE:** When the thirteenth bit of the word in the Modbus response is 1, **Link** is added to the duplex mode string (**TP-Full Link**, **TP-Half Link**, etc.).
Redundancy Status	The Ethernet port is: • learning or forwarding information • discarding information • disabled
Success Rate	Successful transmissions (percentage)
Total Errors	Number of detected errors

NOTE: This page is updated every 5 seconds.

Expanded View

Click **Toggle Detail View** to see more statistics:

Statistic	Description
Speed	Transmission speed
Duplex	Transmission speed (duplex)
Frames Transmitted	Number of frames successfully transmitted
Frames Received	Number of frames received
Bytes Transmitted	Number of bytes successfully transmitted
Bytes Received	Number of bytes received
Inbound Packet Errors	Number of detected inbound packet errors
Inbound Packets Discarded	Number of inbound packets discarded
Outbound Packet Errors	Number of detected outbound packet errors
Outbound Packets Discarded	Number of outbound packets discarded
Excessive Collisions	Number of excessive Ethernet collisions
Late Collisions	Number of late Ethernet collisions
CRC Errors	Number of detected cyclic redundancy check errors
Carrier Sense Errors	Number of detected carrier sense errors. A carrier sense error is detected when a port tries to transmit a frame, but cannot do so because no carrier is detected.
FCS Errors	Number of detected frame check sequence (FCS) errors. An FCS error is detected when a frame is corrupted during transmission as indicated by its checksum value.
Alignment Errors	The number of byte alignment errors that are detected. A byte alignment occurs when the number of bits in a frame is not divisible by 8. An alignment error also triggers an FCS error.
Internal MAC Trans. Errors	The number of detected transmit errors that are not late collisions, excessive collisions, or CRC errors.
Internal MAC Rec. Errors	The number of detected receive errors that are not late collisions, excessive collisions, or CRC errors.
SQE Test Errors	The number of detected signal quality error (SQE) instances. Some Ethernet transceivers use an SQE heartbeat to indicate it is connected to a host interface. This detected error indicates that a transceiver has no heartbeat. (Every transceiver does not necessarily produce a heartbeat.)

I/O Scanner Page

Access the Page

View the scanner status and connections statistics on the **I/O Scanner** page.

Access the page from the **Diagnostics** tab (**Diagnostics > Menu > Connected Devices > I/O Scanner**).

NOTE: This page is updated every 5 seconds.

Diagnostic Information

This table describes the scanner status and connection statistics:

SCANNER STATUS	*Running:* The I/O scanner is enabled.
	Stopped: The I/O scanner is disabled.
	Idle: The I/O scanner is enabled but not running.
	Error: The I/O scanner returns unexpected values from the device.
CONNECTION STATISTICS	*Transmissions Sent:* This is the number of transactions per second.
	Valid Connections: This is the number of valid active connections.

In the *SCANNED DEVICES STATUSES* table, the colors that appear in each block indicate these states for specific remote devices:

Color	Indication	Status
gray	*Not Configured*	A device is not configured.
black	*Unscanned*	The scanning for a specific device is disabled.
green	*Scanned*	A device is being scanned successfully.
red	*Fault*	A scanned device is returning detected errors.

NOTE: Position the cursor over a block to get information for a specific device.

Messaging Page

Access the Page

Access information about the Modbus connections to the module on the **Messaging** page

Access this page from the **Diagnostics** tab (**Diagnostics > Menu > Connected Devices > Messaging**).

NOTE: This page is updated every 5 seconds.

Diagnostic Information

This page shows information for the open Modbus TCP connections on port 502:

Field	Description
MESSAGING STASTICS	This field contains the total number of sent and received messages on port 502. These values are not reset when the port 502 connection is closed. Therefore, the values indicate the number of messages that were sent or received since the module was started.
ACTIVE CONNECTIONS	This field shows the connections that are active when the **Messaging** page is refreshed.
Success Rate	This field shows the percentage of successful requests out of the total number of requests.

QoS Page

Access the Page

Access the **QoS** (quality of service) page from the **Diagnostics** tab (**Diagnostics > Menu > Services > QoS**).

NOTE:

- Configure this service in the EcoStruxure Control Expert application.
- Click **Detail View** to expand the list of parameters.
- This page is updated every 5 seconds.

Diagnostic Information

When you enable QoS, the module adds a differentiated services code point (DSCP) tag to each Ethernet packet it transmits, thereby indicating the priority of that packet:

Field	Parameter
SERVICE STATUS	*Running:* The service is correctly configured and running.
	Disabled: The service is disabled.
	Unknown: The status of the service is undefined.
ETHERNET/IP TRAFFIC	Configure the priority levels to prioritize the management of data packets for the displayed message types.
MODBUS/TCP TRAFFIC	
NTP TRAFFIC	This is the DSCP value for network traffic.
PRECISION TIME PROTOCOL	Configure the Precision Time Protocol (PTP) values for the displayed message types. **NOTE:** The Precision Time Protocol QoS attributes are 2 and 3 (class 48h, instance 1). Use these attributes to obtain QoS values for the Precision Time Protocol.

Considerations

Take measures to effectively implement QoS settings in your Ethernet network:

- Use only network switches that support QoS.
- Apply the same DSCP values to all network devices and switches.
- Use switches that apply a consistent set of rules for handling the different DSCP values when transmitting and receiving Ethernet packets.

NTP Page

Access the Page

Access this page from the **Diagnostics** tab (**Diagnostics > Services > NTP**).

NOTE: This page is updated every 5 seconds.

The **NTP** page contains information about the NTPv4 network time service.

NTPv4 and subsequent supporting version(s) display NTPv4 for the **Client** mode.

Network Time

The Network Time Service synchronizes computer clocks over the internet for the purposes of event recording (sequence events), event synchronization (trigger simultaneous events), or alarm and I/O synchronization (time stamp alarms).

The NTP page displays information about the network time service for the **Client** mode.

NTPv4 Diagnostic Information

This table shows the combined information from the *Client* and *NTPv4 Disable Status* pages:

Field	Description	
SERVICE STATUS	*Running*	The NTP service is correctly configured and running.
	Disabled	The NTP service is disabled.
	Unknown	The NTP service status is unidentified.
MODE	*Client* mode	The controller module receives time data from a remote NTP server.
SYNC	*Client* mode	The time for the controller module is synchronized to an external NTP server.
CURRENT DATE	*System*	The local date for the controller module.
	UTC	The same date in UTC.
CURRENT TIME	*System*	The local time for the controller module.
	UTC	The same time in UTC.
TIME ZONE	—	This field shows the time zone in terms of plus or minus Universal Time, Coordinated (UTC).

Field	Description	
DST STATUS	*On*	Daylight Saving Time (DST) is configured and running.
	Off	DST is disabled.
	Unknown	The DST status is indeterminable.
SERVICE STATISTICS	*Root Delay*	This estimate represents the total amount of time (ms) of the roundtrip delay of the referenced (current) time.
	Root Dispersion	This estimate represents the total amount of variance (ms) between the server time and the *CURRENT TIME*.
	Accuracy	This time corresponds to the *time_correct_within* variable in the NTPv4 diagnostic *C* structure.

Redundancy Page

Access the Page

The **Redundancy** page shows the redundancy status for each port on the communications module. (Configure the RSTP service in EcoStruxure Control Expert.)

Access this page on the **Diagnostic** tab (**Diagnostics > Services > Redundancy**).

Diagnostic Information

This table describes the diagnostics information on the **Redundancy** page. The information shown in this table depends on which Ethernet services you enable in the **Services** configuration page, page 101:

Field	
• *RSTP STATUS*	This color code indicates the status of the RSTP service: • *green:* The service is running on the communications module. • *red:* An error is detected. • *gray:* The service is not running on the communications module.
• *LAST TOPOLOGY CHANGE*	These values represent the date and time that the last topology change was received for the corresponding *Bridge ID* (below).
• *ROUTER BRIDGE STATISTICS*	*Bridge ID:* This unique bridge identifier is the concatenation of the bridge RSTP priority and the MAC address.
	Bridge Priority: Configure the RSTP operating state of the **Bridge ID** in EcoStruxure Control Expert.
• *INTERNAL INTERFACE* • *ETH1* • *ETH2* • *ETH3* • *ETHERNET rack PORT*	This color code indicates the redundancy status for each port: • *green:* The port is learning or forwarding information. • *yellow:* The port is discarding information. • *red:* The port detects errors. • *gray:* RSTP is disabled for the port.

Alarm Viewer Page

Access the Page

View the alarm status that corresponds to the EcoStruxure Control Expert application in the **Alarm Viewer** page. Access this page from the **Diagnostics** tab (**Diagnostics > Menu > System > Alarm Viewer**).

NOTE: This page is updated every 5 seconds.

Diagnostic Information

The **Alarm Viewer** page reports detected application errors. You can read and sort information about alarm objects on this page.

Each alarm has a timestamp, a description, and an acknowledgement status:

- *red:* critical
- *green:* acknowledged
- *blue:* information

  NOTE: These alarms do not require acknowledgement.

This table describes the components of the table on the **Alarm Viewer** page:

Field	Description	
Type	This column describes the alarm type.	
Status	*STOP*	You are required to acknowledge a critical alarm.
	ACK	You acknowledged an alarm.
	OK	An alarm does not require acknowledgment.
Message	This column contains the text of the alarm message.	
Occurrence	This column contains the date and time at which the alarm occurred.	
Acknowledged	This column reports the acknowledged status of the alarm.	
Zone	This column contains the area or geographical zone from which the alarm comes (0: common area).	

Rack Viewer Page

Access the Page

View a graphical representation of the local rack on the **Rack Viewer** page.

Access this page from the **Diagnostics** tab (**Diagnostics > Menu > System > Rack Viewer**).

Diagnostic Information

This page shows a topological view of the modules that are connected to the controller module, as configured in the EcoStruxure Control Expert application. Most modules have a diagnostic window that is displayed when you double-click the respective communications module.

Program Viewer Page

Access the Page

Open the **Program Viewer** to view and monitor the EcoStruxure Control Expert programs that are in run mode:

Step	Action
1	Select the **Monitoring** tab.
2	Expand the **Program Viewer** (**Menu > Program Viewer**).
3	Click **Open Program Viewer** to view the controller module program sections (on the left) and the selected section (on the right).

Controller Programming Languages

EcoStruxure Control Expert supports these programming languages:

- Ladder (LD)
- Instruction List (IL)
- Function Block Diagram (FBD)
- Structured Text (ST)
- Sequential Function Chart (SFC)
- Function block diagram LL984

Click the controller module program section in the navigation tree to view the selected program section.

Variable Animation

The status of boolean variables is indicated by the color:

Color	Action
green	The value is TRUE.
red	The value is FALSE.
yellow	The value is of a type that is neither TRUE nor FALSE. (Use the Tool Tip below to find information about the variable name, type, address, and comment.)

Links Animation

The links to boolean variables are displayed in different colors depending on the value of the variable to which they are connected:

Color	Value
green	The value is TRUE.
red	The value is FALSE.
yellow	This is the value of other links.

Tool Tip

The Tool Tip help bubble appears when the cursor hovers over a variable. The bubble displays this information:

- The value of the variable if only its name is visible in the **Program Viewer**.
- The type, name, address, and comment if only its value is visible in the viewer.

Click the variable to display the bubble permanently. Right-click the variable to close the bubble.

The Program Viewer gets the program directly from the controller module. It can detect a program change to automatically synchronize to the controller module without user intervention or configuration. The available sections are displayed.

The **Program Viewer** displays status messages in the **Console** pane at the bottom of the page. Here are some examples:

- A generic error is detected.
- The controller module is reserved by someone else.
- The controller module must be reserved.
- The response was not built.
- The request has invalid parameters.
- An incorrect sequence exists.
- The response is too big for the available response buffer.
- The module is not configured.
- The action is not permitted on the object.
- An application/controller-module compatibility error (RELOAD) is detected.
- A general error is detected.

Values in the **Program Viewer** sections are refreshed more than once per second.

EcoStruxure Control Expert Project Settings

In the **Property** value column, select (check) the **Program Viewer Information** and **Data Dictionary** boxes in the EcoStruxure Control Expert project settings to make the **Program Viewer** available with automatic synchronization of the controller module program in the **Program Viewer** page.

Program Structure

Click the flowchart icon in the upper-right side of the **Program Viewer Information** page to expand the **Structural View** of the program in the left pane of the page. The **Structural View** is a navigation tree for the controller-module program.

Expand (**+**) **Program** in the structure.

Select a programming section in the expanded structure to see that section in the right pane of the page.

You can configure the parameters in the URL to show or hide the navigation tree (at the left of the controller module **Program Viewer**) to show or hide the console (at the bottom of the controller module **Program Viewer**) and to focus on a specific section or object in the controller module program.

Upgrade the Communications Module in an Existing Project

Introduction

You can replace an older communications module in an existing M580 project with a BMENOC0302(H) High Performance Ethernet Communication Module to leverage the improved capabilities and features of the module.

Use the M580 Application Update tool to perform this conversion by following the instructions below.

NOTE: The modules that are available for the conversion appear in the table of available *source* and *target* communication modules, page 385.

Upgrade Process

These are the process stages for the module upgrade:

Step	Action
1	Export an existing project from EcoStruxure Control Expert Classic.
2	Convert the project with the M580 Application Update tool.
3	Verify that the communication module in the converted project is at the desired version.

NOTE: These individual stages are explained in detail in the conversion instructions below.

Conversion Instructions

Export the project:

Step	Action
1	In EcoStruxure Control Expert Classic, rebuild the project (**Build > Rebuild All Project**).
2	Select the module you want to update.
3	Open the **Export** dialog box (**File > Export Project...**).
4	Enter a file name for the exported project.
5	Select the .zef format from the **Save as type** pull-down menu.
6	Drive to the desired destination folder.
7	Click the **Export** button to export and save the file to the destination folder.

Convert the project:

Step	Action
1	Launch the M580 Application Update tool (`M580ApplicationUpdate.exe`).
2	In the **Converter** pull-down menu, select one of these conversion instructions: • **Update BME NOC DIO 3x1.y {x=0,1; y=2,3,4} to BME NOC DIO 0302** • **Update BME NOC DIO 3x1.y {x=0,1; y=2,3,4} to BME NOC DIO 0302X**
3	Click the **Choose file** link.
4	Drive to the location of the exported .zef project and select the project.
5	Click the **Open** button to close the window and return to the M580 Application Update tool.
6	Click the **Next** button until you see messages that report the successful conversion. **NOTE:** The name of the converted file in the text field is the same as the name of the exported file with the addition of the `_converted` suffix.

NOTE: The conversion is successful only when the controller module in the project supports the BMENOC0302(H) module. If it does not, replace the controller module.

Verify the update:

Step	Action
1	Click the **Open directory** link
2	Find the converted .zef file in the same location as the exported .zef file.
3	Double-click the converted file to open the project in EcoStruxure Control Expert Classic.
4	Open the **PLC bus**.
5	Double-click the module to verify that the desired version of the module is present in the configuration.
6	Rebuild and save the project.

Available Conversions

Use the M580 Application Update tool to convert the configuration for a BMENOC0301 or BMENOC0311 module to a BMENOC0302(H) module.

Appendices

What’s in This Part

Example: Configuring DHCP Server to Provide IP Addresses to Devices in Local and Remote Subnets

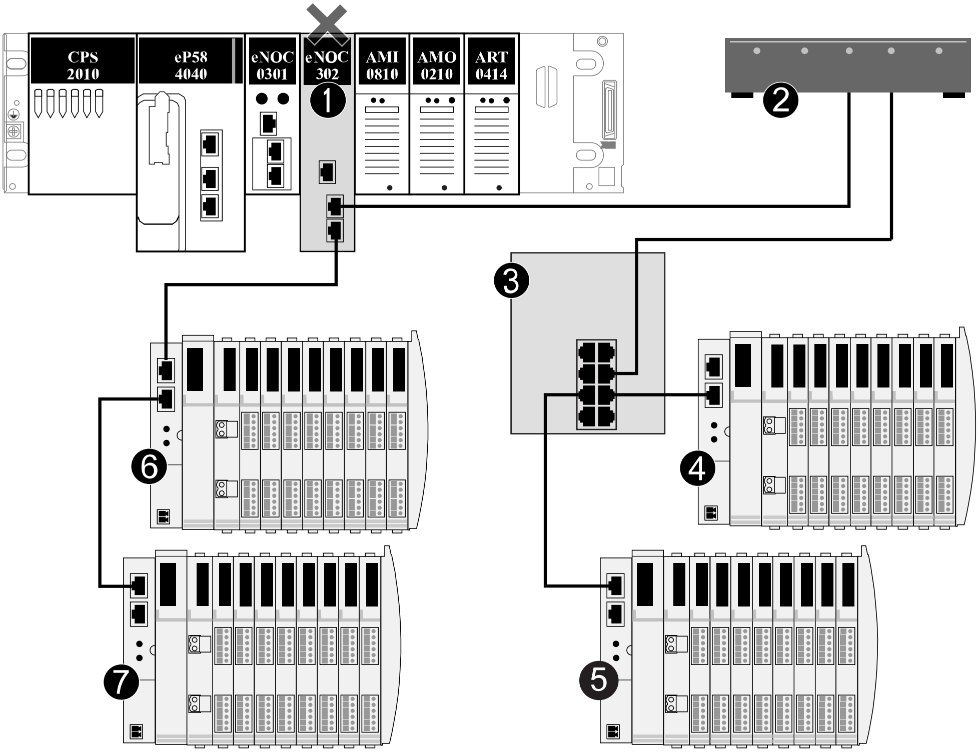
Introduction

This topic illustrates an example of a BMENOC0302(H) Ethernet communication module that provides IP addresses through BOOTP and DHCP to clients in local and remote subnets.

MSP30 Topology Example

MSP30-08040SCZ9MRHHE2A details:

- The remote client subnet is 192.168.20.0/24.
- The local BMENOC0302(H) subnet is 192.168.30.0/24.
- The router interface on the remote subnet is 192.168.20.240.
- The router interface on the local subnet is 192.168.30.240.
- The remote DHCP clients are connected to the remote subnet through the managed switch (TCSESM163F2CU0).
- The local DHCP clients are connected to the BMENOC0302(H) module.
- The router is MSP30-08040SCZ9MRHHE2A.

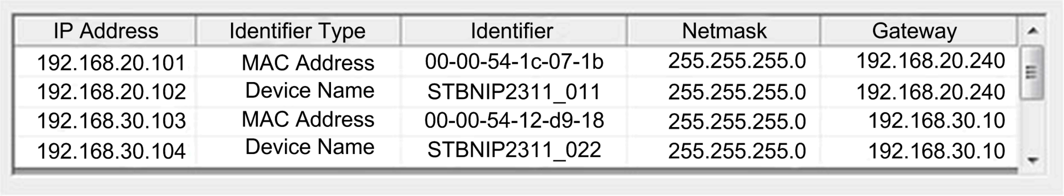


Legend:

Item	Module/Device
1	DHCP server (BMENOC0302(H) module on local rack @ 192.168.30.10)
2	router @ 192.168.30.240 on the DHCP server side and 192.168.20.240 to the managed switch
3	managed switch
4	remote BOOTP client (STBNIP2311 module @ 00:00:54:1c:07:1b)
5	remote DHCP client (STBNIP2311_011)
6	local BOOTP client (STBNIP2311 module @ 00:00:54:12:d9:18)
7	local DHCP client (STBNIP2311_022)

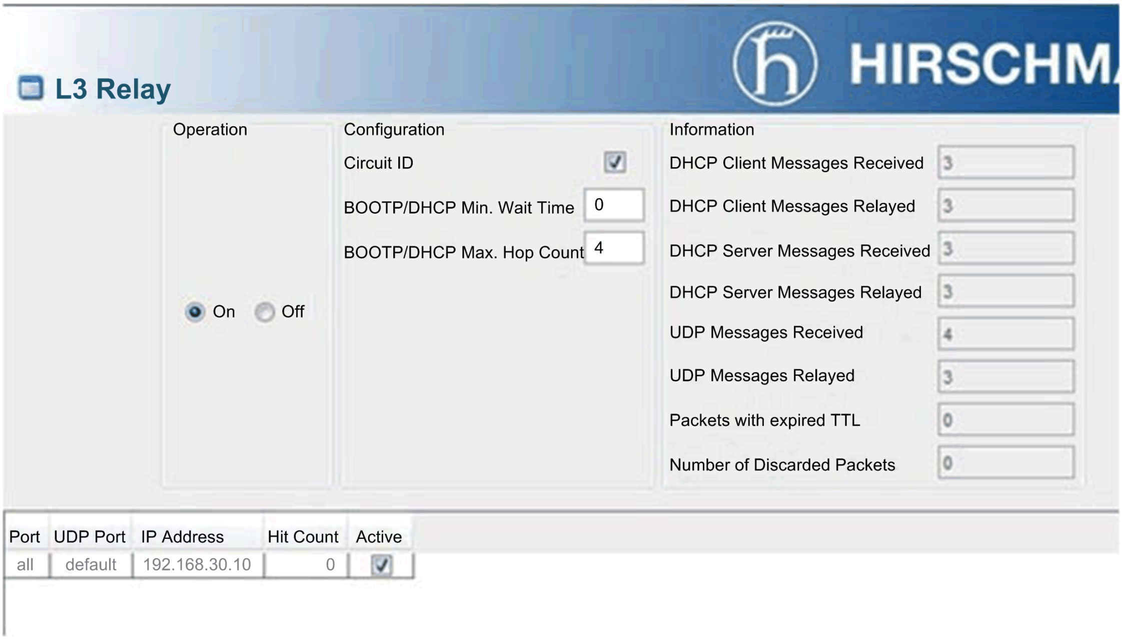
NOTE: In this case, the rack port for the BMENOC0302(H) module is disabled, as indicated by the *X*.

BMENOC0302(H) DHCP server configuration:



IP Address	Identifier Type	Identifier	Netmask	Gateway
192.168.20.101	MAC Address	00-00-54-1c-07-1b	255.255.255.0	192.168.20.240
192.168.20.102	Device Name	STBNIP2311_011	255.255.255.0	192.168.20.240
192.168.30.103	MAC Address	00-00-54-12-d9-18	255.255.255.0	192.168.30.10
192.168.30.104	Device Name	STBNIP2311_022	255.255.255.0	192.168.30.10

MSP30 DHCP relay agent configuration:



L3 Relay

Operation

● On ○ Off

Configuration

Circuit ID ☑

BOOTP/DHCP Min. Wait Time 0

BOOTP/DHCP Max. Hop Count 4

Information

DHCP Client Messages Received	3
DHCP Client Messages Relayed	3
DHCP Server Messages Received	3
DHCP Server Messages Relayed	3
UDP Messages Received	4
UDP Messages Relayed	3
Packets with expired TTL	0
Number of Discarded Packets	0

Port	UDP Port	IP Address	Hit Count	Active
all	default	192.168.30.10	0	☑

Considerations

- Verify that the device names are unique for each DHCP server.
- Verify that DHCP/BOOTP clients that use the same DHCP relay agent also use the same DHCP server.

Detected Error Codes

What's in This Chapter

Overview

The following information contains a list of codes that describe the status of Ethernet communication module messages.

EtherNet/IP Implicit or Explicit Messaging Detected Error Codes

Introduction

If a DATA_EXCH function block does not execute an EtherNet/IP explicit message, EcoStruxure Control Expert returns a hexadecimal detected error code. The code can describe an EtherNet/IP detected error.

EtherNet/IP Detected Error Codes

EtherNet/IP hexadecimal detected error codes include:

Detected Error Code	Description
800D hex	Timeout on the explicit message request
8012 hex	Incorrect device
8015 hex	Either: • Nor resources to handle the message, or • Internal detected error: no buffer available, no link available, impossible to send to the TCP task

Detected Error Code	Description
8018 hex	Either: • Another explicit message for this device is in progress, or • TCP connection or encapsulation session in progress
8030 hex	Timeout on the Forward_Open request
NOTE: The following 81•• hex detected errors are Forward_Open response detected errors that originate at the remote target and are received through the CIP connection.	
8100 hex	Connection in use or duplicate Forward_Open
8103 hex	Transport class and trigger combination not supported
8106 hex	Ownership conflict
8107 hex	Target connection not found
8108 hex	Invalid network connection parameter
8109 hex	Invalid connection size
8110 hex	Target for connection not configured
8111 hex	Requested packet interval (RPI) not supported
8113 hex	Out of connections
8114 hex	Vendor ID or product code mismatch
8115 hex	Product type mismatch
8116 hex	Revision mismatch
8117 hex	Invalid produced or consumed application path
8118 hex	Invalid or inconsistent configuration application path
8119 hex	Non-Listen Only connection not opened
811A hex	Target object out of connections
811B hex	Requested packet interval (RPI) is smaller than the production inhibit time
8123 hex	Connection timed out
8124 hex	Unconnected request timed out
8125 hex	Parameter detected error in unconnected request and service
8126 hex	Message too large for unconnected_send service
8127 hex	Unconnected acknowledge without reply
8131 hex	No buffer memory available
8132 hex	Network bandwidth not available for data

Detected Error Code	Description
8133 hex	No consumed connection ID filter available
8134 hex	Not configured to send scheduled priority data
8135 hex	Schedule signature mismatch
8136 hex	Schedule signature validation not possible
8141 hex	Port not available
8142 hex	Link address not valid
8145 hex	Invalid segment in connection path
8146 hex	Detected error in Forward_Close service connection path
8147 hex	Scheduling not specified
8148 hex	Link address to self invalid
8149 hex	Secondary resources unavailable
814A hex	Rack connection is established
814B hex	Module connection is established
814C hex	Miscellaneous
814D hex	Redundant connection mismatch
814E hex	No more user-configurable link consumer resources: the configured number of resources for a producing application has reached the limit
814F hex	No more user-configurable link consumer resources: there are no consumers configured for a producing application to use
8160 hex	Vendor specific
8170 hex	No target application data available
8171 hex	No originator application data available
8173 hex	Not configured for off-subnet multicast
81A0 hex	Detected error in data assignment
81B0 hex	Optional object state detected error
81C0 hex	Optional device state detected error
NOTE: Every detected 82•• hex error is a register session response detected error.	
8200	Target device does not have sufficient resources
8208	Target device does not recognize message encapsulation header
820F	Reserved or unidentified detected error from target

Explicit Messaging: Communication and Operation Reports

Overview

Communication and operation reports are part of the management parameters.

NOTE: Test the communication function reports at the end of their execution and before the next activation. On cold start-up, verify that the communication function management parameters are checked and reset to 0.

You can use the system bit to examine the first cycle after a cold or warm start. Refer to the *EcoStruxure™ Control Expert, System Bits and Words, Reference Manual*.

Communication Report

This report is common to every explicit messaging function. It is significant when the value of the activity bit switches from 1 to 0. The report in the following table concern errors detected by the controller that executed the function:

Value	Communication report (least significant byte)
00 hex	Correct exchange
01 hex	Exchange stop on timeout
02 hex	Exchange stop on user request (`CANCEL`)
03 hex	Incorrect address format
04 hex	Incorrect destination address
05 hex	Incorrect management parameter format
06 hex	Incorrect specific parameters
07 hex	Error detected in sending to the destination
08 hex	Reserved
09 hex	Insufficient receive buffer size
0A hex	Insufficient send buffer size
0B hex	No system resources: the number of simultaneous communication EFs exceeds the maximum that the controller can manage
0C hex	Incorrect exchange number
0D hex	No telegram received

Value	Communication report (least significant byte)
0E hex	Incorrect length
0F hex	Telegram service not configured
10 hex	Network module missing
11 hex	Request missing
12 hex	Application server is active
13 hex	UNI-TE V2 transaction number incorrect
FF hex	Message refused

NOTE: The function can detect a parameter error before activating the exchange. In this case the activity bit remains at 0, and the report is initialized with values corresponding to the detected error.

Operation Report

This report byte is specific to each function, and specifies the result of the operation on the remote application:

Value	Operation report (most significant byte)
05 hex	Length mismatch (CIP)
07 hex	Incorrect IP address
08 hex	Application error detection
09 hex	Network is not operational
0A hex	Connection reset by peer
0C hex	Communication function not active
0D hex	• Modbus TCP: transaction timed out • EtherNet/IP: request timeout
0F hex	No route to remote host
13 hex	Connection refused
15 hex	• Modbus TCP: no resources • EtherNet/IP: no resources to handle the message; or an internal detected error; or no buffer available; or no link available; or cannot send message
16 hex	Remote address not allowed

Value	Operation report (most significant byte)
18 hex	• Modbus TCP: concurrent connections or transactions limit reached • EtherNet/IP: TCP connection or encapsulation session in progress
19 hex	Connection timed out
22 hex	Modbus TCP: invalid response
23 hex	Modbus TCP: invalid device ID response
30 hex	• Modbus TCP: remote host is down • EtherNet/IP: connection open timed out
80 hex...87 hex: Forward_Open response detected errors:	
80 hex	Internal detected error
81 hex	Configuration detected error: Adjust the length of the explicit message or the requested packet interval (RPI) rate
82 hex	Device detected error: target device does not support this service
83 hex	Device resource detected error: no resource is available to open the connection
84 hex	System resource event: unable to reach the device
85 hex	Data sheet detected error: incorrect EDS file
86 hex	Invalid connection size
90 hex...9F hex: Register session response detected errors:	
90 hex	Target device does not have sufficient resources
98 hex	Target device does not recognize message encapsulation header
9F hex	Unidentified detected error from target

Glossary

A

adapter:

An adapter is the target of real-time I/O data connection requests from scanners. It cannot send or receive real-time I/O data unless it is configured to do so by a scanner, and it does not store or originate the data communications parameters necessary to establish the connection. An adapter accepts explicit message requests (connected and unconnected) from other devices.

B

BOOTP:

(*bootstrap protocol*) A UDP network protocol that can be used by a network client to automatically obtain an IP address from a server. The client identifies itself to the server using its MAC address. The server, which maintains a pre-configured table of client device MAC addresses and associated IP addresses, sends the client its defined IP address. The BOOTP service utilizes UDP ports 67 and 68.

C

CIP™:

(*common industrial protocol*) A comprehensive suite of messages and services for the collection of manufacturing automation applications (control, safety, synchronization, motion, configuration and information). CIP allows users to integrate these manufacturing applications with enterprise-level Ethernet networks and the internet. CIP is the core protocol of EtherNet/IP.

control network:

An Ethernet-based control network contains controller modules, SCADA systems, an NTP server, computers, AMS, switches, etc. Two kinds of topologies are supported:

- flat topology: The modules and devices in this network belong to same subnet.
- two-level topology: The network is split into an operation network and an inter-controller network. These two networks can be physically independent, but are generally linked by a routing device.

CPU:

(*central processing unit*) The CPU, also known as the processor or controller module, is the brain of an industrial manufacturing process. It automates a process as opposed to relay control systems. CPUs are computers suited to survive the harsh conditions of an industrial environment.

D

Device DDT (DDDT):

A Device DDT is a DDT predefined by the manufacturer and not modifiable by user. It contains the I/O language elements of an I/O module.

device network:

An Ethernet-based network within an RIO network that contains both RIO and distributed equipment. Devices connected on this network follow specific rules to allow RIO determinism.

DHCP:

(*dynamic host configuration protocol*) An extension of the BOOTP communications protocol that provides for the automatic assignment of IP addressing settings, including IP address, subnet mask, gateway IP address, and DNS server names. DHCP does not require the maintenance of a table identifying each network device. The client identifies itself to the DHCP server using either its MAC address, or a uniquely assigned device identifier. The DHCP service utilizes UDP ports 67 and 68.

DIO network:

A DIO network contains distributed equipment for which I/O scanning is performed by a controller module that runs the DIO scanner service on the local rack. DIO network traffic is delivered after RIO traffic, which takes priority in an RIO network.

DIO:

(*distributed I/O*) Also known as distributed equipment. DRSs use DIO ports to connect distributed equipment.

distributed equipment:

Ethernet devices (Schneider Electric devices, computers, servers, or third-party devices) that support exchanges with a controller module or other Ethernet I/O scanner service are described as *distributed equipment*.

DNS:

(*domain name server/service*) A service that translates an alpha-numeric domain name into an IP address, the unique identifier of a device on the network.

DRS:

(*dual-ring switch*) A ConneXium extended managed switch that has been configured to operate on an Ethernet network. Predefined configuration files are provided by Schneider Electric to download to a DRS to support the special features of the main ring / sub-ring architecture.

DSCP:

(*differentiated service code points*) This 6-bit field is in the header of an IP packet to classify and prioritize traffic.

DTM:

(*device type manager*) A DTM is a device driver running on the host computer. It provides a unified structure for accessing device parameters, configuring and operating the devices, and troubleshooting devices. DTMs can range from a simple graphical user interface (GUI) for setting device parameters to a highly sophisticated application capable of performing complex real-time calculations for diagnosis and maintenance purposes. In the context of a DTM, a device can be a communications module or a remote device on the network.

See FDT.

E

EDS:

(*electronic data sheet*) EDS are simple text files that describe the configuration capabilities of a device. EDS files are generated and maintained by the manufacturer of the device.

EF:

(*elementary function*) This block is used in a program to perform a predefined logical function.

A function does not have information about the internal state. Multiple calls to the same function using the same input parameters return the same output values. You will find information on the graphic form of the function call in the [*functional block (instance)*]. Unlike a call to a function block, function calls include only an output that is not named and whose name is identical to that of the function. In FBD, each call is indicated by a unique [number] from the graphic block. This number is managed automatically and cannot be modified.

Position and configure these functions in your program to execute your application.

You can also develop other functions using the SDKC development kit.

EtherNet/IP™:

A network communication protocol for industrial automation applications that combines the standard internet transmission protocols of TCP/IP and UDP with the application layer common industrial protocol (CIP) to support both high speed data exchange and industrial control. EtherNet/IP employs electronic data sheets (EDS) to classify each network device and its functionality.

Ethernet:

A 10 Mb/s, 100 Mb/s, or 1 Gb/s, CSMA/CD, frame-based LAN that can run over copper twisted pair or fiber optic cable, or wireless. The IEEE standard 802.3 defines the rules for configuring a wired Ethernet network; the IEEE standard 802.11 defines the rules for configuring a wireless Ethernet network. Common forms include 10BASE-T, 100BASE-TX, and 1000BASE-T, which can utilize category 5e copper twisted pair cables and RJ45 modular connectors.

explicit messaging:

TCP/IP-based messaging for Modbus TCP and EtherNet/IP. It is used for point-to-point, client/server messages that include both data, typically unscheduled information between a client and a server, and routing information. In EtherNet/IP, explicit messaging is considered class 3 type messaging, and can be connection-based or connectionless.

F

FAST:

A FAST task is an optional, periodic processor task that identifies high priority, multiple scan requests, which is run through its programming software. A FAST task can schedule selected I/O modules to have their logic solved more than once per scan. The FAST task has two sections:

- IN: Inputs are copied to the IN section before execution of the FAST task.
- OUT: Outputs are copied to the OUT section after execution of the FAST task.

FDR:

(*fast device replacement*) A service that uses configuration software to replace an inoperable product.

FDT:

(*field device tool*) The technology that harmonizes communication between field devices and the system host.

FTP:

(*file transfer protocol*) A protocol that copies a file from one host to another over a TCP/IP-based network, such as the internet. FTP uses a client-server architecture as well as separate control and data connections between the client and server.

G

gateway:

A device that connects networks with dissimilar network architectures and which operates at the Application Layer of the OSI model. This term may refer to a router.

H

HMI:

(*human machine interface*) System that allows interaction between a human and a machine.

HTTP:

(*hypertext transfer protocol*) A networking protocol for distributed and collaborative information systems. HTTP is the basis of data communication for the web.

I

implicit messaging:

UDP/IP-based class 1 connected messaging for EtherNet/IP. Implicit messaging maintains an open connection for the scheduled transfer of control data between a producer and consumer. Because an open connection is maintained, each message contains primarily data, without the overhead of object information, plus a connection identifier.

I/O scanner:

An Ethernet service that continuously polls I/O modules to collect data, status, event, and diagnostics information. This process monitors inputs and controls outputs. This service supports both RIO and DIO logic scanning.

IP address:

Internet protocol address. This 32-bit address is assigned to hosts that use TCP/IP.

IPsec:

(*internet protocol security*) An open set of protocol standards that make IP communication sessions private and secure for traffic between modules using IPsec, developed by the internet engineering task force (IETF). The IPsec authentication and encryption algorithms require user-defined cryptographic keys that process each communications packet in an IPsec session.

isolated DIO network:

An Ethernet-based network containing distributed equipment that does not participate in an RIO network.

L

local rack:

An M580 local rack contains a controller module and a power supply. A local rack can include one or two racks from the same module family, a main rack and an [optional] extended rack.

local server:

The local rack functionality offered by Schneider Electric’s EtherNet/IP communication modules allows a scanner to assume the role of adapter. The local server enables the module to publish data through implicit messaging connections. Local servers are typically used in peer-to-peer exchanges between controller modules.

M

MAC address:

media access control address. A 48-bit number, unique on a network, that is programmed into each network card or device when it is manufactured.

MAST:

A client (MAST) task is a deterministic processor task that is run through its programming software. The MAST task schedules the RIO module logic to be solved in every I/O scan. The MAST task has two sections:

- IN: Inputs are copied to the IN section before execution of the MAST task.
- OUT: Outputs are copied to the OUT section after execution of the MAST task.

MB/TCP:

(*Modbus over TCP protocol*) This is a Modbus variant used for communications over TCP/IP networks.

Modbus:

Modbus is an application layer messaging protocol. Modbus provides client and server communications between devices connected on different types of buses or networks. Modbus offers many services specified by function codes.

N

NIM:

(*network interface module*) A NIM resides in the first position on an STB island (leftmost on the physical setup). The NIM provides the interface between the I/O modules and the fieldbus client. It is the only module on the island that is fieldbus dependent. That is a different NIM is available for each fieldbus.

NTP:

(*network time protocol*) Protocol for synchronizing computer system clocks. The protocol uses a jitter buffer to resist the effects of variable latency.

P

PAC:

programmable automation controller. The PAC is the brain of an industrial manufacturing process. It automates a process as opposed to relay control systems. PACs are computers suited to survive the harsh conditions of an industrial environment.

port 502:

Port 502 of the TCP/IP stack is the usual port that is usually reserved for Modbus TCP communications.

R

RIO network:

This Ethernet-based network contains three types of RIO devices: a local rack, an RIO drop, and a ConneXium extended dual-ring switch (DRS). Distributed equipment may also participate in an RIO network through a connection to DRSs or BMENOS0300 network option switch modules.

RPI:

(requested packet interval) The time period between cyclic data transmissions requested by the scanner. EtherNet/IP devices publish data at the rate specified by the RPI assigned to them by the scanner, and they receive message requests from the scanner at each RPI.

RSTP:

(*rapid spanning tree protocol*) Allows a network design to include spare (redundant) links to provide automatic backup paths if an active link stops working, without the need for loops or manual enabling/disabling of backup links.

S

SNMP:

(*simple network management protocol*) Protocol used in network management systems to monitor network-attached devices. The protocol is part of the internet protocol suite (IP) as defined by the internet engineering task force (IETF), which consists of network management guidelines, including an application layer protocol, a database schema, and a set of data objects.

SNTP:

(*simple network time protocol*) See NTP.

subnet:

The subnet is that portion of the network that shares a network address with the other parts of the network. A subnet may be physically or logically independent from the rest of the network. A part of an Internet address called a subnet number, which is ignored in IP routing, distinguishes the subnet.

T

TCP:

(*transmission control protocol*) A key protocol of the internet protocol suite that supports connection-oriented communications, by establishing the connection necessary to transmit an ordered sequence of data over the same communication path.

TFTP:

(*trivial file transfer protocol*) A simplified version of *file transfer protocol* (FTP), TFTP uses a client-server architecture to make connections between two devices. From a TFTP client, individual files can be uploaded to or downloaded from the server, using the user datagram protocol (UDP) for transporting data.

Index

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

+ 33 (0) 1 41 29 70 00

www.se.com

As standards, specifications, and design change from time to time, please ask for confirmation of the information given in this publication.

NNZ44174.01